Réseaux et Protocoles
NFP 104

G.Florin S.Natkin

Plan du cours

INTRODUCTION NOTION GÉNÉRALES

1. NIVEAU PHYSIQUE

- 1.1. Transmission sur un canal
- 1.2. Éléments de technologie

2. NIVEAU LIAISON

- 2.1 Liaison point à point
- 2.2 Liaison dans les réseaux locaux

3. NIVEAU RÉSEAU

- 3.1 Problèmes généraux de la couche réseau
- 3.2 Exemple de protocole : IP

4. NIVEAU TRANSPORT

- 4.1 Problèmes généraux de la couche transport
- 4.2 Exemples de protocoles : TCP/UDP

Bibliographie

Andrew Tanenbaum : 'Réseaux', Pearson Education, 4ième édition 2003. Traduction en français de Computer Networks

Claude Servin : 'Réseaux et Telecom' , Dunod, Paris 2006

James F Kurose, Keith W Ross: 'Computer Networking', Addison Wesley 2001

Laurent Toutain : 'Réseaux locaux et Internet' Hermès

Stevens W.R : 'TCP/IP Illustrated', Addison Wesley 1993.

Bouyer, G : "Réseaux synchrones étendus PDH et SDH", Hermès

Les pages WEB : Groupes de travail, constructeurs, cours universitaires, ... Et les **RFC Internet**

Premier Chapitre

Introduction

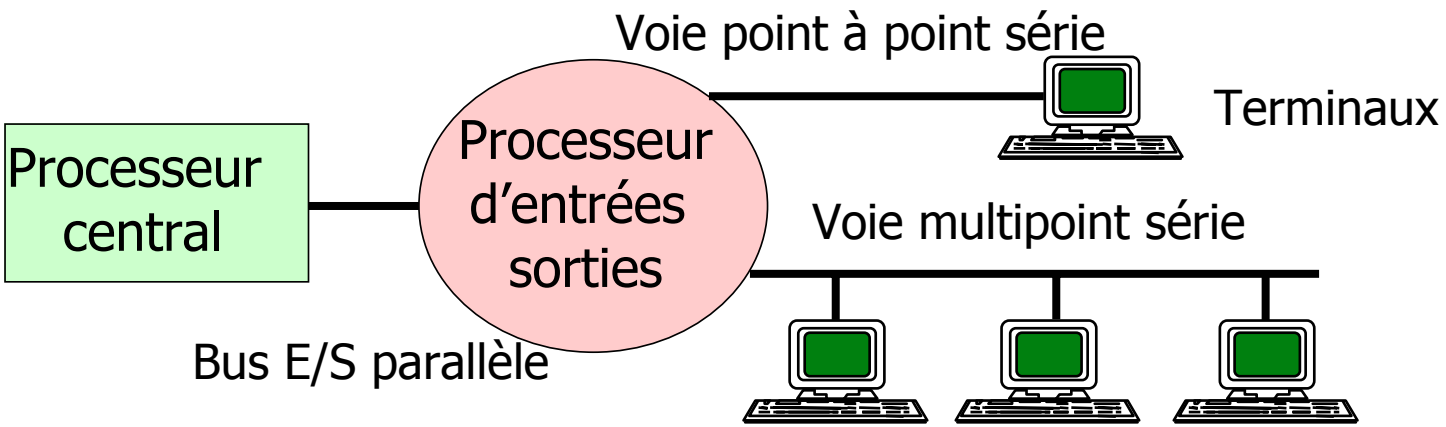
Notions générales concernant les réseaux

Introduction Notions générales

Origine des réseaux - Evolution

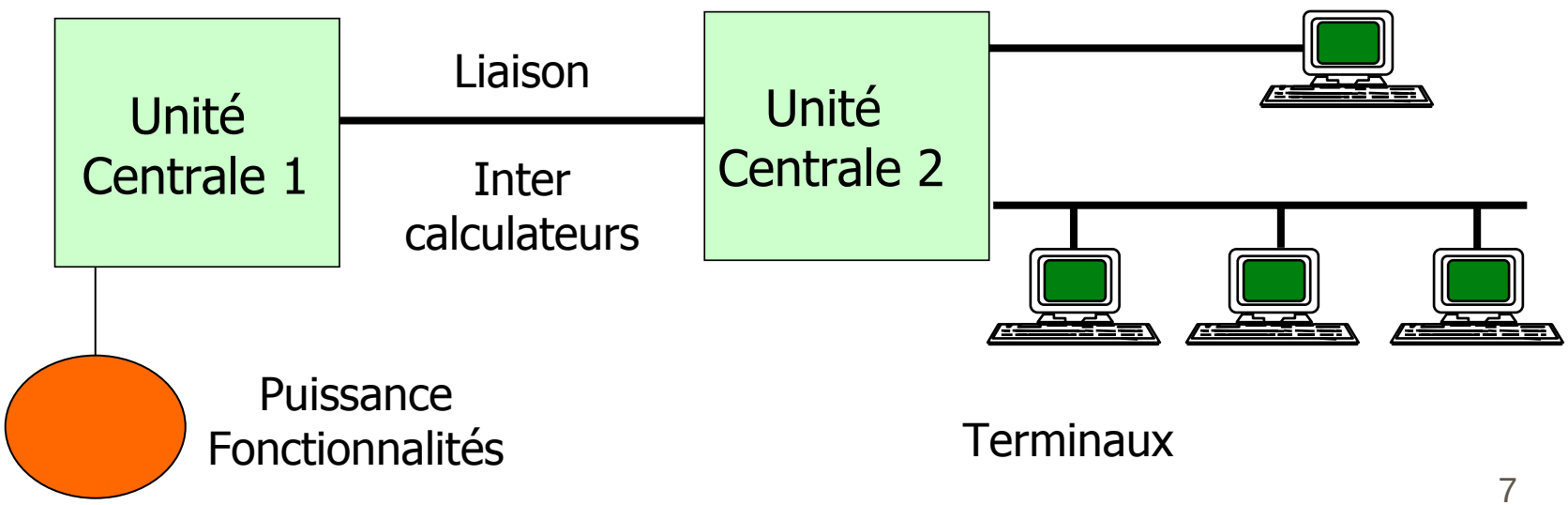
Télétraitement

- Au début de l'informatique : problème majeur
  rentabiliser des unités centrales coûteuses.
- Egalement: installer les terminaux près des utilisateurs.
- **Solution :** Mettre **en place une organisation des moyens** informatiques qui s'adapte à l'approche centralisée.
- S'affranchir des **contraintes géographiques** de localisation des systèmes informatiques.



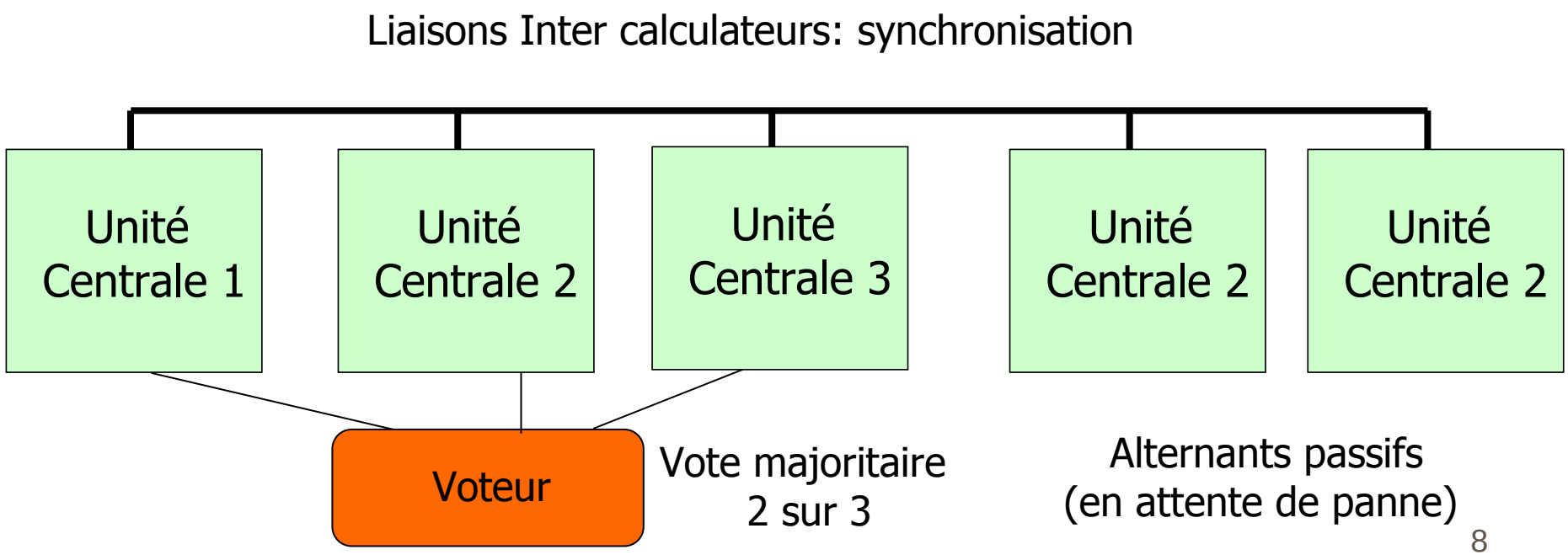
Interconnexion des processeurs: Gestion des ressources

- **A) Optimisation des ressources (partage)**
- **B) Donner accès à des ressources distantes rares/chères.**
  - Exemples 1 : puissance de calcul, logiciels spéciaux …
  - Exemples 2 : périphériques spéciaux rapides, archivage …



Interconnexion des processeurs: Sûreté de fonctionnement

- **La tolérance aux pannes**
- Permettre à des applications sensibles de continuer de fonctionner en présence de pannes.
- Exemple : Architecture Projet Apollo 1965.



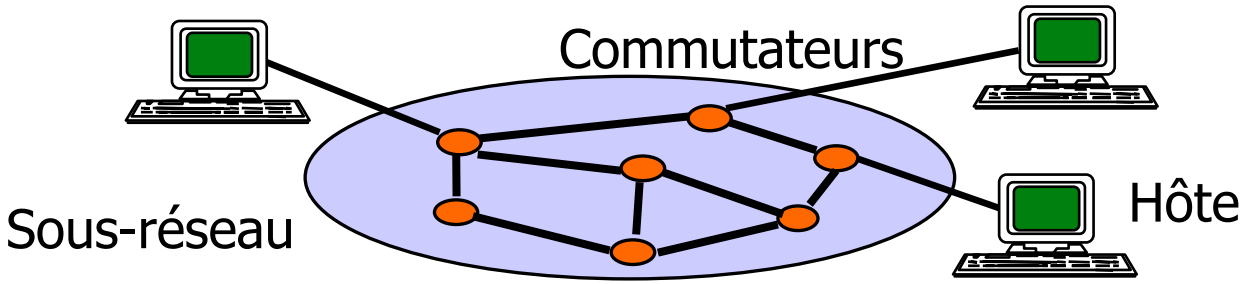
Les réseaux généraux d'ordinateurs

- **Expérimentation du réseau ARPA** ("D-ARPA Defense Advanced Research Project Agency")
  - **Ensemble de travaux** universités/industries sur contrat militaires à partir de 1967 avec des objectifs initiaux de sûreté de fonctionnement.
  - **Développement des principes essentiels** des réseaux informatiques
  - **Protocoles de communications couches basses**
    - Niveau liaison et réseau : développement de la commutation de paquets.
  - **Protocoles d'applications** (courant 1970) : Sessions à distance, Transport de fichiers, Messagerie …
  - **Succès considérable** de l'expérience ARPA => Internet.
  - **Développement de projets similaires** : Exemple cyclades.
- **Architectures constructeurs :**
  - IBM SNA ("System Network Architecture" 1974)
- **La normalisation** : Exemple X25 (1974)

Notions relatives aux réseaux généraux d'ordinateurs

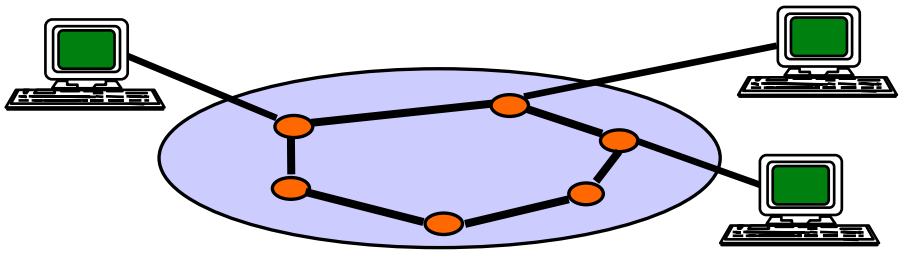
- **Réseau général** : Ensemble des systèmes informatiques **autonomes** capables **d'échanger** des informations en univers **hétérogène.**
  - **Autonomes** : Mémoire propre à chacun des sites.
    - Pas de synchronisation matérielle puissante (mémoire partagée).
  - **Echanges** : Communications en mode message asynchrone "Asynchronous message passing" (idée de faible couplage).
    - Sur des distances quelconques => Débits plus faibles (mais en accroissement constant, et en fonction des moyens financiers).
  - **Hétérogènes** : faire fonctionner ensemble des systèmes d'origines différentes.

Terminologie

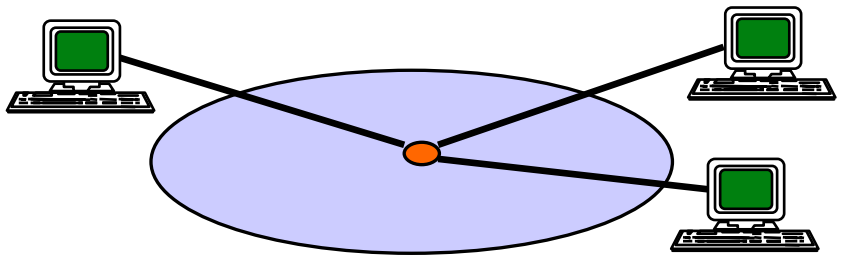


- **Ordinateurs Hôtes ("Hosts Computers")**
  - Les systèmes informatiques interconnectés.
- **Sous-réseau de communication ("Communication Subnet")** : Le moyen de communication des hôtes.
  - **Des calculateurs spécialisés (ou non) dans la commutation :** Commutateurs, noeuds de commutation, routeurs, "Packet Switches","Nodes","Routers"
  - **Des voies physiques de communication:** Liaisons spécialisées, voies, canaux, réseaux de transport d'informations.

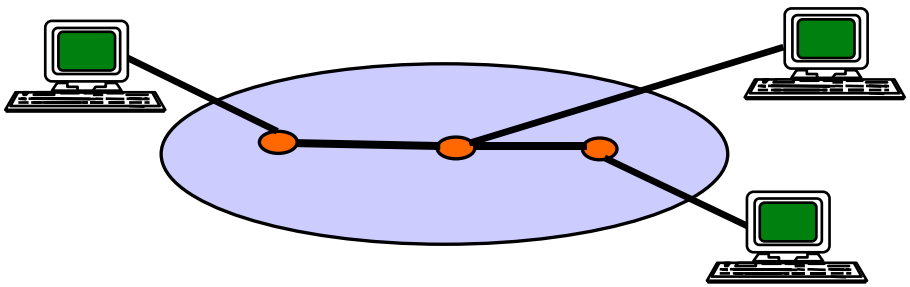
Topologies



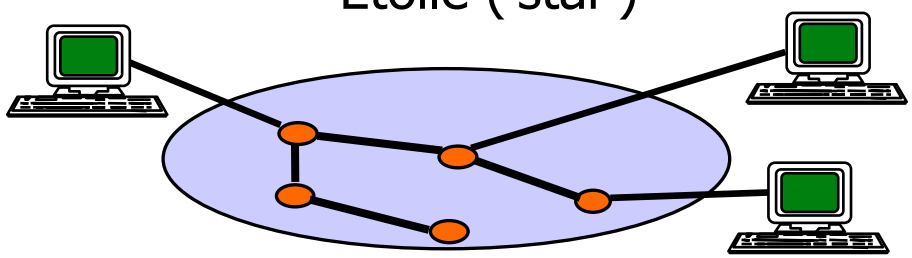
Boucle ou anneau (‘ring’)



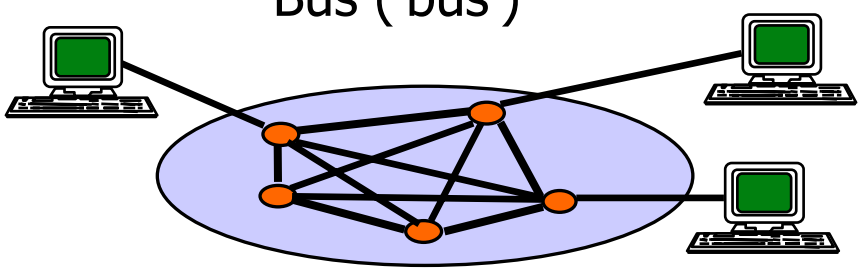
Etoile (‘star’)



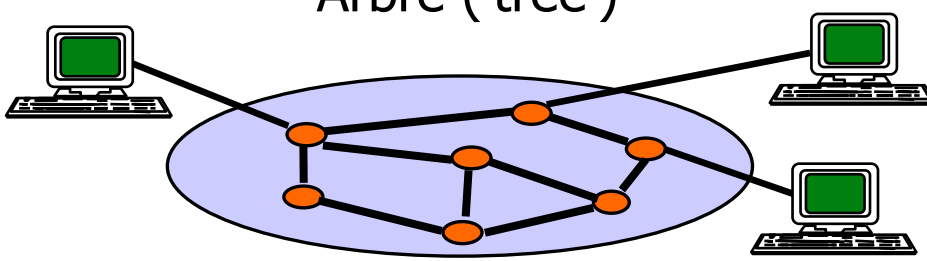
Bus (‘bus’)



Arbre (‘tree’)



Complète ou clique (‘fully connected’)



Maillée (‘meshed’)

Réseaux généraux et communications inter-personnelles

- **Convergence de deux tendances** dans les années 1970: informatique et técommunications => **la télématique** (1981)
- **Réseaux généraux** Résultat de l'expérience ARPA.
  - Les réseaux d'ordinateurs sont des supports fiables de transport de messages textuels =>Développement des messageries.
- **Techniques de commutation**
  - La construction des autocommutateurs évolue vers les techniques numériques et la commutation temporelle synchrone => Premier commutateur temporel 1975.
- **Concrétisation: le RNIS Réseau Numérique à Intégration de Service (début des années 1980)**
  - Intégration sur le même support de transmissions voix, données et images (faiblement animées)
  - Débits de base 64 kilobits/seconde, 2 mégabits/seconde

Réseaux locaux ("LAN Local Area Networks")

- **Réseaux locaux :** communications en mode message asynchrone à débit élevé sur des distances plus faibles.
- **A) Expérimentation** : d'architectures de réseaux en boucles ou sur bus à courte distance (réseaux filaires et hertziens).
- **B) Réseau Ethernet** : réseau filaire en bus (à partir de 1972).
  - **Diffusion à grande échelle** à partir de 1980 (10 Megabits/s).
  - **Améliorations constantes :** 100 Mb/s, 1Gb/s, années 2000 10 Gb/s).
- **C) Réseaux locaux radio** : depuis 1970

  Réseau Aloha => Réseaux Wifi (1997).

Systèmes répartis (‘Distributed systems, N-tiers, Peer to peer’)

- **Notion de système réparti ou distribué** (à partir de 1980)
  - Utilisation d’un réseau dans une approche d'homogénéisation.
  - Système et application commune permettant d'offrir un service réseau équivalent à celui d'un processeur unique.
- **Résolution des problèmes de désignation.**
  - Exemple : localisation des fichiers.
- **Résolution de problèmes de gestion de ressources.**
  - Exemple : sélection d'un processeur pour exécuter une tâche.
- **Résolution de problèmes de synchronisation.**
  - Exemple : contrôle d'exécution répartie en séquence, en parallèle.
- **Exemples de systèmes:** Mach, Chorus.
- **Evolution ultérieure :** les systèmes d'objets répartis, les approches à composants logiciels (Corba, EJB).

Développement des applications des réseaux et systèmes répartis

- **Communications inter personnelles**
  - Téléphonie, diffusion radio/télévision, informations (Web).
- **Algorithmique parallèle**
  - Calcul intensif, grilles de calcul (‘Grid computing’).
- **Informatique industrielle**
  - Systèmes répartis de contrôle de procédés.
- **Informatique d'entreprise**
  - Architectures client-serveur.
- **Intelligence artificielle**
  - Réseaux de neurones, systèmes multi-agents.
- **Multi média**
  - Jeux en réseaux.

Introduction Notions générales

Problèmes résolus dans les réseaux

Objectif

- **Présenter des problèmes importants** de la construction des réseaux.
- **En se limitant aux couches basses.**
- **Problèmes que l'on doit résoudre** dans un ou quelquefois dans plusieurs des niveaux d'une pile de protocoles.
- **Donner des idées générales de solutions.**
- **Des solutions précises** seront revues à propos de chaque niveau.

Z L'ordre de présentation des problèmes n'est pas significatif de leur importance.

1) Modulation, Synchronisation, Représentation des informations

- **Fonction de base des réseaux : acheminer** des suites binaires, structurées, interprétables.
- **Problèmes de synchronisation:**
  - **Synchronisation trame** : A quel moment une suite binaire significative est elle présente ?
  - **Synchronisation bit** : A quel moment un bit (un symbole) doit-il être échantillonné.
- **Problèmes de modulation**: comment sont représentés les unités d'information (bits, symboles).
- **Problèmes de codage** : comment sont codés les données (octets, caractères, numériques, …).

2) Gestion des connexions: les modes avec ou sans connexion

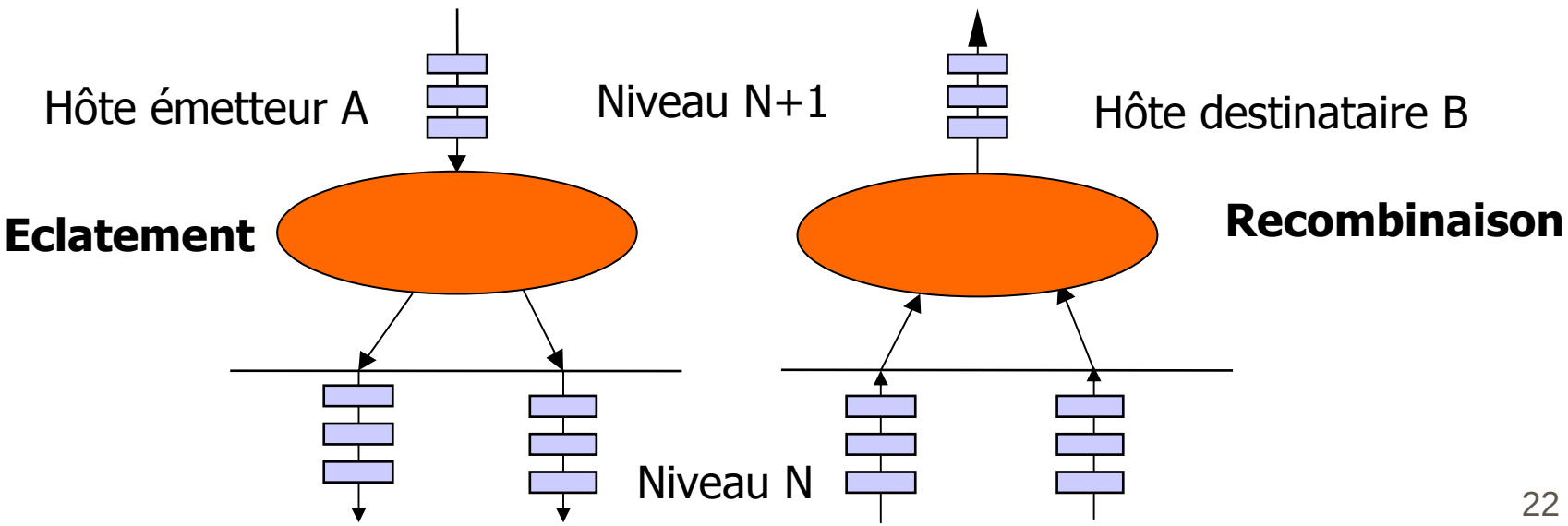
- **Mode sans connexion (non connecté)**:
  - Les échanges peuvent être réalisés à tout instant sans précautions particulières.
  - Exemples : courrier postal, protocoles IP, UDP, courrier électronique SMTP ….
- **Mode en connexion (mode connecté)** :
  - Les échanges sont contrôlés par des connexions: délimitation dans le temps par deux phases d'ouverture et de fermeture de connexion.
  - Exemples : téléphone, protocoles PPP, TCP ….

Gestion des connexions : le mode connecté

- **Ouverture:** Délimitation de début des échanges.
  - **Désignation** du destinataire à atteindre .
  - **Négociation** de qualité de service QOS "Quality Of Service".
    Paramètres **qualitatifs** Ex: mode simplex, à l'alternat, duplex.
    Paramètres **quantitatifs** Ex : débit, taux d'erreur résiduel accepté.
  - **Désignation de la connexion** : notion de référence de connexion.
- **Fermeture:** Délimitation finale des échanges.
  - **Déconnexion abrupte** ("Destructive release"):
    Perte des informations en transit.
  - **Déconnexion ordonnée** ("Orderly release"):
    Echange des données en cours avant fermeture
  - **Déconnexion négociée** ("Negociated release")
    Fermeture avec l'accord des deux parties.

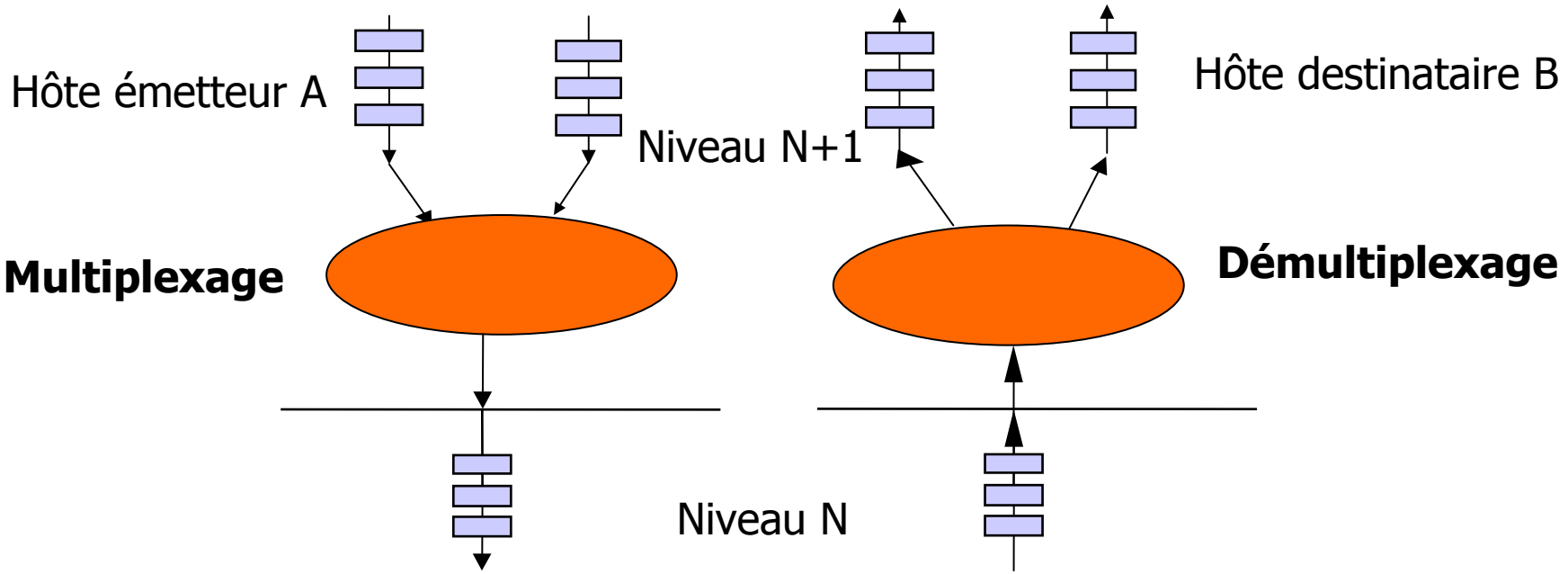
3) Eclatement/Recombinaison

- **Éclatement ("Splitting")** : Une connexion de niveau N+1 est éclatée sur plusieurs connexions de niveau N.
- **Utilisation :** pour améliorer le débit (usage peu fréquent).
- **Opération inverse** sur le site distant: la **recombinaison =>**Problème : **reconstruire la séquence** initiale (utilisation de numéros de séquence).



4) Multiplexage (accès multiple) 'Multiplexing'

- **Multiplexage ('switching') :** Acheminer plusieurs flots de messages de niveau N+1 sur un même flot de niveau N.
- **Problème à résoudre**: **rassembler** puis **séparer** les messages appartenant aux différents flots de communication.

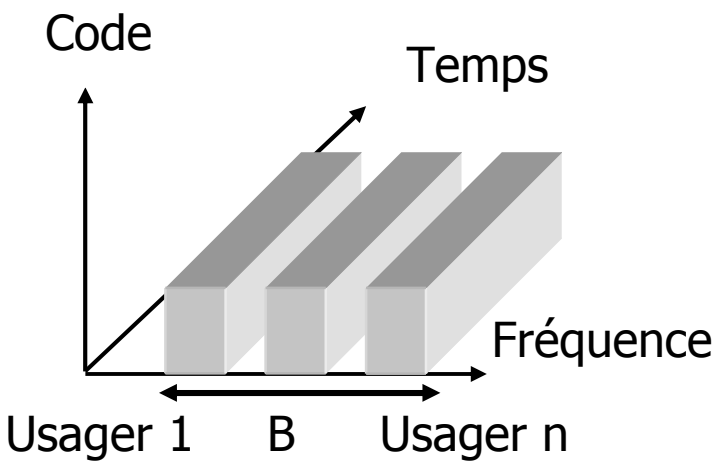


Utilisation du multiplexage

- **Multiplexage de flots appartenant à des usagers différents**: protocoles d'accès multiple.
  - Exemple: téléphonie, réseaux d'ordinateurs.
- **Multiplexage de flots appartenant à un même usager.**
  - Exemple: ouverture de plusieurs connexions différentes entre deux hôtes pour le même usager.
- **Multiplexage de flots appartenant à des protocoles différents.**
  - Exemple: multiplexage des protocoles TCP/IP et Novell sur le même câble Ethernet**.**
- **=> Le Multiplexage est omniprésent en réseaux à tous les niveaux.**

Solutions de multiplexage : a) Multiplexage en fréquence

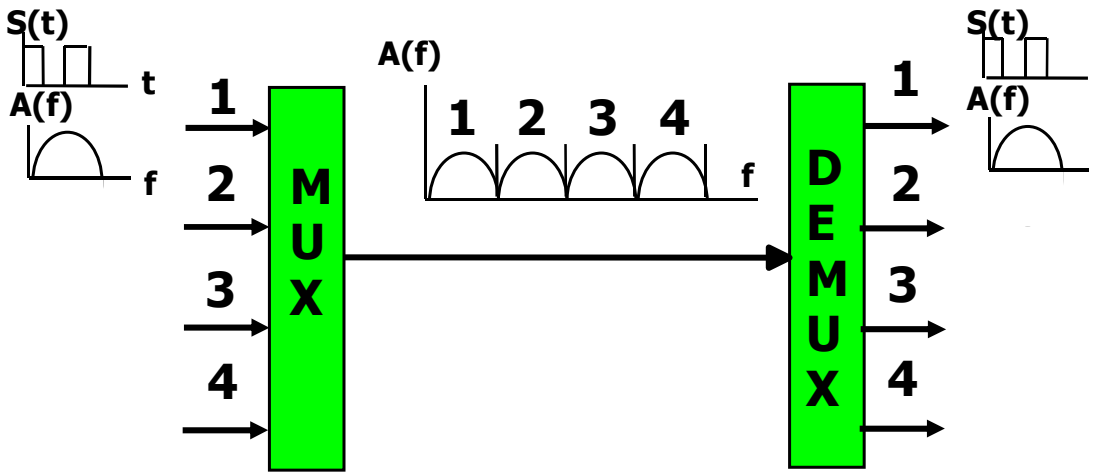
- **FDMA 'Frequency Division Multiple Access'.**
- **AMRF 'Accès multiple à répartition de fréquences'.**
- **Multiplexage analogique :** chaque usager utilise une bande de fréquence fixée selon un codage prédéfini pendant une plage de temps (selon réservation fréquence/temps).



- Les usagers sont séparés par les fréquences.
- La largeur de bande est limitée par le nombre n d'usagers (pour un usager B/n).
- Chaque usager dispose en continu de son canal: qualité garantie mais perte de ressources si l'usager ne transmet pas.
- Les usagers n'ont pas à se synchroniser.
- Les usagers utilisent le plus souvent le même codage (mais les codages peuvent aussi être différents).

Multiplexage fréquentiel : exemple du multiplexage de voies physiques

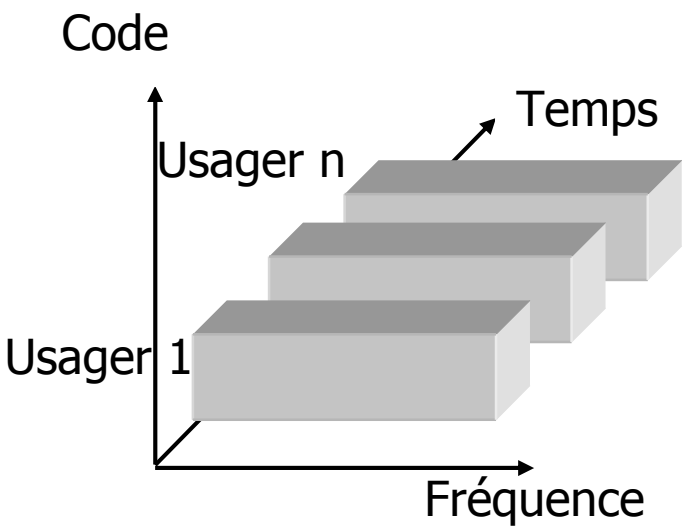
- **Principe :** Le spectre d'une voie rapide est découpé en bandes étroites associées à chaque voie basse vitesse.



- **Applications anciennes:** téléphonie fixe (groupes primaires à quaternaires) ou cellulaire (Radiocom 2000), réseaux locaux "Broadband".
- **Application actuelle:** modulation OFDM, multiplexage dans des fibres optiques WDM 'Wavelenght Division Multiplexing' (16), Dense-WDM (80-160), Ultra-dense (400).

Solutions de multiplexage : b)Multiplexage en temps (temporel)

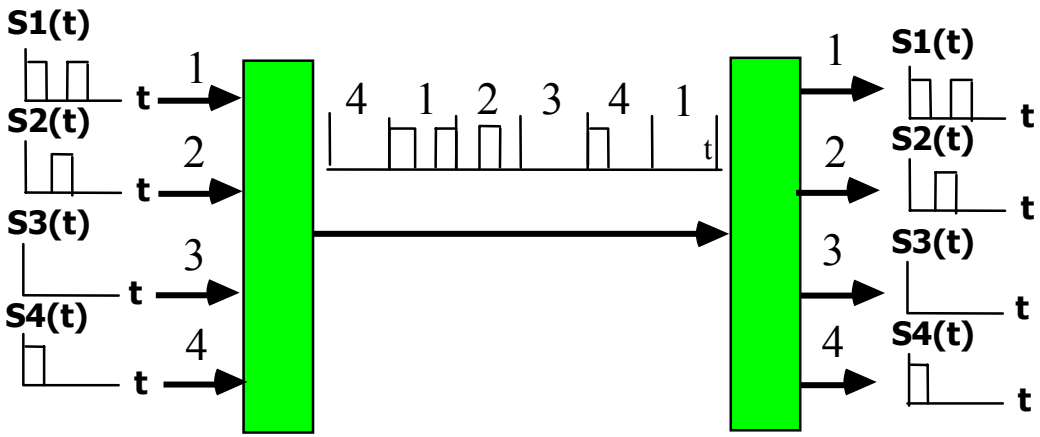
- Chaque usager utilise des d'intervalles de temps (méthode de partage du temps), d'une bande de fréquences prédéfinie, le plus souvent selon le même principe de codage.



- Les usagers sont séparés par le temps.
- Les usagers émettent et reçoivent de façon discontinue dans le temps : ils doivent se synchroniser.
- Meilleure optimisation possible des ressources.

Multiplexage temporel (1) : multiplexage temporel synchrone

- **TDMA** "Time Division Multiple Access",
- **AMRT** "Accès Multiple à Répartition de Temps"
- **Exemple**: multiplexage temporel synchrone de voie physique



- **Principe** :
  - Des intervalles de temps constants d'une voie haute vitesse sont périodiquement attribués à des échantillons de n voies basse vitesse (même s'il n'y a n'a rien à transmettre).
  - On forme des trames répétées de manière synchrone.
- **Applications** :
  - Téléphonie fixe et RNIS-BE (Réseau Numérique à Intégration de Services Bande Étroite). Téléphonie mobile GSM.

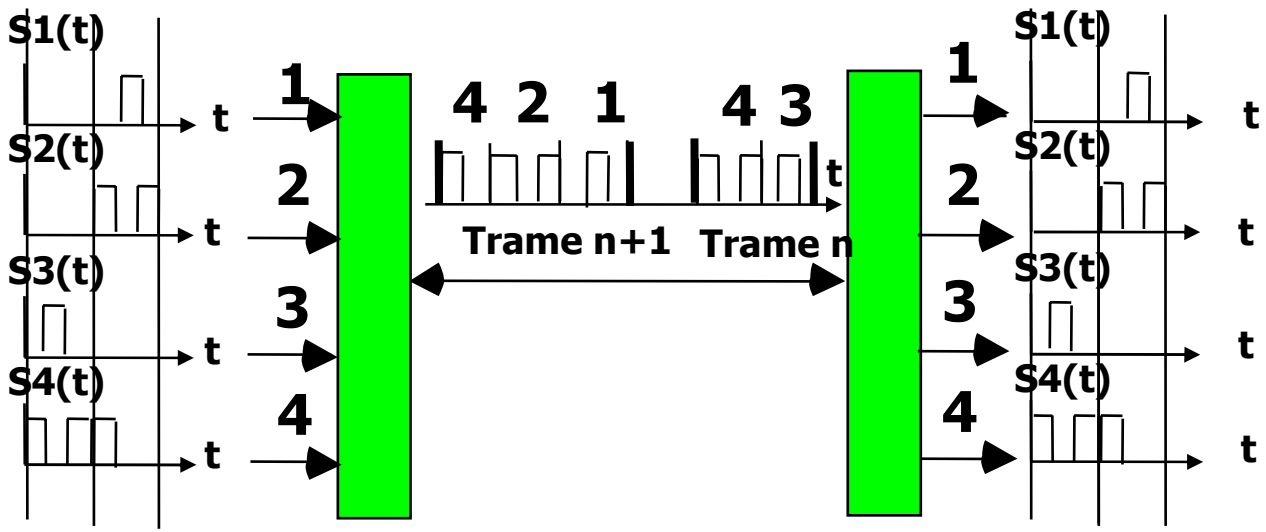
Multiplexage temporel (2) : multiplexage temporel asynchrone

- **ATDM "**Asynchronous Time Division Multiplexing"
- **MTA** "Multiplexage Temporel Asynchrone"

  En français assez souvent => **Multiplexage statistique.**
- **Principe** : Les unités de données significatives (messages) des voies basses vitesses sont acquises selon leur rythme d'arrivée et sont émises sur la voie haute vitesse.
- **Multiplexage des réseaux informatiques à tous les niveaux**: multiplexage de niveau liaison sur niveau physique, niveau réseau sur liaison, transport sur réseau, ...etc
  - **Réseaux locaux**: partage d'une voie commune en compétition Ethernet, Wifi.
  - **Internet** : protocoles PPP, MPLS , IP, TCP
  - **Réseau ATM** "Asynchronous Transfer Mode".

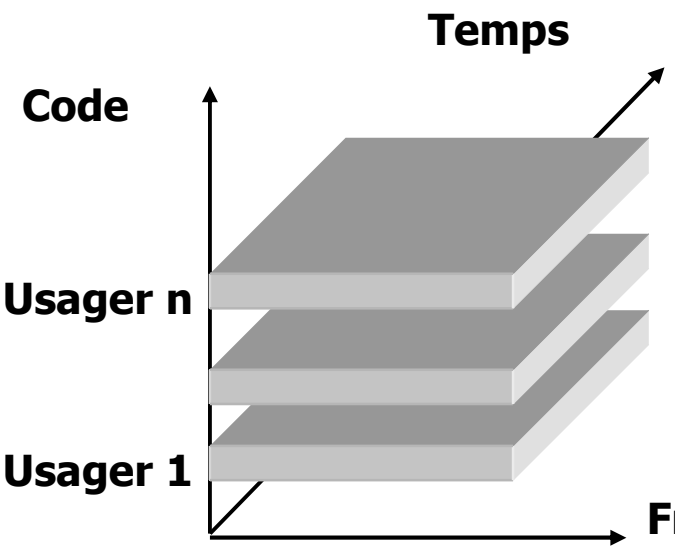
Exemple : le multiplexage temporel asynchrone d'une voie physique

- **Notion de multiplexeur statistique :** Partage d'une voie haut débit en plusieurs voies bas débits**.**
- **Principe :**
  - Les données sont échantillonnées lorsqu'elles sont présentes sur des voies basses vitesses et sont rassemblées en trames sur la voie haute vitesse.
  - Un codage d'adresse ou un code préfixe permettent de déterminer si un échantillon est présent ou pas dans la trame.



Solutions de multiplexage : c) Multiplexage en code

- **CDMA 'Code Division Multiple Access'.**
- **AMRC 'Accès multiple à répartition de codes'.**
- Les usagers utilisent des **codes différents** sur une bande de fréquences prédéfinie pendant une plage de temps donnée (selon réservation).

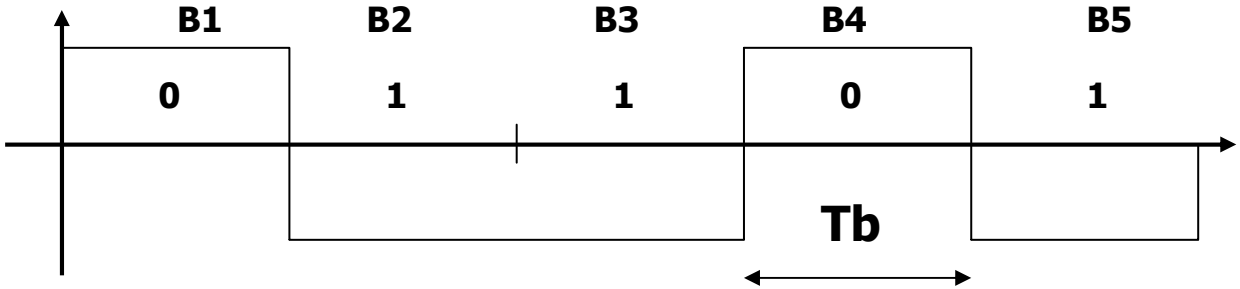


- Les usagers sont séparés par le code utilisé.
- Les usagers peuvent émettre et recevoir de façon continue et simultanément dans la même bande de fréquence.
- Les usagers n'ont pas besoin de se synchroniser dans le temps ni de se distinguer dans le domaine des fréquences..

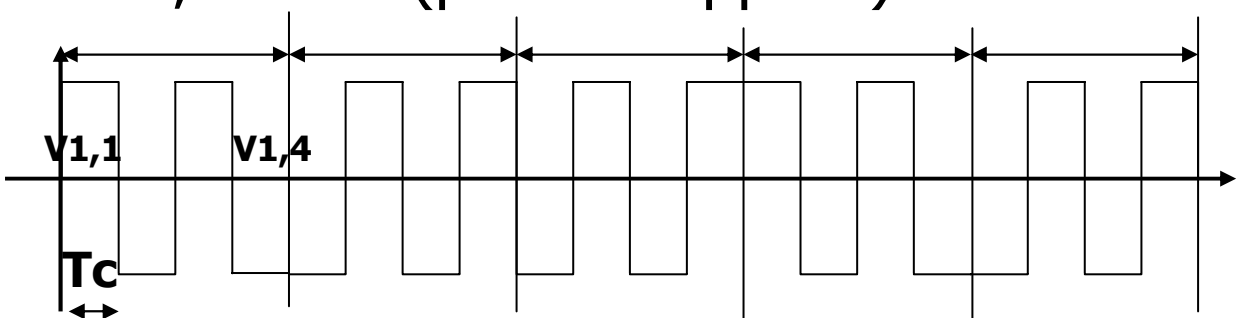
Exemple très simplifié de CDMA : 1) Emission

- Bit Bi =0 ou 1 représenté en niveaux (NRZ).

  0 => +1 et 1 => -1 pendant un temps bit Tb.

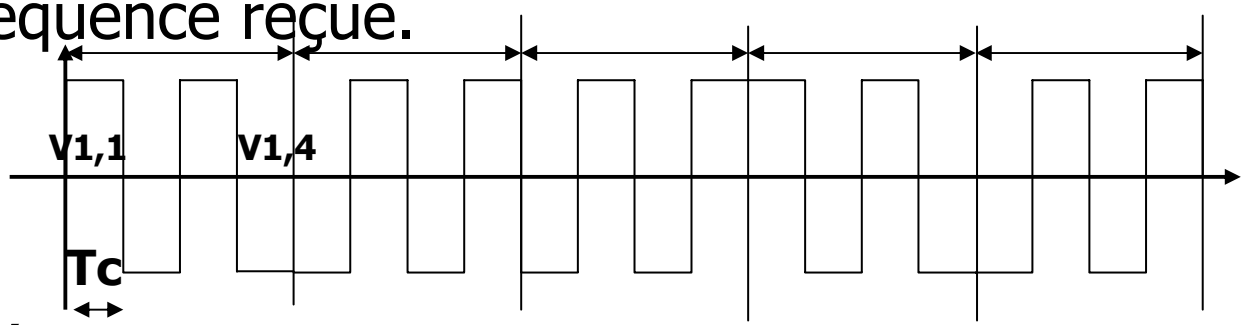


- Pour chaque usager définition d'une séquence d'étalement. Chaque temps bit est divisé en n 'chips' (Tb = n* Tc)
- Exemple pour n=4 : On utilise un code pour 0 C1=+1, C2=-1, C3=+1, C4=-1 (pour 1 l'opposé).



Exemple très simplifié de CDMA : 2) Réception

- Séquence reçue.



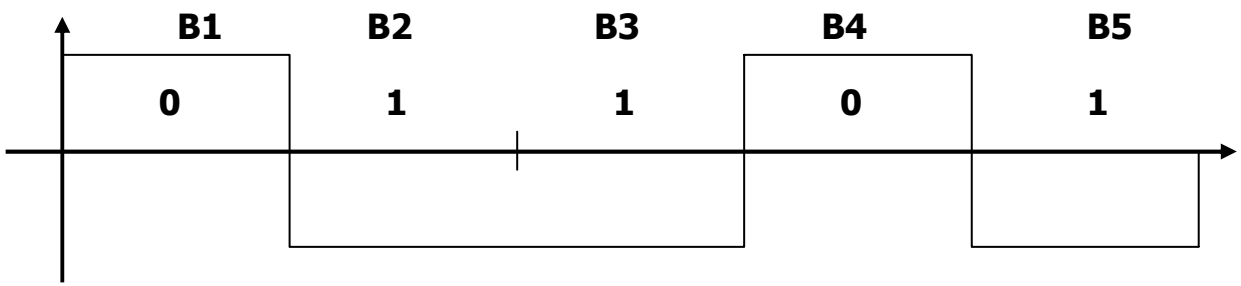
- Décodage pour : C1=+1, C2=-1, C3=+1, C4=-1

  On calcule $Bi = (1/n) \Sigma_{j=1,4} Vi,j\ Cj$

- Pour le bit B1 = (1/4)*(1+1+1+1+1) =1

  Pour le bit B2 = (1/4)*(-1-1-1-1-1) =-1 etc

- On restitue la séquence émise :



Fonctionnement très simplifié de CDMA avec plusieurs utilisateurs 1)

- **Séquences de chips**: On choisit des vecteurs orthogonaux (les produits scalaires de séquences associées à deux usagers s'annulent).
- **Exemple:** Usager-1 (+1, +1, +1, +1)
  Usager-2 (+1, -1, +1, -1)
  Usager-3 (+1, +1, -1, -1)
  Usager-4 (+1, -1, -1, +1)
- **Hypothèses simplificatrices :**
  Bits synchronisés à l'émission.
  Pas de retards différents de propagation.
  Pas de bruit.

Exemple de CDMA avec plusieurs utilisateurs (2)

- **Un scénario de fonctionnement ou les quatre usagers émettent ensemble**
- **Emission** Usager-1 bit a (+a, +a, +a, +a)
  Usager-2 bit b (+b, -b, +b, -b)
  Usager-3 bit c (+c, +c, -c, -c)
  Usager-4 bit d (+d, -d, -d, +d)
- **Signal émis** pendant un bit
  (a+b+c+d),(a-b+c-d),(a+b-c-d),(a-b-c+d)
- **Réception de la transmission de l'usager 2**
  1/4[(a+b+c+d)-(a-b+c-d)+(a+b-c-d)-(a-b-c+d)]
  = 1/4 [4b] = b

Conclusion CDMA

- **Idée d'utiliser les codages :** pour distinguer les usagers dans une approche de transmissions simultanées (en compétition avec collisions).
- **Transposer l'idée dans le monde réel : problèmes importants de mise au point**
  - Séquences de chips **non synchronisées.**
  - Séquences de chips **non orthogonales** : beaucoup d'usagers => nombreuses séquences de codage pas toutes orthogonales**.**
  - Présence de **bruit**.
  - En utilisation mobile et communication hertzienne : Effet **Doppler** du aux mobiles en mouvement**.**
  - Communications hertziennes : **Réflexions multiples**.
- **Applications**
  - Transmissions militaires**.**
  - Téléphonie mobile de troisième génération UMTS**.**

5) Commutation (‘switching’)

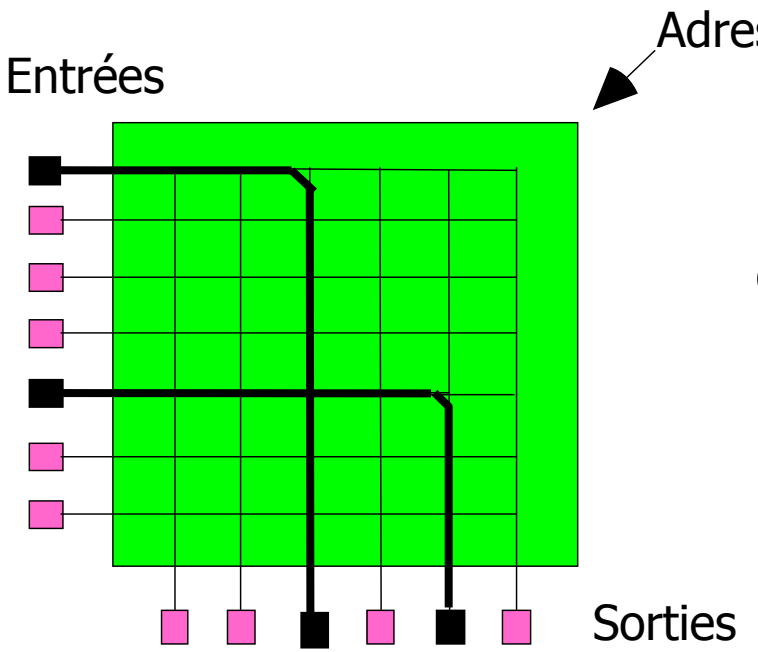
- **Objectif général :** Acheminer un message dans un réseau maillé en visitant des commutateurs successifs.
- **Commuter (‘switching’)**: Réaliser l'aiguillage d'une donnée d'une voie entrante d’un commutateur sur une voie sortante.
- **Router (‘routing’)**: Déterminer le chemin à suivre pour aller d'un point à un autre (calculer des routes optimales).
- **Optimiser**: Gérer des files d'attente associées aux voies (en entrée, en sortie ou au milieu)
  - pour satisfaire des critères de **qualité de services** (temps de réponse, gigue, débit …).
  - et limiter au maximum **les pertes de messages dues à la congestion**.
- **Un problème majeur en réseaux** : la commutation est traitée aux niveaux 2 et 3.

Solutions de commutation : a) Commutation de circuit

- **Principe : Un chemin permanent** (**un circuit**) est établi entre deux entités communicantes et assure un débit (une bande passante) **fixe**.
- **Avantages**
  - Existence d'un canal **totalement disponible** et indifférent au type de données transférées.
  - Permet de **réserver une capacité nette pour des trafics synchrones ou temps réel**: voix, image, données à échéances.
- **Inconvénients**
  - La capacité mise à disposition **n'est pas toujours adaptée au besoin** et peut-être parfois **extrêmement mal employée** (données sporadiques en informatique, voix ou images compressées).

Solutions de commutation de circuit : a1) Commutation spatiale

- **Principe :** Établissement d'un lien métallique permanent au moyen d'aiguillages d'interconnexion.
- **Exemple type ("crossbar") :** Commutateur N entrées N sorties : N^2 aiguillages commandés par adresses destinations.

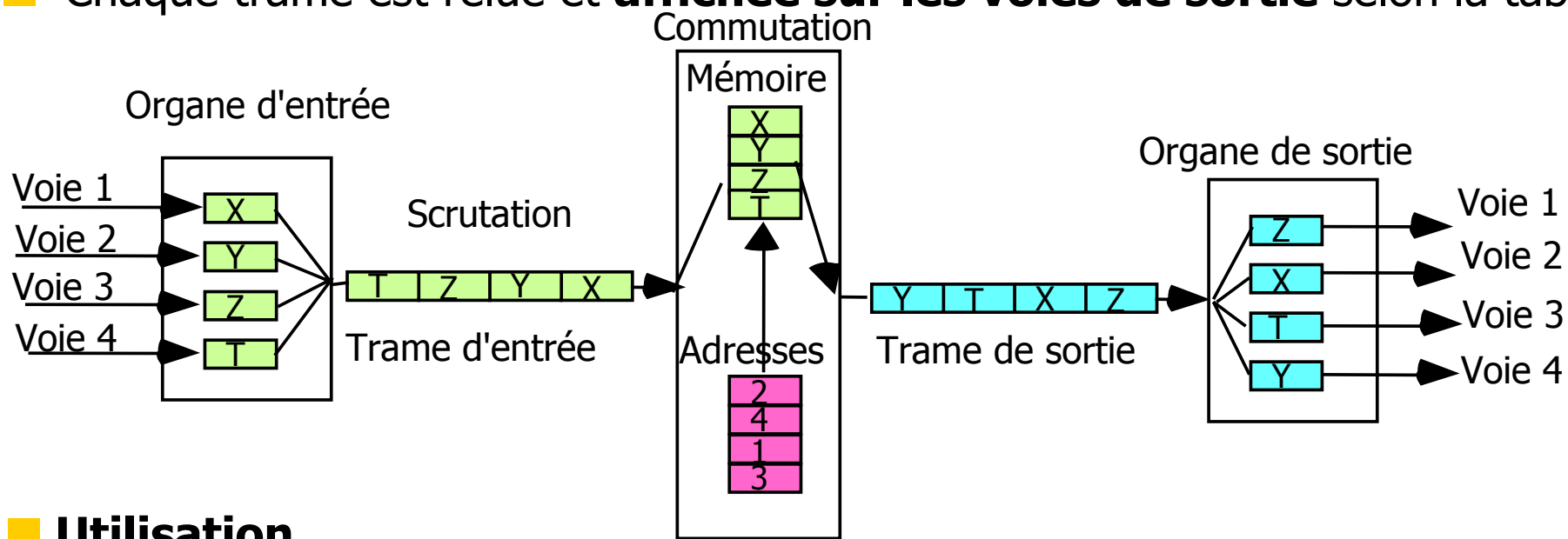


- **Optimisation pour N grand** : diminuer le nombre de points d'aiguillages => notion de réseau d'interconnexion

- **Utilisation**
  - Commutation téléphonique ancienne.
  - Commutation électronique haut débit (accès mémoire, commutateurs gigabits, ...).

Solutions de commutation de circuit : a2) Commutation temporelle synchrone

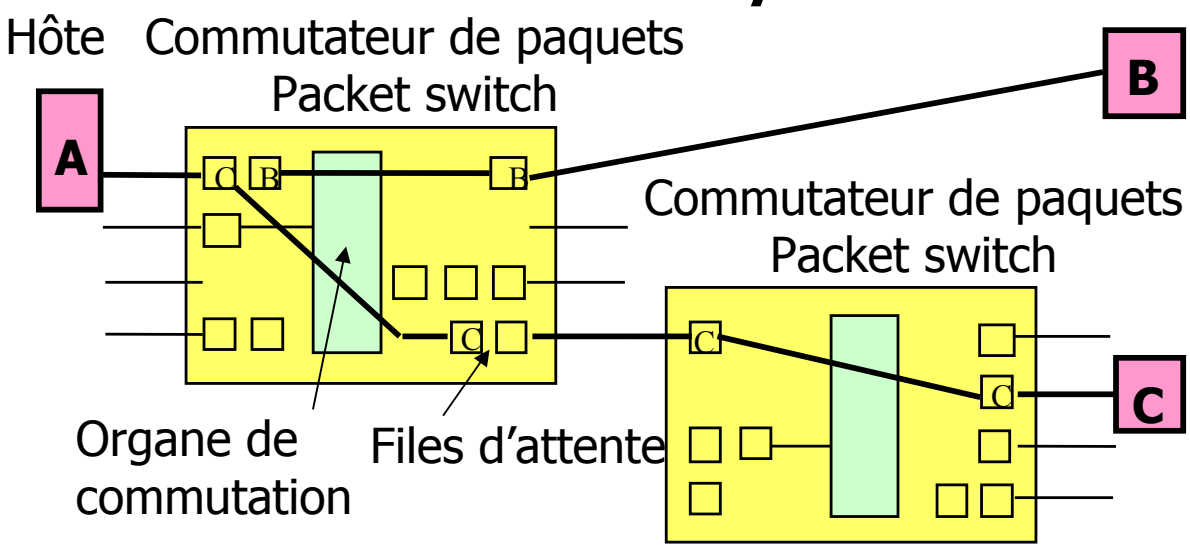
- Chaque entrée est **échantillonnée** de façon **synchrone**.
- On constitue une trame d'entrée. On écrit la trame **en mémoire**.
- Une table de **correspondance** entre voies d'entrée et voies de sortie définit les règles d'aiguillage.
- Chaque trame est relue et **affichée sur les voies de sortie** selon la table.



- **Utilisation**
  - Commutation téléphonique filaire actuelle, RNIS.

Solutions de commutation : b) Commutation de paquets

- **'Packet switching' : Une commutation temporelle asynchrone.**
- Les paquets ont une entête avec **l'adresse du destinataire**.
- Les paquets arrivent de façon **asynchrone** (multiplexage asynchrone).
- Les paquets sont **mis en file d'attente** entrée après acquisition sur une voie d'entrée,
- **Les paquets sont aiguillés vers une file de sortie** en fonction de l'adresse destinataire à atteindre par un organe de commutation.
- **Les paquets en file de sortie sont renvoyés** sur la voie de sortie.



Commutation de paquets : Terminologie, Concepts associés

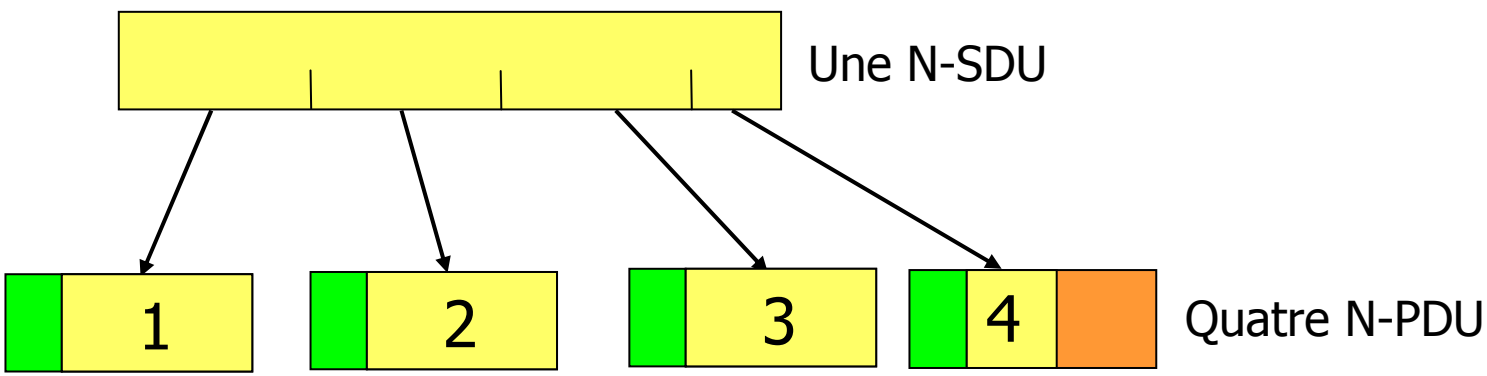
- **Commutation de messages, de paquets, de cellules**
  - **Commutation de messages:** ancienne commutation de données de taille quelconque => problèmes d'allocation de tampons (Exemple : réseau SITA au début).
  - **Commutation de paquets:** la taille des paquets est bornée par connexion usager et permet une meilleure gestion mémoire (X25).
  - **Commutation de cellules:** la taille des cellules est strictement fixée et identique pour tous les usagers => efficacité maximale en gestion mémoire. (ATM)
- **Circuits virtuels et datagrammes**
  - **Circuits virtuels :** les paquets empruntent le même chemin.
  - **Datagrammes :** les paquets circulent indépendamment les uns des autres.

Commutation de paquets : Avantages et inconvénients

- **Avantages**
  - **Apporte une grande adaptabilité aux débits** soumis par les usagers.
  - **Optimise les voies** de communication.
- **Inconvénients**
  - L'opération de commutation est plus **complexe** qu'en commutation de circuits et les débits commutés sont plus **faibles.**
  - Les trafics voix image ont des caractéristiques **synchrones** qui rendent délicate l'utilisation de la commutation de paquet.
- **Applications**
  - Commutation des réseaux informatiques : Ethernet, ATM, MPLS, IP, X25.

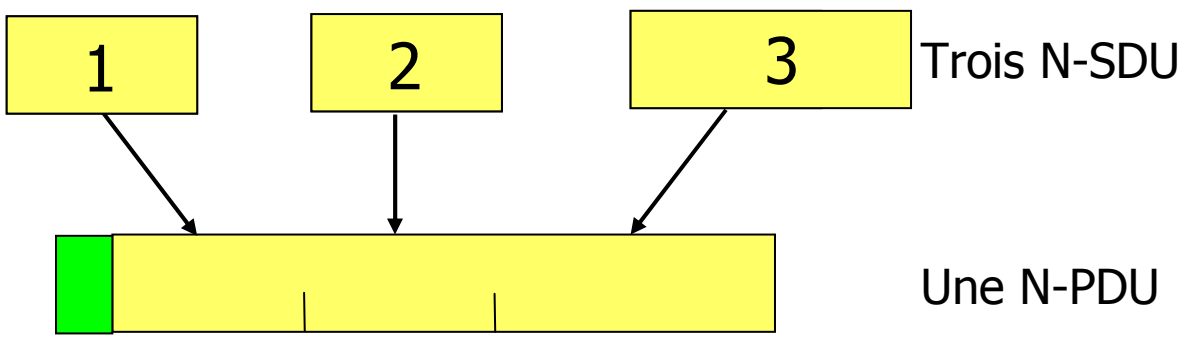
6) Fragmentation / Segmentation 'Fragmentation'

- **Problème:** Dans le cas ou une information usager à transporter au niveau N+1 (N-SDU) est **trop longue** pour la taille (d'un maximum imposé) des messages du niveau (N-PDU).
  - **Exemple:** Ethernet taille maximum 1500 octets
  - **Notion de MTU** : 'Medium Transmission Unit'.
- **Autre situation:** La voie de communication est trop bruitée.
  - Exemple : Wifi.
- **Solution (rencontrée dans certains niveaux 2,3,4):** découper les messages longs en morceaux plus courts (segmentation ou fragmentation).
- **Opération inverse : réassemblage** (identifiant de fragment, position).



7) Groupage (‘Concaténation’)

- **Problème :** dans le cas ou une information à transporter (N-PDU) est **trop petite** par rapport à la taille des messages qui optimisent les performances du niveau (taille fixe ou d'un minimum donné)
- **Optimisation** (groupage) : regroupement de messages courts (rare).
- **Opération inverse** : **dégroupage** (basé sur la position des différents messages courts).



8) Contrôle de flux ‘Flow control’

- **Problème :** différences de vitesse entre l’émetteur et le destinataire (hétérogénéité des puissances de calcul, de la capacité mémoires, des débits des voies de communication)

  => Arrivées sporadiques qui ne peuvent être traitées par le destinataire (pertes de messages).
- **Solution indispensable (niveaux 2, 3, 4) :** Adaptation de la vitesse de l'émetteur à celle du récepteur par rétroaction au moyen de messages spécifiques: les crédits.
- **Contrôle de flux inter sites "Peer flow control"**
  - Entre deux niveaux appariés => contrôle de flux défini dans un protocole.
- **Contrôle de flux inter niveaux "Inter layer flow control "**
  - Entre un niveau N+1 et le niveau N : la solution dépend de l'implantation du fournisseur de logiciel système/réseau.

9) Contrôle d'erreur 'Error control'

- **Problèmes :** Perte de messages
  - **Bruit** sur les voies de communication, dans les mémoires.
  - **Incapacité de stocker** (congestion).
- **Solution indispensable (niveaux 2, 3, 4) :** contrôle de **l'intégrité des données transférées**
  - Codes **auto correcteurs d'erreurs :** Définir un code qui permet de savoir s'il y a eu erreur et qui permet de corriger cette erreur.
  - Codes **détecteurs d'erreurs**: Définir un code qui permet de détecter la modification ou la perte des données échangées et retransmettre en cas d'erreur.
- **Contrôle total ou contrôle partiel :**
  - Contrôle sur la **totalité** du message : protocole et usager.
  - Contrôle sur les **entêtes** protocolaires : traiter uniquement des problèmes protocolaires (exemple : erreurs de routage).

10) Contrôle de séquence Propriétés d'ordre

- **Problème :** assurer le respect de propriétés d'ordre dans les communications
- **A) Ordre local (livraison en séquence, besoin fréquent)**
  - On attend en général d'un protocole qu'il préserve l'ordre des données qui lui sont confiées par un émetteur.
  - Si le réseau perd, duplique ou déséquence les informations il faut restituer les données dans l'ordre des soumissions.
- **B) Ordre global** (cas des communications en **diffusion**)
  - Tous les destinataires d'une suite de messages reçoivent les messages dans le même ordre.
- **C) Ordre causal**:
  - Tous les destinataires reçoivent les messages dans le même ordre qui est celui de la relation de causalité des instants d'émission.
  - Un message en cause un autre s'il est délivré avant l'émission de l'autre.
  - Ordre causal = ordre de précédence intersites déduit des communications par messages.

11) Qualité de service 'QOS Quality Of service'

- **Problème**: respect de propriétés dans les communications.
- **Point de vue ancien**: fournir deux niveaux qualitatifs
  - **Transfert de données normales** ("Normal data flow") : Les données habituellement échangées.
  - **Transfert de données expresses** ("Expedited data transfer") : Les données devant circuler rapidement (exemple alarmes, exceptions).
- **Point de vue actuel:** essentiel pour les données multimédia
- **A) Propriétés qualitatives ou 'sémantiques'**
  - Exemple : propriétés d'ordre, existence de canaux unidirectionnels ou bidirectionnels …
- **B) Propriétés quantitatives : qualité de service temporelle (données temps réel, multimédia)**
  - Nombreux paramètres quantitatifs de qualité: Temps de transmission (latence), Variation du temps de transmission (gigue), taux d'erreur.
- **Contrats de qualité de service:**
  - Responsabilité de l'usager 'usage parameter control' et responsabilité du prestataire de service.

12) Contrôle de congestion ‘Congestion control’

- **Problèmes**:
  - Éviter en présence d'une surcharge la **diminution anormale des performances** => continuer à satisfaire les contrats de qualité de service en présence des surcharges acceptées.
  - Éviter le **phénomène d'écroulement** (‘thrashing’) c'est à dire l'effondrement du trafic utilisateur transporté.
- **Solutions** (indispensables aux niveaux 2 et 3) :
  - **Allouer suffisamment de ressources** : bande passante, puissance de commutation, tampons.
  - **Éviter la destruction de messages** par manque de ressources conduisant à des pertes par le développement de protocoles de prévention ou de traitement de la congestion par exemple utilisation de messages de signalement de surcharge.

13) Compression

- **Problème :**
  - **Éviter de transmettre un volume** de données important risquant de gaspiller la bande passante disponible.

- **Solutions :**
  - **Quelquefois présentes au niveau physique et surtout de niveau application.**
  - **Détecter les redondances** présentes dans les données et les supprimer en utilisant un codage plus efficace.
  - Quelquefois utile dans les données informatiques mais **très efficace dans la transmission d'images et de son.**

14) Désignation et liaison ‘Naming, Binding’

- **Problèmes**:
  - **Désignation :** construire des ensembles de noms logiques ou d’adresses physiques (dans le contexte de grands réseaux).
  - **Liaison :** établir un lien entre un nom logique et une adresse physiques permettant de retrouver la localisation d’un destinataire.
- **Solutions de désignation (niveaux 2, 3, 4) :**
  - **Définition de la structure** des noms et adresses dans les réseaux (approches hiérarchiques)
  - **Définition des autorités** responsables de l’attribution des noms et adresses
- **Solutions de liaison (niveaux 2, 3, 4) :**
  - **Définition des services et protocoles de recherche** (utilisant surtout **des annuaires)** pour établir une liaison entre un nom et un attribut qui est le plus souvent une adresse.
  - **On peut aussi associer** à un nom de nombreux autres types d’attributs utiles comme des clés, des alias, des serveurs ….

15) Sécurité 'Security'

- **Problème**: résister aux actions de malveillance pour assurer
  - **Confidentialité :** donner accès en lecture aux personnes autorisées.
  - **Intégrité :** donner accès en écriture aux personnes autorisées
  - **Protection :** vérifier de manière générale les droits de chaque usager.
  - **Authentification :** vérifier l'identité d'un usager.
  - **Non répudiation :** assurer qu'un usager est bien l'auteur d'une action.
- **Solutions indispensables (niveaux 2, 3, 4) :**
  - **Utiliser des fonctions cryptographiques** : chiffres, fonctions de hachage, générateurs de nombres aléatoires.
  - **Utiliser des protocoles de sécurité** : des échanges de messages dédiés à la sécurité.

16) Administration de réseau ('Network management')

- **Problème**: Assurer le suivi de l'exploitation d'un réseau dans cinq domaines.
  - **Gestion des configurations :** suivre le parc des moyens matériels et logiciels déployés et suivi des versions.
  - **Gestion des pannes :** suivre l'état de fonctionnement des appareils.
  - **Gestion de la comptabilité:** établir les consommations de ressources des usagers pour facturer.
  - **Gestion des performances :** suivre les performances des différents dispositifs (débits, temps de réponse).
  - **Gestion de la sécurité :** définir la politique de sécurité et administrer les paramètres de la sécurité (identificateurs, clés, mots de passe)
- **Solutions indispensables aux niveaux 1,2,3,4 :**
  - **Utiliser des logiciels d'administration** qui implantent des **protocoles d'administration** et des services de **présentation des données** d'administration.

Introduction Notions générales

Modèle de référence pour les architectures de réseaux :

Le modèle OSI

Introduction : Notion de modèle de référence

- **Pourquoi définir un modèle de référence d'architecture de réseaux**?
- **1) Pour permettre la construction rationnelle** des logiciels réseaux.
  - Modularité, Extensibilité,
  - Parallélisme, Tolérance aux pannes.
- **2) Pour régler des problèmes d'incompatibilité** entre différentes choix techniques.

  => **Notion d'ouverture (architecture de système ouvert).**

Architectures de systèmes ouverts
Réalisation de l'ouverture (1)

- **Objectif :** Assurer que des **logiciels réseaux hétérogènes** peuvent être intégrés dans des ensembles coopérants de grande dimension sans entraîner des coûts trop importants.

Techniques employées

- **1) La normalisation ("Standardization"):**
  - Production de spécifications papiers faisant référence pour la définition des différents aspects d'un fonctionnement réseau.
  - => Une norme précise des fonctionnements communs obligatoires ('mandatory') et donc par suite autorise des variantes sur les points optionnels et surtout sur les points non spécifiés.

Architectures de systèmes ouverts
Réalisation de l'ouverture (2)

- **2) L'interopérabilité ("Interoperability") :**
  - Production d'implantations qui peuvent effectivement échanger des informations significatives.
  - => Non seulement la connexion physique est réalisée mais également tous les protocoles employés sont compatibles.
- **3) La portabilité ("Portability"):**
  - Un même logiciel 'portable' peut être exécuté sur une grande variété de machines.
  - => On peut ainsi utiliser une implantation commune qui aura de bonnes chances d'être compatible avec elle même.

Architectures de systèmes ouverts
Réalisation de l'ouverture (3)

- **4) L'extensibilité ("Scalability") :**
  - Signifie qu'un système ouvert peut supporter des extensions de configuration jusqu'à une taille très importante (arbitraire si possible).
  - => Dans son développement un système ouvert peut accompagner l'extension des demandes des applications en leurs assurant des performances acceptables.
  - => Terminologie : Passage à l'échelle.
- **5) L'intégration ("Integration") :**
  - Signifie que les applications doivent avoir une interface bien définie et que les implantations respectent ces spécifications.
  - => Les applications peuvent être assemblées (composées) pour former un système complexe fonctionnant correctement dans la mesure ou les clients respectent les interfaces des applications serveuses.
  - => Terminologie : Interface de programmation d'application API 'Application Programming Interface'.

Organisation des architectures de réseaux

Organisation en couches ou niveaux ('Layers', OSI)

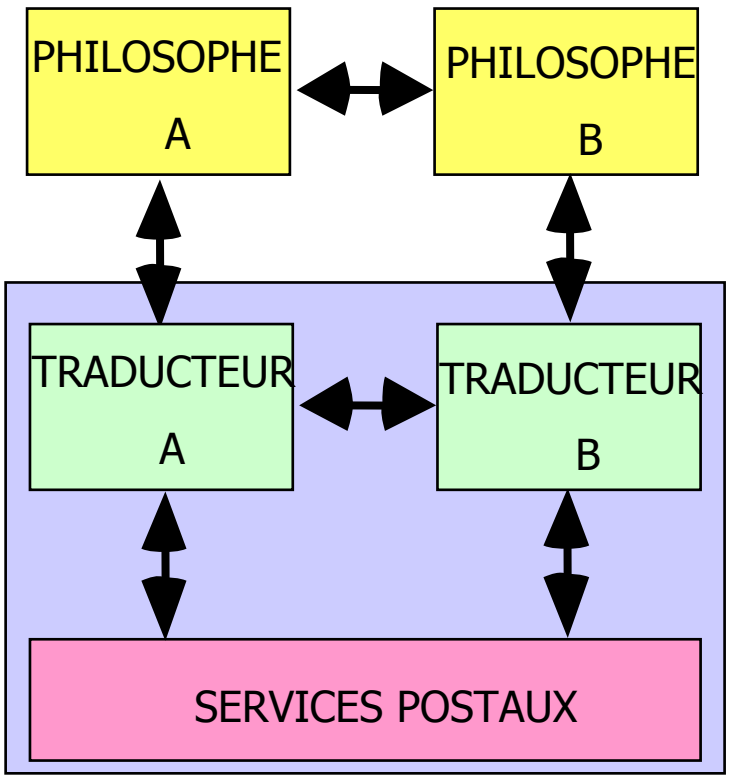
- Pour modulariser et structurer les différentes fonctions:
  - Eviter les approches "fourre-tout" ou tous les aspects sont mêlés dans le même logiciel.
- On situe dans une hiérarchie de couches (une pile ou "stack") les différentes fonctions à réaliser (Dijskstra 1968).
  - Chaque niveau est défini en termes du service rendu au niveau supérieur.
  - Chaque niveau est défini par la façon de dialoguer avec un niveau analogue.

Organisation orientée composants logiciels (objets)

- On ne pas une seule pile (une hiérarchie de réutilisation).
- Tous les composants sont sur le même plan: utilisables via leur API par d'autres composants (Exemple CORBA).

Organisation en couches
Services et protocoles: un exemple

- **Exemple** ('A.S. Tannenbaum'): Deux philosophes qui ne parlent pas la même langue souhaitent mener un débat philosophique par la poste).
  - Ils produisent des **textes philosophiques**.
  - Ils utilisent les **services de traducteurs.**
  - Les textes **circulent par la poste.**



Notion de service (flèches verticales)

- **Interface du service de traduction : requêtes ou primitives.**
  - Pourriez vous envoyer un texte à mon ami le philosophe B qui habite à telle adresse.
  - Oui c'est possible ou Non j'ai trop de travail.
  - Un texte pour vous est arrivé de B.
- **Interface du service postal**
  - Mettre une lettre à la poste
  - Effectuer un envoi recommandé (qualité du service).
  - Guichet surchargé (file d'attente)
  - Le bureau de poste est fermé.

Notion de protocole : (flèches horizontales)

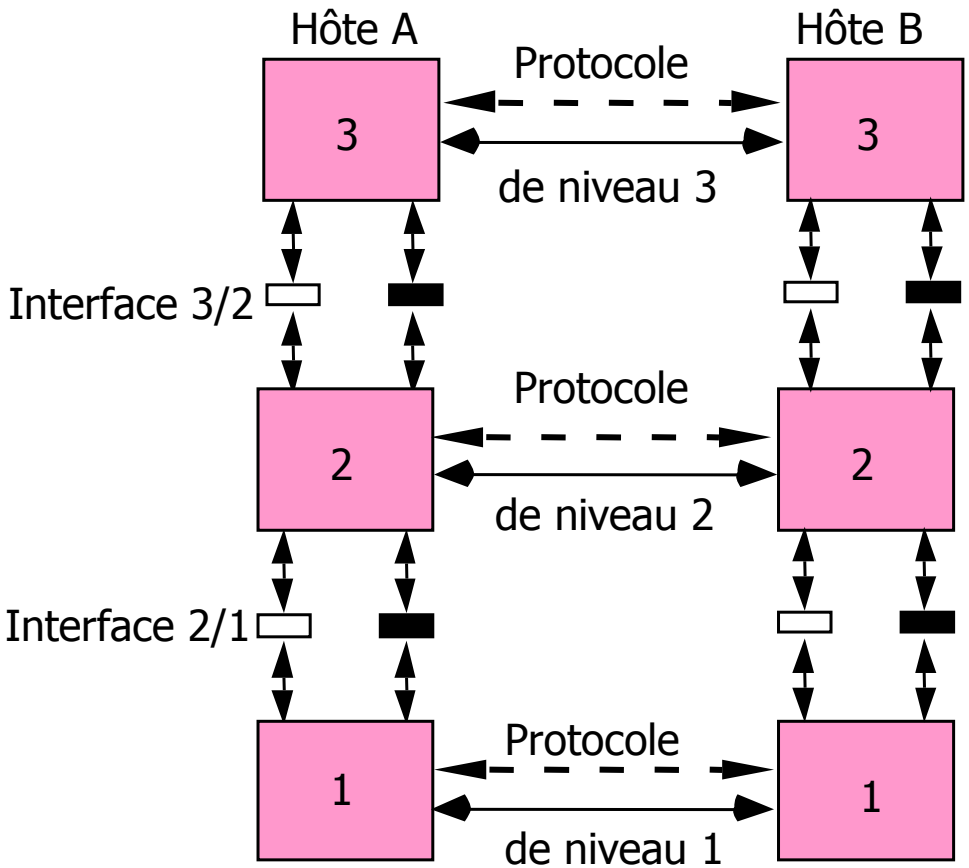
- **Protocole entre philosophes**
  - Cher et estimé collègue.
  - Thèse, antithèse, synthèse.
- **Protocole entre traducteurs**
  - Votre cinquième paragraphe du dernier texte était incompréhensible.
  - Dans une langue "pivot" (anglais) : Que pensez vous d'utiliser pour le prochain envoi l'allemand afin de ne pas perdre la main.

Architecture de communication : Présentation formelle des notions

- **Architecture de réseau ("Network Architecture")**
  - Spécification d'un ensemble de fonctions découpées en niveaux hiérarchisés
- **Couches ou niveaux ("Layers")** définis par:
  - **Une interface de service** : des primitives d'accès au service.
  - **Un protocole de communication** entre un niveau N et un niveau N distant.
- **Services et protocoles**
  - Le niveau **N** communique avec le niveau **N+1** auquel **il fournit un service**.
  - Le niveau **N** communique avec le niveau **N-1** auquel **il demande un service**
  - Les services rendus servent à établir finalement **un dialogue** (protocole) entre **deux niveaux N appariés**.
  - Le niveau **le plus bas** est celui de la **communication effective** sur une voie physique de bits d'information.

Représentation des niveaux, protocoles et services

- **Exemple d'une architecture à trois niveaux.**
- Sur le schéma: distinction flots de contrôle et flots de données (exemples RNIS, ATM).
- Selon les choix de conception ces flots:
  - Circulent sur le même canal physique ou la même voie logique
  - Circulent sur des canaux physiques ou des voies logiques différentes.



Notions liées aux services : Primitives de service

- **Primitive de service ('IDU Interface Data Unit')**
  - **Une fonction** précise activée dans un niveau par un autre niveau.
  - **Exemples :** demande de connexion, demande de transfert, …
  - **Caractéristiques associées** : type, paramètres d'appel et de réponse
    - type de l'opération (de la primitive).
    - adresse du destinataire et adresse de l'émetteur.
    - spécification de qualité de service attendu.
    - données usager.

Notions liées aux services : Services et entités de services

- **Service**
  - **Ensemble de primitives** échangées par un niveau donné et le niveau supérieur. Les primitives de service circulent dans les deux sens N vers N+1 et N+1 vers N.
  - **Profil d'appel des primitives :** analogue de la signature.
  - **Contraintes d'enchaînement :** l'interprétation d'une primitive dépend de l'état. Par exemple : dans un certain état d'un niveau certaines primitives sont utilisables et d'autres pas.
- **Notion d'entité de service**
  - **Instance** d'un ensemble de primitives permettant la réalisation effective du service (analogue de l'instanciation de classe en approche objet).

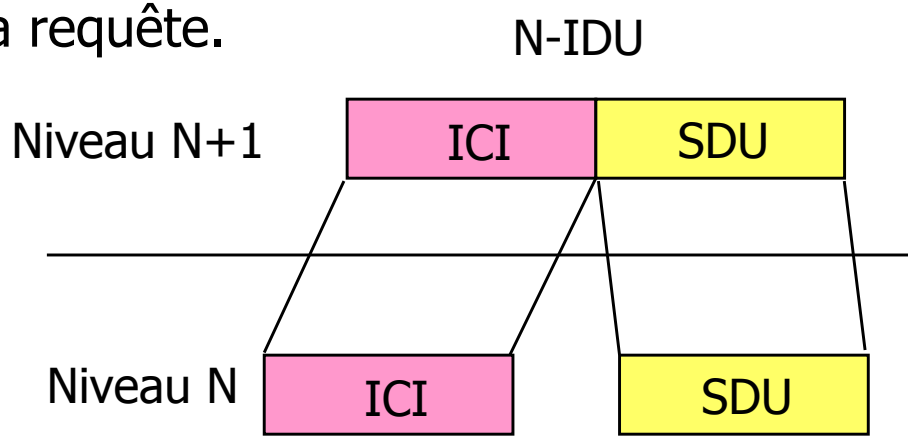
Notions liées aux protocoles

- **Unités de donnée protocolaires ('PDU Protocol Data Unit')**
  - **Spécification d'un ensemble de données** typées, échangées entre deux niveaux appariés. **Selon les niveaux :** trames, paquets, segments, messages.
  - **Exemple :** demande de connexion, transfert de données....
  - **Caractéristiques associées** : les informations transportées
    - type de l'opération
    - adresse du destinataire
    - informations auxiliaires transportées
    - données usagers
- **Protocole**
  - Définition de l'ensemble **des PDU** échangées par un niveau avec un niveau pair.
  - **Définition des contraintes d'enchaînement.**
    - Dans un état d'un niveau certains messages sont interprétables et d'autres pas.
    - L'interprétation d'un même message peut être différente selon l'état.

=> **Indéterminisme** des applications réseaux.

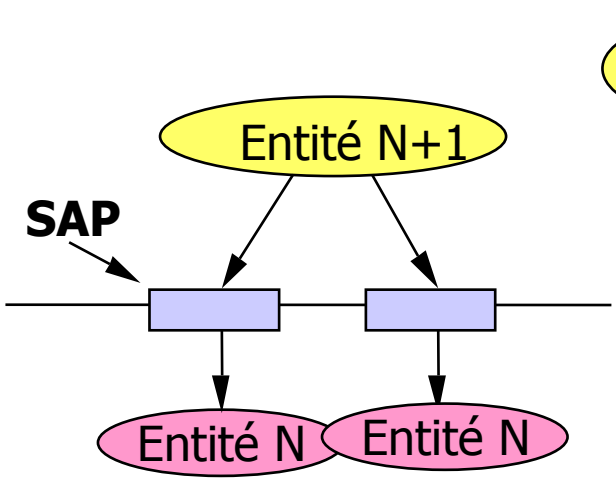
Approfondissements: Services

- **Unité de données d'interface : IDU** 'Interface Data Unit'
  - **Objets échangés entre les niveaux** lors de l'émission d'une primitive de service => Composés de deux parties:
- **1) Unités de données de service : SDU** 'Service Data Unit'
  - La partie que l'usager souhaite **transmettre effectivement** à l'usager distant.
- **2) Informations de contrôle de l'échange : ICI** 'Interface Control Information'
  - L'ensemble des informations de contrôle qui permettent au niveau destinataire de traiter correctement la requête.
- Type de la requête, adresse destinataire,
- Autres informations dans le dialogue entre niveaux (par exemple pour réguler le flux d'informations sur l'interface de service).
- La normalisation ne spécifie pas comment sont échangées les SDU.

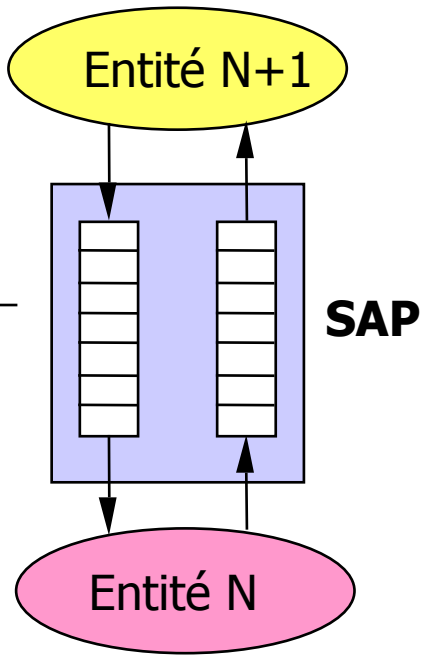


Approfondissements: Point d'accès de services

- **Guichet** permettant à une entité de demander un service à une autre entité .
- **Sa réalisation** dépend des choix d'implantation du logiciel réseau par le fournisseur. Exemples: en appel systèmes, appel de procédure.
- Le point d'accès de service est l'élément essentiel de la **désignation** dans les réseaux.
- Autres dénominations: **port, porte, "sockets", prises**.



SAP : Service Access Point



Réalisation d'un SAP en schéma producteur consommateur

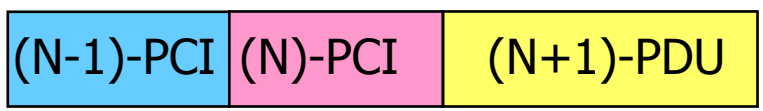
Approfondissements: Protocoles

- **Unités de données de protocole ("PDU Protocol Data Unit")**
  - L'ensemble des objets échangés entre niveaux appariés.
  - Composé d'un (N+1)-PDU et d'une information de contrôle (N)-PCI.
- **Informations de contrôle protocolaire ("PCI "Protocol Control Information")**
  - Ensemble des informations de contrôle de l'échange entre niveaux.
  - Exemples :

  - adresses émetteur et destinataire
  - type
  - version de protocole utilisée.

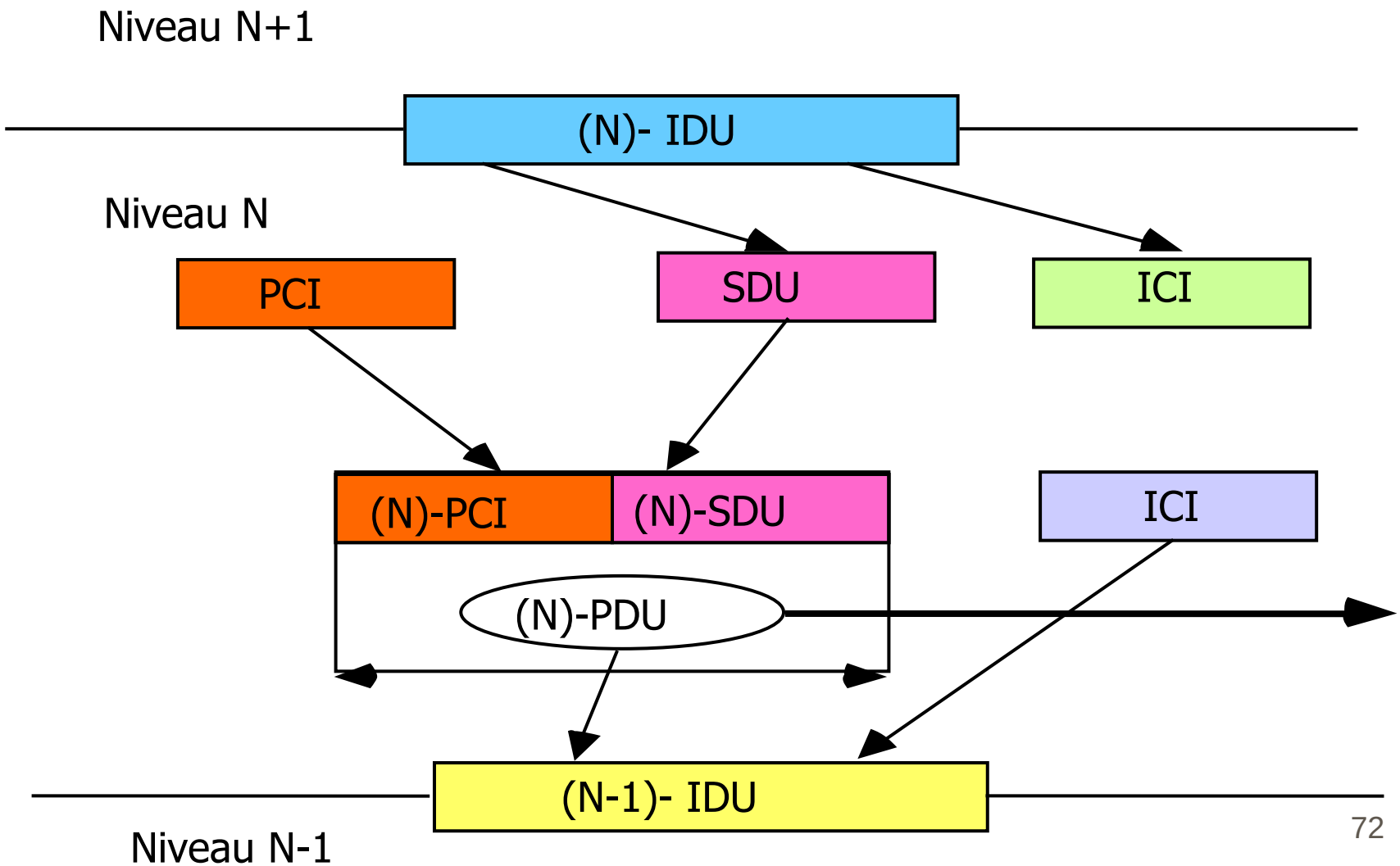
(N+1)-PDU Message (Transport)

(N)PDU Paquet (Réseau)

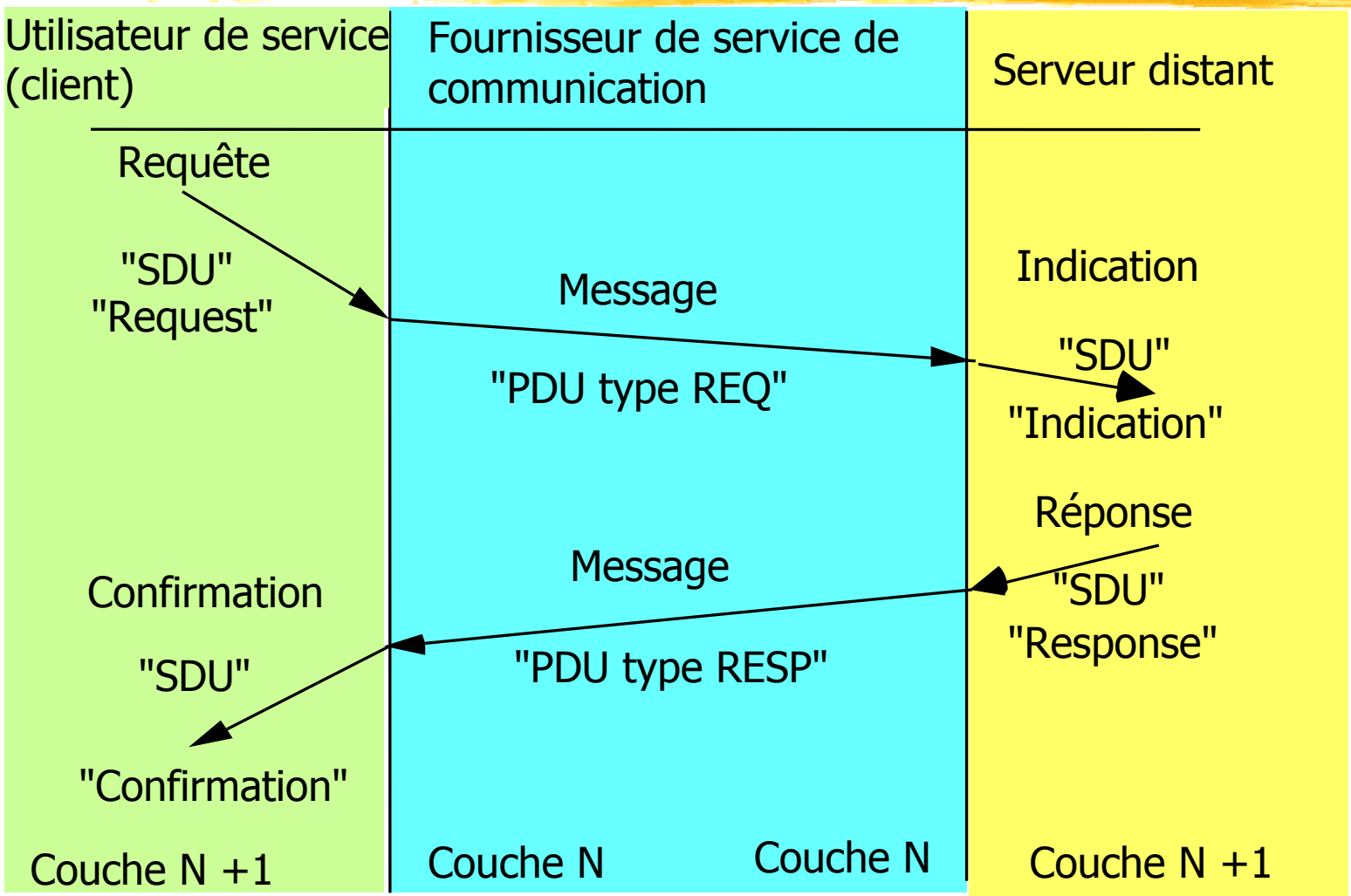


(N-1)-PDU Trame (Liaison)

Approfondissements: Résumé de la structuration OSI



Exemple d'un dialogue avec accord confirmé OSI ('handshake')



Commentaires: dialogue avec accord confirmé OSI ('handshake')

Primitives De Service

- **Requête ("Request") :**
  - Initialisée par le niveau N+1 pour obtenir un service du niveau N
  - Exemple de primitives : Connect-Request, Data-Request.
- **Indication ("Indication") :**
  - Le niveau N avise le niveau N+1 de l'activation d'un service.
  - Exemple de primitives : Connect-Indication, Data-Indication.
- **Réponse ("Response") :**
  - Réponse du niveau N+1 au niveau N sur une indication
  - Exemple : Connect-Response.
- **Confirmation ("Confirmation") :**
  - Du niveau N au niveau N+1 en terminaison du service requis;
  - Exemple : Connect-Confirmation)

Unités de protocole

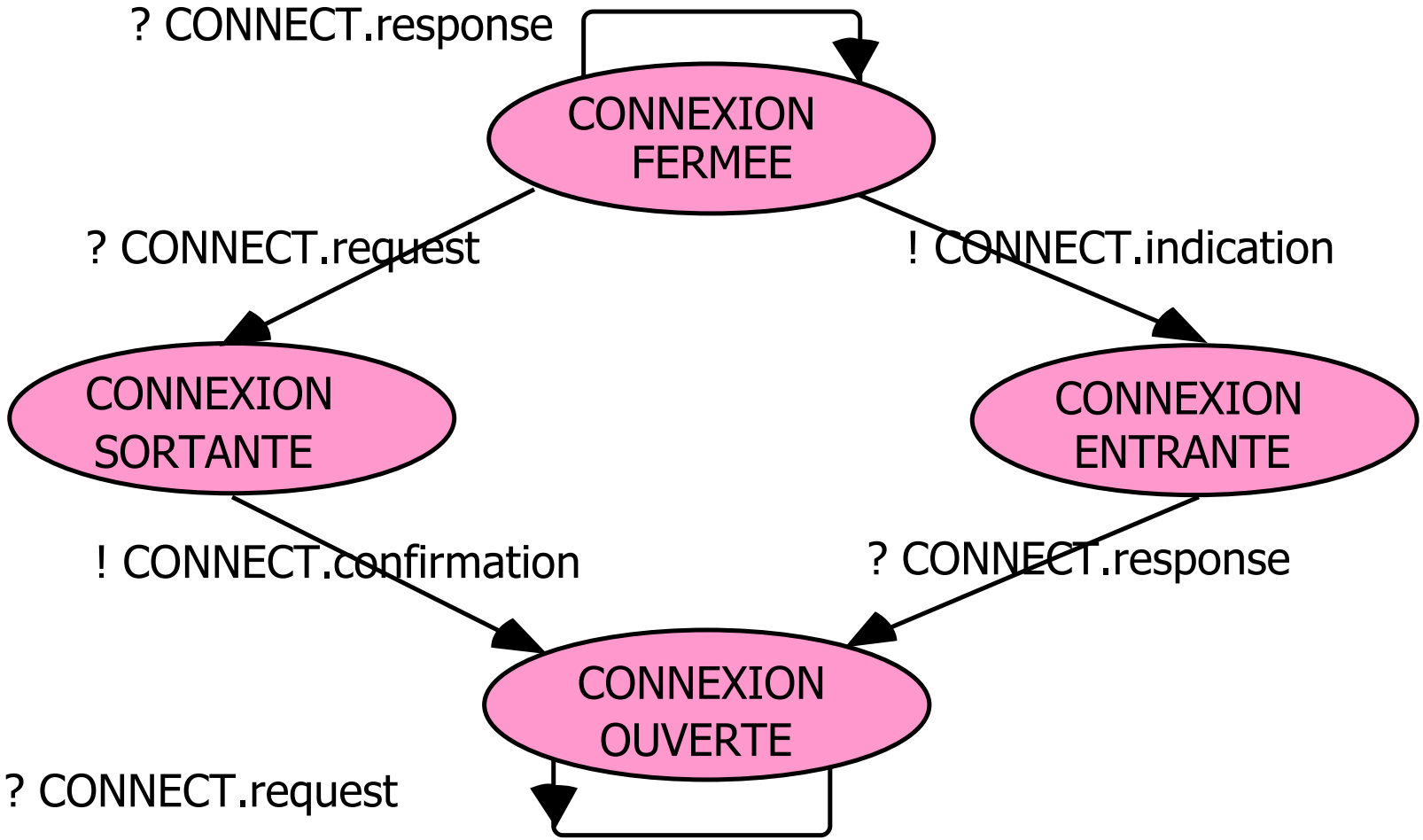
- **REQ:** PDU qui achemine la requête entre le client et le serveur:
- **RESP:** PDU qui achemine la réponse entre le serveur et le client.

Introduction à la spécification des services et des protocoles

- **Définition de toutes les séquences correctes** d'échanges de d'unités de service et de protocole
  - **"ASDC Abstract Service Definition Convention"**
  - **"FDT Formal Definition Techniques"**
- **Exemple de solution (la plus répandue)**

  - Automate d'états finis ("Finite State Machine")
  - Notion d'état ("state")
  - Notion de transition ("transition") entre états.
  - Génération d'événements (émissions d'information) !info.
  - Traitement d'événements (réceptions d'information) ?info.
- **Besoin de méthodes formelles de spécification** et de preuve de comportements corrects 'FDT'
  - Détection des interblocages, des réceptions non spécifiées.

Exemple partiel du service d'accord confirmé de connexion

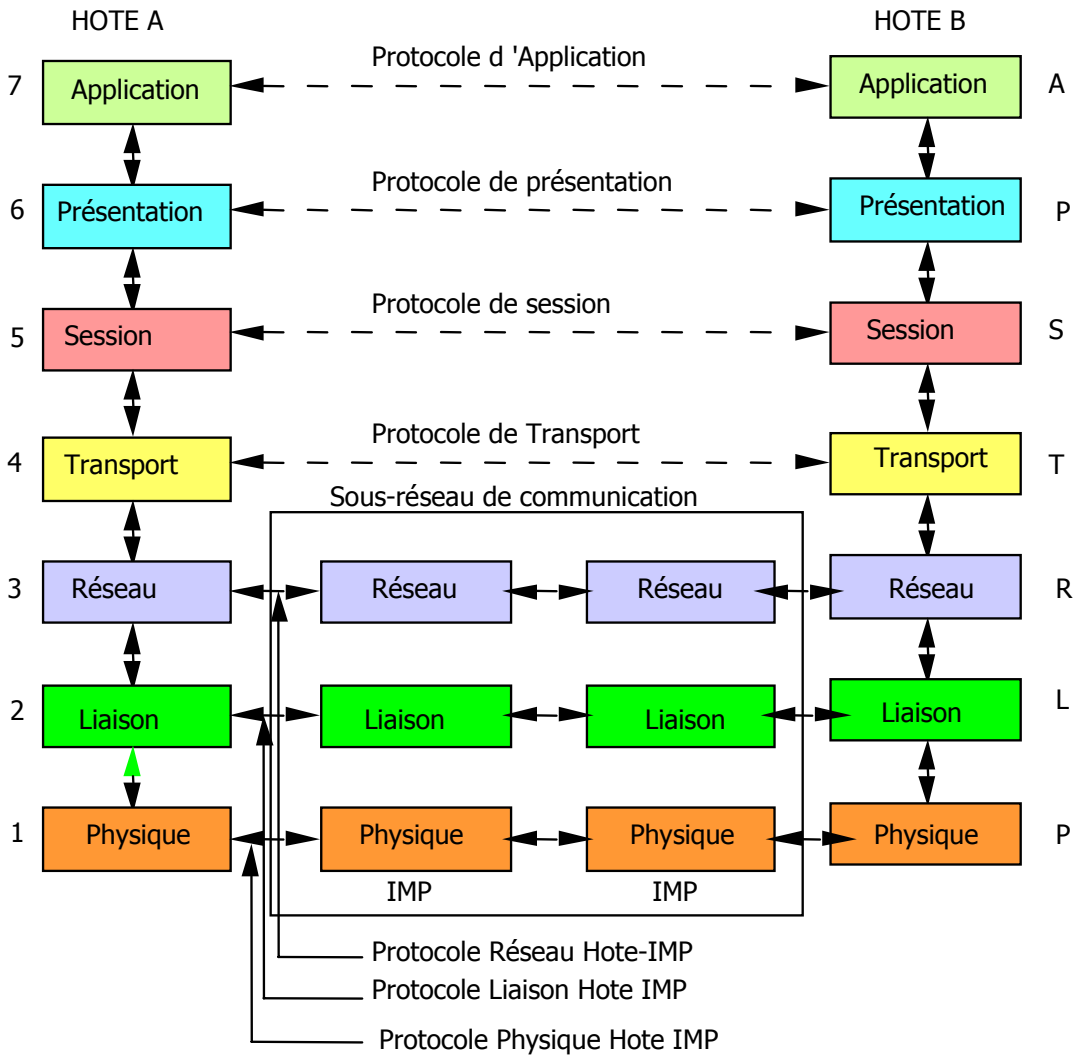


Le modèle OSI (ISO/IEC 7498) 'Open System Interconnection'

Principes de base du modèle

- Les fonctions à exécuter doivent être divisées en **niveaux séparables** du point de vue physique et logique.
- Les fonctions associées dans un niveau doivent avoir une **finalité cohérente**.
- Chaque couche doit contenir **un volume suffisant** de fonctions afin de minimiser le nombre des couches.
- Les protocoles doivent **agir uniquement à l'intérieur** de la même couche.
- Les interfaces entre couches doivent être aussi simples que possible de manière à **minimiser les échanges entre couches.**
- Les couches doivent pouvoir être **modifiée**s sans que soient affectés les services qu'elles offrent.
- Une fonction **devrait n'apparaître qu'une seule fois.**
- L'ensemble **doit être efficace en termes de performances**

Les sept couches



1 - Niveau Physique

- **Objectif:** Fournir les moyens nécessaire à l'activation au maintien et à la désactivation des connexions physiques destinées à la transmission de suites binaires.
- **Mécaniques** : Exemples: connecteurs, forme des prises, utilisation des broches pour les différents signaux.
- **Électriques :** Exemples: modulations, utilisation des niveaux disponibles pour coder un bit, durées.
- **Procéduraux :** Exemples: protocoles de synchronisation entre l'émetteur et le récepteur, multiplexage, transmission bidirectionnelle, à l'alternat.
- **Notion de niveau physique**
  - Définit la globalité de la chaîne de transmission,
  - Exemple : Ethernet au niveau physique sur paire torsadée (100 Base T).
- **Notion d'interface standard**
  - Entre la voie physique de transmission et le système informatique (interface ETTD-ETCD). Constitue une partie du niveau physique.
  - Exemple : interface Ethernet sur paire torsadée RJ45.

2 - Niveau Liaison (1) : Voies multipoints partagées

- **Objectif : Le niveau liaison** assure le transfert d'informations entre deux calculateurs reliés par une voie physique.
- **Selon le support physique et les options de conception** une partie des fonctions suivantes est offerte.

Voies multipoints

- **Partage** de l'accès au médium ("MAC Medium Access Control")
- Autres fonctions (contrôle d'erreur, de flux, administration, …).
- **Exemples :**
  - Réseau Ethernet filaire IEEE 802-3 (10 Mb/s à 10 Gb/s)
  - Réseau Hertzien WIFI 802-11 (1 Mb/s à 54 Mb/s)
  - Gestion de boucles à jeton IBM ISO 8802-5, FDDI ANSI X3T9
  - Réseaux locaux industriels (FIP, Profibus, CAN, …)

Niveau Liaison (2) : Techniques en point à point

- **Gestion de liaison** entre deux voisins reliés par une voie physique quelconque (typiquement une liaison spécialisée).
- **Fonctions réalisées**
  - Délimitation/mise en correspondance d'unités de données.
  - Multiplexage de flots de données d'origine différentes.
  - Contrôle d'erreur : transformer une voie bruitée en une voie de taux d'erreur acceptable.
  - Contrôle de flux.
  - Contrôle de séquence.
  - Établissement et libération de connexions.
  - Fonctions d'administration de liaison
  - Fonctions d'authentification et de sécurité.
- **Exemples d'implantations**
  - Niveaux liaisons en connexion type HDLC 'High Level Data Link Communication' (LAPB, LAPD …).
  - **Internet PPP: 'Point to Point Protocol'.**

Niveau Liaison (3) : Techniques de commutation rapide

- **Utilisation de la commutation** pour acheminer rapidement des trames au niveau liaison entre des stations reliées en mode multipoint.
- Solutions considérées maintenant comme de niveau liaison.
- **1) Commutation de réseaux locaux ('LAN Switching').**
  - Utilisation de techniques de commutation pour acheminer plus efficacement les trames de réseaux locaux Ethernet.
- **2) ATM : Asynchronous Transfer Mode**
  - Réseau numérique à intégration de service large bande réalisant une commutation de cellules à haut débit utilisé pour acheminer rapidement les datagrammes IP (appoche hétérogène avec IP).
- **3) MPLS : Multi Protocol Label Switching**
  - Utilisation de techniques de commutation pour acheminer rapidement des datagrammes IP en cœur de réseau (très intégré à IP).
- **4) Relais de trames FR Frame Relay.**
  - Une simplification pour améliorer la vitesse de commutation des commutateurs X25.

3 - Niveau Réseau

- **Objectif : Réaliser la commutation de paquets** dans un sous-réseau de communication principalement en déterminant comment les paquets sont commutés d'un hôte source vers un hôte destinataire.
- **Selon les options :** une partie des fonctions suivantes est réalisée
  - **Adressage uniforme** de tous les hôtes connectés au réseau.
  - **Routage** (en point à point ou en diffusion) : Échange d'unités de données entre sites reliés par un réseau (indépendamment de la technologie du réseau, local, maillé,...).
  - **Contrôle de congestion**
  - **Fragmentation.**
  - **Multiplexage** (des connexions de réseau sur des connexions de liaison).
  - **Contrôle de séquence.**
  - **Contrôle d'erreur** (détection et correction des erreurs d'hôte à hôte).
  - **Contrôle de flux.**
  - **Gestion des connexions.**
- **Exemple :** IP Internet Protocol
  - X25 niveau paquet.

4 - Niveau Transport

- **Objectif :** Assurer un service de transmission fiable entre processus (donc de bout en bout, "end to end").
  - Le premier des niveaux utilisable directement par l'utilisateur final pour développer des applications.
  - Il résume les fonctions de télécommunications des couches basses.
- **Fonctions réalisées (selon les options de conception)**
  - **Gestion des connexions**.
  - **Multiplexage** des connexions de transport sur des connexions de réseaux.
  - **Contrôle d'erreur**.
  - **Contrôle de flux**.
  - **Contrôle de séquence.**
  - **Segmentation.**
  - **Gestion de la qualité de service.**
- **Exemples de protocoles de transports:**
  - **Internet TCP** 'Transmission Control Protocol' (en connexion).
  - **Internet UDP** 'User Datagram Protocol' (sans connexion).
  - **Internet RTP** 'Real-Time Transport Protocol' (orienté données multimédia).
  - **Autres protocoles :** Novell SPX "Sequenced Packet eXchange" , IBM SNA Niveau "Transmission Control, OSI

5 - Niveau Session (1): L'approche OSI

- **Objectif :** le niveau session structure et synchronise le dialogue entre deux entités.
  - Le transport offre **une voie logique** de communication par **message asynchrone** ("**un tube**") sans organisation spécifique des données échangées.
  - **La session** permet de structurer les échanges pour les coordonner.

L'approche OSI de la couche session

- **La session** définit des fonctions de **reprises sur pannes et de synchronisation**
- **Structuration des échanges en unités de travail**
  - **Activités** (travail important)
  - **Dialogue** (partie d'un travail)
  - **Notion de point de synchronisation** pour délimiter des parties d'un échange.
- **Fonctions de reprise** arrière en cas de panne.
- **Difficulté majeure:** la session OSI a été définie **prématurément** alors que le domaine de la structuration et de la synchronisation répartie était peu avancé.

Niveau Session (2) : L'appel de procédure distante

- **Objectif :** l'appel de Procédure Distante APD 'RPC Remote Procedure Call' permet à un usager d'exécuter une procédure ou une fonction sur un autre site
  - En lui passant dans un message d'appel des paramètres d'appel.
  - En recevant en retour des paramètres en résultat.
  - Un mode de communication synchrone à deux messages.
- **Problème difficile**
  - Assurer en environnement réparti une sémantique pour l'appel de procédure distante voisine de celle connue en univers centralisé.
- **Exemples:**
  - SUN-RPC : 'Remote Procedure Call'
  - OSF-DCE:Open Software Foundation/Distributed Computing Environment
  - OMG-CORBA: 'Object Management Group/Common Object Request Broker Architecture'
  - Java-RMI : 'Remote Method Invocation'.
  - Web Services : SOAP 'Simple Object Access Protocol'.

6 - Niveau Présentation

- **Objectif :** gérer la représentation (le codage) des données échangées.
- **Les conversions**
  - Nécessaires pour tous les types de données
  - Types chaînes de caractères.
  - Types numériques: entiers, flottants, ...
  - Types complexes: articles, ensembles, unions, tableaux...
- **Définition de deux notions.**
  - **Syntaxe abstraite** : permettant la définition d'une grande variété de structures de données (analogue de la syntaxe de définition de types dans un langage évolué).
  - **Syntaxe de transfert** : une représentation unique dans le réseau utilisée pour transférer les données dans les messages.
- **Exemples de niveau présentation:**
  - Réseaux publics: Syntaxe abstraite ASN1 **X208** Syntaxe de transfert **X209.**
  - **SUN-OS: XDR** : "eXternal Data Representation".
  - **CORBA IDL** 'Interface Definition Languages' , **CDR** 'Common data Representation.
  - **Web Services : WSDL** 'Web Services Definition Languages' , **XML** 'eXtended Markup Language'

7 - Niveau Application

- **Objectif :** Le niveau application est défini pour fournir à l'utilisateur des fonctions dont il a besoin couramment.
  - **en termes d'un cadre de développement** d'une application informatique répartie (structuration objet),
  - **en termes de "bibliothèques" de protocoles** (fonctions réseaux) prédéfinies qui déchargent l'utilisateur de travaux répétitifs de programmation d'applications souvent utilisées.
- **Les moyens précédents** permettent à l'utilisateur de développer **ses propres applications** considérées elles aussi comme de niveau application.

Quelques exemples de protocoles de niveau application (1)

- **Désignation :** Créer des espaces de noms et gérer des annuaires permettant à un utilisateur de retrouver des caractéristiques associées à ces noms (principalement l'adresse réseau d'un correspondant).
  - **Exemples** : Internet DNS 'Domain Name System' , LDAP 'Lightweight Directory Access Protocol'.
- **Messagerie :** Permettre d'échanger du courrier électronique entre usagers
  - **Exemples** : Internet SMTP 'Simple Mail Transfer Protocol', protocoles de relève de courrier POP 'Post Office Protocol', IMAP 'Internet Mail Access Protocol', format d'échange MIME 'Multimédia Internet Mail Exchange'.
- **L'échange de documents électroniques :** Permettre d'échanger des documents structurés (textes, images, son, vidéo).
  - **Exemples** : Documents généraux WEB Protocole HTTP, Langages de structuration de documents HTML 'Hyper Text Markup Language' et XML 'Extended Markup Language'
  - **L'échange de données informatisées:** échange de documents administratifs standards sur des réseaux de transmission de données (commandes, factures, ...) entre agents économiques. Normes EDI 'Electronic Data Interchange'

Quelques exemples de protocoles de niveau application (2)

- **Transfert de fichier** : Déplacer des fichiers (plats en général d'un site à un autre).
  - Très nombreux protocoles proposés
  - **Exemple** : Internet FTP 'File Transfer Protocol'
- **Accès aux fichiers distants :** Accès unifié en univers réparti à différents fichiers (réalisation des requêtes d'accès).
  - **Exemple** : SUN-OS NFS 'Network File System'.
- **Gestion transactionnelle :** Cohérence, persistance de données distribuées (assurer le maintien cohérent de données accédées en parallèle en présence de pannes). Optimiser.
  - **Exemple** : X-Open DTP 'Distributed Transaction Processing'. OSI TP.
- **Accès aux bases de données distantes**: Permettre à un client d'accéder à une base de données distante (le plus souvent au moyen de requêtes SQL)
  - **Exemple** : ODBC 'Open Data Base Connectivity'.

Conclusion : Modèle de référence en couches

- **Un incontestable succès :** très nombreuses réalisation de modèles d'architectures de réseau en couches dont l'Internet.
- **Des critiques diverses de ces modèles**
  - Les niveaux ne sont pas également remplis.
  - Certaines fonctions sont déplacées au cours du temps => Nécessaire évolution historique.
  - Certaines fonctions peuvent être répétées à plusieurs niveaux (selon les choix des profils). Par exemple Contrôle d'erreur ou Contrôle de flux. => Ces répétitions dues à des développements séparés ne sont pas forcément inutiles.
  - Les modèles en couches des réseaux sont dominés par une approche télécommunications et n'intègrent pas assez les approches informatiques.

  Exemple : gestion événementielle des applications en mode message => Évolution vers l'approche objet.
- **En fait les modèles en couches des architectures de réseaux sont utilisés dans les aspects transmission des informations.**
- **Plus on se rapproche des applications informatiques traditionnelles, plus on retrouve la structuration de ces applications** (exemple organisation de la couche application).

Introduction Notions générales

Quelques exemples d'architectures de réseaux

Réseaux d'infrastructure (réseaux dorsaux, "Backbone Networks")

- **Objectif :** Définir des infrastructures de télécommunications permettant d'acheminer des données numériques
  - Sur n'importe quelle distance.
  - En général pour des volumes et des débits très importants
  - Pour des données variées : téléphonie, informatique, multimédia etc (les différents trafics qu'un opérateur ou un grand compte doit supporter).
- **Deux générations successives:**
  - PDH "Plesiochronous Digital Hierarchy".
  - SDH "Synchronous Digital Hierarchy".

SDH "Synchronous Digital Hierarchy"

- **Hiérarchie numérique synchrone : G707 , G708, G709** Normes dérivées des travaux SONET ("**S**ynchronous **O**ptical **NET**work") (Bellcore).
- **Solution de niveau physique et liaison :**
  - Traite les problèmes de PDH => un réseau d'infrastructure plus souple.
  - Verrouillage de trames, justification et pointeurs acheminés dans les trames qui permettent de compenser les problèmes de délais de propagation et décalages d'horloges.
  - Pointeurs : L'accès à des circuits de faibles débits dans des trames de débits élevés se fait simplement (extraction rapide).
  - Réseaux maillés SDH et boucles SDH.
- **Exemples de normes et de débits :**

Trame	Débit
STM-1 (OC3)	155,520 Mb/s
STM-4 (OC12)	622,080 Mb/s
STM-16 (OC48)	2488,320 Mb/s
STM-64 (OC192)	9953,280 Mb/s
STM-128 (OC384)	20 Gb/s
STM-256 (OC768)	40 Gb/s

RTC Réseau Téléphonique Commuté

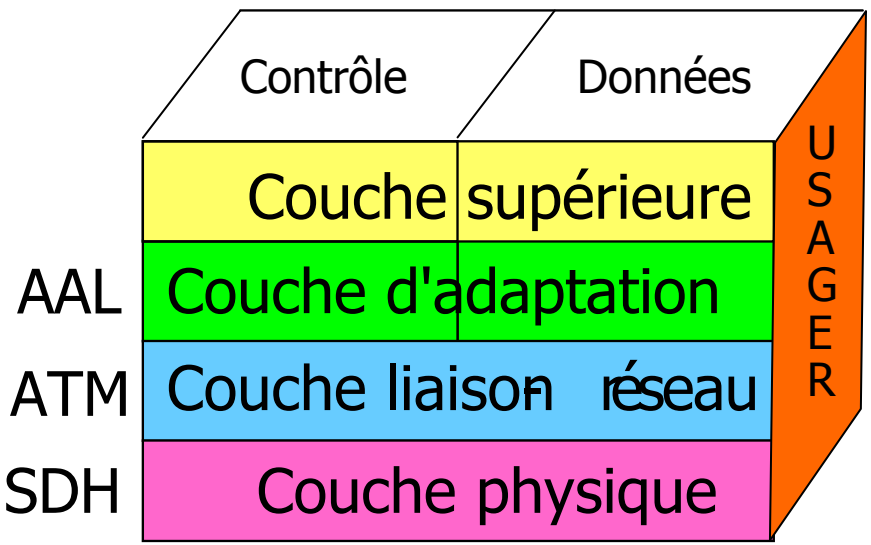
- **POTS** 'Plain Old Telephone System'.
- **Réseau filaire** de transport de la voix mais aussi ouverture au transport de données numériques (informatique, fax, ...).
- **Interface de l'usager**
  - Canaux de 300 à 3400 Hertz de bande passante.
  - Protocole de signalisation dans la bande : Émission d'appel décrochage de combiné), Enregistrement du numéro demandé, Surveillance, Libération du circuit.
- **Fonctionnement interne**
  - Multiplexage de voies MIC 64 Kb/s
  - Réseau de signalisation : Système de signalisation no 7
  - Très grand nombre de problèmes à résoudre : Tolérance aux pannes, Équilibrage de charge, Acheminement international.

Architectures de réseaux sans fils

- **Réseaux personnels sans fils (WPAN)**
  - Bluetooth : IEEE 802.15.1.
- **Réseaux locaux sans fils (WLAN)**
  - WiFi : IEEE 802.11.
- **Réseaux métropolitains sans fils (WMAN)**
  - Wimax : IEEE 802.16.
- **Réseaux étendus sans fils (WWAN)**
  - GSM : Global System for Mobile.
  - GPRS : General Packet Radio Service.
  - UMTS : Universal Mobile Telecommunication System.

Réseaux ATM
'Asynchronous Transfer Mode'

- **Objectif :** Utilisation de la commutation de paquet pour construire un réseau à intégration de services à haut débit.
  - RNIS-LB large bande B-ISDNBroadband Integrated Service Data Network
- **Hiérarchie des protocoles**



Réseaux Locaux
'Local Area Networks'

- **Objectif :** Définir des moyens de communication d'entreprise **à débit élevé** (10Mb/s à 10 Giga bits/s) **sur des distances courtes (kilomètres).**
  - Le niveau liaison dans les réseaux locaux de type IEEE802 ou ISO 8802 définit le tramage et résout le problème d'accès au médium ("MAC Medium Access Control"). Il définit le codage de niveau physique.

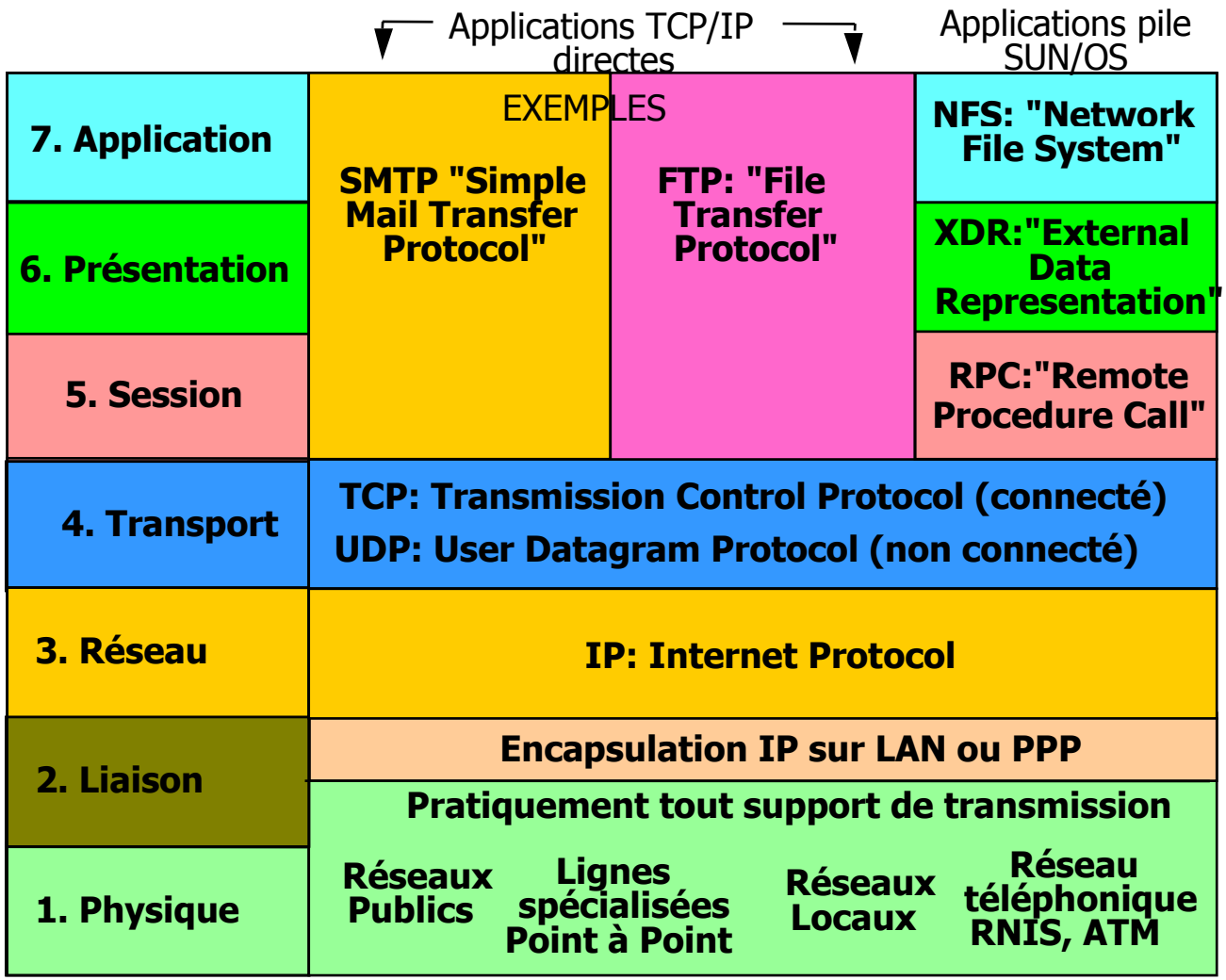
<table>
<tr><td rowspan="2">LIAISON</td><td>Point à point 802 - 2</td></tr>
<tr><td>Accès au médium 802.X (3->N)</td></tr>
<tr><td>PHYSIQUE</td><td>802.X (3->N)</td></tr>
</table>

- **Exemples de quatre architecture de réseaux locaux**
  - IEEE 802-3 Ethernet : Réseau à compétition sur bus filaire.
  - IEEE 802.11 Wifi : Réseau à compétition sur voie radio.

Architectures complètes : Internet

- **Héritier du réseau ARPANET**
  - Construction du réseau ARPANET à partir de 1967 sous forme de contrats entre l'ARPA (ministère de la défense des USA) et des organismes industriels et universitaires.
  - Développement de protocoles de transmission (couches basses) aujourd'hui abandonnés puis développement d'applications : beaucoup sont encore utilisées (transfert de fichiers FTP, accès distant TELNET,...).
- **INTERNET**
  - Création suite à la séparation d'ARPANET en 1983 entre un réseau militaire et un réseau civil.
  - Architecture à 7 couches en évolution constante pour tenir compte des progrés technologiques et recherche.

Organisation de la pile Internet



Conclusion

Conclusion : Notions Générales

- **Domaine des réseaux informatiques** : un axe essentiel du développement des techniques numériques (données, son, images).
- **Domaine vaste et complexe** qui comporte aussi bien des techniques de télécommunications (plutôt matériels) que des aspects de plus en plus sémantiques (logiciels) des interactions entre machines.
- **Grande hétérogénéité** des propositions malgré des efforts de normalisation et d'homogénéisation multiples.
- **Evolutivité** des concepts et des techniques très importante.
- **Importance considérable** des perspectives de développement industriel.

Second Chapitre

Niveau Physique

Plan du cours physique

- **Transmission sur un canal**
  - **Transmission en bande limitée**
  - **Transmission en présence de bruit**
  - **Détection et correction des erreurs**
  - **Représentation des signaux**
- **Technologie du niveau physique**
  - **Les contrôleurs**
  - **Les interfaces standards**
  - **Les modems**
  - **Les voies de communication**
  - **Les réseaux au niveau physique**
    - **Réseau téléphonique commuté**
    - **Réseaux PDH et SDH**

Rappel des objectifs du niveau physique

- **Définition :**
  - **Transmission effective** des informations binaires sur une voie physique en s'adaptant aux contraintes du support physique utilisé.
- **Problèmes à résoudre**
  - **Problèmes de synchronisation** : délimitation des informations significatives.
  - **Problèmes de modulation** : représentation des bits (électronique ou optique).
  - **Problèmes mécanique** : réalisation des connecteurs (connectique).

Niveau Physique

Première partie

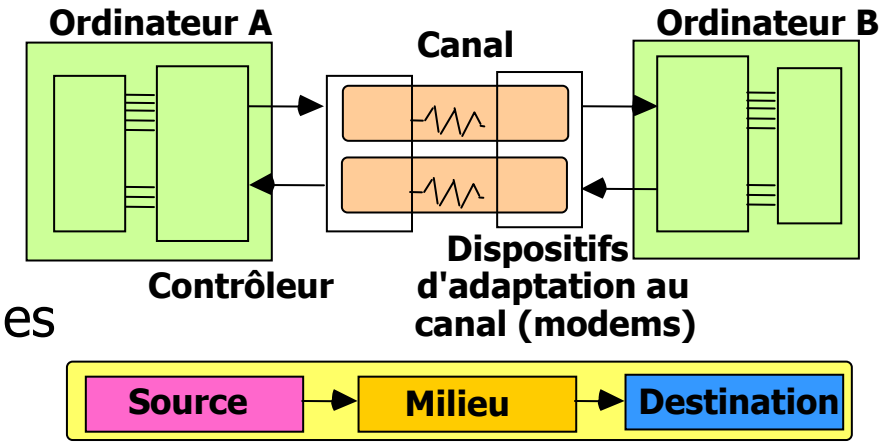
Transmission du signal

Introduction : canal de transmission

- **Disposer d'un support**

Physique qui véhicule les signaux électromagnétiques:

  - fils métalliques => signaux électriques
  - Atmosphère => ondes radio
  - fibre optique => lumière



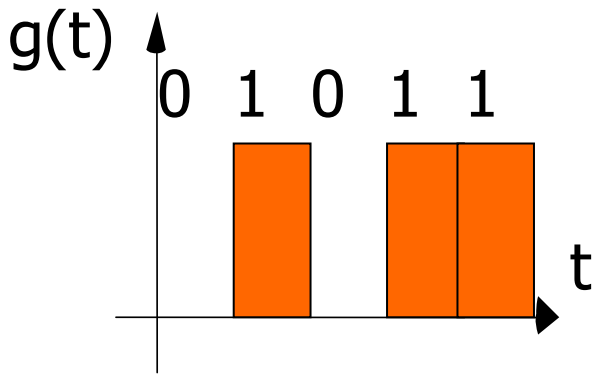
- **Canal de transmission :** une source (dispositif d'adaptation en émission), un médium (un milieu de transmission) et une destination (dispositif d'adaptation en réception).
- **Type de canal de transmission étudié : unidirectionnel** (ou simplexe).

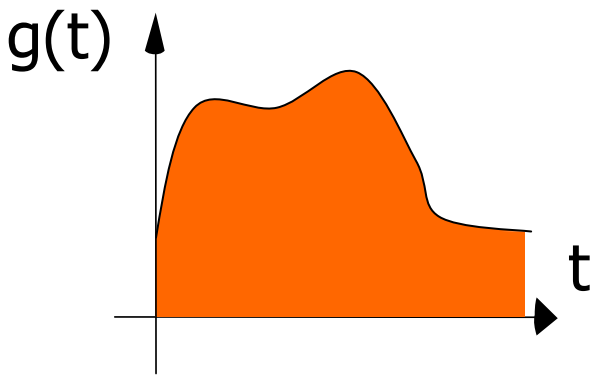
Problème principal du canal : le débit binaire

- **Quel débit d'information peut-être transmis** par un canal de transmission en fonction des caractéristiques de ce canal ?
  - **La bande passante** : bande des fréquences qui sont transmises par le canal.
  - **La déformation du signal** : distorsions apportées par les imperfections de la transmission.
  - **Le bruit** : influences externes provoquées dans le canal par le milieu extérieur.

Etude 1 : Canal sans bruit en bande limitée

- **La bande de fréquence est limitée à une valeur B**.
- **On ne s'intéresse pas au problème des bruits additifs.**
- **La source code** les données à émettre (les bits) par **une fonction g(t)** du temps : une représentation dans le domaine temporel d'un signal numérique ou analogique/continu.





- **Outil de cette étude :** l'analyse de Fourier.
- **Objectif** : Introduire l'importance de la disponibilité d'une large bande passante => passer dans le domaine fréquentiel.

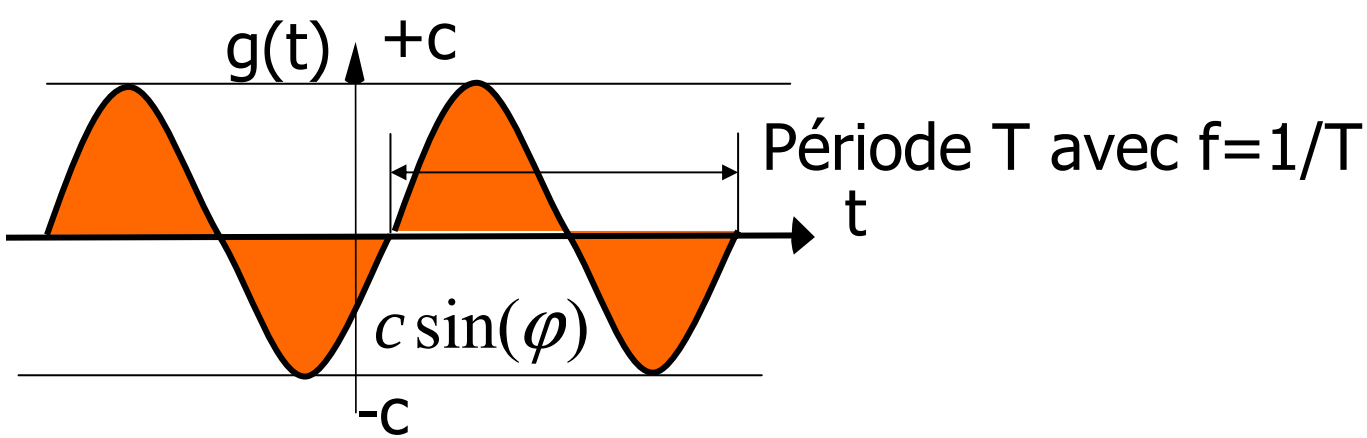
Fonctions sinusoides

- On décompose un signal selon un somme de fonctions sinusoïdes.
- Le signal analogique élémentaire est le sinus (ou le cosinus).

$$g(t) = c\sin(2\pi ft + \varphi)$$

- Signal **périodique** caractérisé par trois paramètres:

  amplitude c, fréquence f, phase φ



a) Cas ou la fonction g(t) est périodique

- **Correspond à une présentation dans un cas très particulier**: g est périodique (par exemple une horloge).
- **g(t) peut-être représentée** comme une somme infinie de fonctions sinus ou cosinus.

$$g(t) = c_0 + \sum_{n=1}^{+\infty} c_n \cos(2\pi nft - \varphi_n)$$

  - **f la fréquence** du signal périodique f = 1/T ou T est la période.
  - **Chaque terme** en cosinus est caractérisé par :

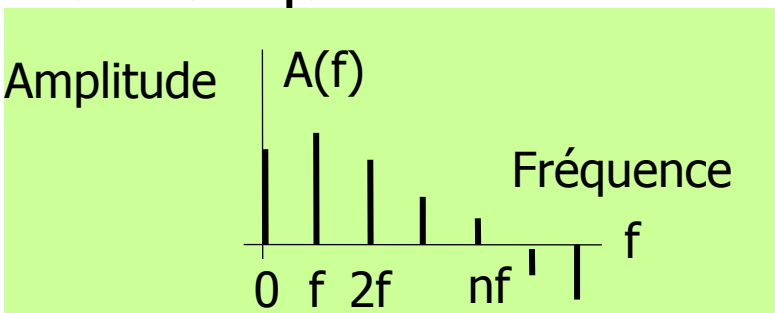
    . nf est une **harmonique** du signal.

    . une composante **d'amplitude:** c_n

    . une composante **de phase**: φ_n

Représentation spectrale

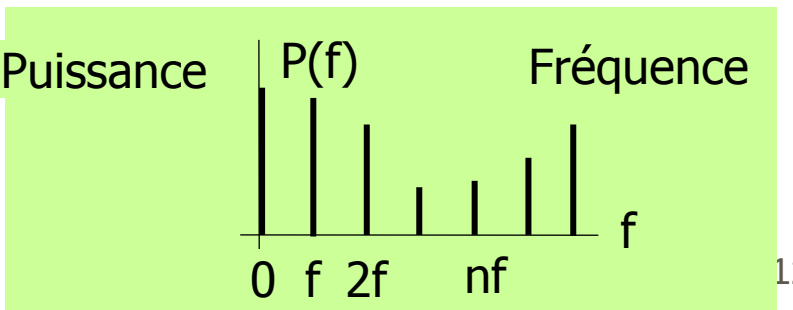
- **Spectre d'amplitude** : Représentation des amplitudes c_n en fonction des fréquences.
- **Fonction périodique=> spectre de raies** : une raie est associée à chaque harmonique.



- **Spectre de puissance** :Représentation des puissances contenues dans les différentes harmoniques.

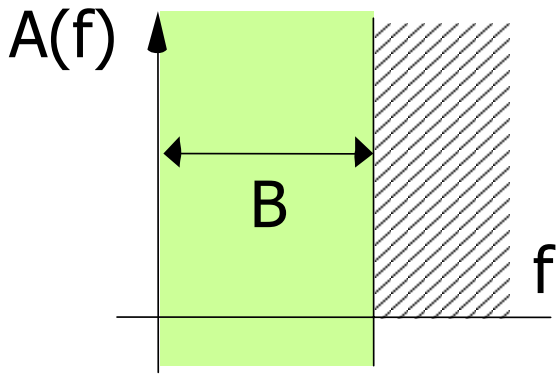
Puissance moyenne d'un signal

$$P = \frac{1}{T}\int_0^T g(t)^2\,dt = \sum_{n=0}^{+\infty} c_n^2$$

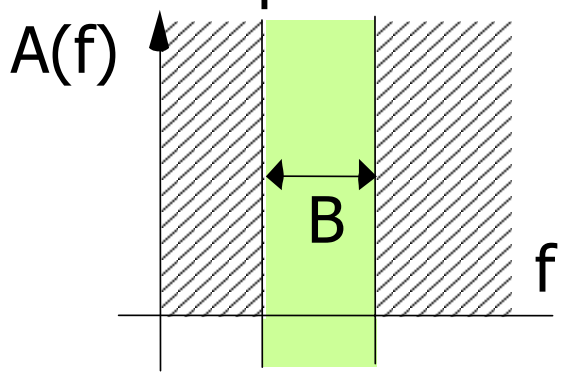


Première application de cette représentation

- **Lorsque l'on transmet un signal on le déforme** de manière différente selon les fréquences.
- **Déformation fondamentale** : on ne transmet jamais toutes les fréquences => **Les fréquences élevées disparaissent.**
- **Un canal se comporte comme un filtre.**
- **Exemple :** Bande passante réseau téléphonique commuté 300-3400 Hz

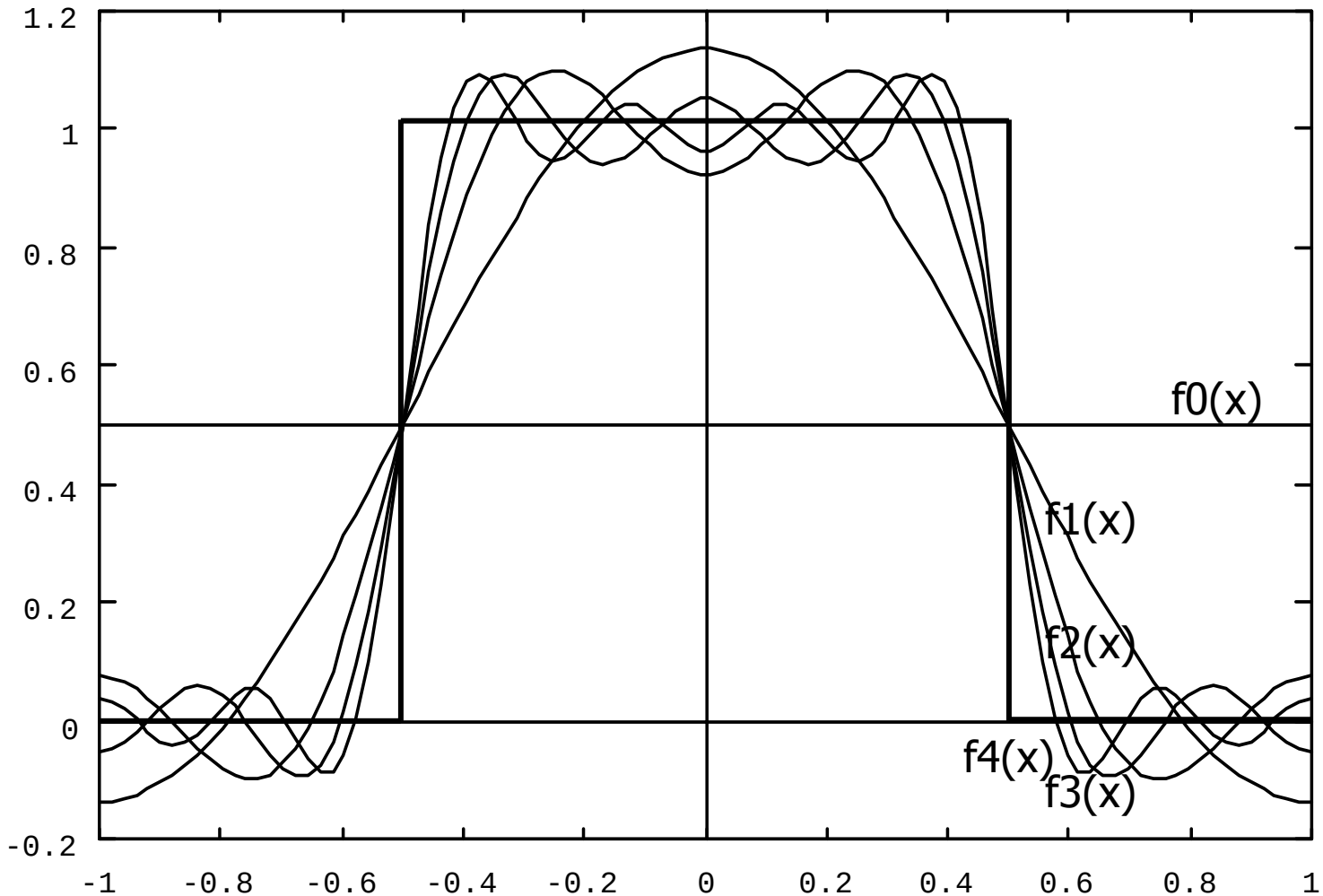


Filtre passe-bas



Filtre passe-bande

Distorsion due à la suppression des fréquences élevées sur une horloge



b) Cas ou la fonction g(t) est non périodique

- **Intégrale de Fourier**
- Un signal **non périodique** peut être mis sous la forme d'une intégrale de fonction sinusoïdale:

$$g(t) = \frac{1}{\pi} \int_0^{+\infty} S(\omega) \cos(\omega t - \varphi(\omega)) d\omega$$

- **Spectre continu** : Pour toutes les fréquences f (avec $\omega = 2\pi f$ la pulsation) on a :
  - une amplitude $S(\omega)$
  - une phase $\varphi(\omega)$

Notion de fonction de transfert

- **Canal par nature imparfait** => chaque composante est déformée de façon différente selon la fréquence.
- **Fonction de transfert du canal A(ω), B(ω) .**
  - **Atténuation en amplitude : A(ω)** le coefficient multiplicatif qui caractérise l'atténuation en fonction de la fréquence (résistance, dispersion d'onde).
  - **Retard de phase : B(ω)** le coefficient additif caractérisant le retard en fonction de la fréquence.
- **Modifications apportées au signal:** Si le signal g(t) est émis, le signal reçu est alors r(t):

$$g(t) = \frac{1}{\pi} \int_0^{+\infty} S(\omega) \cos(\omega t - \varphi(\omega)) d\omega$$

$$r(t) = \frac{1}{\pi} \int_0^{+\infty} A(\omega) S(\omega) \cos(\omega t - \varphi(\omega) + B(\omega)) d\omega$$

Mesure de l'atténuation: le décibel

- **L'atténuation** (affaiblissement) **est une perte de puissance** qui s'exprime en décibels:

  $S=10*\log_{10}(PS/PE)$ (valeur négative).

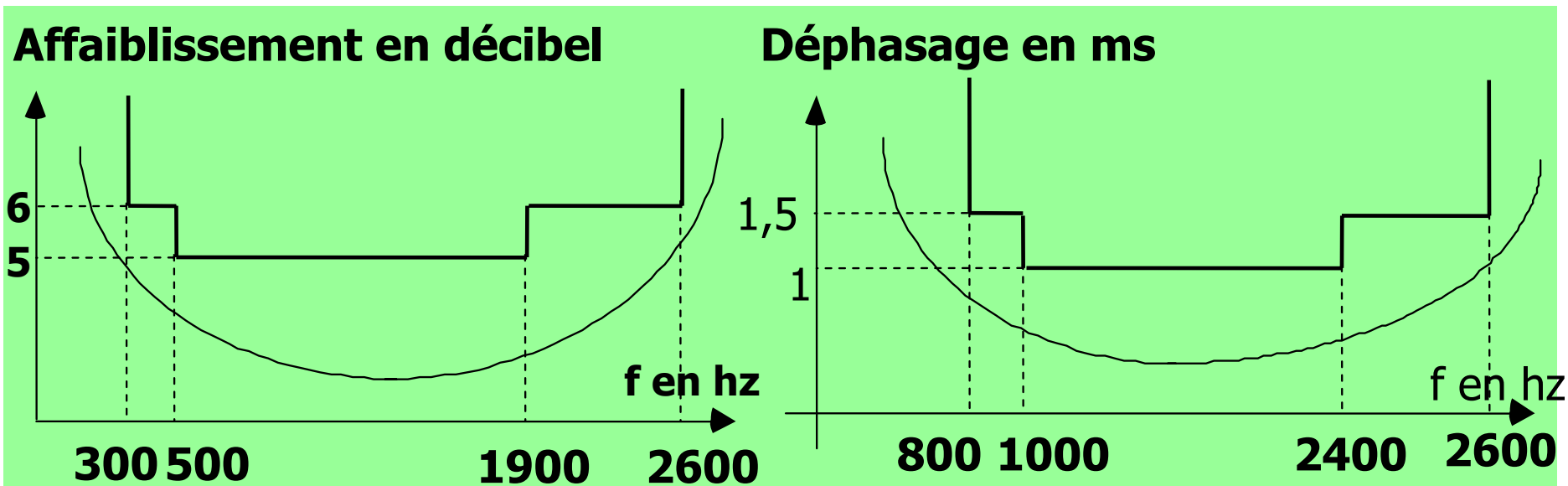
- **Le décibel est la représentation en logarithmes** d'un rapport de puissances (meilleure échelle, et les pertes en cascade s'exprime par des additions).
  - On peut aussi exprimer les pertes sur des tensions
  - $N=20*\log_{10}(TS/TE)$ puisque $P=U^2/R$.
- **Une puissance de signal** peut être donnée en décibel par watt ou par milliwatt dBmW:

  $SdBmW=10*\log_{10}P$ (ou P est en milliwatt).

- **Pour contrecarrer l'affaiblissement** on utilise des amplificateurs qui procurent un gain.
  - $G=10*\log_{10}(PS/PE)$ a une valeur positive.

Engagements contractuels sur les voies de communication

- **Notion de gabarit** : Performance les plus mauvaises offertes par la voie en termes d'atténuation et de déphasage: les gabarits.
- **Exemple:** Gabarit de réseau téléphonique commuté



Résultat d'échantillonnage de Shannon, Nyquist

- **B La largeur de bande d'un filtre en hertz** : on transmet un signal au travers de ce filtre.
- **R La rapidité de modulation en 'bauds'** : le nombre d'intervalles élémentaires par unité de temps (secondes) qui permettent l'échange d'un échantillon (d'un symbole).
- **V La valence d'un signal échantillonné :** le nombre de symboles différents qui peuvent être distingués par intervalle.
- **Q La quantité d'information par intervalle** en '**bits**'

$$Q = \log_2 V$$

- **C Le débit maximum d'informations en 'bits/seconde'**

$$C = R \log_2 V = 2B \log_2 V$$

Interprétation de Shannon, Nyquist

- **Théorème d'échantillonnage**
  - Un signal peut (théoriquement) être reconstruit à partir d'une fréquence d'échantillonnage égale à deux fois la largeur de bande (deux fois la fréquence maximale du signal pour un filtre passe-bas).
  - Soit encore : toutes les fréquences inférieures à la moitié de la fréquence d'échantillonnage peuvent être exactement restituées.
  - Exemple : Le son CD est échantillonné 44100 fois par seconde => on ne peut restituer correctement que les fréquences de 0 à 22050 Hz.

- **Résultat de débit maximum** pour un signal à support de largeur de bande B.
  - Le débit maximum théorique est atteint pour R = 2B (en échantillonnant 2B fois par unité de temps on atteint le débit maximum).
  - Dans une bande B pour augmenter le débit au il faut augmenter V la valence (le nombre de symboles par intervalles élémentaires).

Etude 2: Transmission en présence de bruits.

- **Objectif de la théorie de l'information de Shannon**
  - Modéliser un canal soumis à un bruit additif.
  - Déterminer la capacité maximum de transmission d'un canal

- **Origine des bruits**
  - **Thermiques :** Bruit de fond des résistances.
  - **Diaphoniques :** Influence permanente d'un conducteur sur un autre.
  - **Impulsionels :** Influences transitoires des impulsions
  - **Harmoniques :** Phénomènes de battements, de réflexions.

Entropie d'une source

- **Hypothèses :**
  - Une source émet des messages (ou symboles) pris dans un ensemble (un alphabet) donné fini (cas infini non traité ici)

$$X = x_1, x_2, x_3, ..., x_k, ..., x_M$$

  - Les messages émis sont aléatoires sinon il n'y a pas de communication d'information => Ensemble des **probabilités a priori**

$$p(x_1), p(x_2), p(x_3), ..., p(x_k), ..., p(x_M)$$

- **L'entropie d'une source H** : c'est la quantité d'information moyenne apportée par la source
  - Quantité d'information apportée par le message k : $-\log_2 p(x_k)$
  - Quantité moyenne : espérance mathématique pour tous les messages possibles :

$$H = -\sum_{k=1}^{M} p(x_k) \log_2 p(x_k)$$

Influence du bruit: probabilités a posteriori

- **Le récepteur reçoit des messages** qui appartiennent à un ensemble qui n'est pas nécessairement identique à celui émis par la source.

$$Y = y_1, y_2, y_3, \ldots, y_i, \ldots, y_N$$

- **Le bruit intervient pour modifier un message** émis xk en un message reçu yi selon une probabilité **a posteriori** (probabilité conditionnelle) : la probabilité que l'émetteur ait envoyé xk sachant que le récepteur a vu arriver yi

$$p(x_k / y_i)$$

Information mutuelle et capacité d'un canal

- **Information mutuelle de deux messages émis et reçus**
  - La quantité d'information apportée lorsqu'on reçoit yi alors que xk a été émis:

$$I(x_k, y_i) = \log_2\left(\frac{p(x_k / y_i)}{p(x_k)}\right)$$

  - Exemples: Si y_i et x_k sont indépendants $I(x_k, y_i)=0$

    Si $p(x_k/y_i)=1$ on retrouve $-\log_2(p(x_k))$
- **Information mutuelle moyenne** : source/destinataire

$$I(X,Y) = \sum_X \sum_Y p(x_k et\ y_i) I(x_k, y_i)$$

- **Capacité d'un canal** : La valeur maximum de l'information mutuelle moyenne sur toutes les distributions a priori.

$$C = \max_{p(x_k)} I(X,Y)$$

Résultats de Shannon

- **Premier résultat de Shannon : une source n'est caractérisée que par son entropie.**
  - On ne change rien sur l'information générée par la source en changeant de codage.
  - La seule mesure de l'information qui compte est l'entropie (son débit en bit/unité de temps).
- **Second résultat de Shannon : débit maximum C**
  - **Si H $\leq$C** il existe une codification des messages qui sur une période suffisamment longue permet de transmettre les messages avec une probabilité d'erreur résiduelle aussi faible que l'on veut.
  - **Si H>C** il n'existe pas de codification qui assure sur une période de durée arbitraire une transmission sans erreurs.

Interprétation de Shannon

- **Dans le premier cas : capacité du canal excédentaire**
  - Sur une longue période cet excédent est **important.**
  - Il permet d'ajouter des **redondances** (sans changer l'entropie de la source) => On peut générer des codes correcteurs d'erreur aussi efficaces que l'on veut.
  - On abaisse ainsi le taux d'erreur résiduel **arbitrairement.**
- **Dans le second cas : capacité du canal déficitaire**
  - Sur une période courte on peut transmettre correctement mais ensuite on aura des erreurs non corrigées.
  - Avant ce résultat on pouvait penser que le bruit introduisait une borne **infranchissable** sur le taux d'erreur résiduel.
  - Shannon montre que le **bruit intervient sur le débit du canal et non sur sa précision.**

Obtention d'un débit élevé

- Pour **augmenter** le débit d'un canal à taux d'erreur donné on peut:
  - **Augmenter la complexité de codage** des équipements terminaux pour se placer au plus près de la capacité maximale (des limites du théorème).
  - **Augmenter la capacité du canal** (bande passante, puissance) en conservant des techniques de codage simples.
  - **Jouer sur les deux aspects.**

Résultat particulier de Shannon

- **Canal de bande passante limitée** : B.
- **Puissance moyenne du signal** : S
- **Puissance moyenne d'un bruit additif** : N.

$$C = B \log_2 (1 + \frac{S}{N})$$

  - **Bruit blanc**: énergie répartie de façon uniforme sur le spectre
  - **Gaussien**: l'apparition d'un bruit suit une loi de Gauss.
  - **Exemple**: B= 3100 Hz 10 $\log_{10}$ S/N = 20 db

    S/N = 100 C = 3100 * 6,6 = 20600 b/s
- **Remarque**: Dans ce cas Shannon montre que le nombre de niveaux max V qui peuvent être discriminés est donné par:
  - 2B $\log_2$ V = B $\log_2$ (1 + S/N)

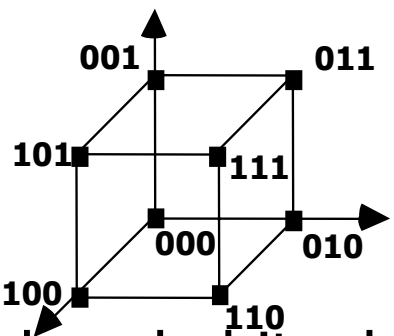
$$V = \sqrt{1 + \frac{S}{N}}$$

Détection et correction des erreurs

Introduction: Généralités concernant les codes de blocs

- **Existence de bruits** qui perturbent les transmissions.
  - **Suite binaire émise M** : un n uple binaire (un bloc).
  - **Suite binaire reçue M'=> M'≠M** (des bits sont modifiés).
- **Distance entre deux messages**

$$d(M,M') = \sum_i \left| M(i) - M'(i) \right|$$



  - **Distance de Hamming** : c'est le nombre de bits dans M et M' qui sont différents.
  - **Géométriquement :** c'est une distance dans l'espace à n dimensions entre les points M, M'.

Solution 1: Détection des erreurs et retransmission

- **Une idée de redondance temporelle (dans le temps).**
- **A) Détection d'erreur par adjonction de redondances** : à tous les messages x_i transmis on ajoute $f(x_i)$.

$$\{ x_i \} \dashrightarrow \{ y_i = (x_i \ , f(x_i)) \}$$

- **B) Vérification de f par le récepteur** : un message reçu y_j

$y_j = (x_j , z_j)$ est correct si $z_j = f(x_j)$.

- **C) Retransmission si erreur.**
- **Notion de code :** A partir de l'ensemble des symboles à émettre x_i on créé un ensemble y_i de messages émis: **un code.**
- **Notion de distance d'un code** : c'est la plus petite distance entre deux symboles différents (distance des messages les plus proches $\inf_{j,k}[d(y_j, y_k)]$
  - **Un code de distance D est détecteur de D-1 erreurs.**
  - **Des erreurs de distance supérieure à D-1 peuvent ne pas être détectées.**

Solution 2: Correction des erreurs (codes correcteurs)

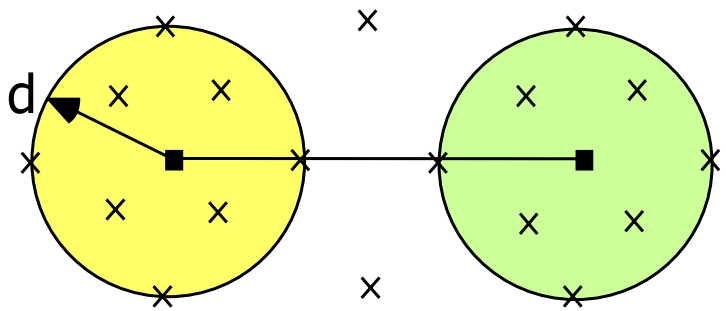
- **Une idée de redondance spatiale (masquage d'erreur).**
- **A) Comme précédemment adjonction de redondances** : à tous les messages x_i transmis on ajoute $f(x_i)$.
- **B) Vérification de f par le récepteur** : si un message reçu y_j appartient au code on l'accepte.
- **C) Correction immédiate** quand le message reçu est erroné

  Message faux $y_j = (x_j , z_j)$ avec $z_j \neq f(x_j)$.
  - On fait l'hypothèse que **le bruit altère probablement un petit nombre de bits.**
  - On **corrige un message erroné par le mot du code correct le plus proche** du message erroné reçu.

$$(x_i, f(x_i)) = \inf_k (d((x_k, f(x_k)),(x_j, z_j))$$

Codes correcteurs: représentation géométrique

- On trace des sphères centrées sur chaque mot d'un code (les messages corrects) de rayon d (distance de Hamming).



Messages corrects: carrés noirs

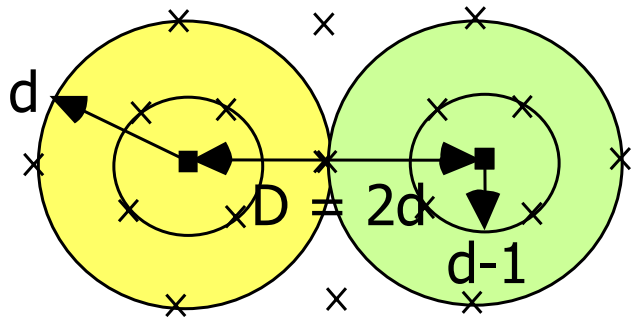
Messages incorrects : croix.

- **Les erreurs de faible poids** (portant sur un petit nombre de bits) ont une probabilité d'apparition forte par rapport aux erreurs de fort poids => **correction des messages** dans une sphère **par le message au centre de la sphère.**
- **Possibilité de faire des erreurs de correction** si le poids des erreurs est trop élevé.

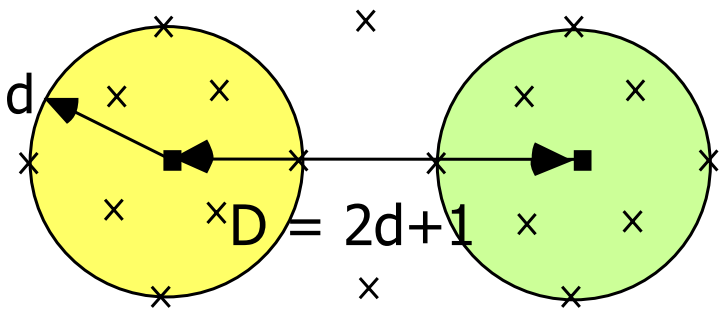
Codes correcteurs: Nombre d'erreurs corrigées

- **Code correcteur de distance D.**

- **Si D=2d :** correction des erreurs de poids **d-1.**



- **Si D=2d+1 :** correction des erreurs de poids **d.**



- **Problème de construction d'un code correcteur**: écarter les mots du code dans l'espace, le plus possible et de façon régulière (par le choix de la fonction de redondance f).

Paramètres d'un code

- a) Taux d'erreur brut
  - Qualité de la voie

$$t = \frac{\text{Nombre de messages faux}}{\text{Nombre de messages total}}$$

- b) Efficacité d'un code
  - En détection

$$e = \frac{\text{Nombre de messages reconnus faux}}{\text{Nombre de messages total}}$$

- c) Taux d'erreur résiduel
  - Erreurs non détectées ou non corrigées

$$q = \frac{\text{Nombre de messages finalement faux}}{\text{Nombre de messages total}}$$

- b) Rendement d'un code
  - Surcharge due au codage

$$r = \frac{\text{Nombre de bits utiles reçus}}{\text{Nombre total de bits transmis}}$$

Exemple 1 de codes : Codes linéaires en blocs

- **Mots du code (blocs):** ensemble de n-uples binaires (vecteurs de n bits) formant un espace vectoriel sur (0,1) pour les opérations ou exclusif $\oplus$, et logique . (notion de corps de Galois à deux éléments GF(2)).

$\oplus$	0	1
0	0	1
1	1	0

.	0	1
0	0	0
1	0	1

- **Notion d'espace vectoriel :** Si X_1 et X_2 appartiennent au code
  - $X_1 \oplus X_2$ appartient au code.
  - Quelque soit k scalaire $k.X_1$ appartient au code => ici 0 appartient au code.

Matrice de génération d'un code linéaire

- **Matrice** G rectangulaire qui fait correspondre à un message Y de k bits un mot du code X de n bits (G a k lignes, n colonnes).

$$X_{(1,n)} = Y_{(1,k)} \cdot G_{(k,n)}$$

- **G ajoute une redondance** de m= n-k bits.
- **Exemple:** le contrôle de parité (code linéaire le plus simple).

$$(x_1, \dots, x_k, x_{k+1}) = (x_1, \dots, x_k) \begin{bmatrix} 1 & & 0 & 1 \\ & \ddots & & \vdots \\ & & 1 & 1 \\ 0 & & 1 & 1 \end{bmatrix}$$

$$x_{k+1} = x_1 \oplus x_2 \dots \oplus x_k$$

- **Notion de code séparé** : Bits d'informations, puis bits de redondance (contrôle, parité).
  - G est de la forme $G = [I_{k,k} P_{k,n-k}]$ $X = (y_1, y_2, \dots, y_k, x_{k+1} \dots, x_n)$

Matrice de contrôle d'un code linéaire

- **V' l'espace orthogonal de V** : tout mot du code X dans V a **un produit scalaire nul** avec tout vecteur X' dans V'.
- **Tout message X' en erreur** (n'appartenant pas au code) **appartient à V'**.
- **Tout message X' (en erreur)** a un produit scalaire non nul avec au moins un vecteur V'
- **Soit H (n, n-k)** la matrice génératrice de l'orthogonal : la matrice formée au moyen des vecteurs d'une base de V'.
- **Syndrome d'erreur d'un message:** $S = X H$
  - Si **S = 0** le message est présumé **correct**
  - Si **S** différent de **0** le message est **erroné.**

Construction de la matrice de contrôle

- **Très facile** : pour les codes séparés
  - On vérifie que si $G = [\, I_{k,k} P_{k,n-k} \,]$
  - Alors $H = \begin{bmatrix} P_{k,n-k} \\ I_{n-k,n-k} \end{bmatrix}$
- **Exemple** du contrôle de parité: $G = \begin{bmatrix} 1 & 0 & 1 \\ & \ddots & 1 \\ 0 & & 1 \end{bmatrix}$

$H^T = [\, 1\ 1\ \ldots\ 1\ 1\ 1 \,]$

$S = (X_1, \ldots, X_{K+1})H$

$S = X_1 \oplus \ldots \oplus X_{K+1}$

- **Application** : codes de Hamming (mémoires)

Exemple 2 de codes : Codes polynomiaux

Principe de la génération de la redondance.

- A un message par exemple 1 1 0 1 1 1 on associe un polynôme -->

$$x^5 \oplus x^4 \oplus x^2 \oplus x \oplus 1$$

- On veut ajouter deux bits de redondance on multiplie par x2 -->

$$x^7 \oplus x^6 \oplus x^4 \oplus x^3 \oplus x^2$$

- On choisit un polynôme générateur de degré inférieur au degré du polynôme message:
- Exemple : $x^2 \oplus x \oplus 1$

Les bits de redondance ajoutés au message sont les coefficients du polynôme reste dans la division du polynôme message par le polynôme générateur.

$$\begin{array}{l|l} x^7 \oplus x^6 \oplus \quad x^4 \oplus x^3 \oplus x^2 & x^2 \oplus x \oplus 1 \\ \underline{x^7 \oplus x^6 \oplus x^5} & \\ \quad x^5 \oplus x^4 \oplus x^3 \oplus x^2 & \underline{\quad\quad\quad} \\ \quad \underline{x^5 \oplus x^4 \oplus x^3} & x^5 \oplus x^3 \oplus 1 \\ \quad\quad x^2 & \\ \quad\quad \underline{x^2 \oplus x \oplus 1} & \\ \quad\quad\quad x \oplus 1 & \end{array}$$

- Le message transmis est donc: 1 1 0 1 1 1 1 1

Codes polynomiaux: Présentation formelle

- **Message de k bits** : M
- **Polynôme associé au message M(x) de degré k-1**: On décale de m positions → $x^m M(x)$
- **G(x) le polynôme générateur** du code de degré m.
- **On effectue la division:** $x^m M(x) = G(x) Q(x) \oplus R(x)$
- **On obtient un reste R(x**) de degré m-1 au plus qui a la propriété suivante. $x^m M(x) \oplus R(x) = G(x) Q(x)$
- **L'ensemble des polynômes** $x^m M(x) \oplus R(x)$ forment **un code polynomial** (tout **mot du code donne un reste nul** dans la division par G(x))
- **Tout mot hors du code** (message erroné) **donne un reste non nul**.

Codes polynomiaux: Propriétés

- **Nombreuses propriétés** des codes polynomiaux: 2 exemples.
- **A) Un code polynomial détecte toute erreur simple.**
  - Une erreur simple est une erreur additive de la forme: $E(x)=x^i$
  - Pour que cette erreur soit indétectée, E(x) doit être divisible par G(x).
  - Si G(x) a plus d'un seul terme il ne peut être diviseur de E(x).
- **B) Un code polynomial qui génère m bits de redondance détecte toutes les rafales d'erreurs de longueur < m.**
  - Une rafale de longueur k est une erreur additive de la forme:
    $$E(x)=x^i\,(x^{k-1}\oplus x^{k-2}\oplus \ldots \oplus x^2\oplus x\oplus 1)$$
  - Pour que cette erreur soit indétectée, E(x) doit être divisible par G(x).
  - A) Si G(x) a un terme constant il ne peut pas avoir x^i en facteur.
  - B) Si k-1 est plus petit ou égal à m-1 le degré de
    $$(x^{k-1}\oplus x^{k-2}\oplus \ldots \oplus x^2\oplus x\oplus 1)$$
    est inférieur au degré de G(x) et G(x) ne peut donc le diviser.

Codes polynomiaux: Exemples de polynômes

- **CRC-12** : Définition d'un polynôme générant 12 bits de redondance (systèmes télécoms, caractères 6 bits).

$$G(x)=x^{12}\oplus x^{11}\oplus x^{3}\oplus x^{2}\oplus x\oplus 1$$

- **Avis V41 CCITT** : Définition d'un polynôme CRC-CCITT (protocoles de liaisons en point à point dérivés de HDLC).

$$G(x)= x^{16}\oplus x^{12}\oplus x^{5}\oplus 1$$

- **CRC-IEEE 802** : Définition d'un polynôme générant 32 bits de redondance pour les réseaux locaux: Ethernet, Wifi.

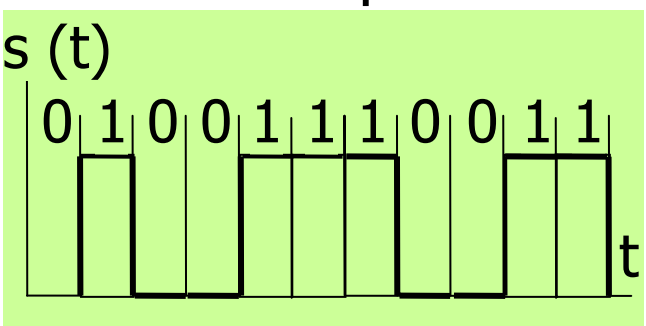
$$G(x)=x^{32}\oplus x^{26}\oplus x^{23}\oplus x^{22}\oplus x^{16}\oplus x^{12}\oplus x^{11}\oplus x^{10}\oplus x^{8}\oplus x^{7}\oplus x^{5}\oplus x^{4}\oplus x^{2}\oplus x\oplus 1$$

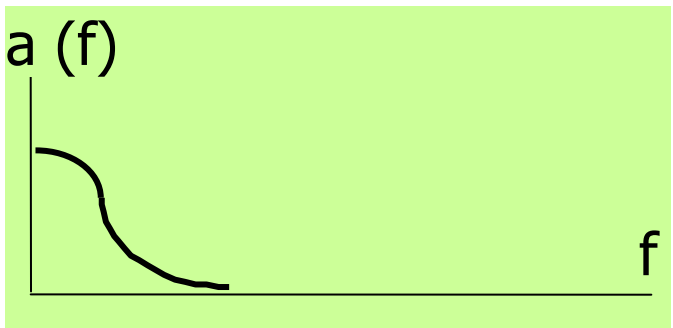
Niveau Physique

Représentation des signaux : Synchronisation, Modulation

Introduction : Transmission en bande de base

- On s'intéresse à une **transmission série.**
- Le signal est dans **sa représentation de base** (tel qu'il est généré par un système informatique).
- **Exemple type:** le codage NRZ-L en représentation temporelle et fréquentielle.

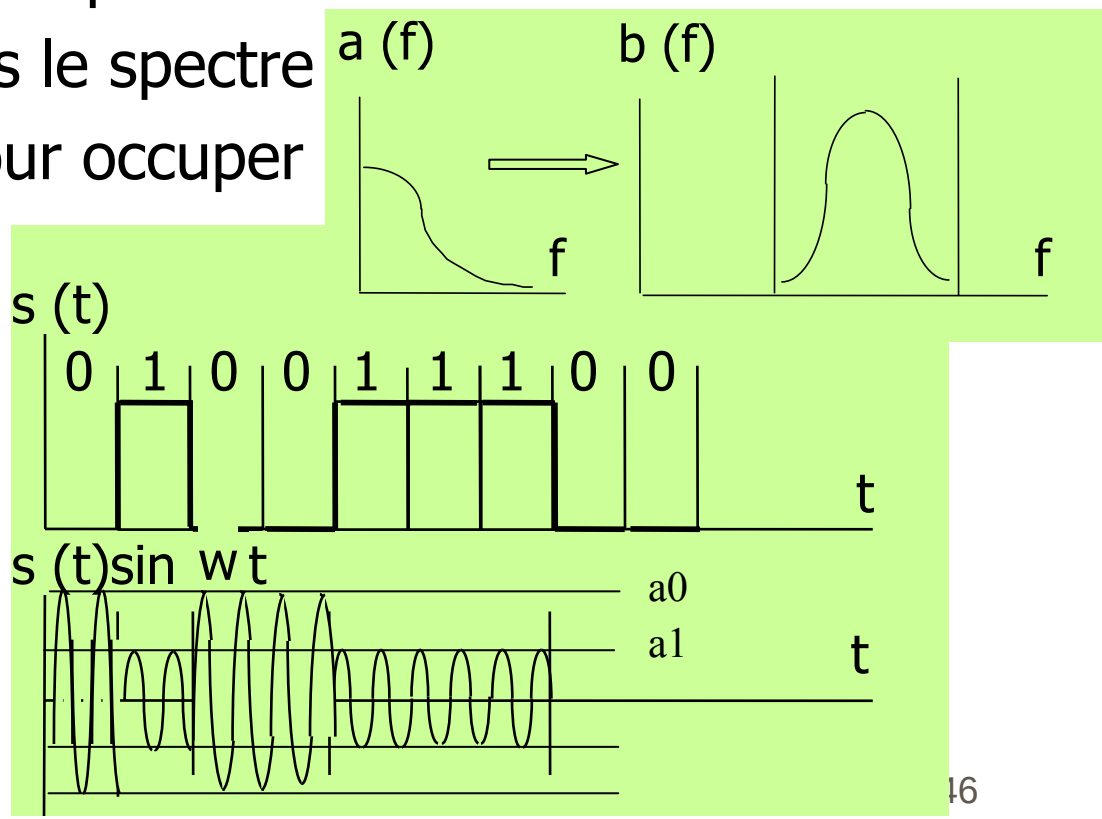




- **Le signal occupe une bande de fréquence** 'naturelle' en fonction de sa représentation et il est envoyé directement sous cette forme dans un médium de communication.

Transmission en modulation

- **Modulation (d'onde porteuse)** Le signal s(t) est représenté au moyen d'une onde porteuse qui modifie le codage de base pour s'adapter à un canal de transmission.
  - Positionnement dans le spectre
  - Remise en forme pour occuper la bande disponible.



- **Exemple type:**

la modulation d'amplitude

Introduction : Problèmes de synchronisation

- **Le récepteur d'un signal doit connaître la position** de chaque bit pour échantillonner correctement les valeurs.
- **Nombreuses difficultés:**
  - **Echantillonnage des bits** en présence de multiples aléas de fonctionnement (gigue, dérive entre les horloges, délais de propagation,...).
  - **Détermination du début des suites binaires** significatives (notion de trames de bit).

Terminologie concernant les signaux (1)

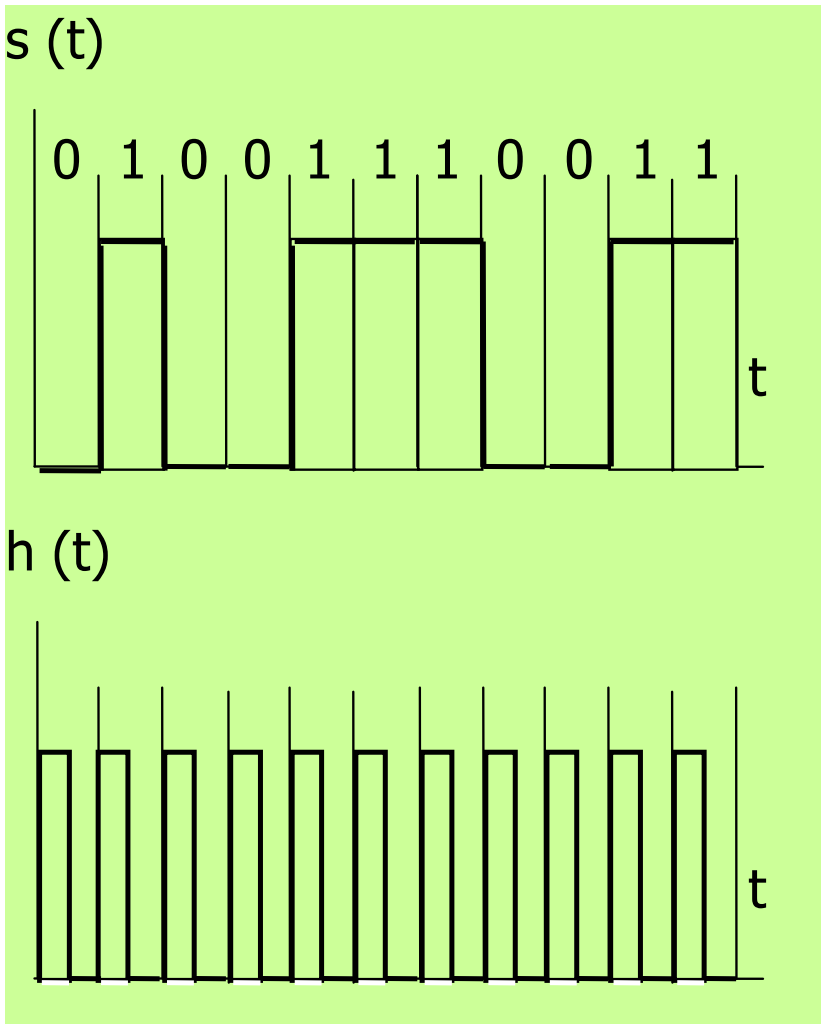
- **Signaux Isochrones** (égaux)
  - **Il existe un écart fixe** entre deux signaux successifs.
  - **Exemple son** : le réseau téléphonique échantillons de 8 bits isochrones selon un intervalle de 125 microseconde.
  - **Exemple image** : codage vidéo à 50 images par seconde images espacées de 40 millisecondes.
  - **L'intervalle constant** doit être reproduit fidèlement chez le récepteur sous peine de perte de qualité de la restitution.
- **Signaux anisochrones** (non égaux)
  - Il n'y a **pas d'intervalle fixe** entre les signaux.
  - Il peut être très important de **restituer l'espacement variable** de l'émission lors de la délivrance au récepteur (contraintes temps réel).

Terminologie concernant les signaux (2)

- **Signaux synchrones** (ensembles)
  - Des signaux synchrones sont à la même cadence (sont rythmés par la **même horloge**).
- **Signaux asynchrones** (antonyme du précédent)
  - Des signaux asynchrones n'apparaissent pas selon un rythme constant défini par une horloge mais apparaissent **aléatoirement**.
- **Signaux plésiochrones** (voisins)
  - Des signaux plésiochrones sont rythmés par des horloges dont les **fréquences** sont **peu différentes** (plésio = voisin).

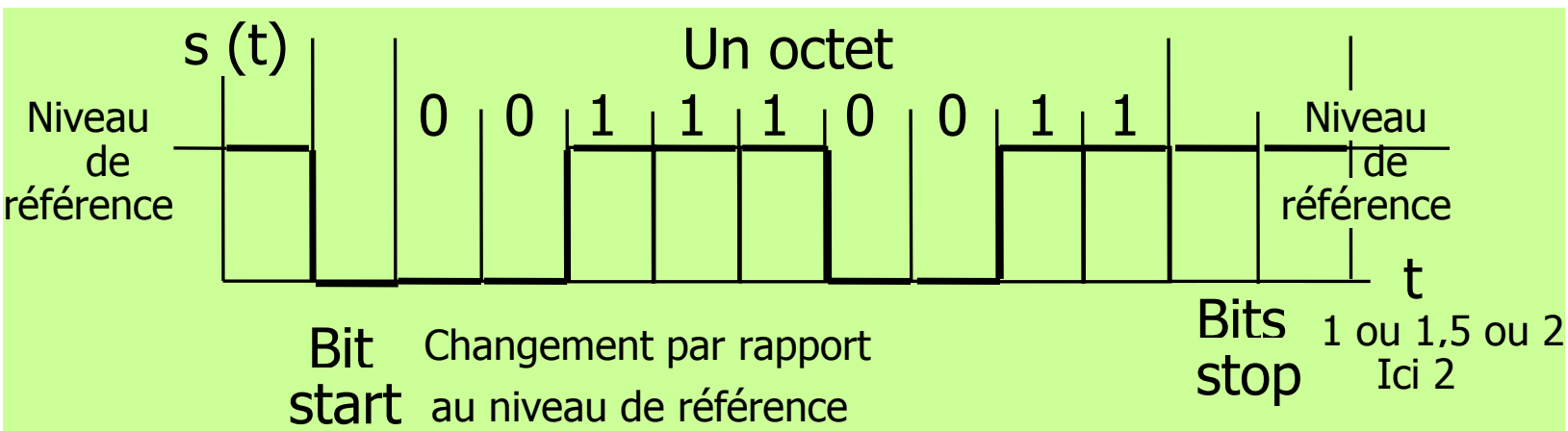
Transmission synchrone

- **Horloge transmise sur un canal** spécial entre émetteur et destinataire.
  - **Les deux sites utilisent la même base de temps**
    => Génération et échantillonnage selon le même rythme.
  - **Nécessite une bonne qualité d'acheminement de l'horloge**
    => Solution assez coûteuse en bande passante nécessitant un canal spécial pour l'horloge.



Transmission asynchrone ‘Start Stop’

- **Pas d'horloge commune =>**asynchronisme sur l'instant de commencement d'une suite binaire.
- **On ne transmet que des suites binaires courtes isochrones** (en fait des octets).
- **On table sur une dérive relative** entre l'horloge d'émission et de réception qui permet de ne pas perdre la synchro bit.
- **Adapté à des débits faibles.**



Transmission plésiochrone

- **Les horloges émetteur et destinataire** ont des fréquences **différentes mais voisines** (dérive max tolérée liée au débit).
- **Le récepteur synchronise une horloge de réception sur l'horloge d'émission.** Exemple: usage d'un préambule qui est un signal d'horloge présent dans tout message.
- **Le message est ensuite échantillonné correctement** par le récepteur mais **il est limité en taille** selon les dérives relative des horloges tolérées.
- **Adaptation aux vitesses relatives** des horloges:

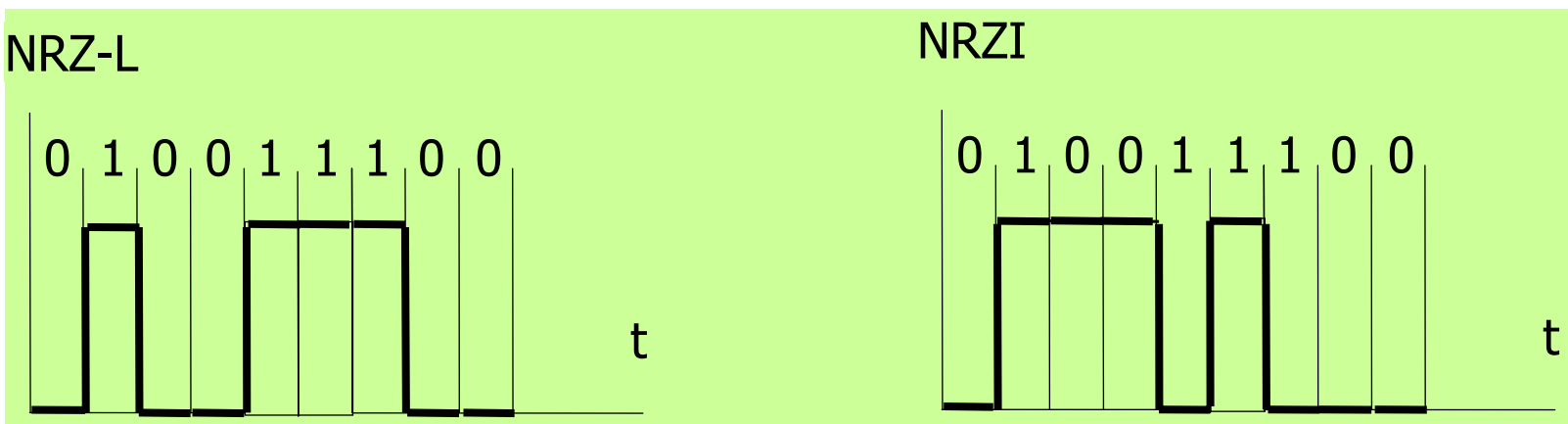
  => techniques de **justification.**
- **Détermination de la position** des informations significatives:

  => techniques de **pointeurs.**
- **Détails : PDH et SDH.**

Techniques de codage
Signaux bande de base en amplitude

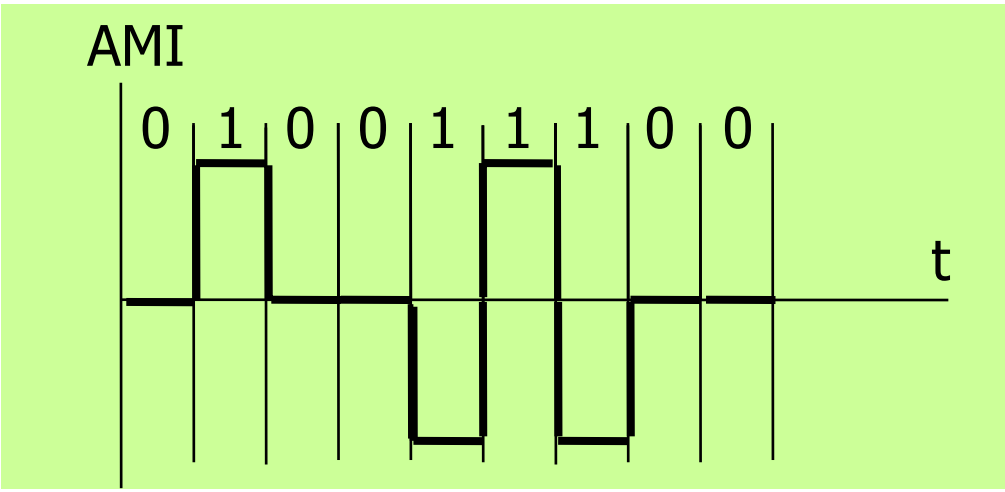
Codages NRZ "Non Retour à Zéro"

- **Le niveau est constant sur un intervalle (il n'y a pas de transition de retour à zéro)**
  - **NRZ-L** ("Level") On utilise deux niveaux pour coder le 0 et le 1. **Exemple: V24**.
  - **NRZI** ("Inverted") Codage différentiel des 1. Chaque nouveau 1 entraîne un changement de polarité par rapport au précédent 1 alors que le 0 n'entraîne pas de modification. **Exemple: Ethernet 100 sur fibre.**



Signaux bande de base en amplitude
Les codages AMI et pseudo ternaire

- **Trois niveaux : 0 , +V, -V volts (code bipolaire).**
- **Bipolaire AMI:** "Alternate Mark Inversion" Un 0 est représenté par le signal 0 volts et un 1 est représenté alternativement par +V et -V volts : Exemple: RNIS Bus d'abonné.



- **Pseudo ternaire :** On inverse le 0 et le 1 dans le schéma précédent.

Techniques d'embrouillage ("Scrambling")

- **Problème des codages NRZ ou AMI** : Absence de transitions dans de longues séquences de symboles identiques.
- **Solution =>** Utiliser un codage préliminaire qui permet de forcer l'apparition de transitions dans ces séquences.

Embrouillage par code polynomial

- **Technique employée dans un embrouilleur additif :** Ou exclusif avec une séquence pseudo aléatoire (0 et 1 aléatoires).
- **La séquence pseudo-aléatoire** peut-être obtenue comme le quotient d'une division répétée indéfiniment par un polynôme générateur (circuit LFSR Linear Feedback Shift Register).
- **Opération ou exclusif avec la même séquence** à l'arrivée.
- **Exemple: Réseau ATM** => Polynôme $X^{31} \oplus X^{28} \oplus 1$

Codages HDB3 et B8ZS

- **Codage** pour résoudre le problème des suites de 0 dans le code AMI.
- **On remplace des suites de 0 de taille fixe** (soit 4 soit 8 bits) sans transitions par des suites de longueur identique possédant des transitions.

B8 ZS "Bipolar with eight zero substitution"

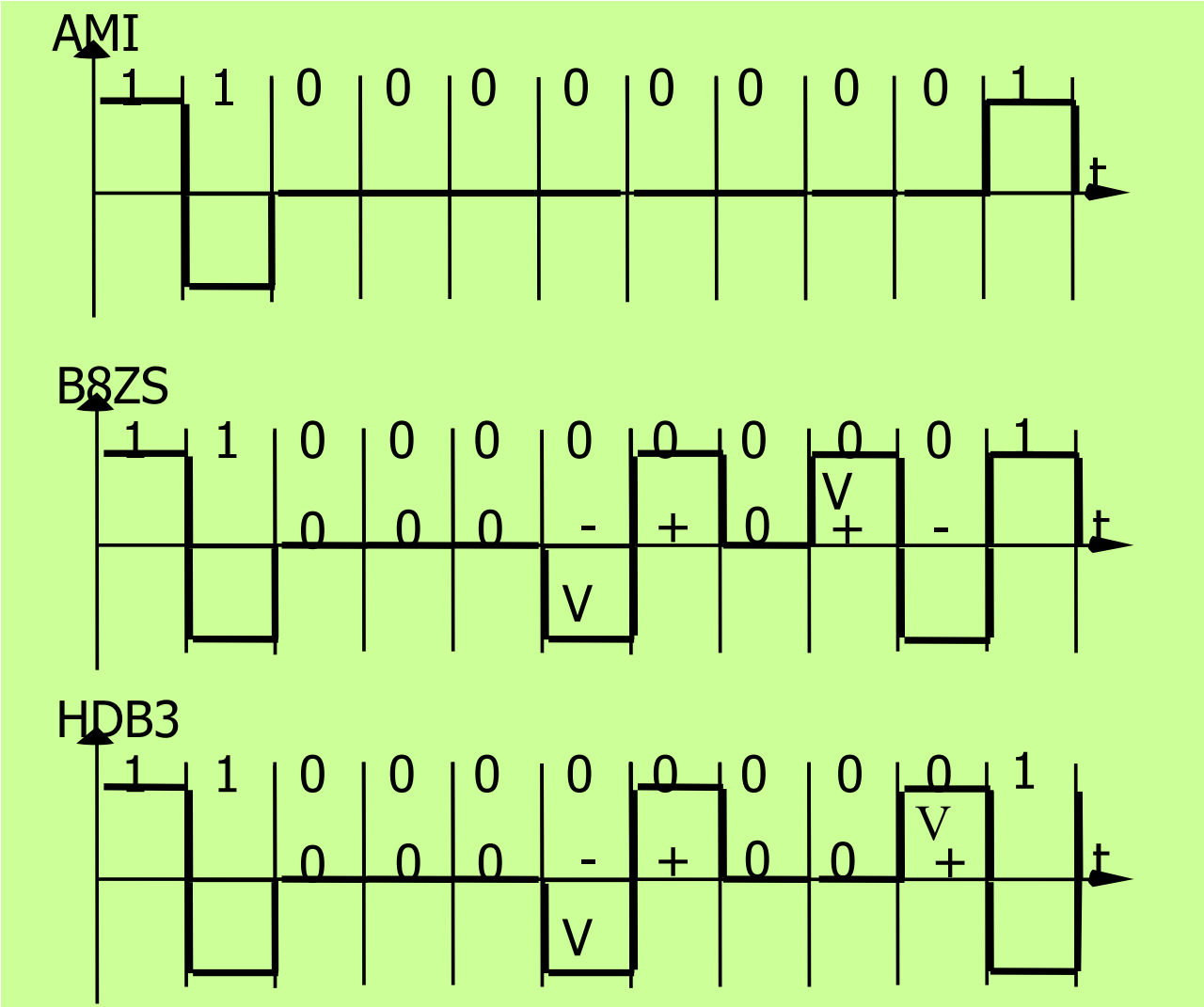
- Si la dernière excursion était positive (+) : 8 bits à 0 => 000+-0-+
- Si la dernière excursion était négative (-) : 8 bits à 0 => 000-+0+-
- **Utilisation: Réseau téléphonique RNIS américain.**

HDB3 "High Density Bipolar - three zero"

- 4 bits consécutifs à 0 sont remplacés selon la polarité de la dernière excursion et le nombre d'excursions depuis la dernière violation.
- **Utilisation : RNIS européen.**

Polarité précédente	Nombre d'excursions impair	Nombre d'excursions pair
-	000 -	+00+
+	000+	-00-

Exemples B8ZS et HDB3

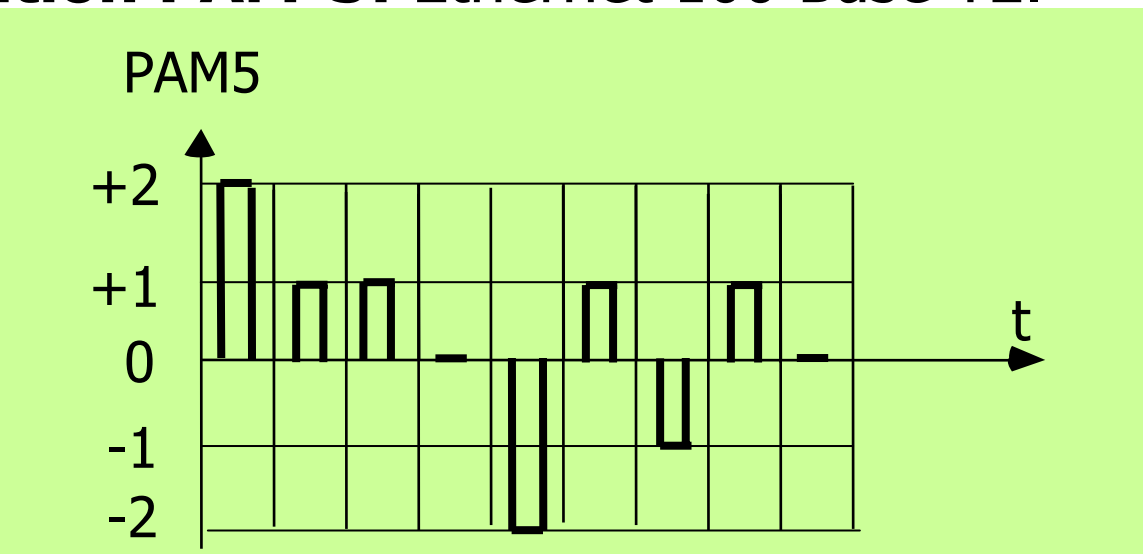


Codages nB/mB

- **On représente n bits à transmettre par m bits** (n<m).
- **On peut ainsi choisir des configurations** m bits qui présentent un bon équilibre de 0 et de 1 (un nombre suffisant de transitions).
- **Exemple: Ethernet 100 Mb/s** (et FDDI) Codage 4B/5B : 4 bits usagers sont transmis par des groupes de 5 bits.
  - On choisit des configurations binaires telles qu'il existe toujours au moins une transition par groupe de trois bits.
  - Les autres configurations sont utilisées pour la signalisation (délimiteurs de trames, acquittements en fin de trame...)
- **Autres exemples**: Réseau Ethernet Gigabit et FC ("Fiber Channel") Codage 8B/10B.

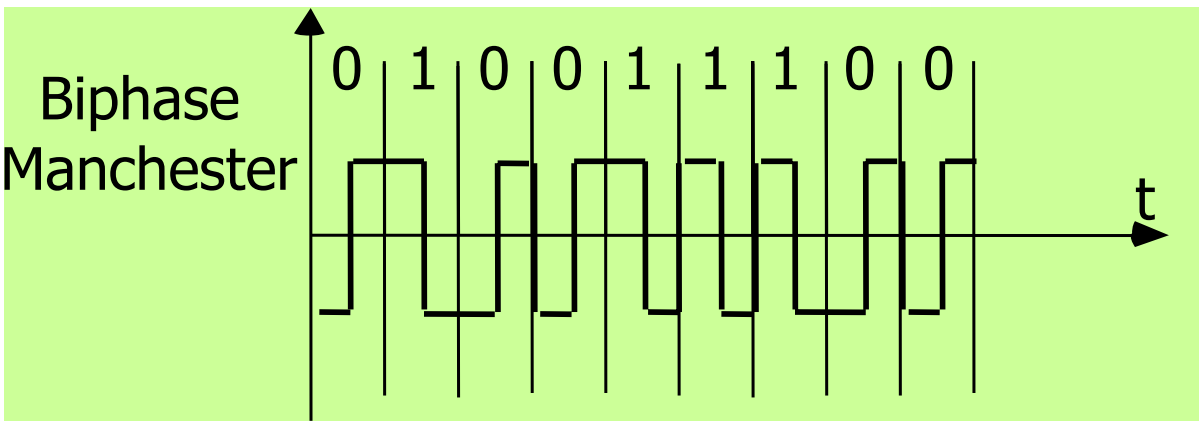
Signaux bande de base en amplitude
Modulation d'impulsion en amplitude

- **PAM** 'Pulse Amplitude Modulation'.
  - **Impulsions d'amplitudes différentes** (codages RZ avec retour à zéro) => transitions sur tous les bits.
  - **Exemple PAM-5 :** on utilise 5 impulsions (autres versions PAM-8 PAM-10 PAM-12 PAM-16).
  - **Utilisation PAM-5:** Ethernet 100 Base T2.



Signaux bande de base en phase
Code biphase (code Manchester)

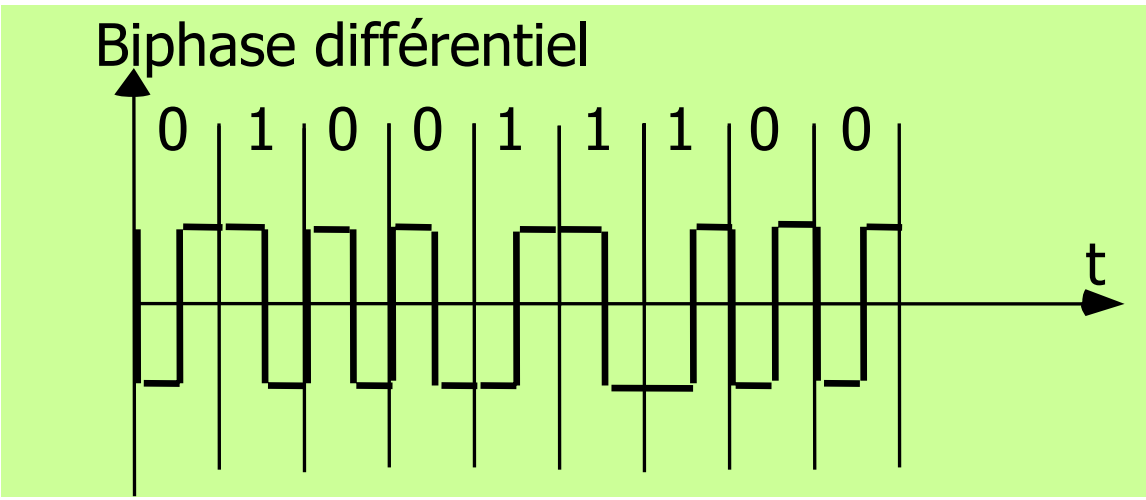
- **Utilisation de deux signaux d'horloge en opposition de phase.**
- **Le signal présente toujours une transition** au milieu de l'intervalle:
  - **Pour coder un 0** transition vers le haut.
  - **Pour coder un 1** transition vers le bas.



- **Exemple: Ethernet 10 Mb/s**.
- **Avantages:** Bonne occupation de la bande. Mise en œuvre très simple. Pas de composante continue (pas de problèmes sur les suites de symboles identiques). Transitions en milieu de période.
- **Inconvénients:** Nécessite une bande passante très large.

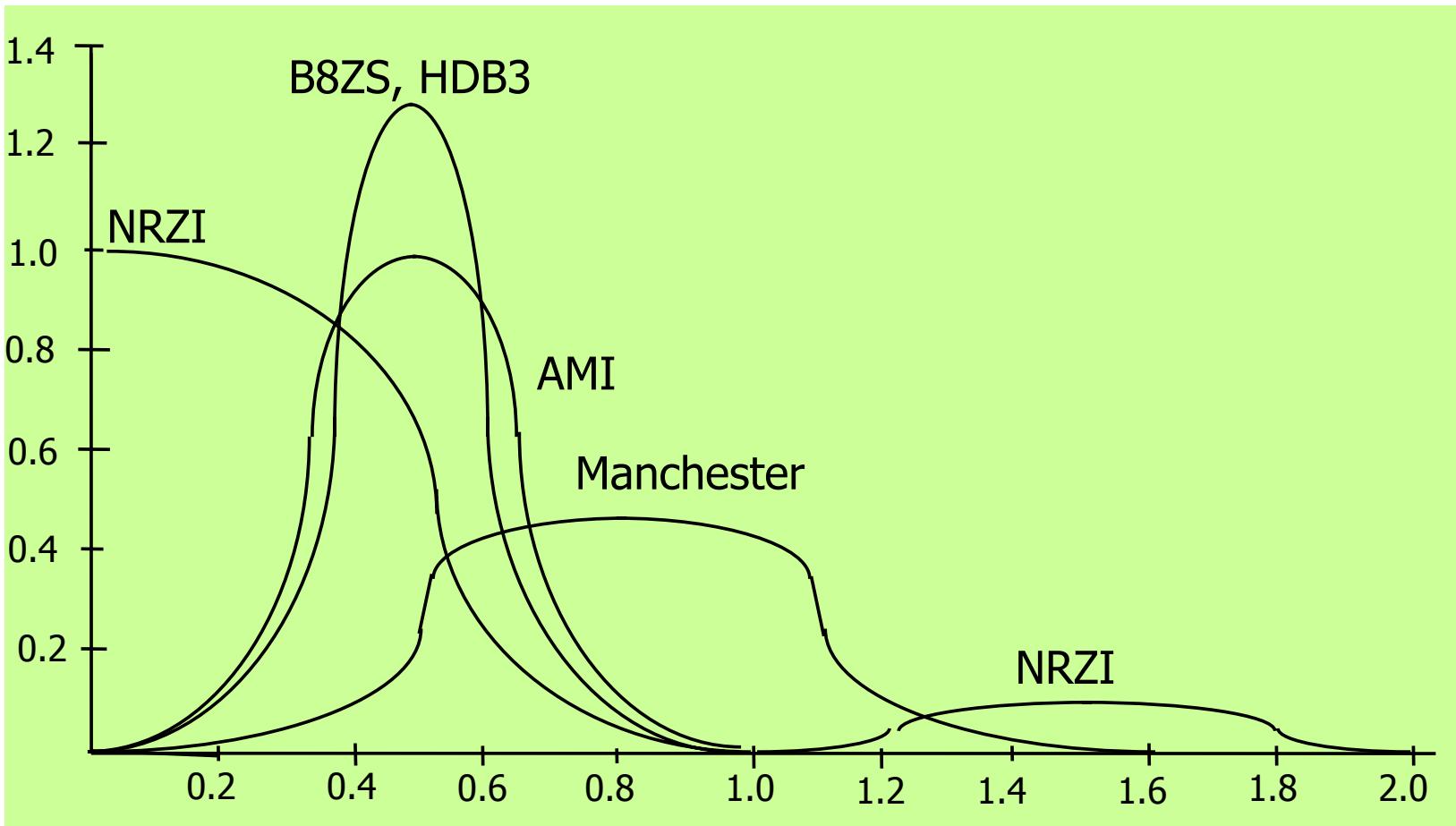
Signaux bande de base en phase Code biphase différentiel

- **Codage en phase:** On a toujours deux signaux en opposition de phase.
- **Codage du 0 :** transition en début de période (le 0 provoque toujours un changement de signal => modulation différentielle).
- **Codage du 1 :** absence de transition (le 1 conserve le même signal).
- **Exemple: Boucle IBM IEEE 802.5.**
- **Avantages Inconvénients:** Voisins de ceux du code Manchester.



Comparaison des différentes techniques de codage

- Puissance par unité de bande passante de différents codages en fonction de la fréquence normalisée.



La transmission analogique Modulation d'onde porteuse

- **Limitation des bandes de fréquence disponibles**
  - Exemple : Téléphone 300 à 3400 Hertz.
- **Affaiblissement et déformation des signaux** dus aux caractéristiques des câbles, ou des voies hertziennes....
- **Modems :** Nécessité de dispositifs d'adaptation de la source au canal (modulation à l'émission / démodulation à la réception).
- **Terminologie UIT-CCITT** : les ETCD ('Équipement de terminaison de circuit de données').

Transmission en modulation
Principe général

- **Signal de base** : S(t)
- **Porteuse sinusoïdale** : $P(t)=A_o \sin(\omega_o t + \phi_o)$
- **Modulation d'onde porteuse** :

  On transforme S(t) en X(t) = f(S(t)) en introduisant l'information du signal de base dans l'une des composantes.
  - Amplitude A
  - Fréquence ω
  - Phase ϕ

Modulation d'amplitude, de fréquence, de phase

- **Modulation d'amplitude** : $X(t) = f(S(t)) = A(S(t) \sin(\omega_0 t + \phi_o)$
- Exemple de base: Utilisation de deux amplitudes

  Codage du 0 : $A1 \sin(\omega_0 t + \phi_0)$ Codage du 1 : $A2 \sin(\omega_0 t + \phi_0)$

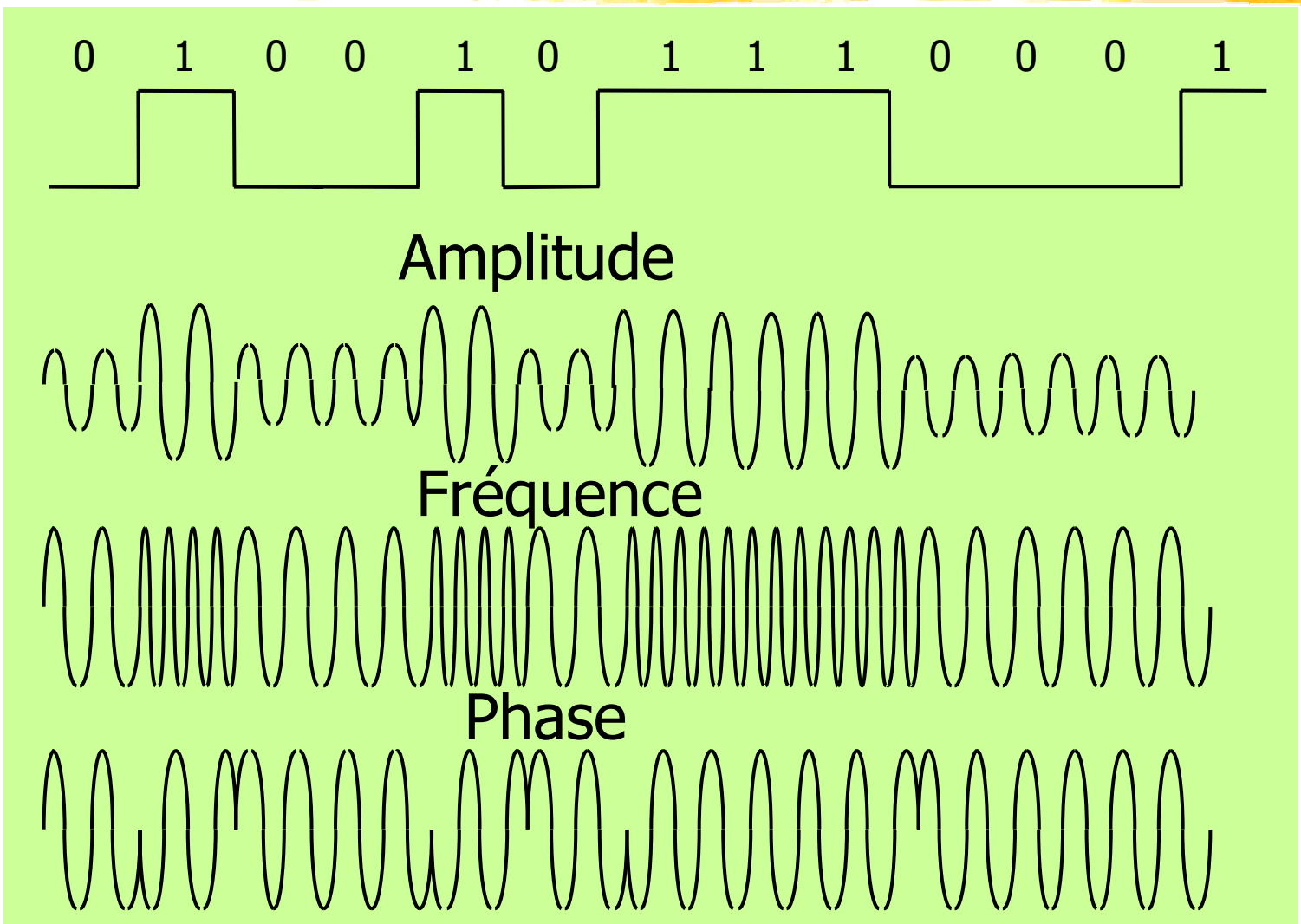
- **Modulation de fréquence** : $X(t) = f(S(t)) = Ao \sin(\omega(S(t)t + \phi_o)$
- Exemple de base: Utilisation de deux fréquences ω_1 et ω_2 .
- Modulation FSK "Frequency Shift keying"

  Codage du 0 : $Ao \sin(\omega_1 t + \phi_o)$ Codage du 1 : $A_o \sin(\omega_2 t + \phi_o)$

- **Modulation de phase** : $X(t) = f(S(t)) = Ao \sin(\omega_0 t + \phi(S(t))$
- Exemple de base: Utilisation de deux phases
- Modulation PSK "Phase Shift keying"

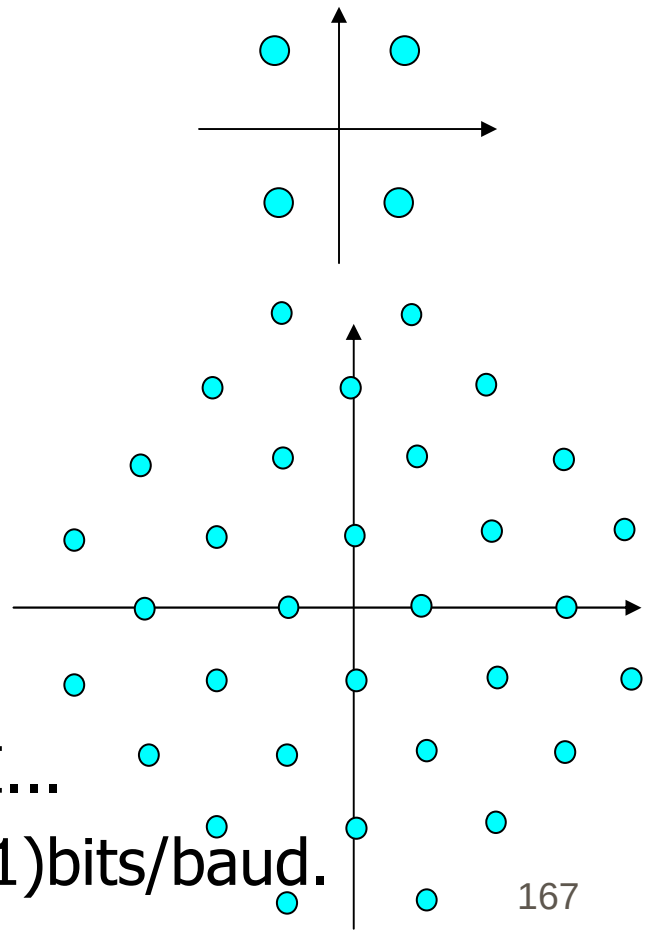
  Codage du 0 : $Ao \sin(\omega_0 t + \phi_1)$ Codage du 1 : $A_o \sin(\omega_0 t + \phi_2)$

Transmission en modulation
Exemples



Modulations combinées d'amplitude et de phase

- **Exemple 1** : Modulation QPSK 'Quadrature Phase Shift Keying'.
  - Quatre signaux de phases différentes et de même amplitude.
  - **Utilisation** : Wifi.
- **Exemple 2** : QAM 'Quadrature Amplitude Modulation
  - Modulation d'amplitude et de phase.
  - Diagramme spatial ou constellation
  - **Utilisation:** nombreuses ADSL,WIFI...
  - **Exemple:** QAM-32 Modem V32 (4+1)bits/baud.

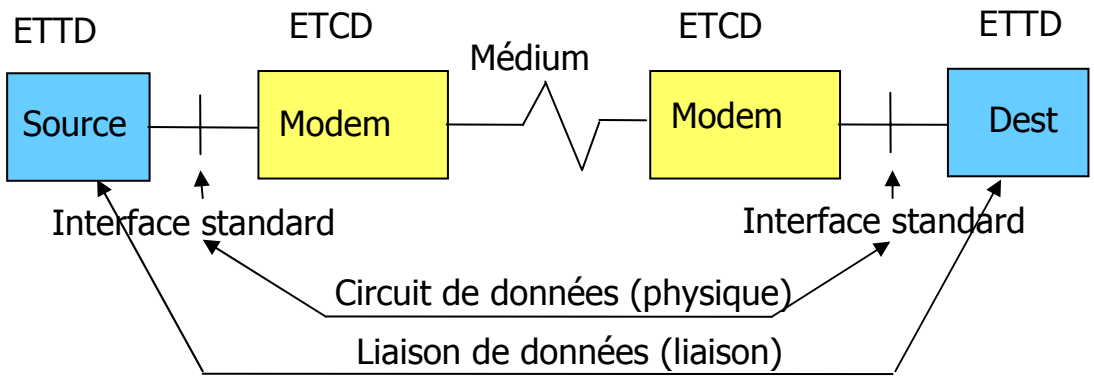


Niveau Physique

Seconde partie

Technologie du niveau physique

Introduction



- ETTD : Équipement terminal de traitement des données
- ETCD : Équipement de terminaison de circuit de données.

- **Différents éléments intervenant dans la transmission**
  - Contrôleur de communication.
  - Interface standard.
  - Modem.
  - Médium/Voie de communication.

Technologie du niveau Physique

Contrôleurs de communications (cartes réseaux)

Différentes fonctions d'un contrôleur (d'une carte réseau) (1)

- **Carte réseau = carte d'entrée/sortie**
  - **Dirigée** par des commandes d'entrées sorties
  - **Générant** des interruptions.
  - **Lecture/écriture** des informations en mémoire centrale (mode canal , DMA 'Direct Memory Access').
  - **Orientation principale**: communications séries.
- **Transmission en émission**
  - **Gestion des délimiteurs**.
  - **Emission données**: acquisition des mots mémoire, sérialisation, modulation.
  - **Génération** des codes détecteurs (ou correcteurs).

Différentes fonctions d'un contrôleur (d'une carte réseau) (2)

- **Transmission en réception**
  - Gestion des délimiteurs.
  - Désérialisation, écriture avec gestion de mémoire tampon (registres à décalages, ou mémoire locale carte).
  - Vérification des codes.
- **Gestion de l'accès au médium** (réseaux locaux).
- **Administration/configuration**
  - Adaptation aux différents débits
  - Adaptation aux différents modes:synchrone, asynchrone
  - Gestion du mot d'état (‘status word).
- **Logiciel système de pilotage : pilote (driver).**

Technologie du niveau Physique

Interfaces standards

Introduction aux interfaces standards

- **Notion d'interface standard:** L'ensemble des spécifications mécaniques et des utilisations des signaux électriques permettant une normalisation des interfaces de communication physique ETTD-ETCD.
- **Exemples:** (très très grande variété).
  - **Interfaces séries ITU/CCITT** V24/V28 RS232C; RS422/V11, RS423/V12 , V35, X21….
  - **Interface USB** (Universal Serial Bus).
  - **Interface Firewire** IEEE 1394.
  - ….
- **Autre besoin :** interfaces ETCD médium de communication.
  - **Interfaces téléphoniques** (pratiquement différentes dans tous les pays , RJ11)
  - ….

Exemple de l'interface V24

- **L'avis V24 du CCITT définit les spécifs fonctionnelles** de la jonction avec de nombreux modems bas débit.
- **L'avis V28 correspond aux caractéristiques électriques** des signaux de la jonction.
- **Spécifications V24/V28 (ITU), RS232C (EIA)** très voisines
- **Les protocoles sont codés** par des niveaux ouverts/fermés sur des signaux (nomenclature série 100) correspondant à des broches du connecteur DB25. Exemple: signal 125 broche 22.
- **Trois étapes principales**
  - Établissement / libération du circuit
  - Initialisation
  - Transmission

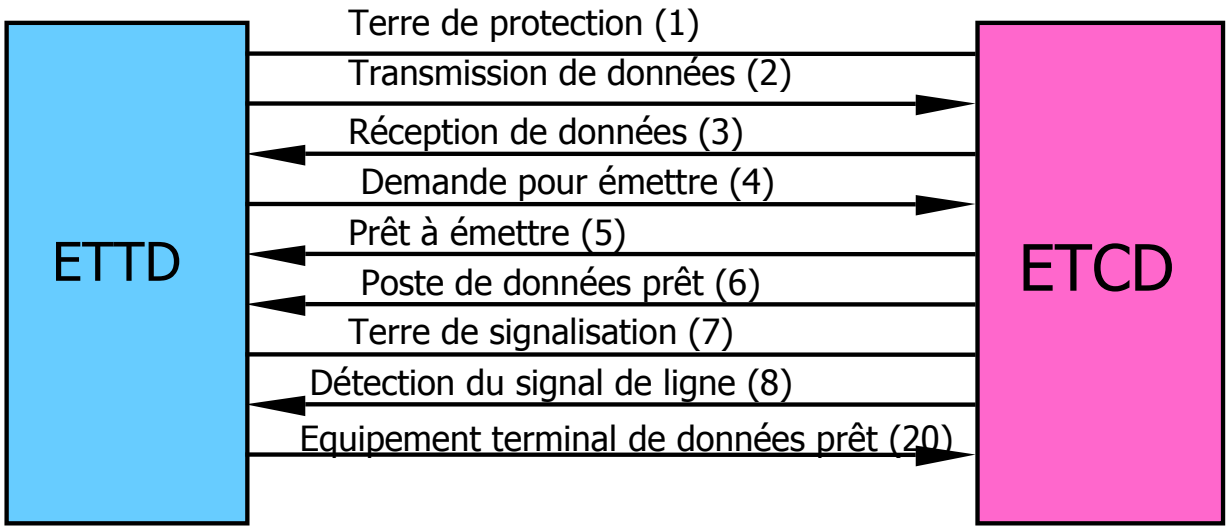
Interface V24 : Quelques signaux importants

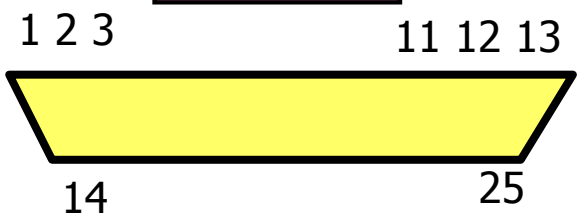
- **Établissement et libération du circuit**
  - 125 (22): Indication d'appel (coté modem), ETCD -> ETTD.

  (ex : sonnerie du téléphone)
  - 108 (20): Autorisation de connexion (du modem) ETTD->ETCD.
  - 107 (06): Acquittement (le modem est relié à la ligne) ETCD -> ETTD.
- **Initialisation**
  - 105 (04): Demande pour émettre ETTD-> ETCD
  - 106 (05): Prêt à émettre ETCD -> ETTD
  - 109 (08): Détection de porteuse ETCD -> ETTD
- **Transmission**
  - 103 (02): Données ETTD -> ETCD
  - 104 (03): Données ETCD -> ETTD
  - 113 (24): Horloge émission (générée par le terminal)

Interface V24 : Compléments

- **Représentation logique de l'interface**





- **Connecteur (mâle coté ETTD)**
- **Normes électriques V28**
  - (-3V) à (-25V) : 1 logique
  - (+4V) à (+25V) : 0 logique (on prend -5,+5 v)

Technologie du niveau Physique

Modems

Catégories de modems

- **Modulateurs/démodulateurs**
- **1) Modems téléphoniques** : utilisation du téléphone fixe sur paire téléphonique.
  - **Modems anciens (débits faibles)**
  - **Modems ADSL ‘Asymmetric Digital Subscriber Line’**
- **2) Modems cables**
  - **HFC ‘Hybrid Fiber Coaxial Cable’** : utilisation des câbles de télédistribution.

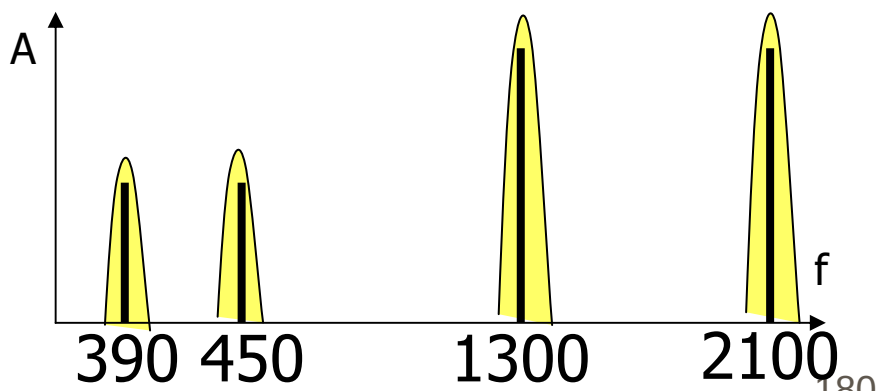
Modems anciens
Exemple : Avis V23

- **Débit** : 600/1200 bits/seconde (Modem du minitel)
- **Transmission** : asynchrone
- **Support utilisable** : Réseau commuté (2 fils) ou LS (4 fils)
- **Mode** : Duplex à l'alternat (ancien)

Duplex intégral : 600b/s ou 1200 b/s voie de retour 75b/s.

- **Principe** : modulation de fréquence pour les deux canaux

	Voie descendante	Voie de retour
1	2100 Hz	390 Hz
0	1300 Hz	450 Hz



Modems anciens
Exemple : Avis V32

- **Débit :** 2400,4800,9600 bits/seconde.
- **Transmission :** synchrone.
- **Support utilisable** : 2 fils ou Réseau Téléphonique Commuté.
- **Mode :** duplex intégral
- **Principe de modulation :** QAM 'Quadrature Amplitude Modulation)
- **Rapidité de modulation**: 2400 bauds ;

Valence: 32 => Le débit réel 12 000 b/s est supérieur au débit utile permettant l'utilisation d'un code correcteur implanté par le modem.

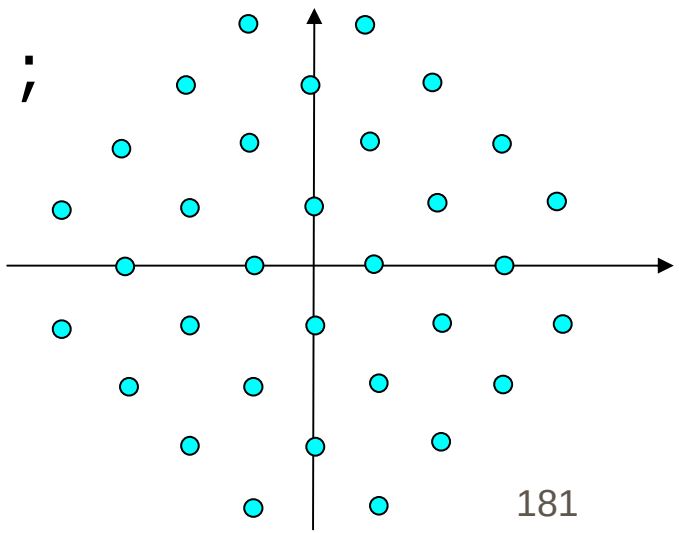


Tableau des normes de modems anciens

Avis	Mode	Débit	Modul	Support	Trans
V21	Async	300b/s	Fr	RTC-LS2	Dup
V22	Syn/Async	600-1200b/s	Ph	RTC-LS2	Dup
V22bis	Syn/Async	1200-2400b/s	Am Q	RTC-LS2	Dup
V23	Syn/Async	1200-75b/s	Fr	RTC-LS2	Dup
V26	Syn	2400b/s	Ph	LS4	Dup
V27	Syn	4800b/s	Ph-D	LS4	Dup
V29	Syn	9600-4800b/s	Ph+A	LS4	Dup
V32	Syn/Async	9600-4800b/s	Am Q	RTC-LS2	Dup
V32bis	Syn/Async	14400-4800b/s	Am Q	RTC-LS2	Dup
V33	Syn	14400-12000b/	Am Q	LS4	Dup
V34	Syn/Async	33600-2400b/s	Am Q	RTC-LS2	Dup

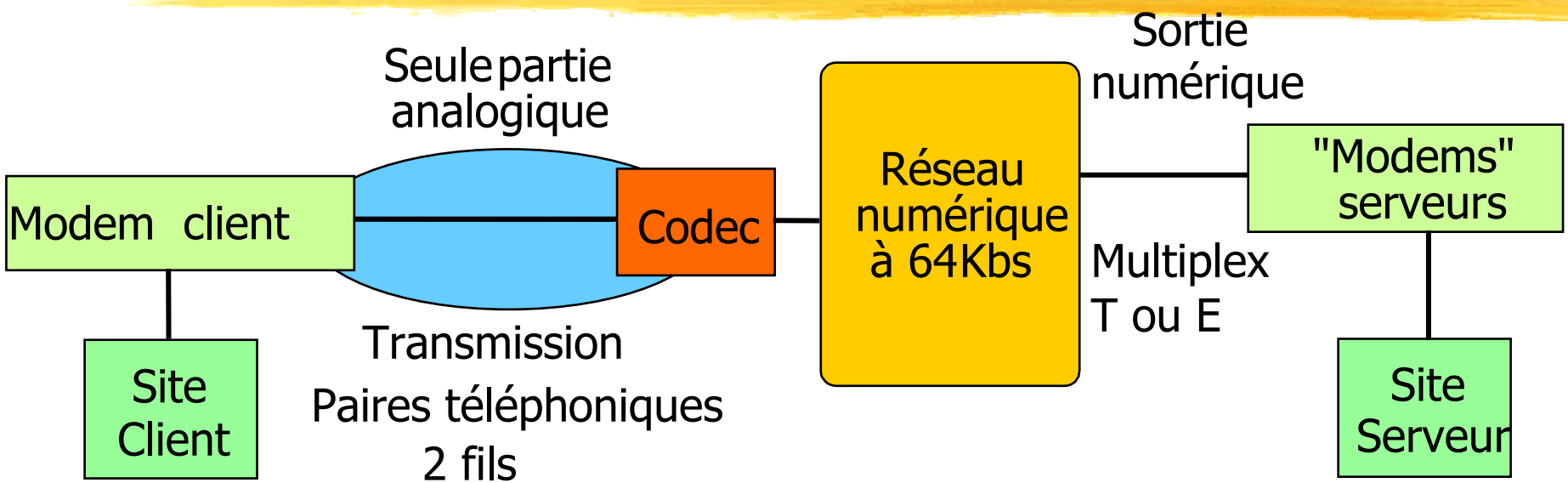
Explications

- Syn: Synchrone
- Async: Asynchrone
- Am Q : modulation d'amplitude et de phase (QAM).
- Ph : modulation de phase
- Fr : modulation de fréquence
- RTC : Réseau téléphonique commuté
- LS 2/4 : liaison spécialisée 2 ou 4 fils.
- Dup : mode bidirectionnel simultané

Le modem V90/V92 (56 Kb/s)

- **Rappel**
  - **Les modems traditionnels** considèrent le RTC (Réseau Téléphonique Commuté) comme **un réseau entièrement analogique**.
  - **Mais le RTC est pratiquement numérique** à 64Kb/s.
  - **Le téléphone numérisé MIC** avec un bruit de quantification des codecs de l'ordre de 36db offre un débit maximum théorique de l'ordre de 35 Kb/s
  - **D'ou la norme V34** constituant une limite pour l'approche analogique avec un débit de 33,6 Kb/s.

Principes d'une nouvelle génération de modems sur RTC



- **En fait le réseau téléphonique est numérisé à 64 Kb/s** sur toute son étendue (sauf le rattachement abonné).
- **Les serveurs d'accès peuvent être directement rattachés** au réseau numérique via des multiplexes PDH (T Etats-Unis ou E Europe).
- **La seule partie analogique est le rattachement usager** sur lequel le rapport signal à bruit permet un débit supérieur à 33,6 kbs.

La solution V90 (Rockwell , US Robotics)

- **Le modem V90 offre deux débits différents**
  - 33600 b/s sens client vers serveur (transmis en mode V34).
  - 56000 b/s sens serveur vers client (le plus intéressant à étudier)
- **A 56 kb/s le coté serveur** envoie des configurations numériques sur 8 bits vers le client.
  - **En fonction de la loi de quantification** le codec serveur génère des "symboles" analogiques
  - **Le modem client interprète la modulation reçue** (les signaux analogiques reçus) pour reconstituer les octets transmis.
- **Problème de mise en œuvre**:
  - **Le codec est conçu pour traiter des signaux de voix humaine** (les niveaux de quantification les plus faibles sont privilégiés) d'ou un problème de discrimination dans le domaine des autres niveaux.
  - **Utilisation d'une technique de codage** sur la partie analogique qui s'effectue avec perte de 8Kb/s ramenant le débit atteint à 56 Kb/s.

Conclusion Modem V90

- **Impossibilité de fonctionner à 56 Kb/s dans les deux sens:**
  - V90 : 34 Kb/s quand même => pas très grave car la voie descendante est souvent privilégiée par rapport à la voie montante.
  - Modem V92 voie montante à 48 Kb/s plus d'autres améliorations

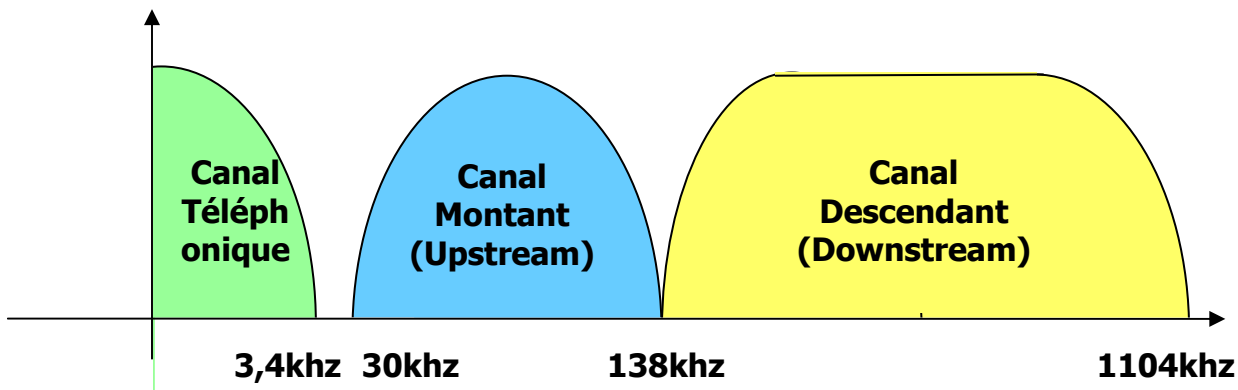
- **Pour que le débit 56 Kb/s fonctionne** il faut que le RTC soit entièrement **numérique MIC et selon la même loi de quantification.**
  - Pas de section analogique
  - Pas de conversion loi A en loi μ interne.
  - Pas de section avec codage non MIC (exemple pas de sections impliquant une conversion vers ADPCM)
  - Lors de l'établissement de la communication les modems opposés doivent tester s'ils peuvent effectivement fonctionner à 56 Kb/s sinon repli en V34 (ou moins encore).

Le modem ADSL ANSI T1.413, UIT G992 ('Asymmetric Digital Subscriber Line')

- **Idée principale :** Utilisation de la bande passante des paires téléphoniques de raccordement abonné/autocommutateur.
  - Exemple: 1,1Mhz sur moins de 3000 m.Pas utilisable au delà de 6000 m.
- **Solution fréquente :** Multiplexage fréquentiel de trois canaux.

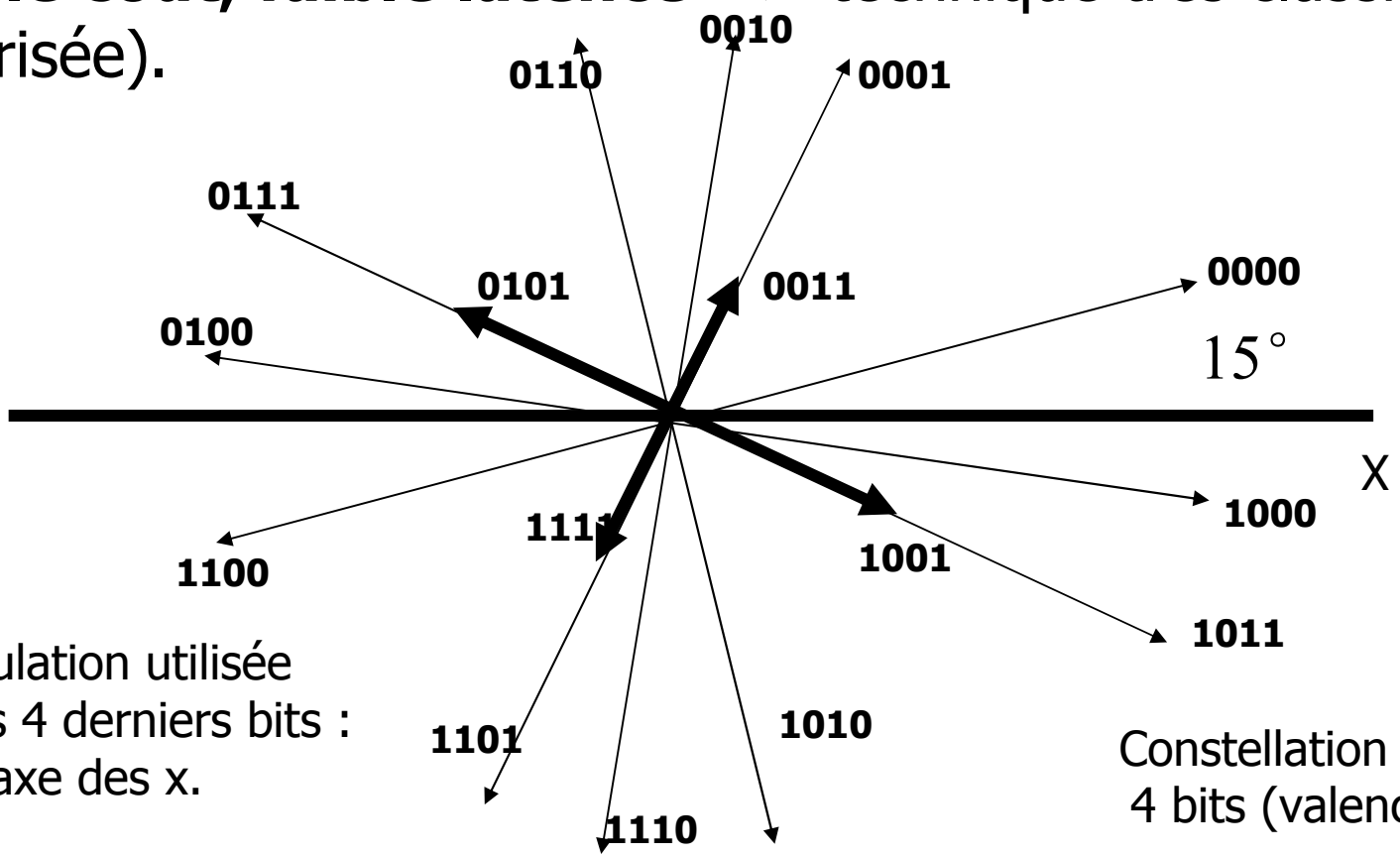
1) **Canal téléphonique** (3,4khz) : vers le réseau téléphonique classqiue.
2) **Canal numérique montant (upstream)** abonné -> central (max 1 Mb/s).
3) **Canal numérique descendant (downstream)** central -> abonné (max 8 Mb/s).

=> **Débits asymétriques dépendant de la qualité et de l'offre.**



Codage des voies numériques
1) CAP 'Carrierless Amplitude Phase'

- **Modulation d'amplitude et de phase** en s'adaptant en débit à la qualité de la bande disponible pour un canal numérique.
- **Faible coût, faible latence =>** technique très classique (bien maîtrisée).

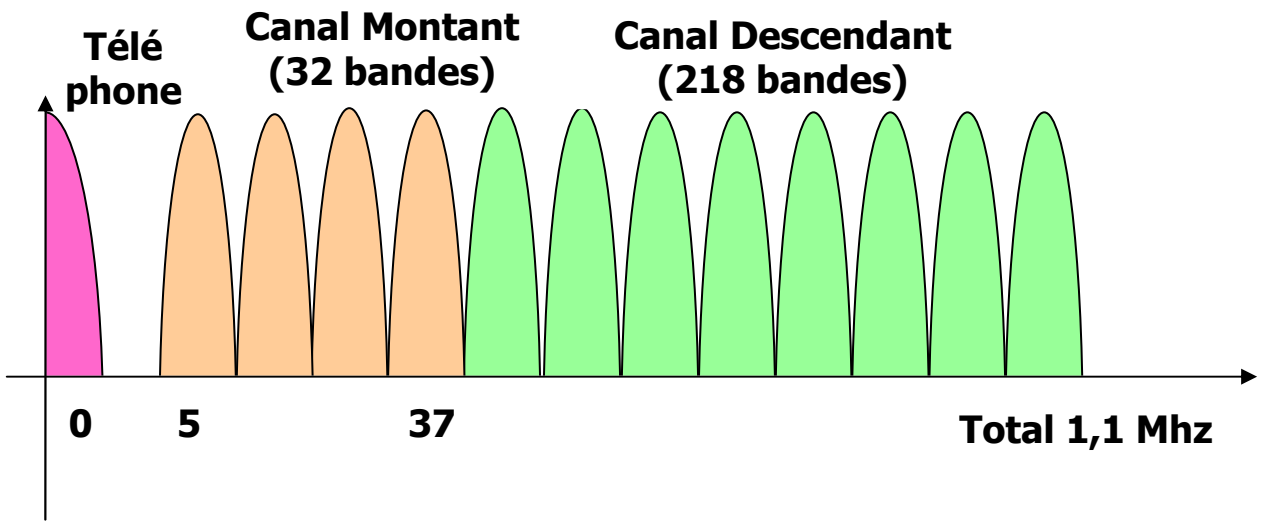


Modulation utilisée pour les 4 derniers bits : axe des x.

Constellation utilisée 4 bits (valence 16).

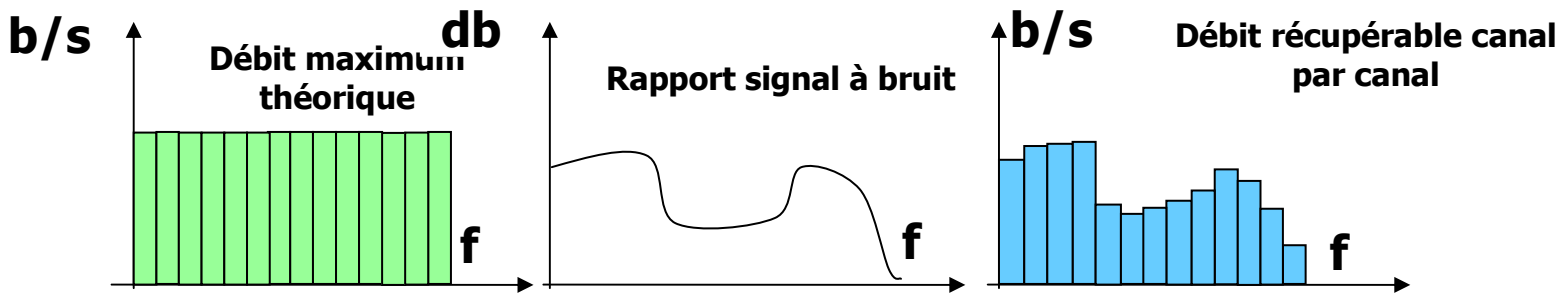
Codage des voies numériques
2) DMT 'Discrete MultiTone'

- **Technique plus fiable et plus sophistiquée, la plus élégante.**
- **Division du spectre en 256 canaux de 4 Khz** (exactement 4312,5 Hz) => Partage des canaux dépendant de l'opérateur.
- **Exemple de partage :** 1 canal téléphonique, 5 non utilisés, 32 flux montant et 218 flux descendant gérées indépendamment en modulation d'amplitude et de phase.



Technique de découverte adaptative du rapport signal à bruit (bande par bande).

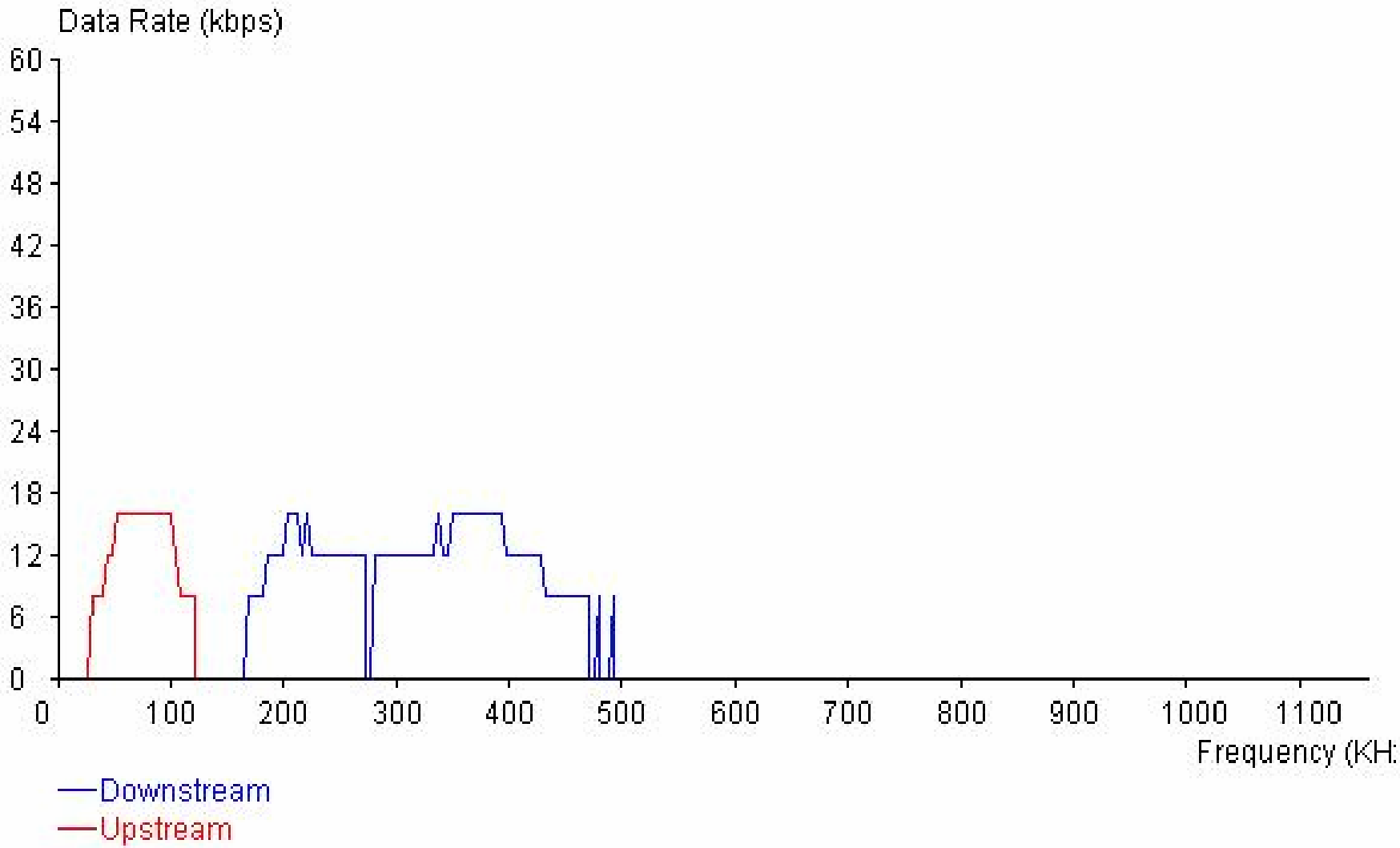
- **La qualité de la transmission n'est pas la même** : pour tous les abonnés, et dans chacun des 256 canaux (section de la paire torsadée, imperfections de la paire, longueur, …)
- **Débit théorique maximum par canal de 4khz 60 Kb/s** : soit pour le canal montant 1,5 Mb/s et pour le canal descendant 14,9 Mb/s.



- **Exemples de** performances du canal descendant.

Débit	Section	Distance
1,5 Mb/s	0,5 mm	5,5 Km
6,1 Mb/s	0,4 mm	2,7 Km

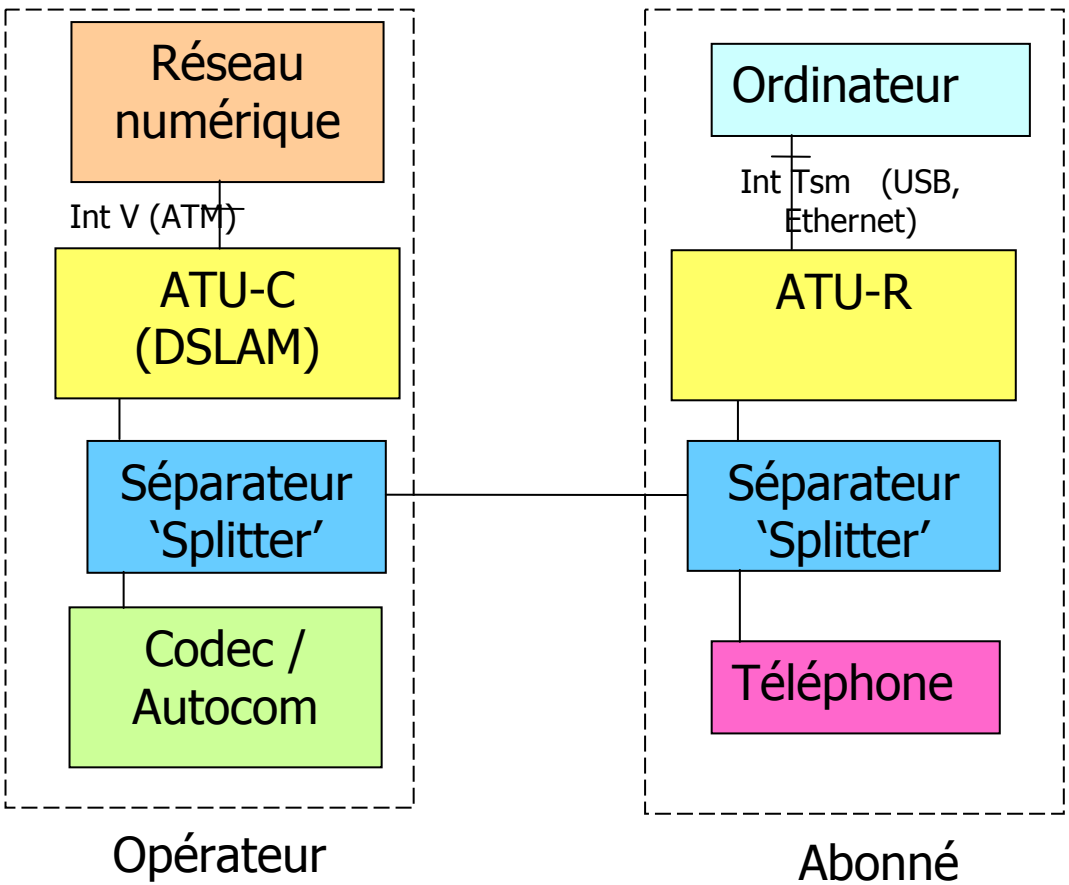
Un exemple de diagramme réel



ADSL: un ensemble complexe de techniques de transmission

- **Embrouillage** des signaux.
- **Correction automatique** d'erreur par code correcteur d'erreurs dans les trames (FEC Forward Error Correcting Code Reed-Solomon).
- **Codage QAM** pour chaque canal: rapidité de modulation 4000 baud, définition de la constellation utilisée permettant selon le rapport signal à bruit jusqu'à 15 bits par intervalle).
- **Tramage** : 68 trames ADSL sont regroupées dans une super trame.
- **Synchronisation** des trames.

ADSL: aspects architecturaux



- **ATU ADSL Transceiver Unit** (C Central, R Remote): l'essentiel du modem ADSL.
- **DSLAM Digital Subscriber Line Access Multiplexer** : multiplexeur d'accès ADSL chez l'opérateur
- **Interface V** : nom générique ADSL de l'interface avec le réseau numérique de l'opérateur (exemple un réseau ATM +IP)
- **Interface Tsm** : nom générique de l'interface entre le modem ADSL et l'ordinateur usager (branchement sur port USB ou sur port ethernet).

Technologie du niveau Physique

Voies de communications (média)

Introduction

- **Objectif du chapitre : apécifier les caractéristiques** des câblages utilisés couramment dans l'entreprise.
  - **Paires torsadées.**
  - **Fibres optiques.**
  - **Câbles coaxiaux.**

Les paires torsadées

- **Deux conducteurs en cuivre, isolés**.
- **Enroulés de façon hélicoïdale** autour de l'axe (l'enroulement permet de réduire les inductions électromagnétiques parasites de l'environnement).
- **Utilisation courante**
  - **Réseaux téléphoniques**.
  - **Réseaux locaux**.

Paires torsadées blindées STP "Shielded Twisted Pairs"

- **Fournies en câbles de 2 paires**.
- **Chaque paire est blindée.**
- **L'ensemble est également blindé**.
- **Possibilités importantes** -> 500 Mhz.
- **En fait différentes difficultés**.
  - **Plus coûteux à l'achat** et à **l'installation.**
  - **Assez encombrant.**
  - **Problèmes posés par les courants dans les blindages** lorsque les appareils reliés sont à des potentiels différents.
- **Exemple d'utilisation:** Boucle IBM avec le connecteur DB9 ou le connecteur IBM UDC "Universal Data Connector" Performances exploitées faibles en débit => 16 Mb/s.

Paires torsadées non blindées UTP "Unshielded Twisted Pairs"

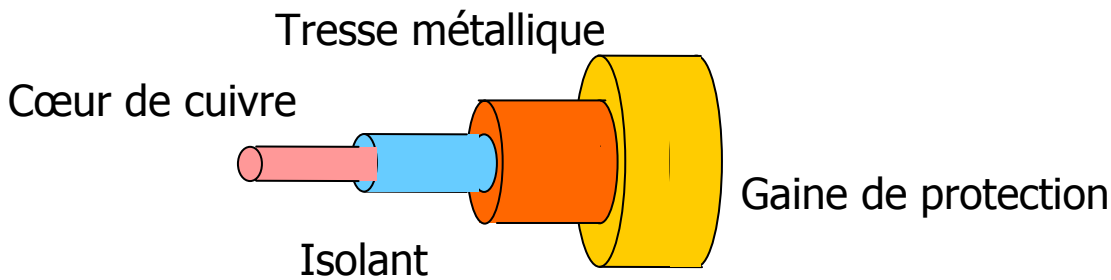
- **Fournies en câbles de 4 paires.**
- **Possibilités significatives** (en développement permanent).
- **Pour un coût raisonnable.**
- **Existence de différentes catégories** de paires UTP aux caractéristiques très normalisées (normes EIA 568):
  - **Impédanc**e caractéristique.
  - **Influence d'une paire sur l'autre** en db.
  - **Atténuatio**n (en db).

Les catégories EIA568B "Commercial BuildingTélécommunications"

- **Cat 1:** Anciennement recommandée pour le téléphone (actuellement non recommandée).
- **Cat 2**: Anciennement recommandée pour la boucle IBM 4Mb/s. (actuellement non reconnue).
- **Cat 3**: Recommandée pour les bandes de fréquences jusqu'à 16 MHz. Utilisation typique pour le réseau Ethernet 10 Mbit/s.
- **Cat 4**: Anciennement recommandée jusqu'à 20 MHz typiquement pour la boucle IBM 16 Mbit/s.
- **Cat 5**: Performance jusqu'à 100 MHz pour le réseau Ethernet à 100 Mbit/s (actuellement non recommandée).
- **Cat 5e**: Recommandée pour les performances jusqu'à 100 MHz pour Ethernet à 100 Mbit/s et Ethernet Gigabit.
- **Cat 6, 6A**: Performance jusqu'à 250 MHz (Ethernet Gigabit ou 10G).
- **Cat 7**: Performance jusqu'à 600 MHz (en fait quatre paires torsadées blindées STP). (Ethernet Gigabit ou 10G).

Câbles coaxiaux

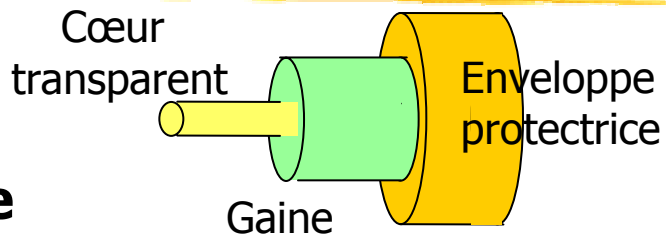
- **Deux conducteurs concentriques**: un **conducteur central** le coeur, un matériau **isolant** de forme cylindrique, une tresse concentrique **conductrice**:



- **Meilleures caractéristiques que les paires torsadées** (largeur de la bande passante et protection contre les rayonnements parasites).
- **Actuellement en perte de vitesse au profit des paires torsadées ou des fibres.**

Fibres optiques

- **Fibre optique = guide de lumière.**
  - **Un cœur transparent (plastique, verre)**
  - **Entouré d'une gaine puis une enveloppe**
  - **Le cœur a un indice de réfraction plus élevé que la gaine pour confiner la lumière**



- **Système de transmission à fibre : Trois composants**
  - **Conversion d'un signal électrique en signal optique** source de lumière : Diode **LED** "Light Emitting Diode" ou Diode laser.
  - **Fibre optique** : qui joue le rôle d'un guide d'ondes lumineuses.
  - **Détecteur de lumière** : photodiode de type **PIN** ("Positive Intrinsic Negative") ou à avalanche, qui restitue un signal électrique.
- **Différents types de fibre**
  - **Fibres à saut** d'indice, **à gradient** d'indice**, à cristaux photoniques**.
  - **Fibres multimodes, fibres monomodes**.

Avantages des fibres optiques

- **Bande passante très importante** (de l'ordre de 1 GHz/km => débit binaire très important).
- **Faible affaiblissement** de l'ordre de 0,25 dB/km.
- **Plusieurs dizaines de kilomètres** entre amplificateurs.
- **Peu sensible aux perturbations** électromagnétiques
- **Taux d'erreur très faibles**
- **Utilisation dans des environnements difficiles** (variation de température...).
- **Matière première bon marché**.
- **Légèreté et faible volume** qui permettent la pose de tronçons longs avec plusieurs fibres.
- **Possibilité de multiplexage en longueur d'onde** (WDM).

Problèmes des fibres optiques

- **Les raccordements** (épissures optiques) restent délicats à réaliser sur le terrain et introduisent un affaiblissement.
- **Les dérivations ne peuvent s'effectuer qu'au prix d'une perte de puissance importante** (limitation de l'utilisation de la fibre optique pour la diffusion).
- **Régénération, amplification et commutation** optique restent des sujets de recherche.

- **Malgré cela : Bande passante très importante, affaiblissement très bas,**

=> Les liaisons optiques sont très utilisées.

Technologie du niveau Physique

Réseaux au niveau physique

Exemple du réseau téléphonique commuté et des réseaux PDH, SDH

Introduction à la transmission téléphonique numérique

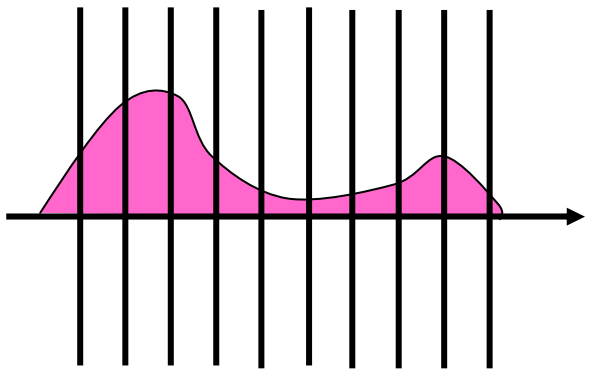
- **La transmission numérique offre des performances supérieures à la transmission analogique:**
  - **Faible taux d'erreur** des liaisons numériques. Les répéteurs numériques, régénérateurs de signaux, ... ne connaissent pas les inconvénients des amplificateurs utilisés par les supports analogiques.
  - **Les informations de type voix, images et données, représentées sous forme numérique peuvent facilement être multiplexées** sur un même support de transmission.
- **Utilisation de la numérisation dans le Réseau Téléphonique Commuté** => RNIS Réseau Numérique à Intégration de Services (bande étroite):
  - **Canaux de communication B à 64 Kb/s**

MIC Modulation d'Impulsion Codée PCM "Pulse code modulation"

- **Une technique pour la transmission** du son sans compression utilisée par le téléphone (également disques compacts audio, fichiers sons…).
- **Le MIC se définit par trois caractéristiques principales :**
  - **Echantillonnage.**
  - **Quantification.**
  - **Codage.**
- **Notion de codec** (codeur/décodeur): convertisseur d'un signal analogique sonore en signal numérique et inversement.

MIC Echantillonnage

- **Echantillonnage : le processus de choix** des instants de prise de mesure sur un signal continu (par exemple: la voix parlée).

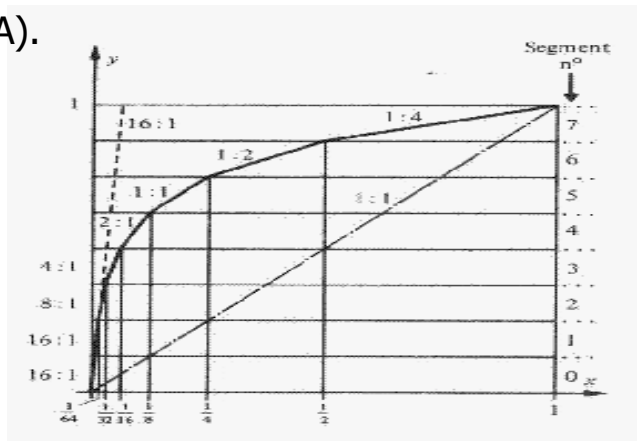


- **Bande passante visée** : B=4000 Hz.
- **Fréquence d'échantillonnage Nyquist :** R = 2B = 8000 échantillons/s
- **Période :** 1 échantillon/125µs)
- **Échantillons sur 8 bits** => Débit binaire 64 Kb/s.

MIC Quantification

- **Quantification :**
  - **le processus de conversion** d'une mesure d'un signal continu en une valeur numériques entière.
  - **L'amplitude entre deux valeurs** voisines définit la précision.
  - **L'erreur qui résulte du processus de quantification** est appelé le **bruit de quantification.**
- **Établissement de la correspondance effective**
  - Mesure du signal analogique sur 13 bits en Europe (14 bits aux USA).
  - Fourniture d'un nombre sur 8 bits
  - Le plus simple serait une correspondance linéaire:
  - En fait on utilise une courbe semi-logarithmique

garantissant une précision relative constante
(les petites valeurs sont quantifiés avec 16 fois
plus de précision que les grandes).



- **En Europe** : Loi A. **Aux USA,Japon** : Loi Mu.

MIC Codage

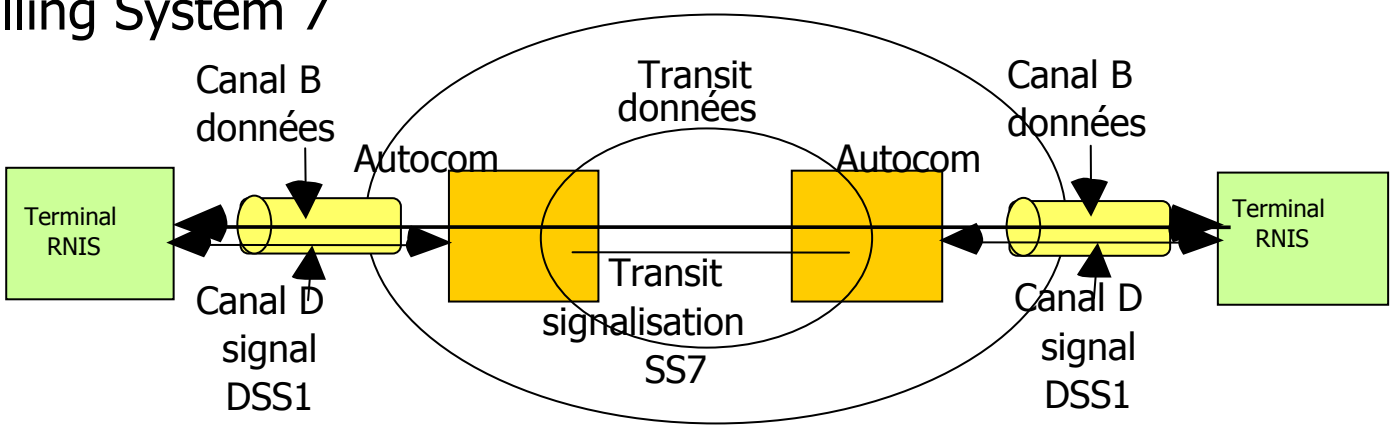
- **Codage:** Étape finale de représentation numérique des octets => Utilisation d'un codage de la valeur numérique quantifiée.
- **En Europe:** méthode ADI "Alternate Digital Inversion"
  - 1 bit sur 2 inversé : 0000 -> 0101
- **Aux USA** : méthode ISC "Inverse Symmetrical Coding"
  - Inversion par groupes de deux bits : 0011 -> 1100.

Autres procédés de codage numérique de la voix

- **Techniques de compression** pour un codage limitant la bande passante utilisée => propriétés de la voix humaine.
- **Modulation MIC différentielle (DPCM "Differential Pulse code modulation")**
  - **On code la différence** entre la valeur courante et la valeur précédente
  - **Exemple DPCM** : Si des écarts entre échantillons de ± 16 incréments, ou plus, sont très peu probables alors un codage sur 5 bits est suffisant.
  - **Lorsque l'écart est supérieur** à ± 16 incréments de quantification l'encodage utilise plusieurs échantillons pour rétablir la situation.
- **Modulation DELTA**
  - **On code sur un seul bit** chaque échantillon en indiquant s'il est plus grand ou plus petit que le précédent de une unité.
  - **Problèmes de rétablissement en cas de variations rapides.**

Réseau téléphonique : Fonctions de signalisation

- **Sur tout réseau téléphonique, nombreuses fonctions** de **signalisation indispensables d'accès au réseau** et d'administration (exemple : l'usager décroche, compose un numéro, le prestataire taxe ...)
- **Réseau téléphonique ancien:** signalisation dans la bande => problèmes.
- **Réseau téléphonique numérique**: un réseau numérique de signalisation (hors bande) utilise un protocole 'système de signalisation n°7) SS7 Signalling System 7

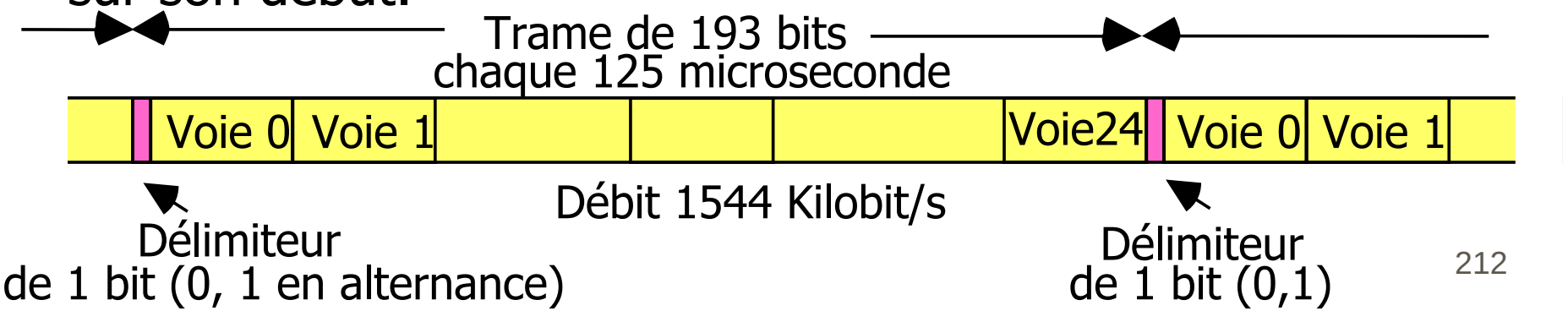


Les hiérarchies de multiplexage dans les réseaux téléphoniques

- **Nécessité du multiplexage** : pour optimiser les supports
  - Transport des circuits téléphoniques en multiplexage de circuits basse vitesse 64 Kb/s sur des voies haut débit.
  - Un échantillon chaque 125 micro secondes => Trames de n échantillons répétées chaque 125 micro secondes.
- **Rappel:** deux grandes classes de techniques de multiplexage
  - **Multiplexage par Répartition de Fréquence** (MRF ou FDM "Frequency Division Multiplexing") une bande de fréquence à chaque signal transporté => Solution ancienne
  - **Multiplexage à Répartition dans le Temps MRT** ("TDM Time Division Multiplexing") une tranche de temps à chaque signal transporté.
- **Les deux approches** industrielles de multiplexage temporel.
  - **PDH Plesiochronous Digital Hierarchy**
  - **SDH Synchronous Digital Hierarchy**

Problèmes posés par la transmission à haut débit en continu: Verrouillage

- **Détermination de l'emplacement des informations significatives**
  - **Les infos multiplexées sont "tramées":** assemblées dans des trames de taille fixe les trames sont émises en continu.
  - **Exemple du multiplex T1** pour 24 voies téléphoniques MIC (États-Unis)=> Il faut pouvoir retrouver les trames en réception en présence de multiples aléas de transmission (bruits, désynchronisations ...).
- **Notion de "verrouillage de trame"**: Une trame est dite "verrouillée" pour le récepteur s'il est correctement synchronisé sur son début.

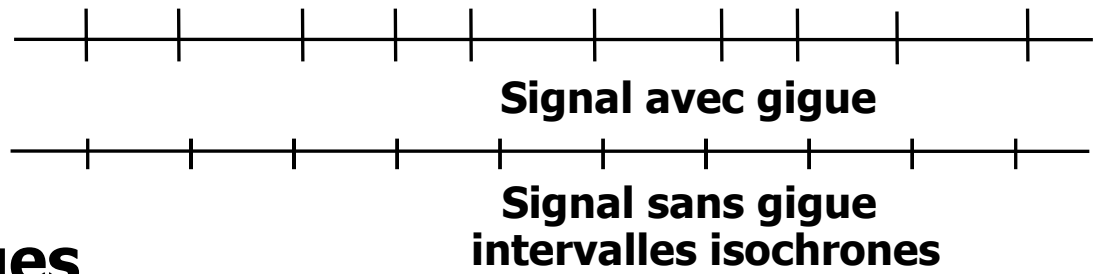


Problèmes de synchronisation d'horloges

- **Différences de fabrication => Différences de fréquence**
  - Les informations sont générées avec des horloges qui ne peuvent générer de manière parfaites des fréquences identiques.
  - Définition de tolérances sur les
- **Dérive d'horloges ("Wander")**
  - La fréquence d'une horloge dépend de la température et peut donc subir des dérives lentes (à l'échelle de plusieurs heures ou même plus) autour d'un valeur moyenne.
  - La variation en fréquence d'une horloge par rapport à une autre doit être bornée
  - Exemple: une horloge pour générer le multiplex PDH à 2,048 Mb/s doit-être à plus ou moins 5 10^{-5} ou encore + - 50 ppm (parties par million).

Problèmes de synchronisation d'horloges

- **Gigue ("Jitter")**
  - La fréquence d'un signal peut présenter des variations instantanées (autour d'une fréquence moyenne) lorsque des retards différents sont introduits dans le traitement des signaux.



- **Déphasages**
  - Les signaux issus de lignes différentes présentent des retards de propagation liées à la distance.

  => Déphasages entre les signaux.

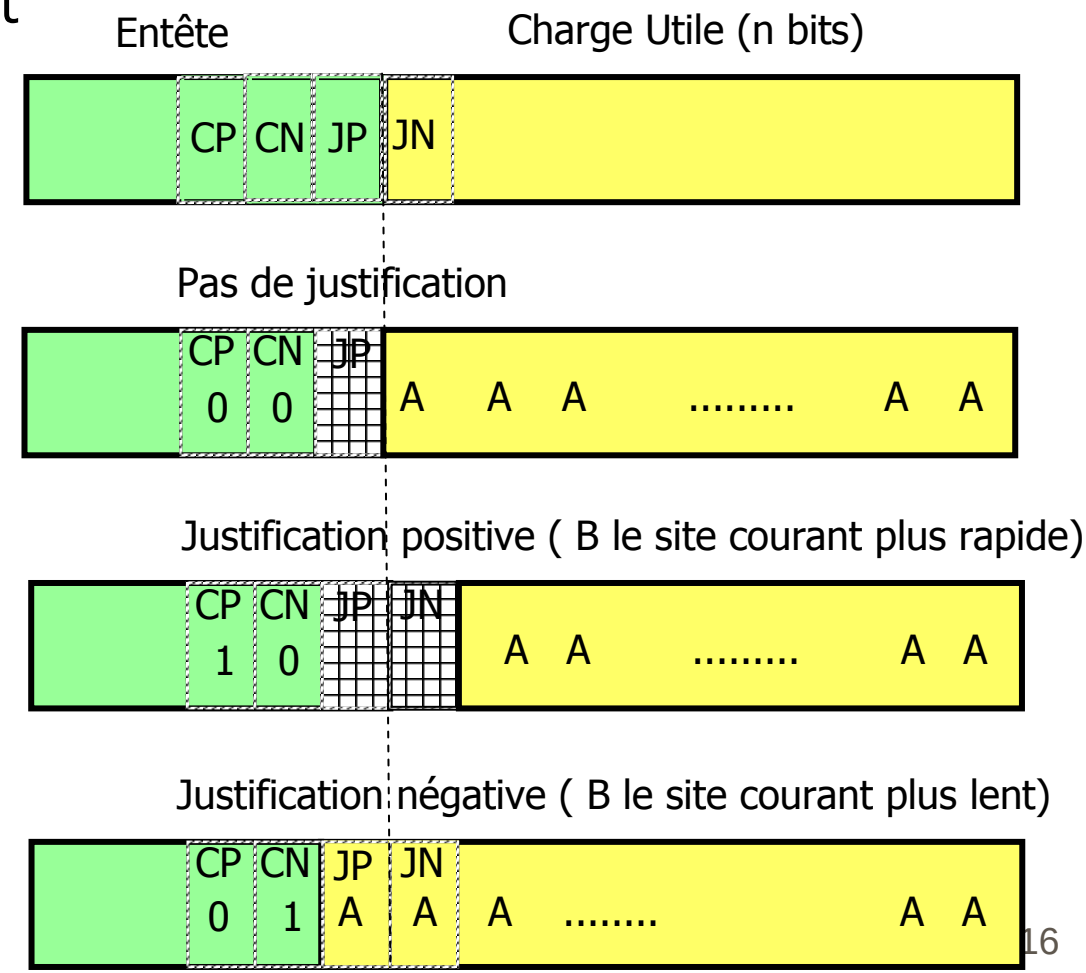
- **Débits**
  - Les différents appareils intervenant dans une chaîne de transmission ne traitent pas exactement les informations à la même vitesse.

Solutions de synchronisation d'horloges: techniques de justification

- **Justification Positive, Nulle ou Négative**
- Deux sites reliés par une voie haut débit:
- Les horloges de A et B sont **identiques** => Justification **nulle**
  - **Pas de justification**: on réémet le **même nombre** de bits
- L'horloge de A est plus rapide que celle de B => Justification **négative**
  - Pour un nombre de bits donné arrivant en B de A, on doit en renvoyer **plus**.
- L'horloge de A est plus lente que celle de B => Justification **positive**
  - Pour un nombre de bits donné arrivant en B de A on doit en renvoyer **moins**.

Fonctionnement de la justification

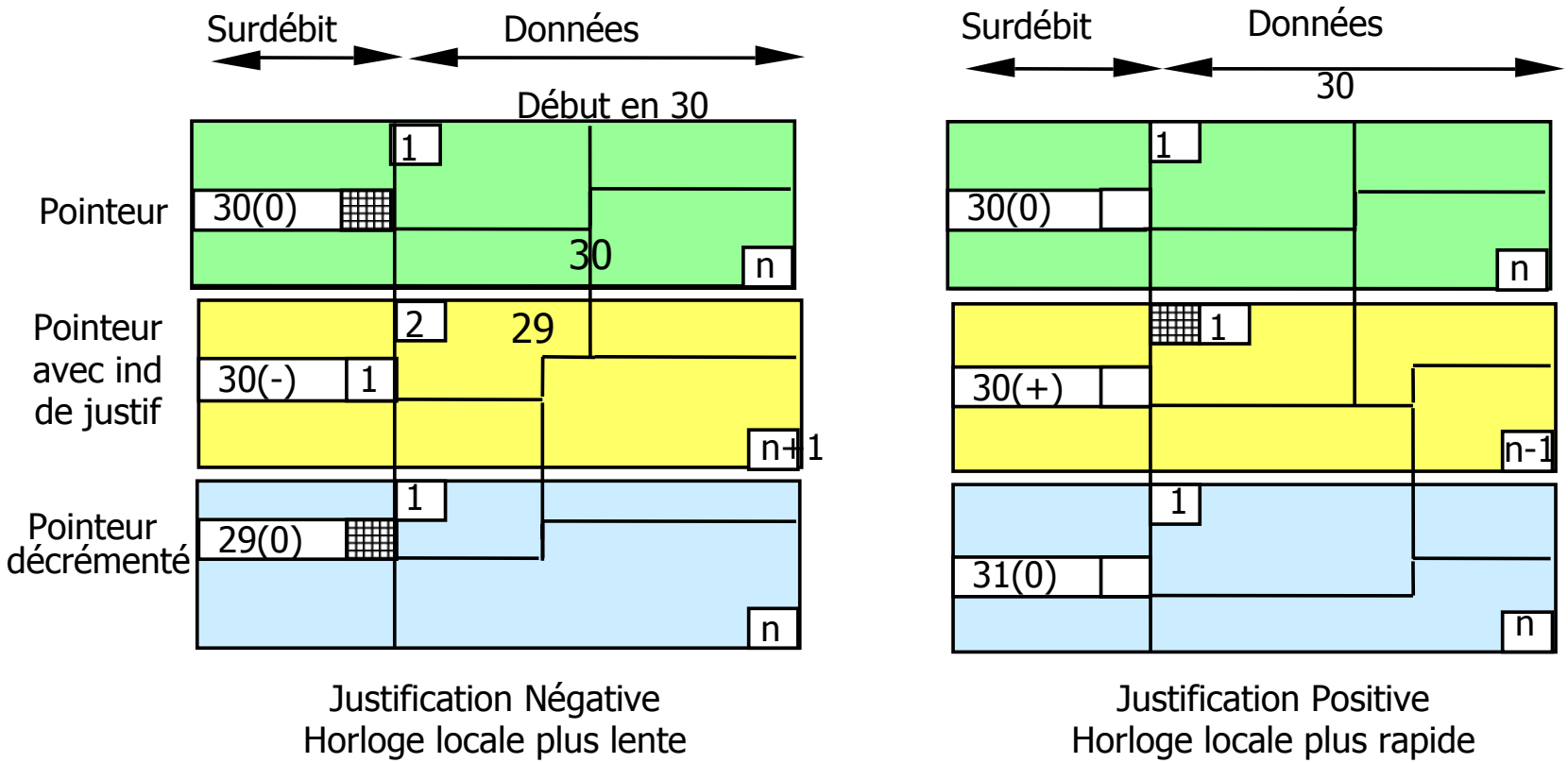
- **Présentation de la justification dans le cas d'un seul bit** : on ajoute ou l'on retranche un seul bit par trame.
  - **CP:** bit de contrôle de justification positive
  - **CN:** bit de contrôle de justification négative
  - **JP:** bit de justification positive
  - **JN:** bit de justification négative



Utilisation des pointeurs

- **Situation du problème :**
  - Quand des trames successives sont soumises à des justifications **successives de même sens**
  - => **Les délimitations entre les zones transportées** glissent à l'intérieur des conteneurs de charge utile
- **Utilisation de pointeurs.**
  - **Un pointeur est associé à la charge utile** de la trame.
  - **Sa valeur ne change pas** tant que la position relative de cette charge utile dans la trame ne varie pas.
  - **Le début des informations** significatives peut avancer ou être retardé par rapport à la valeur initiale => Modification du pointeur
  - **La charge utile peut "flotter"** dans l'espace alloué dans les trames.
  - **Pointeur placé dans une entête protocolaire** : **le surdébit**.
  - **Permettant d'accéder aisément** (indirection) aux infos transportées.

Fonctionnement des pointeurs en SDH



- **Remarque:** Présentation comme en SDH, la justification porte sur des octets et les trames ont une représentation matricielle.
- **On montre trois trames** successives dans les deux cas de justification.

Explications fonctionnement des pointeurs

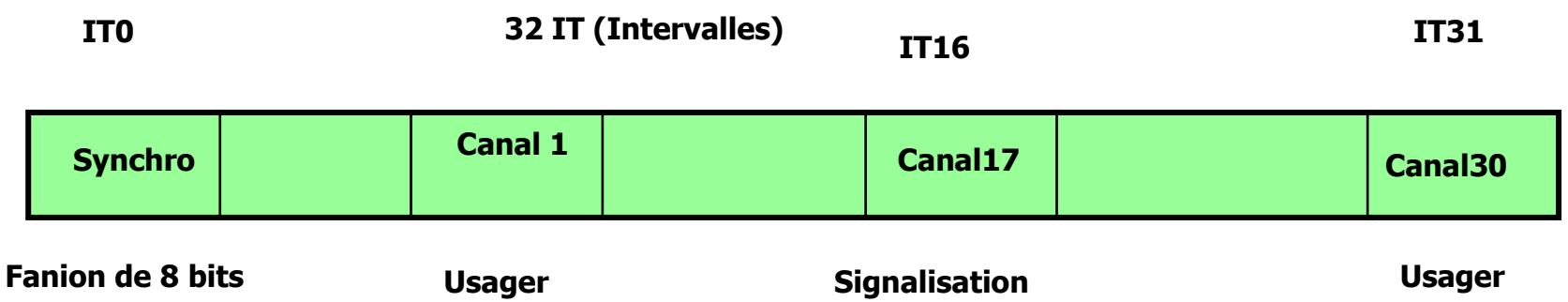
- **Si l'horloge locale a une fréquence plus faible** que celle du site d'origine du signal transporté:
  - On associe au pointeur une **indication de justification négative (-)**.
  - **Un octet de donnée est retranché vers le surdébit**.
  - **Dans la trame suivante** la valeur du pointeur est diminuée de 1.
- **Si l'horloge locale a une fréquence plus élevée**
  - On associe au pointeur une **indication de justification positive (+).**
  - **Un octet de bourrage est inséré** dans la charge utile.
  - **Dans la trame suivante** le pointeur est augmenté de 1.

PDH "Plesiochronous Digital Hierarchy"

- **Hiérarchie numérique plésiochrone :** solution maintenant un peu ancienne.
- **Solution de niveau physique et liaison :**
  - Modulation, synchronisation bit et trame en présence de délais de propagations et décalages d'horloge (techniques de justification).
  - Réalisant le multiplexage temporel de voies téléphoniques MIC (64kb/s).
- **Exemple de normes et de débits :**

Trame	Débit	Nb circuits	Avis CCITT
TN1 – E1	2 048 Kb/s	30	G.704, G.706
TN2 – E2	8 448 Kb/s	120	G.741, G.742
TN3 – E3	34 368 Kb/s	480	G.751
TN4 – E4	139264 Kb/s	1920	G.751

Exemple PDH : le multiplex E1 à 2 048 Kb/s G704 (TN1)



- **Surdébit:**
  - Nécessaire au maintien et à la récupération du synchronisme: l'IT0 acheminé dans le premier octet entête.
  - Utilisé en version multitrame il offre des possibilités de transmission d'informations multiples et complexes.
- **Débit utile:** on a 30 voies téléphoniques MIC et une voie IT16 permettant l'acheminement de la signalisation pour les 30 voies.

G704
Utilisation du surdébit (IT0)

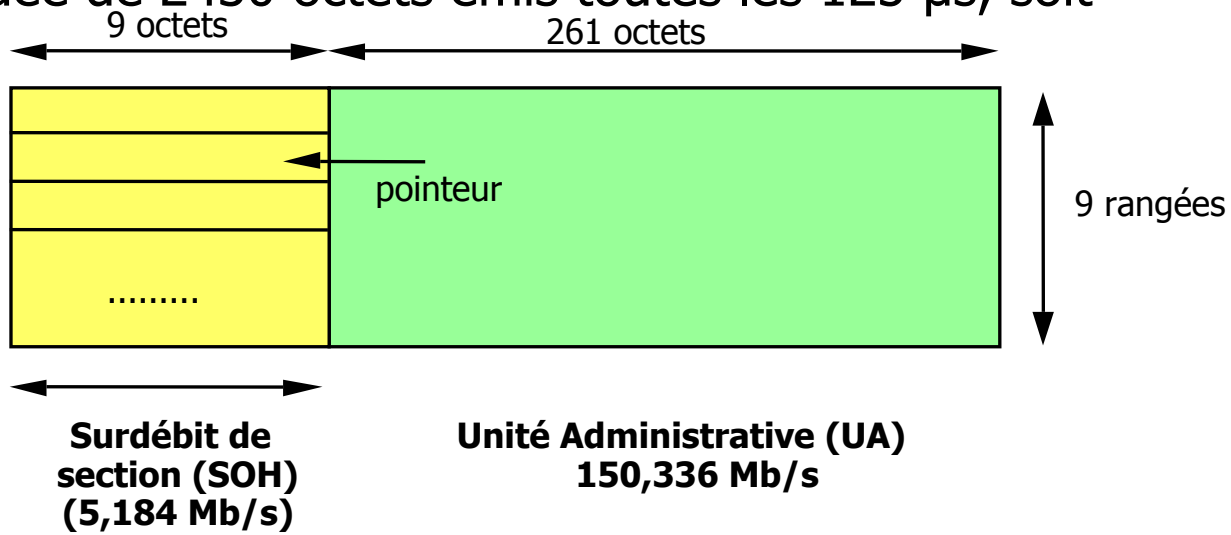
- **En regroupant 16 trames** (multi-trame) on dispose de 128 bits chaque 2 millisecondes utilisés pour de nombreuses fonctions:
  - **Une zone de verrouillage de trame** pour définir le début des trames (la synchronisation trame).
  - **Une zone pour la justification bit**.
  - **Une zone pour des signaux d'alarme** (en cas de mauvais fonctionnement).
  - **Une zone de contrôle de qualité : CRC-4**.
  - **Des zones libres d'usage spécifiques** (par exemple à caractère national).

Introduction aux réseaux SDH Synchronous Digital Hierarchy

- **Problèmes de la hiérarchie PDH**
  - **Manque de flexibilité** de la hiérarchie plésiochrone: pour accéder à une donnée dans un multiplex il faut démultiplexer tous les niveaux.
  - **Développement à partir de 1980** d'une nouvelle technologie:
  - **SONET Synchronous Optical Network** (Bellcore) **=> SDH.**
- **Originalité synchrone :** tous les noeuds d'un réseau utilisant la hiérarchie numérique synchrone sont **synchronisés**
  - Horloges de référence d'une précision spécifiée à **10-11.**
  - On a encore des dérives sensibles (utilisation à très haut débit).
- **Originalité pointeurs :** la technique **des pointeurs**
  - Permet de compenser les problèmes de variation de fréquence du signal
  - Mais aussi d'accéder directement aux données à l'intérieur des trames par des mécanismes d'indexation simples ...

Exemple STM-1 "Synchronous Transport Module 1"

- **Multiplex primaire de la hiérarchie SDH**
- **Trame de base** constituée de 2430 octets émis toutes les 125 µs, soit 155,520 Mb/s.
- **Organisée** en 9 rangées de 270 octets.



- **Dans une trame STM-1**, les informations sont placées dans des conteneurs qui peuvent être vus comme une structure hiérarchisée de groupage.
  - Le conteneur + son surdébit forment un conteneur virtuel (VCn).
  - STM1 -> Un conteneur VC4 ou plusieurs conteneurs plus petits: VC-1 (1,544 à 2, 048 Mb/s), VC-3 (34,368 à 44,736 Mb/s).

Conclusion SDH

- **Utilisation de SDH très importante et très variée**
  - Pratiquement toutes les Transmissions à longue distance.
  - Réseaux téléphoniques (RTC et mobiles)
  - Transport de cellules ATM, de paquets IP (IP sur SDH).
  - Niveau physique pour Ethernet 10 Gigabits/s.

Trame	Débit
STM-1 (OC3)	155,520 Mb/s
STM-4 (OC12)	622,080 Mb/s
STM-16 (OC48)	2488,320 Mb/s
STM-64 (OC192)	9953,280 Mb/s
STM-128 (OC384)	20 Gb/s
STM-256 (OC768)	40 Gb/s

- **Construction d'architectures de réseaux SDH**
  - Avec des répartiteurs, des brasseurs, des multiplexeurs.
  - Topologie en boucle SDH privilégiée.
- **La hiérarchie SDH est encore en développement.**

Niveau Liaison
"Data link Layer"

Protocoles Point à Point

Réseaux locaux

Niveau Liaison
"Data link Layer"

Protocoles Point à Point
"Point to Point Protocols"

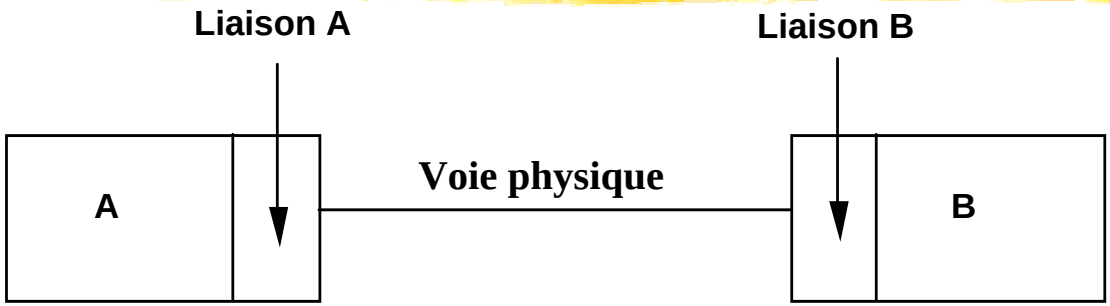
Introduction.
I Solutions générales.
II Protocoles industriels
Conclusion.

Niveau Liaison En Point à point

Introduction

Problèmes résolus dans les protocoles de niveau liaison en point à point

Situation du sujet: le niveau liaison point à point



- **Le niveau liaison contrôle les communications** réalisées au niveau physique entre des sites reliés par une voie point à point (implantation logicielle).
- **Le niveau liaison masque les caractéristiques** de la couche physique (tous types de voies) aux entités de réseau.

  Exemple type: PPP dans l'Internet
- **Le niveau liaison point à point résout des problèmes non résolus au niveau physique** (transparent suivant).
- **Niveau liaison point à point: très stabilisé** (théorie 1960/1970, normes 1970/1990)

Principaux problèmes résolus dans le niveau liaison point à point

Selon les choix de conception on trouve **tout ou partie** de solutions au problèmes **suivants :**

- 1 **Délimitation** des trames.
- 2 Contrôle **d'erreurs.**
- 3 Contrôle **de flux.**
- 4 Contrôle **de séquence.**
- 5 Gestion **des connexions.**
- 6 **Multiplexage.**
- 7 Fonctions **d'administration.**

1 - Délimitation des trames ("tramage", "framing").

- **Fonction basique de tous les protocoles de liaison : acheminer des trames** (des suites binaires structurées).
  - **Nécessite** de retrouver la **synchronisation trame** en relation avec **les erreurs** (trames bruitées).
- **Fonction considérée comme de niveau liaison**
  - Synchronisation bit => niveau physique (horloges).
  - Synchronisation trame => plutôt niveau liaison (délimiteurs)
- **Exemple type :** Fonction quasi unique dans les protocoles de liaison sans connexion
  - Un seul type de trame existe qui doit essentiellement être délimité

    => Protocole **SLIP** ("Serial Line Interface Protocol")

2 - Détection et correction des erreurs : A) Contrôle d'erreurs ("Error control")

- **Bruits au niveau physique** : Solution niveau liaison
  => codes détecteurs d'erreurs.

- **A) Si le taux d'erreurs est inacceptable**

Taux d'erreur de la voie physique : exemple 10^{-3} à 10^{-5} par bits.

=> Le protocole de liaison amène **le taux d'erreur résiduel à un niveau acceptable** au moyen de retransmissions (exemple > 10^{-12} par bits).

=> Protocoles avec contrôle d'erreurs : **LAPB** ('Link Access Protocol B'), IEEE 802.11 (**WiFi** "Wireless Fidelity").

2 - Détection et correction des erreurs : B) Destruction silencieuse

- **B) Si le taux d'erreurs du au bruit est considéré comme acceptable** :

  - **Pas de contrôle d'erreur** (le contrôle d'erreur est renvoyé au niveau transport).
  - => **Destruction silencieuse** des trames erronées ("Silent Discard").
  - => **Exemple type : PPP** ('Point to Point Protocol' en mode standard, **Ethernet** IEEE802.3).

3 - Contrôle de flux ('Flow Control')

- **Cas 1 : Contrôle de flux jugé indispensable**
  - **Problème : adapter la vitesse de l'émetteur à celle du récepteur**.
  - **Si le débit d'un émetteur est trop important** relativement aux performances d'un récepteur
  - On doit **réguler** les échanges : ralentir l'émetteur pour éviter les **pertes** de trames dues à **l'impossibilité de stocker** les trames entrantes en cas de **trafic élevé**.
  - => Exemples : Protocoles **LAPB**, **IEEE 802.3x** (Contrôle de flux en Ethernet)

- **Cas 2 : Si les pertes de trames d'un récepteur sont jugées acceptables :** pas de contrôle de flux

  => Exemples : Protocoles **SLIP**, **PPP** (mode standard).

4 - Livraison en séquence (respect de la causalité)

- **Voie physique : médium 'causal' :**
  - Causalité de propagation des ondes: les bits ne peuvent **remonter le temps** (arriver avant d'être émis) ou se **dépasser** sur les câbles**.**
- **Problème:** les erreurs de transmission et les retransmissions produisent des **modifications de la séquence émise (lacunes, déséquencements)** ou des **duplications.**
  - Le contrôle de séquence assure, en liaison avec le contrôle d'erreur et le contrôle de flux, la délivrance des trames au récepteur dans l'ordre **exact** de la soumission par l'émetteur
- **Exemples :**
  - Protocoles de liaison avec contrôle de séquence : Protocoles **LAPB**, **IEEE 802.11 Wifi.**
  - Protocole sans contrôle de séquence : **SLIP**, **PPP** en standard.

5 - Gestion des connexions

Protocoles en mode connecté

=> Identification de flots séparés par échanges préalables de messages de connexion (à la fin messages de déconnexion).

- Exemples de mode connecté **BSC** (Binary synchornous Communications) **, LAPB, LLC2** (Logical Link Control 2), **PPP**.

Protocoles en mode non connecté

=> Les échanges ont lieu avec des trames de données qui peuvent être transmises à tous moments.

- Exemples d'implantation du mode sans connexion: **SLIP**, **LLC1** (Logical Link Control 1 sur réseau local), **Ethernet**, **Wifi**.

6 - Multiplexage

- **Multiplexage** : gestion de plusieurs flots de données identifiés et délivrés à des entités de réseau différentes.
  - **Utilisation 1 :** Coexistence de trafics pour des architectures de réseaux différentes sur la même voie physique :

    Exemples : Coexistence Internet, Novell, Apple Talk , SNA, OSI ….
  - **Utilisation 2** : coexistence d'un trafic pour des entités de réseaux différentes dans une même architecture:

    Exemples : En Internet protocoles ICMP, ARP , RARP…
- **Solution statique** : Utilisation d'une zone identifiant les différents flots selon des valeurs fixes (normalisées).
- **Exemples : PPP** , **Ethernet/DIX** (Digital/Intel/Xerox), **LLC/SNAP** (Logical Link Control/Subnetwork Access Protocol)

7 - Administration du niveau liaison

- **Fonctions d'administration ('Network Management')** : généralement selon une approche normalisée et par niveaux
  - Exemple : **SNMP** 'Simple Network Management Protocol'.
- **Cas du niveau liaison** : des fonctions spécifiques sont définies dans les protocoles (réglages, adressage, …)
- **Essentiellement valable pour le protocole PPP Internet:** Un protocole qui développe beaucoup cet aspect
  - **Gestion des configurations** : négociation de paramètres de fonctionnement, fourniture d'adresses.
  - **Gestion des pannes** : mécanismes de diagnostic (bouclage).
  - **Gestion de la sécurité** : authentification d'un nouveau client.
  - **Gestion des performances** : pas d'exemple au niveau liaison.
  - **Gestion de facturation des ressources** : pas d'exemple de liaison.

Niveau Liaison En Point à point

Chapitre I

Solutions générales pour la réalisation des protocoles de liaison en point à point

Délimitation des trames

Contrôle d'erreur, de flux, de séquence

I.1
Délimitation des trames

Délimitation des trames
Position du problème

- **Retrouver la synchronisation trame :**
  - **mise en correspondance "une pour une**"
  - entre une trame émise et une trame reçue.
- **Difficultés du problème**
  - Le bruit peut faire perdre la synchronisation trame.
  - Le bruit peut créer des trames fictives (à une trame émise peut correspondre une trame ou plusieurs trames reçues en raison du bruit découpant ou créant des trames parasites).
- **La solution de base:**
  - Se donner un mécanisme de **séparation des trames**.
  - Associer à une trame **un code polynomial détecteur d'erreur.**
  - Quand on décide qu'**une trame est délimitée en réception** on vérifie la **correction** au moyen du code polynomial.
  - Si elle est **incorrecte ou séparée** en plusieurs trames incorrectes **on abandonne les informations reçues** (destruction silencieuse).

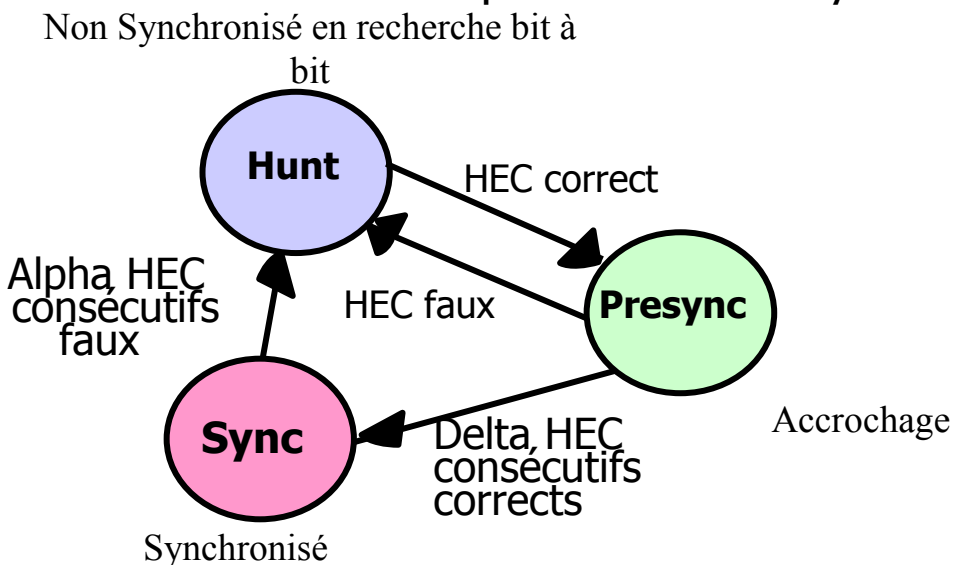
Trames sans délimiteurs (dans un flux continu synchrone)

- **Solution utilisée pour des trames de longueur fixe en flux continu:**
  - Pour chaque trame : un code polynomial correct **se répète en position fixe**.
  - On fait fonctionner un automate qui se **recale sur chaque bit considéré comme un début de trame** et vérifie l'exactitude du code polynomial.
  - **Approche probabiliste :** On peut se recaler une fois par hasard mais pour delta trames successives correctes => très faible risque d'erreur de synchro.

  - **Exemple ATM**

- Cellules de 53 octets

- HEC : Header Error Control (code polynomial)



- **Pour des trames de longueur variable (pas d'exemple industriel)**
  - Difficultés supplémentaires

Trames avec délimiteurs: Transparence caractère ('Character Stuffing')

- **Trames constituées de caractères** d'un alphabet normalisé (ex ASCII,EBCDIC) => les trames sont multiples de 8 bits.
- **Caractères de contrôle** : des codes caractères particuliers.
- **Solution retenue en** : **BSC**, **SLIP**, **PPP** mode standard.

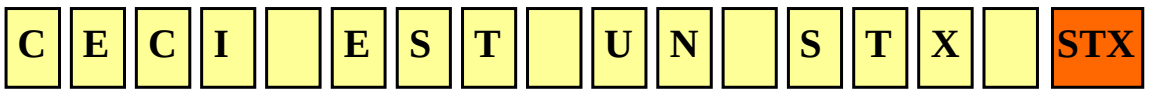
Exemple dans les protocoles BSC ('Binary Synchronous Communication')

- **STX** ("**S**tart of **T**e**X**t") - Délimiteur début de bloc de texte
- **ETX** ("**E**nd of **T**e**X**t") - Délimiteur fin de bloc de texte
- **DLE** ("**D**ata **L**ink **E**scape") - Échappement de liaison

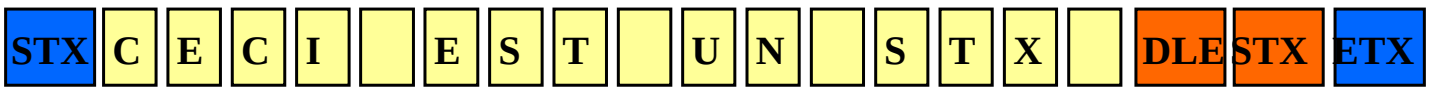
- **Pour une trame purement alphanumérique: pas de problème d'ambiguïté** entre caractères de contrôle et données dans les blocs texte.
- **Si une trame comporte des caractères de contrôle** parmi des caractères alphanumériques d'un bloc => **transparence caractère**
  - Tout caractère de contrôle (qui n'est pas le délimiteur début ou fin) apparaissant dans le bloc est précédé de DLE.
  - ETX -> DLE ETX; STX -> DLE STX; DLE -> DLE DLE
  - A la réception les DLE ajoutés pour la transparence sont retirés.

Exemple de transparence caractère en BSC

- **Bloc à transmettre :**



- **Bloc transmis** :



- **Pas de problème :** pour restituer le bloc initial
  - Pour distinguer les délimiteurs réels en début et fin de bloc (sans DLE devant)
  - Pour distinguer les codes caractères identiques transmis dans la charge utile (ils sont précédés d'un DLE que l'on supprime à la réception).

Trames avec délimiteurs: Transparence binaire ('Bit Stuffing')

- **Les trames sont constituées de suites binaires**.
- **Chaque trame est délimitée** par une suite binaire réservée.

- **Exemple de la famille des protocoles SDLC/HDLC/LAPB:**
  Délimiteur : drapeau, fanion ou "flag" huit bits **01111110**.
- **Le fanion ne doit jamais apparaître** dans la suite binaire d'une trame sous peine de décider la fin de trame.
- **Procédure de transparence binaire**

  - En sortie quand on a **5 bits 1 consécutifs** => on insère automatiquement **un bit 0** après

  - **En entrée le bit 0 suivant 5 bit 1 est enlevé** (sauf bien sur pour les fanions début et fin qui sont interprétés comme des délimiteurs).

Exemple de transparence binaire en SDLC/HDLC/LAPB

- **Suite binaire à transmettre :**

01001000001111101111100

- **Suite binaire après adjonction des bits de transparence :**

01001000001111101011111000

- **Trame transmise avec ses délimiteurs (fanions) :**

01111110	01001000001111101011111000	01111110

- **En réception : suppression des bits de transparence et des fanions.**

Trames avec délimiteurs: Violation de code

- **Problèmes des techniques de transparence** basées sur l'utilisation de délimiteurs formés de **configurations binaires légales** (caractères de contrôle, fanions, ...).
  - **Allongement des trames** du aux informations de transparence (quelques pour cent).
  - **Temps perdu à l'émission et à la réception** (surtout en logiciel).
- **Autres solutions :** définir des modulations utilisées comme délimiteurs (en plus de la modulation des données trame 0,1) .
  - **Augmenter la valence** des signaux de niveau physique pour créer des délimiteurs.
  - Ces signaux **ne peuvent donc apparaître dans le flot normal** des données d'une trame.
- **De nombreuse variantes** de ce principe ont été développées.
  - Les problèmes de la transparence sont résolus.
  - Mais le modulateur doit être plus complexe.
  - Solution très utilisée (par exemple dans les réseaux locaux).

Exemple de délimiteurs dans le protocole IEEE 802.3u Ethernet 100 Base T

- **Code 4B/5B**
  - Données binaires utilisateur: groupes de 4 bits
  - Données transmises: pour chaque groupe de 4 bits un groupe de 5 bits.
  - 16 configurations nécessaires pour les données utilisateurs.
  - 16 configurations disponibles pour des besoins protocolaires : symboles inter trames, délimiteurs etc
- **Délimiteurs**
  - Symbole J 11000 First Start Of Frame (Invalid code)
  - Symbole K 10001 Second Start of Frame (Invalid code)
  - Symbole T 01101 First End Of Frame (Invalid code)
  - Symbole R 00111 Second End of Frame (Invalid code)
  - Symbole I 11111 Idle (Invalid code)
- **Structure d'une trame**

I I I I | J K ..... Groupes données de trame ..... T R | I

- **Solutions en FDDI, Ethernet 1000 , 10G : très similaires.**

I.2
Contrôle d'erreur, de flux, de séquence

Introduction : présentation générale des protocoles étudiés

- **Solutions de complexité croissante** aux problèmes:
  - Contrôle d'**erreur.**
  - Contrôle de **flux**
  - Respect de la **causalité** (livraison en séquence).
- **Traitement conjoint des problèmes**
- **Codage des solutions en langage style ADA**
- **Protocoles examinés** (selon l'approche de A. Tannenbaum)

- Protocole 1 **"Sans contrôle d'erreur et de flux"**
- Protocole 2 **"Envoyer et attendre"**
- Protocole 3 **"Bit alterné"**
- Protocole 4 **"A fenêtre glissante et réception ordonnée"**
- Protocole 5 **"A fenêtre glissante et rejet sélectif"**

Protocole 1 : Sans contrôle d'erreur ni contrôle de flux

- **Pas de contrôle de flux**
  - **La couche réseau du récepteur est toujours prête à recevoir:** les pertes de trames dues au manque de contrôle de flux sont négligeables (temps de calcul négligeables, mémoire toujours disponible pour stocker).
  - **Ou le contrôle de flux est prévu ailleurs** (assuré dans un autre niveau).
- **Pas de contrôle d'erreur**
  - **Les pertes de trames dues au bruit sont tolérables**.
  - **Ou le contrôle d'erreur est prévu ailleurs** (assuré dans un autre niveau).

- **Nature de la solution**
- **Solution de base** d'un protocole sans connexion qui se contente d'acheminer des trames et laisse aux niveaux supérieurs toutes les tâches.
- **Mise en place de la programmation** pour les solutions suivantes.
- Le code ne décrit qu'une transmission **unidirectionnelle.**

- **Solution des protocoles Internet :** SLIP, PPP en mode standard.

Protocole 1 : Déclarations

```
-- Zone de données utilisateur (paquet réseau)
-- Par exemple: taille de paquet 1500 octets.
type paquet is array (integer range 1..1500) of character ;
-- Type de trame de niveau liaison utilisée.
-- Une seule information : la charge utile (le paquet).
type trame is record
     info : paquet ;
end record;
-- Type événement en entrée.
-- Un seul événement : l'arrivée d'une trame
type Type_Evénement = (Arrivée_Trame ) ;
```

Protocole 1 : Codage de l'émetteur

```
procedure émetteur_1 is
s               : trame ;      -- La trame liaison en émission
tampon          : paquet ;     -- Le paquet réseau à émettre
begin
   loop
   recevoir_couche_réseau (tampon) ;    -- Paquet à envoyer
   s.info := tampon ;                   -- Préparer une trame
   envoyer_couche_physique(s) ; -- La faire émettre
   end loop                             -- Boucle infinie
end emetteur_1 ;
```

Protocole 1 : Codage du récepteur

```
--
-- Procédure exécutée par le récepteur
--
procedure récepteur_1 is
événement    : Type_Evénement ;-- Événement à traiter;
r            : trame ;          -- La trame en réception
begin
   loop
   attendre (événement) ;       -- Attendre une arrivée
   recevoir_couche_physique (r) ;-- Prendre trame arrivée
   envoyer_couche_réseau(r.info);-- Passer à l'utilisateur
   end loop                     -- Boucle infinie
end récepteur_1;
```

Protocole 2 : Arrêt et attente ("Stop and Wait")

- **Solution simpliste uniquement de contrôle de flux**.

- **Idée de base :** pour ne pas saturer le récepteur **freiner l'émetteur**.
- **Solution de rétroaction du récepteur sur l'émetteur**:

  . Le récepteur informe l'émetteur qu'il peut **accepter un nouvelle trame** en envoyant une trame de service ne contenant pas de données.

  . Cette trame s'appelle en réseau **un crédit (CDT)** : un crédit d'une unité donne un droit pour émettre une nouvelle trame.

  . L'émetteur **doit attendre d'avoir un crédit** pour envoyer une trame.

  => Famille des solutions de contrôle de flux: **Solutions basées crédits.**

Protocole 2 : Codage de l'émetteur

```
--
-- Code présenté => Solution unidirectionnelle
-- Une voie de retour pour des trames de service (toujours des crédits).
-- Déclarations : pas de changement par rapport au protocole 1
--
procedure émetteur_2 is
-- Les variations par rapport au code du protocole 1 sont en italique.
événement : Type_Evénement ;          -- Un événement à traiter
s         : trame ;                   -- Trame en émission
tampon    : paquet ;                  -- Paquet à émettre
begin
    loop
    recevoir_couche_réseau (tampon) ; -- Un tampon à envoyer
    s.info := tampon ;                -- Préparer une trame
    envoyer_couche_physique(s) ;      -- La faire émettre
    attendre(événement) ;             -- Attendre un crédit
    end loop                          -- Boucle infinie
end émetteur_2 ;
```

Protocole 2 : Codage du récepteur

```
--
procedure récepteur_2 is
--
-- Les variations par rapport au code du protocole 1 sont en italique
événement: Type_Evénement;          -- Evénement à traiter
r          : trame;                 -- Une trame en réception
s          :trame;                  --Une trame de crédit
begin
    loop
    attendre (événement) ;          -- Attendre arrivée de trame
    recevoir_couche_physique (r);   -- Prendre trame arrivée
    envoyer_couche_réseau(r.info) ; -- Passer à l'utilisateur
    envoyer_couche_physique(s);     -- Envoyer le crédit
    end loop                        -- Boucle infinie
end récepteur_2;
```

Protocole 3 : PAR "Protocole avec Acquittement et Retransmission"

- **Solution globale: problèmes de contrôle de flux, d'erreur, de séquence**
  - Pour une voie physique **bruitée** (utilisation d'un code détecteur d'erreur)
  - et avec un récepteur à **capacité de traitement** et de **mémoire limitée.**

- **Nombreuses appellations pour une classe de protocoles voisins.**
  - Protocole **PAR ("Positive Acknowledgment with Retry").**
  - Protocole du **Bit alterné ABP ("Alternate bit protocol").**
  - Protocole **ARQ ("Automatic Repeat Request").**

- **Introduction des notions suivantes**
  - A) **Acquittement positif** (d'une trame correcte).
  - B) **Délai de garde** (pour retransmission sur erreur).
  - C) **Identification des trames** (par un numéro de séquence).

- **Exemples d'implantation de protocoles de ce type :** BSC 'Binary Synchronous Communications', GSM 'Global System for Mobile', WIFI.

A) Acquittements positifs "Positive Acknowledgments"

- **Acquittement positif** : une information protocolaire qui circule pour indiquer la bonne réception d'une trame de donnée.
- **Utilisation des acquittements positifs**

Règle 1 : Une trame n'est **acquittée positivement** que si elle est **reçue correctement** (code détecteur correct).

Règle 2 : Toute **trame correcte doit être acquittée positivement** afin que l'émetteur ne la retransmette plus.

- **Remarques**

- Stratégie de reprise : en acquittement positif la reprise sur erreur est **confiée à l'émetteur** qui doit s'assurer qu'une trame a bien été reçue.

- Acquittements positifs et crédits

=> L'acquittement positif a une signification dans **le contrôle d'erreur** alors que le crédit sert dans le **contrôle de flux**.

=> Une trame unique (baptisée acquittement "ACK") est souvent utilisée (exemple dans le protocole PAR) à double sens pour signifier :

. **dernière trame correcte** (acquittement positif)

. **acceptation d'une autre trame** (crédit de une trame).

Acquittements négatifs "Negative Acknowledgments"

- **Acquittement négatif** : une information protocolaire indiquant la mauvaise réception d'une trame de donnée.
- **Utilisation des acquittements négatifs**

- **Règle 1** : Une trame n'est acquittée négativement que si le destinataire ne l'a pas reçue alors qu'il sait qu'elle a été émise.

Apprentissage : . Signal indiquant une trame en erreur (rare).
. Absence d'une trame dans une suite numérotée.

- **Règle2** : Un acquittement négatif est une demande de retransmission.

- **Remarques:**

- **Stratégie de reprise : en acquittements négatifs** le traitement des erreurs est confié au récepteur qui doit découvrir l'absence d'une trame, en demander la retransmission pour que celle-ci soit bien reçue.
- **Le protocole PAR n`a pas besoin** des acquittements négatifs.
- On peut concevoir de **multiples variantes** de protocoles utilisant à la fois des stratégies d'acquittements négatifs et positifs.
- Des protocoles probabilistes **basés uniquement sur** les acquittements négatifs sont possibles.

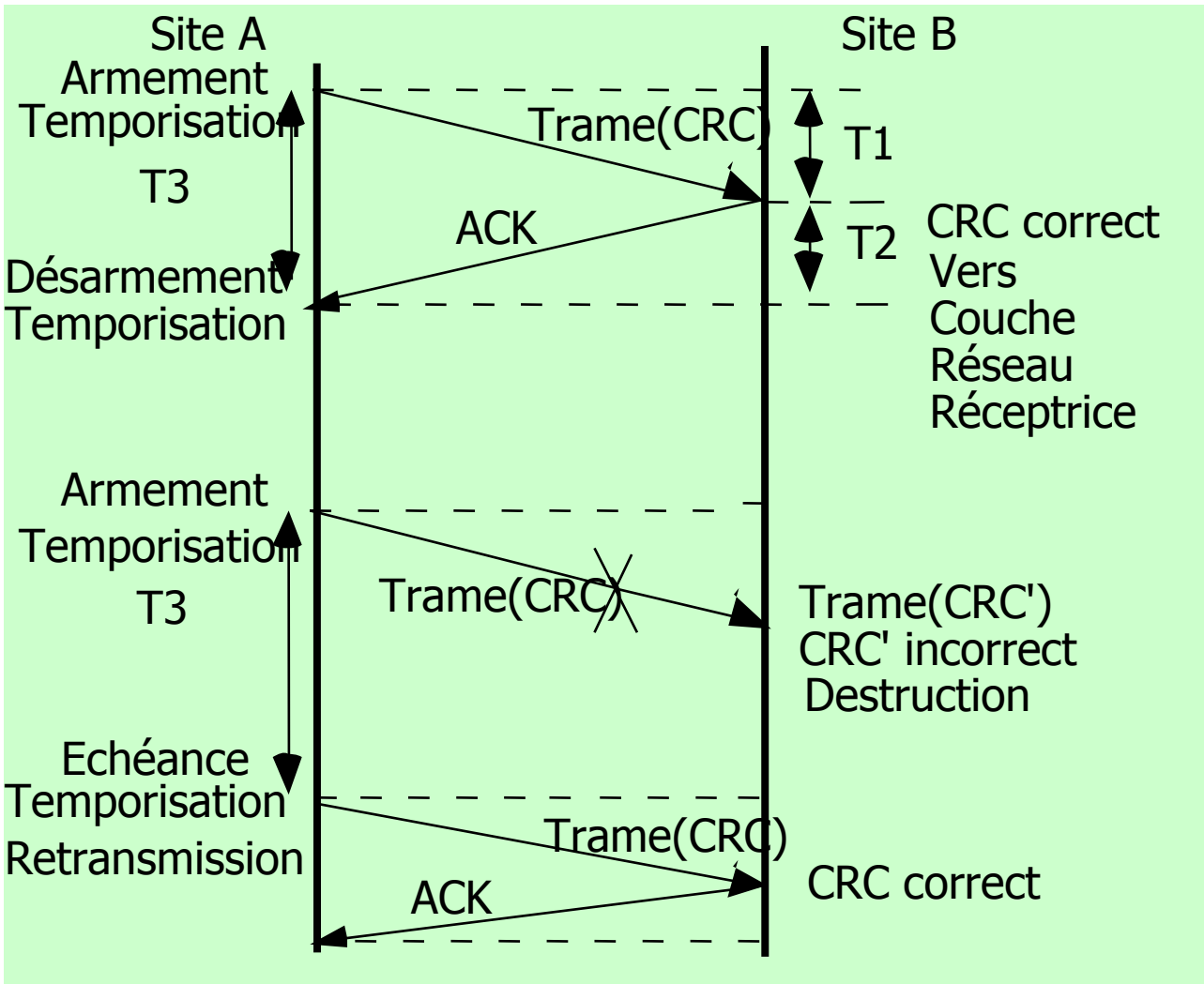
B) Délais de garde : Temporisateurs , "Timers"

- Nécessité de **conserver copie d'une trame** .

  . Si la trame est bruitée on doit la retransmettre.

  . Si la trame est correcte mais l'acquittement positif bruité on ne sait pas si la trame a été reçue => on retransmet.
- Problème: en l'absence d'acquittement positif on ne peut conserver **indéfiniment** les copies.
- Utilisation **d'un délai de garde** (temporisateur)
  - **Réveil de l'émetteur à échéance.**
  - **Retransmission de la trame**
- Protocole PAR : seul mécanisme de reprise, l'émetteur **retransmet systématiquement une trame** sur échéance du délai.
- **Remarque** : L'usage du délai de garde ralentit la reprise sur erreur car le délai de garde doit être nettement supérieur à la somme des délais d'émission et de retour d'acquittement.

Fonctionnement avec délais de garde (temporisateurs)

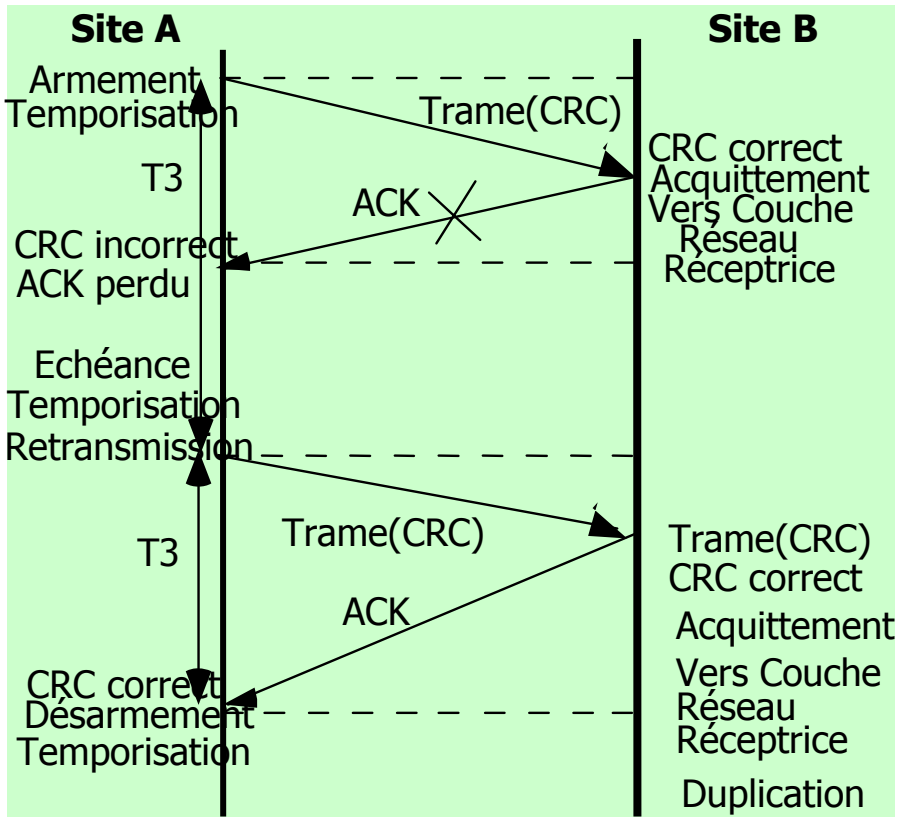


C) Identification des trames : numéros de séquence ("Sequence numbers")

Exemple de problème sans identifiant

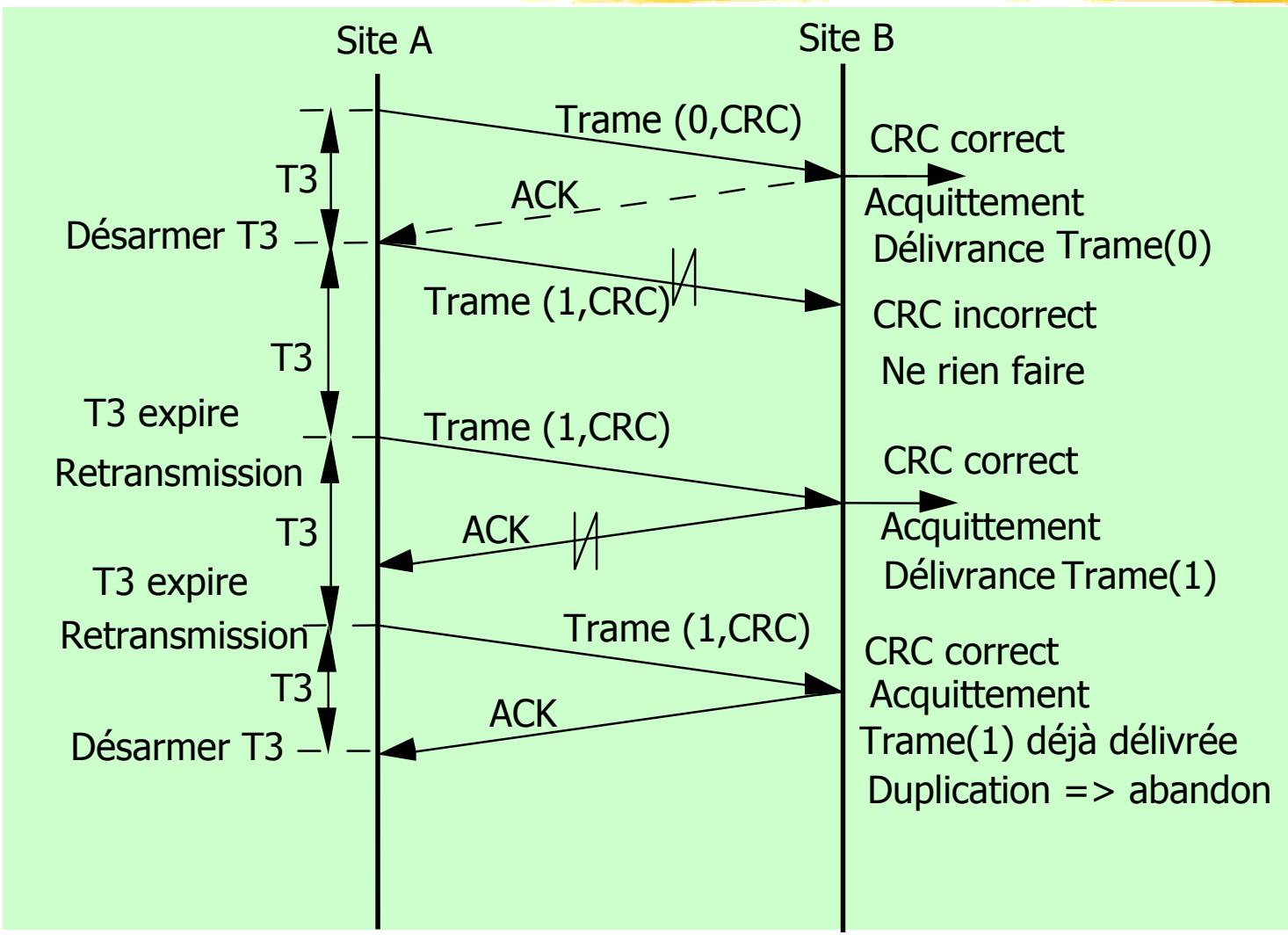
- Une trame est reçue **correctement** mais l'**acquittement positif correspondant se perd ...**
- La couche liaison récepteur reçoit **deux fois la trame** puisque l'émetteur réémet.

=> **Duplication non détectée (non respect de la séquence)**

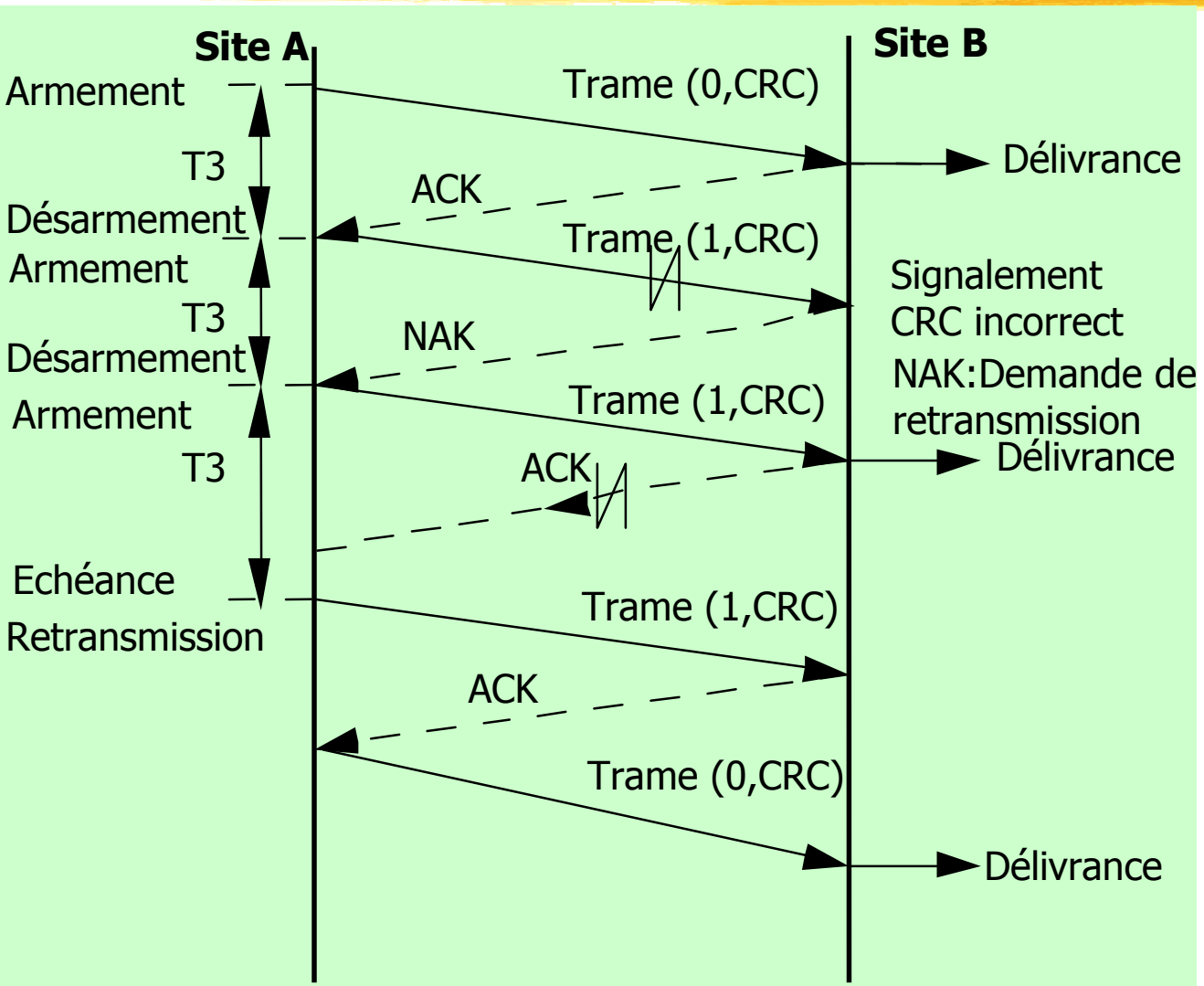


- Nécessité **d'un numéro de séquence** de trame (identifiant entier de la trame) pour éviter les **duplications** et assurer le **respect de la séquence d'émission** (causalité).

Exemple de fonctionnement du protocole PAR



Fonctionnement en utilisant en plus des acquittements négatifs



- Les acquittements négatifs **ne sont pas indispensables** car le fonctionnement de base est fourni par **les acquittements positifs, les temporisateurs et les numéros de séquence**.
- Les acquittements négatifs, servent à **accélérer les retransmissions** en cas d'erreur.

Protocole 3 PAR : Codage
Présentation de la solution codée

- **Utilisation unique des acquittements positifs**.
- Utilisation d'un **délai de garde** (fonctions d'armement et de désarmement d'un temporisateur).
- **Identification des trames** sur un bit (numéros de séquence 0 et 1) => Protocole de bit alterné.
- Solution **unidirectionnelle** (un seul sens est décrit)
- Utilisation d'une voie de retour pour la **circulation des acquittements positifs.**

Protocole 3 PAR : Codage des déclarations

```
--
-- Variations par rapport au protocole 2 en italique
--
-- Déclarations globales
--
-- Type numéro de séquence d'amplitude 0..maxseq=1
maxseq: constant :=1;
type numero_sequence is integer range 0..maxseq;
type paquet is array ( integer range 1..128 ) of character ;
type trame is record
        seq : numero_seq ;
        info : paquet ;
end record;
type Type_Evénement = (Arrivée_Trame, Erreur_Trame, Horloge);
```

Protocole 3 PAR : Codage de l'émetteur

```
procedure emetteur_3 is
événement          :Type_Evénement;          -- Evénement à  traiter;
s                  :trame;                   -- Trame en émission;
tampon             :paquet;                  -- Paquet à émettre
Proch_Trame_A_Envoyer  :  numero_sequence;   -- Num prochaine trame émise
begin
Proch_Trame_A_Envoyer := 0;                  -- Init pour le premier message
recevoir_couche_réseau (tampon);             -- Un tampon est à envoyer
loop
    s.seq := Proch_Trame_A_Envoyer ;         -- Préparer numéro  de trame
    s.info := tampon ;                       -- Partie information usager
    envoyer_couche_physique(s) ;             -- Faire partir la trame
    démarrer_horloge (s.seq) ;               -- Armer un délai de garde
    attendre(événement) ;                    -- Attendre un crédit / un acquit
    if événement = arrivée_trame then        -- C'est l'acquittement attendu
            désarmer_horloge(s.seq) ;        -- Plus besoin d'un réveil
            inc(Proch_Trame_A_Envoyer:);     -- +1 pour le prochain message
            recevoir_couche_réseau (tampon); -- Un tampon est à envoyer
    endif   -- Cas d'une retombée de garde : on ne fait rien donc on réemet
end loop                                     -- Boucle infinie
end emetteur_3;
```

Protocole 3 PAR : Codage du récepteur

```
procedure récepteur_3 is
événement: Type_Evénement;          -- Un événement à traiter;
r          : trame;                 -- Une trame en réception
s          : trame;                 -- Une trame de crédit /acquit (ack)
trame_Attendue: numero_sequence     -- Numéro de séquence prochaine trame à recevoir
begin
trame_Attendue := 0 ;               -- Initialisation attente de la trame 0
loop
attendre (événement) ;              -- Attendre arrivée trame
-- Deux cas possibles: la trame est correcte ou en erreur
if événement == Arrivée_Trame then  -- Cas d'une trame correcte
    recevoir_couche_physique (r);   -- Prendre la trame arrivée
    if r.seq ==Trame_Attendue then  -- C'est la bonne trame
        envoyer_couche_réseau(r.info) ;  -- La passer à l'utilisateur
        inc (Trame_Attendue) ;      -- Préparer trame suivante
    endif;
-- Dans tous les cas : en séquence ou pas on doit acquitter
envoyer_couche_physique(s);         -- Envoyer le crédit / acquit
endif     -- Dans le cas d'une erreur on ignore la trame reçue
end loop                            -- Boucle infinie
end récepteur_3;
```

Protocole 3 PAR : Conclusion
1) Problèmes de robustesse

- Fonctionnement **correct** dans toute configuration:

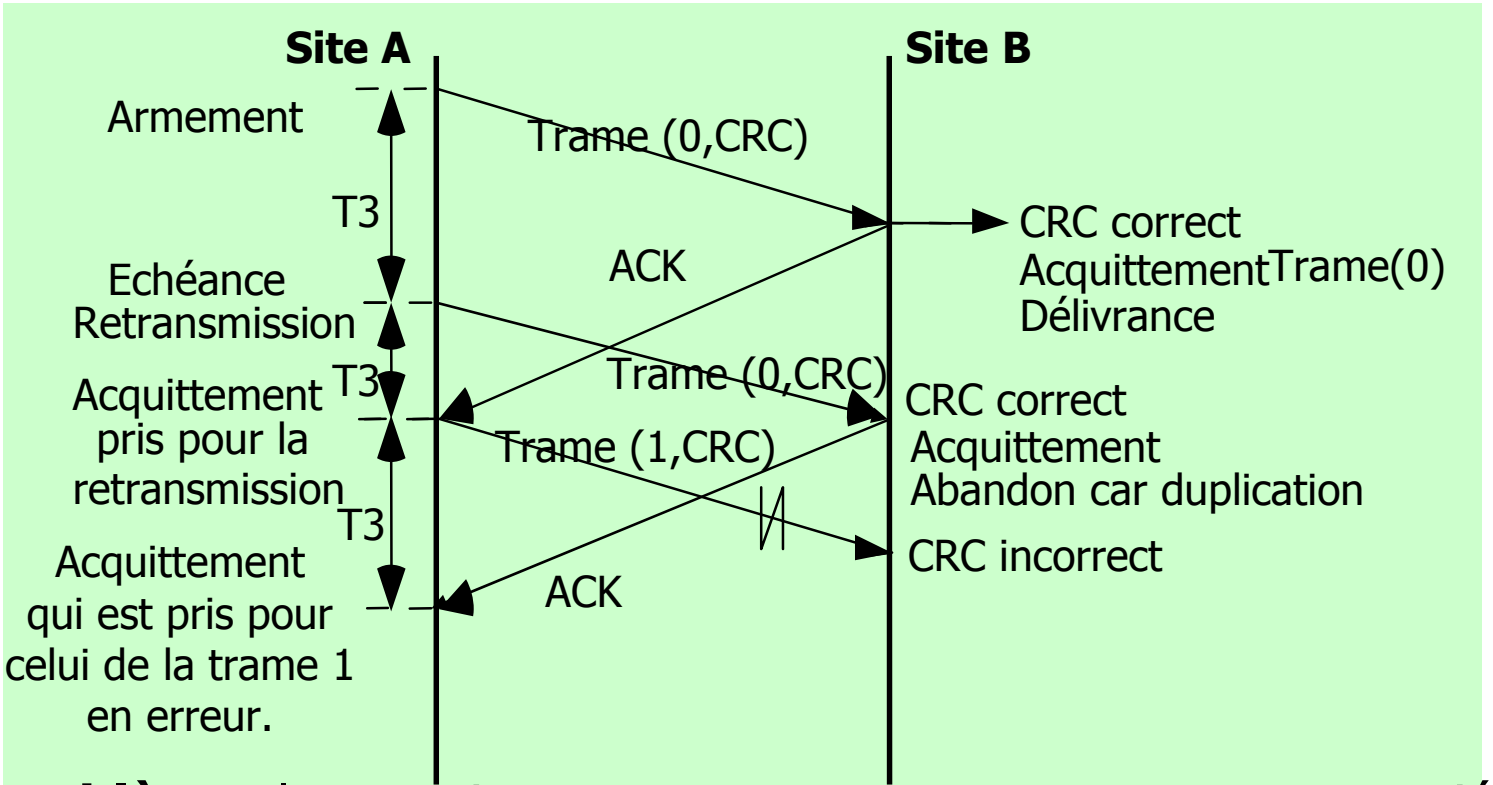
  . de **perte de messages** ou **d'acquittements.**

  . de **réglage des délais de garde.**

  . de **délai d'acheminement**.
- **Besoin de preuve d'un protocole :**
  - Sémantique (preuve de programme)
  - Temporelle (contraintes de synchro et de temps réel)
- Difficulté: Le protocole **n'est pas généralement pas reconfiguré** à chaque fois que l'on change de support physique ou de débit.

Exemple de problème de robustesse

- **Combinaison d'un délai de garde court et d'une perte de message.**



- **Le problème:** les acquittements ne sont pas correctement associés aux messages.
- **Solution :** Identifier dans les acquittements le numéro du message acquitté. => Solution des protocoles à fenêtres glissantes.

Protocole 3 PAR : Conclusion
2) Problèmes de performance

- **Chaque acquittement positif occupe une trame** uniquement dédiée à cette fonction => Relativement coûteux pour une opération très simple.

. ***Gestion d'une interruption*** *pour chaque acquittement*.

Sauvegarde du contexte + Traitant d'interruption + Déroutement couche liaison + Restauration du contexte.

. ***Réservation puis libération d'un tampon*** *pour chaque acquittement.*

- **Si les délais de transmission sont longs, l'attente d'un acquittement est longue => émetteur inactif**

. **Le taux d'utilisation** de la voie peut devenir très faible**.**

. **Exemple des voies satellite** (temps de propagation 250 milliseconde): Le taux d'utilisation de la voie peut tomber à quelques pourcent.

Protocole 4 : A fenêtre glissante avec réception des trames en séquence

- **Objectifs : corriger les défauts** du protocole PAR.

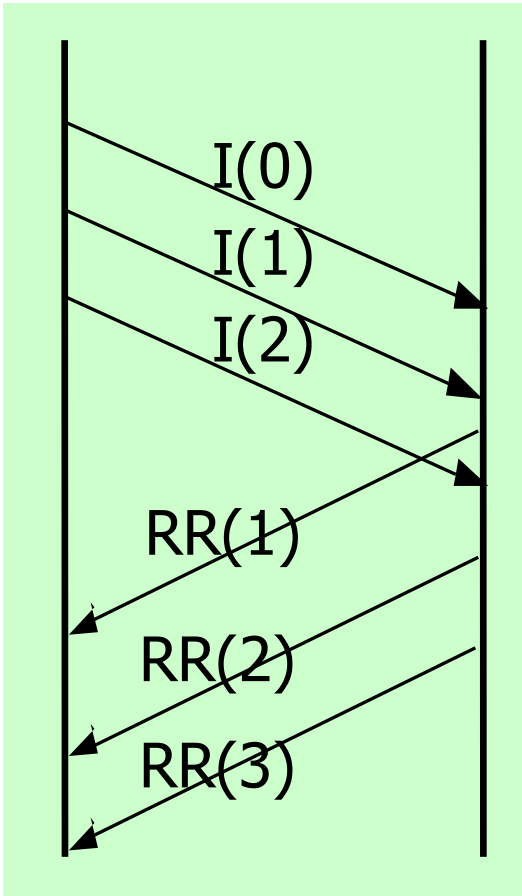
- **Protocole robuste** qui traite correctement le problème de contrôle d'erreur, de flux, de livraison en séquence
  . en présence de perte de messages ou d'acquittements
  . pour tout réglage des délais de garde

- **Protocole performant** qui optimise l'utilisation de la voie en fonction **des temps de transmission**
  . Protocole performant pour différents modes de fonctionnement.

- **Reprise** des outils déjà vus:
  - **Acquittements positifs**, **délais de garde**, **numéros de séquence** de message.
  - Utilisation des notations des protocoles type **LAPB**.

Amélioration 1 : Emission en anticipation ("Pipelining")

- **Anticipation des émissions :** ne pas attendre l'acquittement d'une trame avant d'envoyer les suivantes.
  - Pour éviter d'être **bloqué en attente** d'acquittement: arrêt & attente.
  - Pour résoudre le **problème d'inactivité** de la voie si les temps de transmission sont **longs** avec des messages **courts**.



- **Remarque :** L'anticipation **augmente la longueur** des trames en enchaînant la transmission de plusieurs trames consécutives. On **minimise** donc l'importance relative du temps de **propagation aller retour** et on **améliore le taux d'utilisation du canal**.

- **Exemple d'échange :** I(n) ("Information") : Trame d'information numéro n
  RR(n) ("Receiver Ready") : Trame d'acquittement pour I(n-1)

Anticipation : Règles de fonctionnement (1)

- **Règle 1** - L'émetteur doit **conserver copie des trames** jusqu'à réception de l'acquittement correspondant.

  => Pour retransmission si les trames ont été bruitées.

- **Règle 2** - Chaque trame est **identifiée par un numéro de séquence.**

  Les trames **successives** sont numérotées circulairement (modulo Maxseq+1) par des entiers **successifs**.

  => Pour respecter à la réception l'ordre d'émission.

  => Pour pouvoir éventuellement détecter des erreurs de transmission par des lacunes dans la suite des numéros reçus.

  Fonctionnement avec la règle 2:

  - L'expéditeur maintient une **variable d'état V(s)** qui définit le numéro de **la prochaine trame à émettre**.
  - Chaque trame est transmise avec **un numéro de séquence en émission N(S)** qui est la valeur courante de V(S) au moment de l'émission.

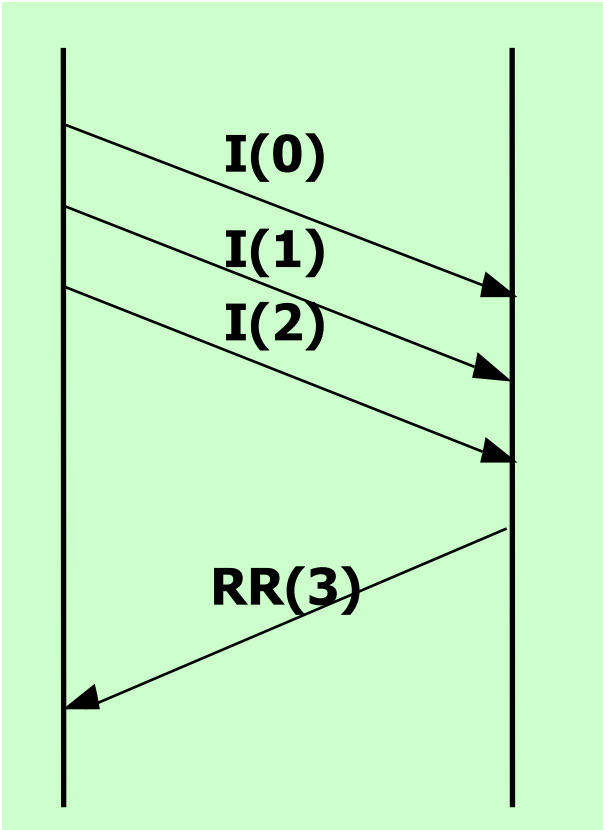
Anticipation : Règles de fonctionnement (2)

- **Règle 3** - Utilisation indispensable d'un ensemble de numéros de séquence de cardinal plus élevé que pour le bit alterné (2 numéros) pour permettre une anticipation réelle.
  - **Sur n bits** : 2^n numéros différents. Trames numérotées de 0 à Maxseq = 2^n - 1.
  - **Choix Maxseq=7** **n = 3** Peu de possibilités d'anticipation.
  - **Choix Maxseq=127** **n = 7** Encombrement dans les messages.

- **Règle 4** - L'anticipation ne peut pas être **autorisée sans limite**.
  - On n'exercerait **aucun contrôle de flux**.
  - On ne disposerait pas de la **capacité mémoire suffisante** pour les copies de trames en attente d'acquittement.
  - **Notion de crédit maximum statique** (taille maximum de la fenêtre d'anticipation).

Amélioration 2 : Regroupement des acquittements

- Il est **inutile et coûteux** d'envoyer **un acquittement pour chaque trame** d'information.
- **On peut acquitter plusieurs trames d'information I par une seule trame RR** d'accusé de réception à condition d'adopter une convention:
- **Acquittement pour I(n-1) vaut pour I(n-2), I(n-3), ... Etc toutes les trames en attente d'acquittement.**



Exemple d'émission avec anticipation et de regroupement des acquittements

Regroupement des acquittements : Règles de fonctionnement (1)

- **<u>Règle 1</u>** - **Le récepteur maintient une variable d'état V(R)** qui désigne le **numéro de séquence de la prochaine trame attendue**

  => Cette variable est incrémentée de 1 chaque fois qu'une trame est reçue en séquence sans erreur.

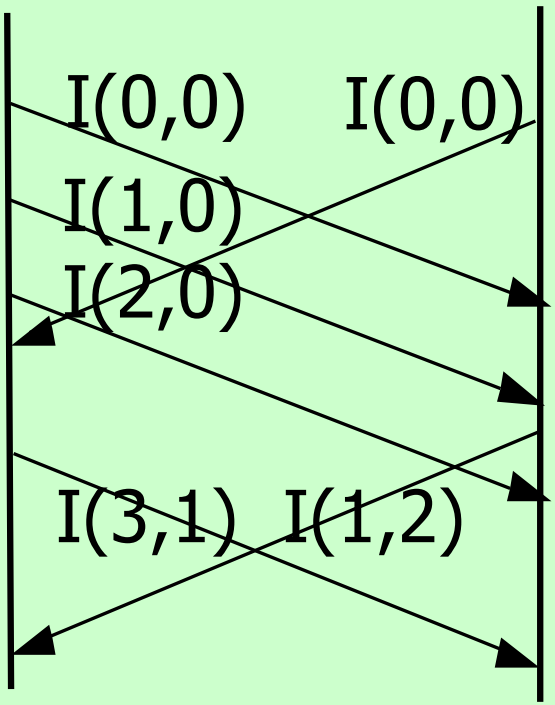
- **<u>Règle 2</u>** - **La variable de réception V(R) est reportée dans le champ N(R)** (le numéro de séquence de réception porté dans les acquittements retournés à l'émetteur RR(N(R)).

- **<u>Règle 3</u> - Cohérence initiale** des numéros de séquence et numéros d'acquittement: N(S) et N(R) = 0 au début de la communication.

- **<u>Règle 4</u> - La signification de l'acquittement:** RR(N(R)) acquitte toutes les trames en attente d'acquittement dont le numéro N(S) est inférieur ou égal à N(R)-1 => **Non pas une seule trame dont le numéro serait N(S)=N(R)-1.**

Amélioration 3 : Acquittements insérés dans les trames infos (piggybacking)

- **Insertion d'un champ acquittement** (N(R)) dans l'entête des trames d'infos.
  => Toute trame d'information devient **un acquittement positif pour des trames du trafic échangé en sens inverse**.
- **I(N(S),N(R)) acquitte** toutes les trames d'information transmises dans l'autre sens avec des numéros de séquence N(S) inférieur ou égal à N(R)-1
- **L'acquittement inséré coûte quelques bits par trame d'information**
- **Peu de trames** d'acquittement explicites.
- **Beaucoup plus de possibilités** d'acheminer des acquittements, sauf si le trafic d'informations est très faible dans un sens: retour à un acquittement explicite.

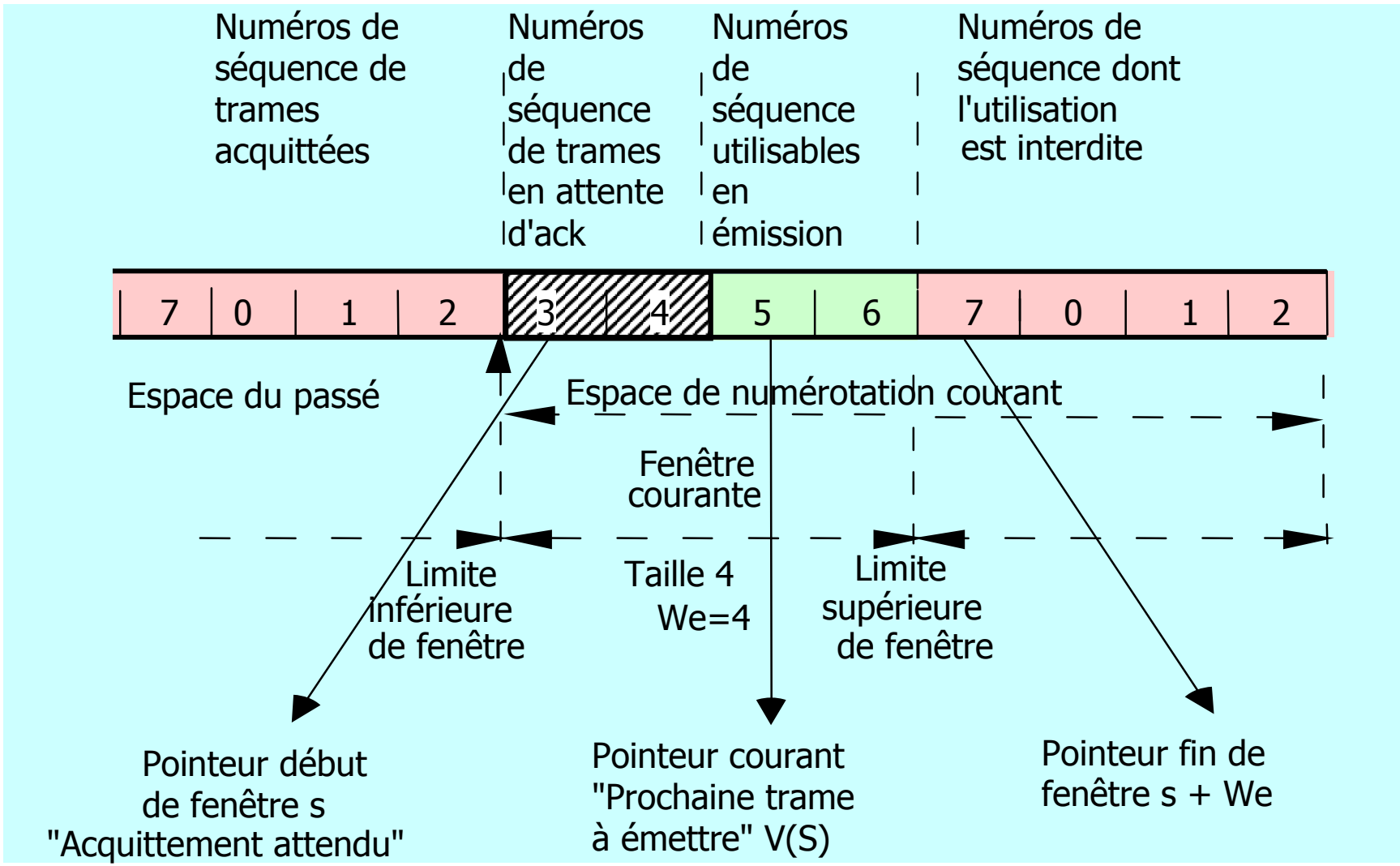


Exemple d'émission avec anticipation, acquits insérés et regroupés

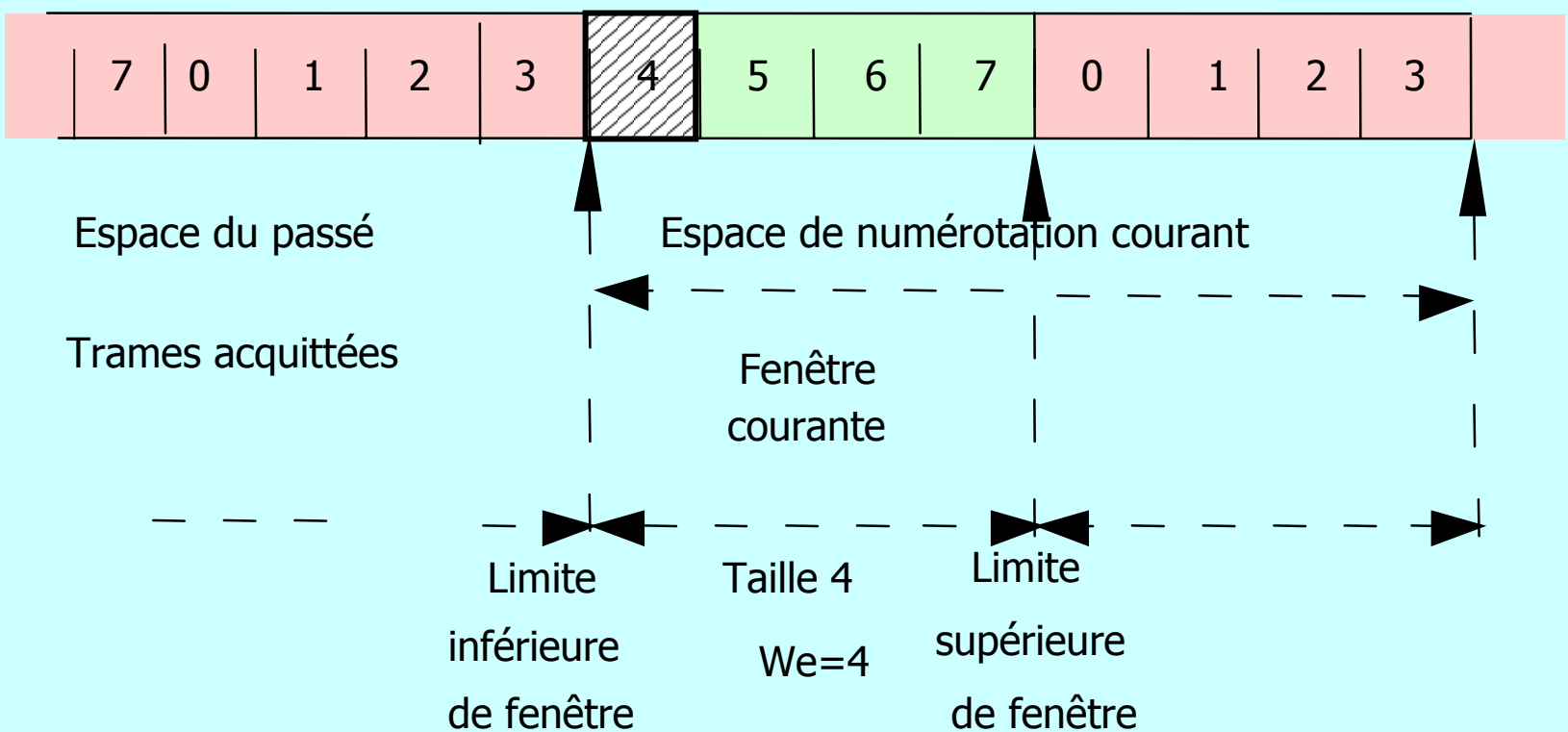
Réalisation des améliorations : Notion de fenêtre d'émission

- **La fenêtre d'émission** ("Sender's Window") est l'ensemble des numéros de séquence des trames dont l'émission en anticipation est autorisée.
  - Trames d'information déjà émises et en attente d'acquittement.
  - Trames à émettre prochainement.
- **Définie par : s =< N(S) < s+ We**:
  - **s** est le numéro de la **plus ancienne trame non acquittée**, qui est la limite inférieure de la fenêtre.
  - **s+We** est la limite supérieure de la fenêtre, qui est le **numéro de la première trame** dont l'envoi est **interdit**.
  - **We crédit maximum constant:** taille max de la fenêtre en émission.
- **Quand une (ou plusieurs) trames sont acquittées** la fenêtre d'émission glisse circulairement vers le haut.
  - Notion de protocole à **fenêtre glissante** (ou coulissante) (**"Sliding Windows Protocols")**

Protocoles à fenêtres glissantes : Exemple de fenêtre d'émission



Protocoles à fenêtres glissantes : exemple de glissement après RR(4)



Réalisation des améliorations : Notion de fenêtre de réception (1)

- **Fenêtre de réception** ("Receiver's Window") : l'ensemble des numéros de séquence des trames que le récepteur est **autorisé à recevoir**.

  => Toute trame dont le numéro de séquence correspond à un numéro de la fenêtre de réception est **acceptée**.

  => Toute trame dont le numéro de séquence est à l'extérieur de la fenêtre de réception est **détruite**.

Réalisation des améliorations : Notion de fenêtre de réception (2)

- Une **trame reçue correctement et dont le numéro de séquence correspond au niveau bas** de la fenêtre en réception:

  => peut-être **délivrée à l'utilisateur** car elle est en séquence (respect de l'ordre d'émission),

  => La fenêtre en réception peut **glisser d'une unité** vers le haut,

  => La trame peut-être **acquittée** vis à vis de l'émetteur,
- Ces opérations sont réalisées **de façon plus ou moins rapide** sans incidence sur le fonctionnement correct du protocole.
- La fenêtre d'émission et la fenêtre de réception **peuvent être de tailles différentes.**

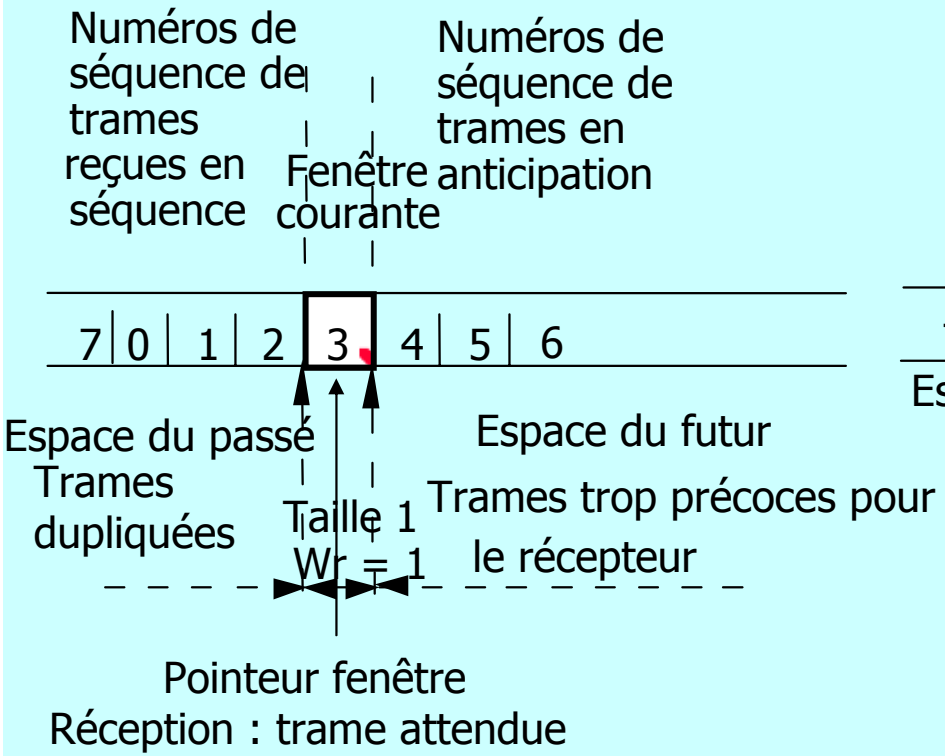
Fenêtre de réception dans le protocole 4

- **Taille de la fenêtre en réception : 1**
  - Le récepteur est donc obligé de **recevoir** les trames correctement les unes après les autres **exactement dans l'ordre d'émission** (un seul tampon suffit).
  - Quand une trame est en **erreur** le récepteur (qui persiste a ne vouloir qu'une seule trame) **perd toute la série** de trames émises en anticipation par l'émetteur :

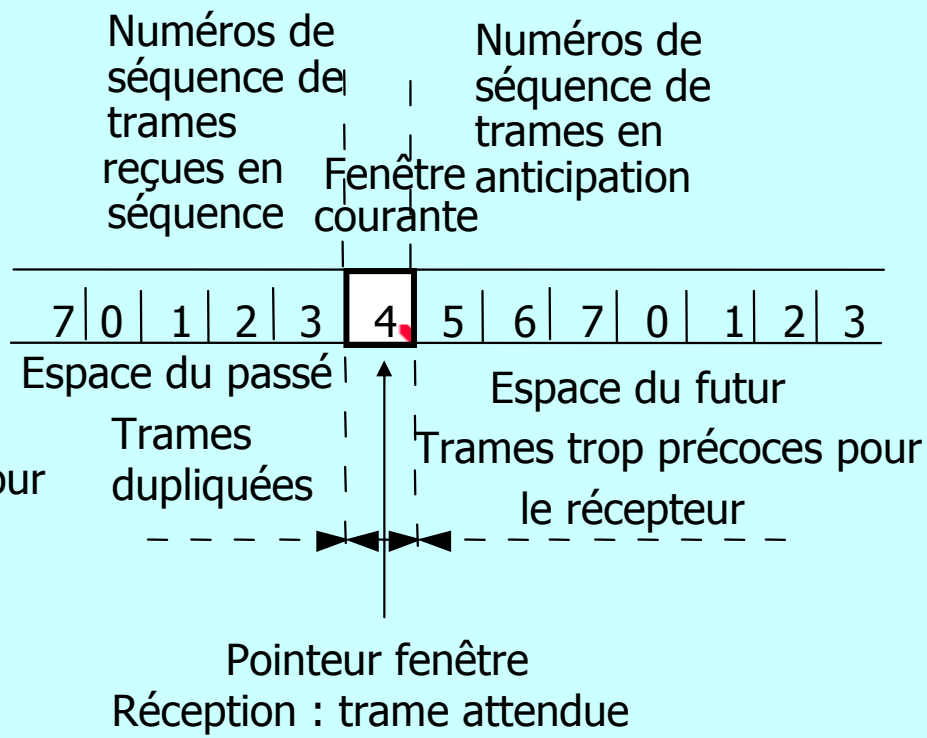
    => Effort d'anticipation perdu (optimisation à venir protocole 5)
- **Stratégie de reprise sur erreur**
  - Stratégie de délai de garde et acquittement positif
  - Stratégie d'acquittement négatif
- **Technique retour arrière de n : "Go back n"**
  - Lorsque le récepteur **constate une lacune** dans la séquence des trames et il **demande la retransmission** de toutes les trames non acquittées avec : N(S) > N(R)-1

Exemple de fenêtre en réception dans le protocole 4

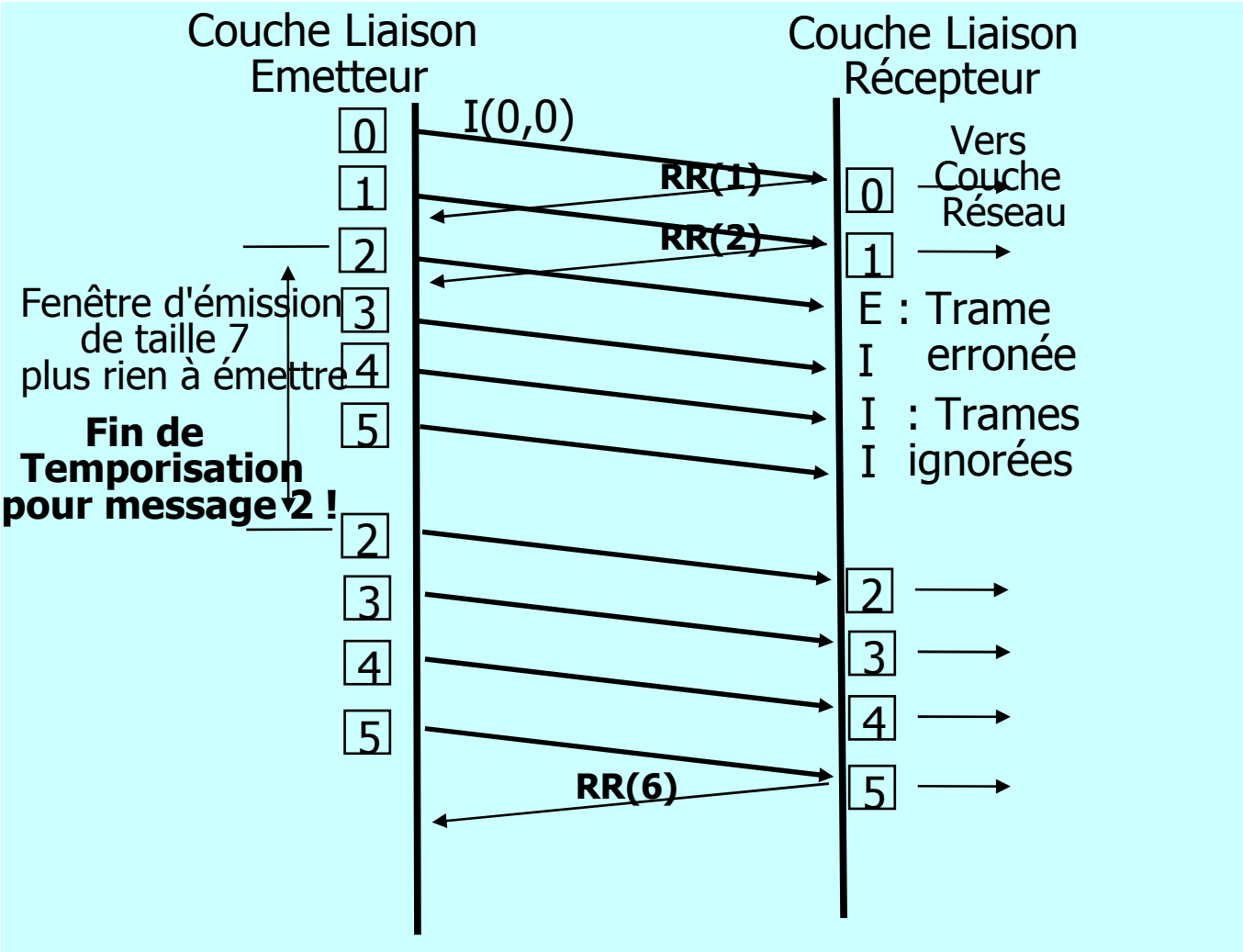
Fenêtre en réception



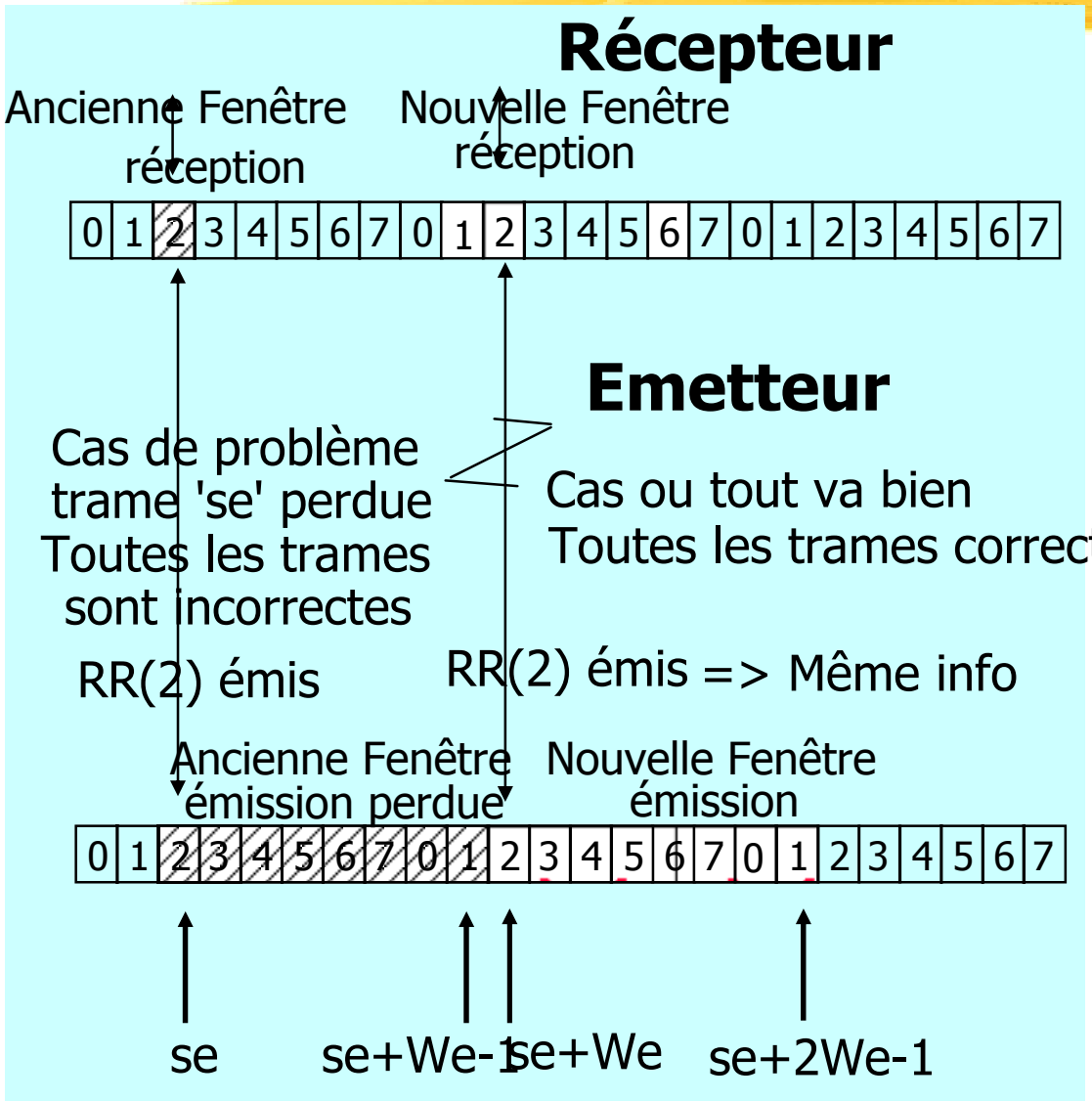
Fenêtre en réception après arrivée de I(3)



Exemple de fonctionnement du protocole 4



Approfondissement 1 : taille maximum de la fenêtre en émission



- **Hypothèse:** 8 numéros de séquence et taille de fenêtre 8
- Si RR(2) se perd pour une fenêtre correcte : retransmission acceptée des trames des trames 2, 3, 4

=> En fait duplication.

- Il y ambiguité quand: se = se + We mod (maxseq +1)

=> **On ne peut pas utiliser une fenêtre comportant la totalité des numéros**.

Approfondissement 1 : formalisation problème de taille maximum (1)

- **Hypothèse : We = Maxseq +1**
- Un émetteur émet toute sa fenêtre: **s < N(S) < s+We-1**
- ***Cas 1 : La première trame s est en erreur***

  Si s est en erreur, le destinataire retourne un acquittement portant N(R) = s => Dans ce cas l'acquittement signifie qu'aucun message n'est reçu correctement (attente de s).
- ***Cas 2 : Toutes les trames sont reçues correctement***

  Le destinataire retourne un acquittement N(R) = s+We

  => Dans ce cas l'acquittement signifie que toutes les trames sont reçues correctement (attente de s+We).

Approfondissement 1 : formalisation problème de taille maximum (2)

- **Pour qu'il n'y ait pas d'ambiguïté possible sur la signification il faut que les deux acquittements:**

  N(R) = s et N(R) = s+We

  soient distinguables (soient des valeurs différentes)
- Si We = Maxseq +1: Pas de distinction entre cas 1 et cas 2

  N(R) = s = s+We = s + Maxseq +1 mod (Maxseq+1)
- **Pour que** les We+1 nombres allant de s à s+We soient tous distincts modulo Maxseq+1 il faut We < Maxseq+1.

  => On doit prendre au plus We = Maxseq
- **<u>Exemples</u>** : n = 3 , 2^n = 8, We=7 trames en anticipation.

  Cas n = 7 , 2^n = 128, We = 127 trames en anticipation.

Approfondissement 2 : Niveaux de contrôle de flux

- **1 Contrôle de flux par fenêtre d'émission (rappel)**

  Avec une fenêtre glissante, si le destinataire s'abstient de retourner des acquittements, il est assuré de ne pas recevoir plus de We trames d'informations s'il retient ses acquittements.

  Problème - Cette technique ne peut plus s'appliquer lorsqu'un site peut obliger l'autre à acquitter (mode commande réponse).

- **2 Suspension temporaire des échanges**

  Permet au destinataire de **demander l'arrêt** temporaire (**puis la reprise**) du protocole.

  Exemple : RNR (non Prêt à recevoir) demande la suspension
  RR ou REJ permettent de reprendre

  Autres exemples : (XOFF, XON) ou (WACK, ENQ) , Ethernet 802.1q

  Ne convient pas si un nombre important de trames peuvent être émises pendant la transmission de la demande d'arrêt.

Protocole 5 : A fenêtre glissante et rejet sélectif

- **Objectif :** Conserver le **bénéfice de l'anticipation** en **cas d'erreur**.
- **Solution :** utiliser une fenêtre de réception de **taille supérieure à 1**

  => Wr>1 définit **la plage des numéros N(S)** de trames d'informations **acceptables par le destinataire**.

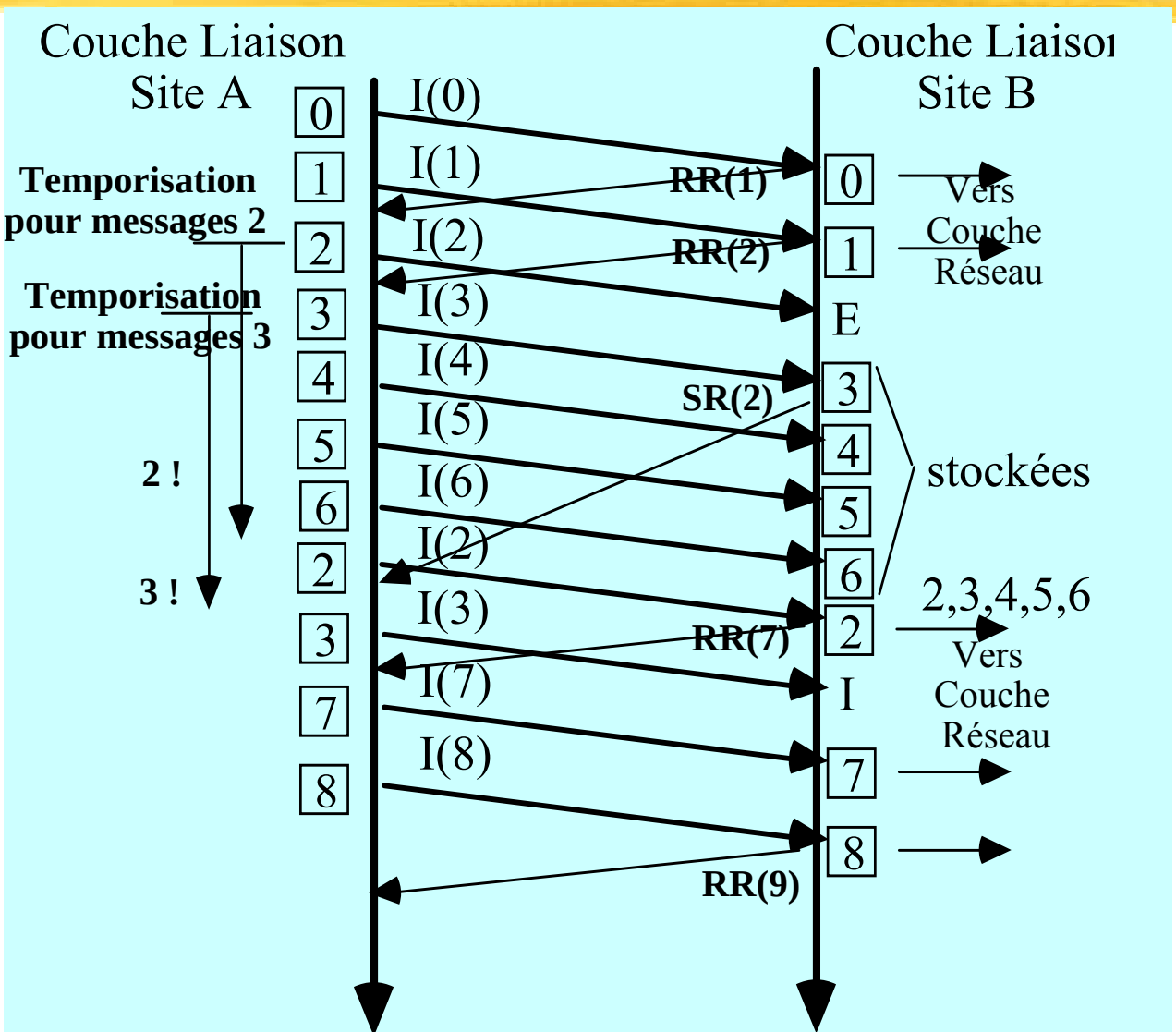
  => Le récepteur accepte **des trames déséquencées** (avec des lacunes dans la numérotation).

  => Le récepteur doit donc **gérer pour chaque trame de la fenêtre un booléen indiquant l'arrivée correcte**.

  => Le récepteur reconstitue la séquence complète des trames émises par **retransmission sur échéance de délai de garde ou sur acquittement négatif**.

Exemple de fonctionnement en mode de rejet sélectif

- L'acquittement négatif est baptisé également ***rejet sélectif*** (SR(N(R) "Selective Reject")
- C'est une **demande de retransmission d'une seule trame d'information en erreur** (de numéro de séquence N(R)).



Approfondissement 1 : Problèmes de taille de fenêtres en rejet sélectif

- **Dimensionnement des fenêtres émission et réception**

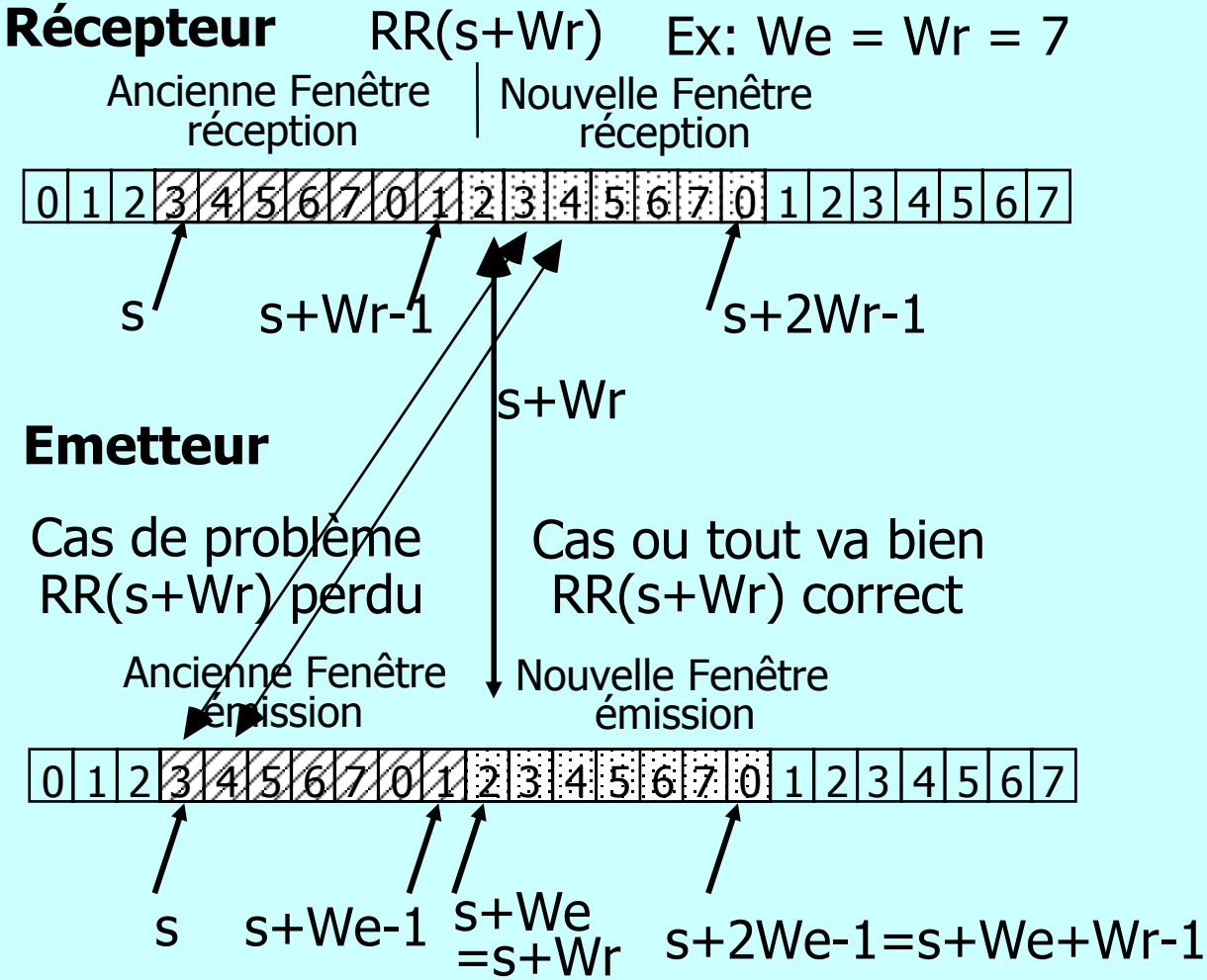
  - We < Wr : pas très utile car Wr-We tampons en réception ne sont jamais utilisés.

  - We > Wr : on ne traite pas complètement le problème de perte du bénéfice d'anticipation en cas d'erreur.

  - We = Wr : choix rationnel.

- **Problème :** le protocole en rejet sélectif fonctionne t'il avec We = Wr = maxseq (la valeur déterminée pour le protocole 4).

Approfondissement 1 : Perte d'un acquittement pour une fenêtre pleine



Retransmissions acceptées des trames 3,4,.. : duplication

Approfondissement 1 : Formalisation du problème de taille en rejet sélectif

- **Si We = Wr = W** : il y ambiguïté entre deux fenêtres d'émission successives quand

  s = s + 2*We -1 mod (maxseq +1)

  => W = maxseq/2 +1

  => On ne peut pas utiliser plus de la moitié des numéros de séquence disponibles.

- **Relation plus générale** : si We différent de Wr

  s = s + We + Wr -1 mod (maxseq +1)

  We + Wr - 1 = maxseq +1

  Ambiguïté si We + Wr = maxseq +2

Approfondissement 2 : Gestion de temporisateurs multiples

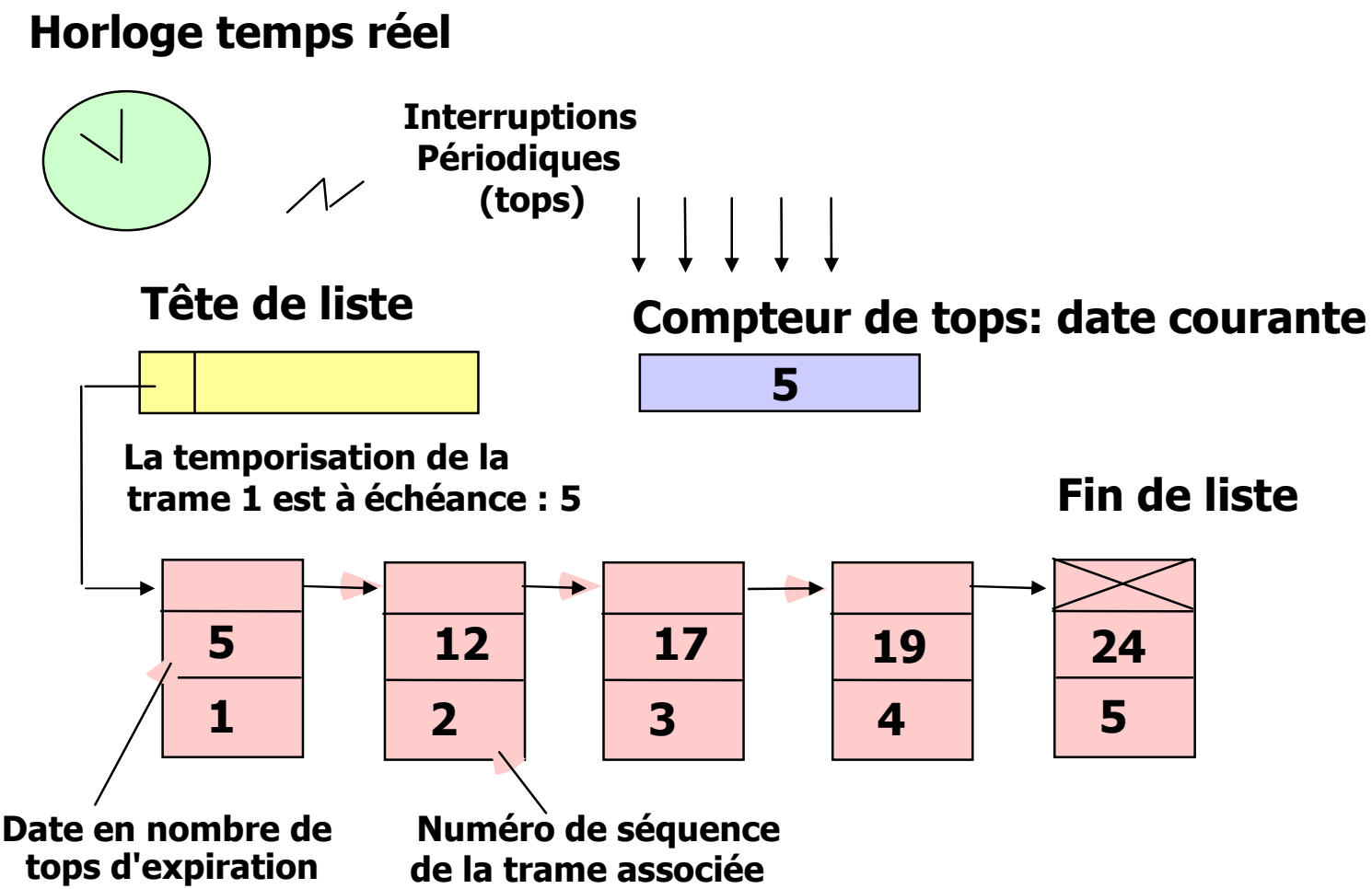
- **Les protocoles à fenêtres glissantes à rejet sélectif impliquent une temporisation associée à chaque message**.
- **Gestion d'un échéancier** (une liste chaînée par ordre croissant de dates d'échéances).
- **Armement d'un temporisateur ("Start_timer")**

  . Insertion d'événements à générer dans la liste.
- **Désarmement d'un temporisateur ("Stop_timer")**

  . Retrait de la liste de l'événement associé
- **Échéance d'un temporisateur ("Alarm")**

  . Déclenchement de traitement de temporisateur sur arrivée en échéance du temporisateur de la tête de liste.

Approfondissement 2 : Schéma avec temporisateurs multiples



Conclusion : Solutions générales aux problèmes des protocoles de liaison

- **Existence de solutions satisfaisantes** construites pour des voies physiques bruitées à bas débit.

  . Solution au problème du **contrôle d'erreur, de contrôle de flux, de contrôle de séquence.**
- **Réutilisation** de ces solutions dans les réseaux sans fils (bruités et d'un débit pas très élevé).
- **Renouvellement du problème** :

  A) Liaison sur câble **coaxial ou fibre** optique : **faible bruit et haut débit.**

  B) Les applications visées sont de plus en plus multimédia: besoin de solutions de communication à **qualité de service**

  => La couches liaison traite **la délimitation, le multiplexage, l'administration** mais **pas de contrôle d'erreur ni de flux** (reportés au niveau transport)

Niveau Liaison En Point à point

Chapitre II

Protocoles industriels

II.1 Protocole à trames de bits

II.2 Protocole PPP

Protocoles de liaison en point à point : Exemples industriels

II.1
Protocoles à trames de bits

Introduction : Rappel des protocoles synchrones en mode caractère

- **Les premiers protocoles de liaison implantés** (définis avant 1970).
- **Protocoles en mode caractère BSC "Binary Synchronous Communication"**
  - L'unité d'information **est le caractère**.
  - Certains caractères sont réservés aux besoins du protocole : les ***caractères de contrôle subsistent dans le jeu de caractères ASCII)***
  - Il a existé **de multiples versions de protocoles basées sur des principes** et une utilisation des caractères de contrôle souvent **très voisins**.
    - Exemple : Protocole BSC 3780

Evolution vers les protocoles à trames de bits

- **Volonté de construire des protocoles indépendants d'un jeu de caractères**.
  - Si l'on transporte des caractères 10 bits ou des mots 60 bits **il faut découper et coder** les données au format caractère du lien.
  - Il faut traiter à chaque fois le problème de l**a transparence en fonction** du code caractère et traiter spécifiquement les caractères de contrôle.
- **Conséquence pour les protocoles à trames de bits:**
  - a) La charge utile devient **une suite de bits**.
  - b) Abandon des caractères de contrôle et **définition d'un format de trame au moyen de zones** ayant un rôle et un codage binaire (structure de message).

Protocoles à trames de bits : Historique et normes associées (1)

- IBM a développé vers 1970 **SDLC** ("**Synchronous Data Link Communication**") pour SNA.
- SDLC a été soumis à l'ISO pour normalisation. Modifié il est devenu **HDLC "High-level Data Link Communication".**
- SDLC a été soumis à l'ANSI pour devenir le standard américain qui l'a modifié et est devenu **ADCCP ("Advanced Data Communication Control Program")**
- Le CCITT a adopté et modifié HDLC qui est ainsi devenu le **LAP ("Linkage Access Protocol")** pour le standard de réseau **X25**.
- Le CCITT a modifié X25 dont le LAP qui est devenu le **LAPB** ("**Linkage Access Protocol**" **B** ou Balanced)

Protocoles à trames de bits : historique et normes associées (2)

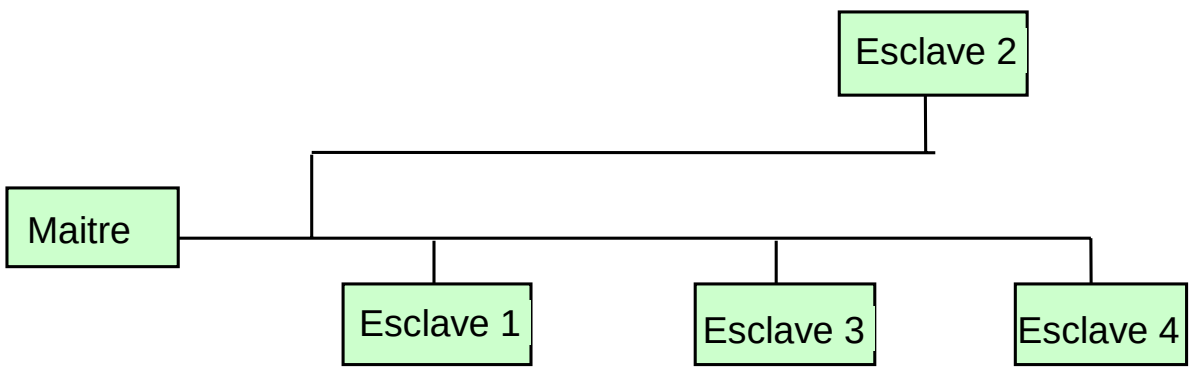
- Les IEEE ont normalisé comme l'un des standards de liaison sur les réseaux locaux une version modifiée légèrement : le **LLC2 ("Logical Link Control type 2")**
- Une autre version : le **LAPD ("Linkage Access Protocol on the D channel")** est définie pour servir de protocole de liaison sur les canaux D du RNIS.
- Une version **LAPX** est spécifiée pour fonctionner avec des échanges à l'alternat (**LAP semi-duplex**).
- Une version **LAPM** (‘**LAP Modem**’) est spécifiée pour la corrections d’erreurs dans les modems.
- Le standard Internet **PPP (‘Point to Point Protocol**’) en version de base reprend différents idées des protocoles à trames et possède une version avec contrôle d’erreur qui est un version de LAPB.

Protocoles à trames de bits : Principes généraux

- **Nombreux points communs** : protocoles à fenêtres.
- Mais **variantes de détail**, liées à des contextes d'utilisation qui rendent ces protocoles incompatibles
- **Quelques choix communs**
  - Utilisation du **principe d'anticipation.**
  - **Numéros de séquence**
    - En général sur 3 bits (au maximum 7 trames en anticipation).
    - Variantes sur 7 bits (exemple LAPD).
  - **Regroupement des acquittements.**
  - **Acquittements insérés** ("piggybacking").
  - Choix dans la plupart des cas **d'une fenêtre en réception de taille 1** (sauf HDLC, ADCCP).

Protocoles à trames de bits : Rappel voies multipoint

- **Deux modes** de fonctionnement :
  - **Symétrique ('Balanced')** : point à point.
  - **Dissymétrique ('Unbalanced') :** en **scrutation** maître esclave (mode multipoint) **("polling- selecting").**
- **Organisation arborescente (en grappe) des voies multipoint**
  - **Site maître** (calculateur): qui dispose de toutes les fonctions (autre terme: primaire) et supervise les communications.
  - **Sites esclaves** (terminaux): qui ne disposent pas de certaines fonctions (autre terme: secondaire) et qui obéissent au maître.



Protocoles à trames de bits : la gestion des voies multipoint

- **Une classification des fonctions** est effectuée : **fonctions primaires et fonctions secondaires (**pour caractériser les modes maître et esclave).

- Sont considérées comme **primaires :**

  - Mise **en ligne ou hors ligne** (mise d'une station en l'état d'émettre ou de recevoir).

  - **L'initiative dans la validation des données** (sollicitation des acquittements).

  - Les traitements en **cas d'erreur de protocole**.

- Sont considérées comme **secondaires :**

  - **L'initiative d'émission de données** (une fois le mode d'échange établi)

  - … **les autres fonctions**

Protocoles à trames de bits : les différents modes d'échange

Distinction dans les modes d'utilisation des fonctions primaires et secondaires

- **Mode normal de réponse NRM** ("Normal Response Mode") : **Configurations dissymétriques**

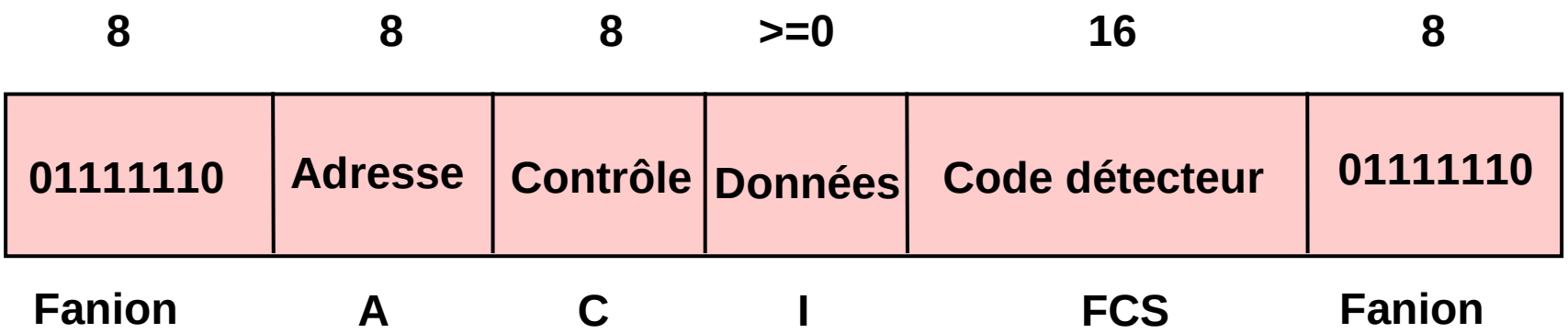
  - L'initiative de l'attribution du mode d'échange est réservé uniquement à la station primaire.

- **Mode asynchrone ABM** ("Asynchronous Balanced Mode") : **Configurations symétriques**

  - Mode asynchrone symétrique applicable aux liaisons point à point en réseau..

  - Une station recevant une commande doit y répondre immédiatement (voir plus loin bit P/F).

Protocoles à trames de bits : la structure des trames (1)



- **1) Les fanions et la transparence binaire (bit stuffing)**

  - Une méthode pour délimiter les trames qui préserve **les propriétés de transparence** des données utilisateur.

  - Chaque trame **commence et se termine par la chaîne 01111110** (Terminologie Drapeau, fanion ou flag)

  - A l'émission **quand on détecte une donnée avec une de suite de 5 bits 1** consécutifs on insère un 0, **en réception le bit 0 suivant 5 bit 1** est automatiquement enlevé.

  - Un **fanion** délimiteur 01111110 est donc **toujours considéré comme tel**.

Protocoles à trames de bits : la structure des trames (2)

- **2) Le champ adresse** (A)

  Permet de traiter les voies multipoint (adresse d'une station secondaire).

  Pour les voies point à point (on distingue les commandes des réponses).
- **3) Le champ contrôle (C)** contient :
  - Le **type** de la trame.
  - Le **numéro de séquence**
  - Les **acquittements**.
  - Le **bit de commande réponse.**
- **4) Le champ données (I)**

  Il peut être arbitrairement long (à traiter en liaison avec le CRC dont l'efficacité décroît en raison de la probabilité d'erreurs en rafale).
- **5) Le champ détection d'erreur (FCS "Field Check sequence") .**

  Généré par le polynôme du CCITT: X16 + X12 + X5 + 1

  En LAPB une variante est appliquée pour détecter les pertes de fanions.

Protocoles à trames de bits : les trois catégories de trames

- **1. Information :Trame I ("Information")**

  Transportent des informations significatives

- **2. Supervision : Trame S ("Supervision")**

  Utilisées pour superviser les échanges de trames I.

  Exemples : Envoyer un acquittement explicite

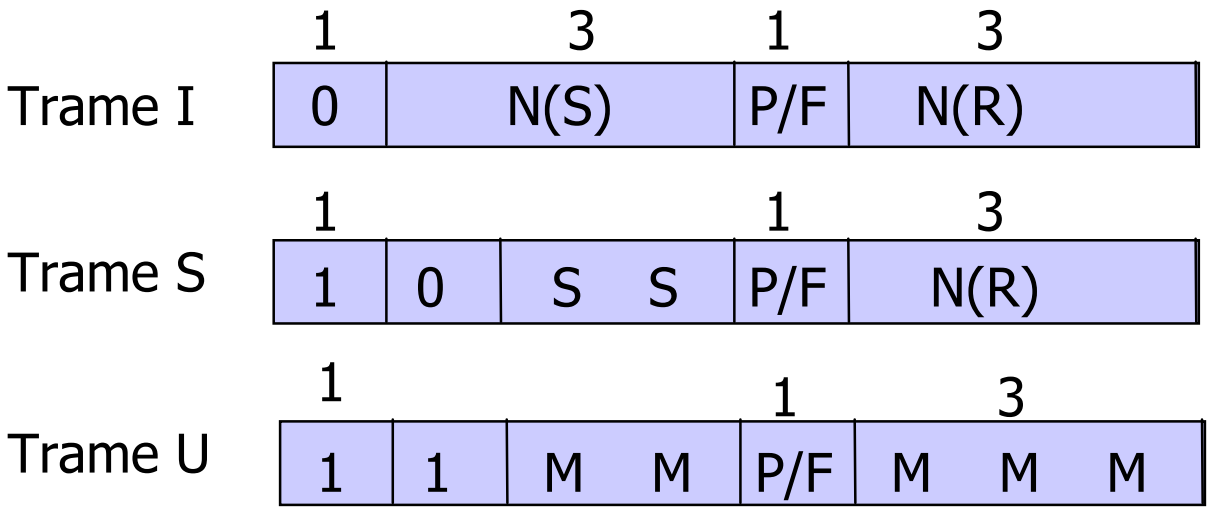
  Demander une suspension temporaire

- **3. Gestion : Trame U ("Unnumbered")**

  Assurent les fonctions nécessaires avant et après l'échange des données.

  Exemples : connexion, déconnexion d'une station, traitements d'erreurs de protocole.

Protocoles à trames de bits : les trois catégories de trames



- **N(S)** : numéro de séquence en émission
- **N(R)** : numéro de séquence de la prochaine trame non encore reçue.
- **S** : type de la fonction de supervision
- **M** : type de la trame non numérotée

Le Bit P/F : Scrutation/Fin d'émission Commande/Réponse ("Poll/final")

- **Première Signification en mode normal de réponse : Invitation à émettre ou fin d'émission** (un primaire gère un groupe de secondaires):

  - Dans les trames de commande le bit à 1 **noté P** signifie "invitation pour la station adressée à émettre (**polling**)".

  - Dans les trames de réponse en provenance de stations secondaires ce bit à 1 **noté F** signifie **fin** de transmission.

- **Seconde Signification** en mode asynchrone équilibré **: Commande Réponse.**

  - La station A recevant une trame avec le bit P l'interprète comme une **commande** émise par le primaire distant B.

  Exemple type: commande d'acquittement immédiat à destination du secondaire local.

  - A doit répondre immédiatement à la commande par **une trame de réponse avec le bit F positionné** car le site distant B a armé un délai de garde pour retransmettre.

Problème du bit P/F en mode symétrique

- **Les deux stations possèdent** :

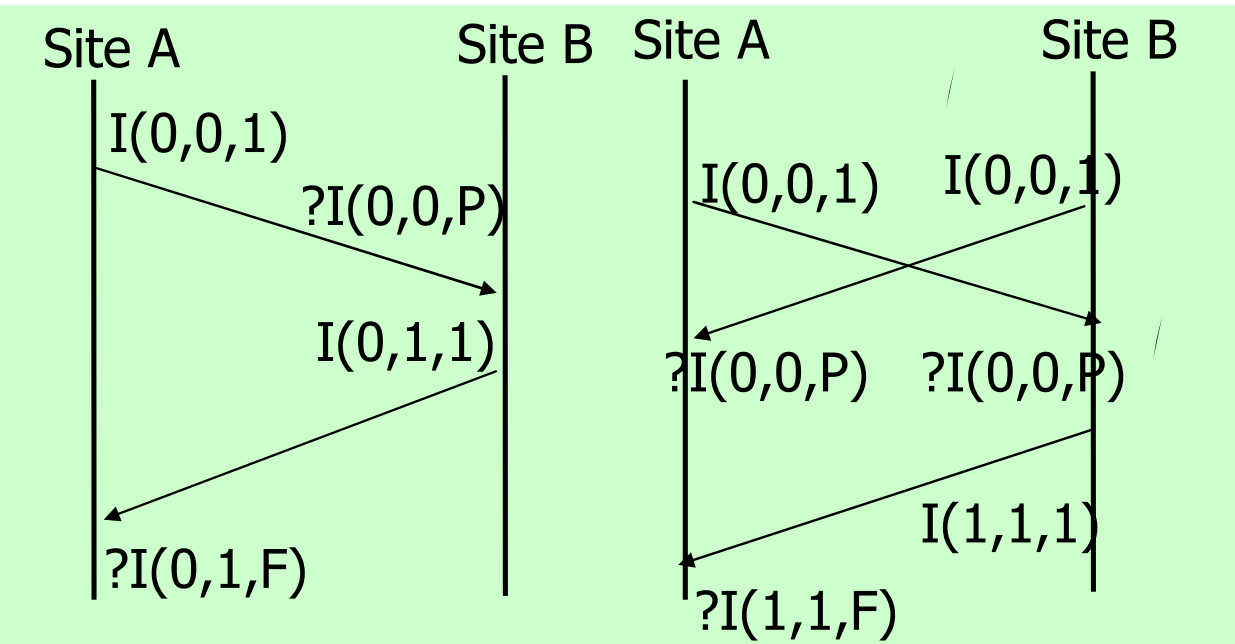
- les fonctions primaires (émettent des trames de commande avec bit P).
- les fonctions secondaires (émettent des trames de réponses avec bit F).

Mais un seul bit pour deux significations

- Une station A **recevant un bit P/F** ?

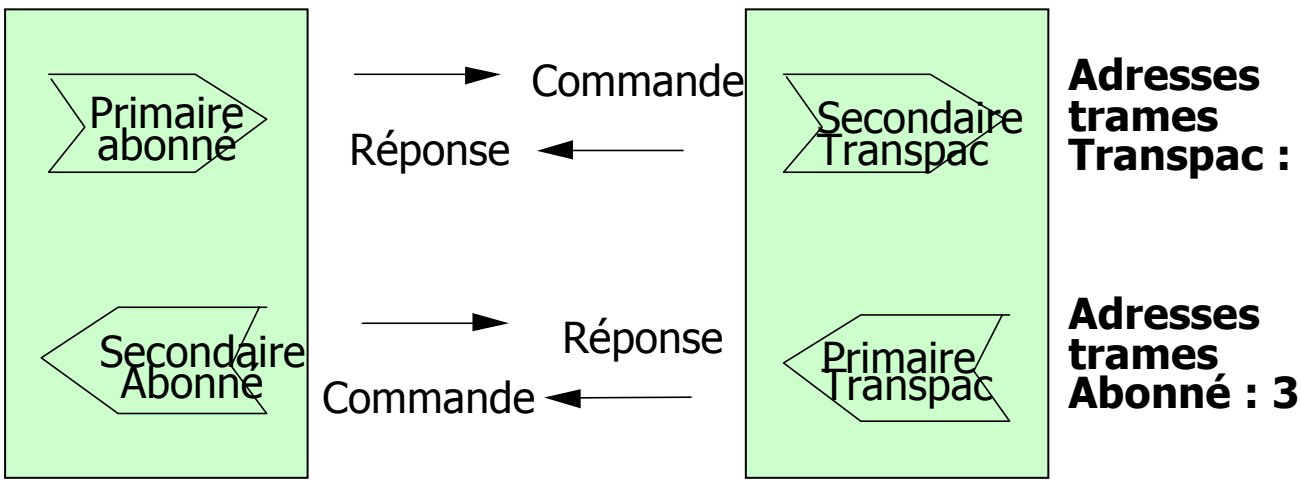
1- **C'est un bit P** que B a émis par ses fonctions primaires.

2- **C'est un bit F** en réponse à un bit P que A avait envoyé avant.



Solution bit P/F en mode symétrique : disposer de deux bits

- 1) En LAPB on ne dispose pas de deux bits dans l'octet de contrôle.
  On conserve la structure du champ contrôle => **On utilise l'adresse**.
  - Si une station agit comme primaire elle place l'adresse de la station éloignée (cas d'une initiative d'émission, bit P)
  - Si une station agit en secondaire elle place dans les trames son adresse (cas d'une réponse bit F).



- 2) Dans les protocoles LAPD, Relais de trame on a **défini** le format des trames pour faire place à un bit de plus C/R (commande/réponse).

Les quatre types de trames de supervision (1)

- **Type 0 - RR Prêt à recevoir ("Receiver Ready")**

  - Station prête à recevoir des trames I.

  - Permet d'accuser réception des trames dont le numéro de séquence est inférieur ou égal à N(R)-1.

  => Trame utilisée lorsqu'il n'y a pas suffisamment de trafic dans un sens pour transporter les acquittements.

- **Type 1 - REJ Rejet – ("Reject")**

  - Acquittement négatif (protocole 4).

  - Le champ N(R) désigne la première trame en séquence qui n'a pas été reçue.

  =>L'émetteur doit réémettre toutes les trames depuis N(R).

  - Toutes les trames jusqu'à N(R)-1 sont acquittées.

Les quatre types de trames de supervision (2)

- **Type 2 RNR- Non prêt à recevoir "Receiver not ready"**

  - Indique une incapacité temporaire à accepter les trames d'informations suivantes (en cas de saturation des tampons par exemple).

  - Mécanisme de contrôle de flux plus fort que celui de la fenêtre (un état d'exception) qui finit avec RR ou REJ.

- **Type 3 SR - Demande de retransmission sélective "Selective Reject"**

  - Demande de retransmission de la seule trame dont le numéro est contenu dans N(R)

    Non applicable au LAPB et SDLC.

    Applicable à HDLC et ADCCP.

Trames non numérotées (type U) : Ouverture de connexion

- **Distinction des modes d'échange** en ouverture:
  . Mode normal de réponse **NRM**
  . Mode symétrique asynchrone **ABM**
- **Distinction de deux formats de trames :**
  . **Format standard** : Champ commande sur 8 bits (n° séquence 3 bits).
  . **Format étendu** : Champ commande sur 16 bits (n° séquence 7 bits).
- **Demande à une station éloignée de se mettre en ligne dans un mode d'échange**

  . SNRM : Mise en mode normal de réponse standard ("Set Normal Response Mode")

  . SNRME: Mise en mode normal de réponse étendu ("Set Normal Response Mode Extended")

  . SABM : Mise en mode asynchrone symétrique standard ("Set Asynchronous Balanced Mode")

  . SABME : Mise en mode asynchrone symétrique étendu ("Set Asynchronous Balanced Mode Extended")

Trames non numérotées (type U) : Autres trames de gestion de connexion

- **DISC** : Déconnexion ("Disconnect")

  . Demande de fin du mode opérationnel (par exemple dans le cas où une station suspend son fonctionnement pour une opération de maintenance).

- **UA** : Réponse d'accusé de réception non numéroté ("Unnumbered Acknowledge")

  . Acceptation par laquelle une station notifie son accord à une commande non numérotée (commande U)

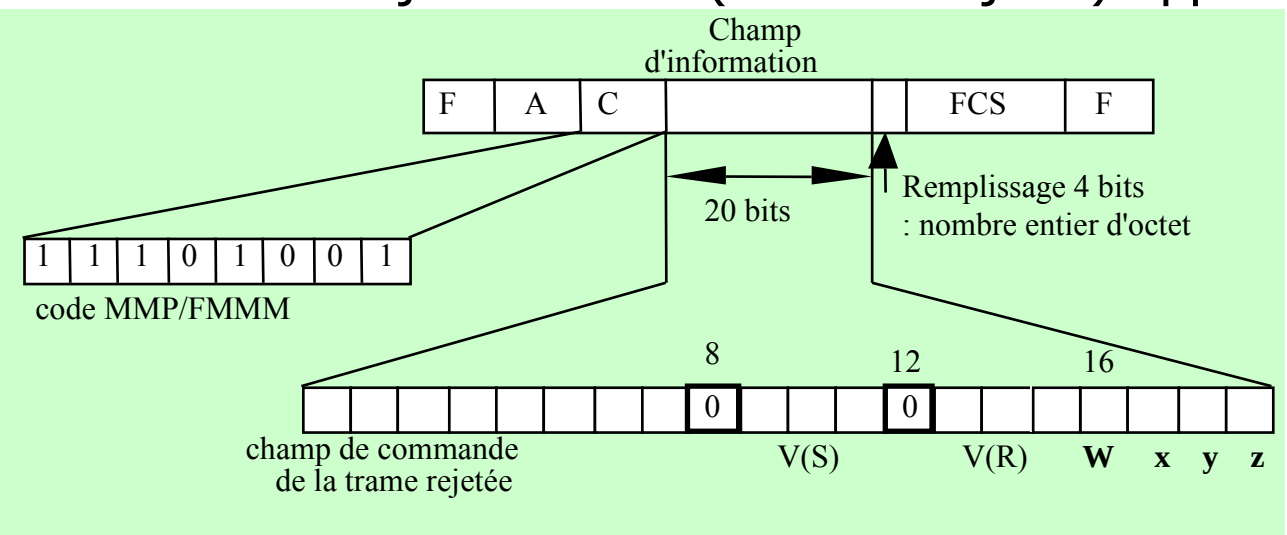
  . Il n'y a pas d'autre champ utilisé

- **DM** : Réponse mode déconnecté ("Disconnect Mode")

  . Indique que la station n'est pas en ligne et ne peut pas accepter de trame.

Trames non numérotées : Trames de rapport d'erreurs protocolaires

- **Trames qui indiquent qu'une condition d'erreur ne peut être corrigée** par retransmission car la trame est incorrecte sémantiquement (erreur de protocole).
- **FRMR**: Rejet de trame ("FRaMe Reject") Applicable au LAPB.



V(S) : Numéro d'émission en cours à la station secondaire.

V(R) : Numéro de réception en cours à la station secondaire.

bit W - Commande invalide : Exemple : une trame de supervision de type 3 demande de répétition sélective n'est pas admise en LAPB.
bit x : Présence indue d'un champ d'information (les trames S et U n'ont pas de champ d'information en général sauf FRMR et CMDR).
bit y : Champ d'information de la trame reçue trop grand pour une station secondaire (ou trop petit). Exemple : Trame de moins de 32 bits interdite en HDLC.
bit z : Numéro de séquence incorrect - accusé de réception d'une trame non émise.

Trames non numérotées (type U) : Autres trames

- **UI** : Trame de gestion d'information non numérotées ("Unnumbered Information")

  Pour les protocoles sans connexion, un seul type de trame UI (applicable à PPP).

  Pour échanger des informations protocolaires.

  Exemple : obtention dynamique d'une adresse (LAPD)
- **XID** : Trame d'identification ("eXchange IDentifier")

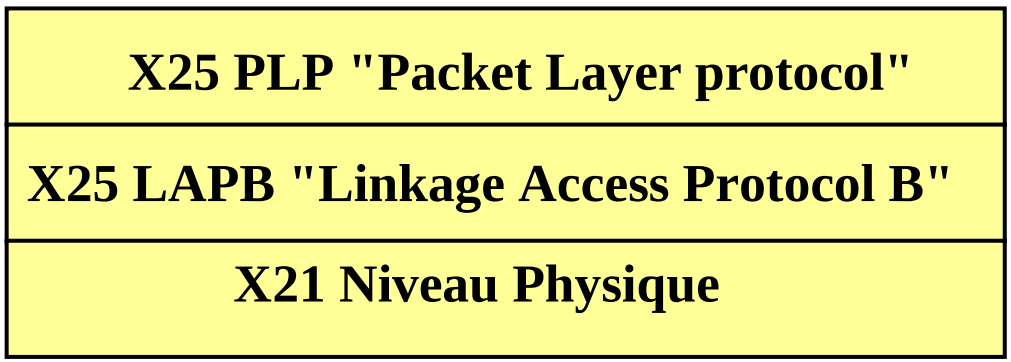
  Pour échanger **les identificateurs** de stations.

  Cette trame peut comporter un champ d'information géré par le niveau supérieur

  (Applicable au LAPD)

Protocoles à trames de bits : Le protocole LAPB

- **Rappel de la situation du protocole** (niveau liaison dans les réseaux publics "Transpac").



- **Résumé des trames utilisées en LAPB**

SABM : Ouverture de connexion.
UA : Acquittement non numéroté.
DISC : Fermeture de connexion.
DM : Indication de mode déconnecté.
FRMR : Erreur de protocole.
I : Trame d'information.
RR : Acquittement explicite.
RNR : Suspension temporaire.
REJ : Rejet d'une suite de trames I.

Protocoles à trames de bits : Le protocole LAPD

- **Rappel de la situation du protocole :** niveau liaison pour le canal D dans le réseau numérique à intégration de services (RNIS)

X25 PLP	Q930-I451 ProtocoleD
Q921-I441 LAPD	
I431 "Interfaces S/T"	

- **Principes généraux du protocole LAPD**

- En mode connecté

Echange de trames numérotées I en mode asynchrone équilibré étendu (SABME).

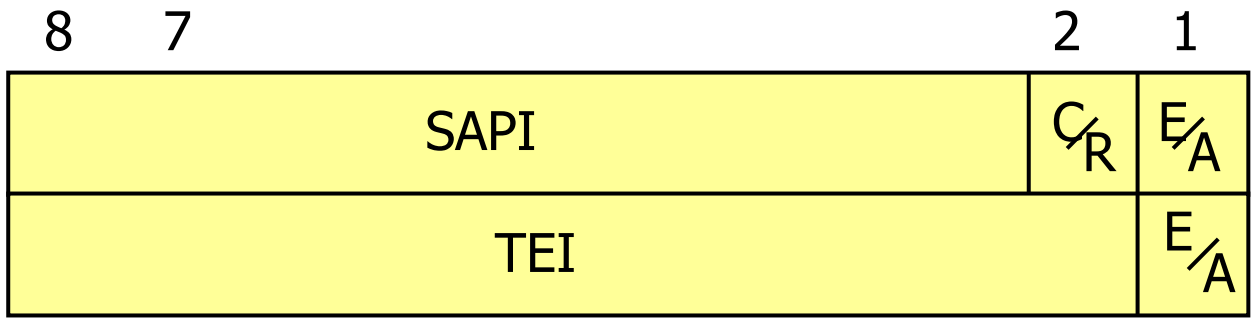
Amélioration des possibilités d'adressage: affectation dynamique d'adresse (XID).

En cas de violation de protocole il y a réinitialisation complète sans notification de cause (pas de FRMR).

- En mode non connecté

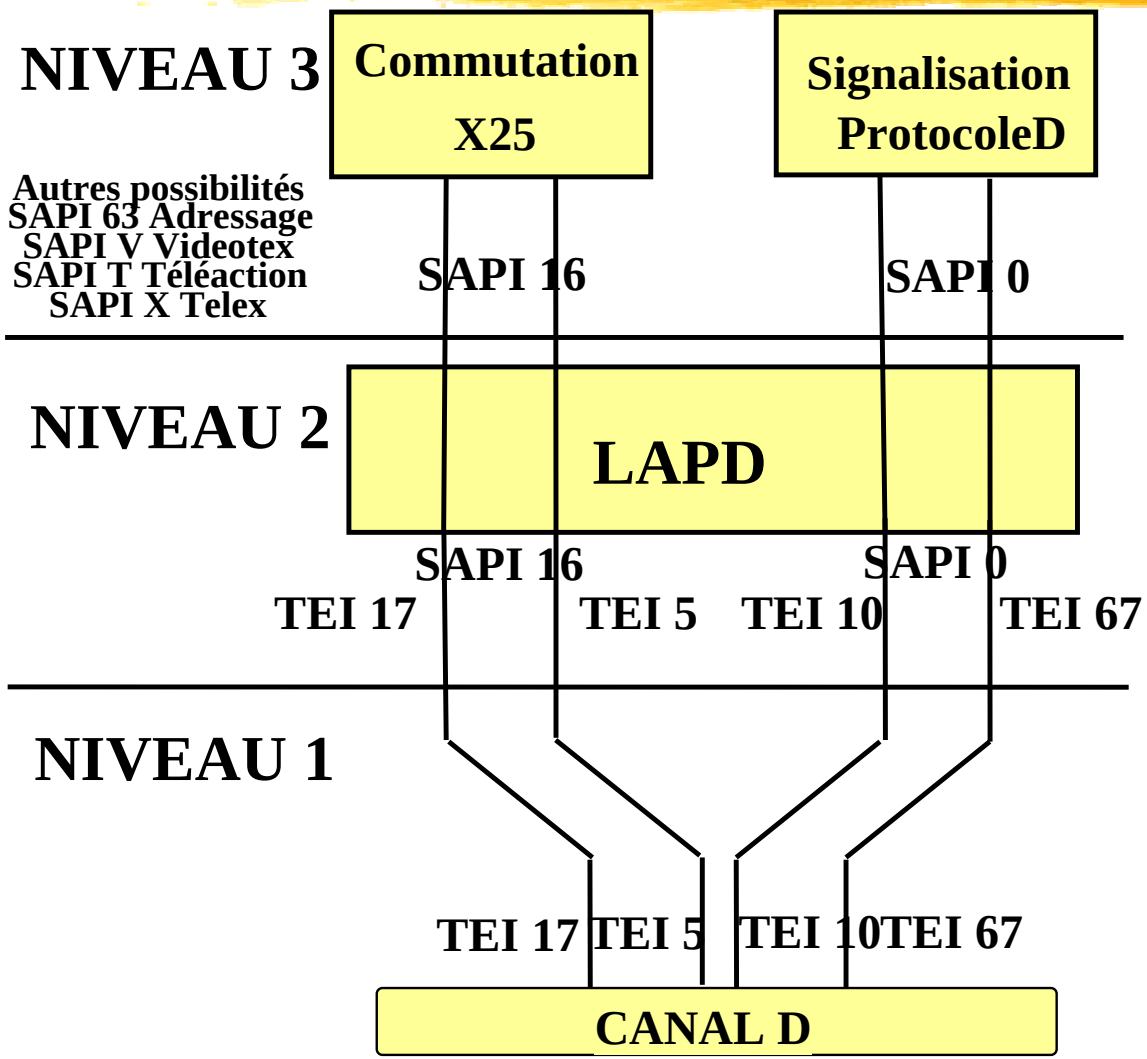
Echange de trames d'informations non numérotées **UI** sans correction d'erreur ni contrôle de flux.

Protocole LAPD : Format des adresses



- **E/A :** Bits d'extension d'adressage.
- **C/R :** Bit distinguant les commandes des réponses.
- **SAPI : 'Service Access Point Identifier'** Permet le multiplexage de différents flux de réseau sur une liaison.
- **TEI : 'Terminal End-Point Identifier'** Adresse de l'appareil physique (une appareil peut utiliser plusieurs adresses).

Protocoles à trames de bits : Adressage en LAPD



LAPD : Gestion des identificateurs de terminaux

- **a) Affectation statique**
  - Générée par le fabricant de 0 à 63.
  - Problème en cas de configurations conflictuelles.
- **b) Affectation dynamique**
  - Existence d'une fonction de la couche liaison permettant de demander un TEI à la mise sous tension d'un terminal.
  - TEI non déjà affecté entre 64 et 126.
  - Utilisation de trames UI typées de SAPI 63 TEI 127 pour le protocole d'attribution d'adresse.
    - Demande d'identité
    - Identité refusée
    - Vérification d'identité
    - Réponse à la vérification
    - Suppression d'identité.
    - Demande de vérification.

LAPD : Résumé des trames utilisées

- **SABME** : Ouverture de connexion mode étendu.
- **UA** : Acquittement non numéroté.
- **DISC** : Fermeture de connexion.
- **DM** : Indication de mode déconnecté.
- **XID** : Echange d'identificateur.
- **UI** : Information non numérotée.
- **I** : Trame d'information.
- **RR** : Acquittement explicite.
- **RNR** : Suspension temporaire.
- **REJ** : Rejet d'une suite de trames information.

II.2
Protocole PPP
("Point to Point Protocol")

1 Généralités
2 La transmission de données
3 La configuration de liaison
4 La configuration de réseau
5 La compression d'entêtes

Protocole PPP
("Point to Point Protocol")

II.2.1
Généralités

A) Historique
B) Objectifs généraux
C) Architectures
D) Organisation du protocole

A) Historique des protocoles de liaison point à point avec IP

- **IP** (1980) : défini pour l'interconnexion de **réseaux**.
- **Besoin d'une adaptation** des paquets IP aux différents réseaux visés: IP sur réseaux locaux (Ethernet, …)
  - IP sur réseaux longues distance (X25)
- Besoin d'un **protocole de liaison point à point** pour acheminer les paquets IP sur voies séries.
  - **Solutions propriétaires** très simples (début 1980).
  - **Normalisation minimum: SLIP** (1984,RFC en 1988)
  - Création d'un groupe de travail IETF **pour une solution complète : PPP** (RFC 1134 nov1989)
- Améliorations successives de PPP : RFC complémentaires.
  - Version en cours **RFC 1661** (juillet 1994)

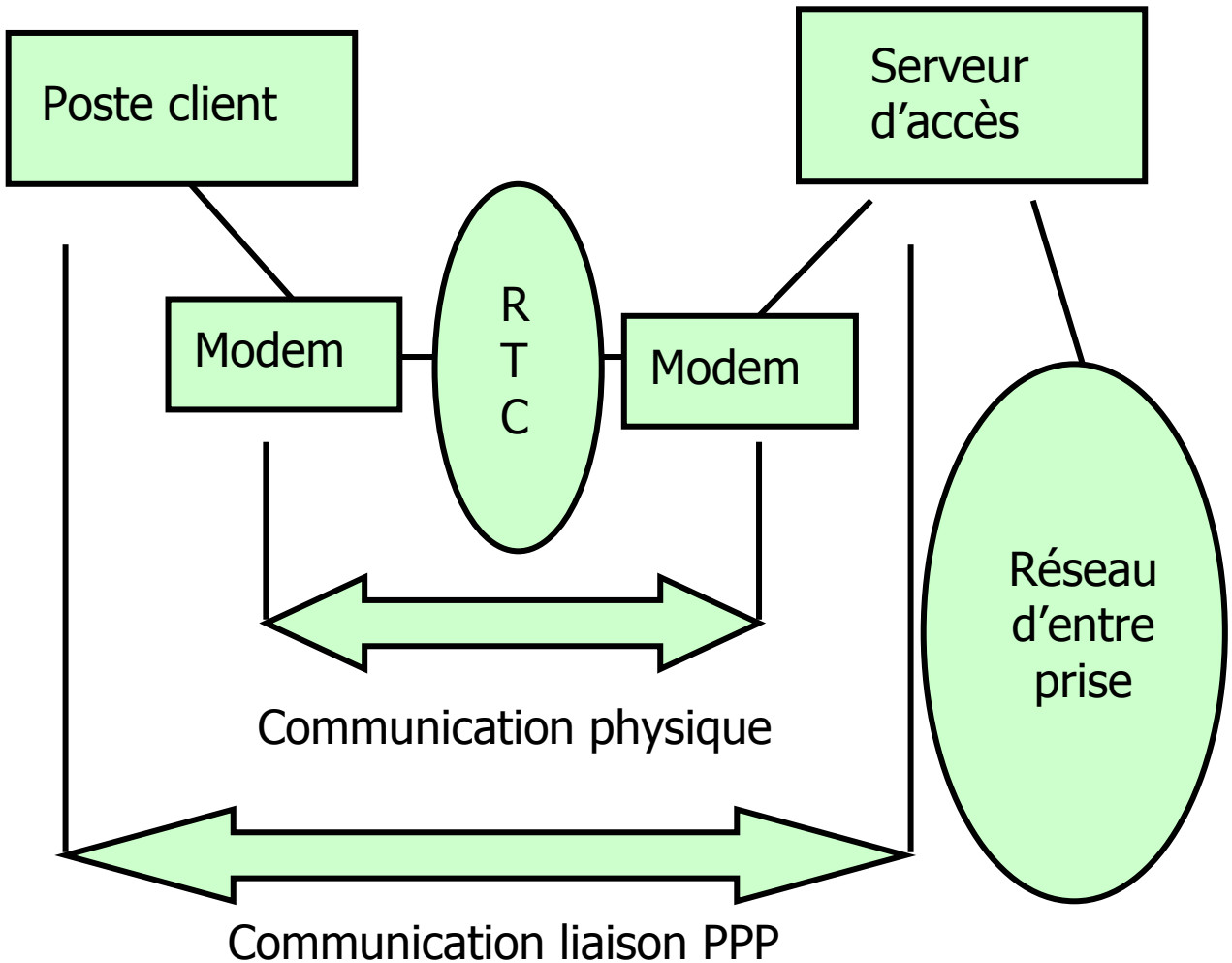
A) Protocole SLIP (RFC 1055 juin 1988) ("Serial Line Internet Protocol")

- **Solutions propriétaires 3COM, SUN** 1984 (R. Adams)
  - SLIP Implanté en 1984 sur BSD. RFC en 1988.
- **Solutions adoptées dans SLIP:**
  - **Délimitation en transparence caractère**
    - Définition d'un caractère de fin de trame "END" (0xC0) et d'une transparence: si END apparaît dans les données séquence d'échappement 0xDB, 0xDC.
    - Si 0xDB apparaît dans les données => émission 0xDB, 0xDB
  - **Uniquement prévu pour transporter de l'IP**
  - **Amélioration** : compression des entêtes (RFC1144 Van Jacobson) => CSLIP ('Compressed SLIP').
- **SLIP: un protocole qui a été quand même très utilisé.**

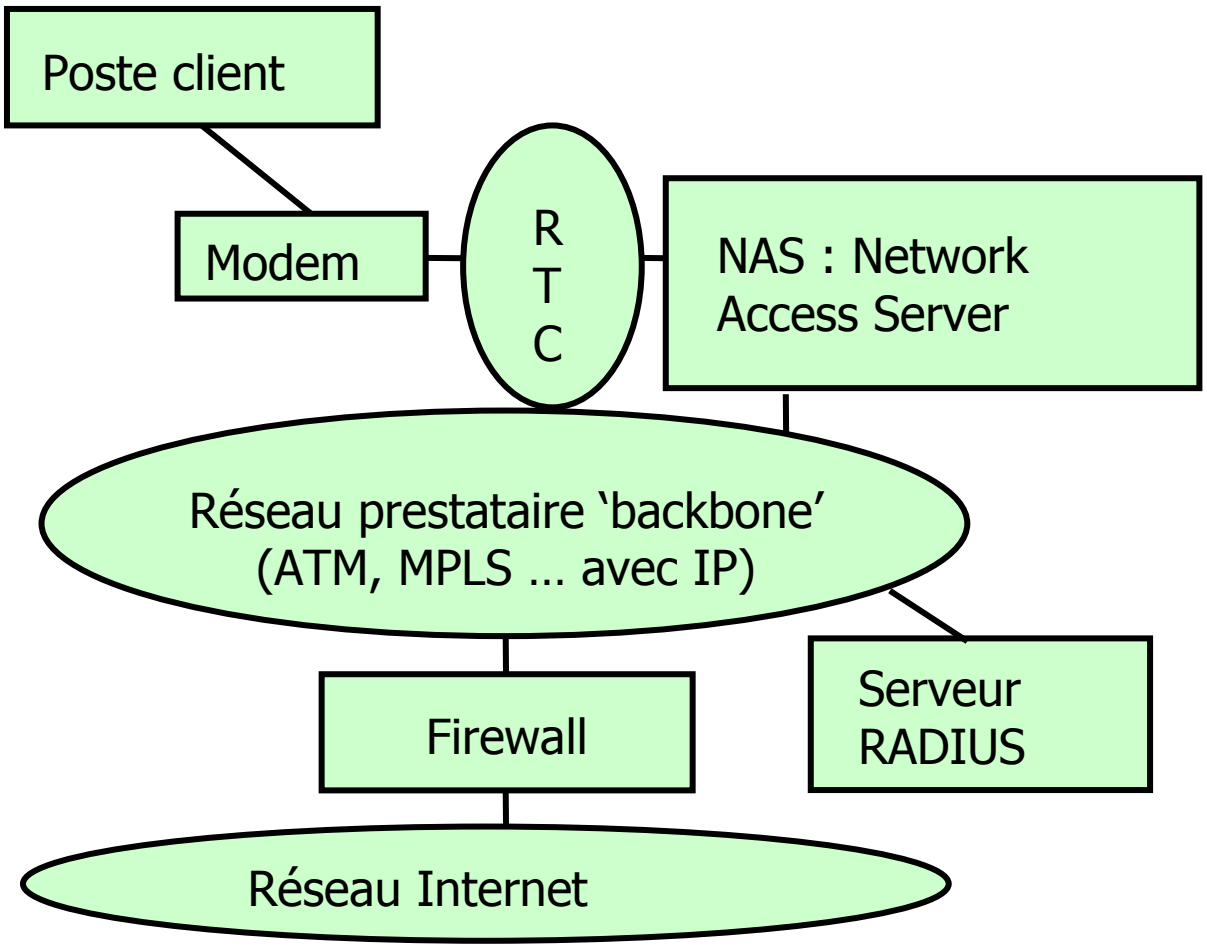
B) Objectifs généraux de PPP

- **Pallier les insuffisances** de SLIP
  - **Pas de** multiplexage, de traitement d'erreurs, d'affectation d'adresse IP, d'authentification d'accès=> Protocole presque vide
  - **La RFC n'a jamais été approuvée définitivement.**
- **PPP : une solution universelle** pour la **communication** au niveau **liaison en point à point.**
  - **Grande variété d'appareils visés :** hôtes, routeurs, points d'accès
  - G**rande variété de protocoles** de niveau 3 (multiplexage de flots d'origines très diverses Internet, Appletalk, IPX, …)
  - **Grande variété de voies de communication** tous débits.
    - Liaisons spécialisées séries synchrones, asynchrones (ADSL…).
    - Architectures de réseaux pouvant être utilisées comme des voies de communication point à point: RTC, RNIS, X25, FR, ATM, réseaux locaux.
  - **Implantation de solutions** pour de très nombreux problèmes.

C) Architectures de réseaux avec PPP : Solution de base



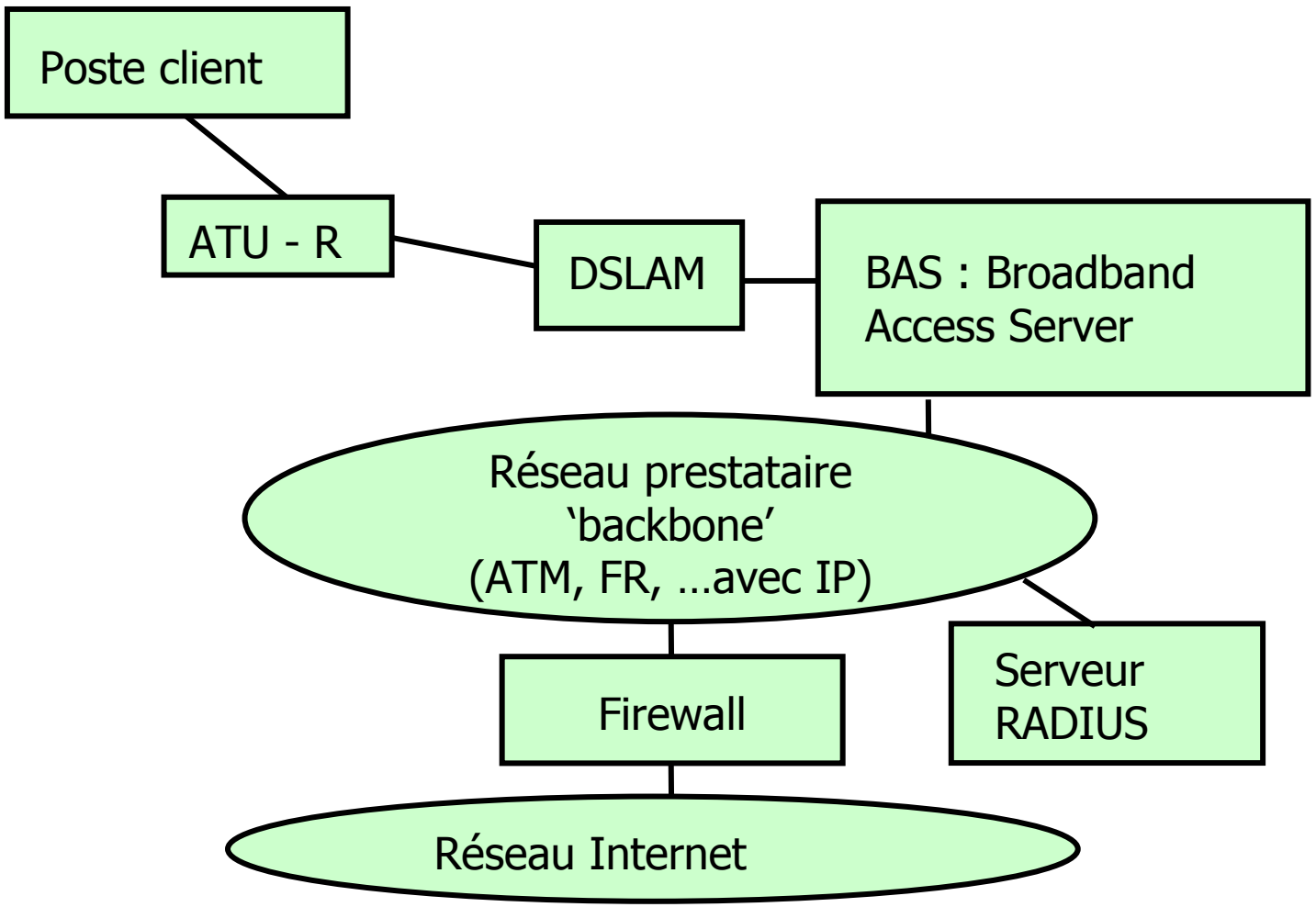
C) Architecture de fournisseur d'accès Internet (FAI, ' ISP ')



C) Notion de NAS : ‘ Network Access Server ’

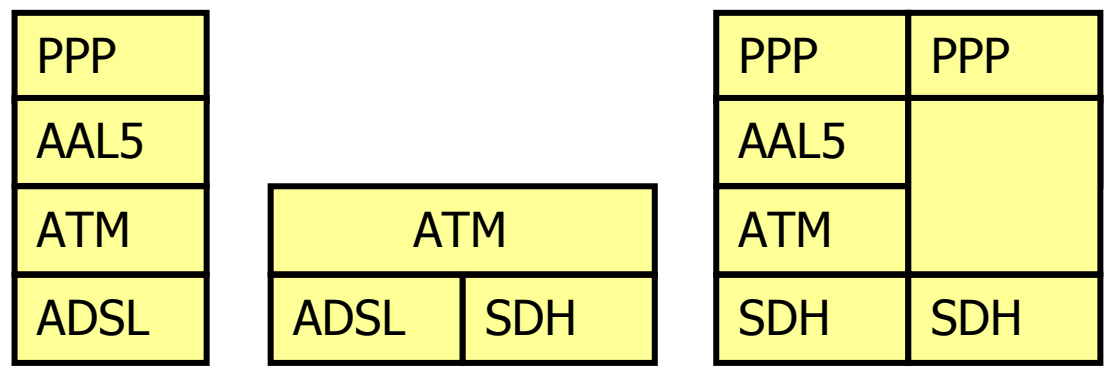
- **NAS (‘Network Access Server’) :** Systèmes dédiés supportant de nombreux types d’accès physiques séries (modems V90, canaux B, T2 , V35, Ethernet, …).
- **NAS : relais de négociation en PPP** pour l’accès Internet : l’adresse IP, type de compression, …
- **NAS :** installés sur tout le territoire (dans les autocoms).
- **Utilisation d’un serveur centralisé** pour l’authentification et la comptabilité (RADIUS).
- **NAS :** achemine ensuite les données en PPP/IP sur tous les types de média voulus (ATM, Ethernet, SDH, FR) avec le poste serveur.

C) Architecture de fournisseur d'accès avec ADSL



C) Notion de DSLAM et de BAS

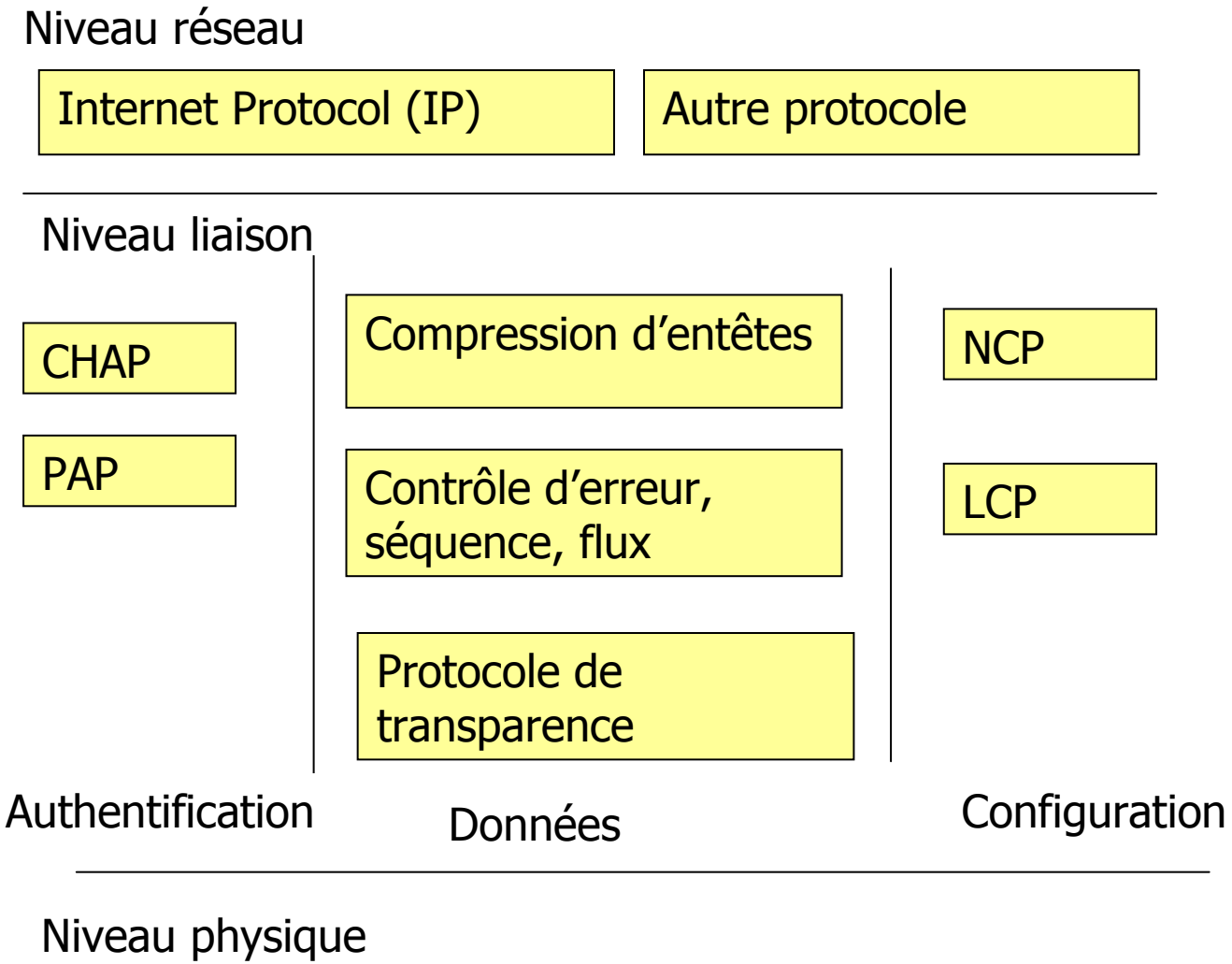
- **ATU-R 'A**DSL **T**ransceiver **U**nit- **R**emote terminal end' : Le modem (coté usager)
- **DSLAM 'D**igital **S**ubscriber **L**ine **A**ccess **M**ultiplexer' : un multiplexeur de voies ADSL (avec modem coté prestataire).
- **BAS** '**B**roadband **A**ccess **S**erver' : le NAS pour voies ADSL.



D) Organisation générale de PPP : les grandes parties

- **Protocole de transmission de données :**
  - Un protocole pour communiquer (encapsuler des datagrammes provenant de plusieurs protocoles de niveau réseau …).
- **Protocole de contrôle de liaison** :
  - LCP : 'Link Control Protocol'
  - Un protocole pour établir, configurer, tester une connexion de liaison.
- **Protocoles de contrôle réseau** :
  - NCPs : 'Network Control Protocols'
  - Une famille de protocoles pour établir, configurer des paramètres pour les protocoles de niveau réseau.
- **Protocoles d'authentification** :
  - Une famille de protocoles pour contrôler l'accès au réseau.

D) Organisation générale de PPP : la suite des protocoles PPP



Protocole PPP
("Point to Point Protocol")

II.2.2
La transmission des données

A) Mécanismes de transparence

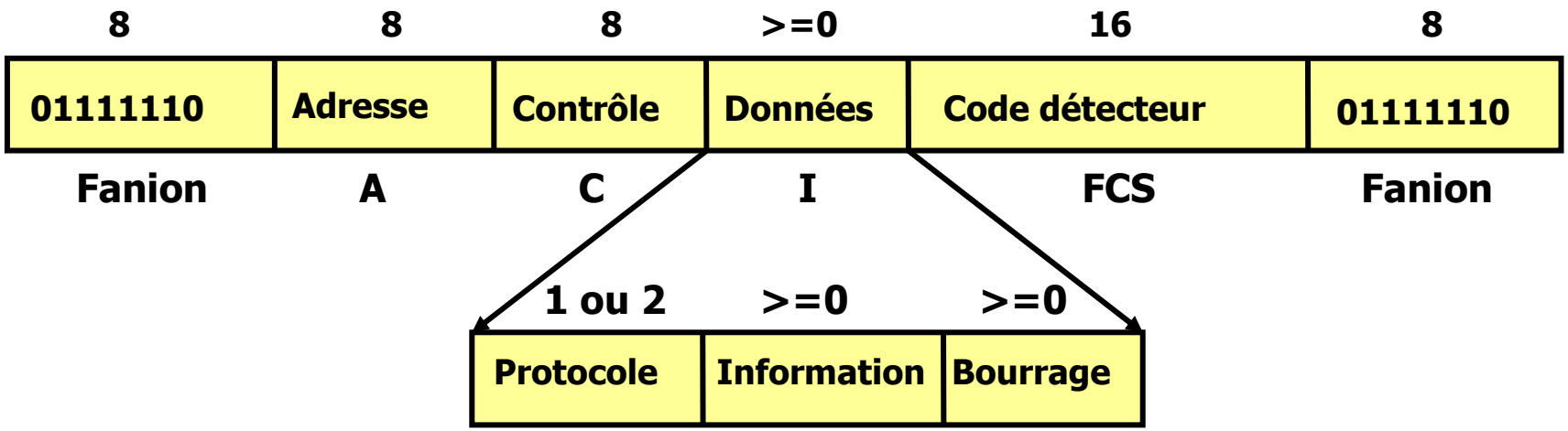
B) Contrôle d'erreur, de flux, de séquence.

C) Multiplexage (encapsulation multi-protocole)

Introduction : organisation générale de la transmission des données

- **Protocole de transparence**
  - Doit fonctionner avec les principales voies de communications existantes (synchrones à trame de bits, asynchrones avec format caractères) => Deux sortes de transparence.
- **Protocole de contrôle d'erreur, de flux, de séquence**
  - Très voisin de LAPB.
- **Multiplexage (encapsulation multi protocole)**
  - PPP permet le multiplexage de différents flots provenant de différents niveaux réseaux (codés sur 2 octets).
- **Compression des entêtes** (Van Jacobson RFC 1144).
  - Compression des entêtes PPP : Recherche d'un surcoût minimum.
  - Ex: PPP utilise 8 octets pour le tramage pouvant être réduits à 2 ou 4 octets lorsque des mécanismes de compression sont utilisés.
  - Compression des entêtes IP, TCP.

Introduction : Format de la trame PPP



- **Reprend le format** de la trame HDLC.
- **Ajoute une possibilité** d'encapsulation multi-protocole.

A) Délimitation (mécanismes de transparence) : Généralités

- **Adaptation à la voie physique (RFC1662 'PPP in HDLC Framing')**
- **Deux types de voies** => deux méthodes différentes de transparence:
  - **Voies synchrones au niveau bit** ('bit synchronous')

    => Transparence binaire HDLC.

    Comme dans les protocoles à trames de bits on rajoute un 0 après tout séquence de 5 bits à 1 **('bit stuffing').**

  - **Voies asynchrones par octet** ou **synchrone au niveau octet**

    => Transparence caractère PPP.

Transparence caractère : Le mode par défaut

- **Délimiteur de trame**
  - Comme en HDLC fanion 01111110 en hexa 0x7F.
- **Caractère d'échappement ('escape')**
  - Caractère 01111101 en hexa 0x7d.
- **Caractères de contrôle** : Ils dépendent de la voie utilisée.
  - Les caractères de contrôle soumis au mécanisme de transparence sont définis par une table de bits **ACCM** ('**Async Control Character Map**').
  - Si le bit ACCM est à 1 le caractère associé est remplacé par une séquence d'échappement de deux caractères:
    - Le caractère escape.
    - Le caractère de contrôle en ou exclusif avec 0x20.

Transparence caractère (suite)

- Les caractères compris entre 0 et 31 (0x00 et 0x20) **sont en général réservés au pilotage des modems**.
- **Si on veut les utiliser au niveau liaison,** il faut les mettre dans la table **ACCM**.
- **Exemples d'application de la transparence PPP:**
  - 0x05 est codé 0x7d, 0x25. (Contrôle modem code 5)
  - 0x7e est codé 0x7d, 0x5e. (Flag Sequence)
  - 0x7d est codé 0x7d, 0x5d. (Control Escape)
  - 0x03 est codé 0x7d, 0x23. (ETX)
  - 0x11 est codé 0x7d, 0x31. (XON)
  - 0x13 est codé 0x7d, 0x33. (XOFF)

B) Contrôle d'erreur, de flux, de séquence : Choix PPP deux solutions

- **1 Solution standard (par défaut) :**

- Transmission **non fiable**,
- Pas de contrôle d 'erreur, de flux, de séquence.

- **2 Solution fiable (en option) :**

- Transmission **fiable** en mode connecté
- Contrôle d'erreur, de flux, de séquence

Contrôle d'erreur, de flux, de séquence : Transmission non fiable

- **RFC 1662 'PPP in HDLC Framing' : Mode par défaut.**
- **Protocole de liaison sans connexion.**
- Pas de de contrôle **d'erreur**, de **séquence**, de **flux**.
- Les trames incorrectes (FCS faux) sont **détruites.**
- **Une seule trame UI 'Unnumbered Information'**

Fanion 0x7E	Adresse 0xFF	Contrôle 0x03	Données	FCS	Fanion 0x7E

- **Fanion :** un octet (0x7e) pour la synchro trame. Un seul fanion entre deux trames.
- **Adresse :** un octet (0xff), adresse diffusion en multipoint ('All-Stations address'**).**
- **Champ Contrôle :** un octet (0x03), type 'Unnumbered Information' (UI) avec bit Poll/Final (P/F) bit à zéro.
- **Champ code polynomial :** Frame Check Sequence (FCS) deux octets. Possibilité de négocier un code sur 32-bit (quatre octets).

Contrôle d'erreur, de flux, de séquence : Transmission fiable

- **Transmission fiable :** 'Numbered Mode ' RFC 1663 PPP Reliable Transmission
- **Mode négociable** en début de transmission :
  - Si l'on considère que la liaison n'est pas assez **fiable.**
  - Si l'on veut éviter des **problèmes** avec la **compression**.
- **Protocole défini en fait** par la norme **ISO 7776** (Description of the X.25 **LAPB-Compatible** DTE Data Link Procedure).

- **Remarque :** Possibilité d'utiliser des tailles de fenêtre de 1 à 127, modes d'ouverture de connexion SABM (1 à 7) ou SABME (1 à 127).

C) Multiplexage et zone de données

- **Format de la zone données de la trame**
  - Type de protocole réseau transporté
  - Charge utile (paquet réseau, LCP, …)
  - Bourrage
- **Type de protocole réseau transporté**
  - **Valeur sur deux octets** définie par le RFC1340 définissant les codes des protocoles autorisés (IANA-ICANN).
  - **Quelques exemples de valeurs possibles**

    0x0021 IP 0x002B IPX

    Ox8021 IPCP 0xC021 LCP

    0x002D TCP/IP avec compression

    0x002F TCP/IP Sans compression
  - **Codage du type de protocole** sur deux octets ramené à un octet par négociation si nécessaire.

Zone données : taille maximum et bourrage

- **Zone des données usager :** longueur maximum en PPP
  - Terminologie liaison : **MRU** 'Maximum Receive Unit'
  - **Valeur négociable** à l'établissement de la liaison : valeur par défaut = 1500 octets.
  - **Fragmentation** par le niveau réseau.
- **Bourrage**
  - Si le médium de transmission utilise un **format fixe** (taille de paquets, de cellule ATM, …)
  - et que cette taille obligatoire de la trame ne correspond pas à la taille de l'information à transporter.

Protocole PPP
("Point to Point Protocol")

II.2.3
La configuration de liaison
(LCP ‘ Link Configuration Protocol ’

A) Gestion de connexion avec LCP

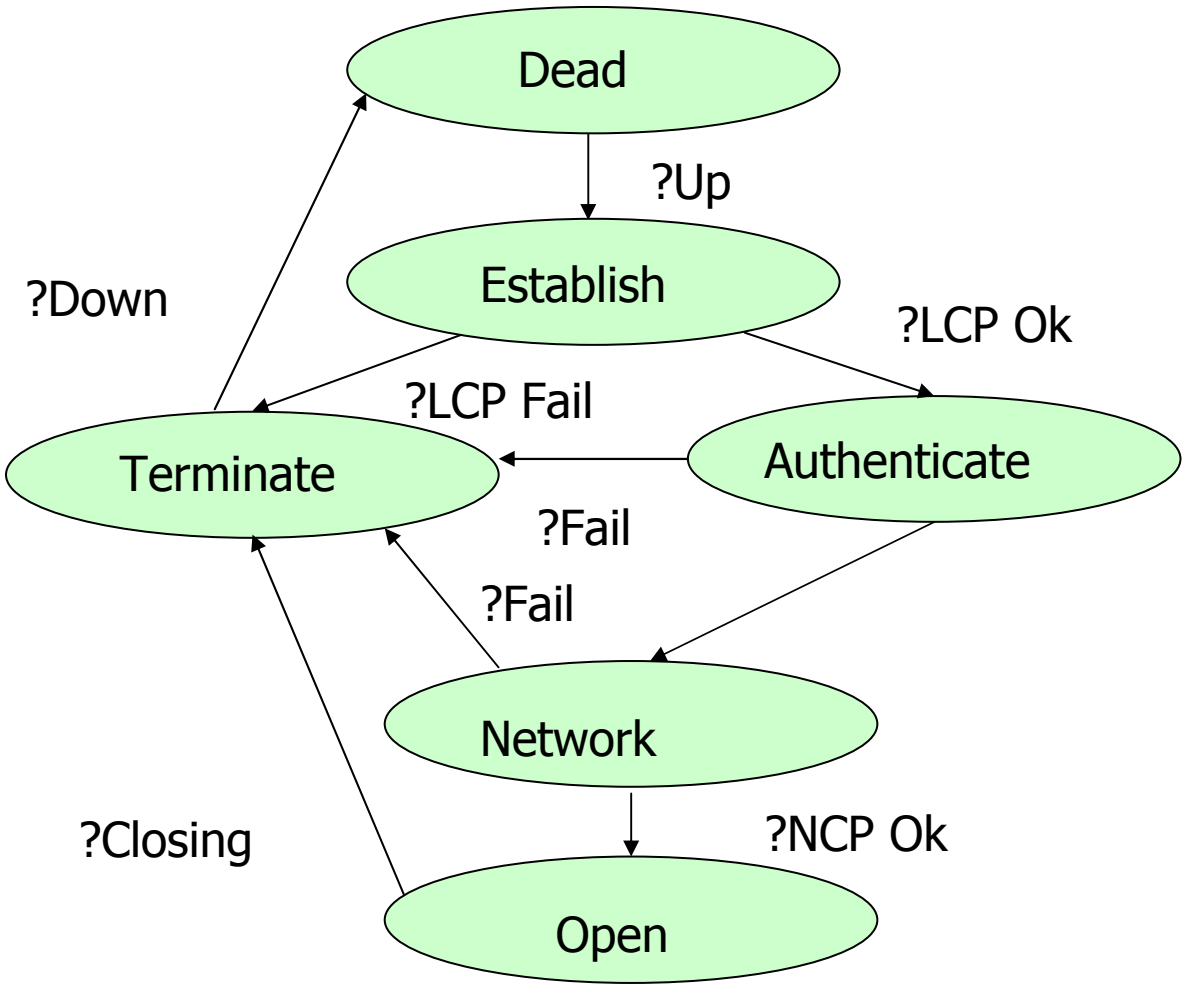
B) Le protocole LCP

C) Les options de configuration LCP.

A) LCP et la gestion de connexion

- **LCP permet** d'ouvrir, de fermer la liaison PPP : **gestion de connexion.**
- **LCP permet de négocier** des paramètres de fonctionnement à l'ouverture (puis de renégocier),
  - Existence de **paramètres par défaut** automatiquement échangés au début entre pairs sans intervention opérateur.
  - **Configuration possible par l'opérateur** à l'ouverture.
  - **Exemples de négociation**
    - Définir le format d'encapsulation (négociation de la compression)
    - Définir la taille maximale des trames (taille MRU).
- **LCP permet de détecter** certaines conditions d'erreur (liaison en fonctionnement correct, en panne, en boucle)

Automate LCP



Établissement d'une connexion PPP : Commentaire des différentes phases (1)

- **Liaison non opérationnelle** (‘Link Dead’)
  - Le niveau physique n’est pas prêt.
  - Un événement externe (détection de porteuse, démarrage opérateur, …) permet de passer à l’état prêt.
- **Liaison en cours d’établissement** (‘Link Establishment phase’).
  - Le LCP (‘Link Control Protocol’) établit les paramètres de liaison.
- **Authentification** (‘Link Authentication Phase’)
  - L’authentification (si elle est demandée) prend place aussitôt que possible après établissement des paramètres de liaison.
  - Si l’authentification échoue on termine.

Établissement d'une connexion PPP : commentaire des différentes phases (2)

- **Négociation des paramètres de réseau** ('Network-Layer Protocol Phase').
  - Chaque niveau réseau (comme IP, IPX, ou AppleTalk) configure ses propres paramètres ('Network Control Protocol').
- **Ouvert** ('Open Phase')
  - Après avoir atteint cet état PPP peut transporter les paquets de données.
- **Terminé** ('Link Termination Phase')
  - PPP termine dans différents cas: perte de porteuse, mauvaise qualité, mauvaise authentification, expiration d'un délai d'inactivité, fermeture décidée par l'opérateur.
  - LCP échange des paquets de terminaison.
  - Informe le niveau réseau de la fermeture.

B) Le protocole LCP

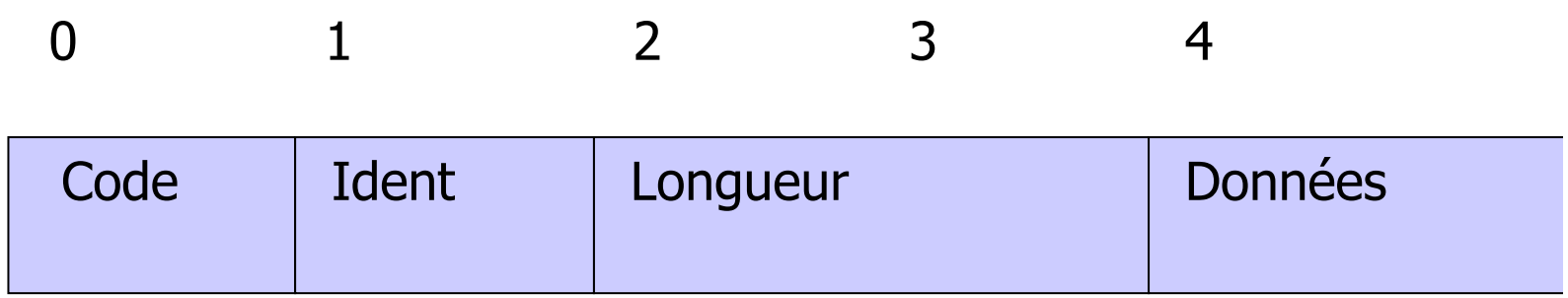
Protocole qui permet principalement la négociation des options de configuration d'une liaison PPP.

- Existence d'une configuration **par défaut**.
- LCP permet de **modifier ces options**.
- Chaque **extrémité propose ses options.**

Principales notions

- Définition des messages LCP et principes de la négociation.
- Définition des **options (attributs) négociables.**
- Les messages LCP sont encapsulés dans **la zone information d'une trame PPP (type de protocole C021).**

Protocole LCP : Format de la trame LCP



- **Représentation uniquement de la charge utile LCP.**
- **Code (' code')** : sur un octet le type LCP.
- **Identificateur (' Identifier')** : sur un octet il permet d'associer les requêtes et les réponses.
- **Longueur ('Length') : s**ur deux octets, la longueur inclut le code, l'identificateur et la donnée.
- **Données ('Data')** : la zone données est vide ou son format est défini par le code type : contenu essentiel les valeurs négociées.

Protocole LCP : Liste des types LCP (codes)

Code	Désignation du paquet
1	Configure-Request
2	Configure-Ack
3	Configure-Nak
4	Configure-Reject
5	Terminate-Request
6	Terminate-Ack
7	Code-Reject
8	* Protocol-Reject
9	* Echo-Request
10	* Echo-Reply
11	* Discard-Request
12	* RESERVED

Codes valables pour IPCP et LCP

*** Codes valables pour LCP seulement**

Protocole LCP : Description détaillée de types LCP (1)

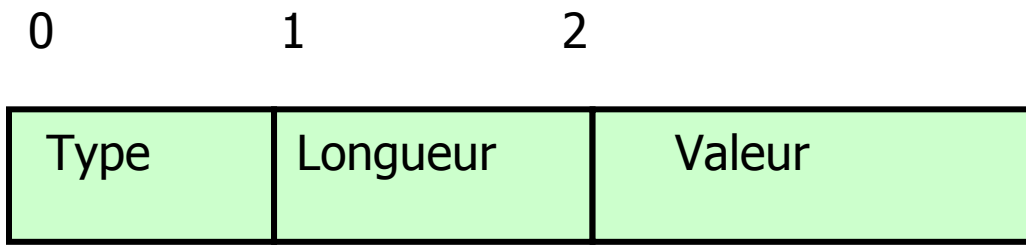
- **Configure-Request**
  - Pour ouvrir une connexion : le paquet Configure-Request contient toutes les options que l'on modifie par rapport aux valeurs par défaut
- **Configure-Ack**
  - Si toutes les options de configuration sont reconnues et acceptées réponse : configure-Ack.
- **Configure-Nak**
  - Si toutes les options de configuration sont reconnues mais certaines ne sont pas acceptables: réponse Configure-Nak.
  - Le champ données contient les valeurs de configuration non acceptées
- **Configure-Reject**
  - Si certaines options de configuration ne sont pas reconnues ou ne sont pas acceptables dans le cadre d'une négociation prévue par l'administrateur réseau alors la réponse est un 'configure-Reject'.

Protocole LCP : Description détaillée de types LCP (2)

- **Terminate-Request et Terminate-Ack**
  - Pour fermer une connexion LCP.
- **Code-Reject :** Type LCP inconnu (champ code).
- **Protocol-Reject :** Type de protocole inconnu (champ proto).
- **Echo-Request et Echo-Reply**
  - Permet de tester une liaison PPP : échange d'un nombre magique sur 4 octets caractéristique de l'émetteur si une valeur a été négociée (sinon 0). Le nombre magique doit être celui du site distant. Si c'est celui du site local il y a une boucle.
- **Discard-Request**
  - Outil de test de liaison : une émission simple, local vers distant, avec destruction immédiate du paquet.
  - Utilisation : Déverminage, test de performance, …
  - Contient un nombre magique s'il a été négocié (sinon 0).

C) Valeurs négociables : Options de configuration

- **Données négociables :** codées type, longueur, valeur



- **Différents types de données négociables**

0	RESERVED	7	Protocol-Field-Compression
1	Maximum-Receive-Unit	8	Addre-and-Contr-Field-comp
2	Async-Control-Character-Map	9	FCS Alternatives
3	Authentication-Protocol	10	Padding protocol
4	Quality-Protocol	11	Numbered mode
5	Magic-Number		etc …..
6	Reserved		

Description de quelques options (1)

- **Maximum-Receive-Unit (MRU)**
  - La valeur de la taille maximum par défaut est de 1500 octets. Une implantation doit toujours être capable de recevoir cette taille.
  - Par cette négociation on peut indiquer au site distant que l'on peut recevoir des paquets de plus grande taille ou que l'on demande la transmission de paquets de plus petite taille.
- **Async-Control-Character-Map**
  - Permet de redéfinir la tables des codes caractères qui seront soumis à la transparence caractères.
  - La table est sur 32 bits (4 octets) et concerne les codes caractères de 0x0000 à 0X0020.
- **Authentication-Protocol**
  - Permet de définir le protocole d'authentification utilisé :
  - PAP (code protocole 0xC023, CHAP (code 0xC223).

Description de quelques options (2)

- **Quality-Protocol**
  - Pour négocier le type de protocole de gestion de la qualité de la liaison (valeur sur 2 octets). Principal choix: LQR 'Link Quality Report' 0xC025
- **Magic-Number**
  - Pour detecter les liaisons qui bouclent ('looped-back links').
  - Chaque coté tire aléatoirement un nombre aléatoire sur 4 octets (le nombre magique qui doit changer à chaque nouvelle ouverture.
- **Protocol-Field-Compression (PFC)**
  - Pour demander la compression de la zone protocole (de 2 à 1 octet).
  - Le FCS est alors calculé sur la trame compressée (pas sur la trame originale non compressée).
- **Address-and-Control-Field-Compression (ACFC)**
  - Pour demander la compression des zones adresses et contrôle dans les trames PPP. Le FCS est calculé sur la trame compressée.

Protocole PPP ("Point to Point Protocol")

II.2.4

La configuration de réseau

(NCP ‘ Network Configuration Protocol ’

Introduction NCP

- **NCP permet la négociation d'options** de configuration du **niveau 3 réseau** dans le cadre du niveau 2 liaison.
- **Fonctions nécessairement dépendantes** du protocole réseau utilisé => Existence de protocoles **spécifiques** NCP par type de réseau.
- **Exemples de RFC NCP existantes :**

  RFC 1332 IPCP pour IP

  RFC 2023 IPV6CP pour IPV6

  RFC 1552 IPXCP pour IPX

  RFC 1378 ATCP pour AppleTalk

  RFC 1377 0SINLCP pour OSI

  RFC 2097 NBFCP pour NetBeui ….. Etc

Le protocole IPCP ‘IP Configuration Protocol’

- **Protocole qui utilise les messages de LCP** (types 1 à 7) pour négocier des options relatives à IP:
- **Exemple 1) Type d’option 2: IP-Compression-Protocol**
  - Une façon de négocier la compression des entêtes IP : théoriquement un code sur deux octets permet de sélectionner la compression souhaitée => Pratiquement deux possibilités : **002d** Compression TCP/IP Van Jacobson ou par défaut pas de compression.
- **Exemple 2) Type d’option 3 : IP-Address**
  - L’émetteur peut dans un Configure-Request demander d’utiliser une adresse IP s’il en connaît une (0 sinon).
  - Le site distant peut accepter l’adresse proposée ou rejeter par un NAK en retournant une autre adresse.

  => Deux zones adresses IP (proposée et affectée) pour la négociation,
- **Affectation d’adresses plus utilisée** => DHCP

Protocole PPP ("Point to Point Protocol")

Conclusion

Raisons du succès du protocole PPP

- **Hégémonie du protocole IP**

=> PPP est obligatoirement le niveau liaison le **plus répandu.**

- **Mais aussi beaucoup de qualités**
  - Protocole de convergence adapté à tout
    - **pour toutes les voies point à point et tous les réseaux .**
    - **pour toutes les architectures de réseau et tous les protocoles**.
  - **Rassemble** la plupart des concepts et solutions..
  - **Enrichit** les fonctions habituellement dévolues au niveau liaison par des fonctions d'administration et de sécurité (LCP, NCP, authentification).
  - **En constante amélioration.**

Conclusion

Évolution des protocoles de liaison industriels

- **Historiquement : grande variété de propositions** qui diffèrent souvent peu : des options jugées nécessaires dans le domaine visé.

  => Trop grande hétérogénéité.

- **Une orientation des protocoles les plus anciens vers des solutions assez riches** en termes de contrôle d'erreur, de flux, de séquence

  => Solutions jugées plutôt coûteuses justifiées par les taux d'erreurs.

- **Changement d'orientation** avec Internet :
  - Redistribution des fonctions en s'orientant vers un découpage ou le rôle de contrôle d'erreur, de flux, de séquence est allégé si nécessaire.
  - Les fonctions d'administration sont renforcées.

Bibliographie : Cours de liaison point à point

- A. S. Tannenbaum 'Computer Networks' Prentice Hall.
- W.R. Stevens "TCIP/IP Illustrated, The protocols" , Addison Wesley.
- L. Toutain 'Réseaux locaux et Internet' Hermés.
- Sites web et RFC.

Niveau Liaison

RÉSEAUX LOCAUX
“LAN, Local Area Networks”

Introduction
1 Réseaux locaux partagés
2 Réseaux locaux commutés.
Conclusion

Introduction: caractéristiques communes des réseaux locaux

- Un réseau local dessert un **ensemble** de stations.
- Problème principal posé => **accès multiple.**
- Selon des protocoles situés aux **niveaux physique et liaison**.
- Un réseau local ne dessert **qu'une organisation** correspondant à un domaine privé. Il échappe généralement aux contraintes d'un opérateur de télécommunications

  => Notion de **réseau local d'entreprise ou de réseau local domestique**
- Problème induit par les réseaux locaux: **l'interconnexion des réseaux locaux**.

Caractéristiques techniques communes des réseaux locaux

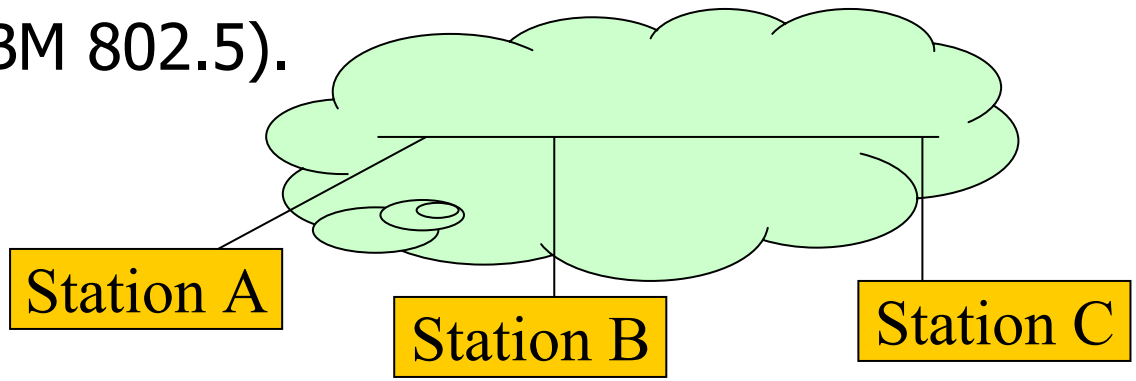
- 1) Le diamètre de la surface desservie **dépend de la technologie adoptée**: il n'excède pas généralement pas **quelques kilomètres.**

- 2) Le débit binaire nominal est au minimum mesuré en **dizaines de mégabits par seconde** (jusqu'à 10 gigabits/s) avec des **taux d'erreurs faibles**.

- 3) **Normalisation** : IEEE 8.02 (**I**nstitute of **E**lectrical and **E**lectonics **E**ngineers) - ISO 8802.

Les deux méthodes de réalisation : a) Réseaux locaux partagés ‘shared’

- **1) Utilisation d’une voie commune multipoint**
  - Bus série (Ethernet IEEE 802.3).
  - Bande de fréquence hertzienne (WIFI 802.11).
  - Boucle (IBM 802.5).



- **2) Partage de la voie commune multipoint**: le problème du contrôle de l'accès au médium (MAC "Médium Access Control") consiste à déterminer la station qui, à un instant donné, a le droit d’émettre.

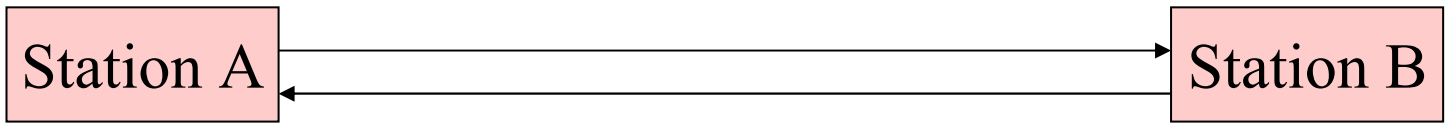
Les deux méthodes de réalisation : b)Réseaux locaux commutés 'switched'

- **1) Utilisation de techniques de commutation** pour faire communiquer des stations au niveau liaison (niveau 2).
- 2) Exemples de techniques de commutation employées :
  - Commutation **temporelle asynchrone** (à mémoire partagée, à médium partagé).
  - Commutation **spatiale** (matrices d'aiguillages).
- 3) Pour des raisons de compatibilité, les réseaux locaux commutés offrent , **les mêmes méthodes d'accès, l'adressage, les mêmes formats de trames** que les réseaux partagés => possibilité de mixage des approches partagées et commutées.

Réseaux locaux partagés: mode de fonctionnement ‘ half duplex ’

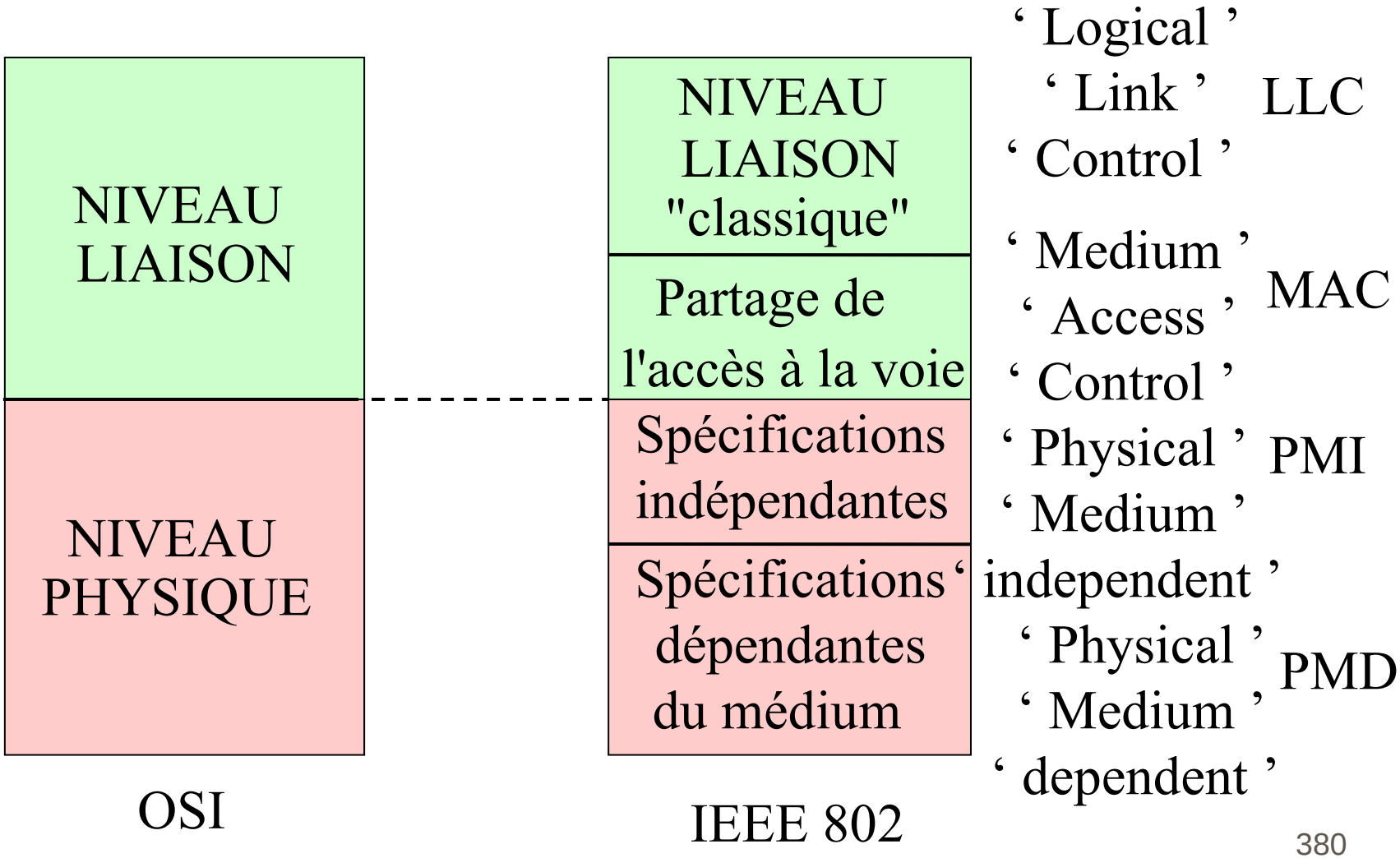
- **Existence d’une voie unique de communication partagée**
  - **Une seule** station peut **émettre** à un instant donné
  - Vers **un ou plusieurs destinataires** (médium à diffusion)
  - Mode de communication demi-duplex: **half duplex**.
- Limitation du **débit de transmission** de l’ensemble des stations au **débit du médium partagé**.
- **Contraintes de distances** liées au médium, au codage des données et au protocole de partage.

Réseaux locaux commutés: mode bidirectionnel (‘ full duplex ’)



- **Connexion directe entre deux stations ou entre une station et un commutateur (câblage en étoile)**
  - Une seule station connectée => **pas de partage de voie commune.**
  - Une station peut **émettre** à un instant donné vers le commutateur et **recevoir** en même temps.
  - Mode de communication **bidirectionnel simultané** (**full duplex**) possible.
- **Possibilité de parallélisme : débit plus important.**
- **Moins de contraintes de distance**.

Découpage en couches lié aux réseaux locaux IEEE 802



Réseaux locaux

Réseaux locaux partagés

1 Critères de classification des méthodes de partages.

2 Réseaux en compétition.

3 Réseaux Ethernet.

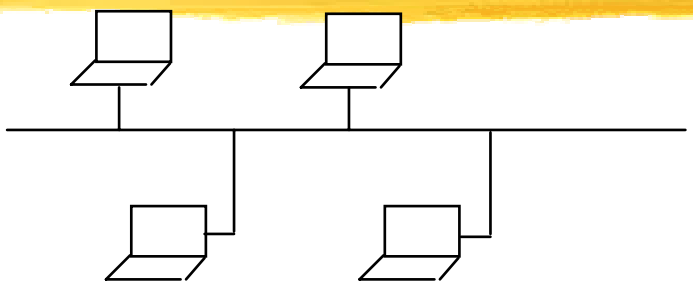
4 Réseaux WIFI.

Réseaux locaux partagés

Critères de classification des réseaux locaux partagés

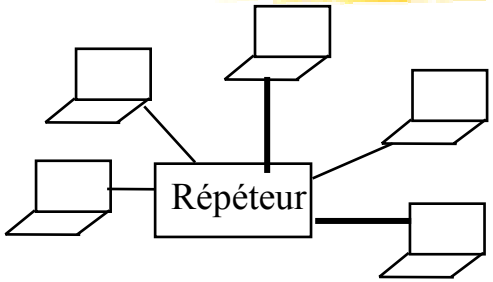
- **Critères qualitatifs**
- **Critères de performance**
- **Critères de sûreté de fonctionnement**

Critères qualitatifs : Topologie Bus ou voie hertzienne



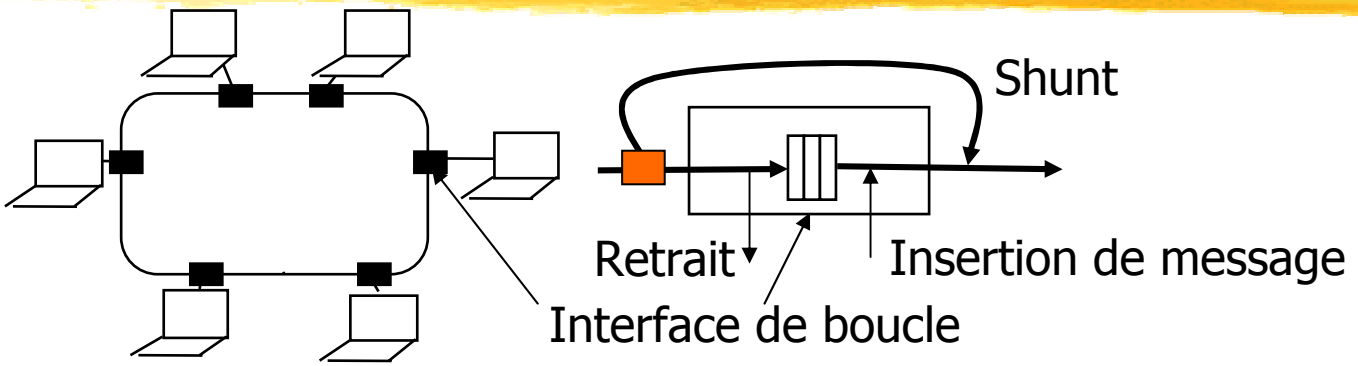
- Chaque station est **directement reliée aux autres par un canal unique** de communication (bus série coaxial ou voie hertzienne).
- Un message véhiculé par le canal **peut-être reçu par toutes les stations**: diffusion très facile à réaliser mais notion de promiscuité ' promiscuous mode' => problèmes de sécurité.
- Une station vérifie, **d'après l'adresse contenue dans le message**, si le message lui est destiné.
- **Médium passif** : électronique plus simple, moins de pannes.
- **La bande passante** de la voie limite les performances.
- **Exemples: Ethernet** (sur câble)**, WIFI** (sans fil).

Critères qualitatifs : Topologie Étoile



- Un concentrateur (un 'hub') **relie les stations**.
- Un concentrateur est un répéteur de signal: "**Repeater Hub**".
- Les répéteurs peuvent être **interconnectés** (arbre).
- Le câblage en étoile permet de **découpler chaque station du reste du réseau**.
- Problème de la **fiabilité** du répéteur: électronique active.
- **Exemple: répéteur Ethernet.**
- **Le câblage type** en matière de réseaux locaux partagés.

Critères qualitatifs : Topologie Boucles ou Anneaux (1)



- Les stations sont rattachées au moyen **d'interfaces** selon **une topologie en boucle**.
- **Une interface de boucle retarde le message dans un registre et régénère le signal.**
- Un message envoyé par une station **fait un tour complet** et est **retiré par son émetteur**.
- **L'adresse destinataire permet de déterminer** si une interface donnée doit prélever le message ou non.

Critères qualitatifs : Topologie Boucles ou Anneaux (2)

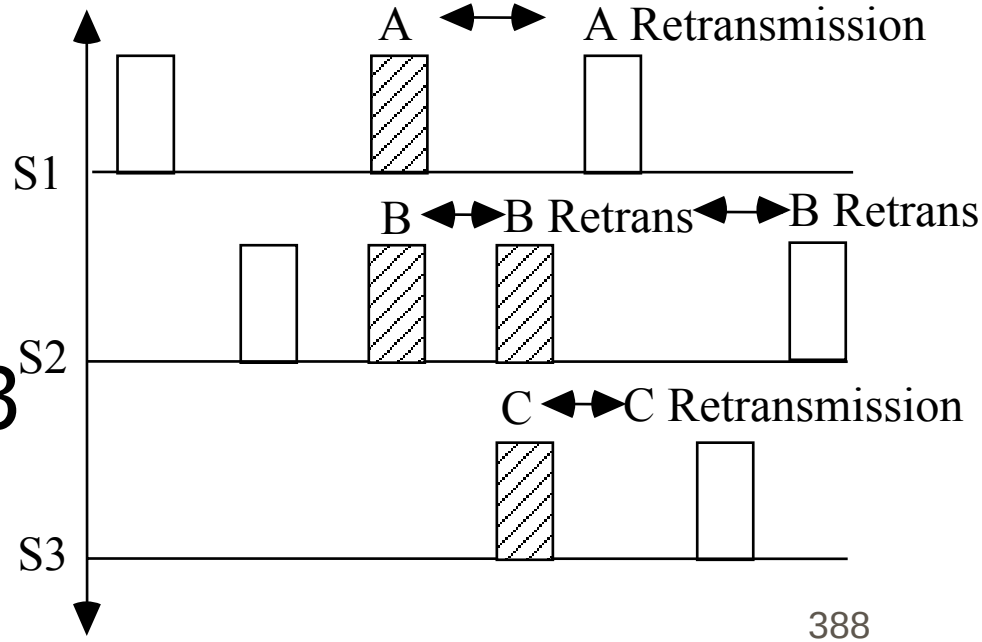
- On doit définir la **politique de partage de la boucle:**
  - **Plateau tournant** (‘ Slotted ring ’ J.R. Pierce)
  - **Jeton circulant** (‘ Token ring ’ E.E. Newhall).
- Un anneau est **une structure active**, **(régénération de signal/retard dans les stations)**.
  - **Problèmes de fiabilité** dus aux interfaces
  - Nécessité de **prévoir le retrait** ("shunt") de stations sur panne.
- **Exemples** : Boucle à jeton ‘ Token Ring ’ IBM 802.5 , Boucle ‘ FDDI ’ ANSI X3T9

Critères qualitatifs : partage en coopération

- **Une approche classique**.
  - Les stations **coopèrent et** par un dialogue préalable définissent qui peut accéder à la voie.
- Implique peu ou prou une **connaissance globale**.
- Exemples **de protocoles en coopération**:
  - **Passation de jeton** : Bus à jeton (802.4), Boucle à jeton (802.5), FDDI (X3T9).
  - **Scrutation** (‘ polling ’) : 100 Base VG Anylan, bus de terrain.
  - **Réservation statique** d’intervalles temporels (TDMA: ‘Time division Multiplexing Access’).

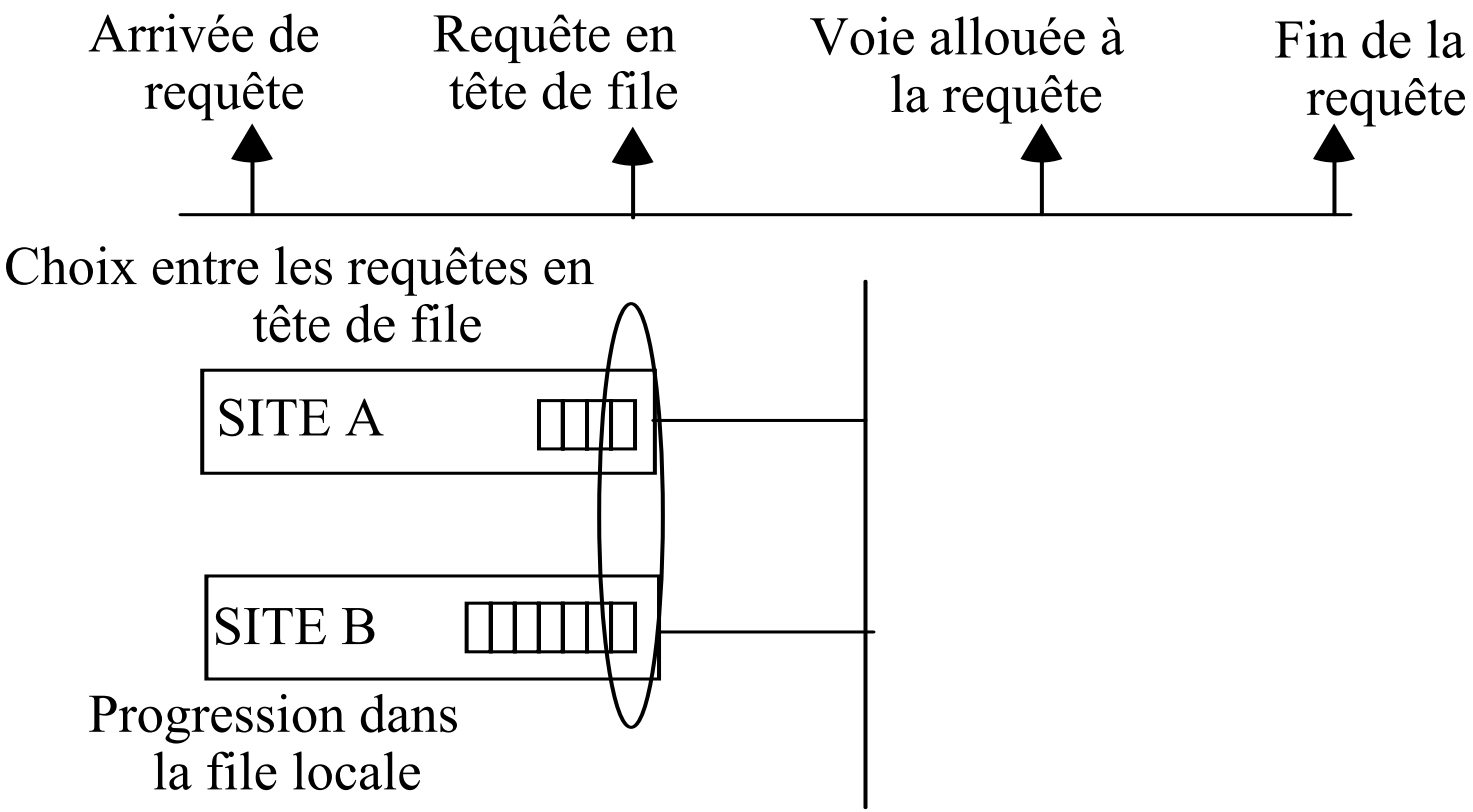
Critères qualitatifs : partage en compétition (‘ contention ’)

- Une approche **probabiliste.**
  - Les stations s'emparent de la voie **sans certitude sur son inoccupation.**
  - Il y a nécessairement des **collisions** d'accès à la voie.
- Connaissances **locales**
- Ex: **Ethernet** 802.3 **WIFI** 802.11.



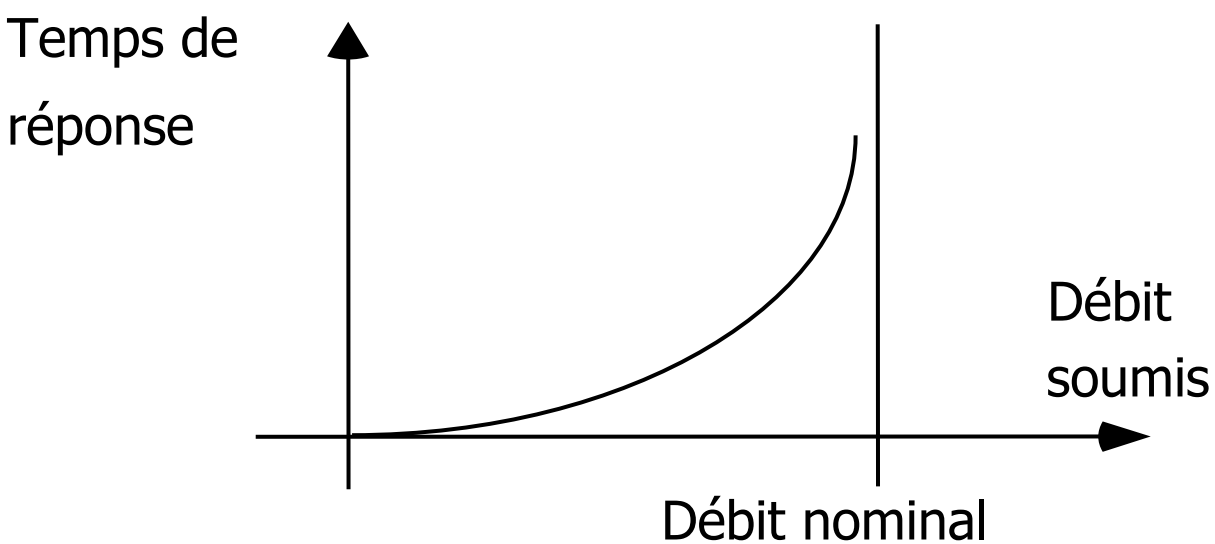
Critères de performances

- **Réseau local => Ajout d'une attente de plus :**
  le temps d'accès à la voie commune (au médium).



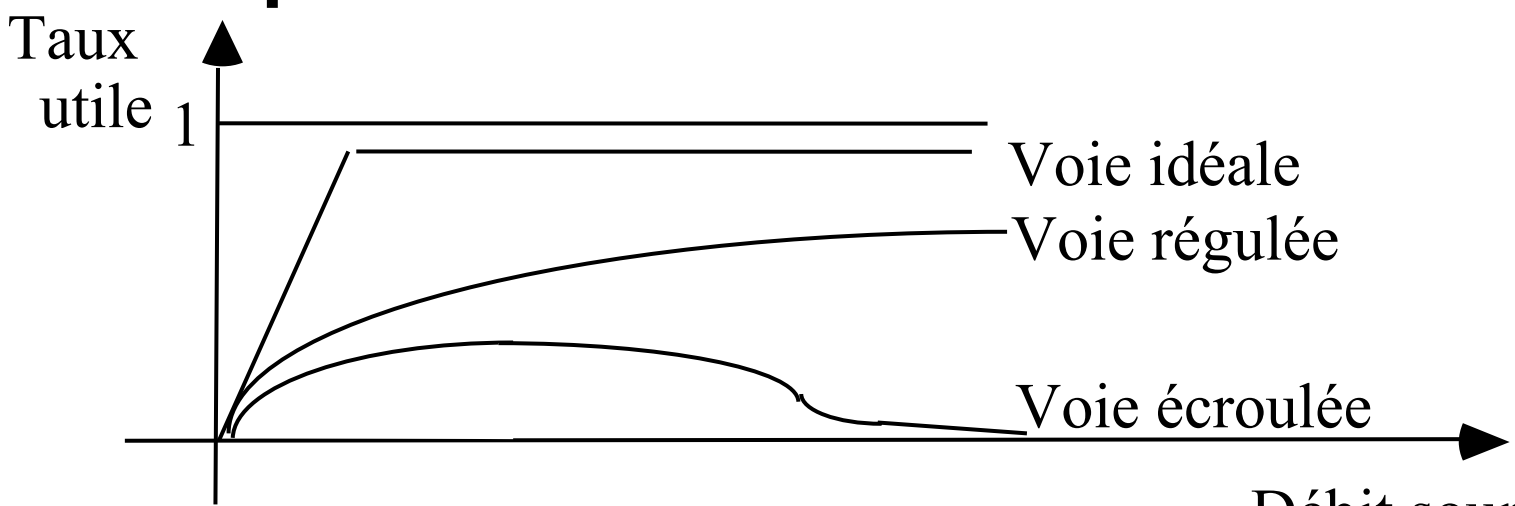
Critères de performances : Point de vue de l'utilisateur

- **Qualité de service temporelle (QOS)**
  - **Temps de réponse** / latence (moyenne), **gigue** (second moment)
  - **Équité des services** ou garantie d'un niveau de service.
- Si l'on soumet une voie à un trafic de plus en plus élevé : phénomène de **congestion**.
- Exemple : si le débit soumis croit vers le débit maximum le temps de réponse tend vers l'infini (**voie saturée)**.



Critères de performances : Point de vue global

- **Maximisation du débit global.**
- Prévention de **l'écroulement** (' thrashing ').
- Si le trafic écoulé continue à croître avec la charge la voie est dite **adaptative** à la charge ou **régulée**. Si le trafic diminue avec la charge et tend vers 0 la voie est **non adaptative** ou **écroulée.**



Critères de sûreté de fonctionnement

- Le réseau local doit être **sûr de fonctionnement** ("dependable") (**Panne** du réseau **: arrêt de** nombreuses fonctions de l'entreprise).
  - Critères quantitatifs classiques, **Fiabilité, Disponibilité, ...**
- **Evitement des pannes** : un dispositif (un ensemble de fonctions) "peu fiable" ne doit pas être **indispensable** au fonctionnement.
- **Partage d'accès 'centralisé' (dissymétrique)**
  - Un dispositif joue un rôle primordial dans le partage (exemple un arbitre)
- **Partage d'accès 'décentralisé' (symétrique)**
  - Aucun site n'est essentiel au partage de la voie.
- **Tolérance aux pannes** : introduction de redondances.

Conclusion: propriétés principales

- **Critères qualitatifs**
  - Topologie d'interconnexion.
  - Coopération / Compétition.
- **Critères de performance**
  - QOS : Temps de réponse.
  - Débit global, Protocole Adaptatif/Non adaptatif.
- **Sûreté de fonctionnement**
  - Centralisé / Décentralisé.

Réseaux locaux partagés

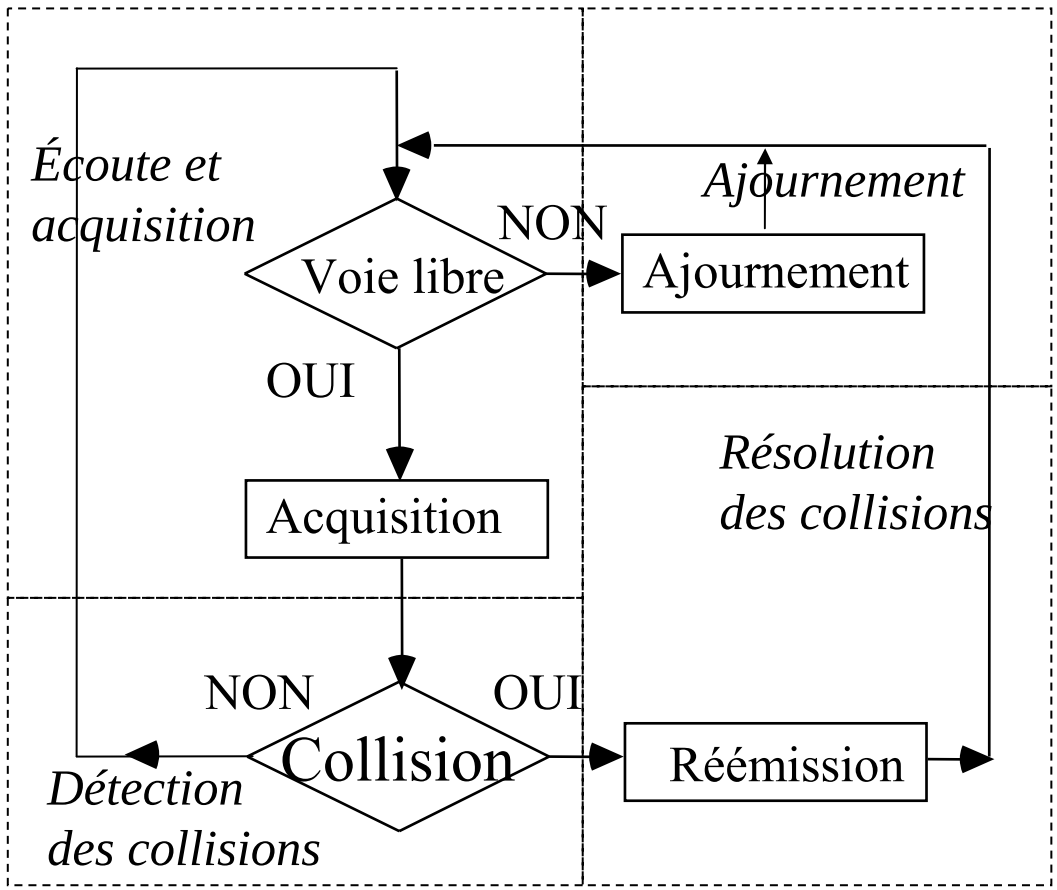
Protocoles de partage d'une voie commune en compétition

Introduction: protocoles en compétition (‘contention protocols’)

- **Étude des protocoles avec accès en compétition** :
  - Émission sans être certain d’être le seul à émettre.
  - Nécessité de prévoir des retransmissions.
- **Type de médium partagé :** Bus ou voie radio.
- **Solution :** Réseau filaire **Ethernet** 802.3
- **Solution :** Réseau radio ‘ wireless ‘ **WIFI** 802.11
- Une **proportion très importante** des protocoles de partage de voie commune.

Introduction: caractéristiques d'un protocole en compétition

- **Écoute et Acquisition :** actions entreprises pour **s'emparer de la voie** commune.
- **Ajournement :** actions entreprises si l'on **constate que la voie est occupée.**
- **Détection des collisions :** moyens par lesquels **un conflit d'accès à la voie est détecté.**
- **Résolution des collisions :** stratégie adoptée pour **retransmettre** une trame en collision.

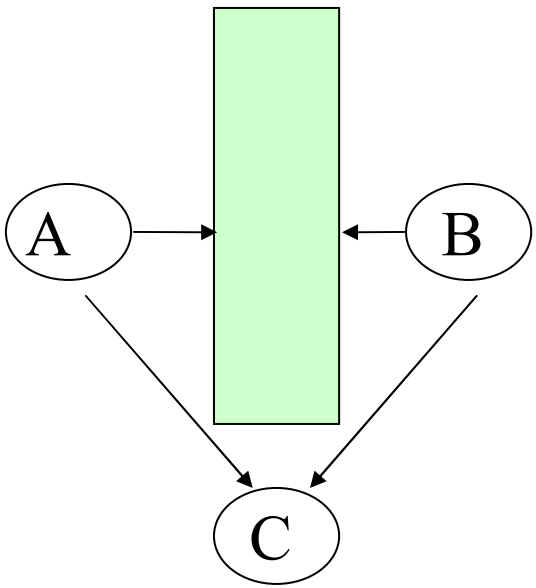


A) Écoute préalable et acquisition

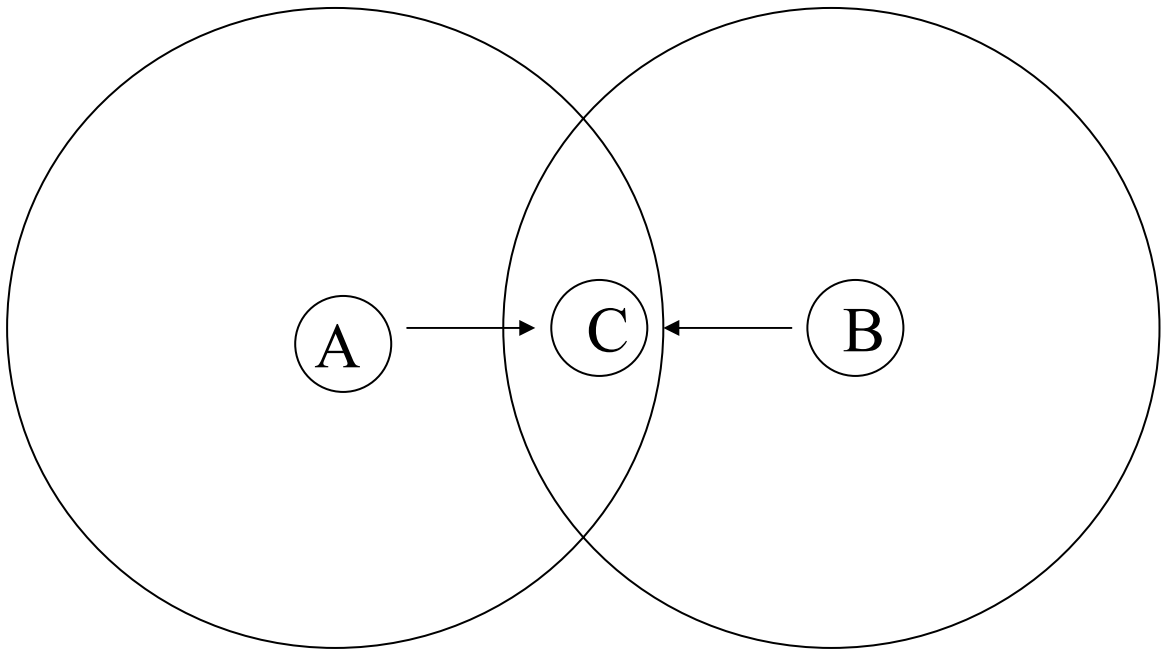
- **Émission sans écoute préalable**: (Émission sourde N. Abramson) "Aloha Pur" .
  - Un émetteur passe immédiatement en toutes circonstances en **mode acquisition**.
  - Les stations **n'écoutent pas** la voie.
- **Émission avec écoute préalable**:(CSMA "Carrier Sense Multiple Access", L.Kleinrock) Ethernet, WIFI
  - Si la voie est détectée libre, l'émetteur passe en **mode acquisition**.
  - Si la voie est détectée occupée, l'émetteur passe en **mode ajournement**.

Écoute préalable en réseau radio: le problème des stations cachées

- Réseaux radios **: écoute préalable possible** mais des difficultés.

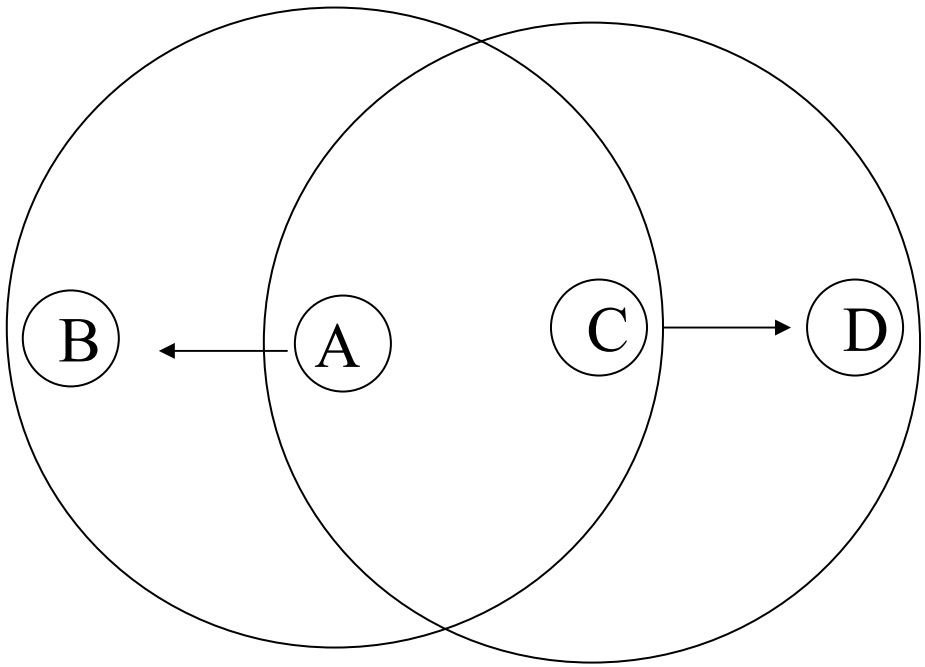


Station cachée: Obstacle



Station cachée: Affaiblissement

Écoute préalable en réseau radio: le problème des stations exposées

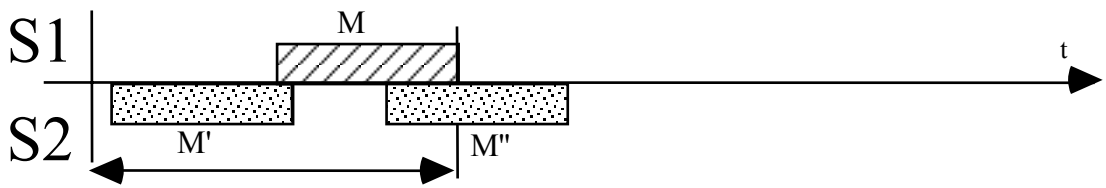


Stations exposées A et C

- A **émet** vers B.
- C qui fait de l'écoute pour **émettre vers D** constate la transmission de A et **attend sa fin**.
- D est hors de portée de A donc **l'attente de C est inutile.**

Intervalle de vulnérabilité: cas d'une acquisition sans écoute

- Intervalle de temps pendant lequel **deux stations ne peuvent commencer d'émettre** sans écoute et provoquer une collision
- Il suffit que **le dernier bit d'une trame se superpose avec le premier bit** d'une autre trame pour qu'il y ait **collision**.



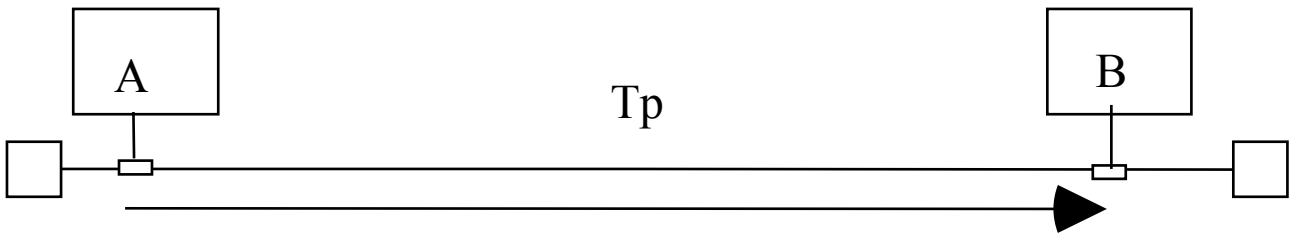
Intervalle de vulnérabilité 2T

Hypothèse: messages de durée T fixe

- Pour qu'une trame de durée T soit transmise sans collision il faut qu'aucune autre trame de durée T ne soit transmise pendant un intervalle 2T =>**performance très médiocre.**

Intervalle de vulnérabilité: cas d'une acquisition avec écoute

- **Intervalle de temps** pendant lequel deux stations peuvent émettre et provoquer une **collision** malgré l'écoute.



- A et B situées aux **extrémités** du médium, A **écoute** le canal, ne détecte rien, décide d'émettre à t_0 .
- **Soit Tp le temps de propagation** entre A et B (fonction de la vitesse de la lumière , des retards introduits sur le câble par les éléments matériels : transmetteurs, répéteurs, ...).
- **B peut commencer** à émettre entre **t_0** et **t_0 + Tp** (pour lui la voie est libre) => on a une **collision.**
- **L'intervalle de vulnérabilité est Tp.**

B) Ajournement (‘deference’)

Ajournement Persistant (Ethernet)

- **Émission immédiate** si la voie est libre ou dès que la trame courante est finie.
- Hypothèse de la solution: **Le trafic sur la voie est faible**
  - La probabilité pour que deux nouvelles demandes apparaissent pendant la transmission d'une trame est faible.

Ajournement non Persistant (WIFI)

- Emission immédiate si la voie est libre. Si la voie est occupée **différer la transmission** comme s’il y avait **collision**.
- Hypothèse de la solution: **Le trafic sur la voie est élevé**
  - Si une trame a risqué d'interférer avec une autre c'est qu'il y a de la charge qui nécessite déjà d’appliquer un retard adaptatif.

C) Détection des collisions : Par écoute de la voie

- **Quand l'écoute des collisions est possible**
- La stratégie d'écoute dépend du médium **utilisé**
  - Si on a deux **signaux séparés** (transmit/receive) exploitation **en parallèle du** signal émis et du signal reçu (exemple 10 Base T).
  - Mesure de la **puissance** moyenne du signal.

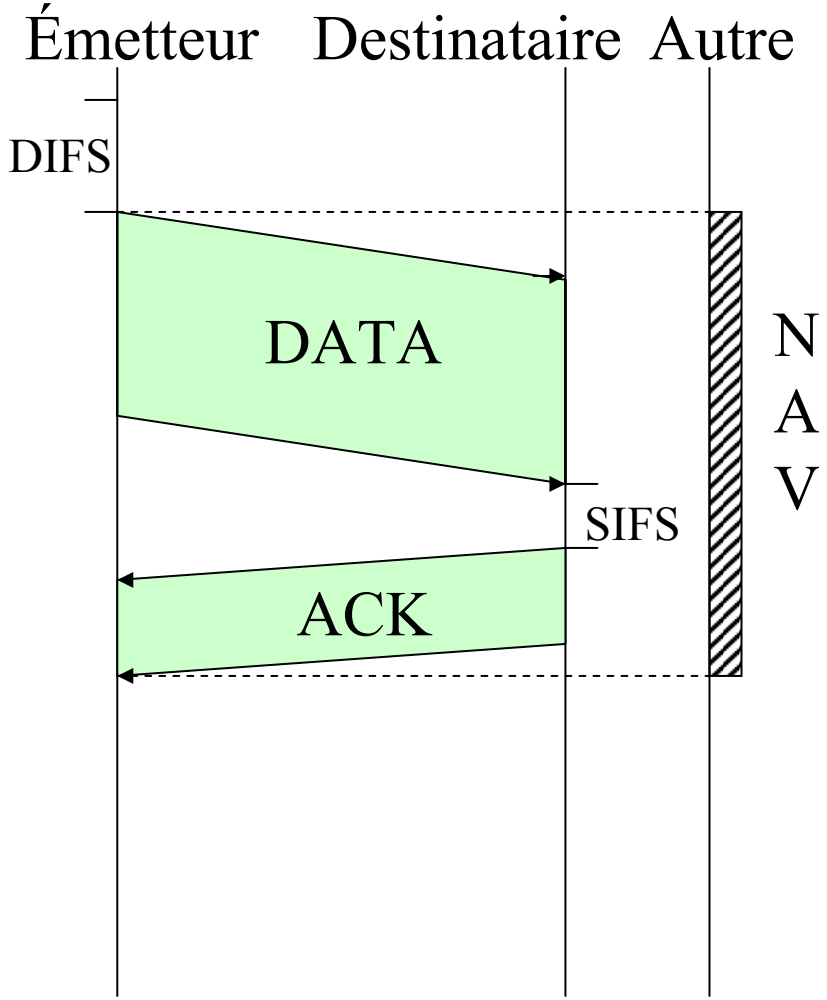
  La puissance moyenne sur la voie en cas de collision est **anormale** (plusieurs signaux sont superposés exemple 10 Base 2, 10 Base 5).
- **L'écoute suppose** l'existence d'une **durée minimum** de la collision permettant la détection.
- Solution type **: le CSMA/CD** ' Carrier Sense Multiple Access / Collision Detection ' **Ethernet 802.3**

Détection des collisions : Cas d'une Écoute impossible

- **Réseaux radios : l'écoute des collisions est non prévue** (très coûteuse ou impossible).
- A la place utilisation d'un **protocole de liaison classique** :
  - code détecteur d'erreurs, accusé de réception positif si la trame est correcte
  - délai de garde, retransmission si la trame est incorrecte
- Les collisions sont **traitées comme des erreurs de transmission** sur les trames**.**
- **Solution du réseau WIFI** (un protocole de base et un protocole plus sophistiqué, le **CSMA/CA** ` Carrier Sense with Multiple Access/ Collision Avoidance ').

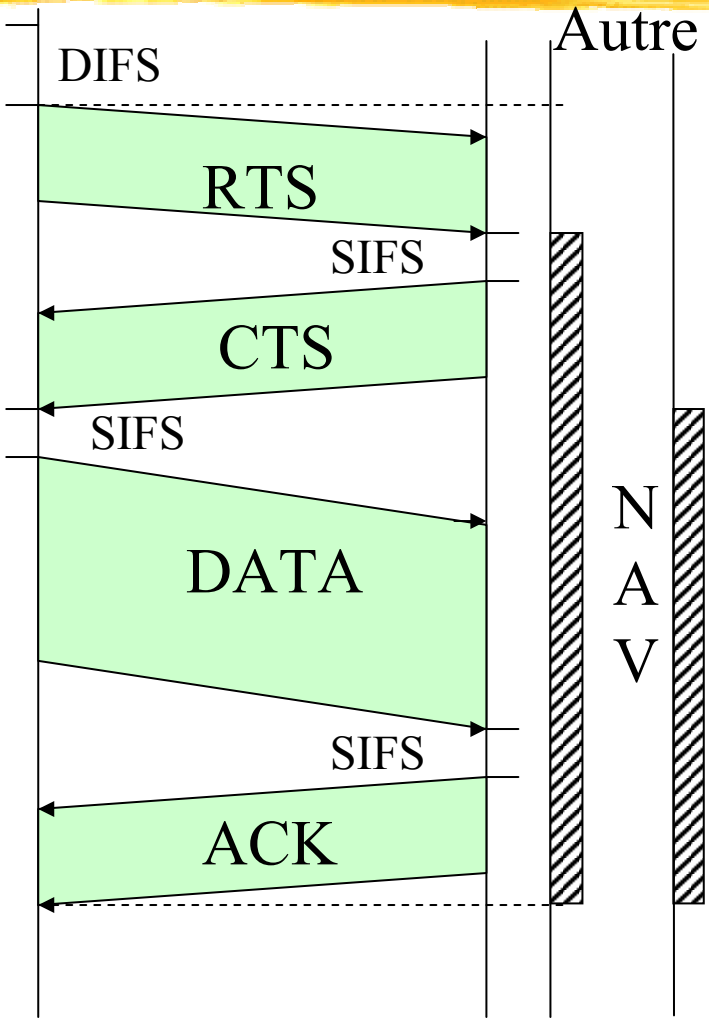
802.11-WIFI : Le mode de base de détection des collisions

- L'émetteur **détecte la voie libre** pendant un délai DIFS.
- Il émet une trame **Data**.
- Une collision **peut avoir lieu** sur la trame data. L'émetteur attend une trame de réponse de type Ack.
- Ack doit être émis **après une attente courte baptisée SIFS**.
- Si l'Ack n'est pas transmis c'est qu'il y a eu **problème**. Le réseau est à nouveau partageable après DIFS.
- Pendant ce temps **les autres sont en attente** (indicateur NAV ' Network Allocation Vector ').

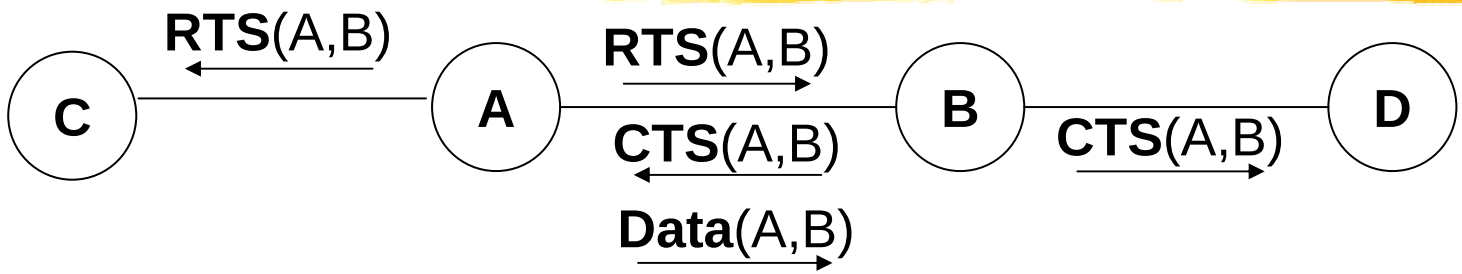


802.11-WIFI : le CSMA/CA (‘ Collision Avoidance ’) avec l’échange RTS/CTS

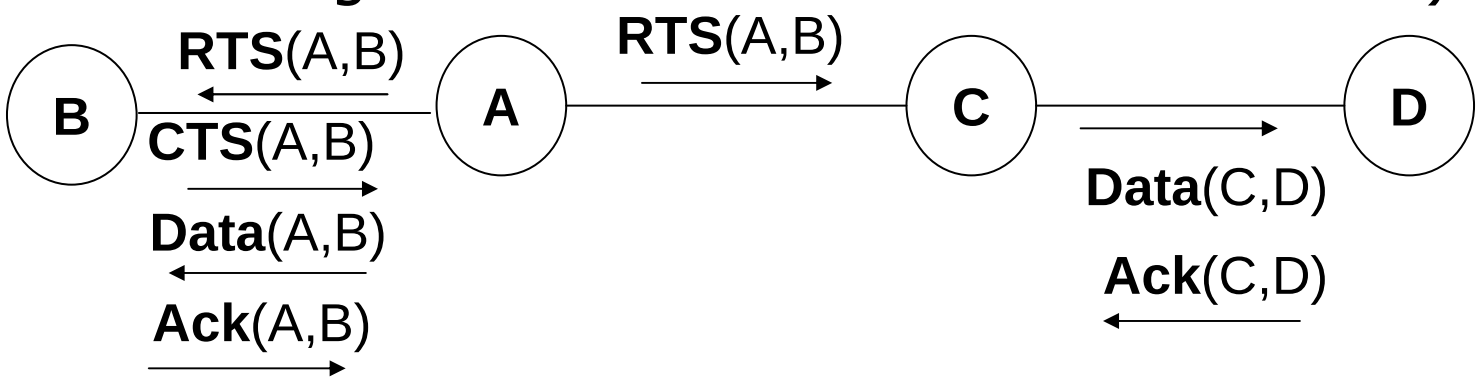
- **Après un silence DIFS** l’émetteur émet un message court RTS (Request to send) signalant **qu’il veut émettre**.
- Le destinataire transmet une réponse courte **d’acceptation** CTS (Clear To Send).
- RTS-CTS réussi: la trame est **transmise.**
- La collision **peut avoir lieu** que sur le message court RTS => limitation de la durée d’une collision.
- CTS correct indique **qu’il n’y a pas eu de collision** sur RTS (détection des collisions).
- La trame data suivie de son acquittement positif peut être échangée **sans collision**.



Le CSMA/CA et le problème des stations cachées et exposées



- La station cachée D qui perçoit le CTS reste silencieuse (la durée du message DATA circule dans RTS et dans CTS)



- La station exposée C entend RTS(A,B) mais pas le CTS(A,B): C déduit qu'une transmission de C vers D ne peut interférer en B ni géner A tant que A transmet DATA (A,B).

D) Résolution des collisions : retransmission non adaptative

La prochaine tentative après une collision est effectuée selon **une distribution qui ne dépend pas du débit soumis au réseau** (non adaptative à la charge).

- Exemple: Tirage aléatoire d'une durée d'attente selon **une distribution statique** ou même dépendante du site.

- A forte charge de toutes façons les stations provoquent de plus en plus de collisions et la voie est non régulée => **écroulement.**

Résolution des conflits : Réémission adaptative

La prochaine tentative après une collision est effectuée après une attente proportionnelle à la charge.

- **Solution centralisée :** Un site d'administration **mesure en permanence le trafic** par observation de la voie.
- **Il diffuse périodiquement** ses mesures aux stations qui les utilisent pour définir un délai de **retransmission adaptatif** en fonction de la charge.

- **Solution répartie :** Chaque site se base sur des **connaissances purement locales** pour déterminer sa politique de **retransmission**.
- **Excellente solution**: prendre comme indicateur de charge **le nombre de collisions** qu'un message vient de rencontrer.

- **Algorithme du retard binaire exponentiel** (BEB ' Binary Exponential Backoff ') : **deux versions très voisines** de cette solution sont utilisées en Ethernet et WIFI.

Résolution des conflits: Algorithme du retard binaire Ethernet

```
Retard_Binaire (nb_collision: entier)
ST : flottant :=51.2 ; fact_mult, delai : flottant ;
début
        si ( nb_collision < 10 ) alors
                fact_mult := 2**nb_collision ;
        sinon
                fact_mult := 2**10;
        finsi;
        délai := ST * int (random*fact_mult);
        attendre (délai);
fin;
```

Commentaires : algorithme du retard binaire Ethernet

- On attend **un délai distribué aléatoirement,**

(**random** est un générateur de nombre aléatoire [0,1[).

- **Uniformément distribué** sur un intervalle,
- **Qui double** à chaque collision,
- Pendant **les 10 premières tentatives.**
- On évalue l'attente en nombre entiers de "slot time" ST.

(**int** est une fonction qui rend la valeur entière par défaut).

- **On montre que cette solution est non écroulée pour moins de 1024 stations.**
- On fait **au maximum 16 tentatives** (caractéristique non intégrée au retard binaire).

Réseaux locaux partagés

Réseaux locaux Ethernet

Historique
Niveau liaison
Ethernet 10 Mb/s
Ethernet 100 Mb/s
Ethernet Gigabit
Ethernet 10 Gigabits
Auto négociation

Historique Ethernet

- 0rigine **R.M Metcalfe** (Rank Xerox Palo Alto). Début des travaux 1973. Article CACM 1976 (Metcalfe et Boggs).
- Protocole en compétition sur coaxial à **2,94 Mb/s** (1976).
- Brevet **Ethernet** (1977): début de l'industrialisation.
- Norme DIX ("Digital Intel Xerox") 10 Base 5 (1980)
- Normalisation **IEEE 802.3** (1983)
- 10 Base 2 (1986), 10 Base T (1991), 10 Base F (1994)
- Ethernet 100 Mb/s : 802.3u (1995)
- Ethernet gigabit/s : 802.3z/802.3ab (1998)
- Ethernet 10 Gigabits/s : 802.3ae (1999-2006)

Ethernet niveau MAC : principales caractéristiques de la version de base

- protocole d'accès au médium en **compétition.**
- **écoute** de porteuse (CSMA).
- **ajournement** persistant (1-persistant).
- **détection** de collisions par écoute (CD).
- **retransmission** avec retard binaire.
- **destruction silencieuse** des messages bruités**.**
- **sans connexion.**
- pas de **fragmentation**, de **reprise** sur erreur, de **contrôle de flux**.
- délivrance **'au mieux'** ('Best Effort').

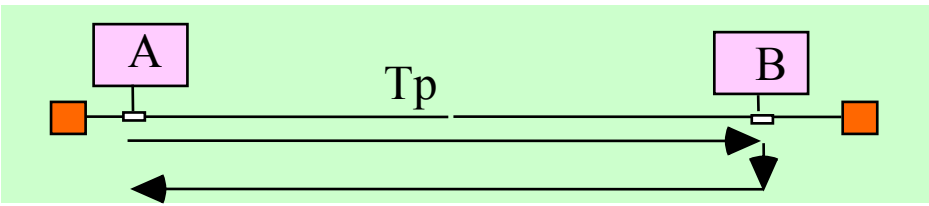
Ethernet : Notion de 'tranche canal' ('ST Slot Time')

- **Principe CSMA/CD** : toute collision doit être **détectée** par le niveau MAC et celui-ci doit assurer la **retransmission.**
- ST est un **délai fixé par la norme Ethernet**.
- **a)** Dans le délai **ST toute station détecte à coup sur toute collision.**
- **Conséquences :**
  - ST fixe **la taille minimum d'une trame.**
  - Après qu'une station a pu transmettre pendant une durée au moins égale à ST**, elle a acquis la voie : elle ne doit plus rencontrer de collision.**
- **b)** Dans le délai **ST** toute **collision** est terminée (on ne poursuit pas une collision sur toute la durée d'un message éventuellement long)**.**

Ethernet : Fixation de ST > Délai d'aller retour

- **Délai d'aller retour ('Round Trip Propagation Delay')**

A et B situées aux extrémités du réseau. Une collision sur une trame de A vers B n'est perçue en A qu'à t_0 + **2 Tp**



Tp le temps maximum de propagation du signal

- **Ethernet 10 Mb/s** : Délai d'aller retour 2 Tp = 46,4 µs.
- **ST > 2 Tp** fixé à 51,2 µs = 46,4 + 4,8 µs (512 temps bit)

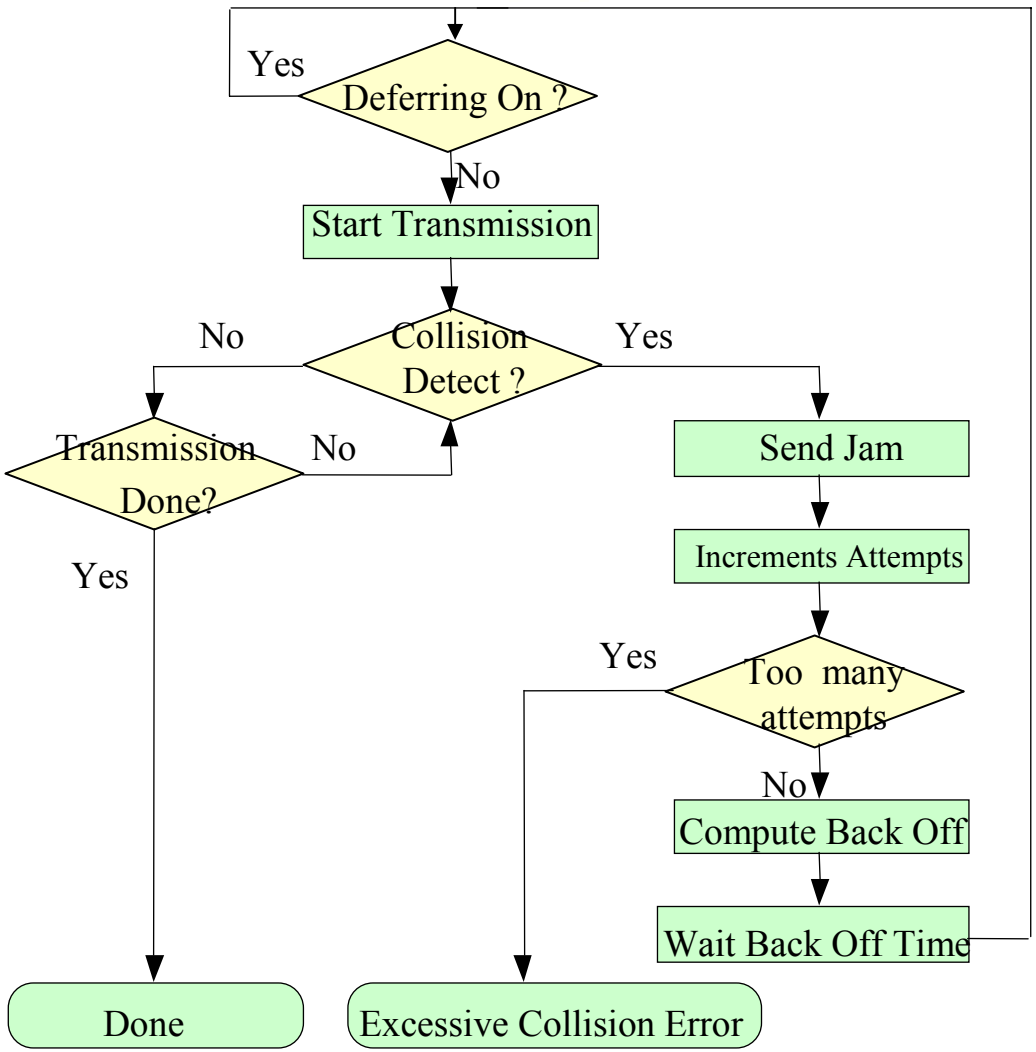
=> La taille minimum d'une trame Ethernet est de 64 octets (valeur fixée sans compter le préambule).

Ethernet : Le renforcement de collision (‘ brouillage ’ ‘ jam ’)

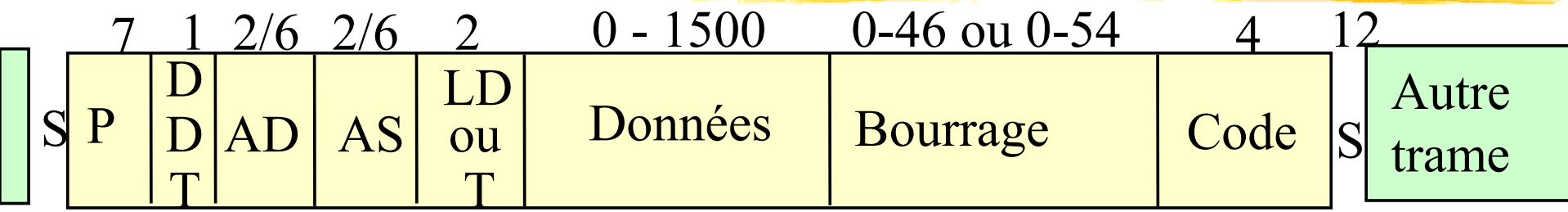
- **Brouillage ("Jam") :** Après détection de collision l'émetteur transmet sur le médium une information non significative.
  - La durée du brouillage est le plus souvent de 3,2 µs .
- Idée de **durée minimum d’une collision** (renforcement de collision): toute trame en collision à une durée minimum pour être détectable par tous (au moins 96 temps bits).
- Idée de **limitation de la durée d’une collision** à la durée maximum nécessaire à la détection de la collision plus la durée du brouillage. Le brouillage fixe la définition de ST donc la durée maximum d’une collision.

ST : Délai d'aller retour (46,4 µs)+Brouillage (4,8 µs)

Ethernet : comportement général

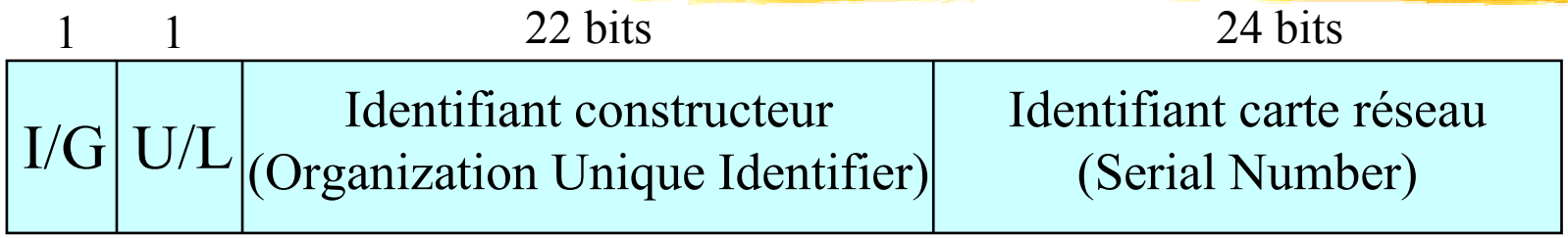


Ethernet : structure de la trame



- P : Préambule 'Preamble' : 7 octets 101010… synchro bit.
- DDT : Délimiteur début de trame 'Start Of Frame': 10101011 synchro octet.
- AD : Adresse Destination 'Destination address' (6 octets) possibilité 2 octets.
- AS : Adresse Source 'Source Address' (6 octets) (ou 2 octets).
- LD : Longueur des données 'Lenght' (802.3) ou T: Type de la trame (DIX).
- Données + Bourrage: La charge utile de 1500 octets au plus (Entête + Données + Bourrage + Code : longueur min 64 octets).
- Code: Code polynomial détecteur d'erreur (FCS 'Frame Check Sequence').
- S : Silence inter-trame 'IFG Inter Frame Gap' (9,6 micro seconde soit 96 temps bit).

Ethernet : adressage IEEE 802



- **Notation :** 6 groupes de 2 chiffres hexadécimaux 00-DD-01-30-C3-17
  Format canonique grand boutiste sur les octets, petit boutiste sur les bits.
- **I/G :** Adresse Individuelle/Groupe (**Individuelle** = 0 ; **Groupe** = 1)

"**Broadcast**" (tous les bits à 1) adresse diffusion générale FF-FF-FF-FF-FF-FF

"**Multicast**" G=1 + adresse de diffusion sur groupe : 01-00-5E-00-A8-76

- **U/L :** Adresse Universelle (unique) / Locale (non unique) (U=0 ; L= 1)
- **Identifiant constructeur** (' OUI Organisation Unique Identifier') (RFC 1340), **carte** (' SN Serial Number ').
- **Exemple** : 00-AA-00-08-C3-98 , Les trois octets de gauche 00-AA-00 désignent le constructeur INTEL, les trois octets de droite 08-C3-98 sont l'adresse unique d'une carte réseau.

Ethernet partagé : performance

- **Temps de réponse pour un utilisateur :**
  - Débit soumis < 50% : bon comportement , temps d'attente moyen < 1 ms.
  - 50% < Débit soumis < 80% : délai supportable < 10 milliseconde.
  - Débit soumis > 80% : mauvais comportement (exemple 100 stations, débit soumis 90 % à 10 Mb/s temps d'attente moyen 10 seconde).
- **Débit maximum :**
  - **Une idée fausse :** on ne récupère pas plus de 37% de la bande passante.
  - **Mesure** sur un Ethernet réel de 10 Mb/s avec 24 stations émettant en permanence des messages de 64 octets: **taux de trafic utile 90%.**
  - Trois paramètres influencent le débit maximum récupérable :
    - **Le nombre de stations connectées:** 1 à 1024, en fait moins de 200 souhaitable.
    - **Le débit soumis par station**: normalement le trafic est très sporadique avec un débit soumis en moyenne faible (un débit constant élevé n'est pas prévu).
    - **Taille des trames** : effet des collisions plus important pour des trames petites, remplissage en données utilisateurs moindre pour des trames petites (64 = 46+18 14888 trames/s 5,476Mb/s , 1518= 1500+18 812 trames/s 9,744 Mb/s).

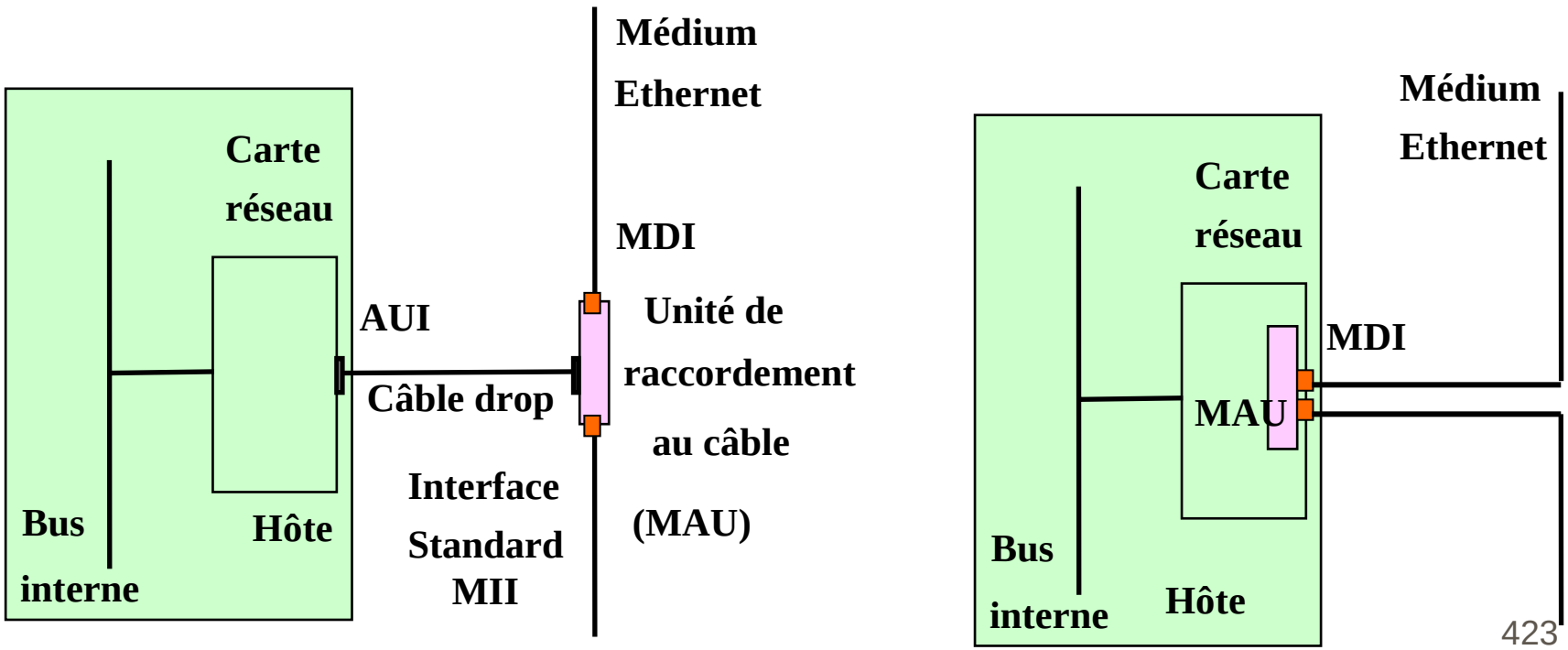
Réseaux locaux Ethernet

Ethernet:

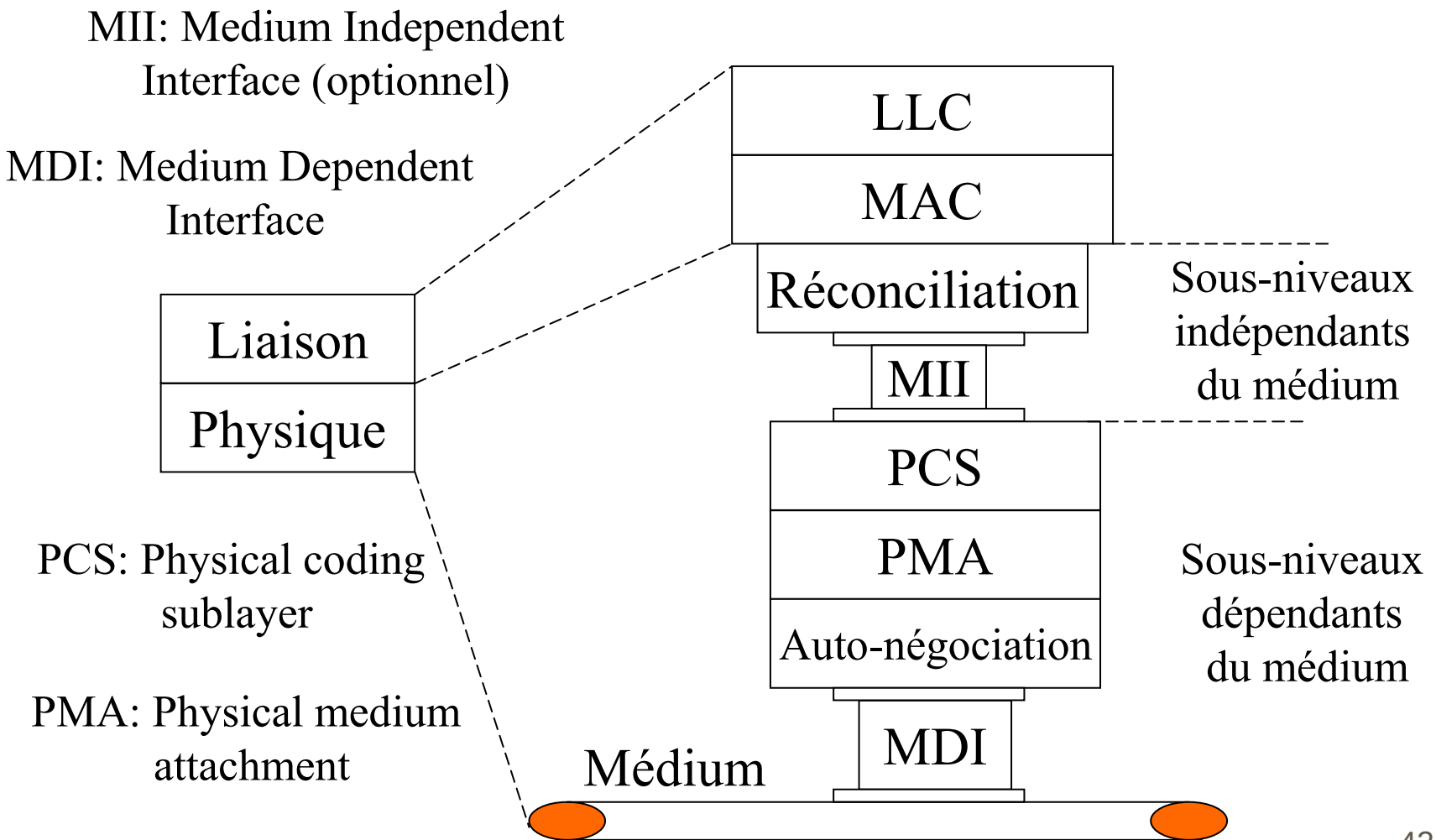
les standards à 10 Mb/s au niveau physique

Ethernet 10 Mb/s : configuration au niveau physique

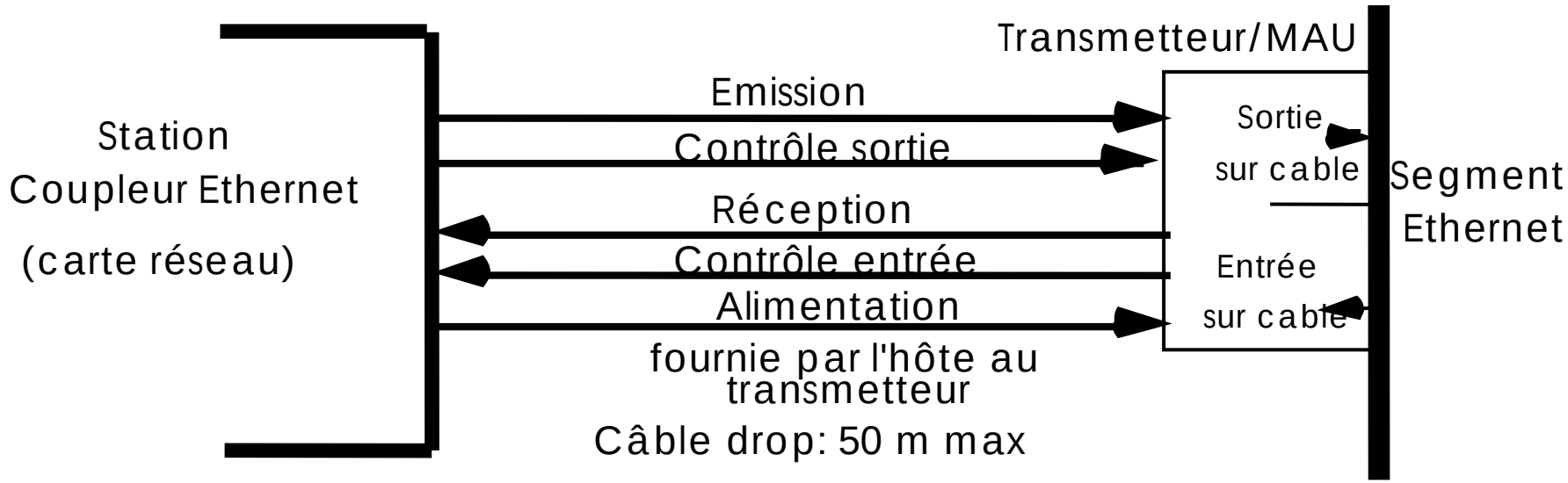
- **Interface indépendante** (MII "Medium Independent Interface"): à 10 Mb/s AUI : "Attachment Unit Interface" , Câble drop/"Transceiver Cable"
- **Raccordement au médium** (PMA "Physical Medium Attachment"): Transmetteur "Transceiver" ou MAU "Medium Attachment Unit"
- **Prise sur le câble** (MDI "Medium Dependent Interface")



Modèle générique de découpage en couches du niveau physique ETHERNET



L'interface indépendante à 10 Mb/s ("AUI : "Attachment user interface")



Alimentation, **Données en émission**, **Données en réception**

Contrôle en entrée: *Transmetteur disponible* IDL : aucun signal.
Transmetteur indisponible CS1 : horloge demi fréquence bit Ethernet.
Erreur qualité du signal CS0 : horloge à la fréquence bit Ethernet.
(1) Signal impropre (coupure, MAU HS) (2) Collision

Contrôle en sortie (*Très peu Implanté).*

Les différents standards de niveau physique Ethernet à 10 Mb/s

Système de désignation physique Ethernet : A, L, B

- **A** : Définit la vitesse : 1, 10, 100, 1000, 10G.
- **L** : Deux valeurs BASE : bande de base. BROAD : bande large (broadband).
- **B** : Définit la longueur maximum d'un segment exprimée en centaines de mètres ou définition d'un type de médium.
- **Exemple : 10 BASE 2** 10 Mb/s en bande de base, longueur max d'un segment 200m (environ), 100 BASE T2 100 Mb/s en bande de base sur deux paires torsadées.

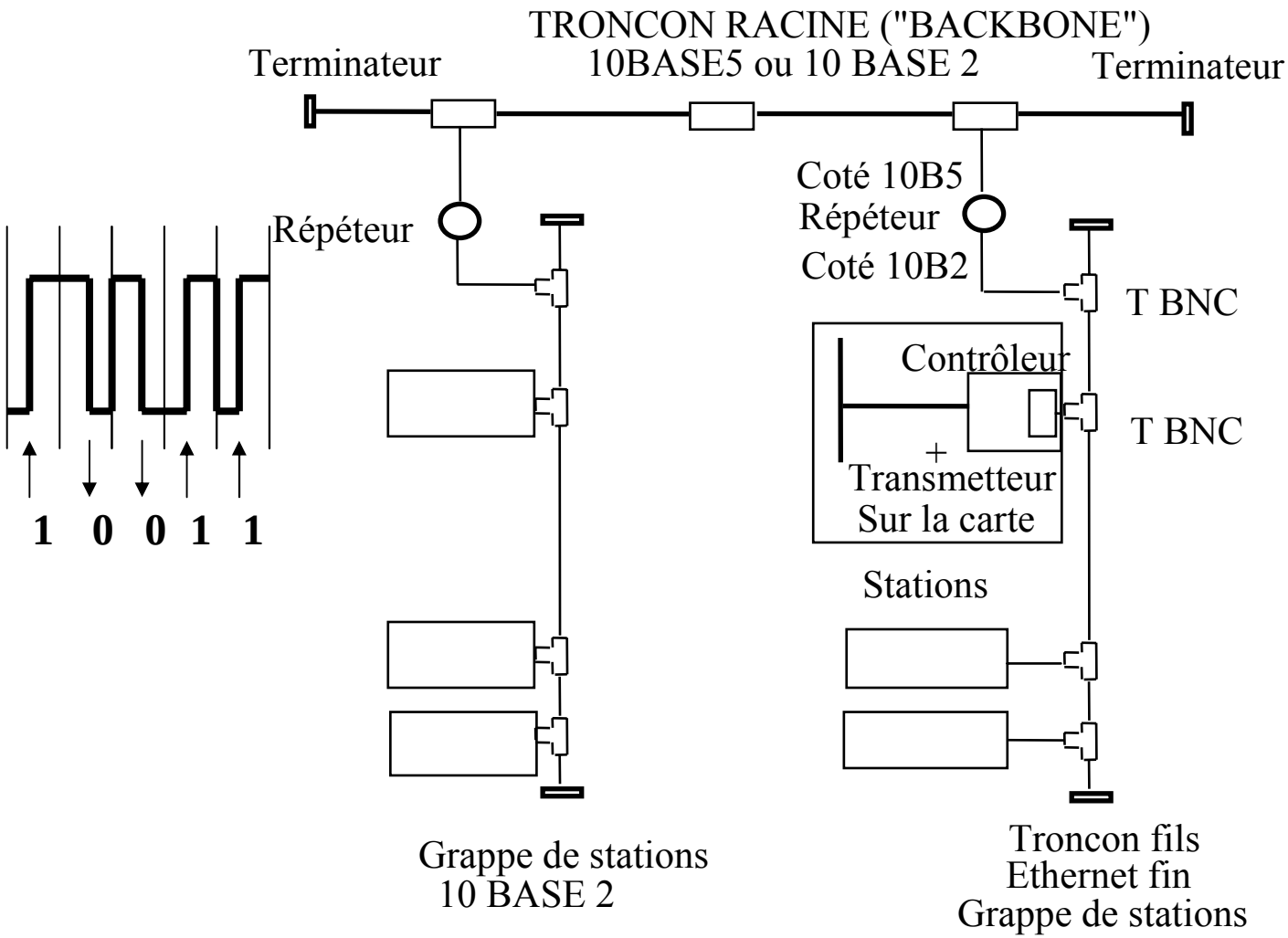
Principaux standards à 10 Megabits/s

Ethernet gros	**10 BASE 5**
Ethernet fin	**10 BASE 2**
Ethernet paire torsadée	**10 BASE T**
Ethernet fibre optique	**10 BASE FL**

10 BASE 2 Ethernet fin "Thin Ethernet" , "Thinnet"

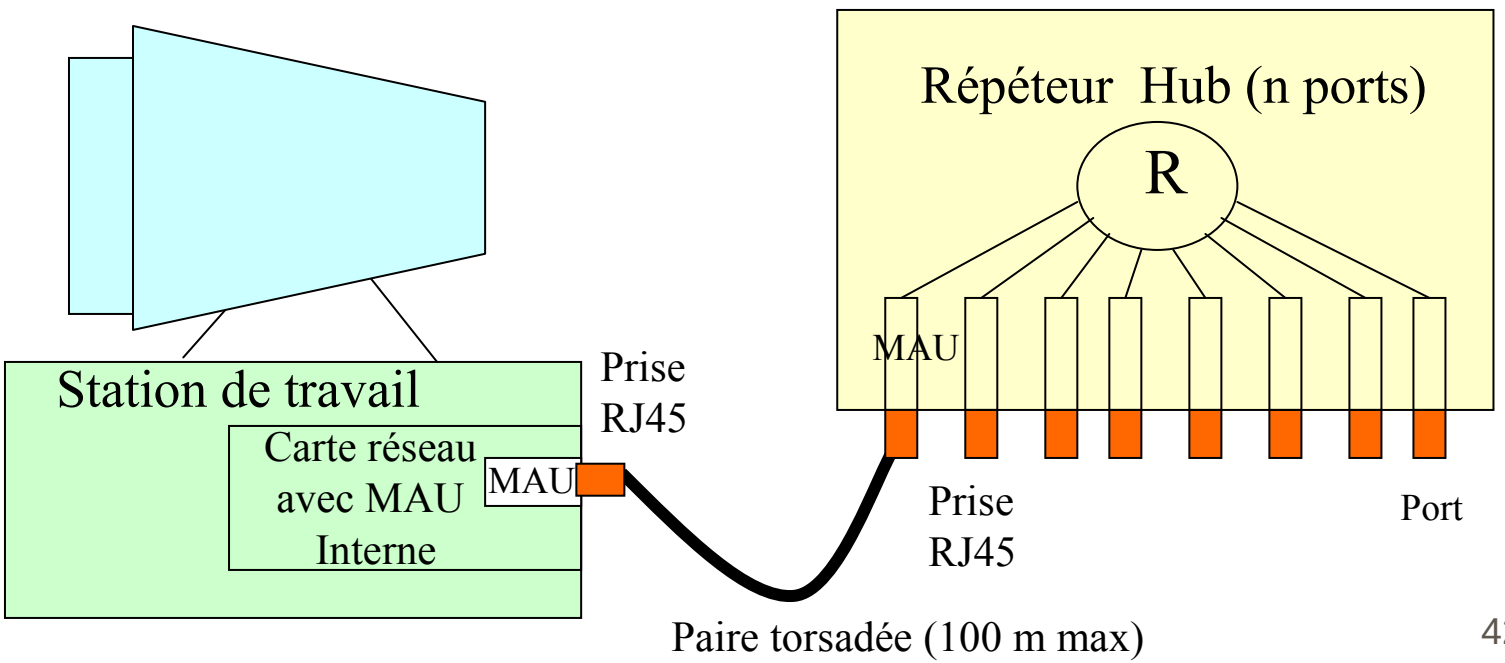
- Spécification très voisine du 10 BASE 5.
- Version plus économique pour réseaux de stations ou de micros.
  - **Caractéristiques du câble coaxial fin**
- Diamètre 0.2 pouce 5 mm.
- Impédance caractéristique 50 Ohms.
- Longueur maximum du tronçon 185 m.
- Espacement des transmetteurs 5 m.
- Maximum de 30 transmetteurs par tronçon.
- Raccordement au câble : prise BNC (Bayonet Neil-Concellman).
  - **Caractéristiques de la signalisation**
- **10** Mégabits/seconde**, bande de base,** code **"Manchester".**

10 BASE 2 ETHERNET FIN
Architecture du réseau



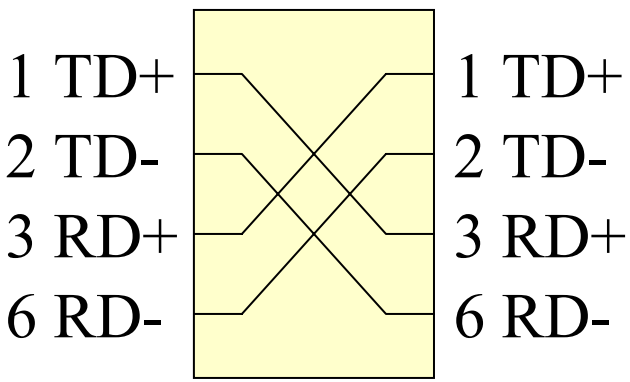
10 BASE T
Ethernet sur paires torsadées

- A partir des idées de **Starlan** réseau en compétition sur paires torsadées: **10 Base T** est devenu **le standard le plus répandu.**
- Utilisation de la **paire torsadée** : médium **économique**, redéfinition des prises standards (MDI RJ45) et des transmetteurs (MAU).
- Topologie en étoile avec des **répéteurs** autre terminologie (concentrateurs, 'hubs', **répéteurs** multiports, ' repeater hubs ').



10 BASE T : signalisation

- Ethernet 10 BASE-T utilise **deux paires** torsadées de catégorie 3 ou supérieure.
  - Paire émission **TR transmit data**
  - Paire réception / écoute des collisions **RD receive data**
- Connecteur normalisé **8 broches** RJ-45
- Broches 1 TD+, 2 TD- , 3 RD+, 4 inutilisé, 5 inutilisé, 6 RD- , 7 inutilisé , 8 inutilisé).



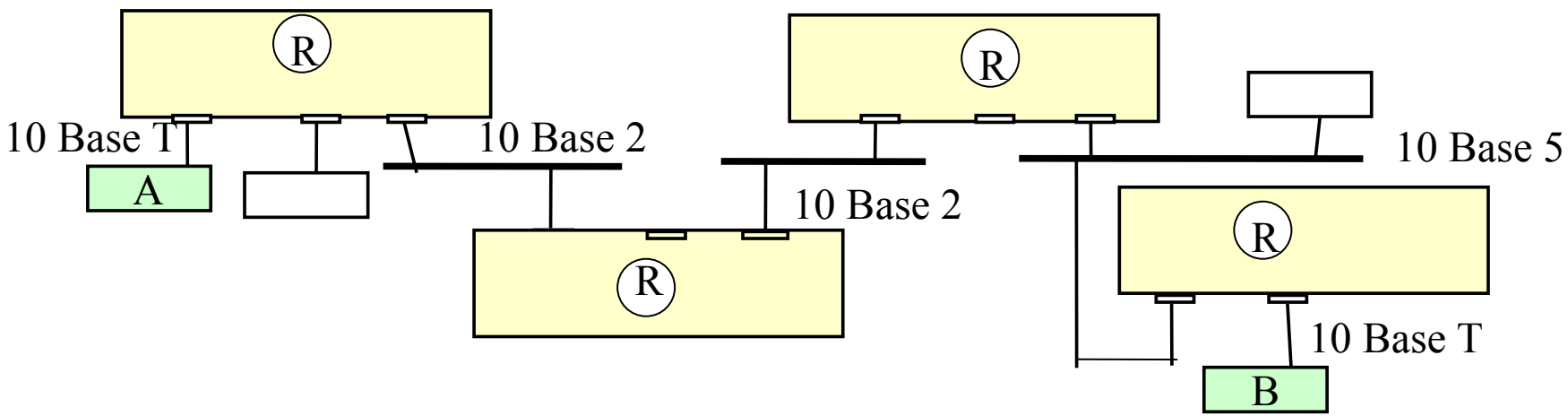
Construction d'un câble croisé

Fonctionnement d'un répéteur multiport

- **Propagation** : Un signal valide arrivant sur l'un des ports du répéteur est **régénéré et rediffusé** sur les autres ports.
  - Restauration de l'amplitude (' signal strenght ')
  - Re-synchronisation selon l'horloge du répéteur (' timing ')
  - Restauration de la forme des bits (' symmetry ')
- **Collisions** : Si **deux ports sont en collision**, le répéteur doit le détecter et générer **un signal de renforcement**
  - **Renforcement de collision** : en cas de détection de collision entre deux entrées A et B, le répéteur génère sur tout ses ports de sortie un signal de brouillage ' jam ' (une suite de 32 bits 01010). La durée du fragment, bits transmis et brouillage est de 96 bits au moins.
  - **Extension des fragments** : si une suite de bits de taille inférieure à 96 bits circule, un répéteur doit générer à la place au moins 96 bits.

10 BASE T Fonctionnement ` multi segments ' , ` multi répéteurs '

- On peut mêler les standards 10 BASE 2, 10 BASE 5, 10 BASE T en respectant **la contrainte de détection des collisions.**
- Empiriquement on peut cascader au plus **quatre répéteurs multiports** et au plus **cinq segments dont trois coaxiaux.**
- Sinon technique précise de **validation d'une architecture par évaluation du diamètre de collision**.



- **Exemple d'interconnexion** de réseaux 1OBASE 2, 5 et T avec 4 répéteurs

10 BASE F : Ethernet sur fibre optique

- Utilisation d'un **médium fibre optique en point a point** entre transmetteurs optiques.
- Une fibre en **émission**, une fibre en **réception**.
- Câblage en étoile avec des **répéteurs** (hubs).
- **Plusieurs variantes 10base FL (Link), 10 Base FB (Backbone), 10 Base FP (Passive).**
- **10 BASE FL – Fibre** 62,5/125 micromètres, émission LED, longueur d'onde 1300 nm.
- Longueur maximum d'un **segment 2km**.
- Nombre maximum de stations : **1024**
- **Solution fibre optique plus chère** mais la plus résistante aux perturbations électromagnétiques ou aux écoutes.

Conclusion : Ethernet 10

- **Un standard qu'on n'installe plus** mais une base installée **encore très importante.**
- Le débit de 10 Mégabits/s **reste suffisant** pour la plupart des applications.
- **Pas forcément nécessaire** de transformer très rapidement toutes les infrastructures sauf besoin précis (scientifique ou multimédia).

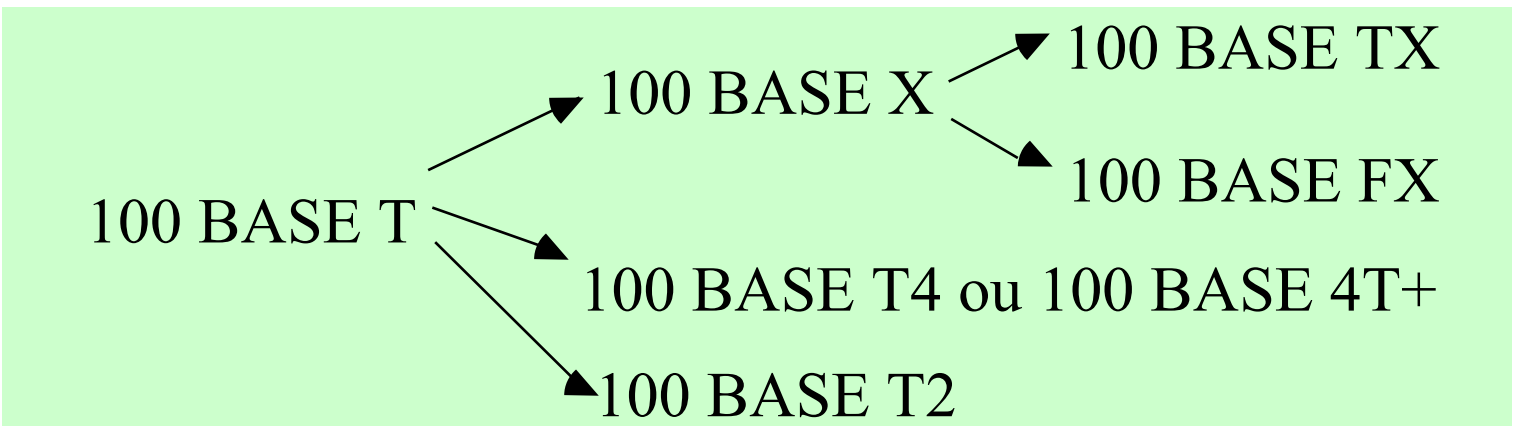
Réseaux locaux Ethernet

Ethernet:

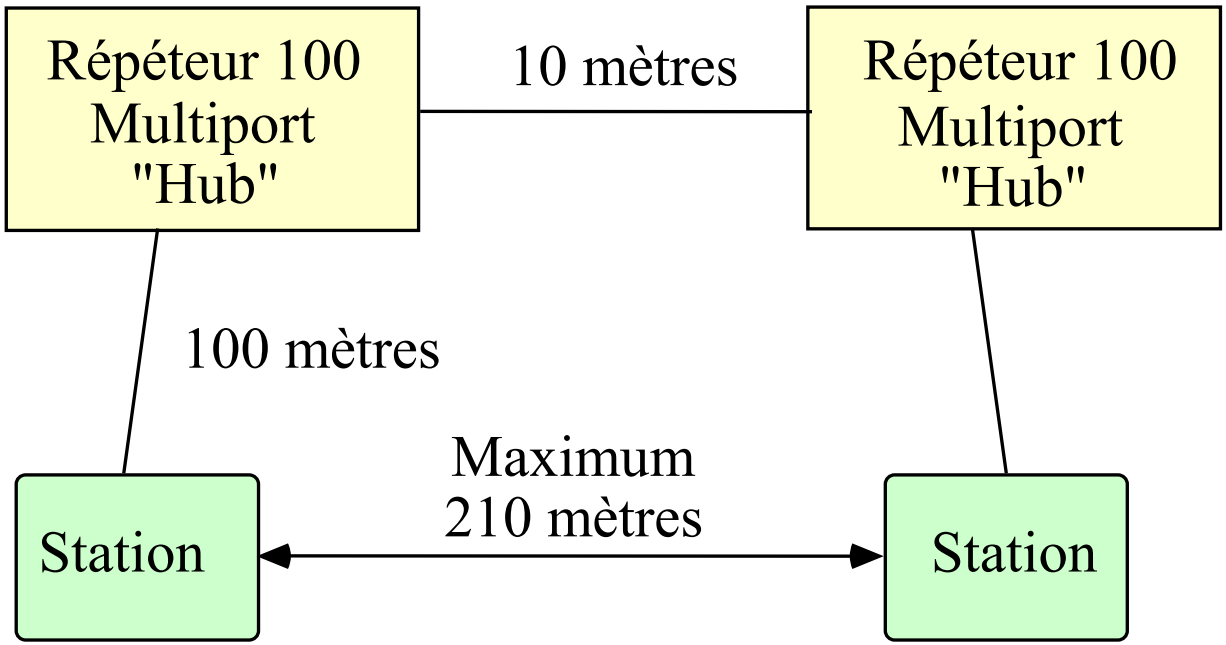
les standards à 100 Mb/s au niveau physique

Standard 1OO BASE T "Fast Ethernet" "IEEE 802.3u"

- **Objectif** : conserver une **compatibilité maximum** avec le standard 10 Mb/s => Changer uniquement le débit.
- **Ethernet** 100 base T conserve d'Ethernet 10 **tout le niveau liaison** : le protocole **CSMA/CD**, le **format des trames** etc...
- **Modifications temporelles**: **ST 5,12 µs , IFG 0,96 µs.**
- **Niveau Physique : plusieurs standards** selon les supports de communication (adaptation du nombre de paires et adaptation des codages).



Ethernet 100 Base TX sur paires torsadées: architecture



- Fonctionnement avec des **répéteurs 100 type II** : Distance station répéteur 100 mètres, distance répéteur-répéteur 10 m.
- **Répéteur Ethernet 10/100**: un seul débit 10 Mb/s ou 100 Mb/s à un instant donné.

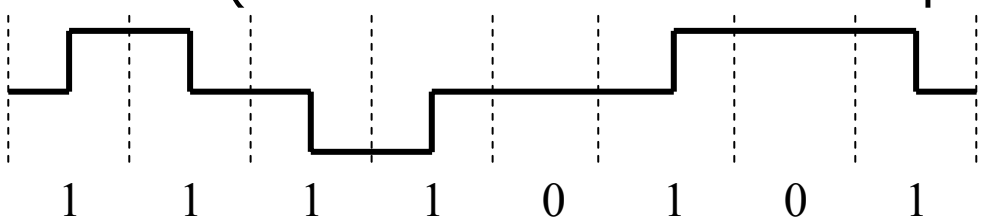
Le standard 100 Base TX: quelques éléments de niveau physique

- **Utilisation de 2 paires UTP 5 norme EIA 568** avec connecteur RJ45 (également STP avec connecteur DB9).
- **Gestion des signaux sur les paires comme en 10 BAS T**

. une paire pour l'**émission** (données)

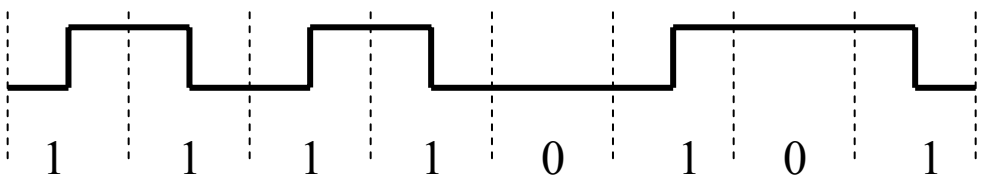
. une paire pour la **réception/détection des collisions**

- **Débit possible sur UTP5 125 Mb/s (et plus).**
- **Utilisation d'un niveau physique (très voisin de FDDI):**
  - **Codage 4B/5B** : Au moyen d'une table, 4 bits ('nibble') sont mis en correspondance avec 16 codes groupes de 5 bits, pour garantir un front tous les trois bits au plus. Les autres codes groupes de 5 bits servent pour la signalisation (3) ou sont inutilisés (11).
  - **Modulation MLT3** ('**M**ulti **L**evel **T**ransmission-3' pseudo ternaire).



Le standard 100 Base FX

- Appartient au standard 100 base X donc il est **très voisin du standard 100 Base Tx**.
- Fonctionnement sur **deux fibres** multi modes au lieu de deux paires torsadées.
- Distance maximum : **400 mètres** en mode half duplex et **2000m** en mode full duplex.
- Utilisation du code **4B/5B** / modulation **NRZI**.



Le standard 100 Base T4

- **Uniquement en mode half duplex avec utilisation de 4 paires UTP de catégorie 3** (ou supérieures)
  - Trois paires pour l'émission/réception (données)
  - Une paire pour la détection des collisions
- **Modulation à 25 Mbaud/s sur UTP3.**
- **Utilisation d'un code ternaire**
  - Code à trois niveaux: -1 , 0 , +1 (codage de '**trits**')
- **Code 8B6T ("8 bits" codés par 6 "trits").**
  - **3 trits** représentent 3*3*3 = 27 valeurs.
  - 27 valeurs différentes permettent de coder **4 bits** (plus d'autres).
  - On code 8 bits sur 6 symboles ternaires (2 fois 3 trits successifs).
- **Débit: (25 mbaud) * 4 bits =100 Mb/s.**

Le standard 100 Base T2

- **Utilisation de 2 paires UTP 3 ou de qualité supérieure.**

  - 'Dual duplex baseband' : on émet simultanément sur deux paires dans les deux sens. Sur une paire le signal qui circule est une somme des signaux émis dans les deux sens

  - Le signal reçu est obtenu par soustraction dans le signal qui circule du signal émis.

- **Utilisation d'un code à 5 niveaux (valence 5)**
  - -2, -1 , 0 , +1, +2
- **Code PAM 5x5** (Pulse Amplitude Modulation).
  - On code 4 bits par deux symboles 5 niveaux (en parallèle sur 2 paires)
  - **Débit:** (25 mbaud) * 4 bits =100 Mb/s.
- **Une norme astucieuse mais chère et sortie tardivement**

Conclusion : Ethernet 100

- **Le standard de base** Ethernet.
- En **développement important.**
- Prix des cartes **bon marché**.
- Inconvénient important pour un réseau local d'entreprise en mode partagé =>**faible extension géographique**
- Utilisation plus importante **en mode commuté**
  - Distances **plus grandes**.
  - **Inter fonctionnement** entre Ethernet 10Mb/s et 100Mb/s).

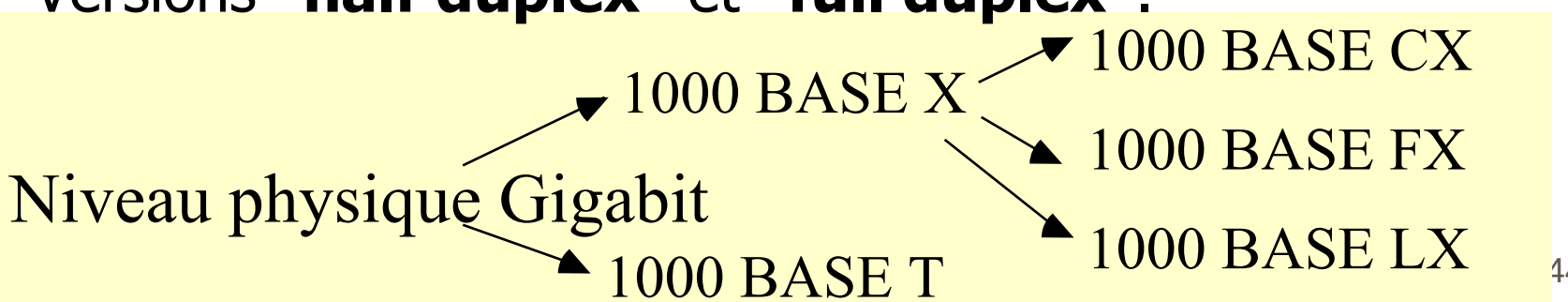
Réseaux locaux Ethernet

Ethernet:

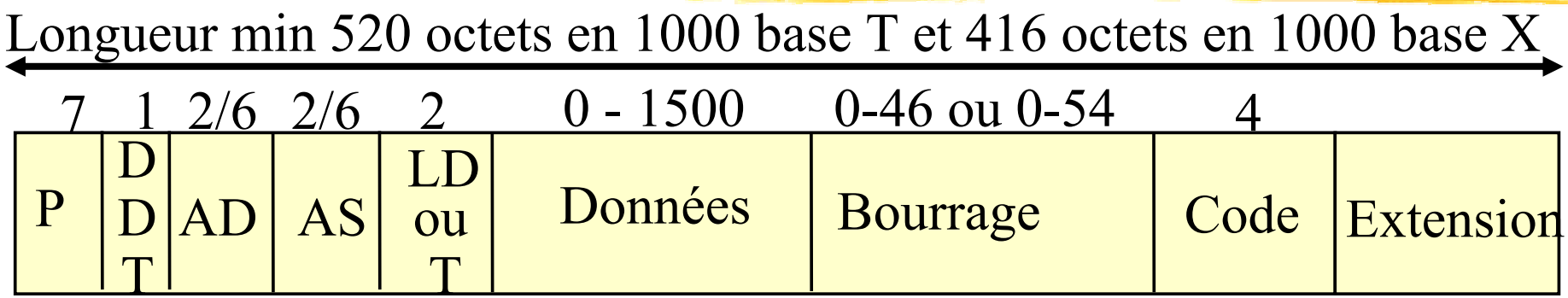
les standards au Gigabit/seconde

Introduction: Ethernet Gigabit

- Refaire une **multiplication par 10 du débit** (1995-1998).
- Pour aller vite récupération des deux technologies: **Ethernet 802.3 et Fibre Channel X3T11.**
- Utiliser la **technologie 0,3 microns** (interfaces à bas prix).
- Créer un réseau gigabit qui **apparaisse** du point de vue des couches supérieures comme un **réseau Ethernet habituel**.
  - . **Format** des trames identique.
  - . Niveau MAC **compatible** (adresses, diffusions, ...)
  - . Administration **identique** (10 , 100, 1000 Mb/s).
- Versions **"half duplex"** et **"full duplex"**.

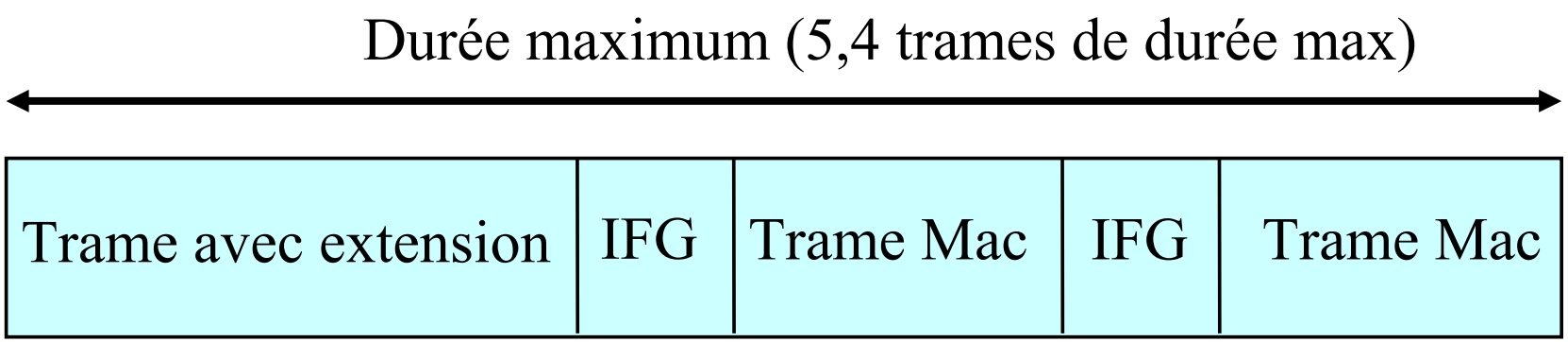


Niveau MAC : Ethernet Gigabit partagé IEEE 802.3z Half Duplex



- **Format des trames inchangé** (compatibilité logicielle) => Nécessité de détecter les collisions sur la trame la plus courte.
- **Longueur min 512 bits à 1000 Mb/s** donne un délai de détection de collision de 512 nanosecondes => **trop court**.
- **Allongement de la trame minimum** au niveau physique.
- Une trame Mac de 64 octets est complétée si nécessaire par **bourrage (notion d'extension de trame).**
- On atteint ainsi un diamètre de collision de 200 m : **les stations peuvent se trouver à 100 m** d'un hub gigabit.

Niveau MAC : Ethernet Gigabit Rafales de trames ('Frame Bursting')



- Transmission d'une **première trame** avec éventuellement bits d'extension pour acquérir la voie.
- Ensuite **émission d'une rafale de trames** (' frame burst ') sans repasser en mode acquisition pour une durée maximum de 5,4 trames de taille maximum**.**
- Les silences inter trames sont **garnis par des bits d'extension.**

Niveau MAC 802.3x : Contrôle de flux ("Flow Control")

- **Haut débit** => ne pas perdre de trames par écrasement dans les tampons d'entrée du récepteur .
- Introduction dans Ethernet Gigabit **d'une technique de contrôle de flux** au niveau liaison (mode full duplex).
- Une solution retenue très rustique **de type arrêt et attente (X-On / X-Off).**
- Une trame de contrôle MAC (8808) baptisée "**Pause**" (0001), permet à un destinataire de **demander à un émetteur de suspendre pour un certain délai ses émissions** (en nombre de slots, exemple 3).
- La même trame "**Pause**" **avec un délai nul** permet de mettre fin avant terme à l'arrêt.

	2	2	2	
Entête	Contrôle MAC x8808	Code Pause x0001	Durée de pause x0003	Bourrage
	Type/longueur	Donnés + Bourrage		

Niveau physique : Ethernet Gigabit 1000 Base X

Standard 1000 BASE LX

- Laser ondes longues sur fibres monomodes ou multi modes (1300 nanomètres).
- Distance 550m à 3 km selon les fibres.

Standard 1000 BASE SX

- Laser ondes courtes (850 nanomètres) sur fibres multi modes.
- Distance 250 à 550 m selon les fibres.

Standard 1000 BASE CX

- Ethernet gigabit sur paires torsadées (2 paires STP Shielded Twisted Pairs)
- Distance 200m.

Niveau physique : Codage 1000 Base X

- **Origine : Fibre Channel niveau 1 FC-1**
- **Objectifs du codage** 8B/10B NRZ (origine IBM)

  => **Minimisation du bruit** (des erreurs)
  - Maintenir "l'équilibre" i.e. le même nombre de bits à 1 que de bits à 0.

  => **Amélioration de la synchro bit**

  => **Détection d'erreurs**

  => **Séparation données/contrôles**
  - Pour des données normales utilisateur **D-type**
  - Pour des données protocolaires **K-type** (Caractères spéciaux par exemple délimiteurs, signalisation)

Niveau physique : codage 1000 Base X 8B10B

- **Code 8B/10B: Représentation des octets 8 bits par des symboles 10 bits.**
  - 512 symboles pour les données (2 représentations par octet)
  - Quelques symboles pour la signalisation
  - Les autres symboles sont invalides (permet la détection d'erreurs)
- **RD "Running Disparity"**
- Indicateur de la disparité entre les 1 et les 0.
  - Si un groupe de bits à autant de 1 que de 0 => RD inchangée
  - Si un groupe de bits à plus de 1 que de 0 => RD positive
  - Si un groupe de bits à moins de 1 que de 0 => RD négative
  - Chaque donnée significative à deux représentations définies par des tables :L'une en cas de RD positive, L'autre en cas de RD négative En cas de RD inchangée la norme définit un des deux codes à utiliser.

Niveau physique : codage 8B10B Exemple

- **Exemple : Le caractère spécial (**type K) **Délimiteur** appelé "virgule"

Octet x'BC' b'1011 1100'

Découpage 3 bits+5 bits 101 11100

Notation normalisée Zxx.yy

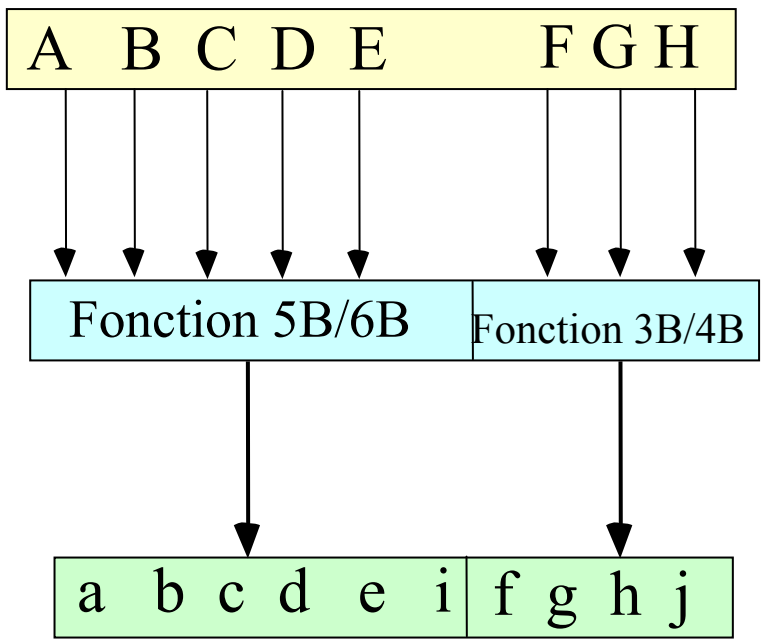
Z: Type K ou D Virgule : K28.5

xx:DEFGH décimal .yy:ABC en décimal

- **Symboles émis sur 10 bits**

avec RD négative 001111 1010

avec RD positive 110000 0101



Niveau physique : Ethernet Gigabit 1000 Base T

- **Support 4 paires torsadées de catégorie 5.**
- **Récupération des progrès** réalisés a propos de l'Ethernet 100
  - 100 Base T a montré qu'on peut transmettre à 125 Mb/s sur UTP 5.
  - 100 Base T4 a montré qu'on peut transmettre sur trois paires.
  - 100 base T2 a montré que l'on peut transmettre en PAM 5 simultanément dans les deux sens.
- **Transmission octets par octets**
  - Encodage initial (code correcteur d'erreur FEC forward error correction)
  - A chaque intervalle, transmission sur 4 paires d'un octet sous la forme de 4 symboles PAM5 (chaque symbole de valence 5 code 2 bits plus une information de code correcteur d'erreur).
  - Débit atteint 125 mbaud/s x 8 bits = 1 gigabit/s.

Conclusion : Ethernet Gigabit

- Une **utilisation déjà importante** dans les réseaux locaux au niveau infrastructure (‘ backbone ’).
- **Cartes réseaux sur PC à un prix accessible.**
- Le mode **half duplex a été maintenu** mais le standard a vocation a être **utilisé en mode full duplex**.
- Avenir de ce standard : la **connexion filaire** de stations de travail.

Réseaux locaux Ethernet

Ethernet:

les standards à 10 Gigabits/s

Introduction: Ethernet 10 Gigabits

- **Groupe de travail IEEE** à partir de 1999. Publication de la norme IEEE802.3 ae en 2002**. Fin des travaux 2006**
- **Objectif poursuivi**: créer un réseau compatible Ethernet.
  . **Format** des trames identique (adressage, tailles min et max, **...**)
  . Compatibilité fonctionnelle norme 802.3 (contrôle de flux, …)
- **Trois particularismes**
  - Version **"full duplex" uniquement.**
  - Distance possible sur fibre monomode **40 km**.
  - Compatibilité d'une version 10G avec le réseau longue distance sur fibre optique : **SONET OC-192c / SDH VC4-64C**
- **Niveau physique** : codages X, R, W médium S, L, L4, E, T

Niveau physique: Les codages

- 10G BASE X utilise le codage **8B/10B**.
- 10G BASE R utilise le codage **64b/66b**.
- 10G Base W utilise comme support de communication des conteneurs SONET OC192 ou SDH VC 64 .

Niveau physique: Fibres optiques

- **10G Base S**
- Deux fibre multimodes 850 nanomètres série.
- Distance maximum 65 mètres
- **10G Base L4**
- Deux fibre multimodes 1310 nanomètres multiplexage en longueur d'onde (utilisation du multiplexage en longueur d 'onde WDM Wavelenght Divison Multiplexing.
- Distance maximum 300 mètres
- **10G Base L**
- Deux fibre monomodes 1310 nanomètres
- Distance maximum 10 kilomètres
- **10G Base E**
- Deux fibre monomodes 1550 nanomètres
- Distance maximum 40 kilomètres

Exemples d'Ethernet 10G sur fibre

- En combinant un choix de codage et un choix de fibre.

- 10GBase-SR 850-nm serial LAN Multimode 65
- 10GBase-LX4 1,310-nm WDM LAN Multimode 300
- 10GBase-LR 1,310-nm serial LAN Single-mode 10,000
- 10GBase-ER 1,550-nm serial LAN Single-mode 40,000
- 10GBase-SW 850-nm serial WAN Multimode 65
- 10GBase-LW 1,310-nm serial WAN Single-mode 10,000
- 10GBase-EW 1,550-nm serial WAN Singel-mode 40,000

Niveau physique: Ethernet 10G Base T

- Le standard à 10 GB sur paires torsadées
- Utilisation de 4 paires torsadées UTP catégorie 6 ou 7 (Class E ou Class F).
- Distances maximum dépendantes de la paire utilisée.
  - 100 m sur catégorie 7 (class F)
  - 55 m sur catégorie 6 (class E)
- Mode full duplex avec annulation d'écho.
- Code PAM 10 , 10 niveaux codent 3 bits.
- Rapidité de modulation 833 Mégabaud.
- Débit 4 paires x 833 mégabaud x 3 bits/baud=10GB.
- Difficultés de mise au point.

Conclusion : Ethernet 10 Gigabit

- Encore **en cours de mise au point.**
- **Des produits déjà diffusés.**
- Pour l'instant un standard pour **opérateurs ou réseaux très importants**.

Réseaux locaux Ethernet

Ethernet:

L’auto négociation

Caractéristiques générales de l'auto-négociation

- L'auto-négociation sert à **déterminer la meilleure option possible de communication** entre deux extrémités d'une voie sur paires torsadées.
- Il fonctionne par **échange des capacités de communication** entre partenaires Ethernet (entre deux stations connectées directement ou entre une station et un répéteur ou un commutateur).
- La négociation réalisée par un protocole situé au **niveau physique**.
- Uniquement définie pour les **versions Ethernet paires torsadée** à partir de 802.3u Ethernet 100.
- Ce protocole est exécuté à **l'initialisation d'une voie** de communication ou lors d'une **restauration manuelle** d'une liaison.
- Origine **NEC** Nwau, normalisation **IEEE 802.3 u** (1995)

Tableau des modes de communication négociables

Priorité	Type	Débit
1	10 Base T Half duplex	10 Mb/s
2	10 Base T Full duplex	2x10 Mb/s
3	100 Base TX Half-duplex	100 Mb/s
4	100 Base T4 Half-duplex	100 Mb/s
5	100 Base T2 Half-duplex	100 Mb/s
6	100 Base TX Full-duplex	2x100 Mb/s
7	100 Base T2 Full-duplex	2x100 Mb/s
8	1000 Base T Half-duplex	1000 Mb/s
9	1000 Base T Full duplex	2X1000 Mb/s

Fonctionnement de l'auto-négociation

- En 10 base T émission initiale d'une **séquence d'impulsions** (**NLP 'Normal Link Pulse'**) puis entretien de l'état opérationnel par des NLP chaque 16 milliseconde.
- L'auto-négociation transforme à l'initialisation les signaux NLP en signaux baptisés **FLP ('Fast Link Pulse'),** qui véhiculent des mots de 16 bits **LCW** ('Link Code Word) qui codent le niveau de communication possible.

S0	S1	S2	S3	S4	A0	A1	A2	A3	A4	A5	A6	A7	RF	Ack	NP

- **Le LCW de base**
  - S0 à S4 : Sélecteur de réseau local (code 00001 pour Ethernet)
  - A0 : 10 Base T HD, A1 : 10 Base T FD, A2 : 100 Base TX HD, ….
  - RF : Remote Fault, signalisation d'erreur à l'extrémité distante
  - Ack : Acquittement de réception d'un mot LCW de 16 bits reçu.
  - NP : Next Page existence d'un autre mot à suivre .
- Un système distant qui possède les fonctions d'auto négociation acquitte et **propose son propre niveau**.

Modes de détection parallèle

- En l'absence du protocole d'auto négociation (pas de signaux FLP) possibilité **de détection parallèle** ('parallel detection'):
  - Si un système ne sait générer que le signal NLP le mode de communication sélectionné est le 10 Base T half duplex.
  - Observation des signaux générés par le partenaire: si un partenaire génère des signaux en 100 base TX ou 100 Base T4, la forme caractéristique de ces signaux permet à un mécanisme dit de détection parallèle de se positionner quand même à 100 Mb/s.
  - Existence de problèmes possibles avec l'auto négociation (sélection d'un niveau trop bas) ou problèmes avec le mode de détection parallèle.

- L'auto négociation en Ethernet gigabit est **modifiée (nouveau LCW,** négociation pour **l'utilisation du contrôle de flux**) .

Conclusion : Ethernet

- **Le standard en réseau local filaire qui a effacé tous les autres.**
- **Un ensemble de concepts qui ont considérablement évolués. En fait Ethernet a été 'réinventé' plusieurs fois ce qui a assuré sa survie et son succès.**
- **Un avenir assuré pour ce standard: prochaine étape envisagée 100 Gigabits/seconde ?**

Bibliographie

- Charles E. Spurgeon, ` Ethernet : the definitive guide ' , Oreilly, 2000
- Alexis Ferréro, ` Les réseaux locaux commutés et ATM ', InterEdition, 1998
- Sites web :
  - Gigabit alliance http://www.10gea.org/

Réseaux locaux partagés

Réseaux locaux sans fils (Wireless LAN) IEEE 802.11 ‘ WIFI ’

Généralités
Niveau liaison
Niveau physique

Généralités Wifi: différentes catégories de réseaux sans fils

- **1) Réseaux personnels sans fils (WPAN)**
  - **Bluetooth (IEEE 802.15.1), HomeRF, ZigBee** IEEE (802.15.4)**, Infrarouges**
- **2) Réseaux locaux sans fils (WLAN)**
  - **WiFi IEEE 802.11, HiperLAN, DECT**
- **3) Réseaux métropolitains sans fils (WMAN)**
  - **Norme IEEE 802.16 (boucle locale radio) Wimax**
- **4) Réseaux étendus sans fils (WWAN)**
  - **GSM (*Global System for Mobile)*, GPRS (*General Packet Radio Service*), UMTS (*Universal Mobile Telecommunication System*).**

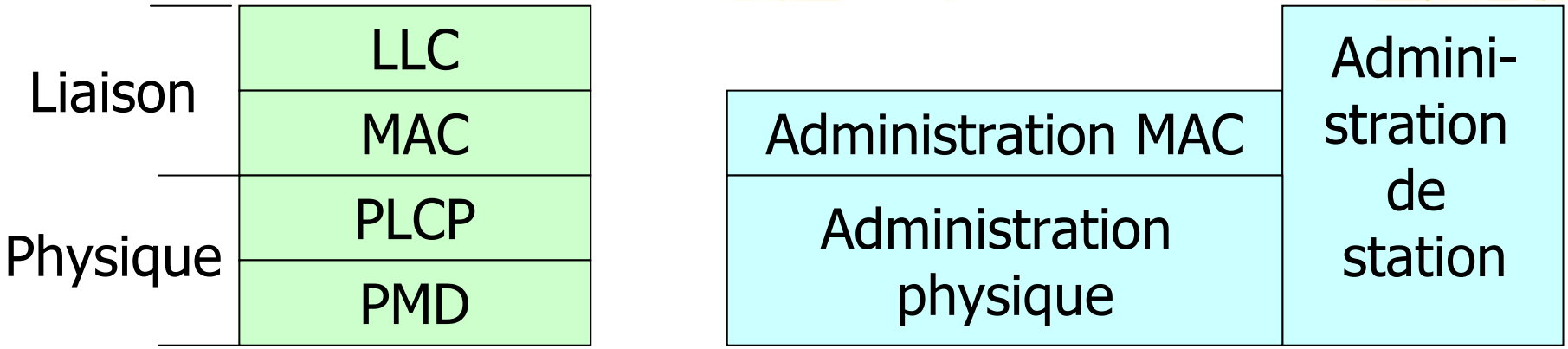
Généralités Wifi ('Wireless Fidelity')

- **Recherches** sur les réseaux locaux sans fils depuis le **début des années 1970**.
- **Normalisation** wifi: fin des années **1990**.
- **Avantages** du sans fil
  - **Ne pas avoir à câbler** un bâtiment.
  - Plus de **souplesse** et de **mobilité**.
- **Déploiement** des réseaux Wifi
  - Réseaux **domestiques.**
  - En **entreprise.**
  - Lieux de **fort passage** ('hotspots').
  - **Fournisseurs** d'accès sans fils WISP 'Wireless Internet Service Provider'

Généralités WIFI: Quelques éléments d'architecture

- **1) WIFI : un réseau local radio**.
  - Définition sur les deux niveaux **physique** et **liaison**.
- **2) WIFI : deux organisations architecturales**.
  - Le mode **infrastructure (centralisé)**.
  - Le mode **ad 'hoc (distribué)**.
- **3) WIFI : deux protocoles différents** d'accès au médium.
  - **PCF** 'Point Coordination Function' **(en coopération)**.
  - **DCF** 'Distributed Coordination Function' **(en compétition)**.
  - Pouvant être utilisés simultanément par une station.
- **4) WIFI : différents niveaux physiques** selon le débit, le codage, la bande de fréquences utilisée.
  - 802.11, 802.11a , **802.11b , 802.11g**, en cours 802.11n.
- **5) Consortium de développement : WIFI Alliance**

Généralités WI FI : Le modèle de référence WIFI



- LLC ‘ Logical Link Control ’.
- MAC ‘Medium Access Control’.
- PLCP ‘Physical Layer Convergence Protocol’.
- PMD ‘Physical Medium Dependent’.
- Administration physique (Physical Management).
- Administration MAC (MAC Management).
- Gestion de Station (Station Management).

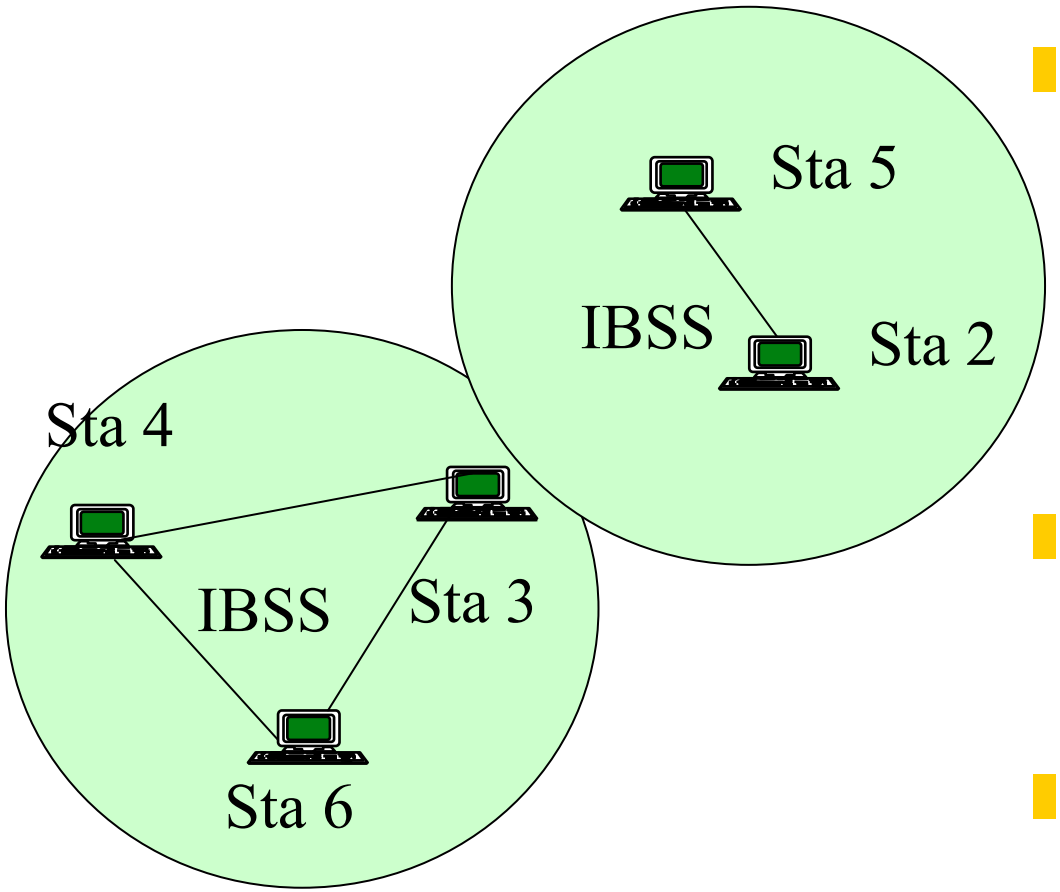
Réseaux locaux partagés
wifi

LE NIVEAU LIAISON

‘MAC Medium Access Control’

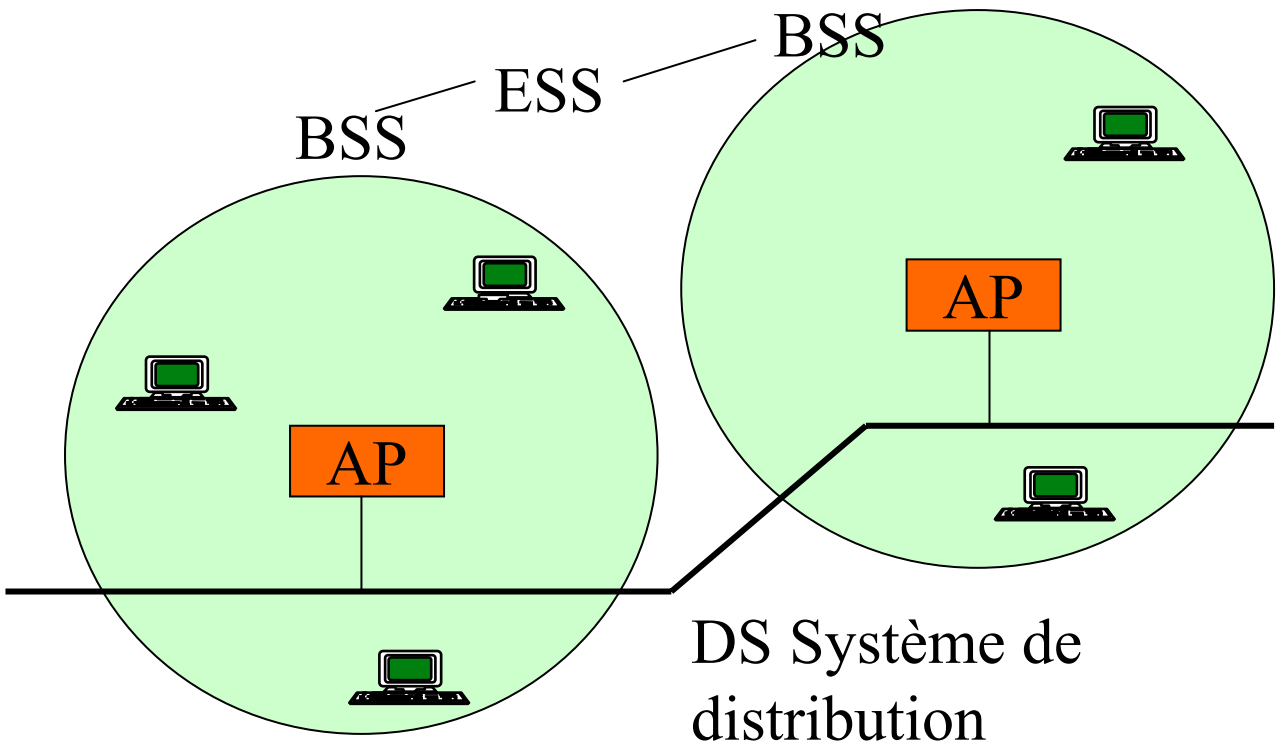
Mode ad ’hoc
Mode infrastructure

Le niveau liaison Wifi : Le mode ad'hoc (distribué)



- IBSS 'Independent Basic Service Set ' : ensemble de stations avec coupleurs sans fils, communicantes dans la même bande.
- Terminologie: mode ad 'hoc, 'peer to peer'
- Protocole DCF : Distributed Coordination Function.

Le niveau liaison Wifi : Le mode ' infrastructure ' (centralisé)



- **AP 'Access Point'** commutateur.
- **Station de travail** avec un coupleur WIFI.
- **BSS** (Basic Service Set): un seul AP.
- **ESS** ('Extended Service Set') : plusieurs AP connectés par un autre réseau (réseau Ethernet ou sans fil).
- Changement de point d'accès: **handover/ roaming** (itinérance).

Le niveau liaison Wifi : DCF ' Distributed Coordination Function '

- 1) Protocole en **compétition** avec **écoute** (CSMA).
- 2) Ajournement **non persistant**.
- 3) **Détection** de collisions par accusé de réception.
- 4) **Retransmission** sur collision (binary backoff).
- 5) Gestion de la **fragmentation.**
- 6) Pas de **gestion de connexion**.
- 7) Pas de **contrôle de flux**.
- 8) Pas de garantie de livraison **sans erreurs**.
- 9) Pas de **qualité de service** (en version de base)

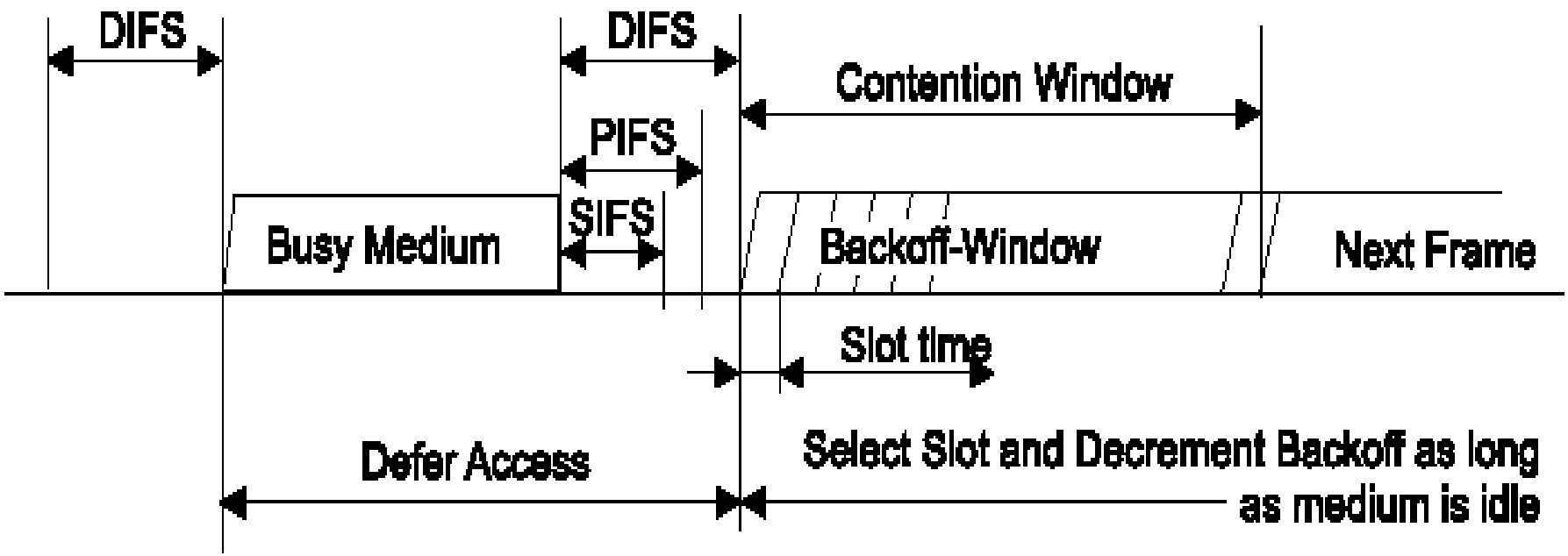
Le niveau liaison Wifi : DCF
Ecoute et ajournement non persistant

- **1) Ecouter** la voie, **transmettre** si la voie est libre.
- **2)** Si la voie est **occupée** choisir un intervalle **d'attente aléatoire ('backoff')** dans l'intervalle [0,CW] (CW contention window).
- **3) Décompter** des intervalles de temps (tranche canal 'Slot Time) quand le medium est libre.
- **4)** Le décompte est **suspendu** quand le médium redevient occupé.
- **5)** Quand l'intervalle d'attente devient **nul et que la voie est libre** commencer à **émettre** (une trame de données ou de contrôle par exemple RTS).

Le niveau liaison Wifi : DCF
Diagramme d'écoute et d'ajournement

Immediate access when medium is free >= DIFS

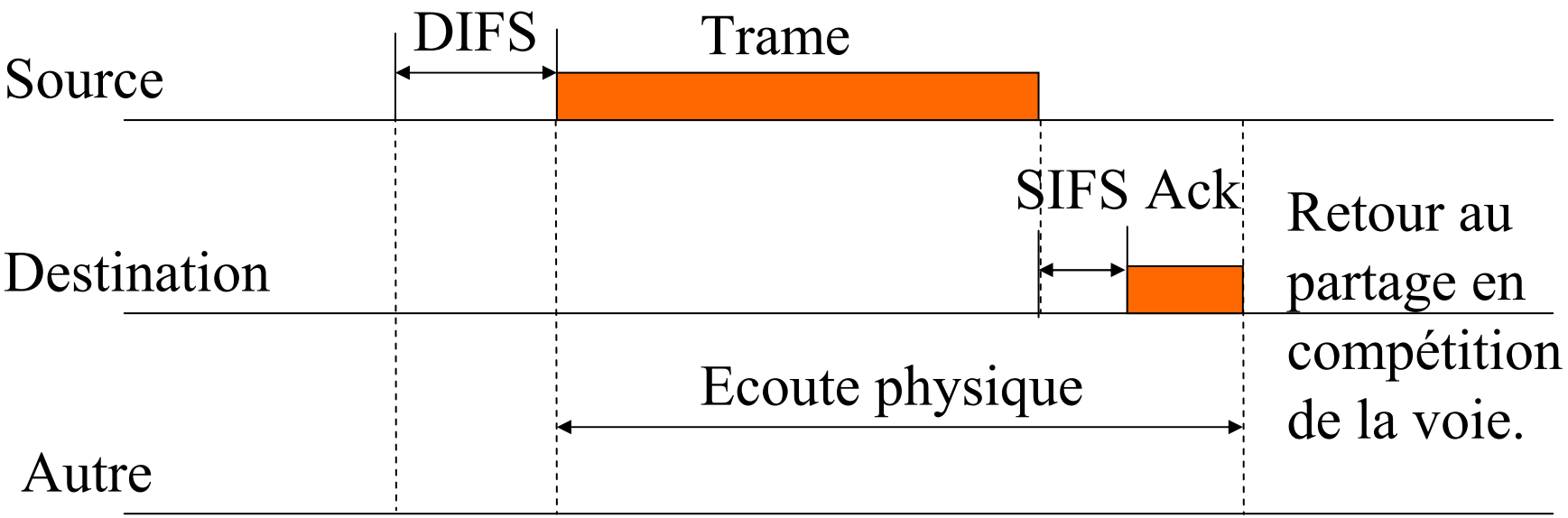
Schéma extrait de la norme 802.11



Notions utilisées

- Silence inter trame (IFS Inter-Frame Spacing)
  DIFS: Distributed IFS (50µs); PIFS: Point IFS(30µs); SIFS: Short IFS (10µs)
- Fenêtre de collision CW ('Contention Window')
- Tranche canal ('Slot Time') 20 µs
- Attente en nombre entier de tranches ('Backoff')

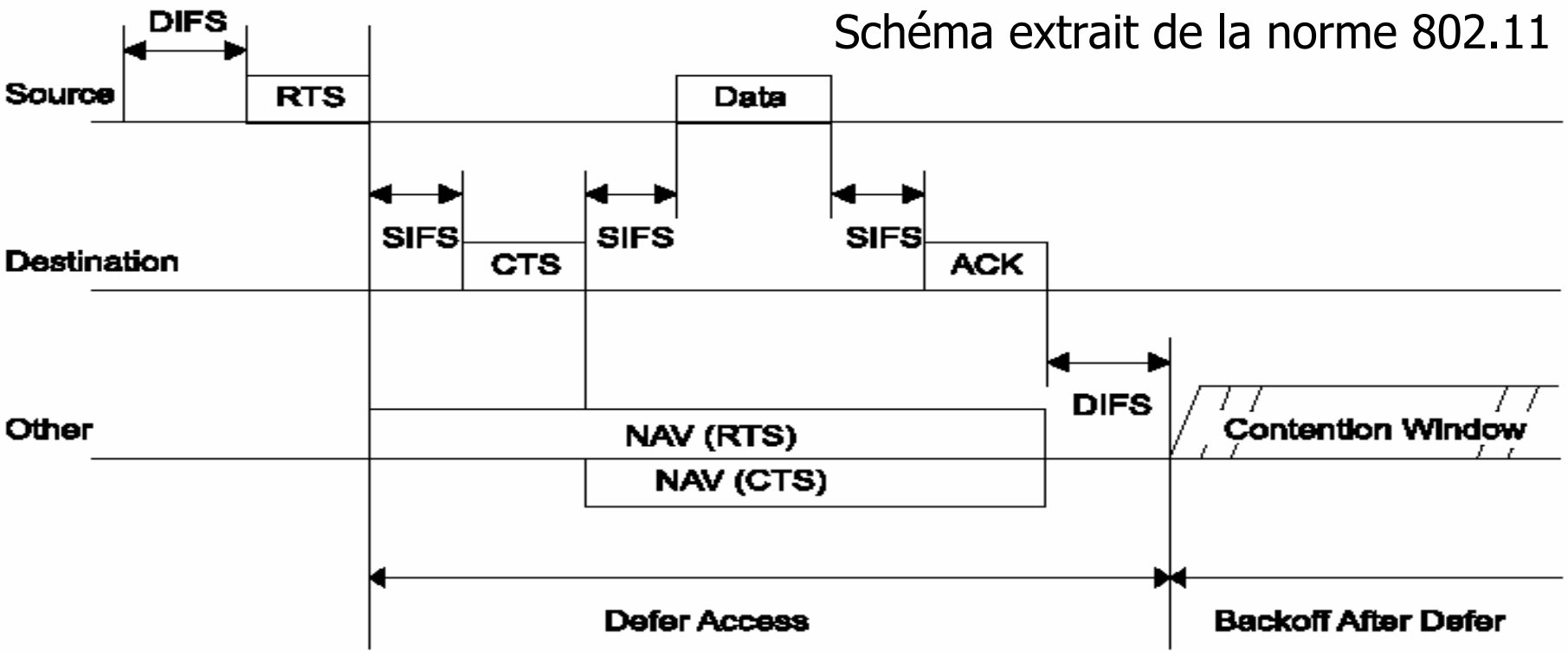
Le niveau liaison Wifi : DCF
Transmission directe d'une trame



- **Acquittement positif ACK** obligatoire après un silence SIFS (protocole PAR 'Positive Acknowledgment Retry' ou encore ARQ Automatic Repeat ReQuest).
- **Mécanisme d'écoute physique** pour une autre station.

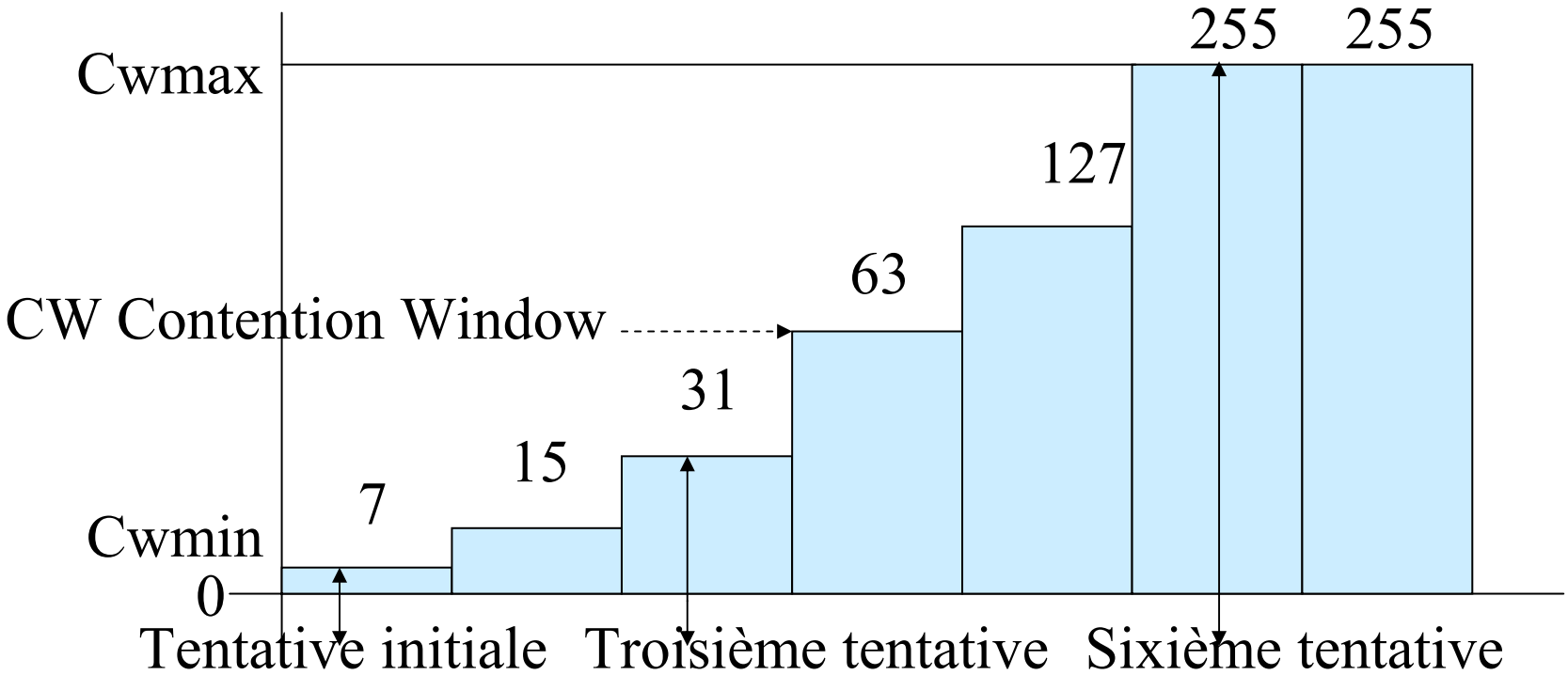
Le niveau liaison Wifi : DCF
Evitement des collisions : CSMA/CA

Schéma extrait de la norme 802.11



- Echange RTS CTS (Request To Send/Clear To Send) pour une trame données.
- Utilisation des silences courts SIFS (l'échange est prioritaire)
- Acquittement positif ACK obligatoire.
- Mécanisme d'écoute virtuelle (indicateur NAV pour une autre station).

Le niveau liaison Wifi : DCF
L'algorithme du retard binaire



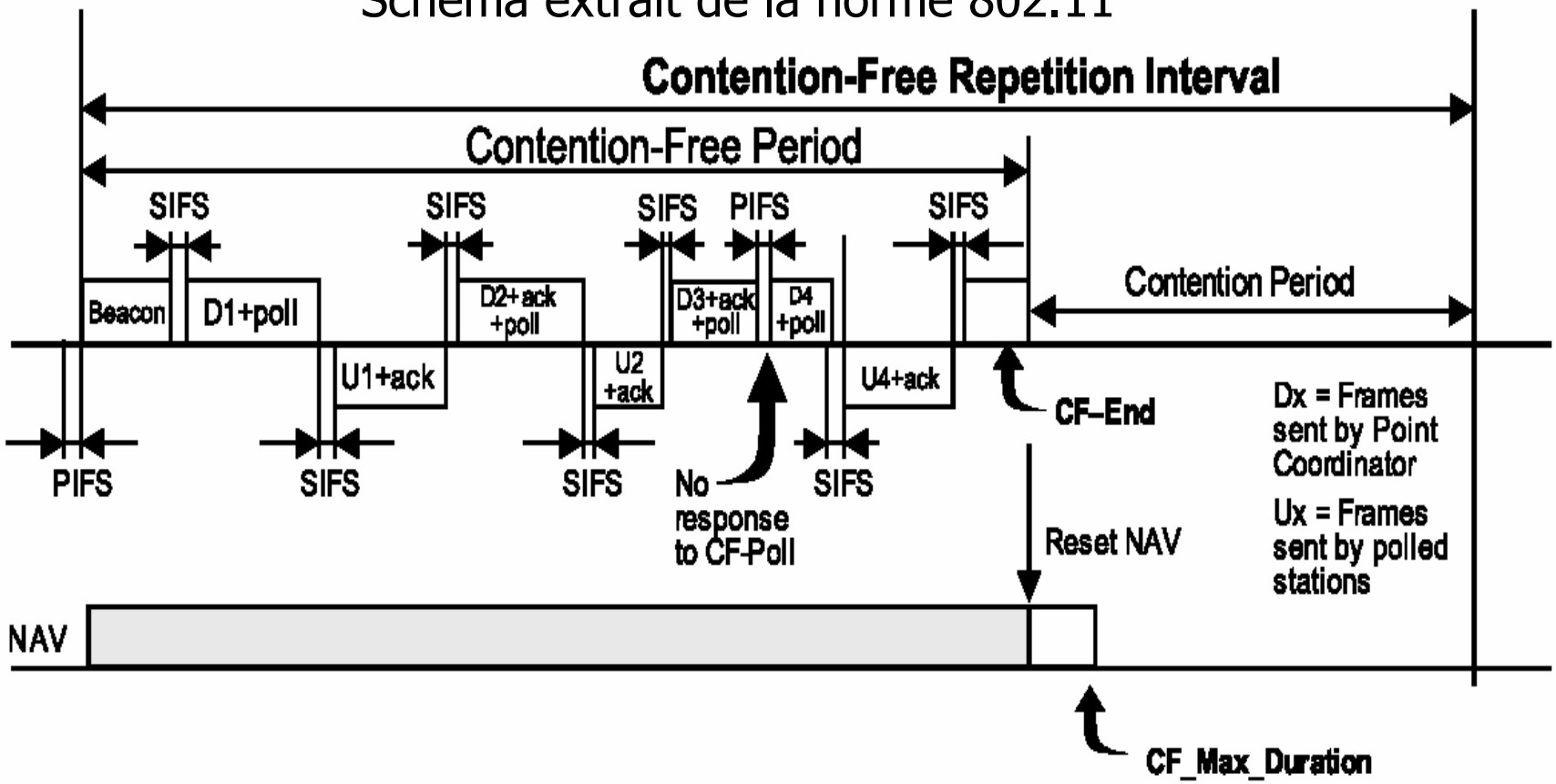
Attente = Random() * ST (Backoff Time).
Random = Entier aléatoire uniformément distribué sur [0,CW].
CW = Entier entre CWmin CWmax qui double à chaque tentative.
ST = Valeur caractéristique du niveau physique (Slot Time).
802.11a et g : cwmin=15 cwmax=1023; **802.11b** : cwmin=31 cwmax=1023

Le niveau liaison Wifi : PCF ' Point Coordination Function '

- 1) Fonctionnement en **scrutation ('polling')** par le PC ('**P**oint **C**oordinator').
- 2) Une station **émet** si elle est **autorisée par le PC.**
- 3) Le PC **sélectionne** une station en plaçant son adresse dans la trame.
- 4) Les trames sont acquittées. Si l'acquittement ne revient pas le **PC ou la station effectuent la retransmission.**
- 5) **PCF** a plutôt été destiné à des échanges à **qualité de service**.

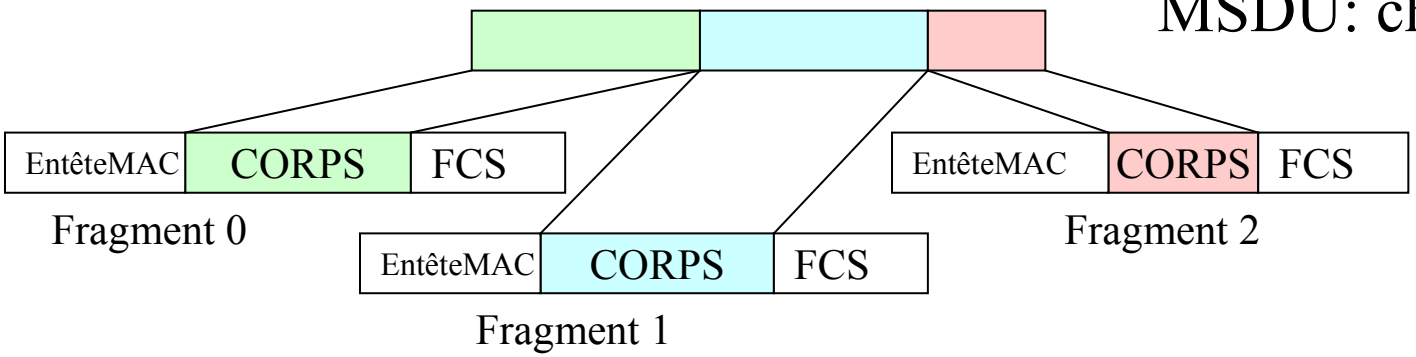
Le niveau liaison Wifi : PCF
Protocole de scrutation

Schéma extrait de la norme 802.11



- Intervalle de répétition: mode PCF (contention free) puis DCF.
- PIFS puis trame Beacon : ouverture d'un intervalle sans collision (mode PCF)
- CF-end fin de séquence Poll Ack sous contrôle du PC.

Le niveau liaison Wifi : autre fonction Fragmentation



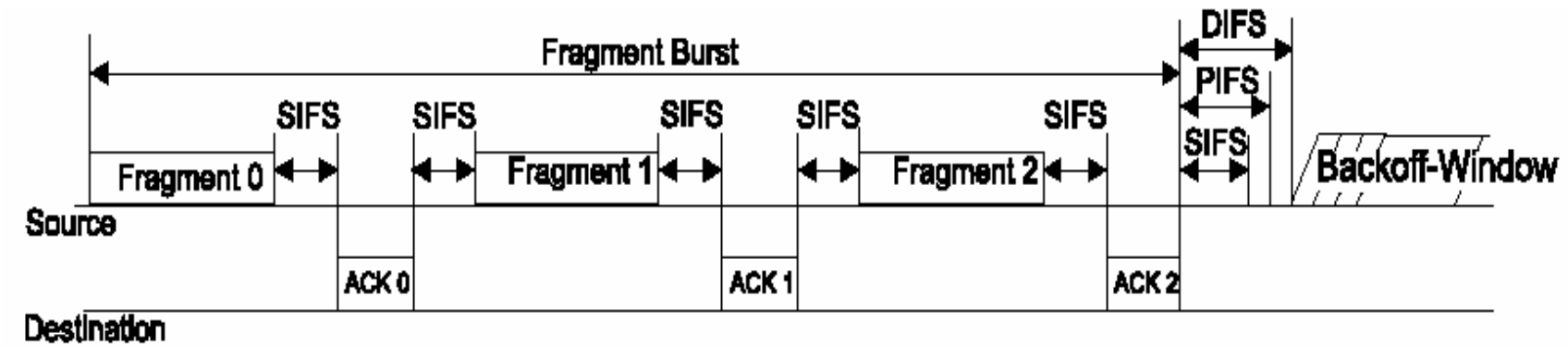
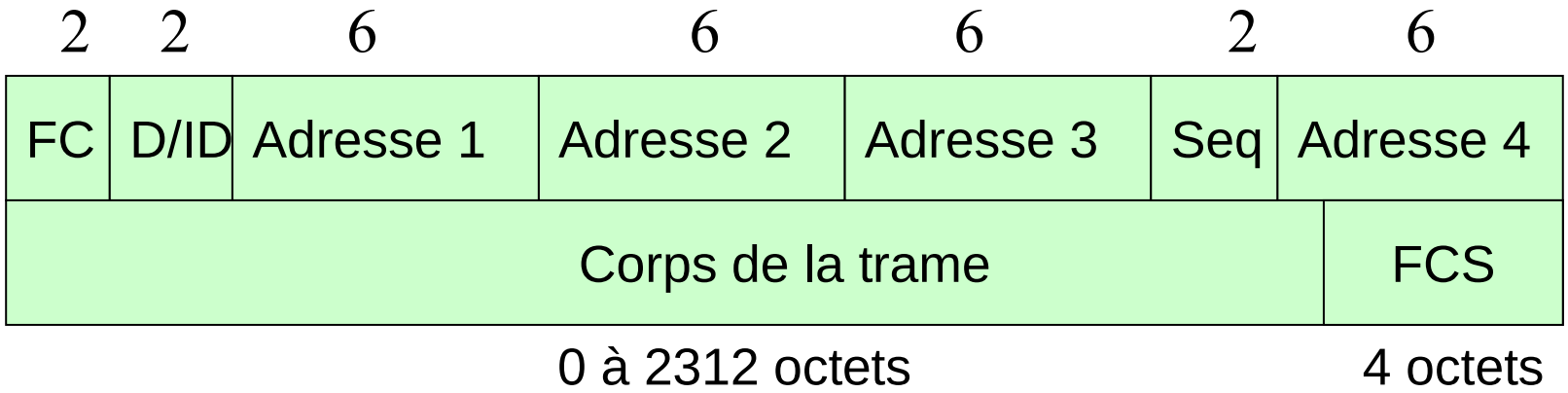


Schéma extrait de la norme 802.11

Le niveau liaison Wifi : Format de trame (MAC)



- **1) FC** (Frame Control): version de protocole, type de trame …etc.
- **2) Durée / ID** : Durée d'utilisation du canal de transmission.
- **3) Champs adresses** : Une trame peut contenir jusqu'à 4 adresses (mode ad'hoc adresse 1 destination et adresse 2 source).
- **4) Contrôle de séquence** : pour la fragmentation (numéro de fragment sur quatre bits et numéro de séquence de la trame sur douze bits).
- **5) Corps de la trame** : charge utile d'au maximum 2312 octets.
- **6) FCS** (Field Check Sequence): somme de contrôle de niveau MAC : $x32+ x26+ x22+ x16+ x 2+ x11+ x10+ x8+ x7+ x5+ x4+ x2 + x+1$

Approfondissement: la zone contrôle de trame (FC)

Version (2 bits)		Type (2 bits)		Sous-type (4 bits)			
To DS (1)	From DS (1)	More Frag (1)	Retry (1 bit)	Power mngt (1)	More data(1)	WEP (1 bit)	Order (1 bit)

- **1) Version du protocole**: Actuellement 0 en première version.
- **2) Type et sous type:** Définition du type de la trame (2 bits + 4 bits).
- **3) To et From DS (Distribution System)** : Trame vers ou en provenance du système de distribution (AP point d'accès). Les 2 bits à 0 mode Ad 'hoc.
- **4) More** : Il reste des fragments à émettre (bit more de la fragmentation).
- **5) Retry**: La trame est une retransmission d'une trame précédente erronée.
- **6) Power management** : A 1 la station entre en mode économie.
- **7) More data** : A 1 des données sont à émettre vers une station en économie.
- **8) WEP** : A 1 la trame est chiffrée en WEP (Wireless Equivalent Privacy).
- **9) Order** : Trame de la classe de service strictement ordonné.

Approfondissement: quatre cas de transmission en wifi

- **Cas 1** (mode ad'hoc): Transmission **directe** entre deux stations (dans un IBSS).
- **Cas 2** (mode infrastructure) : Transmission d'une **station vers le point d'accès** (qui doit ensuite relayer vers une station destinataire).
- **Cas 3** (mode infrastructure) : Transmission par un **point d'accès** d'une trame vers son **destinataire**.
- **Cas 4** (mode infrastructure avec réseau de distribution sans fil): Transmission **intermédiaire** d'une trame **d'un point d'accès à un autre point d'accès.**

Approfondissement: rôles des adresses MAC

Cas	To DS	From DS	Adresse 1	Adresse 2	Adresse 3	Adresse 4
1	0	0	DA	SA	BSSID	N/A
2	1	0	BSSID	SA	DA	N/A
3	0	1	DA	BSSID	SA	N/A
4	1	1	RA	TA	DA	SA

- Toutes les adresses sont au format IEEE 802 sur 48 bits.
- DA (Destination Address) : Adresse destination .
- SA (Source Address) : Adresse source.
- BSSID : Adresse du 'Basic Service Set' soit l'adresse MAC de l'AP (mode infrastructure) soit l'adresse de l'IBSS (mode ad 'hoc).
- RA : adresse de l'AP destinataire (dans le système de distribution sans fils).
- TA : adresse de l'AP source (dans le système de distribution sans fils).
- N/A : Non applicable.

Réseaux locaux partagés wifi

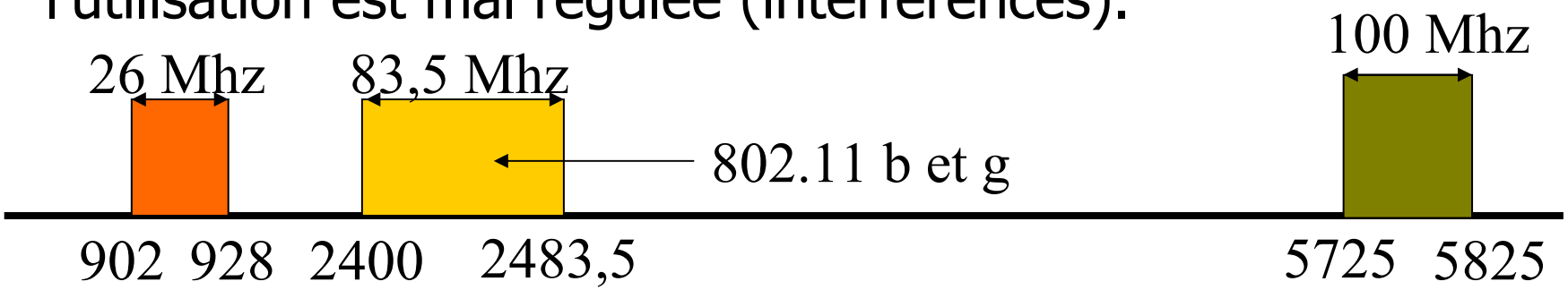
Le niveau physique

802.11 b

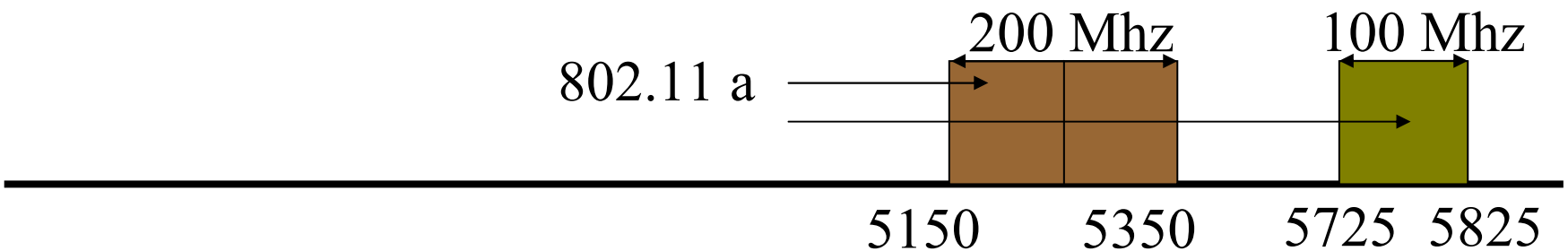
802.11 g

WIFI niveau physique : bandes de fréquences utilisées

- **Bandes de fréquences "ISM "** (**I**ndustrie /**S**cientifique /**M**edicale), aucune autorisation n'est nécessaire mais l'utilisation est mal régulée (interférences).



- **Bande de fréquences "U-NII"** (Unlicensed-National Information Infrastructure). IEEE 802.11 a



WIFI niveau physique : Principaux standards

- IEEE 802.11 (1997) : 1 et 2 Mb/s (ISM 2,4 Ghz).
- **IEEE 802.11b** (1999) : 1, 2, 5.5, 11 Mb/s (ISM 2,4 Ghz)
- IEEE 802.11a (2001) : 6, 9, 12, 18, 24, 36, 48, 54 Mb/s (U-NII 5Mhz).
- **IEEE 802.11g** (2003) : 1,2, 5.5, 11,6, 9, 12, 18, 24, 36, 48, 54 Mb/s (ISM 2,4 Ghz) Compatible 802.11b
- IEEE 802.11n : En développement.

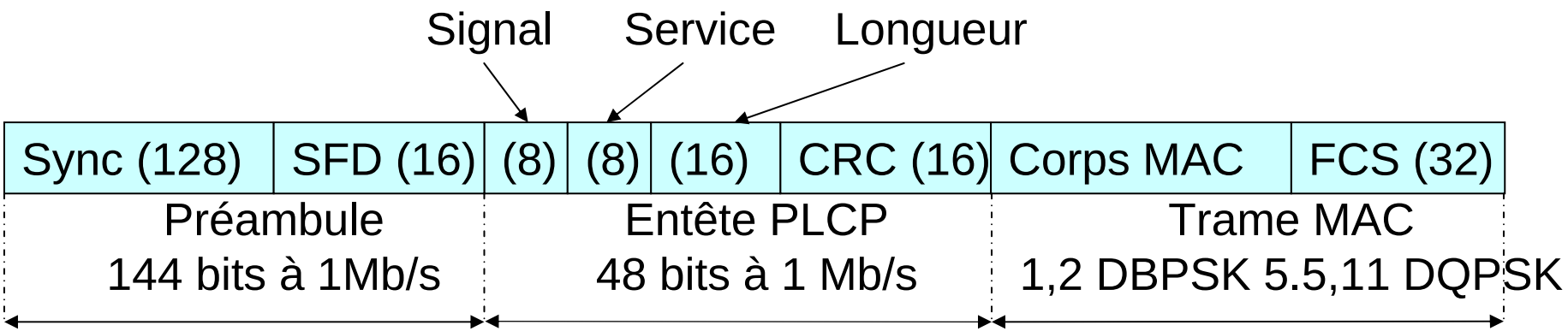
Réseaux locaux partagés wifi : Niveau physique

Le niveau physique wifi selon la norme 802.11b

Niveau physique 802.11b : Principales caractéristiques

- **1) IEEE 802.11**(1997) : Débits 1 Mb/s ('Basic Rate') et 2 Mb/s ('Extended Rate') selon trois codages,
  - **FHSS** ('Frequency Hopping Spread Sequence')
  - **DSSS** ('Direct Sequence Spread Spectrum')
  - **IR** (Infra rouge).
- **2) IEEE 802.11 b** (1999): **Amélioration des codages** 802.11 1997 (baptisés HR/DSSS ('High rate') pour atteindre les débits de 5,5 Mb/s et 11 Mb/s (Enhanced rates).
- **3) 802.11b : Adaptation du débit** (technique de codage) en fonction du rapport signal à bruit ('variable rate shifting').
- **4) 802.11b : 'Indication' des distances en intérieur** : 11 Mbit/s (30 à 45 m) ; 5,5 Mbit/s (45 à 75 m) ; 2 Mbit/s (75 à 100 m) ; 1 Mbit/s (100 à 300 m).
- **5) 802.11b : Utilisation de la bande ISM** 2,4 à 2,4835 Ghz.

Le niveau physique WIFI 802.11 b : Format des trames DSSS

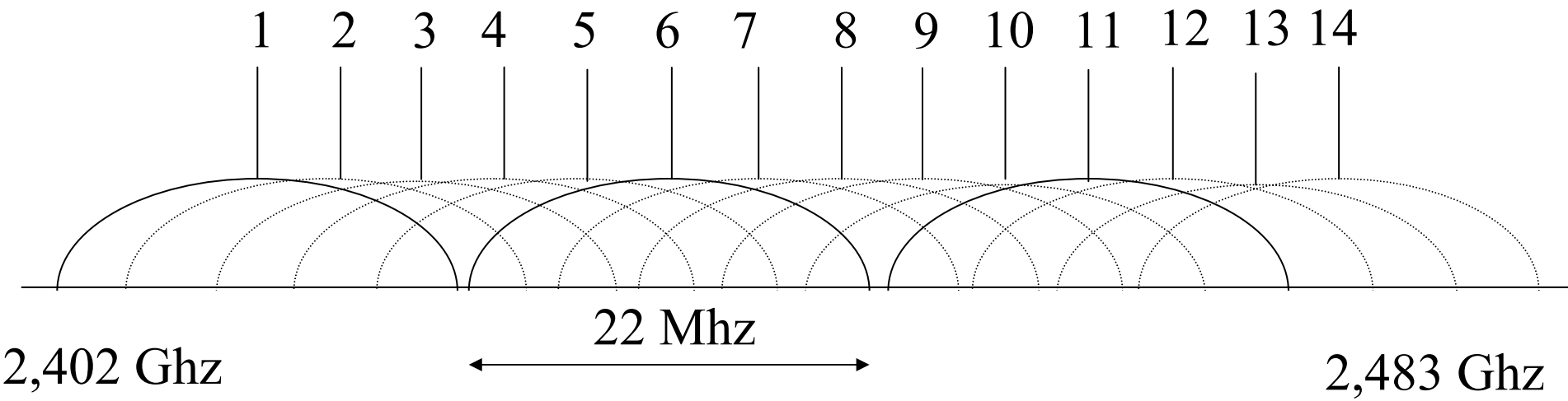


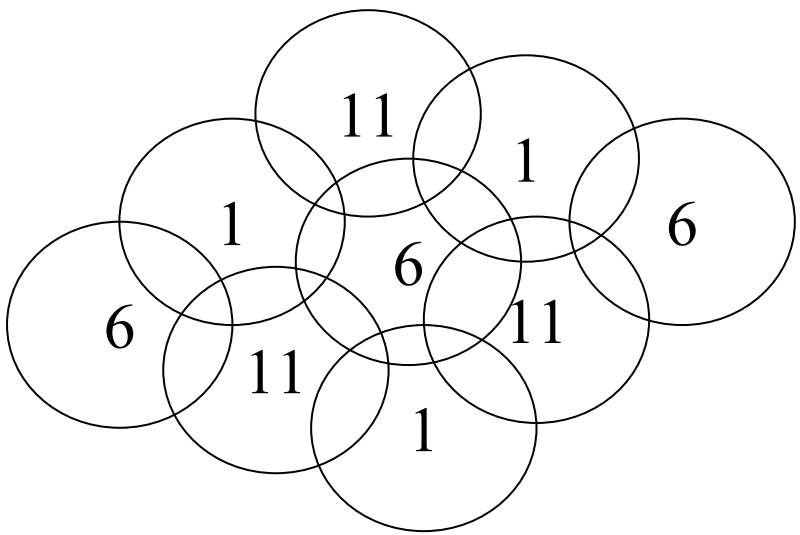
- **Préambule ‘ Preamble ’** :

  Synch (128 bits): Séquence 0101 … (128 bits entête normal, 56 bits entête court).
  SFD (16 bits Start Frame Delimiter): Délimiteur début F3A0 ou 1111 0011 1100 0000
- **PLCP ‘ Physical Layer Convergence Protocol ‘ .**

  Signal (8 bits): débit en centaine Kb/s hexa 0A, 14, 37, 6E pour 1, 2, 5,5, 11 Mb/s.
  Service (8 bits) : à 0.
  Longueur (16 bits): de la trame en octets (pour déterminer la fin).
  Header Error Check Field : CRC (16 bits) sur l’entête PLCP selon
  G(X)= X**16 + X**12 + X**5 + 1
- **Trame de niveau MAC avec code polynomial FCS**.

WIFI 802.11b : Le découpage en canaux de la bande 2,4 Ghz





- Trois canaux disjoints dans la bande 2,4 Ghz
- Une utilisation des trois canaux en mode cellulaire.

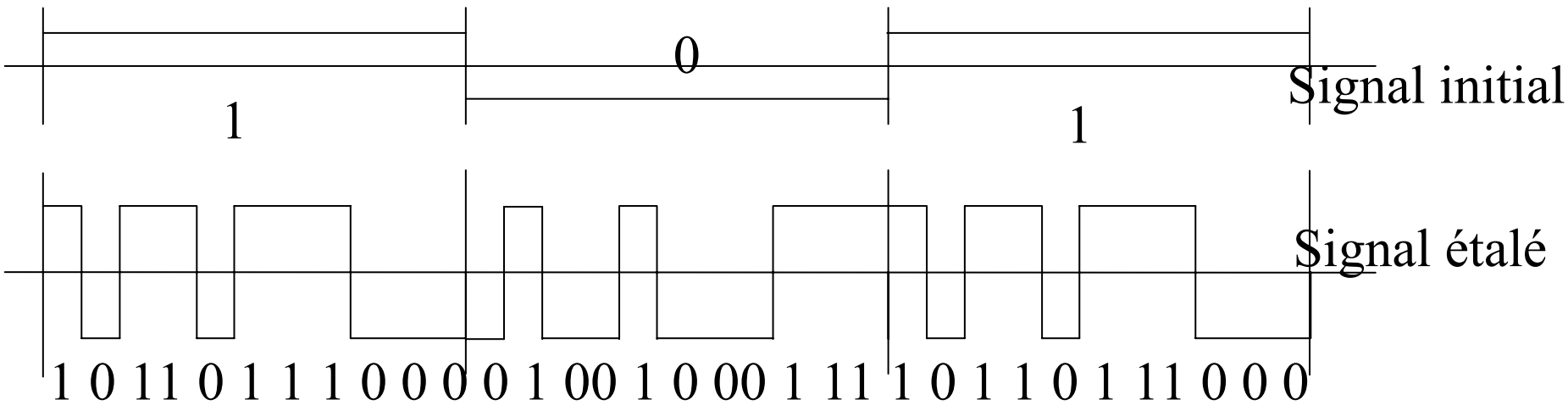
Niveau physique 802.11b : Codages et modulations

Débit en b/s	Nb bits codés par symbole	Longueur du symbole	Débit en symboles /s	Modulation
1 Mb/s	1 bit	11 bits code Barker	1 Méga symboles /s	DBPSK
2 Mb/s	2 bits	11 bits code Barker	1 Méga symboles /s	DQPSK
5,5 Mb/s	4 bits	8 signaux code CCK5,5	1,375 Méga symboles /s	QPSK
11 Mb/s	8 bits	8 signaux code CCK11	1,375 Méga symboles /s	QPSK

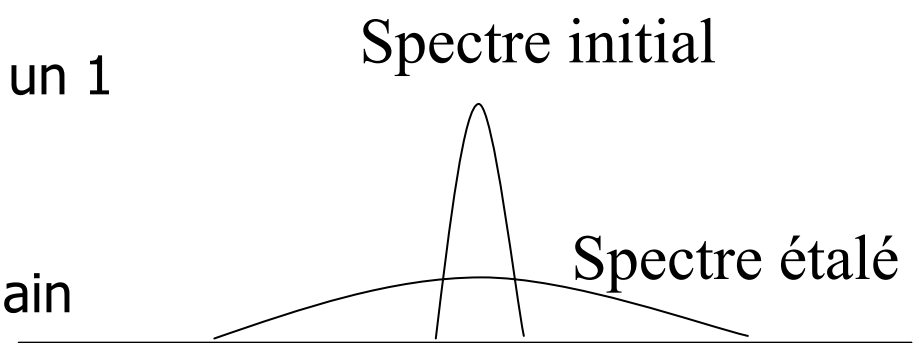
- **DBPSK :** Differential Binary Phase Shift Keying **DQPSK :** Differential Quadrature Phase Shift Keying **QPSK** Quadrature Phase Shift Keying
- **CCK** : Complementary Code Keying (Optionnel **PBCC** Packet Binary Convolutional Coding)

Niveau physique 802.11b: Code DSSS Direct Sequence Spread Spectrum

Etalement de spectre à séquence directe

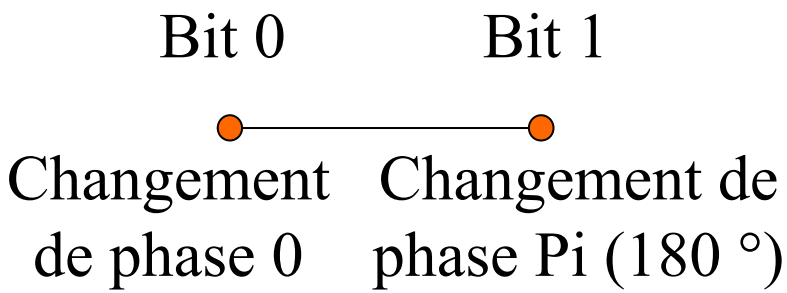


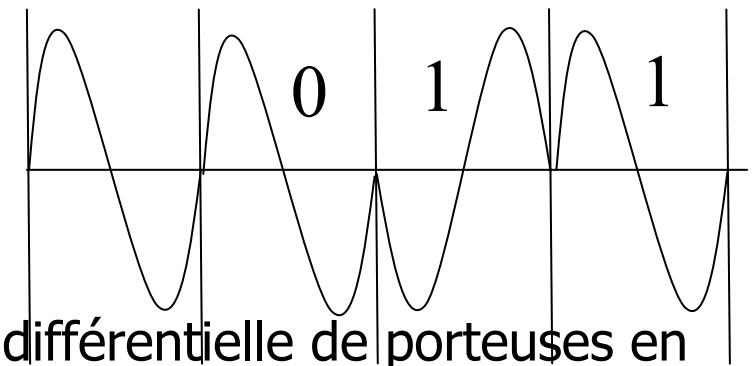
- **Choix d'une séquence** de 11 bits (séquence 'Barker') pour représenter un 1 (10110111000). Son complément représente un 0 (01001000111).
- **Introduction de redondances** permettant la correction d'erreurs / gain de 10 décibel.



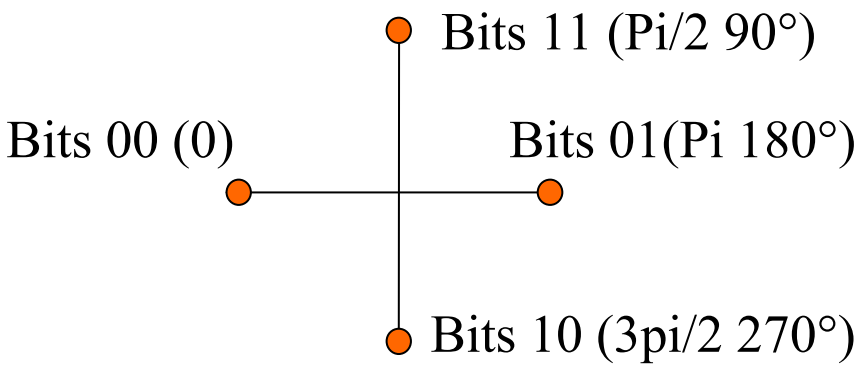
Niveau physique 802.11 b : Modulations utilisées en 802.11 b

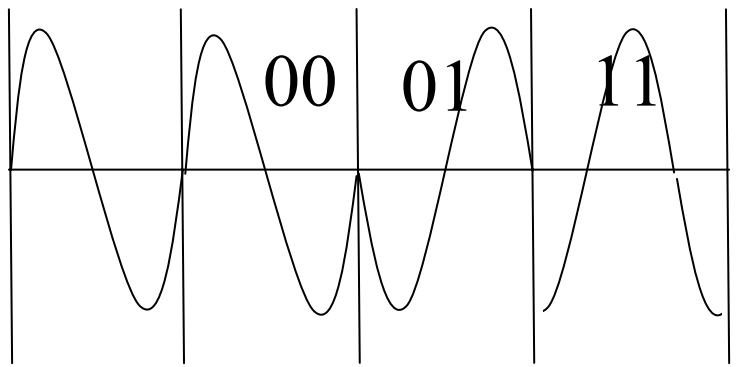
- **Débit 1 Mb/s** : **Modulation de phase différentielle** binaire (un bit par intervalle) (‘**DBPS**K **D**ifferential **B**inary **P**hase **S**hift **K**eying’).





- **Débit 2 Mb/s** : Modulation de phase différentielle de porteuses en quadrature (deux bits par intervalle). **DQPSK D**ifferential **Q**uadrature **P**hase **S**hit **K**eying.





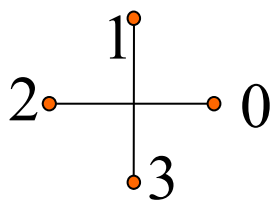
Niveau physique 802.11b : Code CCK 'Complementary Code Keying'

- **Exemple du code CCK 11 Mb/s** : on génère des symboles de 8 signaux DPSK (mot de code de 8 signaux pouvant avoir quatre phases différentes).

- **Première étape** : 8 bits (d1, d2, d3, d4, d5, d6, d7, d8) à transmettre génèrent 4 valeurs de phase Φ_1 , Φ_2 , Φ_3, Φ_4 égales aux valeurs des dibits, (d1, d2) (d3, d4) (d5, d6) (d7, d8). (les phases Φ_i sont codées 0, 1, 2, 3).

- **Seconde étape** : les quatre phases Φ_1 , Φ_2 , Φ_3, Φ_4 génèrent les 8 phases $(\Psi_1, \Psi_{2,\ldots,} \Psi_8)$ codées 0, 1, 2, 3 des huit signaux QPSK d'un symbole transmis (comme dans un code linéaire).

- $$(\Psi_1, \Psi_{2,\ldots,} \Psi_8) = (\Phi_1 , \Phi_2 , \Phi_3, \Phi_4) \begin{vmatrix} 1\,1\,1\,1\,1\,1\,1\,1 \\ 1\,0\,1\,0\,1\,0\,1\,0 \\ 1\,1\,0\,0\,1\,1\,0\,0 \\ 1\,1\,1\,1\,0\,0\,0\,0 \end{vmatrix} + (0\,0\,0\,2\,0\,0\,2\,0)$$



- $c = (e^{j\psi 1\pi/2}, e^{j\psi 2\pi/2}, e^{j\psi 3\pi/2}, e^{j\psi 4\pi/2}, e^{j\psi 5\pi/2}, e^{j\psi 6\pi/2}, e^{j\psi 7\pi/2}, e^{j\psi 8\pi/2})$

Réseaux locaux partagés wifi : Niveau Physique

Le niveau physique wifi selon la norme 802.11g

Niveau physique 802.11g : Principales caractéristiques

- **1) Héritage 802.11a** (2001) :
  - **Utilisation de la solution OFDM** (‘**O**rthogonal **F**requency **D**ivision **M**ultiplexing’) avec les mêmes modulations (DBPSK, QPSK et QAM).
  - Débits repris en 802.11g **DSSS-OFDM**: 6, 9, 12, 18, 24, 36, 48, 54 Mb/s

- **2) Compatibilité 802.11b** (1999) :
  - **Utilisation de la bande ISM** 2,4 à 2,4835 Ghz.
  - **Utilisation des standards 802.11b ERP-DSSS**: 1, 2 **ERP-CCK**: 5.5, 11 Mb/s

  (ERP Layers ‘Extended Rate Physical layers’ ensembles de codages/modulations).

- **3) Codage/modulation optionnels:** CCK/OFDM PBCC/DPSK **ERP-OFDM**: 6, 9, 12, 18, 24, 36, 48, 54 et **ERP-PBCC**: 5.5, 11, 22, 33Mb/s

- **4) 802.11g (2003): Adaptation du débit** aux conditions de transmission

- **5) 802.11g:** distances de 40m à 140 m pour des débits de 6 à 54 Mb/s

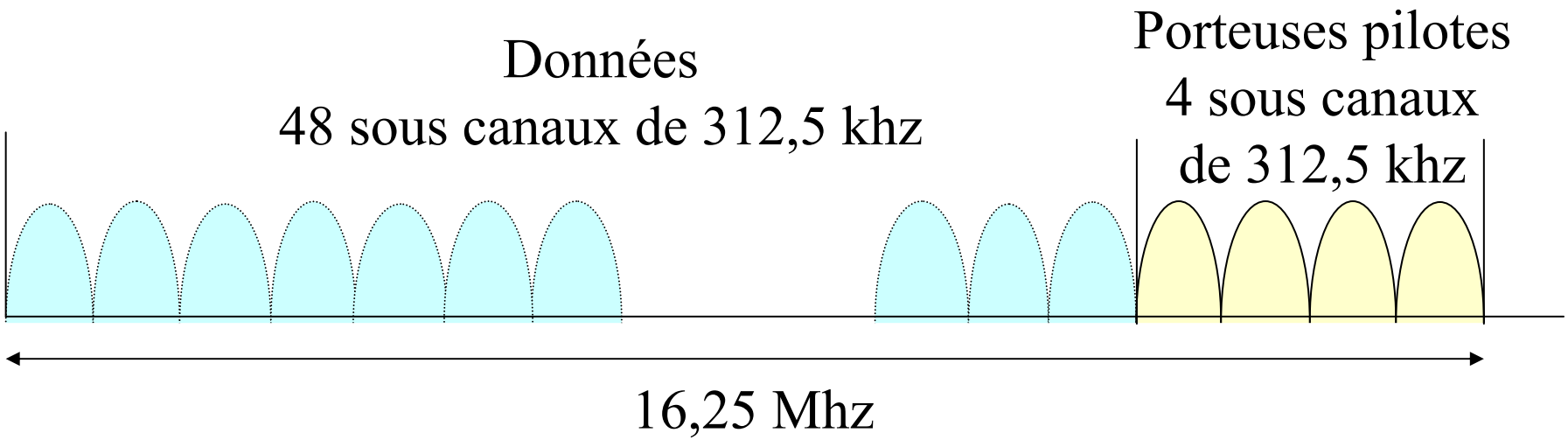
Niveau physique 802.11g : OFDM
Orthogonal Frequency Division Multiplexing

- **1) Technique majeure dans les transmissions numériques (années 1960)**
- **2) Applications multiples:** Radio numérique (DAB Digital Audio Broadcasting), Télévision numérique hertzienne DVB-T (Digital Video Broadcasting - Terrestrial, Accès aux réseaux numériques par le téléphone filaire ADSL (Asymmetric Digital Subscriber Line).
- **3) FDM Multiplexage fréquentiel : utilisation d'un découpage en sous canaux => OFDM = modulation à porteuses multiples**
  - Modulation dans chaque sous canal minimisant les interférences entre canaux.
  - Solution au problème des trajets multiples dus aux réflexions: canaux étroits et intervalles de garde (91 ns avec écho 500ns ou 3200 ns + garde de 800 ns).
- **4) O Orthogonalité: Espacer très régulièrement les canaux**
  - Les porteuses forment un ensemble orthogonal qu'il est facile de séparer

$$\psi_k(t) = e^{jk\omega t}; \int_T \psi_k(t)\psi_l^*(t)dt = 0 \text{ si } k \neq l; = T \text{ si } k = l$$

- **5) Utilisation de circuits spécialisés de transformée de Fourrier** (pour la modulation dans les sous canaux).
- **6) C pour Codage** (COFDM): Entrelacement des bits, Réservation de sous canaux pour transmettre des informations de redondances => correction d'erreurs.

WIFI 802.11g : Le découpage d'un canal OFDM



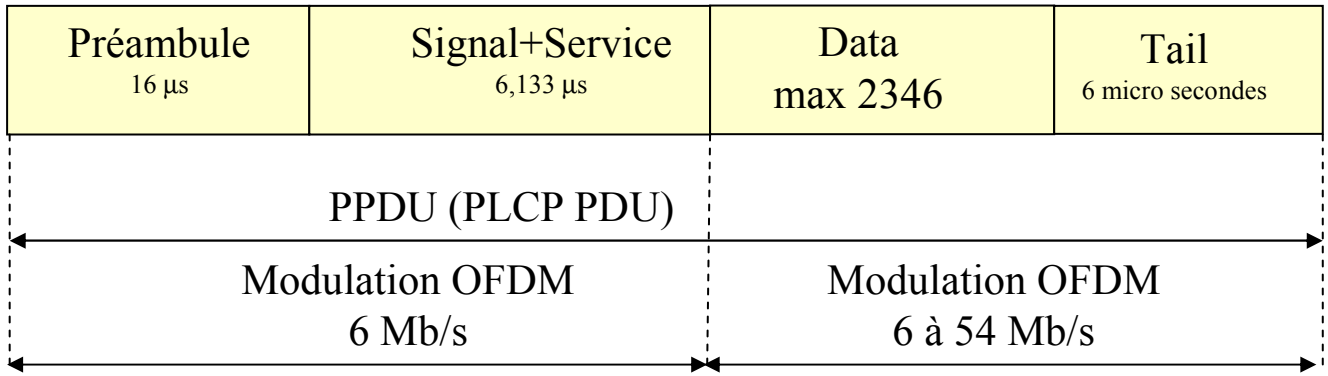
- **Canaux de données (et de redondances):** 48
- **Données codées à l'émission** au moyen d'un code correcteur d'erreurs. Selon les débits ½ ou 1/3 ou ¼ des canaux sont utilisés pour la redondance de données.
- **Sous canaux servant à la transmission de porteuses 'pilotes':** 4
  - On y émet des séquences de données fixes.
  - Utilisées pour évaluer les délais de propagation et les interférences de symboles.

Niveau physique 802.11g : Codages et modulations en OFDM

Débit binaire	Débit par sous canal	Bits codés par symb	Débit en M symboles /s	Code (FEC)	Modulation
6 Mb/s	0,125 Mb/s	1 bit	12 Ms/s	24/48	BPSK
9 Mb/s	0,1875 Mb/s	1 bit	12 Ms/s	36/48	BPSK
12 Mb/s	0,25 Mb/s	2 bits	24 Ms/s	48/96	QPSK
18 Mb/s	0,375 Mb/s	2 bits	24 Ms/s	72/96	QPSK
24 Mb/s	0,5 Mb/s	4 bits	48 Ms/s	96/192	16-QAM
36 Mb/s	0,75 Mb/s	4 bits	48 Ms/s	144/192	16-QAM
48 Mb/s	1 Mb/s	6 bits	72 Ms/s	192/288	64-QAM
54 Mb/s	1,125 Mb/s	6 bits	72 Ms/s	216/288	64-QAM

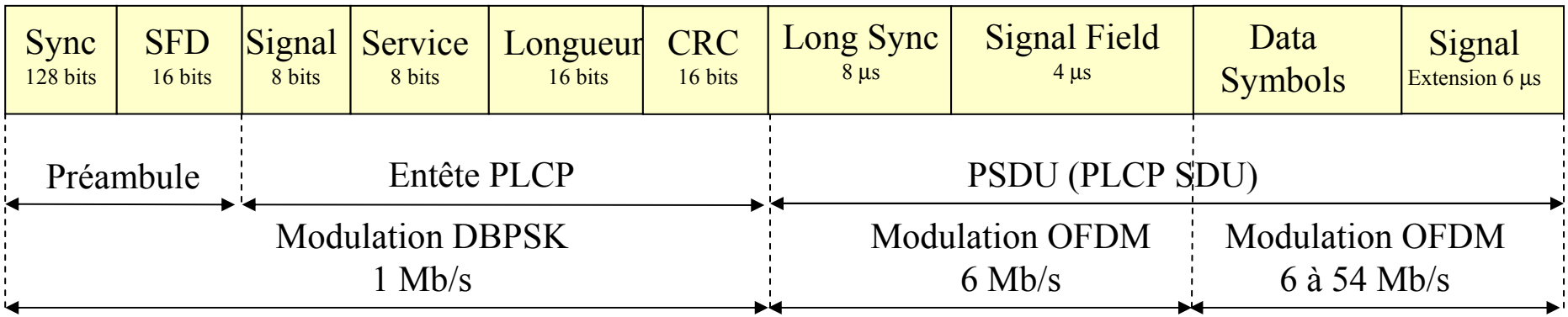
FEC : Forward Error Correction **QAM :** Quadrature Amplitude Modulation

WIFI 802.11g : Le format de trame DSSS-OFDM (pur 802.11g)



- **PPDU (**PLCP PDU): Une unité de protocole de niveau physique.
- **L'entête PLCP** est transmise à 6 Mb/s.
- **Préambule:** Synchronisation pour une transmission OFDM.
- **Zone signal** (24 bits): le débit de transmission utilisé pour la zone données et sa longueur (un symbole de 4 µs à 6 Mb/s). La zone service est de 2/3 de symbole soient 16 bits (inutilisée).
- **La zone de données** contient une trame de niveau MAC d'une longueur maximum de 2346 octets.
- A la fin **TAIL** est une zone de silence de 6 micro secondes.

WIFI 802.11g : Le format mixte d'une trame compatible 802.11b



- **PSDU (**PLCP SDU): la charge utile d'une trame au niveau physique après le préambule et la zone PLCP comme en 802.11b à 1 Mb/s (la partie signal de la zone PLCP code par un débit à 3 Mb/s tous les débits OFDM).
- Les deux premières zones sont toujours émises à 6 Mb/s:
  - **Long Sync (Long Training Sequence)** est une séquence de synchronisation de deux fois 4 micro seconde plus un intervalle de garde.
  - **OFDM Signal** définit le débit et la longueur pour la partie OFDM Data Symbols.
- **Data symbols**: la partie MAC habituelle suivie de 6 µs de silence (signal extension).

Liste des autres standards 802.11

- **802.11c :** Protocole de réalisation des ponts 802.11
- **802.11d :** Harmonisation internationale des réseaux 802.11.
- **802.11e :** Qualité de service en wifi (transmission de la voix et de la vidéo).
- **802.11f :** Itinérance entre AP (‘roaming’).
- **802.11h :** Gestion de la bande 5 Ghz en Europe réglementation fréquences, relation avec HiperLAN2 .
- **802.11i :** Amélioration de la sécurité.
- **802.11j :** Adaptation à la réglementation japonaise.

WIFI : Conclusion

- **Avantages**

- **Supprime les câblages** (construction ' ad-hoc ').
- **Débit acceptable** pour un grand nombre d'applications.

- **Inconvénients**

- **Surcharges** protocolaires (11 Mb/s => 6,38 Mb/s réels).
- **Problèmes des transmissions** hertziennes.
  - Distances assez faibles, Interférences
- **Problèmes de sécurité**
- Mise en œuvre de **l'itinérance entre cellules** (Roaming)
- **Qualité de service** (téléphone sur wifi).

Bibliographie Wifi

- Documents web
- Normes relatives au Wifi. http://www.ieee.org/
- Davor Males, Guy Pujolle 'Wifi par la pratique' , Eyrolles, 2002
- Mustafa Ergen, 'IEEE 802.11 Tutorial', Université de Californie à Berkeley, Juin 2002

Réseaux locaux

Réseaux locaux commutés

"Lan Switching"

1 Notions générales.

2 Techniques de commutation.

3 Techniques de routage.

4 Réseaux locaux virtuels.

Commutation de réseaux locaux

1
Notions générales

Besoin de réseaux locaux commutés

Nombreuses limitations du mode Ethernet partagé (Ethernet unidirectionnel ' Half duplex ')

- **Performances en débit** : Pour des grands réseaux (milliers de stations) et de gros volumes de données (accès fichier, multimédia).
- **Difficultés d'extension en distance:** problème des collisions.
- **Sécurité :** tout message est accessible par toutes les stations.

- **Problème essentiel du mode partagé:** mode diffusion générale de tous les messages (mode ' promiscuous '), tous les tronçons sont reliés par des répéteurs donc toutes les stations voient passer toutes les trames point à point, toutes les diffusions, toutes les collisions, toutes les erreurs …

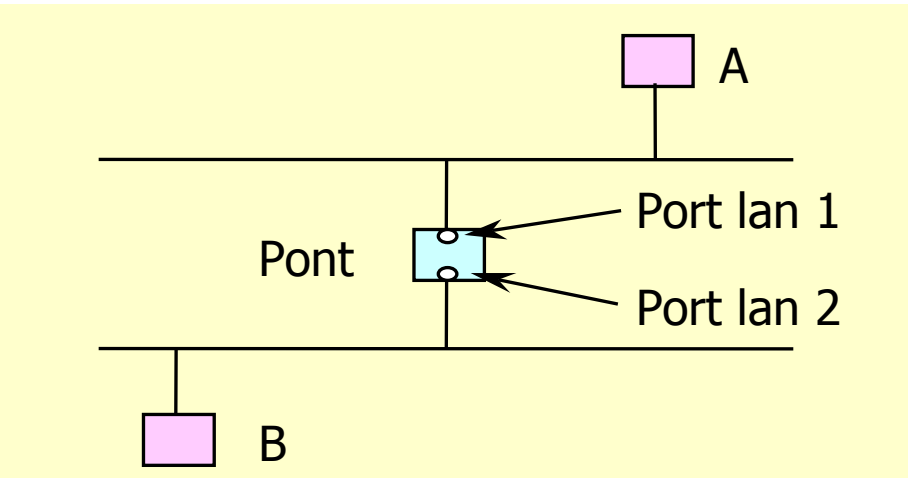
La dégradation des performances et de la sécurité a conduit progressivement à une inadaptation de l'architecture partagée.

Évolution historique (1) : répéteurs, ponts

Dispositifs pour améliorer les réseaux Ethernet

- **Répéteur ou ' hub '** (pour mémoire) : **recopie de niveau physique**, bit à bit des informations de tronçon à tronçon (propre au mode partagé).
- **Pont (' bridge ')** : historiquement **un matériel de connexion entre deux réseaux locaux** agissant en recopiant toutes les trames d'un tronçon sur l'autre (agit au niveau trame par exemple en stockage et retransmission)

  => **Ne propage pas les erreurs ou les collisions** : limitation des domaines de collision à chaque réseau local).

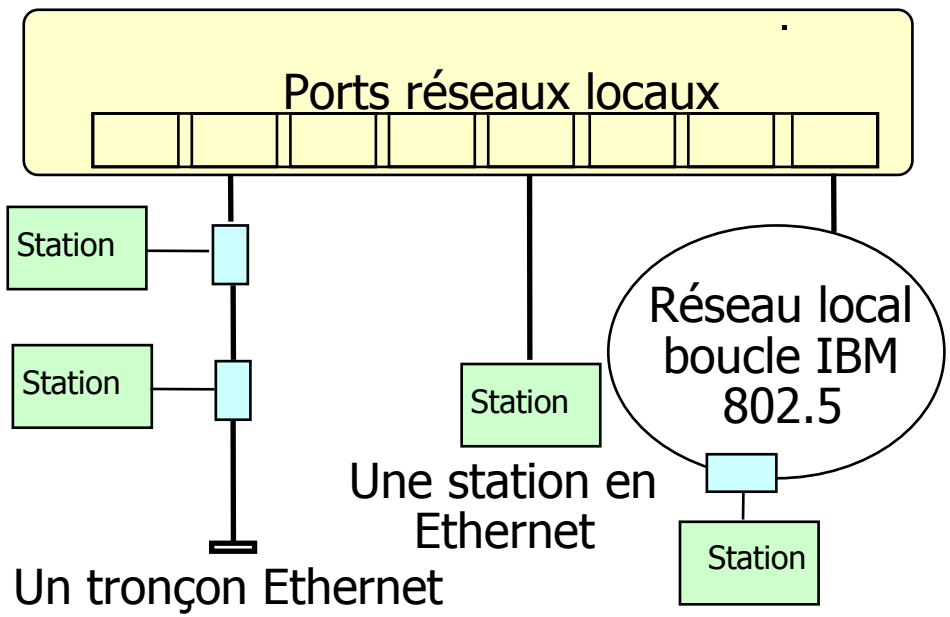


Évolution historique (2) : ponts filtrants, commutateurs

- **Ponts filtrants ('filtering bridges')** : Historiquement la première version de la commutation des réseaux locaux: un pont filtrant agit entre deux tronçons et ne laisse passer que le trafic devant transiter d'un tronçon à l'autre ('forwarding')

  => Les diffusions s'étendent encore à tous les tronçons.

- **Commutateur de réseau local ('lan switch')** : Un dispositif capable de commuter un grand nombre de tronçons de réseaux locaux (8, 16 , 32...) de standards éventuellement différents.



Commutateur de réseaux locaux : architecture en couches, routage

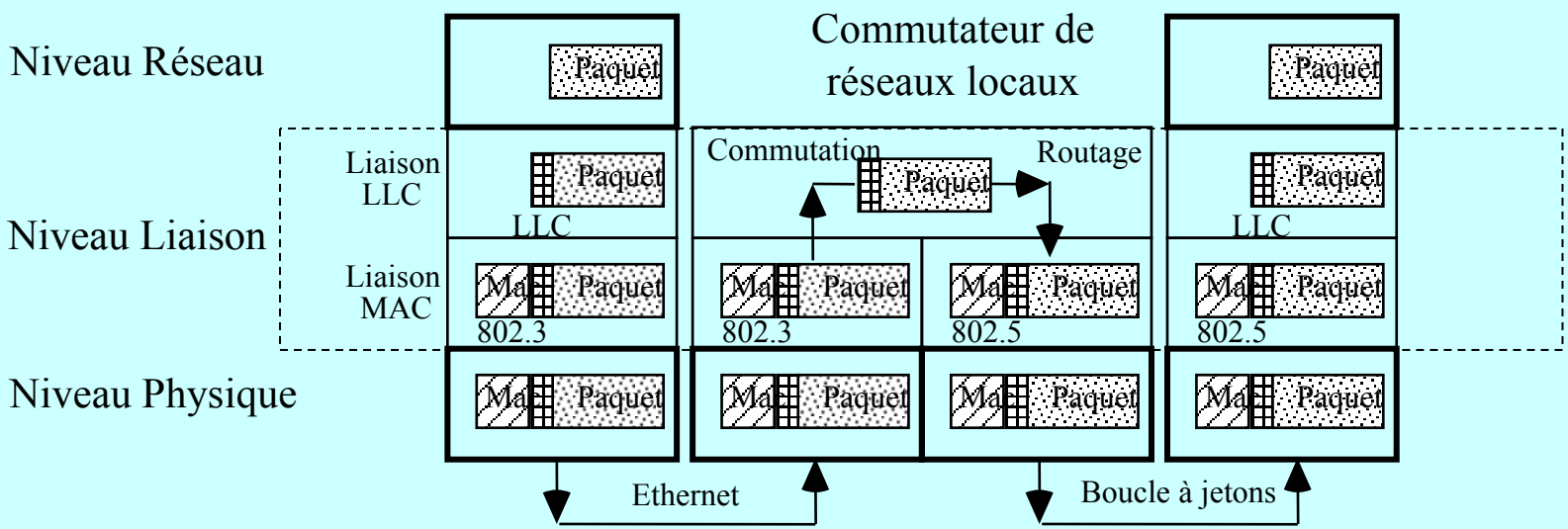


Table de routage

Adresse	Port	Date
42:AB:E3:10:98:33	3	19:27
5C:72:C2:C3:51:01	5	19:29

Commutateurs de réseaux locaux: Le fonctionnement ‘ transparent ’

- **Transparence (‘Transparent Bridges/Switches’)**

On branche physiquement une configuration de commutateurs et de stations quelconque et l’ensemble doit s'autoconfigurer sans aucune intervention humaine (fonctionnement ‘ plug and play ’).

- **Apprentissage ‘a posteriori’ (‘Backward Learning ’) :**

A chaque arrivée de trame sur un port, les commutateurs notent dans la table de routage l’adresse d’émission avec son port.

- A l'initialisation **les tables de routage sont vides.**

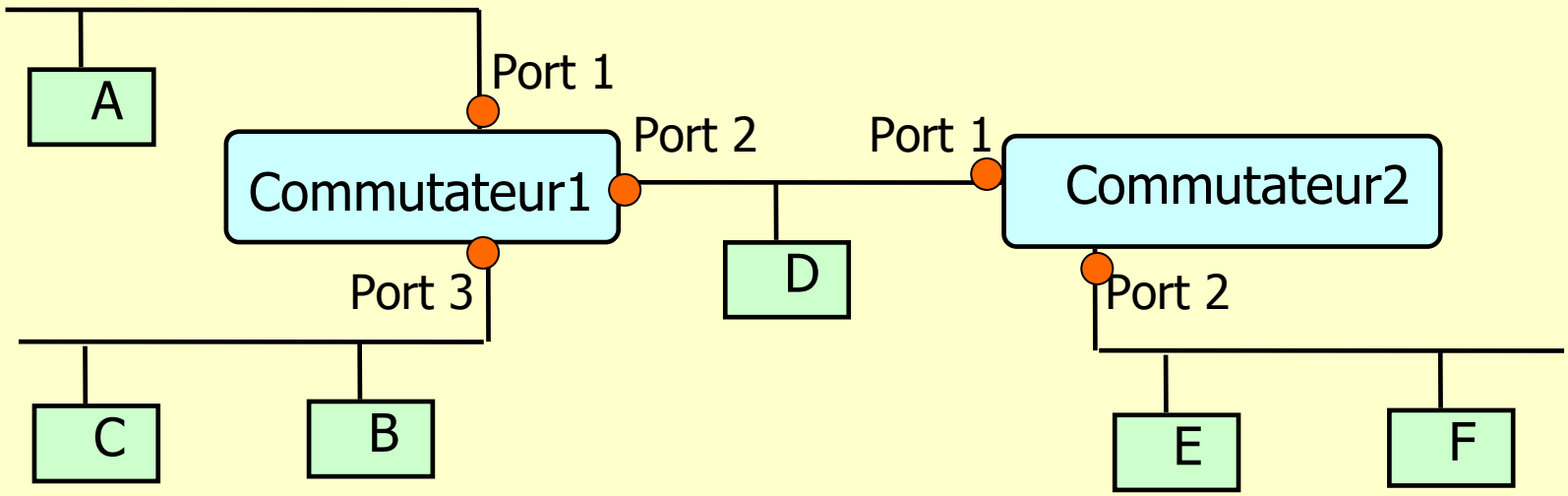
- Toutes les trames circulant sur les tronçons reliés à un commutateur sont **écoutées.**

- Les **adresses sources et les ports** sont notés dans la table.

- Si la destination d’une trame à relayer n'est pas connue, la trame est diffusée **sur tous les ports pour atteindre son destinataire**.

- Si une destination est connue la trame est **recopiée sur le seul port** de l’adresse destination sauf si c'est le même port que celui de la source.

Commutateurs de réseaux locaux: Exemple de l'apprentissage



- Trame C vers D : le commutateur 1 diffuse sur les ports 1, 2 et apprend que la station C est du coté du port 3. La trame est délivrée à D.
- Trame B vers C : le commutateur 1 ne retransmet pas la trame et apprend que la station B est du coté du port 3. La trame est délivrée à C.
- Trame F vers A : les commutateurs 1 et 2 reçoivent et diffusent. Le commutateur 2 apprend que F est du coté du port 2 et le commutateur 1 que F est du coté du port 2. La trame est délivrée à A.

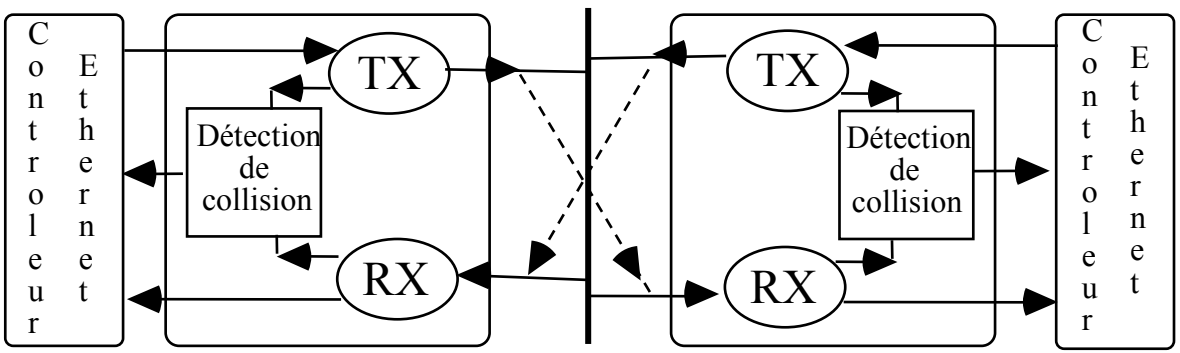
Commutateurs de réseaux locaux: un peu de terminologie

Selon le point de vue par lequel on considère le dispositif ou selon ses fonctions.

- Commutateur de réseaux locaux **"Lan switch"**
- Pont (par analogie des fonctions) **"Bridge"**
- Commutateur de niveau 2 **"Layer 2 switch"**
- Commutateur multi-niveaux (2 niveaux supportés, LAN et IP ou LAN et ATM) **"Multi layer switch"**
- Proxy LAN emulation client**" Proxy LEC "** ou Dispositif de périphérie (d'un réseau ATM) **" Edge device "**

Mode Ethernet Bidirectionnel (1) (‘ Full Duplex ’)

- **Rappel Mode Ethernet partagé :** protocole à l'alternat "half duplex" soit on émet soit on reçoit. En 10 BAS T connexion avec deux paires. Paire émission TX + paire réception RX. RX sert en écoute de collision si TX émet) sinon RX reçoit seule.

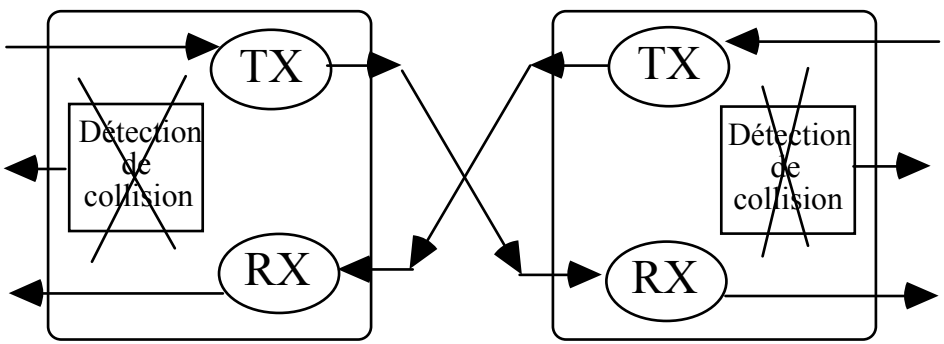


- Avec l’utilisation d'un commutateur Ethernet et avec une seule station par port : pas de collisions

  => Possibilité de **suppression de la gestion des collisions.**

Mode Ethernet Bidirectionnel (2) (‘ Full Duplex ’)

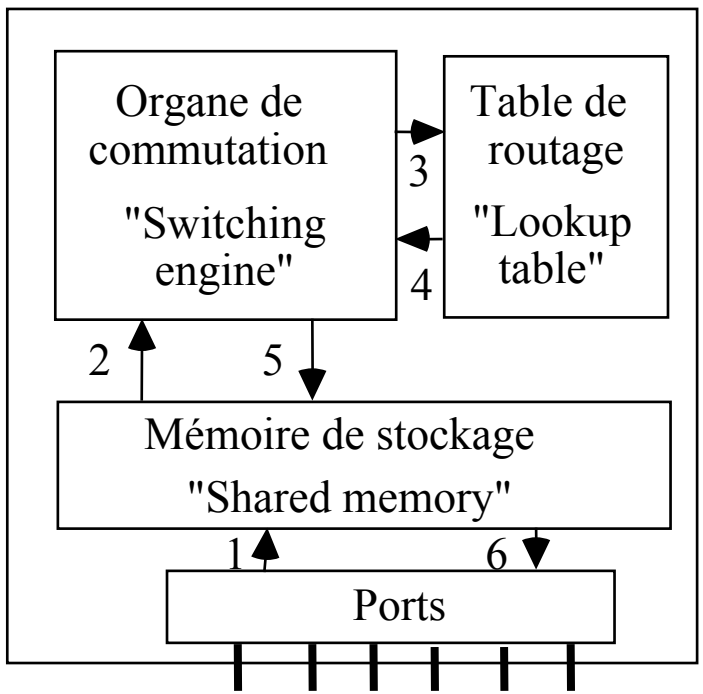
- **Modification de l'interface ethernet pour utiliser les deux paires simultanément en transmission**
- On connecte **directement TX sur RX en mode bidirectionnel.**
- Reste le **format** des trames et les techniques de communication **physique.**
- Notion de carte et de port **Ethernet "full duplex"** avec débit double de 2 fois 10 Mb/s ou 100 Mb/s ou 1 Gb/s (10 Mb/s émission 10 Mb/s réception).



2
Techniques de commutation

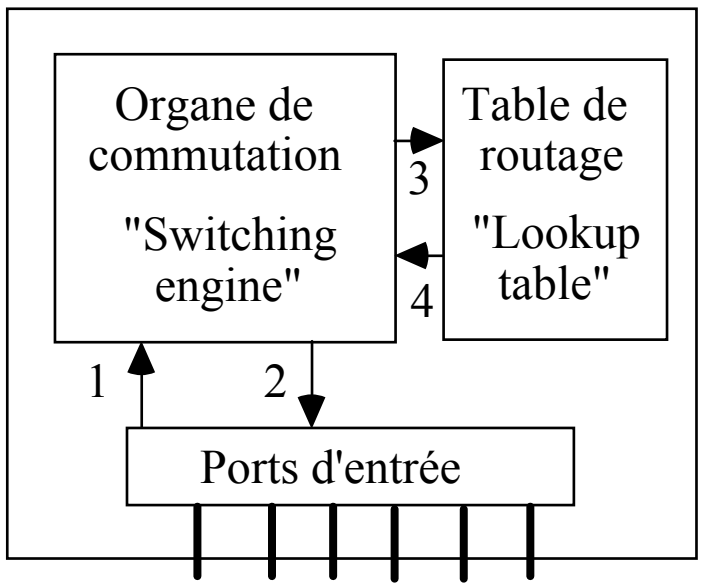
Commutation en mémoire partagée (en stockage et retransmission)

- Terminologie anglaise: ` **shared memory** ’
` **store and forward** ’
- La trame entrante est **entièrement reçue** et **stockée dans la mémoire.**
- 2 Le module de commutation **extrait l'adresse de destination**.
- 3 L'adresse est **envoyée à la table de routage**.
- 4 Le **résultat** de la recherche est **retourné**.
- 5 L'adresse du port sortie est **propagée**
- 6 La trame est renvoyée à partir de la mémoire **sur le port de sortie approprié**.
- **Problème:** le retard de commutation.



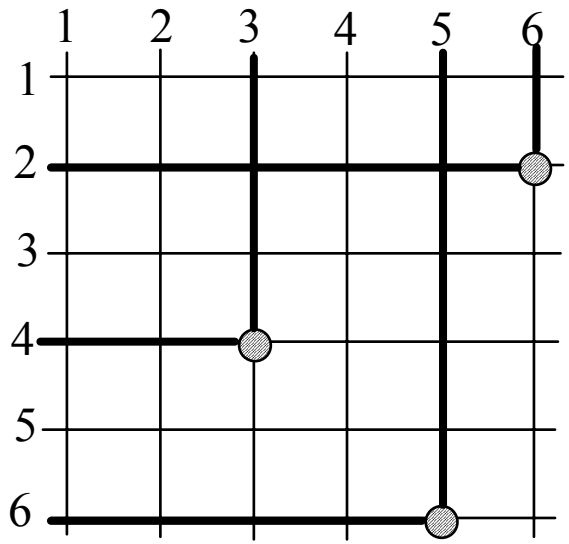
Commutation à la volée

- Terminologie anglaise: **'Cut through', 'On the fly', 'fragment free', 'fast forward'.**
- 1 L'organe de commutation **lit le début de trame et** extrait **l'adresse** destination. La trame **continue d'arriver.**
- 2 L'adresse est **envoyée à la table de routage.**
- 3 Le résultat de la recherche est **retourné.**
- 4 Le module de commutation **renvoie la trame sur le port de sortie dès que possible** (de préférence avant qu'elle ne soit entrée totalement).
- **Problèmes** : tests de correction de la trame et conflits d'accès aux ports de sortie.



Techniques pour la commutation à la volée : commutation spatiale

- Rappel commutation spatiale (‘ cross bar ’, ‘ matrix ’)
- Pour N ports de communication (N petit), utilisation d'une matrice NxN de points de connexions (Matrice "Cross point")
- Utilisation d'aiguillages bâtis autour de circuits intégrés assemblés en matrice.
- Retard de commutation très faible et commutation en parallèle possible.



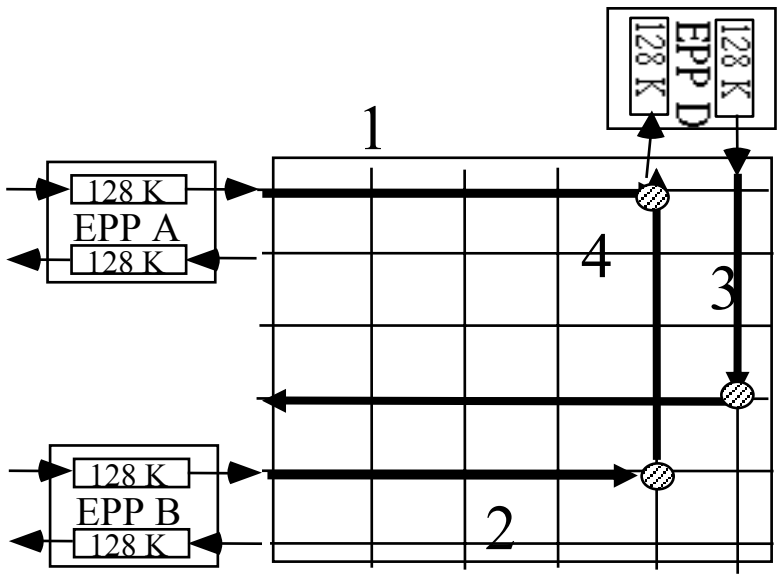
Commutation spatiale: Gestion des conflits d'accès par files d'attente.

- Exemple : Kalpana EPS 1500 2-15 ports 10BAS2, 10BAST, 10BASFL, AUI, 1700 adresses par port, 6000 adresses par commutateur.
- "EPP Ethernet Packet Processor" Processeur de ports d'entrée/sortie avec gestion de tampon (256 trames 1518 octets)
- (1) Port A: trame à destination de D

- Port D occupé en sortie par B (2)
- EPP A stocke sa trame en entrée

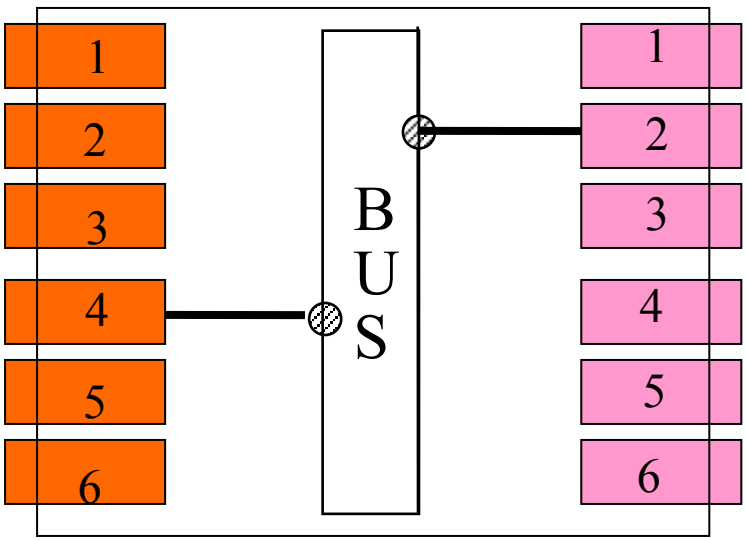
- (1) Port A: trame à destination de D

- Autre cas (3): le Port D est occupé en entrée (Ethernet half duplex impossible de sortir)
- Le port D stocke une trame commutée dans ses tampons de sortie (4).



Architecture de commutateurs à bus

- Pour N ports de communication, utilisation d'un **bus haut débit** qui assure la **fonction de commutation**.
- Nécessité de **tampons pour traiter les conflits d'accès.**



- Gestion des **adresses et des files d'attente** au niveau des cartes de gestion de port.

Techniques pour la commutation à la volée: commutateurs de cellules à bus

Architecture de commutateur type ATM

- Utilisation d'une **architecture** existante à commutation de cellule (en anglais ' **cell backplane switch '**).
- Découpage d'une trame en **cellules courtes de taille fixe** (cellule ATM 53 octets dont 48 de charge utile)**.**
- Chaque cellule est étiquetée avec un **entête qui définit le port de sortie.**
- Les cellules sont commutées puis **stockées dans les tampons du port de destination.**
- La trame est **réassemblée** à partir de ses cellules **et transmise.**

Comparaison des deux types d'architectures

- En stockage et retransmission

  - On examine **la trame en entier**.
  - On peut **détecter tous les cas de trames erronées.**
  - **Seules** les trames **correctes sont relayées.**
  - Le temps de commutation est plus **important.**

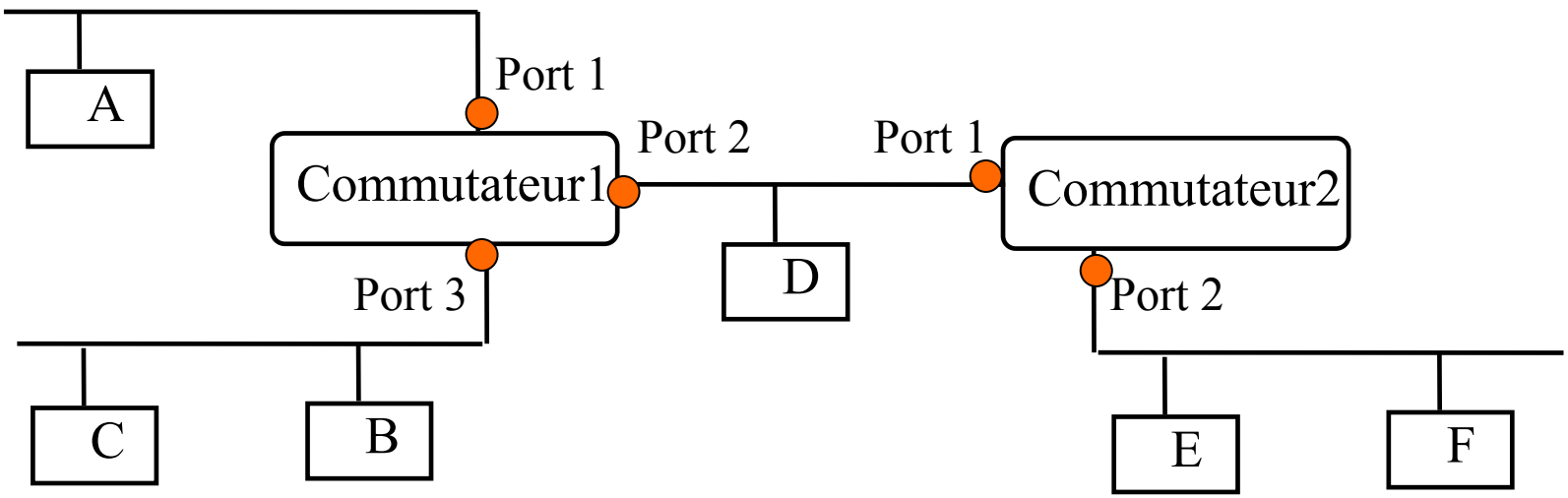
- En commutation à la volée

  - Seuls les **premiers octets sont stockés.**
  - La trame est passée au destinataire **sans examiner la fin.**
  - Certaines trames en **erreur peuvent être relayées.**
  - Le **retard de commutation** d'une trame est **plus faible**.

3
Techniques de routage

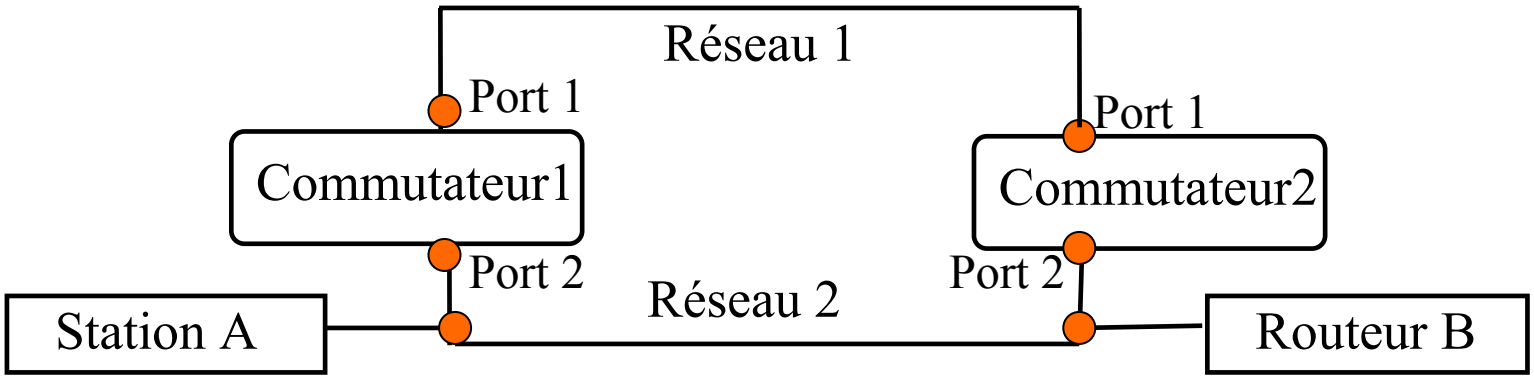
Position du problème de routage

- En présence d'un ensemble de commutateurs de réseaux locaux interconnectés: **comment trouver le chemin qui permet d'aller d'un point à un autre.**
- Construction pour chaque commutateur **d'une table de routage (' forwarding data base ')**



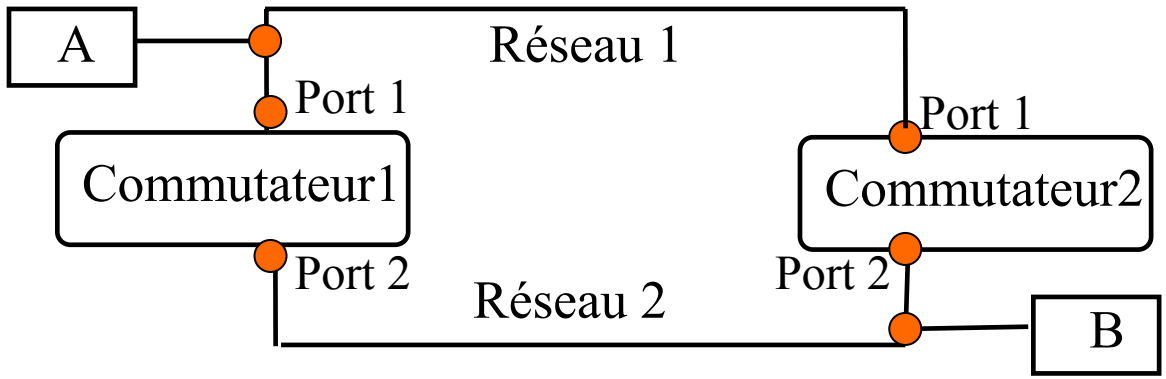
- Rappel: volonté **d'auto configuration automatique** du routage au moyen de techniques d'apprentissage

Problème de l'existence de deux ponts en parallèle: la tempête de diffusions



- Trame en diffusion générale adresse FF:FF:FF:FF:FF (par exemple protocole ARP, B routeur veut apprendre l'adresse de A) : A reçoit mais les deux commutateurs 1 et 2 reçoivent aussi par leur port 2 cette diffusion et selon la règle **répercutent** la trame sur le réseau 1 par leur port 1.
- Le commutateur 1 reçoit alors une diffusion du commutateur 2 sur son port 1 et doit donc **replacer** cette trame sur le réseau 2 par son port 2 (il en est de même pour le commutateur 2).
- Les trames en diffusion vont donc circuler indéfiniment entre les commutateurs (notion de tempête de diffusion ' broadcast storm ')

Problème de l'existence de deux ponts en parallèle: l'instabilité des tables



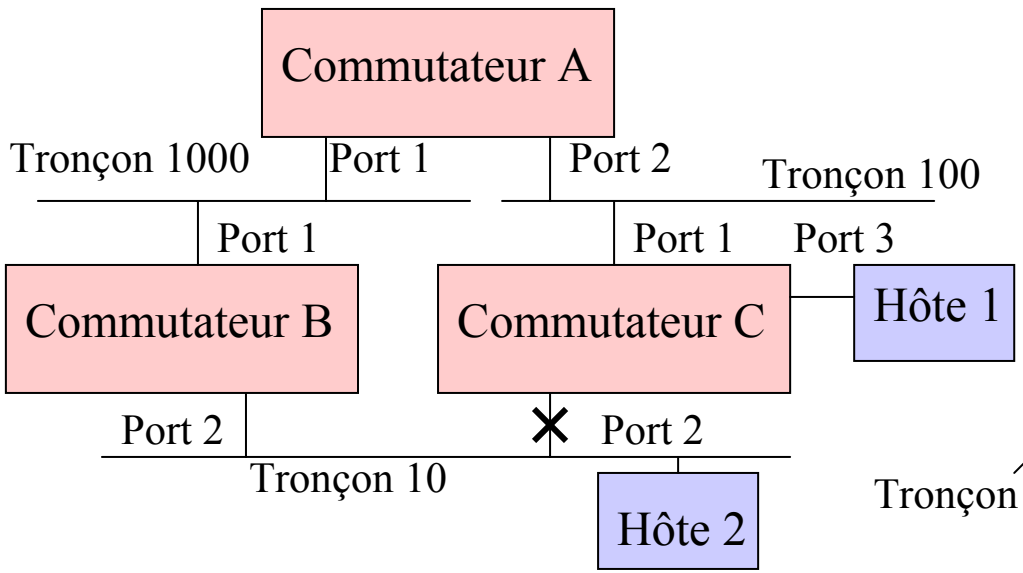
- Trame de A vers B: le commutateur 1 apprend que **la station A est accessible par son port 1.**
- Mais les deux commutateurs 1 et 2 reçoivent par leur port 1 et **répercutent** la trame sur le réseau 2 par leur port 2. B reçoit.
- Le commutateur 1 reçoit une seconde fois la trame sur son port 2 et donc **considère que la station A est accessible par son port 2 ce qui est faux.** Le commutateur 2 fait la même erreur.
- Les communications s'enchaînent et les tables de routage **oscillent en permanence.**

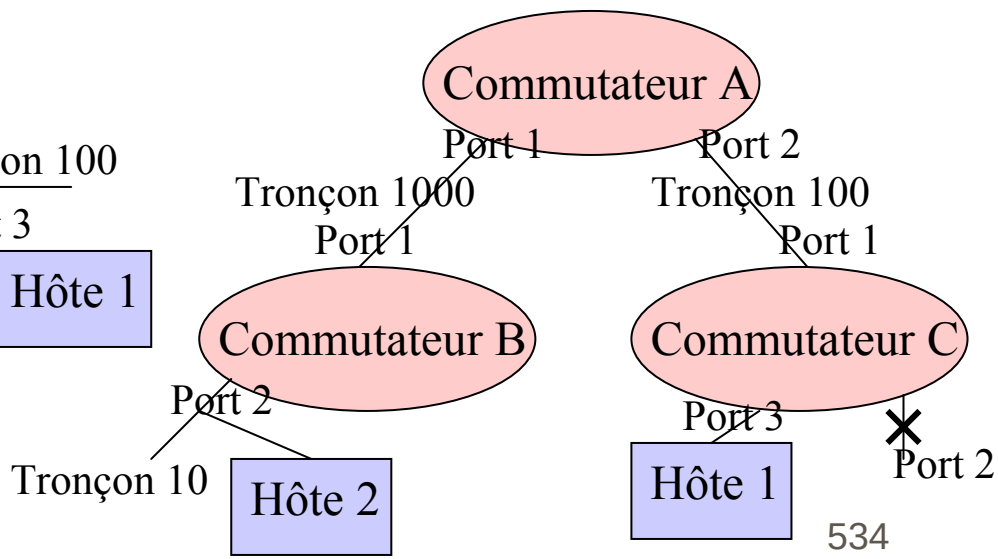
Le routage par arbre couvrant : principes généraux (1)

- **La norme 802.1 D** définit le protocole de routage par arbre couvrant : **STP‘ Spanning tree protocol ’.**
- La solution la plus immédiate au problème des circuits : **n'avoir qu'un seul chemin** pour aller d'un point à un autre au moyen de commutateurs de réseaux locaux.
- Soit le graphe dont les sommets sont les commutateurs et les arcs sont les tronçons de réseaux locaux: pour n'avoir qu'un seul chemin d'un point à un autre du graphe on utilise un **arbre couvrant.**
- **L'arbre couvrant est construit automatiquement** par dialogue entre les commutateurs: utilisation de trames Hello et BPDU ‘ Bridge Protocol Data Units ’ (Radia Perlman 1992).

Le routage par arbre couvrant : principes généraux (2)

- Principe de routage: en fonctionnement normal, pour aller d'un point à un autre on utilise l'arbre. **On va d'un commutateur à la racine puis de la racine à un autre commutateur** ce qui définit un chemin unique.
- Certains chemins sont **abandonnés.** Ils ne peuvent servir que sur panne et reconfiguration du réseau avec un autre arbre.





Étapes de construction de l'arbre couvrant : Étape 1

- **Élection du commutateur racine de l'arbre**

Le commutateur de plus faible priorité puis de plus faible adresse mac est élu par échange de messages.

Priorité : 0 à 32768 définie par configuration

Identifiant unique: adresse MAC unique de l'un des ports.

En réunissant les deux notions: **Identificateur de pont** (BID ' Bridge ID '): priorité (2 octets), adresse MAC (6 octets)

Exemple: priorité 32768 (8000 en hexadécimal), adresse MAC 00:A0:D6:13:43:65, identificateur du commutateur pour les comparaisons 8000:00A0:D613:4365.

Étapes de construction de l'arbre couvrant : Étape 2

Élection du port racine de chaque commutateur

- Pour chaque commutateur on doit déterminer **un port unique** qui le connecte à la racine: le port racine (' root port ').
- On détermine tous les **chemins** du commutateur vers la racine
- Pour chaque chemin on calcule **le coût** du chemin à partir d'un coût attribué à chaque lien (chaque tronçon de réseau local).
- On choisit **le port racine** comme celui de **plus court chemin** et en cas d'égalité de plus faible priorité (selon une priorité définie par configuration).

Étapes de construction de l'arbre couvrant : Étape 3

- **Élection du port désigné**

- Un **tronçon** est possiblement **connecté à la racine** par **plusieurs commutateurs** (plusieurs ports de commutateurs).
- On doit déterminer l'un de ces ports de commutateur comme **port ` désigné '.** C'est un port unique qui permettra au tronçon de communiquer avec la racine.
- On sélectionne comme **port désigné celui de chemin le plus court jusqu'à la racine** (puis celui dont l'adresse MAC est la plus petite).
- Les autres ports sont **bloqués.**

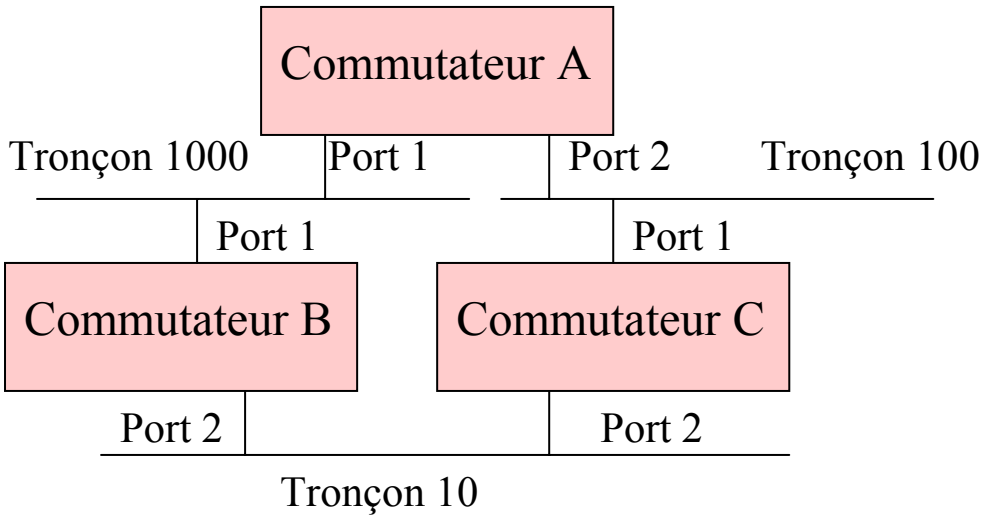
Approfondissements : coûts des différents types de réseaux locaux

Débit	Coût Recommandé	Intervalle de coût recommandé
4Mbps	250	100 à 1000
10Mbps	100	50 à 600
16Mbps	62	40 à 400
100Mbps	19	10 à 60
1Gbps	4	3 à 10
10Gbps	2	1 à 5

Approfondissements : états d'un port

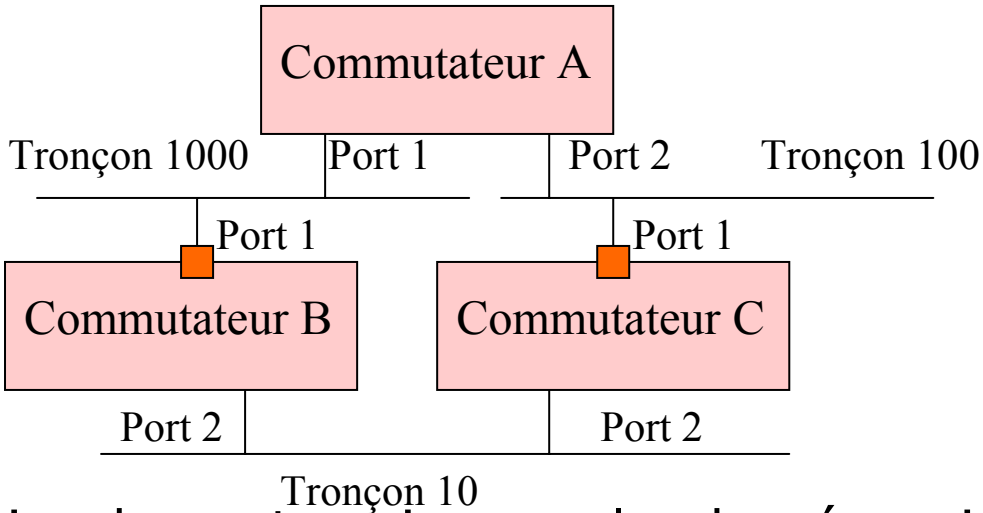
- Les ports participants à l'algorithme de l'arbre couvrant peuvent être dans cinq états:
  - **Listening - Écoute** : émet/reçoit des BPDU pour la construction de l'arbre mais ne relaye pas les trames.
  - **Learning - Apprentissage** : apprend l'existence d'adresses MAC mais ne relaye pas les trames.
  - **Blocking - Bloqué** : en écoute uniquement des trames de type BPDU, ne relaye pas les trames normales.
  - **Forwarding - Relais** : port qui émet et reçoit des trames de la racine.
  - **Disabled - Déconnecté** : port sans aucune activité.

Approfondissements : un exemple de fonctionnement Étape 1



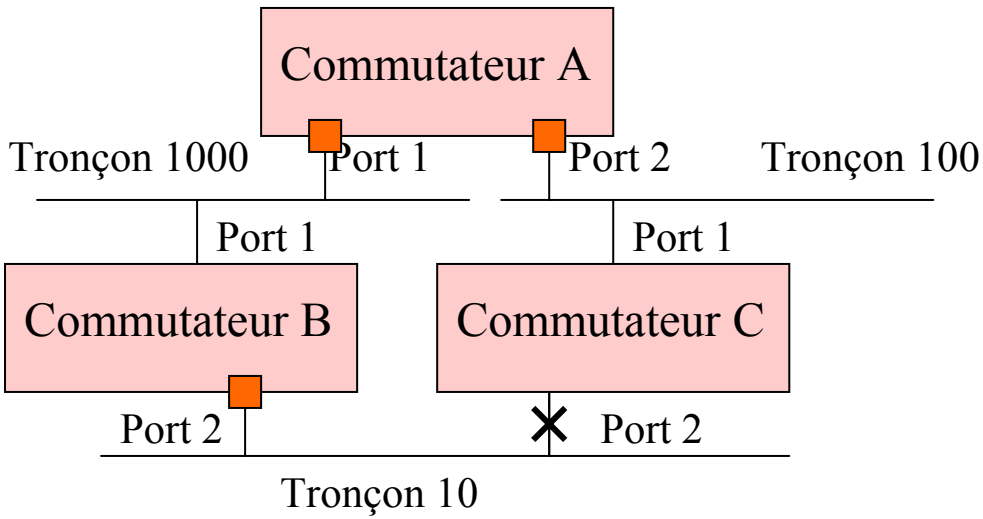
- Étape 1 : Élection du commutateur racine avec les données suivantes
  Commutateur A : Priorité=10 , MAC=00B0D7000001, ID=000A00B0D7000001
  Commutateur B : Priorité=27 , MAC=00B0D7000002, ID=001B00B0D7000002
  Commutateur C : Priorité=32768 , MAC=00B0D7000003, ID=800000B0D7000002
- On élit le **commutateur A**.

Approfondissements : un exemple de fonctionnement Étape 2



- Étape 2 : Élection des ports racine avec les données suivantes
  Commutateur A : Port 1 Coût 4 Priorité 128, Port 2 Coût 19 Priorité 128
  Commutateur B : Port 1 Coût 4 Priorité 128, Port 2 Coût 100 Priorité 128
  Commutateur C : Port 1 Coût 19 Priorité 128, Port 2 Coût 100 Priorité 128
- **Commutateur B** Chemin de B à A: Par le port 1 coût 4, par le port 2 coût 119 donc port 1 moins cher port racine.
- **Commutateur C** Chemin de C à A: Par le port 1 coût 19, par le port 2 coût 104 donc port 1 moins cher port racine.

Approfondissements : un exemple de fonctionnement Étape 3



- Étape 3 : Élection des ports désignés avec les données suivantes
Tronçon 1000 : Commutateur A Port 1 Port désigné commutateur A racine
Tronçon 100 : Commutateur A Port 2 Port désigné commutateur A racine
Tronçon 10 : Commutateur B Port 2 coût 4, et Commutateur C Port 2 coût 19. Le coût le plus faible gagne. Le port 2 commutateur B est désigné. Le port 2 commutateur C n'étant pas désigné est bloqué.

Conclusion : construction de l'arbre couvrant

- La principale conséquence de la méthode de routage par arbre couvrant est **qu'il ne sert à rien de créer des chemins redondants pour améliorer les performances** => on ne garde qu 'un seul chemin actif les autres sont bloqués.
- Les seules **redondances de chemin utilisables le sont pour des objectifs de sûreté de fonctionnement**.

Annexe: ponts en routage par la source "Source routing bridges"

- La solution par arbre couvrant **optimise mal les ressources** offertes par les différents commutateurs.
- Solution différente au problème de routage adoptée par IBM pour les boucle à jeton et utilisée **uniquement** dans ce cas
- Chaque station émettrice **doit connaître la topologie d'ensemble du réseau des commutateurs.**
- Pour chaque trame transmise l'émetteur place dans la trame **la liste des commutateurs à traverser.**
- On peut avoir plusieurs routes pour aller d'un point à un autre et **optimiser les communications.**
- => **Allongement des tailles maximum de trame** possible uniquement pour les communications entre commutateurs.

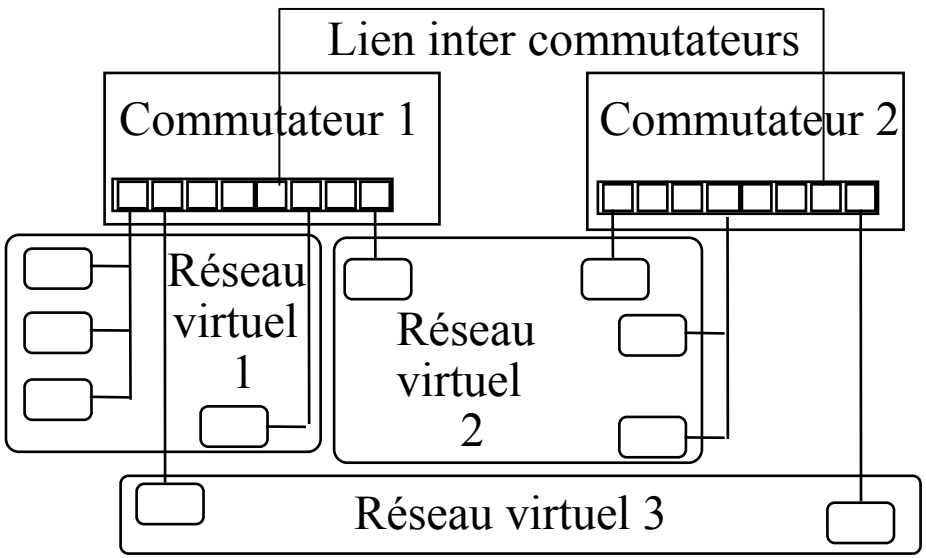
4
Les réseaux locaux virtuels

Position du problème des réseaux locaux virtuels

- Création de **"sous réseaux locaux"** regroupant des stations sur une base logique et non topologique.
- Les groupes cohérents indépendamment de la localisation géographique des stations forment des **réseaux locaux virtuels ou VLAN ' Virtual LAN '.**
- **Principe fondamental:** ne permettre les communications entre stations **qu'à l'intérieur d'un réseau virtuel.**
  - Un VLAN définit un sous réseau fermé de stations, un domaine de collision, un domaine de diffusion générale et de diffusion sur groupe.
- Pour communiquer **entre deux réseaux virtuels** il faut passer par **des stations appartenant aux deux VLAN** et utiliser le routage de niveau réseau => dans un système de VLAN, il doit donc être **possible pour une station d'appartenir à plusieurs VLAN.**

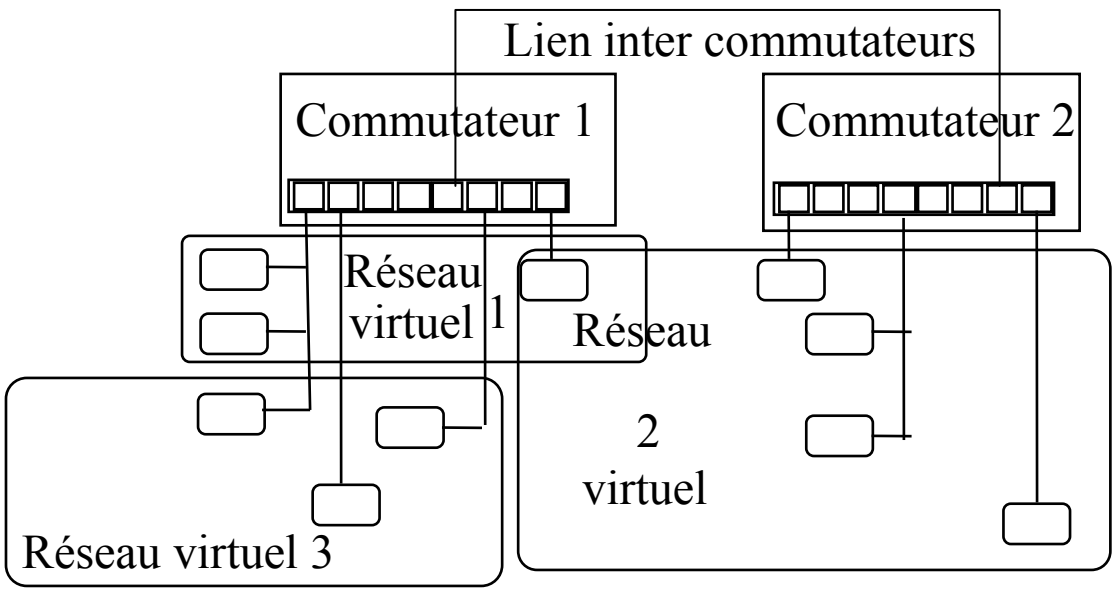
Différentes catégories de VLAN
VLAN de type 1 : par ports

- Les stations connectées à un même port d'un commutateur font **obligatoirement partie du même VLAN.**
- Simplification de la gestion des VLAN: le numéro de VLAN est associé au numéro de port par l'administrateur réseau (notion de VLAN ' port based ')
- La sécurité est excellente: **toutes les stations sur un tronçon peuvent communiquer** parce qu'appartenant au même VLAN.



Différentes catégories de VLAN
VLAN de type 2 : par adresses MAC

- Chaque station **caractérisée par son adresse MAC** peut appartenir **à un VLAN donné** (notion de VLAN ' address based ').
- Un administrateur doit décider **l'affectation d'une station à un VLAN.**
- Le numéro de VLAN doit apparaître dans la **table de routage.**



Différentes catégories de VLAN
VLAN de type 3 : par adresses réseaux

- Chaque station est également **caractérisée par son adresse IP.**
- Si le commutateur de réseau local analyse les charges utiles des trames et qu'il s'agit de paquets IP, une station peut appartenir **à un VLAN donné** sur la base de son **adresse IP** (notion de VLAN ' network address based ').
- L 'administrateur système doit définir les VLAN par des **ensembles d'adresses IP ou des plages d'adresses IP** (des sous réseaux IP).

Autres possibilités de définition des VLAN
(si l 'on analyse les charges utiles).

- Par **type de protocole** destinataire.
- Par **numéro de port TCP.**
- Par identificateur de **compte utilisateur** (login par exemple en telnet).
- ….

Fonctionnement des VLAN
Solution des messages de signalisation

- Les commutateurs échangent **des messages de signalisation (courts)** comportant une adresse MAC et le numéro de réseau virtuel associé pour mise à jour des tables de routage.
- Un message de signalisation est **généré lors de la mise sous tension** d'une station et propagé à tous les commutateurs du réseau.
- Les tables de routage sont **échangées périodiquement**.
- Problèmes de performances **si le réseau est grand :**
  - **Surcharge d'échanges** de messages de signalisation
  - **Taille des tables de routage** et **coût de la recherche en table.**

Fonctionnement des VLAN: Solution de l'estampillage de trames

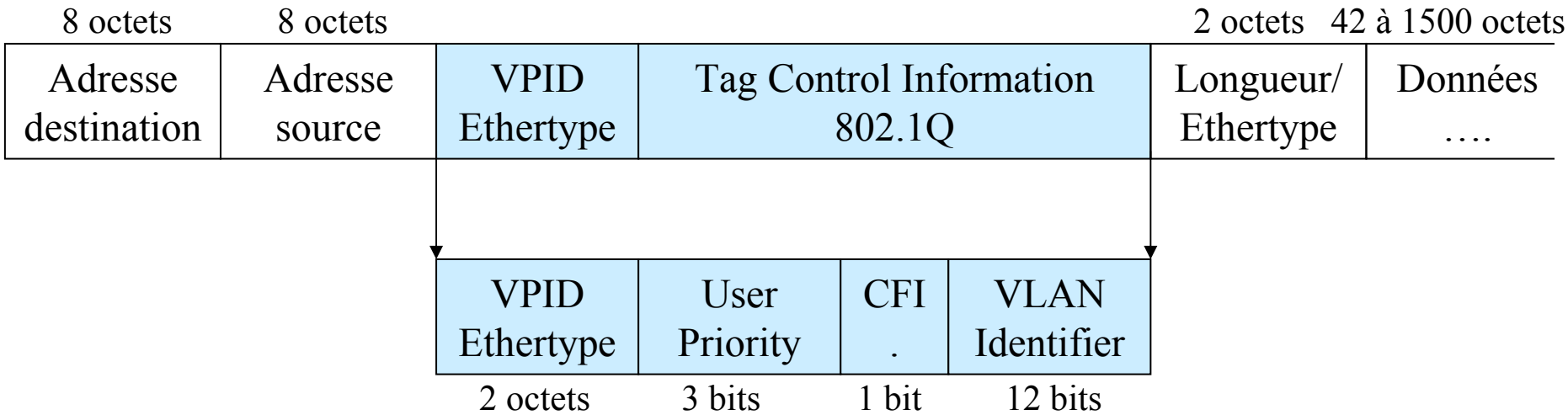
La norme 802.1 Q (' Frame tagging ')

- **Estampillage:** ajouter à la trame Ethernet (après l'adresse source) des informations (4 octets) définissant le VLAN de l'émetteur.

8 octets	8 octets	4 octets		2 octets	42 à 1500 octets
Adresse destination	Adresse source	VPID Ethertype	Tag Control Information 802.1p/Q	Longueur/ Ethertype	Données ….

- **Solution de VLAN par ports:** Pour éviter les problèmes en cas de diffusion un seul VLAN est défini par port (la solution ne concerne que les VLAN de type 1).
- **Connaissance des commutateurs:** VLAN d'appartenance des stations qui lui sont connectés (en fait VLAN des ports du commutateur).
- **Utilisation des estampilles:** uniquement pour les échanges entre commutateurs ce qui évite les problèmes dus au fait de rallonger la taille maximum de la trame Ethernet de 1518 à 1522 octets.

Approfondissement: signification des différents champs



- **Champ VPID** ('VLAN Protocol Identifier'): Code ETHERTYPE fixé à 0x8100.
- **Champ UP** (' User Priority '): 3 bits permettant de définir 8 niveaux de priorité pour une trame (sans rapport avec les VLAN mais introduit à propos d 'une modification du format de la trame Ethernet).
- **Champ CFI** (' Canonical Format Identifier '): Indique pour le routage par la source que le format est standard.
- **Champ VID** ('Vlan Identifier'): Indique pour quel VLAN circule la trame (12 bits).

Approfondissement: fonctionnement de base de l'estampillage de trames:

- **Réception de trame non estampillée**: le commutateur affecte le numéro de VLAN du port d'arrivée (ou un numéro par défaut en l'absence de configuration).
- **Réception d'une trame estampillée**: générée soit par un commutateur 802.1 Q soit par une station capable de générer le format 802.1Q. Le commutateur considère le VLAN de l'estampille (VLAN pour lequel circule la trame)
- **Décision de routage** (' forwarding decision ')
  - Trame Broadcast, Multicast et Unicast destinataire encore inconnu: la trame est renvoyée sur tous les ports ayant même VLAN que la trame.
  - Trame Unicast de destinataire connu: la trame est renvoyée sur le port destinataire si le VLAN du port correspond au VLAN de la trame.
- **D'autres stratégies** propriétaires peuvent être implémentées par des commutateurs pour des VLAN par adresses.

Conclusion :avantages et inconvénients des réseaux locaux virtuels

- **Avantages** des réseaux virtuels:
  - Améliorer **la sécurité** en limitant la circulation des trames (totale indépendance des trafics sur un câblage commun).
  - Améliorer **les performances** en limitant l'étendue des diffusions au réseau virtuel d'appartenance de l'émetteur.
  - Améliorer **la gestion de la configuration** du réseau: définition des groupes d'utilisateurs sans se soucier de l'endroit ou ils sont connectés.
- **Inconvénients** des réseaux virtuels:
  - Lourdeur de **l'administration.**
  - **Contrainte** des mécanismes de sécurité sur les usagers.
  - **Ralentissement** des communications entre réseaux virtuels.

Commutation de réseaux locaux

Conclusion

Commutateurs de réseaux locaux: Avantages (1)

- **Organisation :** S'adapte à toute topologie existante, organisée par tronçons de réseaux, en reliant des ensembles de matériels dispersés géographiquement sur des tronçons de réseaux locaux séparés.
- **Extension en distance :** On peut s'affranchir des contraintes de distance maximum des réseaux locaux par des liaisons spécialisées entre commutateurs.
- **Extension en performances :**
  - Limitation du trafic aux tronçons concernés ,
  - Limitation des domaines de collision ,
  - Possibilité de limitation des diffusions avec les réseaux locaux virtuels.
- **Sécurité :** On peut interdire sélectivement le franchissement des commutateurs. En particulier le mode écoute générale des réseaux locaux ("promiscuous") qui permet beaucoup de piratage peut-être très réduit avec les réseaux locaux virtuels.

Commutateurs de réseaux locaux: Avantages (2)

- **Tolérance aux pannes :** Approche très tolérante puisqu'elle sépare différents tronçons qui peuvent s'arrêter séparément (possibilité de traiter par exemple le problème des avalanches ou tempêtes de diffusion).
- **Interconnexion de réseaux locaux hétérogènes** : Possibilité de supporter plusieurs standards de réseaux locaux ayant des formats de trames voisines (par exemple type IEEE 802) en résolvant néanmoins des problèmes non négligeables d'hétérogénéité.
  - Adressage (unification autour de l'adressage IEEE 802)
  - Débits 802.3 10 Mb/s , 100 Mb/s 802.5 16 Mb/s ("Token ring") 802.11
  - Taille maximum des trames 1500 ou 4500 octets … Segmentation non prévue au niveau liaison.
- **Coûts:** plus importants que pour les répéteurs mais très supportables et en diminution constante.

Commutateurs de réseaux locaux: Inconvénients

- **Retards :** Lors de la traversée des commutateurs.
- **Limitation dues aux tailles des tables de routage :** le nombre des adresses utilisables n'est pas illimité (existence de tables dont la taille ne peut-être arbitraire).
- **Limitation du débit supporté :** un commutateur peut entrer en surcharge ou en congestion en cas de pic de trafic => Nécessité d'assurer un dimensionnement correct d'une architecture de commutateur.
- **Une administration un peu plus complexe :** malgré le mode transparent en cas d'utilisation des réseaux virtuels.

Ensemble d'inconvénients faibles en regard des avantages

Niveau Réseau
"Network Layer"

Généralités: problèmes de routage et de commutation.

Protocole IP ("Internet Protocol")

Niveau Réseau
"Network Layer"

Généralités problèmes de routage et de commutation

Introduction

Adressage

Routage/Commutation

Contrôle de congestion

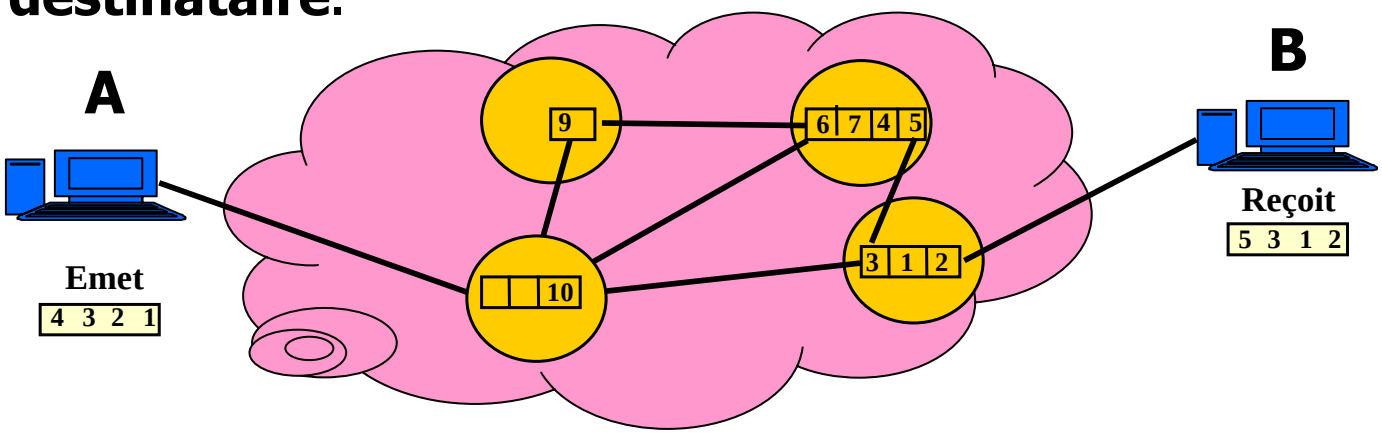
Problèmes généraux de réalisation du niveau réseau

Introduction

Choix de conception.
Réseaux à datagrammes.
Réseaux à circuits virtuels.

Réseaux à datagrammes

- **Modèle analogue à celui de la poste**
  - **Les datagrammes sont des paquets routés indépendamment** les uns des autres.
  - **Chaque datagramme comporte dans son entête l'adresse du destinataire.**



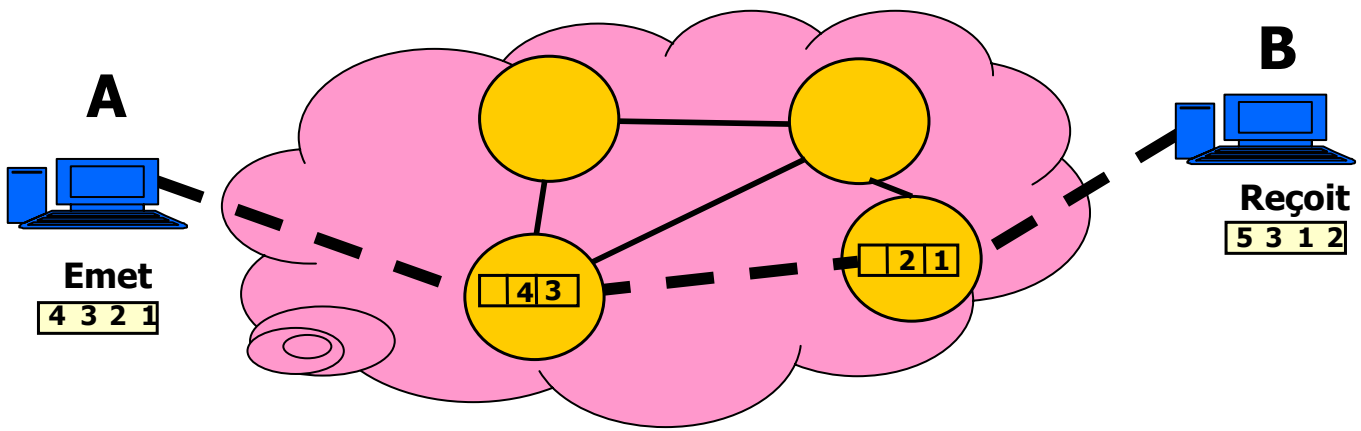
- **Choix coûteux du point de vue du routage**
  - Chaque datagramme subit un routage.
- **Choix permettant plus d'optimisation**
  - On peut tenir compte d'informations récentes pour réaliser le routage

Options des réseaux à datagrammes

- **Solutions des problèmes d'adressage et de routage.**
- **Difficultés pour la livraison en séquence**:
  - Non respect de l'ordre local d'émission du site A lors de la délivrance en B
  - Les messages ne se suivent pas.
- **Difficultés pour le contrôle de flux** d'hôte à hôte
  - Difficulté de rétroaction du destinataire sur l'émetteur.
- **Pas de contrôle d'erreurs**
  - Possible mais non retenu.
- **Gestion des ressources**
  - Optimisation globale favorisée par le traitement sur chaque datagramme.
  - Difficulté pour satisfaire des contraintes de QOS spécifiques de chaque usager
    - Difficulté pour la réservation des tampons, de la bande passante.
    - Traitement des problèmes de congestion.
- **Les problèmes non résolus sont reportés au niveau transport.**
- **Exemple : IP Internet Protocol**
  - **Le dernier des protocoles à datagrammes mais le plus important.**
  - **Autres exemples historiques : Cyclades, Decnet ...**

Réseaux à circuits virtuels

- **Modèle analogue à celui du téléphone**
  - **Circuit virtuel** = **chemin fixe** établi entre deux hôtes.
  - **Le chemin est initialisé au préalable:**
    - Exemple "**paquet d'appel**" routé de l'émetteur au destinataire.
  - **Les données empruntent ensuite toujours ce chemin**
    - S'il y a une panne => il faut reconstruire le circuit virtuel.
  - **Simplification des opérations de commutation**.



Réseaux à circuits virtuels à bas débits

- **Un ensemble d'options assez complètes** pour une couche réseau consistante: **Exemple X25 niveau paquet**
- **Fonctions d'adressage et de routage.**
- **Gestion des connexions.**
- **Livraison en séquence** (d'hôte à hôte)
  - Respect de l'ordre local d'émission du site A lors de la délivrance en B
  - Les paquets se suivent sur le circuit virtuel.
- **Contrôle de flux** (d'hôte à hôte)
  - Par rétroaction sur le circuit virtuel.
- **Allocation préalable de ressources** sur un circuit virtuel.
  - Assure la fourniture d'une qualité de service donnée.

Réseaux à circuits virtuels à haut débit

- **Relais de trames FR, "Frame Relay"**
  - **Un allégement des choix X25** (réseaux publics de transmission de données) afin de permettre la commutation efficace de trafics de plus hauts débits.
- **ATM "Asynchronous Transfer Mode**
  - Réseau numérique à intégration de service large bande.
  - Transmission de données multimédia image, voix, données => Satisfaction des contraintes de qualité de service pour ces trois types de données.
- **MPLS "Multi Protocol Label Switching"**
  - Mêmes objectifs que ATM avec des améliorations et **en s'intégrant mieux à IP**.

Choix techniques des réseaux à circuits virtuels hauts débits

- **Adressage et routage**
- **Communication en mode connecté**
- **Livraison en séquence**
  - Paquets à la suite selon un circuit virtuel.
- **Sans contrôle d'erreur**
  - Haut débit : taux d'erreur faible.
  - Contrôle d'erreur par retransmission inadéquat pour le multimédia.
- **Sans contrôle de flux**
  - Inutile pour les trafics isochrones (sons, images).
- **Protocoles supplémentaires** d'adaptation spécifique du type de données échangées.
- **Garantie de qualité de service** (mode circuit virtuel).

Relations entre mode avec ou sans connexion et mode datagramme ou CV

Rappel

- **Mode connecté :** Les échanges ne peuvent avoir lieu qu'entre deux événements ouverture et fermeture.
- **Mode non Connecté :** Les échanges dans une peuvent prendre place à tout moment.
- **Mode datagramme :** Pas de chemin fixe.
- **Mode Circuit virtuel :** Même chemin fixe.

Associations entre modes de fonctionnement

Association entre réseaux à circuits virtuels et communications en connexions

- Désignation des connexions ou des circuits virtuels.
- Facilite la gestion des ressources et la qualité de service (QOS).
- Facilite la correction des erreurs de transmission.
- Facilite la livraison en séquence d'hôte à hôte.
- Facilite le contrôle de flux d'hôte à hôte.

Association entre mode datagrammes et communications sans connexions

- Difficultés pour la livraison en séquence d'hôte à hôte.
- Difficultés pour le contrôle de flux d'hôte à hôte.
- Difficultés pour la gestion des ressources et la qualité de service.

Différences entre modes de fonctionnement

- **A) Possibilité de construire un protocole en mode connecté sur un réseau à datagrammes** (puisqu'on peut communiquer).
- **B) Utilisation dans un réseau à circuits virtuels comme X25 du routage du paquet d'appel** pour transmettre des datagrammes (option "fast select").

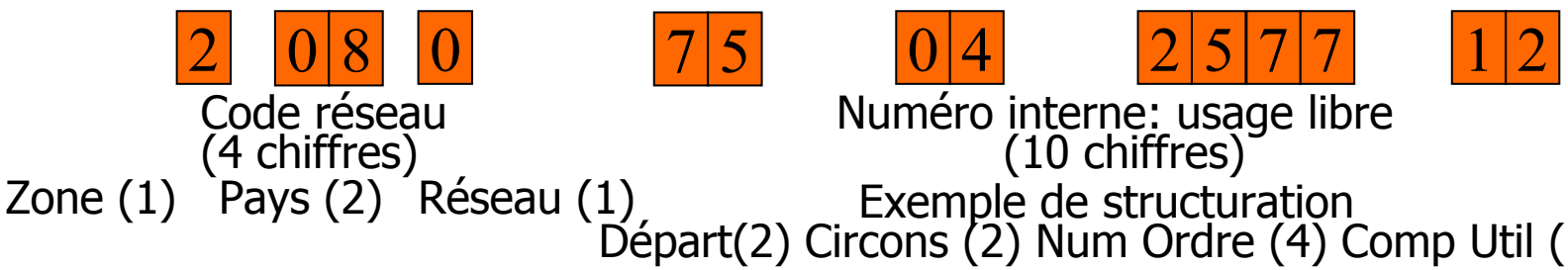
- **Conclusion : Il est tout à fait impropre d'utiliser les équivalences**
  - **Datagramme = Mode non connecté.**
  - **Circuit virtuel = Mode connecté.**

Problèmes généraux de réalisation du niveau réseau

Adressage

Adressage/Nommage/Désignation : Position du problème (1)

- **Addressing, Naming** : L'ensemble des techniques associées aux adresses et aux noms (ici au niveau 3 réseau).
- **1) Politique de structuration** des différentes zones.
  - Comment est organisée la structure de données nom ou adresse.
  - Exemple : les adresses X121 des réseaux publics de transmission de données (type TRANSPAC).



- **2) Définition des autorités administratives** compétentes dans l'attribution des différents champs des noms et adresses.
  - Exemples d'autorités : Organismes internationaux, Opérateurs de télécoms, fournisseurs d'accès réseaux, sociétés délégataires.
  - Ex adresses IP : ISOC -> IANA -> RIPE -> Opérateur -> Entreprise.

Adressage/Nommage/Désignation : Position du problème (2)

- **3) Politique de stockage et d'accès** aux différents moyens de désignation : adresses ou noms.
  - Utilisation dans les annuaires : exemple DNS
  - Utilisation dans les tables de routage (tous les protocoles).
- **4) Politique d'utilisation des adresses dans le cadre des opérations de routage**.
  - **Cas des datagrammes.** Dans chaque paquet figurent:
    - L'adresse de l'émetteur
    - L'adresse du destinataire.
  - **Cas des circuits virtuels.**
    - Dans le paquet appel figurent les adresses émetteur et destinataire.
    - Dans chaque paquet figure ensuite l'identifiant du circuit virtuel.

Définitions relatives aux noms et adresses

- **Nom/Adresse**
  - **Nommer**: Identifier de façon unique.
  - **Adresser**: Identifier de façon à retrouver dans un réseau.
- **Nom/Adresse physique ou logique**
  - **Physique:** Identification associée à des aspects matériels invariants (numéro de série, adresse unique MAC) => Plutôt une adresse.
  - **Logique:** Identification selon une chaîne de caractères quelconque => Plutôt un nom.
- **Nom/Adresse fixe ou mobile**
  - **Fixe**: qui ne peut suivre le déplacement de l'appareil connecté Exemple: numéro de téléphone attaché à un autocommutateur).
  - **Mobile**: qui permet de retrouver un destinataire indépendamment de la localisation géographique. Exemple : numéro de téléphone portable localisable dans une cellule du réseau.

Adressage global / Adressage local

- **Pour aller à un endroit on peut :**
  - Donner l'adresse de l'endroit => **Notion d'adresse "globale"**

    Exemple : Mr X, appartement y, z rue de .....
  - Définir le chemin à emprunter => **Notion d'adresse "locale"**

    Exemple : Tourner à gauche puis 2 ième à droite ....
- **Adressage global**
  - Un identifiant unique du site destinataire acheminé dans chaque message pour déterminer son routage et sa délivrance.
  - Exemple: adresse IP, adresse ATM.
- **Adressage local**
  - Un chemin défini par la suite des décisions à prendre lors de la traversée de chaque commutateur.
  - Exemple: identification des circuits virtuels en X25, ATM , MPLS ....

Structuration des adresses globales : Adressage plat

- **Définition :** Les adresses sont définies dans une zone de n bits sans règles de structuration particulière permettant la localisation.
- **Avantages**
  - Peu de problèmes d'administration.
  - Pas de perte de place : les adresses sont plus courtes

  => Minimisation de l'encombrement dans les messages.
- **Inconvénients**
  - N'offre pas de moyen lié à la structure des adresses pour retrouver un correspondant

  =>Inadapté aux grands réseaux.
- **Exemple :** Adresses de liaison de niveau MAC : IEEE 802.
- **Conclusion :** Utilisation dans les petits réseaux.

Structuration des adresses globales : Adressage hiérarchique

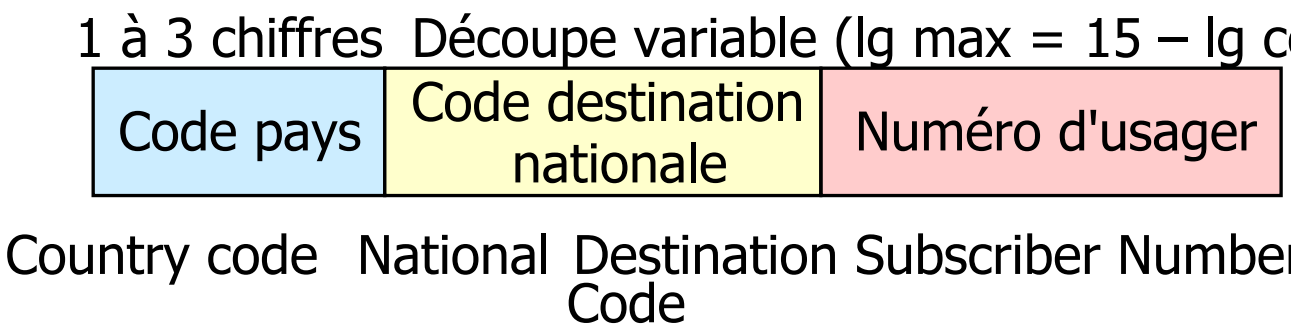
- **Définition :** les adresses sont définies avec des règles de structuration par champs qui correspondent à des regroupements géographiques/topologiques **:**
- **Exemple de découpage géographique :** pays, région, entreprise, réseau, sous/réseau ...
- **Notion de hiérarchisation** du routage/ agrégation de routes:
  - Atteindre un routeur dans le pays puis atteindre un routeur dans la région etc ….
- **Tous les systèmes d'adressage** de niveau 3 sont hiérarchiques.

Adressage hiérarchique : avantages, inconvénients

- **Avantages**
  - **Adapté aux grands réseaux** => Possibilité de gérer de grands espaces. L'administration des adresses peut se faire de façon locale à chaque unité de découpage.
  - **Adapté à la localisation**
    - Les adresses permettent de localiser le destinataire.
    - Au moyen d'un routage hiérarchisé.
- **Inconvénients**
  - **Encombrement** Le découpage opéré peut-être plus ou moins efficace dans chaque champ
    - =>généralement les adresses sont volumineuses.
    - grande perte de capacités d'adressage (gaspillage d'adresses).
  - **Changement de localisation difficile**
    - Un site qui change de localisation doit changer d'adresse.
    - Sauf mise en place un mécanisme de redirection.

Exemple d'adressage hiérarchique 1) Adressage E164

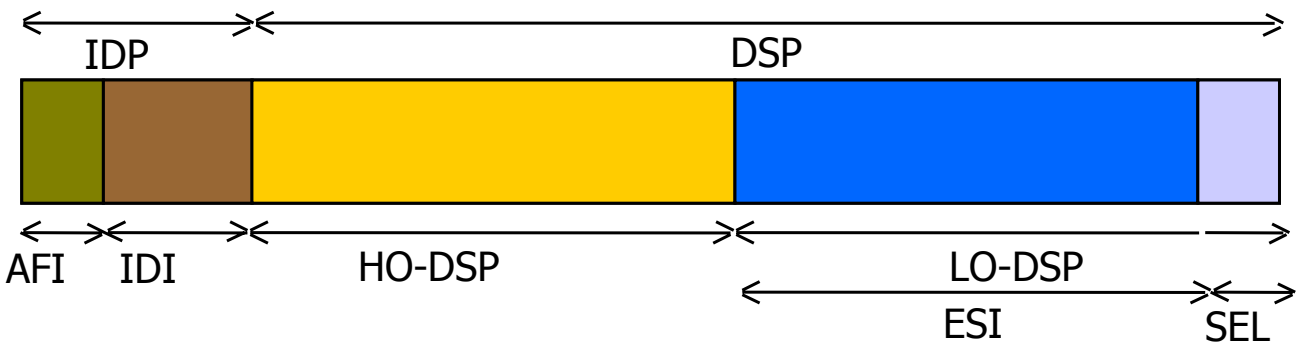
- **Utilisation :** Le plan de numérotation du téléphone classique. **RTC**: **R**éseau **T**éléphonique **C**ommuté ou **PSTN/POTS**: **P**ublic **S**witched **T**elephone **N**etwork ou **P**lain **O**ld **T**elephone **S**ystem
  - Mais aussi réseaux de téléphonie mobile, réseau RNIS (Réseau Numérique à Intégration de Service), réseaux ATM publics ….
- **Structure :**
  - **Adresse basée sur un système de 15 chiffres décimaux** maximum codés en BCD (Binary Coded Decimal) deux chiffres par octets.
  - Forme générale d'une adresse téléphonique géographique.



  - Autres structures : Services, adresses résaux (code pays spécial).
  - Représentation en DNS: 2.4.2.4.5.5.5.1.e164.arpa (numéro 1-5554242).

Exemple d'adressage hiérarchique 2) Adressage normalisé OSI 8348

- **Utilisation :** réseaux ATM privés
- **Format variable en deux parties** pour des adresses de NSAP **("Network Service Access point")**
- **Taille maximum** : 20 octets.
- **Structure :**

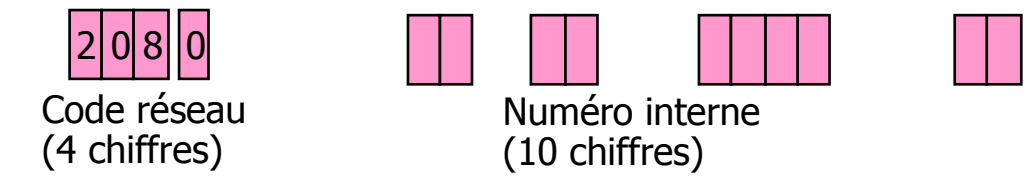


- **IDP "Initial Domain Part"** : Spécifie le format de l'adresse (AFI) puis l'autorité responsable de l'attribution de cette adresse (IDI).
- **DSP "Domain Specific Part"** : Spécifie plus particulièrement un site.

  Séparé en trois parties: "High Order Bits" (pour le routage), une partie identificateur de site ("End System Identifier) plus un sélecteur d'application (SEL) ("Low Order Bits").

Exemples d'adressage hiérarchique
Autres exemples

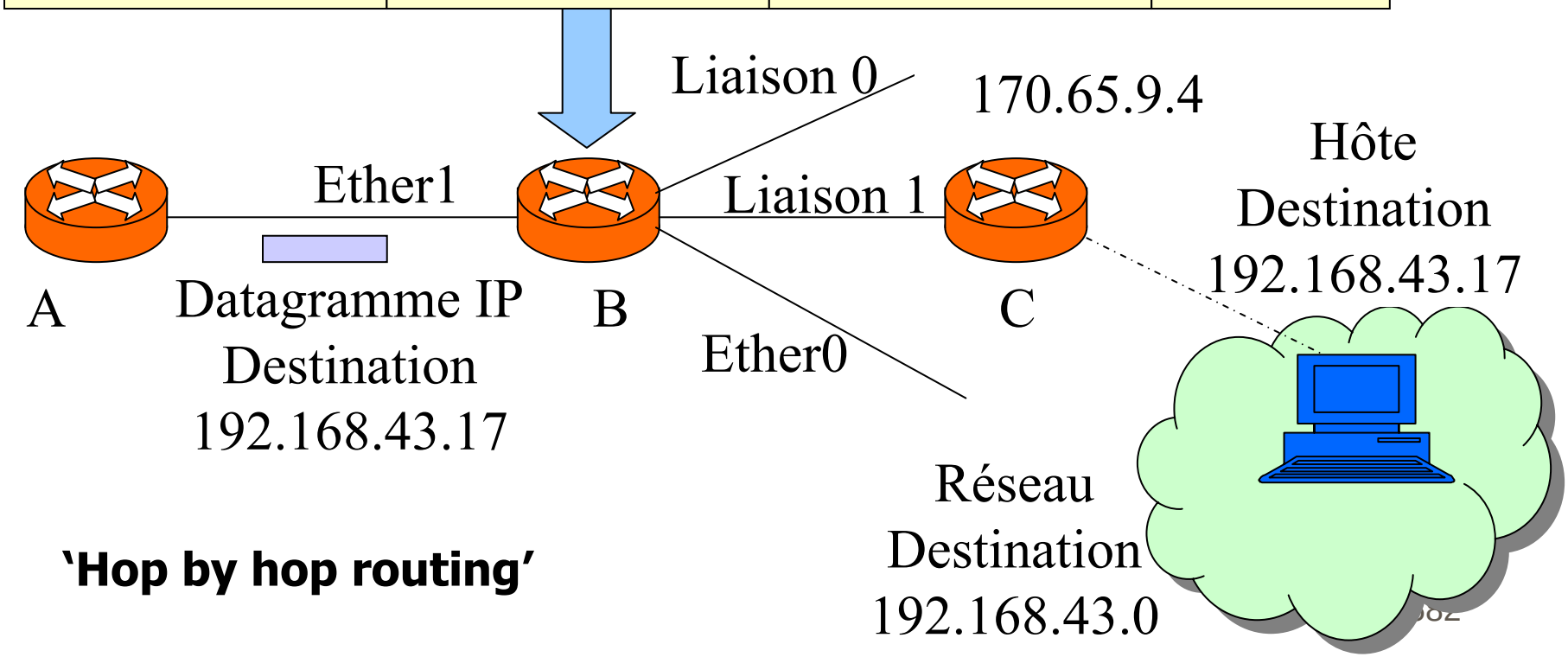
- **3) Adressage des réseaux publics X121**
  - Sur 14 chiffres décimaux (4 chiffres pour le réseau, 10 chiffres pour l'adresse hôte dans le réseau).
  - Pour un réseau différentes possibilités de structuration de la partie numéro interne.



- **4) Adressage IP (Internet niveau 3)**
  - **Deux normes d'adressage** correspondent aux deux versions successives majeures.
  - **IPV4** Adresse sur 32 bits.
  - **IPV6** Adresse sur 128 bits.
  - Traité en détail dans le cours IP.

Utilisation des adresses globales : routage saut par saut en IP

Préfixe Dest	Masque Dest	Prochain saut	Interface
192.168.43.0	/24	170.65.9.4	Liaison 1
…	…	…	…

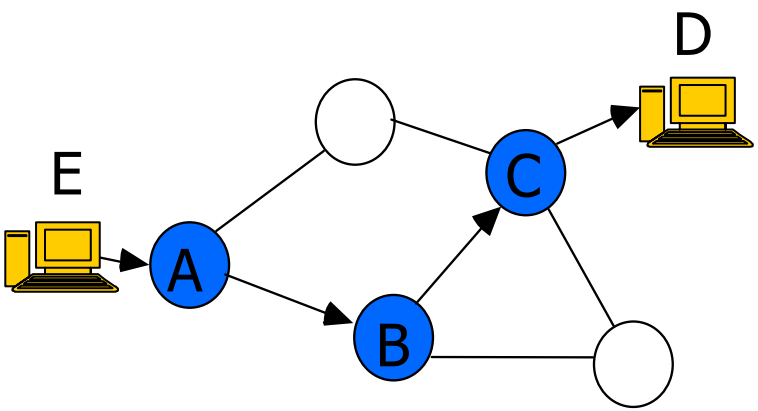


'Hop by hop routing'

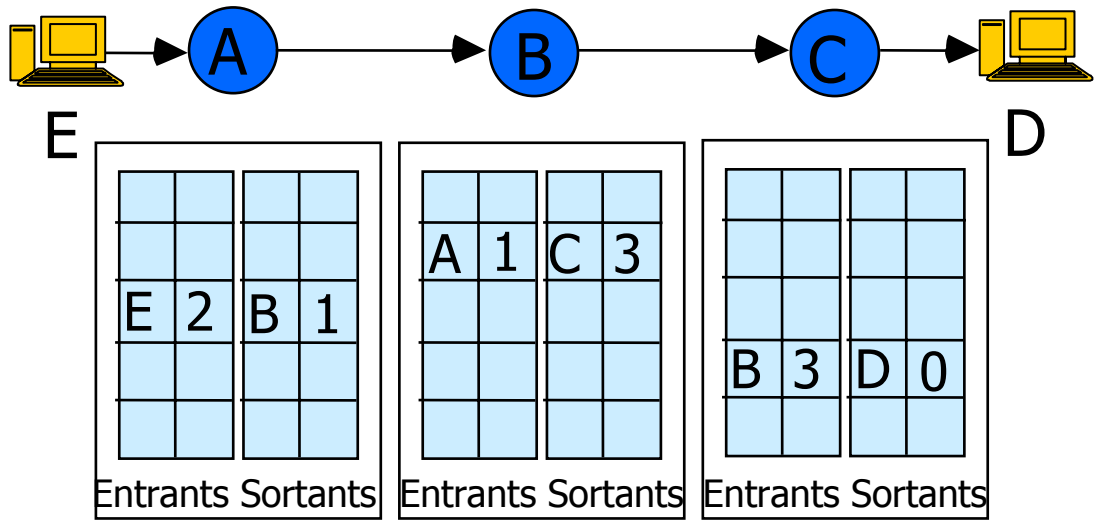
Adresses locales dans les réseaux à Circuits Virtuels (CV)

- **Problème d'identification d'un circuit virtuel.**
  - **Comment désigner un chemin** pour aller d'un point à un autre dans un réseau à circuits virtuels (CV) ?
- **Nom unique global de CV.**
  - **Par exemple :** adresse appelant, adresse appelé, numéro CV.
  - **Solution très encombrante** Exemple: 30 digits pour des adresses X121 plus un numéro de CV.
  - **N'apporte pas de solution particulière** pour le routage.
- **Nom contextuel local de CV.**
  - **Gestion locale** de noms de petite taille (par exemple 12 bits).
  - **Problèmes d'homonymie** : à l'ouverture d'un CV sur un site on lui attribue un numéro libre.
  - **Chaque commutateur traversé** doit pouvoir modifier le numéro du circuit virtuel en lui attribuant un nom unique libre localement.
  - **Dans chaque commutateur il faut une table de routage** : correspondance entre les circuits entrants et sortants.

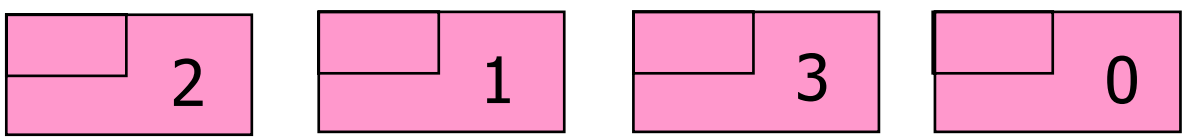
Fonctionnement de l'adressage local dans les circuits virtuels



Réseau avec un CV E A B C D



Tables des CV entrants et sortants en A B C
(tables de routage locales)



Paquets de E vers A | Paquets de A vers B | Paquets de B vers C | Paquets de C vers D

Évolution des entêtes de paquets (désignation du circuit virtuel)

Problèmes généraux de réalisation du niveau réseau

Commutation et routage

Introduction/généralités
Exemples de protocoles de routage

Introduction / généralités

Acception traditionnelle

- **Commutation (‘switching’)** : terminologie créée dans le contexte des réseaux téléphoniques.
  - **Opération de retransmission d’une information** d’un port entrée vers un port de sortie **réalisée plutôt par du matériel** spécialisé en mode circuit.
  - **Exemple:** la commutation de réseaux locaux, ATM.
- **Routage (‘routing’)** : terminologie créée dans le contexte des réseaux IP.
  - **Opération de retransmission d’information réalisée par un ordinateur** utilisant un logiciel, des tables de routage, une technique de stockage et retransmission.
  - **Exemple:** le routage IP.

Notions de commutation et de routage

- **Pour acheminer un paquet** : deux problèmes à résoudre.
  - **Problème A) Construire des routes** (des tables de routage dans les routeurs) pour aller d'un émetteur à un destinataire

    => Le travail réalisé par un protocole de routage (dynamique).
  - **Problème B) Réaliser effectivement l'acheminement** : acquérir un paquet, consulter son entête, consulter une table de routage, renvoyer le paquet dans la bonne direction.
- **Terminologie classique pour les problèmes A et B**
  - **Problème A)** Découverte des routes ('route discovery), annonce des routes ('route advertisment').
  - **Problème B)** Relayage ('forwarding'), commutation ('switching').
- **Spécialisation terminologique constatée.**
  - **Problème A) Routage ('routing')** : le problème de construction des routes (des tables de routage) par un protocole de routage.
  - **Problème B) Commutation ('switching')**: l'acheminement, le relayage d'un paquet par un commutateur.

Les trois idées principales pour réaliser le routage dans un réseau maillé

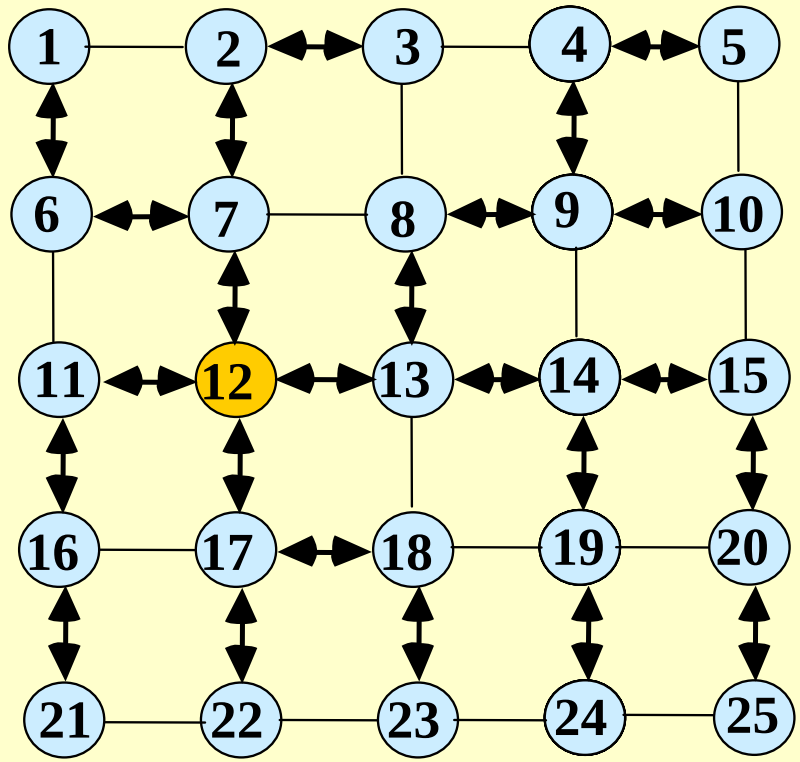
- **1) Routage utilisant la diffusion** (‘Broadcast routing’)
  - Tous les sites du réseau étant atteints, seuls les destinataires adressés dans le message délivrent le message => Besoin d’une diffusion
  - **Diffusion naturelle sur une voie commune** partagée (réseau local).
  - **Diffusion réalisée sur réseau maillé par inondation** (‘Flooding’).
    - Chaque nœud retransmet le paquet sur tous ses liens adjacents.
- **2) Routage défini par la source** (‘Source routing’)
  - La route est déterminée par l’émetteur et insérée dans le paquet.
  - Les routeurs successifs appliquent cette route pour acheminer le paquet.
- **3) Routage défini saut par saut** (‘Hop by hop routing’)
  - Chaque routeur doit connaître des routes vers tous les destinataires en échangeant des informations sur la topologie du réseau.
  - Chaque routeur applique une décision de routage vers un routeur voisin.

La gestion des tables de routage en mode saut par saut

- **Dans un routeur :** à partir d'une adresse destinataire **choisir la meilleure direction de sortie.**
  - Les noeuds mémorisent dans des tables de routage des informations du type: destinataire ; prochain site à atteindre (par quel port l'atteindre, quel contrôleur de voie physique à utiliser).
- **Utilisation du routage saut par saut**
  - **Dans le mode datagramme** -> pour chaque paquet.
  - **Dans le mode circuit virtuel** -> à l'établissement de chaque circuit virtuel (ou lors de sa reconstruction).
- **Routage saut par saut :** de très loin le plus employé au niveau 3.
  - Les autres solutions sont employées de façon marginale.

Routage saut par saut : Notion d'arbre couvrant (arbre de routage)

- Le **réseau de communication** est considéré comme un **graphe**.
- Chaque commutateur est considéré comme la **racine d'un arbre couvrant ('spanning tree').**
- **L'arbre définit les chemins** pour atteindre tous les hôtes du réseau.
- **Exemple d'un arbre couvrant** pour le noeud 12 dans un réseau de type grille.



- **Exemple de table de routage** pour le nœud 12

Destinataire	1	2	3	4	5	6	...
Adjacent Préféré	7	7	7	13	13	7	...

Arbre couvrant des plus courts chemins

- **Construction d'un arbre couvrant qui optimise un critère de coût.**
  => Notion d'arbre couvrant des plus courts chemins.

Différents critères de coûts (métriques, "metrics")

- **Longueur du chemin** : en nombre de commutateurs traversés.
- **Délai de traversée du chemin** : temps effectif d'acheminement (somme des délais d'attente dans les files des commutateurs, de propagation, …).
- **Débit d'un chemin** : débit des commutateurs et des voies empruntées.
- **Bande passante des voies empruntées sur un chemin** : un chemin qui ne comporte que des voies à 100 Mb/s est jugé préférable à un chemin qui emprunte une voie à 56000b/s.
- **Fiabilité d'un chemin** : on utilise les chemins selon leur fiabilité.
- **Coût monétaire effectif :** différences entre une voie louée à coût marginal minime ou une voie avec tarification au volume (Transpac).

Gestion des tables de routage

- **Routage statique (sans prise en compte de coûts)**
  - Définition statique de tables (compatibles).
- **Routage dynamique avec des coûts fixes (stabilisés)**
  - **Principe d'optimalité (Bellman):** Si J est sur le chemin optimal de I à K alors le chemin de J à K est aussi optimal.
  - **Justification du routage saut par saut (tables de routage locales à chaque routeur).**
- **Routage dynamique (coûts variables, modification permanente des coûts)**
  - **Prise en compte des informations les plus récentes.**
  - **Problème de routage : problème de contrôle optimal (mesure de coût puis rétro action sur le routage du réseau)**
    - Possibilité de bouclage pour certains messages.
    - Besoin d'une convergence rapide vers une solution stable si les coûts se stabilisent.

Qualités d'un algorithme de routage

- **Correct :** Permet d'atteindre effectivement le destinataire.
- **Adaptatif :** Évolutif en fonction de la charge (évite la congestion).
- **Optimal :** Choix du meilleur chemin au sens de critères
  - **Usager =>** Satisfaction d'un contrat de service (QOS)
  - **Exploitant réseau =>** Taux d'utilisation des ressources réseaux.
- **Robuste :** Prévu pour tolérer les pannes du réseau
- **Stable :** Convergence rapide d'un routage dynamique vers de nouvelles tables lors de modifications (sans oscillations).
- **Simple :** Comportement facile à décrire (peu de types de messages différents).
- **Efficace:** Ne consommant pas beaucoup de ressources (temps de calcul, nombre de messages échangés).
- **Équitable :** Traite tous les usagers de la même façon.

Algorithmes de routage : Quelques définitions

- **Routages plats** ("flat routing")
  - Gestion d'un seul chemin par destinataire.
- **Routages multi-chemins** ("multi path")
  - Gestion de plusieurs chemins.
- **Routage non hiérarchique**
  - **Pour un grand réseau nécessité d'une très grande table**
    - Encombrement mémoire important.
    - Temps de recherche important (de nombreuses recherches par seconde)
- **Routage hiérarchique : Exemple à deux niveaux**
  - Division du réseau en régions ou domaines.
  - Deux types de routages pouvant relever de solutions différentes
    - **Routage intra-domaine** (chaque domaine réalise un routage interne).
    - **Routage inter-domaine** (pour chaque domaine un (ou plusieurs) commutateur sont spécialisés dans le trafic inter domaines (trafic externe).

Classification des algorithmes de routage

- **Critère d'adaptabilité en fonction de la charge.**
  - **Routage "statique" , "non adaptatif", "prédéterminé"**
    - Tables non modifiées en fonction des conditions de charge.
    - Détermination manuelle des tables.
  - **Routage "dynamique", "adaptatif", "évolutif".**
    - Tables modifiée périodiquement en fonction de la charge et des pannes.
- **Critère de sûreté de fonctionnement.**
  - **Routage "centralisé" , "dissymétrique".**
    - Un site spécialisé calcule les tables de routage et les distribue.
    - => Problèmes de panne et de surcharge du site centralisé.
  - **Routage "décentralisé" , "réparti" , "symétrique".**
    - Toutes les stations calculent les informations de routage .
    - **Routage local :** connaissances locales (longueur des files d'attente).
    - **Routage global :** connaissances échangées avec le reste du réseau (le voisinage immédiat ou tout le réseau).
  - **Routage hybride :** Pouvant utiliser à la fois un site central et un algorithme réparti.

Exemples de protocoles de routage

Routages par inondation "Flooding"

- **Rappel**
  - Construction d'un protocole d'inondation pour atteindre tous les destinataires potentiels et délivrer à qui de droit.
- **Inondation générale ("Flooding")**
  - Pour tout paquet arrivé dans un routeur on retransmet une copie sur tous les adjacents.
  - Complément indispensable: stratégie d'arrêt de l'inondation.
- **Inondation sélective ("Selective Flooding")**
  - On transmet le paquet sur tous les adjacents d'un commutateur pris dans une liste.
    - Soit tirée aléatoirement dans l'ensemble des voisins.
    - Soit correspondant à des sites en direction du destinataire
  - Nécessité d'une stratégie d'arrêt de l'inondation (comme précédemment).

Stratégies d'arrêt de l'inondation

- **Solution 1 : Liste des sites déjà visités**
  - On évite de renvoyer une copie du paquet au site dont on l'a reçu (puisqu'il le connaît déjà).
  - On note dans chaque message la liste des sites déjà visités pour ne pas leur renvoyer (risque d'allongement excessif des messages).
- **Solution 2 : Durée de vie**
  - On note dans chaque copie le nombre de commutateurs traversés.
  - La copie est détruite après traversée de n commutateurs.
  - On doit prendre n > diamètre du réseau => beaucoup de surcharge.
- **Solution 3 : Estampille**
  - Les paquets sont identifiés : estampille comportant l'adresse émetteur et le numéro de séquence de l'émetteur.
  - La première copie reçue fait l'objet d'une inondation et les copies ultérieures sont détruites.
  - Nécessite de mémoriser des estampilles sur chaque routeur.
  - Coût de l'inondation: seulement deux copies d'un même message sur chaque liaison (une dans chaque sens).

Conclusion : Routages par inondation

- **Avantages**
  - **Technique très simple** à mettre en oeuvre.
  - **Très robuste :** détermine un chemin s'il existe.
  - **Détermine le chemin le plus court** (permet de délivrer des paquets très vite sauf problèmes de surcharge).
- **Inconvénients**
  - **Induit une charge élevée**.
- **Utilisation de l'inondation**
  - **Domaine des cas de pannes fréquentes** ou importantes (réseaux militaire, reconstruction d'un réseau).
  - **Implantation de protocoles à diffusion générale**
    Exemple : utilisation de l'inondation dans l'échange des informations de topologie et de coût dans les protocoles de routage (OSPF).

Routages par la source "Paquets fléchés","Source Routing"

- **Principe général**
  - **Chaque émetteur doit connaître la topologie d'ensemble** du réseau et déterminer son arbre de routage.
  - **Dans chaque paquet l'émetteur définit le chemin** qu'il doit emprunter pour atteindre son destinataire.
  - **Les routeurs appliquent le chemin**.
- **Variantes**
  - **1) Définition stricte** du chemin ('strict source routing').
  - **2) Définition non stricte** du chemin ('loose source routing') : définition d'un sous-ensemble de routeurs **qui doivent être traversés obligatoirement** mais le routage peut ajouter d'autres relais intermédiaires.

Conclusion : Routages par la source

- **Avantages**
  - Facilite le travail de chaque routeur intermédiaire (pas de calcul de route, commutation uniquement).
- **Inconvénients**
  - Volume d'informations dans les messages (une route).
  - Nécessité d'une connaissance globale de la topologie de tout le réseau par tous les émetteurs.
  - La connaissance globale peut être vieille.
- **Exemples :**
  - Commutateurs de réseaux locaux IBM.
  - Routage par la source en IP.
  - Algorithme du 'cranckback' en ATM (ouverture d'un CV).

Routages saut par saut : Routages statiques

- **Principe général**
  - **Les tables de routage stables définies manuellement**, modifiées rarement (lors de l'insertion ou de la disparition de sites) et invariantes.
  - **Les tables sont établies en fonction de critères de topologie et de performance** (évalués hors ligne) par le concepteur du réseau.
- **Avantages**
  - **Technique très simple** à mettre en œuvre.
- **Inconvénients**
  - **Non adaptatif en cas de charge élevée.**
  - **Sans apprentissage automatique** beaucoup trop de routes à définir.
- **Utilisation**
  - **Utilisable pour un réseau de petite taille**.
  - **Au démarrage du réseau le routage statique permet la définition de connaissances** de routage indispensables minimales (routes de délivrance locale) avant mise en oeuvre d'un routage dynamique.

Routages saut par saut dynamiques centralisés (adaptatifs)

- **Principe général**
  - **Existence d'un site central** particulier (RCC "Routing Control Center") : par exemple **le site administrateur du réseau** : chargé de la gestion des abonnés, de la facturation....
  - **Émission par les commutateurs** d'informations de charge vers cet administrateur.
    - **Périodiquement**
    - **Sur franchissement de seuils.**
  - **Le central calcule les tables de routage** (algorithme des plus courts chemins exécuté en mode centralisé) et les renvoie aux routeurs.
- **Variante: Serveurs de routes**
  - **Plusieurs sites calculent des tables** pour des routeurs qui ne font qu'appliquer les routes calculées.

Conclusion routages dynamiques centralisés (adaptatifs)

- **Avantages**
  - **Simple à mettre en oeuvre**.
  - **Solution optimale** sur les valeurs utilisées
  - **Solution correcte sans boucles**.
  - **Trafic stable d'informations échangées**.
  - **Pas de calcul** dans les commutateurs.
- **Inconvénients**
  - **Vulnérabilité de l'administrateur** => architecture redondante.
  - **Temps de calcul important** => charge du central
  - **Propagation des informations de charge**
    - Temps de collecte + calcul des tables + transmission des tables
      => Usage de tables anciennes
  - **Trafic élevé au voisinage du central de routage**.
- **Utilisation**
  - **A été très fréquente dans les réseaux commerciaux**.

Routages distribués (adaptatifs) Routages locaux ou isolés

- **Notion de routage local ou isolé.**
  - Les commutateurs utilisent uniquement des informations **locales** (informations qu'ils peuvent connaître sans communiquer).
  - Exemples : longueur des files d'attente, taux d'occupation des voies.
- **Terminologie fréquente :** Routage de type pomme de terre chaude ("Hot potatoe routing")
- **Version de base irréaliste**
  - **On se débarrasse le plus rapidement possible** d'un paquet.
  - **On transmet un paquet dans la direction** qui est actuellement la moins chargée (par exemple la file d'attente la plus courte).
  - **Avantages :** Simplicité , symétrie des comportements.
  - **Inconvénients :** Existence de boucles, l'arrivée n'est pas garantie.
- **Version réaliste**
  - **On transmet un paquet** dans la direction la moins chargée dans un ensemble de choix possibles correspondant à la direction du destinataire (fonctionnement en association avec un autre routage exemple statique)

Routages dynamiques distribués à connaissances globales

- **Principes**
  - **Distribué :** Aucun routeur ne joue un rôle particulier dans l'algorithme.
  - **Global :** Tous les routeurs cherchent à disposer de connaissances globales sur le réseau pour effectuer un routage adaptatif optimal.
- **Solution 1: Apprentissage a posteriori de chemins ("Backward Learning")**
  - **On effectue des expériences** successives de communication : par exemple en inondation.
  - **On apprend des chemins possibles** et leurs coûts pour atteindre un destinataire.
  - **On adopte le meilleur chemin.**
  - **On recommence périodiquement le mode apprentissage** pour prendre en compte des modifications de topologie ou de charge.
  - **Exemple :** Fonctionnement en mode transparent des commutateurs de réseaux locaux (mode transparent).

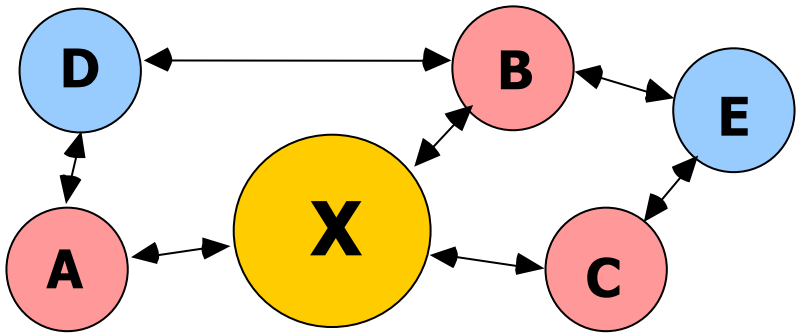
Solution 2 : Arbre des plus courts chemins "Open Shortest Path First”

- **Principe : détermination de l’arbre couvrant des plus courts chemins par chaque routeur.**
  - Pour cela il doit connaître l'état des liaisons de tous les autres routeurs.
  - Il peut alors appliquer un algorithme de plus courts chemins centralisé : pour calculer son arbre de routage et donc ses tables de routage.
- **Fonctionnement par information mutuelle**
  - Chaque routeur envoie à tous les autres l'état de ses liaisons **("link state")** lorsqu'il le juge nécessaire.
  - Chaque routeur répond à des demandes d’état des autres routeurs.
- **Avantages/ Inconvénients**
  - Solution distribuée : bon calcul de routes sur les liaisons connues.
  - Solution plutôt coûteuse en échanges de messages: applicable à un réseau pas trop gros (routage intra-domaine).
- **Exemple: routage OSPF Internet "Open Shortest Path First”, une solution à états de liaisons “link state”.**

Solution 3 : Routage par échange des tables (vecteurs de distance)

- **Principe :** détermination des tables de routages par échange des tables en fait des vecteurs de distances avec les voisins.
- **Nombreuses appellations : "Algorithme de Mac Quillan" ou "Bellman Ford Fulkerson réparti"**
- **Chaque routeur gère deux tables**
  - une table des coûts de transit vers ses voisins immédiats déduites d'infos locales.
  - une table de routage avec les coûts vers tous les autres sites du réseau.
- **Chaque routeur émet périodiquement sa table de routage vers ses voisins et reçoit de ses voisins leurs tables de routage.**
- **Il obtient le coût pour atteindre tous les destinataires en passant par chaque voisin en additionnant:**
  - le coût pour atteindre un voisin immédiat
  - le coût de ce voisin à tous les autres site.
- **Choix des routes :** l'adjacent préféré pour atteindre un destinataire est le voisin de coût minimum (pour atteindre ce destinataire).
- **Exemple : Internet protocole RIP "Routing Internet Protocol".**

Exemple de fonctionnement des vecteurs de distance



Un réseau

Coût Local	A	B	C
X	26	20	18

Coût Distant	D	E
A	64	130
B	56	115
C	45	210

Tables de coûts

Pour aller de X à D ou de X à E

Par A 26 + 64 = 90 26 + 130 = 156

Par B 20 + 56 = 76 20 + 115 = 135

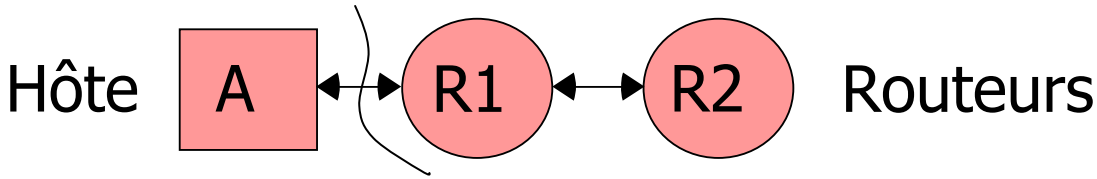
Par C 18 + 45 = 63 18 + 210 = 228

Optimisation

X	A	B	C	D	E
Adjacent	A	B	C	C	B
Coût	26	20	18	63	135

Table de routage

Vecteurs de distances : le problème du comptage vers l'infini



- **Métrique utilisée le nombre de sauts :**
  coût(A-R1) = 1 , coût(R1-R2) = 1 , coût(A-R2) = 2
- **A<->R1** tombe en **panne** : coût(A-R1) = ∞
- **Nouveau calcul des tables en R1 :** R2 indique à R1 qu'il un chemin vers A de longueur 2 (en fait celui passant par R1).
- **R1 considère à tort qu'il a retrouvé un chemin vers A** de longueur coût (R1-R2) + coût(A-R2) soit 1+2=3 par R2
- **Problèmes**
  - **Les paquets de R1 vers A bouclent entre R1 et R2**.
  - **A chaque calcul des tables de routage le coût augmente de 1** (notion de **comptage vers l'infini 'count to infinity')**.

Solutions au problème du comptage vers l'infini

- **Solution de base : Introduire une borne dans le coût des chemins** (notion de durée de vie 'time to live').
  - **Si un chemin a un coût plus élevé que la borne on considère que le chemin est coupé** => on ne retransmet plus sur cette route.
  - **La borne** met fin au bouclage des paquets (réglage de 15/16 à 30/35).
- **Autre solution : Horizon partagé** ("split horizon")
  - **A) Limiter le volume des tables échangées** en ne renvoyant pas une information dans la direction d'où elle vient
    - C1 indique à C2 qu'il a une route vers A.
    - Pour éviter les boucles C2 ne retransmet pas cette route vers C1 (puisqu'elle est connue de C1).
  - **B) Solution dite d'empoisonnement des routes de retour** (RIP, IGRP) ('reverse poison'): au lieu de ne pas retransmettre la route on retransmet un coût infini. Le comportement B) est différent de A) lorsqu'on a plusieurs chemins possibles dans la table.

Conclusion: routages à vecteurs de distances

- **Avantages**
  - **Algorithme complètement réparti, simple à développer.**
- **Inconvénients**
  - **Problème de la fréquence d'échange des tables**
    - **Pour limiter le trafic et les traitements** => Peu d'échanges**.**
    - **Pour que les modifications de topologie soient prises en compte rapidement** => Échanges fréquents.
    - **Apprentissage d'un événement dans tout le réseau** => D étapes d'échanges de tables ou D est le diamètre du réseau.
    - **Exemples de réglage :** Arpanet 0,640 seconde, Internet 30 secondes.
- **Très nombreuses utilisations**
  - **Internet:** Protocoles RIP, EGP "Routing Information Protocol" (dans Internet l'utilisation de RIP diminue depuis 1990 au profit de OSPF).
  - **XNS "Xerox Network System" :** Protocole GWINFO.
  - **Apple talk RTMP :** "Routing Table Maintenance Protocol"
  - **Autres constructeurs** : Novell, 3COM, Ungermann-Bass, Banyan.

Conclusion: Protocoles de routage

- **Problème très important de l'algorithmique des réseaux.**
  - Composant nécessaire dans tous les réseaux.
  - Très nombreuses solutions proposées.
  - Solutions assez souvent retenues : basées sur la construction d'arbres couvrants des plus courts chemins dans chaque routeur.
- **Solutions particulières pour des types de réseaux particuliers.**
  - Routage sur des réseaux à topologie régulière (grille hypercube).
  - Techniques de routage au niveau application pour l'accès aux contenus (réseaux pairs à pairs, tables de hachage distribuées).

Problèmes généraux de réalisation du niveau réseau

Qualité de service et contrôle de congestion

Introduction

- **Spécification d'un contrat de qualité de service**
  - Les obligations temporelles du fournisseur mais aussi du client.
  - Les obligations du fournisseur en terme de pertes de paquets (sûreté).
  - Les outils pour vérifier le respect de ces obligations.
- **Le contrôle de congestion (« Congestion control »)**
  - Le mécanisme global dans un réseau qui permet de respecter en toutes circonstances les objectifs de qualité de service (paramètres temporels, taux de perte).
- **Le contrôle de trafic (« Usage parameter control »)**
  - Le mécanisme à l'entrée du réseau qui permet de détecter les violations du contrat de trafic d'un client.
- **Le contrôle de flux (« Flow control »)**
  - Un mécanisme par rétroaction du destinataire sur l'émetteur pour éviter de perdre des informations par saturation du destinataire.

Phénomènes de congestion

- **Situation de surcharge**: trop de paquets dans le réseau (ou dans une région du réseau, un commutateur donné).
- **Augmentation des délais** de transmission => non conformité à la définition de qualité de service.
- **Pertes de paquets.**
- **Phénomène d'écroulement** : Le réseau ne transporte plus le trafic usager
- **Pourquoi ?**
- **Destruction de paquets** pour désengorger les commutateurs
- **Protocole de traitement de surcharge** (qui implique de plus en plus d'échanges).

Nécessité des mécanismes de QOS et de congestion

- **L'accroissement des ressources ne résout pas seul le problème.**
  - Utilisation à faible charge (relativement aux ressources) => Fonctionnement correct mais sous-réservation des ressources
  - Apparition de surcharges : La congestion peut apparaître lors des surcharges qui sont inévitables sauf contrôle de trafic entrant très strict.
- **Un volume de mémoire important ne résout pas le problème => Pics de trafic de longue durée.**
  - Même si la mémoire est de taille infinie:
  - Temporisateur +retransmission => congestion
- **Des communications très rapide ou des commutateurs rapides ne peuvent solutionner le problème**
  - En fait les tampons peuvent être saturés plus vite.

Classification des techniques de contrôle de congestion

- **Selon la position dans le temps**
  - Traitement préventif de la congestion.
  - Traitement curatif de la congestion.
- **Selon l'endroit ou s'applique le mécanisme**
  - Contrôle du trafic d'entrée d'un usager.
  - Politique de réservation des ressources (gestion des tampons, des capacités des voies)
  - Techniques curatives de traitement de la congestion pour limiter les surcharges (exemple: paquets d'indication de surcharge).

Traitement préventif par limitation du trafic d'entrée

- **Limitation du trafic de chaque hôte**
  - **Selon le contrat** défini au préalable entre le client et le prestataire
  - **Pour éviter une surcharge non connue**.
  - **Également pour découpler le trafic soumis** par chaque usager du débit de sa voie physique de rattachement.
- **Fonctionnement**
  - **Obliger l'usager à respecter** les règles concernant le débit soumis.
  - **Nombreux paramètres possibles**
    - Nombre moyen de paquets par unité de temps.
    - Durée des rafales autorisées au débit maximum.
  - **Le prestataire doit doter chaque accès d'un contrôleur de trafic** d'entrée
    - Exemple pour le débit : technique de seau percé "leaky bucket".

Conclusion : Limitation du trafic d'entrée

- **Avantages**
  - **On évite les comportements inattendus** de certains usagers.
  - **Une technique indispensable** pour limiter les difficultés.
- **Inconvénients**
  - **Si la limitation du volume global entrant est trop importante** (acceptation d'un faible nombre d'usagers)

    => **Sous-réservation** des ressources
  - **Si la limitation est trop légère**

    => **La saturation et la congestion peuvent apparaître.**
  - **N'empêche pas l'installation de la congestion.**
- **Utilisation**
  - **Tous les réseaux à qualité de service : Relais de trames ("Frame Relay"), ATM.**

Traitement préventif de la congestion Allocation des ressources

- **Cas des circuits virtuels**
  - **En fonction du contrat** de service négocié.
  - **Au moment de la construction du circuit**.
- **Réservation dans les commutateurs traversés**
  - **Des ressources** pour l'écoulement correct du trafic (tampons, débit).
  - **Si la réservation est impossible**: le circuit n'est pas accepté.
  - **Si la réservation a pu avoir lieu**: le trafic doit pouvoir s'écouler.
- **Avantage** :
  - **Si la gestion est rigoureuse la congestion est impossible**.
- **Inconvénient** :
  - **Les usagers n'utilisent pas les ressources qu'ils réservent.**
  - **Le fournisseur du service s'il optimise** par la sur-réservation des ressources dont il dispose => implique l'acceptation d'un certain niveau de congestion.

Traitement curatif de la congestion Destruction de paquets

- **Engorgement d'un commutateur**
  - **Insuffisance de mémoire** pour stocker.
  - **Insuffisance de puissance de calcul** et de **débit** des voies.
- **Destruction de paquets**
  - **Incontrôlée** dans les tampons d'entrée.
  - **Selon une politique contrôlée**
    - Après avoir exploité les informations de libération de ressources.
    - **Pour maintenir un trafic "plus prioritaire" au détriment d'un autre** (gestion séparée des files d'attente, des circuits virtuels).
    - **Notion de bit de priorité à la destruction.**
- **Conclusion**
  - **La solution la moins bonne pour des données informatiques** : il faut retransmettre ensuite).
  - **Si la destruction est acceptée ou inévitable**
    - Acceptation du taux d'erreur comparable au taux d'erreurs de transmission.
    - Acceptation d'erreurs récupérables (compensables, qu'on peut corriger).

Traitement curatif de la congestion Paquets de surcharge 'Choke Packets'

- **Principe :** Sur franchissement de **seuil** d'alerte demande aux voisins de limiter ou de suspendre les communications:
  - **Par des paquets spécifiques de surcharge** (trafic en plus)
  - **Par des bits de surcharge** insérés dans les paquets ("piggybacking")

    Exemple: ATM bit EFCN "Explicit Forward Congestion Notification"
  - **L'indication de surcharge :**
    - S'applique à un **CV** ou une **voie** physique
    - Peut **s'arrêter aux voisins** du site surchargé ou être **propagée** j**usqu'aux hôtes générant le trafic** d'entrée.
- **Avantages** :
  - Peut réellement traiter une situation de congestion.
- **Inconvénients** :
  - Ne s'applique qu'à des échanges qui peuvent ralentir
  - En surcharge lenteur de circulation des informations de surcharge.

Traitement curatif de contrôle de congestion basé crédit ou basé débit

- **Sur quel critère traiter la congestion**:
  - Espace mémoire occupée ou débit des paquets sur les voies.
- **Crédits :** Une information qui concerne l'espace mémoire disponible (le nombre de tampons libres).
  - **=> Algorithmes basés crédits** : faire circuler des crédits.
- **Débits :** Une information qui concerne le nombre de paquets ayant circulé par unité de temps dans un passé récent.
  - **=> Algorithmes basés débits** : mesurer le débit effectif et faire fonctionner le CV à ce débit qui est le maximum possible
- **Avantage des solutions basées débit**
  - **Dans le cas des réseaux à haut débit**: l'espace mémoire (très important) reste inoccupé trop longtemps.

Conclusion

Quelques tendances

- **Augmentation des débits disponibles, de la taille des réseaux.**
- **Support des applications multimédia de masse**:
  - Intégration de service : données classiques, téléphone, visio conférence, diffusion télévision, radio.
  - Prise en compte de la qualité de service temporelle.
  - Convergence des réseaux: informatique, téléphonie filaire et mobile.
  - Nouvelles applications : jeux en réseau,
- **Support de la mobilité et des communications radio:**
  - **Routage ad'hoc** : protocoles de routage pour des communautés d'usagers dynamiques auto organisées.
  - **Itinérance ('roaming') :** localisation d'un usager dans l'une des cellules du réseau permettant de l'atteindre.
  - **Continuité ('handover') :** de service lors d'un changement de cellule

Bibliographie : Cours généralités routage/commutation

- A. S. Tannenbaum 'Computer Networks' Prentice Hall.
- Documents en ligne et RFC.

Niveau Réseau
"Network Layer"

Le protocole IP
"Inter-network Protocol"

Introduction.

IP version 4.

IP version 6.

Routage IP et protocoles annexes.

Conclusion.

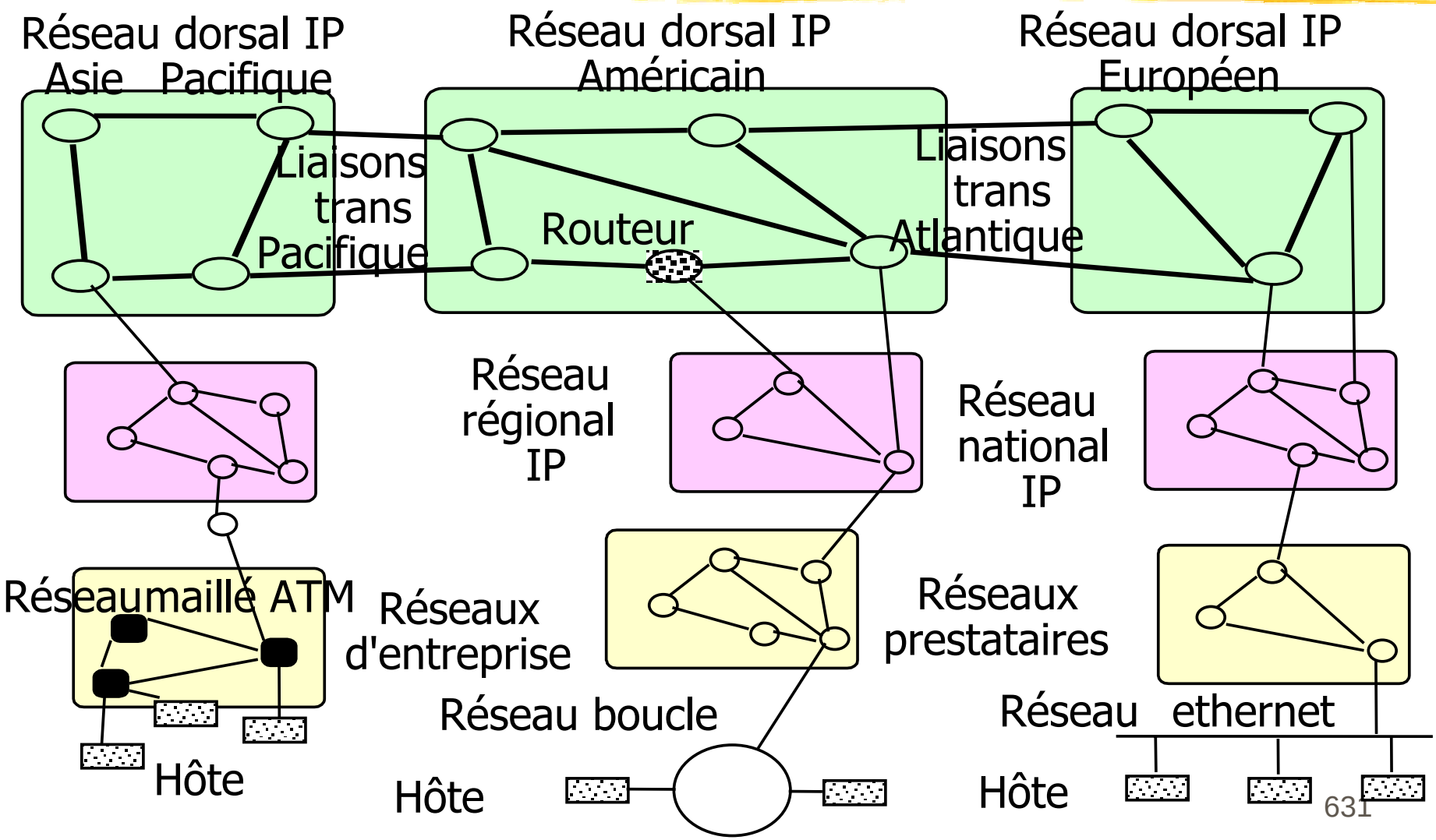
IP

Introduction - Généralités

Objectifs généraux de IP

- **IP : un réseau de réseaux**.
  - Protocole d'interconnexion de réseaux locaux ou généraux.
- **Fonctionnement en mode datagrammes** (pas de circuits virtuels).
- **En version de base pas de qualité de service temporelle**
  - **Fonctionnement au mieux ("best effort")**
  - **Existence de protocoles additionnels** pour la qualité de service.
- **Recherche d'une optimisation globale** des infrastructures de communication disponibles.
- **Robustesse** d'acheminement.
  - Reconfiguration automatique en cas de panne.

IP: un réseau mondial de plus en plus hiérarchisé

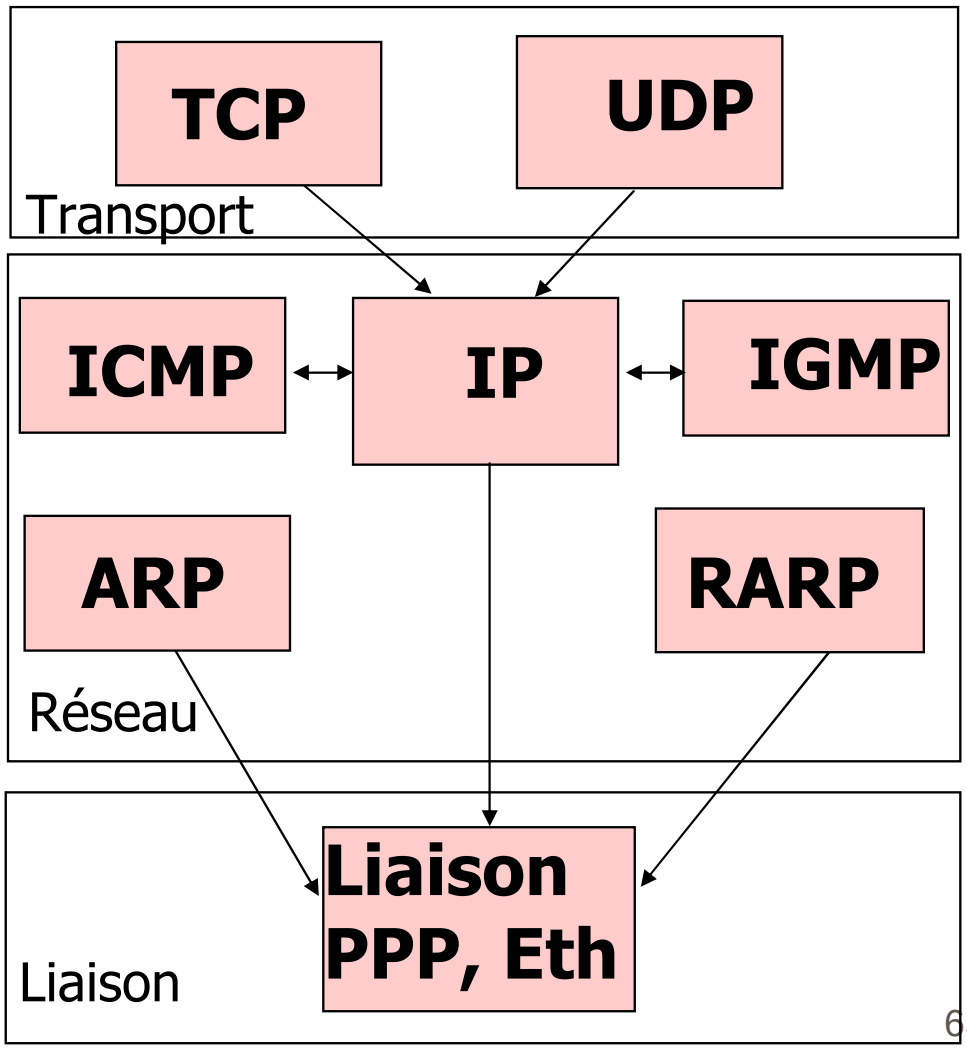


Fonctions réalisées par IP

- **Communications au niveau 3 sans connexion.**
  - Fonction de routage
  - Mode datagramme (pas d'information d'état).
- **Adressage universel.**
  - Assurant l'accès à n'importe quel type d'hôte.
- **Communications sans contrôle d'erreur, de flux, de séquence.**
  - Mode datagramme : envoi de paquets **sans contrôle.** Quelques cas d'erreurs sont détectées (sur l'entête, insuffisance de tampons) IP transmet un paquet ICMP**.**
  - => Contrôle d'erreur, de flux … à la charge de TCP.
- **Fragmentation/Réassemblage.**
  - Adaptation de la taille des datagrammes aux possibilités offertes par les différentes couches liaisons.

La suite des protocoles TCP/IP en version 4

Diagramme des principaux protocoles en version 4



Définition simple des différents modules

- **ICMP : "Internet Control Message Protocol"**
  - Fonctions annexes au routage et traitement d'erreurs.
- **IGMP : "Internet Group Management Protocol"**
  - Gestion des groupes pour la diffusion.
- **ARP, RARP** : **"Address Resolution Protocol" , "Reverse Address Resolution Protocol "**
  - Correspondance d'adresses liaison et d'adresses IP.
- **Couche liaison** (encapsulation des paquets IP).
  - Sur liaison point à point : **"PPP Point to Point Protocol"**
  - Sur réseaux locaux : **Ethernet/DIX, "LLC/SNAP Logical Link Control/Sub Network Access Protocol".**

IP : Historique

- **Travaux sur les réseaux**: Protocole TCP -> Nombreux contributeurs (Article IEEE 1974 TCP Vinton Cerf, Robert Kahn) mais aussi Jon Postel : adoption RFC **IP** (RFC 760 janvier 1980)
- **Protocole IP** : séparé de TCP, Codage à partir de 1978.
- **Différentes améliorations** : **Stabilisation => IP Version 4 (RFC 791** septembre 1981**).**
- **Diffusion significative :** à partir du début des années 1980.
- **Grande importance du couple UNIX-TCP/IP** : ensemble cohérent permettant de faire du réseau à coût raisonnable (UNIX Berkeley sur DEC/VAX 1983 - Université de Californie).
- **Développement des protocoles annexes**: protocoles de routage, support de la qualité de service, de la sécurité …..
- **Restructuration importante** de l'adressage pour suivre le développement mondial: **toujours en cours IP V6** (1995).

Le contrôle de l'Internet : Principaux organismes

- **ISOC "Internet Society"**
  - Organisation principale chargée de la croissance, de l'évolution technique, des aspects sociaux, politiques, ...
- **IAB "Internet Architecture Board"**
  - Définition de l'architecture du réseau et de ses protocoles. Arbitrage des conflits.
- **IESB "Internet Engineering Steering Group"**
  - Administre les normes sur proposition de l'IETF.
- **IETF "Internet Engineering Task Force"**
  - Définition, expérimentation des protocoles, groupes de travail par domaines.
  - **RFC "Request For Comments" :** normes de l'Internet.
- **IRTF "Internet Research Task Group"**
  - Recherche à long terme.

Le contrôle de l'Internet : Gestion des noms, adresses et paramètres

- **IANA "Internet Assigned Number Authority"**
- **Puis ICANN "Internet Corporation for Assigned Names and Numbers"**
  - **Organisme chargé de l'affectation** des adresses, mot-clés, paramètres, ... pour l'ensemble des protocoles Internet
  - **Politique de gestion** : adresses, noms de domaines, définition des MIB ("Management Information Base")… etc
- **Délégation de certains responsabilités** (espace d'adresses)
  - **Amérique : INTERNIC** "Internet Network Information Center".
  - **Europe : RIPE NCC** "Réseaux IP Européens Network Computing Center"
  - **Asie : APNIC** "Asia Pacific Network Information Center".

IP

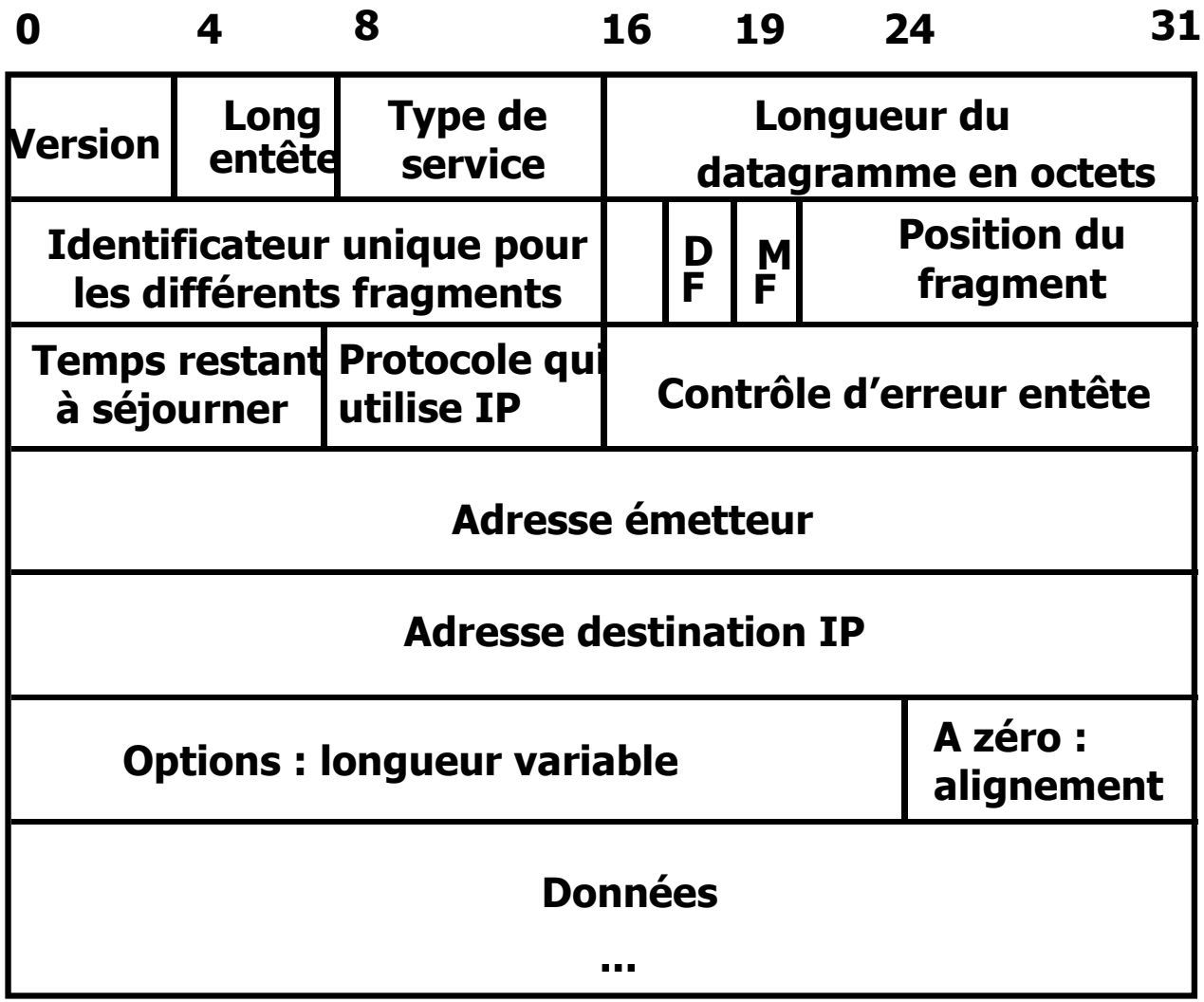
Chapitre I
Le protocole IP en version 4

Structure des datagrammes
Fragmentation
Adressage

IP Version 4

I.1
Structure des datagrammes

Format du datagramme IP V4



Convention :
Transmission grand boutiste 'big endian'
Le bit 0 est envoyé en tête.

Détail des différents champs (1)

- **Numéro de version IP** "IP version number" : **4 bits** Ici IP v4
- **Longueur de l'entête** "IP Header Lenght" : **4 Bits**

  Longueur de l'entête en mots de 32 bits (Min 5 -> Max 15)

  Option: au plus 40 octets (entête standard 20 + option = 60).
- **Type de service** TOS "Type Of Service") : **8 bits**

  Qualité de service

00	**01**	**02**	**03**	**04**	**05**	**06**	**07**
Precedence			D	T	R	M	

- 3 bits ("Precedence") Priorité 0 normal à 7 contrôle réseau
- 4 bits indicateurs ("Flags" D T R M) + Un bit inutilisé

D "Delay" minimiser le délai T "Throughput" maximiser le débit

R "Reliability"max de fiabilité M "Monetary" min de coût

Redéfinition du TOS : QOS Multimédia => Diffserv.

Détail des différents champs (2)

- **Longueur datagramme (16 bits)** "Total lenght"

  . Longueur totale du datagramme en octets incluant entête et données => Longueur au maximum 65535.
- **Identificateur unique (16 bits)** "Ident field"

  . Valeur entière utilisée pour regrouper les différents fragments d'un message fragmenté.

  . Un datagramme peut être fragmenté à l'émission: un routeur qui détecte qu'un datagramme est trop long pour la voie de sortie le fragmente.
- **Ne pas fragmenter (1 bit)** DF "Don't Fragment"

  . Le datagramme même ne doit pas être fragmenté.

  . Le destinataire ne peut traiter les fragments.

  Ex : téléchargement de code en une fois.

Détail des différents champs (3)

- **Dernier fragment MF** "More Fragment" **: 1 bit.**

  . Indique le dernier fragment d'un message fragmenté (0)

  . ou un fragment courant (1).

- **Position du fragment** "Fragment Offset" **: 13 bits.**

  . Détermine la position d'un fragment dans un message (8192 fragments possibles).

  . Chaque fragment sauf le dernier comprend un nombre entier de groupes de 8 octets.

- **Temps restant à séjourner TTL** "Time To Live" : **8 bits**

  . Ancienne version (RFC 791) :Mesure du temps de séjour dans le réseau en secondes depuis l'émission (255 s max).

  . Actuellement: Initialisé à une valeur entière (par ex 30).

  Décrémenté par chaque routeur => Le paquet est détruit lorsque le compteur passe à zéro (pour éviter les boucles).

Détail des différents champs (4)

- **Protocole utilisateur "Protocol" : 8 bits**

  . Protocole qui utilise IP. Nombreuses valeurs normalisées pour le démultiplexage des paquets entrants

  . Exemples ICMP=1, TCP=6, UDP=17

- **Contrôle d'erreur entête** "Header Checksum" **: 16 bits**

  . Contrôle d'intégrité sur l'entête du paquet.

  . Un paquet d'entête erronée est détruit pour éviter des erreurs de délivrance.

Méthode de calcul

- L'entête est considérée comme une suite de mots de 16 bits.
- On fait la somme des mots de 16 bits en complément à 1.
- On prend le complément à 1 du résultat.

=> A chaque traversée de commutateur: comme il n'y a que la zone TTL qui change de un, le calcul de la nouvelle somme de contrôle est simple.

Détail des différents champs (5)

- **Adresse source** "Source address" **: 32 bits**

  . Adresse IP de l'émetteur.

- **Adresse destination "**Destination address" **: 32 bits**

  . Adresse IP du destinataire.

- **Données "Data"**

. Zone de donnée utilisateur d'une taille maximum de 64 K octets.

Zone des options

- **Utilisée pour spécifier des compléments de protocole** qui n'ont pas à être toujours présents.
- **Utilisation des options** : beaucoup moins forte en IPV4 qu'en IPV6.
- **Longueur variable** : de 4 à 40 octets.
- **Alignement sur des frontières de mots de 32 bits** => Bourrage si le dernier mot n'est pas complètement utilisé.
- Les options **ne sont pas toutes traitées** par certains routeurs.

Les cinq classes d'options en IPV4

- **Protocoles de sécurité : IPSEC** "IP Security"
- **Enregistrement** de la route suivie "Record Route".
- **Enregistrement de la route et estampillage** par la date de traversée de tous les routeurs "Record and Timestamp".
- **Routage par la source non contraint** "Loose Source Routing" : définition d'une liste partielle de routeurs devant être visités.
- **Routage par la source contraint** "Strict Source Routing" : liste stricte des routeurs à visiter.

IP Version 4

I.2
Fragmentation
(ou segmentation)

Solutions pour la fragmentation (Segmentation)

- **Objectif: adapter la taille des datagrammes** à la taille maximum des trames de liaison (taille médium).
- **MTU** ('Maximum Transfer Unit') : pour une voie donnée la taille maximum des trames (souvent 1500 octets Ethernet).

Solutions de fragmentation non retenues en IP V4.

- **Fragmentation transparente 1 = fragmentation et réassemblage pour chaque saut** : le routeur émetteur sur une voie fragmente si nécessaire et le routeur destination réassemble s'il y a eu fragmentation.
- **Fragmentation transparente 2 = fragmentation de bout en bout** : on ne fragmente qu'à l'entrée du réseau et on ne réassemble qu'à la sortie => implique l'apprentissage du MTU de chemin (path MTU) plus petit MTU d'un chemin.

Fragmentation en IP V4 : Une fragmentation non transparente

- **Pour un réseau donné** (une voie de communication) un émetteur (ou un routeur) fragmente un datagramme si nécessaire et les **fragments poursuivent** jusqu'au destinataire qui est le seul à réassembler.
- **Il peut donc y avoir plusieurs fragmentations successives sans réassemblage** (sauf au terme).

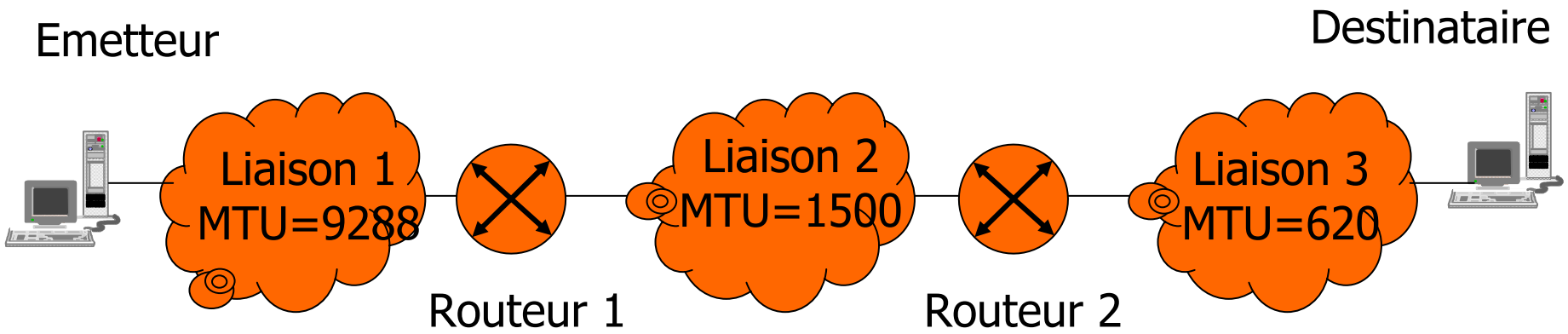
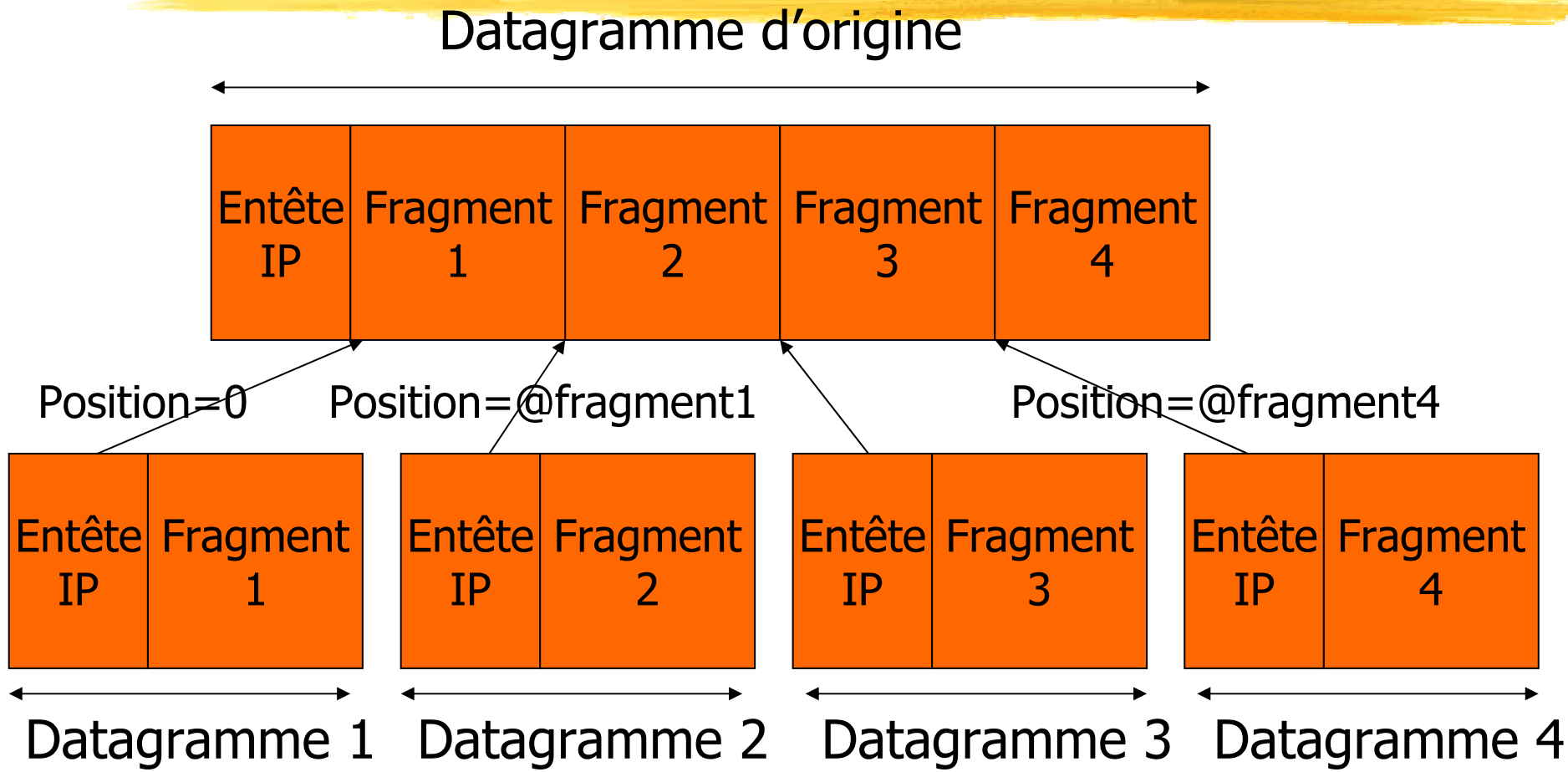


Schéma général de la fragmentation



Suite des datagrammes associés à une fragmentation :
la position (offset) permet de reconstruire le datagramme d'origine

Fragmentation IP V4 : exemple d'école de fonctionnement

- **Datagramme à fragmenter**

I=3204	P=0	M=0	L=41	Message A Transmettre

- **Après fragmentation pour un MTU=28**

I=3204	P=0	M=1	L=28	Message

I=3204	P=1	M=1	L=28	A transm

I=3204	P=2	M=0	L=25	ettre

- **Entête IPV4** : informations pour la fragmentation
  I : Identificateur de fragment. P : Position d'un fragment dans le datagramme origine (offset). M : Indicateur dernier fragment ('more'). L : Longueur du datagramme (avec entête 20 octets).
- **Attention** : P la position ('offset') est en multiple de huit octets.

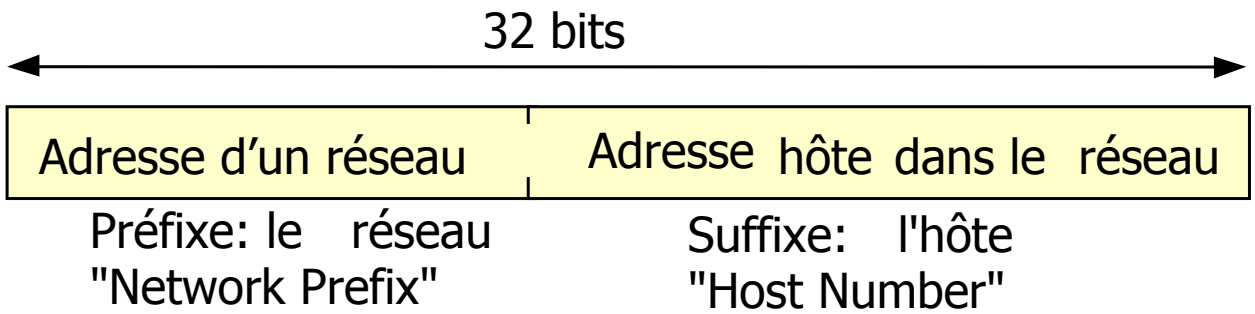
IP Version 4

I.3
Adressage

A) Par classes (‘Classfull’)
B) Avec Sous réseaux (‘Subnetting’)
C) Avec masque variable (‘Variable lenght mask’)
D) Sans classes (‘Classless’)
E) Mécanismes additionnels

Introduction : Adressage dans IP un réseau de réseaux

- **Toute machine connectée à IP appartient à un réseau**
  - Notion d'adresse de réseau et d'adresse hôte à l'intérieur d'un réseau



- **Cadre général de l'adressage IPV4**

  - **Adressage uniforme au moyen d'adresses universelles sur 32 bits** - 4 octets.

  - **Notation "Décimale Pointée"** ('dotted decimal') 4 octets d'une adresse : a.b.c.d Exemple d'adresse: 192.200.25.1

  - Transmission des adresses : **grand boutiste (big endian)**

Evolution de l'adressage IP

- **Améliorations** successives pour faire face:

  - Demande d'adresses IP à satisfaire (croissance très rapide).

  - Nombre de réseaux IP également en croissance : taille des tables de routage.

- **Solution : Hiérarchiser de plus en plus** l'adressage en relation avec le routage => Quatre étapes successives.
  - **Hiérarchisation à deux niveaux :**

  Adressage par classes 'Classfull'
  - **Hiérarchisation à trois niveaux :**

  Adressage IP par sous réseaux 'Subnetting'
  - **Hiérarchisation complète à n niveaux de l'adresse d'hôte:**

  Adressage IP avec masque variable VLSM ('Variable lenght Subnet mask')
  - **Hiérarchisation complète à n niveaux de l'adresse IP**

  Adressage IP sans class CIDR ('Classless Inter Domain Routing').

Utilisation des adresses IP dans le routage

- **Table de routage (ensemble de routes)**

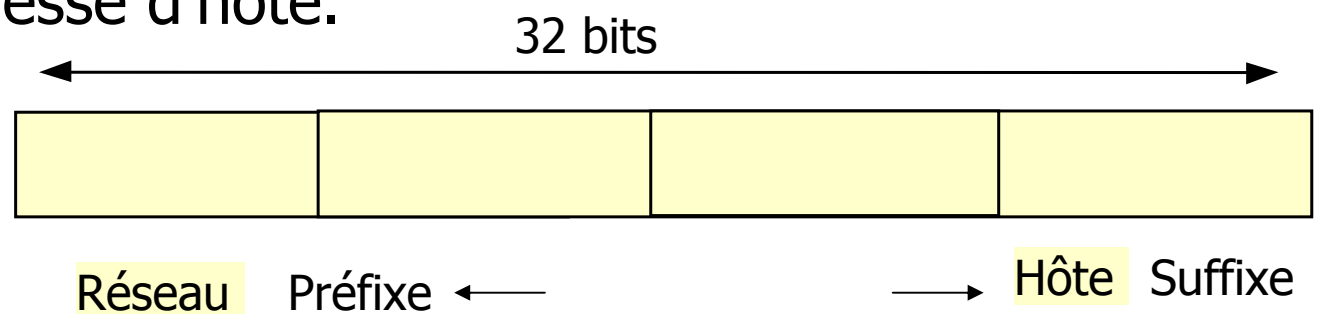
Adresse destination	Différents champs	Direction d'envoi
163.173.0.0 (réseau)	xxxxxxxxxxxxxxx	/dev/eth0
…….	…….	…….

- **Datagramme à router**
  - Adresse destination e.g.h.i (par ex 136.173.36.60).
  - A partir du préfixe détermination de l'adresse réseau par application d'un masque (adresse avec un préfixe de bits à 1 par exemple 16 bits à 1 soit 255.255.0.0)

    => adresse réseau 136.173.0.0.
  - Comparaison de l'adresse réseau de destination avec les destinations des différentes routes dans la table.

A) L'adressage IP V4 par classes "IP Classfull" RFC 791 (1981)

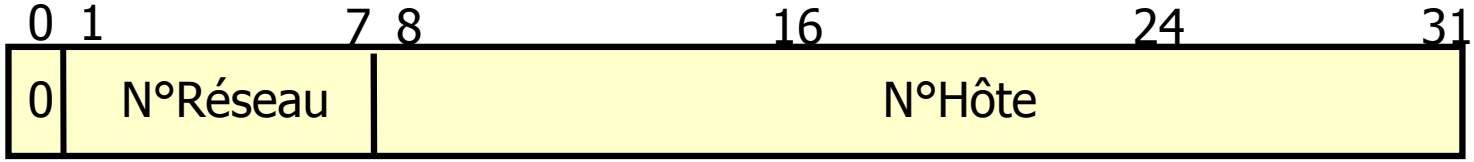
- **Hiérarchisation de base des adresses avec deux niveaux**
- **Réseaux de types et de tailles très différentes**: idée de distinction de trois classes A, B, C selon les tailles de réseau => Trois frontières différentes entre adresse de réseau et adresse d'hôte.



- **Une répartition des adresses entre les trois classes** qui permet automatiquement de déterminer la classe (la taille du préfixe) => donc de trouver l'adresse du réseau d'appartenance (par analyse de l'octet de fort poids).

Classe A : Grands réseaux

- Préfixe sur 8 bits, suffixe sur 24 bits.
- 126 Réseaux de 16777214 = 2**24 -2 hôtes.
- La moitié de l'espace d'adressage.



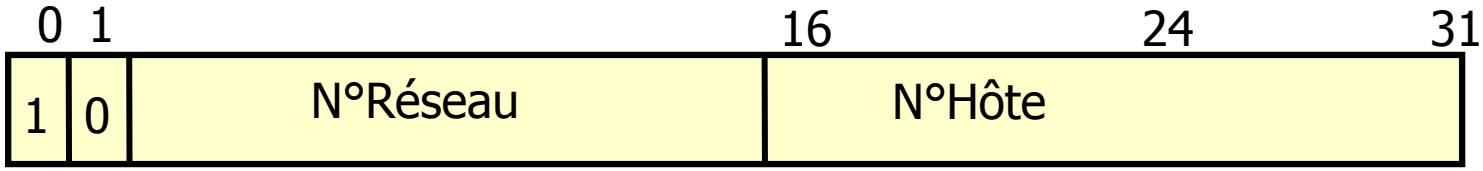
N°de Réseau: a de 1 à 126 (0 et 127 sont réservés).

N°d'hôte: a.0.0.0 et a.255.255.255 réservés.

- Plan d'adressage réservé aux très grands groupes => gestion stricte.

Classe B : Réseaux moyens

- Préfixe sur 16 bits, suffixe sur 16 bits.
- 16384 Réseaux de 65534 = 2**16 -2 hôtes.
- Le quart de l'espace d'adressage.



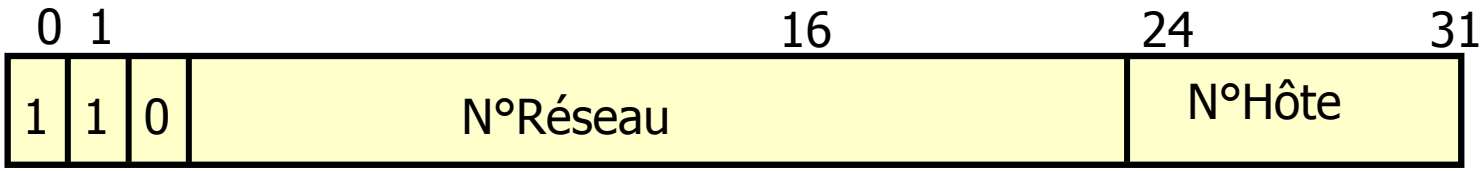
N°de Réseau: a.b de 128.0 à 191.255.

N°d'hôte: 1 à 65534 (a.b.0.0 et a.b.255.255 réservés)

- Plan d'adressage pour entreprises moyennes => gestion laxiste au départ.

Classe C : Petits Réseaux

- Préfixe sur 24 bits, suffixe sur 8 bits.
- 2097152 Réseaux de 254 = 2**8 - 2 hôtes.
- Le huitième de l'espace d'adressage.



N°de Réseau: a.b.c de 192.0.0 à 223.255.255

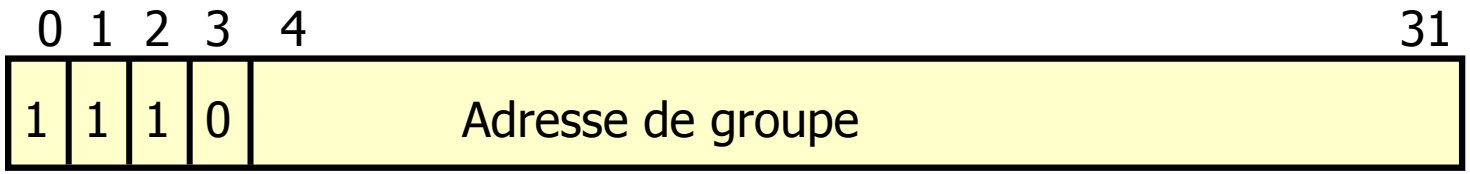
N°d'hôte: 1 à 2544 (a.b.c.0 et a.b.c.255 réservés)

- Plan d'adressage peu demandé au départ

=> utilisation avec la croissance du réseau.

Adresses de classe D : Diffusion sur groupe ("Multicast" IP)

- **Préfixe 4 bits 1110**, suffixe 28 bits: identifiant de groupe (adresse) de diffusion.
- Adresses de 224.0.0.0 à 239.255.255.255



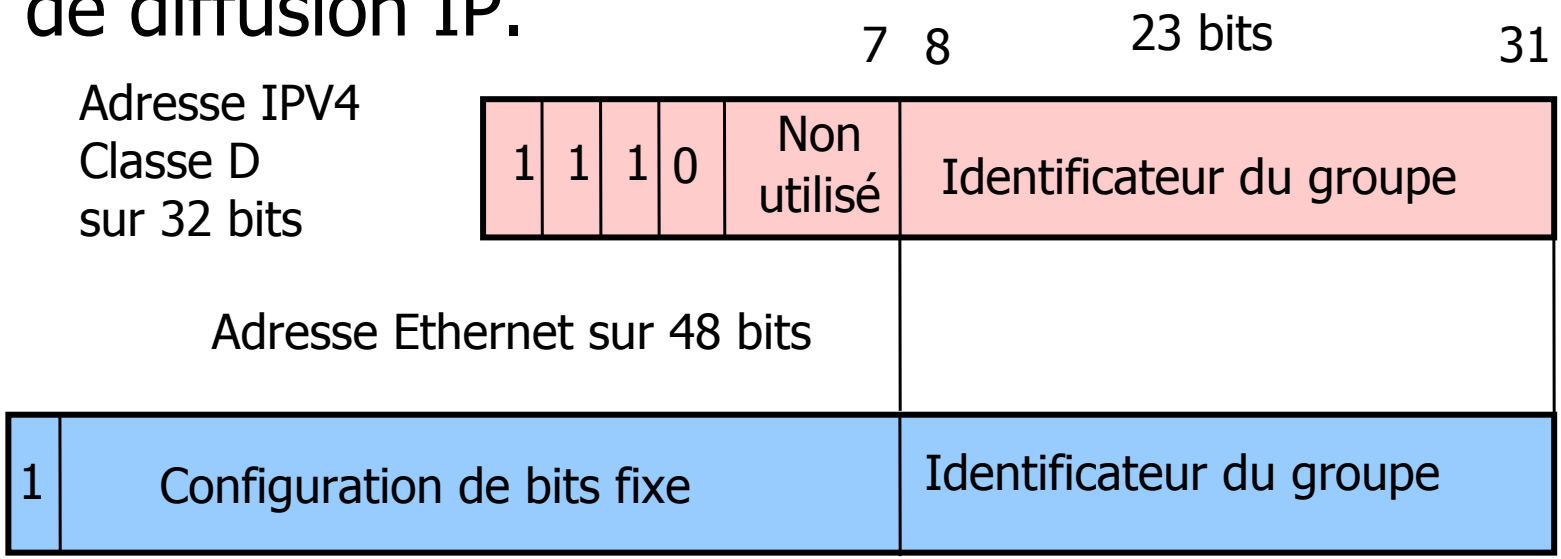
- **Groupes permanents 224.x.y.z (224/8)**

  224.0.0.2 tous les routeurs d'un sous-réseau

  224.0.1.1 groupe "Network Time Protocol"
- **Autres groupes** non permanents : 225 à 239.
- **Pour mémoire classe E** : Réservée Préfixe 11110

Adresses de classe D : Diffusion sur IP et sur Ethernet

- **Transformation** d'une adresse de groupe IP en une adresse de groupe Ethernet.
- **Pour déterminer automatiquement** une adresse de diffusion Ethernet à partir d'une adresse de diffusion IP.



Adresses particulières IPV4 : Adresses point à point (RFC 1340)

- **Adresse point à point ('unicast'):** atteindre un seul destinataire.
- **Adresse destination dans un réseau : une adresse de destination dans une table de routage (atteindre un hôte dans un réseau)**

  - On note le préfixe adresse de réseau suivi de 0 en partie hôte (xyz.00).
  - Autre notation la notation a.b.c.d/n (/n indique un préfixe sur n bits)

    Exemple: a.b.0.0 <=> a.b.c.d/16

  - 0.0.0.0 : L'Internet => l'adresse **destination** par défaut (atteindre un hôte qui se trouve dans l'Internet).
- **Adresse destination moi-même (l'hôte courant): 127/8**

  - 127/8 Comme adresse de destination le même hôte (pour permettre à deux utilisateurs sur le même site de communiquer par IP).

  - Adresse de rebouclage "Loopback"

  - Toutes les adresses classe A "127.a.b.c" sont affectées à cette fonction.

  => Utilisation habituelle de l'adresse : 127.0.0.1 ("localhost").
- **Adresse source moi-même (l'hôte courant): 0.0.0.0 ou 000.xyz**

  - **0.0.0.0 : Adresse source** d'une station qui ne connaît pas son adresse (utilisable également 000.xyz l'hôte xyz dans son réseau).

Adresses particulières v4 : Adresses de diffusion générale

- **Idée de diffusion générale ('Broadcast') :** atteindre tous les hôtes d'un réseau IP donné.
- **En IPV4 pour construire une adresse de diffusion : mettre** des 1 partout dans la partie adresse hôte.
- **Cas de l'adressage par classe : Adresses destination: a.255.255.255 , a.b.255.255 et a.b.c.255:**

  Diffusion à tous les hôtes du réseau a.0.0.0 (classe A) ou a.b.0.0 (classe B) ou a.b.c.0 (classe C)
- **Cas particulier : Adresse dest 255.255.255.255**

  - Idée naturelle au départ diffusion à tout l'Internet

  => abus.

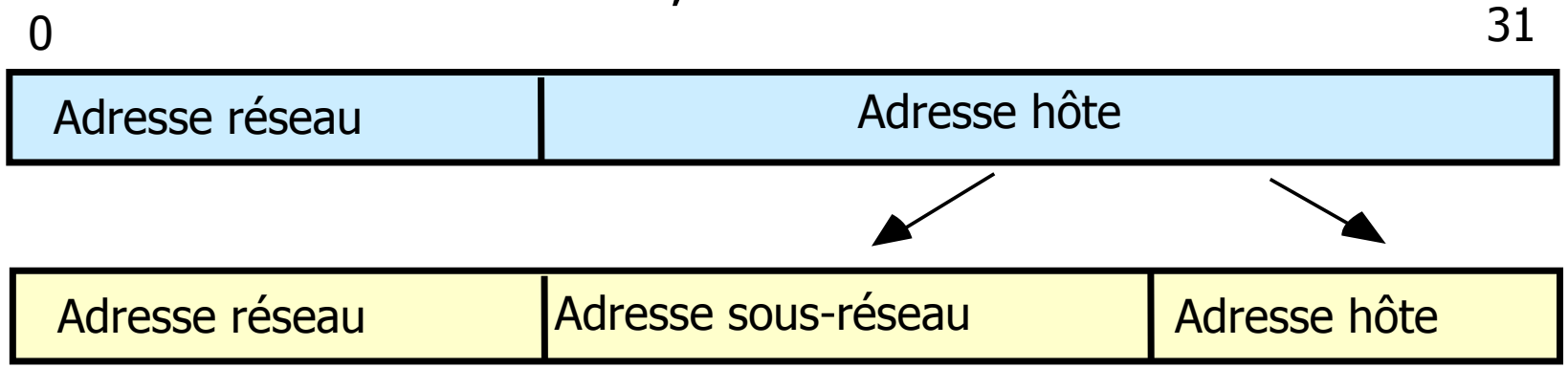
  - Ensuite diffusion limitée au sous-réseau de l'hôte émetteur (non délivré hors du contexte local).

Conclusion : Adressage IPV4 par classes

- **Gaspillage d'adresses dans les années 1980.**
  - L'espace d'adressage paraissant très suffisant,
  - Le réseau était confidentiel
  => Les adresses ont été distribuées sans prévoir.
- **Les besoins exprimés par les entreprises** sont souvent supérieurs à la classe C sans justifier la classe B
  Si attribution de plusieurs classes C : gonflement des tables de routage **=> Attribution de classe B**
- **L'adressage IPV4 par classes sur 32 bits** (4 294 967 296 adresses) **est devenu tout à fait inadapté**
- **Mais les attributions anciennes ont été préservées** dans les plans d'adressages ultérieurs.

B) Adressage par sous-réseaux "IP Subnetting" RFC 950 (1985)

- **Hiérarchisation à trois niveaux:**
  - Possibilité offerte de structurer l'espace d'adressage interne à un réseau de classe A, B ou C en deux niveaux
  - Trois notions : réseau, sous-réseau et hôte.



- **Pour quoi faire** :
  - Mieux structurer un espace d'adressage interne.
  - Sans impact sur l'Internet mondial.
  - Eviter des demandes de blocs d'adresses

Adressage par sous-réseaux : Notion de masque ("Subnet Mask")

- **Souplesse souhaitée**: La frontière entre adresse sous-réseau et adresse d'hôte est variable selon les besoins de l'entreprise (définie par l'administrateur du réseau).
- **Nécessité de fournir** le découpage retenu à chaque machine d'un sous réseau et aux routeurs.
- **Le masque :** permet le filtrage des adresses destination pour trouver l'adresse du sous-réseau d'appartenance.
- **C'est une configuration de bits à 1 que l'on applique en et logique** sur une adresse IP pour sélectionner la partie adresse réseau + sous réseau
- **Exemple** un réseau de classe B : 135.28/16
  - On souhaite le découpage de l'espace interne 10 bits pour l'adresse de sous-réseau et 6 bits pour l'adresse d'hôte :
  - Valeur du masque : **255.255.255.192 ou en notation de** préfixe étendu **/26** (ou encore en héxadécimal 0xFFFFFFC0).

Conclusion IPV4 et les sous-réseaux

Avantages

- **Les tables de routages de l'Internet ne croissent pas en taille** (seuls les routeurs internes gèrent les sous-réseaux)
- **L'espace d'adressage privé est mieux géré** (lors de la création de nouveaux réseaux on évite de demander des adresses).
- **Si un réseau modifie sa structure interne il n'est pas nécessaire de modifier les routes** dans l'Internet

Inconvénients

- **Il faut gérer le masque** en plus de l'adresse.
- **On ne définit qu'une seule façon de hiérarchiser** les adresses : **rigidité du découpage** (un seul pour toute l'entreprise => difficile à changer).

C) Masques de longueur variable VLSM Variable Length Subnet Mask

- **Besoin: créer des sous réseaux de taille différente.**

Exemple

- Classe B 135.8.0.0/16 découpé par le masque 255.255.254.0 ou /23 (soit 2**7 = 128 sous-réseaux de 2**9 – 2 = 510).
- Il se créé un nouveau sous_réseau de 15 hôtes (extension prévisible à 50).
  - Si on lui attribue une adresse de sous-réseau /23 on va perdre environ 500 adresses.
  - Il serait par contre très intéressant de lui attribuer une adresse /26 d'un sous réseau de 64 – 2 = 62 hôtes.

- **La solution : VLSM Variable Length Subnet Mask (RFC 1009 en 1987) :** masques de taille variable.

Problèmes posés par VLSM : 1) Gestion des masques

- **Chaque sous-réseau** possède sa propre **taille.**
  - Pour déterminer **correctement le numéro de réseau** quelque soit sa taille.
  - **Le protocole de routage interne doit utiliser un masque** (un préfixe étendu) différent pour chaque sous-réseau
  - **Il doit transférer ces masques dans chaque route.**

=> **Modifier les protocoles de routage**

- **RIP V2** ('Routing Information Protocol' RFC1388) La version 2 permet de déployer VLSM.
- **OSPF** ('Open Shortest Path First')

Problèmes posés par VLSM : 2) Correspondance la plus longue

- **Recherche de "correspondance la plus longue"** (‘Longest Match based forwarding algorithm’)
  - Au cas ou plusieurs routes sont dans une table,
  - La route de plus long préfixe est la plus précise
- **La route de plus long préfixe doit être sélectionnée et utilisée.**
- **Exemple :** datagramme vers l'hôte 136.1.6.5 avec 3 routes vers les destinations suivantes :
  - 136.1.0.0/16 : 10001000 00000001
  - 136.1.4.0/22 : 10001000 00000001 000001
  - 136.1.6.0/23 : 10001000 00000001 0000011

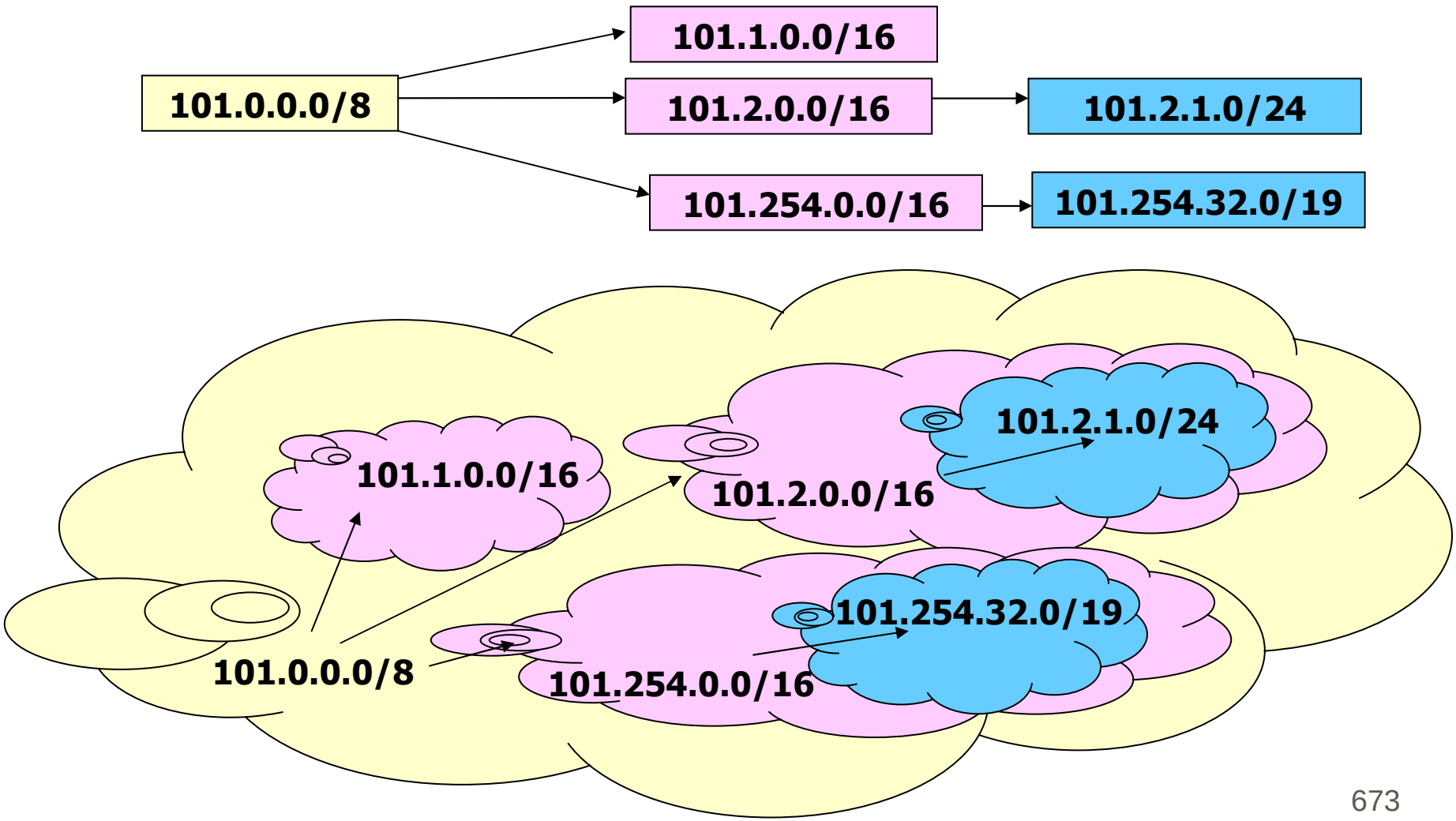
  =>Les trois routes conduisent au but

  =>Le routeur choisit la route 136.1.6.0/23.

Problèmes posés par VLSM : 3) Agrégation des routes

- **Déploiement d'un réseau VLSM => Pour l'agrégation des routes les adresses doivent être assignées 'topologiquement '.**
  - Les blocs d'adresses sont découpés hiérarchiquement.
  - Les blocs d'adresses sont attribuées selon la topologie du réseau
- **On réduit la quantité d'information dans les tables de routage** => on peut agréger en une seule route, les routes pour l'ensemble des blocs contenus dans un bloc destination.
- **On diminue le temps de recherche en table** => amélioration des performances du routage.

Exemple de gestion d'adresse avec agrégation 'topologique' en VLSM



Conclusion : IPV4 avec VLSM

Avantages

- **L'utilisation de plusieurs masques permet un usage plus efficace de l'espace d'adressage attribué à une organisation** : il n'est pas nécessaire de se conformer à la taille unique des sous-réseaux.
- **On réduit le volume des tables nécessaires au routage** au niveau dorsal ('backbone') d'une organisation.

Inconvénients

- **Nécessite l'adaptation des protocoles de routage** pour échanger les masques: RIPV1 -> RIPV2
- **Ne permet de structurer correctement que le domaine d'adresse privé** d'une organisation.

D) Routage sans classe : CIDR 'Classless Inter Domain Routing'

- **Problème récurrent après 1990 (web) en IP v4:**
  - Saturation de l'espace d'adressage et croissance de la taille des tables de routage (aux plus haut niveaux).
- **Solution : Hiérarchisation complète des adresses V4.**

=> Extension de l'approche VLSM à tout l'espace d'adressage de l'Internet.

=>Suppression des frontières établies par l'adressage en classes (classless)

- **Prolongation importante de l'adressage V4.**
  - En améliorant l'utilisation des adresses encore disponibles.
  - En diminuant le volume des tables de routage par agrégation des routes.
- **Solution CIDR : RFC1517 à 1520 (1993).**

Contraintes pour le déploiement de CIDR

Hôtes et routeurs doivent supporter l'adressage CIDR et ses conséquences

- **Mêmes conséquences** que VLSM.
- **Les adresses de destination doivent être échangées** par les protocoles de routage avec leur préfixe (qui peut être de taille quelconque).
- **Les routeurs doivent implanter** un algorithme de "correspondance la plus longue".
- **Les adresses doivent être distribuées** sur une base topologique pour agréger les routes.

Distribution des adresses IP dans l'adressage sans classe

- **Attribution d'adresses par blocs** dont la taille est toujours sur **n bits soit 2^n adresses à chaque fois.**
- **L'utilisation d'un bloc d'adresses libres** de n bits doit correspondre à une adresse de réseau valide sur 32-n bits.
- **Notation en CIDR :**
  - Un bloc d'adresses en CIDR: 212.37.24.160/27
  - Utilisable comme une adresse de réseau:

    L'adresse de réseau en binaire :

    11010100.00100101.00011000.101 | 00000
  - Le masque comporte 27 bits à 1 en tête, et 5 bits à 0 à la fin.

    11111111111111111111111111 | 00000
  - On peut donc aussi noter le masque en notation décimale pointée :

    11111111.11111111.11111111.111 | 00000

    Soit : 255.255.255.224

Exemple de distribution

- **Construction d'un nouveau réseau IP** : comprenant environ 2000 adresses.
- **On doit attribuer un bloc de 2**11** : soient 2048>2000 adresses =>11 bits adresse hôte masque /21.
- **On dispose du bloc libre :** 194.16.32.0/19

  => On peut attribuer les blocs
  - Réseau (1) 194.16.32.0/21 ou 11000010 00010000 00100|000 00000000
  - Réseau (2) 194.16.40.0/21 ou 11000010 00010000 00101|000 00000000
  - Réseau (3) 194.16.48.0/21 ou 11000010 00010000 00110|000 00000000
  - Réseau (3) 194.16.56.0/21 ou 11000010 00010000 00111|000 00000000
  - Exemple: réseau (2) Première adresse d'hôte utilisable 194.16.40.1

    Dernière adresse 194.16.47.254
- **On ne peut pas faire d'autre choix** : car ce ne seraient pas des adresses de réseaux avec préfixe /21 sur 21 bits et suffixe sur 11 bits.

Application de CIDR : distribution des adresses de classe C restantes

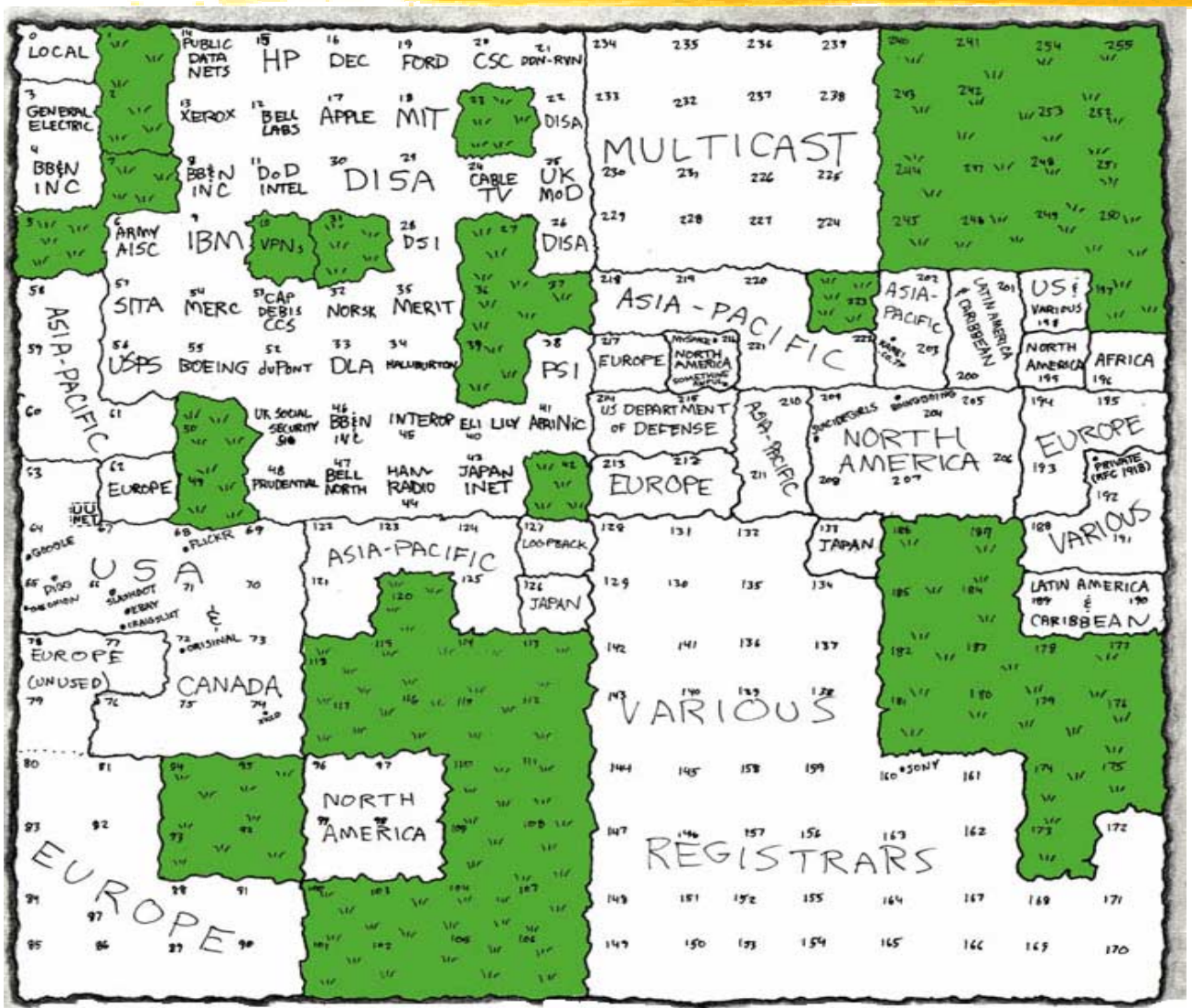
- **Adresses restantes** : dans la classe C (peu de demandes).
- **Solution d'administration**: séparer les classe C restantes en quatre catégories administrées par continent.
  - 194.0.0.0 - 195.255.255.255 Europe RIPE
  - 198.0.0.0 - 199.255.255.255 Amérique nord et sud ARIN
  - 200.0.0.0 - 201.255.255.255 //
  - 202.0.0.0 - 203.255.255.255 Asie Pacifique APNIC
- **Distributions indépendantes** par région de blocs de taille quelconque aux FAI.
- **Agrégation de routes**: une adresse 194.x.y.z doit être envoyée sur un routeur européen.

Conclusion IPV4 avec CIDR

- **CIDR alloue efficace des adresses v4**
  - CIDR permet de **coller** assez finement aux demandes.
  - **Récupération** d'anciennes adresses A, B ou C.
  - Un prestataire Internet 'ISP' **attribue librement** ses adresses.
  - La découpe peut opérer à tous les niveaux.
- **CIDR permet d'agréger les routes à tous les niveaux**
  - **Contrôle de la taille** des tables de routage.
  - **Facilite l'administration** des routeurs.

- **CIDR présente les inconvénients de la hiérarchisation**: Si une organisation souhaite changer de prestataire sans changer d'adresse on doit créer une route d'exception ce qui est coûteux (autre solution voir plus loin NAT).

Carte de l'Internet : occupation de l'espace IPV4 en 2006



Adressage IP Version 4

E) Mécanismes additionnels pour l'adressage (économiser et faciliter l'administration)

1. Liaisons dénumérotées
2. Adresses publiques et privées
3. Traduction d'adresses (NAT)
4. Distribution d'adresses (DHCP)

E1) Liaisons dénumérotées (RFC 1812)

- **Toute carte réseau est identifiée par une adresse IP unique.**
- **Pour une liaison point-à-point, il faut attribuer un numéro de réseau pour une voie qui ne contient que deux interfaces** => perte d'adresses IPv4.
- **Solution : Notion de liaison point-à-point dénumérotée et de routeur virtuel.**
  - On supprime les adresses des interfaces réseau pour une liaison dénumérotée en contradiction avec la notion de route (adresse IP à atteindre "next hop")
  - Les deux routeurs situés aux deux extrémités de la liaison sont des demi routeurs qui forment un seul routeur virtuel (la liaison point à point est en fait interne au routeur virtuel).
- **Avantages** : gain de deux adresses, gestion simplifiée
- **Inconvénients** : on ne supporte pas les cas compliqués à plusieurs routeurs, les routeurs virtuels sont complexes et non standardisés.

E2) Adresses publiques et adresses privées (RFC 1918)

- **Les organisations qui veulent créer un Internet privé peuvent utiliser sans demande les adresses réservées:**
  - 10/8 (10.0.0.0 à 10.255.255.255)
  - 172.16/12 (172.16.0.0 à 172.31.255.255)
  - 192.168/16 (192.168.0.0 à 192.168.255.255)
- **Les adresses privées ne sont routées que dans les réseaux privés (non routées dans l'Internet mondial).**

Avantages

- On évite ainsi **beaucoup de demandes d'adresses.**
- On a **moins de risque d'une utilisation 'sauvage' d'adresses publiques** dans des réseaux privés.

Inconvénient

- **On ne peut pas communiquer avec l'Internet mondial.**

E3) Traducteurs d'adresses IP : NAT Network Address Translation RFC 1631

- **Motivation : l'économie des adresses IP mais aussi :**
  - **Une entreprise ayant créé un Internet privé** (RFC 1918) souhaite avoir ensuite accès à l'Internet mondial.
  - **Une entreprise souhaite cacher au monde extérieur** son plan d'adressage interne.
  - **Une entreprise souhaite se rendre indépendante** des adresses fournies par son fournisseur d'accès Internet.
- **La solution NAT : modification des adresses dans les datagrammes**
  - **Traduction des adresses IP** (dans un routeur ou dans un équipement de transit par exemple mur pare feux 'firewall').
  - **Basée sur l'acquisition** du datagramme, la **consultation** de table, la **modification** d'adresse, la **retransmission.**

NAT : Traduction statique et dynamique

- **Traduction statique (Static NAT) :**
  - Traduction d'une adresse IP d'entreprise vers une adresse IP extérieure
  - => **Toujours la même traduction réalisée.**
  - Typiquement adresse privée interne vers adresse publique du réseau mondial (facilite l'administration)

- **Traduction dynamique (Dynamic NAT) :**
  - Traduction d'une adresse IP d'entreprise (privée) vers une adresse IP publique prise dans une réserve
  - => **Un hôte n'a pas toujours la même adresse IP.**
  - Facilite l'administration et économise les adresses

NAT : Les quatre approches de traduction (1)

- **NAT avec traduction pour des transactions en sortie uniquement (unidirectionnel en sortie).**
  - Traduction NAT classique des adresses IP internes en adresses IP externes.
- **NAT avec transactions dans les deux sens (NAT unidirectionnel en sortie et unidirectionnel en entrée) :**
  - **Cas précédent ou des hôtes internes requièrent des serveurs externes** (traduction unidirectionnelle en sortie)
  - **Cas ou des clients externes requièrent des serveurs internes :** NAT en relation avec le DNS donne une adresse publique d'un serveur interne et transforme cette adresse en adresse privée (traduction unidirectionnelle en entrée).

NAT : les quatre approches de traduction (2)

- **NAT Bidirectionnel (avec traduction dans les datagrammes des adresses sources et destination) :**
  - Cas ou les espaces d'adressage internes et externes se recouvrent (adresses privées ou usage anormal d'adresses publiques). Exemple: mise en relation NAT de deux réseaux privés construits sur le bloc 10.0.0.0/8.
- **Traduction d'adresse et de numéro de port**
  - **NAT surchargé 'Overloaded'** ou encore **NAT with PAT 'Port Address Translation.**
  - Traduction du couple (adresse IP, numéro de port TCP ou UDP) vers un autre couple (adresse, numéro de port).

  => **Une adresse IP dans une réserve et un numéro de port sur 16 bits sont donc réattribués**.

  => **La solution la plus utilisée.**

Exemple : NAT unidirectionnel avec traduction de numéro de port

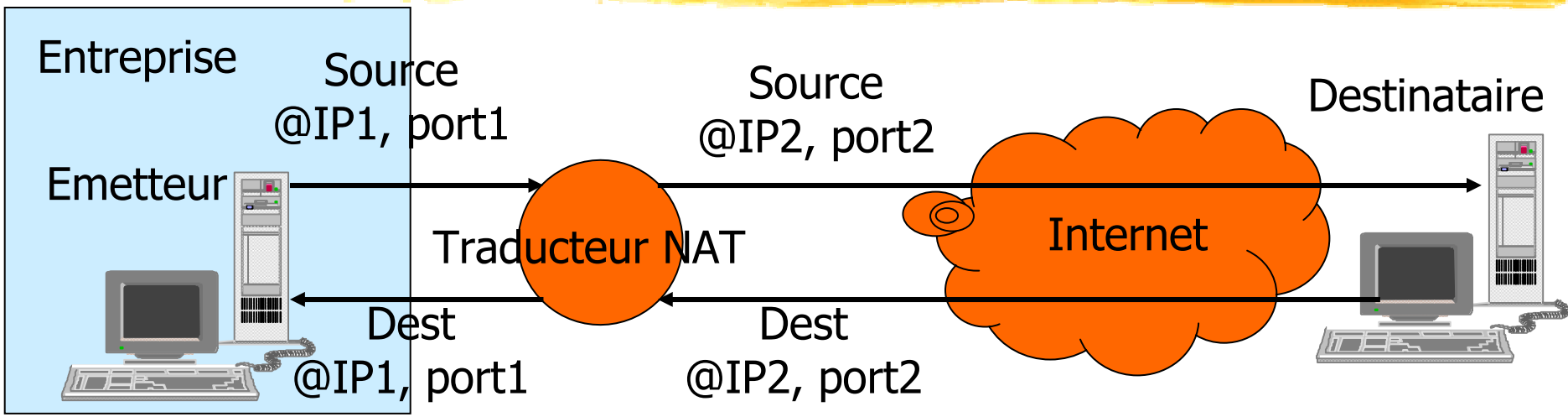


Table du traducteur

@IP1 , port 1	@IP2 , port 2

- L'adresse @IP1 peut être privée.
- L'adresse @IP2 doit être publique : un seule adresse peut servir 65536 – 4096 = 61440 applications (numéro de port sur 16 bits et les 4096 premiers numéros sont réservés aux ports bien connus).

Conclusion NAT

- **Solution simple, peu coûteuse et très efficace** : la solution qui a assuré la survie de l'adressage IP V4
- **NAT une solution qui pose aussi des problèmes**
  - Ne respecte pas le principe: chaque interface une adresse IP (problème d'identification des sources)
  - Le mode datagramme IP devient plus ou moins connecté.
  - NAT: uniquement prévu pour TCP et UDP, viole le principe d'indépendance des couches (mélange réseau/transport).
  - Si des applications placent des adresses IP dans les datagrammes il faudrait que NAT modifie à deux endroits.
  - Problème des mécanismes de sécurité avec chiffrement de charges utiles encapsulant des datagrammes donc des adresses IP.
  - NAT retarde le déploiement de IP V6.

E4) Distribution des adresses IP : La solution de base RARP

- **Problème : Attribution à un site d'une adresse IP.**
- **RARP 'Reverse Address Resolution Protocol' RFC 903**
  - Une solution pour la correspondance entre adresses MAC et IP.
  - Utilisée pour les machines sans disques et facilite l'administration.
- **Solution sur réseau local.**
  - Chaque hôte dispose d'une adresse IP fixe.
  - Un hôte qui démarre demande son adresse IP fixe.
  - Diffusion sur Ethernet de requête RARP en donnant son adresse MAC.
  - Un serveur RARP fournit l'adresse IP correspondant à l'adresse MAC.
- **Problèmes**
  - Ne fonctionne qu'avec des adresse IP fixes.
  - RARP ne gère que la distribution d'adresses IP.
  - Nécessite un serveur RARP sur chaque tronçon (ou mise en œuvre de serveurs proxy, relais de requêtes).

Distribution des adresses IP: La solution BOOTP

- **BOOTP 'Bootstrap Protocol' RFC 951, 1048, 1084**
  - Faire un outil pour le démarrage des stations sans disques.
  - Utiliser UDP et le routage IP pour avoir un seul serveur par entreprise.
  - Fournir différentes informations à l'initialisation (routeur par défaut, masque, serveur de fichiers de boot pour station sans disque …).
- **Solution BOOTP pour les adresses IP sur UDP.**
  - Protocole avec différents formats de messages au dessus de UDP.
  - Un hôte qui démarre diffuse une demande d'adresse IP en UDP sur le port serveur bootp 67 (réponse en diffusion sur le port client 68).
- **Problèmes**
  - Ne fonctionne qu'avec des adresse IP fixes.

Attribution dynamique d'adresses : DHCP (RFC 941)

- **DHCP 'Dynamic Host Configuration Protocol'**
  - Une solution de **distribution d'adresses sur réseau local**.
  - Utilise les **formats** de message du protocole **BOOTP**.
  - Requête d'adresse IP en diffusion sur réseau local: **DHCPDISCOVER.**
  - Réponse d'un serveur DHCP proposant une adresse: **DHCP OFFER.**
  - Acceptation d'une adresse offerte par le client: **DHCPREQUEST.**
  - Acquittement par le serveur: attribution d'adresse: **DHCPACK.**
- **Gestion de bail** : location d'adresse IP pour une période limitée pour récupérer les adresses inutilisées.
- **Comme Bootp** : fourniture d'autres informations utiles.
- **Possibilité de déclarer des relais DHCP** sur des routeurs pour atteindre d'autres tronçons.

Conclusion : DHCP

- **Principal avantage :**
  - **Administration simplifiée des adresses (administration centralisée).**
  - **Pas de problèmes d'erreurs dues à l'utilisation de la même IP.**
- **Deux solutions**

A) Un hôte reçoit **toujours la même** adresse IP: IP fixe comme en Bootp (pour des serveurs).

B) Un hôte reçoit **une adresse IP prise** dans un ensemble d'adresses disponibles.

- Une même adresse peut servir à désigner **des hôtes différents dans le temps.**
- Il n'est pas nécessaire **d'avoir autant d'adresses que d'abonnés** si tous les abonnés ne se connectent pas en même temps.

Conclusion Adressage IPv4

- **Les problèmes de l'adressage IPv4** : tarissement des adresses, grossissement des tables de routage, trop grande centralisation de distribution.
- **Ont reçus des solutions astucieuses qui permettent à IPV4 de durer** : CIDR, NAT, DHCP …

- **Le plan d'adressage Internet IPv4 devrait néanmoins tôt ou tard arriver à saturation**

  - Incertitude très grande sur la date effective

  - Liée au développement des services Internet consommateurs d'adresses: Internet fixe, mobile, téléphonie, commerce électronique, domotique…

  - Et à la façon de régler les problèmes d'adressage dans tous ces cas
- **Les difficultés prévisibles de l'adressage IPV4 ont amené à spécifier une version nouvelle de IP IP Version 6.**

IP

Chapitre II

Le protocole IP en version 6

Généralités IPV6

Structure des datagrammes

Adressage

Introduction IPv6

- **Besoin d'un nouveau protocole** qui apporte des réponses aux **limitations du plan d'adressage v4.**
- **Incorporer** aussi les évolutions technologiques (**sécurité, performances, administration**)
- **Etude à partir des années 1990** : différentes propositions pour un futur IP baptisé tout d'abord: IP NG
- **Processus de choix difficile à l'IETF** => Choix techniques principaux = adoption des RFC **1994-1995**.
  - IP v5 : Protocole ST2 RFC 1819: multimédia, en connexion
  - IP v7 : Réseau OSI sans connexion CLNP
  - **IP v6** : choix/fusion entre propositions CATNIP, TUBA, SIPP
- **Décision définitive 1998**
- **Implantations disponibles** en cours à partir de 1995-1996, routeur IPV6 2001 depuis phase d'expérimentation/déploiement.

Critères de conception IPv6

Adressage / Routage

- **Grand espace d'adressage hiérarchisable**.
  - Adressage pour au moins un milliard de réseaux.
- **Autorisant un routage hiérarchisé.**
  - Diminution des tailles des tables
- **Distribution d'adresses facilitée** en répartissant les possibilités d'attribution.

Déploiement

- **Une transition 'sans jour j'.**
- **Tous les changements à effectuer sur tous les types d'appareils doivent être précisés** (protocoles annexes ICMP/IGMP, hôtes, routeurs, administration réseau, ...).

Modifications par rapport à IPV4

- **Capacité d'adressage quadruplée**
  - 128 bits soit 16 octets (au lieu de 32 bits).
- **Simplification du format d'entête standard**
  - Optimisation pour un routage simplifié.
  - Suppression des champs inutiles au routage.
  - Alignement sur des frontières de mots
- **Etiquette de flot**
  - Identifier des flots d'octets pour permettre la réservation de ressource => qualité de service.
- **Pas de somme de contrôle d'entête**
- **Amélioration des extensions et des options**
  - Sous forme d'extensions à l'entête minimum.

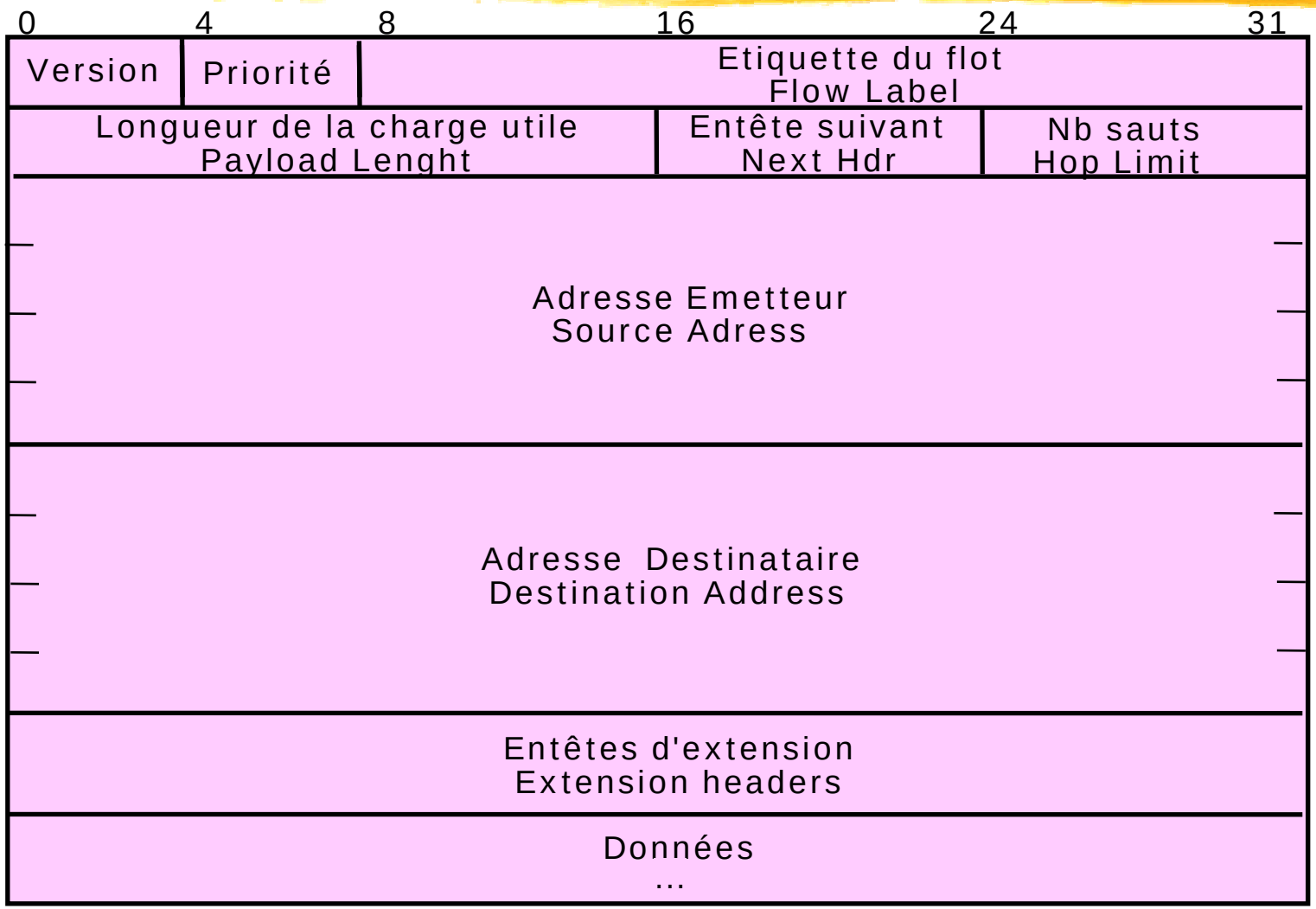
Fonctionnalités requises pour IPv6

- **Support de l'autoconfiguration** ('plug and play')
- **Support de mécanismes de sécurité** (confidentialité, authentification, intégrité)
- **Support de la qualité de service temporelle** (existence de mécanismes pour la réservation de ressources).
- **Support du mode diffusion.**
- **Support de la mobilité.**
- **Support d'artères à tous les débits**.

IP Version 6

Structure du datagramme

Format du datagramme IPv6



Détails concernant les champs IPv6 (1)

- **Numéro de version IP (4 bits) "IP version number"**

  Ici valeur 6 (IPv6).

- **Classe de trafic (4 bits) "Traffic Class" "Priority"**

Permet la définition de la priorité entre les flots de datagrammes.

Valeurs de 0 à 7 pour les flots pouvant ralentir en cas de congestion.

0 Pas de priorité particulière
1 Trafic de fond ("news")
2 Trafic non attendu ("mail")
3 Réservé pour usage futur
4 Trafic en rafale attendu ("ftp")
5 Réservé pour usage futur
6 Trafic interactif et (X11)
7 Commandes: routage, admin

Valeurs de 8 à 15 trafic "temps réel" non susceptible de ralentir (multimédia)

Remarque : Problème de la QOS temps réel: encore en développement. Autre découpage proposé: Traffic Class plus riche sur 8 bits et étiquette de flot seulement 20 bits.

Détails concernant les champs IPv6 (2)

- **Etiquette de flot (24 bits) "Flow label"**

  - En relation avec l'adresse émetteur une étiquette de flot identifie un flot de données:

  => On peut allouer des ressources à ce flot pour lui assurer un certaine qualité de service.

  - Utilisation en liaison avec RSVP "Resource Reservation Protocol".

- **Longueur de la charge utile (16 bits) "Payload Lenght"**

  - A la différence de IPv4 on ne compte pas les 40 octets de l'entête.

Détails concernant les champs IPv6 (3)

- **Prochain entête "Next Header"**

  - De nombreux entêtes d'extension sont prévus pour compléter l'entête de base selon les besoins.
  - Les entêtes forment une liste.
  - Cette zone détermine le type du premier entête.
  - Le dernier entête définit le protocole utilisateur.
    - 0 Informations de routage saut par saut
    - 4 Protocole internet
    - 6 Protocole TCP
    - 17 Protocole UDP
    - 43 Entête de routage
    - 44 Entête de fragmentation (par la source)
    - 45 Protocole de routage inter domaine
    - 46 Protocole de réservation (RSVP)
    - 50 Confidentialité de la charge
    - ……..

Détails concernant les champs IPv6 (4)

- **Nombre de sauts max (8 bits) "Hop Limit"**

  - Comme dans IPv4 le nombre maximum de commutateurs pouvant être traversés (ancienne zone 'Time To Live' avec un nouveau nom qui correspond à la fonction).

  - Le diamètre du réseau 256 est jugé trop faible par certains commentateurs.

- **Adresse source (128 bits) ("Source address")**

  - Adresse IP de l'émetteur.

- **Adresse destination (128 bits) ("Destination address")**

  - Adresse IP du destinataire.

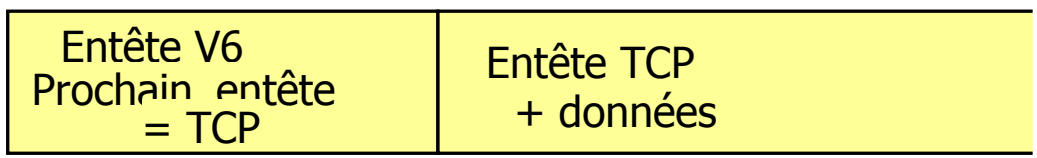
- **Données "Data"**

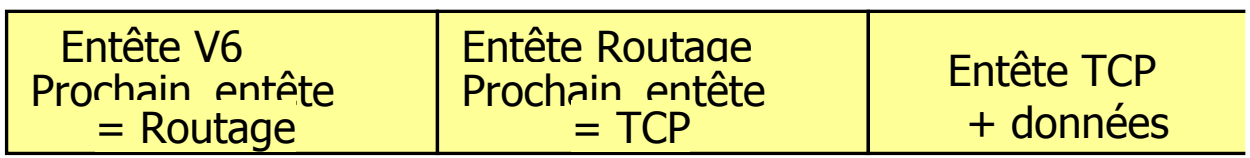
  - Zone de donnée d'une taille max de 64 Ko.

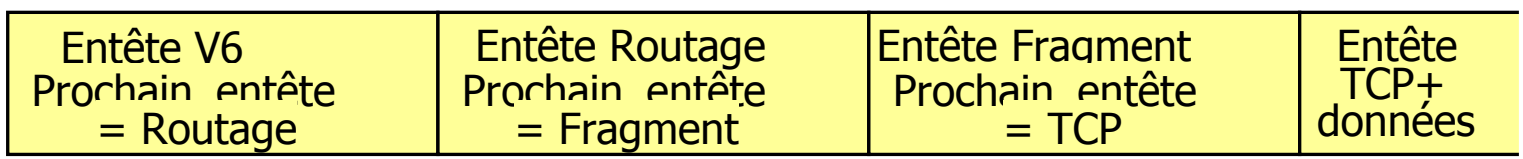
  - Une entête d'extension particulière permet de définir des longueurs sur 32 bits jumbograms.

IPV6 : Les entêtes d'extension "Extension Headers" RFC 1883

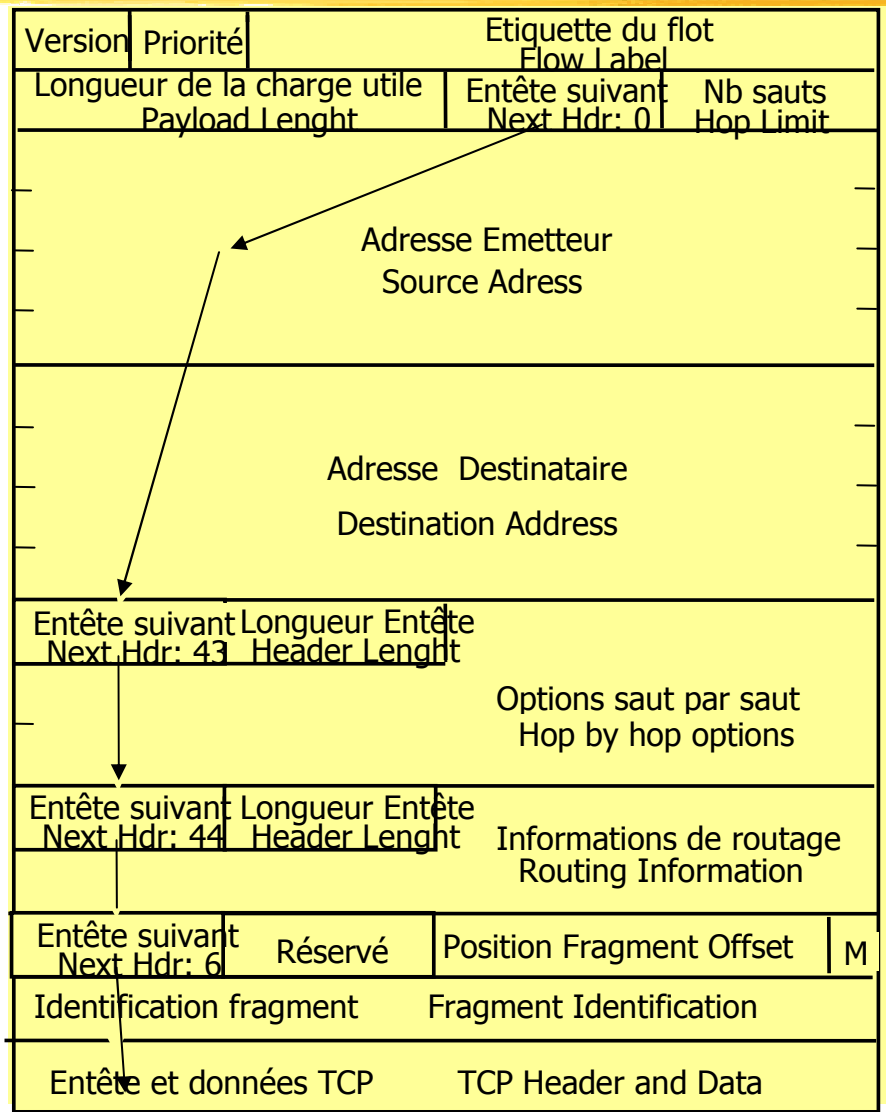
- **Nombreuses options prévues par le protocole** codées dans un nombre variable de champs d'extension en début.
- **Les extensions ne sont pas traitées par les routeurs** sauf l'extension infos "pour chaque saut" ("Hop by hop").
- **Les entêtes forment une liste.**







IPV6 : Les entêtes d'extension
Autre présentation de la liste



Description de quelques entêtes d'extension (1)

- **Les champs d'extension et leur ordre**
  - Une extension ne peut apparaître qu'une fois.
  - On doit rencontrer les extensions dans l'ordre suivant
- **Infos saut par saut 0 "Hop-by-hop header"**
  - Définit des informations pour chaque routeur rencontré par le datagramme.
  - Différents types d'informations sont précisées sur un octet.
  - En particulier le code 194 " Jumbo Payload Lenght" définit un paquet dont la taille dépasse 64K (jusqu'à 32 bits)
- **Routage 43 "Routing Header"**
  - Définit un routage par la source comme en IPV4.

Description de quelques entêtes d'extension (2)

- **Authentification 51 "Authentication Header"**
  - La méthode proposée par défaut par IPV6 utilise une clé secrète connue de l'émetteur et du destinataire.
  - La clé combinée avec le paquet transmis est compressée avec l'algorithme MD5 ("Message Digest 5").
  - Beaucoup d'autres approches de sécurité IPSEC.
- **Pas de prochaine entête 59 "No next header"**

Entêtes de fragmentation (2)

- **Généralités**
  - **La fragmentation ne dispose plus** d'informations toujours présentes dans l'entête de tous les datagrammes.
  - **Part non fragmentable**: entête de base plus quelques extensions (routage).
  - **Part fragmentable** : les autres extensions et les données
- **Entête 44 "Fragment Header"**
  - Définit une fragmentation avec des paramètres voisins de ceux de V4
  - Bit M, identificateur, déplacement (offset) du fragment.
- **Solution de fragmentation transparente** (bout en bout).
- **Découverte du MTU de chemin** :
  - Le plus grand MTU possible qui ne conduise pas à une fragmentation sur le parcours.
  - L'émetteur émet un paquet avec le bit Don't Fragment (taille du paquet inférieure ou égale au MTU local; prise en compte aussi du MSS Maximum Segment Size taille max définie pour les messages TCP).
  - Si le MTU nécessite une fragmentation dans un routeur intermediaire
    - Ce routeur transmet un message ICMP "I can't fragment"
    - L'émetteur recommence avec une taille plus petite.

IP Version 6

Adressage

IPv6 : Choix d'une adresse 128 bits

- **Rappel des principes de base**
  - Une adresse IP v6 adresse une interface (pas à un hôte).
  - C'est un identifiant unique pour une interface
  - C'est un moyen de localisation de cette interface.
- **Adressage IP V6 : ambition à terme** d'être le principal système d'adressage au niveau mondial => **effet grille-pain**
- **Choix** : entre des adresses de taille fixe (plus rapide à traiter) et des adresses de taille variable => **Taille fixe grande**
- **Choix 128 bits :** un choix de compromis entre 64 bits (jugé trop faible) et 160 bits adresse OSI (trop grand ou trop OSI).
  - A priori 3.9 * 1018 adresses par mètre carré de surface terrestre.
  - Si l'on utilise très mal les adresses disponibles (comme dans le téléphone) => 1500 adresses par mètre carré.

IPv6 : Trois catégories d'adresses

- **Adressage "Unicast" point à point.**

  - Une adresse pour un seul destinataire => le paquet est délivré à l'interface identifiée par l'adresse (comme en IP v4).

- **Adressage "Multicast" diffusion**

  - Une adresse pour un ensemble de destinataires => le paquet est délivré à toutes les interfaces du groupe identifié par l'adresse (comme en IP v4).

- **Adressage "Anycast"**

  - Une adresse pour un ensemble de destinataires => le paquet est délivré à l'une quelconque des interfaces appartenant au groupe identifié par l'adresse

  - Utilisation possible, accès à un seul serveur appartenant à un groupe de serveurs (exemple trouver un serveur au moins).

IPv6 : Représentation des adresses

- **Notation en hexadécimal** par groupes de 16 bits avec des deux points comme séparateurs.

  128 bits = 32 chiffres hexadécimaux = 8 groupes de 4 chiffres

  0ECD:AB56:0000:0000:FE34:98BC:7800:4532

Deux raccourcis d'écriture sont prévus

- **Omission des zéros en tête de groupe**.

  ECD:AB56:0:0:FE34:98BC:7800:4532

- **Plusieurs groupes de 16 bits à zéro peuvent être remplacés par ::**

L'abréviation :: ne peut apparaître qu'une fois dans une adresse.

ECD:AB56::FE34:98BC:7800:4532

IPv6 : Adresses particulières

- **Adresses de réseaux**

  - **Adressage de type CIDR** => Tout découpage réseau/ sous réseau est possible (selon des plans d'adressages).

  - La notation **adresse_Ipv6/n** définit la valeur du masque (les n bits en fort poids forment l'adresse de réseau, les autres bits sont à 0).

- **Adresse non spécifiée ("Unspecified")**

  - Pour un site en initialisation qui demande à un serveur son adresse réelle (seulement utilisable comme adresse source).

  0:0:0:0:0:0:0:0 ⇔ ::

- **Adresse de rebouclage ("Loopback")**

  - L'adresse pour s'envoyer des messages (ne peut circuler sur le réseau).

  0:0:0:0:0:0:0:1 ⇔ ::1

IPv6 : Plans d'adressage
Adresses de plus haut niveau

0::/8	00000000	Adresses IPv4
200::/7	0000001	Adresses OSI CLNP
400::/7	0000 010	Adresses Novell IPX
2000::/3	001	Adresses agrégées
4000::/3	010	Adresses prestataires
8000::/3	100	Adresses géographiques
FE80::/10	1111111010	Adresses locales lien
FEC0::/10	1111111011	Adresses locales site
FF00::/8	1111 1111	Adresses de diffusion

Récupération de la base existante OSI-CLNP, Novell

- **Protocoles de réseaux existants non IP**
  - CLNP "ConnectionLess Network Protocol"
  - IPX "Internetwork Packet Exchange"
- **Le plan d'adressage v6 propose au moyen de préfixes de reprendre ces adresses réseaux existantes**

  => migration facilitée pour ces protocoles.
- **Conversions d'adresses**

  - IPX 80 bits (10 octets) à compléter à 121 bits

  - Problème pour les adresses CLNP-NSAP "Network Service Access Point" 20 octets=160 bits à faire rentrer dans 121 bits ….

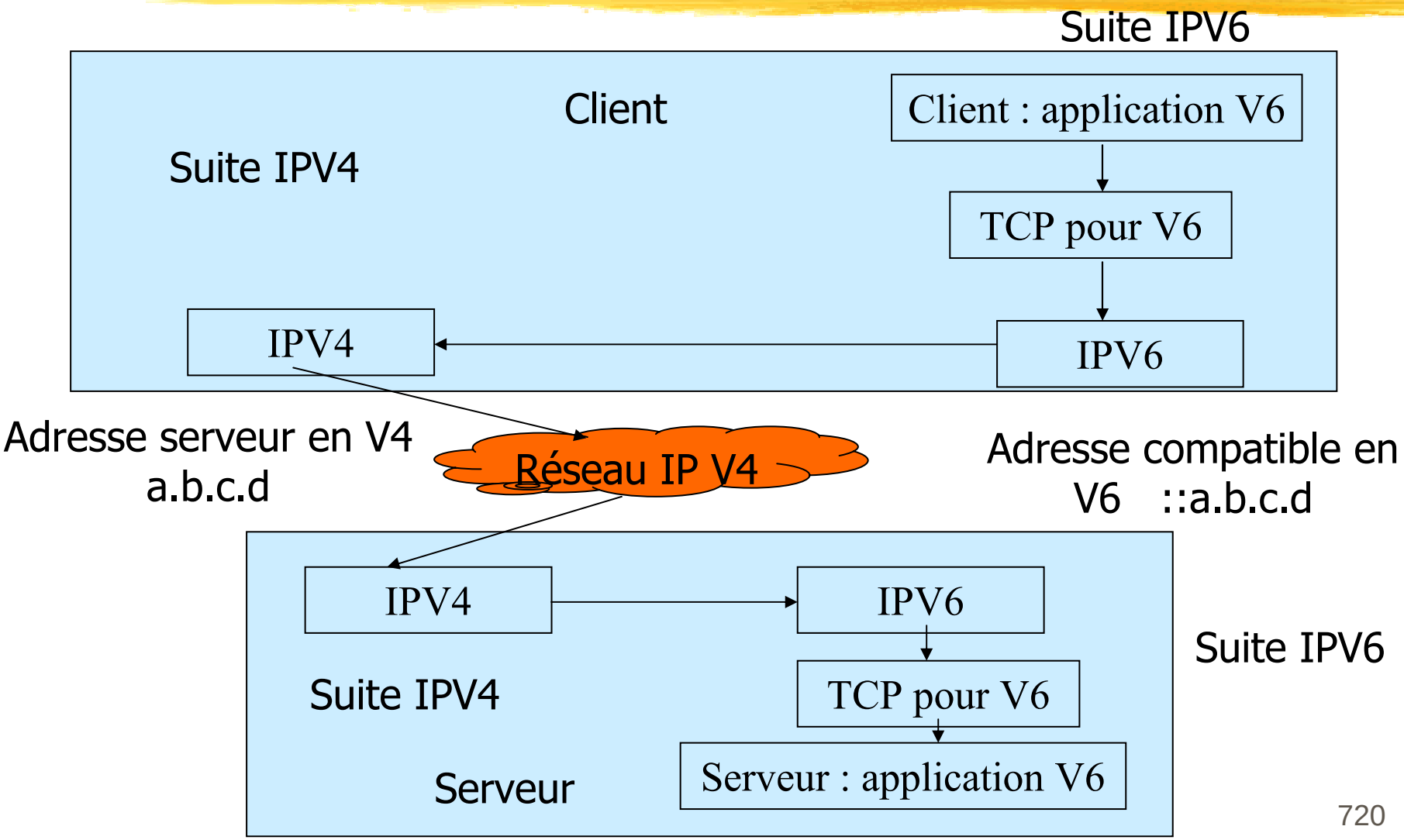
Adresses IPV6 compatibles IPV4 Transition par encapsulation

- **Phase de transition IPV4 vers IPV6:** situation ou l'on communique encore en IPV4 pour démarrer IP V6.
- **Solution d'encapsulation ("tunnelling")** de datagrammes IP v6 dans des datagrammes IP v4 (IPV6 acheminé par IP v4 et délivré à distance à une pile IP v6 après désencapsulation).
- **Adresse IPV6 compatible IPv4** "IPv4 Compatible address"

  Un hôte IP à une adresse IPV4 et une adresse IP v6 en rajoutant des 0 devant l'adresse ipv4 pour en faire de l'IP v6.

  Forme: 0:0:0:0:0:0:a.b.c.d soit ::a.b.c.d
- **Un site IP v6 souhaitant communiquer** avec un autre site IP v6 **au moyen de IP v4** utilise une suite IPV4, une adresse IP V4 et une adresse IP v6 compatible IP v4.

Encapsulation IPV6 dans IPV4

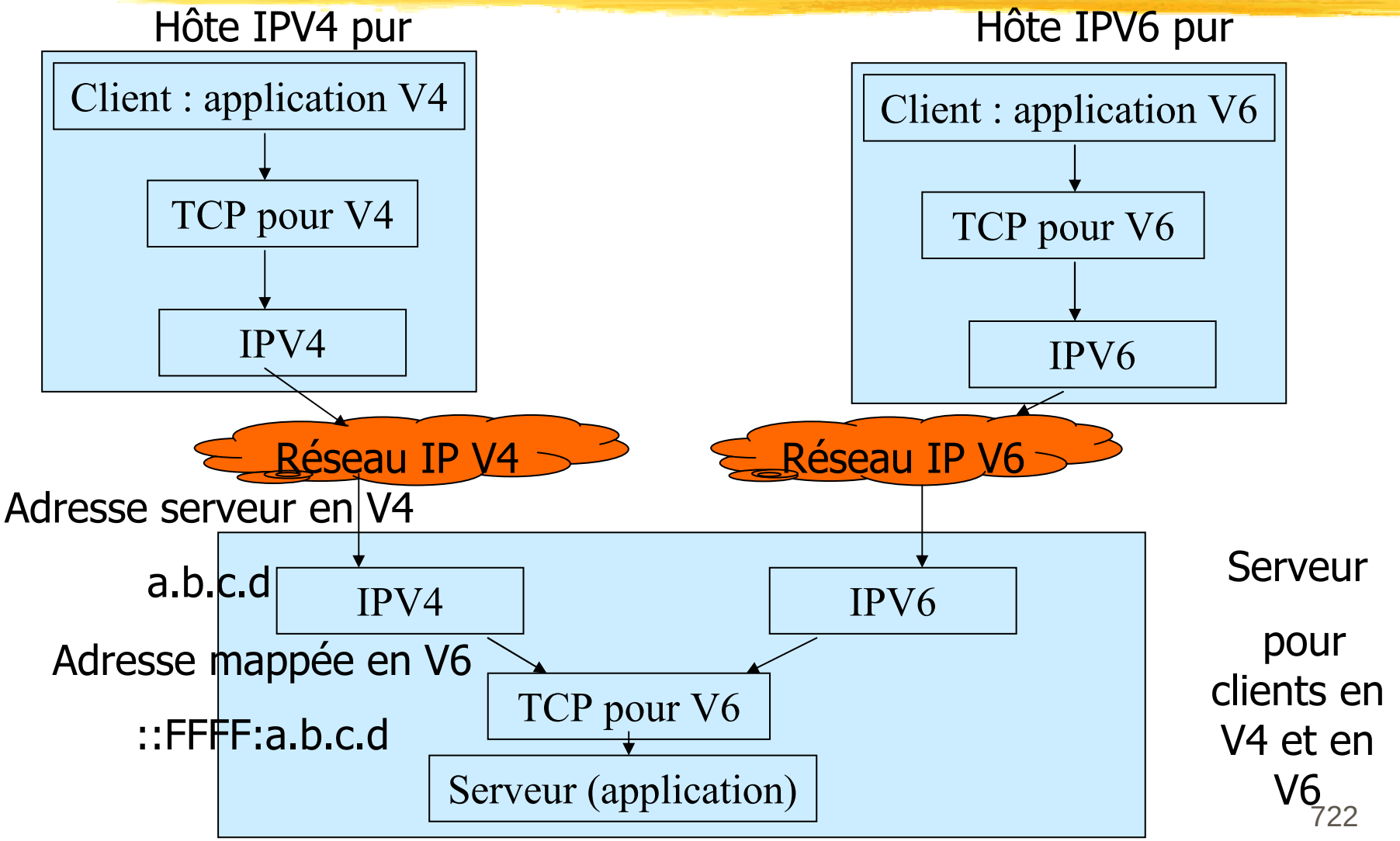


Adresses IPV4 représentées en IPV6 (‘mapped’)

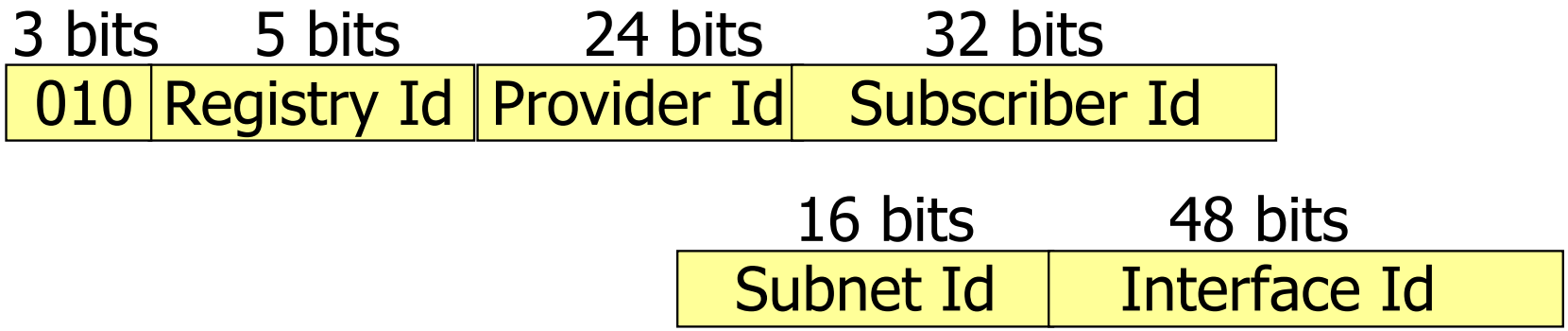
- **Transition IPV4-> IPV6 par transformation (mappage)**
  - En **émission** une requête de transmission pour un datagramme avec une adresse IP v4 représentée en IP v6 est **traité par une pile IP v4.**
  - En réception, le datagramme reçu par IP v4 est présenté à son destinataire (TCP) comme s'il s'agissait **d'un datagramme arrivé en IP v6 avec une adresse mappée**.
- **Adresse IPV4 représentée par une adresse IPV6** "IPv4 mapped IPv6 address"

  0:0:0:0:0:FFFF:a.b.c.d soit ::FFFF:a.b.c.d
- **Seul un trafic Ipv6 acheminé par IPV4** peut utiliser une adresse IP V6 mappée.
- **On peut communiquer à partir de sites IP V4 vers des sites IPV6** comme si l'on se trouvait dans le domaine d'adressage IP V6.

Adresses IPV4 représentées en IPV6 (‘mapped’)

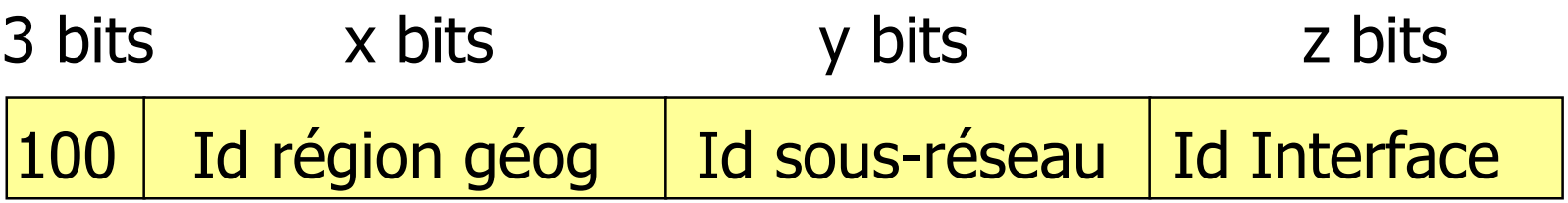


IPV6 : Plan d'adressage prestataire 'Provider based unicast address'



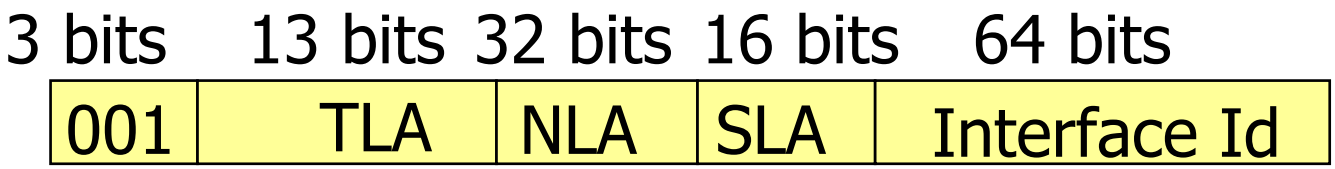
- **Quatre autorités ("Registry") prévues**
  - IANA Internet Assigned Numbers Authority
  - RIPE-NCC Réseaux IP Européens Network Coordination Center
  - INTERNIC Inter Network Information Center
  - APNIC Asia Pacific Network Information Center
- **Non utilisé** : trop dépendant des prestataires à tous niveaux (changement de prestataire => changement d'adresse).

IPV6: Plan d'adressage géographique 'Geographic based unicast address'



- **Ces adresses seraient distribuées** selon des contraintes géographiques (pays, région, …).
- **Les opérateurs / les monopoles de Télécom** devraient jouer un rôle majeur.
- **Beaucoup de problèmes de mise en œuvre :** répartition d'entreprises sur différentes zones géographiques.
- **Non utilisé**

IPV6 : Plan d'adressage agrégé 'Aggregatable global addresses'



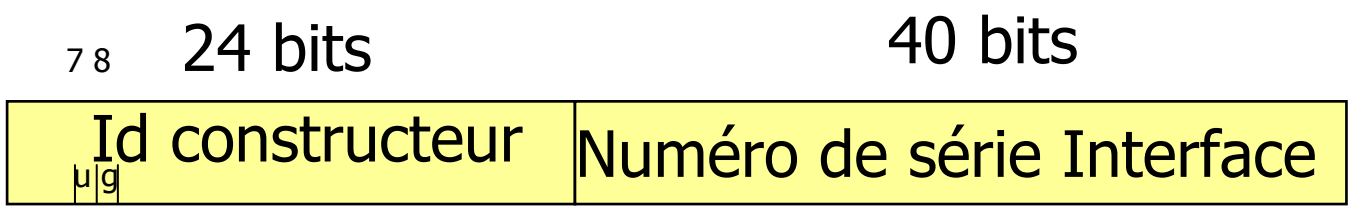
- **Préfixe** 2000::/3 (3bits)
- **TLA ('Top Level Aggregator') (13 bits)**
  Agrégation de plus haut niveau: ce niveau représente de très grands ensembles d'adresses (ex: grands opérateurs internet)
- **NLA ('Next Level Aggregator') (32 bits)**
  Agrégation de niveau intermédiaire: ce niveau représente des ensembles d'adresses de taille intermédiaire (prestataires de service moyens). Ce niveau est hiérarchisable.
- **SLA ('Site Level Aggregator') (16 bits)**
  Agrégation au niveau d'un site (ex une entreprise). Ce niveau est hiérarchisable.
- **Le plan d'adressage actuellement en service.**

Identificateur d'interface 64 bits 'Interface Identifier'

- **IEEE** : normalisation des identificateurs d'interface sur 64 bits

=> Format EUI 64: identifier toutes les interfaces de réseaux (IEEE 802, Appletalk, ..)

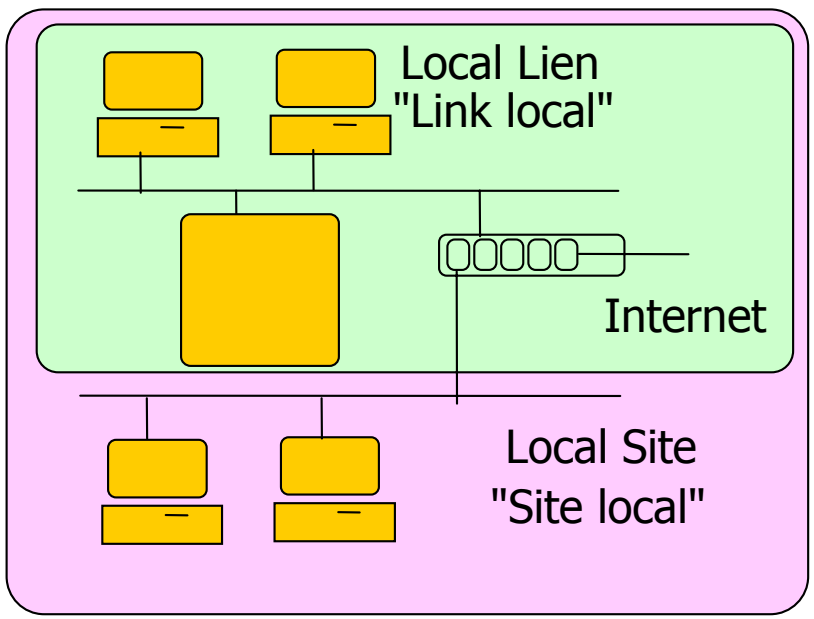
Format d'identificateur d'interface pour IP v6 (dérivé de EUI 64)



- **Bit u (universel) :** 1 => l'adresse est unique, 0 sinon (bit inversé en EUI)
- **Bit g (group) :** 1 => adresse d'un groupe ; 0 => adresse individuelle.
- **Identifiant constructeur** : 24 bits comme pour les adresses MAC IEEE
- **Numéro de série interface:** 40 bits (par exemple on porte de 24 bits à 40 bits la partie numéro de série des adresses IEEE802 en ajoutant FFFE).

IPV6 : Adresses locales

- **Adresses de portée locale**
  - Ces adresses ne sont pas valides à l'extérieur d'une certaine portée
  - => Les routeurs ne les acheminent pas.
- **Permettent de construire des réseaux Internet privés** (à l'abri d'un mur anti feu) comme dans le cas des adresses réservées IP v4.
- **Deux portées locales**
  - Locale site
  - Locale lien ou tronçon



IPV6 : Format des adresses locales

- **Portée locale site (préfixe FEC0::/10)**

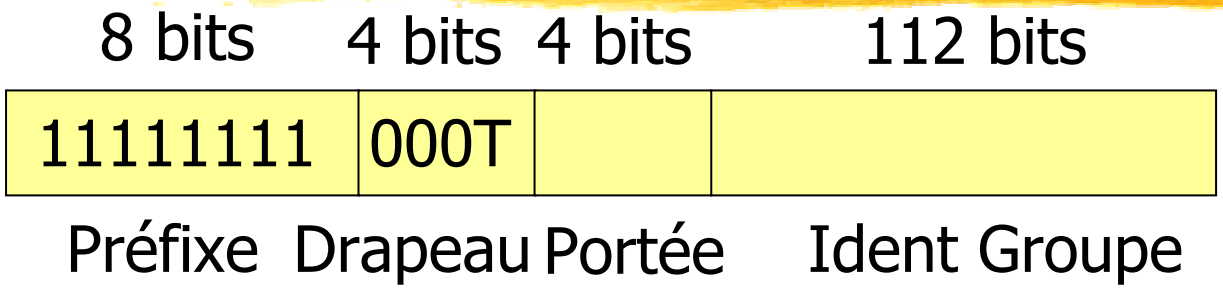
10 bits	n bits	m bits	118-n-m bits
1111111011	0		

- **Portée locale lien (un tronçon de réseau local) (préfixe FE80::/10)**

10 bits	n bits	118-n bits
1111111010	0	Interface Id

  - On doit mettre une adresse d'interface unique.
  - On prend l'adresse IEEE de la carte réseau accédant au réseau local.

IPV6 : Les adresses de diffusion



- **Préfixe ('Prefix) :** FF/8 **soient** 8 bits à 1 => 11111111
- **Drapeau ('Flag') :** 4 bits 000T

  T=1 adresse permanente ; T=0 adresse temporaire
- **Portée ('Scope') :** 4 bits XXXX Valeurs de portée de diffusion
  - 1 Diffusion limitée à un seul système
  - 2 Diffusion limitée à une seule liaison locale
  - 5 Diffusion limitée à un seul site
  - E Diffusion de portée globale à l'Internet
- **Identifiant de groupe ('Group Id') :** 112 bits

IPV6 : Exemple de quelques adresses de groupes prédéfinies

- **1 : L'ensemble des systèmes**

  FF05::1 Portée de site local

  FF02::1 Portée lien local.

- **2 : L'ensemble des routeurs**

  FF05::2 Portée de site local

  FF02::2 Portée lien local.

- **C : L'ensemble des serveurs de configuration DHCP** "Dynamic Host Configuration Protocol"

  FF02::C Portée lien local (tous les serveurs DHCP sur un tronçon, utilisation type de DHCP).

Conclusion IP V6

- **IP V6 : une amélioration certaine par rapport à IPV4**
  - Surtout pour ce qui concerne l'adressage
  - Mais aussi pour la prise en compte d'améliorations techniques diverses (sécurité, extensions …).
- **Mais démarrage très lent à grande échelle**.
  - **Très nombreux détails à régler**
  - **IPV6 constitue un effort de portage** que les utilisateurs n'ont pas envie de supporter tant que l'adressage IP V4 tient.
  - **On demande beaucoup plus à IPV6 qu'à IPV4.**
- **Développé depuis 1995**
  - Transition prévue au départ sur 15 ans.
  - Chaque année le lancement à grande échelle est annoncé.
  - Chaque année IP V6 se développe un peu.
- **Attente de l'application consommatrice d'adresses IP qui forcera la migration vers IPV6.**

Bibliographie IP V6

- RFC1752 'Recommendation for the IP Next Generation Protocol' 1/95
- RFC1809 'Using the Flow Label in IPv6' 6/95
- RFC1881 'IPv6 Address Allocation Management' 12/95
- RFC1883 'Internet Protocol, Version 6 Specification' 12/95
- RFC1884 'IP Version 6 Addressing Architecture' 12/95
- RFC1885 'Internet Control Message Protocol (ICMPv6)' 12/95
- RFC1886 'DNS Extensions to Support IPv6‘ 12/95
- RFC1887 'An Architecture for IPv6 Unicast Address Allocation' 12/95
- RFC1897 'IPv6 Testing Address Allocation' 12/95
- RFC1924 'A Compact Representation of IPv6 Addresses' 4/96
- RFC1933 'Transition Mechanisms for IPv6 Hosts and Routers' 4/96
- RFC1825 'Security Architecture for the Internet Protocol' 8/95

IP

Chapitre III
Le routage IP

Généralités

Routage statique

Routage dynamique

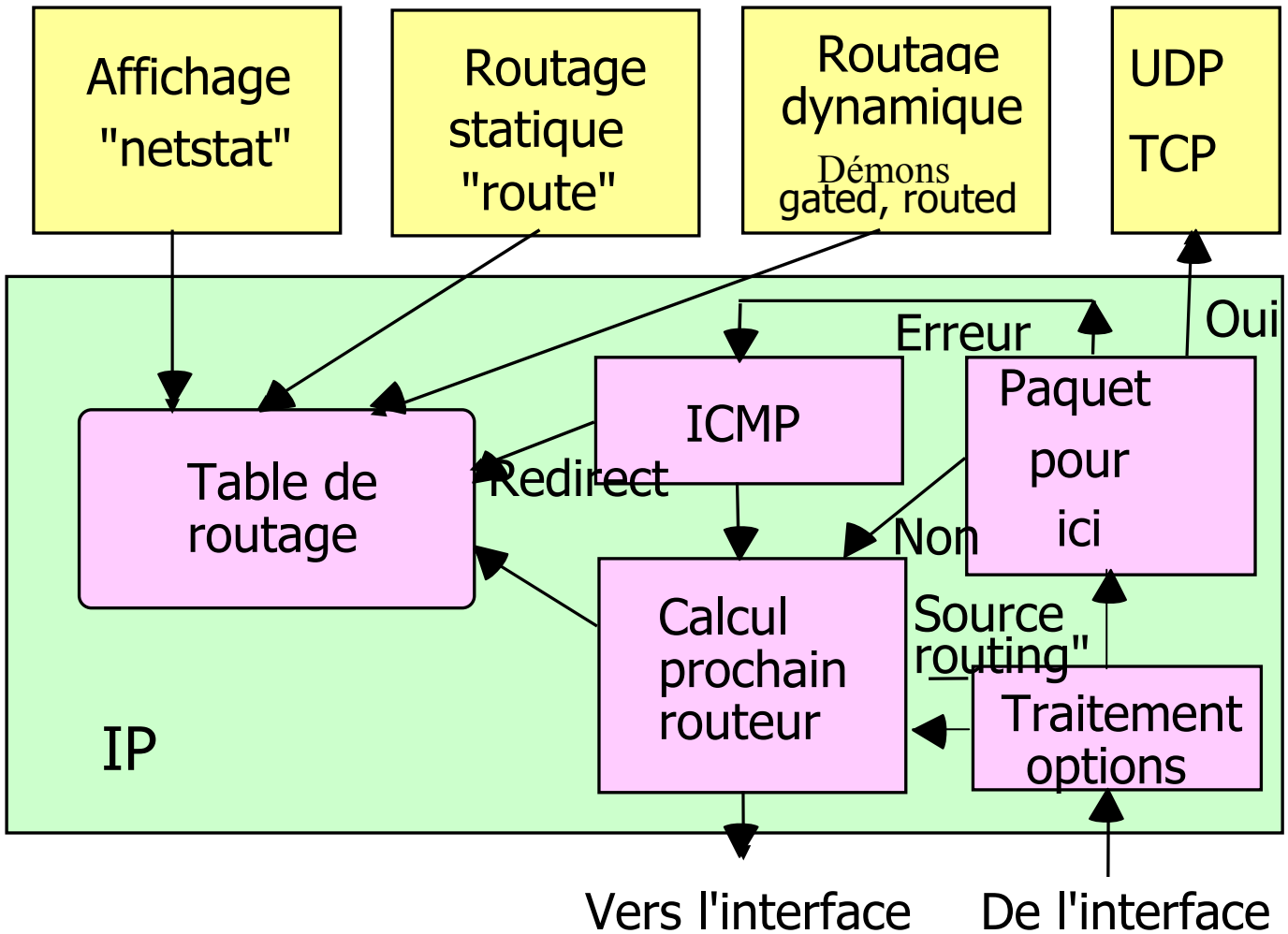
Introduction : Rappel du problème de routage

- **Objectif du routage point à point :** Atteindre un seul hôte destinataire en masquant la traversée d'une série de **réseaux et de routeurs intermédiaires**.
- **Routage multipoint (‘multicast’) :** Atteindre tous les hôtes d’un groupe destinataire.
- **Notion d’hôtes** (**"Hosts"**)
  - Un hôte ne **relaie pas** de messages (il dispose en général d’une seule interface et d'une table de routage simplifiée).
- **Notion de routeurs ("Routers" "Gateways")**
  - Un routeur possède plusieurs interfaces.
  - **Il retransmet un message d'une interface entrante à une interface sortante** s'il dispose des informations suffisantes pour le routage (sinon il note le message 'non délivrable‘).

Routage statique et dynamique

- **Routage statique** : Définition manuelle des routes (une fois pour toute).
- **Routage dynamique :** Mise en œuvre d'un protocole de communication entre routeurs pour l'apprentissage 'automatique' des meilleurs routes (selon des critères de coûts).

Organisation générale du routage IP



Etapes d'une opération de routage

- **Choix d'une route**
  - **Recherche des routes** pour la destination d'un paquet.
  - Choix de **correspondance la plus longue** *longest Match*
  - Eventuellement : choix entre des routes équivalentes selon la qualité de service **(TOS)** et la **métrique** de la route.
  - Eventuellement **répartition de charge**.
- **Transmission**
  - Si le site à atteindre est connecté directement au site courant (par une liaison point à point ou en réseau local)

    => Obtention de **l'adresse liaison** destinataire (ARP)

    => Le message est **envoyé directement**.
  - Sinon **transmission au prochain routeur** (next hop) qui reprend à son compte l'acheminement.

Gestion de la table de routage : A) Liste d'une table

Informations associées à une route (UNIX)

- **Adresse IP destination**
  - Généralement une adresse de réseau (la zone Host-id est à zéro).
- **Adresse du prochain routeur ("Gateway")**
  - A emprunter pour atteindre la destination.
- **Indicateurs ("flags")**
  - U chemin opérationnel,
  - G chemin vers un routeur,
  - H chemin vers un hôte,
  - D chemin créé par une redirection,
  - M chemin modifié par une redirection ….
- **Nombre de références**
  - Nombre de connexion utilisant le chemin (connexions TCP, UDP).
- **Métrique (de la route)**
- **Nombre de paquets envoyés**
- **Interface ("device")** (pilote et carte sur lesquels envoyer le paquet).

Table de routage pour un hôte: Exemple de liste en LINUX

- **Commande route (autre possibilité netstat –r)**

kirov::/users/ensinf/gerard _16 /sbin/route
Table de routage IP du noyau

Destination	Passerelle	Genmask	Indic	Metric	Ref	Use	Iface
163.173.128.0	*	255.255.252.0	U	0	0	0	eth0
loopback	*	255.0.0.0	U	0	0	0	lo
default	bigiron.cnam.fr	0.0.0.0	UG	0	0	0	eth0

kirov::/users/ensinf/gerard _17

- **Commentaire : table de routage d'un hôte =>** trois routes principales
  - **La boucle locale** : loopback (127.0.0.1) pour les messages qui ne sortent pas.
  - **L'accès aux hôtes** sur le même réseau Ethernet : ici 163.173.128.0
  - **L'accès à un routeur par défaut** qui ouvre sur le reste de l'Internet : default (en fait 0.0.0.0). Indicateur G route vers un routeur.

Table de routage pour un routeur: Exemple de liste sur routeur CISCO

mgs>**show ip route**
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default.4/9
Gateway of last resort is 193.51.128.81 to network 0.0.0.0
10.0.0.0 255.255.0.0 is subnetted, 1 subnets
S 10.35.0.0 [1/0] via 192.108.119.147
C 192.168.100.0 is directly connected, Ethernet5
C 192.168.101.0 is directly connected, Ethernet5
C 192.168.55.0 is directly connected, Ethernet3
C 192.168.0.0 is directly connected, Ethernet1
193.51.128.0 255.255.255.248 is subnetted, 1 subnets
C 193.51.128.80 is directly connected, Ethernet2
C 192.168.200.0 is directly connected, Ethernet4
S 192.168.201.0 [1/0] via 192.168.200.61
…… liste coupée ici …. etc …..

Gestion de la table de routage : B) Initialisation Statique (1)

- **Pour créer manuellement une route**
  - **Pour délivrer des datagrammes** sur un réseau local.
  - **Pour atteindre un routeur distant (ex type default).**
- **Exemple en UNIX : Commande ifconfig configure les paramètres du pilote de carte réseau**
  - A chaque définition d'une interface la table de routage est initialisée automatiquement en conséquence.
  - **Exemple** : /etc/ifconfig eth0 kirov up déclare un coupleur ethernet eth0 actif (up) . Différentes autres options
    - "netmask" : définition du masque de sous-réseau.
    - etc ...;

Exemple de liste de paramètres d'interface

```
kirov::/users/ensinf/gerard _16 /sbin/ifconfig eth0
eth0    Lien encap:Ethernet  HWaddr 00:09:3D:00:A9:7F
        inet adr:163.173.129.17  Bcast:163.173.131.255 Masque:255.255.252.0
        adr inet6: fe80::209:3dff:fe00:a97f/64 Scope:Lien
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:214175639 errors:0 dropped:0 overruns:0 frame:0
        TX packets:147434433 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 lg file transmission:1000
         RX bytes:158285671425 (150952.9 Mb)  TX bytes:87060146495
                                                        (83027.0 Mb)
         Interruption:25
kirov::/users/ensinf/gerard _17
```

Gestion de la table de routage: Initialisation Statique (2)

- **Exemple en UNIX : Commande route**
  - **Déclaration explicite d'une route** vers un réseau distant :

route add -net 192.56.76.0 netmask 255.255.255.0 dev eth0

  - **Les routes initialisées statiquement** sont contenues dans un fichier de configuration :
    - Exemples AIX /etc/rc.net, SUNOS /etc/rc.local, SOLARIS 2 /etc/rc2.d/S69inet

Gestion de la table de routage: C) Redirection par ICMP

- **Modification des tables de routage par découverte de chemins :** ICMP protocole additionnel qui achemine des informations de routage et des messages d'erreur pour IP
- **Utilisation des diagnostics ICMP d'erreurs pour améliorer le routage**
  - Exemple: routeur A envoie un message à un routeur B pour atteindre C.
  - B s'aperçoit qu'il ne peut atteindre C
  - B indique par un message ICMP à A ce problème ("host unreachable").
- **Messages ICMP de maintenance des tables**
  - ICMP peut diffuser des demandes de routes (en diffusion totale ou mieux en diffusion sur groupes) "router sollicitation message"
  - Les routeurs à l'écoute répondent : "router advertisement message"

Gestion de la table de routage: D) Routage dynamique

- **Routages dynamiques :**
  - **Les routes dans les tables sont modifiées** par des processus qui implantent des protocoles d'échange de routes.
  - Le routage dynamique permet **l'apprentissage des routes et l'adaptativité aux variations de charge**.
  - **Les tables de routages sont exploitées** de la même façon pour la commutation par IP (qu'elles soient initialisées statiquement ou dynamiquement).
- **Nombreuses possibilités de routage dynamique en IP.**
- **IP : Routage hiérarchisé à deux niveaux (au moins)**
- **Notion de domaine ou AS "Autonomous Systems":**
  - Un domaine correspond à un ensemble de sites, administrés par une seule et même entité (grande entreprise, campus).
- **Deux types de routage: intra et inter domaine.**

Liste de principaux protocoles de routage dynamique en IP

- **Protocoles de routage intra-domaine (IGP "Interior Gateway Protocol") :**
  - **RIP :** "Routing Information Protocol".
  - **IGRP, EIGRP :** "Enhanced Interior Gateway Routing Protocol".
  - **OSPF :** "Open Shortest path First".
  - **Integrated IS-IS :** "Integrated Intermediate System to Intermediate System".
- **Protocoles de routage inter-domaine (IRP "Interdomain Routing Protocol") :**
  - **EGP :** "Exterior Gateway Protocol".
  - **BGP :** "Border Gateway Protocol".

1) Routage dynamique: RIP Routing Information Protocol (RFC 1028)

- **Routage dynamique par échange périodique** de tables entre voisins **("Distance Vector").**
- **Initialisation du protocole**
  - **Emission d'une requête** de demande de table sur toutes les interfaces avec une liste de destination (**Code commande 1).**
  - **Réponse**: S'il y a une route pour la destination =>métrique de la route
    Sinon => métrique = valeur infinie (ex 30) (**Code commande 2).**
- **Fonctionnement en mode établi (‘message request’)**
  - **Mise à jour périodique** : émission vers tous les voisins systématiquement toute les x secondes (typiquement 30).
  - **Si une route n'a pas été rafraîchie** pendant y minutes elle est portée à infini (pour invalidation, y typiquement 3 minutes)
  - **Mise à jour de route** sur événement (changement de métrique).

RIP Routing Information Protocol : Informations complémentaires

- **Métrique d'une route :** nombre de sauts ("hops").
- **Protocole utilisé UDP :** pour échanger les informations avec les autres routeurs.
- **Nom du démon en UNIX: "Routed"**
- **RIP V2 RFC 1388:** extensions dans des champs inutilisés par RIP V1 (diffusion, identification de domaine, échange du masque)
- **Sous UNIX :** commande pour obtenir les informations de routage d'un routeur distant (code commande poll 5) :
  ripquery -n "nom de routeur"
- **Solution de moins en moins utilisée**

2) Routage dynamique: OSPF Open Shortest Path First (RFC 1247)

- **OSPF : remplaçant de RIP**
  - RIP est insuffisant pour les grands réseaux.
  - Déploiement progressif de OSPF après 1990.
- **Solution à état de liaison ("link state").**
  - Collecte par chaque routeur de l'état des liaisons adjacentes.
  - Echange de ces états (plusieurs approches, solution de base inondation).
  - Calcul par tous les routeurs des tables de routage optimales par l'algorithme de Dijsktra ("Shortest Path First").

OSPF : Caractéristiques

- **OSPF: un routage intra domaine mais hiérarchisé à deux niveaux**
  - OSPF permet de gérer à l'intérieur d'un domaine des régions ("areas")
  - OSPF calcule des routes intra-régions, inter régions.
  - Une région particulière permet de connecter les autres ("backbone" "area 0").
  - Pour la maitrise des grands domaines ("autonomous system").
- **OSPF propose une grande variété de métriques**
  - Débit, délai aller-retour, ...
  - Une métrique peut-être attribuée par type de service.
- **Calcul de plusieurs routes possibles** par type de service.
  - Si deux routes sont de coût équivalent: distribution de charge.
- **Echange des informations** de routage OSPF: protocole IP .
- **Nom du démon UNIX**: "Gated« .

Principaux messages OSPF

Type de message	Description
Hello	Découverte des voisins immédiats
Link state update	Diffusion aux voisins
Link state ack	Acquittement réception
Database description	Annonce des états dont dispose un routeur
Link state request	Demande informations à un routeur

3) Routage dynamique: BGP Border Gateway Protocol (RFC 1267, 1268)

- **Protocole de routage Inter-domaine.**
- **Autorise des politiques de routage** spécifique aux grandes organisations
- **Objectif :** contrôler l'acheminement par une organisation (interdire du traffic en transit , orienter le traffic, ...).
- **Trois types de domaines** (identifiés par des entiers 16 bits):
  - **Domaine souche ("stub") :** un seul lien avec l'exterieur
  - **Domaine de transit ("transit AS"):** plusieurs liens avec l'extérieur, et autorise le passage d'informations extérieures à son trafic local (transit)
  - **Domaine multi-liens ("multi homed AS"):** plusieurs liens avec l'extérieur, mais n'autorise pas le passage.
- **Solution par échange de tables** entre voisins comme RIP ("Distance vector", Mac Quillan)
- **Echange des informations de routage** : TCP.

Routage IP : Conclusion

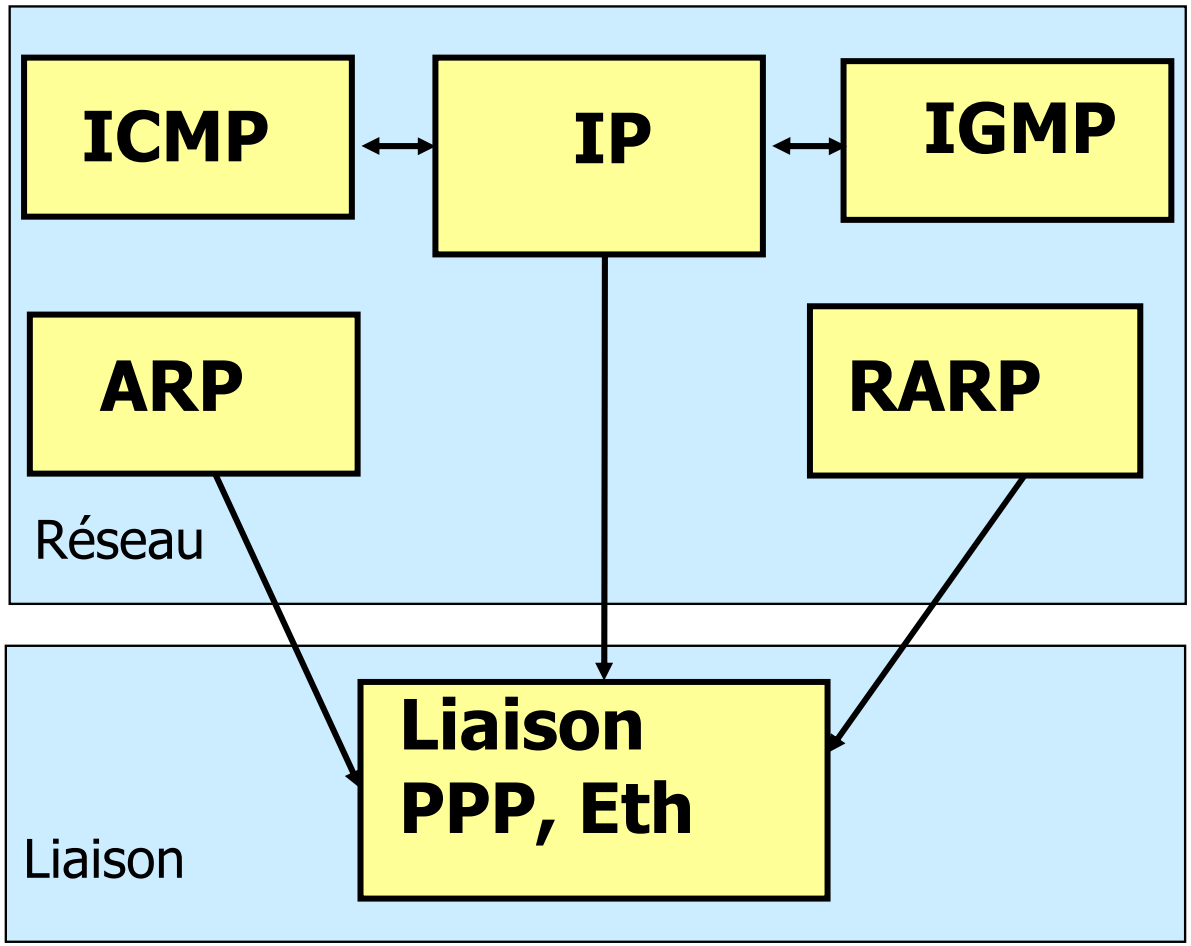
- **Hiérarchisation du routage :** le routage IP est devenu suffisamment hiérarchique avec les notions de sous-réseaux, les domaines (AS), les régions (areas) (mais le réseau Internet est très grand et en développement rapide).
- **Agrégation des routes :**
- **Rôle majeur des opérateurs, des grandes organisations et des fournisseurs de service :** qui administrent le routage pour des domaines importants.
- **Introduction de IP V6 :** Les algorithmes de routages sont basés exactement sur les mêmes principes (OSPF, BGP4) avec un espace d'adresses suffisant.

IP

Chapitre IV

Protocoles complémentaires de IP

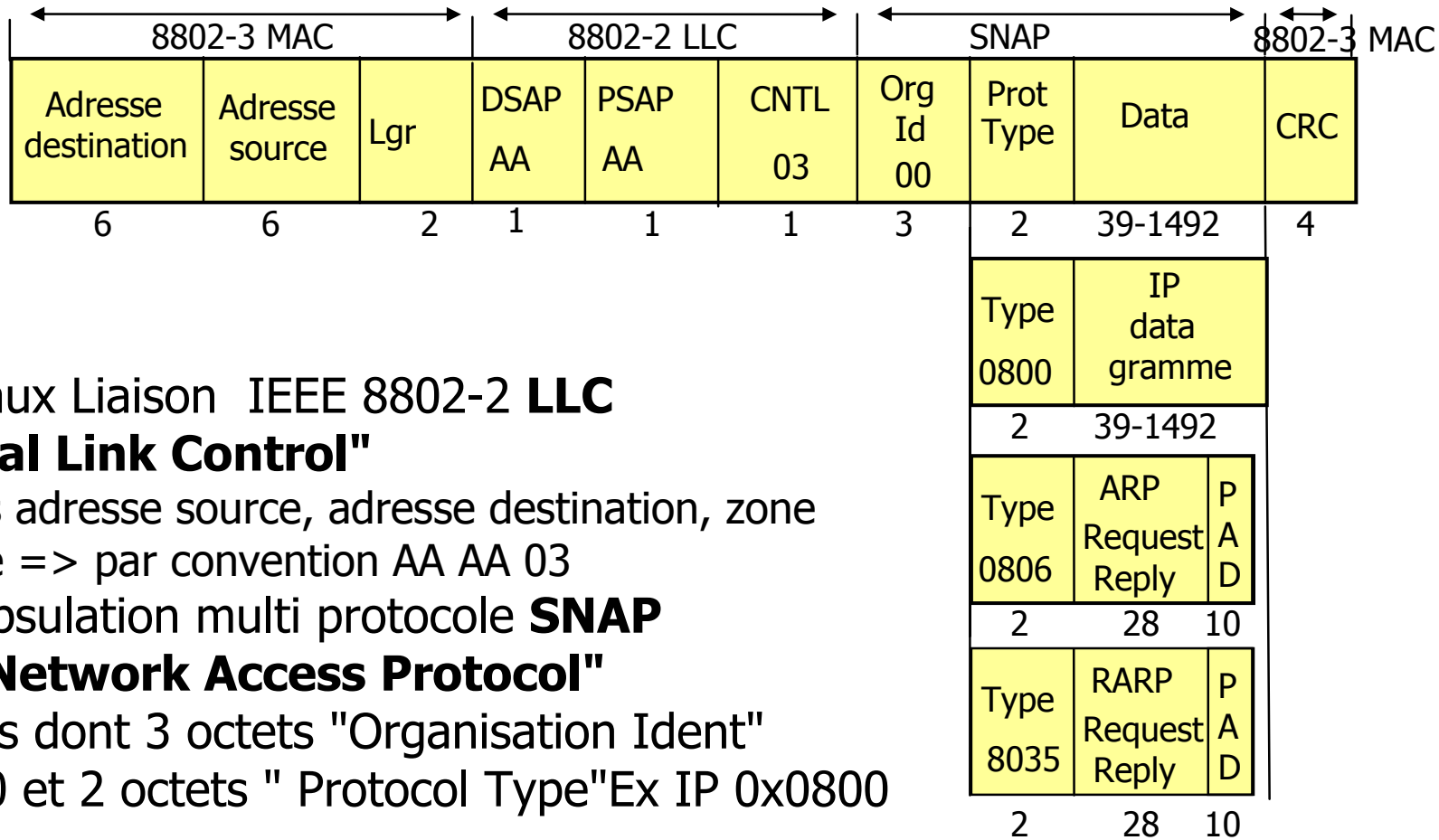
Introduction : rappel des protocoles annexes de IP



A) Couche liaison : Encapsulation Multiplexage

- **Problème abordé:**
  - Encapsuler des paquets IP dans des trames au niveau liaison.
  - Supporter conjointement sur le même réseau local différents protocoles de niveau 3 (IP, IPX, DECNET, CLNP, AppleTalk, ...).
- **Existence de différents protocoles de liaison.**
- **Cas des liaisons spécialisées : protocole PPP**
  - Encapsulation PPP => déjà vue
- **Cas des réseaux locaux : Protocoles Ethernet DIX ou IEEE 802**
  - Encapsulation **Ethernet/DIX ou LLC/SNAP**

Encapsulation IEEE 802



- Niveaux Liaison IEEE 8802-2 **LLC**

"**Logical Link Control**"

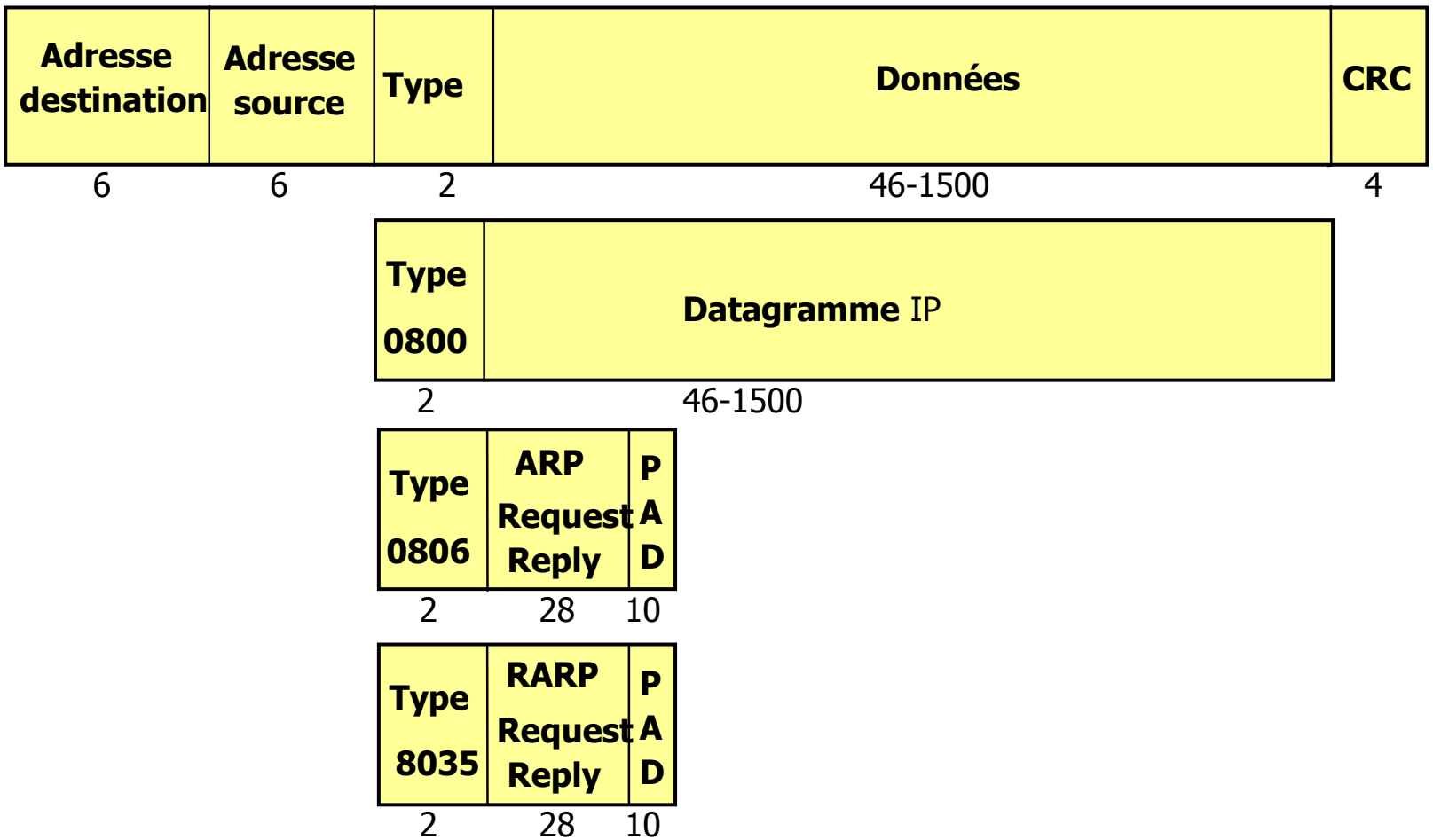
3 octets adresse source, adresse destination, zone contrôle => par convention AA AA 03

- Encapsulation multi protocole **SNAP**

"**Sub-Network Access Protocol**"

5 octets dont 3 octets "Organisation Ident" 000000 et 2 octets " Protocol Type"Ex IP 0x0800

Encapsulation Ethernet DIX



B) Relation entre adresses liaison et réseau: solutions statiques

- **Objectif : Etablir la correpsondance** entre adresses liaison (adresses MAC) et adresses réseau (adresses IP):
  - **Adresse Internet** <=> **Adresse Ethernet**

    @IP @MAC

- **Solutions statiques**
- **A) Gérer dans chaque site une table statique**

  => Mais modification obligatoire de la table à chaque changement de configuration.

  => Solution très lourde.
- **B) Utiliser des adresses IP et des adresses MAC qui se déduisent l'une de l'autre 'facilement'**

  => Solution IP V6.

Relation entre adresses liaison et réseau: Solutions dynamiques

- **Solution dynamique: gestion d'annuaire**
  - Existence de sites serveurs qui connaissent la relation d'adressage**.**
  - Utiliser un protocole client-serveur pour interroger ces serveurs et déterminer dynamiquement les relations entre adresses.
  - Très utile si le réseau est souvent reconfiguré (retrait, installation fréquente de stations de travail).
- **Eviter de renseigner sur chaque site les adresses des serveurs:** utilisation du mode diffusion des réseaux locaux.

B1) Protocole ARP : "Address Resolution Protocol"

- **Problème :** connaître l'adresse Ethernet connaissant l'adresse IP
- **Cas d'utilisation :** communication dans l'Internet sur le même réseau local
  - A sur le même réseau local que B doit dialoguer avec B (@IpB, @MacB).
  - A ne connaît que l'adresse réseau Internet @IpB.
  - A veut connaître l'adresse liaison Ethernet @MacB.
- **Fonctionnement de la recherche locale : Mécanisme de Cache d'adresse (cache ARP)**
  - Table sur chaque site qui conserve les résolutions d'adresses effectuées.
  - A cherche dans son cache local l'adresse MAC de B : si succès fin de ARP et communication avec B.
- **Fonctionnement de la recherche distante: chaque site fonctionne comme serveur de sa propre adresse.**

(1) Diffusion d'un paquet requête ARP contenant @ IpB

(2) Tous les sites du réseau reçoivent le paquet ARP et comparent l'adresse Internet proposée avec leur propre adresse

(3) B répond seul (réponse ARP) en envoyant @MacB.

(4) A dialogue avec B.

Protocole ARP : Compléments de gestion du cache

- **A ayant reçu une réponse de B :** mémorise dans son cache la correspondance (@IpB, @MacB)
- **B dans le message requête à appris la correspondance** (@IpA, @MacA) **:** apprentissage dans le cache
- **Tous les autres hôtes ont reçu la requête :** entrée si nécessaire dans leur cache (@IpA, @MacA).
- **A sa mise en marche une station diffuse une requête ARP sur sa propre adresse:** apprentissage de tout le réseau (ARP gratuit).
- **Les entrées du cache sont invalidées après une durée paramétrable :** gestion dynamique de cache pour tenir compte des modifications de l'architecture.
- **Remarque :** prolongation possible des requêtes de diffusion ARP **notion de proxy ARP** un routeur qui retransmet les requêtes ARP sur ses différentes interfaces de réseau.

Protocole ARP : Exemple (fonctionnement en UNIX)

tulipe::/users/ensinf/gerard _23 /usr/sbin/arp -a
Net to Media Table

Device	IP Address	Mask	Flags	Phys Addr
le0	mac-florin.cnam.fr	255.255.255.255		00:00:94:22:ac:74
le0	cisco-for-turbigo	255.255.255.255		aa:00:04:00:1f:c8
le0	kendatt	255.255.255.255		08:00:20:7d:72:08
le0	savitri.cnam.fr	255.255.255.255		08:00:20:04:3a:04

< Longue liste de toute la table>

tulipe::/users/ensinf/gerard _25 /usr/sbin/ping rita
rita.cnam.fr is alive

tulipe::/users/ensinf/gerard _26 /usr/sbin/arp -a
Net to Media Table

Device	IP Address	Mask	Flags	Phys Addr
le0	mac-florin.cnam.fr	255.255.255.255		00:00:94:22:ac:74
le0	cisco-for-turbigo	255.255.255.255		aa:00:04:00:1f:c8
le0	kendatt	255.255.255.255		08:00:20:7d:72:08
le0	**rita.cnam.fr**	**255.255.255.255**		**08:00:20:12:bd:d5**
le0	savitri.cnam.fr	255.255.255.255		08:00:20:04:3a:04

< Longue liste de toute la table>

B2) Protocole RARP : "Reverse Address Resolution Protocol"

- **Objectif : Déterminer une adresse IP avec une adresse Ethernet**
- **Cas d'utilisation:** une machine qui boute
  - Cas d'une machine sans disque: boute à partir du réseau (coupleur boute = coupleur Ethernet).
  - Nécessité pour utiliser un transfert de fichier simple (TFTP) de connaître l'adresse IP locale ou envoyer le binaire.
  - En ROM coupleur son adresse MAC => obtention de l'adresse IP.

- **Fonctionnement RARP**
  - La station qui amorce diffuse son adresse Ethernet sur le réseau local.
  - Un serveur RARP (nécessaire sur chaque réseau) retourne l'adresse IP.
  - Absence de réponse : retransmissions (nombre limité).
  - Tolérance aux pannes : plusieurs serveurs RARP.

- **En IPv6 ARPet RARP sont intégrés à ICMP.**

C) Protocole ICMP : "Internet Control Message Protocol" (1)

- **Objectif : réaliser différents échanges orientés administration de réseaux** (15 types de messages)
  - Messages d'erreurs.
  - Acquisition d'informations.
- **Requête-Réponse de masque (d'un sous-réseau)**
  - Pour une station sans disque qui souhaite le connaître lors de son initialisation.
- **Requête-Réponse demande de l'heure d'un site distant**
  - Pour développer une synchronisation d'horloge.
- **Acheminement de messages d'erreurs**
  - Durée de vie dépassée => paquet détruit
  - Site inaccessible ("port unreachable error") 16 diagnostics d'échec.
- **Messages ICMP de maintenance des tables de routage**
  - ICMP redirect ou messages de demande routes.

ICMP : Utilitaire ping

- **Requête-Réponse ICMP:** envoi d'un message à un hôte distant et attente d'une réponse (si le répondeur est activé).
  - **Message ICMP type ECHO_REQUEST**
  - **Message ICMP type ECHO_REPLY**
- **Commande Unix: ping**
  - ping "adresse IP" ou "nom de domaine DNS d'un hôte"
  - "nom de station" is alive et/ou statistiques de temps d'aller retour
- **Utilisation:**
  - Hôte distant opérationnel (en fait sa couche IP fonctionne).
  - Le réseau Internet permet d'atteindre un hôte distant.
  - Détection d'un réseau (ou d'un hôte distant) surchargé.
- **Autre option : ping -R** pour "record route" : chaque routeur enregistre son adresse dans le paquet.
  - Utilisation pour tracer les routes (courtes).
  - Ping n'est pas toujours disponible et longueur maximum de la route 9.

Exemple : utilitaire ping

$ping ulysse.cnam.fr

Envoi d'une requête 'ping' sur ulysse.cnam.fr [163.173.136.36] avec 32 octets de données :

Réponse de 163.173.136.36 : octets=32 temps=60 ms TTL=115
Réponse de 163.173.136.36 : octets=32 temps=50 ms TTL=115
Délai d'attente de la demande dépassé.
Réponse de 163.173.136.36 : octets=32 temps=50 ms TTL=115

Statistiques Ping pour 163.173.136.36:
    Paquets : envoyés = 4, reçus = 3, perdus = 1 (perte 25%),
Durée approximative des boucles en millisecondes :
    minimum = 50ms, maximum = 60ms, moyenne = 40ms
$

ICMP : Utilitaire traceroute

- **Objectif :** tracer ou tester une route de longueur arbitraire.
- **Solution :**
  - Emettre des datagrammes IP vers un destinataire avec une valeur de la durée de vie TTL Time To Live croissante (1, 2, 3, jusqu'à une valeur max).
  - A chaque expérience ICMP retourne un diagnostic d'erreur ICMP à l'envoyeur lorsque le TTL passe à 0 => apprentissage du routeur visité sur la route vers le destinataire
- **Commande Unix: traceroute "adresse IP" ou "nom de domaine"**
  - Permet de connaître le chemin emprunté et la durée pour atteindre chaque routeur (trois expériences pour chaque routeur).
- **Autre possibilité : traceroute -g "adresseIP"**
  - Permet également de tester les possibilités de routage par la source
  - -g routage faible : on force IP à emprunter quelques routeurs spécifiés.
  - -G routage strict : on force IP à parcourir strictement une route.

Exemple : utilitaire traceroute

```
$/usr/etc/traceroute cyr.culture.fr
    traceroute to cyr.culture.fr (143.126.201.251), 30 hops max, 40 byte packets
     1  internet-gw                    (163.173.128.2)   0 ms      0 ms      0 ms
     2  renater-gw                     (192.33.159.1)    0 ms     10 ms      0 ms
     3  danton1.rerif.ft.net           (193.48.58.113)   110 ms    80 ms     90 ms
     4  stlamb3.rerif.ft.net           (193.48.53.49)    100 ms   130 ms    100 ms
     5  stamand1.renater.ft.net        (192.93.43.115)    90 ms    60 ms     50 ms
     6  stamand3.renater.ft.net        (192.93.43.17)     70 ms   100 ms     90 ms
     7  rbs1.renater.ft.net            (192.93.43.170)   130 ms   120 ms      *
     8  Paris-EBS2.Ebone.NET           (192.121.156.226) 110 ms    90 ms    100 ms
     9  icm-dc-1.icp.net               (192.121.156.202) 220 ms   110 ms    220 ms
    10   icm-dc-1-F0/0.icp.net         (144.228.20.101)   200 ms   230 ms   290 ms
    11   Vienna1.VA.Alter.Net          (192.41.177.249)   250 ms   210 ms   240 ms
    12   Falls-Church4.VA.ALTER.NET(137.39.100.33)      330 ms   220 ms   180 ms
    13   Falls-Church1.VA.ALTER.NET(137.39.8.2)         270 ms   290 ms   230 ms
    14   Amsterdam2.NL.EU.net          (134.222.35.1)     380 ms   410 ms   460 ms
    15   Amsterdam1.NL.EU.net          (193.242.84.1)     350 ms   380 ms   310 ms
    16   134.222.30.2                  (134.222.30.2)     150 ms   490 ms   530 ms
    17   Rocquencourt.FR.EU.net        (193.107.192.18)   340 ms   340 ms   330 ms
    18   143.126.200.203               (143.126.200.203) 300 ms   410 ms      *
    19   cyr.culture.fr                (143.126.201.251)  460 ms   220 ms   290 ms
$
```

D) Protocole IGMP : "Internet Group Management Protocol"

- **Objectif :** Utiliser les capacités de transmission IP pour réaliser des diffusions sur groupe => Deux problèmes à résoudre :

  => **Disposer d'un protocole d'appartenance à un groupe:** messages vers les routeurs pour entrer et sortir d'un groupe de diffusion

  => **Construire un routage en diffusion** : exemple de protocoles de routage en diffusion sur groupe DVMRP, PIM=>Notion de routeur diffuseur
- **Protocole d'appartenance à un groupe dans l'Internet: IGMP.**
  - **Rappel :** Adresses IPV4 de groupes (classe D) ou Adresse IPV6 (FF/8)
  - **Association à un groupe (abonnement) :** en émettant une requête à son routeur (diffuseur) de rattachement comportant l'identificateur du groupe et l'interface qui doit recevoir les messages.
  - **Désassociation (désabonnement) :** autre requête.
  - **Surveillance :** un routeur diffuseur émet une requête périodique sur toutes les interfaces ou il doit délivrer des messages diffusés pour vérifier l'opérationnalité.
  - **Maintenance de table :** utilisant ces requêtes et ces réponses un routeur gère sa table locale des groupes.
- **En IPv6 IGMP est intégré à ICMP.**

IP

Conclusion

Succès du protocole IP

- **Un protocole de niveau 3 en expansion considérable:** HTTP, SMTP puis les applications puis la téléphonie et la TV.
- **IP couvre les besoins d'interconnexion** : pour des réseaux de transmission de données numériques => **IP est devenu le réseau par excellence d'intégration de services.**
- **IP intègre** toutes les nouvelles offres de moyens de communication physique.
- **Les routeurs IP ont été améliorés en performances** et IP est complété par un protocole de commutation rapide à circuits virtuels (MPLS)
- **IP possède une version 6 :** pour supporter le développement en termes d'adressage et de hiérarchisation.

Compte tenu de son extension et de son adéquation IP devrait continuer d'être utilisé comme protocole unificateur de niveau 3 pendant très longtemps.

Incertitudes

- **Est ce que IPV6 va réussir à se déployer** dans de bonnes conditions en même temps que IPv4 va régresser?
- **Est ce que IP va pouvoir s'ouvrir à l'ensemble large et divers d'usagers et de besoins qui sont visés:**
  - Support de la qualité de service pour des application variées multimédias.
  - Dans le cadre d'un développement énorme => passage à l'échelle.
- **Comment IP et plus généralement l'Internet va gérer son propre effet sur la société?**

Bibliographie Internet Protocol

- W.R. Stevens "**TCIP/IP Illustrated, The protocols**" , Addison Wesley
- S.A. Thomas "**IPng and the TCP/IP protocols**" Wiley
- A.S. Tannenbaum "**Computer Networks**" Prentice Hall
- Cisco "**Internetworking Technology**" Publication interne

Niveau Transport
"Transport Layer"

I) Problèmes et solutions au niveau transport
II) Exemple des protocoles et services de transport dans l'INTERNET

Niveau Transport
"Transport Layer"

Chapitre I

Problèmes et solutions au niveau transport

Généralités: choix de conception
Adressage
Gestion des connexions
Transfert de données

Objectifs du niveau transport

- **Offrir un service de transmission d'informations** pour les besoins d'un utilisateur de session.
  - **Communication de processus à processus** (de bout en bout),
  - **Selon des objectifs de qualité** de service: transport fiable, multimédia
  - **Efficace** : en performances.
  - **Indépendant** de la nature des réseaux traversés: LAN, MAN, WAN
- **Importance du niveau transport**
  - **La couche transport n'est pas une couche de plus** dans une architecture de réseau.
  - **C'est la couche qui résume toutes les couches associées à la transmission** (les couches basses).
  - **C'est la base** sur laquelle repose une application répartie.

Nécessité de la couche transport: 1) Problème d'adressage

- **Niveaux liaison et réseau** : adresses utilisées = Adresses d'équipements physiques (contrôleur de communication, NIC).
  - Exemple : adresses Ethernet, adresses X25, adresses IP.
- **Niveau transport** (communication de bout en bout) : adresses utilisées = adresses d'applications (processus).
- **Toute solution** visant à utiliser les adressages physiques pour faire communiquer des processus est plus ou moins "bricolée".
  - Exemple : Utilisation d'adresses X25 ou Ethernet 'étendues'
  - On ne peut pas faire dépendre la structure de l'adresse d'une application de la nature du réseau qui permet de l'atteindre.
- **Le niveau transport doit offrir un adressage en propre:**
  - On utilise une adresse "logique d'activité" (adresse transport)
  - On réalise la mise en correspondance d'une adresse transport avec une "adresse physique d'hôte" qui est support de l'application.
  - L'adresse de transport est indépendante du réseau utilisé.

Nécessité de la couche transport: 2) Gestion de connexions et QOS

- **Protocole avec connexion**
  - **Permet la spécification de qualité de service** selon les besoins des applications.
  - **Le transport adapte aux besoins des applications** les niveaux physique, liaison et réseau.
  - **Exemples : TCP, RTP et ISO 8072**
- **Protocole sans connexion**
  - **Chaque donnée circule** en utilisant essentiellement des adresses de transport pour atteindre le destinataire.
  - **Protocole simplifié** pour des communications **rapides.**
  - **Exemples : UDP et ISO 8072 ad1**

Exemple de qualité de service: Le contrôle d'erreurs

- **Un utilisateur d'une architecture de réseaux ne doit pas avoir à se préoccuper des erreurs de transmission.**
  - Taux d'erreur résiduel acceptable (erreurs non détectées, non corrigées). Exemple 10^{-12}/bit pour des données informatiques de criticité normale.
  - La plupart des services de transmission offerts par les services réseaux n'offrent pas cette qualité.
    - Niveau liaison sans contrôle d'erreur : Ethernet, ATM.
    - Niveau réseau avec contrôle : X25 problème de panne de commutateur.
    - Niveau réseau à datagrammes sans contrôle d'erreur : IP.
- **Le niveau liaison ou le niveau réseau sont insuffisants pour les besoins en contrôle d'erreurs des applications.**
- **On reporte au niveau transport le lieu principal de correction des erreurs de transmission.**

Choix de conception du niveau transport

- **Services offerts:** selon les options de conception d'une couche transport.
  - **Gestion des connexions** avec négociation de qualité de service.
  - **Fragmentation** (segmentation).
  - **Groupage** (concaténation).
  - **Multiplexage/éclatement** des connexions de transport sur des connexions de réseaux.
  - **Contrôle d'erreur, de flux, de séquence**.
  - **Propriétés de qualité** de service temporelle.
  - **Contrôle de congestion** (du réseau par contrôle d'admission au niveau transport).

Analyse des concepteurs du transport Internet TCP/UDP

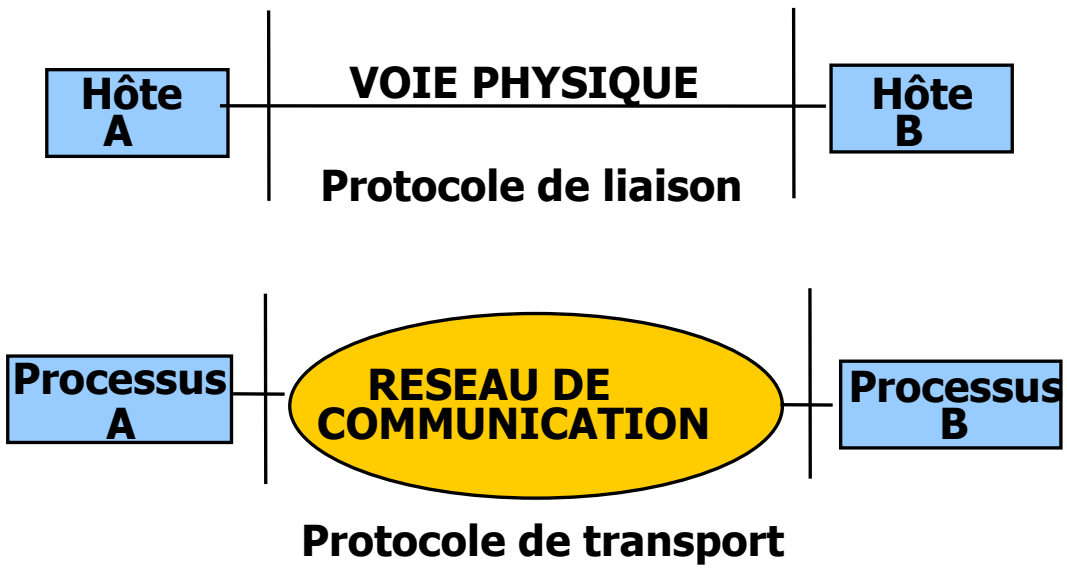
- **TCP : un protocole complet pour des données informatiques.**
  - **Résout les problèmes non traités par IP** (contrôle d'erreur, flux, séquence)
  - **=> TCP un transport fiable**
  - **=> Utilisable pour tous les types de réseaux.**
  - **Pas de négociation initiale**: fonctions toutes disponibles.
  - **Peu adapté** aux réseaux aux fonctionnalités élevées (X25).
  - **Beaucoup de traitements à effectuer**: solution lente.
  - Une solution très **optimisée** pour le service rendu.
- **UDP : un protocole très simplifié**
  - Pour aller vite: réalise uniquement un adressage transport.
- **RTP : un protocole de transport à QOS.**
  - Satisfaction de contraintes temporelles pour le multimédia.

Analyse des concepteurs du transport OSI

- **Etude de l'écart** entre le service à fournir par la couche transport et le service qu'elle obtient de la couche réseau.
- **Permet de déterminer** les fonctions à mettre en oeuvre au niveau de la couche transport.
  - **Meilleur est le service offert par le réseau** (surtout en termes de contrôle d'erreur, livraison en séquence)
  - **Plus simples sont les fonctions** du protocole de transport.
- **Conséquences**
  - **Définition d'un protocole à plusieurs classes (5 classes).**
  - **Classes de transport de base** (classe 0, 1, pour des niveaux réseaux performants) à **complexe** (classe 4 , très complète pour des niveaux réseaux médiocres).
  - **Protocole complexe** (volume de fonctions), négociation d'ouverture de connexion complexe => difficile à implanter.

Particularités des problèmes de réalisation du niveau transport

- **Le service de transport** parait **voisin** du service de **liaison**.
  - Communication en **point à point**.
  - Qualité de service comprenant **contrôle d'erreur et de flux.**
- **Pourrait-on alors utiliser au niveau de la couche transport un protocole de liaison ?**
  - Le problème principal : la différence des moyens de transmission utilisés



Modes de pannes supportées

- **Modes de pannes franches et transitoires dans les systèmes de communication :**
  - **Panne franche** : coupure totale des communications.
  - **Panne transitoire** : pertes aléatoires de message.

- **Au niveau physique comme au niveau réseau** : les deux niveaux supportent des coupures ou des pertes de messages.

=> Peu de différences dans les traitements de niveau liaison et de niveau transport.

Modes de pannes temporelles

- **Panne temporelle** : Délai de transmission anormalement élevé par rapport aux spécifications.
- **Niveau physique** : capacité de stockage du canal très faible (délai de transmission, délai de propagation)
  - => **le temps de transmission** d'une trame est **déterministe**.
- **Niveau réseau** : Le réseau peut **stocker des paquets** pendant un certain temps et ne les **délivrer qu'après ce délai**.
  - **Transmissions à longue distance** avec **surcharge et congestion**.
  - **Des paquets peuvent arriver après un long délai.**
  - => **Le temps de transmission** d'un paquet est **aléatoire (non déterministe).**
  - **Problème important au niveau transport : les "vieux paquets".**

Problèmes de causalité (déséquencement)

- **Niveau physique :**
  - Une voie de communication physique **a un comportement "causal":** les messages émis dans un certain ordre arrivent dans le même ordre.
  - **=> Les suites binaires ne peuvent se dépasser.**

- **Niveau réseau :**
  - Le réseau (en mode datagramme) peut **déséquencer ou dupliquer** les paquets transmis.
    - En raison des **algorithmes de routage** dans les réseaux à datagrammes.
    - En raison des **problèmes de délai de propagation** (déjà évoqués)
  - **=> Les paquets sont possiblement déséquencés.**

Problèmes et solutions au niveau transport

Problèmes d'adressage au niveau transport

Nature des adresses de transport
Adresse = Nom local du processus

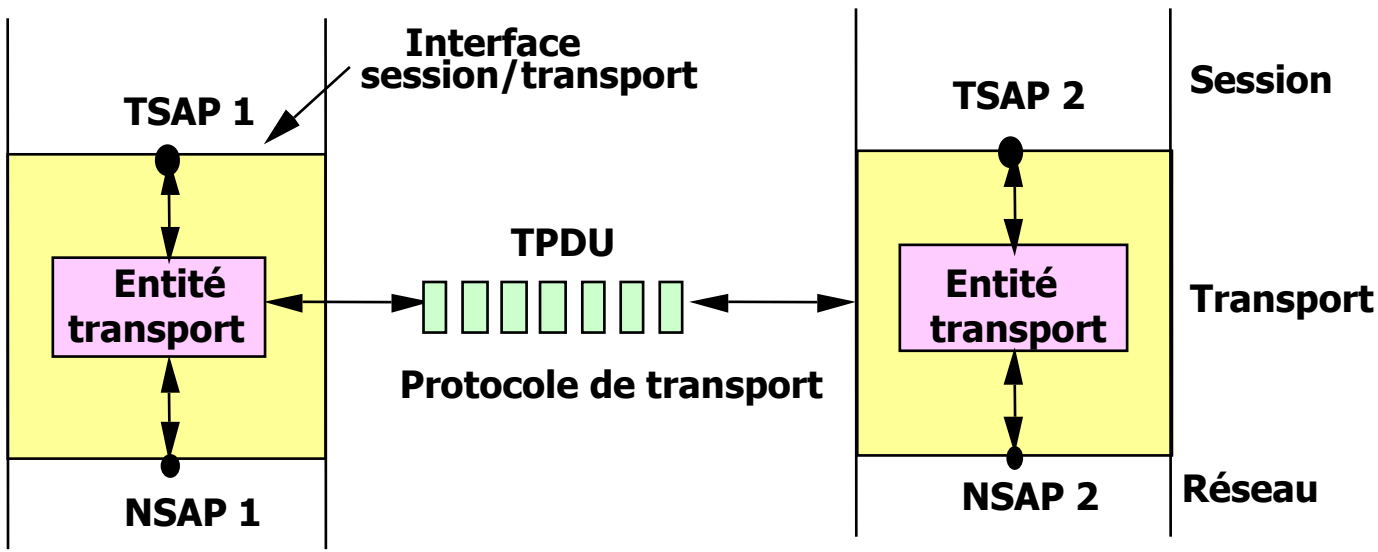
- **Nom local de processus** (source ou destinataire): nom du processus dans le systèmes d'exploitation de l'émetteur ou du destinataire.
- **Solution très médiocre pour résoudre le problème d'adressage :** variantes de syntaxe des noms de processus sur différentes machines.
  - **Gestion des noms** selon les différentes syntaxes.
  - **Difficulté de migration** d'une machine à une autre.
  - **Difficulté de remplacement** d'un processus en cas de panne (changement du nom).
  - **Difficulté pour la répartition de charge** : service rendu par plusieurs instances équivalentes d'un même code.

Nature des adresses de transport Adresse = nom global réseau

- **Nom global : L'adresse d'une source ou d'un destinataire est définie dans une syntaxe réseau unique pour toutes les machines.**
  - **Exemple : en TCP un entier sur 16 bits.**
- **Solution indépendante de la** syntaxe des noms de processus sur différentes machines.
- **Nom global = nom d'une file d'attente de message.**
  - **Besoin de plusieurs communications pour un processus**
  - **Indirection des messages** par un point de passage intermédiaire auquel se raccrochent les processus usagers.
  - **Terminologies pour les files de messages : Point d'accès de service transport, TSAP** "Transport Service Access Point"**,**
    - **Boite à lettre, Port, Porte, "Socket" ou prise.**

Détermination des adresses des processus communicants

- **L'ouverture d'une connexion de transport suppose la connaissance des adresses**
  - **TSAP** ("Transport Service Access Point") (Internet numéro de port) : sélection du processus qui rend le service.
  - **NSAP** ("Network Service Access Point") (Internet adresse IP) : sur quel site se trouve ce processus => Le routage réseau se charge ensuite de le trouver.
- **Problème de mise en correspondance TSAP <-> NSAP.**



Adressage dans une communication client-serveur

- **Scénario d'une communication**
  - **1/ L'utilisateur B (le serveur)** se rattache au TSAP **b** et se met en attente d'une arrivée de connexion.
  - **2/ L'utilisateur A (le client)** qui désire établir une connexion de transport avec B émet une requête de connexion sur le destinataire (NSAP b, TSAP b) en indiquant son adresse d'appelant (NSAP a, TSAP a).
  - **3/ Le serveur et le client** peuvent dialoguer.
- **Le scénario marche parce que B était en attente sur le TSAP** b et que A est supposé connaître b par son adresse.
- **Comment A peut-il connaître cette adresse :**
  - Problème de liaison analogue de l'édition des liens dans les programmes.
  - Solutions de liaison statique ou dynamique.

1) Solution de liaison statique

- **Connaissance des adresses codée dans les programmes**
  - **Tous les utilisateurs de B** ont la connaissance du TSAP de B.
  - **Analogie avec les numéros de téléphone réservés**: 15 , 18.
  - **On installe de façon définitive** un service sur un adresse donnée.
- **Principe simple mais limité.**
  - **Les serveurs dont l'adresse est fixe sont nécessairement assez peu nombreux** => services à caractère générique système.
  - **Gaspillage d'adresses** si l'on utilise une adresse permanente pour un serveur très spécifique ou rarement sollicité.
  - **Il est nécessaire d'utiliser aussi des adresses de TSAP non permanentes**.

2) Liaison dynamique par processus créateur de serveur

- **Notion de processus créateur de serveur (processus "logger")**
  - Processus dont le rôle est de **créer à la demande** des instances de serveur.
  - L'adresse du créateur de serveurs ("logger") est **permanente et connue de tous** les utilisateurs (solution de liaison statique).
- **Fonctionnement avec créateur de serveur sur un hôte NSAP B.**
  - Le client A se connecte au processus créateur et définit le service B à créer:
    - Soit sous la forme d'un nom logique de service (dans une liste)
    - Soit sous forme du nom d'un fichier accessible contenant une image binaire chargeable
  - Le processus créateur alloue une adresse de TSAP libre (**TSAP b**), créé une instance de B et lui attribue l'adresse allouée => il la communique au client A.
  - Le client A se connecte au **TSAP b** pour obtenir le service souhaité.
- **Avantages/inconvénients de la solution**
  - **Solution de désignation, de création et de liaison dynamique complète.**
  - **Ne marche que pour les serveurs pouvant fonctionner au coup par coup** (création à la demande) Exemple : serveur de compilation, serveur d'horloge,….
  - **Inutilisable pour un serveur qui fonctionne en permanence**

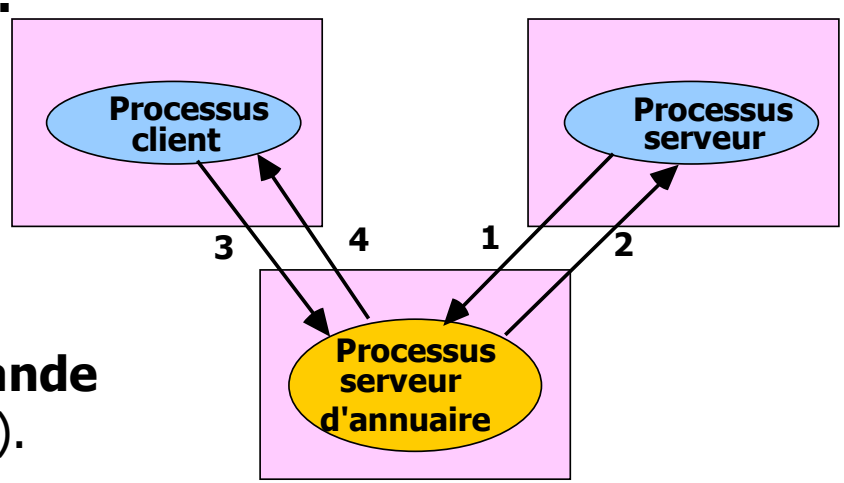
    Exemple : serveur de fichiers.

3) Liaison dynamique par utilisation d'un serveur d'annuaire

- **Principe de la solution**
  - Consiste à faire appel à un processus **serveur d'annuaire**, dont l'adresse (Nsap, Tsap) est permanente et connue de tous (solution statique).
  - Ce processus **gère un annuaire** de services auprès duquel les serveurs peuvent s'enregistrer et qui fournit des renseignements concernant un service:
    - Principalement son **adresse** (Nsap, Tsap)
    - Eventuellement d'autres attributs ou fonctions (protection, description/typage du service, ...).
  - **Solution analogue** de ce que réalisent les renseignements téléphoniques.

Troisième solution : liaison dynamique par utilisation d'un serveur d'annuaire

- **Étapes 1, 2 : Enregistrement dans une base de données des noms de serveurs.**
- 1) B établit une connexion avec le serveur d'annuaire. B émet une requête avec :
  - Le nom du service (chaîne de caractères).
  - L'adresse réseau d'accès au service.
  - L'adresse transport d'accès au service.
  - Tout attribut complémentaire si nécessaire.
- 2) Le serveur d'annuaire **acquitte la demande** d'enregistrement (absence d'homonymie, ...).



- **Étapes 3, 4: Liaison entre un client et un serveur.**
- 3/ A établit une connexion de transport avec le processus serveur d'annuaire.
  - Requête spécifiant le nom logique du service dont il désire connaître l'adresse.
- 4/ Réponse du serveur : l'adresse demandée. A se déconnecte de l'annuaire.

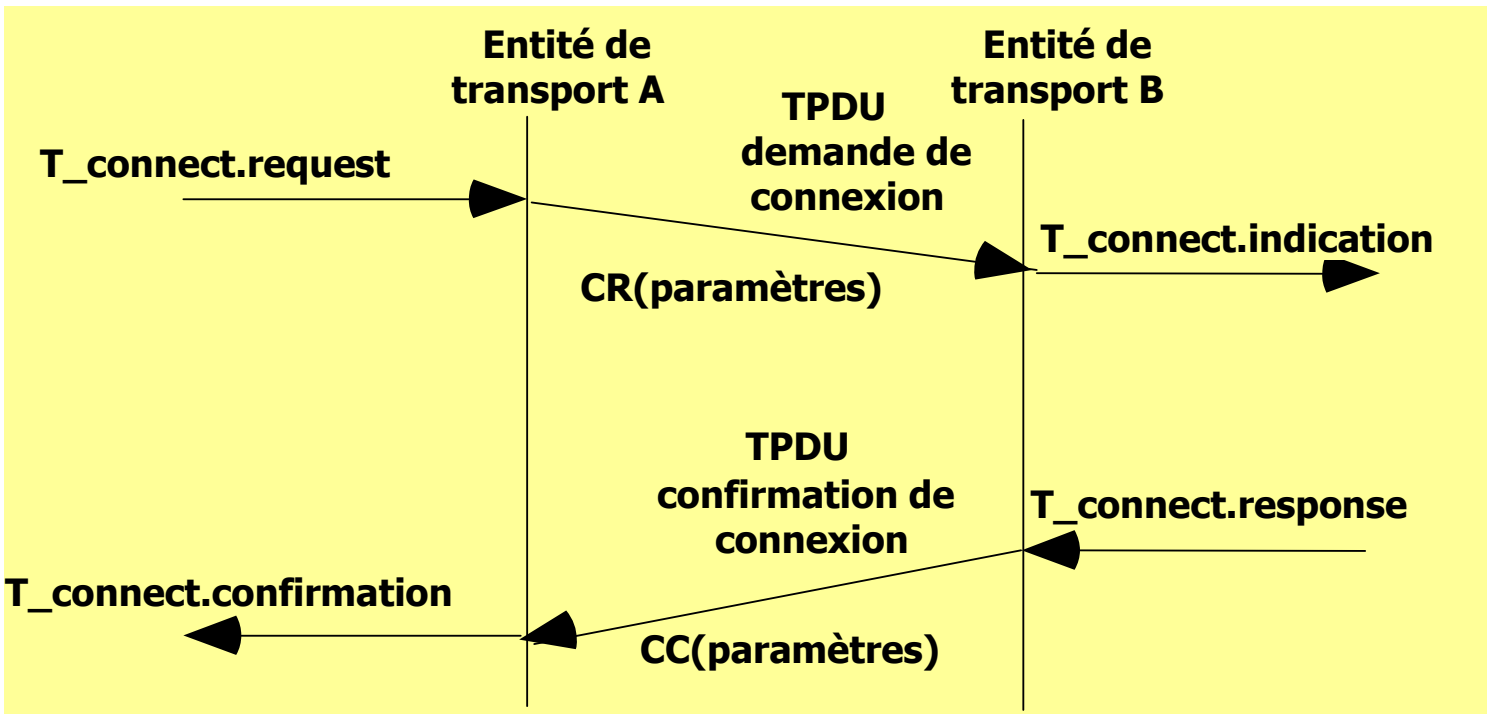
- A n'a plus qu'à établir la connexion avec le serveur.

Problèmes et solutions au niveau transport

Problèmes de gestion des connexions au niveau transport

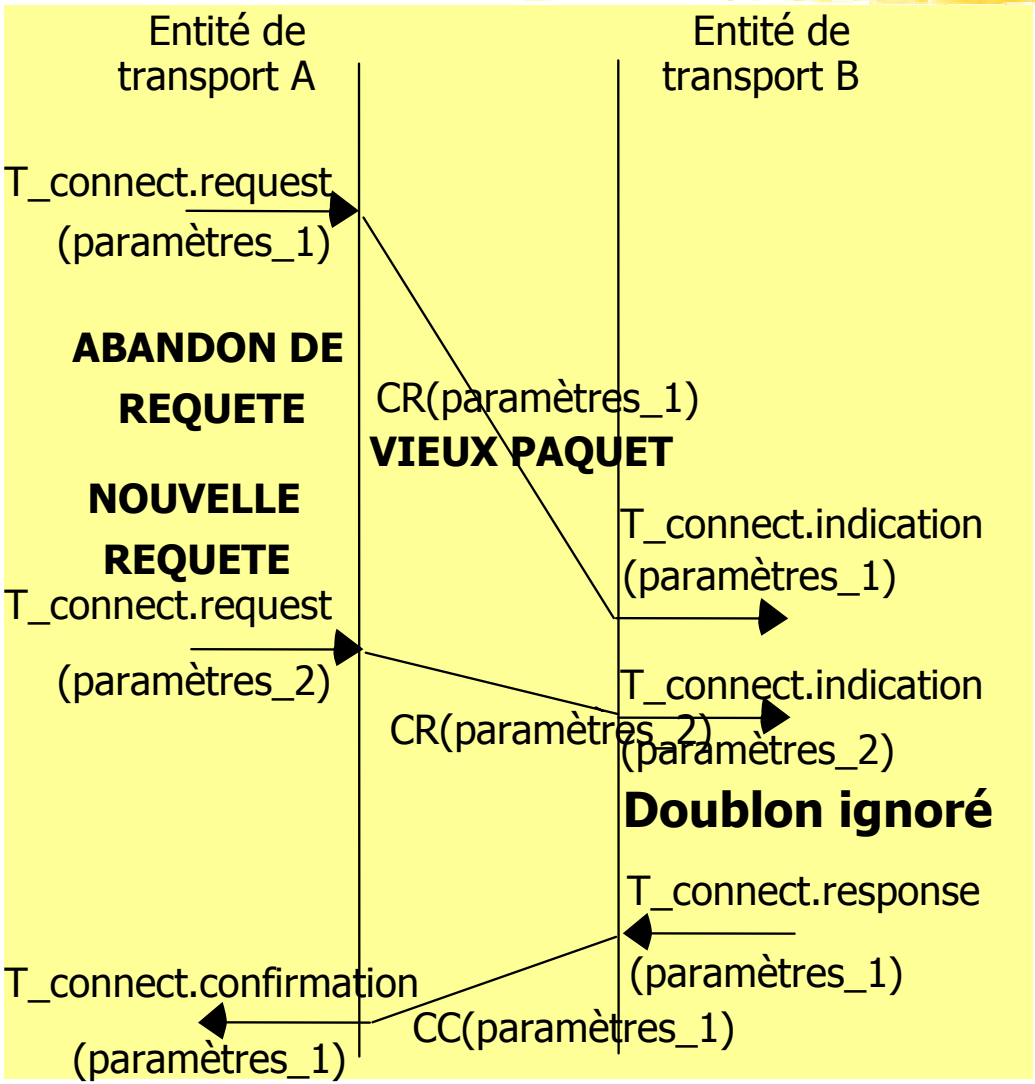
A) Problèmes d'ouverture de connexion
Rappel du schéma de base

- **Rappel : L'accord confirmé de base ('hanshake simple')**



- **Problème:** le réseau peut mémoriser des paquets pendant un temps très long (éventuellement non borné dans l'hypothèse asynchrone).

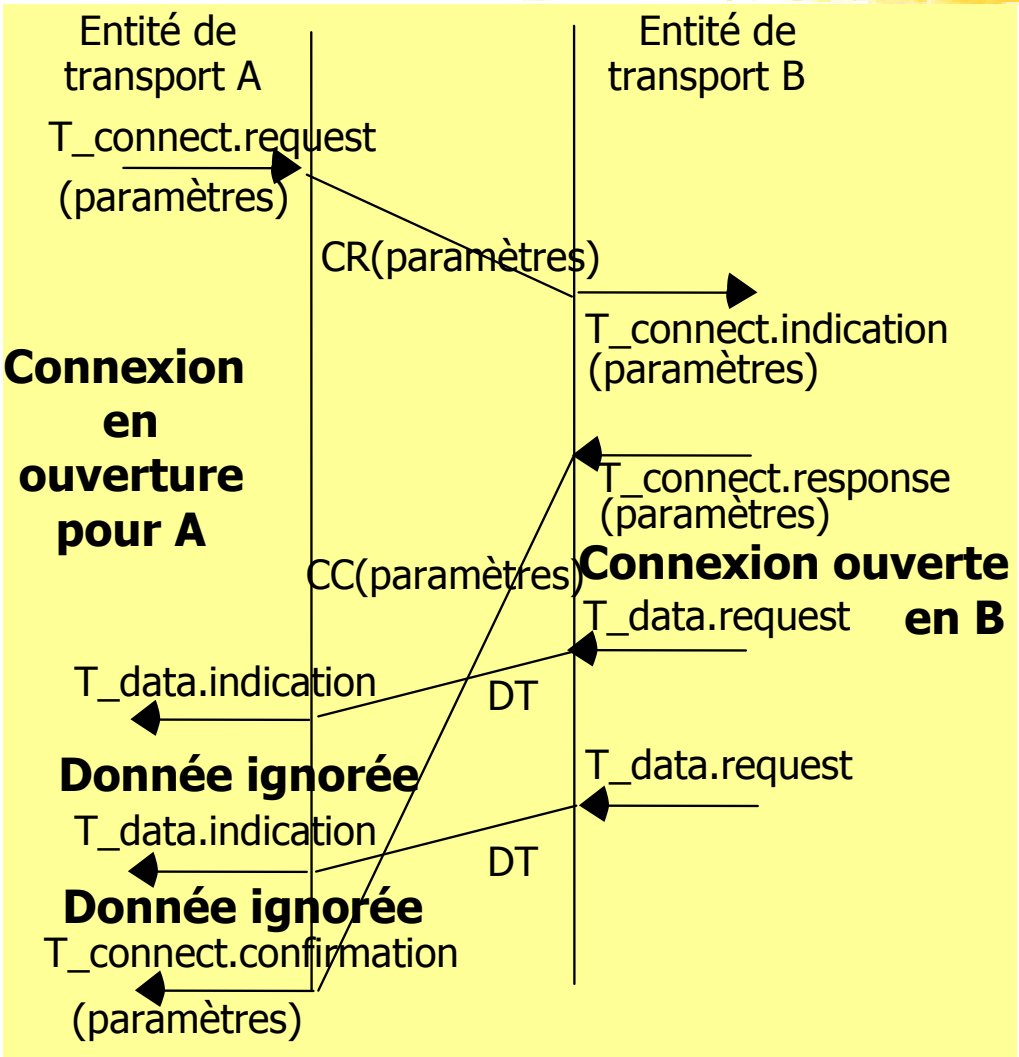
Exemple 1 : comportement incorrect du à un délai élevé



- **Seul responsable de l'erreur** le délai élevé de transmission du premier paquet.
- **Il n'y a pas d'erreur** de transmission et **la séquence des paquets est respectée**.
- On voit que le problème provient surtout **de l'absence d'identification des connexions.**

- **Solution à apporter:**
  Notion de référence de connexion
  et mécanisme de gel des références

Exemple 2 : comportement incorrect du au déséquencement



- **Responsable** le séquencement des paquets **non respecté.**
- D'où une connexion **considérée comme ouverte par un site B** et **non ouverte par l'autre** A (demi-ouverte).
- **La situation peut durer longtemps** si le paquet CC arrive très retardé (ou même n'arrive jamais).

- **Solution à apporter:**

  Ouverture de connexion confirmée en trois étapes ("three way handshake")

Notion de référence de connexion

- **Besoin de créer des références de connexion :**
  - **Un moyen de désigner** une connexion de transport : une communication entre un émetteur et un destinataire.
  - **Un usager peut gérer sur un même TSAP** plusieurs connexions simultanément avec le même ou plusieurs usagers.
- **Problème posé :**
  - Générer à la demande des références uniques pour de nombreux couples d'usagers avec de nombreuses connexions ouvertes.
- **Solution :**
  - Pour chaque site associer un entier unique pour le site et le nom du site : notion d'estampille ('timestamp').
  - Former la référence unique à partir du couple des estampilles émetteur destinataire.
- **Comment créer des entiers uniques ?**

Solutions pour les références : 1) Adresses TSAP à usage unique

- **Référence entière basée sur l'adresse de TSAP**
  - On doit utiliser une adresse de TSAP différente à chaque connexion
  - On s'oblige à ne gérer qu'une connexion par TSAP.
  - Chaque fois qu'on a besoin d'une nouvelle référence de connexion, on crée une nouvelle adresse de transport.
- **Il ne peut plus y avoir d'ambiguïté** entre messages circulant sur des connexions différentes ou en cours d'établissement.
- **Lorsqu'une connexion est fermée**, toutes les messages qui lui étaient associés sont inutilisables (sans destinataire).
- **Solution coûteuse** en adresses de TSAP.
- **Solution qui rend très difficile la liaison** (découverte de TSAP).

Solutions pour les références : 2) Références volumineuses

- **Utilisation d'une référence de connexion longue** contenant des informations de différenciation pour la rendre unique.

- **A) Référence basée sur la date et l'heure d'ouverture de connexion**.
  - Solution correcte chaque référence est différente.
  - Nécessite une gestion correcte du temps absolu.
- **B) Référence basée sur un numéro de séquence.**
  - **Utilisation séquentielle** d'un espace de références grand => il ne repasse par les mêmes numéros qu'avec une faible probabilité.
  - **Nécessite une sauvegarde en mémoire stable** du dernier numéro généré car en cas de panne on ne doit pas réutiliser le même numéro (démarrage à 0 au boute).
- **C) Tirage aléatoire d'une référence** dans un grand espace de numérotation tel qu'on ne peut tirer deux fois la même référence qu'avec une faible probabilité (exemple sur 32 bits ou mieux sur 64 bits).
  - Nécessite un bon générateur d'entiers : avec probabilité faible de collision.
  - Solution probabiliste : on n'est jamais sur de l'absence de collision.
  - La référence unique aléatoire peut-être utilisée comme base de numérotation en séquence des messages.

Solutions pour les références : 3) Référence courte avec gel

- **Utilisation d'une référence de connexion courte réutilisable.**
  - **En général un entier qui correspond à une entrée** dans la table des descripteurs de connexions.
  - **A chaque fois qu'une connexion doit être ouverte** on utilise une entrée libre de la table.
  - **L'index entier est sur n bits, 2^n est petit** et il y r**éutilisation fréquente** des références par des connexions successives.
  - **Il ne faut pas qu'il y ait d'ambiguïté** entre les messages de la précédente connexion utilisant l'entrée et ceux de la nouvelle connexion.
- **Solution** : **gel de référence**
  - **Une référence ne peut être réutilisée** pendant une période suivant la fermeture d'une connexion **=> Gel de la référence**.
  - **Le gel permet à tous les paquets en circulation** sur la connexion fermée d'être reçus et purgés.
  - **On doit disposer d'un mécanisme de limitation de la durée de vie** dans le réseau ('time to live') pour armer un délai de garde correspondant au gel.

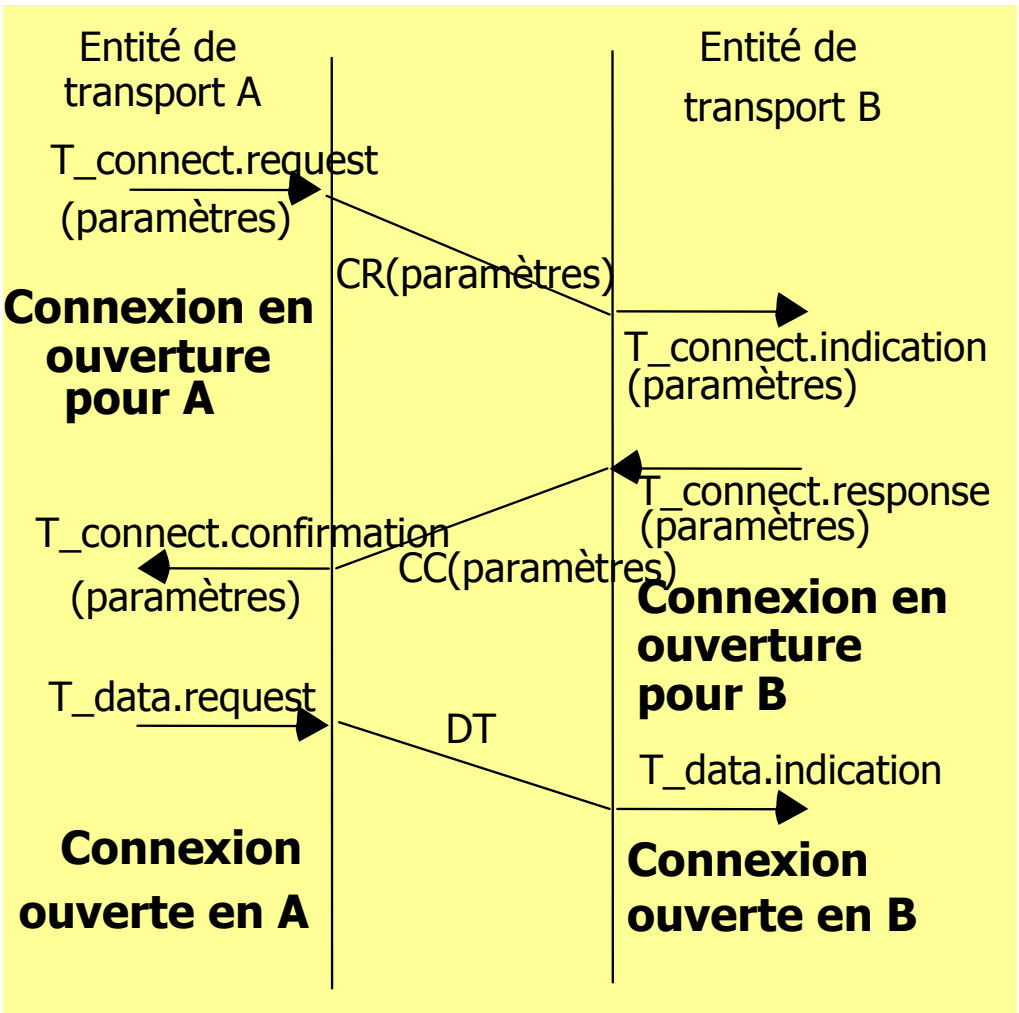
La solution en TCP

- **Utilisation d'une solution 'hybride'.**
- **La référence est basée sur une horloge 'temps réel'**
  - Cette horloge est très lointainement reliée au temps absolu.
- **Il faut plutôt considérer que l'horloge de TCP fonctionne comme un générateur de nombres aléatoires**
  - Solution de génération d'entiers probabilistes sur 32 bits.
  - La référence est utilisée comme base de numérotation en séquence des messages.
- **Crainte des collisions entre numéros :** utilisation en plus d'un mécanisme de **gel de référence**
  - **Notion de durée de vie** des paquets qui permet de dimensionner le gel.

Ouverture de connexion en trois étapes ("three way handshake")

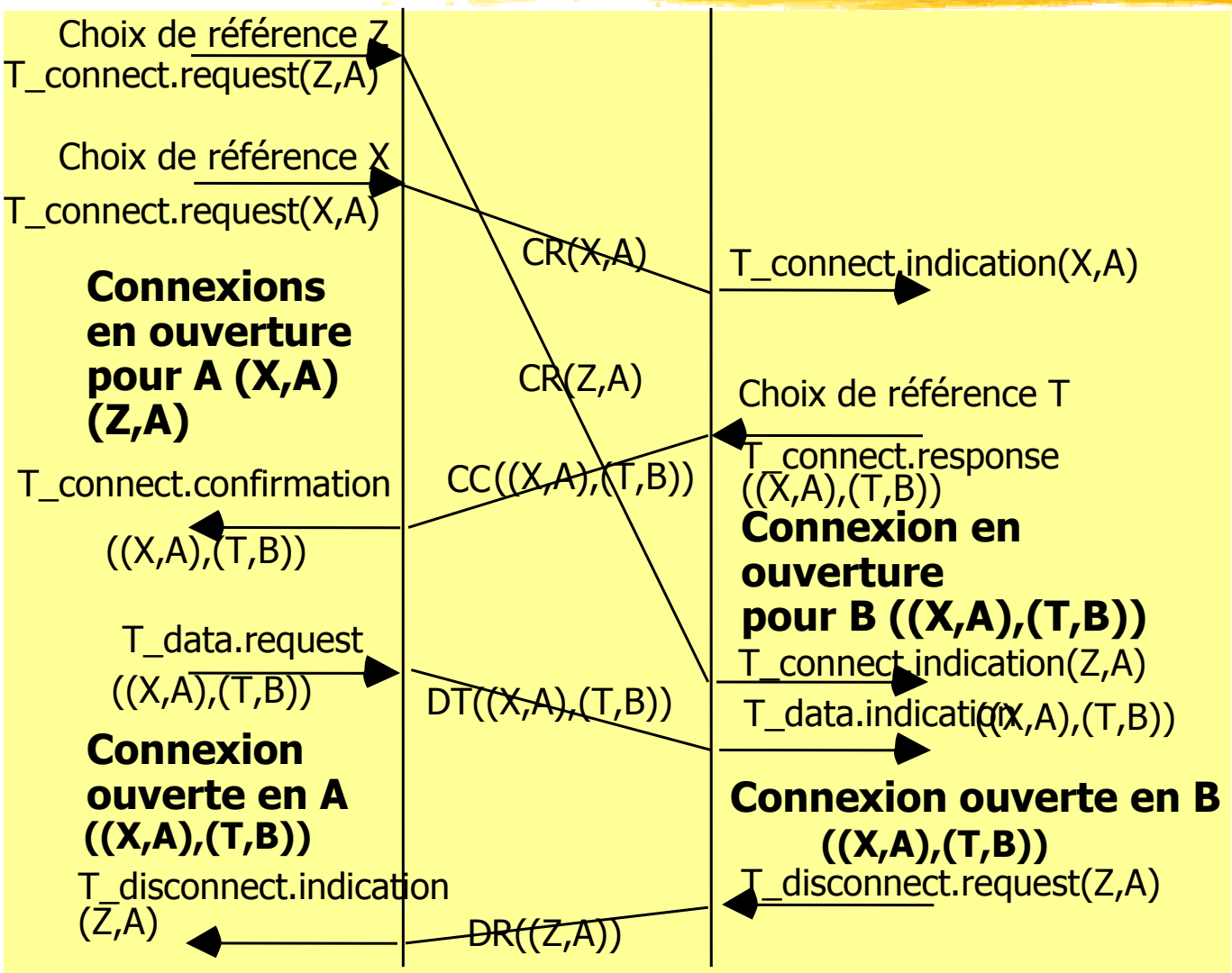
- **Problème posé**: perte d'unités de données de protocole en phase d'ouverture (ouverture de connexion incomplète à cause de messages perdus, anciens ou déséquencés).
- **L'objectif visé**: Avant de commencer à transférer effectivement des données on doit atteindre un état tel que:
  - le demandeur de l'ouverture sait que le destinataire accepte l'ouverture,
  - le destinataire sait que le demandeur sait qu'il accepte l'ouverture de connexion.
- **Solution 1**
  - Ignorer les TPDU ("Transport Protocol Data Unit") de données qui arrivent avant la confirmation d'ouverture connexion => des pertes de données, alors que le réseau est parfaitement fiable.
  - Ceci ne définit pas comment on atteint l'ouverture confirmée (la connexion peut rester indéfiniment en ouverture).
- **Solution 2**
  - Stocker les TPDU de données qui arrivent avant la confirmation de connexion, afin de les présenter lorsque la connexion est établie.
  - On peut avoir à stocker de gros volumes de données.
  - Il n'y a toujours pas de définition de la confirmation d'ouverture.

L'ouverture confirmée en trois étapes



- **Principes**
- **On interdit** à l'entité appelée d'envoyer des données (de considérer que la connexion est ouverte)
- **L'appelé ne peut répondre** qu'une TPDU de confirmation.
- **Jusqu'à ce qu`il ait reçu** une TPDU qui lui confirme la bonne réception de sa TPDU de confirmation de connexion par l'appelant.
- **On utilise pour cela une troisième TPDU**
- Selon le protocole une donnée si l'appelant a des données à envoyer ou une TPDU spécifique (troisième message d'ouverture).

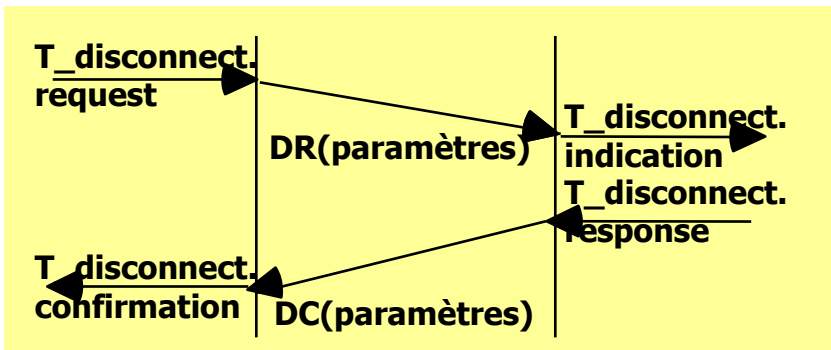
Exemple d'échange confirmé en trois étapes avec références



- **Remarques:**
- Il ne peut y avoir d'ambiguïté entre les connexions référencées Z et X dont l'ouverture concurrente est en cours.
- Le site B a le choix d'accepter ou de rejeter (ici par un disconnect) l'une des deux connexions.

B) Problèmes de fermeture de connexion

- **Fermeture de connexion = atteindre un consensus** sur la connaissance de la libération.
  - Si la connexion doit être libérée A et B doivent atteindre (assez rapidement) un état ou ils décident **tous les deux que la connexion est fermée**:
  - Pour éviter **l'existence indéfinie de connexions** demi-ouvertes
    - L'un a décidé de fermer et l'autre ne le sait pas.
  - Pour terminer les échanges dans une situation claire du point de vue applicatif.
- **Solution de base: accord confirmé**
  - T_disconnect.request, DR
  - T_disconnect.indication
  - T_disconnect.response, DC
  - T_disconnect.confirmation.



- **Ce protocole ne marche pas en présence de tous types de délais de transmission et de pertes de messages.**

Problèmes de consensus : le problème des deux armées

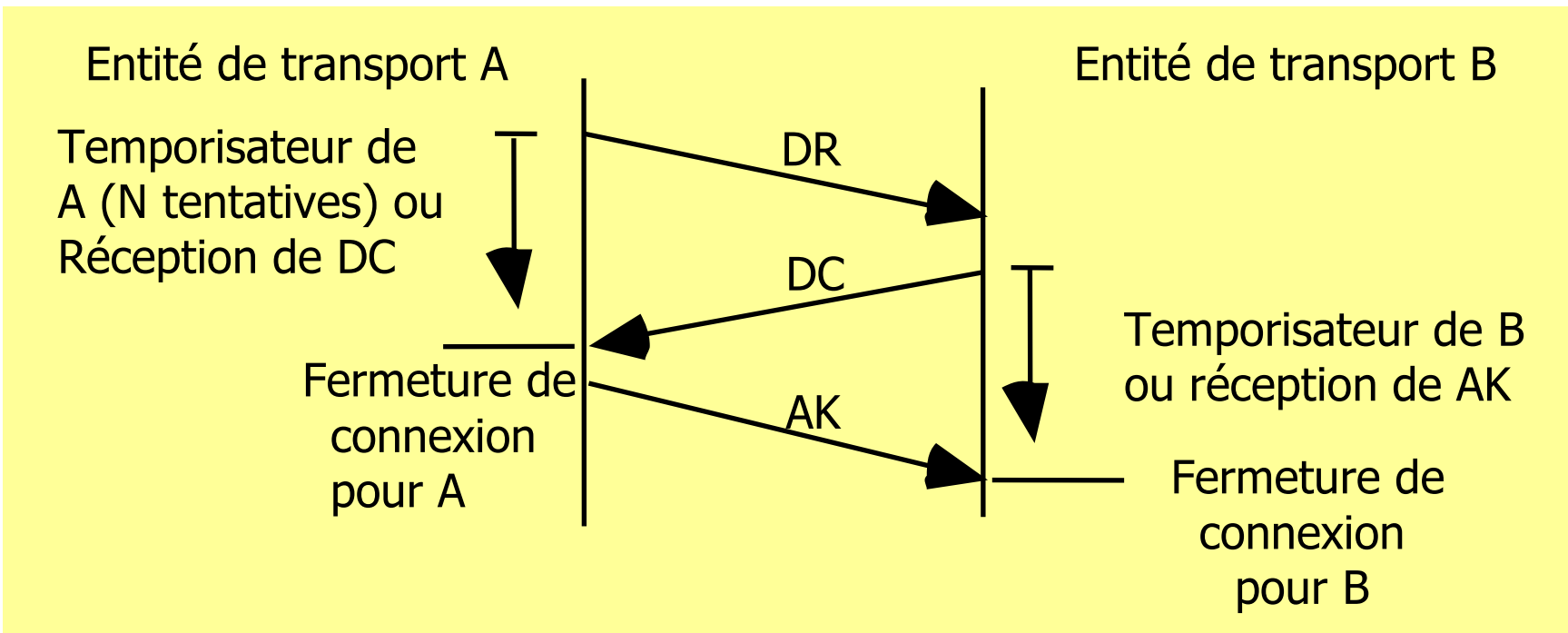
- **La formulation d'un problème de consensus entre deux entités communicantes** analogue au problème de déconnexion:
  - Une armée A campe dans une vallée.
  - L'armée B (son adversaire) est séparée en deux corps d'armées B1 et B2 qui occupent deux collines distinctes séparées par l'armée A.
  - A est plus forte que chacun des deux corps d'armées B, mais A est moins forte que B dans le cas ou les deux corps B1 et B2 attaquent de façon coordonnée.
  - Le seul moyen pour coordonner l'attaque des B est l'usage de messagers qui traversent la vallée et qui sont susceptibles de mettre très longtemps, d'être capturés ... en raison des ennemis
- **Solution asynchrone au problème des deux armées**
  - **Existe-t-il un protocole déterministe :** qui marche dans tous les cas.
  - **En mode message asynchrone :** quels que soient les pertes ou les retards (délais de transmission non bornés).
  - **Permettant aux commandants B de gagner avec certitude :** de se mettre d'accord sur l'heure d'attaque ? (de résoudre le problème du consensus à deux).

Impossibilité d'une solution déterministe en mode asynchrone

- **Présentation informelle du mode de raisonnement.**
- **Recherche d'une solution à deux messagers.**
  - Le commandant de B1 envoie au commandant de B2 le message "Attaque demain« .
  - Le commandant de B2 reçoit le message et fait répondre "D'accord".
  - Ce dialogue n'est pas concluant, car le commandant de B2 ne peut pas savoir avec certitude si sa réponse a bien été reçue (non perdue ou retardée après la date proposée).
- **Recherche d'une solution à trois messagers.**
  - On utilise à un protocole en trois étapes.
  - B1 doit accuser réception de la réponse de B2 à sa proposition.
  - Alors, c'est le commandant de B1 qui ne peut pas savoir si sa confirmation est bien arrivée.
  - S'il y a perte de la confirmation, B2 ne bougera pas, B1 attaquera seul et sera vaincu.
- **Recherche d'une solution à N messagers.**
  - Avec N messagers, le même problème se pose : l'émetteur du Nième message n'est jamais sur de savoir si son message a été reçu ou non.
  - Pour toute solution à N messagers comme le dernier message peut ne pas arriver il ne peut donc être essentiel pour la solution et on doit pouvoir s'en passer.
  - Donc on peut enlever les derniers messagers les uns après les autres.
  - Comme il n'existe pas de solutions à deux messagers il n'existe pas de solution (raisonnement par récurrence).

Solution probabiliste et synchrone pour la déconnexion de transport

- **Solution probabiliste :** une solution qui ne marche pas en toutes circonstances de pannes et de délais de transmission.
- **Solution synchrone :** basée sur des délais de transmission bornés permettant de régler des délais de garde.
- **Schéma général** d'une solution:

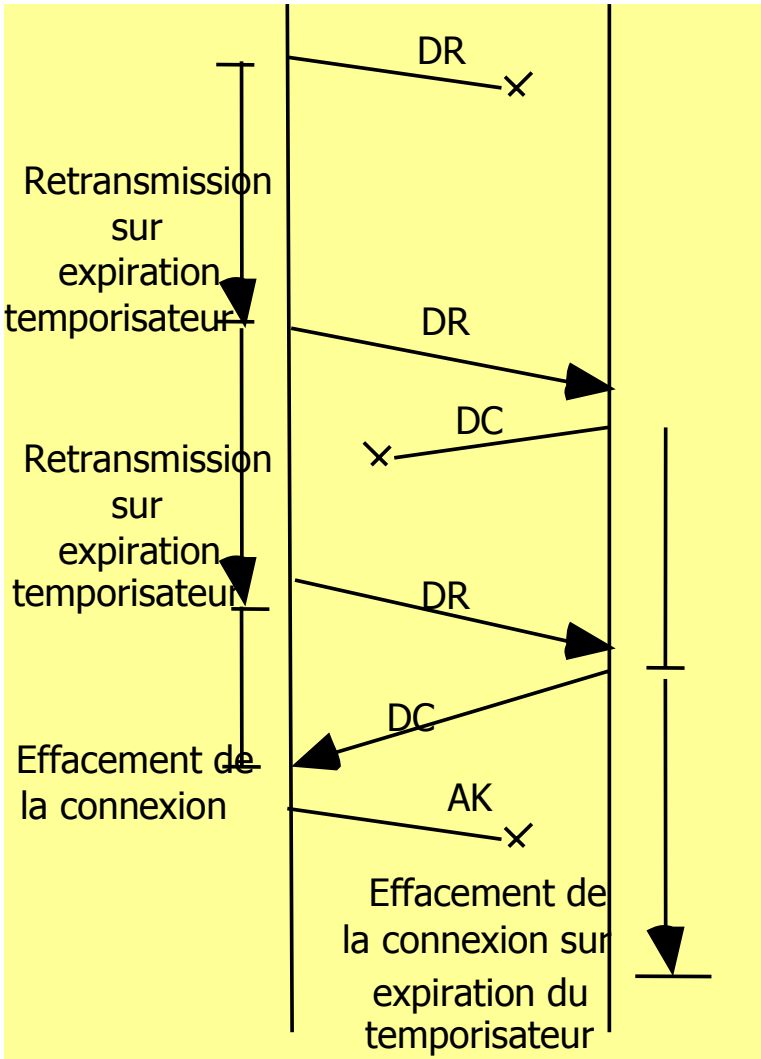


Détails du fonctionnement de la déconnexion

- **Action de A**
  - L'entité de transport A désirant mettre fin à une connexion envoie une TPDU de demande de **déconnexion DR** et arme un temporisateur.
  - A échéance, si elle n'a pas reçu de réponse, elle **retransmet la TPDU.**
  - Au bout de **N retransmissions** sans réponse ou sur réception d'une confirmation DC, l'entité **A ferme la connexion.**
- **Action de B**
  - L'entité de transport B recevant la demande de déconnexion **répond par une TPDU de confirmation** de déconnexion et arme un temporisateur.
  - A **échéance** du temporisateur ou sur réception d'un **acquittement** de sa confirmation, elle **ferme** effectivement la connexion (elle enlève de sa table des connexions ouvertes les informations concernant la connexion considérée).
  - L'entité de transport A initiatrice de la libération **accuse réception de la confirmation** en envoyant une TPDU d'acquittement et en fermant la connexion de son côté.

Analyse du protocole de déconnexion

- **Cas pour lesquels le protocole fonctionne**
  - Basés sur la réussite de certaines communications et sur l'existence de délais de garde.
  - Permettant de régler des temporisateurs en fonction des délais de propagation des messages sur le réseau.
- **Cas pour lequel le protocole ne fonctionne pas**
  - Il existe une probabilité non nulle pour que les demandes de déconnexion ou leurs réponses se perdent toutes.
  - A ferme la connexion sans que B le sache ou B ferme la connexion sans que A le sache => Notion de connexion demi-fermée.



Analyse du protocole de déconnexion

- **Solution synchrone (basée sur le temps) et probabilistes aux connexions demi-fermées**
  - On utilise un **délai de détection d'inactivité** de la connexion : si aucune TPDU n'a été reçue pendant ce délai, une entité est autorisée à fermer la connexion de son côté **(solution synchrone).**
    - Exemples : A sur émission de DR ou B sur émission de DC (temporisateurs)
  - Cette solution amène à fermer des connexions qui ne devraient pas l'être uniquement **parce que les temps de réponse sont trop élevés** par rapport aux délais de garde **(solution probabiliste).**
  - Pour que ça fonctionne il faut que les messages aient un temps de transmission borné (**transmission à délai borné ou "synchrone"**) qui permet de régler le délai de garde d'inactivité et qu'on accepte une probabilité (qui doit être très faible) de fermer des connexions.
- **Conclusion**
  - **Dans le monde des réseaux en présence de pannes (pertes de messages) et délais de transmission, on ne construit de solutions industrielles que probabilistes et synchrones.**
  - **La probabilité d'échec doit rester acceptable.**

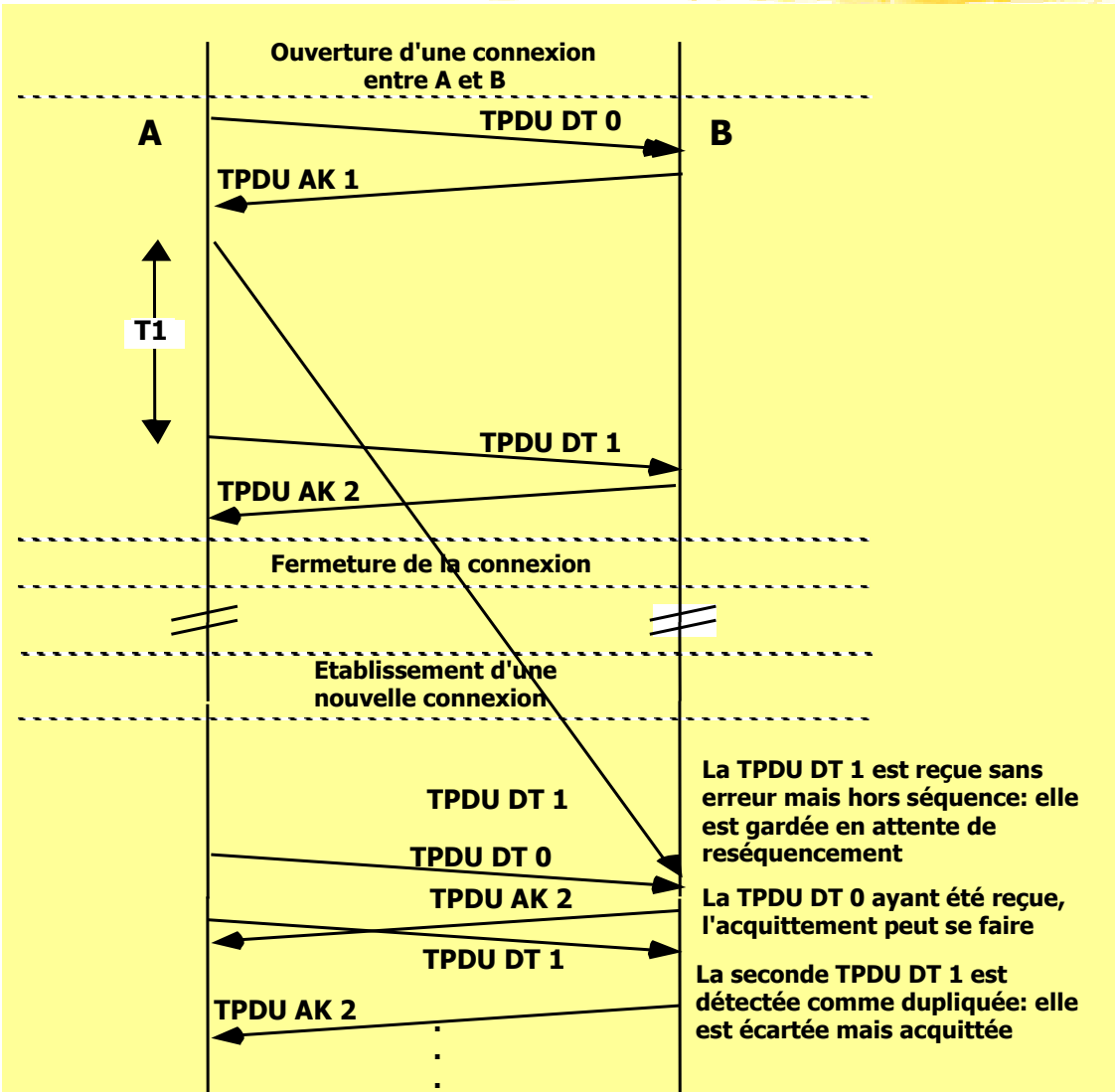
Problèmes et solutions au niveau transport

Problèmes de transmission des données au niveau transport

Contrôle d'erreur au niveau transport

- **Code détecteur d'erreur**
  - On utilise plutôt une somme de contrôle (de type parité) rapide à calculer par programme (par rapport à un code polynomial).
- **Numéros de séquence**
  - La livraison en séquence assuré par une numérotation des TPDU.
- **Acquittements positifs et temporisateurs**
  - Le contrôle d'erreur est assuré comme au niveau liaison par des protocoles à acquittement positif et retransmission sur délai de garde.
- **Solutions identiques à celles des protocoles de liaison.**
- **Difficulté du contrôle d'erreur de transport**
  - Problèmes posés par les vieux paquets et les dé-séquencements.

Exemple 1 : Interférence entre paquets sur des connexions successives



Solutions possibles

- Références de connexion dans les données.

=> Identification des paquets.

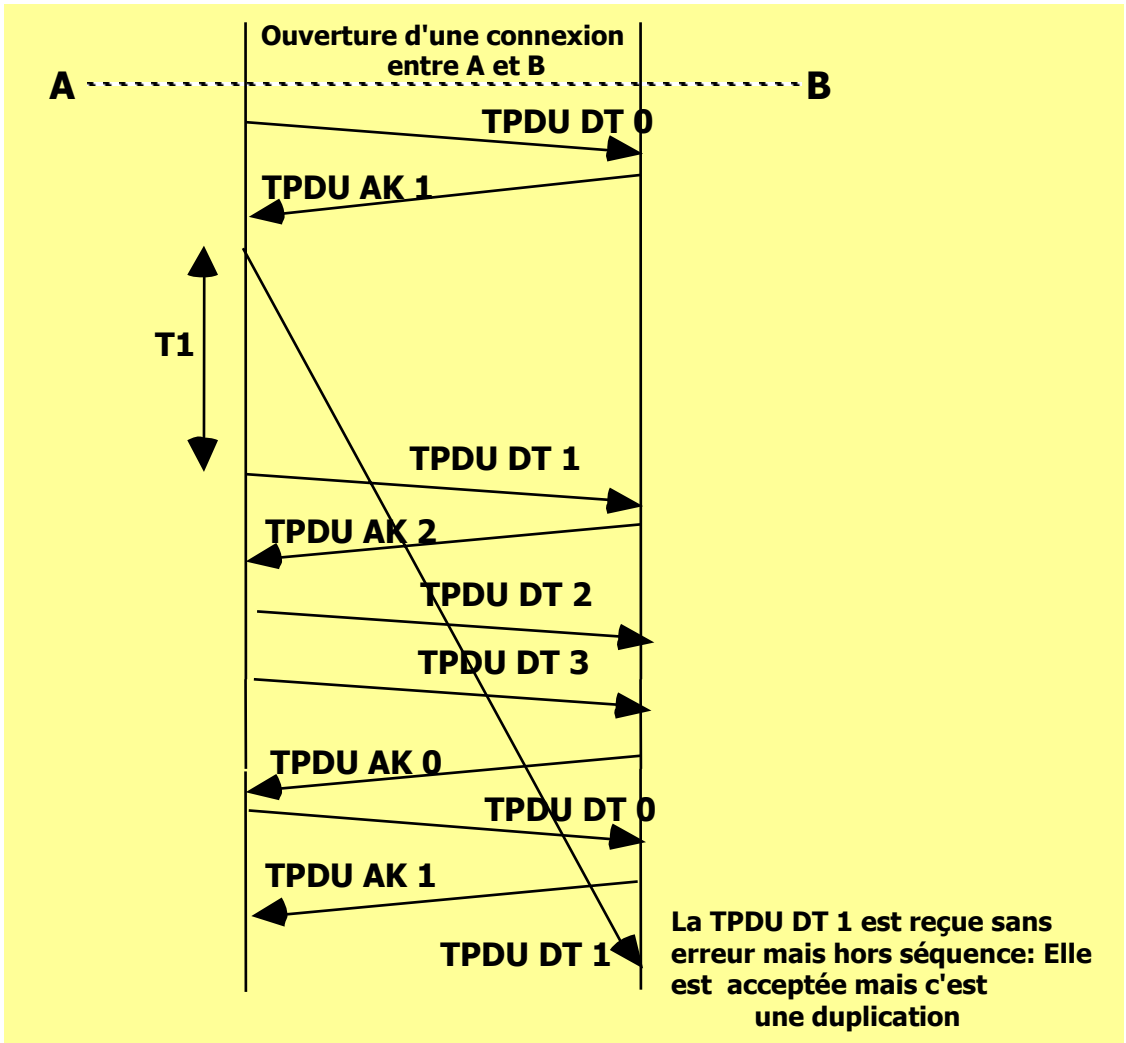
- Utilisation du gel des références.

=> Pas de réception dans une connexion ultérieure.

- Changement de l'espace des numéros de séquence utilisés.

=> Acceptation dans une nouvelle fenêtre de réception.

Exemple 2 : Interférence entre paquets de la même connexion



Solution

- Utilisation d'un numéro de séquence de grande taille (par exemple sur 32 bits ou mieux sur 64 bits).
- La probabilité de réutilisation à court terme d'un même numéro pendant une connexion doit être très faible.

Contrôle de flux au niveau transport

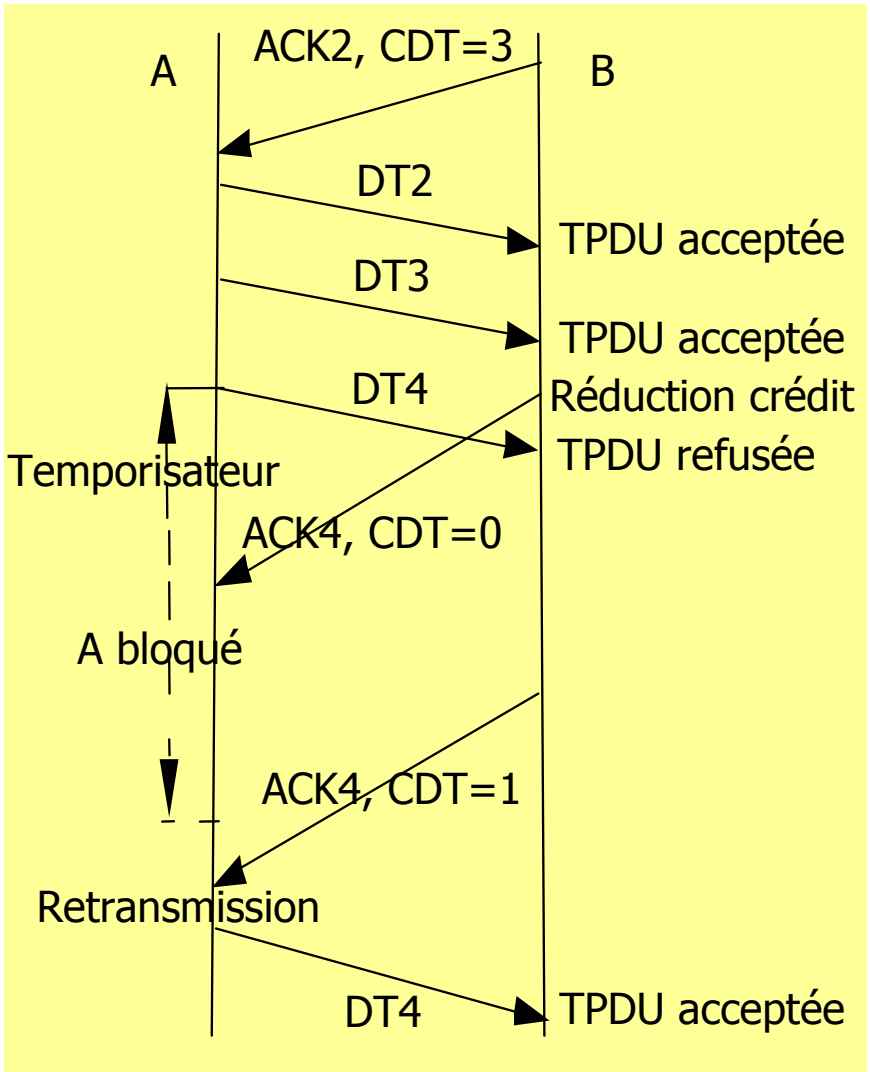
- **Problème de contrôle de flux.**
  - Adaptation de la vitesse du processus émetteur à celle du récepteur.
  - En l'absence de contrôle de flux, des unités de données sont détruites à leur arrivée, faute de tampons libres.
- **Problèmes posés par le contrôle de flux de transport.**
  - Le réseau sous-jacent achemine plus ou moins vite les informations selon sa charge.
  - Le site récepteur prend en compte les messages au niveau transport plus ou moins rapidement selon sa charge.
  - Les traitements d'application prennent plus ou moins de temps selon les données à donner.
  - **=> Très grande variabilité des vitesses de réception.**
  - **=> Nécessité d'un mécanisme de régulation très adaptatif**
- **Remplacement des fenêtres de taille fixe par des fenêtres de taille variable.**

Contrôle de flux par crédit variable

- **Notion de crédit CDT** :
  - **Une zone supplémentaire** dans les données émises par le récepteur et qui sont porteuses d'un acquittement positif AK ('piggybacking').
  - **C'est le nombre d'unités de données que l'émetteur peut émettre en anticipation** (que le destinataire est prêt à recevoir) à partir du numéro d'acquittement porté dans le segment.
  - Le récepteur **accepte** les messages de **numéro AK** au **numéro AK+CDT-1**.
- **Définit un contrôle de flux par fenêtre glissante de taille variable** car le récepteur peut modifier le crédit en fonction de ses capacités.
  - Le récepteur peut accroître la taille de la fenêtre dynamiquement
  - Le récepteur peut réduire la taille de la fenêtre dynamiquement.
- **Remarque**
  - **Le crédit est donné par la définition en absolu** des numéros de séquence autorisés : autorisation d'émettre dans la fenêtre de AK à AK + CDT - 1.
  - **La définition relative par autorisation d'envoi de CDT messages en plus** est impossible puisque les crédits doivent pouvoir être répétés en cas de perte.
    => A chaque répétition **on accorderait l'envoi de CDT messages en plus.**

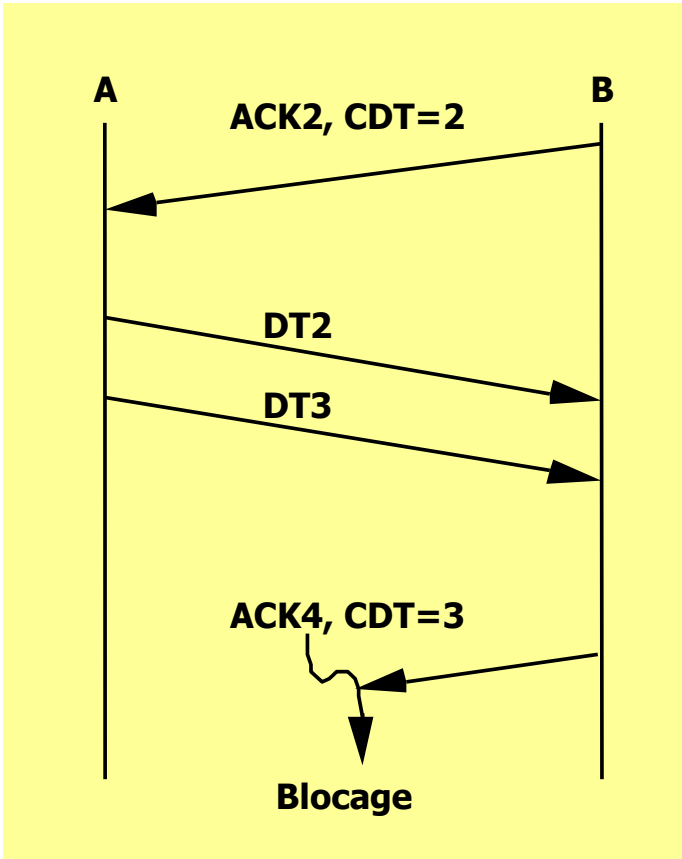
Problèmes de gestion des crédits : 1) Réduction de crédit

- **Problème** : la réduction de crédits peut entraîner la perte de TPDU de données.
- **Exemple:**
  - **Perte de TPDU données** à la suite d'une réduction de la limite supérieure de fenêtre.
  - **Bien que le réseau soit fiable.**
  - **La TDPU DT 4 arrive hors fenêtre** puisque le destinataire B a réduit entre temps le crédit d'émission de l'expéditeur A.
- **Solution:**
  - Retransmission comme si la donnée avait été bruitée.
  - Utilisation d'acquittements positifs et de temporisateur.



Problèmes de gestion des crédits : 2) Interblocage de perte de crédit

- **Problème** : La perte d'un message de crédit peut conduire à un interblocage de message.
- **Exemple :**
  - Interblocage du à une perte de message porteur de crédit.
- **Solution:**
  - Répétition des messages porteurs de crédits.



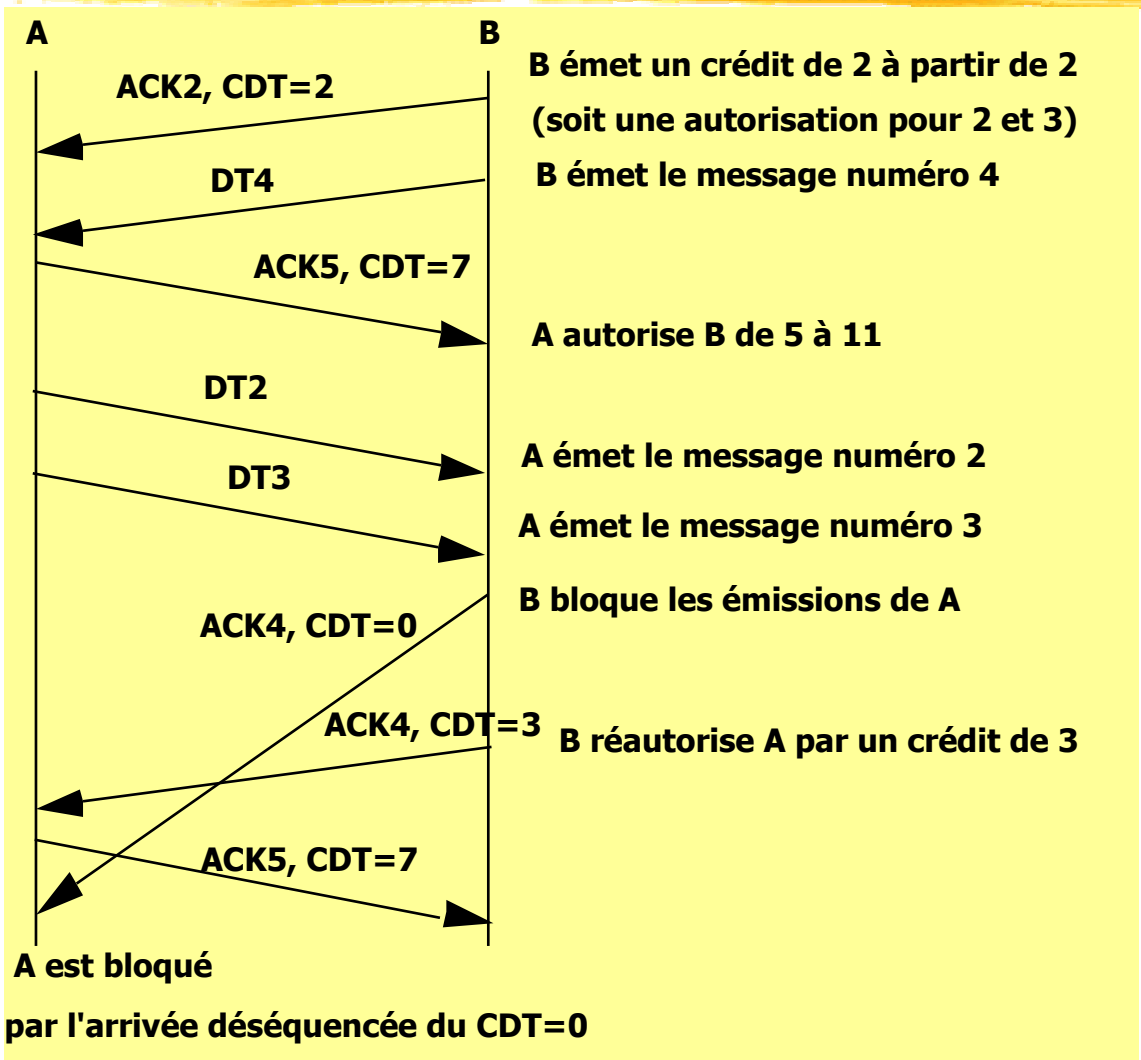
Problèmes de gestion des crédits
Solutions à la perte de crédit

- **1- Le récepteur répète périodiquement** ses messages d'augmentation de crédit (sur temporisateur).
  - Trafic périodique d'acquittements avec crédit (ou "Idle Data Transfer").
  - => Tôt ou tard le site distant reçoit une copie du dernier crédit.

- **2- Le récepteur arme un délai de garde TW** après chaque message d'acquittement/crédit et répète le message à échéance uniquement si l'émetteur n'a pas repris ses émissions entre temps (il considère qu'il est perdu).
  - **Si TW est trop faible:** on risque de réexpédier des acquittements/crédits à un rythme élevé pendant un temps important, ce qui consomme inutilement les ressources du réseau.
  - **Si TW est très élevé:** on rallonge la période de blocage de l'expéditeur quand ce dernier à des TPDU à émettre.

3) Interblocage de déséquencement de crédits

- **Problème** : Le déséquencement des crédits peut conduire à des interblocages.
- **Exemple :**
  - Un crédit ayant dépassé l'autre l'émetteur est bloqué.
- **Une solution:**
  - Utiliser des numéros de séquence pour les augmentations de crédits.
  - Délivrer les augmentations dans l'ordre d'émission.



Problèmes et solutions au niveau transport

Conclusion

Conclusion: Conception des protocoles de transport

- **Problèmes de transport** : finalement assez compliqués.
- **On dispose de solutions correctes** en mode point à point pour les données informatiques classiques.
- **Différents problèmes restent posés** :
  - **Adaptation des solutions existantes** à la montée en débits (gigabits, térabits, …)
  - **Communication de niveau transport en diffusion** (multipoint).
  - **Communications de transport avec qualité de service temporelle** (multimédia).

Niveau Transport
"Transport Layer"

Chapitre II

Exemple des protocoles et services de transport INTERNET

UDP "User Datagram Protocol"
TCP "Transmission Control Protocol"
Un service pour TCP et UDP: les sockets

Exemple des protocoles et services de transport INTERNET

UDP

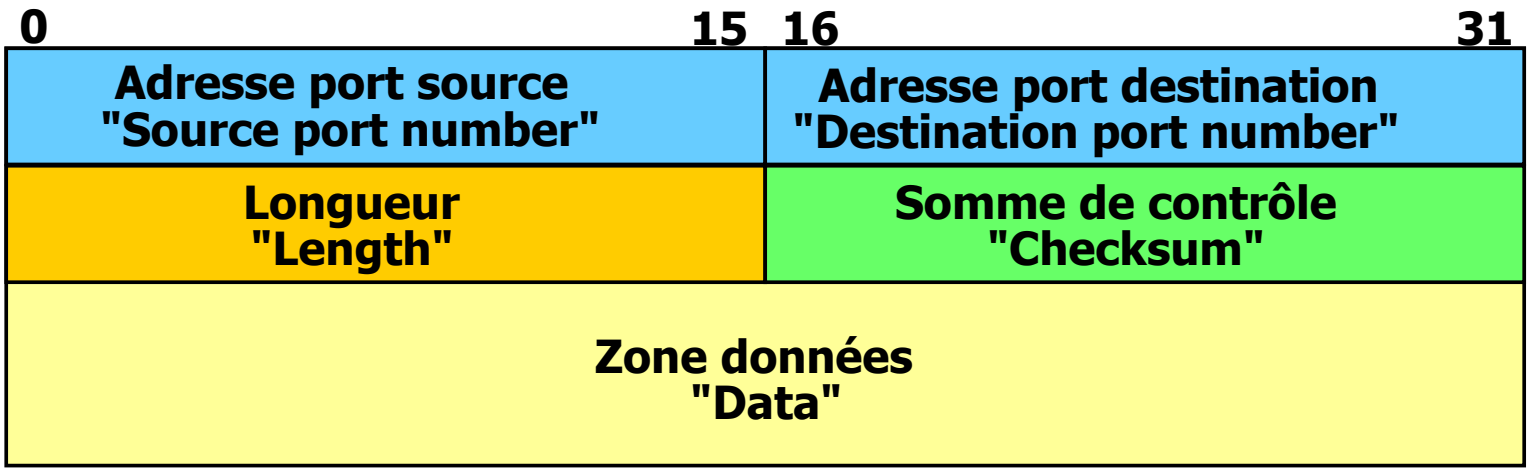
"User Datagram Protocol"

Choix de conception UDP

- **UDP permet l'émission de datagrammes IP en utilisant un protocole très simplifié de niveau transport.**
- **Sans connexion**
  - Pour des relations courtes.
  - **Efficace** (en termes de performances).
  - Simplification des développements de codes applicatifs réseaux.
- **Avec adressage de niveau transport**
  - Identification des **"ports"**

- **UDP implante donc très peu de mécanismes**
  - Adresses de ports (points d'accès de service de transport)
  - Somme de contrôle d'entête (optionnelle)
  - Aucun contrôle de séquence, contrôle d'erreur, ni contrôle de flux.

Format du segment UDP



- Structure très simplifiée (génération et analyse rapide)
- Peu encombrante (8 octets)
- Dédiée essentiellement au problème d'adressage de niveau transport.

Détails concernant les différents champs

- **Zone source port et destination port**
  - Numéros de port identifiant l'utilisateur source et l'utilisateur destinataire (16 bits).
  - Le numéro de port source est optionnel (si non renseigné il est mis à zéro).
- **Zone longueur**
  - Longueur totale en octets du message.
  - Redondant avec la longueur de paquet IP.
  - => La longueur est optionnelle (dans ce cas on la met à zéro).
- **Zone somme de contrôle**
  - Le champ "checksum" couvre la partie entête et la partie données.
  - Il est calculé comme pour l'entête IP par un ou exclusif sur des groupes de 16 bits et complémenté à un.
  - Si la longueur des données est impaire un remplissage par des zéros est effectué sur le dernier octet.
  - => La somme de contrôle est optionnelle (dans ce cas on la met à zéro).

Exemple des protocoles et services de transport INTERNET

TCP

"Transmission Control Protocol"

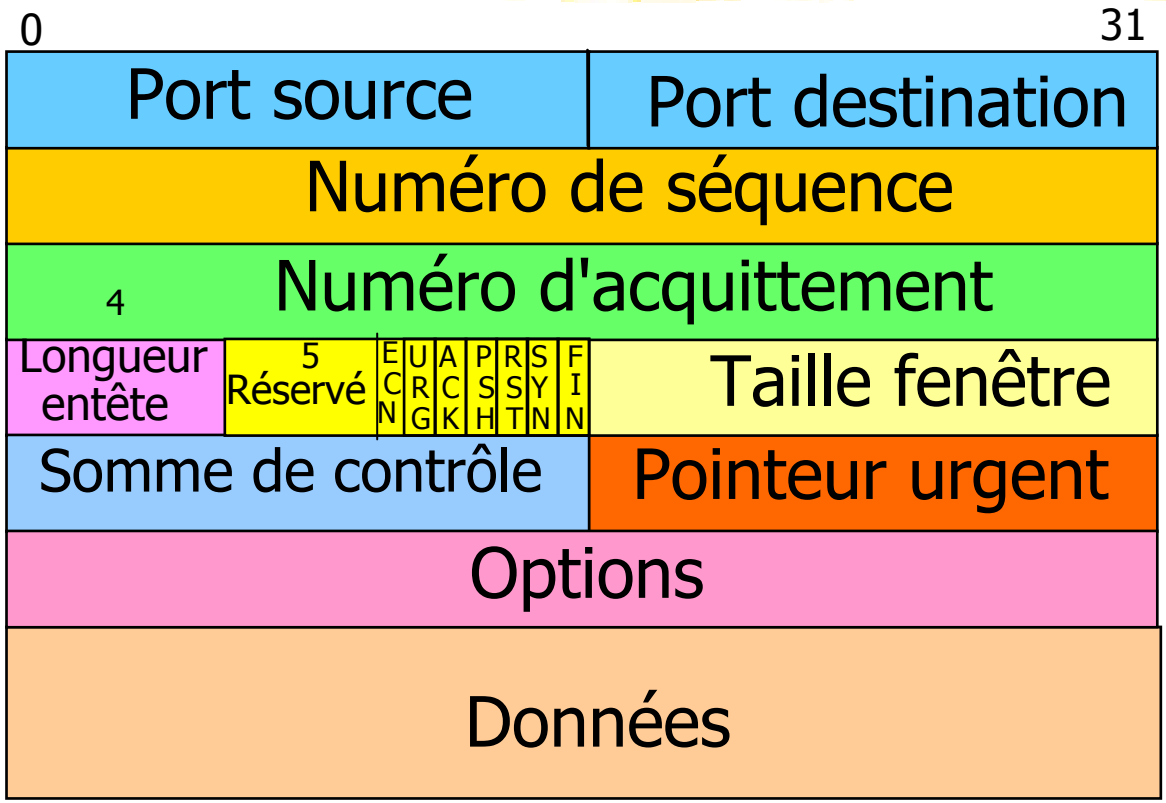
Choix de conception TCP

- **TCP a été conçu en fonction de IP qui est de qualité de service médiocre.**
  - Le réseau sous-jacent (IP) peut **perdre, altérer, dupliquer**, **déséquencer** les paquets.

- **TCP permet un transport de données orienté octets**
  - En mode **connecté, bidirectionnel.**
  - Avec **contrôle d'erreur** (transport fiable).
  - Avec **contrôle de séquence.**
  - Avec **contrôle de flux.**
  - Avec **contrôle de congestion**

- **Existence de trois phases principales**
  - **Ouverture** de connexion
  - **Transfert** de données
  - **Fermeture** de connexion

Format du segment TCP



- Un seul format de TPDU pour tous les mécanismes du protocole.
- Un en-tête relativement long : au minimum 20 octets

Détails concernant les différents champs (1)

- **Numéro de port source ("Source port")**
  - Entier sur 16 bits: identifie le port émetteur.
- **Numéro de port destination ("Destination port")**
  - Entier sur 16 bits: identifie le port récepteur.
- **Numéro de séquence ("Sequence Number") sur 32 bits**
  - Si le bit SYN est non positionné c'est le numéro de séquence du premier octet de données dans la TPDU
  - Si le bit SYN est positionné c'est le numéro de séquence initial (ISN): le premier octet de données a pour numéro ISN+1.
- **Numéro d'acquittement sur 32 bits.**
  - Numéro de séquence de l'octet que l'entité s'attend à recevoir.
- **Longueur de l'en-tête sur 4 bits**
  - En nombre de mots de 32 bits.
  - Nécessaire puisque l'en-tête comporte des options de longueur variable.
- **Zone réservée de 5 bits.**

Détails concernant les différents champs (2)

- **Sept drapeaux (1 bit) déterminent la présence de certains champs** dans le segment TCP et en fait donnent un type (ou plusieurs types) au segment: donnée, acquit positif, établissement ou libération de connexion…

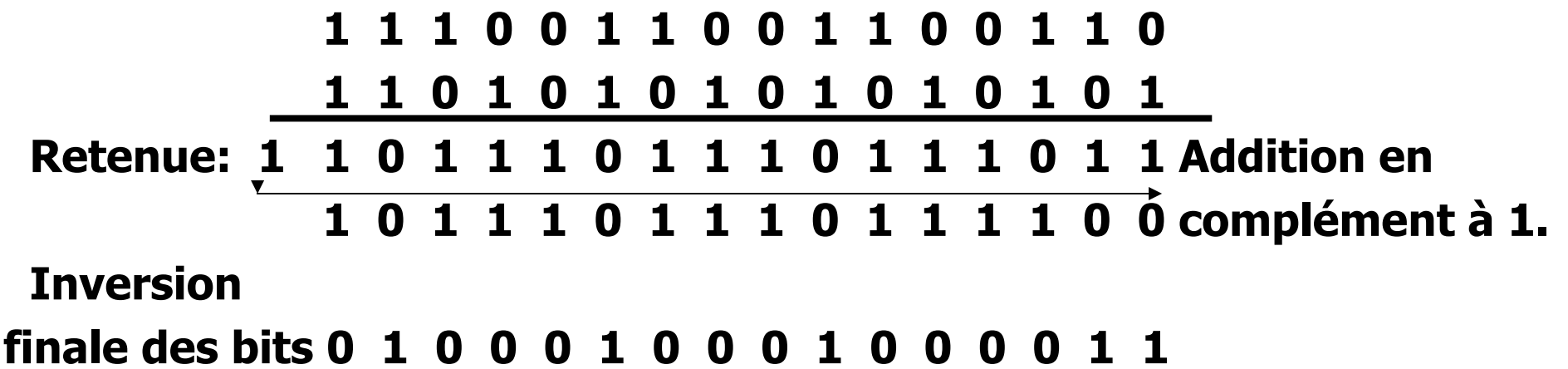
- **URG** ("Urgent") : à 1 si la zone donnée urgente est présente (le pointeur urgent sur la fin de la zone de donnée urgente doit être positionné).
- **ACK** ("Acknowledgment") : à 1 si le champ "numéro d'acquittement" est présent. Il sert aussi à l'acquittement du numéro de séquence initial.
- **PSH** (Push) : une option de service qui demande la transmission immédiate des données en instance dans un segment porteur du bit push. A l'arrivée le destinataire doit délivrer également immédiatement.
- **RST** ("Reset") : l'émetteur ferme abruptement la connexion.
- **SYN** ("Synchronize") : synchronise les numéros de séquence (utilisation principale à l'établissement des connexions).
- **FIN** ("Fin") : l'émetteur n'a plus rien à transmettre et libère la connexion.
- **ECN** ("Explicit Congestion Notification") : notification de congestion.

Détails concernant les différents champs (3)

- **Taille Fenêtre** ('Window Size') (16 bits)
  - Le crédit (taille de la fenêtre de réception) exprimé en nombre d'octets.
  - Mesuré à partir du numéro d'acquittement inséré dans le segment.
- **Somme de contrôle** ('Checksum') (16 bits)
  - Sert à la détection d'erreurs.
- **Pointeur urgent** ('Urgent pointer') (16 bits)
  - Le pointeur urgent pointe sur la fin des données urgentes placées nécessairement à partir du début du segment.
  - Les données urgentes peuvent être suivies par des données normales.
- **Zone options**
  - **Permet l'échange d'informations protocolaires** en extension de l'entête (taille maximum de la zone option 44 octets).
  - **Un code sur un octet distingue les différentes options** (de l'ordre de 25/30 options définies par des RFC).

Approfondissements 1) : calcul de la somme de contrôle UDP/TCP

- **Checksum :** calcul sur le segment décomposé en groupes de 16 bits : calcul **simple, faisable en logiciel** comme en **IP.**



- **Calcul portant sur le segment et un 'pseudo entête IP'.**
  - Les adresses IP source et destination, la longueur, le type.
  - Pour sécuriser la donnée utilisateur mais aussi les principales informations protocolaires (TCP et une partie de l'entête IP).
  - Violation de l'indépendance entre couches.

Approfondissements 2) : Différences Push et Urg

- **Mode push** dans une primitive de service
  - **Force l'émission immédiate** du segment porteur des données en mode push (le segment comporte le bit push).
  - **Les données circulent normalement.**
  - **A destination:** le segment doit être délivré immédiatement.
  - **Exemple:**
    - Terminal en écho local on peut mettre le bit push sur un segment lorsqu'on a atteint le retour chariot.
    - Terminal en écho distant : on doit faire push sur chaque caractère.
- **Mode données urgentes.**
  - **Un bloc urgent** est défini à l'intérieur du segment (utilisation du bit URG et du champ pointeur urgent).
  - **Les données devraient circuler le plus rapidement possible** : mécanismes de QOS ou de priorité IP.
  - Le destinataire devrait recevoir les données dès l'arrivée.

Approfondissements 3) : Exemples de zones options (1)

- **Option 'End of option list'** (code=0) : fin de la liste des options TCP.
- **Option 'No operation'** (code=1) : pour remplissage et alignement sur des frontières de mots de 32 bits.
- **Option MSS ('Maximum Segment Size')** sur 4 octets: Taille maximum des segments (code=2).
  - Annonce de la taille max en octets que le TCP est capable de traiter (dans un segment d'ouverture avec bit syn=1).
  - Absence de MSS : TCP doit toujours accepter 512 octets de données soit un MSS par défaut de 512 + 20 + 4 = 536.
- **Option 'Window Scale'** 10 octets : facteur d'échelle fenêtre (code = 3)
  - Dans les réseaux haut débit la valeur du crédit (taille de la fenêtre sur 16 bits 64K) est trop petite.
  - Sur un octet cette option définit à l'ouverture de connexion un facteur multiplicatif à appliquer à la zone fenêtre.
  - La valeur utilisée est 2^n ou n est le facteur d'échelle.
  - Soit en fait un décalage de n bits sur la valeur de la taille de fenêtre.

Exemples de zones options (2)

- **Option 'TCP SACK permitted'** 2 octets : (code = 4)
  - TCP peut fonctionner en mode rejet sélectif.
  - Dans le cas ou des accusés de réception sélectifs sont autorisés. Cette option met en place le mécanisme.
- **Option 'Timestamp'** 10 octets : datation (code = 8)
  - **Problème** de la mesure du temps d'aller retour (RTT Round Trip Time): les acquittements groupés sont retardés.
  - **Fonction principale de l'option**: faire écho d'une date sur 32 bits pour qu'un émetteur mesure correctement le RTT.
    - Deux estampilles : **TSVAL** la date de l'émission du présent segment ,
    - **TSECR** ('echo reply') la date qui était portée par le segment de donnée dont ce segment est un acquittement positif (bit ACK=1).

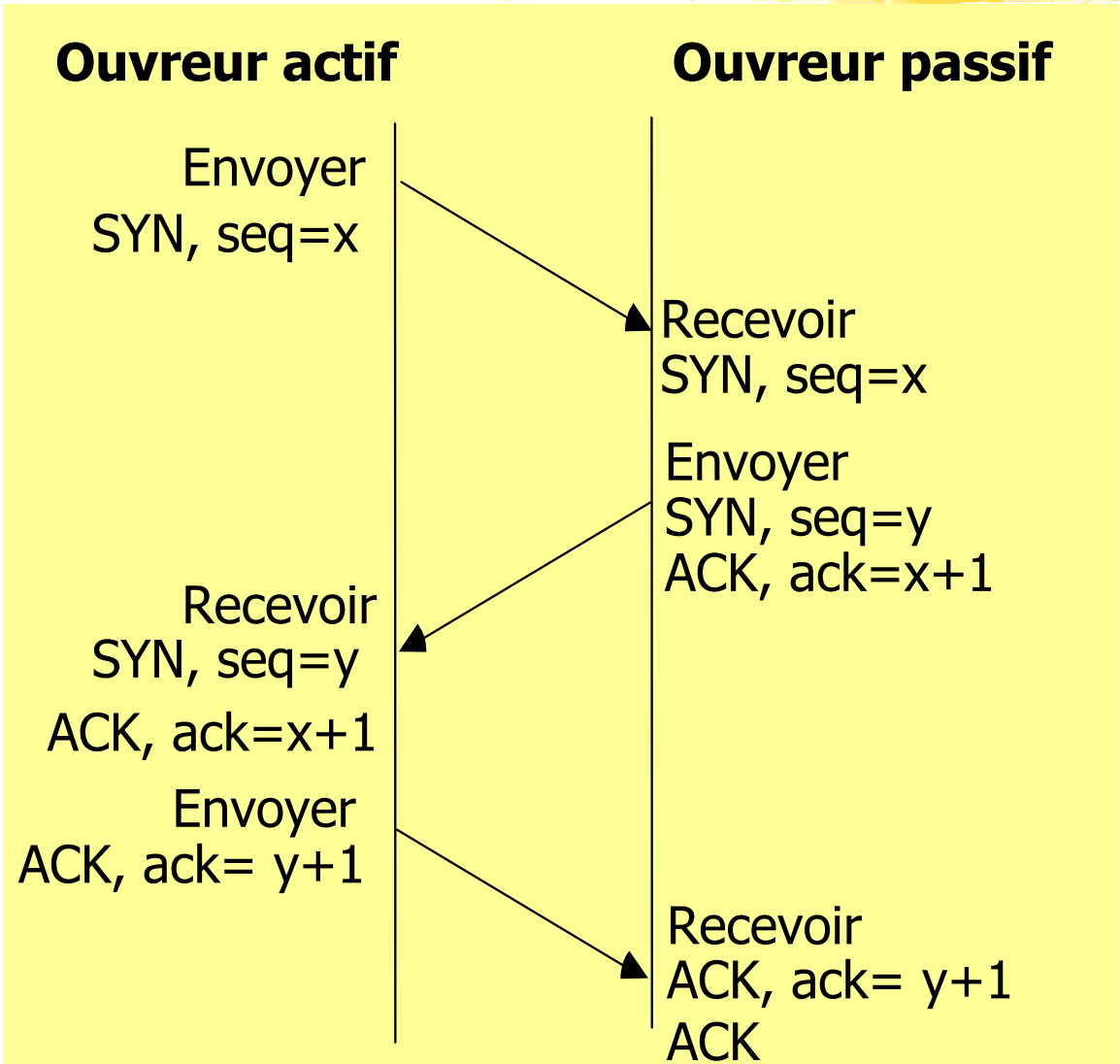
Protocole d'ouverture de connexion en TCP : Principes généraux (1)

- **Ouverture de connexion en trois messages** : "three way handshake" (pour des raisons de fiabilité).
- **Chaque extrémité doit choisir un numéro de séquence** initial ISN.
  - **Ce numéro sur 32 bits est envoyé à l'autre extrémité.**
  - **Chaque numéro de séquence** initial choisi par l'un des communicants doit être **acquitté par l'autre**.
  - **Le couple des numéros de séquence sert de référence initiale** à la connexion.
  - **La norme recommande de choisir les numéros de séquence "au hasard" selon les bits de faible poids d'une horloge** et de ne pas réutiliser ces numéros avant un délai (gel de référence, purge des "vieux paquets" retardés dans le réseau et qui pourraient être mal interprétés).

Protocole d'ouverture de connexion en TCP : Principes généraux (2)

- **Une seule connexion TCP** est établie entre deux ports.
  - TCP ignore les **requêtes** de connexion **ultérieures.**
- **Ouverture de connexion normale** : on distingue le demandeur initial de la connexion (**ouvreur actif**) et l'accepteur (**ouvreur passif**).
- **Cas possible d'ouverture simultanée** : on a deux initiateurs simultanément actifs et l'on doit traiter ce cas particulier.

Diagramme de messages d'ouverture de connexion TCP



Message 1

- Demande de connexion avec SYN=1, un numéro de séquence initial X, et ACK=0 indiquant que le champ d'acquittement n'est pas utilisé.

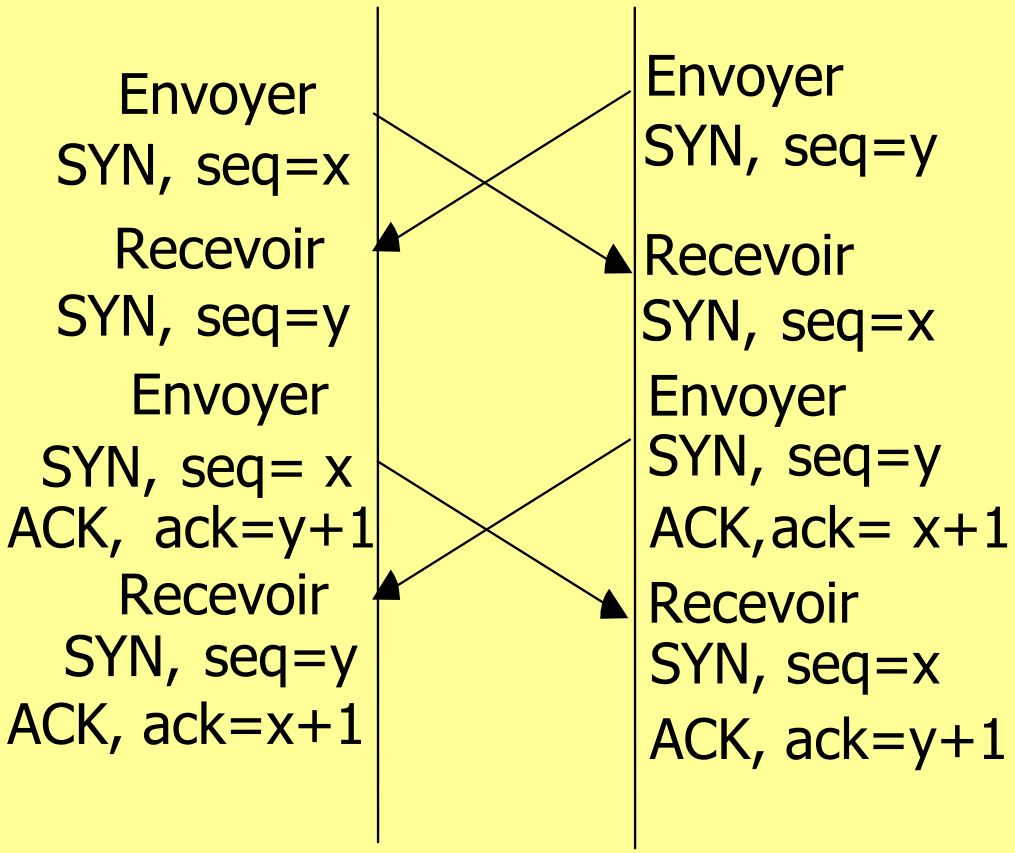
Message 2

- Confirmation de connexion avec SYN=1, un numéro de séquence initial Y et ACK=1 avec le numéro X+1 acquittant le numéro proposé X.

Message 3

- Acquittement du numéro Y proposé avec ACK=1 et un numéro d'acquittement Y+1.

Ouverture de connexion simultanées

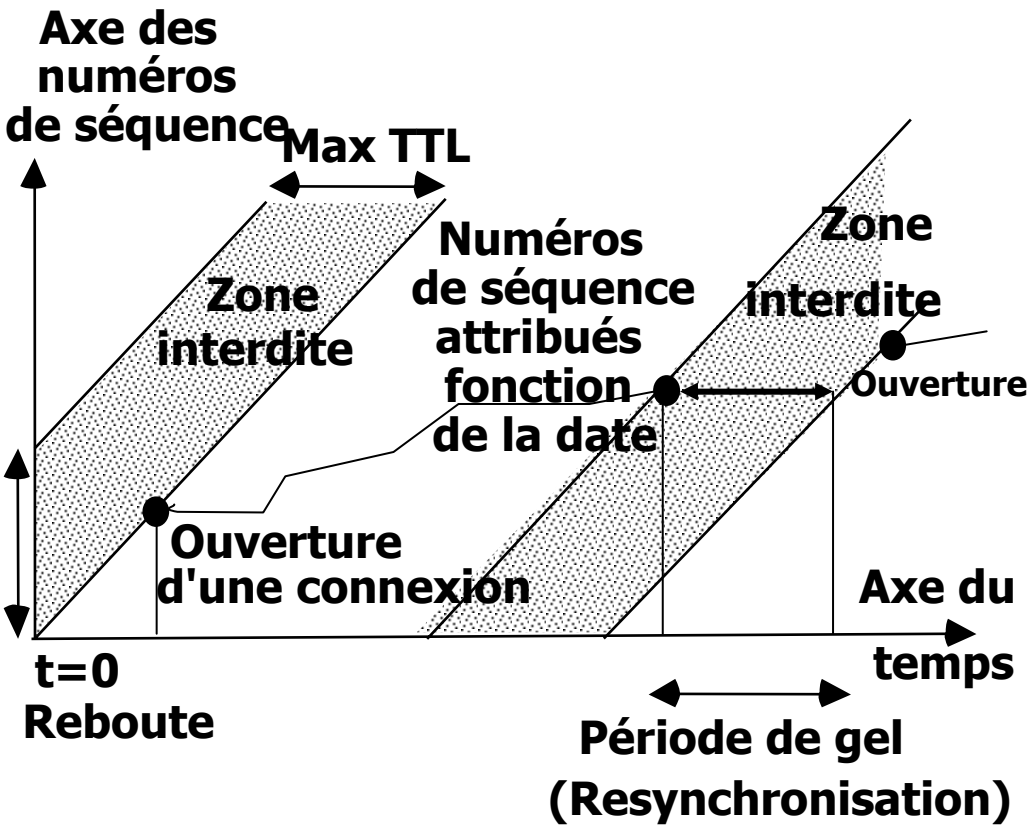


- **Collision d'appel** : deux messages SYN concurrents.
- La norme indique **qu'une seule connexion** doit-être ouverte.
- C'est le cas ici : les deux sites ont négociés en **quatre messages une connexion de référence unique (x,y).**

Approfondissement: Choix des numéros de séquence

- **Des connexions successives doivent recevoir des numéros de séquence initiaux différents et "aléatoires".**
  - Pour différencier les messages ayant circulé sur les deux connexions
- **Pour des numéros de séquence sur 32 bits** le RFC 793 TCP suggère d'utiliser une **horloge sur 32 bits**.
  - **Horloge TCP:** un compteur incrémenté toutes les 4 micro-secondes ce qui le fait recycler environ en 4 heures ("wrap around").
  - **L'incrémentation n'est pas faite un par un** (trop d'interruptions) : mais selon une période qui peut être assez longue (exemple 1/2 seconde => incrémentation de 128000).
  - **Les implantations diffèrent de façon significative** sur le choix de ces valeurs.

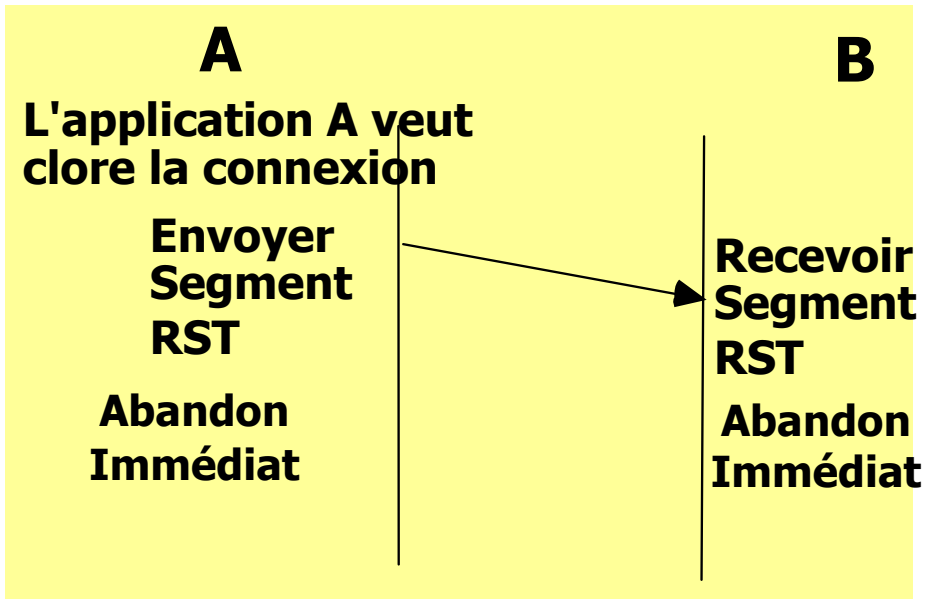
Choix des numéros de séquence : Règles de choix



- **Deux numéros de séquence identiques** correspondant à des messages **différents de connexions différentes** successives entre les mêmes ports ne doivent pas être attribués.
- **Le TTL ("Time To Live")** d'un paquet dans le réseau de communication permet de régler le délai de la zone interdite (durée du gel)..
- **Les numéros de séquence dans la zone interdite ne peuvent être attribués** car des paquets ayant ces numéros peuvent être en transit.

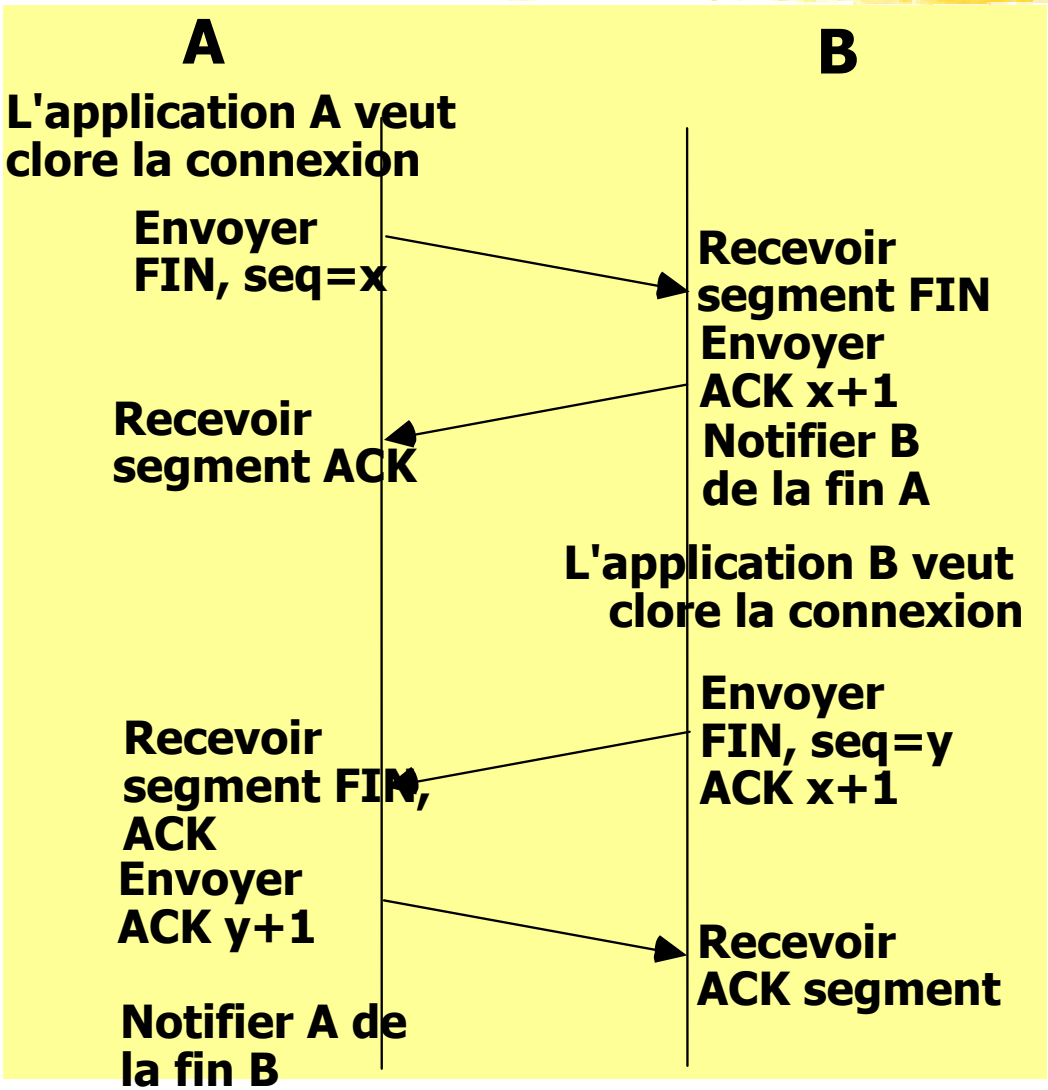
1) Protocole de fermeture de connexion: Libération abrupte

- **TCP Connection reset**
  - **Lorsqu'un segment est transmis avec le bit RST** : une fermeture inconditionnelle (immédiate) de la connexion est réalisée.



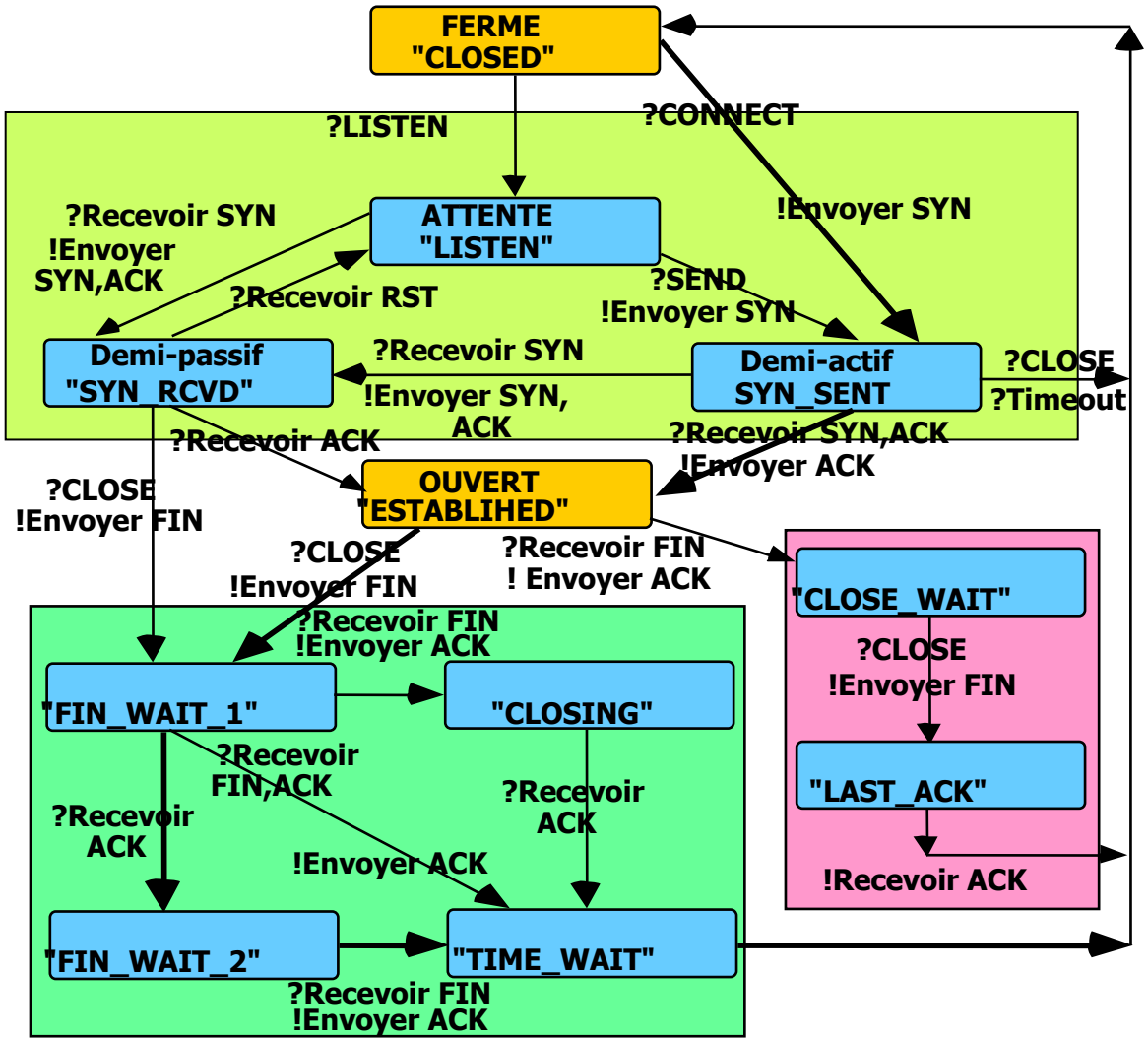
- **Le récepteur abandonne la connexion**.
- **Il notifie à l'application** la fin immédiate de la connexion
- **L'émetteur, le récepteur libèrent tout l'espace de travail** occupé par les segments en cours de transmission.

2) Protocole de fermeture de connexion: Libération négociée



- **TCP Graceful close**
  - **Chaque utilisateur doit générer** sa propre requête de déconnexion.
  - **On garantit la livraison correcte** de toutes les données échangées sur la connexion.

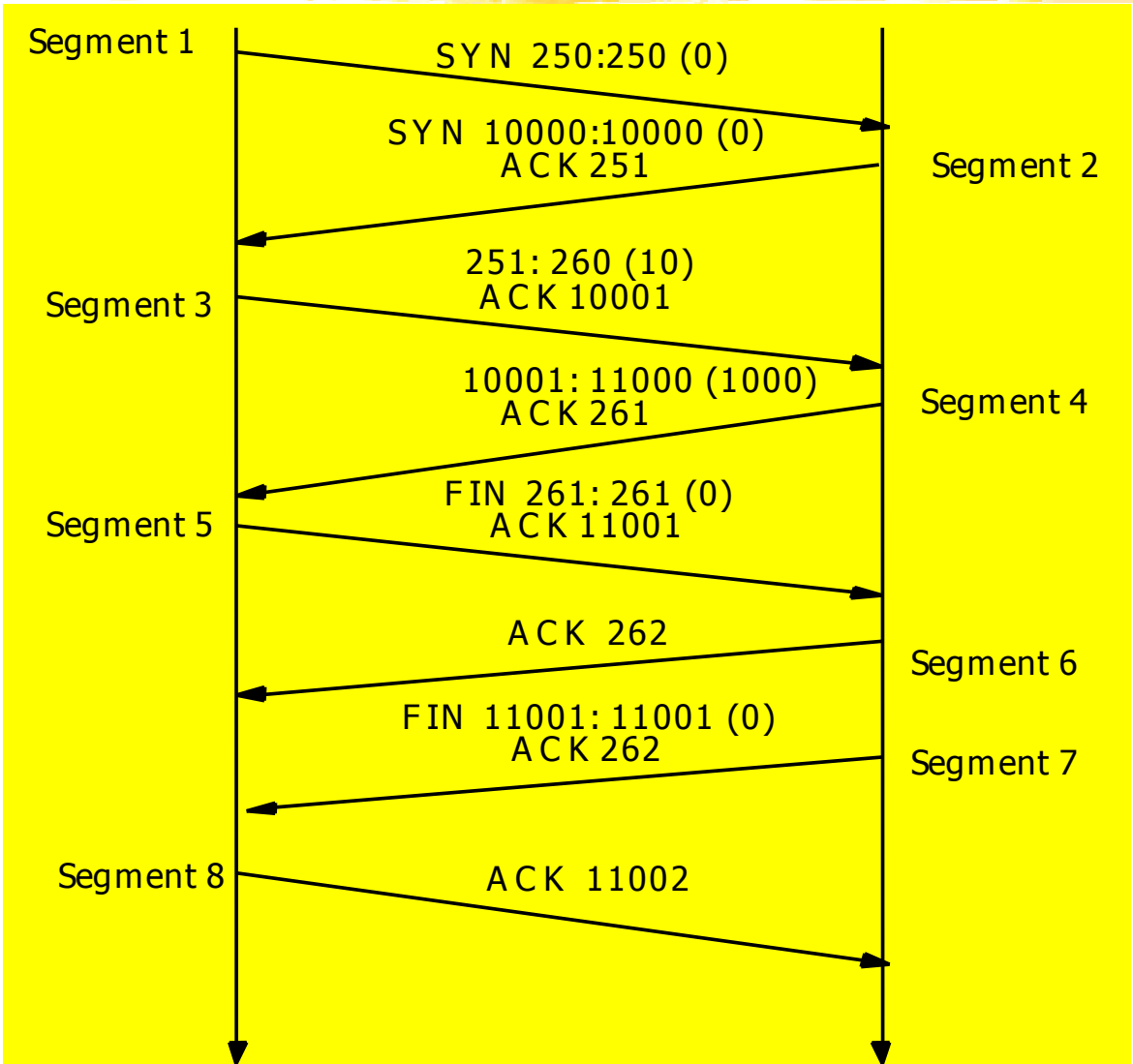
TCP : Automate de connexion/déconnexion



Transfert des données en TCP

- **Structuration des échanges par octets ("stream").**
  - Les blocs d'octets consécutifs à transmettre sont placés dans des segments TCP.
  - TCP décide de l'envoi d'un segment indépendamment des primitives d'émission en effectuant **du groupage** ou de la **segmentation** pour envoyer un segment de taille optimale compte tenu de la gestion des fenêtres de contrôles de flux et de congestion.
- **Contrôle de séquence**
  - Chaque octet est numéroté sur 32 bits.
  - Chaque segment porte le numéro de séquence du premier octet de données.
- **Contrôle de flux**
  - Basé sur le crédit variable (taille de la fenêtre 'window size')
  - Crédit s'exprimant en nombre d'octets.
- **Contrôle d'erreur**
  - Le contrôle d'erreur utilise une stratégie **d'acquittement positif** avec **temporisateur** et **retransmission**.
  - **Son originalité** réside dans l'algorithme de calcul de la valeur du temporisateur de retransmission : la valeur du temporisateur armé lors d'une émission dépend du délai d'aller-retour mesuré pendant une période récente.

Exemple d'échange de segments connexion/deconnexion en TCP

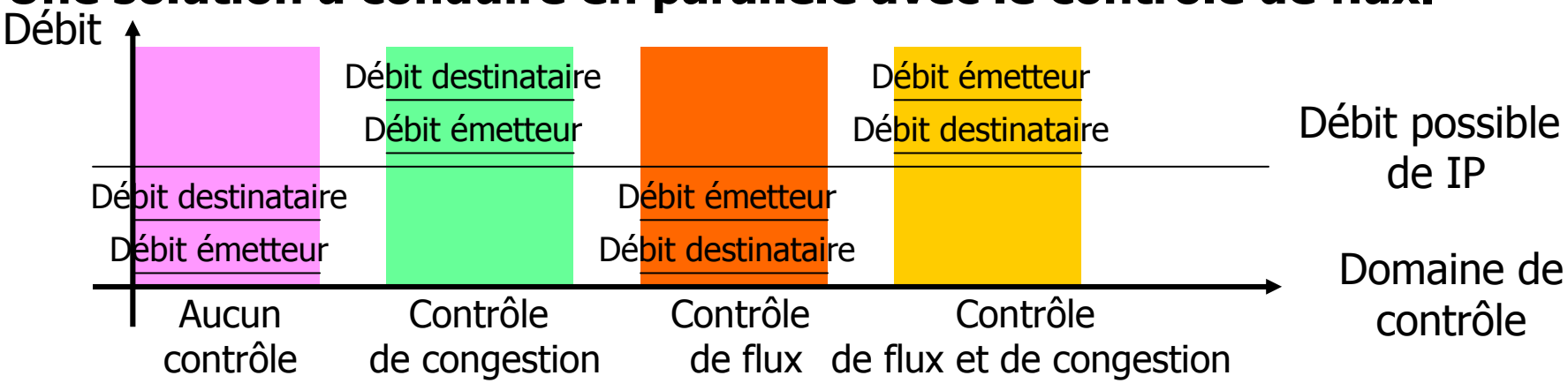


TCP "Transmission Control Protocol"

Approfondissements TCP

TCP: Approfondissements
1) Contrôle de congestion

- **Contrôle de congestion réalisé par TCP** : mais qui s'applique en fait au niveau réseau IP (c'est IP qui devrait faire ce travail).
- **Congestion TCP :** une solution basée sur un contrôle d'admission dans IP
  - TCP ne doit pas soumettre un trafic supérieur à celui de IP.
  - S'il y a des pertes de segments TCP suppose qu'elles sont dues à de la congestion (pas au bruit) => TCP ralentit son débit.
- **Une solution à conduire en parallèle avec le contrôle de flux.**

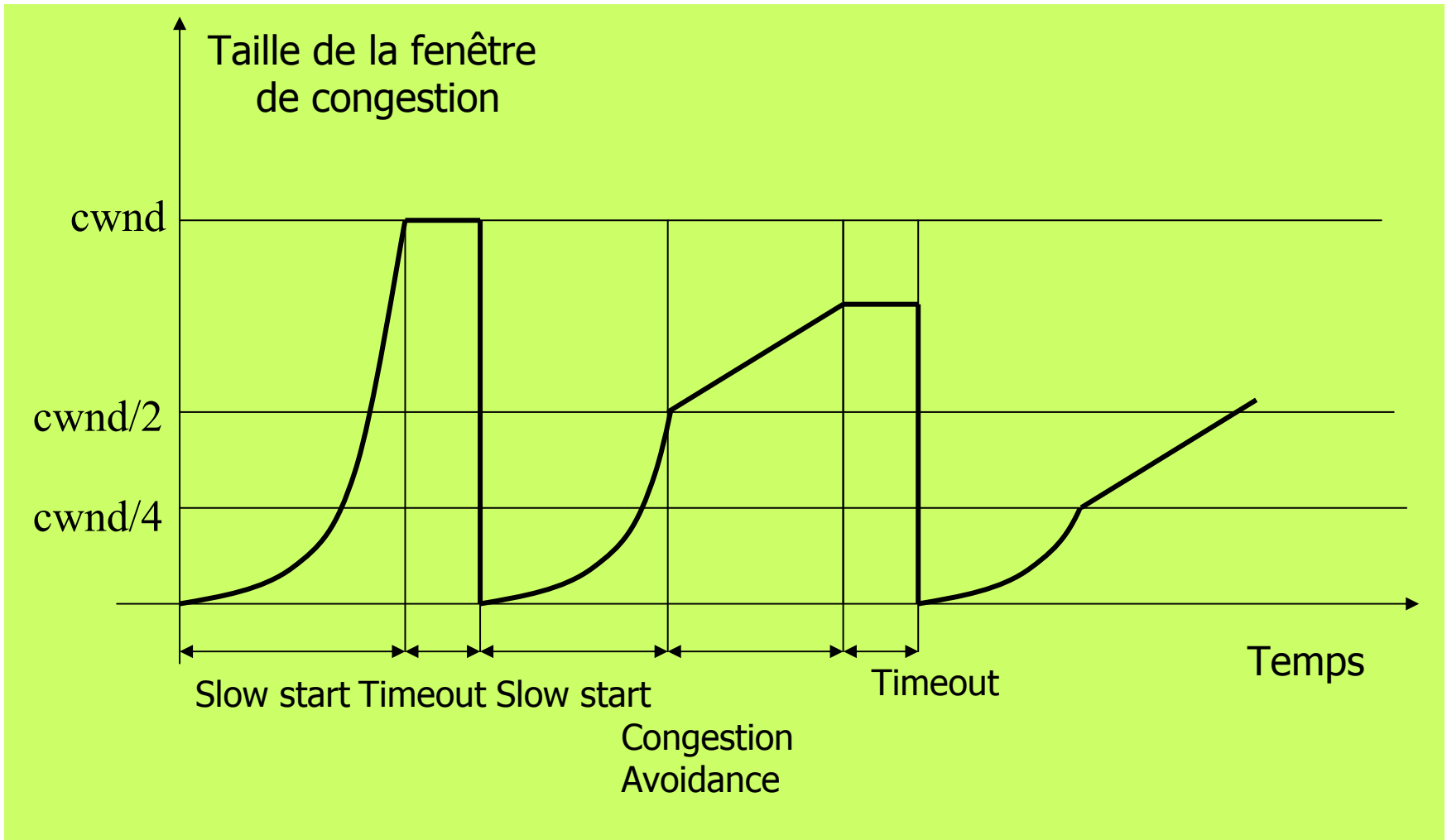


- **=> Une fenêtre pour le contrôle de flux.**
- **=> Une fenêtre pour le contrôle de congestion.**
- **=> La fenêtre réellement utilisée : la plus petite des deux fenêtres.**

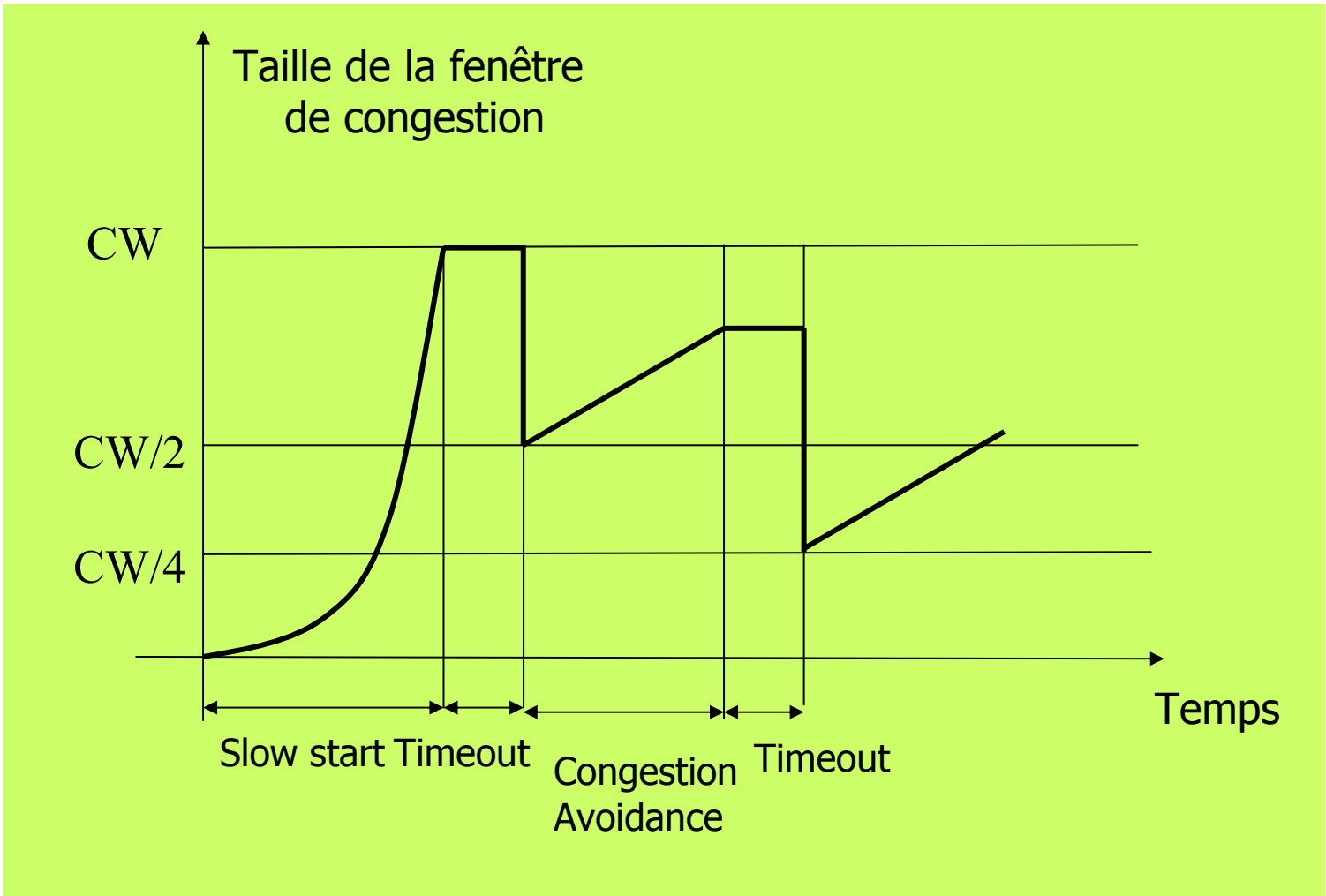
Dimensionnement de la fenêtre de congestion : départ lent 'slow start'

- **Recherche du débit maximum du réseau en TCP Tahoe**
- **A) Démarrage avec croissance exponentielle.**
  - Fenêtre de congestion initiale de taille 1 : un segment de MSS octets.
  - A chaque acquittement positif (transmission réussie): on double la taille de la fenêtre en transmettant 1, 2 , 4, 8, … segments.
  - Jusqu'à un échec de transmission (pas d'acquittement) pour une valeur cwnd: on considère qu'il est du au phénomène de congestion.
  - On repart en démarrage lent jusqu'à la moitié de la valeur soit cwnd/2.
- **B) Recherche fine avec une croissance linéaire.**
  - Phase baptisée évitement de congestion: **'congestion avoidance'.**
  - Lorsque la fenêtre atteint la moitié de sa taille: reprise de la croissance en augmentant par incréments de 1.
- **Optimisation :** ne pas refaire la phase de slow start (TCP Reno).

Evolution de la fenêtre de congestion ('slow start'):TCP Tahoe



Evolution de la fenêtre de congestion ('slow start'): TCP Reno



2) Retransmission rapide Fast Retransmit / Fast recovery

- **Technique des acquittements répétés :** pour signaler un segment hors séquence.
  - Une astuce pour remplacer un acquittement négatif
  - Deux acquittements positifs peuvent apparaître sur simple déséquencement par le réseau.
- **Si on a trois acquittements d'un segment :** indication forte qu'un segment qui suit est perdu.
- **Retransmission du segment sans attendre** le délai de garde => **Solution baptisée 'fast retransmit'.**
- **Ne pas ramener la taille de la fenêtre de congestion à 1** pour un démarrage lent 'slow start' : il y a du trafic à écouler
- **Solution baptisée 'fast recovery'.**
- **Autre solution :** acquittements négatifs dans la zone option (TCP avec SACK).

3) Adaptation du temporisateur aux temps de réponse

- **Notion de RTT 'Round Trip Time' :** mesure du délai d'aller retour (émission d'un segment => réception de son acquittement positif).
- **Notion de RTO 'Retransmission Timeout' :** adaptation des délais de garde en fonction des RTT.
- **Possibilités de différentes formules** donnant RTO en fonction de RTT.
- **La méthode de base :**
  - **RTT mesuré** : de préférence sur chaque segment.
  - **RTT lissé** : sRTT = (1 – g) * sRTT + g * RTT
  - **Valeur de RTO** le temporisateur : RTO = 2 * sRTT
  - **Problème:** ne tient compte que de la moyenne de RTT.

La méthode de Karn/Partridge

- **Utiliser la moyenne et la variance.**
- **RTT courant** : ne pas utiliser les mesures de RTT lors d'une retransmission et doubler le RTO à chaque retransmission car on est probablement en congestion (idée de retard binaire).
- **RTT lissé** : sRTT = (1 – g) * sRTT + g * RTT
- **Ecart type du RTT** 'standard deviation' écart par rapport à la moyenne
  D = (1 – h) * D – h * | RTT – sRTT |
- **Valeurs recommandées** pour les coefficients g et h:
  g = 0.125 : prise en compte sur une moyenne de 8 mesures.
  h = 0.25 : prise en compte sur une moyenne de 4 mesures.
- **Définition de RTO le temporisateur** : RTO = sRTT + 4 D
- **Plus RTT est grand et plus grande est sa variation : plus long est le temporisateur.**
- **Autre méthode : Jacobson/Karels.**

4) Gestion de taille des segments 'Silly Window Syndrome'

- **Situation : traitement lent du récepteur.**
  - **Ré ouverture lente** de petits crédits d'émission.
  - **Sur ces petits crédits** => émission de petits segments.
  - **Syndrome de la fenêtre ridicule ('silly')** => Mauvaise utilisation du réseau car les messages sont de petite taille.
- **Solutions : Le groupage (Algorithme de Naggle)**
  - **Le récepteur ne transmet que des crédits importants.**
    - Une fraction significative de l'espace total des tampons récepteurs.
  - **L'émetteur diffère les émissions**
    - Tant qu'un segment de taille suffisante à émettre n'est pas réuni (définition d'un petit segment : relativement au MSS du destinataire).
    - Ou tant qu'il y a des données non acquittées.

5) Protection contre le passage à zéro des numéros de séquence

- **Passage par zéro ('Wrap Around') :** Twrap =2^{31}/débit binaire
  - Ethernet (10Mbps): Twrap = 1700 s (30 minutes)
  - Ethernet Gigabit (1 Gbps): Twrap = 17 secs.
  - Si la durée de vie dans le réseau (MSL Maximum Segment Life) est possiblement supérieure on a un risque d'interférence entre numéros.
- **PAWS : Protection against wrapped sequence numbers.**
- **Utilisation de l'option 'timestamp'** : estampillage TSVAL 32 bits pour la mesure correcte des délais d'aller retour dans la zone option.
- **A partir d'une horloge à la milliseconde**: repassage par 0 en 24 jours.
- **Solution:** combinaison du numéros de séquence habituel et de l'estampille TSVAL pour obtenir un numéro sur 64 bits : comparaison des valeurs sur 64 bits.
- **Le problème de repassage par zéro ne peut se poser que sur 24 jours:** acceptable en relation avec la durée de vie des segments.

Conclusion TCP

- **Un protocole de transport fiable efficace.**
  - Indispensable dans le domaine des réseaux basse et moyenne vitesse.
  - Des optimisations nombreuses : versions Tahoe, Reno et New-Reno, Sack TCP … toujours étudiées.
- **Problèmes posés**
  - **Passage aux réseaux gigabits en TCP.**
    - Différentes expériences : modifications, dimensionnement pour apporter des réponses à cette question.
  - **Support des applications multimédia**
    - Nécessite la définition de nouveaux protocoles de transports qui respectent les besoins de qualité de service de ces applications (RTP ,….)

Exemple des protocoles et services de transport INTERNET

Un service pour TCP et UDP: les sockets

Généralités interface "socket"

- **Définition en 1982** : une interface de programmation d'applications réseaux (API) pour la version UNIX BSD.
- **Existence de plusieurs autres interfaces réseaux :** TLI, NETBEUI, ...
- **Objectifs généraux**
  - Fournir des moyens de communications entre processus (IPC) **utilisables en toutes circonstances**: échanges locaux ou réseaux.
  - Cacher **les détails d'implantation** des couches de transport aux usagers.
  - Si possible cacher les **différences entre protocoles de transport hétérogènes** sous une même interface (TCP, Novell XNS, OSI)
  - Fournir une interface d'accès qui se rapproche **des accès fichiers pour simplifier la programmation** => En fait des similitudes et des différences importantes entre programmation socket et fichier.

Choix de conception des sockets

- **Une "socket" (prise) est un point d'accès de service pour des couches transport :** essentiellement TCP/UDP mais aussi d'autres protocoles (OSI, DECNET...).
- **Une socket est analogue à un objet** (de communication)
  - **Un type**:
    - Pour quel protocole de transport est-elle un point d'accès de service?
    - Quelle est la sémantique de l'accès au service?
  - **Un nom:** identifiant unique sur chaque site (en fait un entier 16 bits).
  - **Un ensemble de primitives** : un service pour l'accès aux fonctions de transport.
  - **Des données encapsulées :**
    - **un descriptif** (pour sa désignation et sa gestion)
    - **des files d'attente** de messages en entrée et en sortie.

Désignation des sockets

- **Identifier complètement une socket** dans un réseau et pour une couche transport : un couple NSAP,TSAP.

  Exemple Internet TCP: Numéro de port , Adresse IP
- **Gestion par l'IANA**
- **A) Numéros de ports réservés :** numéros de ports réservés pour des services généraux **bien connus** ou **"well-known ports"** (numéros inférieurs à 1023).
  - **Exemples ports UDP** : Echo server: 7, TFTP: 69.
  - **Exemples ports TCP** : Telnet: 23, DNS: 53, HTTP: 80.
- **B) Numéros de ports enregistrés ('registered'):** (entre 1024 et 49151) pour des applications ayant fait une demande.
- **C) Numéros de ports privés ('private') (dynamiques) :** les autres numéros entre 49152 et 65535 qui sont attribués dynamiquement aux sockets utilisateurs (clients).

Choix de conception des sockets avec TCP

- TCP est un transport **fiable** en **connexion** et en mode **bidirectionnel point à point**.
- **Une socket TCP peut être utilisée par plusieurs connexions** TCP simultanément.
- **Une connexion est identifiée par le couple** d'adresses socket des deux extrémités.
- **Un échange TCP est orienté flot d'octets**.
  - Les zones de données qui correspondent à des envois successifs ne sont pas connues à la réception.
  - Pour optimiser TCP peut tamponner les données et les émettre ultérieurement.
  - L'option **"push"** qui permet de demander l'émission immédiate d'un segment.
  - L'option **"urgent"** qui devrait permettre l'échange rapide de données exceptionnelles avec signalement d'arrivée.

Choix de conception des sockets avec UDP

- UDP est une couche transport **non fiable, sans connexion,** en mode **bidirectionnel** et **point à point**.
- **L'adresse UDP d'une socket** (Numéro de port UDP , Adresse IP) sur l'Internet à la même forme que celle d'une socket TCP.
- **Mais les deux ensembles d'adresses sont indépendants** : une communication UDP n'a rien à voir avec une communication TCP.
- Un échange UDP est sans connexion (échange de **datagrammes)**.
- Les zones de données qui correspondent à des envois successifs sont **respectées** à la réception.

Exemple des protocoles et services de transport INTERNET

Les primitives de l'interface socket

Exemple en langage C en UNIX.

Primitive socket

- **Permet la création d'un nouveau point d'accès de service transport:**
  - définition de son type.
  - allocation de l'espace des données.
- **Trois paramètres d'appel**
  - **Famille** d'adresses réseaux utilisées locale, réseau IP, réseau OSI ...
  - **Type** de la socket (du service) sémantique de la communication.
  - **Protocole** de transport utilisé.
- **Un paramètre résultat:** le numéro de descripteur socket.
- **Profil d'appel de la primitive en C**

```
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/in.h>
int socket (int famille, int type, int protocole);
```

Approfondissement des paramètres de la primitive socket

- **Paramètre Famille**
  - AF_UNIX : Communication locale (i-node)
  - AF_INET : Communication Internet
  - AF_ISO : Communication ISO

  ....
- **Paramètre Type**
  - **SOCK_STREAM** : Flot d'octets en mode connecté (ne préserve pas les limites de l'enregistrement)
  - **SOCK_DGRAM** : Datagramme en mode non connecté (préserve les limites de l'enregistrement)
  - **SOCK_RAW** : Accès aux couches basses.
  - **SOCK_SEQPACKET** : Format structuré ordonné (protocoles différents de l'Internet)
- **Paramètre Type de protocole**

Valeur	Relation avec le paramètre type
IPPROTO_TCP	SOCK_STREAM
IPPROTO_UDP	SOCK_DGRAM
IPPROTO_ICMP	SOCK_RAW
IPPROTO_RAW	SOCK_RAW

Primitive bind

- **Primitive pour l'attribution d'une adresse de socket** à un descripteur de socket.
- **N'est pas réalisé** lors de la création du descriptif (socket).
  - Un serveur (qui accepte des connexions) doit définir sur quelle adresse.
  - Un client (qui ouvre des connexions) n'est pas forcé de définir une adresse (qui est alors attribuée automatiquement).
- **Profil d'appel de la primitive**

```
#include <sys/types.h>
#include <sys/socket.h>
int bind (     int s,
               struct sockaddr_in *mon_adresse,
               int longueur_mon_adresse        )
```

- **Trois paramètres d'appel**
  - Numéro du descriptif de Socket (s).
  - Structure de donnée adresse de socket Pour internet type sockaddr_in.
  - Longueur de la structure d'adresse.

Approfondissement concernant la primitive bind

- **Descripteur d'adresse de socket**

```
#include <sys/socket.h>
struct sockaddr_in {
    short           sin_family;
    u_short         sin_port;
    struct in_addr  sin_addr;
    char            sin_zero[8];   };
```

- **Un exemple d'exécution de "bind" pour les protocoles Internet.**

```
struct servent *sp
struct sockaddr_in sin

/* Pour connaître le numéro de port */
if((sp=getservbyname(service,"tcp")==NULL)
/* cas d'erreur */

/* Remplissage de la structure sockaddr */
/* htonl convertit dans le bon ordre */
/* INADDR_ANY adresse IP du site local */
sin.sin_family= AF_INET;
sin.sin_port = sp -> s_port;
sin.sin_addr.s_addr=htonl(INADDR_ANY):

/* Création d'une socket internet */
if ((s=socket(AF_INET,SOCK_STREAM,0))<0)
/* cas d'erreur */

/* Attribution d'une adresse */
if (bind(s, &sin, sizeof(sin)) < 0)
/* cas d'erreur */
```

Primitive listen

- **Utilisé dans le mode connecté lorsque plusieurs clients** sont susceptibles d'établir plusieurs connexions avec un serveur.
- **Indique le nombre d'appel maximum attendu** pour réserver l'espace nécessaire aux descriptifs des connexions.
- **La primitive listen est immédiate** (non bloquante).
- **Profil d'appel :** int listen (int s , int max_connexion)

  s : Référence du descripteur de socket

  max_connexion : Nombre maximum de connexions.

Primitive accept

- **La primitive accept** permet de se bloquer en attente d'une nouvelle demande de connexion (donc en mode connecté TCP).
- **Après accept,** la connexion est complète entre les deux processus.
- **Le site qui émet accept exécute une ouverture passive.**
- **Pour chaque nouvelle connexion entrante** la primitive fournit un pointeur sur un nouveau descriptif de socket qui est du même modèle que le descritif précédemment créé.
- **Profil d'appel**

```
#include <sys/types.h>
#include <sys/socket.h>
int accept (    int ns,
                struct sockaddr_in *addr_cl,
                int lg_addr_cl)

ns              : Référence nouvelle socket
addr_cl         : L'adresse du client.
lg_addr_cl: La longueur de l'adresse.
```

Approfondissement concernant les primitives listen et accept

- **Exemple de code UNIX :** pour un serveur qui accepte des connexions successives et qui créé un processus pour traiter chaque client.

```
#include <sys/socket.h>
/* Adresse socket du client appelant */
struct sockaddr_in from;
quelen = ... ;
if (listen (s, quelen) <0 )
        Cas d'erreur
/* On accepte des appels successifs */
/* Pour chaque appel on créé un processus */
if((g=accept(f,&from,sizeof(from)))<0)
        Cas d'erreur
if ( fork ...
/* Processus traitant de connexion*/
```

Primitive connect

- **La primitive connect** (bloquante) permet à un client de demander l'ouverture (**active**) de connexion à un serveur.
- **L'adresse du serveur doit être fournie.**
- **La partie extrémité locale relative au client** est renseignée automatiquement.
- **Ensuite le client ne fournit plus l'adresse du serveur** pour chaque appel mais le descriptif de la socket (qui contient l'adresses serveur).
- **Profil d'appel**

```
#include <sys/types.h>
#include <sys/socket.h>
int connect (  int s,
               struct sockaddr_in *addr_serv,
               int lg_addr_serv)

s              : La référence de la socket
addr_serv      : L'adresse du serveur.
lg_addr_serv : La longueur de l'adresse.
```

Primitives send, recv

- **Les primitives send, recv (bloquantes)** permettent l'échange effectif des données.
- **Le profil d'appel est identique à celui des primitives read et write sur fichiers** avec un quatrième paramètre pour préciser des options de communications.
- **Profil d'appel**

```
#include <sys/types.h>
#include <sys/socket.h>
int send  (int s, char *zone,
           int lg_zone, int options_com)
int recv  (int s,char *zone,
           int lg_zone, int options_com)
s                          : La référence de la socket
zone                       : La zone à échanger.
lg_zone          : La longueur de la zone.
options_com      : Les options (données urgentes , ....)
```

Primitives sendto, recvfrom

- **Les primitives sendto, recvfrom** permettent l'échange des données dans le mode non connecté UDP.
- **On doit préciser l'adresse destinataire** dans toutes les primitives **sendto** et **l'adresse émetteur** dans les **recvfrom**.
- **Profil d'appel**

```
#include <sys/types.h>
#include <sys/socket.h>
int sendto (int s,
            char *zone,
            int lg_zone,
            int options_com,
            struct sockaddr_in *addr_dest,
            int lg_addr)
int recvfrom (int s,
            char *zone,
            int lg_zone,
            int options_com,
            struct sockaddr_in *addr_emet,
            int *lg_addr)
addr_dest  : L'adresse du destinataire.
addr_emet : L'adresse de l'émetteur.
lg_addr    : La longueur de l'adresse.
```

Primitives shutdown, close

- **Shutdown** permet la terminaison des échanges sur une socket suivi de la fermeture de la connexion :
  - **Profil d'appel** : int shutdown(s , h); Pour la socket s.
  - **h = 0** : l'usager ne veut plus recevoir de données
  - **h = 1** : l'usager ne veut plus envoyer de données
  - **h = 2** : l'usager ne veut plus ni recevoir, ni envoyer.
- **Close :** Permet la fermeture d'une connexion et la destruction du descriptif.
  - Profil d'appel

```
#include <sys/types.h>
#include <sys/socket.h>
int close ( int s )
```

Résumé : Interface socket

- **Fonctionnement en TCP**
  - Serveur.
    socket
    bind
    listen
    accept
    recv, send
    close
  - Client.
    socket
    connect
    recv, send
    close
- **Fonctionnement en UDP**
    socket
    recvfrom, sendto
    close

Bibliographie
Niveau transport

- A.S. Tannenbaum "**Computer Networks**" Prentice Hall
- W.R. Stevens "**TCIP/IP Illustrated, The protocols**" , Addison Wesley
- Internet Web : multiples cours TCP.