



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

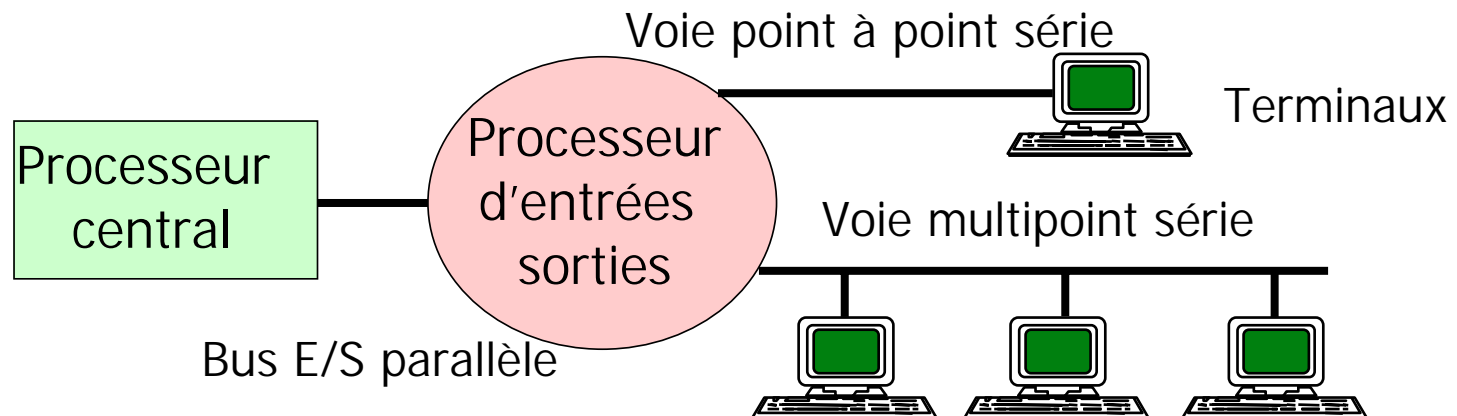
Introduction Notions générales



Origine des réseaux - Evolution

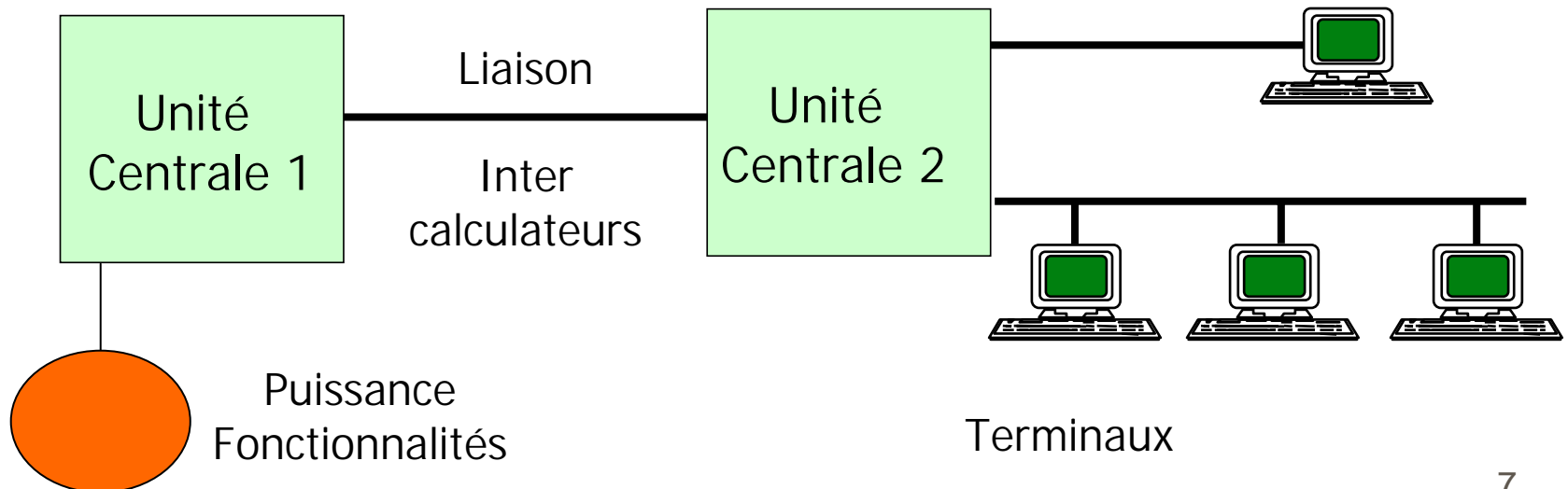
Télétraitement

- 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

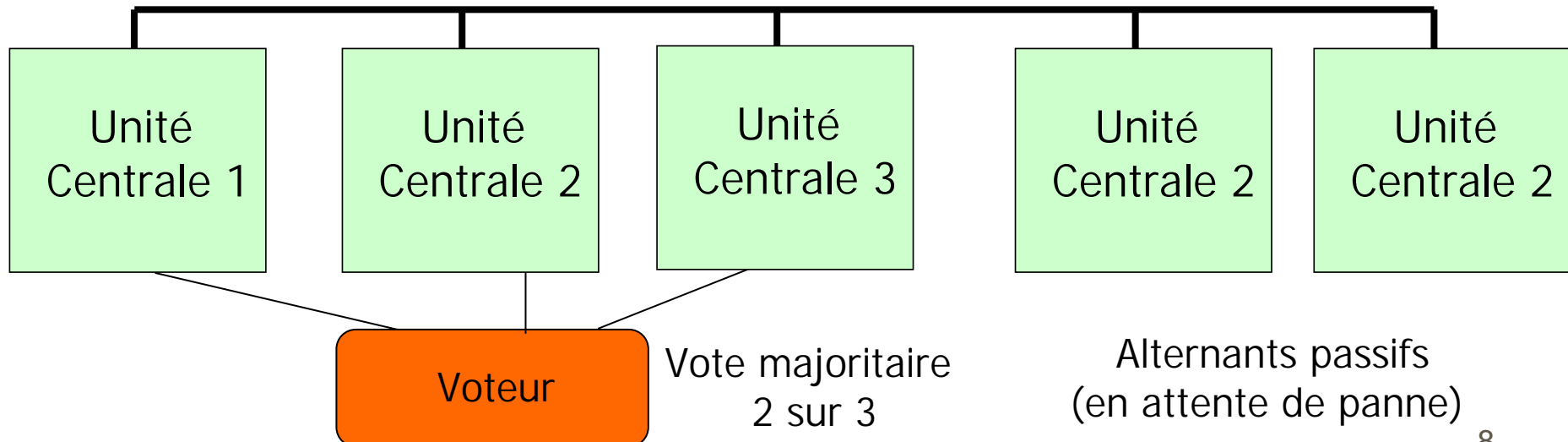
- A) Optimisation des ressources (partage)
- B) Donner accès à des ressources rares/chères.
 - Exemples 1 : puissance de calcul, logiciels spéciaux ...
 - Exemples 2 : périphériques spéciaux rapides, archivage ...



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.

Liaisons Inter calculateurs: synchronisation



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** :

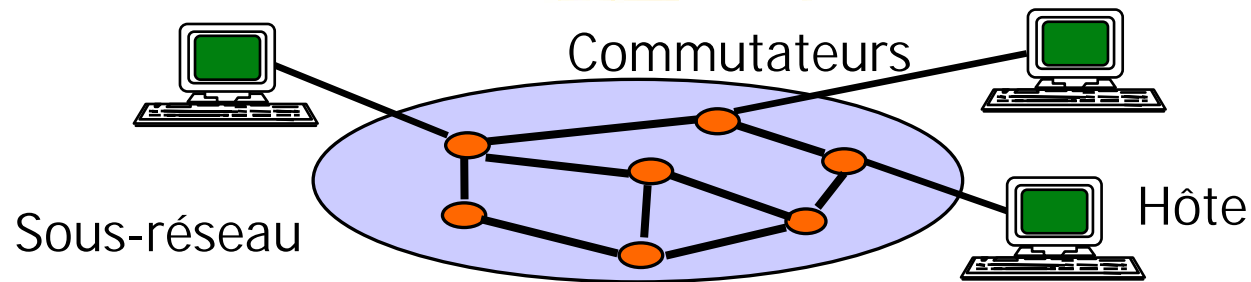
- IBM SNA ("System Network Architecture" 1974)

■ **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



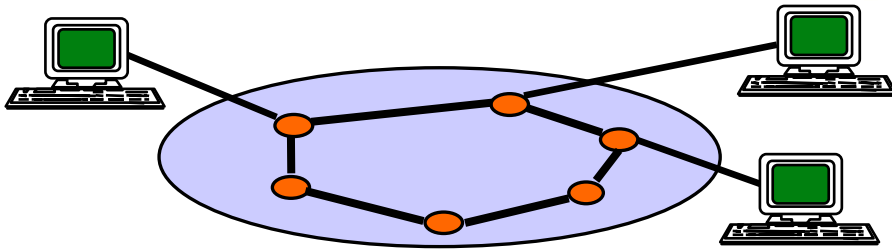
■ Ordinateurs Hôtes ("Hosts Computers")

- Les systèmes informatiques interconnectés.

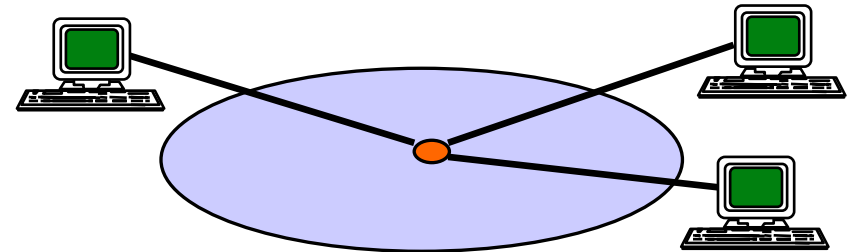
■ Sous-réseau de communication ("Communication Subnet") : Le moyen de communication des hôtes.

- **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.

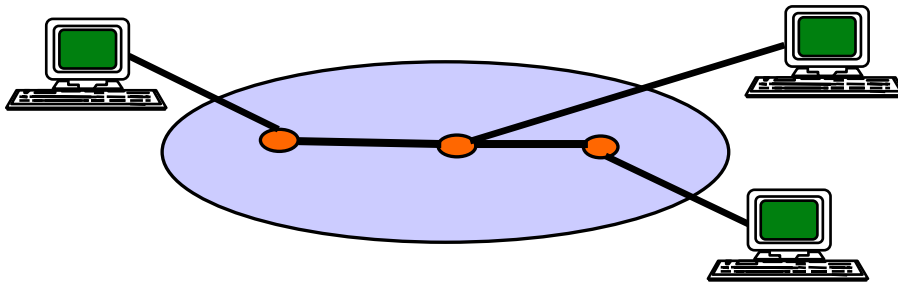
Topologies



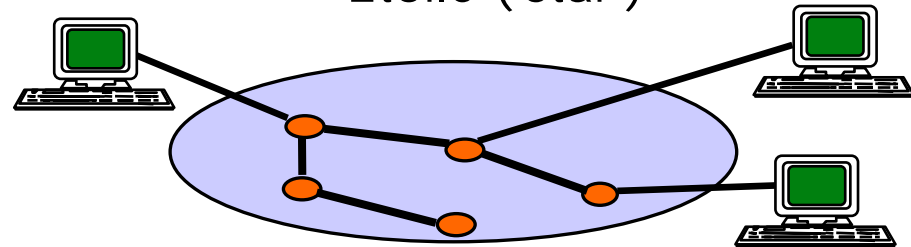
Boucle ou anneau ('ring')



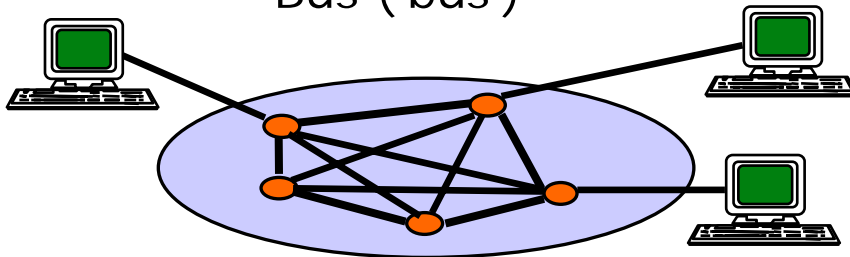
Etoile ('star')



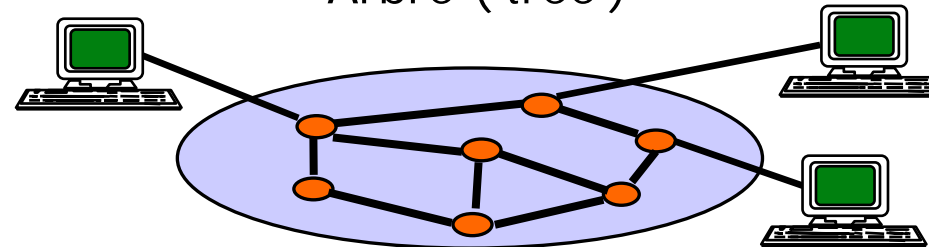
Bus ('bus')



Arbre ('tree')



Complète ou clique ('fully connected')



Maillée ('meshed')¹²

Réseaux généraux et communications inter-personnelles

- **Convergence de deux tendances** dans les années 1970: informatique et télécommunications => **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).

Systemes 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"):

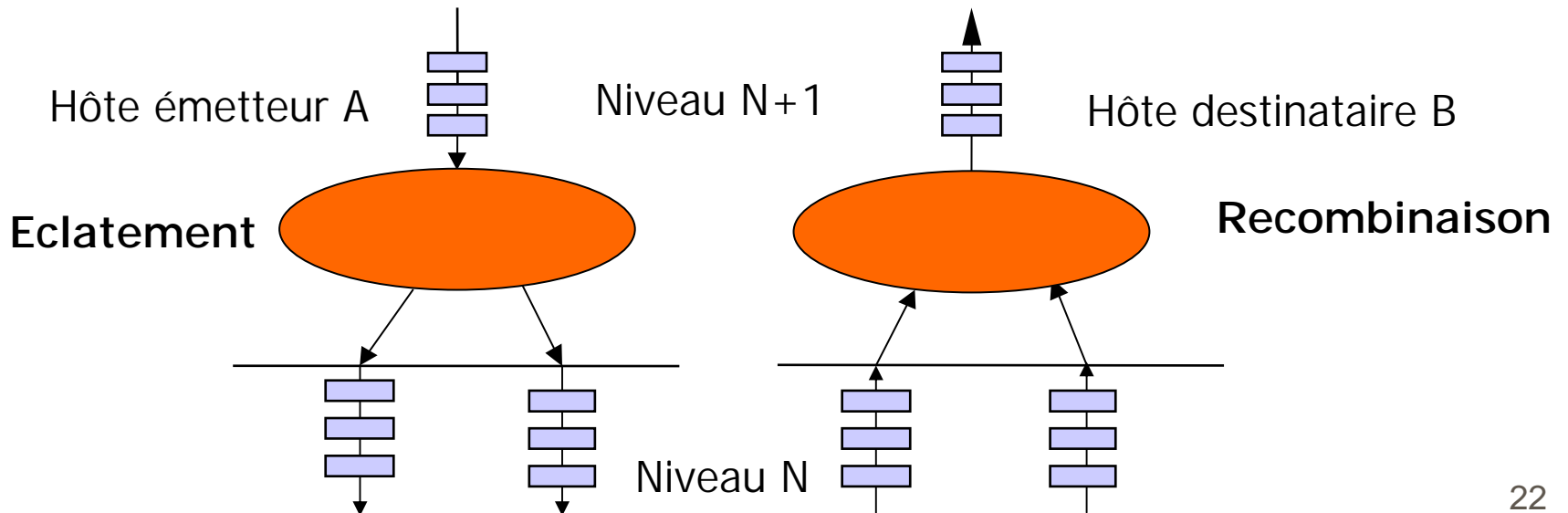
Echange des données en cours avant fermeture

■ **Déconnexion négociée** ("Negotiated release")

Fermeture avec l'accord des deux parties.

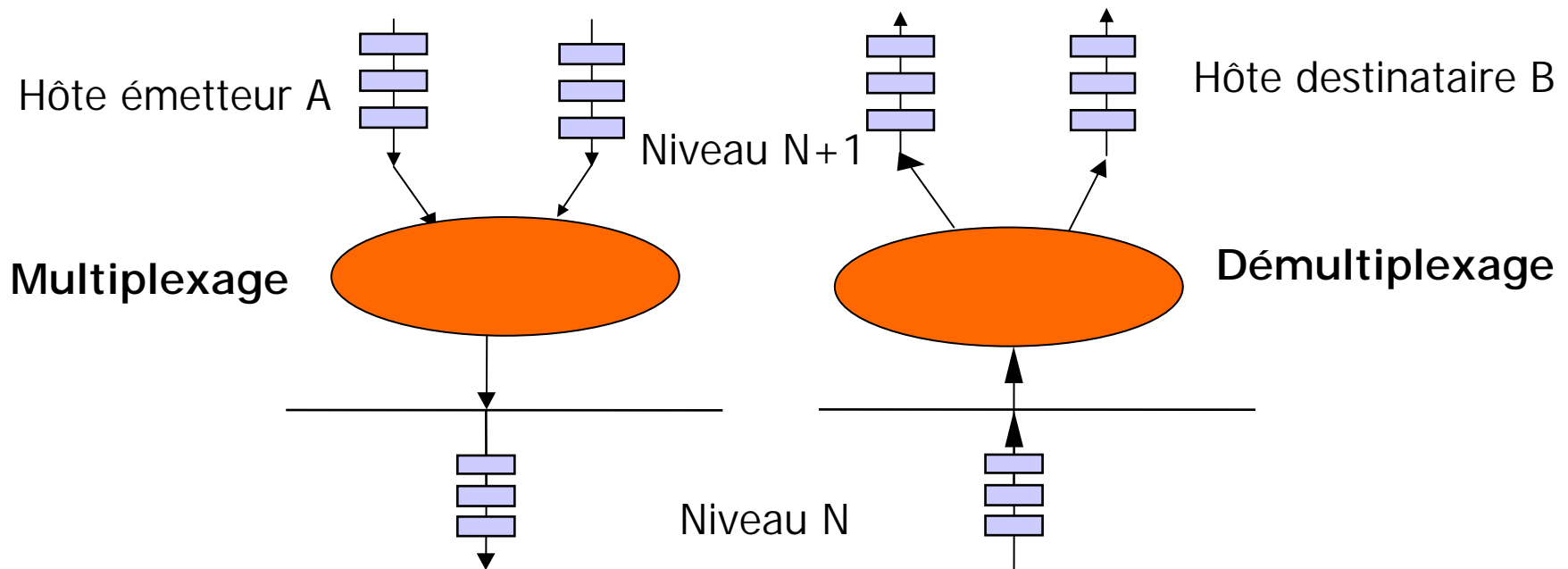
3) Eclatement/Recombinaison

- **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.**

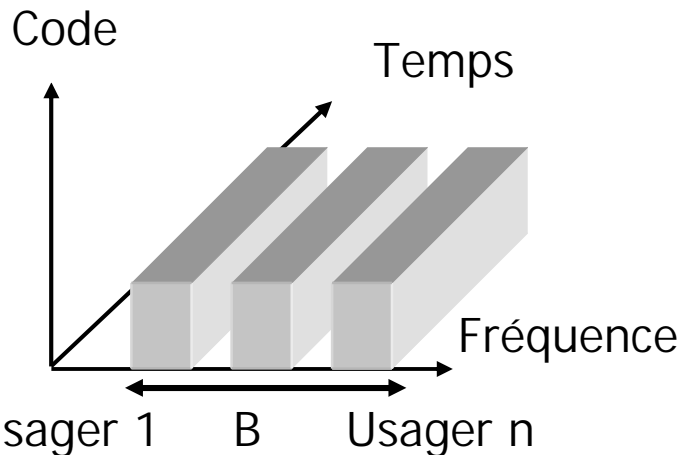
Exemple: multiplexage des protocoles TCP/IP et Novell sur le même câble Ethernet.

- **= > Le Multiplexage est omniprésent en réseaux à tous les niveaux.**

Solutions de multiplexage :

a) Multiplexage en fréquence

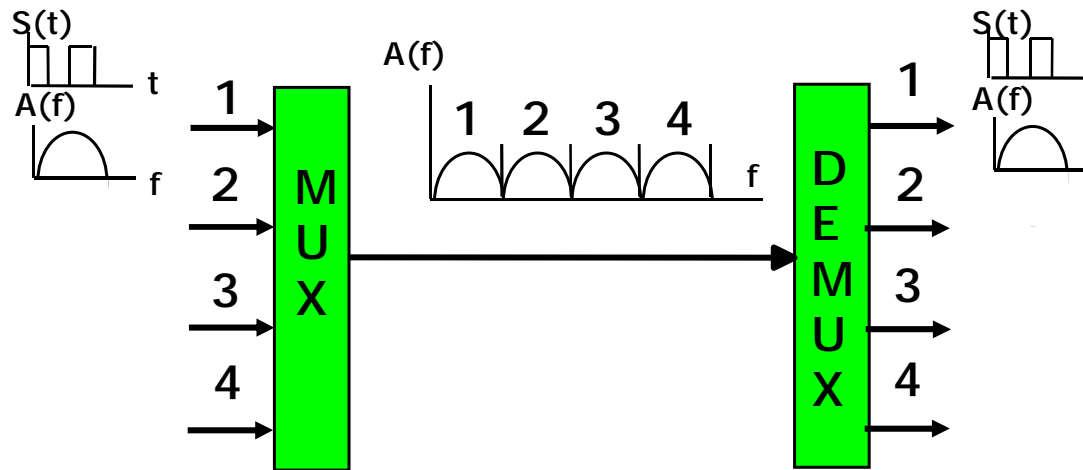
- FDMA 'Frequency Division Multiple Access'.
- AMRF 'Accès multiple à répartition de fréquences'.
- Multiplexage analogique : chaque usager utilise une bande de fréquence fixée selon un codage prédéfini pendant une plage de temps (selon réservation fréquence/temps).



- 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.

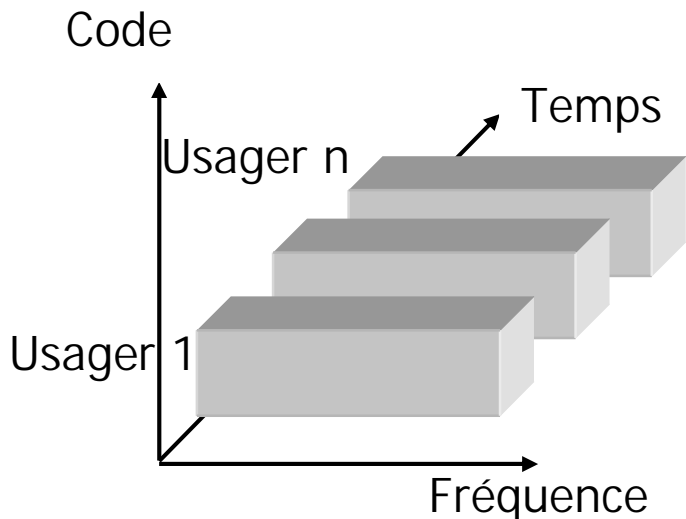


- **Applications anciennes**: téléphonie fixe (groupes primaires à quaternaires) ou cellulaire (Radiocom 2000), réseaux locaux "Broadband".
- **Application actuelle**: modulation OFDM, multiplexage dans des fibres optiques WDM 'Wavelength Division Multiplexing' (16), Dense-WDM (80-160), Ultra-dense (400).

Solutions de multiplexage :

b) Multiplexage en temps (temporel)

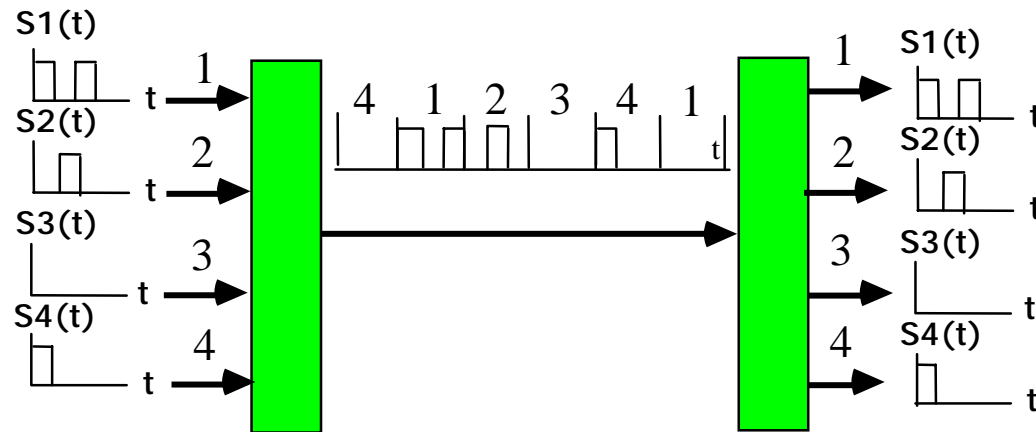
- Chaque usager utilise des d'intervalles de temps (méthode de partage du temps), d'une bande de fréquences prédéfinie, le plus souvent selon le même principe de codage.



- 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 rien à transmettre).
- On forme des trames répétées de manière synchrone.

■ Applications :

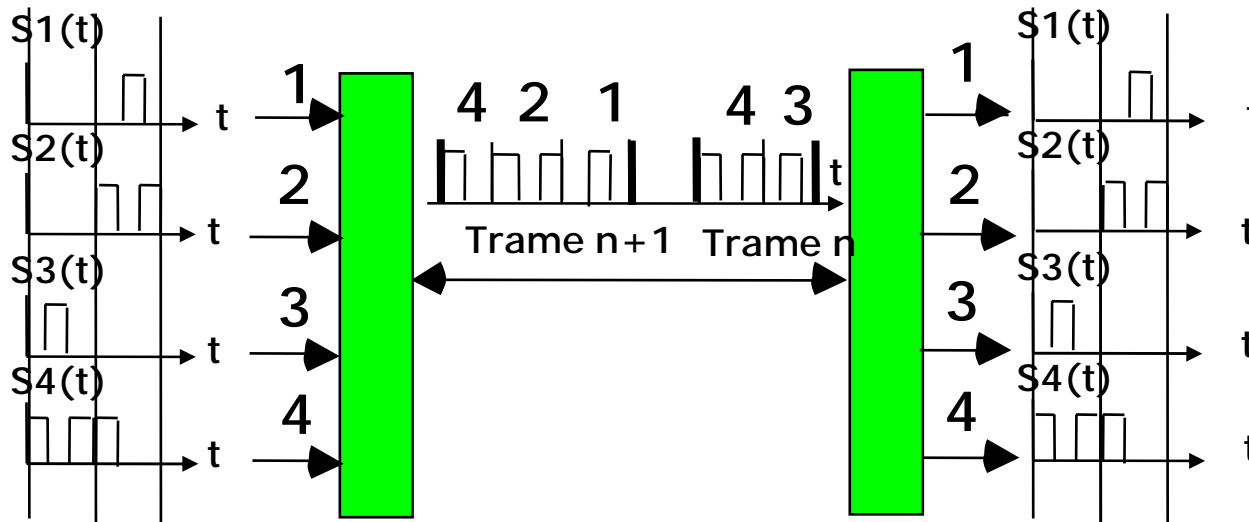
- Téléphonie fixe et RNIS-BE (Réseau Numérique à Intégration de Services Bande Étroite). Téléphonie mobile GSM.

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

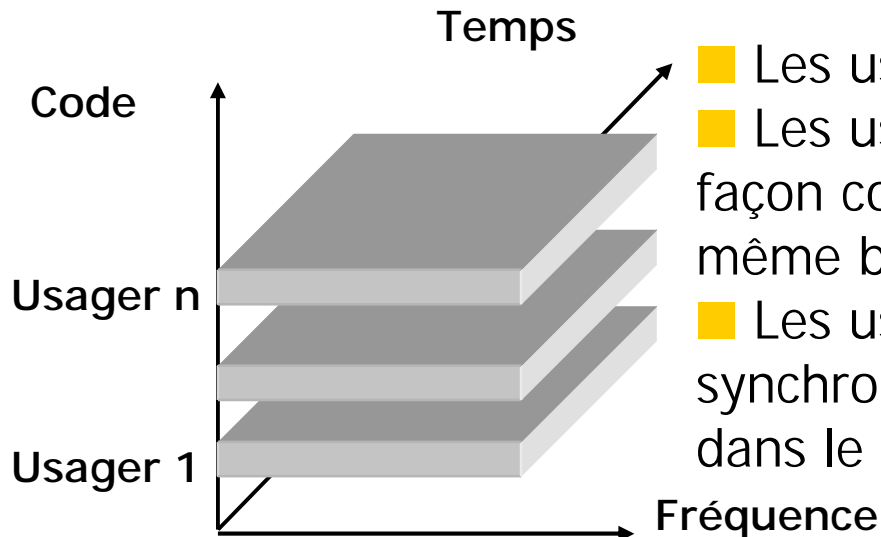
- **Notion de multiplexeur statistique** : Partage d'une voie haut débit en plusieurs voies bas débits.
- **Principe** :
 - Les données sont échantillonnées lorsqu'elles sont présentes sur des voies basses vitesses et sont rassemblées en trames sur la voie haute vitesse.
 - Un codage d'adresse ou un 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).

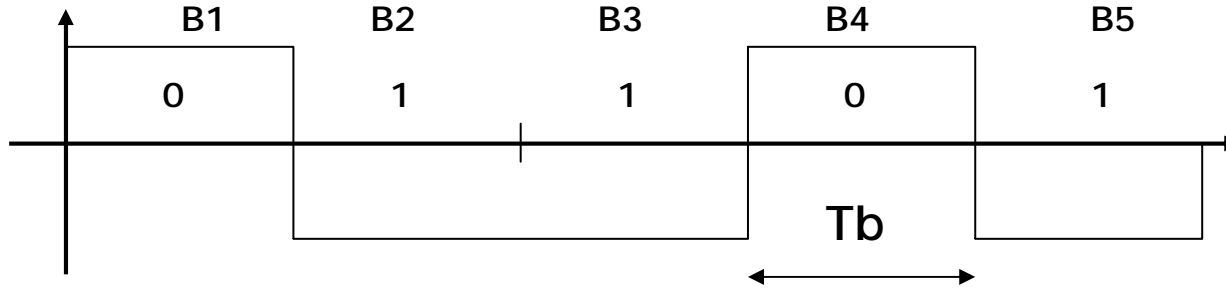


- Les usagers sont séparés par le code utilisé.
- Les usagers peuvent émettre et recevoir de façon continue et simultanément dans la même bande de fréquence.
- Les usagers n'ont pas besoin de se synchroniser dans le temps ni de se distinguer dans le domaine des fréquences..

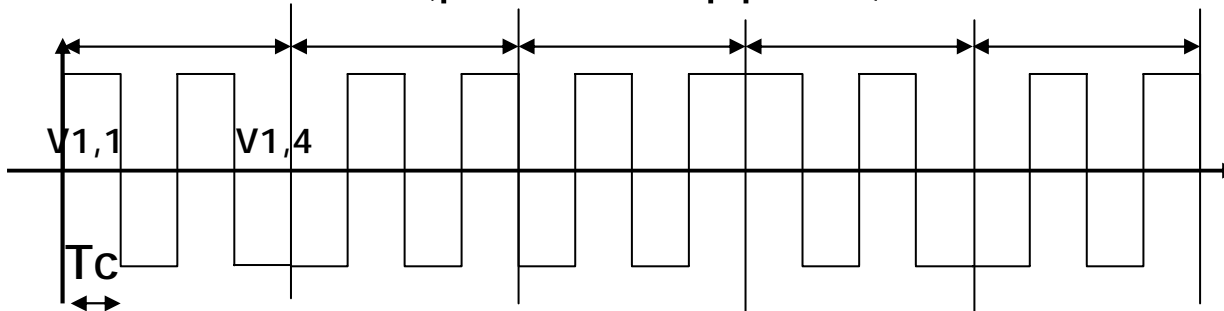
Exemple très simplifié de CDMA :

1) Emission

- Bit $B_i = 0$ ou 1 représenté en niveaux (NRZ).
 $0 \Rightarrow +1$ et $1 \Rightarrow -1$ pendant un temps bit T_b .



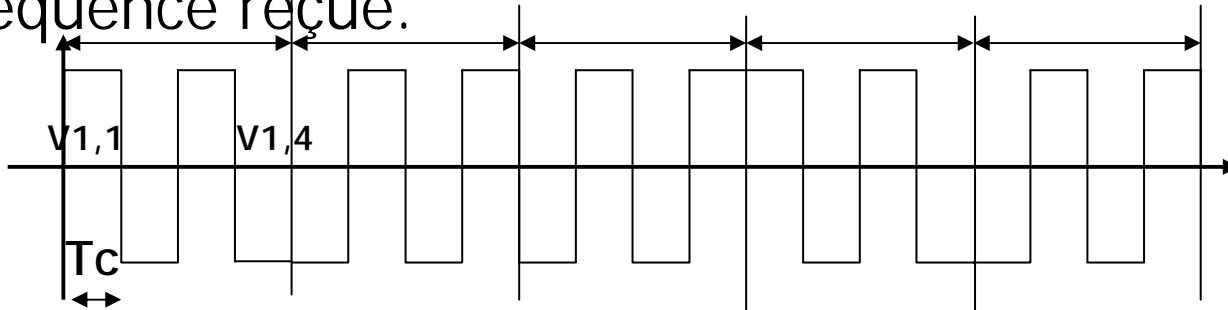
- Pour chaque usager définition d'une séquence d'étalement.
Chaque temps bit est divisé en n 'chips' ($T_b = n * T_c$)
- Exemple pour $n=4$: On utilise un code pour 0 $C_1 = +1$, $C_2 = -1$, $C_3 = +1$, $C_4 = -1$ (pour 1 l'opposé).



Exemple très simplifié de CDMA :

2) Réception

- Séquence reçue.



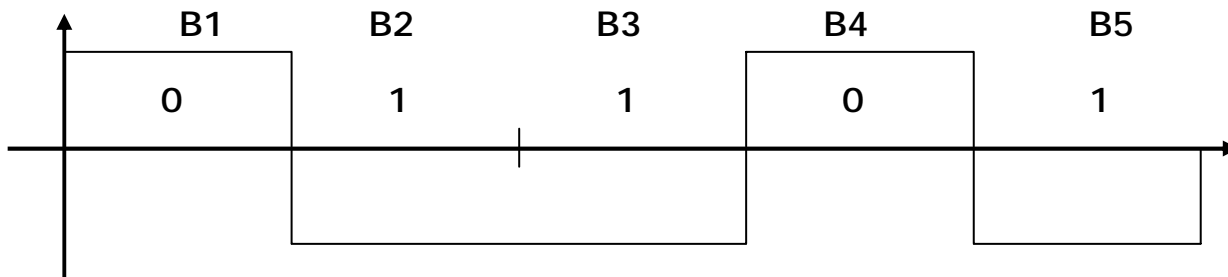
- Décodage pour : $C_1 = +1$, $C_2 = -1$, $C_3 = +1$, $C_4 = -1$

On calcule $B_i = (1/n) \sum_{j=1,4} V_{i,j} C_j$

- Pour le bit $B_1 = (1/4) * (1+1+1+1+1) = 1$

Pour le bit $B_2 = (1/4) * (-1-1-1-1-1) = -1$ 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 où 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 **totalelement disponible** et indifférent au type de données transférées.

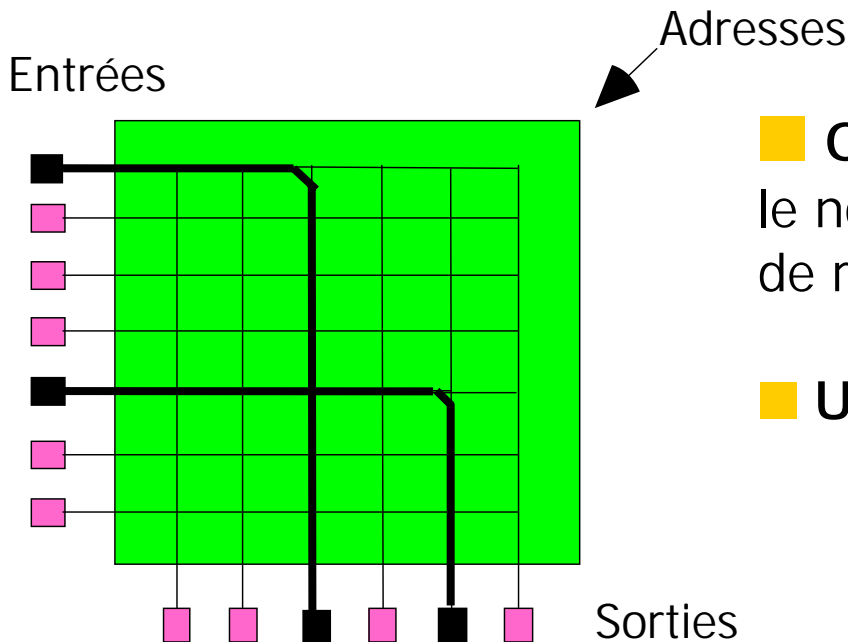
■ Permet de **réserver une capacité nette pour des trafics synchrones ou temps réel**: voix, image, données à échéances.

■ **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

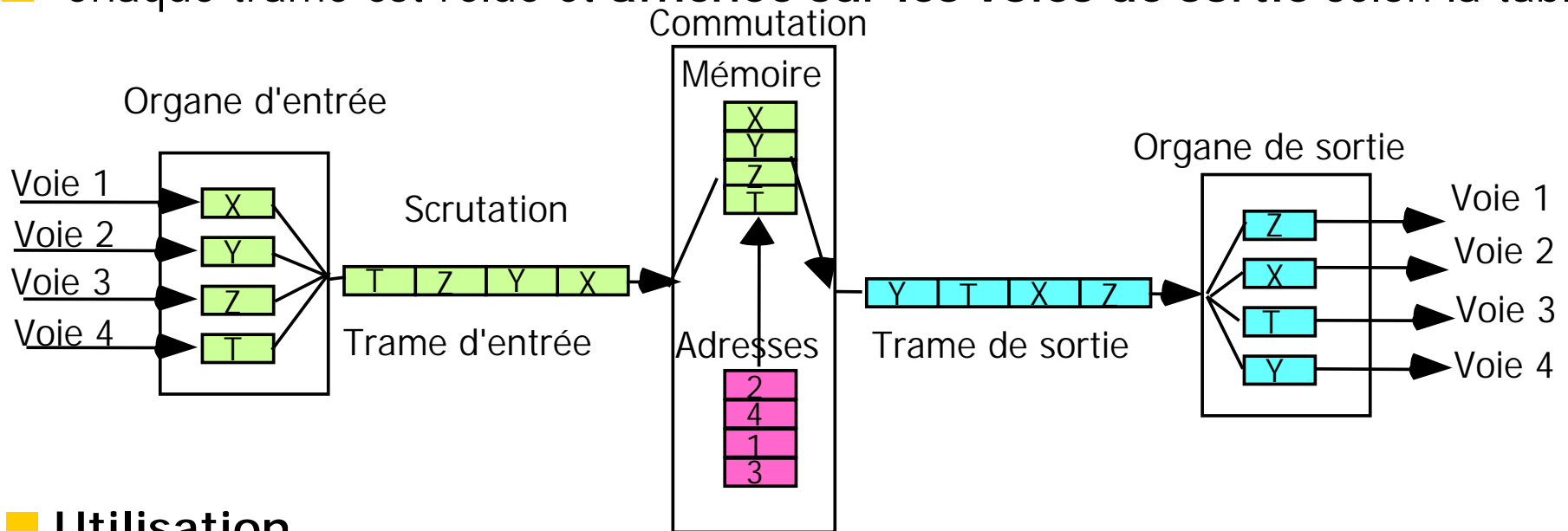
■ Utilisation

- 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.



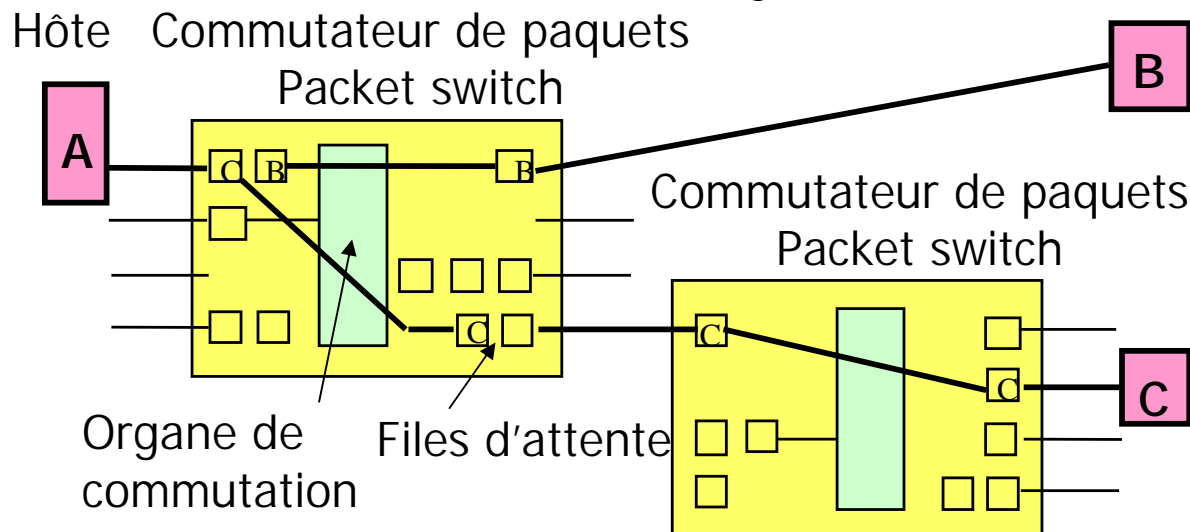
■ Utilisation

- 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 où 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

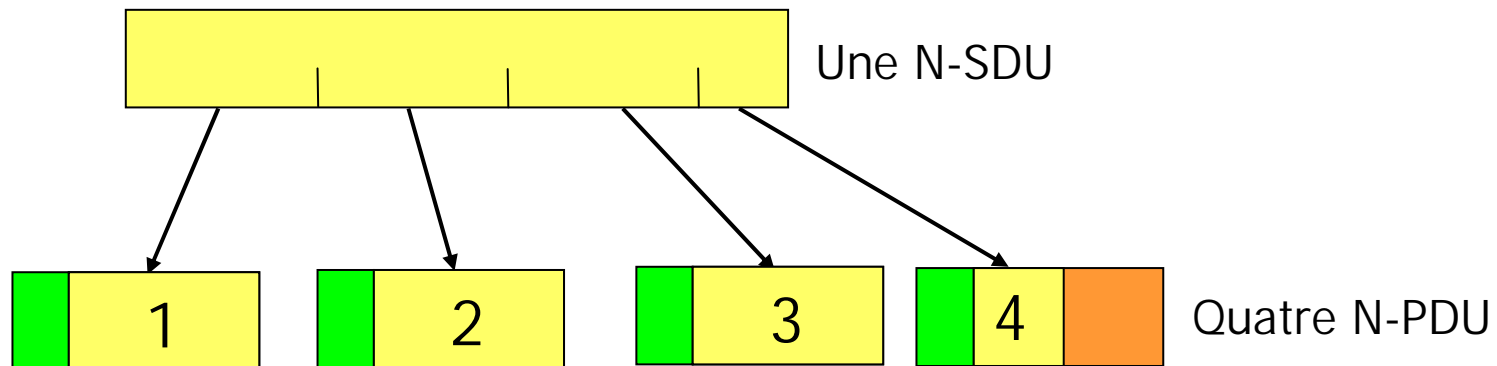
■ **Notion de MTU :** 'Medium Transmission Unit'.

■ **Autre situation:** La voie de communication est trop bruitée.

■ Exemple : Wifi.

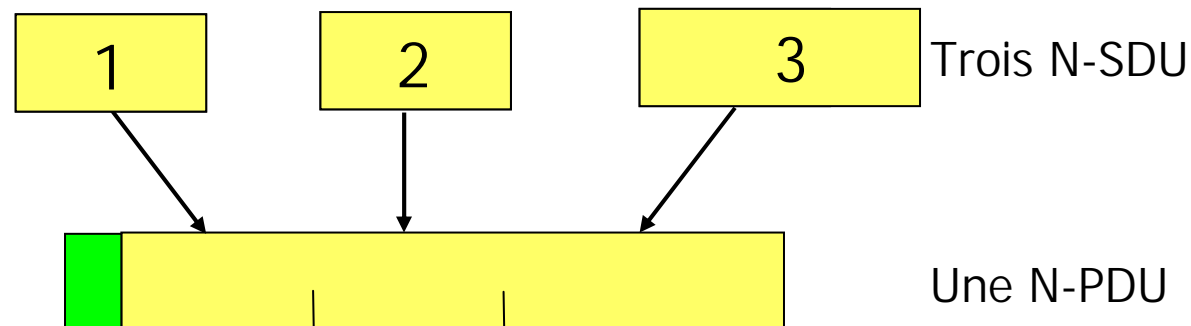
■ **Solution (rencontrée dans certains niveaux 2,3,4):** découper les messages longs en morceaux plus courts (segmentation ou fragmentation).

■ **Opération inverse : réassemblage** (identifiant de fragment, position).



7) Groupage ('Concaténation')

- **Problème** : dans le cas où 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 précedence 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'utilisateur '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 physique 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 différents 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 où 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" où 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 (Dijkstra 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.

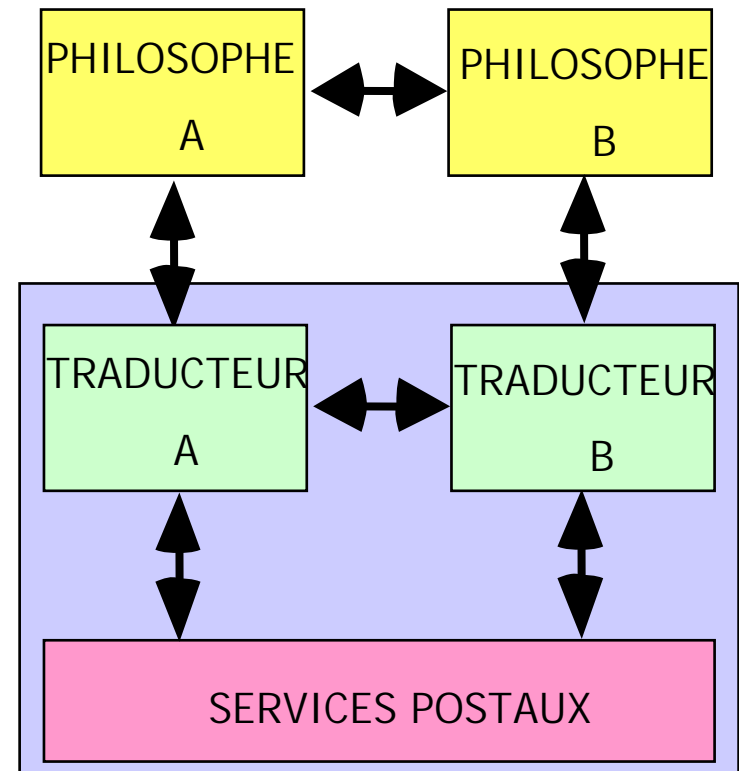
Organisation orientée composants logiciels (objets)

- 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.

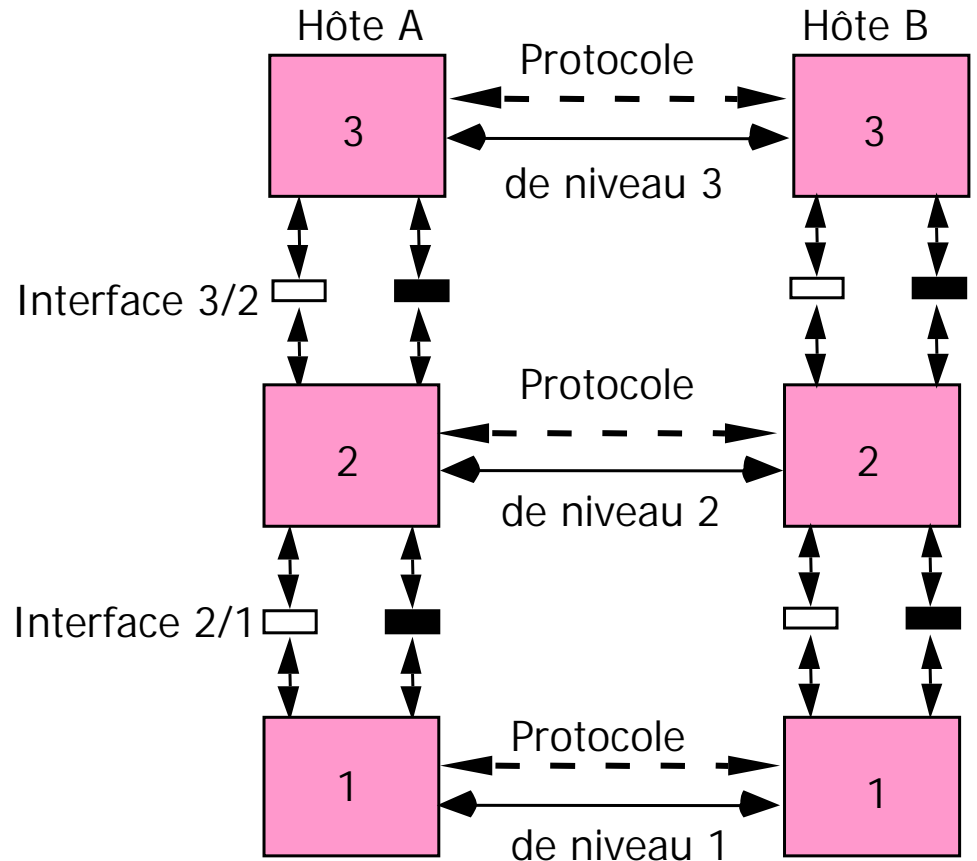
Architecture de communication :

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

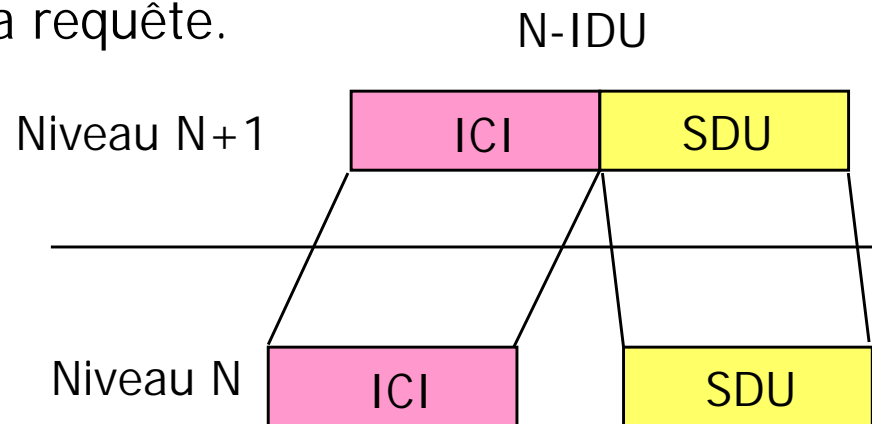
- **Instance** d'un ensemble de primitives permettant la réalisation effective du service (analogue de l'instanciation de classe en approche objet).

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'utilisateur souhaite transmettre effectivement à l'utilisateur 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

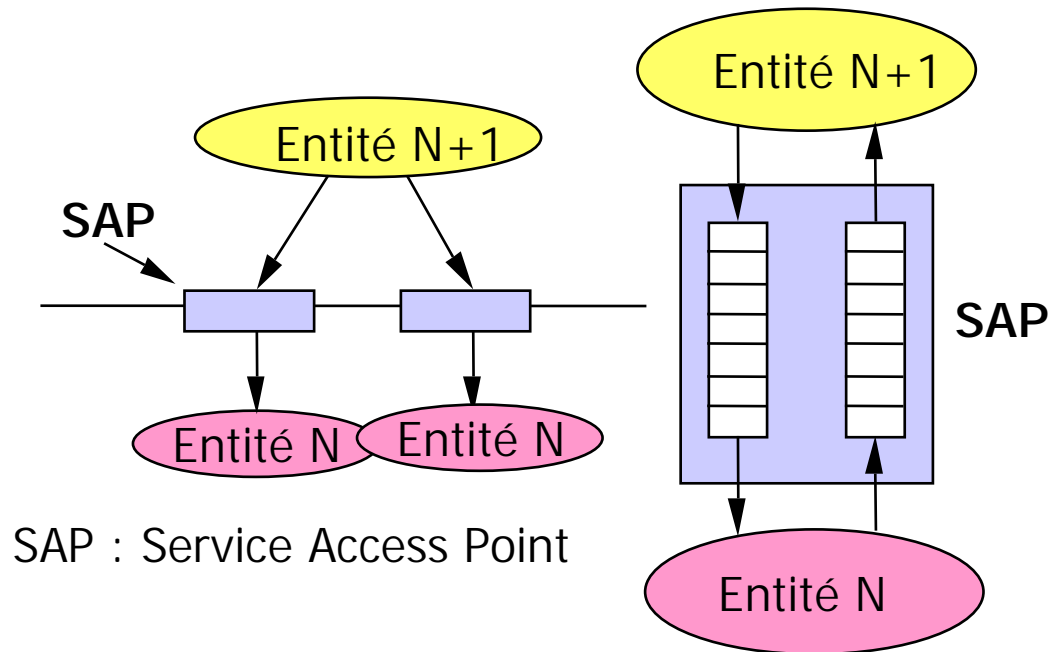
■ **Guichet** permettant à une entité de demander un service à une autre entité .

■ **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**.



Réalisation d'un SAP
en schéma producteur
consommateur

Approfondissements: Protocoles

■ Unités de données de protocole ("PDU Protocol Data Unit")

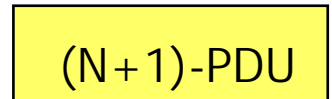
- L'ensemble des objets échangés entre niveaux appariés.
- Composé d'un (N+1)-PDU et d'une information de contrôle (N)-PCI.

■ Informations de contrôle protocolaire ("PCI "Protocol Control Information")

■ Ensemble des informations de contrôle de l'échange entre niveaux.

■ Exemples :

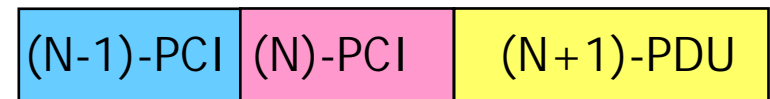
- adresses émetteur et destinataire
- type
- version de protocole utilisée.



(N+1)-PDU Message (Transport)

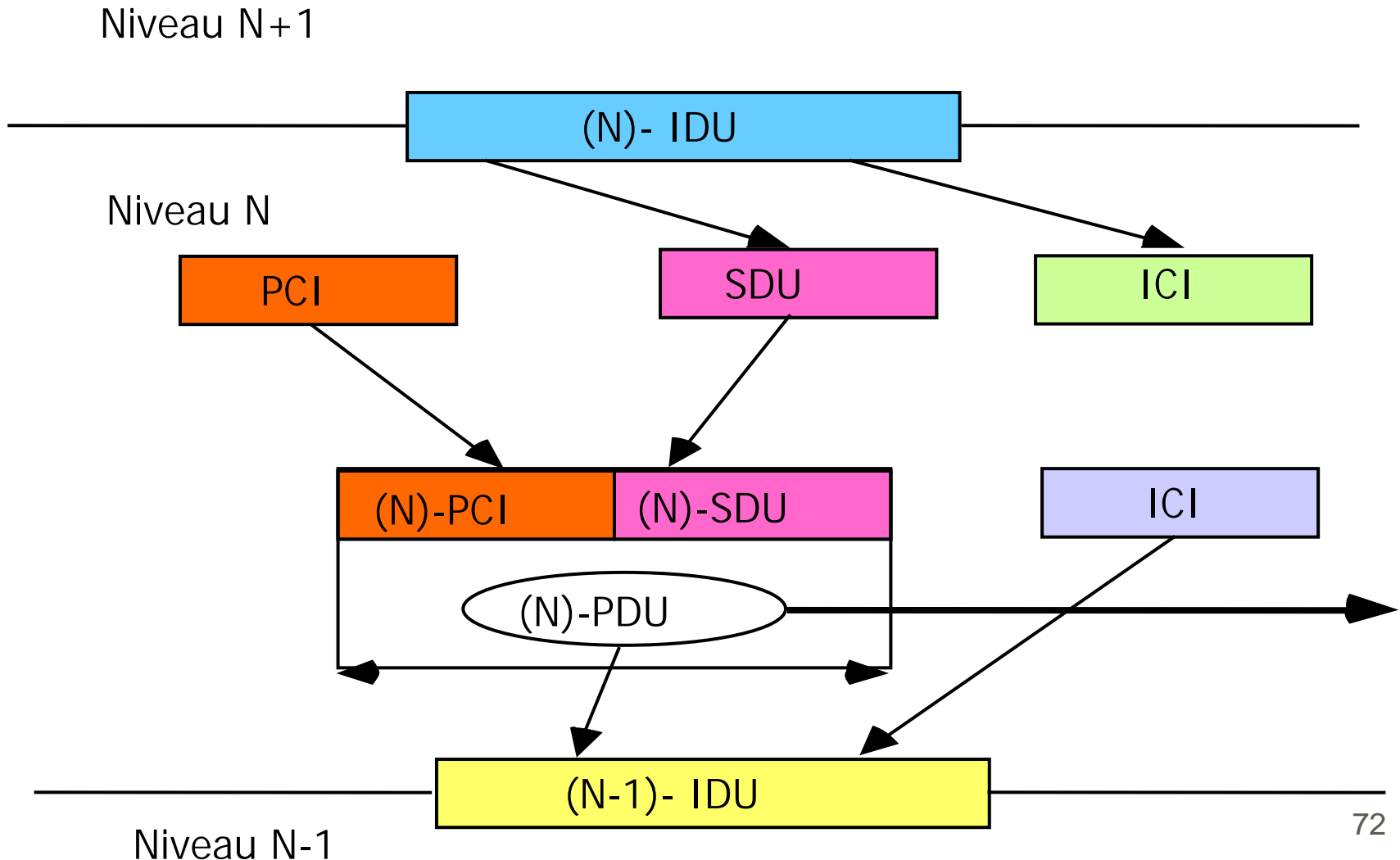


(N)PDU Paquet (Réseau)

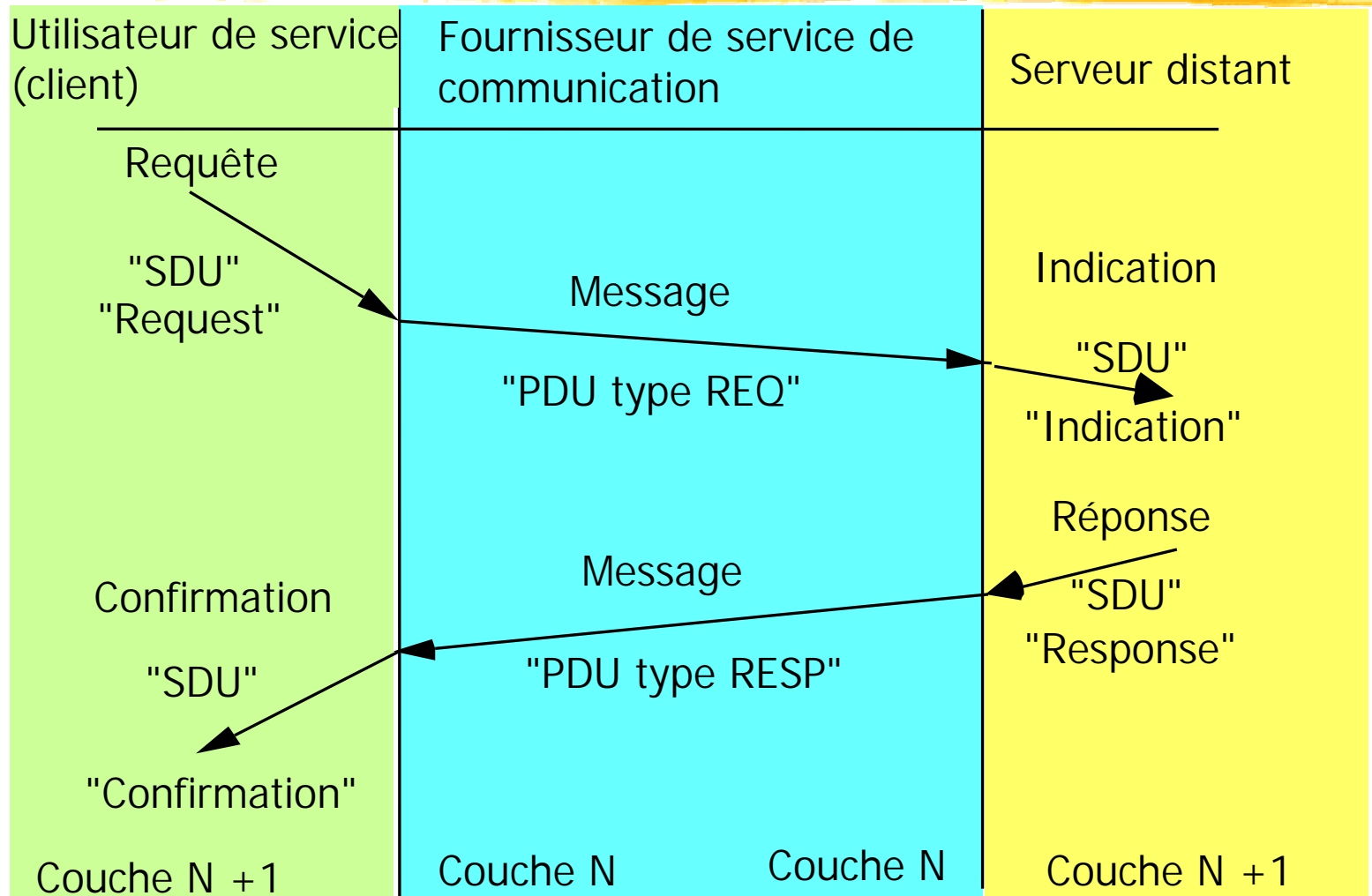


(N-1)-PDU Trame (Liaison)

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)

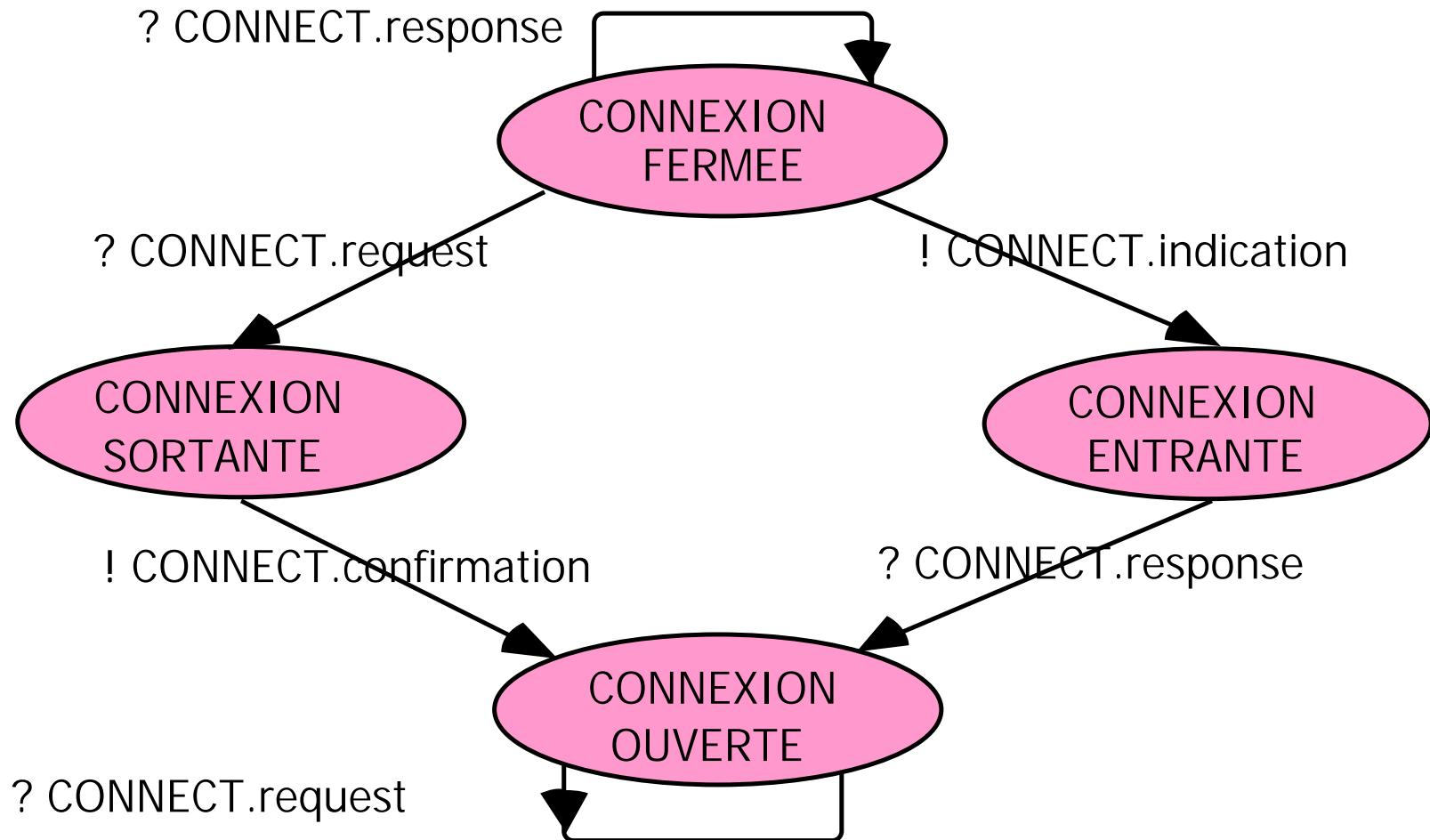
Unités de protocole

- **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

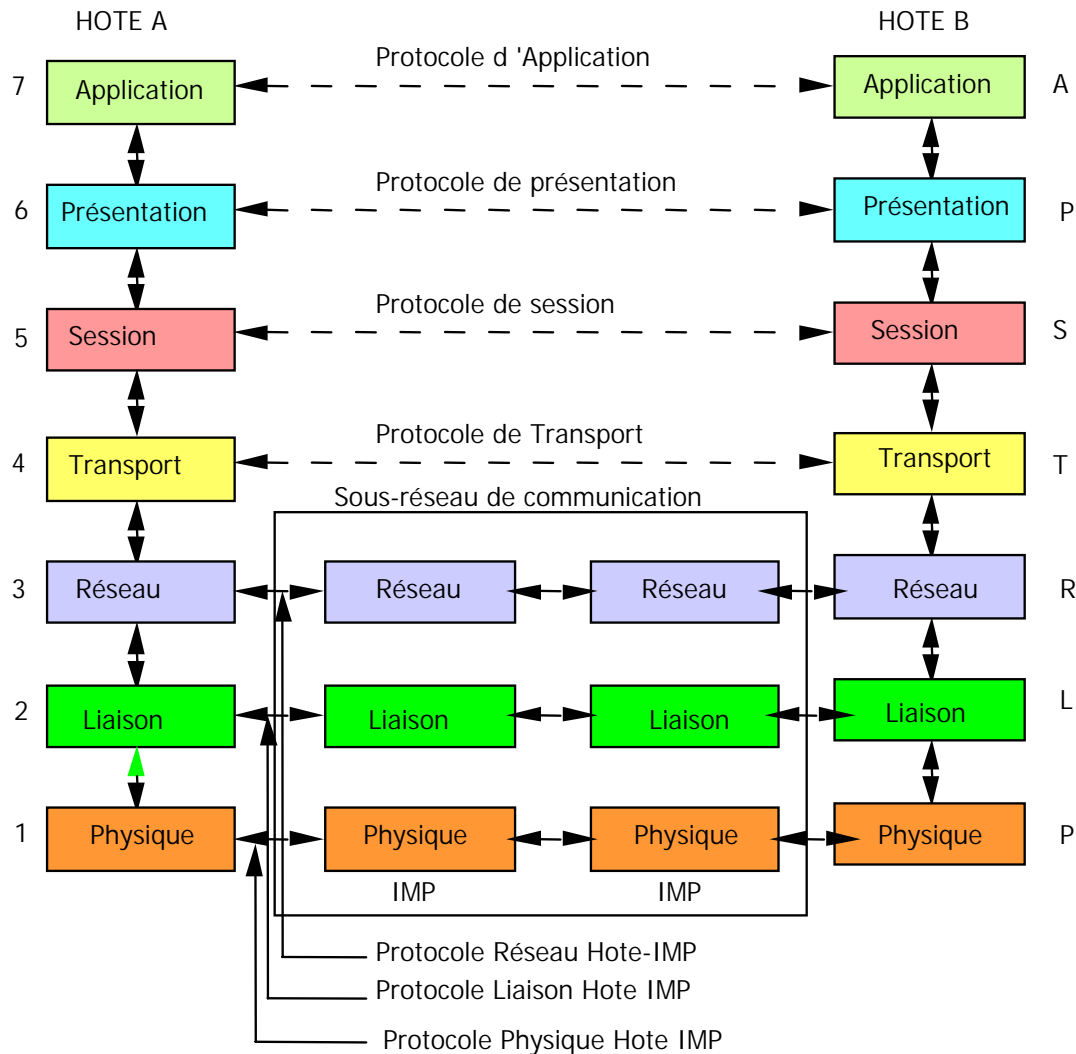


Le modèle OSI (ISO/IEC 7498) 'Open System Interconnection'

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 **modifiées** 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 nécessaires 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 (approche 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 SNA₈₄
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 relèvement 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 réalisations 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"

- **Hierarchie numérique synchrone : G707 , G708, G709** Normes dérivées des travaux SONET ("Synchronous Optical NETWORK") (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'utilisateur**
 - 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.

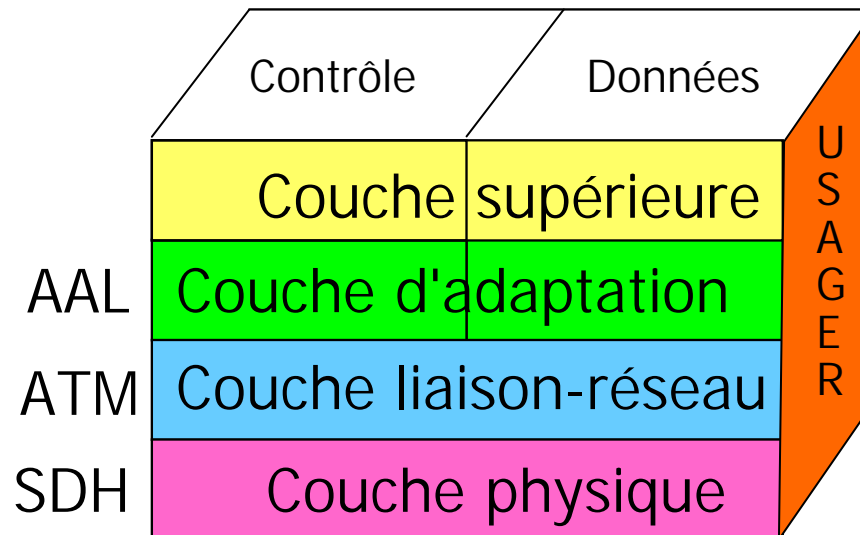
Réseaux ATM

'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-ISDN Broadband 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.

LIAISON	Point à point 802 - 2
	Accès au médium 802.X (3->N)
PHYSIQUE	802.X (3->N)

■ 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

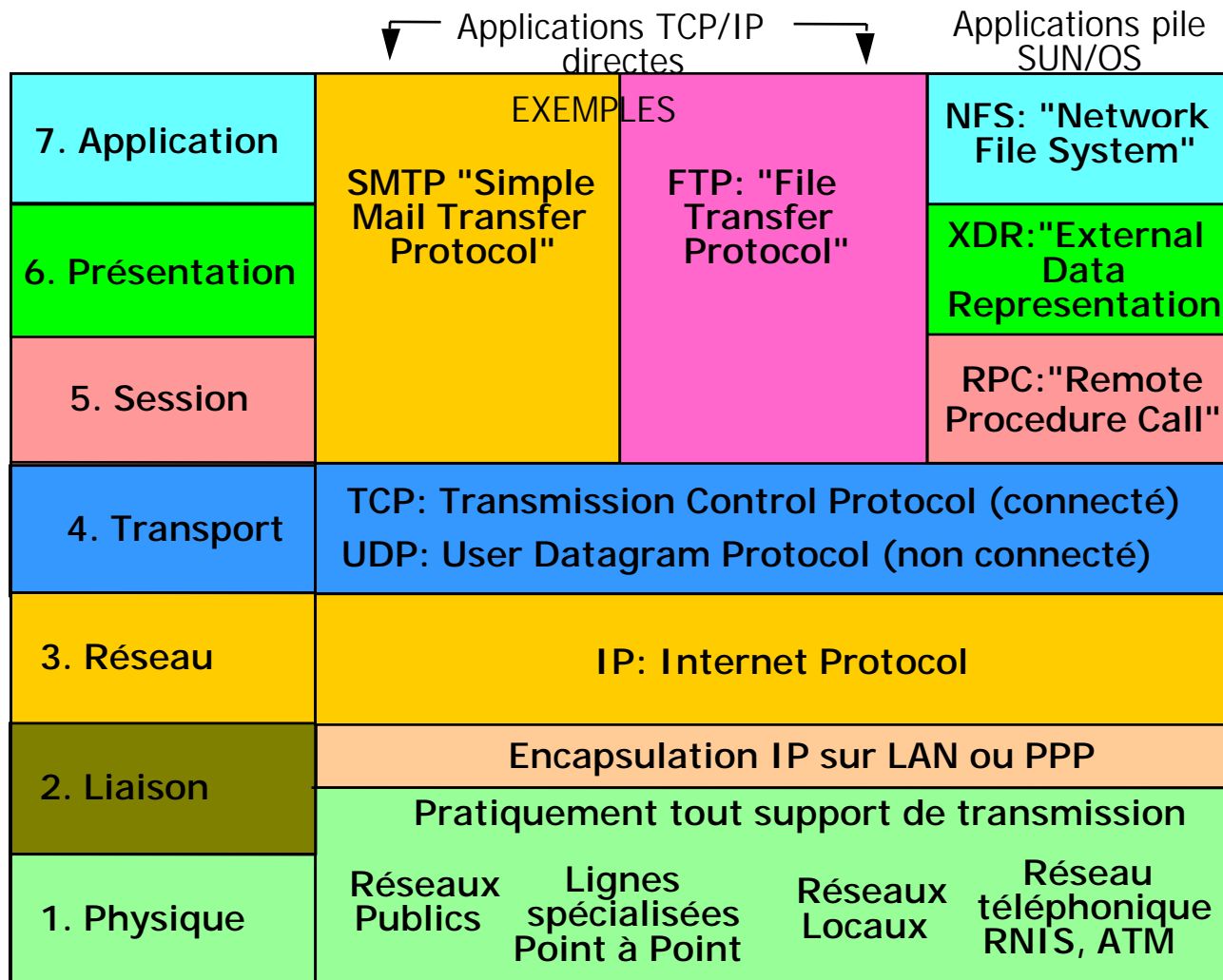
■ Héritier du réseau ARPANET

- 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 progrès technologiques et recherche.

Organisation de la pile Internet



Introduction Notions générales



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.