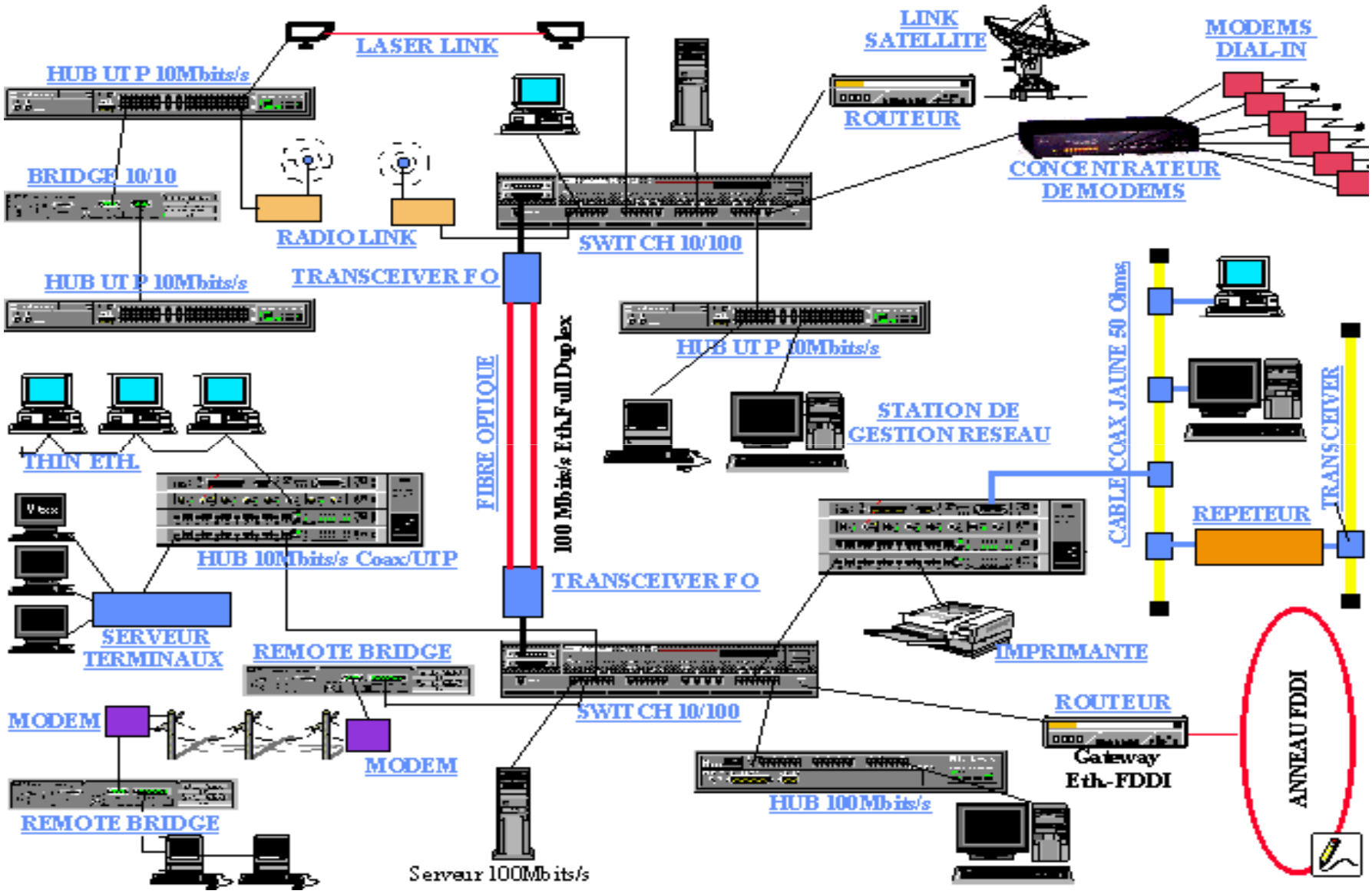
Réseaux



Gilney DAMM

Objectif des réseaux

- Créer un médium de communication
- Partager les ressources, les informations, les périphériques
- Améliorer la fiabilité
- Réduire les coûts
- Echange de données

Idéal

- aucune erreur de transmission
- transfert de très grandes quantités d'information
- temps de réponse ou de transfert négligeable
- transfert de toute information
- présence simultanée en plusieurs endroits

Intérêt d'un réseau

- Services de courriers électroniques
- Services de partage de fichiers
- Services de transfert de fichiers
- Services de partage de périphériques
- Services de terminaux virtuels

- Quelques ouvrages généraux
  - Andrew Tanenbaum - RESEAUX Architectures, protocoles, applications
    - basé sur le modèle OSI
  - Guy Pujolle - Les Réseaux
    - s'attaque aux technologies plus récentes,
    - bonne bibliographie

Type de réseau

Réseau étendu ou WAN

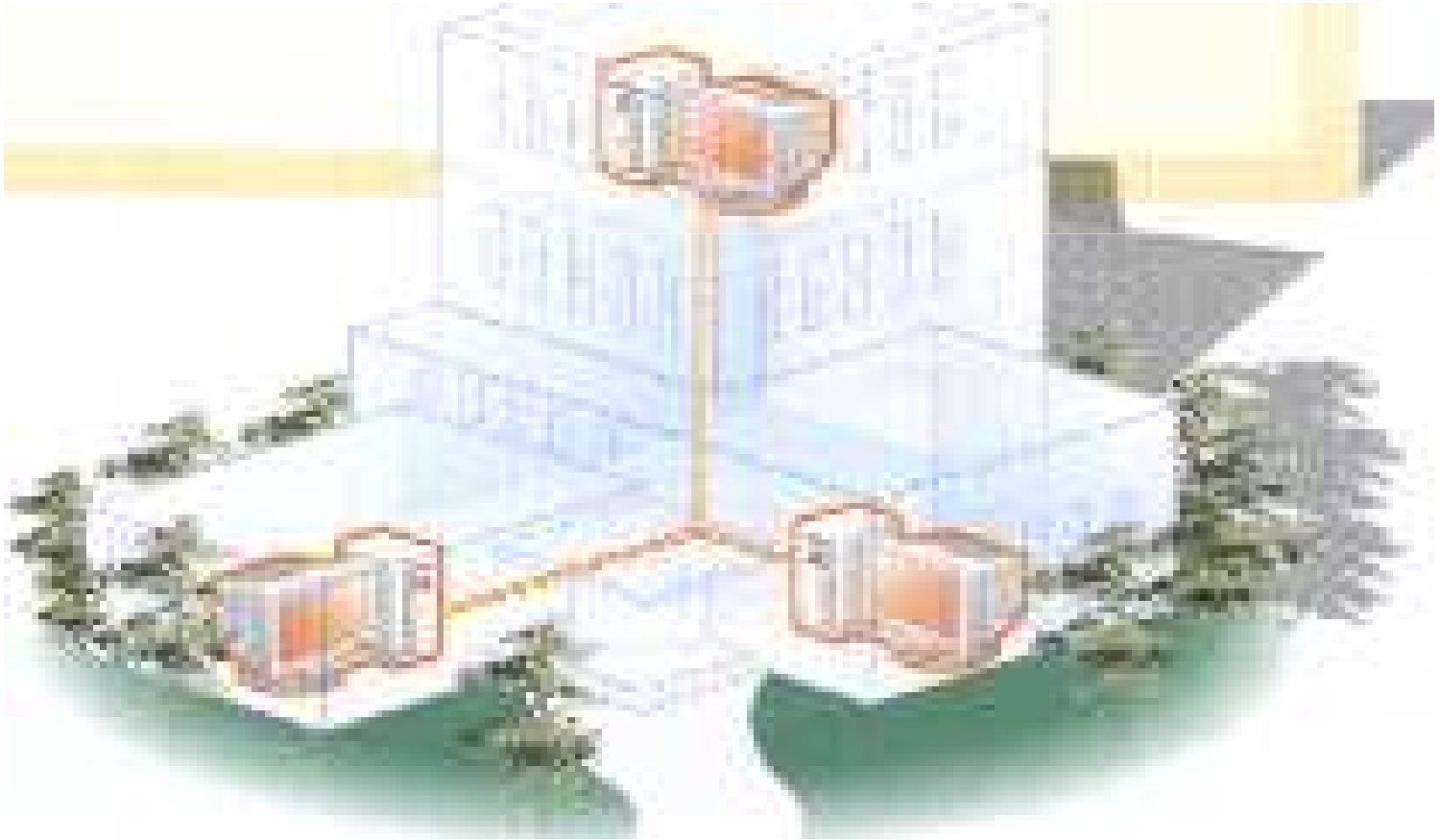
Wide Area Network

Réseau métropolitain ou MAN

Metropolitan Area Network

Réseau local ou LAN

Local Area Network



Classification des réseaux

	WAN Réseaux publics	MAN Réseaux fédérateurs	LAN RLE - RLI
taille géographique	Quelques milliers de kilomètres	100 km	1km
Nombre d'utilisateurs	Plusieurs milliers	1000	100
Opérateur	Public ou privé différent des utilisateurs	groupement d'utilisateurs	l'utilisateur
Facturation	Volume et durée	forfait	gratuit
Débit	500 Mb/s à 20 Gb/s	100 Mb/s	100 Mb/s
Taux d'erreur	10-3 à 10-6	< 10-9	< 10-9
Délai	< 0,5 s	de 10 à 100 ms	de 1 à 100 ms

WAN : Wide Area Network
MAN : Metropolitan Area Network
LAN : Local Area Network

RLE : Réseau local d'entreprise
RLI : Réseau local Industriel

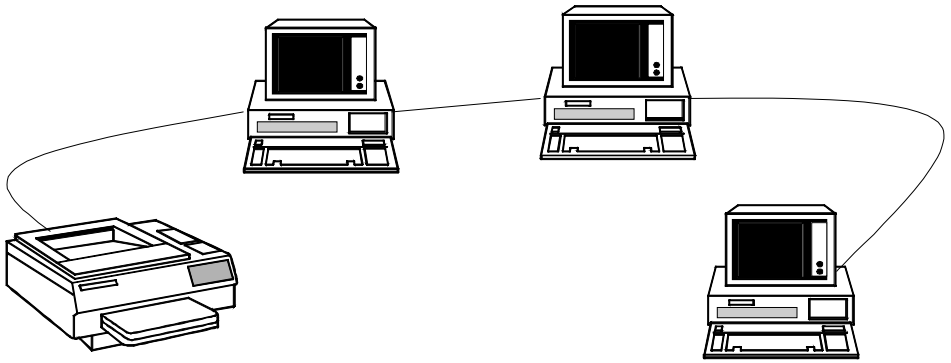
Les débits

- Unités :
  - le baud
  - bit par seconde
  - Kbit / s, Mbit/s ou Mb/s, Gb / s, Tb / s
- Attention !!!
  - 1 octet = 8 bits
  - 1 Ko = 1024 octets (210 octets)
  - 1 Ko = 1024*8 = 8192 bits ≈ 8 Kb
- Connexion parallèle (ordinateur/imprimante)
  - de l'ordre de 115 Kb/s
- Connexion série sur un PC
  - de 75 bit/s à 921 Kb/s
  - Connexion Internet par modem de 14,4 à 56 Kb/s

Architecture poste à poste

Environ 10 postes - pas d'ordinateur central
Réseau dit léger ou peer to peer ou point à point

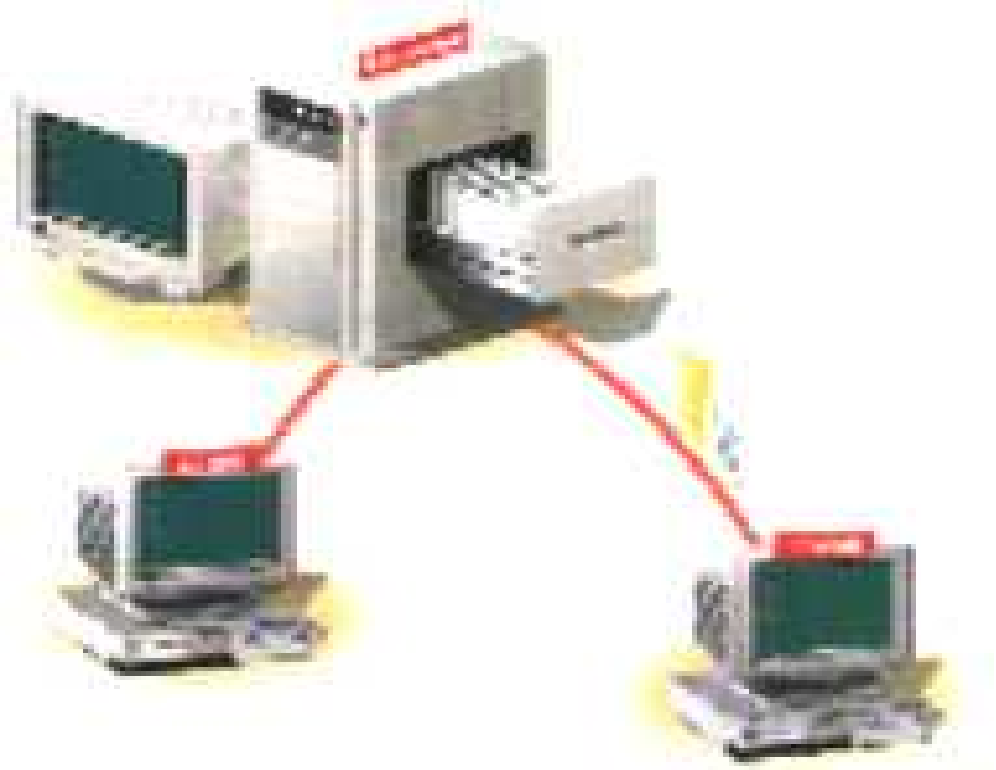
- Utilisation
  - partage d'imprimante
  - partage de disque
  - transfert de fichier entre une dizaine de postes
  - utilisation d'un logiciel se trouvant sur un autre poste



- Caractéristique
  - Ce type de réseau utilise des cartes/adaptateurs réseaux
  - Réseau fondé sur l ’utilisation individuelle des ordinateurs chaque personne connectée à la responsabilité du partage dont elle peut faire profiter à autrui
- Avantages
  - Simplicité et souplesse d’utilisation
  - Prix bas
- Inconvénient
  - Nombre de postes et performances limités
  - Sécurité limité

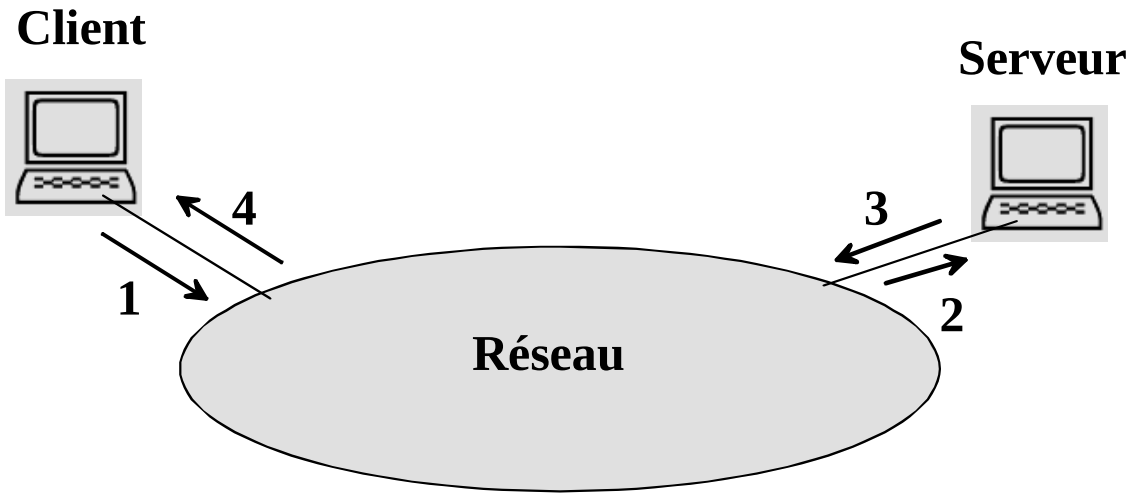
Architecture Client-Serveur

Nombre de postes indifférent



Ordinateur central : SERVEUR

La notion de client et de serveur



un client formule une requête à destination d'un serveur
le serveur répond à une requête issue d'un client

Exemple d’applications client-serveur

- un navigateur Internet (Netscape, Outlook) est une application cliente
- un site Web est une application serveur

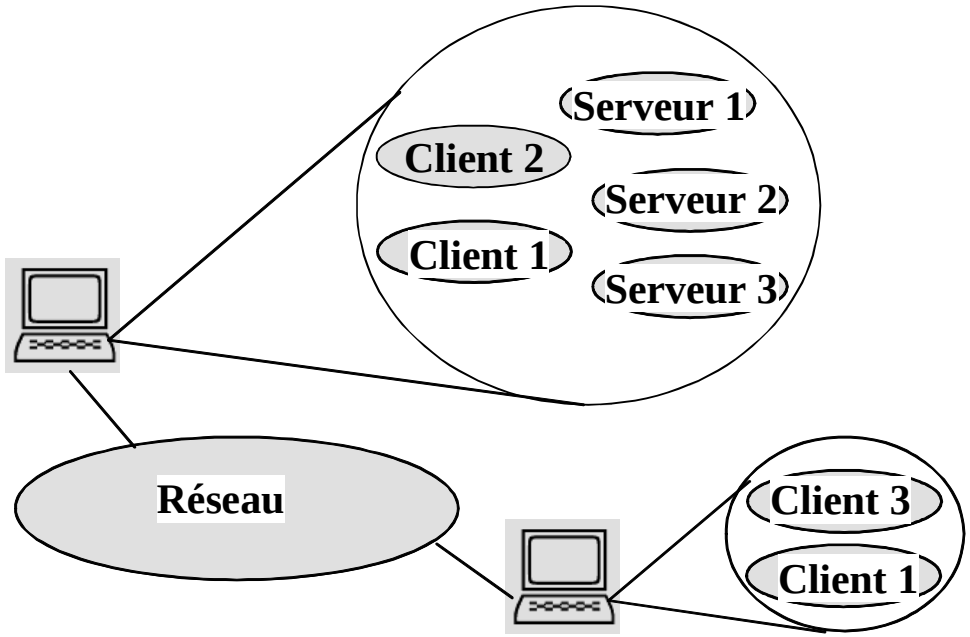
- un agent SNCF dispose d’une application permettant la réservation d’une place de train
- la base de données de réservation de la SNCF « Socrate » est une application serveur

- échange de fichiers sur Internet (téléchargement): client FTP et serveur FTP

- connexion sur une machine distante :
client « telnet », serveur « telnetd »

A un instant donné, une machine peut exécuter « simultanément » plusieurs programmes :

➔ programmes clients et/ou programmes serveurs



➢ **Une machine peut être « cliente » des applications « serveur » qu'elle supporte.**

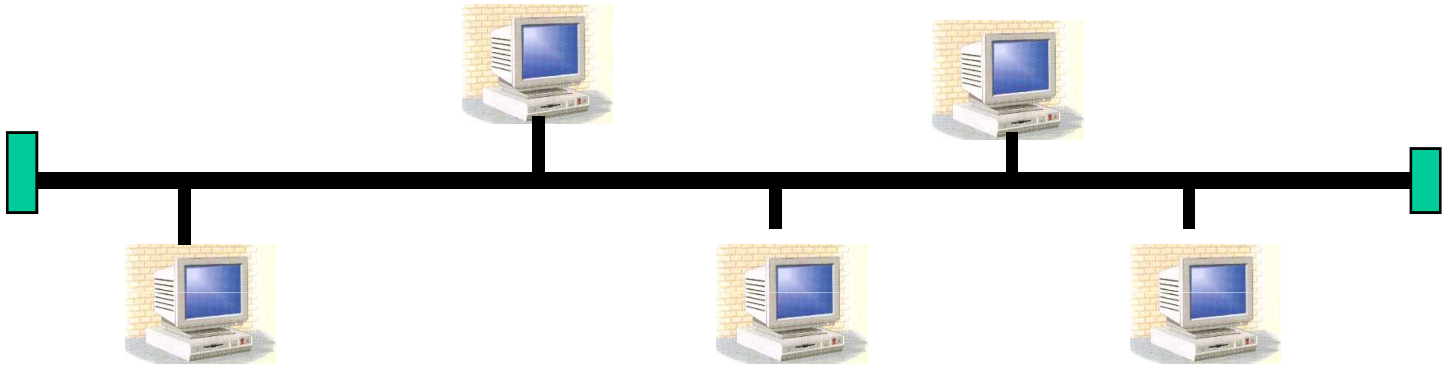
➔ une application « réseau » peut être testée avec une seule machine mais attention les délais de communication sont alors nuls !

➢ Un serveur peut servir « simultanément » plusieurs clients

Topologie Physique

- La topologie représente la manière dont les équipements sont reliés entre eux par le support physique (câble coaxiaux, fibre optique…..)
- Son choix s'appuie sur :
  - Le bilan des équipements informatiques existants
  - L'analyse des besoins immédiats
  - La disposition géographique des équipement et des locaux
  - L 'expression des besoins futurs
  - Les coûts d 'investissement et de maintenance
- 4 topologies sont à distinguées :
  - l'étoile
  - l'anneau
  - le bus
  - l'arbre

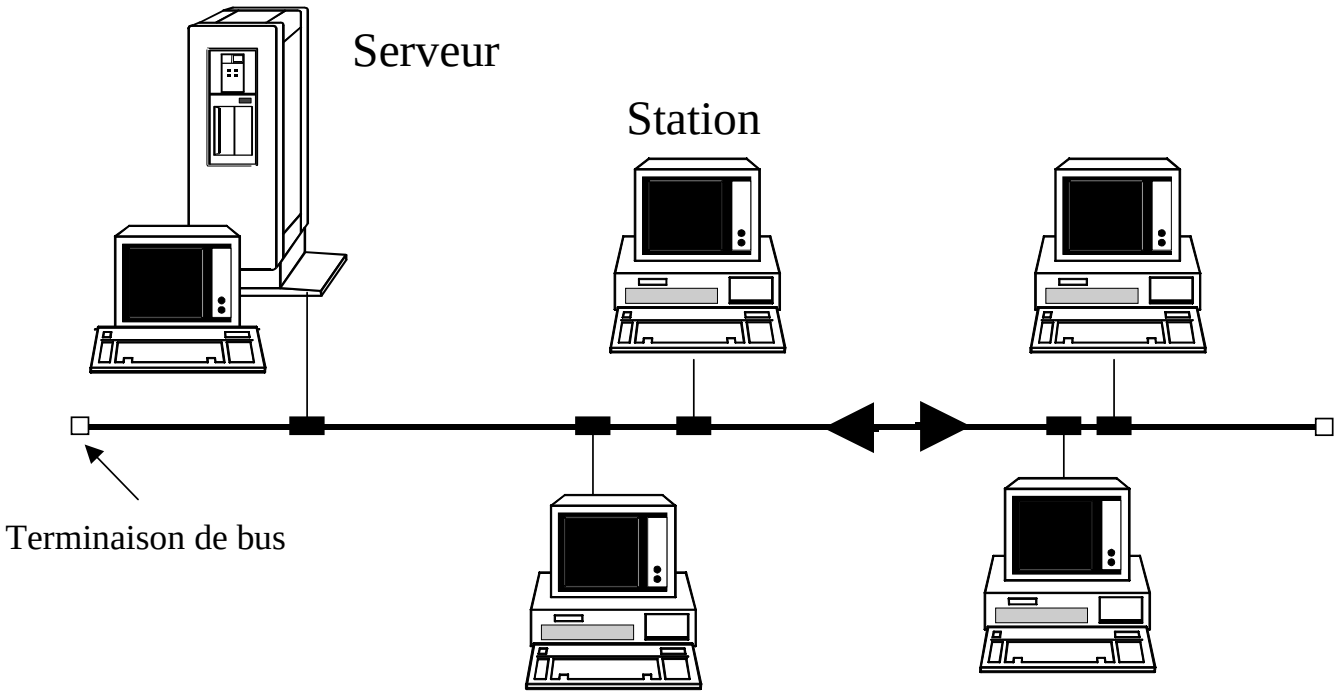
Topologie en BUS



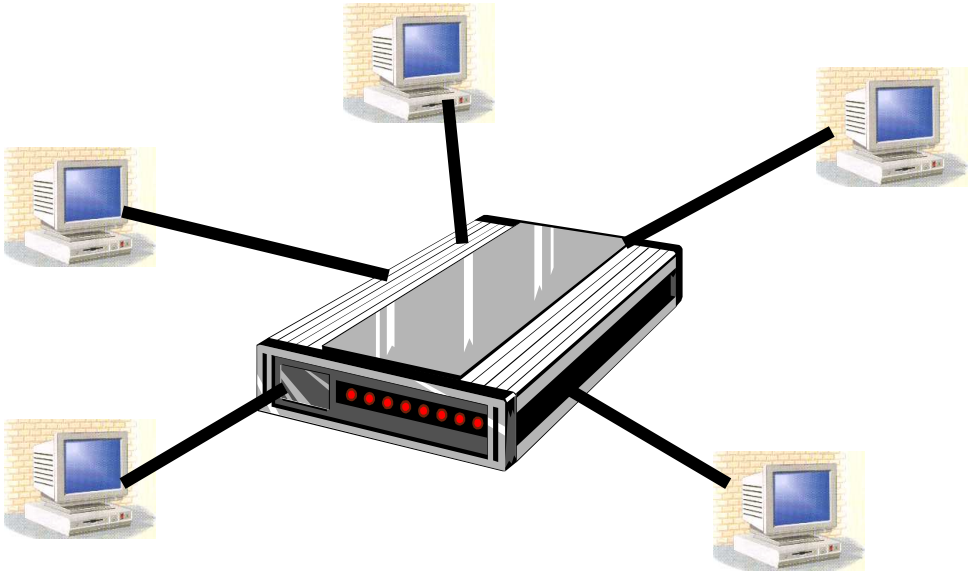
Ordinateurs équipés d ’un «T» et reliés par un même câble terminé par un «bouchon»

Topologie bus

- La communication entre les stations s ’effectue sur un câble commun
- Les stations sont connectées via des jonctions passives qui ne régénèrent pas le signal



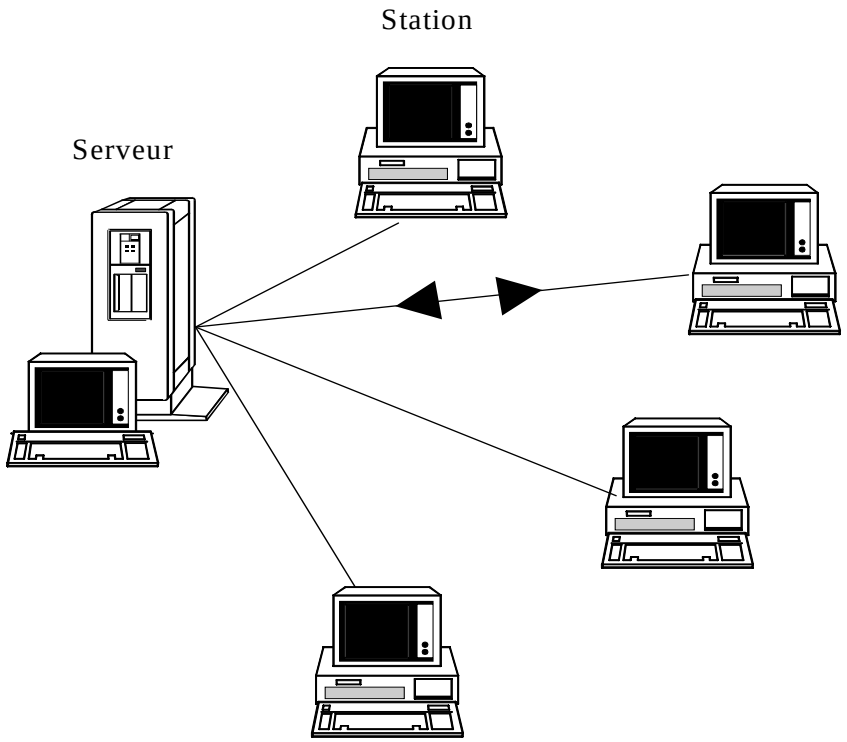
Topologie en étoile



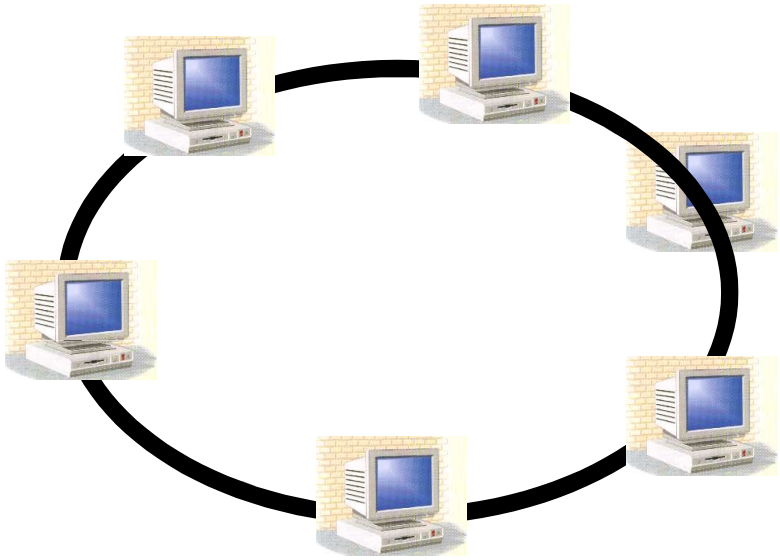
chaque ordinateur est relié à un système central par un câble

Topologie en étoile

- Topologie la plus ancienne
- Chaque station a un accès direct au nœud
- Logiciel centralisé $\Rightarrow$ simplicité du système
- Problème de fiabilité
- Manque de souplesse



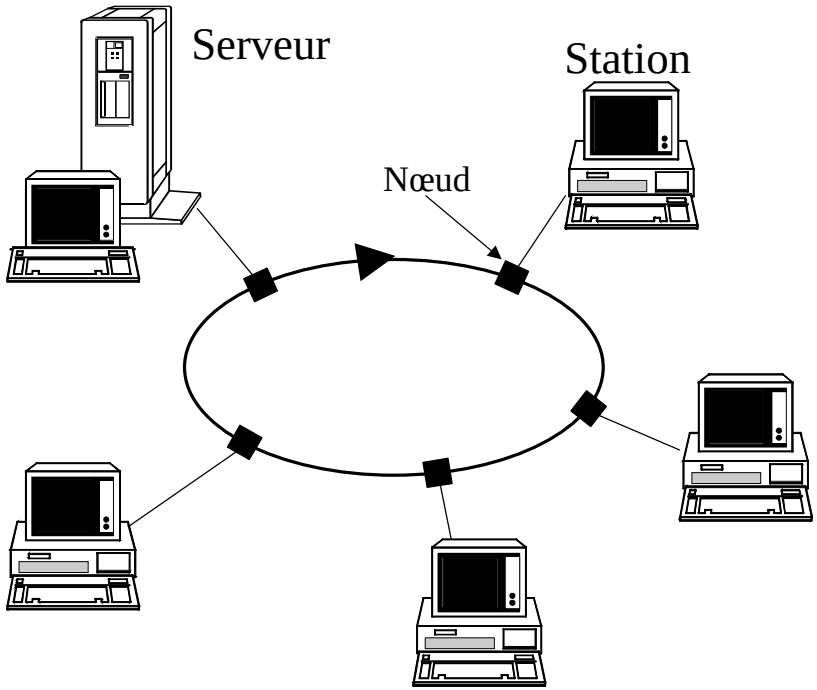
Topologie en anneau



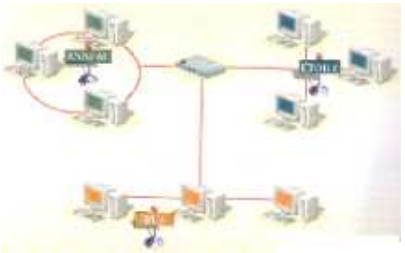
ordinateurs reliés de manière circulaire par un même câble

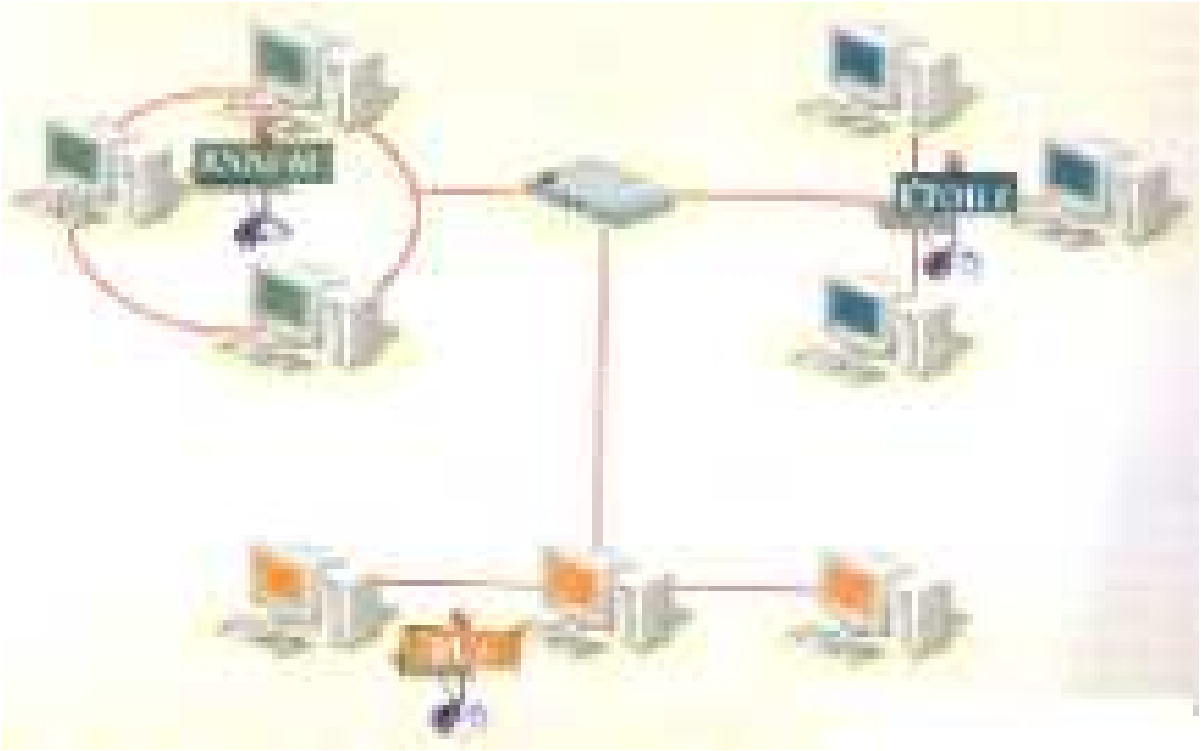
Topologie en anneau

- Les nœuds sont chargés de recevoir les informations en provenance de la station précédente et de les transmettre vers la station suivante.
- Nécessité d'une station de supervision qui puisse intervenir en cas d'erreur
- L'insertion d'un nouveau équipement nécessite la coupure de l 'anneau



Topologie hybride - arbre



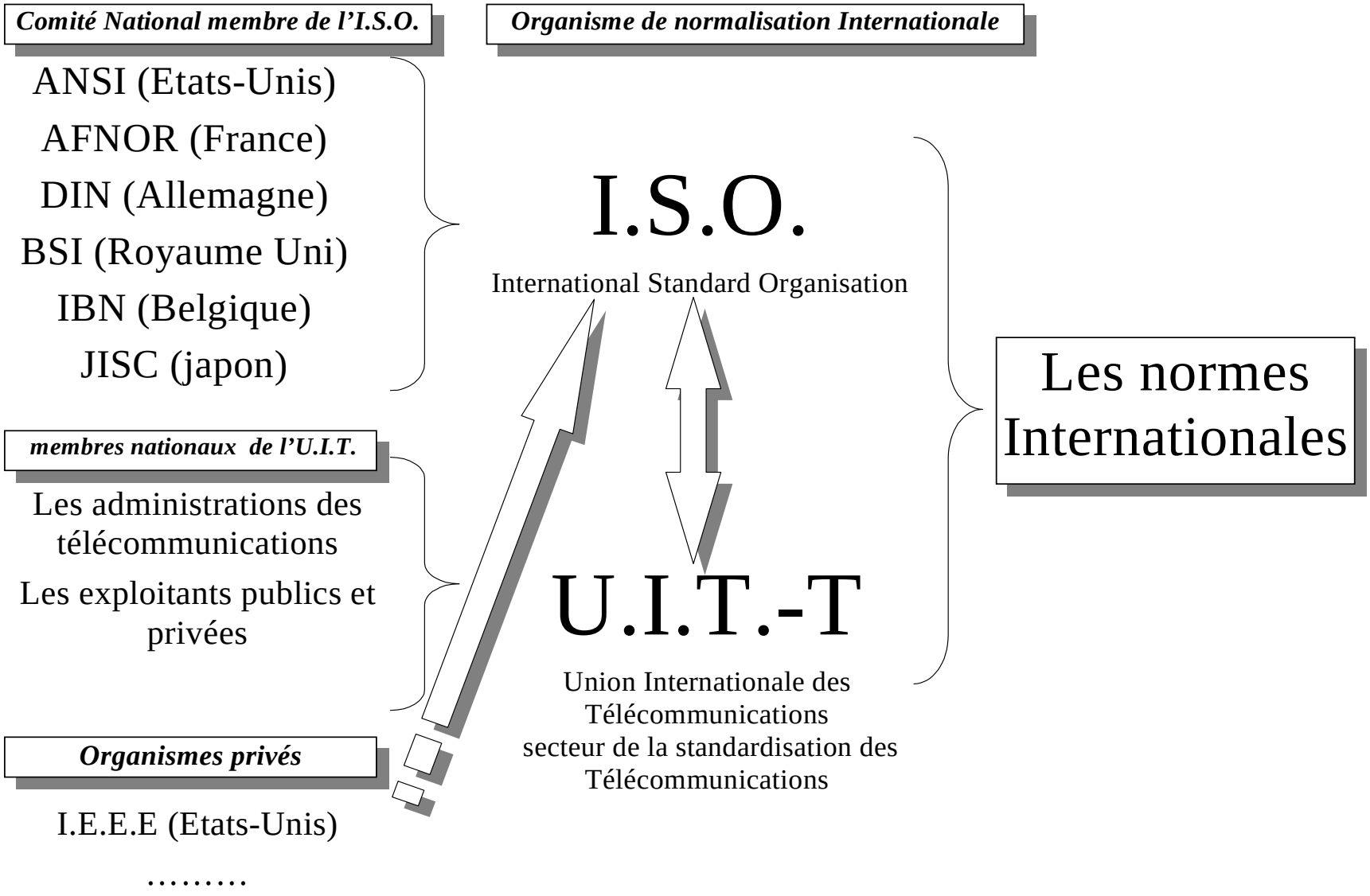


Mélange de topologies - Il est possible de rencontrer plusieurs topologies différentes dans un même réseau local

La normalisation

- Historique
  - Début des années 1970 : apparition de plusieurs réseaux privés indépendants (IBM, DEC, Bull,…)
  - Impossibilité d'interconnecter ces différents réseaux
- Pourquoi les normes
  - Elles permettent de se mettre d'accord sur l'ensemble des éléments nécessaires à la communication pour que les échanges puissent s'effectuer.
  - Besoin de définir des protocoles normalisés ou standardisés afin que seule l 'implémentation des protocoles change
- Comment se fait les normes ?
  - Un ensemble de constructeurs s'est mis d 'accord sur des règles communes.
  - Un constructeur a su imposer aux autres un protocole plus performant ou tout simplement qu'il est le seul à satisfaire la demande des utilisateurs.

Les organismes de normalisation

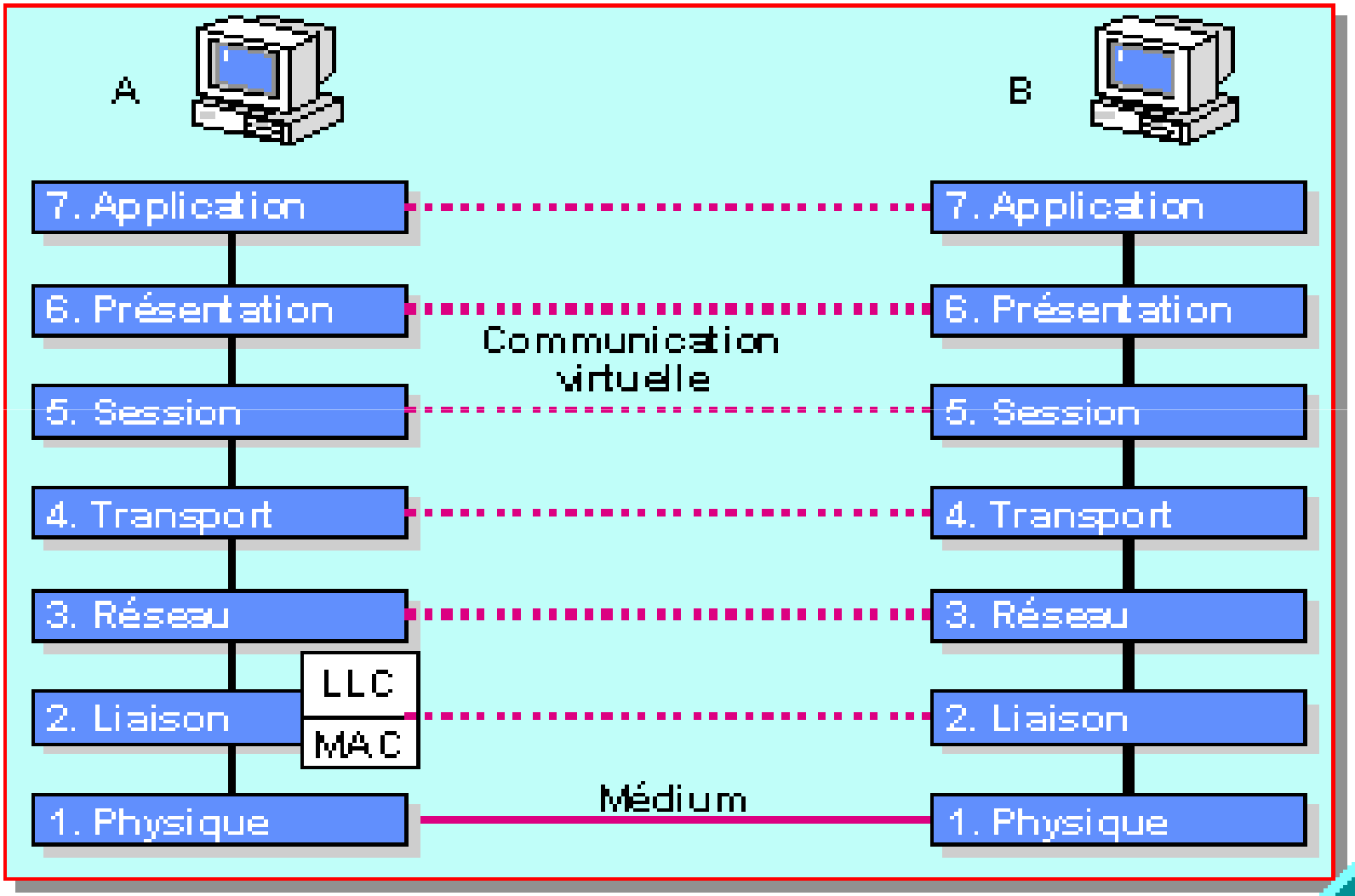


Le modèle O.S.I. de l'I.S.O.

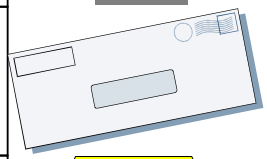
- Pour réaliser le transport d 'information de l'extrémité à une autre du réseau, il faut :
  - Un support de communication (physique ou hertzien)
  - Une structure normalisée des systèmes de communication
- 1977 : ISO démarre une réflexion sur une architecture de réseau en couches,
- 1983 : définition du modèle OSI
  - Open : systèmes ouverts à la communication avec d'autres systèmes
  - Systems : ensemble des moyens informatiques (matériel et logiciel) contribuant au traitement et au transfert de l'information
  - Interconnection

- Le modèle OSI permet
  - De définir un modèle de toute architecture de réseau
  - L'interconnexion de réseaux d'architecture différente
- Le modèle O.S.I. découpe les systèmes de communication en 7 couches et précise le fonctionnement de chacune de ces couches.
  - garantit que 2 systèmes hétérogènes pourront communiquer si :
    - même ensemble de fonctions de communication,
    - fonctions organisées dans le même ensemble de couches,
    - les couches paires partagent le même protocole.
- Pour ce modèle on a plusieurs dénominations :
  - Le modèle O.S.I.
  - Le modèle OSI de l'ISO
  - Norme de l'ISO
  - Norme du modèle OSI
  - Le modèle 7 couches de l 'I.S.O.

Le modèle OSI (Open Systems Interconnection)



Analogie avec l'acheminement du courrier

Couche 7 Application		*Le cadre écrit le contenu du courrier à envoyer*
Couche 6 Présentation		*La secrétaire dactylographie la lettre avec les conventions du courrier commercial*
Couche 5 Session		*La lettre comporte des indications (référence la situant dans la relation inter-entreprise)*
Couche 4 Transport		*Le service courrier de l'entreprise choisit le mode d'acheminement et prépare l'expédition*
Couche 3 Réseau		*La poste se charge d'acheminer la lettre à son destinataire*
Couche 2 Liaison		*Les lettres sont groupées en sacs suivant leur destination*
Couche 1 Physique		*Différents moyens de transports sont utilisés pour l'acheminement les sacs postaux*

Le modèle O.S.I.

Couches hautes

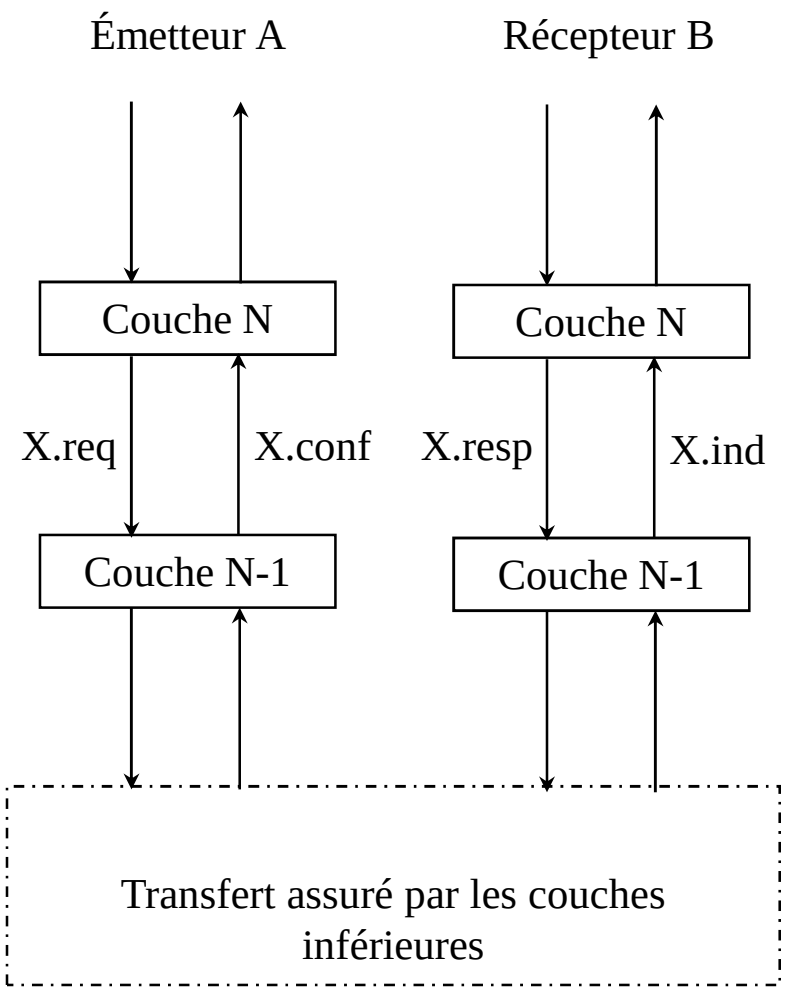
Permettent de mettre en forme l'information et fournissent des services d'accès à la communication

Couches basses

Permettent le transfert à travers le réseau de l'information provenant des couches supérieures

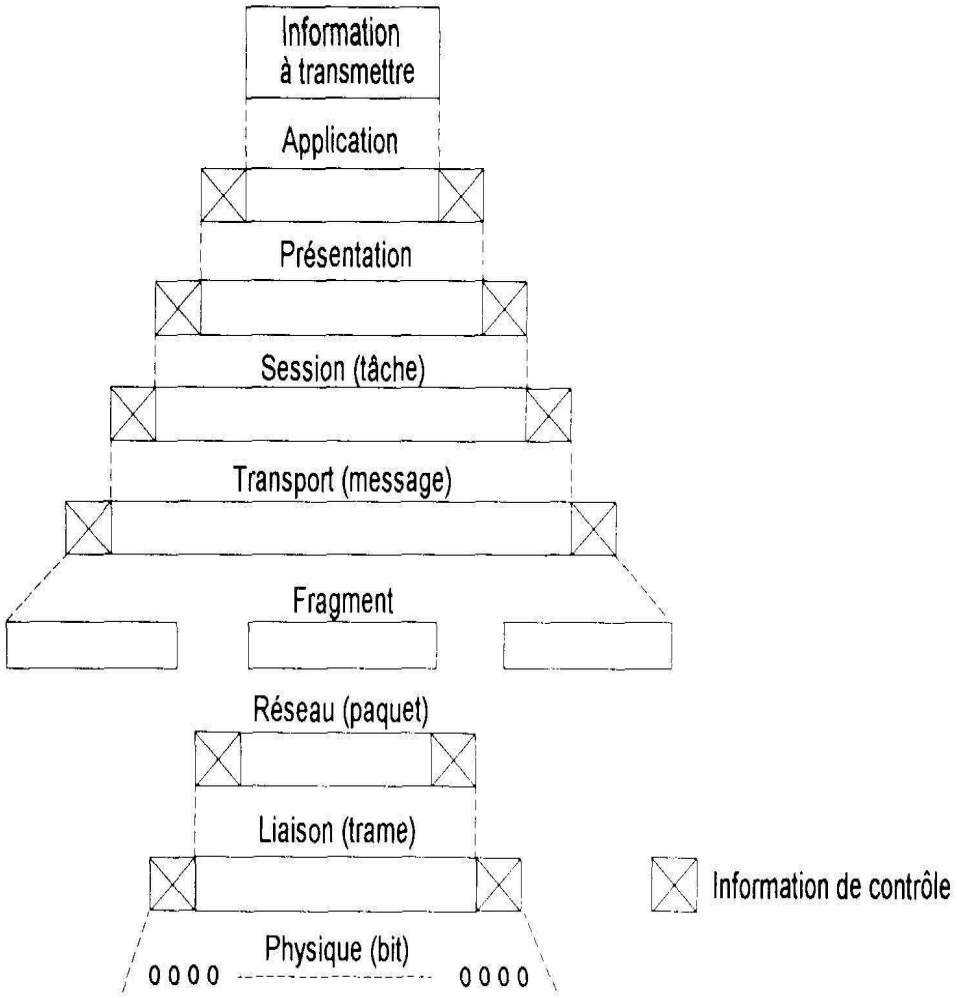
Couche 7	Application
Couche 6	Présentation
Couche 5	Session
Couche 4	Transport
Couche 3	Réseau
Couche 2	Liaison
Couche 1	Physique

Les 4 primitives



- **X.request** (requète) : demande d'un service X de la couche N à la couche N-1
- **X.indication** : réception des données du service X de la couche N-1 à la couche N
- **X.response** (réponse) : réponse à la primitive X.indication de la couche N à la couche N-1
- **X.confirmation** : confirmation de la primitive X.request de la couche N-1 à la couche N

De la couche 7 à la couche 1



Encapsulation des données

- A chaque couche correspond une unité de données (bit, trame, paquet, datagramme, fragment, segment, message)
- Le passage d'une couche à l'autre se fait par ajout d'informations de contrôle

Couche 1: La couche Physique

- Gère la transmission des bits de façon brute sur un lien physique, sans en connaître la signification ou la structure
- Réalise le transfert physique des éléments binaires (bits) entre équipements distants
- Respecte les caractéristiques physiques, électriques et mécaniques définies par des normes
- Génère les signaux pour établir, maintenir et libérer la connexion entre les équipements distants
- Synchronise le transfert d'une transmission en série ou en parallèle
- Génère un signal émis (portée, puissance, modulation, multiplexage,…) sur un support (câble électrique, fibre optique,…)
- Elle gère la transmission de séquences de bits par un support de communication, sans traitement explicite d 'erreurs.

Couche 2: La couche Liaison de données

- But : transformer un moyen brut de transmission en une liaison de données qui paraît exempte d'erreur de transmission à la couche supérieure
- Achemine les données reçues de la couche supérieure en les organisant en blocs de transmission
- Elle assure :
  - la transmission de l 'information
  - l 'assemblage des données en blocs
  - La synchronisation des blocs
  - La détection et correction des erreurs de transfert
  - Gère l'établissement, le maintient et la libération des connexions
  - Supervise le fonctionnement de la transmission
  - Le protocole de la couche liaison de données définit la structure syntaxique des trames et la manière d'organiser et d'enchaîner les échanges

Couche 3 : La couche Réseau

- But : Acheminer les données du système source au système destination quelle que soit la topologie du réseau de communication entre les 2 systèmes terminaux,
- Plus basse couche concernée par la transmission de bout en bout,
- Réalise pour les couches supérieures le transfert de données quelque soit la topologie du réseau,
- Assure le routage (acheminement) des paquets via des routes,
  - Les paquets en provenance d'un émetteur donné et à destination d'un récepteur donné peuvent emprunter le même chemin ou non : routage (choix des chemins à partir des adresses)
- Gère les problèmes d'adressage dans l'interconnexion de réseaux hétérogènes,
- Complexité de la couche dépendante de la topologie du réseau.
- Assure la détection et la correction des erreurs non réglées par la couche 2

Couche 4: La couche Transport

- But : Offrir aux couches supérieures un canal de transport de données de bout en bout fiable et économique quelle que soit la nature du réseau sous-jacent
- Sert d'intermédiaire entre les couches orientées traitement et les couches orientées communication
- Garantit une livraison ordonnée des messages
- Assure le découpage des messages trop longs en paquets
- Assure le rassemblement des paquets pour reformer le message
- canal économique :
  - débit rapide : une communication transport sur plusieurs connexions réseau,
  - réseau coûteux : multiplexage de plusieurs connexions transport sur une seule connexion réseau,

Couche 5: La couche Session

- But : Gérer le dialogue entre 2 applications distantes
- Fiabilité assurée par les couches inférieures,
- Gestion du dialogue :
  - Dialogue unidirectionnel ou bidirectionnel,
  - Gestion du tour de parole,
  - Ouverture et fermeture des sessions entre utilisateurs
  - Fournit des outils de synchronisation du dialogue entre les 2 applications (interruptions du transfert, reprise,…)
  - Elle assure l'établissement et le contrôle de séances de communication : contrôle des accès.
- Mécanisme de points de reprise en cas d'interruption dans le transfert d'informations

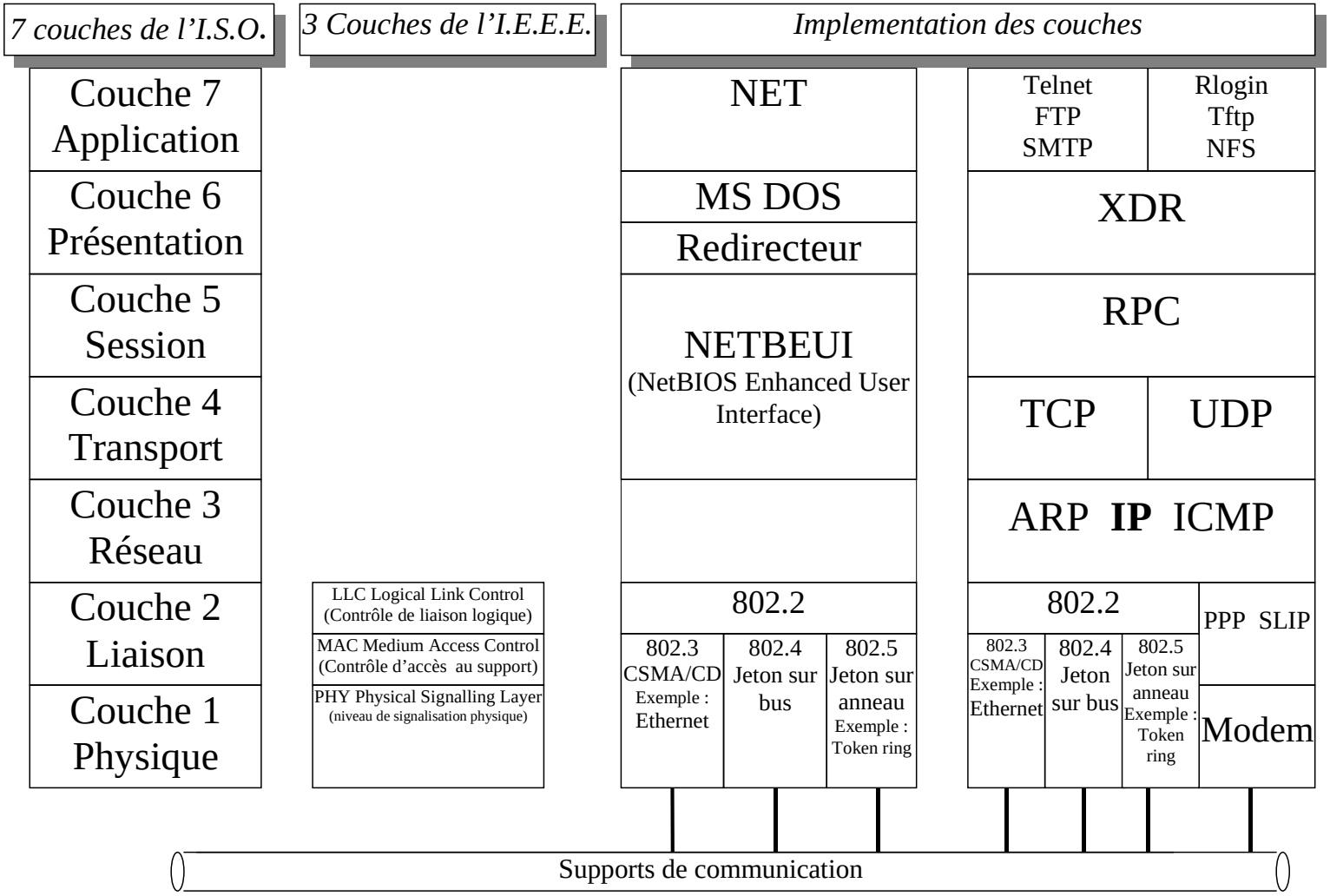
Couche 6: La couche Présentation

- But : Affranchir les applications de la couche supérieure des contraintes syntaxiques
- Effectue la compression des données si elle est nécessaire pour le réseau,
- S'occupe du chiffrement des données et de l'authentification.
  - Comporte des fonctions de traduction, de compression, d'encodage ASCII, de conversion, de cryptage
  - Codage des nombres entiers, réels, caractères, chaînes de caractères,…, objets,…
- Gère les problèmes de différences de représentation des données,
  - Inclue les codes de syntaxe et de présentation des informations
  - Elle convertit les données et les codes au format du destinataire (cryptage, compression de données..)
  - Met en forme les informations pour les rendre compatibles à l'ensemble des systèmes hétérogènes : représentation des données

Couche 7: La couche Application

- But : Fournir des applications réseaux normalisées.
- Est l'interface logiciel entre la machine et l'utilisateur
- Contient les fonctions nécessaires et donne accès à la communication entre systèmes ouverts
- Est chargée de l'exécution de l'application et du dialogue avec la couche 7 du processus distant
- Fournir des protocoles normalisés d'applications réseaux :
  - terminal virtuel,
  - transfert de fichiers,
  - messagerie électronique,
  - gestion et administration de réseaux,
  - consultation de serveurs et de bases de données.

- Modèle O.S.I. dans les réseaux locaux



Les Normes I.E.E.E

- IEEE (Institute of Electrical and Electronics Engineers)
- Nous allons nous concentrer sur les normes de l'IEEE qui est un des organismes les plus actifs dans le domaine des réseaux locaux
- Voici en résumé ces différentes normes

802.1 GESTION DE RESEAU

décrit les relations entre les normes ci-dessous

802.2 LIEN LOGIQUE (LLC)

définit la partie LLC (Logical Link Control) de la couche 2

802.3 RESEAU CSMA/CD

pour les réseaux à topologie bus et méthode d'accès CSMA/CD

802.4 RESEAU TOKEN BUS

pour les réseaux à topologie bus avec méthode d'accès à jeton

802.5 RESEAU TOKEN RING

réseaux en anneau avec méthode d'accès à jeton.

802.6 METROPOLITAN AREA NETWORK

réseaux à l'échelle d'une ville.

802.7 TRANSMISSION LARGE BANDE

c'est une norme qui se base sur les réseaux 802.3 et 802.4.

802.8 RESEAUX FIBRE OPTIQUE

802.9 VOIX + DONNEES

concerne l'utilisation d'un seul support physique pour transporter la voix et les données.

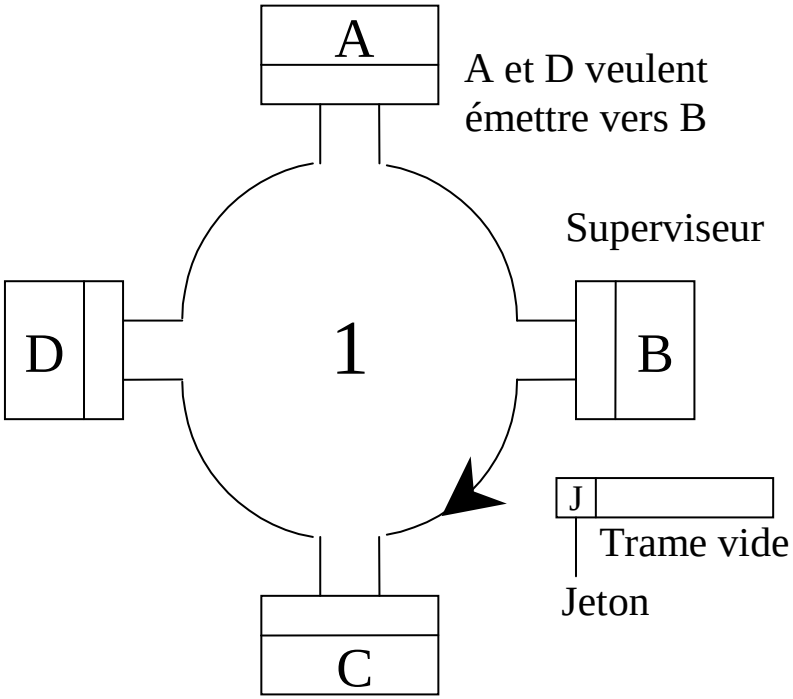
802.10 SECURITE DES RESEAUX LOCAUX

étudie les problèmes de sécurité dans les réseaux.

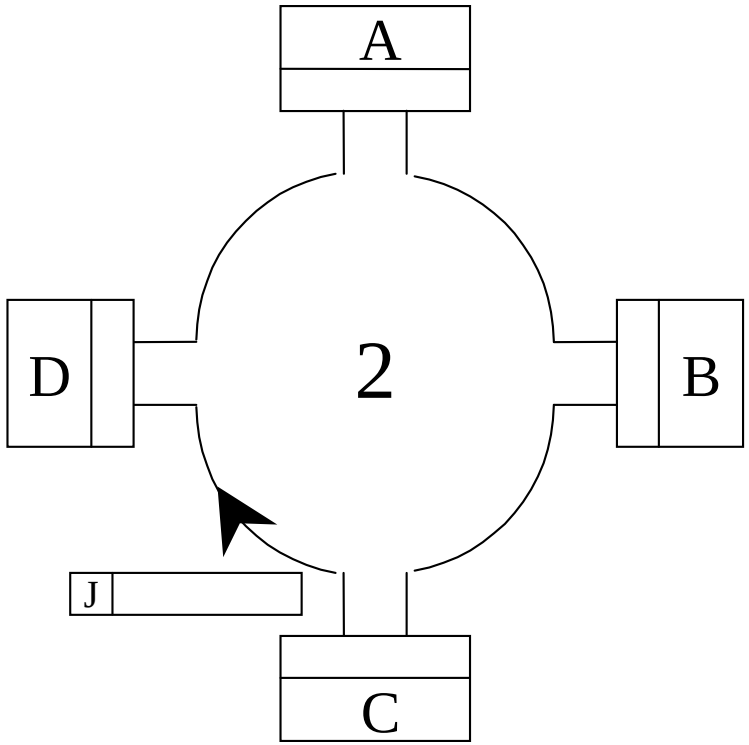
802.11 RESEAUX LOCAUX SANS FIL

transmission infrarouges, micro-ondes, ondes hertziennes, etc.

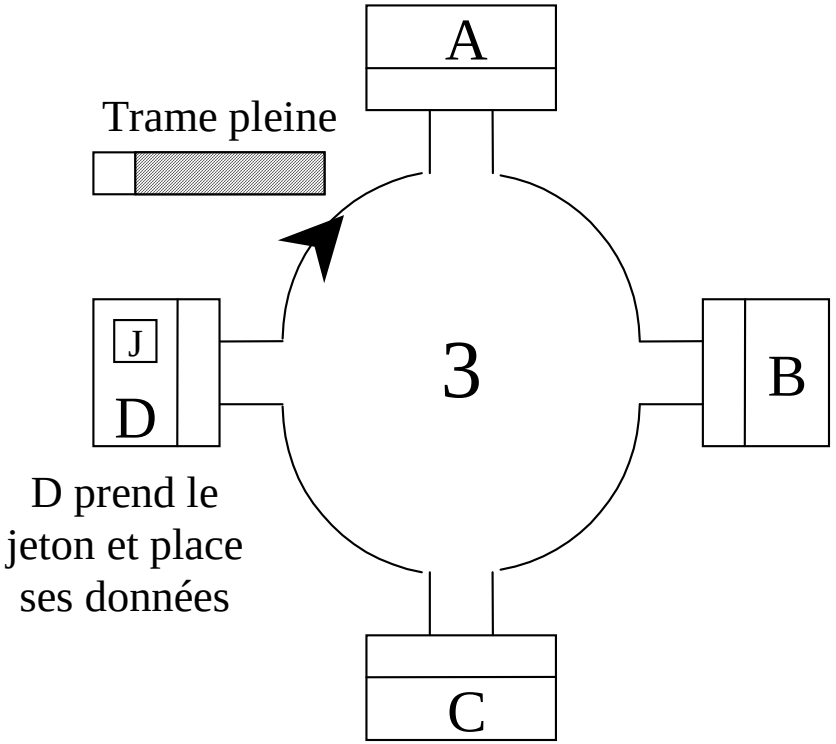
La méthode d’accès à jeton



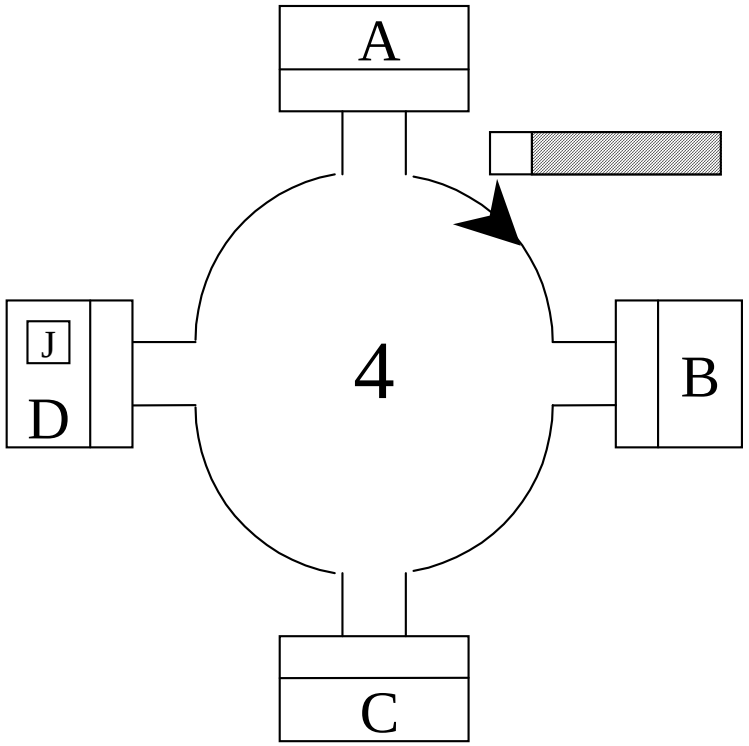
La méthode d’accès à jeton



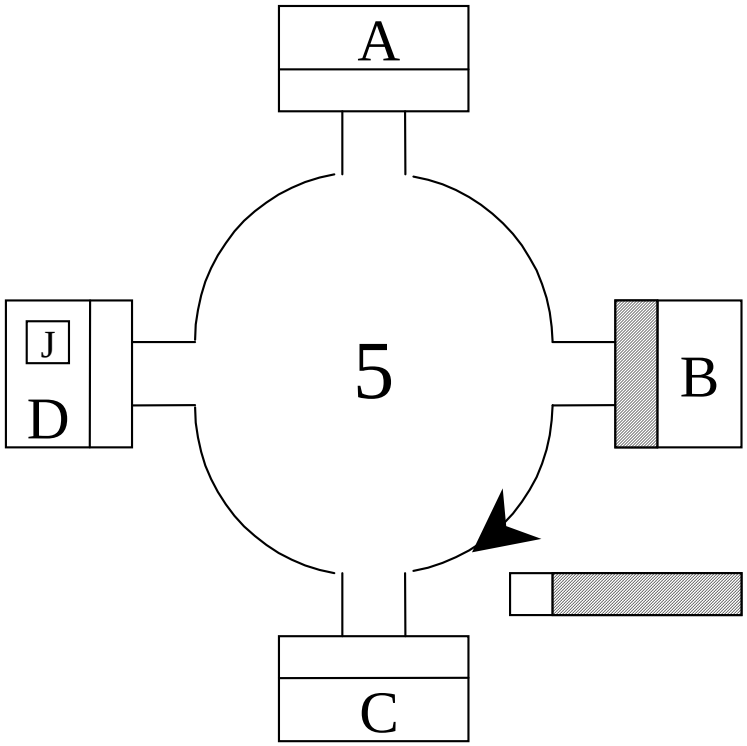
La méthode d’accès à jeton



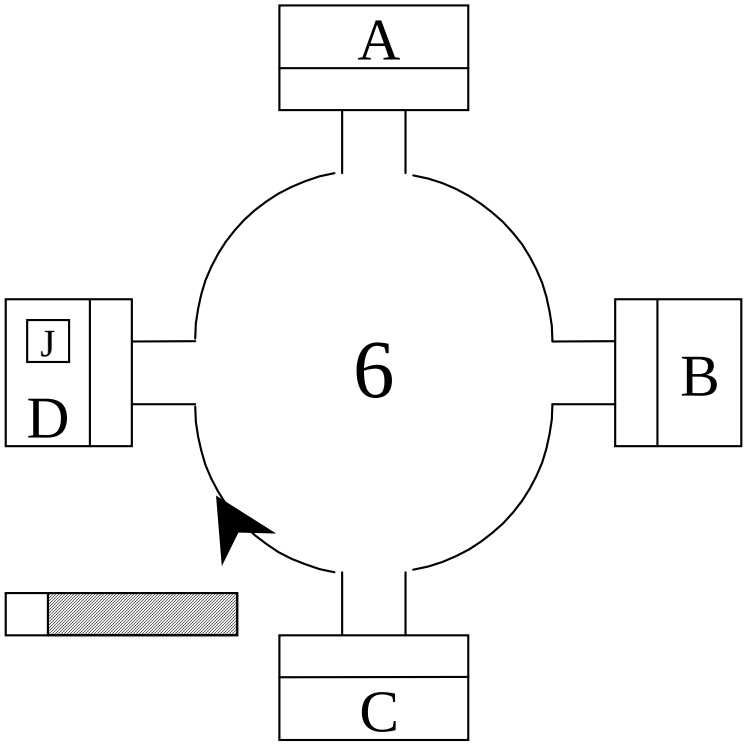
La méthode d’accès à jeton



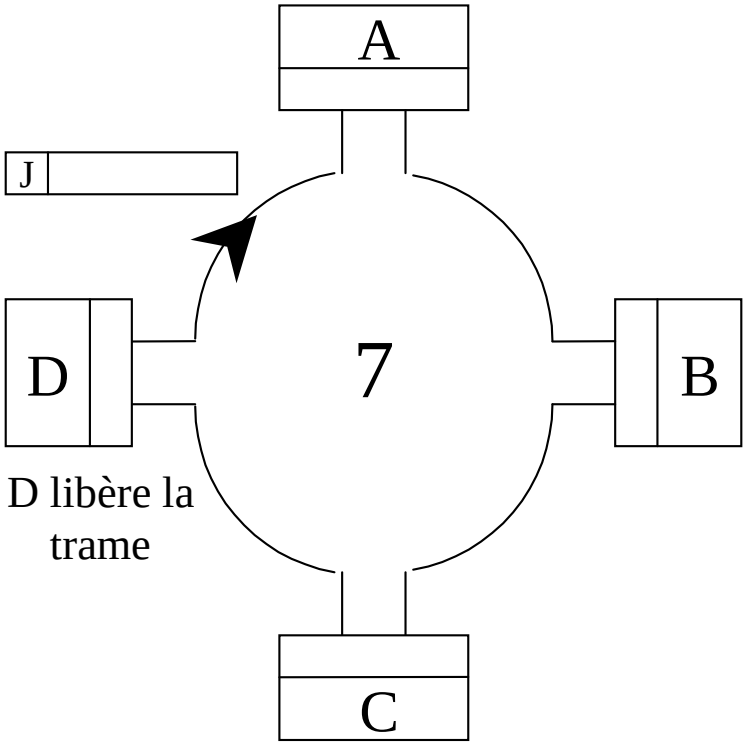
La méthode d’accès à jeton



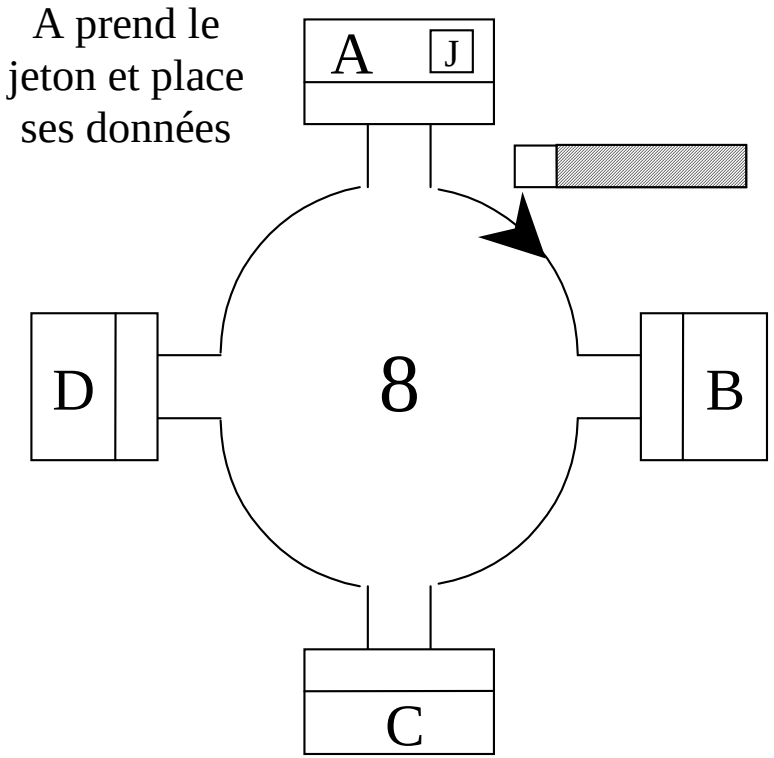
La méthode d’accès à jeton



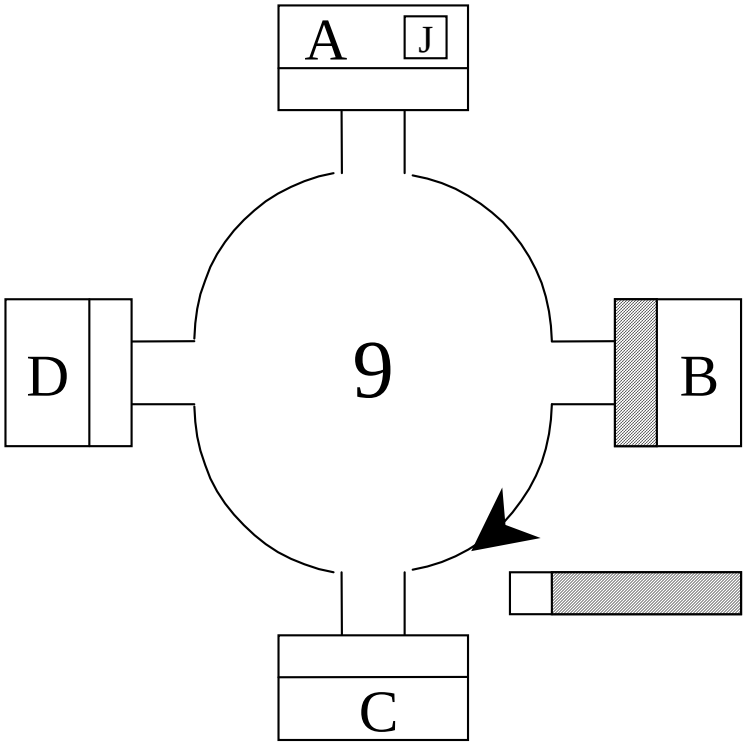
La méthode d’accès à jeton



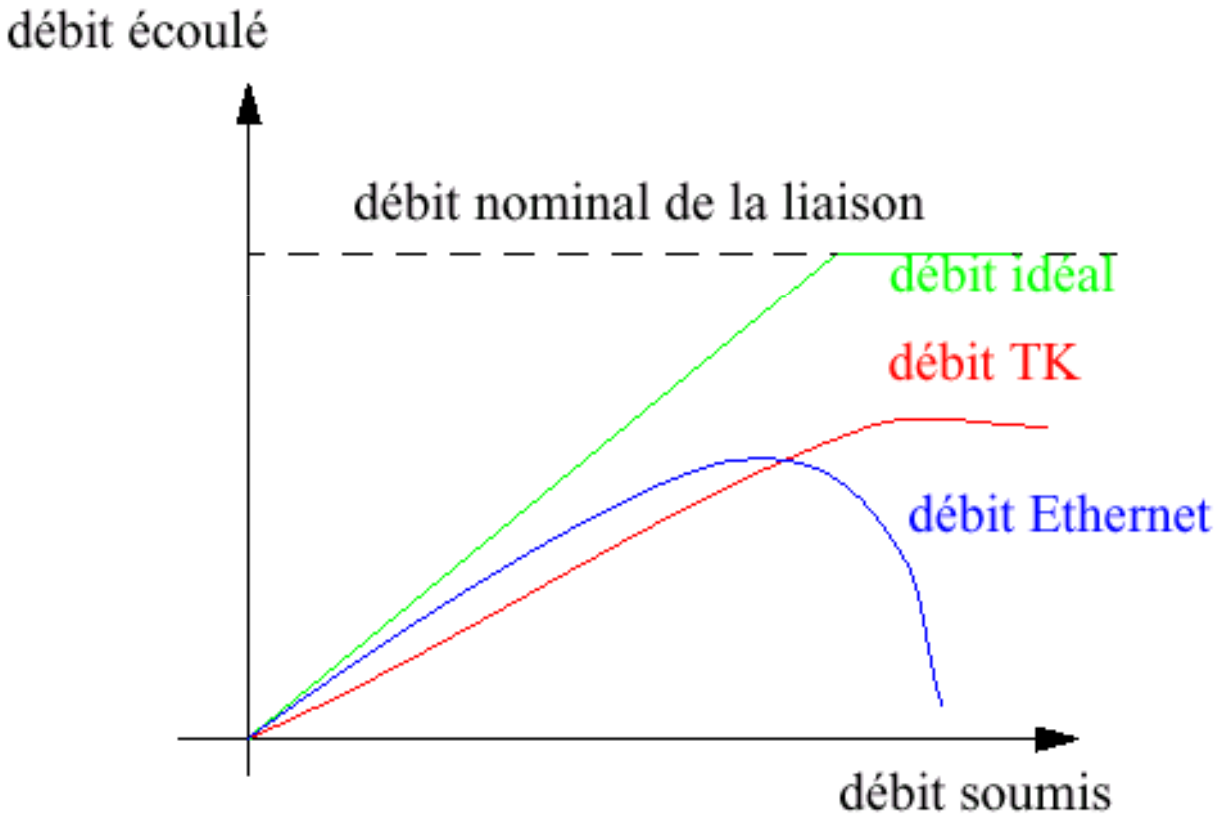
La méthode d’accès à jeton



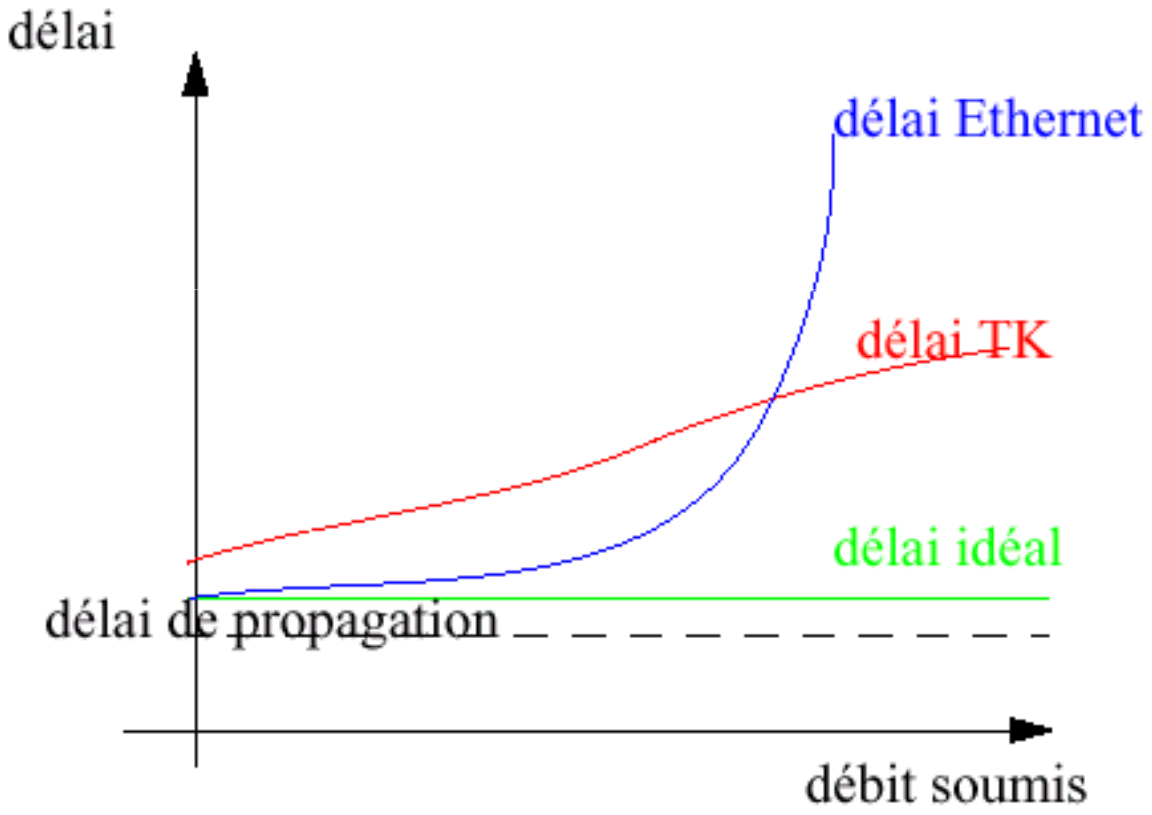
La méthode d’accès à jeton



Comparaison Ethernet/Token Ring (TK)



Comparaison Ethernet/ Token Ring (TK)



Aperçu sur les composants d’un réseau

- les différents média de communication
- l'adaptateur réseau (coupleur)
- le modem (modulateur-démodulateur)
- le répéteur (repeater)
- les concentrateurs (hub) - raccordement en étoile d'un réseau
- les commutateurs (switch)
- les ponts (bridge)
- les routeurs (router et b(ridge)-router)
- le serveur proxy
- le pare-feu (firewall)

Les différents médias de communication

- Le câble cuivre
  - Coaxial
  - à paires torsadées, parfois écrantées ou blindées

- La fibre optique
  - à base de verre ou de plastique
  - mono-mode ou multi-mode

- Les ondes radioélectriques et les rayons infrarouges

  < 300 Ghz

  de 10 Khz à 500 Mhz : radio – télévision

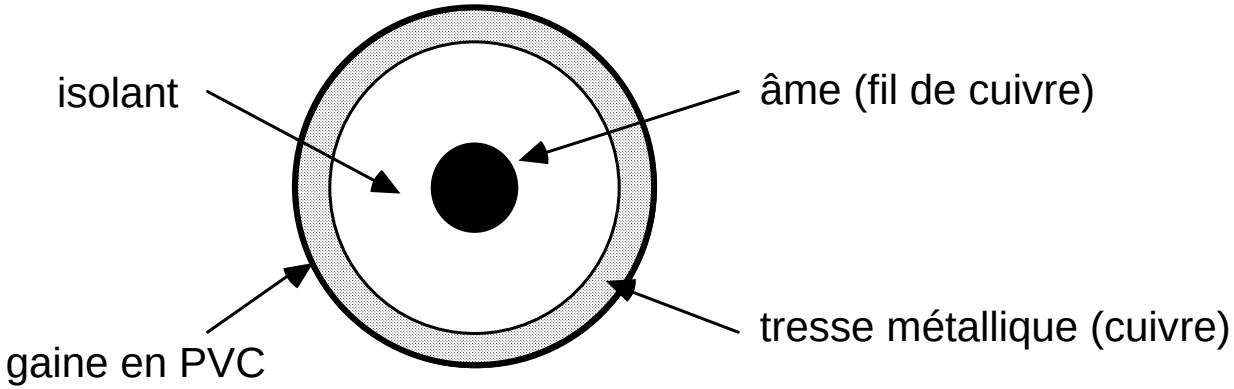
  de 500 Mhz à 40 Ghz : téléphonie mobile avec relais ou par satellite

  > 40 Ghz : limitée à une pièce

  > 300 Ghz (infrarouge)

Les câbles coaxiaux

Le câble coaxial est constitué de deux conducteurs cylindriques concentriques séparés par un isolant.



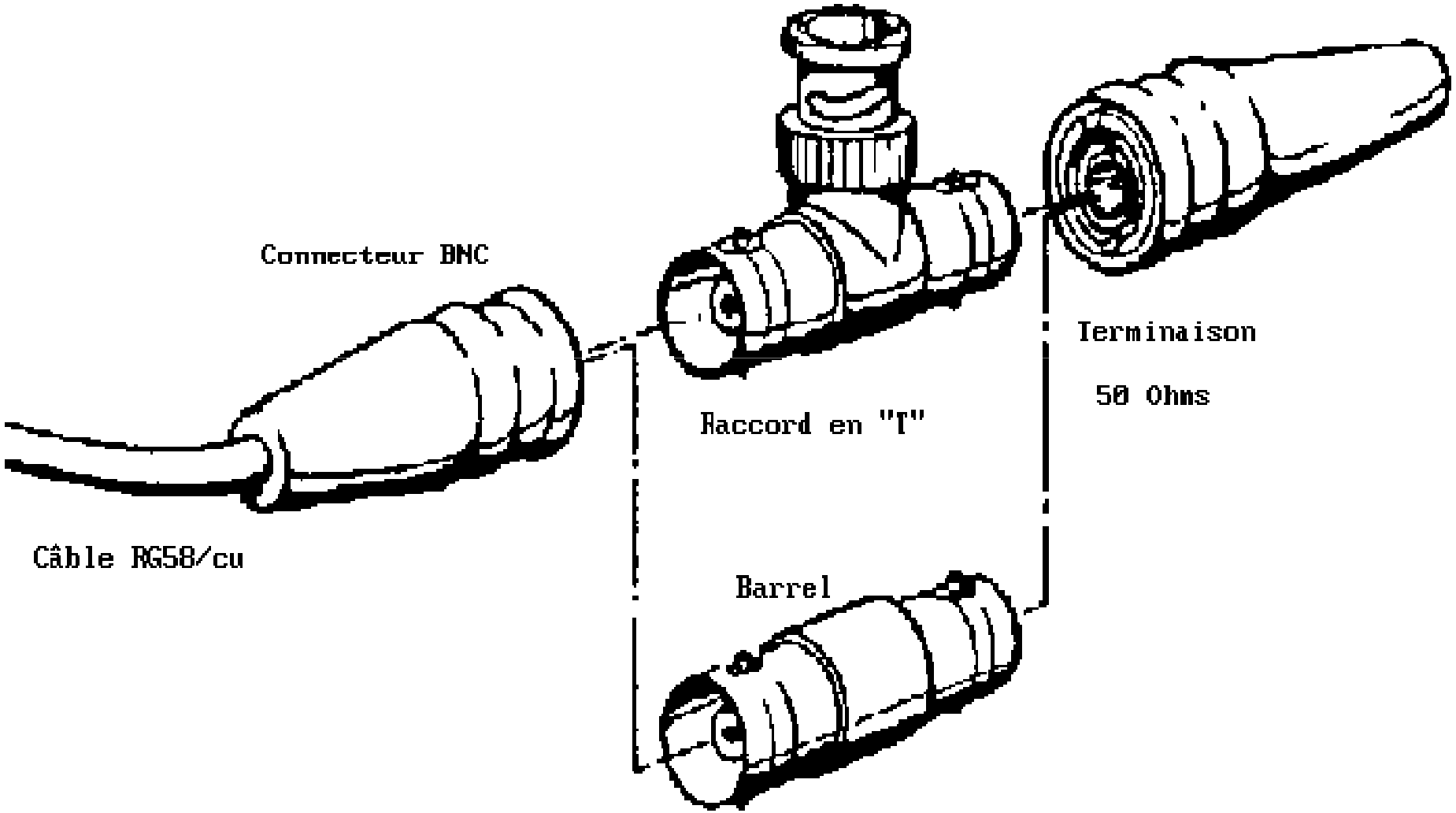
On distingue :

Les câbles utilisés avec un codage en bande de base pour les transmissions numériques : un seul signal (0: pas de courant, 1:présence de courant)

Les câbles utilisés avec un codage large bande pour les transmissions analogiques : plusieurs signaux analogiques peuvent être véhiculés simultanément à des fréquences différentes.

Inconvénient : difficile à installer, difficile à modifier

Thin ETHERNET



Le câble "Thin Ethernet" appelé aussi 10 Base 2 est utilisé pour la distribution d'Ethernet dans les bureaux ou groupes de travail, en reliant en chaîne des ordinateurs connectés à un Hub ou un répéteur.

Caractéristiques d'un segment "Thin Ethernet":

- Câble 50 Ohms RG 58 /CU
- Maximum 185m par segment
- 30 connexions max par segment
- 1 mètre min. entre chaque "noeud"
- Chaque segment doit être terminé par une résistance de 50 Ohms à chaque extrémité

On peut mesurer la qualité d'un câble à l'aide d'un Réflectomètre.

Les câbles à paires torsadées

- Chaque paire est entourée d'une gaine de couleur particulière. Les interférences électriques (diaphonie) entre les deux fils sont minimisées par le fait qu'elles sont torsadées.

- Débit possible pouvant atteindre 155 Mbits/s à une fréquence de 100 Mhz (en catégorie5) sur des distances courtes (90 m).

Câbles multipaires

Les paires torsadées sont généralement assemblées en câbles multipaires (2, 4, 6, 8, 14, 24, 28, 56, 112, 224, ...) protégés par une gaine PVC.

L'écrantage

- C'est une simplification du blindage.
- L'écran est constitué d'une fine feuille d'aluminium et de polyester qui protège la paire torsadée contre les perturbations radioélectriques de fréquences supérieures à 1 Mhz.
- L'écran doit être raccordé à chaque extrémité de la terre.

Principes

- Sur le câble circulent des trames :
  - Suites d'éléments binaires (trains de bits)
- Trame contient l'adresse de l'émetteur et du destinataire

- Un coupleur (carte réseau) est à l'écoute de la totalité des trames qui circulent sur le câble
  - Si une trame lui est destinée :
    - Adresse destinataire = Sa propre adresse physique
    - Il la prend, la traite et la délivre à la couche supérieure
  - Sinon, le coupleur ne fait rien

- Une station qui veut émettre
  - Regarde si le câble est libre
    - Si oui, elle envoie sa trame
    - Si non elle attend que le câble soit libre
- Si 2 stations émettent ensemble, il y a collision

  Les 2 trames sont inexploitables

  Les 2 stations détectent la collision, elles ré-émettront leur trame ultérieurement

Qualités de câble

- les paires torsadées non blindées : UTP Unshielded Twisted Pair
- les paires torsadées blindées : STP Shielded Twisted Pair
- les paires torsadées avec écran : FTP Foiled Twisted Pair
- les paires tors. blindées avec écran: SFTP – Shielded Foiled Twisted Pair

Exemple de câble à paires torsadées

- câble à paires torsadées non blindées d'impédance caractéristique 100 ohms utilisé par AT&T dans son système de câblage PDS
  norme américaine de système de câblage : câble 24 AWG - 0,51mm

- câble à paires torsadées blindées 150 ohms utilisé par IBM dans son système de câblage ICS
  norme américaine de système de câblage : câble : 22 AWG - 0,64mm

- câble à paires torsadées écranté 120 ohms, câble L120 préconisé par France Télécom

Les catégories de câbles à paires torsadées

- Les catégories 1 et 2 : transport de la voix et des données à vitesses réduites

- La catégorie 3 : supporte des débits < 16 Mbits/s
utilisé pour le transport de la voix et des données à un débit < 10 Mbits/s

- La catégorie 4 : supporte des débits < 20 Mbits/s utilisé pour la transmission des données à un débit < 16 Mbits/s

- La catégorie 5 : supporte des débits ~ 100 Mbits/s

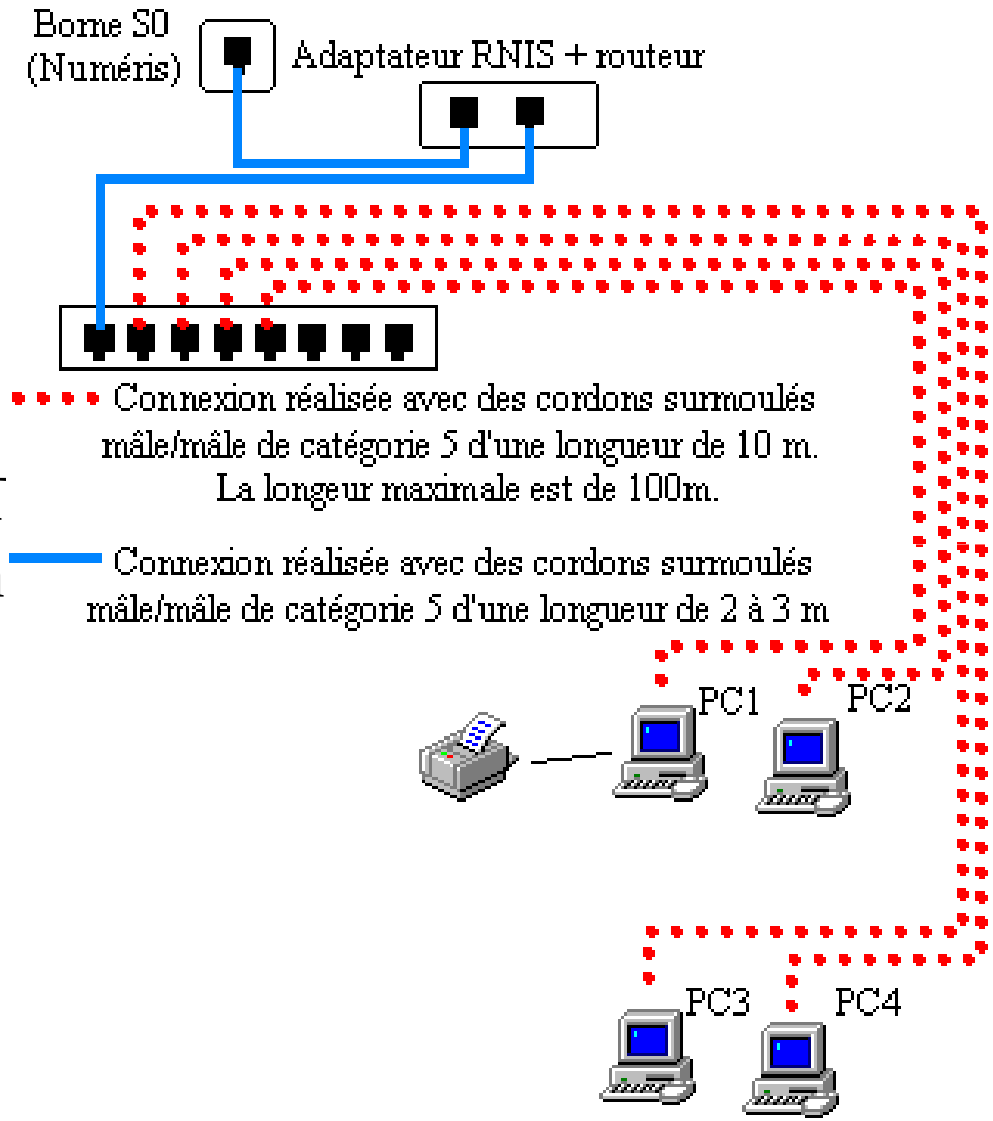
- La catégorie 6

- La catégorie 7

Installation en câbles volants

C'est la solution la plus économique. En effet, certains revendeurs fournissent des kits complets pour mettre en réseau plusieurs postes. Ces ensembles comprennent en général : les cartes réseau; les câbles 10baseT de 10m avec des prises RJ45; un concentrateur qu'il faudra choisir en fonction du nombre de postes qui seront connectés.

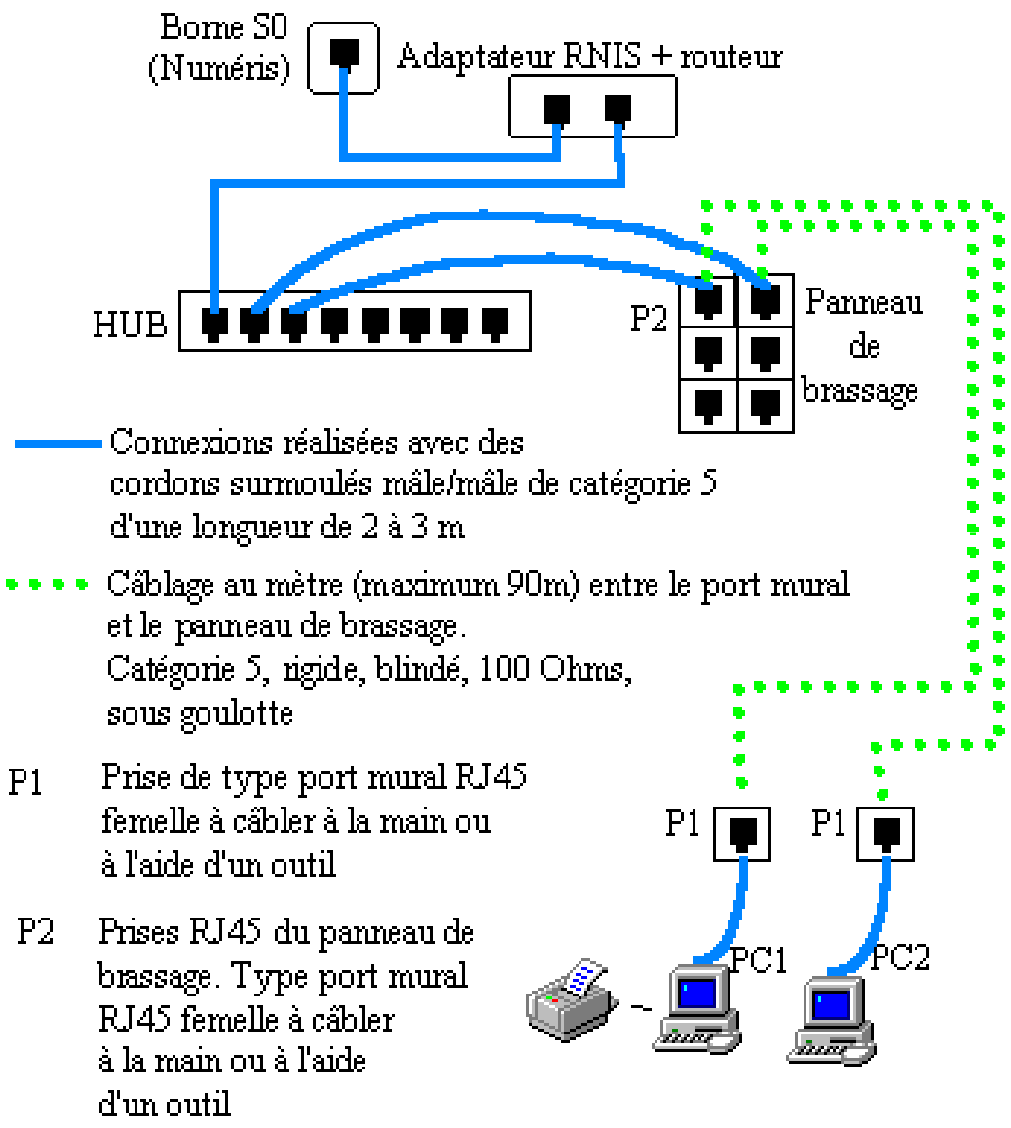
La faiblesse de cette installation réside dans la vulnérabilité des câbles, qui, non cachés, peuvent être arrachés.



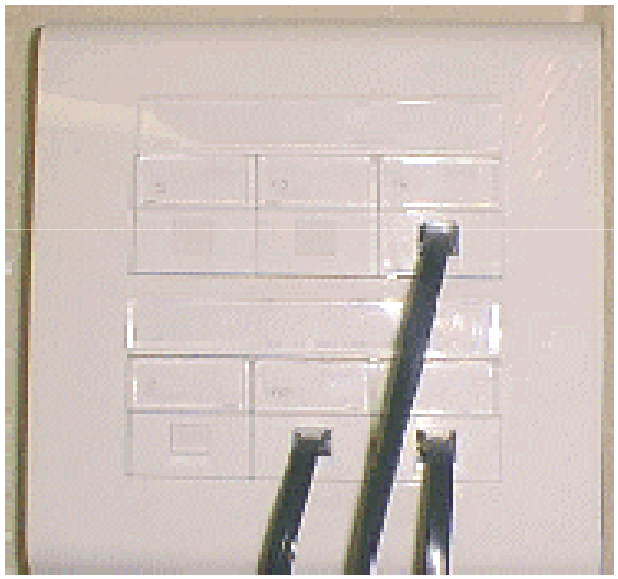
Installation plus solide

Contrairement à l'installation en câbles volants (fils souples), la partie filaire du réseau sera encastrée (fils rigides) en goulottes et débouchera du côté ordinateur comme du côté concentrateur sur des prises femelles. Toutes ces prises sont prévues pour accepter un câblage manuel ou avec outil.

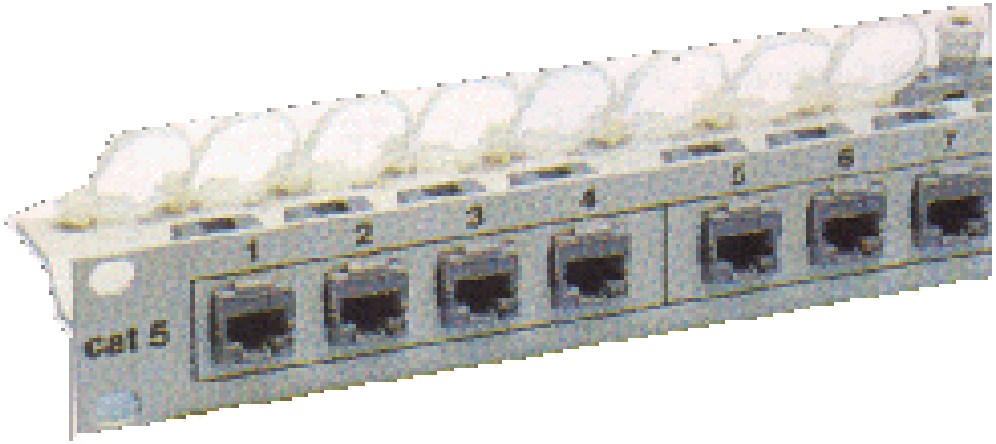
Cette solution présente l'avantage d'avoir des câbles cachés et solidement placés. Dans cette configuration, les câbles sont coupés à la longueur exacte plus vingt centimètres afin de permettre le montage et la maintenance des prises. Le repérage systématique de chaque prise (au départ et à l'arrivée) facilite le dépannage du réseau.



Prise femelle
(côté ordinateur)



Panneau de brassage (côté concentrateur) à monter



Panneau de brassage équipé

Câblage de prises RJ45

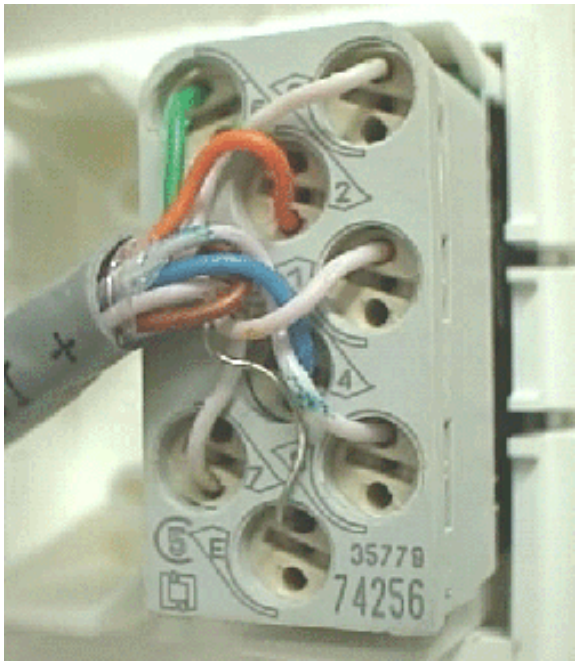
Tout le matériel doit être de catégorie 5 pour accepter par la suite des débits supérieurs

Câblage d'une prise

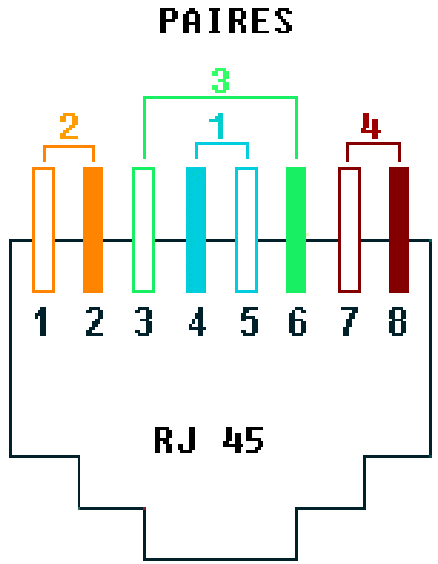
Enlever la gaine sur 20 mm au maximum. On ne doit pas détorsader sur plus de 13 mm, ni sur-torsader

Les fils des câbles sont torsadés par paire. Les couleurs de ces paires sont le plus souvent :

Paire 1 : bleu et blanc/bleu
Paire 2 : orange et blanc/orange
Paire 3 : vert et blanc/vert
Paire 4 : marron et blanc/marron



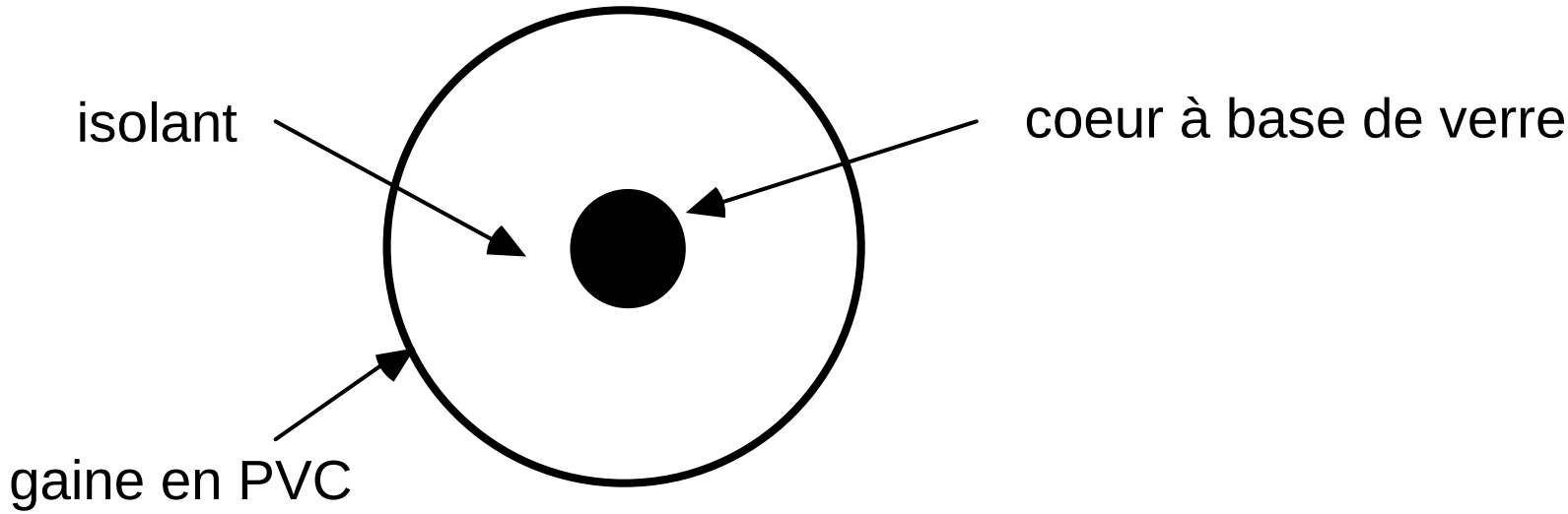
Exemple : la prise Legrand ci-dessus de référence 74256 est de catégorie 5.



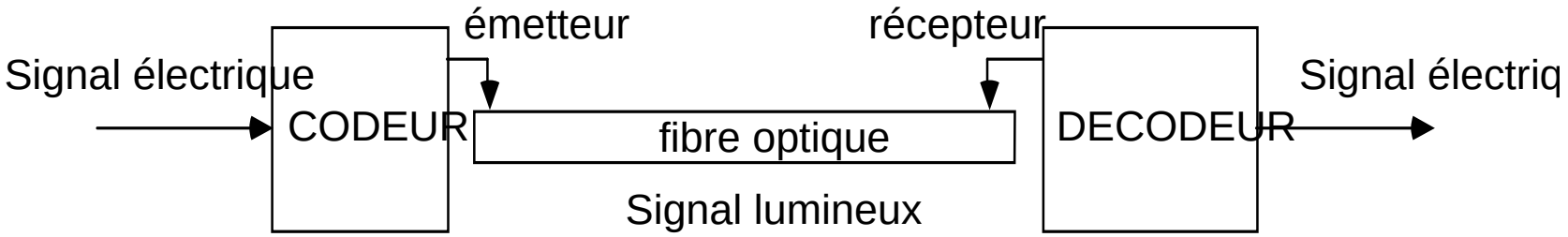
Le fil d'écran (blindage) se connecte sur la neuvième borne (E sur la photo)

La fibre optique

Un guide cylindrique de très faible diamètre à base de verre (ou de plastique) recouvert d'un isolant véhicule un signal lumineux.



Connectique



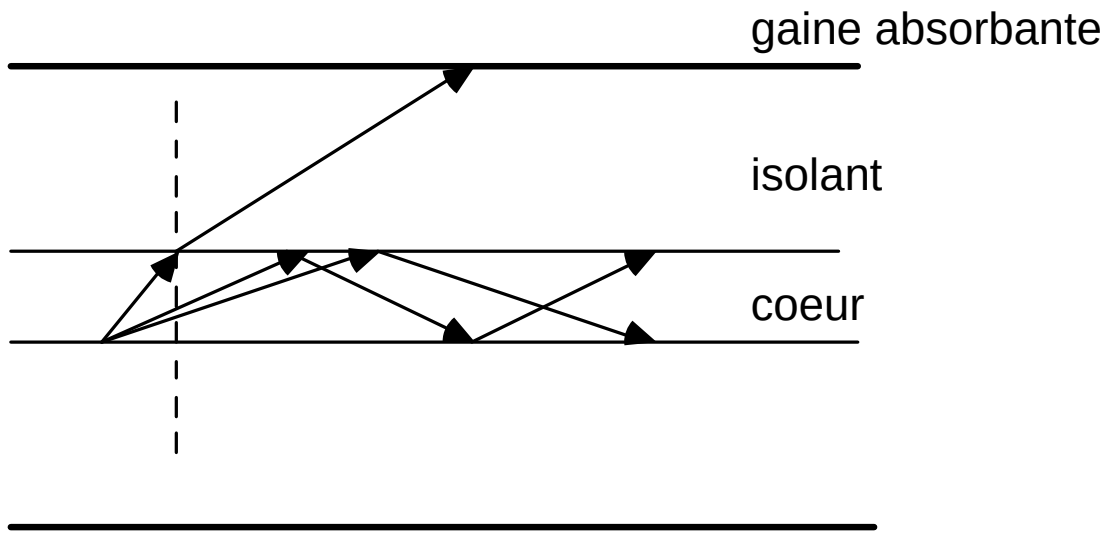
- Ce sont les composants de bout, émetteurs et récepteurs qui limitent les vitesses qui peuvent être atteintes sur les fibres.

- Les raccordements permanents sont (étaient) réalisés par épissurage (soudure, collage). Aujourd'hui il existe différents types de connecteurs.

- Les raccordements (provisoires) sont réalisés à l'aide de connecteurs autorisant de multiples connexions et déconnexions.

Principe de fonctionnement

Un rayon lumineux émis par une diode luminescente ou un laser se propage dans le cœur.



Au contact de l'isolant le rayon

- se réfracte si l'angle d'incidence est faible
- se réfléchit si l'angle d'incidence est fort

La quantité de lumière réfractée ou réfléchie est fonction de la valeur des indices de réfraction du cœur et de l'isolant.

Type de fibre optique

- les fibres multimodes à saut d'indice
  - indice de réfraction du cœur constant < indice de réfraction de l'isolant
  - diamètre du cœur : 100 µ, diamètre de l'isolant : 140 µ

- les fibres multimodes à gradient d'indice --> réseaux locaux
  - indice de réfraction du cœur décroît du cœur vers la gaine
  - diamètre du cœur : 62,5 µ, diamètre de l'isolant : 125 µ
  - diamètre du cœur : 50 µ, diamètre de l'isolant : 125 µ

- les fibres monomodes ---> réseaux des opérateurs longue distance
  - un seul rayon lumineux se propage
  - plus performante - plus difficile à mettre en œuvre (nécessite un laser)
  - indice de réfraction du cœur constant, mais très inférieur à celui de l'isolant
  - diamètre du cœur : 9,5 µ, diamètre de l'isolant : 125 µ

Avantage des fibres optiques

- très large bande passante => débits très élevés possibles
- faible volume, rayon de courbure < 1 cm => souplesse d'installation
- faible poids : 9 fois moins qu'un câble conventionnel
- immunité aux perturbations électromagnétiques
- sécurité : intrusion facile à détecter (affaiblissement à la réception)
- affaiblissement linéique très faible
- affaiblissement assez largement indépendant de la fréquence
- grande résistance mécanique (traction), à la chaleur, au froid, à la corrosion
- facilité de localisation des coupures par télémétrie

Inconvénients de la fibre optique ➔ ils diminuent progressivement

· coût élevé des équipements de bout

Les ondes radioélectriques et les rayons infrarouges

< 300 Ghz : ondes radioélectriques

de 10Khz à 500 Mhz: diffusion d'un émetteur vers des récepteurs dispersés

peu utilisée en transmission de données

de 500 Mhz à 40 Ghz : faisceaux hertziens

- transmission d'un émetteur vers un destinataire unique
- transmission à vue, avec relais (courbure de la terre)
- téléphonie mobile étendue (norme GSM 900 Mhz et DCS 1800 Mhz)
- téléphonie mobile rapprochée (norme CT2 et norme DECT)
- boucle locale radio (3,5 Ghz et 28 Ghz)

➢ transmission via satellite

2 types de réseau satellites

- géostationnaire à 36 000 km de la terre (induit un délai)
- en orbite basse LEOS (Low Earth Orbiting Satellites) entre 750 et 1500 km

>40 Ghz

- **utilisation de guides d'ondes pour compenser l'atténuation due aux gouttes d'eau contenues dans l'atmosphère,**
- ondes millimétriques dont l'utilisation est essentiellement limitée à une pièce.

<u>>300 Ghz : rayons infrarouges</u>

- utilisation limitées à une pièce en utilisant des diodes LED
- utilisation parfois en liaison inter-bâtiments, en utilisant un laser
- émergence des réseaux locaux sans fils, mais actuellement prix non compétitifs

L'adaptateur (carte) réseau - coupleur

• C'est le composant qui fait l'interface entre le poste de l'utilisateur et le médium de communication.

• Pour un PC on parle souvent de carte réseau, elle est insérée dans le micro-ordinateur. Il tend de plus en plus à être intégré directement sur la carte mère. Autrefois il était souvent externe à l'ordinateur et relié par un câble spécial.

• L'adaptateur réseau est fonction du type de médium de communication utilisée (couche physique) et du mode d'accès à la couche physique (sous-couche MAC de la couche de liaison). Certains adaptateurs peuvent implanter simultanément plusieurs normes de communication.(10 base 5, 10 base 2, 10 base T, ethernet 10/100 Mbits/s). Autres types de carte : Token Ring, FDDI, RNIS, ATM.

• **Un adaptateur peut être doté de plus ou moins d' "intelligence", c'est à dire de capacité de mettre en œuvre par lui-même un certain nombre de fonctions (démarrage d'une machine sans disque à partir d'une autre machine du réseau, support d'agent de gestion SNMP, ...)**

Le modem (modulateur-démodulateur)

Le modem est un adaptateur réseau. Il convertit un signal binaire en bande de base en signal analogique (et vice-versa) pour connecter une station au RTC (Réseau Téléphonique commuté).

- V21 : 300 bits/s
- V22bis : 2400 bits/s
- V32 : 9600 bits/s
- V32bis : 14,4 Kbits/s
- V 34 : 28,8 Kbits/s
- V42 bis : 38,4 Kbits/s
- V 90 : 56 Kbits/s dans un seul sens (fournisseur -> client)

Le répéteur

- Interconnexion de réseaux au niveau 1

- Permet d'augmenter la taille d'un réseau
- C'est un régénérateur de signal, équipement non "intelligent"
- Tout ce qui arrive sur un port est recopié sur les autres ports
- Le nombre de répéteurs dans un réseau est limité

- les deux segments de réseau doivent être de même type
  par exemple :
  2 segments ethernet
  2 segments token ring

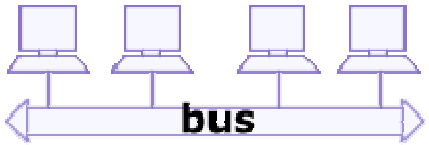
un répéteur répète le signal (avec ses bruits)
c'est une recopie bit par bit

Le hub

- C'est un concentrateur
- Il permet le raccordement en étoile des stations d'un réseau 10 base T.
- Son rôle est de répercuter tout ce qu'il entend sur un brin (port) sur tous les autres brins (ports).
- Il doit être alimenté.
- Le réseau physique est une étoile, le réseau logique est un bus.
- Il peut éventuellement être doté d'un module de gestion à distance, certains ports peuvent alors être inhibés.

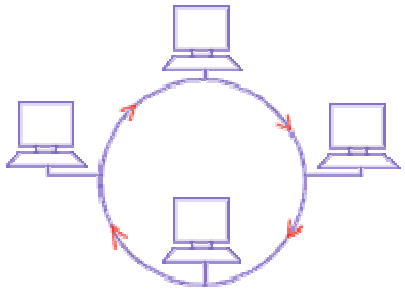
La topologie d'un réseau local - Les concentrateurs

la topologie en bus



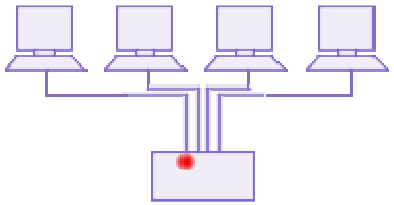
ethernet ➔ hub

la topologie en anneau



token ring ➔ MAU

la topologie en étoile



voir le site **http://www.commentcamarche.com/**

Le commutateur (switch)

Rappel : sur un réseau de type "ethernet", tout ce qui est émis par une station est reçu par toutes les stations du réseau; le rôle du hub est de répercuter tout ce qu'il entend sur un port sur tous les autres ports.

- Un commutateur identifie par leur adresse physique (adresse Mac) chacune des stations auxquelles il est raccordé. Il dispose d'une table (numéro de port, adresse physique).

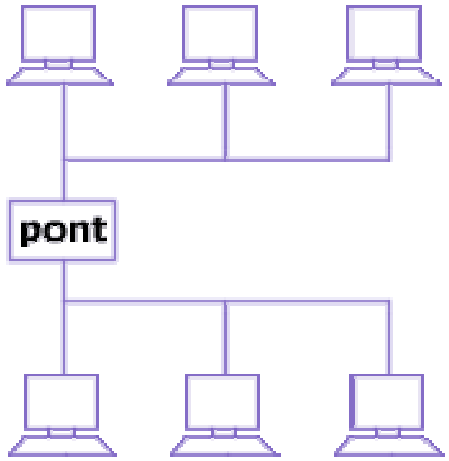
- Chaque trame qu'il reçoit sur un port est décodé au niveau Mac pour déterminer l'adresse physique de la station de destination. Il répercute alors le message reçu uniquement sur le port de la station de destination.

- Le commutateur permet de créer des réseaux locaux virtuels VLAN

Le pont (bridge)

Il permet d'interconnecter deux réseaux au niveau de la couche 2.

Chaque station a une adresse de niveau 2, appelée adresse Mac



- Les deux réseaux peuvent être de type différent
  par exemple :
  - un segment ethernet
  - un segment token ring

Le pont :

- reçoit sur un port toutes les trames qui circule sur un segment donné
- stocke ces trames
- et recopie toutes les trames à destination de l’autre segment sur le port correspondant

- C'est un organe "intelligent", car il joue le rôle d'un filtre. Il analyse les trames reçues sur un segment de réseau et ne recopie sur un autre segment que les trames destinées à ce segment ou à un segment accessible par l'intermédiaire de ce segment.

- Un pont doit être initialisé pour qu'il connaisse la structure du réseau. Cette initialisation peut être automatique, on parle de pont auto-apprenant (auto-adaptatif).

Le routeur

- Il permet l'interconnexion de réseaux au niveau de la couche 3 (réseau)

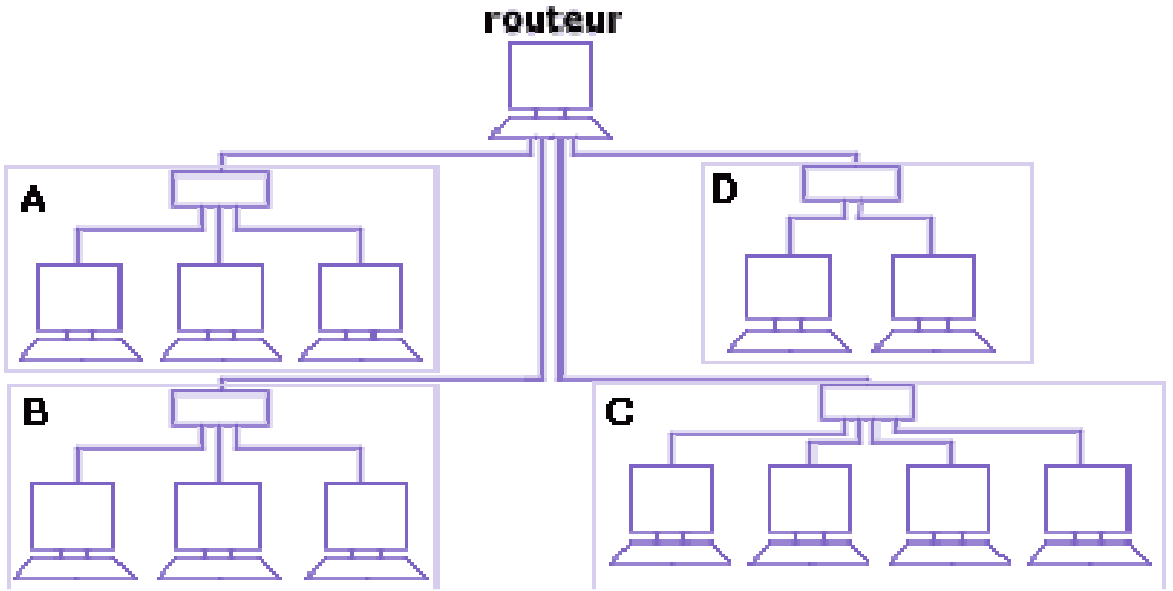
- Chaque station a une adresse de niveau 3, appelée adresse réseau

  Une station connectée à Internet a une adresse internet sous la forme de 4 octets.

  exemple : 130.66.73.77

  Mémoriser une adresse n'est pas simple, on préfère donner à chaque station un nom mnémonique et utiliser un serveur DNS pour établir la correspondance entre le nom mnémonique et l'adresse internet.

- Dans chaque paquet d'informations qui circule sur un réseau IP, on trouve l'adresse IP de destination du paquet. Le routeur est un composant du réseau qui permet de quitter le réseau local pour accéder à un autre réseau.



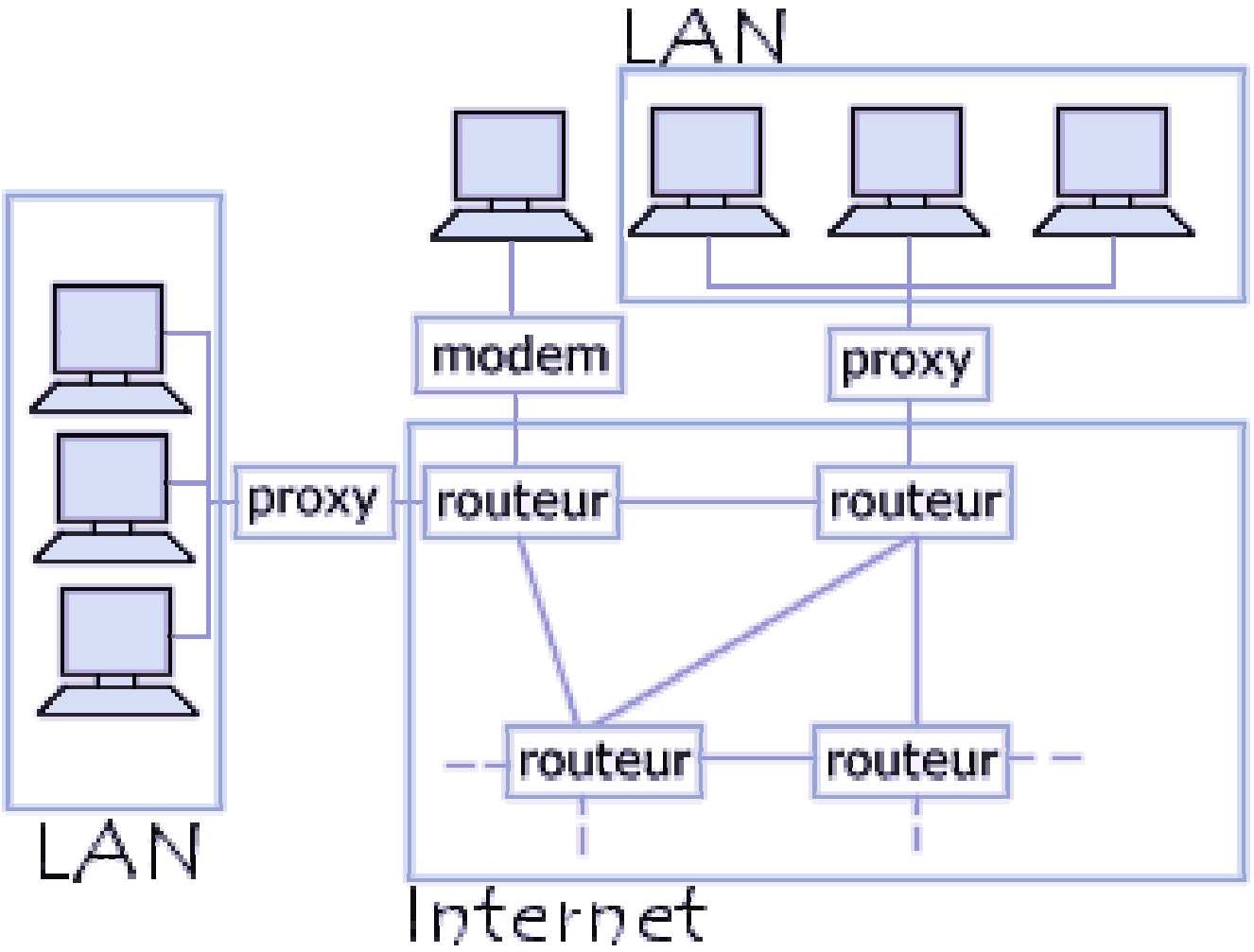
Le rôle du routeur est d'analyser l'adresse de destination des paquets qu'il reçoit et de choisir le meilleur chemin pour les conduire à destination.

Un routeur reçoit des paquets conformes à un protocole couche 3 donné et le réexpédie dans une enveloppe conforme à un autre protocole couche 3 (qui peut être le même ou non que celui du paquet reçu).

Les b-routeur (bridge-router) pont amélioré

Dispositif qui joue le rôle de routeur pour certaines trames reçus conforme à un certain protocole donné, et le rôle de pont pour les trames reçus qui ne sont pas conforme à ce protocole.

Serveur proxy : proxy-cache et firewall (pare-feu)



Machine intermédiaire entre les ordinateurs d'un réseau local et le web

Les serveurs proxy ont deux fonctions principales :

- Une fonction de cache mémoire

  - garde en mémoire les pages les plus souvent visitées pour pouvoir les fournir plus rapidement, on l'appelle alors **serveur proxy-cache.**

  - **Si votre navigateur est configuré** de manière à travailler avec le serveur proxy de votre fournisseur d'accès, lorsque vous demandez une page, votre navigateur interroge d'abord le proxy

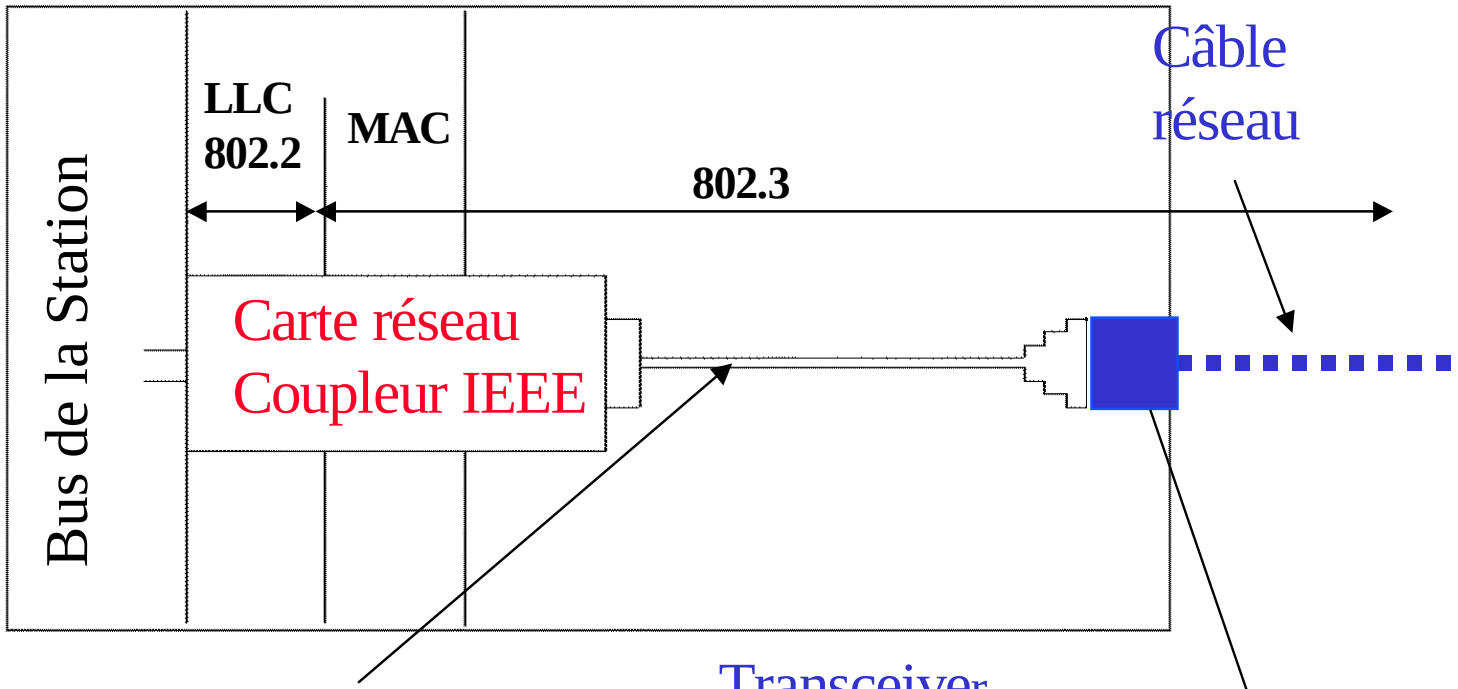
- Une fonction de sécurité

  - le serveur proxy peut servir de **firewall** (pare-feu), c'est alors un système qui filtre les informations en ne laissant, par exemple, passer que les ports (protocoles) choisis pour des raisons de sécurité.

Modèle OSI et IEEE 802

Couche ISO Liaison Physique

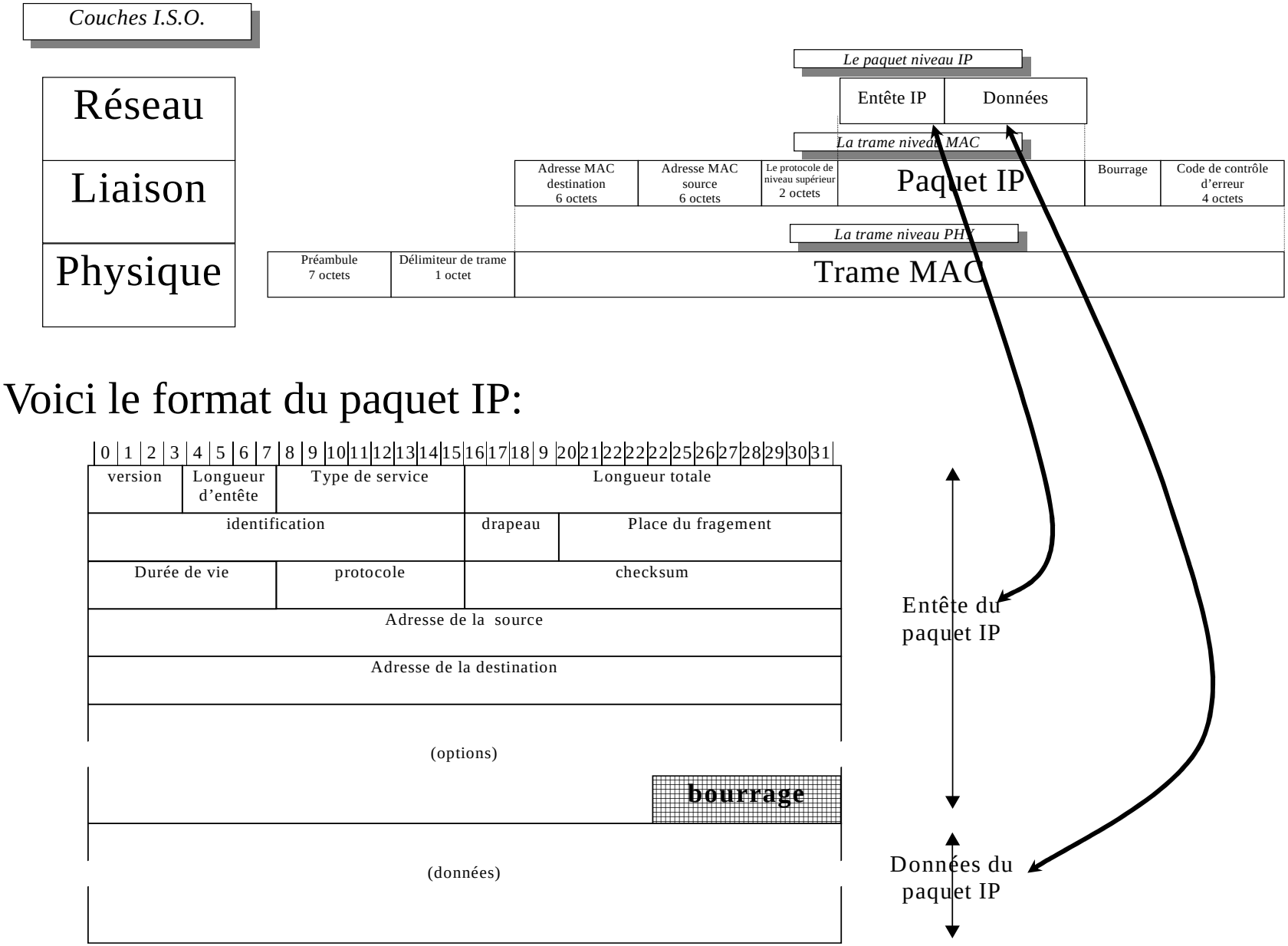
MAC
Media Access Control
LLC
Logical Link Control



Câble transceiver ou Drop
AUI : Attachment Unit Interface

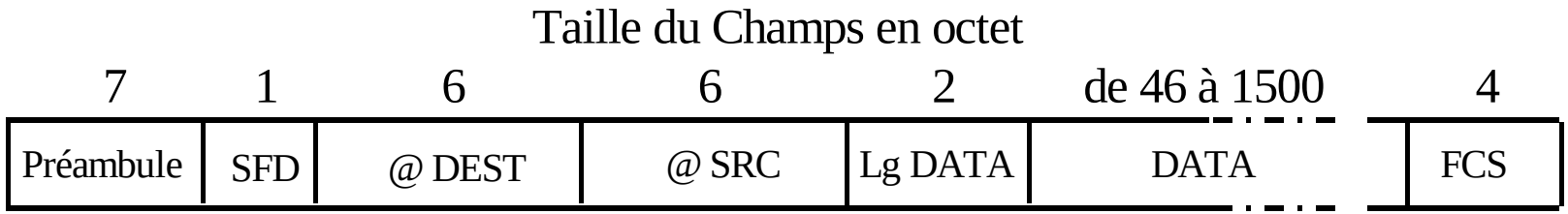
Transceiver
MAU : Medium Attachment Unit
MAU = PMA + MDI
PMA = Physical Medium Attachment
MDI Medium Dependant Interface

Voici l’encapsulation des trames :



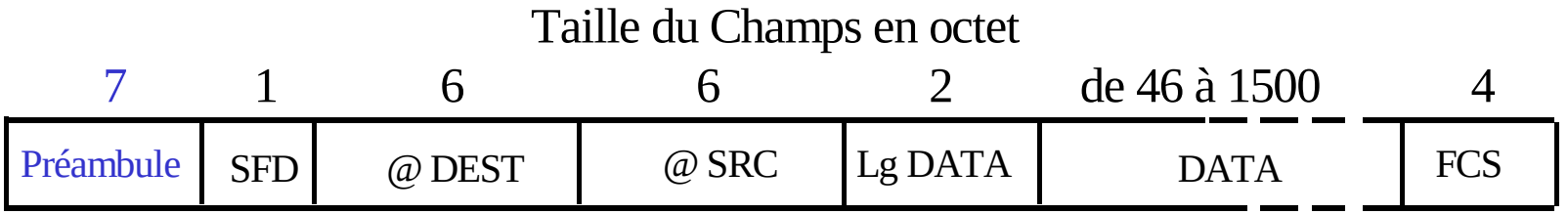
Voici le format du paquet IP:

Format d'une trame IEEE 802.3



- Débit d'émission / réception : 10 Mb/s
  - 10 bits par µs
- Longueur des trames (avec préambule et SFD) :
  - 26 octets réservés au protocole
  - Longueur minimale : **72 octets**
  - Longueur maximale : **1526 octets**

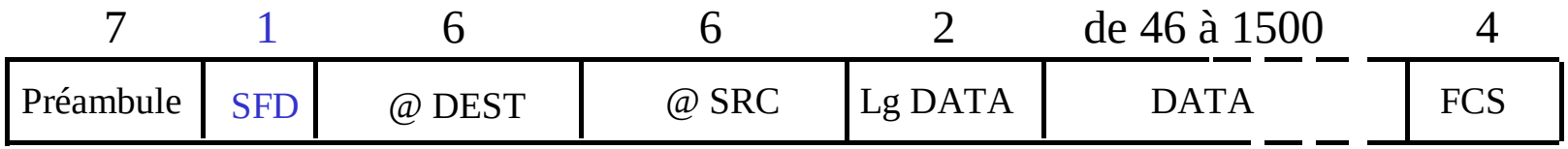
Trame IEEE 802.3 : Préambule



- Taille : 7 octets identiques (10101010) <=> Simple suite continue de bit à 0 et de bit à 1
- Assez long pour servir à la synchronisation de l'horloge locale
- Pas de fin de trame (pas d'échappement)

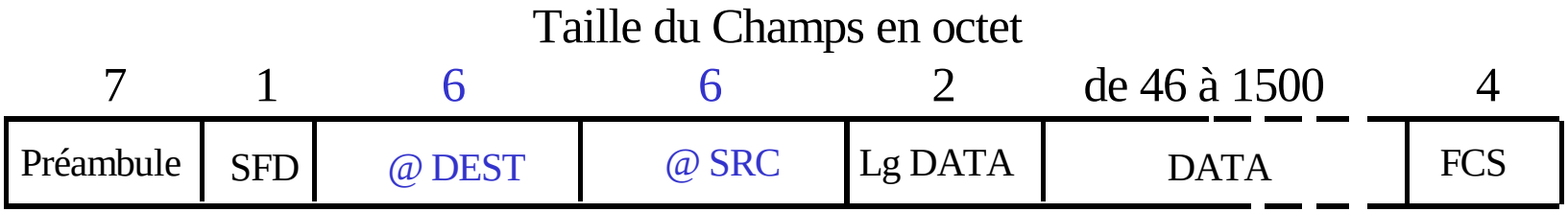
Trame IEEE 802.3 : SFD

Taille du Champs en octet



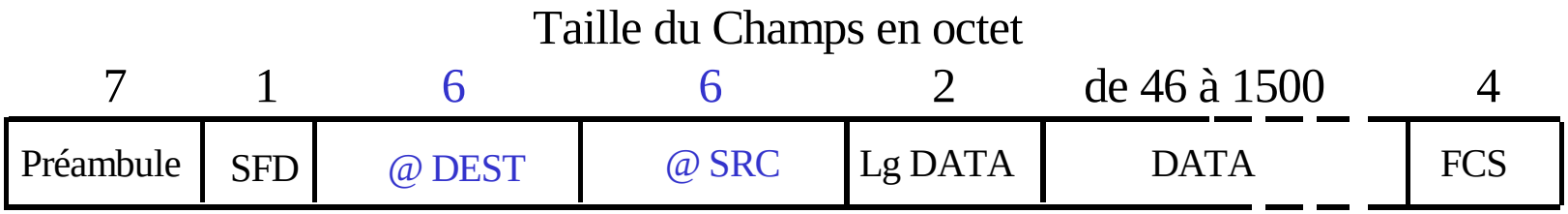
- SFD : Start Frame Delimitor
- Marque le début de la trame
- Taille : 1 octet
- SFD = 10101011

Trame IEEE 802.3 : Adresses



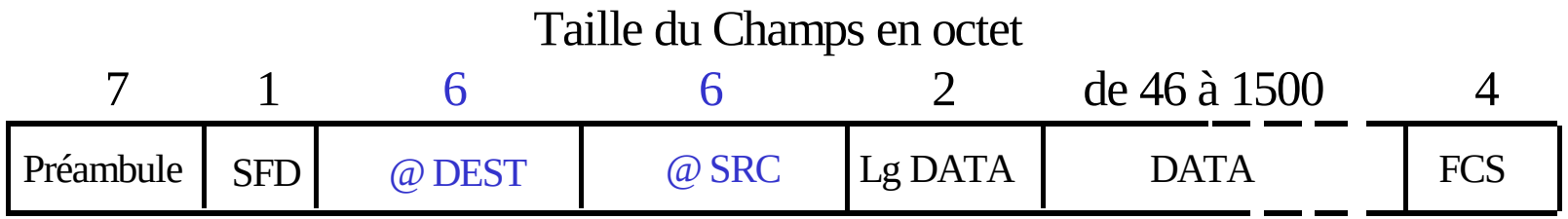
- Détails dans RFC 1700
- Liste : ftp://ftp.isi.edu/in-notes/iana/assignments/ethernet-numbers
- Adresses IEEE 802.3 ou Ethernet : 48 bits (6 octets).
  - syntaxe : 08:00:20:05:B3:A7 ou 8:0:20:5:B3:A7 …..
- 1er bit transmit : spécifie une adresse individuelle (0) ou de groupe (1)
- 2ième bit transmit : spécifie si l'adresse est administré localement (0) ou universellement par IEEE (1)

Trame IEEE 802.3 : Adresses



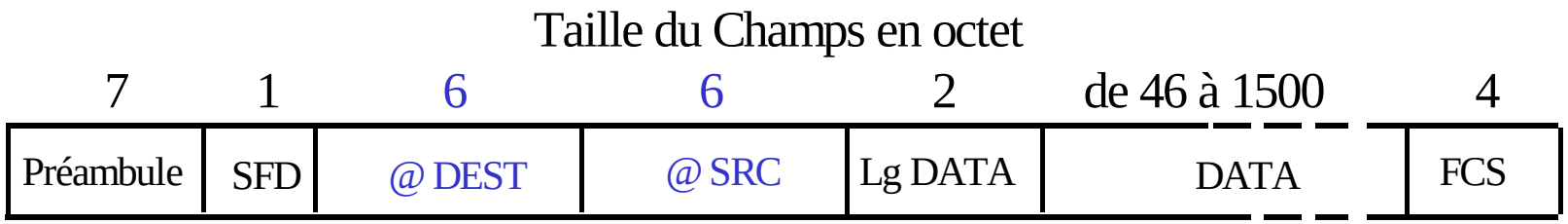
- Exemples :
  - Broadcast (diffusion) : FF:FF:FF:FF:FF:FF
  - Multicast (groupe) : 1er Bit à 1 (1er octet d'adresse impair) :
    - IP (RFC1112) 01-00-5E-00-00-00 à 01-00-5E-7F-FF-FF
    - Spanning Tree 01-80-C2-00-00-00 ou HP Probe 09-00-09-00-00-01
  - Individuelle : 1er Bit à 0 (1er octet d'adresse pair) :
    - 08:00:20:09:E3:D8 ou 00:01:23:09:E3:D5

Trame IEEE 802.3 : Adresses



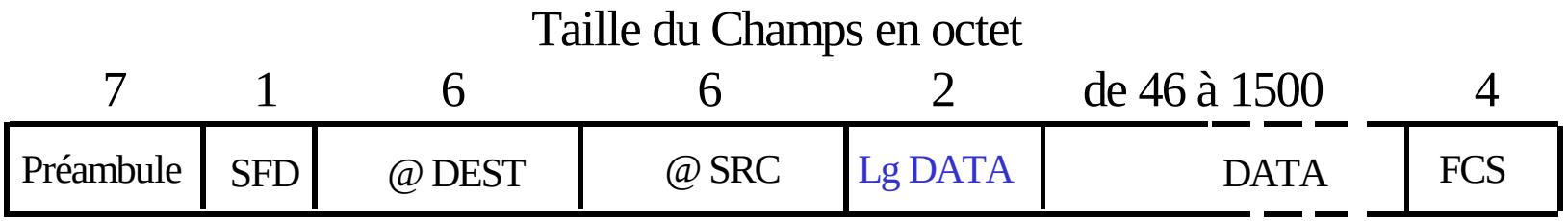
- Plus généralement avec les 2 premiers bits à 0
- Attribuées aux fabricants de coupleur ethernet pour définir l'adresse physique de leur coupleur
- Les 3 derniers octets étant librement alloués par le fabricant
  - $(256)^3$ = 16.78 millions de possibilités pour le fabricant
    - Cisco 00:00:0C:XX:XX:XX
    - Cabletron 00:00:1D:XX:XX:XX
    - Sun 08:00:20:XX:XX:XX
    - HP 08:00:09:XX:XX:XX

Trame IEEE 802.3 : Adresses



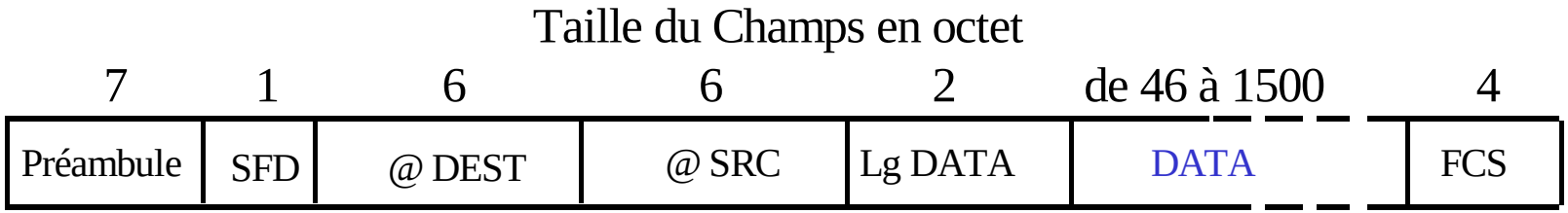
- L'adresse destinataire peut donc représenter :
  - L'adresse physique d'une machine locale
  - L'adressage d'un groupe de machines (multicast)
  - Toutes les machines du réseau local (broadcast)
- L'adresse source représente seulement :
  - L'adresse physique de la station émettrice

Trame IEEE 802.3 : Longueur



- Taille : 2 octets (valeur = 1500)
- Donne le nombre d'octets utilisé par les données dans l trame

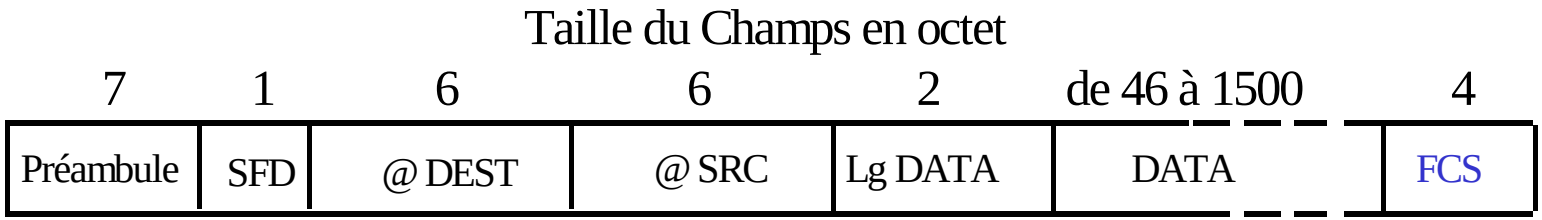
Trame IEEE 802.3 : Données



- 1 < Taille du champs de données utiles < 1500 octets
- Padding : Ajout d'octet(s) sans signification pour envoyer moins de 46 octets de données

=> Longueur minimale de la trame : **72 octets**

Trame IEEE 802.3 : FCS



- FCS : Frame Check Sequence
- Contrôle à la réception de la trame par calcul
  - Calcul = CRC (Cyclic Redundancy Check) (Division polynomiale)
  - CRC sur champs destination, source, longueur et données
- Taille : 4 octets

Format d'une trame Ethernet

@Mac Dest	@Mac Src	Type	Données	FCS

Champ protocole	*0x0800*	*0x0801*	*0x0802*	*0x0803*	*0x0806*
Protocole	*IP Internet*	*X.25 Internet*	*NBS Internet*	*ECMA Internet*	*ARP*

Champ protocole	*0x0807*	*0x809B*	*0x80D5*	*0x80F3*	*0x86DD*
Protocole	*XNS Compatibility*	*Appletalk*	*IBM SNA Service on Eher*	*AppleTalk AARP (Kinetics)*	*IPv6*

Exercice 1

Une station A d'adresse 00-60-8C-54-22-CD veut envoyer une trame à une station B d'adresse 00-72-9F-43-E8-AB. Sachant que la couche LLC a passée 12 octets de données à la couche MAC pour cette communication, représenter la trame que sera expédié par le media de communication.

Exercice 2

Représenter la réponse de la station B à la station A de l'exemple antérieur, en sachant que la réponse comporte 80 octets de données.

Exercice 3

Représenter la trame émise par la station A des exemples précédents, au moment quelle envoie une trame de broadcast avec 20 octets de données.

- **Ouverture de connexion**
- **Fermeture de connexion**
- **Exchange d’information sans erreur**
- **Perte de trames, reprise sur temporisateur**
- **Pertes multiples de trames**
- **Rupture de séquence**

Format d'une trame LLC

DSAP	SSAP	Control	Information
8 bits	8 bits	8 ou 16 bits	n octets

Le champ Control

Il permet de typer les trames qui vont circuler (≠ type de service)

Type de trame	Valeur des bits 1 et 2	Contenu de la trame	Longueur du champ de contrôle
type I	premier bit = 0	trame d'information	2
type S	premier bit = 1 deuxième bit = 0	trame de supervision	2
type U	premier bit = 1 deuxième bit = 1	trame non numérotée	1

DSAP : Destination Service Access Point

SSAP : Source Service Access Point

pour tous les SAP ➔ 7 bits de poids forts : adresse du SAP

le plus faible des 7 vaut 1 pour un SAP global

le plus faible des 7 vaut 0 pour un SAP local

exemple : réseau IP : 0000 011

réseau X25 : 0111 111

réseau IPX : 1110 000

réseau Net bios: 1111 000

pour tous les SAP ➔ 1 bit de poids faible : appelé bit C/R

pour un SSAP

C/R = 0 si la trame LLC est une trame de commande

C/R = 1 si la trame LLC est une trame de réponse

pour un DSAP

C/R = 0 si la trame est destinée à un SAP unique

C/R = 1 si la trame est destinée à un groupe de SAP

Trame de type I (information)

0				N(S)				P/F				N(R)			

Trame de type S (supervision)

1	0	0	0					P/F				N(R)				RR
1	0	0	1					P/F				N(R)				REJ
1	0	1	0					P/F				N(R)				RNR

Trame de type U (non numérotées)

1	2	3	4	5	6	7	8		
1	1	1	1	1	1	1	0	**SABME**	=> Ouverture de connexion
1	1	0	0	1	0	1	0	**DISC**	=> Fermeture de connexion
1	1	0	0	1	1	1	0	**UA**	=> Acceptation d'une connexion
1	1	1	1	1	0	0	0	**DM**	=> Acceptation d'une fermeture
1	1	0	0	0	0	0	0	**UI**	=> Information non numéroté

Exchange d'information après connexion

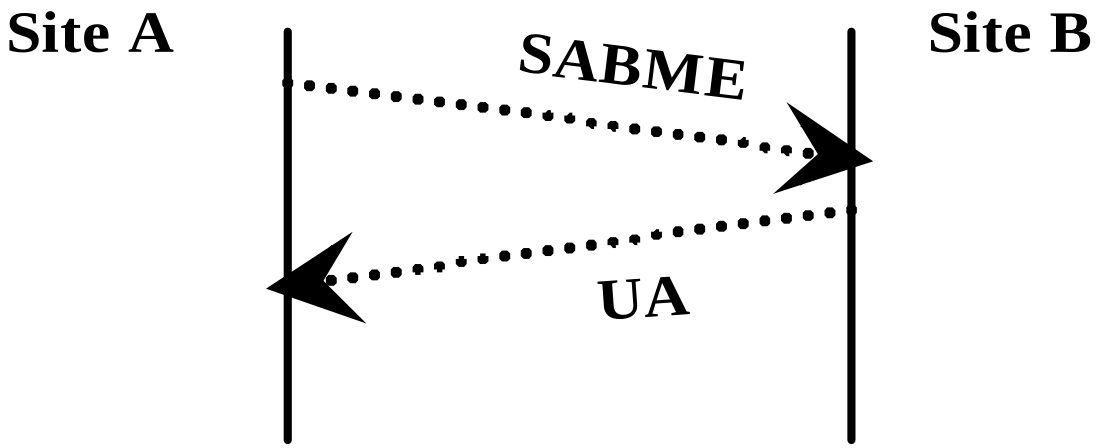
RR : Acquittement de trames
REJ : Rejection de trames
RNR : Inhibe l'émission de nouvelles trames.
SREJ : Demande la retransmission de la trame N(R)

P/F:
- **dans le cas d'une trame de commande (C/R = 0) le bit s'appelle P (Poll), P = 1 : signale la demande d'une réponse immédiate**
- **dans le cas d'une trame de réponse (C/R = 1) le bit s'appelle F (Final), F = 1 : signale la fourniture d'une réponse immédiate**

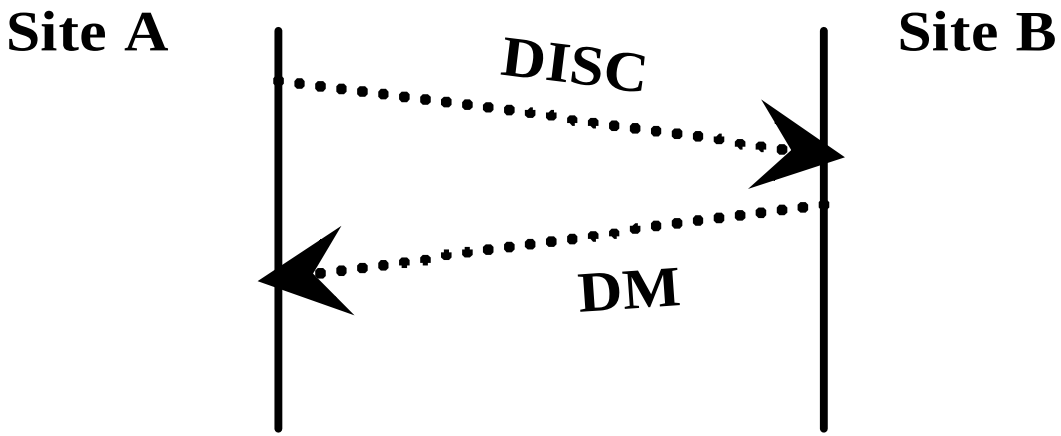
N(S) : compteur de numérotation des trames émises
N(R) : compteur d'acquittement des trames reçues

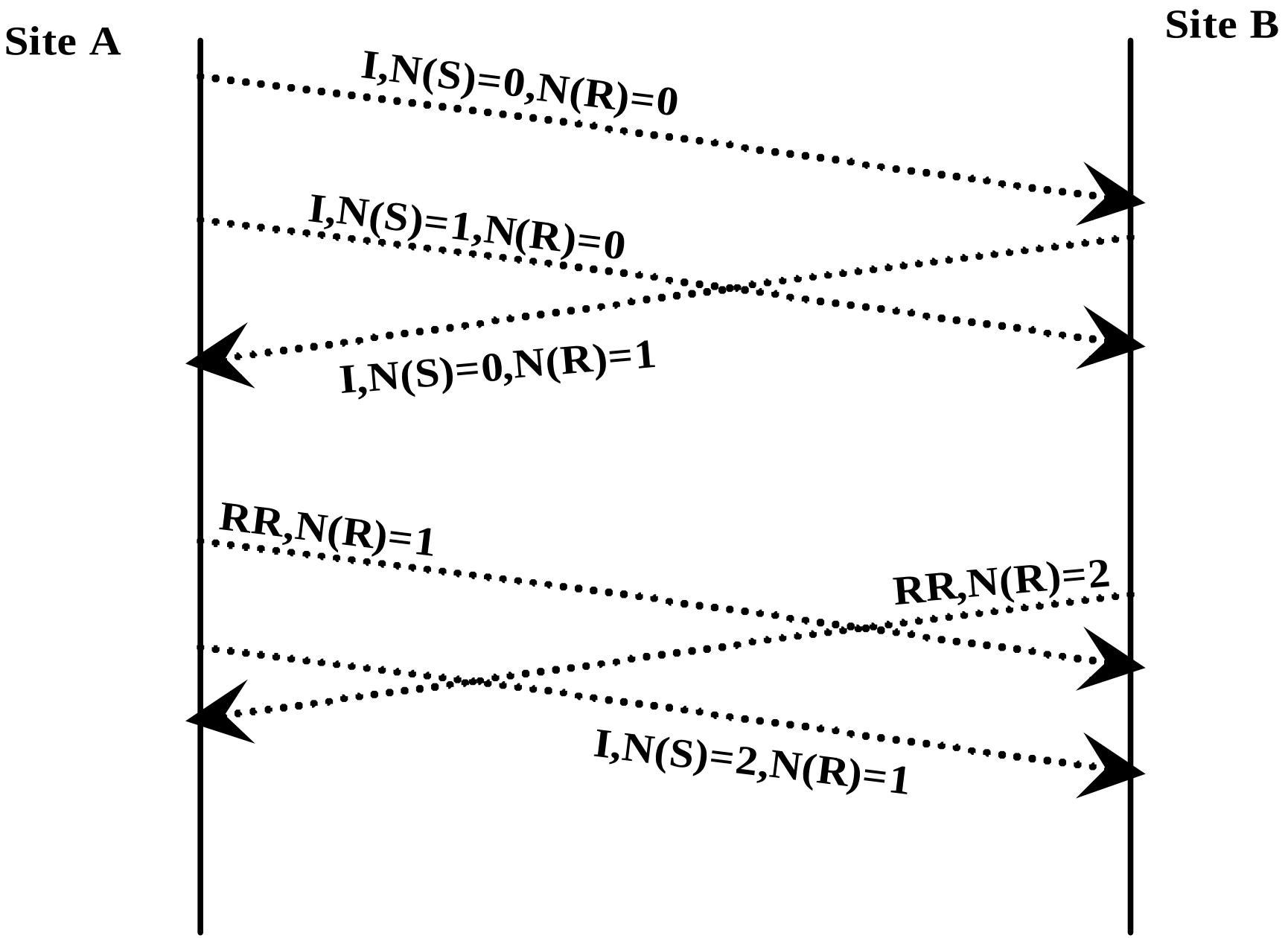
N(R) => contient le numéro de la prochaine trame attendue
=> acquitte ainsi toutes les précédentes

Ouverture d'une connexion LLC type 2



Fermeture d'une connexion LLC type 2





Exemple de transfert d'information sans erreur

Exemple de transfert d'information sans erreur

1. **Le site A envoie sa première trame :**
   - **elle a pour numéro N(S) = 0**
   - **le site A n'a pas encore reçu de trame du site B, il attend donc du site B la trame : N(R) = 0**

2. **Le site B envoie sa première trame :**

- elle a pour numéro N(S) = 0

- il acquitte en même temps la première trame qu'il a reçu du site A en signalant qu'il attend la 2^{éme} trame du site A : N(R) = 1

3. **Le site A envoie sa deuxième trame, il n'a pas encore reçu la première trame du site B :**

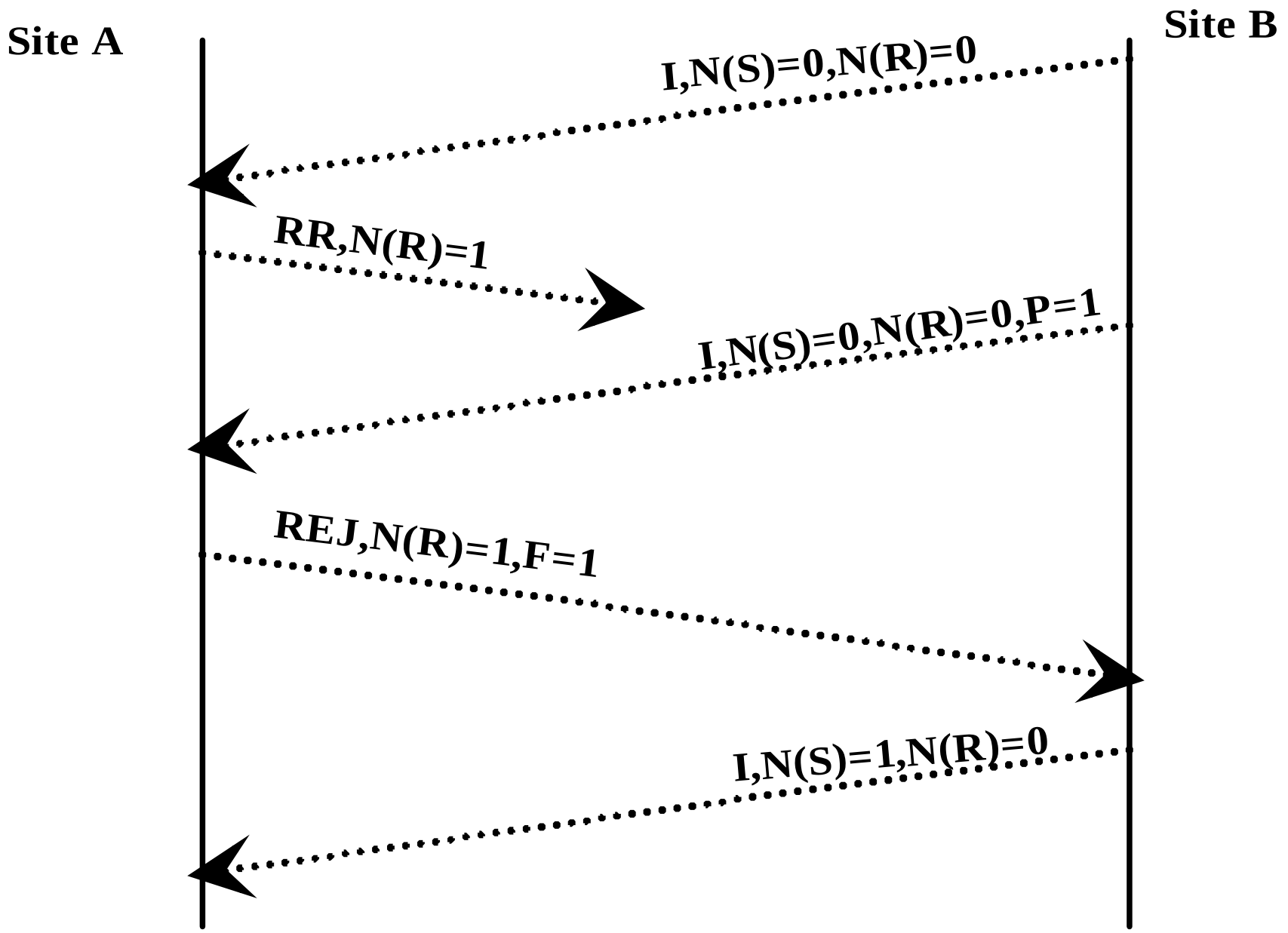
- la trame envoyée a pour numéro N(S) = 1

- la trame attendue du site B est toujours la première : N(R) = 0

4. Le site A n'avait pas de donné à envoyer quand il a reçu la première trame du site B, il ne l'a donc pas acquitté. Au bout d'un certain temps (à échéance d'un temporisateur) , n'ayant toujours pas de données à envoyer, il acquitte la trame reçue du site B avec une trame de supervision, trame sans données, avec le compteur N(R) = 1 (il attend la trame 1 du site B).

5. Le site B, quand il a reçu la deuxième trame du site A, n'avait pas de données à envoyer. A échéance du temporisateur, il effectue l'acquittement par une trame de supervision : N(R) = 2.

6. Au bout d'un certain temps, propre à l'application, le site A envoie sa troisième trame N(S) = 2, et N(R) = 1, acquittement (redondant) de la trame reçu du site B …



Perte de trames, reprise sur temporisateur

Perte de trames, reprise sur temporisateur

1- Le site B envoie sa première trame

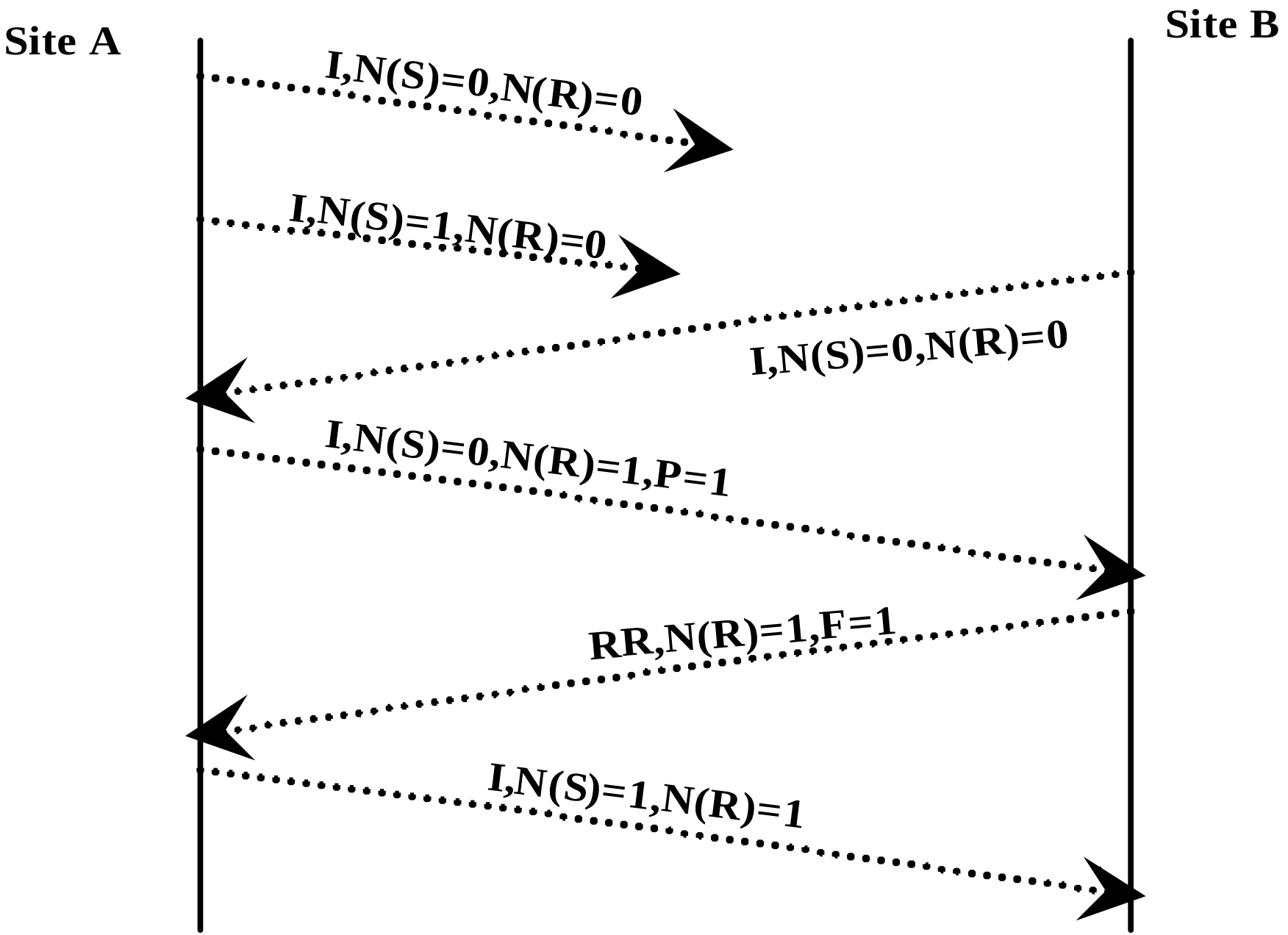
2.2- Le site A reçoit la première trame du site B. N'ayant rien à envoyer, au bout d'un certain temps, il acquitte avec une trame RR, N(R) = 1 (il attend la deuxième trame du site B, trame numérotée 1).

3- Quand la couche LLC reçoit une trame, elle « arme » un temporisateur, l'horloge logicielle décrémente régulièrement le temporisateur ; quand le temporisateur arrive à zéro, le système d'exploitation déclenche l'exécution de la procédure adéquate)

4- L'acquittement émis par A s'est perdu. Au bout d'un certain temps (un autre temporisateur, armé chaque fois qu'une trame est émise), le site B, constatant la non-réception de l'acquittement de la trame envoyée, ré-envoie la même trame, mais en demandant une réponse immédiate, bit P positionné à 1

4- Le site A reçoit une trame qu'il a déjà reçue, il la rejette (trame de supervision de type REJ) tout en signalant qu'il attend la deuxième trame du site B (trame numérotée 1) ; comme la trame reçue est arrivée avec une demande de réponse immédiate, il émet immédiatement cette trame de rejet avec le bit F à 1.

5 - Le site B, ayant reçu l'acquittement de sa première trame, peut poursuivre l'échange …



Pertes multiples de trames

Pertes multiples de trames

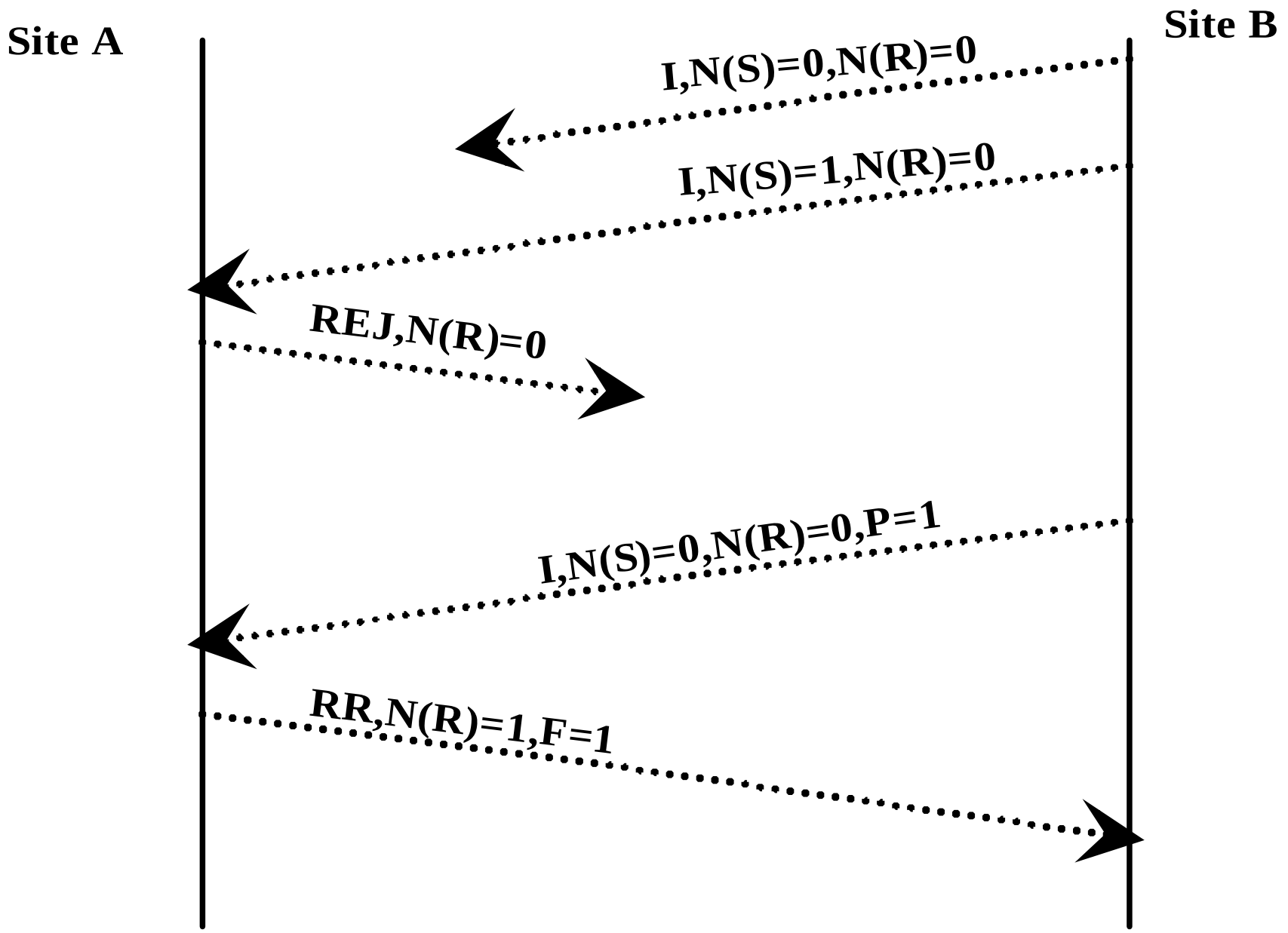
1. **Le site A envoie ses deux premières trames.**

2. **Le site B envoie sa première trame.**

3. **N'ayant pas reçu l'acquittement de sa première trame, il la ré émet, N(S) = 0 avec P = 1. Ayant bien reçu la première trame du site B, il l'acquitte N(R) = 1.**

4. **N'ayant pas de données à envoyer, le site B acquitte, dès sa réception, la trame reçue du site A : trame RR, N(R) = 1, F = 1.**

5. **A réception de l'acquittement de sa première trame, le site A peut reprendre son émission avec l'envoi de sa deuxième trame …**



Rupture de séquence

Rupture de séquence

1. **Le site B envoie ses deux première trames**

2. **Le site A ne reçoit que la deuxième. N'ayant pas reçu la première il rejette la deuxième avec une trame de supervision REJ en signalant qu'il attend la première N(R) = 0.**

3. **Cette trame d'acquittement n'arrive pas à B (décidément la liaison est mauvaise), le site B n'ayant pas reçu l'acquittement de sa première trame reprend l'échange à cet endroit en demandant, pour cette première trame, une réponse immédiate.**

4. **Le site A l'acquitte immédiatement : trame de supervision RR, N(R) = 1, F = 1.**

Exercice 1

Un programme d'une station A d'adresse 00-60-8C-73-E1-F4 de niveau réseau veut envoyer le message "Bonjour" à un programme d'une station B d'adresse 00-72-9F-43-E8-AB en utilisant un service liaison de type LLC1.

Représenter en hexadécimal la trame du niveau LLC et du niveau MAC.

en ascii : A = 41, a = 61

Exercice 2

Un programme de la station A veut échanger des données avec un programme de la station B en utilisant un service liaison de type LLC2.

Représenter en hexadécimal les trames du niveau LLC et du niveau MAC de toutes les trames ayant circulé sur le médium de communication.

Exercice 3

Représenter sur un diagramme temporel les trames échangées entre deux sites **A** et **B** quand :

a) **A** envoie trois trames successives à **B**, qui n'a rien à transmettre à **A**, mais la deuxième trame de **A** est perdue.

b) **A** envoie trois trames à **B** qui lui aussi a trois trames à envoyer à **A**. Mais la 1^{ére} trame de **A** et la 3ème trame de **B** sont perdues.

Le protocole 802.3

CSMA/CD : Carrier Sense Multiple Access / Collision Detect

accès multiple après écoute de la porteuse et détection de collision

exemple : 10 base 2

support : câble coaxial d'impédance 50 ohms, codage manchester

Principe de l'accès au medium de communication

- Une station qui veut émettre écoute le canal.
- S'il est libre, la station émet, sinon elle diffère son émission.
- Si deux stations émettent en même temps, une collision survient.
- L'algorithme BEB (Binary Exponential Backoff) permet de résoudre les collisions en tirant au sort dans chaque station la durée de l'attente avant nouvelle émission.

Principe de l'algorithme du BEB

- A la première tentative : choix entre 0 intervalle d'attente et 1 intervalle.
- A la deuxième : choix entre 0, 1, 2 et 3.
- A la troisième : choix entre 0, 1, ..., 6 et 7.
- A chaque nouvelle tentative, jusqu'à 10 tentatives, on multiplie le nombre de choix par 2. On peut aller jusqu'à 16 tentatives avant de renoncer.

Exercice 1

Soit un réseau local en bus de longueur D. La vitesse de propagation du signal sur le support est V. La capacité de transfert du support est C. Quelle est la longueur minimale L d'une trame pour que le protocole CSMA/CD fonctionne. Justifier la réponse.

Application numérique: D=2,5 km; V = 100 000 km/s; C = 10 Mbits/s

Exercice 2

Le protocole CSMA/CD spécifie une longueur minimale de trame de 512 bits. Quelle est la distance maximale d'un chemin de données entre 2 stations pour un réseau à 100 Mbits/s et une vitesse de propagation de 100.000 km/s ?

Exercice 3

On considère un réseau métropolitain sur fibre optique de débit 100 Mbits/s qui couvre une distance de 100 km. Le signal se propage à une vitesse de 100 000 km/s. Si on choisissait de mettre en œuvre le protocole CSMA/CD, quelle serait la longueur minimale du trame ? Montrer pourquoi le protocole CSMA/CD n'a pas été retenu comme protocole de réseau métropolitain.

Algorithme de reprise après collision :

```
void backoff (int attemps, int *max_backoff)
{ float slot_time = 51,2 ;
  backoff_limit = 10 ;
  int delay ;
if (attemps==1) backoff=2 ;
else {if (attemps <= backoff_limit) maxbackoff=maxbackoff*2 ;
        else maxbackoff = puissance(2, 10)  \*fonction d'élévation
                          d'un nombre x à la puissance y */
}
delay=partie_entière(random() * maxbackoff) ;
                          /* float random() ; fonction - nombre réel pris aléatoirement dans
                          l'intervalle [0,1[ */
wait(delay * slot_time) ; /* void wait(int x)  fonction provoquant une attente de x µs */
}
```

Exercice 4

Soit un réseau local en bus utilisant un protocole de type CSMA/CD et comportant 4 stations notées A, B, C, D. A l'instant t = 0, la station A commence à transmettre une trame dont le temps d'émission dure 6 slots. A l'instant t = 5, les stations B, C, D décident chacune de transmettre une trame de durée 6 slots. Sachant que l'algorithme de reprise après collision est celui donné ci-dessus, faire un diagramme temporel, gradué en slots, décrivant le déroulement des différentes transmissions de trames,

La fonction « random » est supposée retourner les valeurs suivantes:

Station	B	C	D
1er tirage	0.25	0.5	0.75
2éme	0.6	0.25	0.25
3éme	0.33	0.5	0.25

Exercice 5

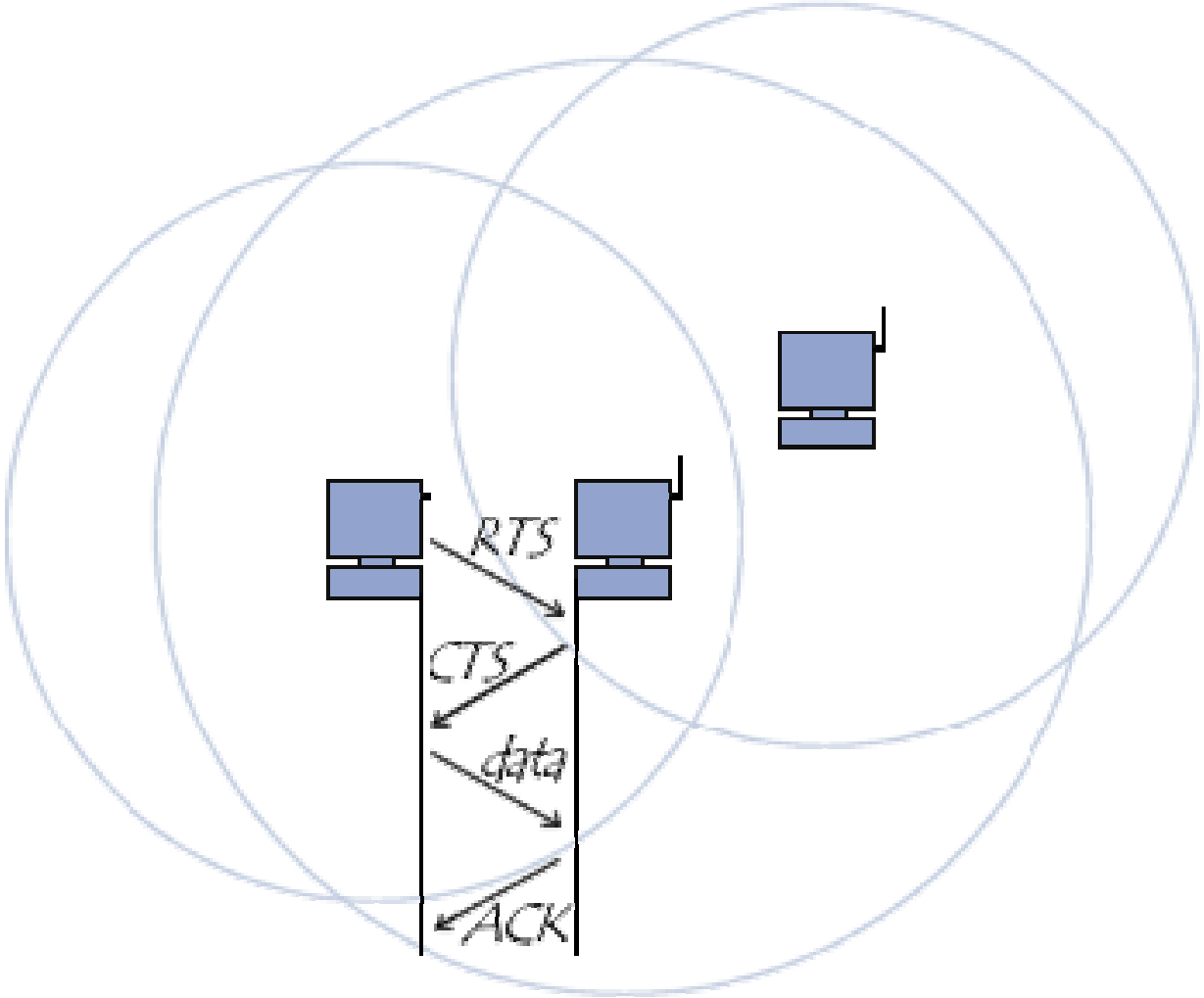
Deux stations A et B entrent en collision. Pour A, il s'agit d'une première collision, alors que pour B, il s'agit de la seconde collision. Quelle est la probabilité qu'il y ait une nouvelle collision entre les stations A et B ?

Le protocole CSMA/CA

Dans un environnement sans fil deux stations communiquant avec un récepteur ne s'entendent pas forcément.
Ainsi la norme 802.11 propose un protocole appelé **CSMA/CA** (*Carrier Sense Multiple Access with Collision Avoidance*).
Le protocole *CSMA/CA* utilise un mécanisme d'esquive de collision basé sur un principe d'accusés de réception entre l'émetteur et le récepteur:

- La station voulant émettre écoute le réseau.
- Si le réseau est encombré, la transmission est différée.
- Si le média est libre pendant un temps donné (appelé *DIFS* pour *Distributed Inter Frame Space*), alors la station peut émettre.
- La station transmet un message appelé *Ready To Send* (noté *RTS)* contenant des informations sur le volume des données qu'elle souhaite émettre et sa vitesse de transmission.
- Le récepteur (généralement un point d'accès) répond un *Clear To Send* (*CTS*), puis la station commence l'émission des données.
- A réception de toutes les données émises par la station, le récepteur envoie un accusé de réception (*ACK*).
- Toutes les stations avoisinantes patientent alors pendant un temps qu'elle considère être celui nécessaire à la transmission du volume d'information à émettre à la vitesse annoncée.

Le protocole CSMA/CA



La détection et la correction d’erreur

1 - La protection contre les erreurs
2 - Le code de parité verticale VRC
3 - Le code de parité longitudinale LRC
4 - Les codes cycliques ou codes polynomiaux
Principe de l'algorithme de calcul du CRC

5 - Les codes correcteurs d'erreurs
Principe du code de Hamming :
code de correction d'erreur

Les canaux de transmission n'étant pas parfaits, ils introduisent des erreurs dans la transmission des données.

Taux d'erreur

$$\text{taux d'erreur} = \frac{\text{Nombre de bits erronés}}{\text{Nombre de bits transmis}}$$

Taux d'erreur d'une ligne téléphonique : de l'ordre de 10^{-4}

- **Pour détecter et éventuellement corriger des erreurs, le codeur ajoute aux données à transmettre des bits dits "de redondance" ou "de contrôle".**
  - **Pour transmettre k bits, le codeur ajoute r bits,**
    - ✓ **on parle de code C(n,k) avec n = k + r**
    - ✓ **les n bits constituent un mot de code**

Rendement d'un code : **rendement = k/n**

Le code de parité verticale : VRC - Vertical Redundancy Check

Ce code est utilisé pour détecter les erreurs lors de la transmission d'un caractère (7 bits)

<u>parité paire</u>

- **on ajoute un bit de parité de telle sorte que le nombre de 1 soit pair**
- **Si le nombre de 1 (avant ajout du bit de parité) est pair**
  - ✓ **Alors bit de parité = 0**
- **Si le nombre de 1 (avant ajout du bit de parité) est impair**
  - ✓ **Alors bit de parité = 1**

<u>parité impaire</u>

- **on ajoute un bit de parité de telle sorte que le nombre de 1 soit impair**

…

Exemple de mise en œuvre du code VRC

bits à transmettre : 0 0 1 1 0 1 0

bit de parité = 1 en parité paire
bit de parité = 0 en parité impaire

bits transmis en parité paire : 0 0 1 1 0 1 0 1
bits transmis en parité impaire : 0 0 1 1 0 1 0 0

Ce code permet de détecter un nombre impair d'erreurs dans le caractère, sans localisation possible

Le code de parité longitudinale :
LRC - Longitudinal Redundancy Check

Ce code est généralement associé au code de parité verticale.
A chaque caractère est ajouté un bit de parité verticale.
A chaque bloc de caractères est ajouté un caractère de parité longitudinale

<u>Exemple de mise en œuvre du code LRC</u>

Caractères à transmettre en parité impaire :
0 1 0 0 0 1 1 1 0 1 1 1 0 0 1 1 0 1 0 1 0

		caractères à transmettre	bit de parité verticale
		0 1 0 0 0 1 1	0
		1 0 1 1 1 0 0	1
		1 1 0 1 0 1 0	1
caractère de parité longitudinale	➔	1 1 0 1 0 1 0	1

Caractères transmis en parité impaire :
0 1 0 0 0 1 1 <u>0</u> 1 0 1 1 1 0 0 <u>1</u> 1 1 0 1 0 1 0 <u>1</u> <u>1 1 0 1 0 1 0 1</u>

Exemple de détection d'erreur

Caractères reçus en parité impaire :

0 1 0 0 0 1 1 0 1 0 1 0 1 0 0 1 1 1 0 1 0 1 0 1 1 1 0 1 0 1 0 1

	bits utiles	*p. verticale*	
	0 1 0 0 0 1 1	0	
	1 0 1 ***0*** 1 0 0	1	⇐ anomalie
	1 1 0 1 0 1 0	1	
parité longitudinale	1 1 0 1 0 1 0	1	

⇑

anomalie

- **Le bit de parité est faux :**
  - ✓**à la deuxième ligne**
  - ✓**à la quatrième colonne**

  ➔**le bit à l'intersection des deux flèches, est un 1 et non un 0**

- **La parité longitudinale permet de détecter**
  - **Une erreur**
  - **Deux erreurs apparues dans la même ligne ou la même colonne**

Dans l'alphabet CCITT nº 5, le mot OSI se code par les 3 caractères
De 7 bits suivants:

O	=	1 0 0 1 1 1 1
S	=	1 0 1 0 0 1 1
I	=	1 0 0 0 0 1 1

On veut émettre une trame comportant ces trois caractères en utilisant pour chaque caractère un huitième bit de parité VRC paire, et pour l'ensemble des trois caractères un bit de LRC.

- ✓Pour chaque caractère, calculer le mot de code sur 8 bits.
- ✓Calculer le LRC du mot OSI ainsi codé.
- ✓Quelle sera la trame effectivement transmise?

Même question en utilisant une parité impaire

Principe du code de Hamming : code de correction d'erreur

Le codage de Hamming est un codage qui permet de faire de la correction d'erreur.

- ✓ **Lors du codage, on ajoute x bits de redondance aux n bits utiles.**
- ✓ **Lors du décodage, les bits de redondance permettent de vérifier qu'il n'y a pas eu d'altération de la chaîne codée.**
- ✓ **Dans le cas où il y une erreur, les bits de redondance permettent de calculer la position de l'erreur**

soit
- **m le nombre de bits à transmettre**
- **n le nombre de bits transmis**
- **x le nombre de bits de redondance**

n = m + x

➢ **Pour un nombre n de bits effectivement transmis, x est égal au nombre de puissance de 2 inférieur à n**

si n = 12 $\Rightarrow$ x = 4 car 2^0, 2^1, 2^2, 2^3 < 12, mais 2^4 > 12

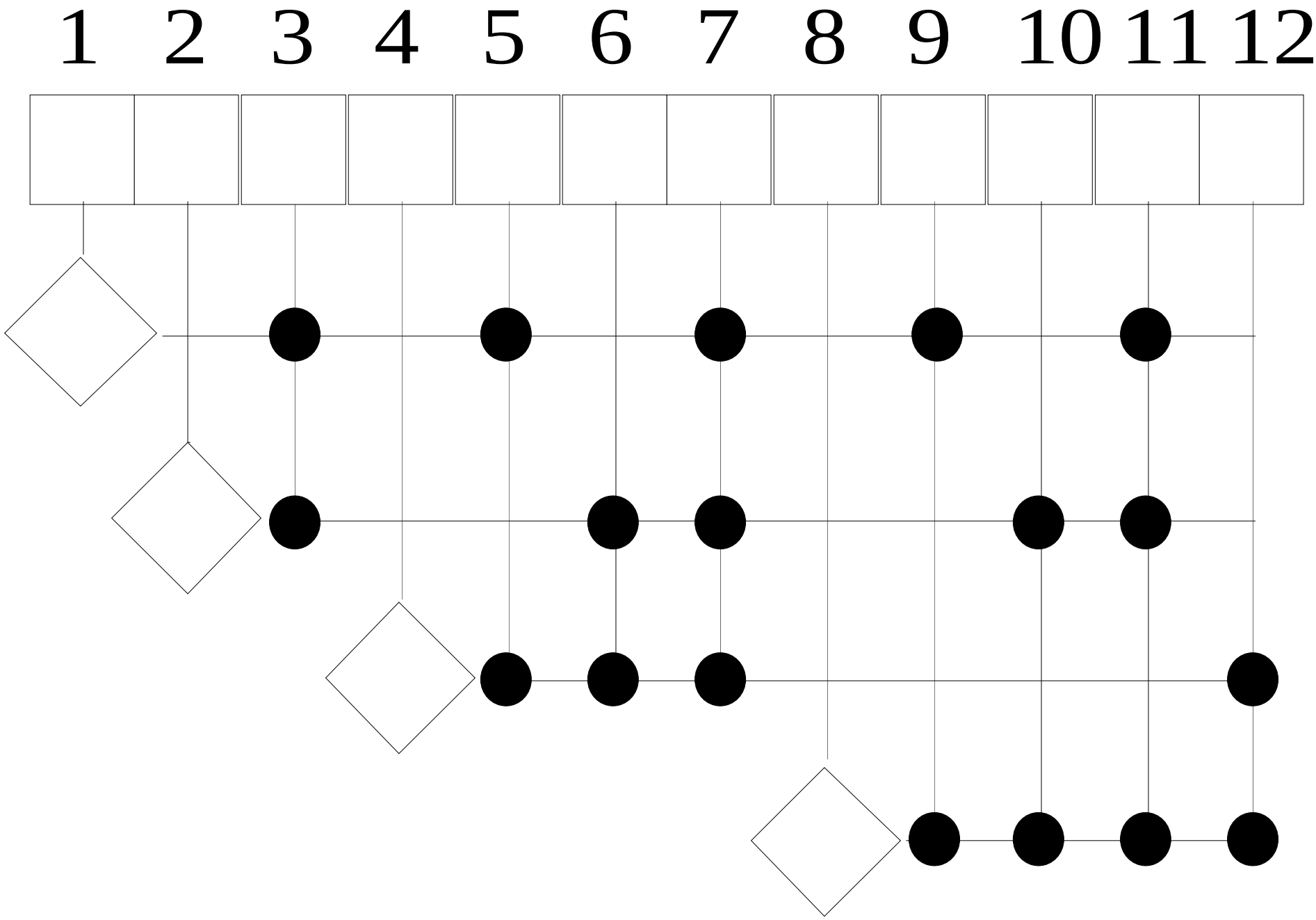
➔ **comme n = m + x, n=12 et x = 4, on en déduit m = 8**

soit le mot à transmettre : $m_1\ m_2\ m_3\ m_4\ m_5\ m_6\ m_7\ m_8$

soit le mot transmis : $n_1\ n_2\ n_3\ n_4\ n_5\ n_6\ n_7\ n_8\ n_9\ n_{10}\ n_{11}\ n_{12}$

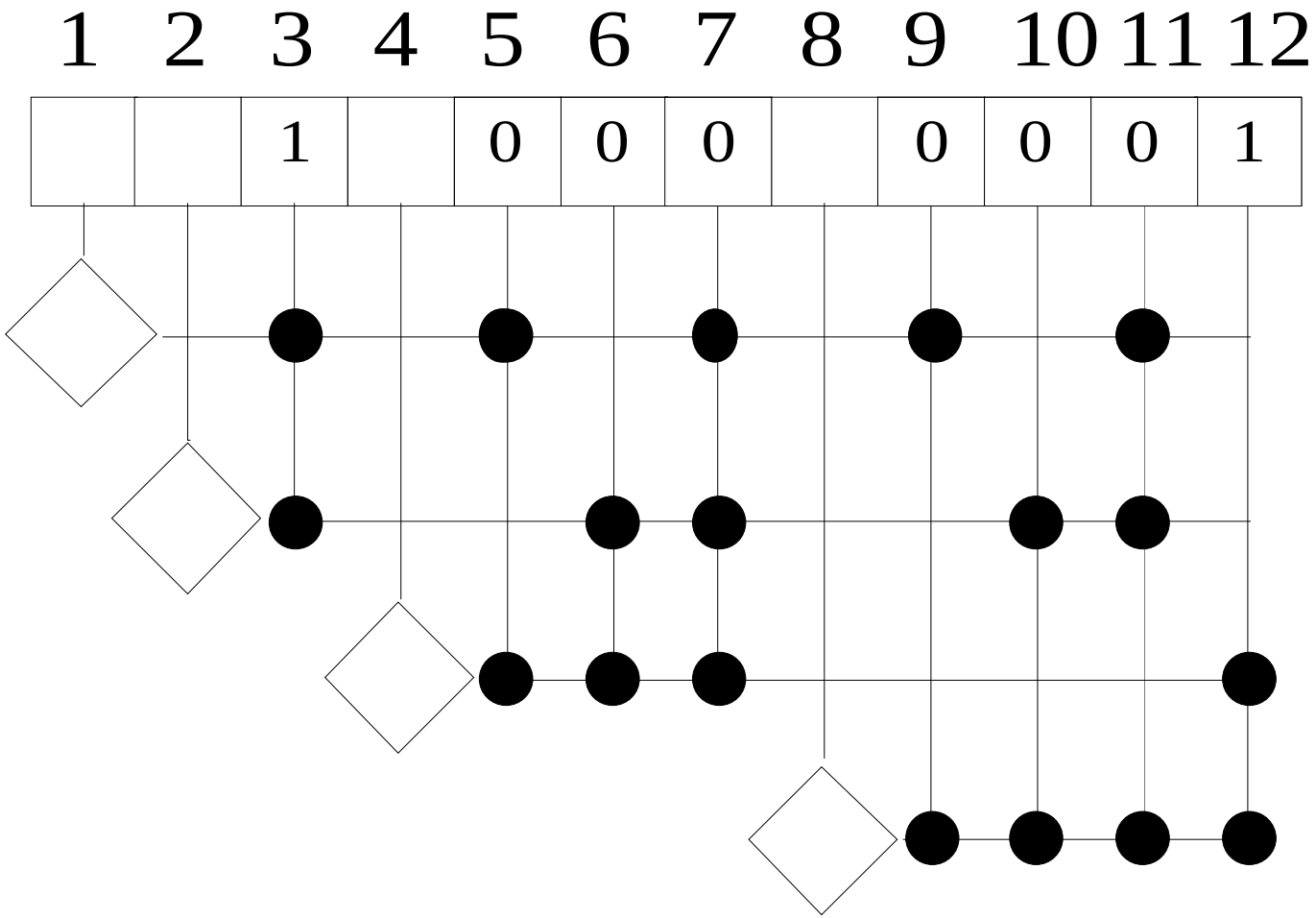
le bit du rang correspondant à une puissance de 2, est le bit de parité paire du mot composé des bits dont le rang utilise dans sa décomposition en puissance de 2 la puissance du rang du bit correspondant.

$1 = 2^0$	n_1 = bit de parité des bits 3, 5, 7, 9 et 11
$2 = 2^1$	n_2 = bit de parité des bits 3, 6, 7, 10 et 11
$3 = 2^0 + 2^1$	$n_3 = m_1$
$4 = 2^2$	n_4 = bit de parité des bits 5, 6, 7 et 12
$5 = 2^0 + 2^2$	$n_5 = m_2$
$6 = 2^1 + 2^2$	$n_6 = m_3$
$7 = 2^0 + 2^1 + 2^2$	$n_7 = m_4$
$8 = 2^3$	n_8 = bit de parité des bits 9, 10, 11 et 12
$9 = 2^0 + 2^3$	$n_9 = m_5$
$10 = 2^1 + 2^3$	$n_{10} = m_6$
$11 = 2^0 + 2^1 + 2^3$	$n_{11} = m_7$
$12 = 2^2 + 2^3$	$n_{12} = m_8$


1 2 3 4 5 6 7 8 9 10 11 12

Exemple : calcul de la clé à émettre

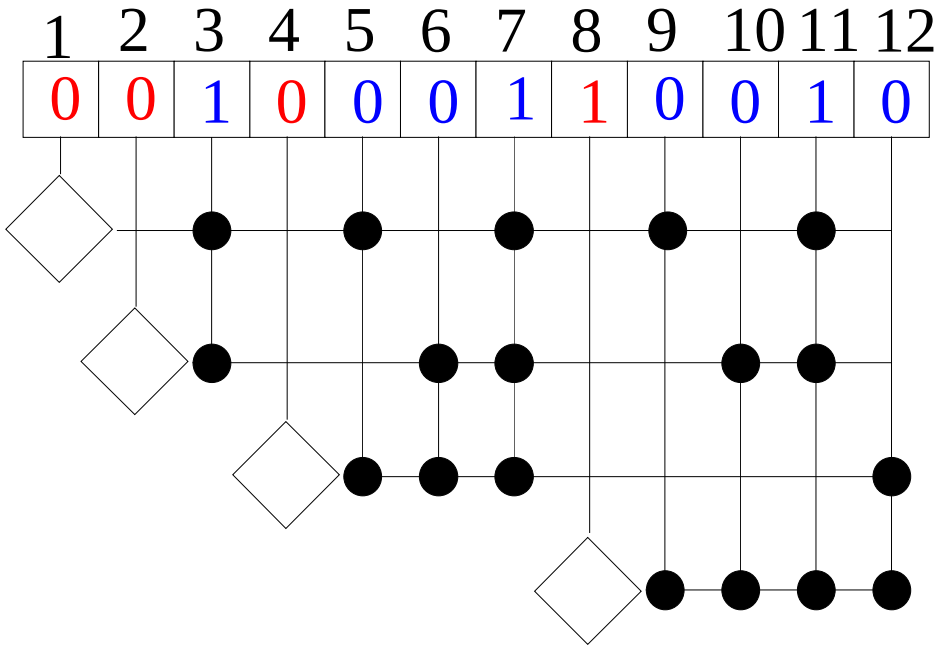
- **soit à transmettre : 1 0 0 0 0 0 0 1**



- **bits transmis : 1 1 1 1 0 0 0 1 0 0 0 1**

Exemple : contrôle de la clé reçue

- **Supposons que le mot reçu soit :** **0 0 1 0 0 0 1 1 0 0 1 0**



- **apparemment** **le mot transmis est : 1 0 0 1 0 0 1 0**

  la clé reçue est : 0 0 0 1

  la clé calculée est : 1 1 1 1

  bits 1 2 4 8

- **les bits de clé erronés sont les bits 1, 2 et 4,**

le bit erroné est donc le 1+2+4 = 7éme bit

le mot réellement transmis était : 1 0 0 0 0 0 1 0

Quelle sera la trame transmise en code de Hamming si l'information à transmettre est:

0 1 0 1 0 1 0 1

Le même pour:

1 0 0 1 0 0 1 1 0 1

Quelle est l'information transmise si la trame reçue est:

1 1 1 0 1 1 1 0 1 1 1 1

Est-elle correcte?

Le même pour la trame:

1 1 1 1 0 1 0 1 1 0 1 0

Et pour la trame:

0 1 1 1 0 0 1 0 0 0 1 1 0 1 0

Les codes cycliques ou codes polynomiaux
Principe de l'algorithme de calcul du CRC

- **Soit k la longueur en bits de la trame à émettre.**
- **On considère les bits de cette trame comme les coefficients d'un polynôme M(x) de degré k - 1.**

exemple : trame 1 1 0 0 1 0 0 ➔ M(x) = $x^6+x^5+x^2$

- **Pour calculer le CRC on utilise un polynôme générateur G(x) de degré r.**

exemple : G(x) = x^5+x^4+1

- **L'entité émettrice et l'entité réceptrice doivent utiliser le même polynôme.**
- **La longueur de la trame effectivement émise est égale à k + r.**
- **Les k bits de poids fort sont les k bits de la trame initiale.**
- **Les r bits de poids faibles sont les coefficients du polynôme reste de la division modulo 2 du polynôme x^r M(x) par le polynôme G(x)**

exemple de calcul du CRC

- soit un polynôme générateur $G(x) = x^5 + x^4 + 1$
- soit la trame à transmettre 1 1 0 0 1 0 0 ➔ $M(x) = x^6 + x^5 + x^2$

$x^r M(x) / G(x)$	=	$(x^{11} + x^{10} + x^7) / (x^5 + x^4 + 1)$
quotient modulo 2	=	$x^6 + x^2$
reste modulo 2	=	x^2

d'où la trame émise : 1 1 0 0 1 0 0 0 0 1 0 0

- Pour contrôler le CRC l'entité destinataire effectue la division du polynôme de degré k -1 + r, associé à la trame reçue, par le polynôme générateur.
- Si il n'y a pas eu d'erreur de transmission, le reste de la division est nul.
- Avec un CRC de r bits, on détecte une erreur dans la trame reçue si celle-ci contient au plus r bits erronés.
- Pour la trame Ethernet- $G(x) = x^{32} + x^{26} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x^1$
- Pour la trame Wi-Fi- $G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x^1 + 1$

Soit à emettre une trame 1 1 0 1 0 1 1 0 1 1 en utilisant le polynôme générateur $x^4 + x + 1$

Quel est le CRC de cette trame?

Quelle est la trame effectivement transmise?

Et pour la trame 1 0 0 0 1 0 0 0 1 1 0 0 1

Comme auparavant, on veut toujours transmettre une trame constitué du mot OSI dans l'alphabet CCITT n^o 5 en utilisant pour chaque caractère un huitième bit de parité VRC paire, mais au lieu d'utiliser un caractère LRC, on souhaite utiliser un CRC avec le polynôme générateur $x^8 + 1$

Quel est le CRC?

Quel est la trame effectivement transmise?

O	=	1 0 0 1 1 1 1
S	=	1 0 1 0 0 1 1
I	=	1 0 0 0 0 1 1

L’Adressage Internet - Adressage IP

➢Objectif

- Fournir un service de communication universel permettant à toute machine de communiquer avec toute autre machine du réseau
- Une machine doit être accessible aussi bien par des humains que par d'autres machines
- Une machine doit pouvoir être identifiée par :
  - une adresse qui doit être un identificateur universel de la machine
  - un nom mnémotechnique pour les utilisateurs
  - une route précisant comment la machine peut être atteinte

➢Solution

- Adresse IP codée sur 32 bits (IPv4)
- Chaque machine possède une adresse IP unique au monde

Adresse IP

- Codée sur 32 bits = 4 octets
- Représentée en notation décimale pointée W.X.Y.Z où W, X, Y et Z ∈ [0…255]
- Représentée en notation binaire pointée xxxxxxxx.xxxxxxxx.xxxxxxxx.xxxxxxxx
- Constituée d'une paire (Net_id, Host_id)
  - Net_id : adresse du réseau (id. réseau)
  - Host_id : adresse de la machine dans le réseau (id. machine)
- Structurée en 5 classes (A, B, C, D et E) selon la valeur du 1er octet

Domain Name System

Les adresses IP sont difficiles à mémoriser – On les associe a des noms

193.105.81.57 = machine.domain.machin.fr

Une seule adresse IP par nom, mais plusieurs noms par adresse IP

Les noms des domaines principaux sont un code de pays (.fr .it .br), ou bien:

.com	commercial
.edu	éducation
.gov	gouvernement
.mil	militaire
.org	organisation
.net	réseau
.int	international

Pour connaître le numéro IP d'une machine en Australie, **susana.library.uwa.edu.au** (utilitaire nslookup)

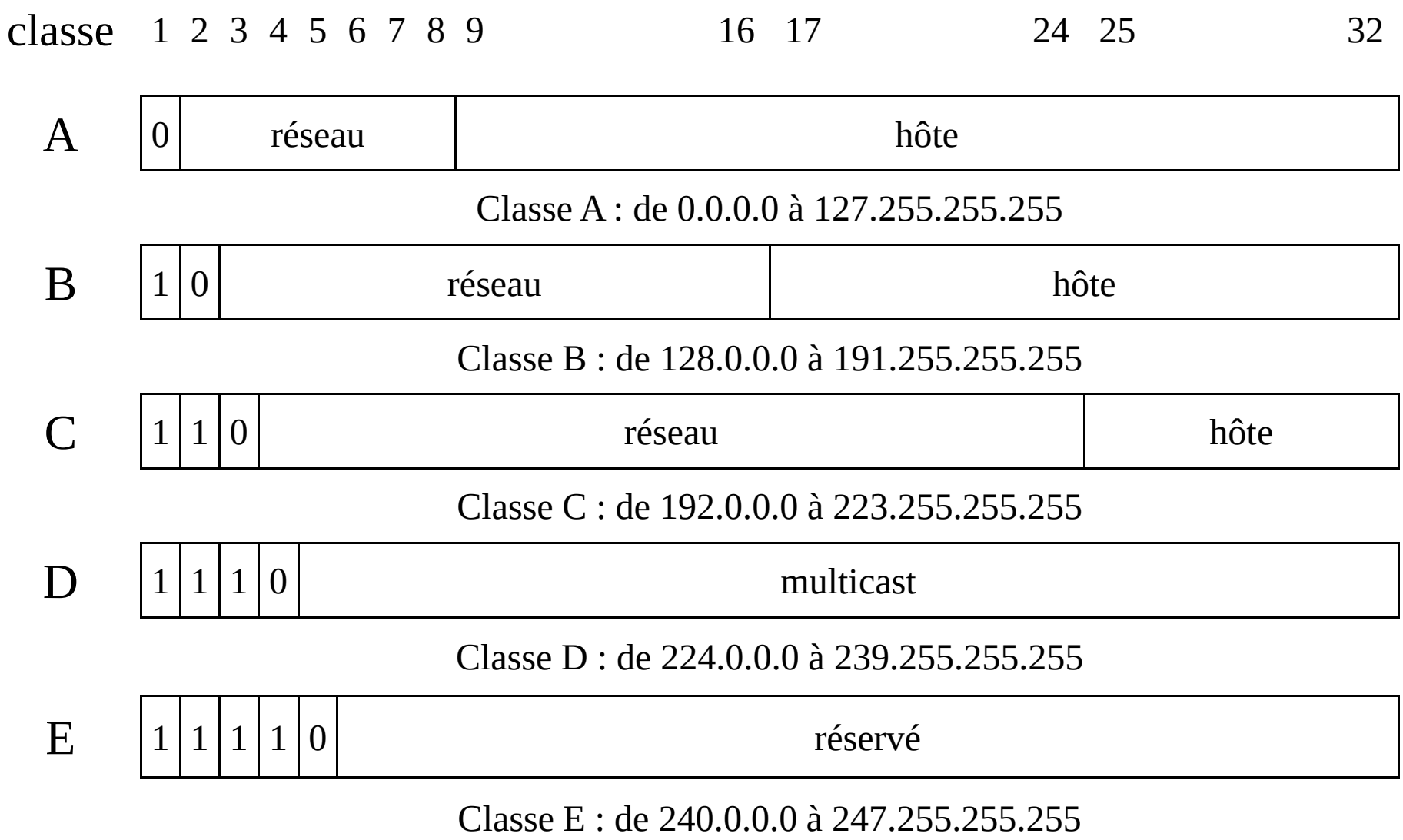
Je connais par cœur les adresses de **a.root-servers.net** (198.41.0.4), …, **m.root-servers.net** (202.12.27.33)

J'appelle 198.41.0.4 qui me dit que pour tout ce qui concerne **.au** il faut demander à **munnari.oz.au** (128.250.1.21)

J'appelle 128.250.1.21 qui me dit qu'il faut demander à **beast.library.uwa.edu.au** (130.95.106.12) pour tout ce qui concerne **.library.uwa.edu.au**

J'appelle 130.95.106.12 qui me dit que l'adresse cherchée est 130.95.245.38

Adresse IP


classe
1 2 3 4 5 6 7 8 9
16 17
24 25
32
A
0
réseau
hôte
Classe A : de 0.0.0.0 à 127.255.255.255
B
1
0
réseau
hôte
Classe B : de 128.0.0.0 à 191.255.255.255
C
1
1
0
réseau
hôte
Classe C : de 192.0.0.0 à 223.255.255.255
D
1
1
1
0
multicast
Classe D : de 224.0.0.0 à 239.255.255.255
E
1
1
1
1
0
réservé
Classe E : de 240.0.0.0 à 247.255.255.255

➢Adresses IP réservées

•0.0.0.0 : adresse non encore connue

- Utilisée par une machine pour connaître sa propre adresse IP au démarrage

•Net_id tout à 0, Host_id : adresse non encore connue

- •Utilisée par une machine pour connaître sa propre adresse IP au démarrage – très peau employé

•Net_id, Host_id tout à 0 : désigne le réseau de la machine

- •La valeur 0 ne peut pas être attribuée à une machine
- •Ex : 130.20.0.0 désigne le réseau de classe B 130.20

•Net_id, Host_id tout à 1 : adresse de diffusion (broadcast)

- •Désigne toutes les machines du réseau
- •Ex : 130.20.255.255 désigne toutes les machines du réseau de classe B 130.20

- 255.255.255.255 (FF.FF.FF.FF): adresse de diffusion (broadcast)
  - Désigne toutes les machines du réseau auquel appartient l'ordinateur qui utilise cette adresse. L'avantage par rapport à l'adresse précédente est que l'émetteur n'est pas obligé de connaître l'adresse du réseau auquel il appartient
- 127.X.Y.Z : boucle locale
  - Utilisée pour permettre les communications inter-processus sur un même ordinateur ou réaliser des tests de logiciels car tout logiciel de communication recevant des données pour cette adresse les retourne simplement à l'émetteur
- Réseaux privés, intranet
  - Les adresses de classe A de 10.0.0.0 à 10.255.255.255, de classe B de 172.16.0.0 à 172.31.255.255 et de classe C de 192.168.0.0 à 192.168.255.255 sont réservées à la constitution de réseaux privés

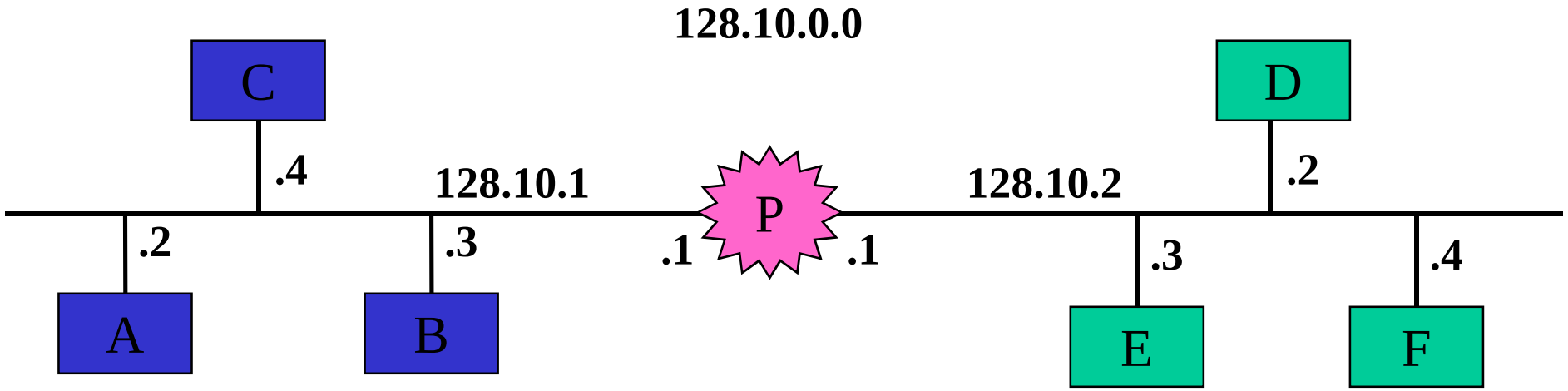
Segmentation en sous-réseaux

Utilité de la segmentation d'un réseau en sous-réseaux

- Réduire le nombre de communications sur un même segment
- Connecter des réseaux d'architectures hétérogènes
- Structurer la gestion des domaines en sous-domaines

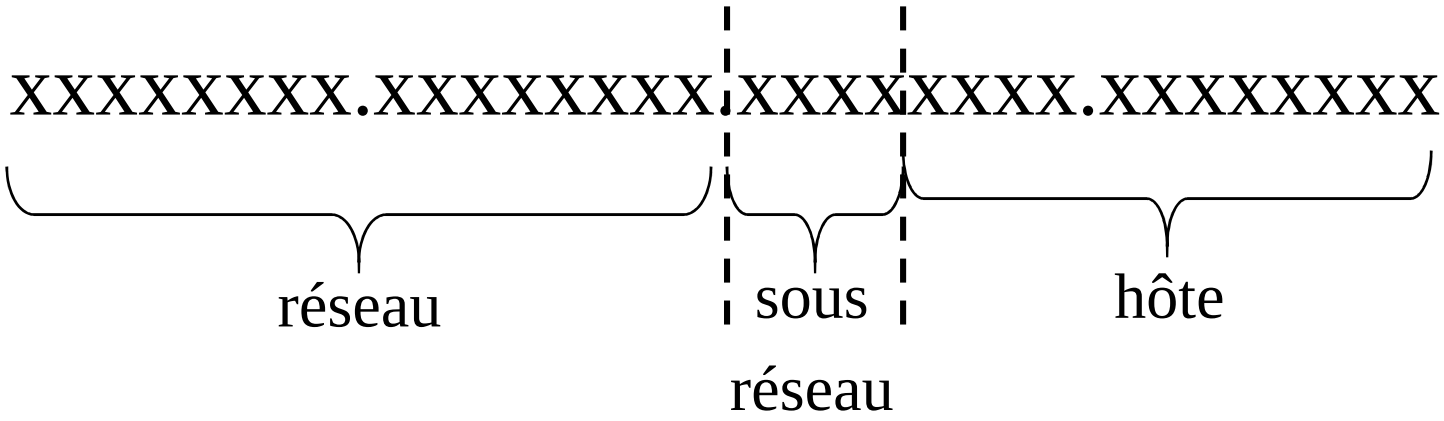
L'adressage IP doit permettre de déterminer si une machine est

- •Une machine du même sous-réseau que moi
- •Une machine d'un sous-réseau différent sur le même réseau que moi
- •Une machine sur un autre réseau



Structure du sous adressage

Exemple



Le champ réseau reste le même (classe A, B ou C)

Le champ hôte est divisé en 2 champs : le sous-réseau et l'hôte

Le masque de sous-réseau ou *netmask*

Permet d'obtenir l'adresse du réseau dans lequel la machine se situe en effectuant un ET logique entre l'adresse IP de la machine et le masque

Le masque de réseau, masque par défaut

Permet de recueillir l'adresse du réseau

Le masque de sous-réseau

Permet de recueillir l'adresse du réseau et du sous-réseau

Exemple : IP 130.53.166.76 et masque 255.255.224.0

```
IP                  10000010.00110101.10100110.01001100
MASQUE              11111111.11111111.11100000.00000000
IP AND MASQUE       10000010.00110101.10100000.00000000
```

$\Rightarrow$ classe B, réseau 130.53.0.0, sous-réseau 130.53.160.0

Notation : adresse IP/masque 130.53.166.76/19 (19=nombre de bits contigus du masque)

Protocole IP

IP : Internet Protocol

- Est au cœur du fonctionnement d'Internet
- Protocole de niveau réseau
- Responsable de la transmission des datagrammes (unité de données) en mode sans connexion
- Assure un service non fiable de délivrance des datagrammes (pas de garantie que les datagrammes arrivent à destination, c'est TCP qui assure cette fiabilité !)
- Responsable de l’adressage et du routage des paquets entre les stations, par l'intermédiaire des routeurs
- Responsable de la fragmentation des données (assemblage/désassemblage)

⇒ Certains datagrammes peuvent être perdus, dupliqués, retardés, altérés ou remis dans le désordre. On parle de remise au mieux (*best effort delivery*). Ni l'émetteur, ni le récepteur ne sont informés directement par IP des problèmes rencontrés. Le mode de transmission est non connecté car IP traite chaque datagramme indépendamment de ceux qui le précèdent et le suivent. Ainsi, en théorie, au moins deux datagrammes IP issus de la même machine et ayant la même destination peuvent ne pas suivre obligatoirement le même chemin

Lors de l'émission, les fonctions assurées sont :

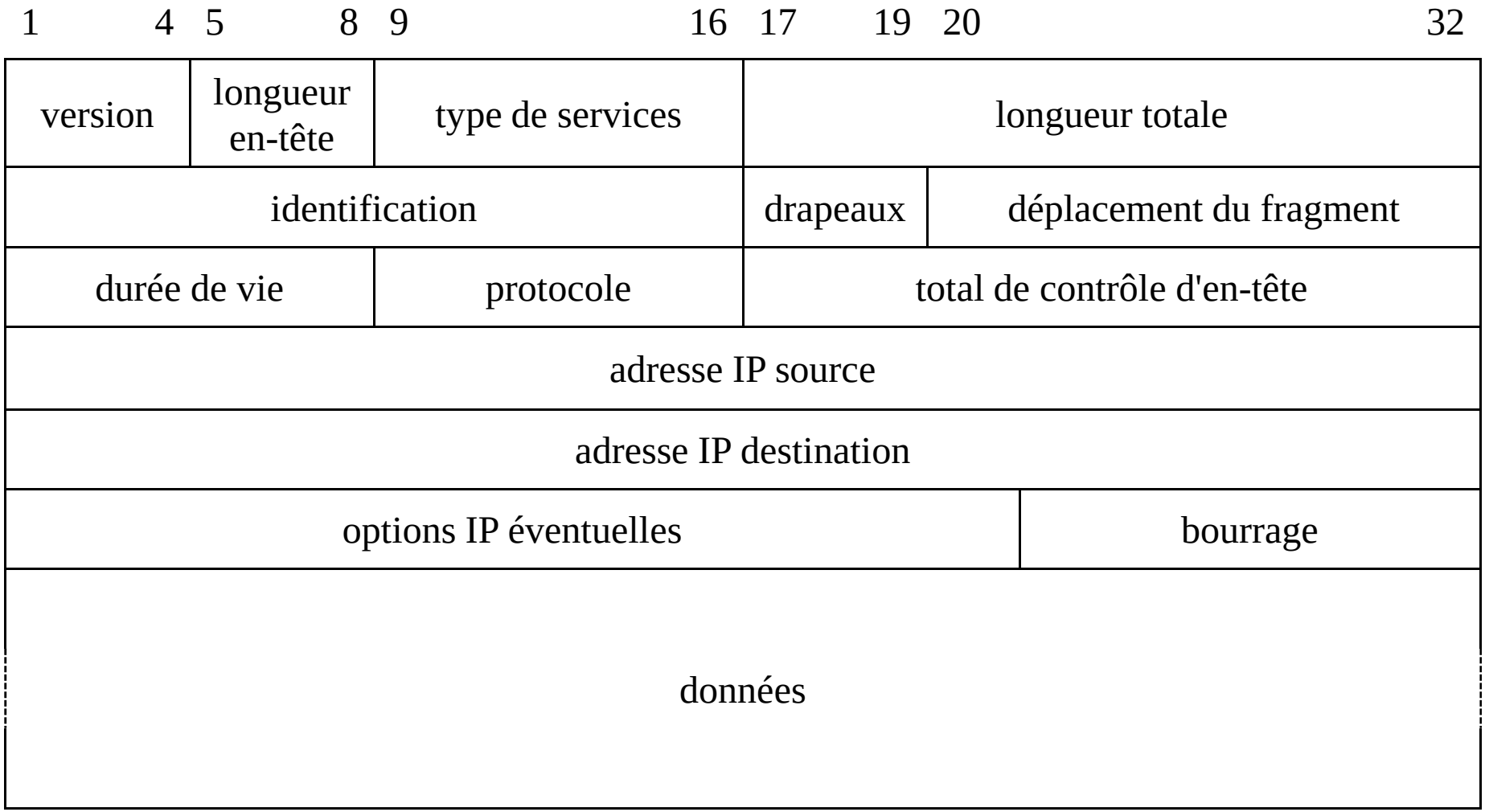
- identification du paquet
- détermination de la route à suivre (adressage et routage)
- vérification du type d'adressage (station ou diffusion)
- fragmentation de la trame si nécessaire

Lors de la réception, les fonctions assurées sont :

- vérification de la longueur du paquet
- contrôle des erreurs
- réassemblage en cas de fragmentation
- transmission du paquet réassemblé au niveau supérieur

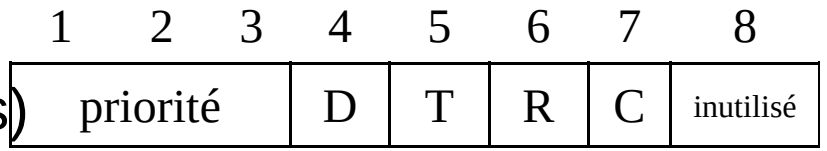
Le datagramme IP

- Constitué d'un en-tête et d'un champ de données, formaté sur 4 octets



- Version
  - la version code sur 4 bits le numéro de version du protocole IP utilisé. Tout logiciel IP doit d'abord vérifier que le numéro de version du datagramme qu'il reçoit est en accord avec lui-même. Si ce n'est pas le cas, le datagramme est tout simplement rejeté. Ceci permet de tester de nouveaux protocoles sans interférer avec la bonne marche du réseau
  - la version courante est la 4, d'où son nom IPv4
  - la version 6 apparaît sous le nom IPv6
- Longueur d'en-tête – IHL (Internet Header Length)
  - représente sur 4 bits la longueur de l'en-tête du datagramme, en nombre de mots de 32 bits (4 octets)
  - permet de déterminer où commencent exactement les données transportées
  - ce champ est nécessaire car un en-tête peut avoir une taille supérieure à 20 octets (taille minimale de l'en-tête où la longueur de l'en-tête vaut 5) à cause des options que l'on peut ajouter

Type de services (TOS : Type Of Services)



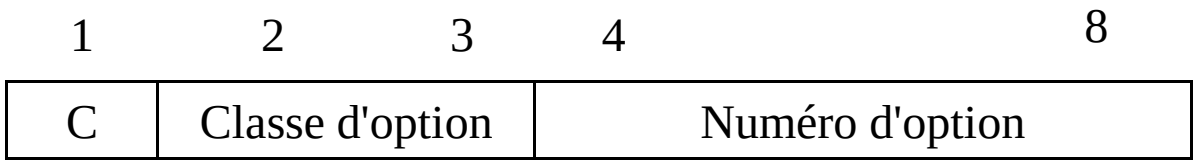
- codé sur 8 bits, le type de services indique la manière dont doit être géré le datagramme par le routeur : qualité de services
  - *priorité* varie de 0 (priorité normale, valeur par défaut) à 7 (priorité maximale pour la supervision du réseau) et permet d'indiquer l'importance de chaque datagramme. Même si ce champ n'est pas pris en compte par tous les routeurs, il permettrait d'envisager des méthodes de contrôle de congestion du réseau qui ne soient pas affectées par le problème qu'elles cherchent à résoudre
  - Les 4 bits D, T, R et C permettent de spécifier ce que l'on veut privilégier pour la transmission du datagramme. Ils servent à améliorer la qualité du routage et ne sont pas des exigences incontournables. Simplement, si un routeur connaît plusieurs voies de sortie pour une même destination, il pourra choisir celle qui correspond le mieux à la demande
  - D est mis à 1 pour essayer de minimiser le délai d'acheminement (par ex : choisir un câble sous-marin plutôt qu'une liaison satellite)
  - T est mis à 1 pour maximiser le débit de transmission
  - R est mis à 1 pour assurer une plus grande fiabilité
  - C est mis à 1 pour minimiser les coûts de transmission
  - Si les quatre bits sont à 1, c'est la sécurité de la transmission qui doit être maximisée

- Longueur totale
  - codée sur 2 octets, contient la longueur totale en octets du datagramme (en-tête + données)
  - la taille maximale d'un datagramme ne peut dépasser 65535 octets
- Identification, drapeaux (flags) et déplacement du fragment (fragment offset)
  - interviennent dans le processus de fragmentation des datagrammes IP
- Durée de vie (Time To Live : TTL)
  - indique le nombre maximal de routeurs que peut traverser le datagramme
  - elle est initialisée à N (souvent 32 ou 64, mais aussi 128 et 256) par la station émettrice
  - elle est décrémentée de 1 par chaque routeur qui reçoit le datagramme et le réexpédie
  - lorsqu'un routeur reçoit un datagramme dont la durée de vie est nulle, il le détruit et envoie à l'expéditeur un message ICMP (Internet Control Message Protocol). Ainsi, il est impossible qu'un datagramme "tourne" indéfiniment dans un réseau

- Protocole
  - permet d'identifier le protocole de niveau supérieur dont le message est véhiculé dans le champ de données du datagramme
  - codé sur 1 octet (ex : 1 pour ICMP, 2 pour IGMP, 6 pour TCP, 17 pour UDP)
  - la station destinatrice qui reçoit un datagramme IP peut ainsi diriger les données qu'il contient vers le protocole adéquat
- Total/somme de contrôle d'en-tête (*header checksum*)
  - Permet de détecter les erreurs survenant dans l'en-tête du datagramme, et par conséquent, l'intégrité du datagramme
  - Porte sur l'en-tête du datagramme et non sur les données véhiculées
  - Pour calculer cette somme de contrôle, on commence par la mettre à zéro. Puis, en considérant la totalité de l'en-tête comme une suite d'entiers de 16 bits, on fait la somme de ces entiers en complément à 1. On complémente à 1 cette somme et cela donne le total de contrôle que l'on insère dans le champ *checksum*.

- Total/somme de contrôle d'en-tête (*header checksum*)
  - A la réception du datagramme, il suffit d'additionner tous les nombres de l'en-tête (y compris le *checksum*). Si on obtient un nombre avec tous ses bits à 1 (FFFF), c'est que la transmission s'est passée sans problème, il n'y a pas d'erreur
  - Ex : soit le datagramme IP dont l'en-tête est le suivant : 4500 05DC E733 222B FF11 checksum C02C 4D60 C02C 4D01. La somme des mots de 16 bits en compléments à 1 donne 6E08 (=46E04 modulo FFFF), son complément à 1 est 91F7 (=6E08 XOR FFFF). Le datagramme est donc expédié avec la valeur 91F7 de checksum. A la réception, la somme des 10 mots de 16 bits doit donner FFFF
- Adresses IP source
  - contient sur 32 bits l'adresse IP de la machine émettrice du datagramme
- Adresses IP destination
  - contient sur 32 bits l'adresse IP de la machine réceptrice du datagramme

- Options IP éventuelles
  - Ce champ est facultatif et de longueur variable
  - Les options concernent essentiellement des fonctionnalités de mise au point. Une option est définie par un champ octet :

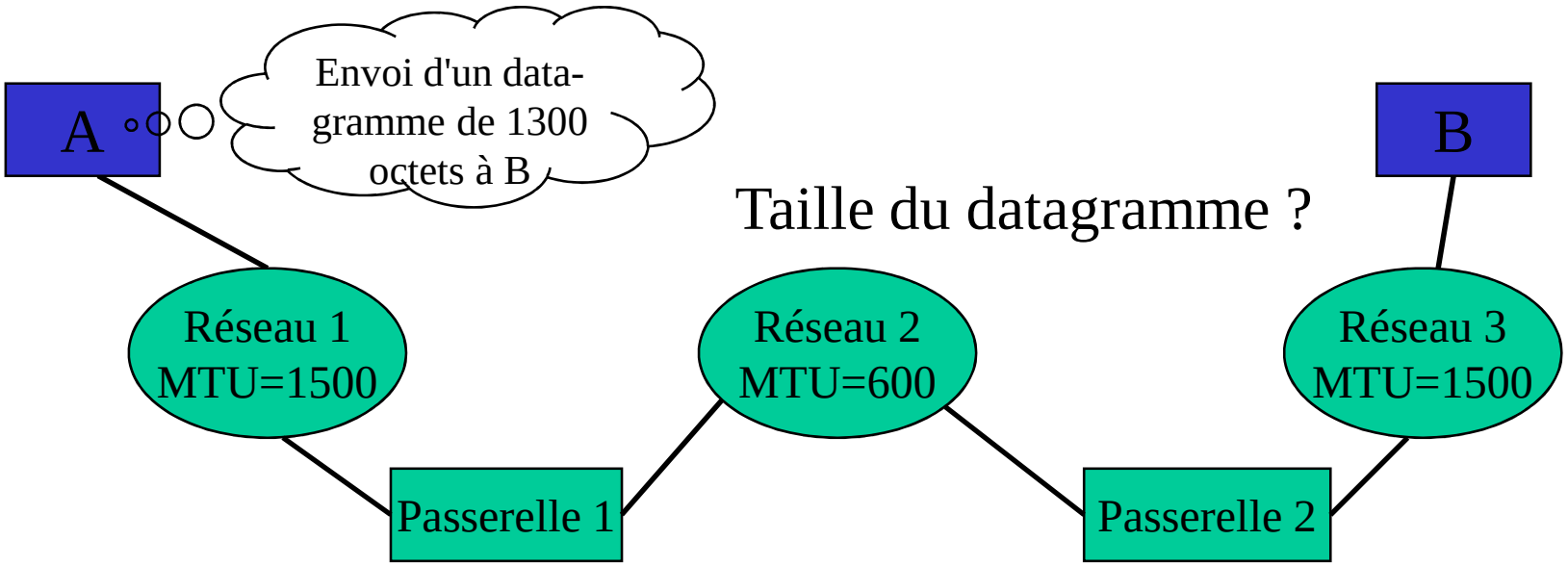


- C (copie) : indique que l'option doit être recopiée dans tous les fragments (C=1) ou bien uniquement dans le premier fragment (C=0)
- les bits classe d'option et numéro d'option indiquent le type de l'option et une option particulière de ce type :
  - Enregistrement de route (classe = 0, numéro = 7) : permet à la source de créer une liste d'adresses IP vide et de demander à chaque passerelle d'ajouter son adresse dans la liste

- Routage strict prédéfini par l'émetteur (classe = 0, numéro = 9) : prédéfinit le routage qui doit être utilisé dans l'interconnexion en indiquant la suite des adresses IP à suivre. Le chemin spécifié ne tolère aucun autre intermédiaire ; une erreur est retournée à l'émetteur si une passerelle ne peut pas appliquer le routage spécifié
    - Routage lâche prédéfini par l'émetteur (classe = 0, numéro = 3) : cette option autorise, entre deux passages obligés, le transit par d'autres intermédiaires
    - Horodatage (classe = 2, numéro = 4) : cette option permet d'obtenir les temps de passage des datagrammes dans les passerelles. Une liste de couples (adresse IP - horodatage) est réservée par l'émetteur ; les passerelles ont à charge de remplir un champ lors du passage du datagramme
- Bourrage/remplissage (*padding*)
    - Le champ options IP ayant une taille variable, il convient de le compléter par des bits de remplissage (0) afin qu'il atteigne une taille multiple de 32 bits (4 octets)

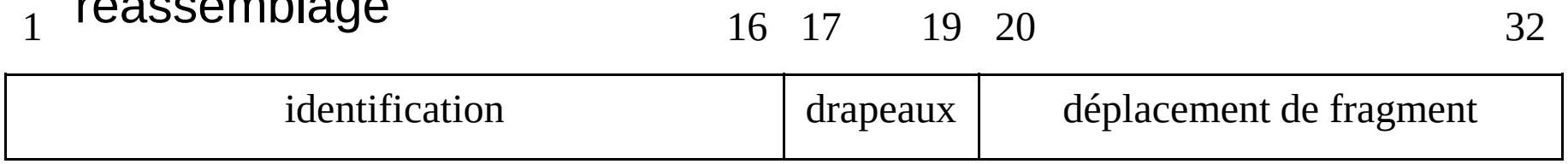
Fragmentation des datagrammes IP

- Problème
  - La taille maximale d'un datagramme IP est 2^{16}=65536 octets
  - Au niveau de la couche liaison, le datagramme est découpé en trames de longueur fonction du protocole utilisé (1500 octets pour Ethernet, 4470 octets pour FDDI,…)
  - Sur Internet, un datagramme transite par des réseaux de technologies différentes
  - Il est, par conséquent, impossible de déterminer une taille maximale de datagramme IP permettant d'être encapsulé dans une seule trame quel que soit le réseau



- Solution
  - Sur toute machine ou passerelle mettant en œuvre TCP/IP, une unité maximale de transfert (MTU : *Maximum Transfert Unit*) définit la taille maximale d'un datagramme véhiculé sur le réseau physique correspondant
  - Lorsque le datagramme est routé vers un réseau physique dont le MTU est plus petit que le MTU courant, la passerelle fragmente le datagramme en un certain nombre de fragments, véhiculés par autant de trames sur le réseau physique correspondant
  - Lorsque le datagramme est routé vers un réseau physique dont le MTU est supérieur au MTU courant, la passerelle route les fragments tels quels (les datagrammes peuvent emprunter des chemins différents)
  - Le destinataire final reconstitue le datagramme initial à partir de l'ensemble des fragments reçus. La taille de ces fragments correspond au plus petit MTU emprunté sur le réseau. Si un seul des fragments est perdu, le datagramme initial est considéré comme perdu : la probabilité de perte d'un datagramme augmente avec la fragmentation

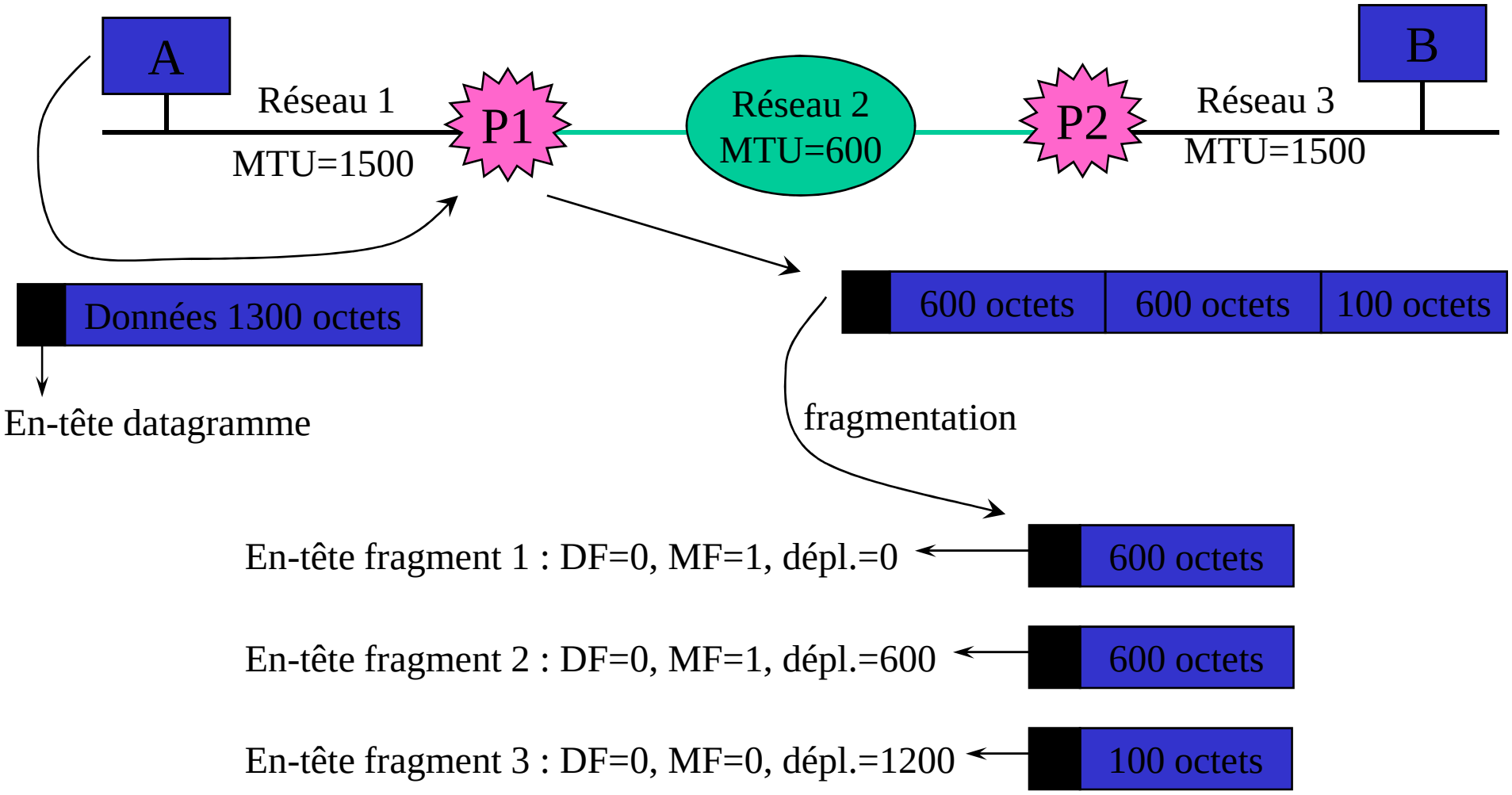
- Les champs *identification*, *drapeaux*, et *déplacement du fragment* du datagramme IP assurent le processus de fragmentation et de réassemblage



- Identification
  - Entier qui identifie le numéro de datagramme initial
  - Tous les fragments d'un même datagramme portent le même numéro
  - Utilisé pour la reconstitution à partir des fragments qui ont tous la même valeur
- Drapeaux
  - Gère la fragmentation sur 3 bits : 0 DF MF
  - Le bit DF (Don't Fragment) demande au routeur de ne pas fragmenter le paquet lorsque DF=1
  - Le bit MF (More Fragment) est à 1 dans tous les fragments, sauf le dernier
- Déplacement du fragment
  - Indique par multiple de 8 octets la position du fragment dans le paquet courant
  - Permet de reconstruire le datagramme initial en positionnant chaque fragment

- Remarques
  - Un datagramme fragmenté n'est réassemblé que lorsqu'il arrive à destination finale. Même si les fragments traversent des réseaux avec un plus grand MTU, les routeurs/passerelles ne réassemblent pas les petits fragments. De plus, chaque fragment est routé de manière totalement indépendante des autres fragments du datagramme d'où il provient
  - Quand le destinataire reçoit le fragment dont le bit fragment à suivre MF est à 0, il est apte à déterminer s'il a reçu tous les fragments du datagramme initial grâce notamment aux champs *identification*, *déplacement* et *longueur totale* des différents fragments
  - Si un fragment doit être à nouveau fragmenté lorsqu'il arrive sur un réseau avec un MTU encore plus petit, ceci est fait comme décrit précédemment, sauf que le calcul du champ déplacement de fragment est fait en tenant compte du déplacement inscrit dans le fragment à traiter

Fragmentation des datagrammes IP



Routage IP

- Routeur
  - Ordinateur ou dispositif effectuant le choix nécessaire à l'acheminement d'un datagramme
  - Le datagramme transite de passerelle en passerelle jusqu'à ce que l'une d'elles le délivre à l'ordinateur de destination
  - Un routeur possède au moins 2 connexions réseaux tandis qu'un ordinateur ne possède généralement qu'une seule connexion

- Table de routage
  - Permet de déterminer vers quelle passerelle envoyer un datagramme en fonction de l'adresse IP du réseau de destination
  - Contient des paires d'adresses (R,P) où R est l'adresse IP d'un réseau de destination et P l'adresse IP de la passerelle du prochain saut
  - Contient une route par défaut

Routage IP

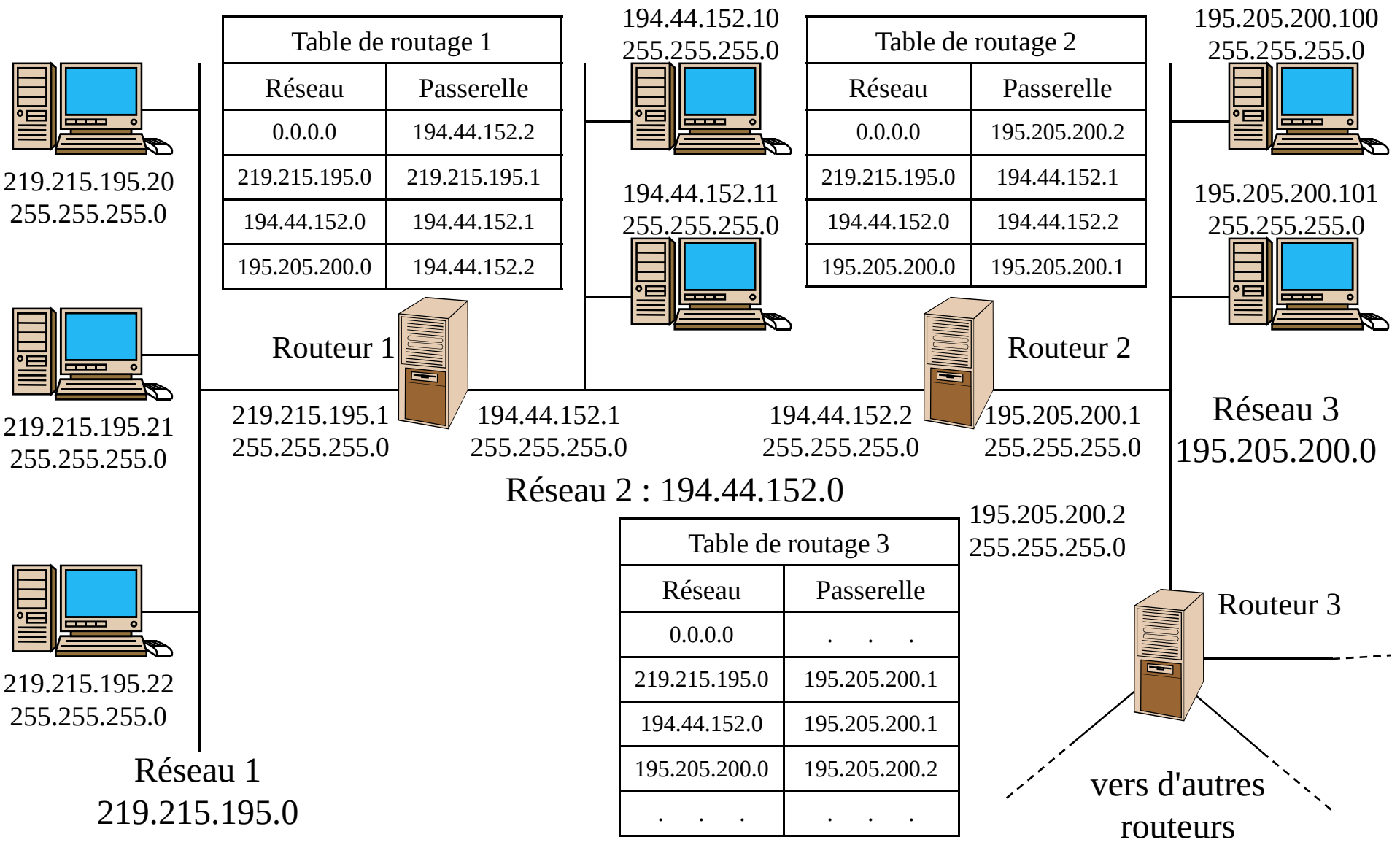


Table de routage 1
Réseau
Passerelle
0.0.0.0
194.44.152.2
219.215.195.0
219.215.195.1
194.44.152.0
194.44.152.1
195.205.200.0
194.44.152.2
Table de routage 2
Réseau
Passerelle
0.0.0.0
195.205.200.2
219.215.195.0
194.44.152.1
194.44.152.0
194.44.152.2
195.205.200.0
195.205.200.1
Table de routage 3
Réseau
Passerelle
0.0.0.0
. . .
219.215.195.0
195.205.200.1
194.44.152.0
195.205.200.1
195.205.200.0
195.205.200.2
. . .
. . .
219.215.195.20
255.255.255.0
219.215.195.21
255.255.255.0
219.215.195.22
255.255.255.0
194.44.152.10
255.255.255.0
194.44.152.11
255.255.255.0
195.205.200.100
255.255.255.0
195.205.200.101
255.255.255.0
Routeur 1
Routeur 2
Routeur 3
219.215.195.1
255.255.255.0
194.44.152.1
255.255.255.0
194.44.152.2
255.255.255.0
195.205.200.1
255.255.255.0
195.205.200.2
255.255.255.0
Réseau 1
219.215.195.0
Réseau 2 : 194.44.152.0
Réseau 3
195.205.200.0
vers d'autres
routeurs

Gestion des erreurs : protocole ICMP

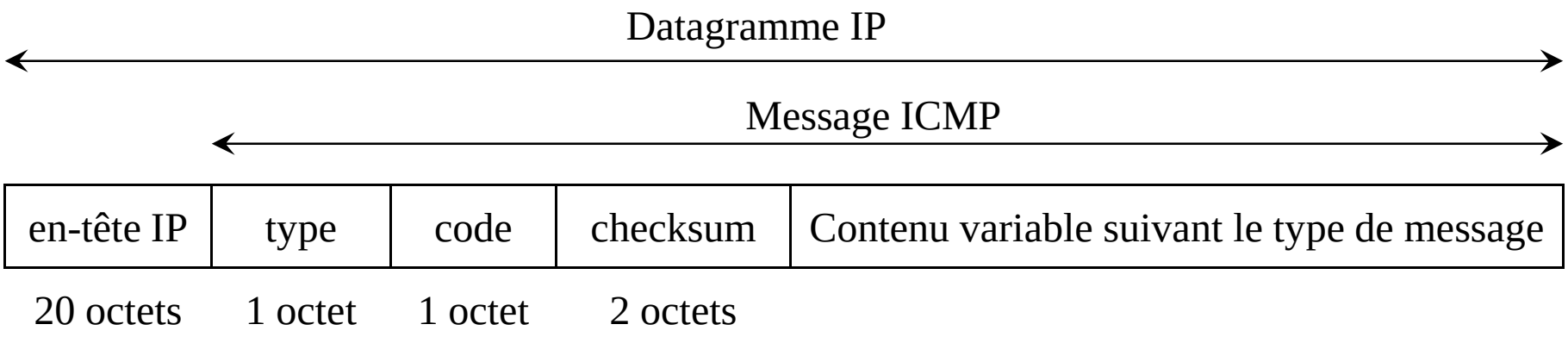
- Protocole ICMP (*Internet Control Message Protocol*)
  - Organise un échange d'informations permettant aux routeurs d'envoyer des messages d'erreurs ou de contrôle à d'autres ordinateurs ou routeurs
  - Bien qu'ICMP fonctionne au-dessus de IP, il est requis dans tous les routeurs, c'est pourquoi on le place dans la couche IP
  - Le but d'ICMP n'est pas de fiabiliser le protocole IP, mais de fournir à une autre couche IP, ou à une couche supérieure de protocole (TCP ou UDP), le compte-rendu d'une erreur détectée dans un routeur
  - Un message ICMP étant acheminé à l'intérieur d'un datagramme IP, il est susceptible, lui aussi, de souffrir d'erreurs de transmission. Mais la règle est qu'aucun message ICMP ne doit être délivré pour signaler une erreur relative à un message ICMP. On évite ainsi une avalanche de messages d'erreurs quand le fonctionnement d'un réseau se détériore

Gestion des erreurs : protocole ICMP

Le protocole ICMP offre une série de services pour les besoins internes des réseaux IP. Le premier d'entre eux est la découverte des routeurs sur chaque segment, réalisée par IRDP (ICMP *Router Discovery Protocol*). Un autre service est la mesure du temps de réponse des messages par l'envoi d'un paquet ECHO_REQUEST puis de l'attente du même paquet ECHO_REPLY en écho. La commande utilisateur qui implémente ceci est appelée ping (pour *Packet Internet Grouper* ou *ping-pong*).

Un autre service important offert par ICMP est la redirection des paquets. Cette fonction permet a un routeur de demander a une station de ne plus envoyer les paquets vers lui, mais vers un autre routeur, et ce a chaque fois qu'il existe une meilleure route. L'intérêt est que les stations n'ont pas besoin d'être dotées d'un logiciel de routage complexe et consommateur de ressources mémoire et CPU.

- Format d'un message ICMP
  - Le champ type peut prendre 15 valeurs différentes spécifiant de quelle nature est le message envoyé
  - Pour certains types, le champ code sert à préciser encore plus le contexte d'émission du message
  - Le champ checksum est une somme de contrôle de tout le message ICMP calculée comme dans le cas de l'en-tête d'un datagramme IP



Types et codes des messages

- Type=0 et 8 : demande (8) et réponse (0) d'écho dans le cadre de la commande ping
- Type=3 : destination inaccessible – message ne peut atteindre sa destination
  - Code=0 : réseau inaccessible – Le réseau n’est pas dans les tables de routage. Le routeur ne sait pas ou emettre le paquet.
  - Code=1 : machine inaccessible – Le réseau est accessible, mais la station n’existe pas sur ce réseau
  - Code=2 : protocole inaccessible
  - Code=3 : port inaccessible
  - Code=4 : fragmentation nécessaire mais bit de non fragmentation positionné à 1
  - Code=5 : échec de routage de source
  - Code=6 : réseau de destination inconnu
  - Code=7 : machine destinataire inconnue
  - Code=8 : machine source isolée (obsolète)
  - Code=9 : communication avec le réseau de destination administrativement interdite

- Type=3
  - Code=10 : communication avec la machine de destination administrativement interdite
  - Code=11 : réseau inaccessible pour ce type de service
  - Code=12 : machine inaccessible pour ce type de service
  - Code=13 : communication administrativement interdite par filtrage
- Type=4 : demande de limitation d'envoi pour éviter la congestion du routeur

  Source Quench – Le récepteur (station ou routeur) indique que l'émetteur doit réduire sa vitesse d'émission tant qu'il reçoit ce message de la passerelle. Puis l'émetteur peut graduellement augmenter sa vitesse d'émission jusqu'à ce qu'il reçoive un nouveau source quench.
- Type=5 : redirection, changement de route
  - Code=0 : redirection pour un réseau
  - Code=1 : redirection pour une machine
  - Code=2 : redirection pour un type de service et réseau
  - Code=3 : redirection pour un type de service et machine

- Type=9 : avertissement de routeur expédié par un routeur
- Type=10 : sollicitation de routeur diffusé par une machine pour initialiser sa table de routage
- Type=11 : TTL détecté à 0
  - Code=0 : durée de vie écoulée avant l'arrivée à destination
  - Code=1 : durée de vie écoulée pendant le réassemblage d'un datagramme
- Type=12 : Mauvais en-tête IP
- Type=13 et 14 : requête (13) ou réponse (14) timestamp, synchronisation horaire
- Type=15 et 16 : requête (15) ou réponse (16) d'information sur l'adresse du réseau
- Type=17 et 18 : requête (17) ou réponse (18) de masque de sous-réseau

Protocoles ARP et RARP

- ARP (*Address Resolution Protocol*)
  - Permet de faire la correspondance entre les adresses logiques (adresses IP) et les adresses physiques (adresses MAC)
- RARP (*Reverse Address Resolution Protocol*)
  - Permet de faire la correspondance entre les adresses physiques (adresses MAC) et les adresses logiques (adresses IP)
- Principe
  - Une machine A veut envoyer un paquet au destinataire B connaissant son adresse IP. Le paquet est encapsulé dans une trame de niveau 2 (Ethernet, FDDI,…)
  - Le module ARP de la machine A envoie une requête ARP dans une trame avec une adresse MAC de diffusion générale braodcast (FF-FF-FF-FF-FF-FF) : toutes les machines du réseau la reçoive
  - La couche ARP de la machine B visée reconnaît que cette requête lui est destinée et répond par une réponse ARP contenant son adresse MAC. Les autres machines ignorent la requête
  - La réponse ARP est reçue par l'émetteur qui peut donc envoyer les paquets avec la bonne adresse MAC de destination

Lors de la réponse, la machine source conserve l'association @IP ↔ @MAC de la requête ARP ⇒ gestion d'une table évitant la répétition de requête ARP

Type de matériel	Type de protocole	Longueur @MAC	Longueur @IP	opération	@MAC source	@IP source	@MAC dest.	@IP dest.
2 octets	2	1	1	2	6	4	6	4

Exemple

- La station 193.157.0.20 envoie la requête ARP à la station 193.157.0.40. L'adresse MAC de destination de la trame au niveau 2 est une adresse de diffusion FF-FF-FF-FF-FF-FF

@MAC source	@IP source	@MAC dest	@IP dest
0080C8D219B3	193.157.0.20	000000000000	193.157.0.40

- La station 193.157.0.40 répond en incluant son adresse MAC

@MAC source	@IP source	@MAC dest	@IP dest
008054B420E5	193.157.0.40	0080C8D219B3	193.157.0.20

Sous Unix la commande arp –a permet de visualiser cette table

```
>arp –a
Station1        192.44.77.1         08:00:20:10:b6:25
Station2        192.44.77.30        08:00:20:03:10:83
>ping station3
Station3 is alive
>arp –a
Station3        192.44.77.32        08:00:20:02:f9:e1
Station1        192.44.77.1         08:00:20:10:b6:25
Station2        192.44.77.30        08:00:20:03:10:83
```

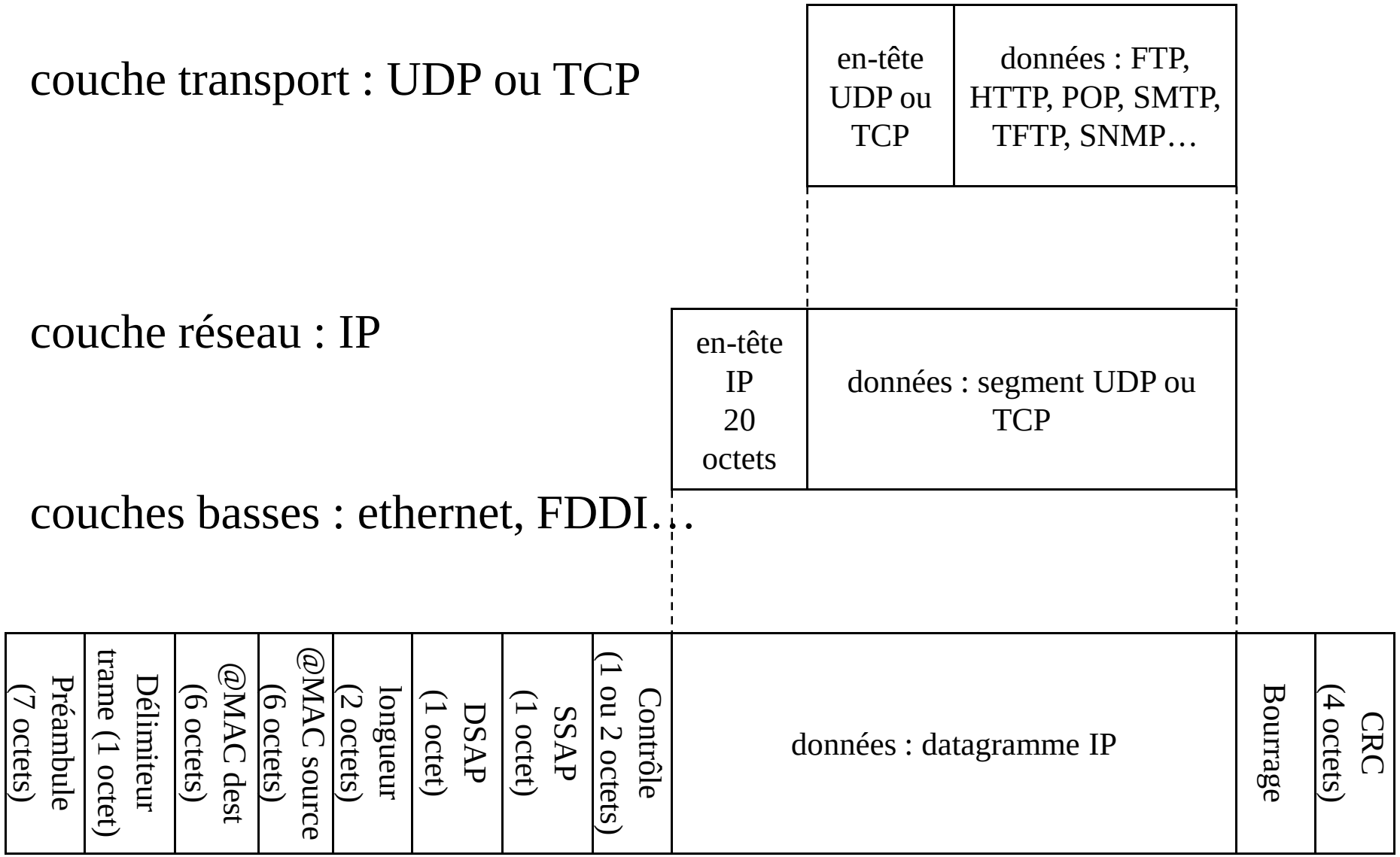
Dans cet exemple, la table contient les adresses pour deux machines station1 et station2. Apres la commande *ping* station3, qui a provoque l'émission de message vers station3, la table contient en plus la correspondance des adresses pour station3.

Encapsulation des données

couche transport : UDP ou TCP

couche réseau : IP

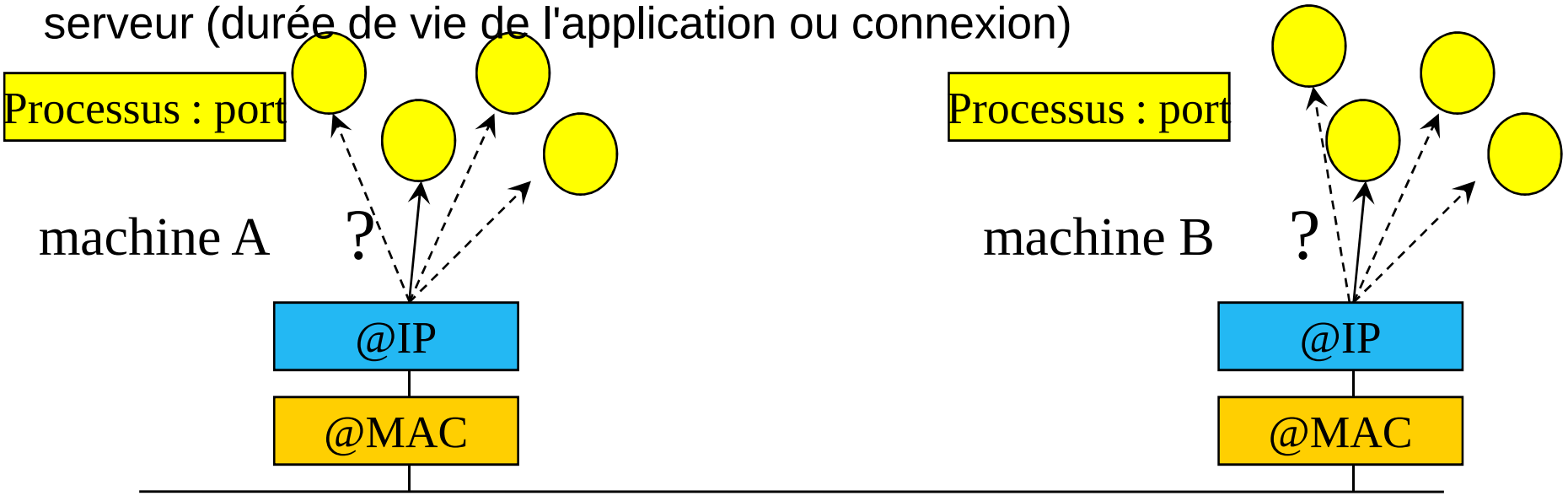
couches basses : ethernet, FDDI…

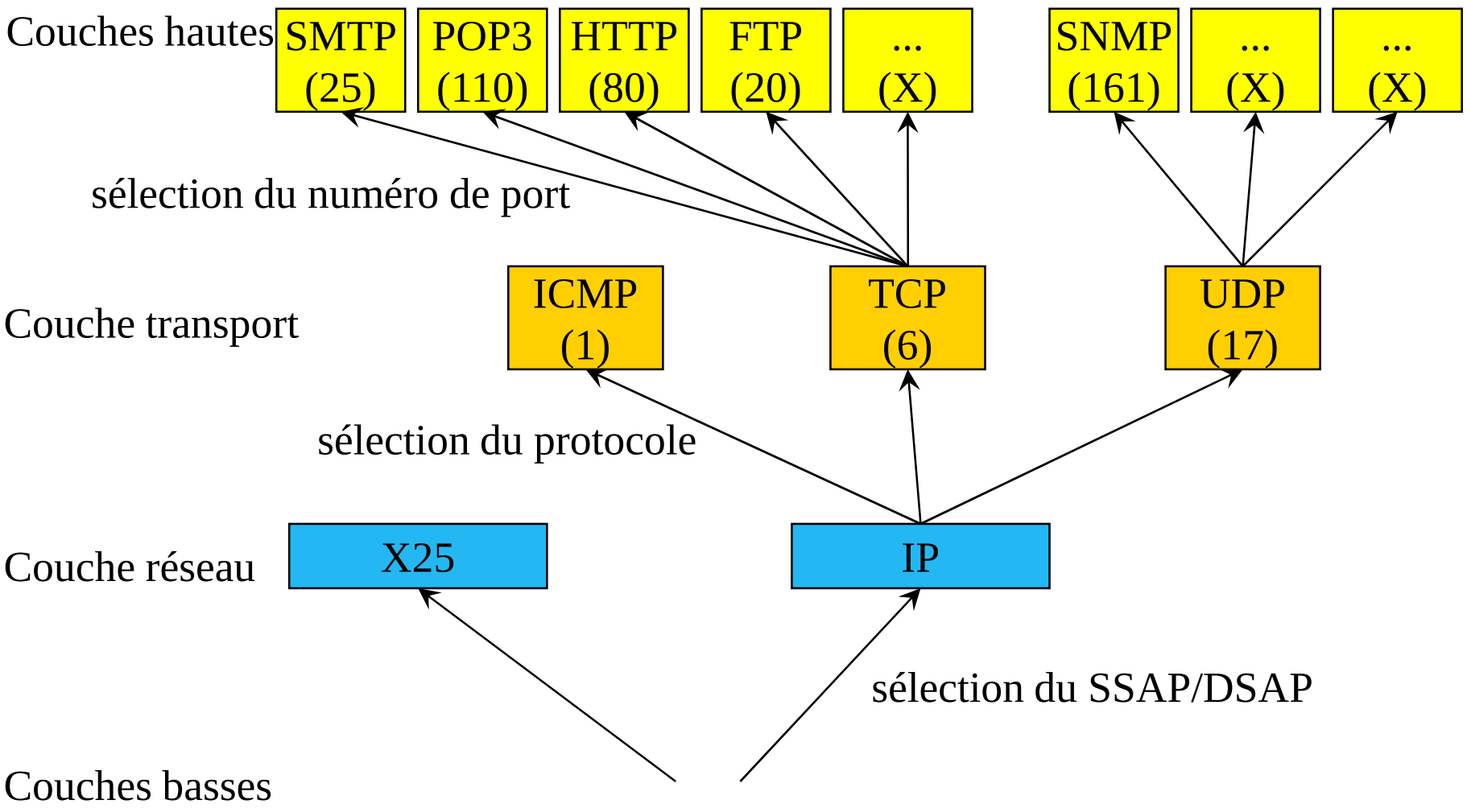


Couche Transport

- Processus désire communiquer avec un autre processus sur une machine distante ⇨ L'adressage de ce processus effectué selon concept abstrait indépendant du système d'exploitation des machines
  - processus créés et détruits dynamiquement sur les machines
  - processus (presque) indépendant des applications locales et distantes
  - il faut identifier les destinations selon les services offerts
  - un processus doit pouvoir assurer plusieurs services
- Ces destinations abstraites permettant d'adresser un service applicatif s'appellent des ports de protocole
- Emission d'un message fait sur la base d'un port source et d'un port destinataire
- Processus disposent d'une interface système leur permettant de spécifier un port et d'y accéder
- Les accès aux ports sont généralement synchrones, les opérations sur les ports sont tamponnées (files d'attente)

- RFC (Request For Comment) numéro 1700 ⇨ liste des numéros de ports attribués aux applications serveurs universellement connues :
  http://www.alternic.org/rfcs/rfc1700/rfc1700.html
  http://www.alternic.org/rfcs/
- Fichier *services* ⇨ défini numéro de port et protocole correspondant utilisé par chaque application serveur sur machine:
  sur Windows 95/98 : \WINDOWS\service
  sur Windows NT4/2000 : \WINNT\system32\drivers\etc\services
  sur Linux/Unix : /etc/services
- Numéros 0 à 1023 ⇨réservés points d'accès universels(durée de vie infinie)
- Numéros > 1024 ⇨ attribués à n'importe quelle application cliente ou serveur (durée de vie de l'application ou connexion)




Couches hautes
SMTP (25)
POP3 (110)
HTTP (80)
FTP (20)
... (X)
SNMP (161)
... (X)
... (X)
sélection du numéro de port
Couche transport
ICMP (1)
TCP (6)
UDP (17)
sélection du protocole
Couche réseau
X25
IP
sélection du SSAP/DSAP
Couches basses

```
# Copyright (c) 1993-1999 Microsoft Corp.
# Ce fichier contient les numéros de port des services les plus connus définis par la RFC 1700
# Format:
# <nom de service>  <numéro de port/<protocole>  [alias...]   [#<commentaire>]

echo        7/tcp
echo        7/udp
…
ftp-data    20/tcp                  #FTP, données (File Transfert Protocol)
ftp         21/tcp                  #FTP, contrôle (File Transfert Protocol)
telnet      23/tcp                  #Prise de contrôle à distance
smtp        25/tcp    mail          #Format SMTP (Simple Mail Transfer Protocol)
…
domain      53/tcp                  #Serveur de nom de domaine (DNS)
domain      53/udp                  #Serveur de nom de domaine (DNS)
…
gopher      70/tcp
finger      79/tcp
http        80/tcp    www www-http  #World Wide Web (HyperText Transfer Protocol)
…
pop2        109/tcp   postoffice    #Post Office Protocol, Protocole Bureau de poste -
Version 2
pop3        110/tcp                 #Protocole Bureau de poste - Version 3
…
nntp        119/tcp   usenet        #Network News Transfer Protocol
ntp         123/udp                 #Protocole d'heure du réseau
…
snmp        161/udp                 #SNMP (Simple Network Management Protocol)
…
irc         194/tcp                 #Protocole IRC (Internet Relay Chat)
…
wins        1512/tcp                #Microsoft Windows Internet Name Service (WINS)
wins        1512/udp                #Microsoft Windows Internet Name Service (WINS)
…
webcache    8080/tcp                #www caching server
```

Par exemple, si la machine A (172.16.16.10) est client, et la machine B (10.155.22.99) est serveur FTP, le processus de communication entre les 2 machines est le suivant:

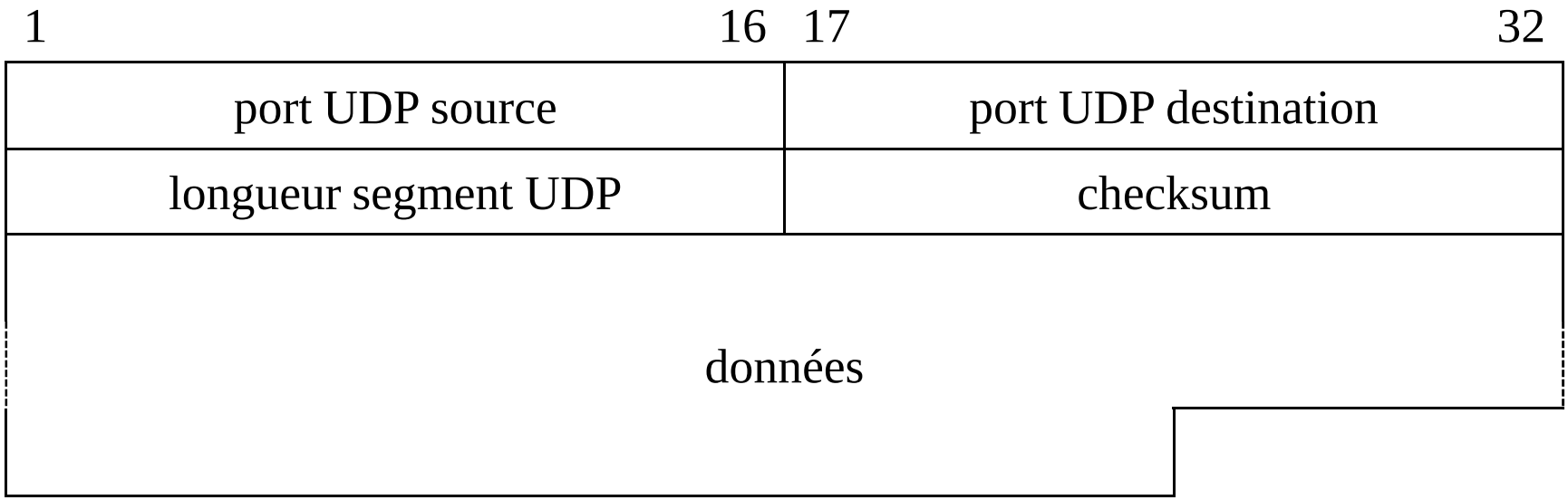
1. Sur A, l'utilisateur tape ftp://10.155.22.99
2. Le système d'exploitation attribue un numéro de Port TCP (supérieur a 1023, par exemple 1029) a l'application ftp
3. TCP crée un paquet contenant les informations suivantes:
   Adresse IP de la source 172.16.16.10
   Numéro de Port TCP de la source 1029
   Adresse de la destination 10.155.22.99
   Numéro de Port TCP de la destination 21
4. Le paquet est envoyé par le réseau a l'adresse IP de B
5. B reçoit le paquet et TCP l'achemine au Port 21 (contrôle FTP)
6. Le service FTP de B retourne un acquittement (ACK) a la machine A (adresse 172.16.16.10, port 1029)

UDP : User Datagram Protocol

- Présentation
- UDP (RFC 768, août 1980) ⇨ protocole de transport sans connexion des services applicatifs
- UDP permet à une application d'envoyer des segments de données vers une autre application avec un minimum de fonctionnalité :
  - contrôle de la validité des données reçues
  - accès à l'application par un numéro de port
- Inconvénients :
  - travaille en mode non connecté ⇨ UDP n'avertit pas l'ordinateur destinataire de l'envoi d'un segment
  - n'utilise pas d'accusé de réception
  - ne re-séquence pas les segments reçus
- Avantages :
  - sa simplicité et sa rapidité

Format du segment UDP

- Constitué :
  - d'un en-tête de taille fixe
  - d'un champ de données de taille variable



- Port UDP source, port UDP destination
  - référence les processus sur machine locale et machine distante
  - UDP multiplexe et dé-multiplexe les datagrammes en sélectionnant les numéros de ports :
    - > une application obtient un numéro de port de la machine locale: dès lors que l'application émet un message via ce port, le champ *port source* du segment UDP contient ce numéro de port
    - > *port source* spécifie aussi le numéro de port utilisé lors de la réponse. S'il n'est pas utilisé, il vaut 0
    - > une application connaît le numéro de port distant afin de communiquer avec le service désiré
    - > lorsque UDP reçoit un segment ⇨ vérifie que celui-ci est pour un des ports actuellement actifs (associé à une application)
    - > délivre le segment à l'application responsable (mise en queue)
    - > si le port est pas actif ⇨ émission d'un message ICMP *port unreachable*, et détruit le datagramme
- Longueur segment UDP
  - longueur totale du segment en octets (entête + données)
  - longueur < 65535 octets
- checksum : contrôle d'erreur

UDP : conclusion

- Protocole simple :
  - ✓ surcoût minimal pour les paquets UDP
  - ✓ surcoût minimal pour le traitement du protocole :
    - pas de contexte
    - très peu de contrôle (détection d'erreur optionnelle)
  - ✓ sans (avec très peu d') augmentation de service :
    - le service fourni est le service disponible
    - multiplexage (nº port)

Mais protocole non-fiable !

TCP : Transmission Control Protocol

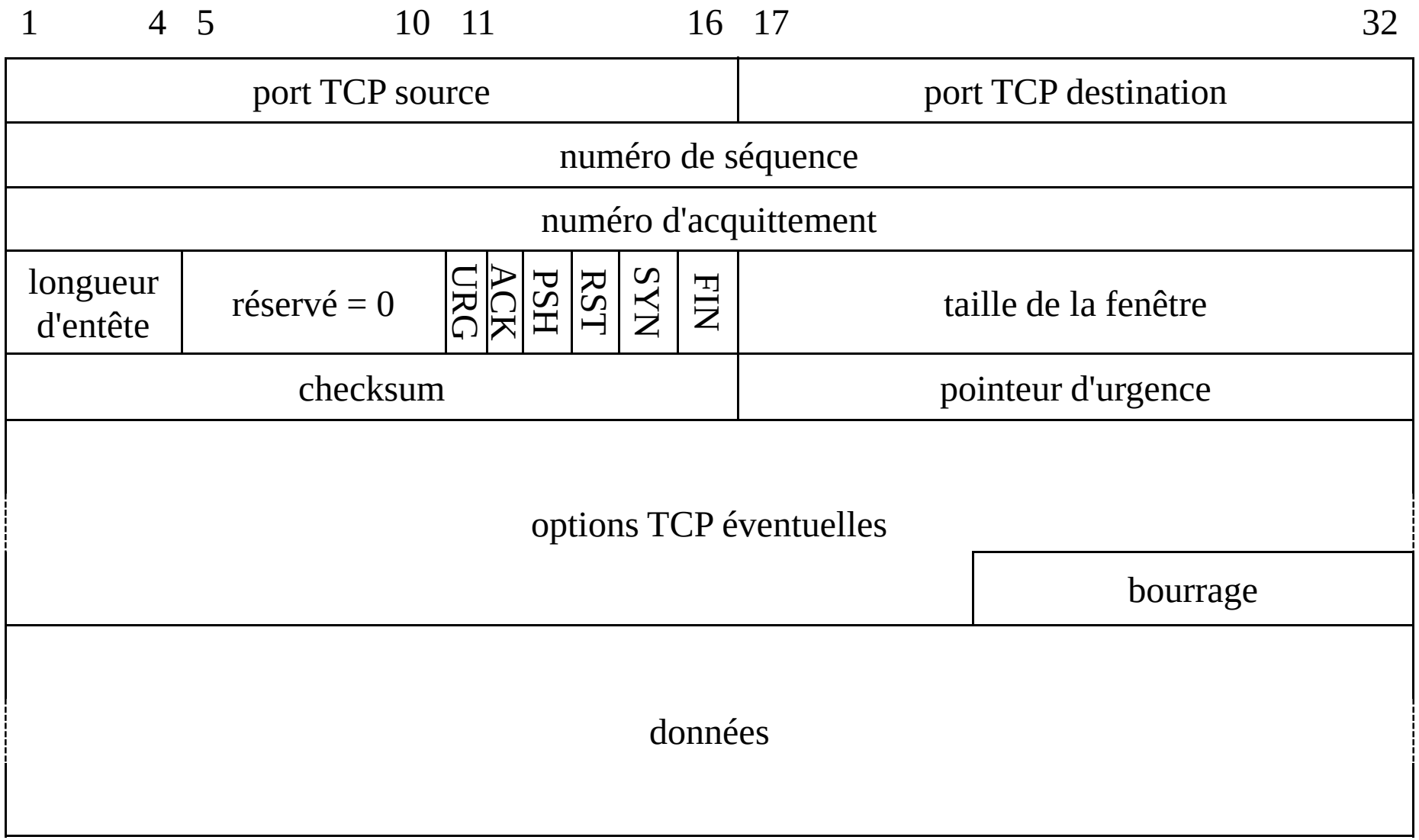
RFC 793, septembre 1981

- transport fiable des données entre 2 machines
- service en mode connecté
  - ✓ établissement de la connexion virtuelle
  - ✓ transfert des données
  - ✓ fermeture de la connexion virtuelle
- connexions bidirectionnelles et simultanées
- transferts tamponnés : découpage en segments de tailles variables
  - ✓ flux non structuré de données (*stream* : suite d'octets)
  - ✓ segmentation et ré-assemblage des segments
- Fiabilité
  - ✓ mécanisme d'acquittement des segments
  - ✓ retransmission en cas d'erreur (données endommagées, perdues, dupliquées)
  - ✓ contrôle de flux
  - ✓ re-séquencement des datagrammes/segments avec IP
  - ✓ gestion des priorités des données et de la sécurité de la communication

Segment TCP

- Le segment est l'unité de transfert du protocole TCP. Il est utilisé pour :
  - ✓ établir les connexions
  - ✓ transférer les données
  - ✓ émettre des acquittements
  - ✓ fermer les connexions
- Formaté sur 4 octets :
  - ✓ en-tête de taille fixe
  - ✓ en-tête optionnel de taille variable
  - ✓ champ de données de taille variable

Format du segment TCP



- numéros de port source et port destination
  - ✓référence les processus sur les machines locale et distante
  - ✓<@IP source, N° port source> + <@IP destination, N° port destination>
  - ✓l'association <@IP, N° port> est appelée *socket*
- numéro de séquence
  - ✓indique le numéro du 1er octet transmis dans le segment (modulo 2^{32})
  - ✓initialisation du numéro de séquence par ISN
- numéro d'acquittement
  - ✓contient le numéro de séquence NP du prochain octet attendu ⇨acquitte implicitement les octets NP-1, NP-2, …
  - ✓une fois la connexion établie, cette valeur est toujours transmise
- longueur d'en-tête
  - ✓indique en nombre de mots de 4 octets (32 bits) la taille de l'en-tête
  - ✓taille de l'en-tête fixe = 20 octets (longueur = 5), taille maximale de l'en-tête = 60 octets (longueur = 15)

- taille de la fenêtre
  - fixée par le destinataire avant l'envoi du 1er segment ⇨fonction taille de sa mémoire tampon de réception ⇨ indique nombre d'octets capable de recevoir
  - ceci est mentionné dans chaque segment (données et acquittement)
  - permet d'envoyer un certain nombre d'octets par anticipation (sans attendre un acquittement) ⇨ permet de gérer le contrôle de flux
- URG, ACK, PSH, RST, SYN et FIN indiquent la nature du segment et la validité de certains champs
- URG (urgent bit)
  - le pointeur de données urgentes est valide
  - les données sont émises sans délai, les données reçues sont remises sans délai à la couche supérieure
- ACK (acknowledgement bit)
  - la valeur du champ numéro d'acquittement est prise en compte
- PSH (push bit)
  - normalement, en émission, TCP reçoit les données depuis l'applicatif (la couche supérieure), les transforme en segments à sa guise, puis transfère les segments sur le réseau. Un récepteur TCP décodant le bit PSH à 1, transmet immédiatement les données reçues à l'application réceptrice (la couche supérieure), sans attendre plus de données de l'émetteur ⇒ fin d'un message

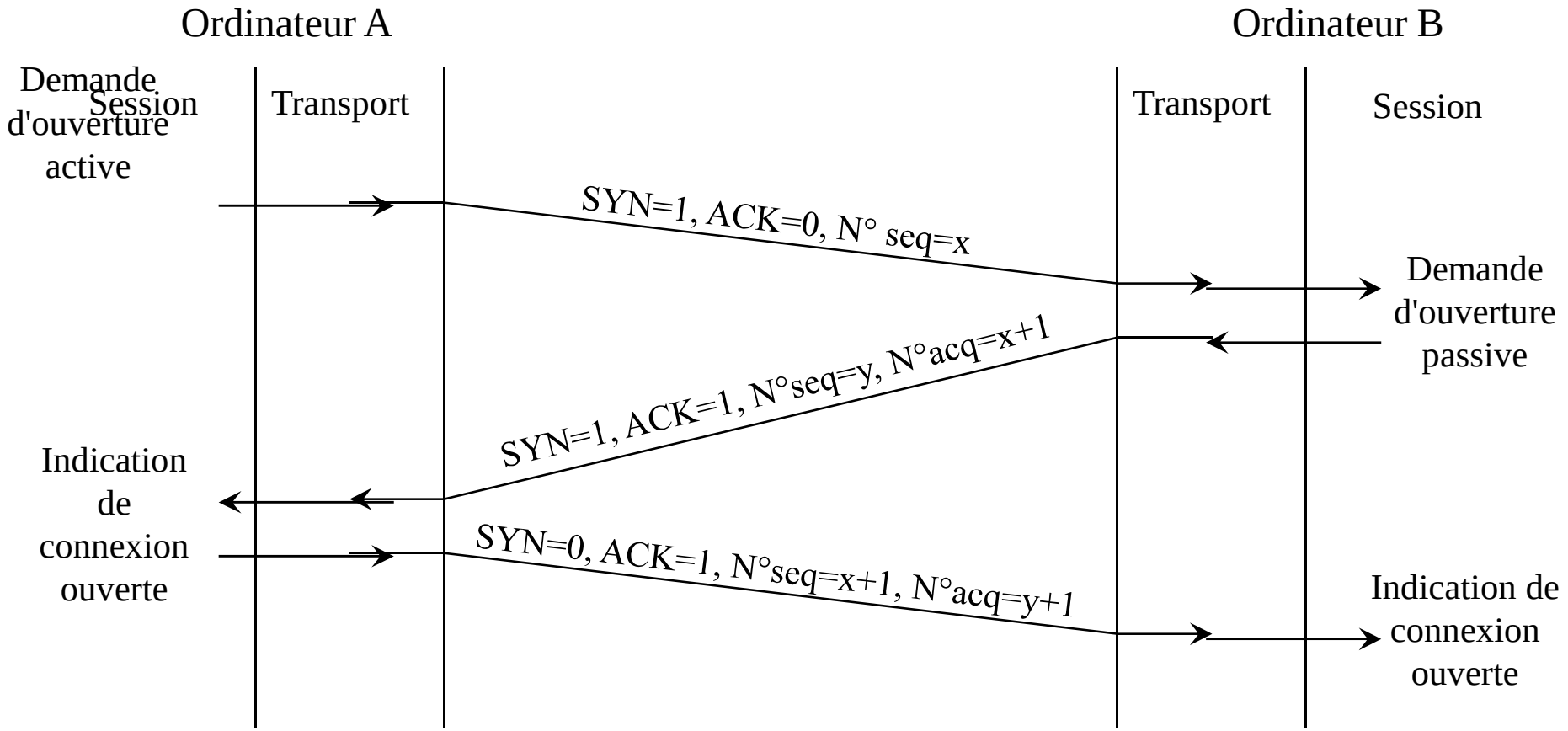
- RST (reset bit)
  - utilisé par une extrémité pour indiquer à l'autre extrémité qu'elle doit réinitialiser la connexion suite à une erreur (désynchronisation)
- SYN (synchronise bit)
  - utilisé à l'ouverture de la connexion - indique où commence la numérotation - SYN occupe lui-même un numéro de séquence
- FIN (final bit)
  - demande la libération de la connexion
- checksum : contrôle d'erreur
- pointeur d'urgence
  - indique les octets devant être traités en priorité lorsque URG est à 1
  - les données sont transmises hors du contrôle de flux
  - permet de transmettre des données sans retard, devant être traitées de manière urgente (par exemple : annulation d'un transfert de fichier)
  - le champ *pointeur d'urgence* indique le numéro du dernier octet de la partie urgente du segment de données qui commence au début du champ de données du segment

Ouverture d'une connexion TCP

- connexion TCP
  - ✓ 1 connexion = 1 paire d'extrémités de connexion
  - ✓ 1 extrémité de connexion = 1 couple (@ IP, numéro de port)
  - ✓ exemple de connexion TCP : ((194.199.53.1, 1034), (19.24.67.2, 21))
  - ✓ une extrémité de connexion peut être partagée par plusieurs autres extrémités de connexions (par exemple : serveur ftp)

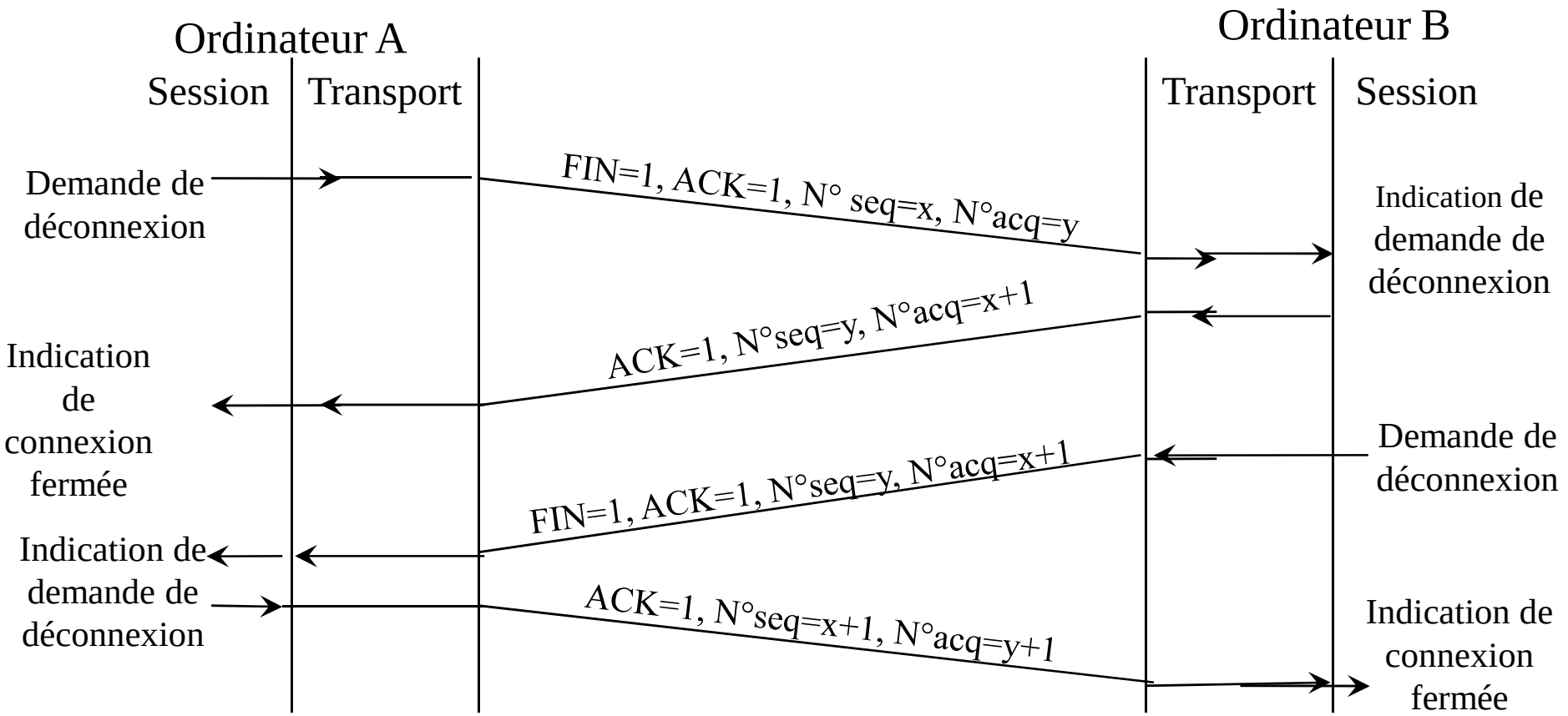
Ouverture d'une connexion TCP

- connexion TCP de type circuit virtuel - établie en trois temps de manière à assurer la synchronisation nécessaire entre les extrémités
  - ✓demande d'ouverture active, initialisation du numéro de séquence
  - ✓acceptation/refus de l'établissement, double établissement simultané
  - ✓indication de connexion ouverte



Fermeture d'une connexion TCP

- Libération de la connexion
  - ✓réalisée lorsque le récepteur reçoit un en-tête TCP avec un bit FIN positionné à 1
  - ✓en trois temps modifiés dans chaque sens indépendamment ⇒ 2 doubles échanges

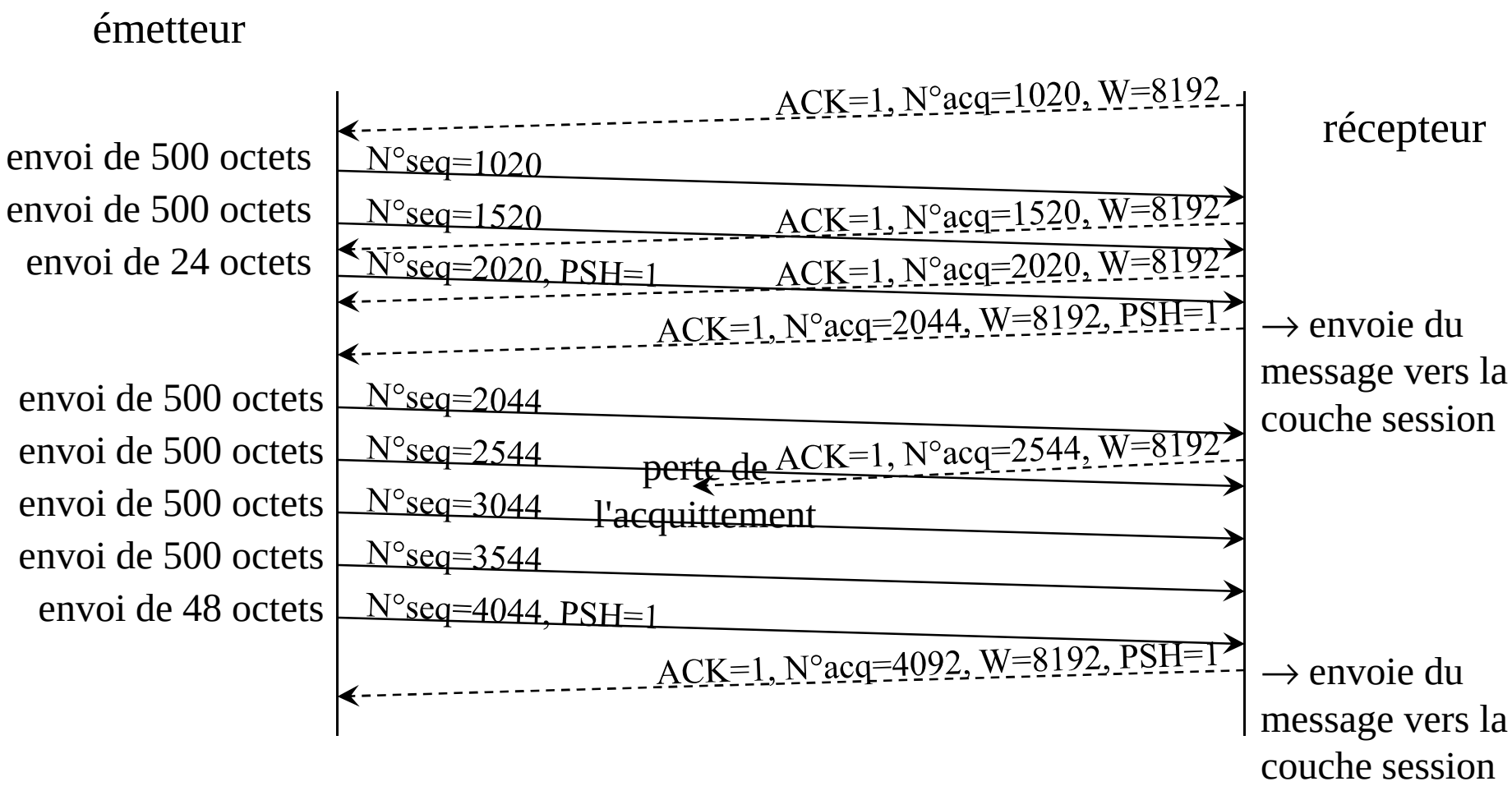


Transfert et acquittement des données

- Après ouverture d'une connexion TCP, les données peuvent être transférées
  - ✓ le numéro de séquence indique le numéro du 1er octet de données du segment
  - ✓ le contrôle de flux est réalisé dans les 2 sens par les numéros d'acquittement (le bit ACK doit être mis à 1) et la taille de la fenêtre
  - ✓ le contrôle de perte des segments est réalisé par un temporisateur
- Acquittement TCP
  - ✓ L'acquittement est pris en compte par le champ *numéro d'acquittement* lorsque le bit ACK est positionné à 1
  - ✓ Les acquittements sont cumulatifs : ils acquittent tous les octets précédents
  - ✓ Nul n'est besoin d'envoyer systématiquement un acquittement
  - ✓ La perte d'un acquittement ne nécessite pas forcément une ré-émission

Acquittement TCP

émission d'un message de 1024 octets, puis d'un message de 2048 octets, par segments de 500 octets maximum, taille de la fenêtre = 8192 octets



Segmentation et contrôle de flux

- Segmentation
  - ✓ les données transmises à TCP constituent un flot d'octets de longueur variable
  - ✓ TCP divise ce flot de données en segments en utilisant un mécanisme de fenêtrage (taille de fenêtre) et de taille de segment
  - ✓ un segment est encapsulé dans un datagramme IP
- Contrôle de flux
  - ✓ au niveau du récepteur :
    - nombre d'octets pouvant être stockés par le récepteur (mémoire tampon) contrôlé par la taille de la fenêtre : nombre maximum d'octets pouvant être émis par l'émetteur sans avoir reçu d'acquittement du récepteur
    - espace de stockage du récepteur presque plein ⇒ diminution de la taille de la fenêtre
    - espace de stockage du récepteur presque vide ⇒ augmentation de la taille de la fenêtre

Contrôle de flux au niveau de l'émetteur :

envoi des octets par segment de taille variable à l'intérieur de la fenêtre d'émission

taille d'un segment = min (taille de la fenêtre, taille maximale du segment = MSS)

Par conséquent :

La fenêtre d'émission est une fenêtre glissante à taille variable

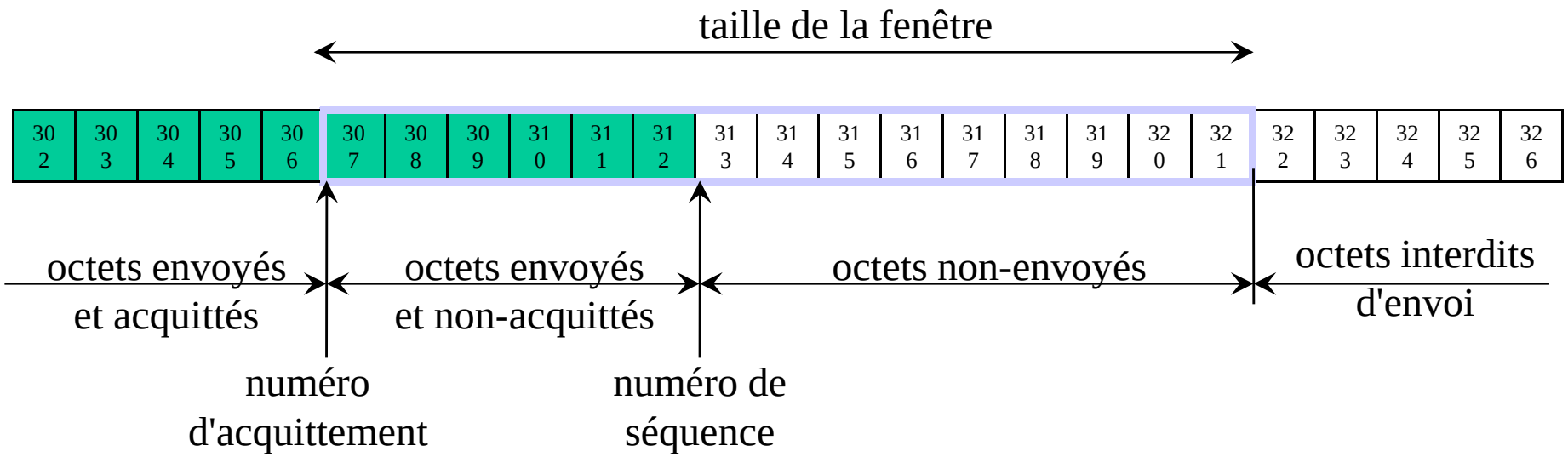
Le mécanisme de fenêtre opère au niveau de l'octet et non pas au niveau du segment

numérotation séquentielle des octets de données

gestion de trois pointeurs par fenêtre

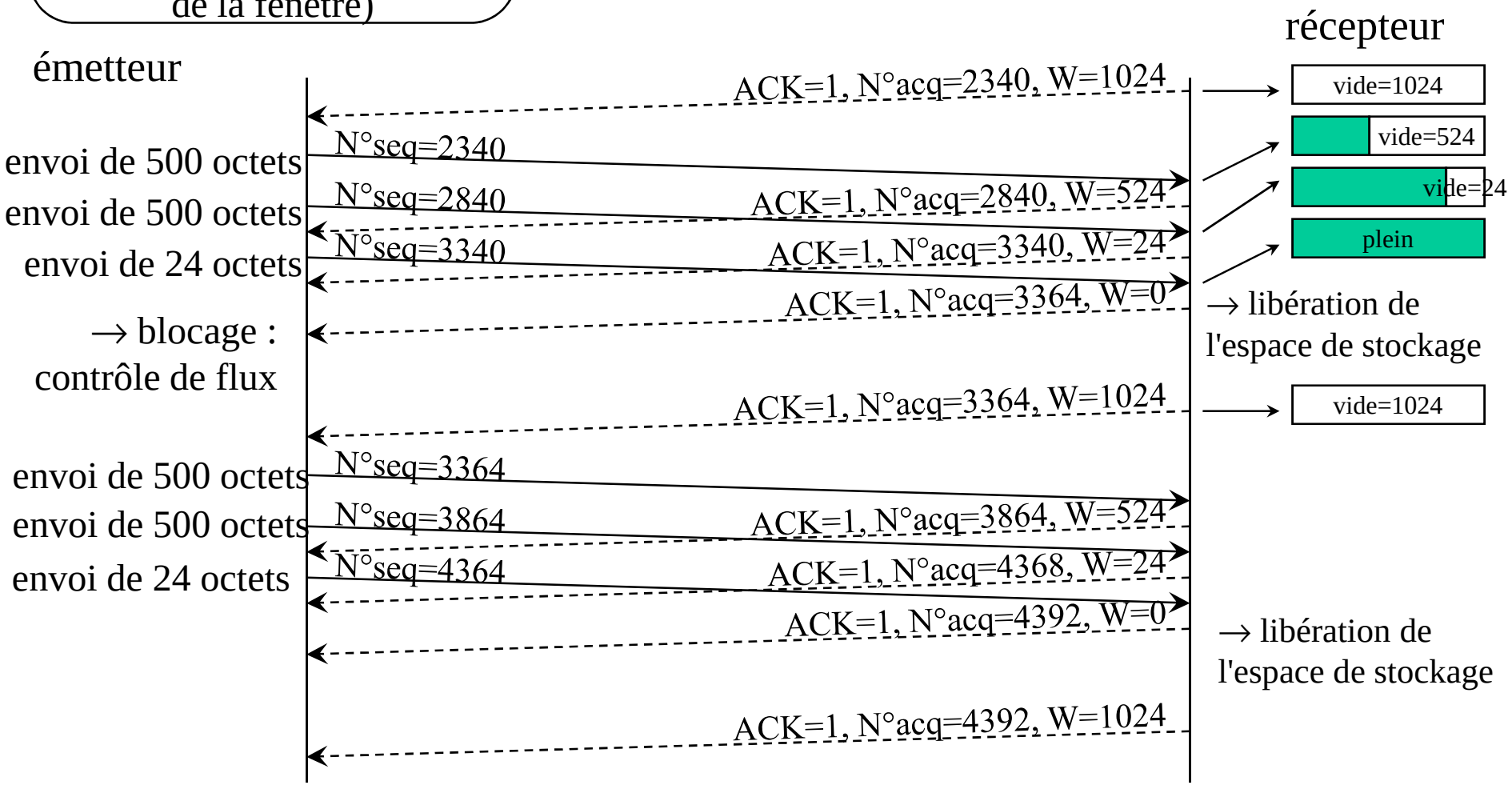
deux fenêtres indépendantes (une pour chaque sens de transmission des données)

pour chacune des deux fenêtres, chaque partenaire possède une copie de chaque variable permettant de gérer localement cette fenêtre



Contrôle de flux

émission d'un message de 5000 octets, par segments de 500 octets maximum, espace de stockage disponible au récepteur = 1024 octets (taille de la fenêtre)



Acquittements et retransmission : gestion des erreurs

Le mécanisme d'acquittement de TCP est cumulatif.

- si un segment a un numéro de séquence supérieur au numéro de séquence attendu (bien que dans la fenêtre), le segment est conservé mais l'acquittement référence toujours le numéro de séquence attendu

Pour tout segment émis, TCP s'attend à recevoir un acquittement

- contrairement à UDP, TCP garantit l'arrivée des messages ⇒ en cas de perte, les deux extrémités se préviennent
- ce concept repose sur les techniques d'acquittement de message : lorsqu'une source émet un segment vers une destination, la source attend un acquittement de la destination. Si l'acquittement ne parvient pas à la source, celle-ci considère au bout d'un certain temps que le segment est perdu et le re-émet
- or, un réseau d'interconnexion offre des temps de transit variables nécessitant le réglage des temporisations :
  - TCP gère des temporisations variables pour chaque connexion en utilisant un algorithme de retransmission adaptative

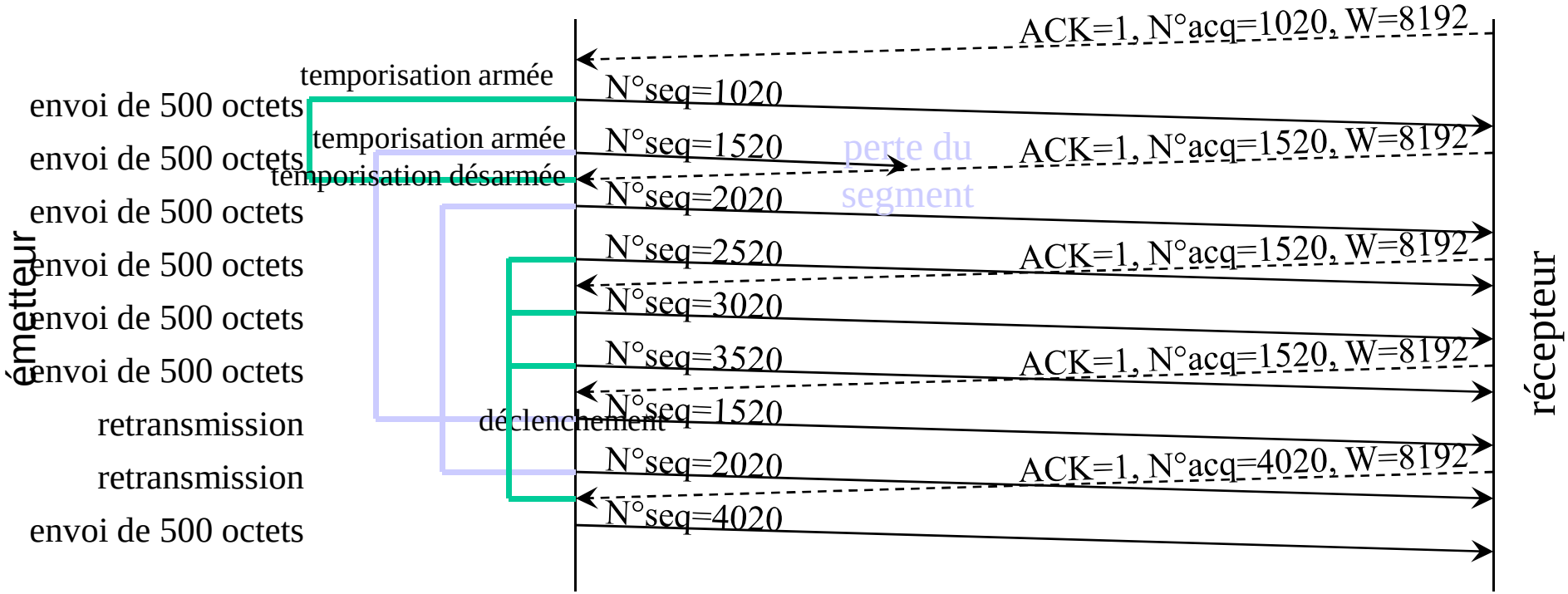
Acquittements et retransmission

Détection des pertes : à l'émetteur, par un temporisateur

moins efficace (mais plus sûr) que par acquittement explicite (négatif) en provenance du récepteur

Récupération des pertes : par retransmission (depuis le segment perdu)

Les segments corrompus sont détruits, cela se traduit par des pertes ! Le mécanisme normal de contrôle des erreurs permet leur retransmission



Gestion de la congestion

- Contrôle de congestion = Contrôle de flux des routeurs
- TCP gère le contrôle de flux de bout en bout mais également les problèmes de congestion liés à l'interconnexion
  - la congestion correspond à la saturation de nœud(s) dans le réseau provoquant des délais d'acheminement de datagrammes jusqu'à leur pertes éventuelles
  - les extrémités ignorent tout de la congestion sauf les délais. Habituellement, les protocoles retransmettent les segments ce qui aggrave encore le phénomène
- Dans la technologie TCP/IP, les passerelles (niveau IP) utilisent la réduction du débit de la source mais TCP participe également à la gestion de la congestion en diminuant le débit lorsque les délais s'allongent

TCP : conclusion

- Protocole offrant un service évolué assurant une transmission :
  - ✓ de bonne qualité
  - ✓ en mode connecté
  - ✓ avec un service de multiplexage (ports)
- ✓ Basé sur le mécanisme de la fenêtre glissante/coulissante, permettant à la fois :
  - ✓ une utilisation optimale de la liaison sous-jacente (physique)
  - ✓ la protection contre les erreurs (détection + correction)
  - ✓ le contrôle des duplications, des pertes, de la séquentialité
  - ✓ le contrôle de flux et de congestion
- ✓ Protocole aux mécanismes complexes entraînant un certain surcoût tant au niveau des traitements que pour les entêtes des segments. Ses performances sont toutefois très correctes, car optimisées

Le protocole Telnet

Il permet d'obtenir un service de communication en autorisant un utilisateur (client) à se connecter à un hôte distant (serveur), ainsi qu'à exécuter des commandes (souvent des commandes UNIX) à partir de ce dernier. On peut donc se connecter à une machine situé n'importe où dans le monde (à condition qu'elle soit configurée pour accepter les sessions Telnet) à partir de chez nous. Telnet reste cependant une interface textuelle.
La communication est établi par un protocole TCP/IP, et est basé sur un système nommé Network Virtual Terminal (NVT).
Pour exécuter telnet, il suffit d'exécuter la commande qui lui permet de démarrer son client Telnet, suivi de l'adresse à contacter (et éventuellement du port), on lui demande ensuite son login (Nom d'utilisateur) et son password (Mot de passe).
Exemple: "telnet 100.100.150.1"
Telnet est inclus de façon native sous UNIX et Windows XX.

ssh

ssh - secure shell client (remote login program)

L'idée:

- **chaque machine a une bi-clés (Clé publique/Clé privée)**
- **échange à l'aide des bi-clés d'une clef de session**
- **codage de la session à l'aide de la clef négociée**

FTP
(File Transfer Protocol)

FTP est le protocole qui définit les transferts de donnés sur un réseau. On l'utilise en général directement depuis un terminal. Les objectifs de ce protocole sont de permettre un partage de fichiers ou programmes sur des machines distantes, de permettre des modifications à distance sur des fichiers, et de transférer des données via un réseau. Ce protocole a été mis en pace dès 1971.

DTP - Data Transfer System. C'est le processus qui établit et gère la connexion pour les données. Il peut être soit actif, soit passif.

PI - Protocol Interpreter. Les parties clients et serveur ont des rôles différents. Le serveur écoute sur le port 21, attend des connections de clients, puis établit une connexion de contrôle de la communication. Il reçoit les commandes **FTP** standard émises par le client **PI**, renvoie des réponses, et dirige le serveur **DTP**.
La partie client initialise la connexion de contrôle entre lui et le serveur **FTP**, envoie les commandes **FTP**, et dirige le client **DTP** si ce procès fait partie du transfert de fichier demandé.

?	Access the Help screen.		**abor**	Abort Transfer
ascii	Switch to ASCII transfer mode		**binary**	Switches to binary transfer mode.
bye	Exits from FTP.		**cd**	Changes directory.
cdup	Change to parent directory on remote system		**close**	Exits from FTP.
delete	Deletes a file.		**dir**	Lists files if connected.
get	Get file from the computer connected to.		**hash**	Sets hash mark printing on / off
help	Access the Help screen and displays information about command if command typed after help.		**lcd**	Displays local directory or if path typed after lcd will change local directory.
ls	Lists files if connected.		**mdelete**	Multiple delete
mdir	Lists contents of multiple remote directories		**mget**	Get multiple files
mkdir	Make directory.		**mls**	Lists contents of multiple remote directories.
mput	Sent multiple files		**open**	Opens address.
pass	Supplies a user password.		**port**	Specify the client port number.
prompt	Enables disables prompt.		**put**	Send one file
pwd	Print working directory		**quit**	Exits from FTP.
rename	Renames a file		**rmdir**	Removes a directory
send	Send single file		**status**	Shows status of currently enabled / disabled options
trace	Toggles packet tracing		**user**	Send new user information