



**Université Paris Descartes**  
**UFR de mathématiques et Informatique**

LICENCE MIA  
3<sup>ème</sup> Année – Semestre 5

Polycopié de cours et de Travaux dirigés

# **RESEAUX**

## **MLI536**

Responsable pédagogique :

**Pr. Ahmed MEHAOUA**



**Auteur(s):** Dominique Seret, Ahmed Mehaoua

Droits de propriété intellectuelle : UFR Mathématiques et Informatique de l'Université Paris Descartes

Dernière modification: 09/09/2013

**Pré-requis :** notion de bases en architecture des ordinateurs et les systèmes d'exploitation

**Description du module :**

- *Volume horaire* : 18h heures de CM, et 24h de TD/TP
- *Objectif général* : comprendre rapidement l'environnement réseau de l'entreprise
- *Objectifs d'apprentissage* :

- § savoir identifier les natures et modes de transmissions
- § savoir situer les différents équipements de l'architecture d'un réseau
- § savoir reconnaître les protocoles de communication utilisés
- § comprendre les services réseau implémentés et les utiliser
- § comprendre les éléments de configuration d'un réseau

- **Résumé** : ce cours présente de manière très progressive les éléments de réseau et leurs architectures. Il décrit les principes de base et s'attache à présenter les solutions les plus fréquentes d'Ethernet à Internet.

- **Mots clés** : Protocoles, Ethernet, Internet, TCP/IP, téléphonie, interconnexion.

**Contact** : ahmed.mehaoua@parisdescartes.fr





Université Paris Descartes  
UFR de mathématiques et Informatique

# **Transparents de cours Réseaux MLI536**

# Cours : Réseaux

## Objectifs de ce cours :

1. Etudier et comprendre le fonctionnement des réseaux informatiques
2. Etudier le fonctionnement et la configuration d'un réseau local **ETHERNET** (cablage, codage des signaux, contrôle des accès au canal de communication)
3. Etudier le fonctionnement d'un grand réseau, **L'INTERNET** (adressage, routage, interconnexion)
4. Utiliser des logiciels de diagnostics et d'analyses de réseaux (sniffer), **WIRESHARK**

## Bonnes pratiques de ce cours :

- récupérer le support du cours et du TD/TP sur MODDLE **avant les séances** et l'étudier
- Consulter le chapitre du livre de référence avant le cours (préparer vos questions)
- participer activement aux séances de TD

## Planning du cours :

# Plan Général

- 1) ARCHITECTURES DES RESEAUX, DEFINITIONS**
- 2) MATERIELS, TRANSMISSION, COUCHE PHYSIQUE**
- 3) LOGICIELS, PROTOCOLES HDLC, COUCHE LIAISON**
- 4) ETHERNET: LES RESEAUX LOCAUX**
- 5) INTERNET: ADRESSAGE, NOMMAGE DES RESSOURCES**
- 6) INTERNET: ROUTAGE DES INFORMATIONS**
- 7) LES EQUIPEMENTS D'INTERCONNEXION (HUB, SWITCH, GATEWAY, ...)**
- 8) LES RESEAUX TELEPHONIQUES**

# **Réseaux Informatiques**

## **Architectures et**

## **Définitions**

Page 5

© A. Mehaoua

# **Plan**

- ❑ DEFINITIONS ET PRINCIPES DE BASE**
- ❑ CLASSIFICATION DES RESEAUX**
- ❑ NORMES ET STANDARDS**
- ❑ HIERARCHIE DES PROTOCOLES**
- ❑ PRINCIPES DE LA COUCHE PHYSIQUE**
- ❑ TYPES D'INFOS ET CODAGE SOURCE**
- ❑ TECHNIQUES DE TRANSMISSION**

Page 6

© A. Mehaoua

## Historique technologique

- 1876: Téléphonie (Graham Bell), 1880 en France
- 1906: Radiodiffusion (Branly, Ducret, Marconi)
- 1930: La télévision
- 1969: Arpanet, 1<sup>er</sup> réseau informatique
- Apparition du transistor dans les années 50
- Numérisation des communications téléphoniques 1970
- 1980: réseau Numéris, intégration de la voix et des données informatiques
- Numérisation de la télévision
  - 1994 MPEG Motion Picture Expert Group (codage source)
    - ✓ Représentation numérique et compression de l'information audiovisuelle
  - 1995 DVB-S (Satellite) Digital Video Broadcasting (codage canal)
    - ✓ Transmission numérique de l'information audiovisuelle
    - ✓ 2001 : DVB-T (Terrestre)



© A. Mehaoua

Page 7

## Qu'est ce qu'un réseau de communication ?

Un ensemble des ressources matériels (modem, routeur, commutateur, câblage, cartes, ...) et logiciels (procédures, règles, protocoles, systèmes d'exploitation, ...) associés à la transmission et l'échange d'information entre différentes entités (ordinateurs, individus, périphériques, processus, ...).

Les réseaux font l'objet d'un certain nombre de spécifications et de normes pour garantir leurs inter-fonctionnement.

# Classification des Réseaux de Communication

## - type d'informations -

- ♦ Les **réseaux de communications** peuvent donc être classés en fonction du **type d'informations** transportées et de la **nature des entités** impliquées. On distingue ainsi trois principales catégories de réseaux :
  - Les réseaux de **télécommunications**
  - Les réseaux de **télédiffusion**
  - Les réseaux **Téléinformatiques**

## La télé-informatique

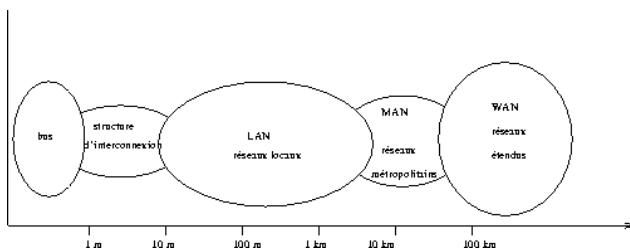


- en 1957 Seymour Cray invente la société CDC et le 1<sup>er</sup> calculateur
- En 1964, Kleinrock du MIT invente la commutation de paquets
- Le réseau ARPANET apparaît en 1969
- Email : 1<sup>ère</sup> application ARPANET inventée par le MIT en 1972
- En 1976, TCP/IP intégré dans ARPANET
- En 1979, Metcalf invente Ethernet et quitte Xerox pour créer 3Com
- Mai 1982 : 235 machines connectées sur Internet

# Classification Des Réseaux de communication

## - Dimension -

- **Bus des ordinateurs** ISA, PCI
- **Réseaux personnels (PAN)** Bluetooth
- **Réseaux locaux (LAN)** Ethernet, WiFi
- **Réseaux métropolitains (MAN)** Gigabit Ethernet,
- **Réseaux étendus (WAN)** Numéris, Internet, GSM, Satellites

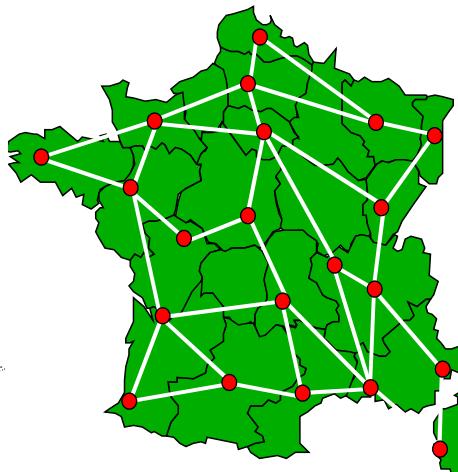
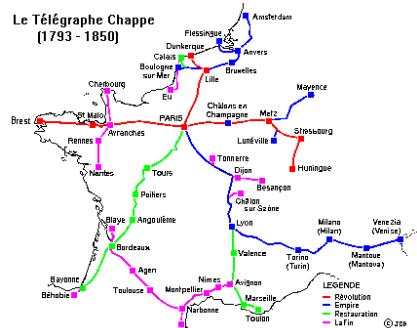


# La problematique des réseaux téléinformatiques

- Comment faire communiquer les ordinateurs/processus sur une seule ligne ?
- La solution
  - Coder les données et les informations de contrôle (logique à deux états)
  - Les transmettre sur la même ligne
- Les protocoles
  - Règles de codage des informations
  - Règles de dialogue entre ordinateurs
    - Gérés par les logiciels et matériels de communication
- Les architectures
  - Cadres d'environnement et de définition des protocoles
  - Ensemble de protocoles, procédures et équipements de communications
  - Permettre l'interconnexion des réseaux hétérogènes aux moyens de dispositifs de conversion

## Architecture matérielle d'un réseau d'opérateur

- **POP**
  - Points de(Of) Présence (équipements commutateur, routeur, multiplexeur)
- **Raccordement des utilisateurs sur les POP**
  - Via la boucle locale (cuivre)
- **Interconnexion des POP**
  - Réseau maillé
  - Fibres optiques

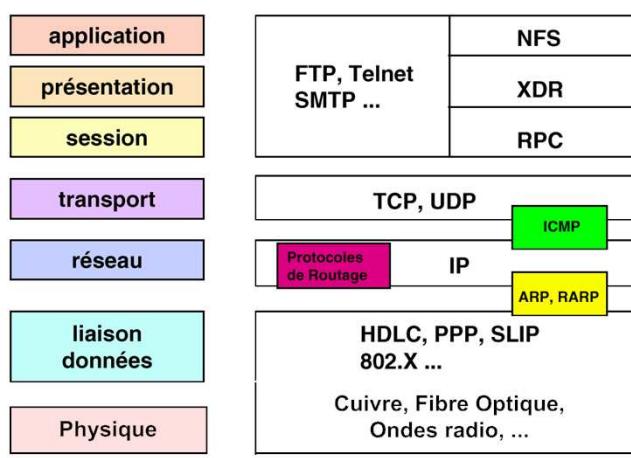


© A. Mehaoua

Page 13

## Exemples d'architectures logicielles:

le modèle ISO (1982) vs le modèle Internet (1969)

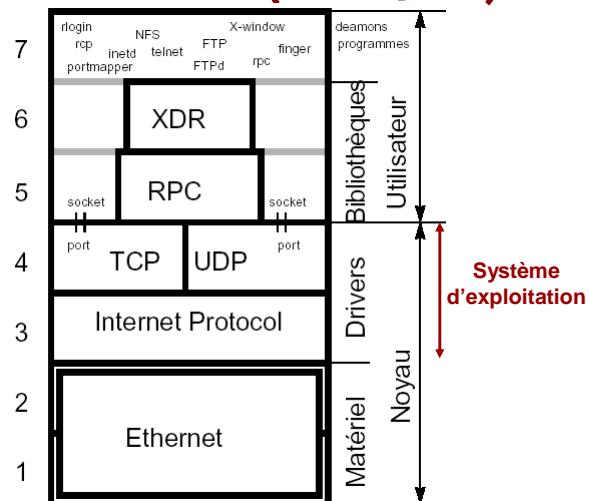


ISO

Internet

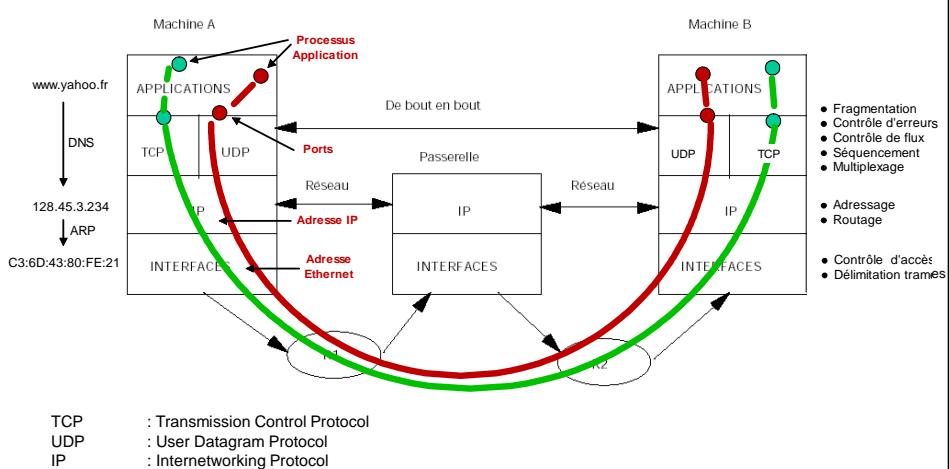
page 14

## Architecture d'un terminal Internet (TCP/IP)



page 15

## Communication Internet/Intranet : le modèle client/serveur

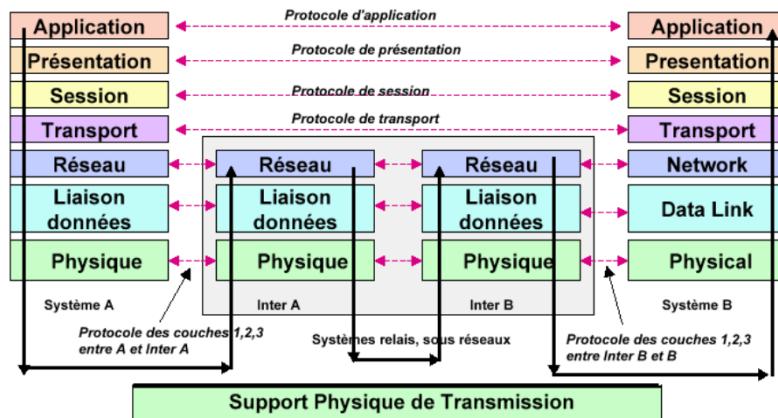


© Ahmed Mehaoua 1999 - page 16

page 16

## LE MODELE ISO

### Principe du Relais



© Ahmed Mehaoua 1999 - page 17

## Organismes de Normalisation

### ♦ Les Organismes Internationaux :

Les organismes de normalisation internationaux cités ci-dessous sont sous l'égide de l'**ONU** et sont les plus **actifs** dans le domaine des **réseaux** et des **télécommunications**.

- **OSI** (Organisation Internationale de Standardisation) ou ISO (International Organisation for Standardisation)
- **UIT** (Union Internationale des Télécommunications) anciennement CCITT (Comité Consultatif International Télégraphique et Téléphonique)

### ♦ Les Organismes Multinationaux :

A ces organismes internationaux, s'ajoutent encore des organismes de différents continents comme l'Europe et les Etats-Unis :

- **IETF** (Internet Engineering Task Force)
- **IEEE** (Institute of Engineers in Electronic & Electrotechnic)
- **ETSI** European Telecommunication Standardization Institute)
- **EBU** (European Broadcasting Union)

## LE MODELE DE REFERENCE ISO de L'OSI

Le Modèle de référence ISO pour Interconnexion des Systèmes Ouverts a été proposé en 1984 par l'OSI (Organisation de standardisation Internationale) :

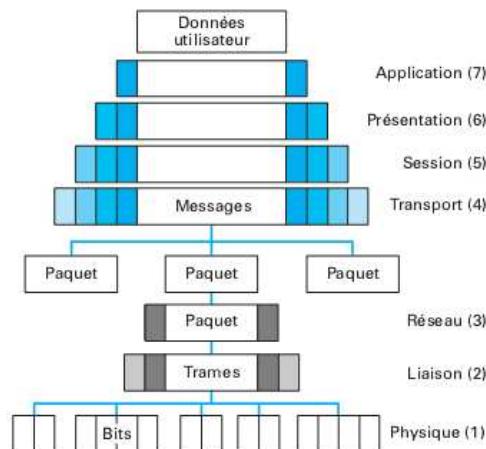
- Modèle fondé sur un principe énoncé par Jules César :
  - « Diviser pour Régner »
- Le principe de base est la représentation des réseaux sous la forme de couche de fonctions superposées les unes aux autres.
  - Leur nombre, leur nom et leur fonction varient selon les réseaux
- L'étude du système de communication revient alors à l'étude de ses éléments élémentaires et offre une plus grande :
  - Facilité d'étude
  - Indépendance des couches
  - Souplette d'évolution

## LE MODELE ISO 7 COUCHES

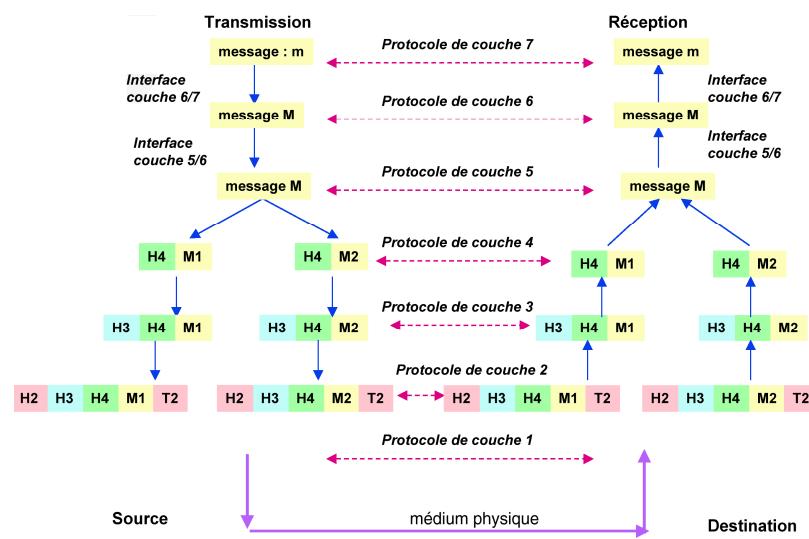
Tableau 2 – Couches du modèle OSI

Niveau	Nom	Fonction	Protocoles
7	Couche application	Assurer l'interface avec les applications.	HTTP, FTP, telnet, SSH, DNS
6	Couche présentation	Formater des données (leur représentation, éventuellement leur compression).	
5	Couche session	Fournir les moyens pour organiser et synchroniser les dialogues et les échanges de données.	
4	Couche transport	Transporter les données et, selon le protocole, gérer les erreurs.	TCP, UDP
3	Couche réseau	Gérer l'adressage et le routage.	IP, ICMP, IGMP, ARP
2	Couche liaison	Définir l'interface avec la carte réseau et la méthode d'accès.	Ethernet, LLC, SNAP, PPP
1	Couche physique	Convertir des données en signaux numériques.	Ethernet, 802.3, 802.5 ( <i>token ring</i> ), 802.11 ( <i>wireless</i> )

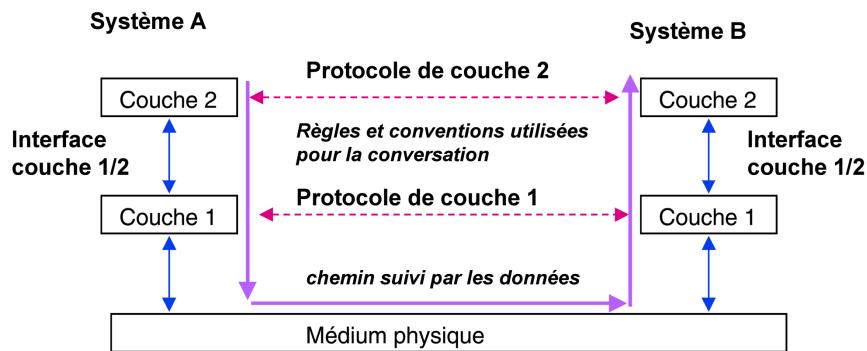
## LE MODELE OSI PRINCIPE DE L'ENCAPSULATION



## LE MODELE ISO Principe de L'encapsulation



## LE MODELE OSI PRINCIPE DU PROTOCOLE



## CARACTERISATION DES RESEAUX

Comment transférer des données d'un point A à un point B ?

### ◆ Transfert en Mode circuit

Toutes les données entre A et B transitent par un même chemin à travers le réseau. Ce chemin est appelé « **Circuit** » et est pré-établi (calculé) pour satisfaire les contraintes de l'application (débit, délai, taux d'erreurs, taux de pertes ....).

Le circuit est calculé lors d'une **phase de mise en connexion** entre A et B (envoi d'un message de contrôle)

Exemples: Réseaux téléphoniques fixes et mobiles : RNIS, GSM, 3G

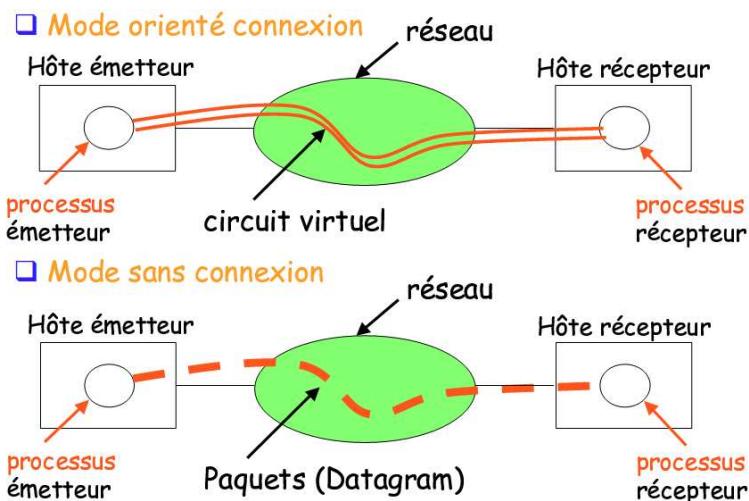
### ◆ Transfert en Mode datagramme

Les données « peuvent » transiter entre A et B par des chemins différents à travers le réseau.

**Pas de phase de mise en connexion** et de calcul d'un chemin entre A et B.

Exemples : Réseau Internet, réseaux 3G.

## Mode de transfert des données

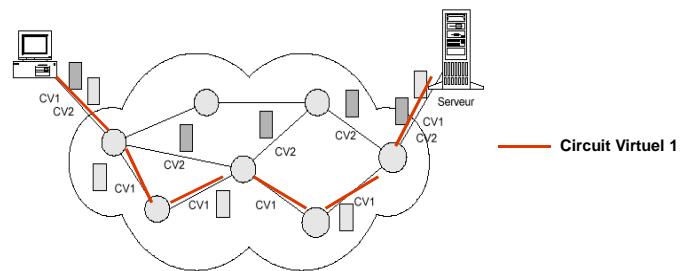


## TRANSFERT EN MODE CIRCUIT

### Principe du service téléphonique :

Toutes les données entre A et B transitent par un même chemin à travers le réseau. Ce chemin est appelé « **Circuit** » et est pré-établi (calculé) pour satisfaire les contraintes de l'application (débit, délai, taux d'erreurs, taux de pertes ....)

1. Si ce circuit est dédié aux communications entre A et B : on parle de **circuit physique**
2. Si ce circuit est partagé entre plusieurs entités (A, B, C, ....) alors on parle de **circuit virtuel**
  
3. Si ce circuit est établi pour une longue période (mois, années) on parle de **circuit permanent**
4. Si ce circuit est établi pour une courte période (transfert des données) on parle de **circuit commuté**

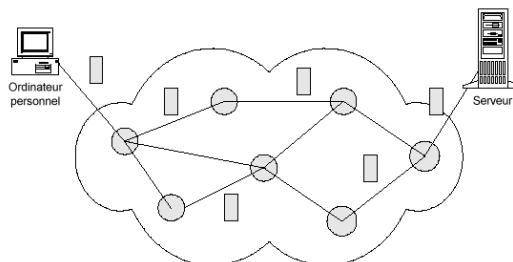


## TRANSFERT EN MODE DATAGRAMME

### Principe du courrier postal

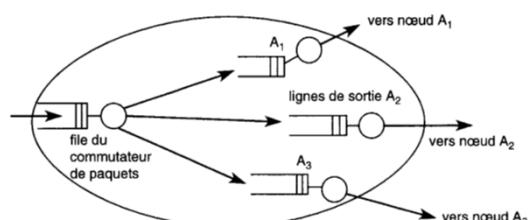
A envoi vers B les différents paquets de son messages avec l'adresse de B sans demande préalable de connexion et de calcul d'un chemin (circuit virtuel) entre A et B.

- C'est aux équipements du réseau d'acheminer ces paquets **individuellement** par des chemins pouvant être différents, et en les **temporisant** si nécessaire.

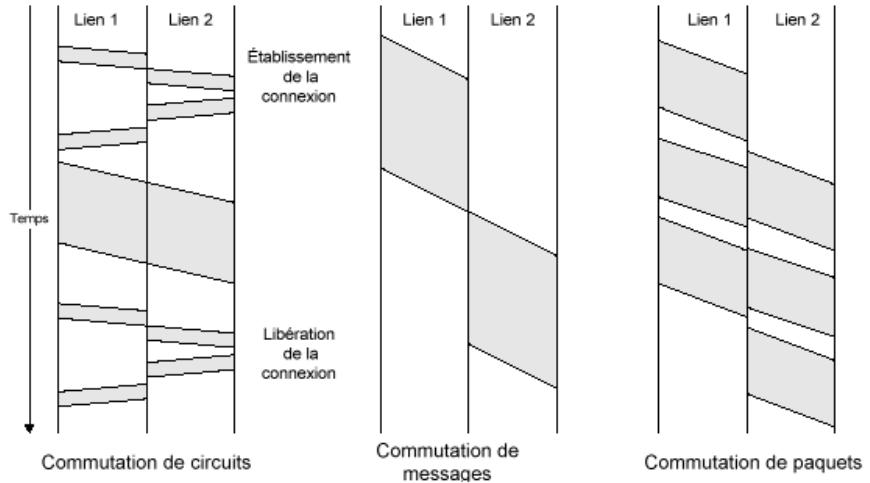


## CARACTERISATION DES RESEAUX TECHNIQUES DE COMMUTATION

- ♦ On s'intéresse au mode de fonctionnement des nœuds du réseau
- ♦ **Commutation** : technique utilisée par les noeuds dans le réseau pour acheminer (aiguiller) les messages de l'émetteur vers le récepteur.
- ♦ Ils existent **5 techniques / variantes** :
  - **commutation de circuit**
  - **commutation de messages**
  - **commutation de paquets**
  - **commutation de trames**
  - **commutation de cellule**



## COMPARAISON DES TECHNIQUES DE COMMUTATION



© Ahmed Mehaoua 2008 - page 29

**Réseaux Informatiques**  
**Couche Physique**

# Codage source et Normes

Informations sous forme binaire 0 et 1 :

- |                      |   |   |
|----------------------|---|---|
| Nombres              | → | Représentation sous forme binaire                     |
| Texte                | → | Code ASCII  |
|                      |   | UNICODE   |
|                      |   | Code Vidéotex   |
|                      |   | ...   |
| Image                | → | Noire et blanc (1 bit : 0 noir et 1 blanc)            |
|                      |   | Nuances de gris (8 bits par point)                    |
|                      |   | Couleur (RVB, 8 bits par couleur → 24 bits par point) |
|                      |   | Compression JPEG                                      |
|                      |   | ...   |
| Parole, Son et Vidéo | → | PCM (Pulse Modulation Code) pour un signal analogique |
|                      |   | Compression DPCM (Son)                                |
|                      |   | Compression MPEG (Vidéo)                              |

## Codage source : code ASCII

**ASCII:** American Standard Code for Information Interchange

Exemples de code ASCII:

Caractère 0 → code ASCII: 30H

Caractère A → code ASCII: 41H

Caractère SP → code ASCII: 20H

SP: Espace

**Bit de parité:** est un bit supplémentaire qu'on ajoute pour faire 8 bits, de telle façon que la somme des éléments binaires modulo 2 soit égale à 0.

poids fort							
000	001	010	011	100	101	110	111
0000 NUL DLE SP 0 ⓥ P \ p	0001 SDH DC1 ! " A Q e q	0010 STX DC2 " 2 B R b r	0011 ETX DC3 # 3 C S c s	0100 EOT DC4 \$ 4 D T d t	0101 ENQ MK % 5 E U e u	0110 ACK SYN & 6 F V f v	0111 BEL ETB , 7 G W g w
1000 BS CAN { 8 H X h x	1001 HT EM } 9 I Y i u	1010 LF SUB ; J Z j z	1011 VT ESC + : K l k {	1100 FF FS ! < L C l o	1101 CR GS - = M ) m }	1110 SO RS . > N ^ n ~	1111 SI US / ? O <-- o DEL
Code ASCII 7 bits							

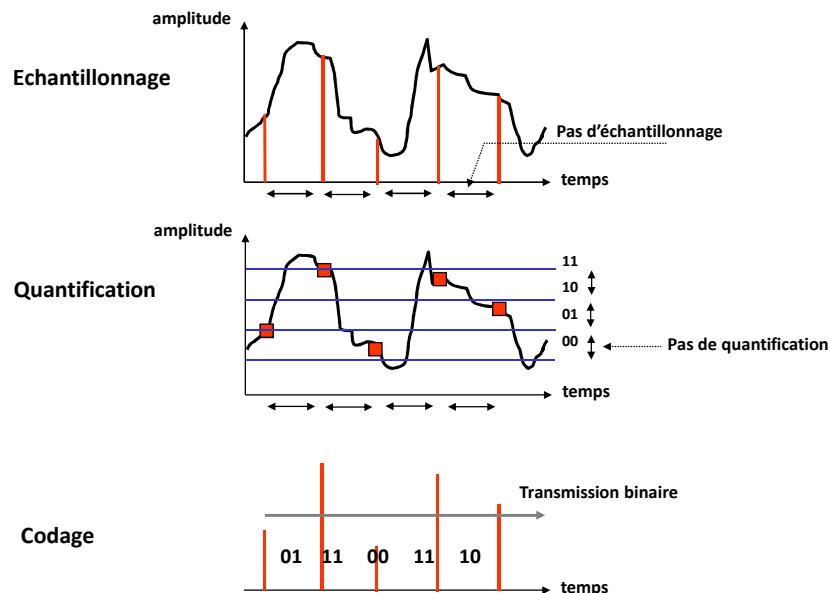
A: 0 100 0001 → Somme des bits (mod 2)=0

Exemples: B: 0 100 0010 → Somme des bits (mod 2)=0

C: 1 100 0011 → Somme des bits (mod 2)=0

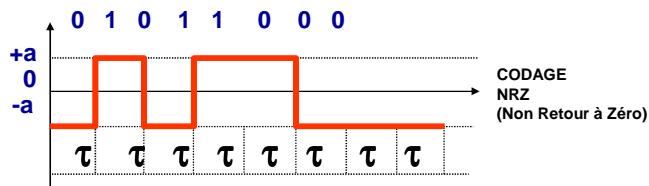
## Codage source : La numérisation

MIC: Modulation par Impulsion et Codage



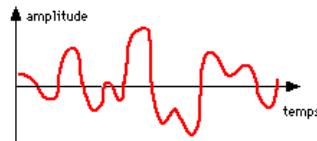
## La couche physique

- **La couche physique** est chargée de la transmission (émission et réception) effective d'un bit ou d'un train de bits continu sous la forme de signaux électriques ou optiques entre les interlocuteurs.
- Cette couche est chargée de la conversion entre bits et signaux électriques ou optiques.
- La transmission numérique (ou bande de base) consiste à convertir (ou coder) les bits en un signal à 2 niveaux : **0 → -a** et **1 → +a**

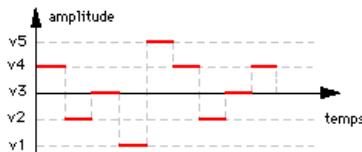


# Transmission

- L'**information** (analogique ou numérique) est véhiculée grâce à un signal physique. Ce signal peut être de nature analogique soit de nature digital (numérique).
- **Transmission analogique:** Un signal analogique est un signal **continu** qui peut prendre une infinité de valeurs.



- **Transmission numérique:** un signal **numérique** varie à des instants déterminés (discontinue) dans le temps et ne peut prendre que des valeurs distinctes dans un ensemble fini.

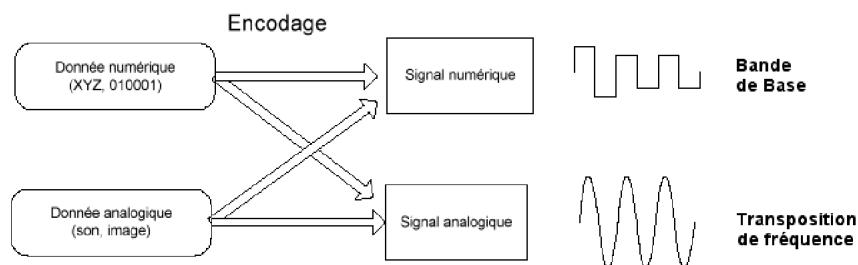


- **Remarque:** 4 combinaisons possibles entre les différents types d'information et les modes de transmission.

## Transmission (suite)

- **4 combinaisons** possibles entre les différents types d'information et les modes de transmission:

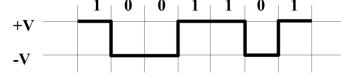
- Information **Analogique** – Transmission **Analogique** (voix sur RTCP)
- Information **Analogique** – Transmission **Numérique** (voix sur GSM ou Internet)
- Information **Numérique** – Transmission **Analogique** (données ordinateur sur RTCP via modem)
- Information **Numérique** – Transmission **Numérique** (données ordinateur sur LAN ou Internet)



## Transmission numérique

- Codage unipolaire sans retour à zéro (NRZ)

- Machine (horloge)



- Codage **Manchester** (simple)

- Inclus le signal d'horloge

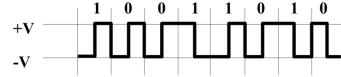
- $\frac{1}{2}$  temps bit à l'inverse de la valeur  
+  $\frac{1}{2}$  temps bit à la valeur.



- Codage **Manchester différentiel**

- Bit 0 = Changement de polarité

- Bit 1 = Polarité du début temps bit identique à précédente



37

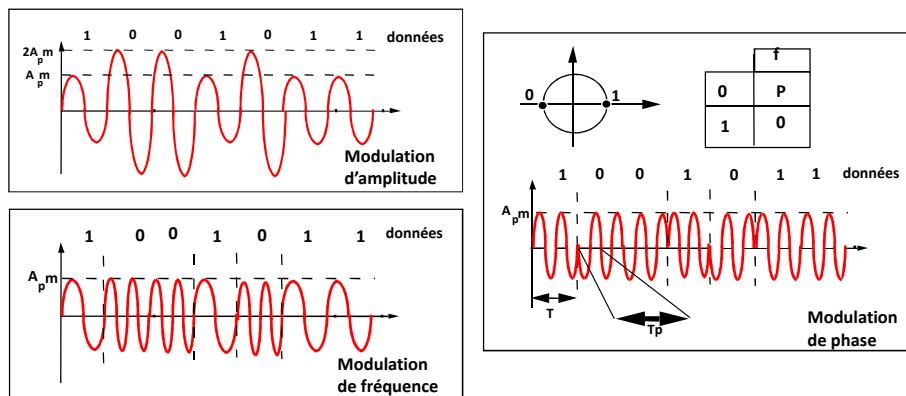
## Transmission analogique

- Un signal est caractérisé par :
  - son amplitude **A**, sa fréquence **f** et sa phase **Φ**, tel que:  
 $y(t) = A \sin(2\pi ft + \Phi)$       avec       $f_{(\text{Hz})} = 1/T$  (T= période)
- Le signal est transporté sous la forme d'une onde adaptée aux caractéristiques physiques du support:
  - **ddp électrique, onde radio-électrique, intensité lumineuse (fibre optique)**
- Le signal se présente sous la forme d'une onde de base régulière appelée **porteuse**.
  - On fait subir des déformations (ou **modulations**) à cette porteuse pour distinguer les éléments du message (0, 1, 00, 01, 10, ....).
  - 4 types de modulations :
    - modulation d'**amplitude**
    - modulation de **fréquence**
    - modulation de **phase** (synchronisation)
    - modulation **combinée** (par exemple de phase et d'amplitude)

38

# Transmission analogique

## la Modulation



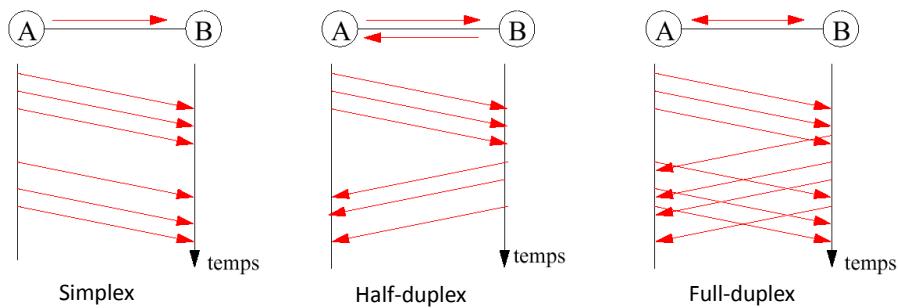
**Modem** (modulateur-démodulateur) entre l'ordinateur  
(numérique) et le système téléphonique (analogique)

39

# Transmission de données

## Modes d'échange

- Unidirectionnel (simplex)
- Bidirectionnel à l'alternat (half-duplex)
- Bi-directionnel (full-duplex)

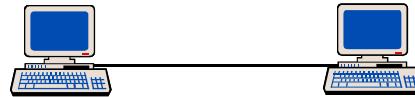
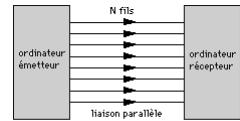


# Transmission de données

## Transmission parallèle

- ✓ Plusieurs bits en même temps

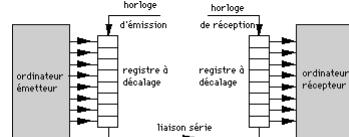
- ✓ 16, 32 ou 64 bits



## Transmission série

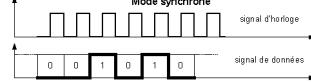
- ✓ 1 bit à la fois

- ✓ Pour les informations de contrôle et les données



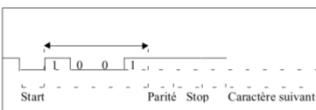
## Transmission synchrone

L'information est transmise sous la forme d'un flot continu de bits à une cadence définie par l'horloge d'émission.



## Transmission asynchrone

Chaque caractère est émis de façon irrégulière dans le temps.



# Multiplexage

## Objectif :

- Optimiser l'usage des canaux de transmission  
→ transit simultané d'un maximum d'informations

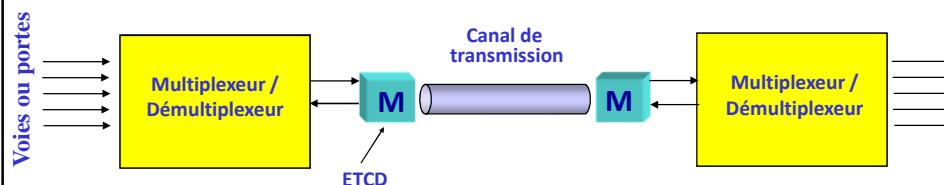
## Principe :

- Traiter le signal pour concentrer des flux d'origines diverses sous forme d'un signal composite unique  
→ signal multiplex

## 3 techniques :

- Multiplexage fréquentiel
- Multiplexage temporel
- Multiplexage temporel statistique

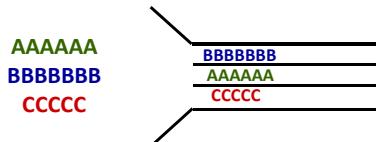
## Equipement:



# Multiplexage: fréquentiel, temporel

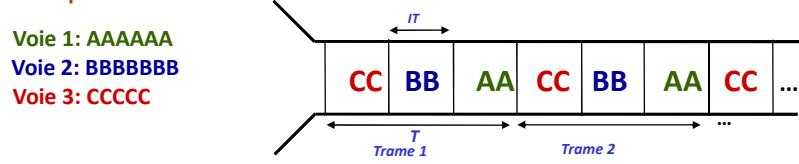
## • Multiplexage fréquentiel

- Découper la bande passante d'un canal en plusieurs sous-bandes, chaque sous-bande est affectée à une voie de transmission



## • Multiplexage temporel

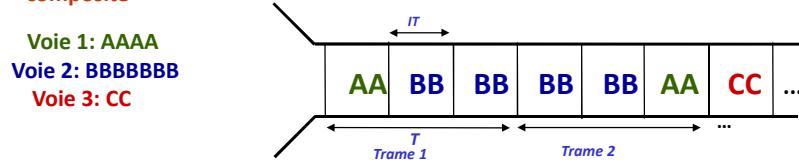
- Appelé aussi TDM (Time Division Multiplexing)
- Prélèvement successif de bits ou (d'octets) sur les différentes voies reliées au multiplexeur pour construire un train de bits (ou d'octets) qui constituera le **signal composite**



# Multiplexage: Temporel statistique

## • Multiplexage temporel statistique

- Appelé aussi STM (statistical Time Division Multiplexing)
- Prélèvement successif de bits ou (d'octets) sur les différentes voies reliées au multiplexeur pour construire un train de bits (ou d'octets) qui constituera le **signal composite**

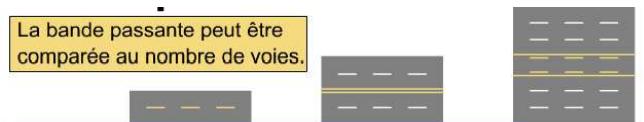


# Définitions

- **Unités (Hz)**
  - La fréquence d'un signal (**Hertz**), est le nombre de périodes (oscillations) par seconde
  - kHz, MHz, GHz ...
- **Bandé Passante (Hz) :**
  - La bande passante, c'est la bande de fréquences dans laquelle les signaux sont correctement reçus
  - $W = F_{\max} - F_{\min}$
- **Rapidité de modulation (signal numérique):**
  - $R$  (bauds) =  $1/\Delta$  ( $\Delta$ : durée d'un élément binaire)

45

## bande passante



Les équipements réseau correspondent aux bretelles, aux feux de signalisation, aux panneaux et aux cartes.



Les paquets sont comparables aux véhicules.

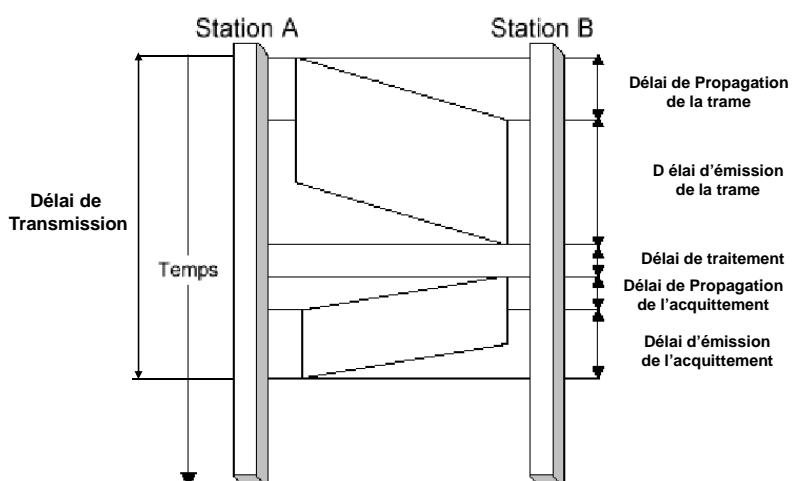


# Débits et Capacité d'un canal

- **Débit binaire:**
  - $D$  (bits/s) =  $n \cdot R$  ( $n$ : nombre de bits/intervalle de modulation)
- **Valence:**
  - $V=2^n$  est appelé **Valence** du signal.
- **Capacité d'une voie de transmission (bit/s ou bps):** est le débit binaire maximal. C'est une fonction directe de la bande passante ( $W$ ) :
  - $C=D_{\max}=W \log_2(1+S/B)$  ( $S/B$  = Signal/Bruit)
  - En effet:
    - Selon Shannon:  $R_{\max} = 1/2 \log_2(1+S/B)$  (canal bruité)
    - Selon Nyquist:  $R_{\max} = 2W$  (canal sans bruit)
- **Remarque:** Lorsque  $V = 2$  (modulation simple), le débit binaire (bits/s) est égal à la rapidité de modulation (bauds). Par abus de langage on parle de débits en bauds ( $V \neq 2$ )

47

## DELAI DE TRANSMISSION



## DELAIS

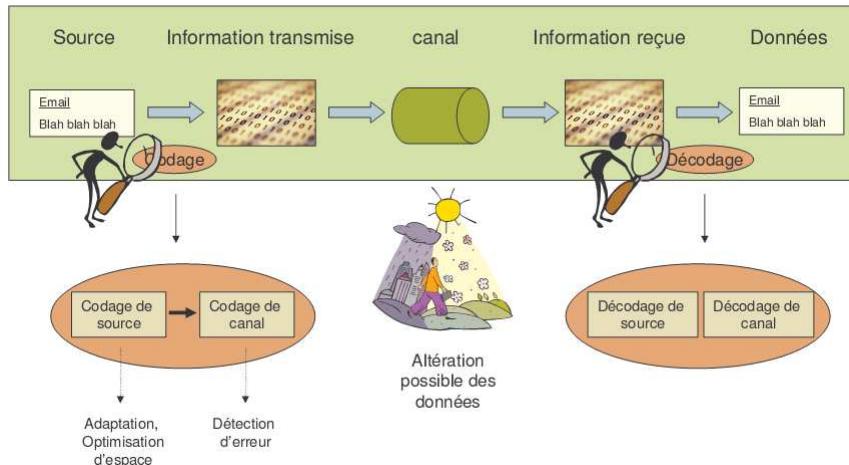
Soit :

- |   |  |
|---|--|
| • C: Capacité/Débit de la ligne (bit/s) | d: distance de propagation (m)         |
| • L: Longueur de la trame (bits)        | L' : Longueur de l'acquittement (bits) |
| • V : vitesse du support (m/s)          |  |

T <sub>e</sub> : délai d'émission de la trame	= L / C
T <sub>p</sub> : délai de propagation de la trame	= d / V
T' <sub>e</sub> : délai d'émission de l'acquittement	= L' / C
T' <sub>p</sub> : délai de propagation de l'ACK	= T <sub>p</sub> = d / V
T <sub>exec</sub> : délai de traitement de la trame/ACK	= négligeable
T: délai de transmission	= T <sub>e</sub> + 2T <sub>p</sub> + T' <sub>e</sub> = ((L+L')/C) + 2d/V
délai de blocage de l'émetteur	= 2T <sub>p</sub> + L'/C

Efficacité d'un protocole	= Taux d'occupation du canal = délai d'émission des données/Délai de transmission = Débit utile / Débit de la ligne
---------------------------	---

## LE CODAGE CANAL



## CONTRÔLE DES ERREURS

Assurer la bonne réception de toutes les données émises

- Téléphonie : 10-3 bits
- vidéo compressée : 10-6
- données informatiques : 10-9

3 opérations à effectuer :

1. détecter une erreur
2. localiser l'erreur dans les données
3. corriger l'erreur

La protection peut s'appliquer à différents niveaux :

1. Au niveau bit ou caractère (bit de parité)
2. Au niveau d'une suite de bits : trame ou paquet, ... (CRC)

Contrôle multiple :

1. codes de contrôle des erreurs (parité, CRC)
2. numérotation de trames
3. vérification de la longueur des trames

## CODES DETECTEURS

- Si l'on veut pouvoir détecter des erreurs, le codage de canal induit toujours un ajout d'information
- Il existe deux principales manières de rajouter cette information :
  - On rajoute à la fin du message un ensemble de bits dédiés au contrôle d'erreur
    - CheckSum (somme de contrôle), **CRC** (Code de Redondance Cyclique), ...
  - On découpe le message en blocs et on calcule un ensemble de bits de contrôle pour chacun des blocs
    - On parle de codages par blocs
    - Parité, ...

# Distance de Hamming

- Distance de Hamming entre deux mots: XOR
  - nombre de bits différents entre 2 mots du code
  - = nombre de bits à 1 dans le résultat du XOR
- Ex:  $10001001 \wedge 10110001$  i.e. 3

$$\begin{array}{r} 10001001 \\ 10110001 \\ \hline 00111000 \end{array}$$

## Déetecter

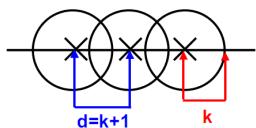
Trame: longueur  $n$  bits =  $m$  données +  $r$  contrôle

- L'ensemble des  $n$  bits est un **mot du code**

Distance de Hamming **d'un code** est  $d$

=>  $d$  erreurs suffisent pour passer d'un mot à un autre

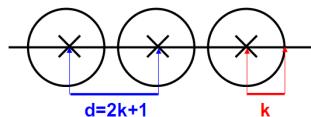
Détecter  $k$  erreurs nécessite une distance de Hamming de  $d=k+1$



## Corriger

Pour corriger  $k$  erreurs,  
il faut que la distance de Hamming soit d'au moins  $d=2k+1$

Le nombre minimal de bits de contrôle  $r$  permettant de corriger **une** erreur (où qu'elle soit) doit vérifier:  $(m+r+1) \leq 2^r$   
Comme on connaît  $m$ , on peut trouver  $r$ .



# CODES DETECTEURS - bits de parité -

- Principe :

- On choisit une convention : parité paire ou impaire
- A chaque bloc de  $k$  bits on ajoute un bit tel que le nombre de 1 dans le bloc de  $k+1$  bits respecte la convention de parité.

- Exemple :

- Soit le message 01011110. On choisit  $k = 4$  et une *parité paire*
- Les deux blocs de 4 bits à coder sont donc 0101 et 1110
- Les deux blocs de 5 bits à transmettre sont donc 01010 et 11101
- Le message transmis est alors 0101011101

- Propriétés :

- Le codage de parité permet de **déetecter un nombre impair d'erreurs**
- Le codage de parité **ne permet pas de corriger les erreurs** détectées

## CODES DETECTEURS

### - bits de parité -

**Parité longitudinale LRC** (longitudinal Redundancy check) :

Pour chaque caractère, on fait la somme des bits à "1" et on ajoute un bit de redondance de parité qui peut prendre la valeur "0" ou "1" selon le type de parité utilisé.

On peut ajouter une **Parité Verticale VRC** (Vertical Redundancy check)

Exemple : donnée initiale codée sur 7 bits (ASCII) : "0011010"

parité paire : "00110101"

parité impaire : "00110100"

1 0 0 1 1 0 0	1
0 1 1 0 0 1 0	1
1 0 0 1 1 0 1	0
<hr/>	
0 1 1 0 0 1 1 0	

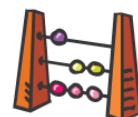
parité LRC et VRC paire

Suite d'éléments binaires émis :  
01100110 10011010 01100101 10011001

## CODES DETECTEURS

### - CHECKSUM -

- Somme de contrôle de  $s$  bits (en général 1) calculée en additionnant les valeurs de blocs de  $b$  bits (en général 8) modulo  $2^s$



- Exemple : Checksum sur 1 octet  $\Rightarrow s = 8 \Rightarrow 2^8 = 256$

- Message à transmettre : 011010100101010101010010

$$01101010 \ 01010101 \ 01010010 = 106 + 85 + 82 = 273$$

$$273 \text{ modulo } 256 = 17$$

- Message transmis : 01101010 01010101 01010010 **00010001**

checksum

## CODES DETECTEURS

### - CRC ou Codes polynomiaux -

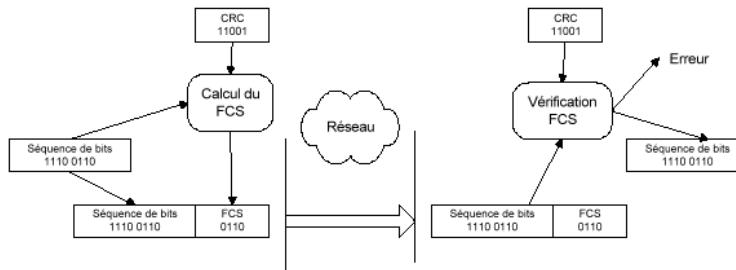
- s'applique sur une suite quelconques de bits
- détection des erreurs plus fiable,
- moins gourmand en ressources

Exemples: codes polynomiaux :

$$\text{CRC-12} = x^{12} + x^{11} + x^3 + x^2 + x^1 + x^0$$

$$\text{CRC-16} = x^{16} + x^{15} + x^2 + x^0$$

$$\text{CRC-CCITT} = x^{16} + x^{12} + x^5 + x^0$$



## CODES DETECTEURS

### - CRC ou Codes polynomiaux -

- On considère une suite de  $n+1$  bits comme un **polynôme de degré  $n$**  où les coefficients ne peuvent prendre que les valeurs 0 ou 1.

– Exemple :  $1001011 = x^6 + x^3 + x + 1 \longrightarrow \text{degré} = 6$

- L'addition et la soustraction de tels polynômes sont de simples Ou-Exclusifs

$$\begin{array}{r}
 101101 = \quad x^5 + \qquad \qquad x^3 + \qquad x^2 + \qquad \qquad x^0 \\
 + \quad 1011 = \qquad \qquad \qquad x^3 + \qquad x^2 + \qquad x + \qquad x^0 \\
 \hline
 100110 = \quad x^5 + \qquad \qquad x^2 + \qquad x \qquad \qquad x^0
 \end{array}$$

$$\begin{array}{r}
 101101 = \quad x^5 + \qquad \qquad x^3 + \qquad x^2 + \qquad \qquad x^0 \\
 - \quad 1011 = \qquad \qquad \qquad - x^3 \qquad - x^2 \qquad - x \qquad - x^0 \\
 \hline
 100110 = \quad x^5 + \qquad \qquad x^2 + \qquad x \qquad \qquad x^0
 \end{array}$$

## CODES DETECTEURS

### - CRC ou Codes polynomiaux -

- On choisit un **polynôme générateur** noté  $G(x)$  de degré d
  - CRC-12       $= x^{12} + x^{11} + x^3 + x^2 + x^1 + 1$
  - CRC-16       $= x^{16} + x^{15} + x^2 + 1$
  - CRC-CCITT     $= x^{16} + x^{12} + x^5 + 1$
  - CRC Eth.      $= x^{32} + x^{26} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x$
- On ajoute au message M à transmettre un bloc **B0** de  $d$  bits à 0
- On effectue la **division de M.B0 par G(x)**, On obtient un reste R de  $d$  bits
  - L'opération de division est la division classique avec l'addition et la soustraction précédentes
- On transmet **M' = M.R**
- Si à l'arrivée on vérifie **M'(x)/G(x) = 0**, alors on considère qu'il n'y pas eu d'erreur.
- Un CRC de  $d$  bits permet de détecter :
  - Avec une probabilité de 1 la présence de paquets d'erreurs de longueur <  $d$
  - Avec une probabilité de  $(1 - 1/2^{d-1})$  la présence de paquets d'erreurs de longueur  $d$
  - Avec une probabilité de  $(1 - 1/2^d)$  la présence de paquets d'erreurs de longueur >  $d$

## CODES DETECTEURS

### - exemple CRC -

Exemple : CRC sur 4 bits

$$\begin{array}{l}
 P(x) = 11010110110000 \\
 \hline
 \begin{array}{r}
 10011 \downarrow \\
 10011 \\
 \hline
 10011 \downarrow \\
 00001 \\
 \hline
 00000 \downarrow \\
 00010 \\
 \hline
 00000 \downarrow \\
 00101 \\
 \hline
 00000 \downarrow \\
 01011 \\
 \hline
 00000 \downarrow \\
 10110 \\
 \hline
 10011 \downarrow \\
 01010 \\
 \hline
 00000 \downarrow \\
 10100 \\
 \hline
 10011 \downarrow \\
 01110 \\
 \hline
 00000
 \end{array}
 \end{array}$$

10011 =  $G(x)$   
 1100001010 =  $Q(x)$

-  $M = 1101011011$   
 - **P(x) = 110110110000**  
 -  $G(x) = x^4 + x + 1$   
 -  $d$  (degré) = 4  
 - CRC = 1110

- Message transmis :  
 $M'(x) = P(x) . R(x)$   
**M'(x) = 1101011011 1110**

$R(x) = 1110$

## **CODES DETECTEURS**

### **- checksum et CRC -**

#### **Problèmes :**

Ces codes détectent les erreurs mais :

- ne les localisent pas
- ne les corrigeant pas\_!!

#### **Solutions :**

1. L'émetteur numérote les blocs de données à transmettre
2. Le destinataire acquitte les blocs reçus
3. L'émetteur retransmait les blocs erronées

Rôle du protocole de communication entre l'émetteur et le destinataire

# Réseaux Informatiques

## Couche Physique

## Codage source et Normes

Informations sous forme binaire **0 et 1** :

- |  |   |
|--|---|
| <b>Nombres</b>   | ➔      Représentation sous forme binaire  |
| <b>Texte</b>   | ➔      Code ASCII<br>UNICODE<br>Code Vidéotex<br>...  |
| <b>Image</b>   | ➔      Noire et blanc (1 bit : 0 noir et 1 blanc)<br>Nuances de gris (8 bits par point)<br>Couleur (RVB, 8 bits par couleur ➔ 24 bits par point)<br>Compression JPEG<br>... |
| Parole, Son et Vidéo ➔ PCM (Pulse Modulation Code) pour un signal analogique |   |
| Compression DPCM (Son)   |   |
| Compression MPEG (Vidéo)   |   |

# Codage source :

## code ASCII

**ASCII:** American Standard Code for Information Interchange

Exemples de code ASCII:

Caractère **O** → code ASCII: **30H**

Caractère **A** → code ASCII: **41H**

Caractère **SP** → code ASCII: **20H**

**SP:** Espace

**Bit de parité:** est un bit supplémentaire qu'on ajoute pour faire **8 bits**, de telle façon que la somme des éléments binaires modulo 2 soit égale à 0.

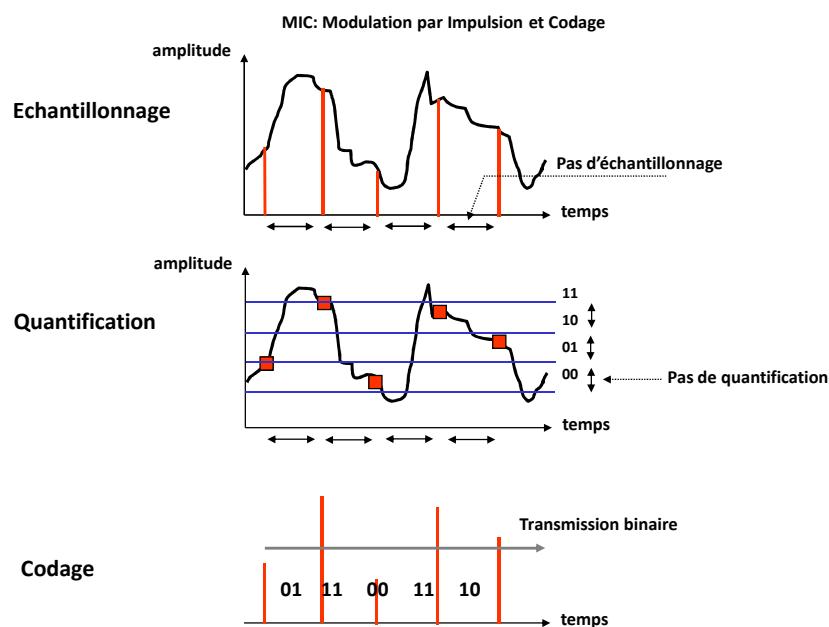
Code ASCII 7 bits										
poids fortés										
000	001	010	011	100	101	110	111			
0000 NUL	0010 DLE	010 SP	011 Ø	100 P	101 \	110 p	111			
0001 SDH	0011 DC1	0102 DC2	0112 Ø	1003 R	1014 Q	1105 b	1116 r			
0010 STX	0011 DC3	0103 Ø	0114 C	1005 S	1016 c	1107 s	1118			
0011 ETX	0010 DC4	0104 Ø	0115 D	1006 T	1017 d	1108 t	1119			
0100 EOT	0011 DC5	0105 Ø	0116 E	1007 U	1018 e	1109 u	1110			
0101 ENQ	0010 DC6	0106 Ø	0117 F	1008 V	1019 f	1110 v	1111			
0110 ACK	0011 SYN	0107 Ø	0118 G	1009 W	1010 g	1111 w	1112			
0111 BEL	0010 ETB	0108 Ø	0119 H	1011 X	1012 h	1113 x	1114			
1000 BS	0011 CAN	0110 Ø	0111 I	1013 Y	1014 i	1115 y	1116			
1001 HT	0010 EM	0112 Ø	0113 J	1015 Z	1016 j	1117 z	1118			
1010 LF	0011 SUB	0114 Ø	0115 K	1017 L	1018 k	1119 l	1110			
1011 VT	0010 ESC	0116 Ø	0117 M	1019 N	1010 m	1111 n	1112			
1100 FF	0011 PS	0118 Ø	0119 Ø	1011 Ø	1012 Ø	1113 Ø	1114 Ø			
1101 CR	0010 GS	0110 Ø	0111 Ø	1013 Ø	1014 Ø	1115 Ø	1116 Ø			
1110 SO	0011 RS	0112 Ø	0113 Ø	1015 Ø	1016 Ø	1117 Ø	1118 Ø			
1111 SI	0010 US	0114 Ø	0115 Ø	1017 Ø	1018 Ø	1119 Ø	1110 Ø			

A: 0 100 0001 → Somme des bits (mod 2)=0

Exemples: B: 0 100 0010 → Somme des bits (mod 2)=0

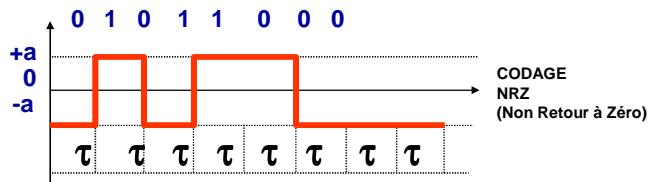
C: 1 100 0011 → Somme des bits (mod 2)=0

# Codage source : La numérisation



## La couche physique

- La couche physique est chargée de la transmission (émission et réception) effective d'un bit ou d'un train de bits continu sous la forme de signaux électriques ou optiques entre les interlocuteurs.
- Cette couche est chargée de la conversion entre bits et signaux électriques ou optiques.
- La transmission numérique (ou bande de base) consiste à convertir (ou coder) les bits en un signal à 2 niveaux :  $0 \rightarrow -a$  et  $1 \rightarrow +a$

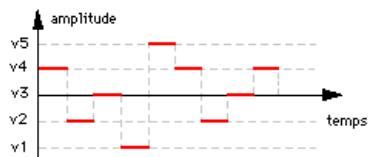


## Transmission

- L'information (analogique ou numérique) est véhiculée grâce à un signal physique. Ce signal peut être de nature analogique soit de nature digital (numérique).
- **Transmission analogique:** Un signal analogique est un signal continu qui peut prendre une infinité de valeurs.



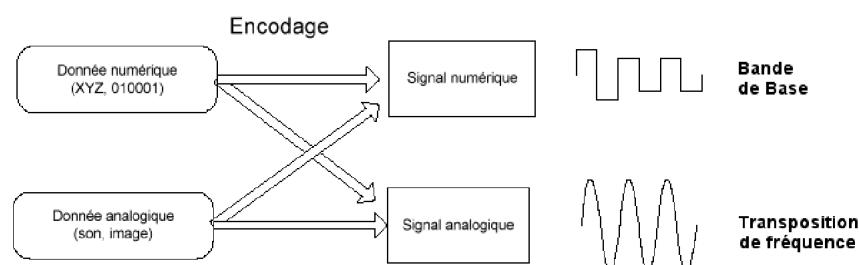
- **Transmission numérique:** un signal numérique varie à des instants déterminés (discontinu) dans le temps et ne peut prendre que des valeurs distinctes dans un ensemble fini.



- **Remarque:** 4 combinaisons possibles entre les différents types d'information et les modes de transmission.

## Transmission (suite)

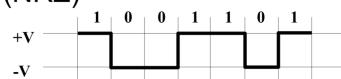
- **4 combinaisons** possibles entre les différents types d'information et les modes de transmission:
  - Information **Analogique** – Transmission **Analogique** (voix sur RTCP)
  - Information **Analogique** – Transmission **Numérique** (voix sur GSM ou Internet)
  - Information **Numérique** – Transmission **Analogique** (données ordinateur sur RTCP via modem)
  - Information **Numérique** – Transmission **Numérique** (données ordinateur sur LAN ou Internet)



## Transmission numérique

- Codage unipolaire sans retour à zéro (NRZ)

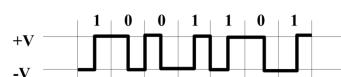
- Machine (horloge)



- Codage **Manchester** (simple)

- Inclus le signal d'horloge

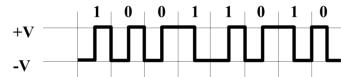
- $\frac{1}{2}$  temps bit à l'inverse de la valeur  
+  $\frac{1}{2}$  temps bit à la valeur.



- Codage **Manchester différentiel**

- Bit 0 = Changement de polarité

- Bit 1 = Polarité du début temps bit identique à précédente

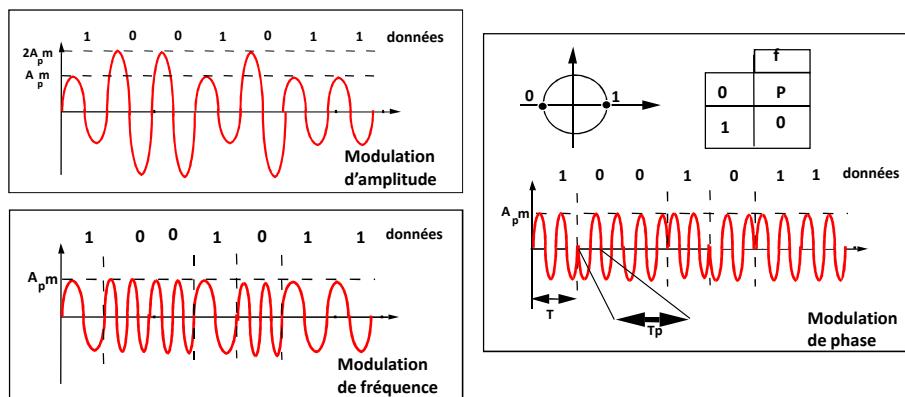


# Transmission analogique

- Un signal est caractérisé par :
  - son amplitude  $A$ , sa fréquence  $f$  et sa phase  $\Phi$ , tel que:  
 $y(t) = A \sin(2\pi ft + \Phi)$  avec  $f_{(\text{Hz})} = 1/T$  ( $T$ = période)
- Le signal est transporté sous la forme d'une onde adaptée aux caractéristiques physiques du support:
  - ddp électrique, onde radio-électrique, intensité lumineuse (fibre optique)
- Le signal se présente sous la forme d'une onde de base régulière appelée **porteuse**.
  - On fait subir des déformations (ou **modulations**) à cette porteuse pour distinguer les éléments du message (0, 1, 00, 01, 10, ....).
  - 4 types de modulations :
    - modulation d'**amplitude**
    - modulation de **fréquence**
    - modulation de **phase** (synchronisation)
    - modulation **combinée** (par exemple de phase et d'amplitude)

9

## Transmission analogique la Modulation



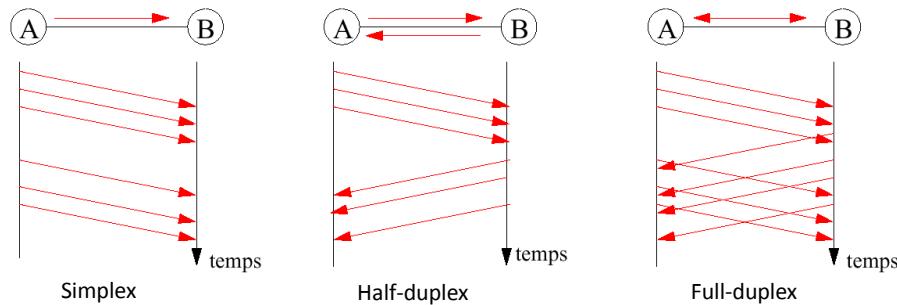
Modem (modulateur-démodulateur) entre l'ordinateur  
(numérique) et le système téléphonique (analogique)

10

# Transmission de données

## Modes d'échange

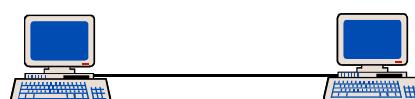
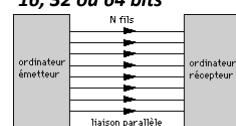
- Unidirectionnel (simplex)
- Bidirectionnel à l'alternat (half-duplex)
- Bi-directionnel (full-duplex)



# Transmission de données

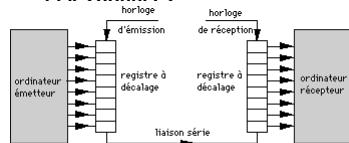
## Transmission parallèle

- ✓ Plusieurs bits en même temps
- ✓ 16, 32 ou 64 bits



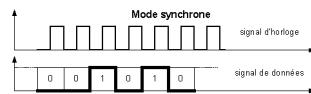
## Transmission série

- ✓ 1 bit à la fois
- ✓ Pour les informations de contrôle et les données



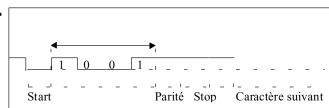
## Transmission synchrone

L'information est transmise sous la forme d'un flot continu de bits à une cadence définie par l'horloge d'émission.



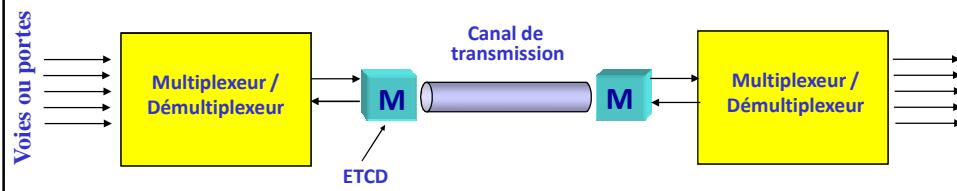
## Transmission asynchrone

Chaque caractère est émis de façon irrégulière dans le temps.



# Multiplexage

- **Objectif :**
  - Optimiser l'usage des canaux de transmission  
→ transit simultané d'un maximum d'informations
- **Principe :**
  - Traiter le signal pour concentrer des flux d'origines diverses sous forme d'un signal composite unique  
→ signal multiplex
- **3 techniques :**
  - Multiplexage fréquentiel
  - Multiplexage temporel
  - Multiplexage temporel statistique
- **Equipement:**

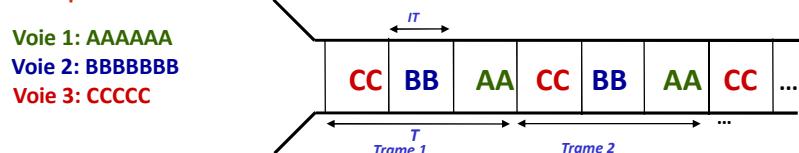


## Multiplexage: fréquentiel, temporel

- **Multiplexage fréquentiel**
  - Découper la bande passante d'un canal en plusieurs sous-bandes, chaque sous-bande est affectée à une voie de transmission
- **Multiplexage temporel**
  - Appelé aussi TDM (Time Division Multiplexing)
  - Prélèvement successif de bits ou (d'octets) sur les différentes voies reliées au multiplexeur pour construire un train de bits (ou d'octets) qui constituera le **signal composite**



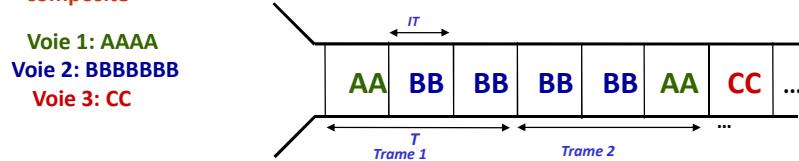
- Appelé aussi TDM (Time Division Multiplexing)
- Prélèvement successif de bits ou (d'octets) sur les différentes voies reliées au multiplexeur pour construire un train de bits (ou d'octets) qui constituera le **signal composite**



## Multiplexage: Temporel statistique

- Multiplexage temporel statistique

- Appelé aussi STM (statistical Time Division Multiplexing)
- Prélèvement successif de bits ou (d'octets) sur les différentes voies reliées au multiplexeur pour construire un train de bits (ou d'octets) qui constituera le **signal composite**



## Définitions

- Unités (**Hz**)

- La fréquence d'un signal (**Hertz**), est le nombre de périodes (oscillations) par seconde
- kHz, MHz, GHz ...

- Bande Passante (**Hz**) :

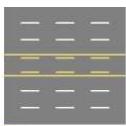
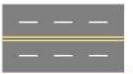
- La bande passante, c'est la bande de fréquences dans laquelle les signaux sont correctement reçus
- $W = F_{\max} - F_{\min}$

- Rapidité de modulation (signal numérique):

- $R$  (bauds) =  $1/\Delta$  ( $\Delta$ : durée d'un élément binaire)

## bande passante

La bande passante peut être comparée au nombre de voies.



Les équipements réseau correspondent aux bretelles, aux feux de signalisation, aux panneaux et aux cartes.



Les paquets sont comparables aux véhicules.



© Cisco Systems, Inc. 1999  
7

## Débits et Capacité d'un canal

- Débit binaire:

- $D$  (bits/s) =  $n \cdot R$  ( $n$ : nombre de bits/intervalle de modulation)

- Valence:

- $V=2^n$  est appelé **Valence** du signal.

- Capacité d'une voie de transmission (**bit/s ou bps**): est le débit binaire maximal. C'est une fonction directe de la bande passante ( $W$ ) :

- $C=D_{max}=W \log_2(1+S/B)$  ( $S/B$  = Signal/Bruit)

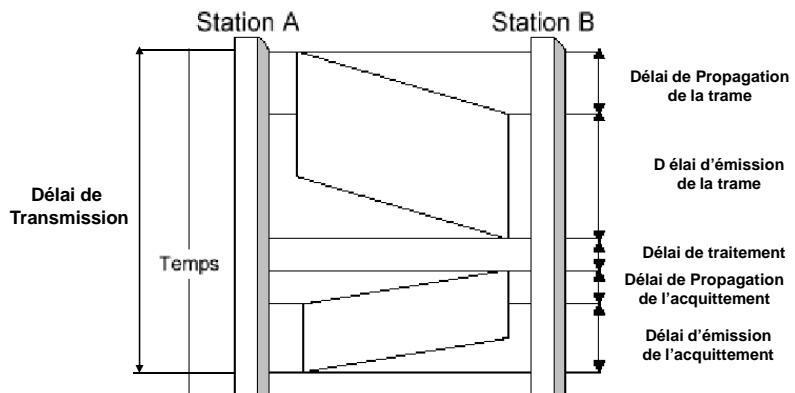
- En effet:

- Selon Shannon:  $n_{max} = 1/2 \log_2(1+S/B)$  (canal bruité)

- Selon Nyquist:  $R_{max} = 2W$  (canal sans bruit)

- **Remarque:** Lorsque  $V = 2$  (modulation simple), le débit binaire (bits/s) est égal à la rapidité de modulation (bauds). Par abus de langage on parle de débits en bauds ( $V \neq 2$ )

## DELAIS DE TRANSMISSION



## DELAIS

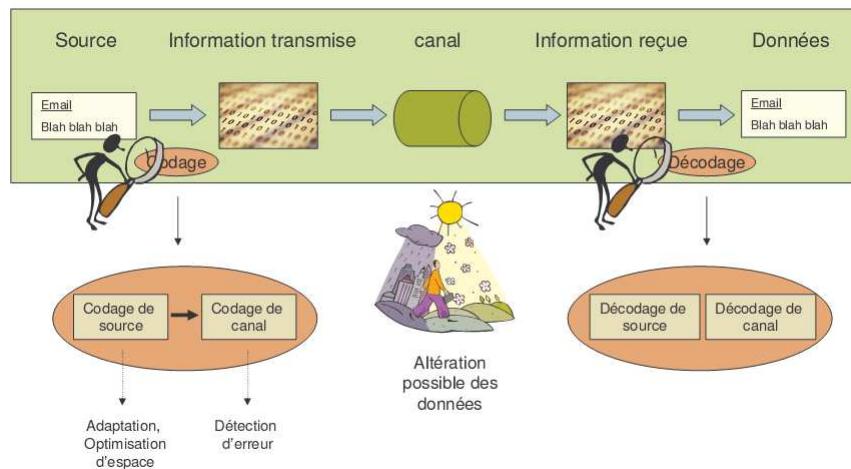
Soit :

- C: Débit de la ligne (bit/s)
- L: Longueur de la trame (bits)
- V : vitesse du support (m/s)
- D: distance de propagation (m)
- L' : Longueur de l'acquittement (bits)

Te: délai d'émission de la trame	$= L / C$
Tp: délai de propagation de la trame	$= D / V$
T'e : délai d'émission de l'acquittement	$= L' / C$
T'p: délai de propagation de l'ACK	$= Tp = D / V$
Texec : délai de traitement de la trame/ACK	= négligeable
T: délai de transmission	$= Te + 2Tp + T'e = ((L+L')/C) + 2D/V$
délai de blocage de l'émetteur	$= 2Tp + L'/C$

Efficacité d'un protocole	= Taux d'occupation du canal = délai d'émission des données/Délai de transmission = Débit utile / Débit de la ligne
---------------------------	---

## LE CODAGE CANAL



## CONTRÔLE DES ERREURS

Assurer la bonne réception de toutes les données émises

- Téléphonie : 10-3 bits
- vidéo compressée : 10-6
- données informatiques : 10-9

3 opérations à effectuer :

1. détecter une erreur
2. localiser l'erreur dans les données
3. corriger l'erreur

La protection peut s'appliquer à différents niveaux :

1. Au niveau bit ou caractère (bit de parité)
2. Au niveau d'une suite de bits : trame ou paquet, ... (CRC)

Contrôle multiple :

1. codes de contrôle des erreurs (parité, CRC)
2. numérotation de trames
3. vérification de la longueur des trames

## CODES DETECTEURS

- Si l'on veut pouvoir détecter des erreurs, le codage de canal induit toujours un ajout d'information
- Il existe deux principales manières de rajouter cette information :
  - On rajoute à la fin du message un ensemble de bits dédiés au contrôle d'erreur
    - CheckSum (somme de contrôle), **CRC** (Code de Redondance Cyclique), ...
  - On découpe le message en blocs et on calcule un ensemble de bits de contrôle pour chacun des blocs
    - On parle de codages par blocs
    - Parité, ...

## Distance de Hamming

- **Distance de Hamming** entre deux mots: XOR
  - nombre de bits différents entre 2 mots du code
  - = nombre de bits à 1 dans le résultat du XOR

10001001  
10110001  
00111000

Ex:  $10001001 \wedge 10110001$  i.e. 3

### Détecter

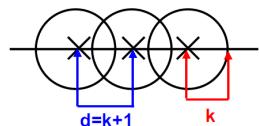
Trame: longueur  $n$  bits =  $m$  données +  $r$  contrôle

➤ L'ensemble des  $n$  bits est un **mot du code**

Distance de Hamming **d'un code** est  $d$

=>  $d$  erreurs suffisent pour passer d'un mot à un autre

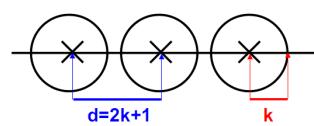
Détecter  $k$  erreurs nécessite une distance de Hamming de  $d=k+1$



### Corriger

Pour corriger  $k$  erreurs,  
il faut que la distance de Hamming soit d'au moins  $d=2k+1$

Le nombre minimal de bits de contrôle  $r$  permettant de corriger **une** erreur (où qu'elle soit) doit vérifier:  $(m+r+1) \leq 2^r$   
Comme on connaît  $m$ , on peut trouver  $r$ .



## CODES DETECTEURS

### - bits de parité -

- Principe :

- On choisit une convention : parité paire ou impaire
- A chaque bloc de  $k$  bits on ajoute un bit tel que le nombre de 1 dans le bloc de  $k+1$  bits respecte la convention de parité.

- Exemple :

- Soit le message 01011110. On choisit  $k = 4$  et une *parité paire*
- Les deux blocs de 4 bits à coder sont donc 0101 et 1110
- Les deux blocs de 5 bits à transmettre sont donc 01010 et 11101
- Le message transmis est alors 0101011110

- Propriétés :

- Le codage de parité permet de **déetecter un nombre impair d'erreurs**
- Le codage de parité **ne permet pas de corriger les erreurs** détectées

## CODES DETECTEURS

### - bits de parité -

**Parité longitudinale LRC** (longitudinal Redundancy check) :

Pour chaque caractère, on fait la somme des bits à “1” et on ajoute un bit de redondance de parité qui peut prendre la valeur “0” ou “1” selon le type de parité utilisé.

On peut ajouter une **Parité Verticale VRC** (Vertical Redundancy check)

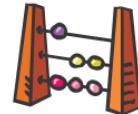
Exemple : donnée initiale codée sur 7 bits (ASCII) : “0011010”  
parité paire : “00110101”  
parité impaire : “00110100”

1 0 0 1 1 0 0	1	parité LRC et VRC paire
0 1 1 0 0 1 0	1	
1 0 0 1 1 0 1	0	

Suite d'éléments binaires émis :  
01100110 10011010 01100101 10011001

## CODES DETECTEURS - CHECKSUM -

- Somme de contrôle de  $s$  bits (en général 1) calculée en additionnant les valeurs de blocs de  $b$  bits (en général 8) modulo  $2^s$



- Exemple : Checksum sur 1 octet  $\Rightarrow s = 8 \Rightarrow 2^8 = 256$

- Message à transmettre : 011010100101010101010010

$$01101010\ 01010101\ 01010010 = 106 + 85 + 82 = 273$$

$$273 \text{ modulo } 256 = 17$$

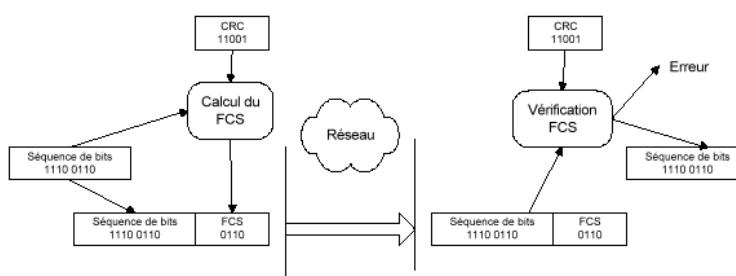
- Message transmis : 01101010 01010101 01010010 00010001  
checksum

## CODES DETECTEURS - CRC ou Codes polynomiaux -

- s'applique sur une suite quelconques de bits
- détection des erreurs plus fiable,
- moins gourmand en ressources

Exemples: codes polynomiaux :

CRC-12	$= x^{12} + x^{11} + x^3 + x^2 + x^1 + x^0$
CRC-16	$= x^{16} + x^{15} + x^2 + x^0$
CRC-CCITT	$= x^{16} + x^{12} + x^5 + x^0$



## CODES DETECTEURS

### - CRC ou Codes polynomiaux -

- On considère une suite de  $n+1$  bits comme un **polynôme de degré  $n$**  où les coefficients ne peuvent prendre que les valeurs 0 ou 1.

– Exemple :  $1001011 = x^6 + x^3 + x + 1 \longrightarrow \text{degré} = 6$

- L'addition et la soustraction de tels polynômes sont de simples Ou-Exclusifs

$$\begin{array}{r} 101101 = x^5 + & x^3 + & x^2 + & x^0 \\ + 1011 = & x^3 + & x + & x^0 \\ \hline 100110 = x^5 + & x^2 + & x & \end{array}$$

$$\begin{array}{r} 101101 = x^5 + & x^3 + & x^2 + & x^0 \\ - 1011 = & - x^3 & - x & - x^0 \\ \hline 100110 = x^5 + & x^2 + & x & \end{array}$$

## CODES DETECTEURS

### - CRC ou Codes polynomiaux -

- On choisit un **polynôme générateur** noté  $G(x)$  de degré  $d$ 
  - CRC-12       $= x^{12} + x^{11} + x^3 + x^2 + x^1 + 1$
  - CRC-16       $= x^{16} + x^{15} + x^2 + 1$
  - CRC-CCITT     $= x^{16} + x^{12} + x^5 + 1$
  - CRC Eth.      $= x^{32} + x^{26} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + + x^2 + x$
- On ajoute au message  $M$  à transmettre un bloc **B0** de  $d$  bits à 0
- On effectue la **division de  $M.B0$  par  $G(x)$** , On obtient un reste  $R$  de  $d$  bits
  - L'opération de division est la division classique avec l'addition et la soustraction précédentes
- On transmet  **$M' = M.R$**
- Si à l'arrivée on vérifie  **$M'(x)/G(x) = 0$** , alors on considère qu'il n'y pas eu d'erreur.
- Un CRC de  $d$  bits permet de détecter :
  - Avec une probabilité de 1 la présence de paquets d'erreurs de longueur  $< d$
  - Avec une probabilité de  $(1 - 1/2^{d-1})$  la présence de paquets d'erreurs de longueur  $d$
  - Avec une probabilité de  $(1 - 1/2^d)$  la présence de paquets d'erreurs de longueur  $> d$

## CODES DETECTEURS

### - exemple CRC -

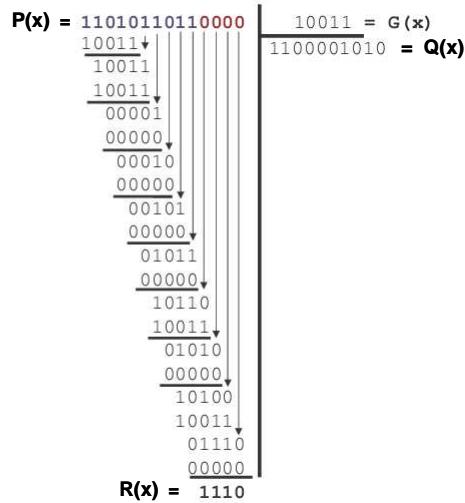
Exemple : CRC sur 4 bits

- $M = 1101011011$
- $P(x) = 110110110000$
- $G(x) = x^4 + x + 1$
- $d$  (degré) = 4
- CRC = 1110

- Message transmis :

$$M'(x) = P(x) \cdot R(x)$$

$$M'(x) = 1101011011 1110$$



## CODES DETECTEURS

### - checksum et CRC -

#### Problèmes :

Ces codes détectent les erreurs mais :

- ne les localisent pas
- ne les corrigeant pas !!

#### Solutions :

1. L'émetteur numérote les blocs de données à transmettre
2. Le destinataire acquitte les blocs reçus
3. L'émetteur retransmet les blocs erronées

Rôle du protocole de communication entre l'émetteur et le destinataire

# **La Couche Liaison**

## **Exemple de la procédure HDLC**

**High-level Data Link Control**

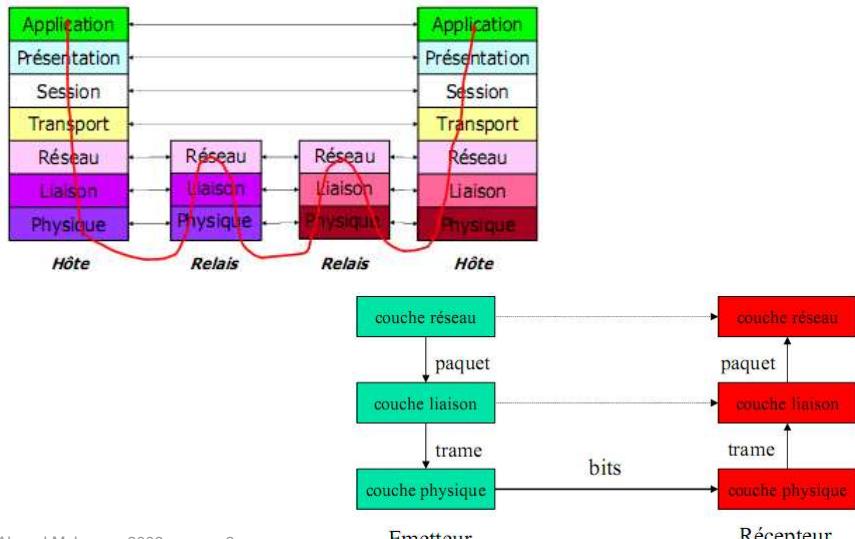
© Ahmed Mehaoua 2008 - page 1

### **Couche Liaison de données : Objectifs**

- Communication (fiable et efficace) entre deux machines adjacentes
  - deux machines physiquement connectées par un canal de transmission
  - La couche liaison récupère des paquets de la couche réseau.
  - Pour chaque paquet, elle construit une (ou plusieurs) trame(s).
  - La couche liaison envoie chaque trame à la couche physique.
- Liaisons de transmission ne sont pas parfaites :
  - Débit binaire limité, le délai de propagation est non nul, il peut y avoir des erreurs de transmission
- **Cette couche doit assurer une transmission exempte d'erreurs sur un canal de communication.**
- **Elle doit aussi assurer un délivrance ordonnée des informations**

© Ahmed Mehaoua 2006 - page 2

## Couche Liaison de données : Services offerts



© Ahmed Mehaoua 2006 - page 3

## Couche Liaison de données : Services offerts

- Gestion (délimitation) de trames
- Contrôle d'erreurs
- Contrôle de flux
- Contrôle d'accès à un canal partagé (MAC)

© Ahmed Mehaoua 2006 - page 4

## DELIMITATION DES DONNEES

### protocole synchrone orienté bit

Données à envoyer

0	1	1	0	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Données transmises sur le support physique

01111110	0	1	1	0	1	1	1	1	0	1	1	0	1	0	1	1	1	1	0	01111110
----------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	----------

Bits de transparence

Données stockées par le récepteur après retrait des bits de transparence

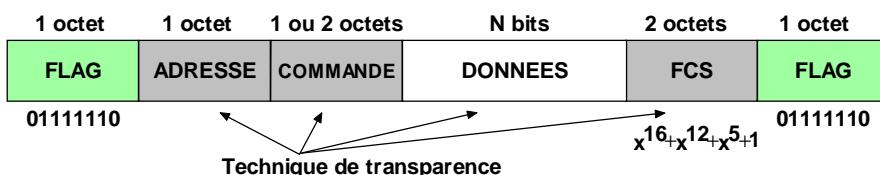
0	1	1	0	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

- Un mécanisme de transparence permet de régler les problèmes d'apparition du fanion dans le bloc de données.
- Avantages : (1) indépendant du code utilisé – (2) trame de taille variable et longue
- Exemples : IOSI HDLC, IETF PPP

© Ahmed Mehaoua 2006 - page 5

## DELIMITATION DES DONNEES

### exemple Trame HDLC



Le champ « DONNEES est généralement de taille constante.

N = 128 ou 256 octets

FCS : Frame Check Sequence (contrôle des erreurs binaires)

© Ahmed Mehaoua 2006 - page 6

## CONTRÔLE DES ERREURS

### 1- Vérification au récepteur de données

Vérification du format des trames :

- longueur, valeurs prédefinies de certains champs

Détection de la corruption des trames :

- champ de contrôle d'erreur → CRC-16 pour HDLC (champ FCS)

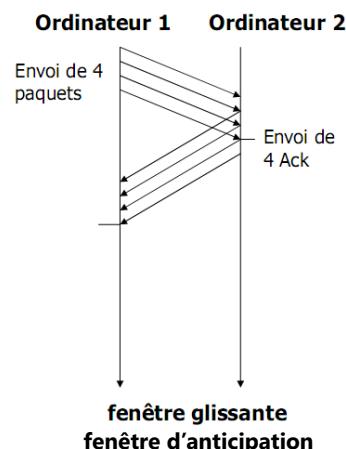
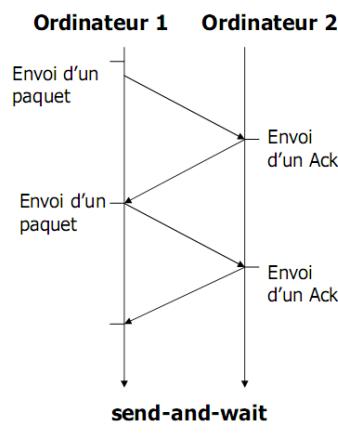
### 2- Information de l'émetteur de données

- Soit implicitement par temporisateur
  - armé à chaque envoi de trame,
  - désarmé lors de la réception d'un acquittement positif
- Soit explicitement par "Nack"
  - le **rejet total** : retransmission de toutes les trames à partir de celle spécifiée
  - le **rejet sélectif** : retransmission de la trame spécifiée

### 3- Retransmission de la trame (perdue ou détruite) par l'émetteur

© Ahmed Mehaoua 2006 - page 7

## CONTRÔLE DE FLUX 2 mécanismes



© Ahmed Mehaoua 2006 - page 8

## CONTRÔLE DE FLUX avec mécanisme SIMPLE et UTOPIQUE « SEND & WAIT »

Hypothèses :

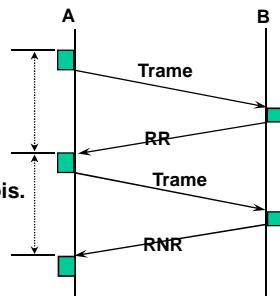
- Transmission de trames de données (I) dans un seul sens
- Canal de communication parfait (pas d'erreurs ni pertes)
- Taille finie des mémoires tampon

Solution :

- Introduction de 2 trames de supervision (S), qui ne transportent aucune information utile et qui sont invisibles aux utilisateurs :
  - RR (Receiver Ready)
  - RNR (Receiver Not Ready)

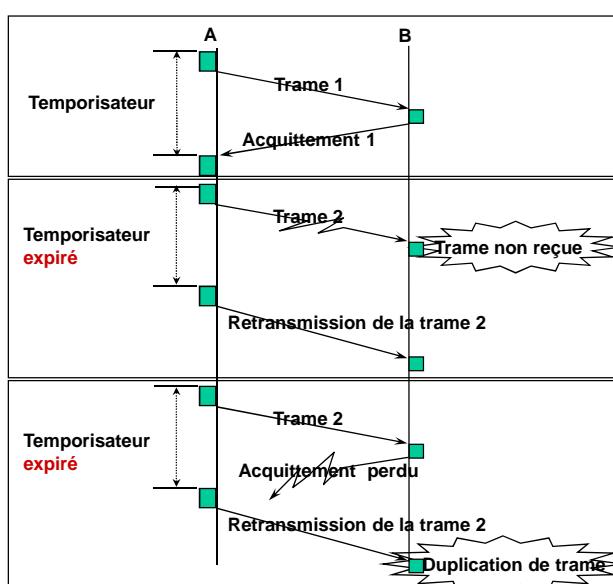
2 variantes :

- Envoie d'une trame de supervision après chaque trame de données,
- Envoie d'une trame RNR si tampon plein, suivie d'une trame RR pour reprendre les envois.



© Ahmed Mehaoua 2006 - page 9

### Problèmes des duplications de trames

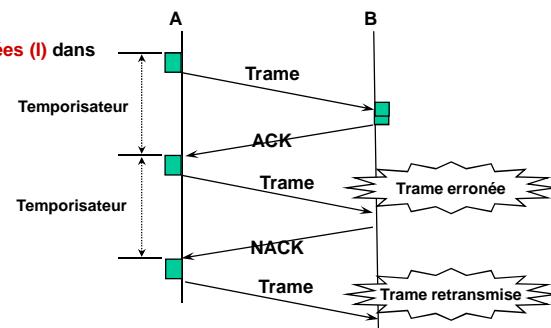


© Ahmed Mehaoua 2006 - page 10

## CONTRÔLE DE FLUX avec mécanismes « SEND & WAIT » et CONTRÔLE DES ERREURS avec « ACQUITTEMENT »

Hypothèses :

- Transmission de trames de données (I) dans un seul sens
- Canal de communication bruité
- Taille finie des mémoires tampon



Problèmes:

- Trames perdues
- Trames erronées
- Duplication de trame

Solution :

- Ajouter un processus d'acquittement positif ou négatif
- Utiliser un **temporisateur ou Timer** pour borner le délai de réception des ACK
- **Numérotation des trames modulo M** (valeur 2, 8 ou 128)
- Ajout d'un champ N(S) dans l'en-tête des trames de données et de supervision
- Ajout de compteurs V(S) et V(R) dans les terminaux émetteurs et récepteurs
- Requière une initialisation de l'échange pour la négociation de la valeur du compteur (protocole en mode connecté)

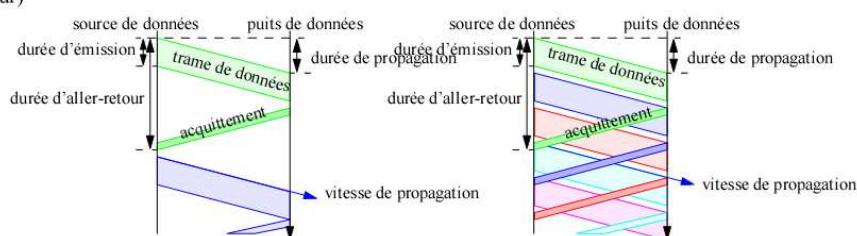
**ATTENTION :** La fonction de Contrôle de Flux et de contrôle d'erreurs peuvent utiliser la même trame de supervision (par exemple RR et RNR)

© Ahmed Mehaoua 2006 - page 11

## Transmission avec fenêtre d'anticipation

Les protocoles simples précédents (bit alterné, “send and wait”, “stop and go”) ont comme principal inconvénient de n'autoriser que la transmission d'une seule trame à la fois.

La liaison de données est alors inoccupée la plupart du temps. De même, l'émetteur (resp. le récepteur) passe son temps à attendre l'acquittement du récepteur (resp. la trame de données de l'émetteur)



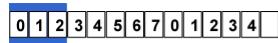
□ On autorise l'émission (resp. la réception) de plusieurs trames d'information consécutives sans attendre l'acquittement de la première (resp. avant d'envoyer l'acquittement).

- on remarque que la source et le puits émettent et reçoivent simultanément
- que la liaison est utilisée de manière bidirectionnelle

© Ahmed Mehaoua 2006 - page 12

## Transmission avec fenêtre d'anticipation (2/3)

### Exemple W = 3



- on peut émettre 0, 1, 2



- on reçoit trame RR demandant 3
- on peut émettre 3, 4, 5



- on reçoit trame RR demandant 5
- on peut émettre 5, 6, 7



- on reçoit trame RR demandant 7
- on peut émettre 7, 0, 1

- etc ...

© Ahmed Mehaoua 2006 - page 13

## Transmission avec fenêtre d'anticipation (3/3)

□ Le nombre maximum de trames consécutives que l'on peut ainsi émettre (resp. recevoir) est la **largeur de la fenêtre d'anticipation** d'émission (resp. de réception).

- Dans l'exemple : la largeur  $W \geq 3$

Pour que la capacité de la liaison de données soit totalement utilisée il faut que :

- $W * L \geq T_a/r * D$ 
  - .  $L$  étant la longueur moyenne d'une trame,  $T_a/r$  la durée d'aller/retour et  $D$  le débit nominal de la liaison.

La largeur de fenêtre peut être :

- fixe
  - . par exemple : HDLC ou X25.3
- variable
  - . par exemple : TCP
  - . dans ce cas sa valeur instantanée est appelée crédit

© Ahmed Mehaoua 2006 - page 14

## CONTRÔLE DE FLUX par fenêtre d'anticipation et CONTRÔLE DES ERREURS par Ack/retransmission groupé ou sélectif

### OBJECTIF :

- Augmenter l'efficacité du dialogue
- Efficacité = délai d'émission des données / délai de transmission total

### PRINCIPE :

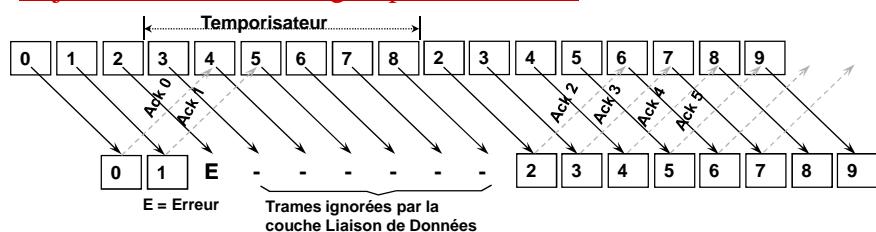
- Emission de plusieurs trames à la suite sans attendre la réception d'un ACK
- Une trame de supervision peut acquitter un groupe de trames de données
- Nombre de trames émises avant ACK =  $N-1$

1. REJET ET RETRANSMISSION GROUPEE (GO-Back-N) de toutes les trames à partir de la trame erronée ou perdue au moyen d'une trame de supervision REJ

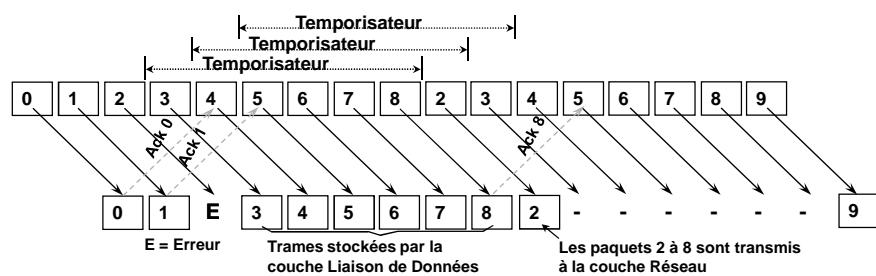
2. REJET ET RETRANSMISSION SELECTIVE (Selective Reject) au moyen de la trame de supervision SREJ

© Ahmed Mehaoua 2006 - page 15

### Rejet et Retransmission groupé : trame REJ

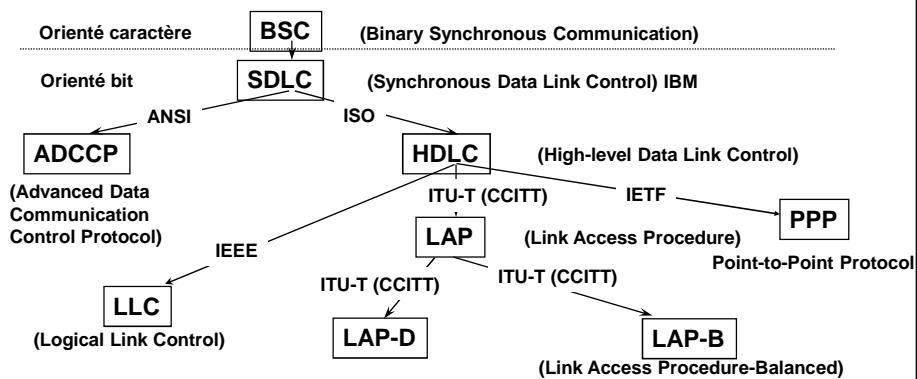


### Rejet et Retransmission sélectif : trame SREJ



© Ahmed Mehaoua 2006 - page 16

## PANORAMA DES PROTOCOLES DE LIAISON DE DONNEES



© Ahmed Mehaoua 2006 - page 17

## HDLC QU'est-ce que c'est ?

- HDLC offre un service de transfert de données fiable et efficace entre deux systèmes adjacents.

High-level Data Link Control :

- ISO 3309 : HDLC frame structure
- ISO 4335 : HDLC : elements of procedure,
- ISO 7448 : MultiLink procedure (MLP),
- ISO 7776 : LAP-B compatible link control procedure,
- ISO 7809 : Consolidated classes of procedures,
- ISO 8471 : HDLC balanced, link address information

- Utilisé comme protocole de la couche Liaison de données dans les normes X.25 (du CCITT) en usage dans les réseaux publics de transmission numériques de données (TRANSPAC, par exemple).

© Ahmed Mehaoua 2006 - page 18

## **HDLC HISTORIQUE**

1960 : BSC (“Binary synchronous communication”) - IBM

- tout premier protocole synchrone :

- . l’horloge du récepteur est maintenue synchronisée même s’il n’y a pas de transmission de données
- . transmission plus rapide (sans resynchronisation)
- . nécessite un contrôleur de communication spécialisé
- l’unité de transmission est le caractère (code ASCII (7 bits) ou EBCDIC (8 bits))
- . par abus : protocole “orienté” caractère

70 : SDLC (Synchronous data link control) - IBM

- l’unité de transmission est la trame
- normalisé par l’ANSI (“American national standard institute) sous le nom ADCCP (“Advanced data communication - control procedure”)

76 : HDLC (“High data link control”)

- protocole basé sur l’élément binaire (“orienté” bit)
- ISO 3309 : HDLC frame structure
- ISO 4335 : HDLC : elements of procedure

© Ahmed Mehaoua 2006 - page 19

## **HDLC HISTORIQUE (2/2)**

80 : adapté pour l’accès au réseau numérique de données

- LAP-B (“Link access procedure-balanced”):
  - . rôles équilibrés (symétriques) entre les deux systèmes adjacents
- normalisé : CCITT X25.2 et ISO 7776

85 : adapté aux réseaux locaux

- protocole de la sous-couche d’homogénéisation LLC (“Logical link control”)
- apparition d’un mode de transmission non connecté (LLC classe 1)
- normalisé : IEEE 802.2 et ISO 8802/2

Autres adaptations :

- Téléx : LAP-X - CCITT T71
- RNIS - canal D : LAP-D - CCITT Q921 ou I441

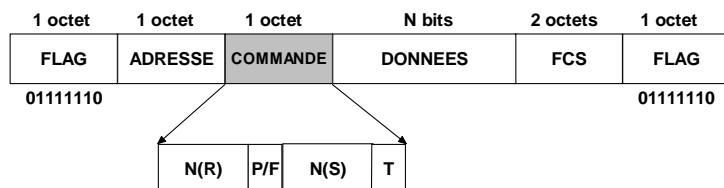
© Ahmed Mehaoua 2006 - page 20

## FONCTIONS

1. DELIMITATION et IDENTIFICATION des trames (Protocole)
2. GESTION de la liaison de données (Procédure) :
  - Etablissement et libération de la liaison de données sur un ou plusieurs circuits physiques préalablement activées,
3. SUPERVISION du fonctionnement de la liaison de données selon :
  - Le mode de transmission (synchrone ou asynchrone)
  - La nature de l'échange (simplex, half-duplex ou full-duplex)
  - Le type de liaison (point-à-point ou multipoint)
  - Le mode de l'échange (hiérarchique ou symétrique)
4. IDENTIFICATION de la source et du destinataire (Adressage)
5. CONTROLE D'ERREURS (Procédure)
6. CONTROLE DE FLUX (Procédure)

© Ahmed Mehaoua 2006 - page 21

## Format trame HDLC CHAMP COMMANDÉ



T (1 bit) : Indique le type de trame

N(S) et N(R) (6 bits) : Indique le numéro des trames émises et reçues

P/F (1 bit) : Demande de réponse immédiate à la suite de l'envoi d'une trame de commande

© Ahmed Mehaoua 2006 - page 22

## HDLC CHAMP COMMANDE

□ Trois types de trames :

- les trames d'**information** (*I Information*)
- les trames de **supervision** (*S Supervisory*)
- les trames **non numérotées** (*U Unnumbered*)

Elles se distinguent notamment par leur champ Commande :

Types de trame	Champ Commande						
	0	<b>N(S)</b>		P/F	<b>N(R)</b>		
<b>S</b>	1	0	Type	P/F	<b>N(R)</b>		
<b>U</b>	1	1	<b>M</b>	<b>M</b>	P/F	<b>M</b>	<b>M</b>

Note : deux formats du champ Commande existent :

- le format normal (8 bits)
- le format étendu (16 bits) : négocié lors de l'établissement de la connexion pour avoir un champ de commande plus grand et ainsi effectuer la numérotation modulo 128.

© Ahmed Mehaoua 2006 - page 23

## HDLC TRAMES DE SUPERVISION

4 types de trames de supervision,

- codées dans le sous-champ Type du champ Commande
  - commande ou réponse
- (ACK + CF) - **RR** ("Received & Ready") - 00 : acquittement
  - . confirme la réception des trames de données de  $n^o < N(R)$
  - . demande la transmission des trames suivantes
- (ACK + CF) - **RNR** ("Received & Not Ready") - 10 : contrôle de flux
  - . confirme la réception des trames de données de  $n^o < N(R)$
  - . interdit la transmission des trames suivantes
- (ACK + RET) - **REJ** ("Reject") - 01 : protection contre les erreurs
  - . confirme la réception des trames de données de  $n^o < N(R)$
  - . demande la retransmission des trames de  $n^o \geq N(R)$
- (ACK + RET) - **SREJ** ("Selective Reject") - 11 : protection contre les erreurs
  - . confirme la réception des trames de données de  $n^o < N(R)$
  - . demande la retransmission de la trame de  $n^o = N(R)$
  - . non-utilisée par LAP-B

© Ahmed Mehaoua 2006 - page 24

## HDLC TRAMES DE GESTION

Trame d'établissement de la connexion - commande :

- **SABM** (Set asynchronous balanced mode) - en format normal
- **SABME** (Set asynchronous balanced mode extended) - en format étendu

Trame de libération de la connexion - commande :

- **DISC** (Disconnection)

Trame de confirmation - réponse :

- **UA** ("Unnumbered acknowledgment") :

Trame de récupération des erreurs -réponse :

- **FRMR** ("Frame reject") :

Trame d'indication de connexion libérée

- **DM** ("Disconnected mode")

© Ahmed Mehaoua 2006 - page 25

## HDLC Trames I, S, U

bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	
Nr		P/F		Ns		0		frame I
								RR
				0	0	0	1	RNR
				0	1	0	1	REJ
				1	0	0	1	SREJ
Nr		P/F		1	1	0	1	
0	0	0	P	1	1	1	1	SARM
1	0	0	P	0	0	1	1	SNRM
0	0	1	P	1	1	1	1	SABM
0	1	1	P	1	1	1	1	SABME
0	1	0	P	0	0	1	1	DISC
0	1	1	F	0	0	1	1	UA
1	0	0	F	0	1	1	1	CMDR/FRMR
0	0	0	F	1	1	1	1	DM

© Ahmed Mehaoua 2006 - page 26

## HDLC VARIABLES

- ❑ Chaque entité tient à jour les trois variables suivantes :
  - **V(S)** = numéro de la prochaine trame d'information à émettre,
  - **V(R)** = numéro de la prochaine trame à recevoir,
  - **DN(R)** = numéro du dernier acquittement reçu.
- ❑ et connaît les constantes suivantes :
  - **T1** = délai de garde au bout duquel une trame non acquittée est réémise.
  - **T2** = délai d'acquittement pendant lequel le récepteur peut retarder l'envoi de l'acquittement d'une trame.
  - **N1** = taille maximum d'une trame.
  - **N2** = nombre maximum de réémissions d'une même trame.
  - **W** = largeur de la fenêtre.
  - etc.

© Ahmed Mehaoua 2006 - page 27

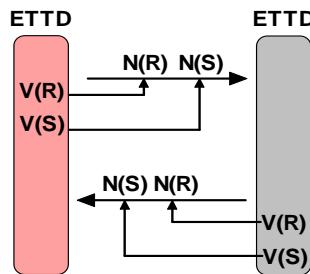
## HDLC NUMEROTATION DE TRAMES

**V(S)** : numéro de la prochaine trame à envoyer (0 à 7)

**V(R)** : numéro de la prochaine trame attendue en réception (0 à 7)

**N(S)** : numéro de la trame

**N(R)** : acquittement des trames reçues de numéro strictement inférieur à **N(S)**



© Ahmed Mehaoua 2006 - page 28

## HDLC TIMERS ET PARAMETRES

TAILLE MINI d'une trame HDLC : **32 octets**

TAILLE MAX d'une trame HDLC : **1150 octets**

Nombre de trames de la fenêtre d' anticipation : **W inférieur ou égal à 7, paramétrable**

**TIMER T1 :**

Durée maximale d'attente d'un acquittement à l'émission d'une trame.

L'expiration de T1 sans réception de ACK entraîne la retransmission de la première trame émise non acquittée.

**T1 = 100,200,400,800,1600 ou 2550 ms paramétrable**

**N2 :**

Nombre maximale de réémissions de la même trame I, avant de considérer la liaison hors service (**N2=10**).

**TIMER T2 :**

Durée maximale d'attente avant d'acquitter une trame reçue, au moyen d'une trame de supervision si aucune trame I disponible.

Temps de transmission de la trame la plus longue : **soit 1150 octets**

© Ahmed Mehaoua 2006 - page 29

## HDLC ENVOI DE TRAMES

### Emission d'une trame I

Vérifier que  $V(S) < DN(R) + W$  puis :

- **N(S) = V(S)** et **N(R) = V(R)** ;
- mémoriser la trame;
- incrémenter **V(S) modulo N**;
- armer le temporisateur (délai de garde **T1**) associé à la trame;
- désarmer T2.

### Emission d'une trame REJ

- **N(R) = V(R)**
- désarmer T2.

### Emission d'une trame RR

- **N(R) = V(R)**
- désarmer T2.

© Ahmed Mehaoua 2006 - page 30

## HDLC : RECEPTION DE TRAMES

- ❑ Sur réception d'une trame
  - Si la trame est invalide
    - alors la trame est ignorée (si FCS incorrect) ou émission d'une trame FRMR (format incorrect).
- ❑ Sur réception d'une trame I
  - Si  $N(S) \neq V(R)$ 
    - alors trame non-attendue (déséquencée)
      - . émettre un trame REJ;
    - sinon /\*  $N(S) = V(R)$  \*/
      - . Armement du temporisateur T2 (délai d'acquittement) associé à  $N(S)$ ;
      - . incrémentation de  $V(R)$ .
    - Si  $DN(R) \leq N(R) < V(S)$  alors
      - . désarmer les temporisateurs T1 des trames de n° compris entre  $DN(R)$  et  $N(R)$ ;
      - .  $DNR(R) = N(R)$  ;
- ❑ Sur réception d'une trame RR
  - Si  $DN(R) \leq N(R) < V(S)$  alors
    - . désarmer les temporisateurs T1 des trames de n° compris entre  $DN(R)$  et  $N(R)$ ;
    - .  $DNR(R) = N(R)$ .

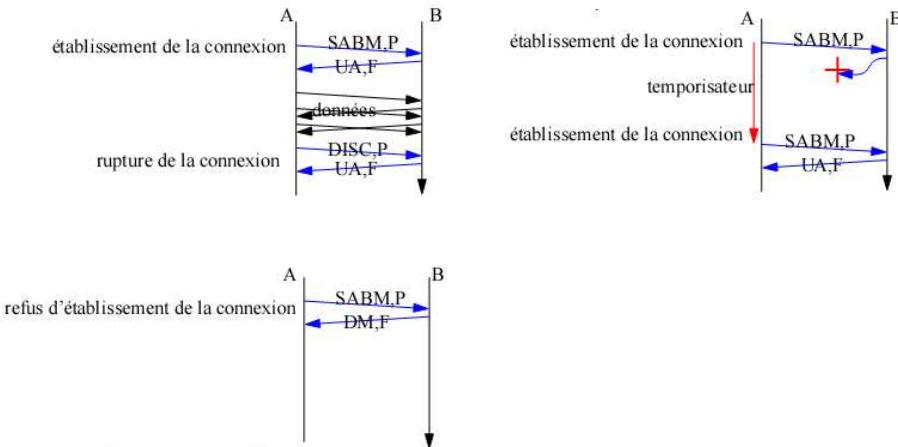
© Ahmed Mehaoua 2006 - page 31

## HDLC RECEPTION DE TRAMES (2/2)

- ❑ Sur réception d'une trame REJ
  - Si  $DN(R) \leq N(R) < V(S)$  alors
    - . désarmer les temporisateurs T1 des trames de n° compris entre  $DN(R)$  et  $N(R)$ ;
    - .  $DNR(R) = N(R)$  ;
    - . Emettre les trames de numéros compris entre  $N(R)$  et  $V(S)$ .
- ❑ A l'expiration du délai T1 associé à une trame
  - Si le nombre de retransmissions n'est pas dépassé ( $< N2$ )
    - . alors on réemet la trame I telle qu'elle a été mémorisée.
- ❑ A l'expiration du temporisateur T2
  - émettre une trame RR.

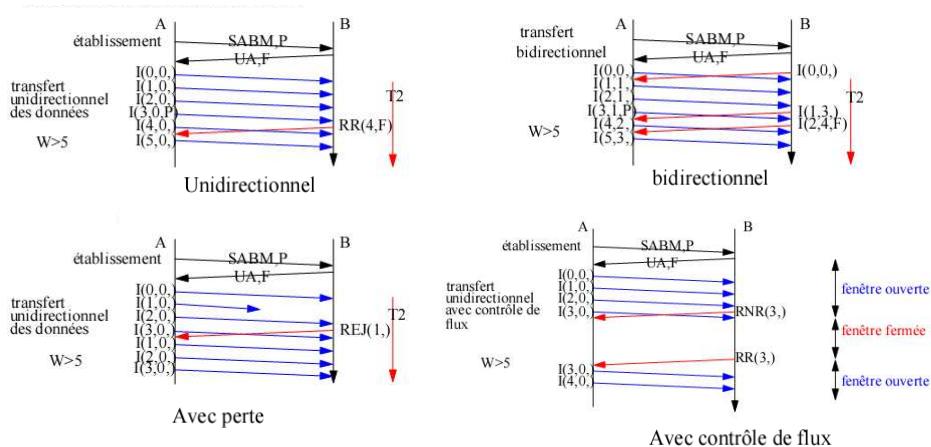
© Ahmed Mehaoua 2006 - page 32

## HDLC ETABLISSEMENT/CLOTURE DE LIAISON



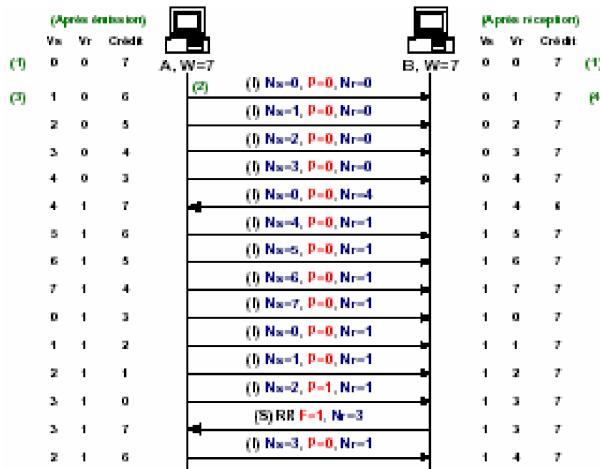
© Ahmed Mehaoua 2006 - page 33

## HDLC SCENARIOS D'ECHANGES



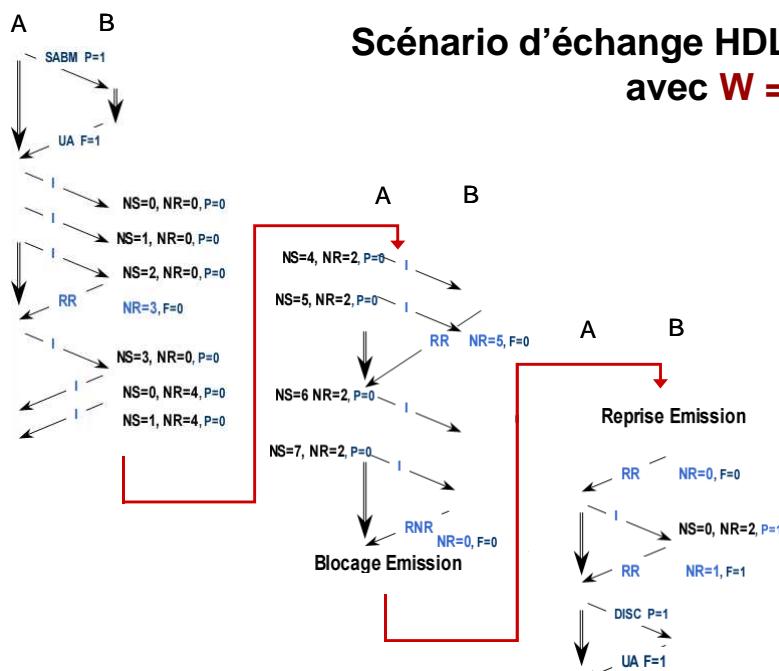
© Ahmed Mehaoua 2006 - page 34

## HDLC SCENARIOS D'ECHANGES



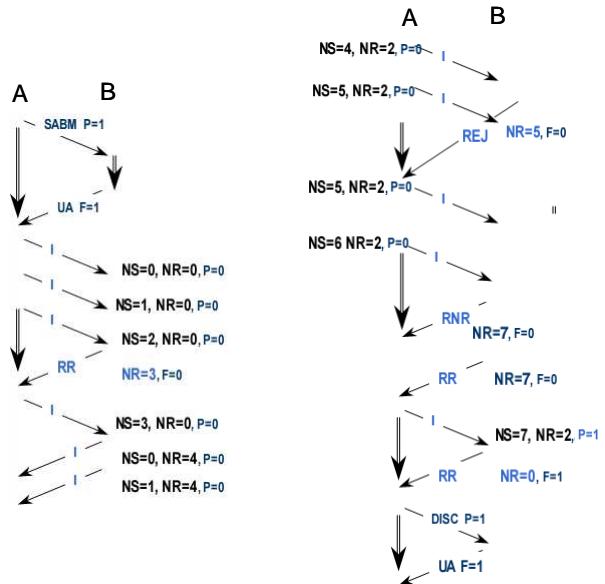
© Ahmed Mehaoua 2006 - page 35

### Scénario d'échange HDLC avec $W = 3$



© Ahmed Mehaoua 2006 - page 36

## Scénario d'échange HDLC avec $W = 3$ (suite)



© Ahmed Mehaoua 2006 - page 37

# **Les Réseaux Locaux Informatiques**

## **- Ethernet -**

### **PLAN**

#### **1. Les principes des réseaux locaux informatiques (RLI)**

Définition : supports, topologies, techniques d'accès  
Architecture IEEE

#### **2. le RLI Ethernet ou 802.3**

Transmission physique  
Trames 802.3 et adressage (MAC)  
Technique d'accès au canal (CSMA/CD)  
Le protocole de contrôle de la liaison (LLC)  
Réseaux Locaux Virtuel (VLAN)



## DEFINITION ET PROPRIETES D'UN RLI

- Les **réseaux locaux informatiques** (en anglais **LAN**, *Local Area Network*) sont destinés principalement aux communications locales, généralement au sein d'une **même entité** (entreprise, administration, etc), sur de **courtes distances** (quelques kilomètres au maximum).
- Les RLI ont donné lieu à des normes définies par l'IEEE dans le groupe 802.

Les RLI sont caractérisés par :

- **1. Une Gestion privée et autonome du réseau**
- **2. un Support physique partagé**
- **3. un mode de communication par diffusion**
- **4. une Transmission numérique en bande de base**



## PARTICULARITES DES RLI

- ♦ Les conséquences des particularités techniques des RLI sont :

### 1. **Les problèmes des accès concurrents**

- trouver une technique de partage du média (si possible équitable)
- pris en charge par le protocole de niveau liaison

### 2. **Les problèmes de confidentialité et de sécurité**

- Exemple : interception des mots de passe des usagers
- pris en charge par les systèmes d'exploitation et les applications (cryptographie)



## CARACTERISTIQUES DES RLI

- ♦ Les différentes solutions de RLI se distinguent par trois choix techniques

### 1. Le type de topologie

Bus, Etoile, anneau, arbre ...

### 2. Le type de support physique

cuivre, coaxial, fibre optique, radio, ...

### 3. La technique d'accès au support

centralisée/distribuée; aléatoire/déterministe, ...

**1 + 2 + 3 = un réseau local informatique particulier**

### Exemple de RLI : Ethernet 10baseT

Ethernet : protocole MAC de type distribué et aléatoire

10 : Débit de 10 Mbp/s

T : support de type paire de cuivre torsadée

Base : Transmission numérique en bâton de base

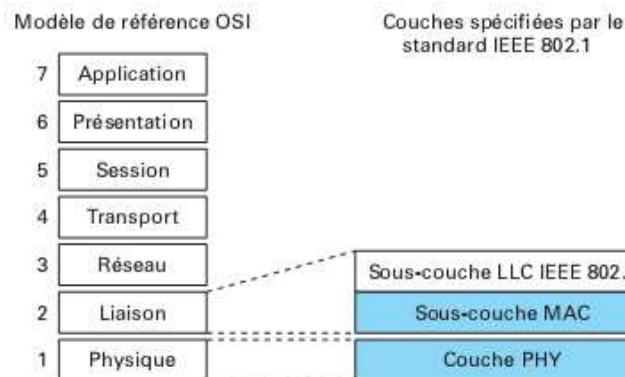
Topologie en étoile (autour d'un concentrateur)



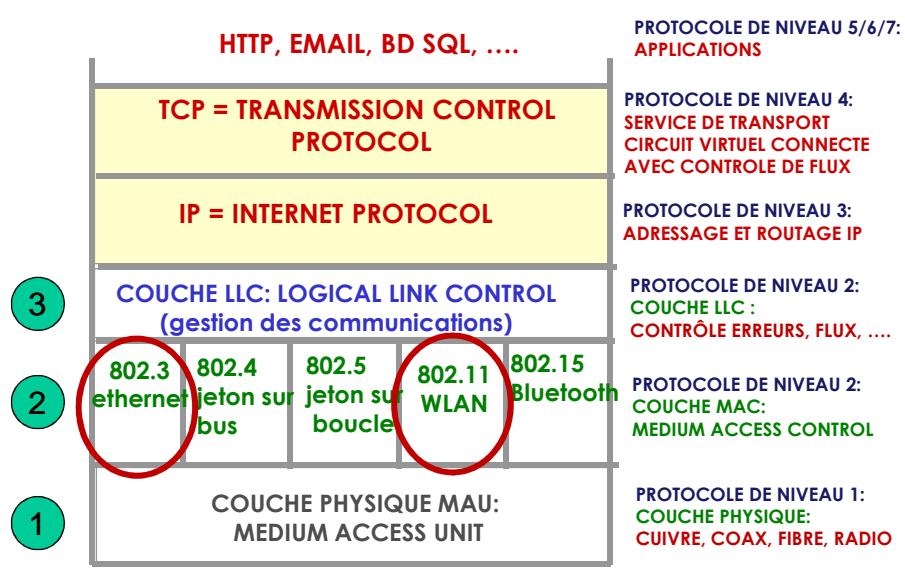
## ETHERNET : Un peu d'histoire

- ♦ 1974 : Inventeur XEROX : Spécification de Ethernet
- ♦ 1976 : INTEL et DIGITAL proposent Ethernet v2 et en font un standard du marché
- ♦ 1980 : IEEE normalise :
  - La technique d'accès de Ethernet (CSMA/CD 802.3)
  - La gestion des collisions
    - Notifications (bourrage de la ligne - JAM)
    - définit la variante CSMA-persistant
    - Algorithme de reprise après collision (Binary Exponential Backoff)
  - Les algorithmes d'émission et de réception
  - Les grandeurs physiques IEEE 802.3 (délais, distances, ...)
  - La structure de la trame Ethernet 802.3
  - Les spécifications des supports physiques
- ♦ 2000 : Ethernet et ses dérivées représentent 80% du marché des LAN

## ARCHITECTURE RLI



## ARCHITECTURE : IEEE 802 LAN et WLAN

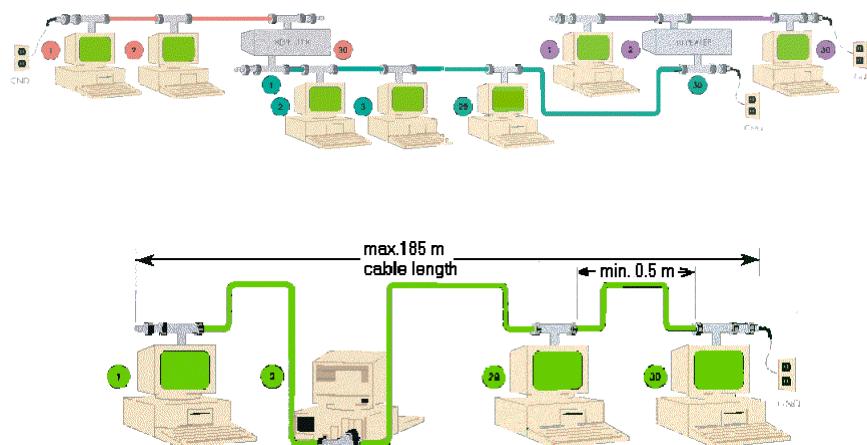


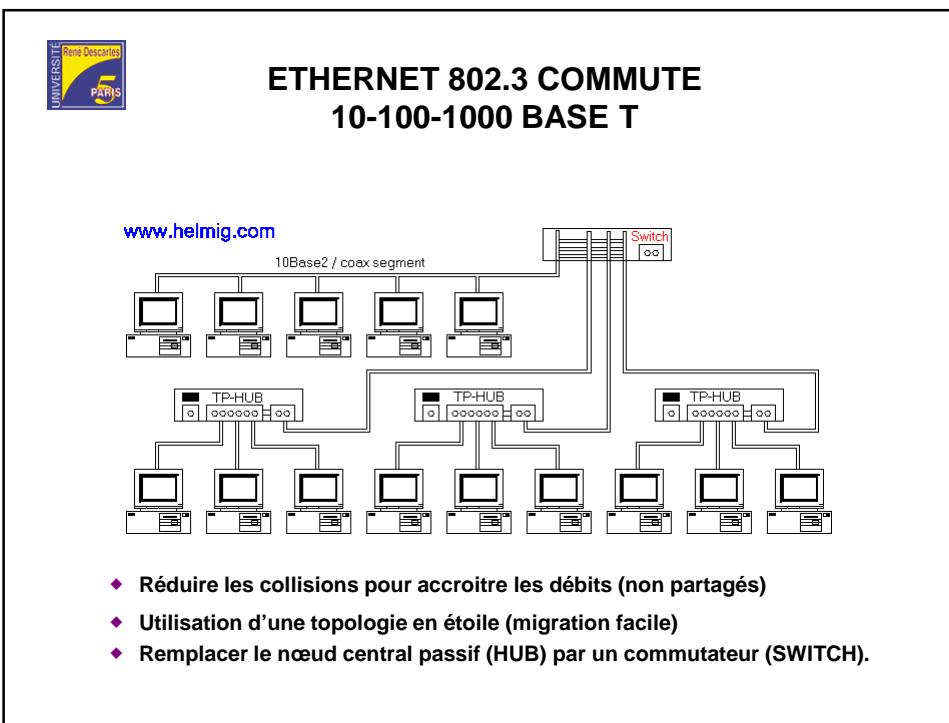
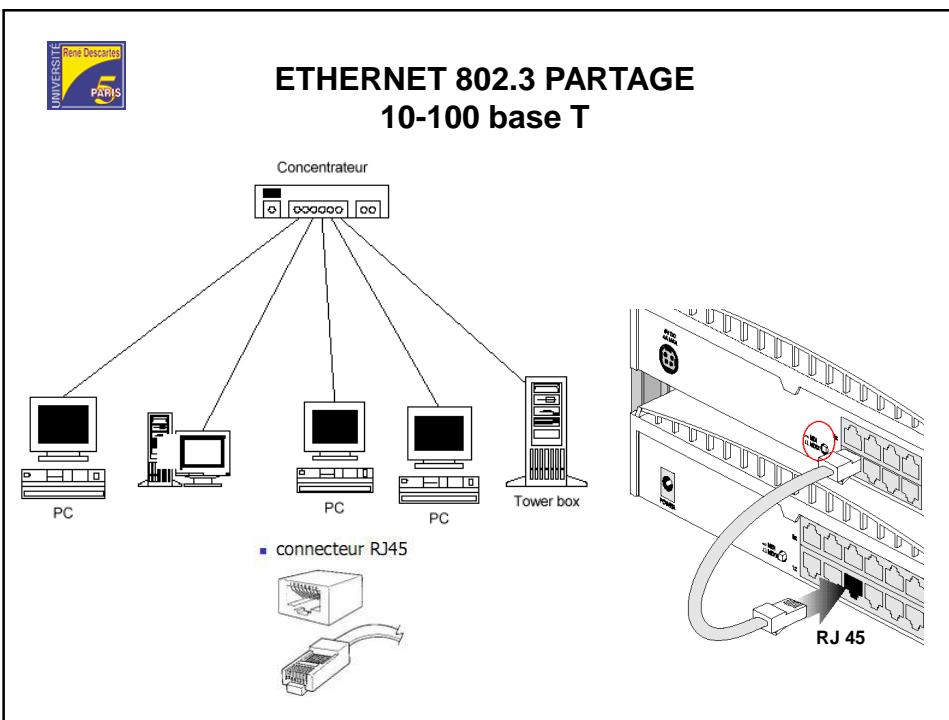
## NORMES 802

- 802.1 High Level Interface, Network Management, Bridging, Glossary
- 802.2 Logical Link Control
- **802.3 CSMA/CD Ethernet**
- 802.4 Token Bus
- 802.5 Token Ring (LAN IBM)
- 802.6 Metropolitan Area Network (DQDB : Double Queue Dual Bus)
- 802.7 Broadband LAN Technical Advisory Group
- 802.8 Fiber Optic Technical Advisory Group
- 802.9 Integrated Service LAN (IsoEthernet), pour isochrone (temps réel)
- 802.10 LAN Security (SILS : Standard for Interoperable LAN Security)
- **802.11 Wireless LAN**
- 802.12 Demand Priority LAN (100VG - AnyLAN)
- 802.14 Cable TV MAN
- **802.15 Wireless Personal Area Network (WPAN), bluetooth**
- **802.16 Fixed Broadband Wireless Access** (sans fil large bande)



## ETHERNET 802.3 10 base 5 et 10base 2





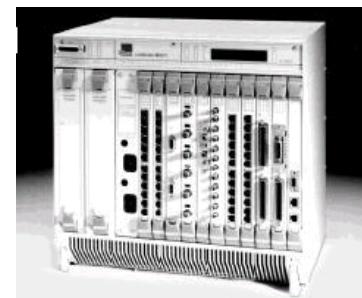


## REPEATER / HUB / SWITCH

Répéteur/adaptateur (UNICOM)



hubs 16/8 ports (HP)



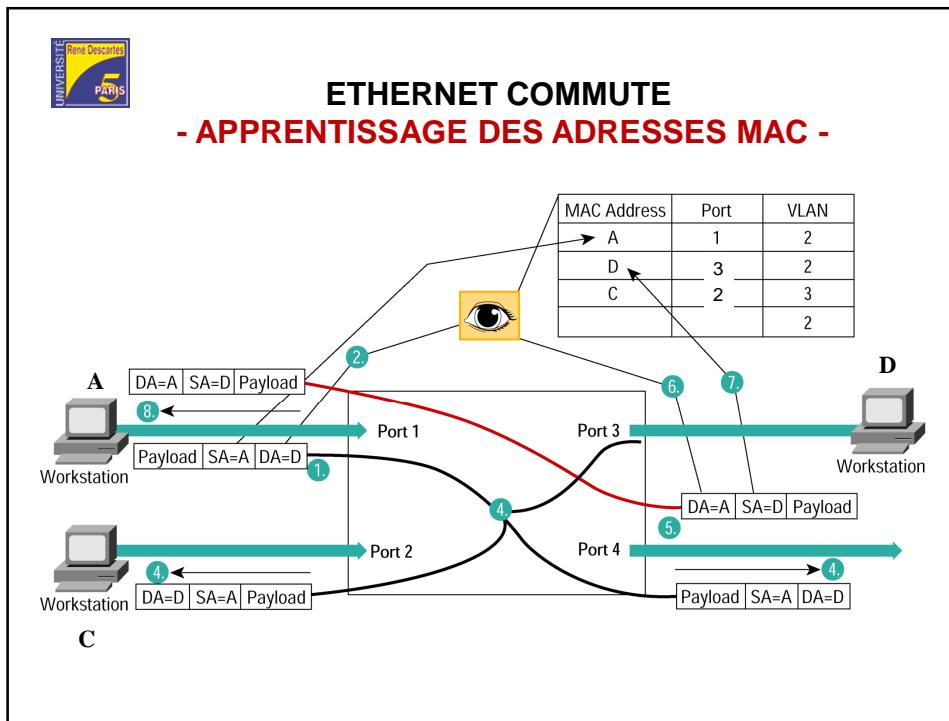
Switch multi Protocole (3com)



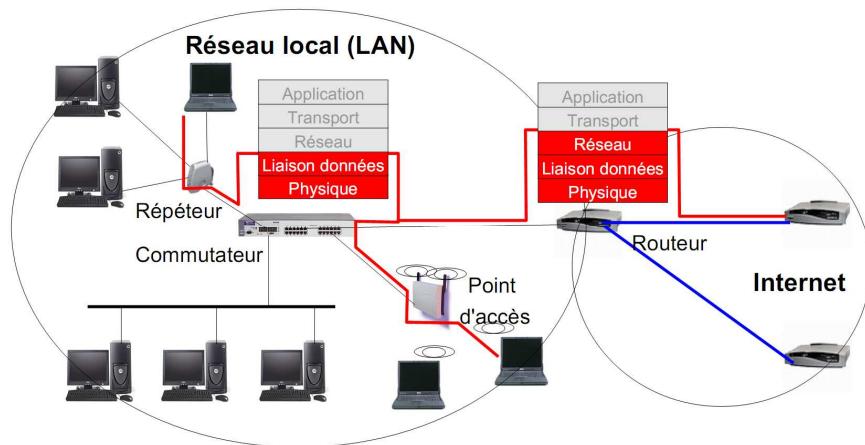
Commutateur/ Switch Netgear



Switch empilables



## ARCHITECTURE IEEE 802 LAN / WLAN

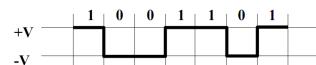


page 15

## ETHERNET 802.3 Transmission Physique

- Codage unipolaire sans retour à zéro (NRZ)

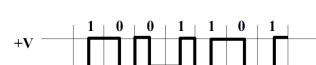
– Machine (horloge)



- Codage Manchester (simple)

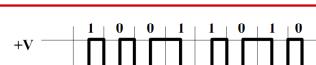
– Inclus le signal d'horloge

- $\frac{1}{2}$  temps bit à l'inverse de la valeur  
+  $\frac{1}{2}$  temps bit à la valeur.



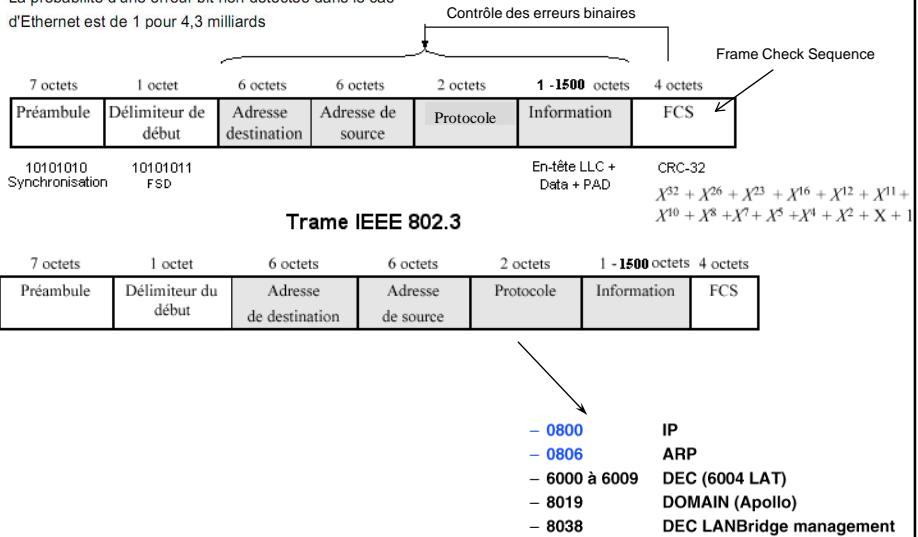
- Codage Manchester différentiel

- Bit 0 = Changement de polarité
- Bit 1 = Polarité du début temps bit identique à précédente  
– Le sens des fils n'a plus d'importance.



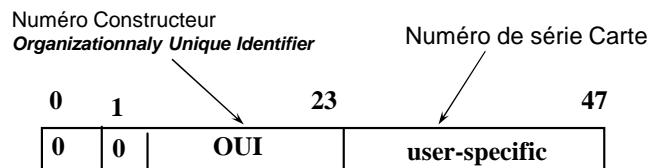
## FORMAT TRAME ETHERNET 802.3

La probabilité d'une erreur bit non détectée dans le cas d'Ethernet est de 1 pour 4,3 milliards



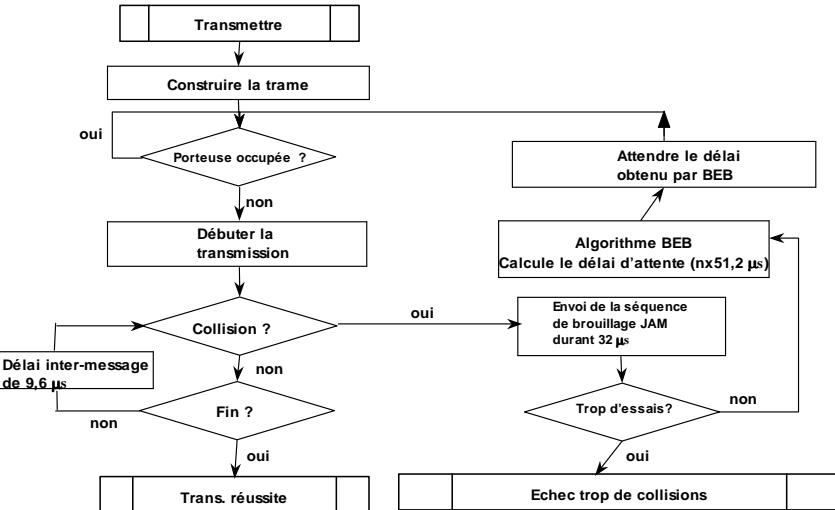
## ADRESSE UNIVERSELLE MAC 802 (2) - Résolution d'adresses (ARP) -

- 00:00:0C:XX:XX:XX : Cisco
- 08:00:20:XX:XX:XX : Sun
- 08:00:09:XX:XX:XX : HP
- 08:00:14:XX:XX:XX : Excelan

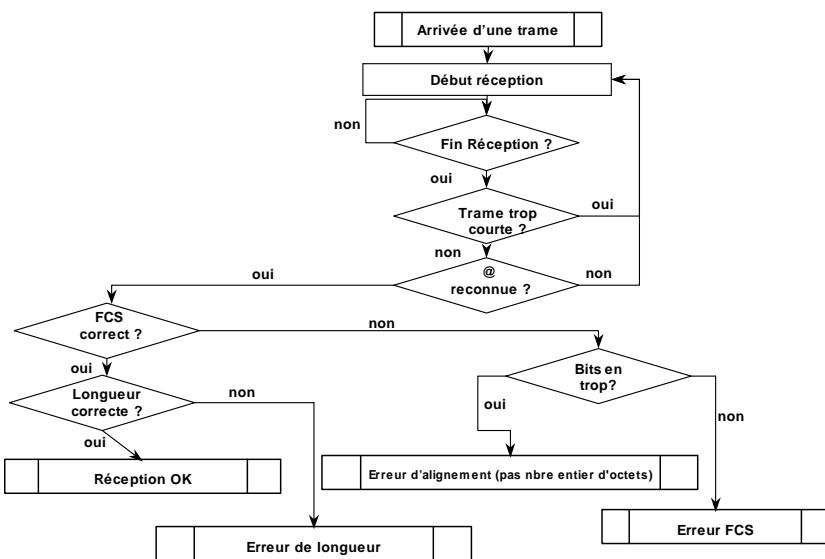


- Le protocole ARP (Adress Resolution Protocol) permet aux stations d'un RLI de trouver automatiquement l'adresse MAC d'une station distante en ne connaissant que son adresse IP.

## MAC 802.3 PRINCIPE D'EMISSION (2)



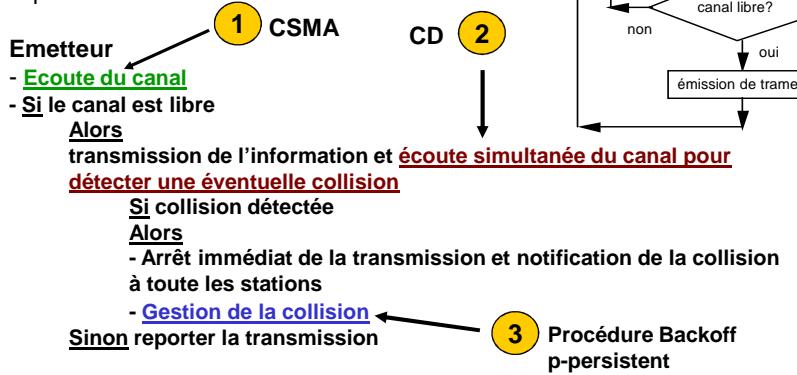
## ETHERNET 802.3 PRINCIPE DE RECEPTION (2)



## CSMA/CD

(CSMA with Collision Detection)

- Amélioration de la méthode CSMA p-persistant :
- Principe :



## ETHERNET 802.3

### ALGORITHME BACKOFF (BEB)

- La procédure BACKOFF utilise 3 fonctions :
    - random()** : tire un nombre réel aléatoire entre 0 et 1.
    - int()** : rend la partie entière d'un réel
    - délai()** : calcul le délai d'attente multiple d'un slot\_time (51.2 microsec) et est compris entre  $[0, 2^k]$ .
- Avec  $k = \min(n, 10)$ ,  $n = \text{nbre de ré-émission déjà faites}$

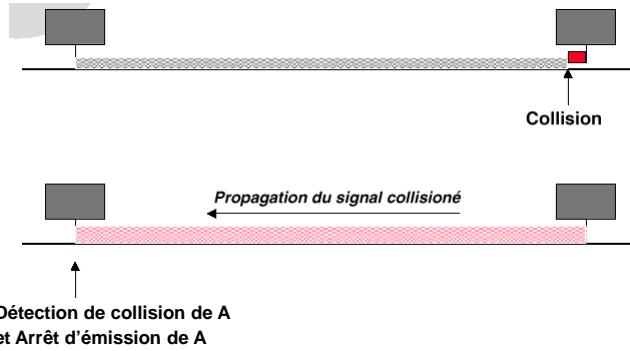
```

Procédure BACKOFF (no_tentative : entier, VAR maxbackoff : entier)
Const slot-time=51.2 (microsecondes); limite_tentative=16;
Var délai : entier;

BEGIN
  Si (no_tentative =1)
    Alors  maxbackoff =2 (borne de temps d'attente maximale)
  Sinon
    Si (tentative < limite_tentative)
      Alors  maxbackoff = maxbackoff*2;
      Sinon  maxbackoff =  $2^{10}$  (au delà de 10 essais la borne devient constante)
    fsi
    délai := int(random() *maxbackoff)
    attendre (délai*slot_time)
  END

```

## ETHERNET 802.3 SPECIFICATION DES GRANDEURS PHYSIQUES (2)

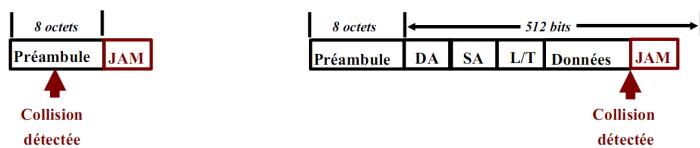


Temps max écouté = Aller + Retour (RTT Round Trip Time) =  $2 \times \text{Distance} / V$   
 Temps d'émission =  $T_e = \text{Longueur de la trame} / \text{Débit du canal}$

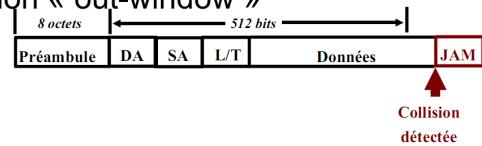
Pour que CSMA/CD fonctionne correctement =>  $T_e \geq RTT$

## ETHERNET 802.3 COLLISIONS ET BROUILLAGE

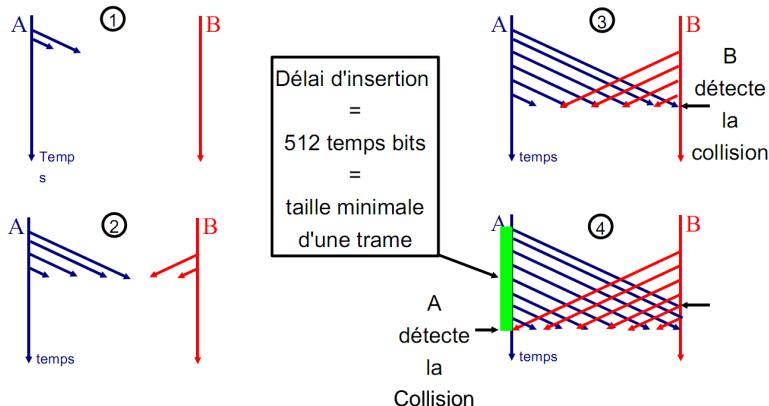
- Collisions « in-window »



- Collision « out-window »



## ETHERNET 802.3 DELAI D'INSERTION et DETECTION COLLISIONS



## ETHERNET 802.3 SPECIFICATION DES GRANDEURS PHYSIQUES (3)

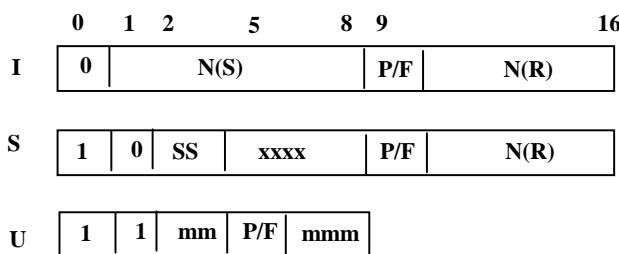
Paramètres	Valeurs
Tranche canal	→ 512 temps bits (64 octets)
Slot-time (10 Mbps)	→ 51.2 µs
Silence inter messages	→ 9.6 ms
Nombre d'essais total	→ 16 (15 retransmissions)
Limite tirage BEB	→ 10
Taille mini. du brouillage	→ 32 bits
Taille maxi. des trames	→ 1526 octets
Taille mini. des trames	→ 64 octets (46 octets pour Data)
Taille des adresses	→ 6 octets

## LOGICAL LINK CONTROL (LLC) 802.2

- ♦ Le but du protocole LLC est de fournir une garantie de livraison des messages appelés LSDU (Link Services Data Unit), la détection et la reprise sur erreur. L'envoi d'un datagramme (ou paquet ne garantit pas à son émetteur que le ou les destinataires ont reçu ce message).
- ♦ Sous-couche commune des sous-couches MAC (**Dérivée de HDLC**)
- ♦ Propose 3 niveaux de service (qualité):
  - **LLC1** - service sans connexion et sans acquittement
  - **LLC2** - service avec connexion et ack
  - **LLC3** - service sans connexion et avec acquittement au choix



## TYPES TRAMES LLC 802.2



- **SS=00:** RR (Receive Ready=Prêt à recevoir),
- **SS=10:** RNR (Receive Not Ready=Non prêt à recevoir),
- **SS=01:** REJ (Reject=Rejet).

**MM-MMM :**

- SABME: Demande d'ouverture de Connexion mode asynchrone équilibré étendu,  
UA: [Réponse] Acquittement non numéroté  
DM: [Réponse] La liaison est déconnectée (Ack négatif suite à un SABME).  
DISC: Fermeture d'une connexion : Disjonction  
FRMR: Rejet du LPDU en raison d'une erreur (la cause et le diagnostic sont mentionnés),  
XID: Echange d'identité entre deux entités LLC,  
TEST: Utilisé pour tester une liaison,  
UI: Information non numérotée (trames porteuses de données)

## DIFFERENCES ENTRE LLC2 et HDLC

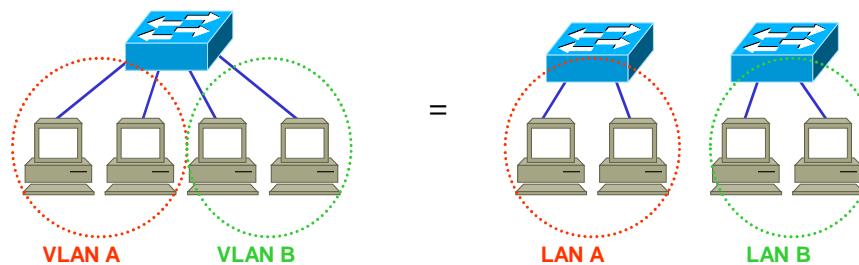
1. La taille maximale de la fenêtre d'anticipation est de **7** dans HDLC et **127** dans LLC.
2. La commande **SREJ** n'existe pas dans LLC (utile pour un canal bruité avec RTT long)
3. HDLC offre plusieurs modes de connexion, alors que dans un réseau local, seul le mode **ABME** (mode asynchrone équilibré étendu) a un sens.
4. Les LPDU **XID** et **TEST** ont été introduits dans LLC pour le besoin du trafic sans connexion. En effet, XID sert à échanger le type de LLC ainsi que la taille de la fenêtre d'anticipation; TEST sert à tester si une liaison logique est active ou pas.

On répond à un XID ou TEST par un LPDU de la même nature.

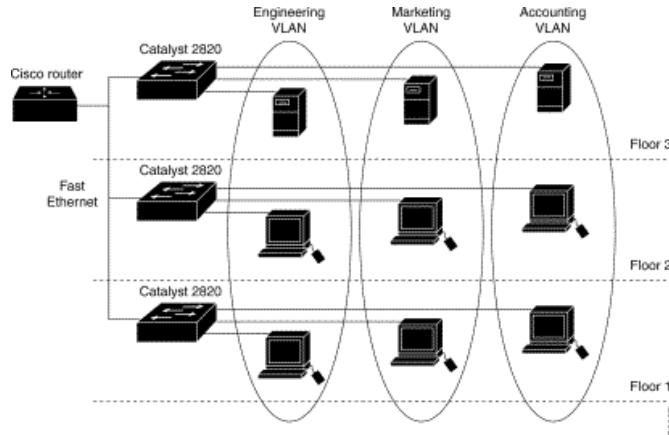
## VLAN: Définition

Définition : Virtual Local Area Network

Utilité : Plusieurs réseaux virtuels sur un même réseau physique



## VLAN: Architecture



## Typologie des VLAN

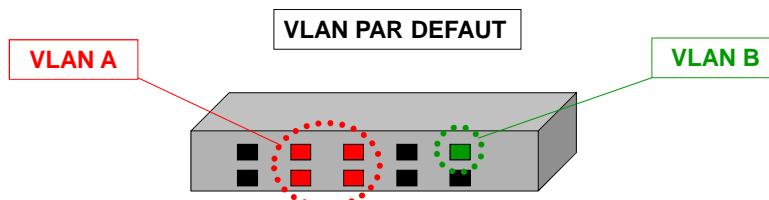
Plusieurs types de VLAN sont définis, selon le critère de commutation et le niveau auquel il s'effectue :

1. **Un VLAN de niveau 1** (aussi appelés VLAN par port, en anglais *Port-Based VLAN*) définit un réseau virtuel en fonction des ports de raccordement sur le commutateur ;
2. **Un VLAN de niveau 2** (également appelé VLAN MAC ou en anglais *MAC Address-Based VLAN*) consiste à définir un réseau virtuel en fonction des adresses MAC des stations. Ce type de VLAN est beaucoup plus souple que le VLAN par port car le réseau est indépendant de la localisation de la station; le défaut est que chaque station doit être manuellement associée à un VLAN.

## VLAN: de niveau 1

**VLAN de niveau 1 ⇔ VLAN par port**

- 1 port du switch dans 1 VLAN
- configurable au niveau de l'équipement
- 90% des VLAN sont des VLAN par port



## VLAN et QOS - IEEE 802.1p et 802.1q -

Tame Ethernet non 802.1p

Destination	Source	Type / Longueur
-------------	--------	-----------------

Tame Ethernet étendue 802.1p

Destination	Source	Tag Control Info	Type / Longueur
-------------	--------	------------------	-----------------

Type de trame	Priorité	Canonical	802.1q VLQN identifiant
2 bytes	3 bits	3 bits	12 bits

Sous champ de contrôle	Description
Type de frame marquée	Toujours à 8100h (type trame Ethernet)
Champ priorité (802.1p)	Valeur représentant le niveau de priorité
« Canonical »	Toujours à 0
802.1q VLQN identifiant	Numéro d'identification du VLAN

# **INTERNET : introduction et adressage**

page 1

## **Plan**

1. Introduction à L'INTERNET: Historique et définitions
2. Protocoles IP : Adressage
3. Protocole ARP : Résolution d'adresses
4. Protocole ICMP : contrôle des erreurs
5. Mode d'accès à Internet : ADSL, PPP

page 2

## Bibliographie

- **TCP/IP : Principes, protocoles et Architecture**  
Douglas E .Comer, Prentice Hall - 4ème édition - 754 pages
- **TCP/IP Illustré vol. 1, 2 et 3**  
W. Richard Stevens, Addison-Wesley 1996
- **Routage dans l'Internet**  
Christian Huitema, Prentice Hall - 2ème édition - 384 pages
- **Réseaux locaux et Internet : Des protocoles à l'interconnexion**  
Laurent toutain, Hermès - 2 ème édition - 732 pages

page 3

## Historique

- ◆ **1969** : Début du réseau (D)ARPANET (4 calculateurs)
- ◆ **DARPA** = Defense Advanced Research Projects Agency
- ◆ **1972** : Démonstration de ARPANET
  - ◆ IMP - Interface Message Processor - mode connecté (X.25)
  - ◆ NCP - Network Control Program – non connecté (ancêtre de TCP)
- ◆ **1977-1979** : Les protocoles TCP/IP prennent leur forme définitive,
- ◆ **1980** - L'université de Berkeley intègre TCP/IP dans Unix (BSD)
- ◆ **1980** - janvier 1983 : Tous les réseaux raccordés à ARPANET sont convertis à TCP/IP

page 4

## Historique (2)

- ◆ 198x – TCPIP devient le Standard de facto pour l'interconnexion de réseaux hétérogènes,
- ◆ 1988 – Mise en place du Backbone de la NSFnet (12 réseaux régionaux)
- ◆ 1992 – EBone et RENATER
- ◆ 199x - explosion de l'offre et de la demande de services Internet y compris pour les particuliers
- ◆ 1995 – Arrêt du Backbone NSFnet
  - ◆ Mise en œuvre des NAPs (Network Access Points)
- ◆ 200x – Internet nouvelle génération

page 5

## Qu'est ce qu'Internet ? 3 définitions

1. Une **famille de protocoles** de communication, appelée :
  - TCP / IP : Transmission Control Protocol / Internetworking Protocol,
  - ou Internet Protocol Suite,
2. Un **réseau mondial** constitué de milliers de réseaux hétérogènes, et interconnecté au moyen des protocoles TCP/IP :
  - Réseaux locaux d'agences gouvernementales, institutions d'éducation, hôpitaux, des commerciaux, ...
  - Réseaux fédérateur de Campus,
  - Réseaux Régionaux, Nationaux, Intercontinentaux (Américains, Européen, Eunet, Ebone, Asiatiques, ...)
3. Une **communauté** de personnes utilisant différents services
  - Courrier électronique, Web, Transfert de fichiers FTP, ...

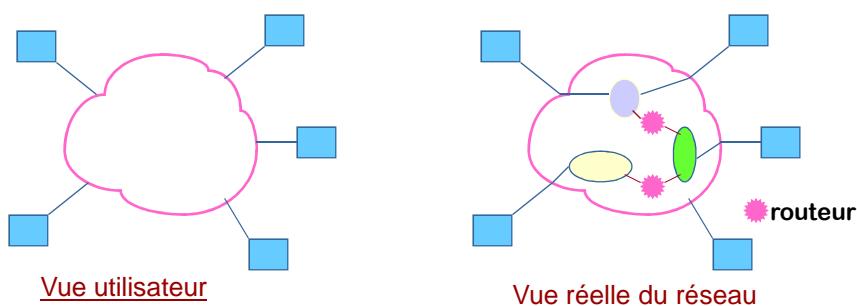
page 6

## Qu'est ce qu'un Intranet ou Extranet ?

1. **Intranet** : un réseau d'entreprise dans lequel les mêmes technologies et protocoles que l'Internet sont mis en œuvres
  - Routeurs, protocoles TCP/IP, protocoles applicatifs : email, web, ...
2. **Extranet** : un Intranet qui offre des accès distants aux usagers/employés/partenaires de l'entreprise
  - Problème de sécurité

page 7

## Structure Physique de l'INTERNET



page 8

## Qui normalise l'Internet ?

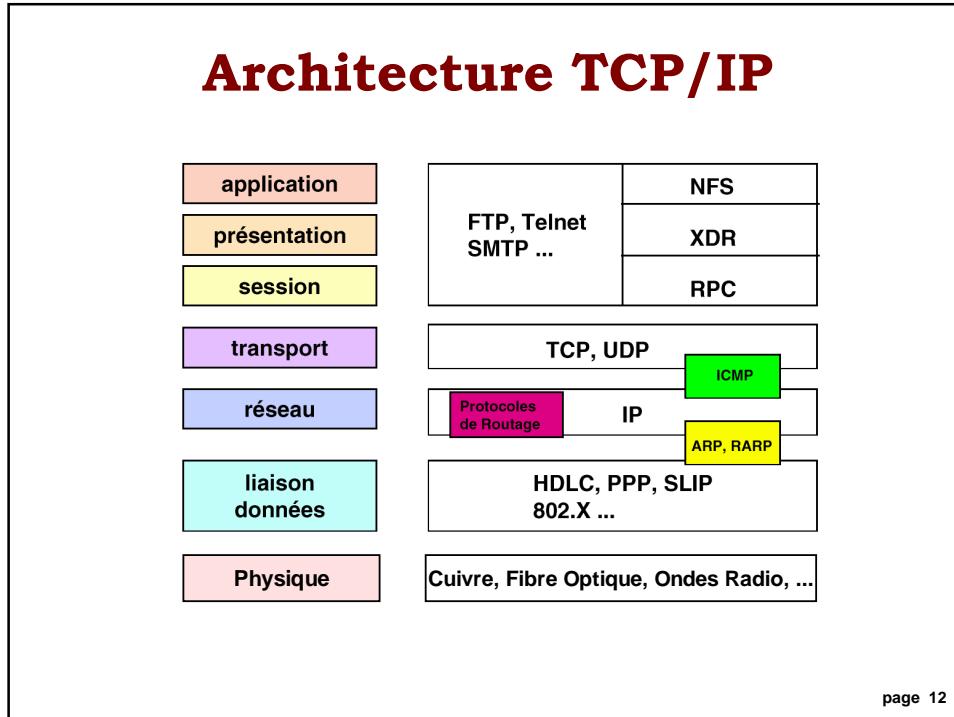
- Technologie INTERNET développée par un organisme bénévole : l'**IETF** (Internet Engineering Task Force) organisé en 8 secteurs de recherche,
- Les normes sont appelées **RFC** (Request For Comment),
  - Exemple : RFC 791 (déscriit IP) – RFC 793 (déscriit TCP)
  - Documents gratuits accessibles à « [www.ietf.org](http://www.ietf.org) »
- Tout le monde peut proposer un RFC
  - L'IAB (Internet Activities Board) gère le processus d'acceptation des RFC
- les standards sont publiés par une association sans but lucratif l'internet society (1992)

page 9

## Qui gère Internet

1. Normes techniques : IETF (internet Engineering Task Force)  
Les normes sont appelées **RFC** (Request For Comment),
  - Exemple : RFC 791 (déscriit IP) – RFC 793 (déscriit TCP)
  - Documents gratuits accessibles à « [www.ietf.org](http://www.ietf.org) »
2. Noms de domaines: **ICANN** (USA), RIPE (France)  
ICANN : Internet corporation for Assigned Names and Numbers;
3. Adresses IP, N°port, N°AS : **ICANN** depuis décembre 1998;
4. Réseaux : ISP (Internet Service Provider), NSP (Network Service Provider)
5. Fibres : Opérateurs télécoms
6. Serveurs, contenus : tout le monde (particuliers, entreprises, université, ...)

page 10



## Normes et RFC

- Couche Liaison :
  - **SLIP** : Serial Line IP RFC 1055
  - **PPP** : Point to Point Protocol RFC 1661
- Couche Réseaux :
  - **IP** : Internetworking Protocol RFC 791 (v4) et RFC 2460 (v6)
  - **ICMP** : Internet Control Message Protocol RFC 792
  - **ARP** : Adress Resolution Protocol RFC 826
  - **RARP** : Reverse ARP RFC 903
  - **IGMP** : Internet Group Management Protocol RFC 1112
- Protocoles Transport
  - **UDP** : User Datagram Protocol RFC 768
  - **TCP** : Transport Control Protocol RFC 793

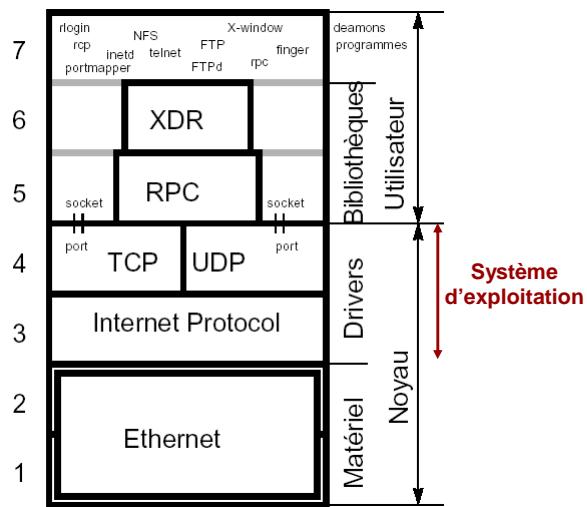
page 13

## Normes et RFC (suite)

- Couche Application :
  - **DNS** : Domain Name Server RFC 1034 UDP (53)
  - **HTTP** : Hyper Text Transfer Protocol RFC 2616 TCP (80)
  - **SMTP** : Simple Mail Transfer Protocol RFC 821 TCP (25)
  - **POP 3** : Post Office Protocol RFC 1939 TCP (110)
  - **MIME** : Multipurpose Internet Mail Extensions RFC 2045 -
  - **FTP** : File Transfer Protocol RFC 959 TCP (20-21)
  - **TELNET** RFC 854 TCP (23)
  - **BOOTP** : Bootstrap Protocol RFC 951 UDP (67-68)
  - **DHCP** : Dynamic Host Configuration Protocol RFC 2131 TCP (546-547)
  - **SNMP** : Simple Network Management Protocol RFC 1157 UDP (161-162)
  - **RIP 2** : Routing Internet Protocol RFC 2453 UDP (520)
  - **OSPF 2** : Open Shortest Path First RFC 2328 -
  - **BGP** : Border Gateway Protocol RFC 1771 TCP (179)
  - **IMAP** : Internet Message Access Protocol RFC 2060 TCP (143)
  - **RTSP** : Real Time Streaming Protocol RFC 2326 TCP (554)
  - **NFS** : Network File system RFC 1094 UDP (2049)

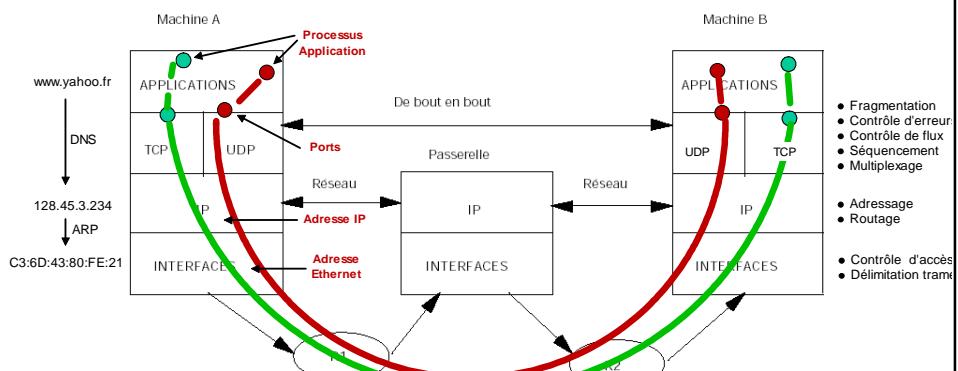
page 14

## Architecture d'un terminal IP



page 15

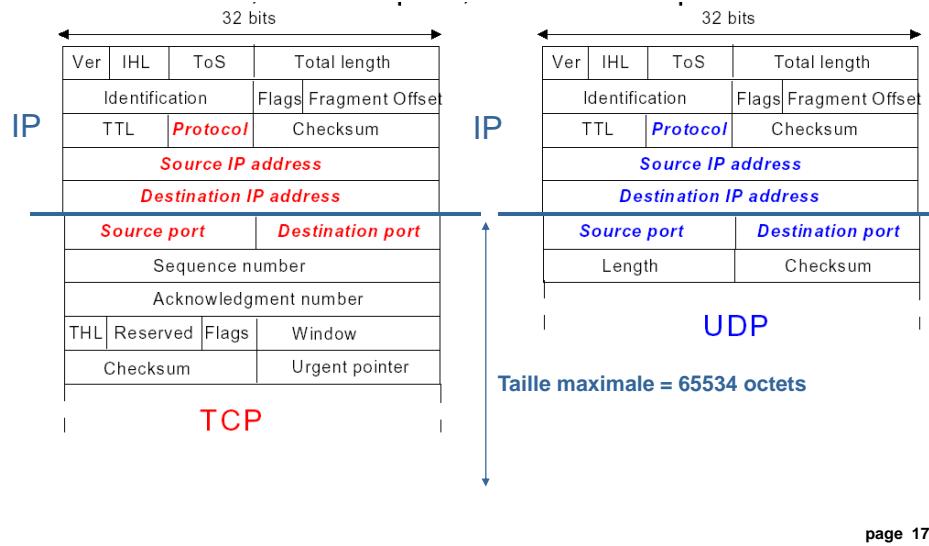
## Communication client/serveur



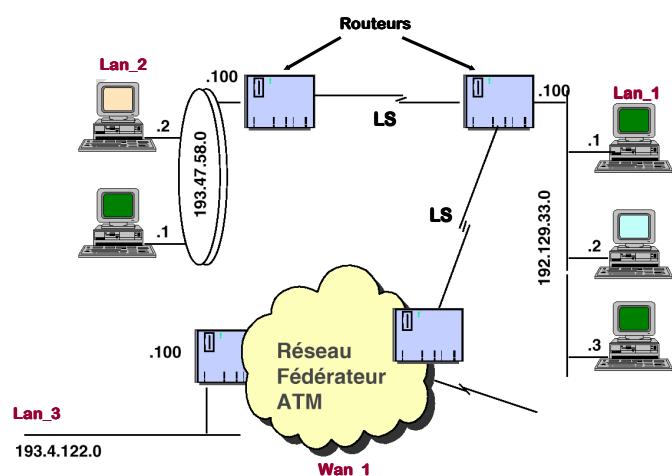
© Ahmed Mehaoua 1999 - page 16

page 16

## Structure des Paquets IP



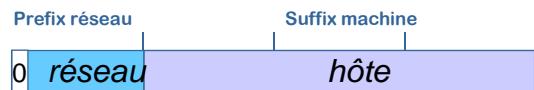
## Adresse réseau



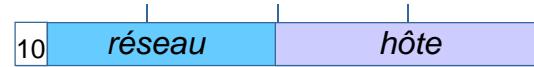
page 18

## Classes d'adresses IP

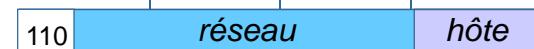
- **Classe A** de 1.x.x.x à 127.x.x.x
  - ↳ 127 réseaux –
  - ↳ 16777214 machines



- **Classe B** de 128.0.x.x à 191.255.x.x
  - ↳ 16384 réseaux –
  - ↳ 65534 machines



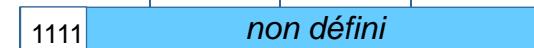
- **Classe C** de 192.0.0.x à 223.255.255.x
  - ↳ 2097152 réseaux –
  - ↳ 254 machines



- **Classe D** de 224.0.0.0 à 239.255.255.255  
(multicast)



- **Classe E** de 240.0.0.0 à 255.255.255.255 (Expérimentale)



page 19

## Adresses IP particulières

- **Adresse de diffusion** : tous les champs sont à « 1 »
  - Exemple : 255.255.255.255
  - Diffusion sur tout le réseau (tous les sous-réseaux sont concernés)
- **Adresse de diffusion dirigée** : le champ « hostid » est tout à « 1 » et le champ « netid » est une adresse réseau spécifique :
  - Exemple : 192.20.0.255
  - ⇒ la diffusion concerne toutes les machines situées sur le réseau spécifié : 192.20.0.255
  - ⇒ désigne toutes les machines du réseau de classe C 192.20.0
- **Adresse de boucle locale** :
  - l'adresse réseau 127.0.0.1 est réservée pour la désignation de la machine locale, c'est à dire la communication intra-machine. Une adresse réseau 127 ne doit, en conséquence, jamais être véhiculée sur un réseau et un routeur ne doit jamais router un datagramme pour le réseau 127.
  - **Adresse de BOOTP** (« hostid » et « netid » tout à zéro), l'adresse est utilisée au démarrage du système afin de connaître l'adresse IP (Cf RARP).
    - Exemple : 0.0.0.0

page 20

## Masque de réseau ou Netmask

- **Masque du réseau** : adresse IP particulière servant à identifier l'adresse du réseau à partir d'une adresse IP de machine.
  - Le masque d'un réseau de classe A = 255.0.0.0
  - Le masque d'un réseau de classe B = 255.255.0.0
  - Le masque d'un réseau de classe C = 255.255.255.0
  - Dans le cas d'un réseau découpé en sous-réseau : le masque est calculé en mettant tous les bits du préfix réseaux à la valeur binaire « 1 », et tous les bits associés au suffix à « 0 ».
- **Adresses réseau** : adresse IP dont la partie « hostid » ne comprend que des zéros;
  - => la valeur zéro ne peut être attribuée à une machine réelle : 192.20.0.0 désigne le réseau de classe C 192.20.0
- **Adresse machine locale** : adresse IP dont le champ réseau (netid) ne contient que des zéros;
  - Exemple 0.0.25.1

page 21

## Netmask

- Permet à une station de savoir si la station destination est dans le même réseau qu'elle ou s'il lui faut envoyer son paquet au routeur qui l'acheminera,
- Exemple station A veut envoyer un paquet à une station B :
  - @ IP A = 172.16.2.4
  - @ IP B = 172.16.3.5
  - @ netmask A : 255.255.0.0
- La station A doit réaliser **3 opérations** :
  1. @ A AND @ netmask A = Res 1
  2. @ B AND @ netmask A = Res 2
  3. comparer Res 1 et Re 2
    - Si Res 1 = Res2 alors station sur le même réseau
    - Sinon station sur des réseaux distants

page 22

A	B	A AND B
0	0	0
0	1	0
1	0	0
1	1	1

## Netmask (2)

172 . 16 . 2 . 4 (@ IP A)  
10101100 . 00010000 . 00000010 . 00000100  
11111111 . 11111111 . 00000000 . 00000000 (mask A = 255.255.0.0)  
10101100 . 00010000 . 00000000 . 00000000 (@ du réseau classe B 172.16.0.0)

172 . 16 . 3 . 5 (@ IP B)  
10101100 . 00010000 . 00000011 . 00000101  
11111111 . 11111111 . 00000000 . 00000000 (mask A = 255.255.0.0)  
10101100 . 00010000 . 00000000 . 00000000 (@ du réseau B 172.16.0.0)

page 23

## Netmask (3)

Autre exemple @ IP C = 125.128.96.12

172 . 16 . 2 . 4 (@ IP A)  
10101100 . 00010000 . 00000010 . 00000100  
11111111 . 11111111 . 00000000 . 00000000 (mask A = 255.255.0.0 - classe B)  
10101100 . 00010000 . 00000000 . 00000000 (@ du réseau classe B 172.16.0.0)

125 . 128 . 96 . 12 (@ IP C)  
01111111 . 10000000 . 01100000 . 00001100  
11111111 . 11111111 . 00000000 . 00000000 (mask A = 255.255.0.0)  
01111111 . 10000000 . 00000000 . 00000000 (@ du réseau classe A 125.128.0.0)

page 24

## Adresses IP Privées

- Classe A : 10.0.0.0 - 10.255.255.255
- Classe B : 172.16.0.0 - 172.31.255.255
- Classe C : 192.168.0.0 - 192.168.255.255

page 25

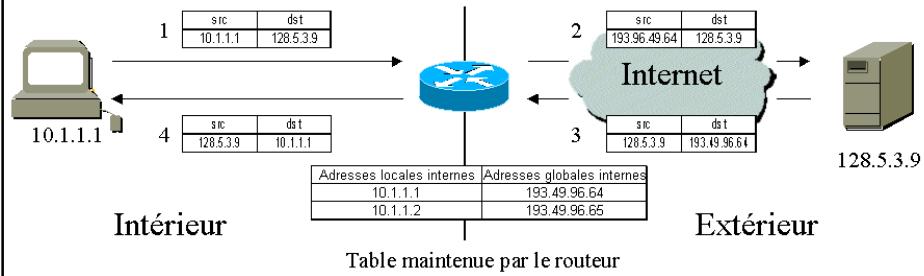
## Problème des adresses IPv4

- L'assignation d'une classe par bit, signifie : la classe A prend 1/2 des adresses, la classe B 1/4, la classe C 1/8 etc.
- **Problèmes** avec une telle assignation :
  1. Gaspillage
  2. Saturation dans les routeurs
  3. Pénurie des adresses encore libres
- **Solutions** ?
  1. Utiliser les adresses IP privées avec un protocole de translation d'adresse (NAT: Network Address Translation)
  2. Fractionner les blocs d'adresses plus finement : « Subnetting » ou « sous-adressage »
    - conserver la taille à 32 bits mais ...
  3. Augmenter la taille du champ adresse :
    - Exemple : IP version 6 (décembre 1998) : champ adresse de 128 bits
    - conséquence : incompatibilité entre les machines

page 26

## « NAT »

- Network Address Translation :



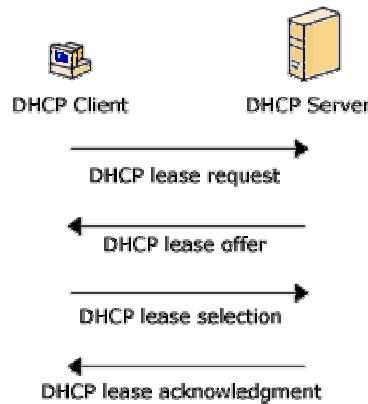
page 27

## Attributions des adresses IP

- Pour communiquer dans un réseau IP, les hosts doivent connaître :
  - L'adresse IP du host
  - Le masque de leur réseau
  - L'adresse IP de la passerelle (Gateway) (optionnel)
  - L'adresse IP du serveur de noms (DNS) (optionnel)
- Configuration statique :
  - Configuration manuelle et permanente.
  - requise pour les serveurs et routeurs
  - Commandes systèmes **ifconfig** (unix) - **netsh** (windows)
- Configuration dynamique :
  - Simplicité et optimisation des adresses IP
  - Adaptée pour les terminaux nomades
  - Utilisation d'un serveur de configuration interrogé par les terminaux au démarrage
  - Les clients et le serveur communiquent au moyen d'un protocole (règles d'échange et format de messages valides) **DHCP** (Dynamic Host Configuration Protocol)

page 28

## Dynamic Host Configuration Protocol (DHCP)



page 29

## Subnetting

- **Constat** : Un site ne contient pas un réseau mais un ensemble de réseaux (exemple : UVSQ)
- **Solution** : scinder une classe en sous-réseaux (ou segment):
  - La partie numéro de machine devient le numéro de sous-réseau et le numéro de la machine dans ce sous-réseau,
  - Combien de bits ( $n$ ) utiliser pour représenter les sous-réseaux ?
    - Si ( $p$ ) sous-réseaux à représenter alors  $p \geq (2^n)$
  - Nombre de bits alloués au numéro de sous-réseau est configurable : c'est le « **sub-netmask** » ou simplement le « **netmask** » du sous-réseau

(1)	Partie internet	Partie locale	
(2)	Partie internet	Sous-réseau	Machine

page 30

## Subnetting Exemple

- Soit un réseau d'entreprise de classe B = 130.96.0.0 constitué de 8 sous-réseaux locaux.
- Pour identifier 8 sous-réseaux, combien de bits faut-il prendre de la partie Host-id ?
  - 3 bits ? =>  $2^3 - 2 = 6$  (insuffisant !!!)
  - 4 bits ? =>  $2^4 - 2 = 14$  (Oui !!!)
- **Masque de sous-réseau** = 255.255.240.0
- Exemple **d'adresse de diffusion restreinte** = 130.96.175.255 pour le **sous-réseau** de net-id = 130.96.160.0

page 31

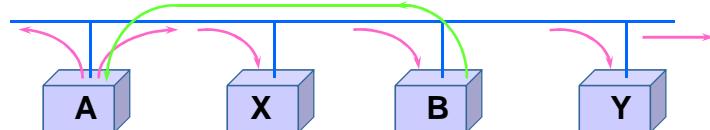
## Résolution des adresses

- Pourquoi ?
  - Dans un **Intranet ou Internet**, les **communications** entre applications se font au moyen des adresses IP des hosts (et des n° de port).
  - Dans un **réseau local**, l'**acheminement** des données se fait au moyen des adresses physiques des émetteurs et des récepteurs.
  - L'unité de transfert est la Trame Ethernet (et non le paquet IP)
  - Les adresses IP sont obtenues par l'interrogation d'un serveur : le **DNS**
  - Comment obtenir l'adresse physique d'une machine distante en connaissant son adresse IP ?
- La Solution :
  - ARP : Address Resolution Protocol
  - utiliser un protocole de type requête/réponse
  - utilise le principe de la diffusion sur le réseau local (broadcast)
  - l'association **adresse physique - adresse IP** de l'émetteur est incluse dans la requête ARP de manière à ce que les récepteurs enregistrent l'association dans leur propre mémoire cache,

page 32

## ARP

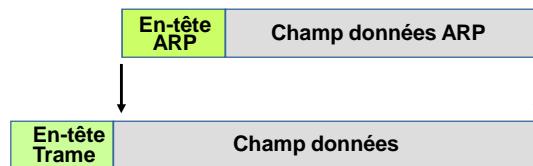
- L'association **adresse physique - adresse IP** de l'émetteur est incluse dans la requête ARP de manière à ce que les récepteurs enregistrent l'association dans leur propre mémoire cache,



- Pour connaître l'adresse physique de B (PB) à partir de son adresse IP (IB), la machine A **diffuse une requête ARP** qui contient l'adresse IP de B (IB) vers toutes les machines;
- la machine B **répond avec un message ARP** qui contient la paire (IB, PB).
- Rem : champ type de la trame Ethernet: 0806 pour ARP

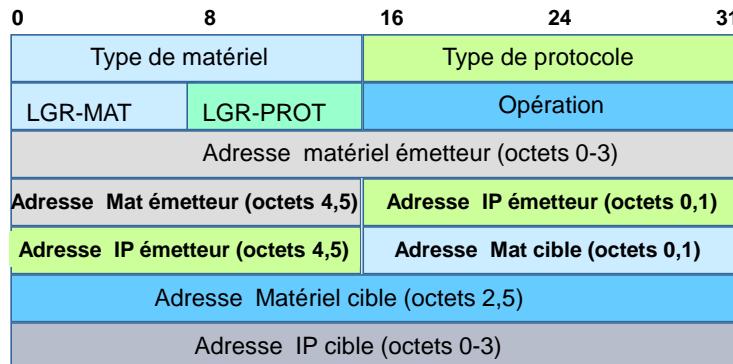
page 33

## ARP : encapsulation



page 34

## Format du message ARP



page 35

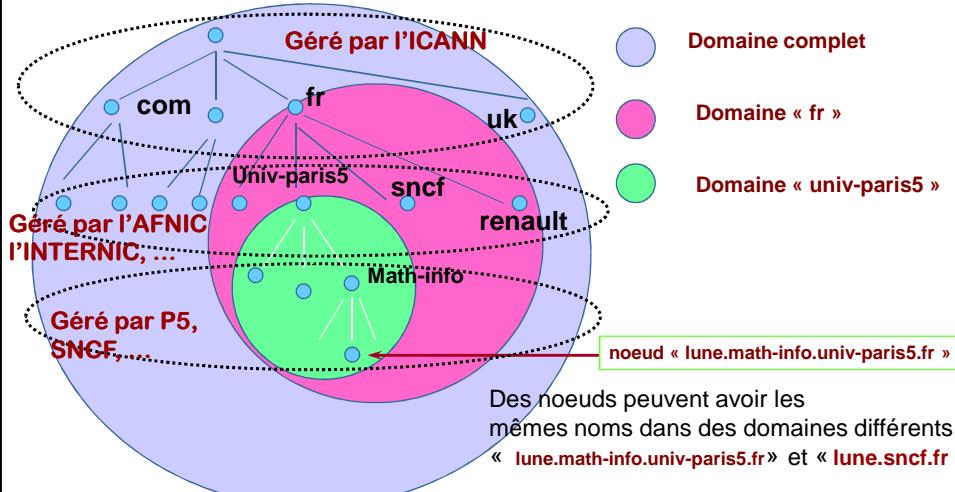
## Nommage des ressources

- **Nommage des ressources** du réseau : utiliser un **NOM SYMBOLIQUE** plutôt qu'une adresse décimale :
  - brune.prism.uvsq.fr                  193.51.25.130
  - www.yahoo.fr                          10.25.123.68
  - Unicité des adresses => unicité des noms
  - Il existe un « plan de nommage » hiérarchique mondiale et un « service de noms » mondial : le **DNS** (Domain Name System)

page 36

## Le domaine

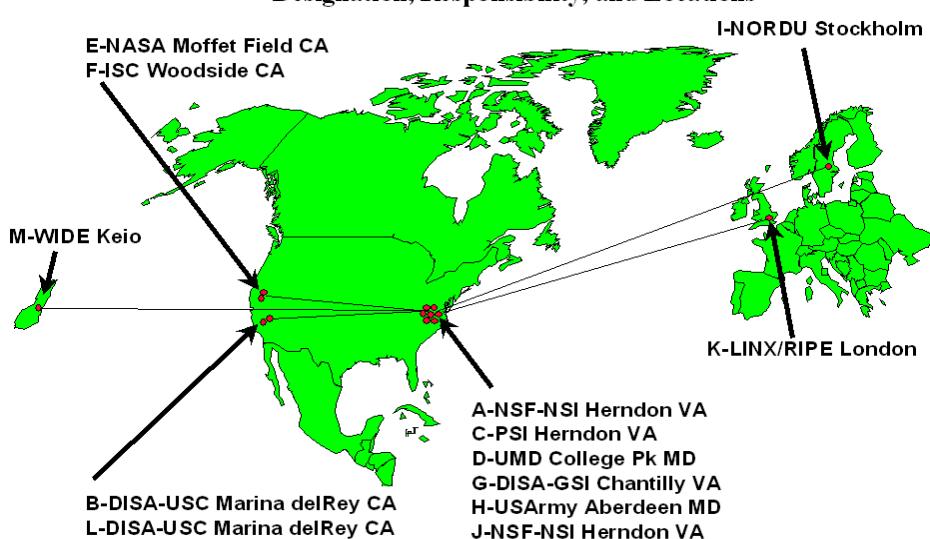
Un domaine est un sous-arbre de l'espace nom de domaine



page 37

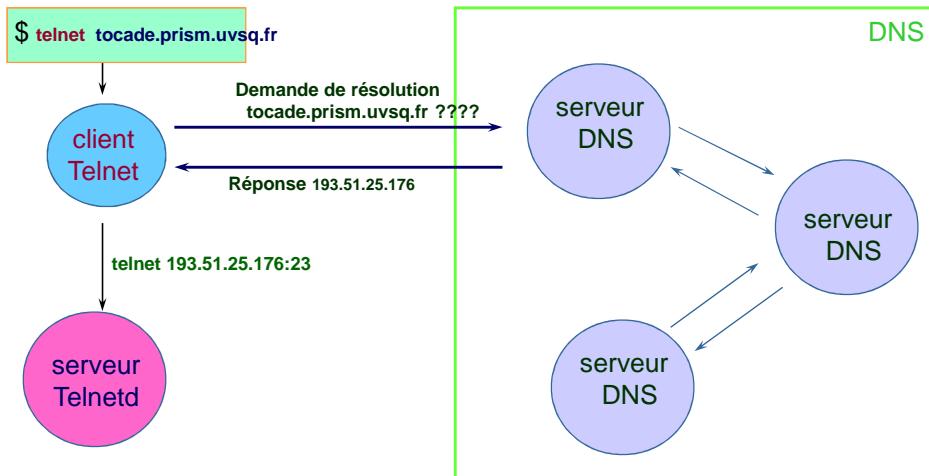
## DNS Root Servers

Designation, Responsibility, and Locations



page 38

## Principe (illustration)



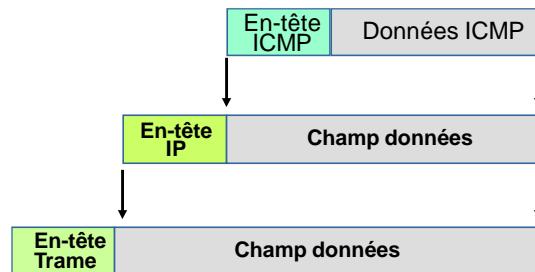
page 39

## Protocole ICMP

- Le protocole **ICMP** (Internet Control Message Protocol) permet d'envoyer des **messages de commande** ou des **messages d'erreurs** vers d'autres machines ou routeurs.
- ICMP rapporte les messages d'erreur à l'émetteur initial.
- Beaucoup d'erreurs sont causées par l'émetteur, mais d'autres sont dues à des problèmes d'interconnexions rencontrés sur l'Internet :
  - machine destination déconnectée**,
  - durée de vie du datagramme expirée**,
  - congestion de routeurs intermédiaires**.
- Si un routeur détecte un problème sur un datagramme IP, elle le détruit et émet un message ICMP pour informer l'émetteur initial.
- Les messages ICMP sont véhiculés à l'intérieur de datagrammes IP et sont routés comme n'importe quel datagramme IP sur l'internet.
- Une erreur engendrée par un message ICMP ne peut donner naissance à un autre message ICMP (évite l'effet cumulatif).

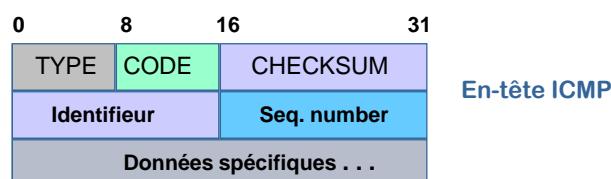
page 40

## ICMP : encapsulation



page 41

## ICMP : format des messages



TYPE            8 bits; type de message

CODE            8 bits; informations complémentaires

CHECKSUM      16 bits; champ de contrôle

IDENTIFIER (16 bits) et SEQUENCE NUMBER (16 bits) sont utilisés par l'émetteur pour contrôler les réponses aux requêtes, (CODE = 0).

page 42

## ICMP : type de messages

<u>TYPE</u>	<u>Message ICMP</u>	<u>TYPE</u>	<u>Message ICMP</u>
0	Echo Reply	13	Timestamp Request
3	Destination Unreachable	14	Timestamp Reply
4	Source Quench	15	Information Request (obsolete)
5	Redirect (change a route)	16	Information Reply (obsolete)
8	Echo Request	17	Address Mask Request
11	Time Exceeded (TTL)	18	Address Mask Reply
12	Parameter Problem with a Datagram		

page 43

## ICMP : les messages d'erreur

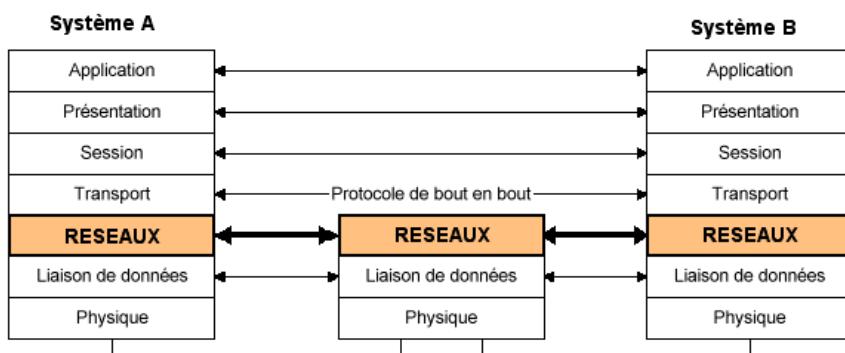
- Lorsqu'une passerelle émet un message ICMP de type destination inaccessible, le champ **CODE** décrit la nature de l'erreur :
  - 0 Network Unreachable
  - 1 Host Unreachable
  - 2 Protocol Unreachable
  - 3 Port Unreachable
  - 4 Fragmentation Needed and DF set
  - 5 Source Route Failed
  - 6 Destination Network Unknown
  - 7 Destination Host Unknown
  - 8 Source Host Isolated
  - 9 Communication with destination network administratively prohibited
  - 10 Communication with destination host administratively prohibited
  - 11 Network Unreachable for type of Service
  - 12 Host Unreachable for type of Service

page 44

# Internet et Routage

page 1

## Couche réseau



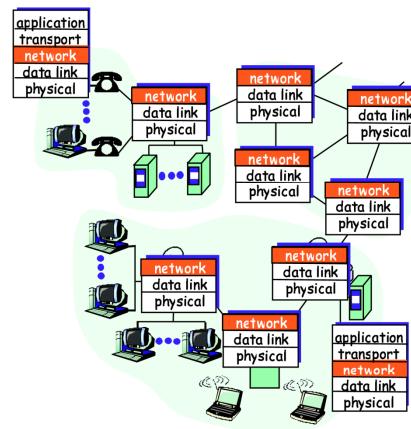
page 2

## Couche réseau: fonctionnalités

- Transporter des paquets de l'émetteur vers le récepteur
- Les protocoles de couche réseau s'exécutent dans chaque hôte et routeur.

Trois fonctions principales :

- *Choix du chemin*: route suivie par les paquets de la source à la destination : *Algorithmes de routage*
- *Commutation*: transporter les paquets du port d'entrée vers le bon port de sortie.
- *Mise en place de l'appel*: Dans les réseaux à circuits, la mise en place du circuit est effectuée par la couche réseau.



page 3

## Couche réseau: fonctionnalités

2 modes de fonctionnement de la couche réseau :

- Mode circuit virtuel
- Mode datagramme

page 4

## Mode Circuit Virtuel

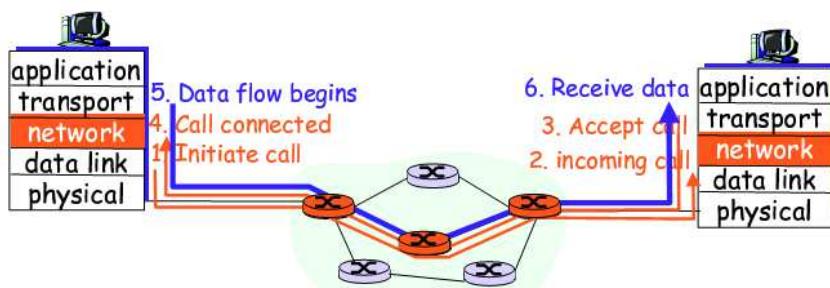
Le « chemin » de la source à la destination se comporte comme un circuit téléphonique

- Avant d'émettre des données, le circuit doit être mis en place
- Chaque paquet contient un identificateur de circuit (et non pas l'adresse de la destination)
- Chaque routeur maintient un « état » pour chaque connexion qui traverse le routeur
  - > Les connexions dans la couche transport ne mettent en jeux que les systèmes terminaux
- Des ressources du lien (bande passante) ou du routeur (mémoire) peuvent être allouées au circuit virtuel
  - > Pour garantir des performances

page 5

## Mode Circuit Virtuel

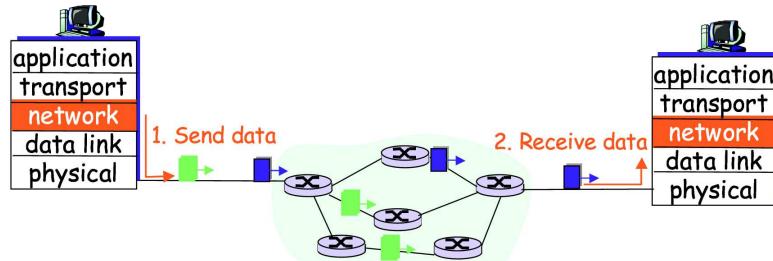
- Utilisés pour mettre en place et gérer un circuit virtuel
- utilisés dans ATM, frame-relay et X.25
- Ne sont pas utilisés (du moins de façon visible) dans l'Internet actuel



page 6

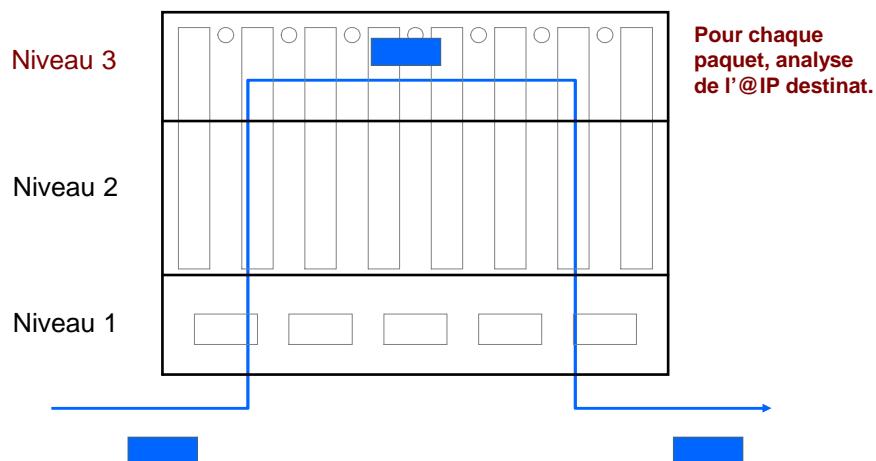
## Mode Datagramme

- Pas de mise en place de circuit
- routeurs: aucun état mémorisé au sujet des connexions
  - > Pas de notion de connexion au niveau réseau
- Les paquets sont typiquement routé en fonction de l'adresse de destination
  - > Des paquets avec la même source et destination peuvent suivre des trajets différents



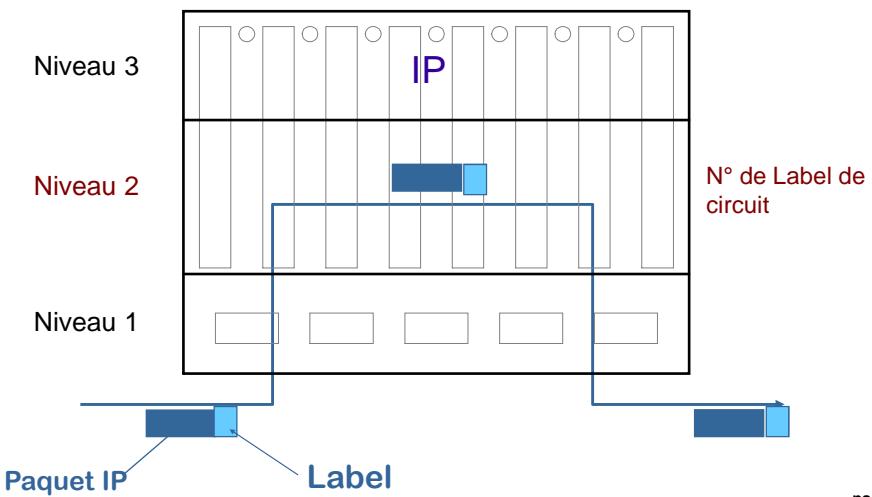
page 7

## Mode datagramme Routage IP

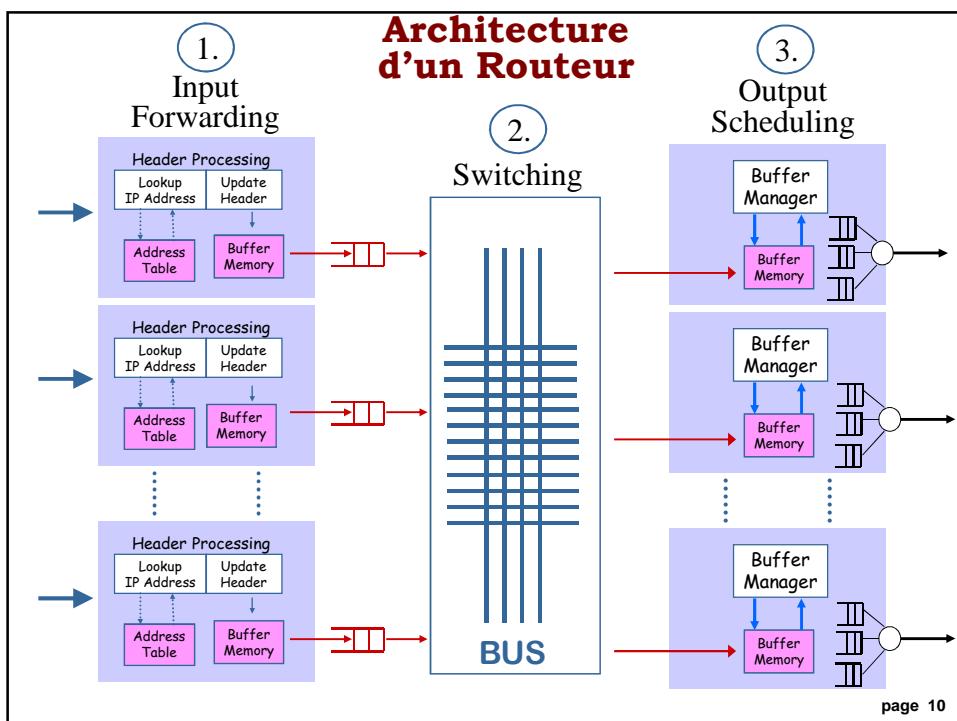


page 8

## Mode circuit virtuel Commutation IP

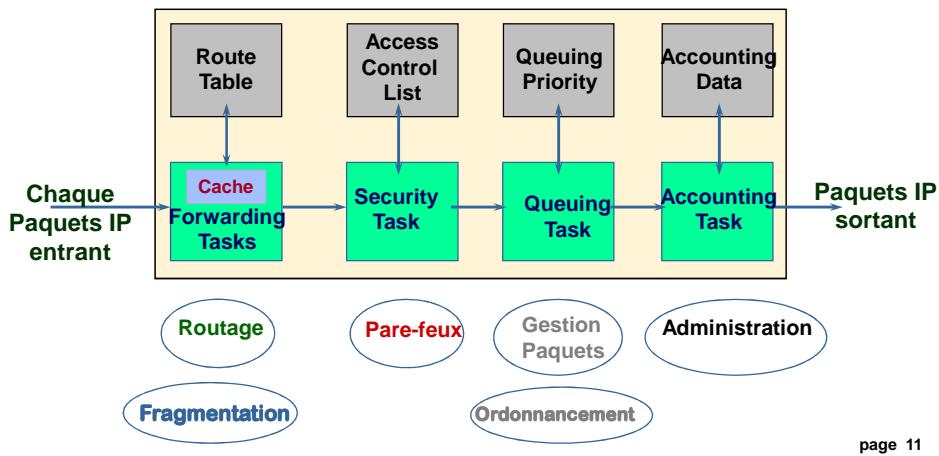


page 9



page 10

## Fonctions d'un Routeur IP



## Tâches d'une passerelle IP

Pour chaque datagramme IP qui traverse une passerelle, le protocole IP :

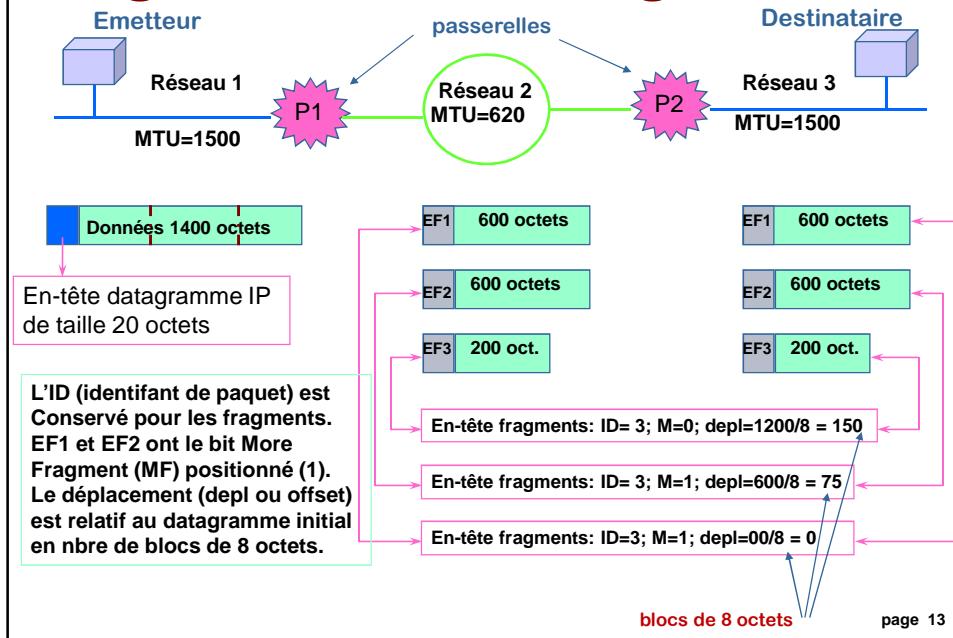
1. détermine si ce sont des données utilisateur (TCP ou UDP) ou de contrôle (ICMP) destinées à la passerelle (analyse du champ « Protocole »)
2. vérifie le checksum, si faux => destruction paquet
3. vérifie la liste de contrôle d'accès (optionnel : fonction de Pare-Feux)
4. décrémente la durée de vie (TTL) du paquet, si nulle => destruction
5. **forwarding: décide du routage** (consulte la table de routage)
6. **fragmente** le datagramme si nécessaire (pour respecter le MTU de la prochaine liaison)
7. **reconstruct l'en-tête IP** avec les champs maj (TTL, ID, FLAG, OFFSET, Checksum)
8. **Switching: transmet** le ou les fragments du paquet IP vers le port de sortie à travers le bus
9. **Scheduling: ordonnancement** du paquet dans la file de sortie
10. Remise du paquet à la couche 2 puis à la couche 1 pour codage et transmission
11. mise à jours des statistiques de trafic (optionnel)

A réception dans l'hôte destinataire, IP :

- vérifie le checksum
- s'il y a eu fragmentation, mémorise puis **réassemble**
- **délivre au niveau supérieur** (TCP, UDP) les données et les paramètres par la primitive DELIVER

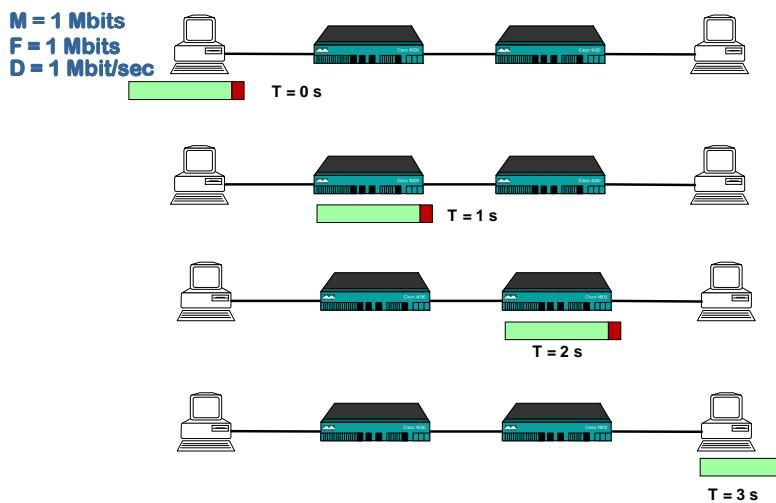
page 12

## Fragmentation des datagrammes IP



## Impact de la taille des paquets

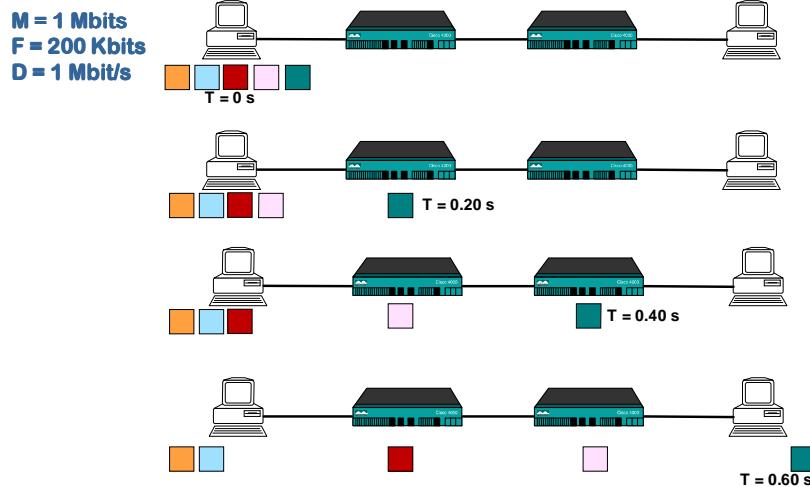
Commutation de paquets IP



page 14

## Impact de la taille des paquets

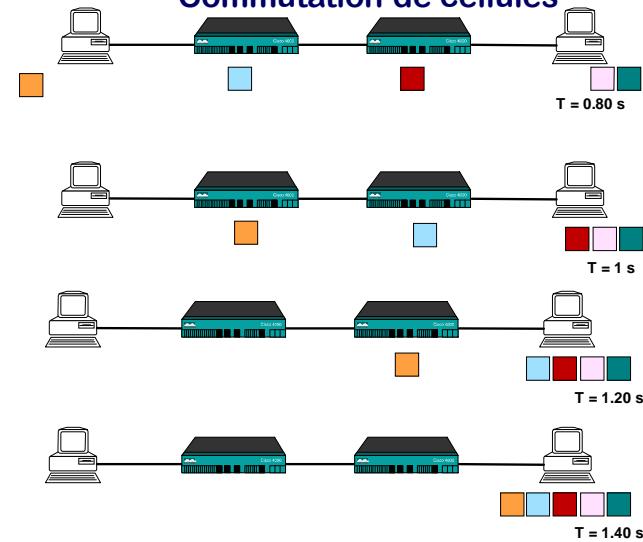
Commutation de cellules



page 15

## Impact de la taille des paquets

Commutation de cellules

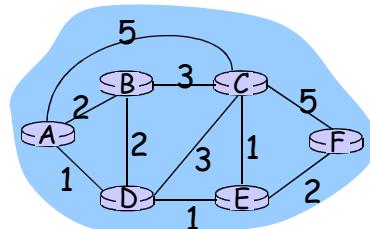


page 16

## Routage: définition

**Objectif:** détermine "meilleur" chemin (sequence des routeurs) à travers le réseau de la source à la destination.

- Utilisation du Graphe par les algorithmes de routage:
- noeuds de graphe sont des routeurs
- Arcs de graphe sont des liens physique
  - poids: délai, nombre de sauts, débits, ...



- "meilleur" chemin:
  - Typiquement un chemin à coût minimum
  - autre déf possible

17

## classification des algo. de routage

### Global ou decentralisé?

#### Global:

- Une connaissance topologique
- **Algorithme "link state"**
- Ex. OSPF (grands réseaux)

#### Decentralisé:

- Processus itératif de calculs, échange des infos avec les voisins
- **Algorithme "distance vector"**
- Ex. RIP (petits réseaux)

### Statique ou dynamique?

#### Statique:

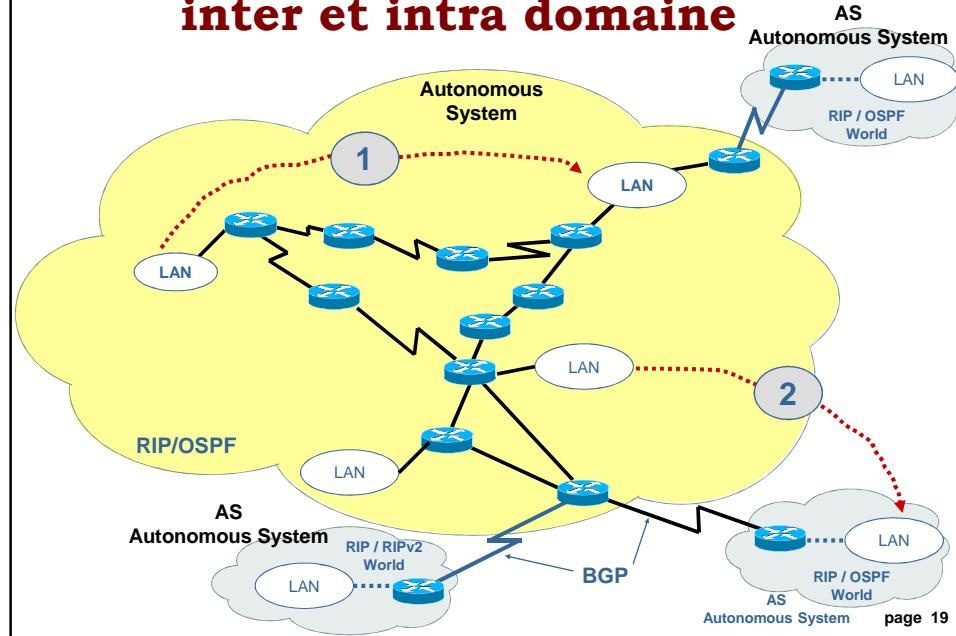
- Les routes changent lentement avec le temps
  - Ex. Télécoms

#### Dynamique:

- Les routes changent rapidement
  - Mise à jour périodique
  - Ex. Internet

18

## Route IP inter et intra domaine



## Qu'est ce qu'un Système Autonome

- Un réseau administré par une unique autorité
- Un réseau soumis à une **même** politique de routage
- Peut être constitué de plusieurs sous réseaux
- Identifié par un numéro de AS qui sont :
  - Attribué par l'ICANN
  - compris entre 1 et 65535 (n° privés entre 64512 et 65535)

page 20

## Routage IP intra-domaine

Distance vector algorithm : ([utilisé avec le protocole RIP](#))

- algorithme simple,
- par diffusion d'un extrait des meilleurs chemins,
- (sous la forme d'un vecteur où chaque entrée contient une distance)
- entre voisins directs (de proche en proche)
- métrique simple : *hop count*.

Link state algorithm (pour information) : ([utilisé avec le protocole OSPF](#))

- 2 phases :
  - . diffusion à tous de la connaissance sur les liaisons locales
  - . calcul local par chacun des meilleurs chemins sur les informations ainsi rassemblées
- exemple : Short Path First

page 21

## Routage IP

- Fonction qui permet de déterminer le meilleure chemin dans un réseau maillé vers une destination identifiée par une adresse IP.
- Utilisation de :
  - **TABLE DE ROUTAGE** (ou table d'acheminement) située dans chaque nœud : information nécessaire pour atteindre le prochain nœud vers la destination. Ex. Table de routage ip (netstat -r)
  - **ALGORITHME DE ROUTAGE** : fonction distribuée sur chaque noeuds qui a pour objectif de calculer les routes optimales pour atteindre une destination. Ex. Bellman-ford, Djikstra,
  - **PROTOCOLES DE ROUTAGE** : pour rôle l'échanges des informations de routes calculées par les **algorithmes de routage** et qui permettent la mise à jour dynamique des **tables de routage**. Ex. RIP, OSPF

page 22

## Exple d'une table de routage

IPv4 Table de routage					
Destination réseau	Masque réseau	Adr. passerelle	Adr. interface	Métrique	
<b>Itinéraires actifs :</b>					
0.0.0.0	0.0.0.0	172.30.32.1	172.30.34.194	25	
0.0.0.0	0.0.0.0	193.48.200.198	193.48.200.198	20	
127.0.0.0	255.0.0.0	On-link	127.0.0.1	306	
127.0.0.1	255.255.255.255	On-link	127.0.0.1	306	
127.255.255.255	255.255.255.255	On-link	127.0.0.1	306	
172.30.32.0	255.255.240.0	On-link	172.30.34.194	281	
172.30.34.194	255.255.255.255	On-link	172.30.34.194	281	
172.30.47.255	255.255.255.255	On-link	172.30.34.194	281	
172.168.56.0	255.255.255.255	On-link	192.168.56.1	276	
192.168.56.1	255.255.255.255	On-link	192.168.56.1	276	
193.48.200.0	255.255.255.0	On-link	193.48.200.198	276	
193.48.200.198	255.255.255.255	On-link	193.48.200.198	276	
193.48.200.255	255.255.255.255	On-link	193.48.200.198	276	
224.0.0.0	240.0.0.0	On-link	127.0.0.1	306	
224.0.0.1	240.0.0.0	On-link	192.168.56.1	276	
224.0.0.2	240.0.0.0	On-link	193.48.200.198	276	
224.0.0.0	240.0.0.0	On-link	172.30.34.194	281	
255.255.255.255	255.255.255.255	On-link	127.0.0.1	306	
255.255.255.255	255.255.255.255	On-link	192.168.56.1	276	
255.255.255.255	255.255.255.255	On-link	193.48.200.198	276	
255.255.255.255	255.255.255.255	On-link	172.30.34.194	281	

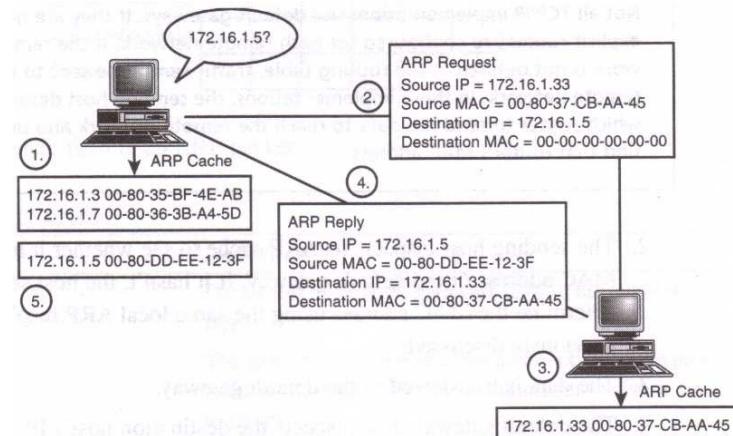
page 23

## Routage IP

- Machines et routeurs participent au routage :
  - Ils possèdent tous deux une table de routage,
  - les machines doivent déterminer si le datagramme doit être délivré sur le réseau physique sur lequel elles sont connectées (**routage direct**) ou bien si le datagramme doit être acheminé vers un routeur; dans ce cas (**routage indirect**), elle doit identifier le routeur appropriée.
  - les routeurs effectuent le choix de routage vers d'autres routeurs afin d'acheminer le datagramme vers sa destination finale.
  - Commande : netstat -r

page 24

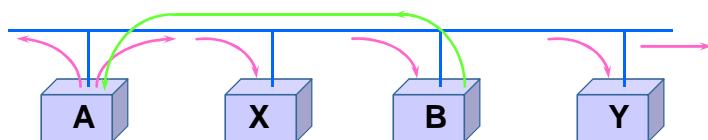
## Cas 1 : Serveur local Routage IP directe ARP (Adresse Resolution Protocol)



page 25

## ARP

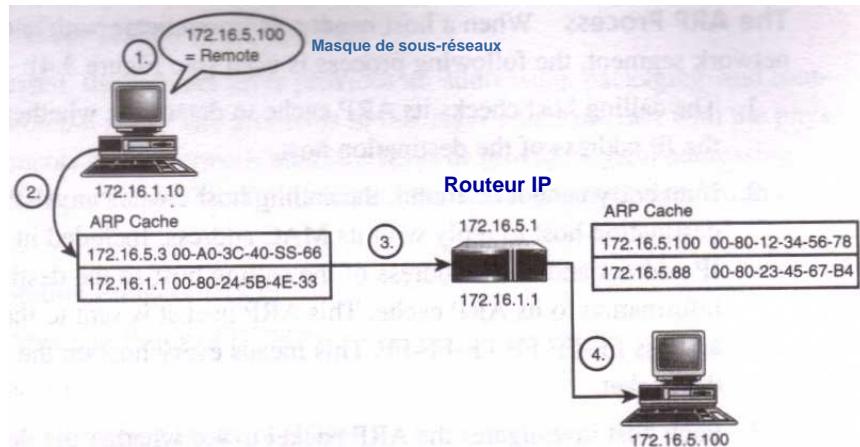
- L'association **adresse physique - adresse IP** de l'émetteur est incluse dans la requête ARP de manière à ce que les récepteurs enregistrent l'association dans leur propre mémoire cache,



- Pour connaître l'adresse physique de B (PB) à partir de son adresse IP (IB), la machine A **diffuse une requête ARP** qui contient l'adresse IP de B (IB) vers toutes les machines;
- la machine B **répond avec un message ARP** qui contient la paire (IB, PB).
- Rem : champ type de la trame Ethernet: 0806 pour ARP

page 26

## Cas 2 : Serveur distant Routage IP indirecte



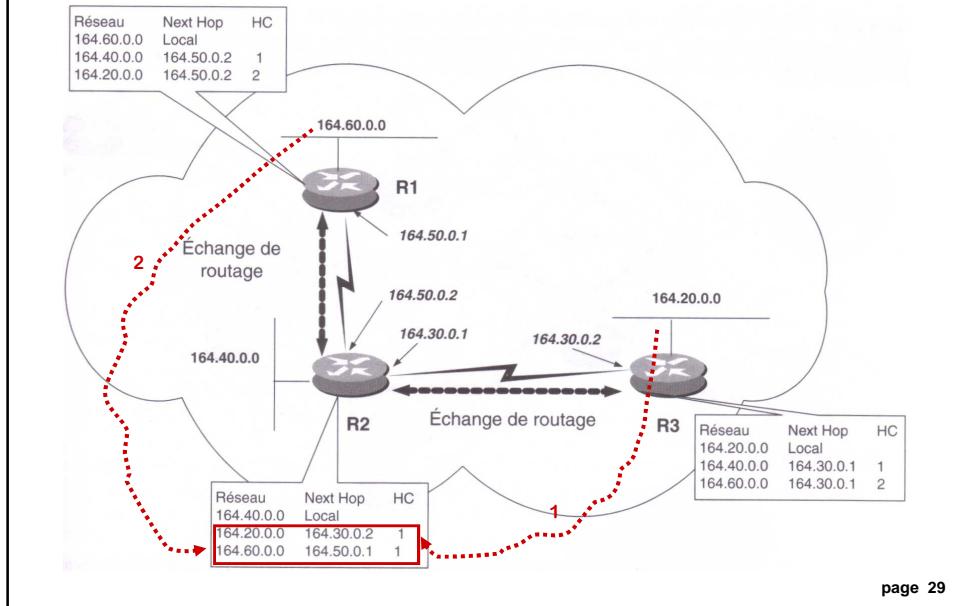
page 27

## Algorithme Distance Vector

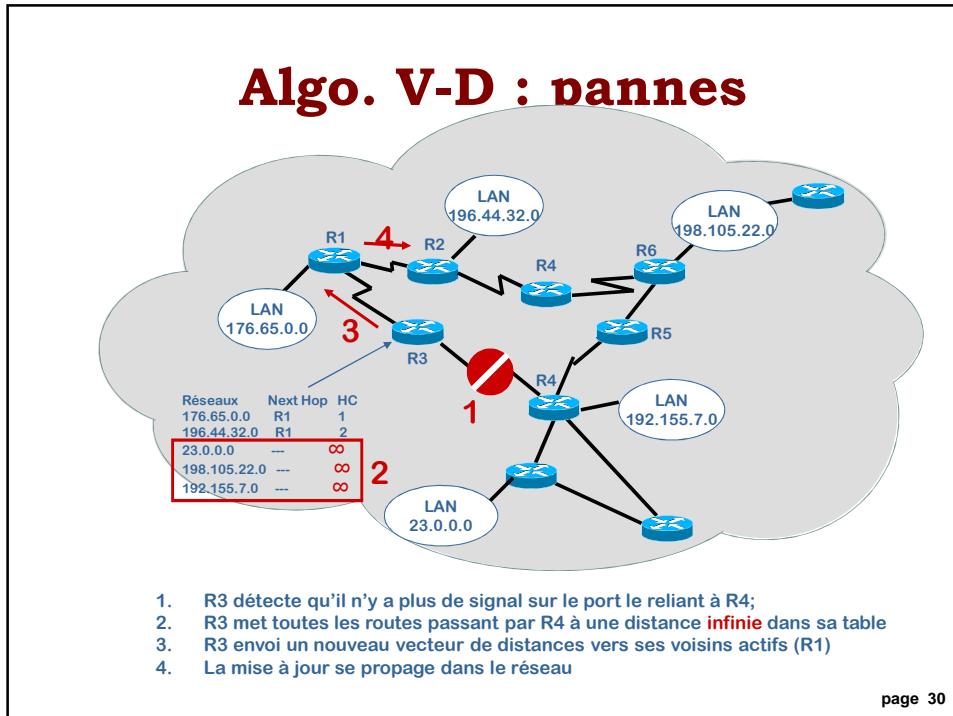
- Basé sur l'algorithme de Belman-Ford, calcul de routes distribué.
- Un routeur diffuse régulièrement à ses voisins les routes qu'il connaît (toute les 30sec. Avec RIP).
- Une route est composée d'une adresse destination, d'une adresse de passerelle et d'une métrique indiquant le nombre de sauts nécessaires pour atteindre la destination.
- Une passerelle qui reçoit ces informations compare les routes reçues avec ses propres routes connues et met à jour sa propre table de routage :
  - si une route reçue comprend un plus court chemin (nombre de prochains sauts +1 inférieur),
  - si une route reçue est inconnue.

page 28

## Algorithme Vector distance



## Algo. V-D : pannes



## Algorithme Distance Vector

### Inconvénients :

- La taille des informations de routage est proportionnelle au nombre de routeurs du domaine,
- Métrique difficilement utilisable : lenteur de convergence,
- Bouclage, éventuellement à l'infini,
- Pas de chemins multiples
- Coût des routes externes arbitraire.

page 31

## RIP

Routing Information Protocol :

- RIP-1 : RFC 1058 - juin 1988.
- RIP-2 : RFC 1388 - juin 1993.

*routed* : Unix RIP routing deamon

commande *netstat -r* : visualise la table de routage

commande *route* : modifie la table de routage

fichier : */etc/hosts* : la table de routage initiale

**RIP + UDP + IP**

- . Port n°520 (service RIP)
- . Infini = 16 hops ↘ étendue limitée
- . Période de diffusion des message de routage [15-45s]
- . Durée de validité d'un entrée (3 mn)
- . Délai aléatoire de diffusion immédiate [0-5s]

**Optimisation :**

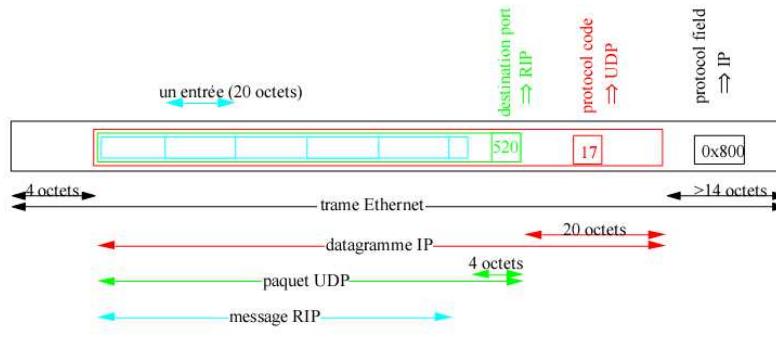
- RIP-1 utilise l'adresse de diffusion locale (255.255.255.255)
  - . Toutes les stations reçoivent une copie du message
- RIP-2 utilise l'adresse multicast réservée (224.0.0.9 : le groupe des routeurs)
  - . Seuls les routeurs RIP reçoivent une copie du message
    - ⇒ moins de surcharge pour les drivers IP des autres stations et autres routeurs.

page 32

## RIP Encapsulation

### Contraintes

- . Les messages de routage ont une longueur limitée : 512 octets
  - ➡ le MTU par défaut des datagrammes IP est de 576 octets !
- . si les informations à transmettre sont plus longues, on diffuse plusieurs messages de routage.
- . le protocole RIP est sans mémoire (“memoryless”), ces messages ne sont pas liés (par ex. pas de n°).



## RIP principe

### Etat initial :

- Chaque routeur connaît son environnement immédiat :
- . son adresse, ses interfaces,
  - . ses (sous-)réseaux directs : distance = 0.

Chaque routeur maintient localement une liste (BdD) des meilleures routes

➡ table de routage <@ de destination, distance, @ du prochain routeur>

Chaque routeur actif diffuse un **extrait** de sa table de routage (message de routage) :

- Périodiquement (30s)
- A tous leurs voisins immédiats
- Une liste de couple <@ de destination, distance>

Tous les routeurs mettent à jour leur tables de routage en conséquence. L'adresse du prochain routeur est implicitement celui de l'émetteur du message de routage.

### Etat des stations :

- Actif (les routeurs) diffusent leurs routes,
- Passif (les stations d'extrémité) écoutent.

## RIP Format des messages

0	7 8	15 16	31 bits
command	version	routing domain	
address family	route tag		
IP address			
subnet mask			
next-hop address			
metric			
address family	route tag		
IP address			
subnet mask			
next-hop address			
metric			

Le champ “**command**”(8 bits) : code le type du message :

- . 1 = demande d'information
  - demande partielle pour certaines destinations (dont les entrées figurent dans la demande)
  - demande totale (s'il y a une seule entrée associée à la demande tel que “address family”=0 et “metric”=16)
- . 2 = réponse
  - l'extrait des meilleures routes du routeur
  - suit à une demande, envoi périodique, envoi spontané

Le champ “**version**”(8 bits) :

- . 1 = RIP-1 ( $\Rightarrow$  les champs “routing domain”, “route tag”, “subnet address”, “next-hop address” sont inutilisés = 0)
- . 2 = RIP-2

Le champ “**routing domain**”(16 bits) :

- . RIP est générique :
  - plusieurs domaines peuvent être gérés simultanément par le même routeur.
- . 0 par défaut et obligatoire pour RIP-1

page 35

## RIP Format des messages (2)

0	7 8	15 16	31 bits
command	version	routing domain	
address family	route tag		
IP address			
subnet mask			
next-hop address			
metric			
address family	route tag		
IP address			
subnet mask			
next-hop address			
metric			

Le champ “**address family**”(16 bits) : code le format d'adressage :

- . les adresses peuvent être de longueur quelconque
- . 2 = IP ( $\Rightarrow$  32 bits)

Le champ “**route tag**”(16 bits) :

- . transmet des informations utilisées par le routage inter-domaine (EGP)
- . 0 pour RIP-1

Le champ “**IP address**”(32 bits) : l'adresse de destination

- . l'adresse d'un réseau IP ( $\Rightarrow$  netid)
- . l'adresse d'un sous-réseau IP ( $\Rightarrow$  subnet mask : subnetid)
- . l'adresse d'une station ( $\Rightarrow$  @IP)
- . l'adresse par défaut ( $\Rightarrow$  n'importe quelle destination : 0.0.0.0)

Le champ “**subnet mask**”(32 bits) :

- . 0 pour RIP-1
- . spécifie la taille du champ “subnetID” dans le champ “hostID” de l'adresse IP.

page 36

# Interconnexion de réseaux

## Interconnexion Qu'est ce que c'est ?

Fonction pour réaliser l'inter-fonctionnement de réseaux hétérogènes

- Hétérogénéité des réseaux à plusieurs niveaux:
  - Matériels
  - Capacité de traitement / stockage
  - Taille de paquets
  - Protocoles
  - Services
- Méthode : Identifier le niveau d'hétérogénéité afin de déterminer les fonctions requises pour établir l'interconnexion (Modèle OSI)
- Selon le niveau d'hétérogénéité considéré :
  - Mise en œuvre d'un dispositif d'interfonctionnement

---

© Ahmed Mehaoua - 2

## Niveaux d'interconnexion

L'interconnexion peut être effectuées à toutes les niveaux :

- couche 1 (Physique) : **modem, répéteur, concentrateur**
  - . techniques de modulation adaptées au support physique
  - . par ex.: interconnexion entre brins (segments) d'un seul Ethernet
- couche 2 (Liaison de données) : **pont, switch**
  - . conversion entre différentes méthodes d'accès
  - . par ex.: interconnexion de réseaux locaux
- couche 3 (Réseau) : **routeur , pare-feux**
  - . prévue pour !
- couches supérieures : **passerelle**, (relai, convertisseur de protocoles)
  - . interopérabilité de niveau applicatif
  - . par ex.: messagerie SMTP<-> X400



© Ahmed Mehaoua - 3

## Matériel

Répéteur/adaptateur (UNICOM)



Hub multi Protocole (3com)



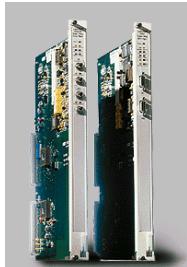
← Mini hubs 16/8 ports (HP)

© Ahmed Mehaoua - 4

## Matériel



Chassis Hub BayNetworks



MAU T.R  
BayNetworks



MAU Olicom



Hub BayNetworks



CAU Olicom

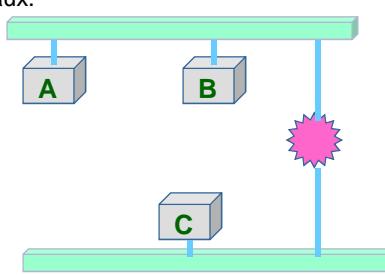


Hubs empilables

© Ahmed Mehaoua - 5

## Répéteurs

- Le signal électrique se déforme et s'atténue d'autant plus que la distance est longue entre deux noeuds.
- Passé une certaine limite (qui dépend du support), il faut le régénérer : amplification, resynchronisation.
- On utilise pour cela un **REPETEUR** :
  - Fonctionne au **niveau Physique** (bit),
  - dispositif actif non configurable
  - permet d'augmenter la distance entre deux stations Ethernet
  - reçoit, amplifie et retransmet les signaux.
- **Limitations :**
  1. ne peuvent être utilisés que sur les mêmes types de segments (Ethernet-Ethernet ou Token Ring-Token Ring).
  2. Pas de conversion de signaux (Optique – > électrique):



© Ahmed Mehaoua - 6

## Concentrateurs / Hubs

- Un **concentrateur** (ou **Hub**, étoile, multi-répéteur) est employé dans les réseaux locaux ETHERNET :
  - a une fonction de répéteur, mais :
  - permet de mixer différents médias (paire torsadée, AUI, Thin ethernet, fibre optique),
  - D'employer une topologie physique en étoile (Ethernet 10BaseT)
- souvent composé d'un châssis pouvant contenir N cartes
- peuvent être «**empilables**» (un seul domaine de collision)
- Hub plat : 8, 16, 24, 32 ports
- Carte dans châssis : 8,16,24 ports.

---

© Ahmed Mehaoua - 7

## Ponts

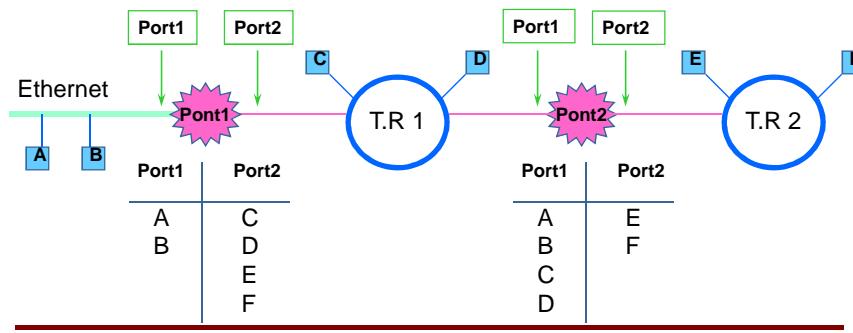
- Aussi appelé **Bridge**,
- Travail sur les trames au niveau **liaison**.
  - Offre les services des répéteurs, avec en plus :
    1. Permet de **segmenter** le réseau en sous-réseaux indépendants
    2. dispositif actif filtrant (collision) :
      - **permet de diminuer la charge du réseau : amélioration des performances.**
      - **Sécurisation des échanges entre segments**
    3. Capable de **convertir** des trames de formats différents (ex : Ethernet - Token Ring).
    4. **Administration** et filtrage configurable à **distance** (agent SNMP)

---

© Ahmed Mehaoua - 8

## Comment les ponts peuvent ils identifier les stations et leurs adresses Physiques ?

C'est le rôle des ponts à **AUTO-APPRENTISSAGE** aussi appelés PONTS TRANSPARENTS



© Ahmed Mehaoua - 9

## Ponts Transparents

- Conçus à l'origine pour interconnecter des réseaux **Ethernet**,
  1. fonctionnent en "auto-apprentissage"
    - découvrent automatiquement la topologie du réseau Ethernet
    - Fonctionne par défaut par inondation
    - le pont construit au fur et à mesure une **table de correspondance** entre adresses sources et segments sur lesquels les trames correspondantes sont acheminées.
  2. Evite les boucles dans des topologies multi-chemins par la mise en œuvre du protocole de l'arbre couvrant (**Spanning Tree Protocol**)
- Aujourd'hui également utilisés pour interconnecter les réseaux **Ethernet** et **Token Ring** :
  3. Convertissent les trames d'un format à l'autre,
    - les stations du réseau Token Ring doivent être configurées de manière à limiter la longueur de leurs trames à 1500 octets (longueur maximum d'une trame Ethernet); nécessaire car il n'existe pas de possibilité de segmentation au niveau de la couche LLC.

© Ahmed Mehaoua - 10

## Pont/Switch à Auto-apprentissage

### - Algorithme -

Lorsque le pont reçoit une trame :

Si une entrée valide dans sa table de pontage correspond à [adresse de destination](#) de la trame alors /\* la table de pontage contient l'adresse de destination \*/

. si le sous-réseau associé à cette entrée est différent du sous-réseau dont est issue la trame,  
. alors /\* la trame doit être pontée \*/

la trame (inchangée) est réémise sur ce sous-réseau.  
sinon / la table de pontage ne contient pas l'adresse \*/

. la trame est diffusée vers tous les sous-réseaux sauf le sous-réseau d'où elle provient.

Si aucune entrée valide dans sa table de pontage correspond à [adresse d'émission](#) de la trame alors /\* la table de pontage ne contient pas l'adresse d'émission \*/

. une entrée est créée dans la table de pontage associant l'adresse et le sous-réseau d'où est issue la trame.

sinon /\* la table de pontage contient déjà l'adresse \*/  
. si le sous-réseau dont est issue la trame et celui de l'entrée sont différents  
. alors

l'entrée est modifiée en conséquence

© Ahmed Mehaoua - 11

## Comment renforcer la fiabilité du réseau en cas de pannes d'un pont/switch ?

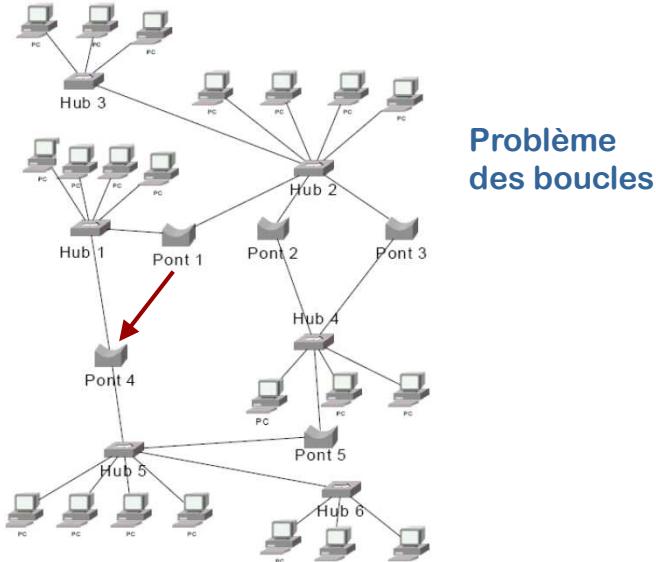
Installer plusieurs ponts/switchs en redondance, pour avoir une [topologie physique](#) multi-chemins.

- **Problème:** risques de bouclage des trames dans le réseau.
- **Solution:** utiliser un algorithme qui :
  1. construit une [topologie logique sans boucles](#) (un arbre) entre les ponts du réseau, garantissant un unique chemin entre deux stations
  2. et qui [reconfigure l'arbre en cas de pannes](#) d'un pont/switch

C'est le rôle du protocole de l'arbre couvrant ([Spanning Tree Protocol](#))

© Ahmed Mehaoua - 12

## Spanning Tree Protocol



© Ahmed Mehaoua - 13

## Spanning Tree Protocol

Definit dans la norme 802.1d

Arbre de recouvrement :

- . construction d'un arbre recouvrant tous les sous-réseaux
- . en éliminant certains ponts, on élimine les cycles
- . il existe plusieurs arbres recouvrants pour une même topologie!

Algorithme de construction d'un arbre de recouvrement total :

- . algorithme d'élection basé sur les adresses + coût + n° port.
- . la racine de l'arbre sera la station de + petite adresse
- . les liaisons actives seront celles de + faible coût à partir de cette racine.
- . en cas d'égalité, on choisit le + petit n° de port (interface de communication).

© Ahmed Mehaoua - 14

# Spanning Tree

## - Algorithme -

Pour construire un arbre couvrant, Les ponts ou commutateurs s'échangent périodiquement des trames de configuration (appelées des BPDU - Bridge Protocol Data Unit) pour invalider les chemins multiples susceptibles de créer des boucles au sein du réseau Ethernet.

L'arbre couvrant regroupe l'ensemble des plus courts chemins entre chacun des commutateurs (ponts) et un commutateur (pont) élu appelé commutateur-racine (pont-racine) (Switch Root).

Ce chemin est établi en fonction de la somme des coûts des liens entre les commutateurs et le commutateur-racine, ce coût étant basé sur la vitesse des ports.

*L'arbre couvrant est construit en 3 étapes :*

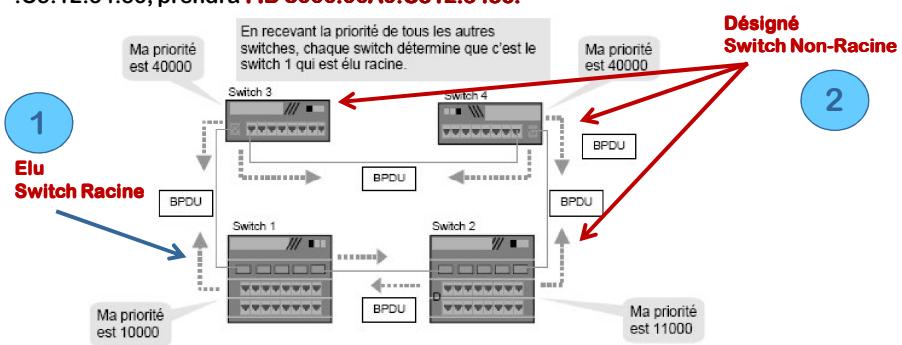
1. Sélection d'un Switch Racine (Commutateur Racine)
2. Sélection d'un port Racine (port root) pour les autres Switchs (Switch non-Root)
3. Sélection d'un port désigné pour chaque segment

© Ahmed Mehaoua - 15

# Spanning Tree

## Etape 1 – Election du Switch-Racine

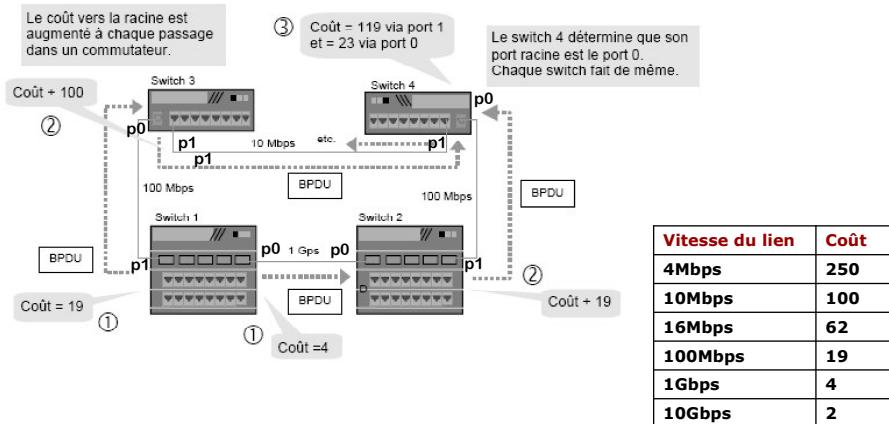
- C'est celui qui possède l'**ID le plus petit**.
- L'ID d'un commutateur comporte deux parties, d'une part, la priorité (2 octets) assigné par l'administrateur du réseau et, d'autre part, l'adresse MAC (6 octets). La priorité 802.1d est d'une valeur de 32768 par défaut (sur 16 bits). Par exemple, un switch avec une priorité par défaut de 32768 (8000 Hex) et une adresse MAC 00 :A0 :C5:12:34:56, prendra l'**ID 8000:00A0:C512:3456**.



© Ahmed Mehaoua - 16

## Spanning Tree

### Etape 2 – Selection d'un port-Racine pour chaque Switch (non racine)

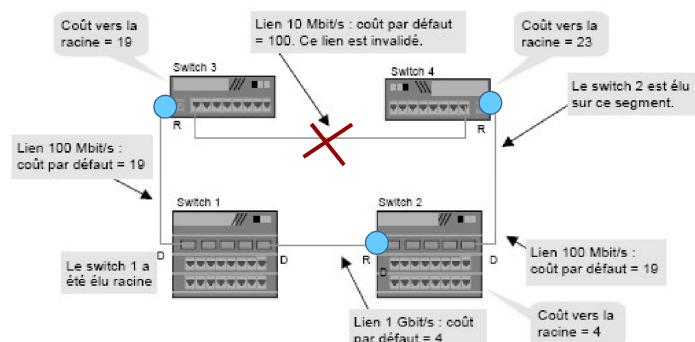


© Ahmed Mehaoua - 17

## Spanning Tree

### Etape 3 – Selection du port Designé pour chaque segment

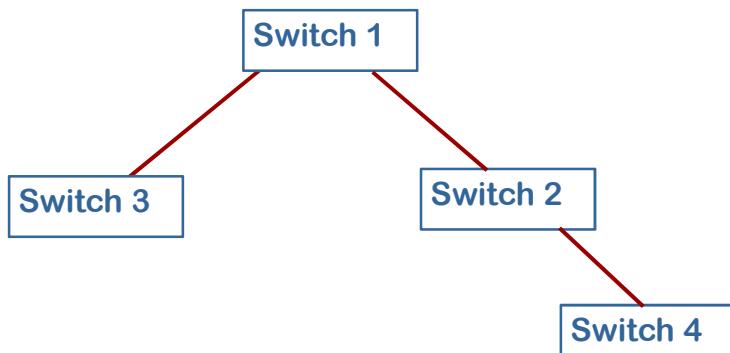
- sur chaque segment Ethernet, on détermine le **commutateur désigné** dont le port racine possède le coût de chemin vers la racine le plus bas. Ce port racine sera appelé **port désigné**. En cas d'égalité, c'est celui qui a la priorité la plus basse et, en cas de nouvelle égalité, celui qui a l'adresse MAC la plus basse.



© Ahmed Mehaoua - 18

# Spanning Tree

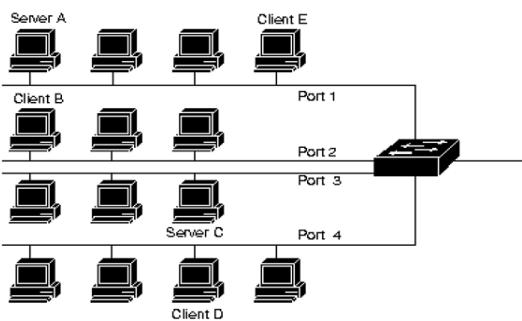
## Etape 4 – Le Spanning tree final



© Ahmed Mehaoua - 19

# Commutateurs

- Aussi appelé **SWITCH**, fonctionnent au niveau Liaison,
- Mêmes fonction qu'un pont mais utilisent des **ports dédiés** et non partagés,
- Commute** les trames au niveau MAC
- Peut gérer **simultanément** plusieurs communications (liaisons)



### Echanges simultanés :

- A (port 1) <--> B (port 2)
- C (port 3) <--> D (port 4)

### Echange non commuté :

- A (port 1) <--> E (port 1)

© Ahmed Mehaoua - 20

## Commutateurs

- Le commutateur établit et met à jour une table d'adresses MAC, qui lui indique sur quel port diriger les trames destinées à une adresse donnée.
- Lorsqu'une trame entre dans le commutateur, celui-ci conserve l'adresse MAC de l'émetteur et le port sur lequel il l'a reçu dans la table d'adresse. C'est ainsi que la table est établie et mise à jour.
- Si l'adresse du destinataire est inconnue, alors il envoie la trame à toutes les machines du réseau. Un commutateur est similaire à un **concentrateur** dans le sens où il fournit un seul domaine de diffusion.
- En revanche, chaque port a son propre domaine de collision. Le commutateur utilise la **micro-segmentation** pour diviser les domaines de collision, un par segment connecté. Ainsi, seules les interfaces réseau directement connectées par un lien point à point sollicitent le medium.
- Si le commutateur auquel il est connecté supporte le **full-duplex**, le domaine de collision est entièrement éliminé..

© Ahmed Mehaoua - 21

## Commutateurs

### - méthodes de transmission -

- La transmission des paquets peut s'opérer de différentes manières:
  1. **cut through** : le commutateur lit juste l'adresse du matériel et la transmet telle quelle. Aucune détection d'erreur n'est réalisée avec cette méthode.
  2. **mode différé (store and forward)** : le commutateur met en tampon, et le plus souvent, réalise une opération de checksum sur chaque trame avant de l'envoyer.

Un commutateur peut dans certains cas :

- prendre en charge plusieurs réseaux virtuels (**VLAN**),
- utiliser le **spanning tree protocol** pour éviter des boucles réseau, en particulier dans une architecture redondante,
- faire office de **routeurs**, on parle alors de commutateur de niveau 3 ou L3.

Les commutateurs ont aussi des fonctionnalités qui permettent à l'administrateur de surveiller le trafic :

- le **port mirroring** (miroirisation de port): le commutateur envoie une copie de tous les paquets à une connexion réseau de surveillance.

© Ahmed Mehaoua - 22

## Routeurs

- Aussi appelé **Router** ou **Gateway (Passerelle)** dans Internet,
- Ils fonctionnent au niveau **réseau** (couche 3 du modèle OSI), c'est à dire avec des adresses logiques (administrées).
- Des stations interconnectés aux moyens de HUBs forment un sous-réseaux, un **routeur** a pour objectif d'interconnecter des sous-réseaux co-localisés ou distants à travers des liaisons longues distances,
- **Avantages par rapport aux Ponts :**
  1. le routeur est indépendant des couches physique/liaison et par conséquent est parfaitement approprié pour interconnecter des réseaux physiques de nature différente (ex. Token Ring / X.25)
  2. Permet des interconnexions à travers des réseaux longues distances,

---

© Ahmed Mehaoua - 23

## Routeur coupe-feux

- Aussi appelé **pare-feux** ou **Firewall**,
- Routeur aux fonctionnalités étendues,
- permet une sécurité accrue (**Access Control List**),
- placés en front d'accès extérieur de manière à protéger le(s) réseau(x) interne(s);
  1. mise en oeuvre des fonctionnalités étendues entre la couche liaison Ethernet et la couche réseau IP par filtrage au niveau trame Ethernet et IP : vérifier si les règles de sécurité (définies par l'administrateur) autorisent le transfert du paquet vers le destinataire
  2. filtrage des requêtes FTP, HTTP, et autres services
  3. prévention contre les chevaux de Troie ou virus par filtrage E-mail, etc,
  4. vérification et enregistrement de toutes les communications.

---

© Ahmed Mehaoua - 24

## Passerelles

- Aussi appelé **Gateway**,
- Fonctionne au niveau 4 ou supérieur,
- Permet d'interfonctionner des systèmes d'information hétérogènes,
- Exemples : entre messageries d'entreprise, serveurs de fichiers, d'impression, ...

---

© Ahmed Mehaoua - 25



Université Paris Descartes  
UFR de mathématiques et Informatique

# Travaux dirigés Réseaux MLI536

## **Architectures, Codage, alphabets, numérisation**

### **Exercice 1.**

- a) Citer les codes ou alphabets que vous connaissez ! Quels sont les symboles représentables ?
- b) Comment représentons la parole, la musique, les images dans les applications usuelles : le téléphone, la télévision, le CD audio, ou le DVD ?
- c) Comment réalise-ton la conversion de ces informations de l'analogique au numérique ?
- d) Quels sont les intérêts du numérique par rapport à l'analogique ?

### **Exercice 2. : Architectures des réseaux**

- e) Qu'est ce qu'un protocole de communication ? une architecture de communication ?
- f) Quels sont les intérêts d'un modèle d'architecture de communication hiérarchique (en couches) ?
- g) Quelles sont les principales fonctionnalités de la couche **Physique** ? de la couche **Liaison** ?

### **Exercice 3.**

On considère un signal de parole de bande passante 4 KHz. On souhaite numériser ce signal et le transmettre sur le réseau NUMERIS en France. a) Quel sera le débit binaire de cette communication ? Veuillez préciser la fréquence d'échantillonnage, l'échelle de quantification, et la résolution de codage.

### **Exercice 4.**

Un CD audio contient 12 chansons pour une durée totale de 46 minutes 14 secondes. Sachant que le son est échantillonné à 44,1 KHz avec 16 bits par échantillon, et qu'il y a deux canaux de son (stéréo), quelle est la quantité d'information enregistrée sur le CD (en Mo) ?

### **Exercice 5.**

Soit à coder en binaire pour la transmission et le traitement informatique, une page A4 (A4 = 297 x 210 mm).

- a) on choisit de représenter chaque pixel par un bit (0 s'il est blanc, 1 s'il est noir). Sachant qu'il y a (pour le fax) 1728 pixels par ligne et 3,85 lignes par mm, quel est le volume de données binaires pour représenter ainsi une page (en mode portrait) ?
- b) Combien de temps faut-il pour transmettre la page numérisée à 9600 bit/s, à 64 Kbit/s ?
- c) Mêmes questions si l'on veut transmettre la page avec 256 nuances de gris (possibles pour chaque pixel).
- d) Que peut-on déduire quant à la méthode de codage utilisée dans un télécopieur classique ?

**Capacité binaire, Délai de transmission, Taux d'erreurs binaires,  
Contrôle d'erreurs**

**Exercice 1. : Débit maximal d'un canal de transmission**

Rappel :

La capacité maximale d'un canal de transmission numérique est la quantité d'information (en bits) pouvant être transmise par unité de temps (seconde). Il se mesure en bit/s et dépend des caractéristiques du support physique (bande passante, impédance) et/ou du signal (nombre de niveaux ou valence).

En 1924, un ingénieur suédois, Henry Nyquist, développa une formule pour exprimer la capacité maximale d'un canal **parfait** et de bande passante **finie**  $W$ . D'après Nyquist, le débit binaire maximal d'un canal non bruité de bande passante  $H$ , devant transmettre un signal composé de  $V$  niveaux discrets significatifs (appelé Valence du signal) est de :

$$(1) \text{ Théo. de Nyquist} \quad \text{débit binaire maximal (parfait)} = C = 2.W.\log_2 V \quad (\text{en bit/s})$$

En 1948, un ingénieur anglais, Claude Shannon, reprenait les travaux de Nyquist pour les étendre à des canaux soumis à des erreurs (aussi appelé bruit). Le rapport signal-sur-bruit (SNR) est un indicateur de la qualité de la transmission d'une information sur un canal bruité. C'est le rapport des puissances entre :

- le signal d'amplitude maximale ( $S$ ), déterminée par la valeur maximale admissible pour que les distorsions du signal restent à une valeur admissible (par exemple 1%);
- le bruit de fond ( $N$ ), information non significative correspondant en général au signal présent à la sortie du dispositif en l'absence d'une information à l'entrée.

Ce rapport signal-sur-bruit (SNR) est exprimé en fonction de l'atténuation du signal ( $S/N$ ) et est mesuré en décibels (dB) selon la formule (2) suivante où:

- $S$  représente le niveau du Signal,
- $N$  représente le niveau du bruit et
- $S/N$  représente l'atténuation du signal  $S$  en fonction du bruit  $N$ :

$$(2) \quad \text{Rapport signal-bruit} = \text{SNR} = 10.\log_{10} (S/N) \quad (\text{en dB})$$

pour rappels      (3)     $\log_2 (X) = \log_{10} X / \log_{10} 2$

$$(4) \quad a^{\log_a (X)} = X$$

D'après Shannon, le débit binaire d'information maximale transmissible sur un canal **bruité** de rapport signal-sur-bruit SNR (en dB) et d'atténuation du signal ( $S/N$ ), de bande passante **finie**  $W$  (en Hz), et quel que soit le nombre de niveaux (Valence) du signal à émettre, est :

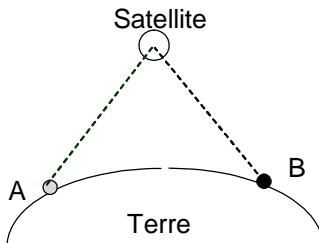
$$(5) \text{ Théo. de Shannon} \quad \text{débit binaire maximal (bruité)} = C = W.\log_2(1 + S/N) \quad (\text{en bit/s})$$

où  $S$  est le niveau du signal et  $N$  est le niveau du bruit et  $W$  la bande passante du canal.

- Soit un canal sans bruit de bande passante 4 KHz. Quel sera le débit maximal  $C$  sur ce canal si l'on transmet un signal binaire à 2 états (Valence  $V=2$ ) ?
- Les canaux de télévision ont une largeur de bande de 6 MHz. Combien de bits par secondes peuvent être transmis si on utilise des signaux numériques à 4 niveaux ? On supposera que le canal est sans bruit.
- Déterminer l'atténuation du signal ( $S/N$ ) correspondant aux rapports signal-bruit suivants SNR : 3 dB, puis 10 dB.
- Si un signal binaire à 2 états (Valence  $V=2$ ) est envoyé sur un canal à 4 KHz, dont le rapport signal-sur-bruit SNR est de 3 dB, quel sera le débit maximum sur ce canal bruité ? Comparer avec a).

### Exercice 2. Délais de transmission

Pour transmettre des messages entre deux terminaux A et B, on utilise un satellite géostationnaire situé à 36 000 km de la terre. La vitesse de propagation est prise égale à 240 000 km/s. On supposera que les messages font 1 kbits chacun, et que le débit binaire de la liaison est de 50 Kbit/s.



- Calculer le Temps d'émission ( $T_e, m$ ) d'un message sur la liaison. Dépend il de la taille du message ?
- Calculer le Temps de propagation ( $T_p, m$ ) terre-satellite-terre d'un message. Dépend il de la taille du message ?
- Calculer maintenant le Temps de transmission total ( $T_t$ ) d'un message de A vers B.
- La liaison satellite étant soumise à des erreurs de communication, A décide d'envoyer un message vers B et d'attendre que B acquitte ce message pour transmettre le message suivant. On supposera que la longueur d'un message d'acquittement est égale à 100 bits. Calculer le Temps de transmission total ( $T't$ ) pour la transmission du message ( $m$ ) et de son acquittement (ack). On supposera qu'il n'y a pas eu d'erreurs.
- Calculer le taux d'utilisation de la liaison, aussi appelé Efficacité ( $E$ ) c'est-à-dire le rapport du Tps d'émission effectif du message ( $T_e, m$ ) sur le Temps de Transmission « total » du message et de son acquittement ( $T'$ ).

### Exercice 3. : Taux d'erreurs binaires

Rappel :

Une liaison est caractérisée par son **taux d'erreurs binaires** ( $Te$ ) appelé BER pour Bit Error Rate en anglais. Ce taux d'erreurs est exprimé par le rapport entre le nombre d'informations (bits) erronées et le nombre d'informations (bits) transmises. Soit,  $Te = \text{Nb de bits erronés} / \text{Nb de bits transmis}$ .

- Si «  $Te$  » est la probabilité pour qu'un bit soit erroné, quelle est la probabilité de recevoir un bit correct ? la probabilité de recevoir  $N$  bits corrects ?
- Dans l'alphabet CCITT n°5, le mot « OSI », se code par les trois caractères de 7 bits suivants : « O » : 1001111 ; « S » : 1010011 ; « I » : 1000011.  
On supposera que le récepteur reçoit la suite de bits suivante : «1001011 1010101 1000011». Quel est le taux d'erreurs «  $Te$  » du canal ?

### Exercice 4. : Détection des erreurs par bits de parité

Rappel :

Pour détecter des erreurs lors des transmissions, il est courant d'introduire des informations complémentaires au message à envoyer, appelées codes de parité verticale et longitudinale :

**VRC** : (Vertical Redundancy Check) : à chaque caractère, on ajoute un bit appelé « bit de redondance verticale » ou « bit de parité », tel que le nombre de bits, à 1, à transmettre, soit pair (parité PAIRE) ou impair (parité IMPAIRE).

**LRC** : (Longitudinal Redundancy Check) : à chaque bloc de caractères, on ajoute un champ de contrôle supplémentaire construit de la façon suivante : On ajoute à chaque colonne (bits de parité VRC inclus), un bit de parité calculé de la même façon que VRC.

- Dans l'alphabet CCITT n°5, le mot « OSI », se code par les trois caractères de 7 bits suivants : « O » : 1001111 ; « S » : 1010011 ; « I » : 1000011.

Donner le mot de code sur 8 bits associé à chaque caractère LRC, puis le VRC correspondant en utilisant une parité PAIRE.

b) Même question que précédemment en utilisant une parité IMPAIRE.

### Couche LIAISON, Protocole HDLC

#### Exercice 1. : Contrôle de flux et Efficacité d'une liaison

Un canal a un débit de 2 Mbit/s (C), un délai de propagation de 20 ms ( $T_p$ ) et une vitesse de propagation sur le support (V) de 260 000 Km/s. On utilise un protocole d'échange de type « envoyer et attendre ». On suppose que le temps de traitement d'une trame est négligeable. On supposera que la longueur d'un acquittement ( $L_{acq}$ ) est de 100 octets (en-tête inclus).

- Quelle est la longueur de la liaison (d) ?
- Quelle taille de trames permet d'obtenir une efficacité de 50% ( $L_m$ ) ?
- On décide de fixer la taille de la trame  $L_m$  à 128 octets (en-tête inclus) et d'utiliser le mécanisme par fenêtre glissante de largeur  $n$  ( $W$ ). Déterminer  $n$  pour obtenir une efficacité optimale de 100% ?
- Quelle sera la longueur minimale (en bits) du champ numérotation de trames pour la question c) ?

#### Exercice 2. : Couche Liaison et protocole HDLC

Rappel : Le format d'une trame HDLC est le suivant.

Drapeau	Adresse	Commande	Données	FCS	Drapeau
---------	---------	----------	---------	-----	---------

Le format du champ de Commande HDLC est le suivant.

0	1	2	3	4	5	6	7
0	N(S)		P/F	N(R)			
1	0	M	P/F	N(R)			
1	1	M'	P/F	M'			

**champ C d'une trame I**

**champ C d'une trame S**

**champ C d'une trame U**

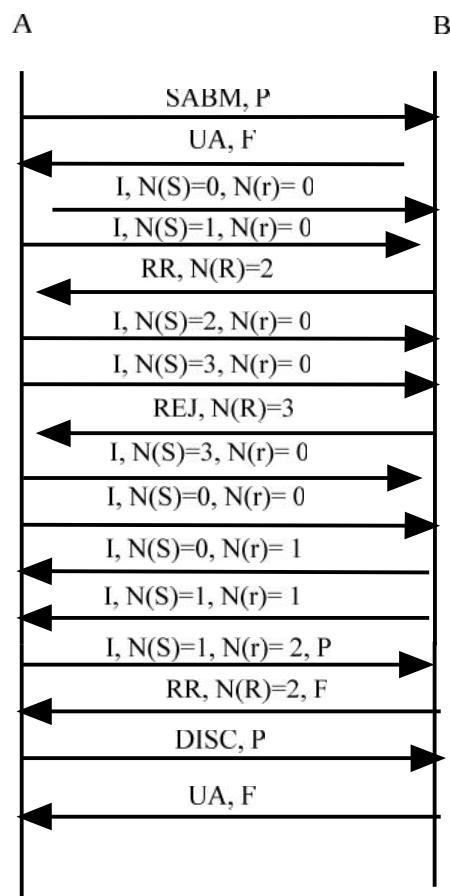
- Quel est le rôle des protocoles de niveau Liaison ?
- Dans le cas de la procédure HDLC, expliquer le rôle des champs N(S) et N(R) ?
- Comment s'effectue la synchronisation des horloges de l'émetteur et du récepteur dans une procédure HDLC ?
- On désire transmettre la suite de bits de données suivante avec le protocole HDLC:  
« 0111111001101110111110 »  
Quelle est la suite de bits réellement transmise au niveau physique ?
- On désire maintenant transmettre la suite des bits de données suivante :  
« 01101101001110110111110 »  
Par suite d'une erreur de transmission, la couche liaison du récepteur reçoit la séquence de données suivante (hors Fanions de début et de Fin) : 01101101001111101111100.  
Comment le récepteur interprète-t-il cette séquence de données ?
- Indiquer le type de reprises sur erreurs qui sera entrepris.

#### Exercice 3. : Analyse de diagramme d'échanges de trames HDLC

Commentez le diagramme d'échange ci-dessous dans le cas d'une liaison HDLC bidirectionnelle.

La numérotation se fait modulo 4 (0, 1, 2, 3). Et la fenêtre d'anticipation est de 2.

- Quels sont les rôles des trames SABM, DISC et UA ?
- Quel est le rôle de la trame RR ? REJ ?
- Combien de trames de données sont échangées entre A et B, et B vers A (hors retransmissions) ?
- Y a-t-il des erreurs de transmission ? Quelles sont les trames concernées ?
- Quel est le terminal qui décide de clore la liaison ? Est-ce que l'autre terminal peut faire de même ?



## Réseaux Locaux, Ethernet

### Exercice 1. : Adressage

Une entreprise dispose d'un réseau Ethernet. Un nouvel employé dans l'entreprise est doté d'un ordinateur ayant une carte Ethernet d'adresse universelle 3E 98 4A 51 49 76 (en hexadécimal). A quel niveau cette adresse est-elle gérée ? Est-il nécessaire de vérifier qu'aucun autre ordinateur ne dispose de la même adresse dans le réseau local ? Est-il possible de la modifier ?

### Exercice 2. : Dimensionnement d'un réseau local

Sur un câble coaxial en cuivre utilisé en Ethernet 10base5, la vitesse de propagation du signal électrique est de  $2.10^8$ m/s. Quelle est la longueur maximum d'un segment pour que le réseau local puisse fonctionner correctement sachant que la taille minimale d'une trame est de 64 octets ?

### Ethernet 3 : Analyse de trames

Rappels : Structure d'une trame Ethernet

Destination (6)	Source(6)	Type(2)	Données(n)
<b>Type</b> (0800 IP, 0806 ARP, 00c0 PPP)			

Rappels : Structure d'un paquet ARP :

Type mat. (2)	Protocole (2)	T. mat (1)	T. prot (1)	OP (2)	Adr. Mac émetteur. (6)	Adr. IP émetteur (4)	Adr. Mac destinataire. (6)	Adr. IP Destinataire (4)
------------------	------------------	---------------	----------------	-----------	---------------------------	-------------------------	-------------------------------	-----------------------------

**OP (0001 requête, 0002 réponse)**

Soient les suites hexadécimales ci dessous correspondant à la capture de deux trames de réseaux local Ethernet par un logiciel sniffer. Les octets de préambules ne sont pas représentés.

**Trame n°1 :**

FF FF FF FF FF FF 08 00 20 02 45 9E 08 06 00 01 08 00 06 04 00  
01 08 00 20 02 45 9E 81 68 FE 06 00 00 00 00 00 00 81 68 FE 05

**Trame n°2 :**

08 00 20 02 45 9E 08 00 20 07 0B 94 08 06 00 01 08 00 06 04 00  
02 08 00 20 07 0B 94 81 68 FE 05 08 00 20 02 45 9E 81 68 FE 06

En vous aidant du manuel de cours et/ou de l'Internet, veuillez :

- Préciser les valeurs et les significations des différents champs des trames échangées.
- Que représente la valeur « FF FF FF FF FF FF » ? Quand a-t-on besoin de l'utiliser ?
- A quoi sert le protocole ARP (Adress Resolution Protocol) ? Donner un sens à ces échanges de trames.

### Exercice 4. Simulation de l'algorithme CSMA/CD

Soit un réseau local Ethernet en bus comportant 4 stations : A,B,C et D utilisant la méthode d'accès au support CSMA/CD.

A l'instant  $t=0$ , la station A commence à transmettre une trame dont le temps d'émission dure 6 slots.

A  $t=5$ , les stations B, C et D décident chacune de transmettre une trame de durée de 6 slots.

L'algorithme de reprise après collision est le suivant :

```

Procédure Reprise_après_collision (attempts : integer ; maxBackOff : integer) ;
(attempts : compteur de tentatives de transmission)
(maxBackOff : borne supérieure de l'intervalle de tirage)

CONST
    slotTime = 51,2-s ;
    backOffLimit = 10 ;
VAR
    r, Delay : integer /*Nombre de slots d'attente avant de retransmettre*/
Begin
{
    if attempts = 1 then maxBackOff := 2 ;
    else {if attempts <= backOffLimit
    then maxBackOff := maxBackOff*2;
        else maxBackOff := 210;}
    r := delay := int(random*maxBackOff);
    wait (delay*slotTime);
}
End;
```

Int() est une fonction qui rend la partie entière par défaut d'un réel.

Random() est une fonction qui tire de manière aléatoire un nombre réel dans [0 ;1[

On considère que la fonction random rend respectivement les valeurs données par le tableau suivant :

Stations	A	B	C	D
1 <sup>er</sup> tirage	2/3	1/4	1/2	3/4
2 <sup>ème</sup> tirage	1/4	3/5	1/4	1/4
3 <sup>ème</sup> tirage	2/5	1/3	1/2	1/8

1°/ Dessiner un diagramme des temps gradués en slots décrivant le déroulement des différentes transmissions de trame.

On adoptera la représentation suivante :

CD



- Un slot occupé par la transmission correcte d'une trame de la station A est notée A
- Un slot occupé par une collision est noté X, avec les noms des stations impliquées mentionnés au dessus du slot.
- Un slot non occupé reste vide

2°/ Calculer sur la période allant de  $t=0$  à la fin de la transmission de la dernière trame, le taux d'utilisation du canal pour la transmission effective des trames

3°/ Calculer le délai moyen d'accès au support. Est-il borné ?

## **IP, Adressage**

### **1. CLASSES D'ADRESSAGE**

- 1.1 Qu'est-ce que CIDR et VLSM ? pourquoi les a-t-on introduit à partir de 1994 ?
- 1.2 Combien de types d'adresses IP différentes connaissez vous ? Citez les et donnez un exemple pour chacun d'eux.

### **2. MASQUE DE RESEAUX**

- 2.1 Quelle est la fonction du masque de réseaux sur un terminal IP ?
- 2.2 A quel moment le terminal fait il usage du masque ?
- 2.3 Pour chacune des classes d'adresses globales (A, B et C) donner le masque de réseau associé.
- 2.4 Soit une machine d'adresse IP 197.178.0.52/24. De quelle classe est cette adresse ? Quel est le masque du réseau ? Quelle est l'adresse du réseau ? Définir l'adresse de diffusion globale et l'adresse de diffusion restreinte pour ce réseau.
- 2.5 Les adresses de diffusion traversent-elles les routeurs ?
- 2.6 Soit la machine C possédant l'adresse 192.168.0.140/255.255.255.128. Nous voulons savoir si les machines A et B ayant respectivement pour adresses 192.168.0.20 (A) et 192.168.0.185 (B) sont sur le même réseau ?

### **3. SUBDIVISION DE RESEAUX**

- 3.1 Quelle est l'intérêt de la subdivision de réseaux ?
- 3.2 Vous êtes l'administrateur du réseau de votre entreprise, à qui l'on vient d'attribuer l'adresse IP 214.123.155.0. Vous devez créer 8 sous-réseaux distincts pour les 8 succursales de l'entreprise, à partir de cette adresse IP.
  - a– Quel est la classe de ce réseau ?
  - b– Quel masque de sous-réseau devez vous utiliser pour optimiser votre plan d'adressage ?
  - c– Combien d'adresses IP (machines ou routeurs) pourra recevoir chaque sous-réseau?
  - d– Quelle est l'adresse réseau et de broadcast du 5ème sous-réseau utilisable ?
  - e– Combien d'adresses IP distinctes est-il possible d'utiliser avec un tel masque, tous sous-réseaux possibles confondus ?

## **Capture, Filtrage et Analyse de trames ETHERNET avec le logiciel Wireshark**

*Wireshark* est un programme informatique libre de droit, qui permet de capturer et d'analyser les trames d'information qui transitent par les interfaces de communication du terminal sur lequel il s'exécute. *Wireshark* est ainsi apparenté aux logiciels appelés « Sniffer » ou « analyseur de trafic ». Il est multi-OS et téléchargeable sur le site [www.wireshark.com](http://www.wireshark.com).

Avec *Wireshark*, il est possible de capturer des trames Ethernet en temps réel directement sur les Cartes de communication du terminal, de sauvegarder les résultats de cette capture dans des fichiers qui peuvent être analysés ultérieurement hors ligne. *Wireshark* supporte un très grand nombre de protocoles de communication et de formats de fichiers de capture : Ethernet, ARP, IP, TCP/UDP, HDLC, etc ... libpcap/tcpdump, Sun's snoop/atmsnoop, LanAlyzer, MS Network Monitor, HPUX nettl, AIX iptrace, Cisco Secure IDS, etc....

Durant ce TP, nous allons :

- 1.lancer le programme Wireshark,
- 2.capturer et analyser une trame Ethernet
- 3.définir des filtres pour la capture et la visualisation des trames
- 4.Enregistrer le résultat de cette capture dans un fichier

### **Etape 1 : Lancement des machines virtuelles VMWARE et de Wireshark**

Sélectionner l'environnement graphique « Xfce4 » à la place de « Gnome »

1.1 – Démarrer le logiciel de virtualisation VIRTUALBOX sur votre poste au moyen de la commande suivante :

**[user1@machine] \$ virtualbox&**  
Lancer la machine virtuelle Serveur en sélectionnant la machine dans la liste (menu de gauche)  
Connectez vous en tant qu'administrateur sur le serveur avec :

**login = etu  
mot de passe = etu&reseaux**

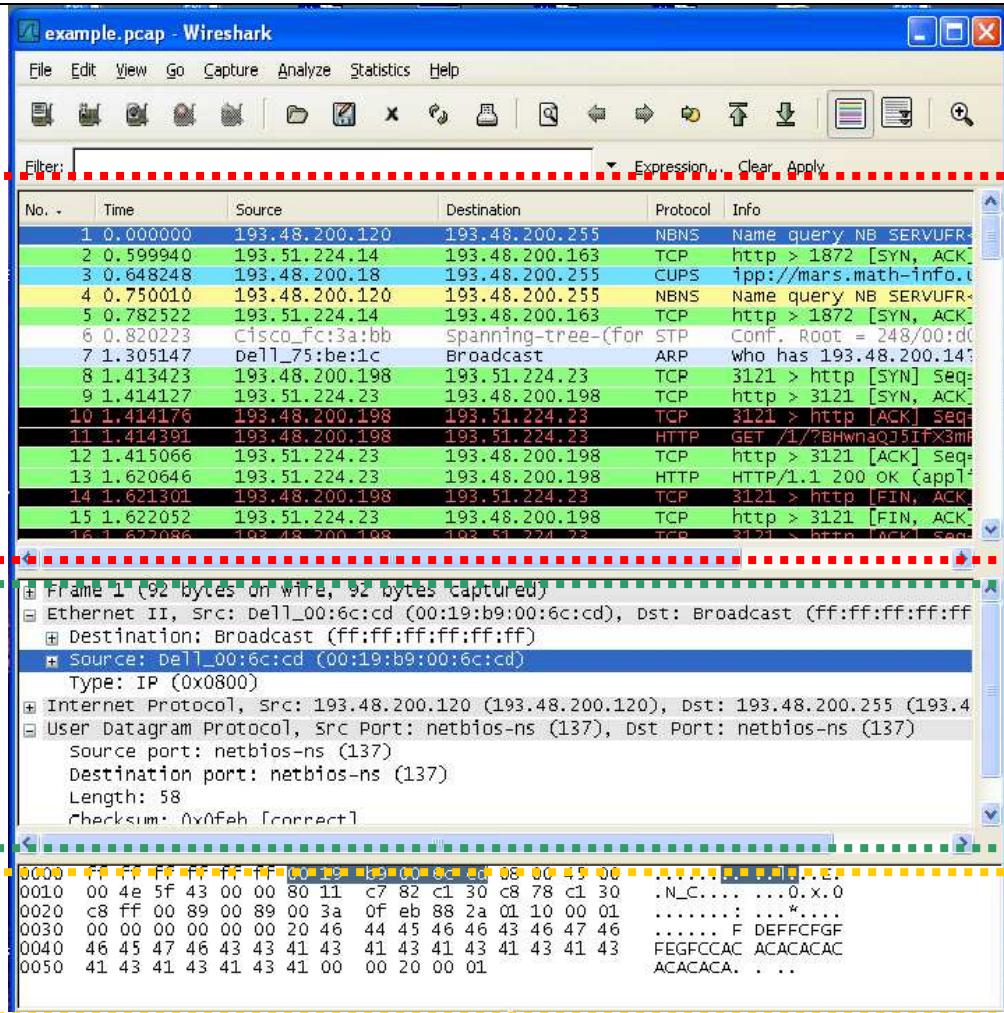
Lancer la machine virtuelle Cliente en sélectionnant la machine dans la liste (menu de gauche)

Connectez vous en tant qu'administrateur sur le client avec :

**login = etu  
mot de passe = etu&reseaux**

**Pour quitter le mode plein écran d'une machine virtuelle, veuillez taper CTRL+ALT+ECHAP**

1.2 Démarrez ensuite l'application *Wireshark*. Créez un raccourci sur votre bureau il vous sera bien utile.  
Voila comment le sniffer se présente.



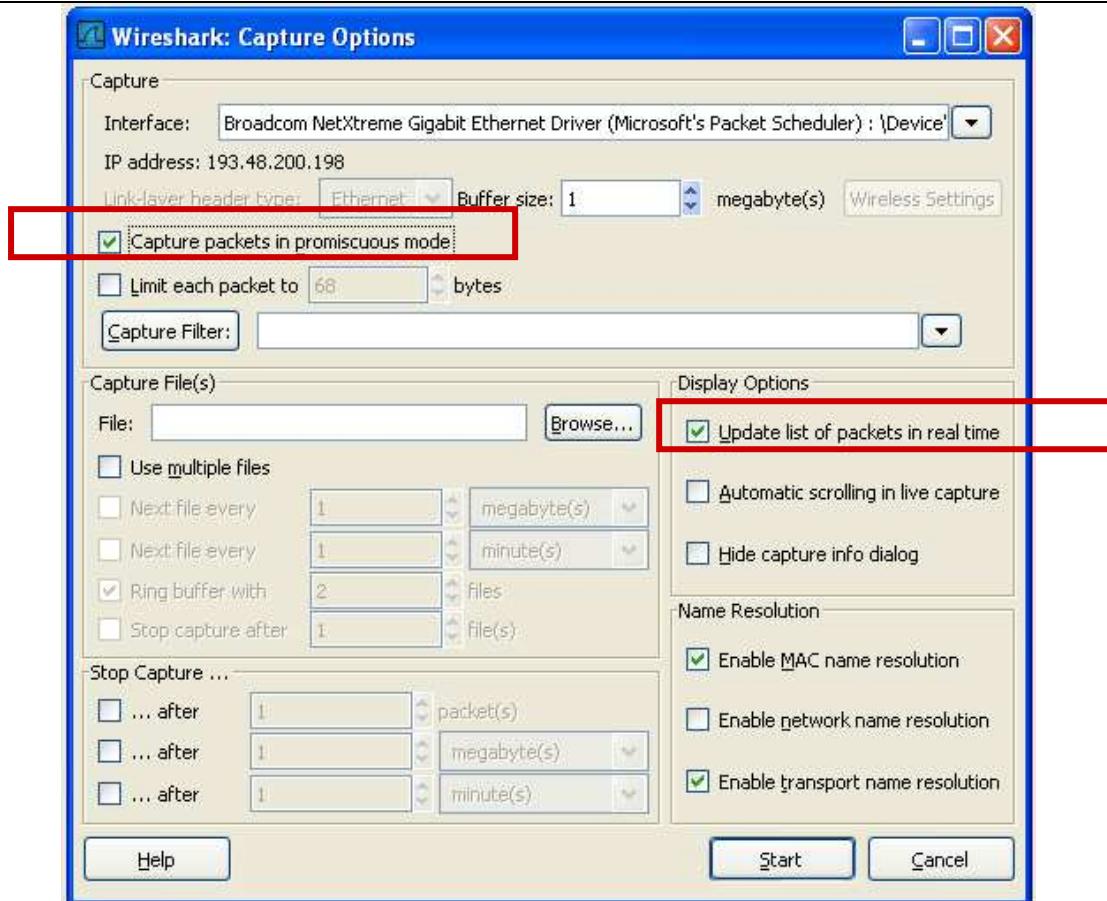
La fenêtre est divisée en trois parties.

1. La **première partie** est de type général, on y trouve des informations de type adresse IP des machines ou encore protocole utilisé lors de l'échange des données.
2. La **deuxième partie** de la fenêtre reprend ici la trame sélectionnée et la détaille soit dans les sept couches du modèles OSI ou dans les quatre couches du modèle IP. Pour plus d'informations à ce sujet des tutoriaux sont disponibles sur le net.
3. La **troisième et dernière partie** est une vision de la trame en codage hexadécimal et ASCII

Nous allons voir maintenant comment capturer les trames sur le réseau sur lequel le sniffer est connecté.

### Etape 2 : Capture de trames sur le réseau

Pour capturer les trames sur le réseau, vous devez allez dans le menu “Capture” et cliquez sur “Start”. La fenêtre suivante s’ouvre.



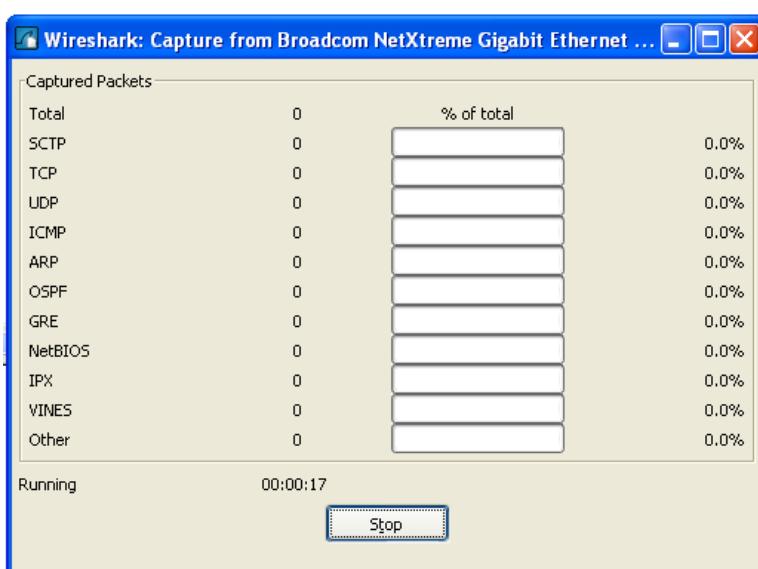
Choisissez l'interface sur laquelle vous voulez “écouter” le trafic. Si vous en avez qu'une le choix ne sera pas très difficile.

Par défaut l'espace réservé à la collecte des données est défini à 1MB. Cela devrait être suffisant. Dans le cas contraire augmentez-le.

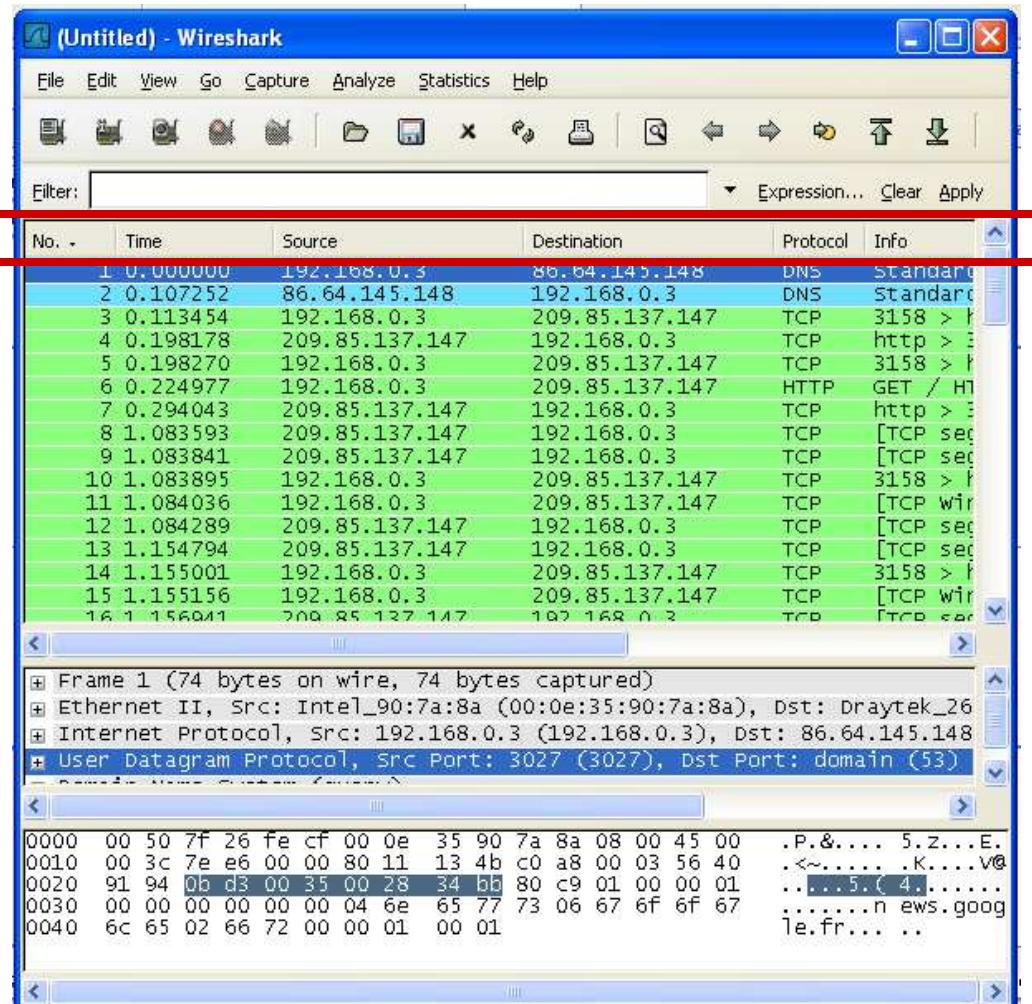
Activer l'option “**Capture packets in promiscuous mode**”. Cette option permet à la carte réseau de lire et d'intercepter tout le trafic sur le réseau. Dans le cas contraire celle-ci n'interceptera que les trames qui lui sont destinées et ainsi vous ne verrez pas toutes les trames Multicast et Broadcast.

Laissez le champ “**Capture Filter**” vide dans un premier temps. Nous verrons par la suite comment le remplir. Nous ne toucherons pas non plus aux autres options.

Il ne vous reste plus qu'à démarrer la capture en cliquant sur “OK”. La fenêtre suivante s'ouvre.



Capturez environ 30 secondes de trafic entre le poste client et serveur. Puis cliquez sur "Stop". Wireshark va alors afficher les trames capturées par votre carte réseau dans un format lisible ci-dessous.



Sur la première partie de cette fenêtre les différentes trames capturées s'affichent et suivant les colonnes nous avons les informations suivantes:

**Première colonne** : numéro de la trame.

**Deuxième colonne** : temps écoulé depuis le départ de la capture et l'arrivée de la trame.

**Troisième colonne** : adresse IP ou nom de la machine émettrice

**Quatrième colonne** : adresse IP ou nom de la machine réceptrice

**Cinquième colonne** : protocole utilisé entre les deux machines

**Sixième colonne** : informations complémentaires

La quantité de données capturées peut vite devenir considérable, d'autant plus que plusieurs communications peuvent être établies en parallèle comme par exemple une connexion à [www.google.fr](http://www.google.fr) et une autre à [www.tplpc.com](http://www.tplpc.com).

C'est pourquoi nous allons voir comment définir un filtre pour capturer une partie de tout ce que voit la carte réseau.

### Etape 3 : Les filtres

Il y a deux sortes de filtres. **Les filtres à la capture** et **les filtres à l'affichage**. Ces filtres n'ont pas la même syntaxe. Pour Unix la syntaxe des filtres à la capture est la même que les filtres utilisés pour la commande tcpdump. Pour en connaître le format, il faut donc utiliser man tcpdump. Quand aux filtres à l'affichage, la

La syntaxe est une syntaxe propriétaire à Wireshark. Pour en connaître la syntaxe, il faut utiliser la commande man wireshark. La section présente donne des exemples pour ces deux types de filtres.

## 1. Filtres de capture

Ne seront conservés que les paquets pour lesquels le filtre est vrai. Les filtres se décomposent en 3 parties :

- le **protocole** à capturer : exemples : ether, fddi, ip, arp, rarp, decnet, lat, sca, moprc, mopdl, tcp ou udp,
- l'identifiant qui peut être src ou dst,
- un champ qui peut être host, net ou port suivi d'une valeur.

Les opérateurs and, or et not peuvent être utilisés pour combiner des filtres.

Filtre	Fonction
host 172.16.0.1 and tcp	ne conserve que les paquets TCP à destination ou en provenance de la machine 172.16.0.1
udp port 53	ne conserve que les paquets UDP en provenance ou à destination du port 53
udp port 53 and dst host 172.16.0.1	ne conserve que les paquets UDP en provenance ou à destination du port 53 et à destination de la machine 172.16.0.1
tcp dst port 80 and dst host 172.16.0.1 and src net 172.16.0.0 mask 255.255.255.0	ne conserve que les paquets TCP à destination de la machine 172.16.0.1 sur le port 80 et en provenance des machines du sous-réseau 172.16.0/24

## 2. Filtres d'affichage

Les filtres d'affichage sont un peu plus fins que ceux de la capture. Seuls les paquets pour lesquels l'expression du filtre est vraie seront gardés. Les expressions sont basées sur les champs disponibles dans un paquet. Le simple ajout d'un champ veut dire que l'on garde le paquet si ce champ est disponible. Maintenant, on peut aussi utiliser les opérateurs ==, !=, >, <, >= et <= pour comparer les champs avec des valeurs. Les expressions ainsi fabriquées peuvent être combinées avec les opérateurs && (pour un et logique), || (pour un ou logique), ^ (pour le ou exclusif) et ! Pour la négation. L'usage des parenthèses est possible.

Voici quelques exemples de champs disponibles

Champ	Type	Signification
ip.addr	adresse IPv4	adresse IP source ou destination
ip.dst	adresse IPv4	adresse IP destination
ip.flags.df	booléen	Drapeau IP, ne pas fragmenter
ip.flags.mf	booléen	Drapeau IP, fragments à venir
ip.ttl	entier non signé sur 8 bits	Time to live
nbdgm.src.ip	adresse IPv4	adresse IP source d'un paquet Netbios Datagram
nbdgm.src.port	entier non signé sur 16 bits	port IP source d'un paquet Netbios Datagram
http.request	booléen	requête HTTP
http.response	booléen	réponse HTTP
icmp.code	entier non signé sur 8 bits	numéro du code d'une commande ICMP
icmp.type	entier non signé sur 8 bits	numéro du type d'une commande ICMP
ftp.request	booléen	requête FTP
ftp.request.command	chaîne de caractères	commande FTP
ftp.reponse.data	chaîne de caractères	donnée de transfer FTP
dns.query	booléen	requête DNS
dns.response	booléen	réponse d'une requête DNS

Voici quelques exemples de filtres

Filtre	Signification
ip.addr == 172.16.0.100	tous les paquets IP en provenance ou à destination de la machine 172.16.0.100

	172.16.0.100
(ip.addr == 172.16.0.100) && (dns.response)	tous les paquets IP en provenance ou à destination de la machine 172.16.0.100 qui sont des réponses à des requêtes DNS
(ip.addr >= 172.16.0.100) && (ip.addr <= 172.16.0.123)	tous les paquets IP en provenance ou à destination des machines comprises entre l'adresse IP 172.16.0.100 et l'adresse IP 172.16.0.123 (comprises)

### 3. Comment définir un filtre pour la capture des trames (Capture Filter)

Allez dans le menu "Capture". Puis cliquez sur "Capture Filters". La fenêtre suivante s'ouvre.

Considérons que notre machine a l'adresse IP 192.168.1.33.

Nous voulons capturer uniquement les trames échangées entre celle-ci et la machine avec l'adresse IP 145.200.80.45.

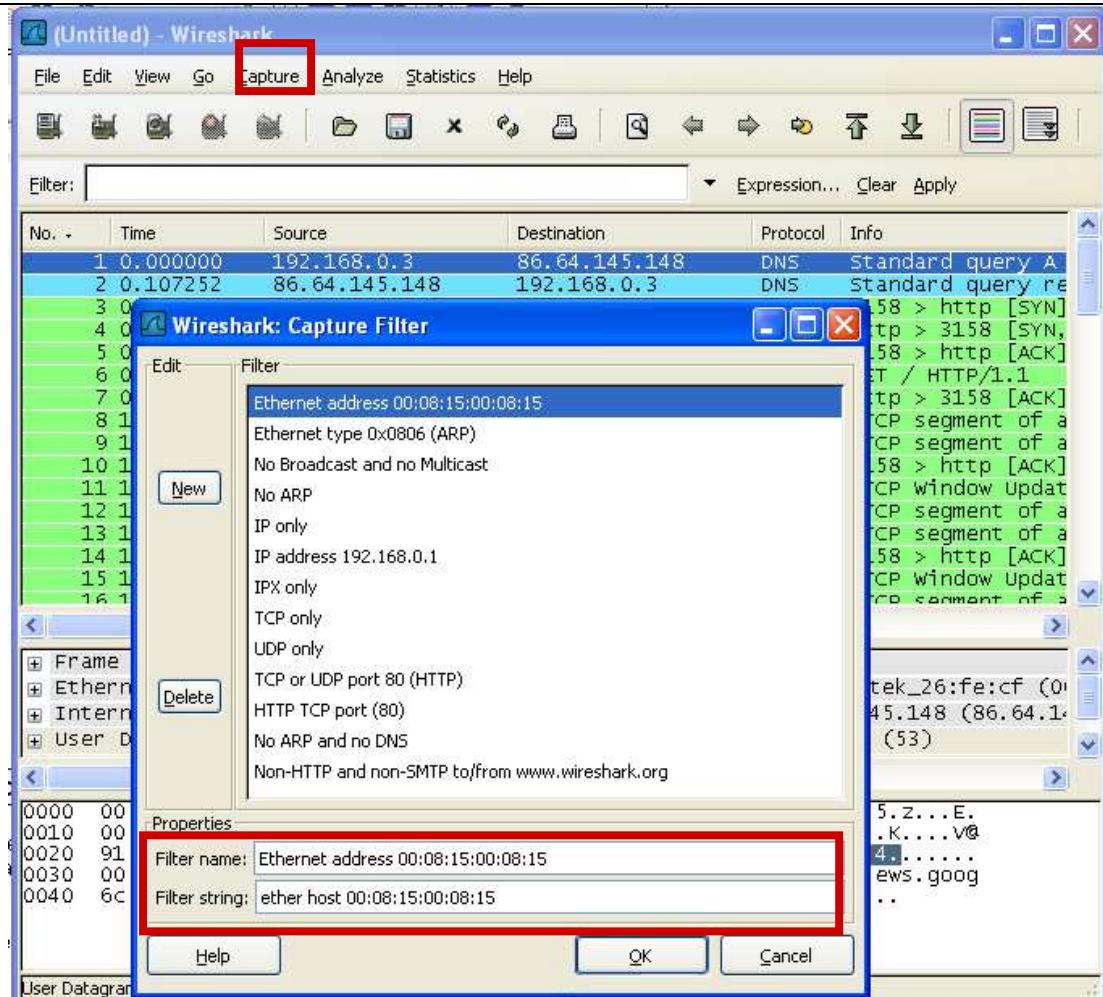
Pour cela cliquez sur "New".

Dans le champ "Filter Name" entrez le nom de votre filtre : mon filtre (par exemple).

Dans le champ "Filter string" entrez la chaîne suivante : host 145.200.80.45. Cliquez maintenant sur "save" et voilà votre filtre est défini vous pouvez cliquez sur "close" pour fermer la fenêtre.

Retournez dans le menu "Capture" et cliquez sur "Start". Reprenez les mêmes options que précédemment. Cliquez sur le bouton "Capture Filter" et sélectionnez votre filtre. Voilà cliquez sur "OK" pour démarrer la capture avec le filtre en question.

Pour plus de détail sur la structure des filtres vous pouvez consulter l'aide en appuyant sur la touche F1 et en allant sur l'onglet "Capture Filter"

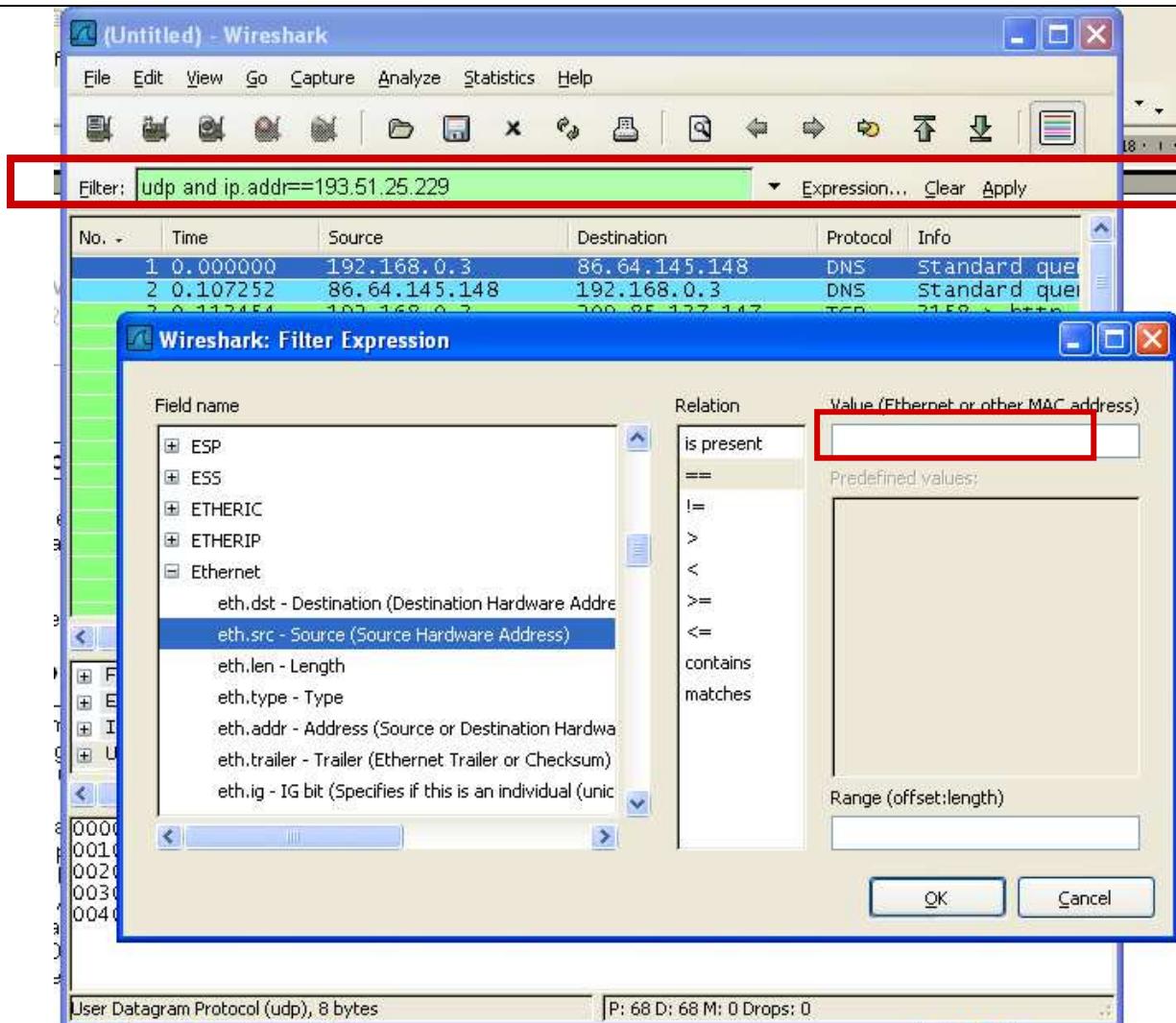


Une autre méthode consiste à capturer toutes les trames dans un premier temps et de filtrer par la suite. L'avantage de cette solution est d'avoir toujours la capture de départ et d'y appliquer par la suite autant de filtres que l'on souhaite. C'est ce que nous allons voir dans le prochain chapitre.

#### 4. Comment définir un filtre pour la visualisation des trames (Display Filter)

Essayons d'appliquer le même filtre que précédemment. Dans un premier temps faites une capture sans appliquer de filtre (reportez vous au premier paragraphe). Stoppez la capture. Allez sur la barre FILTER et sélectionner « EXRESSION ». une fenêtre s'ouvre vous permettant de rédiger des filtres d'affichage. Par exemple on sélectionne le protocole Ethernet et l'adresse source. On tape la chaîne suivante : eth.src==12:23:45:67:34 5A et on valide. Voilà le filtre d'affichage est appliquée. Si vous voulez le sauvegarder cliquez sur "Save".

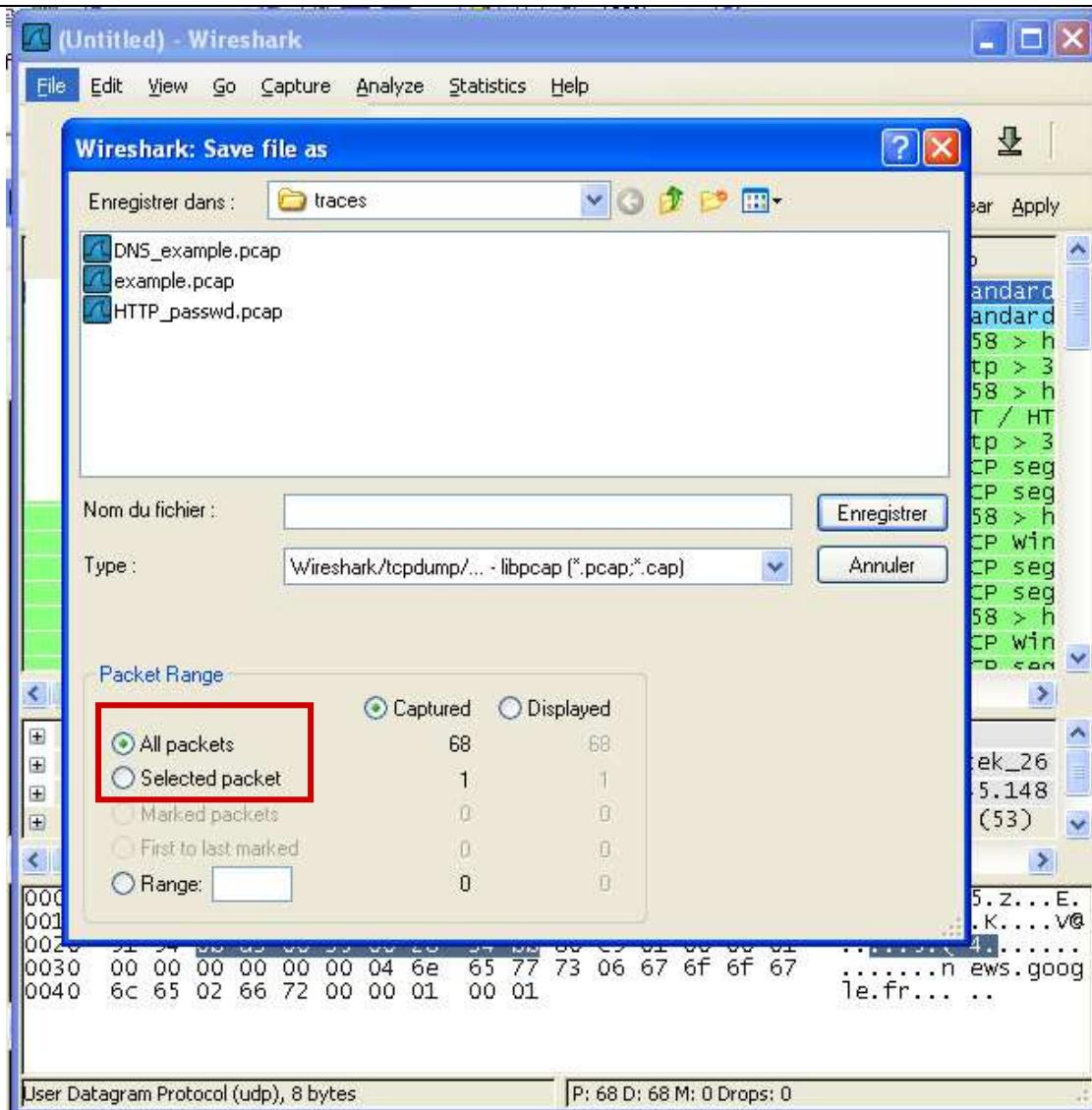
Si maintenant vous voulez l'annuler, effacez la chaîne dans le champ "Filter string" ou cliquer sur « CLEAR ».



#### Etape 4 : sauvegarde d'un résultat de capture

Pour sauvegarder le résultat d'une capture dans un fichier, il faut sélectionner la commande « Save as » dans le menu « File ». Une fenêtre nous proposera de choisir le répertoire et le nom du fichier, ainsi que le format/type de fichier de sauvegarde (conserver le format par défaut libpcap).

Pour n'enregistrer qu'une trame ou une sélection de trames, vous avez à votre disposition ces options dans le menu « Packet Range ».



### Etape 5 : Répondre aux questions suivantes :

**ATTENTION :** Avant de vous connecter sur votre poste avec votre compte client, Veuillez sélectionner l'environnement graphique « Xfce4 » à la place de « Gnome » (onglet en bas de votre écran)

- 5.1 Lancer la machine virtuelle « client linux fedora » puis la machine virtuelle « Serveur linux fedora »
- 5.2 taper sur la console du serveur la commande « ifconfig » (voir le manuel man pour la syntaxe de la commande ifconfig).
  - Combien d'interfaces trouvez-vous ? A quoi correspond l'interface « eth0 », l'interface « lo » ?
  - Identifier les adresses Ethernet (eth0) du serveur.
  - Identifier les adresses IP et le masque réseaux du serveur.
  - Quel est le type d'adresse IP (publique/privée) utilisé par le serveur ?
  - Réitérer les mêmes opérations avec le poste client.
- 5.3 sur le poste Client, lancer le logiciel Wireshark sur votre interface Ethernet (eth0) en mode « administrateur (root) », au moyen de la commande : \$> sudo wireshark
- 5.4 sur le poste Client, taper une commande de type « ping » à destination du serveur et capturer environ 30 secondes de trafic sur le poste serveur (voir le manuel man pour la syntaxe de la commande ping).
- 5.5 Combien de types de trames avez-vous capturé ?
- 5.6 filtrer votre capture pour ne sélectionner que les trames « icmp ». Puis analyser la première trame et indiquer la valeur des champs suivants :
  - en tête Ethernet : champ « TYPE »

- 
- en tête IP : champ « protocole »
  - en tête ICMP : champ « TYPE » et champ « IDENTIFIER »

5.7 Recherchez sur Internet le document RFC 1700. Quelle information mentionne t il en relation avec la trame Ethernet ? le message ICMP ?

5.8 Au moyen des filtres d'affichage sélectionner uniquement les trames dont l'émetteur est le poste client (sur la base de son adresse Ethernet). Puis sur la base de son adresse IP.

5.9 Décrivez la procédure (commandes systèmes, filtres wireshark) permettant de capturer et de filtrer les trames Ethernet transportant uniquement un paquet ARP ayant pour origine (émission) le poste serveur.

5.10 modifier l'adresse IP et le masque de réseau de votre serveur linux fedora avec les valeurs 172.24.0.2 et 255.255.0.0 (consulter le manuel système Linux/unix, « \$> man ifconfig »)

## IP : fragmentation et routage

### 1. Fragmentation des paquets IP

Soit un réseau constitué de 5 routeurs IP (R1 ... R5) et de trois stations A, B et C qui doivent communiquer (Figure 1).

Chaque liaison entre hôtes (station ou routeur) est étiquetée par son **MTU** (Maximum Transmission Unit). Le MTU définit la taille maximale d'un paquet IP qui peut être véhiculé dans les trames d'un réseau physique particulier. Ce paramètre est rattaché à une interface réseau du hôte (de numéro pi et d'adresse IP de classe B) pour fragmenter les données avant leur transmission sur la liaison.

On suppose que A doit émettre **1520 octets** de données vers B.

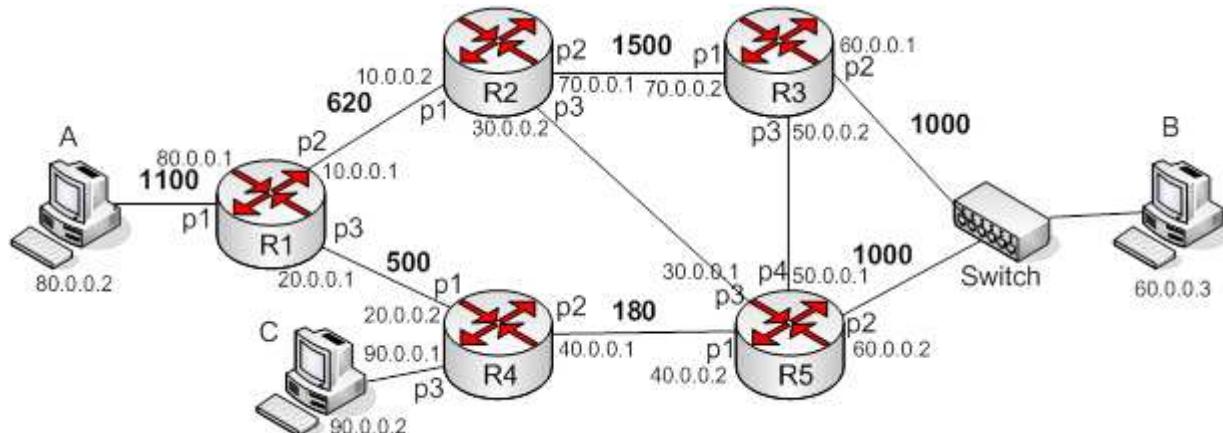


Figure 1 – Architecture de réseau

**1.1** Décrivez les fragmentations réalisées pour la transmission d'un paquet IP émis par A à destination de B, en supposant que le routeur R1 transmet alternativement les trames qu'il reçoit vers R2-R3 puis vers R4-R5. Précisez pour chaque fragment de paquet, les valeurs des champs (Identification, More Fragment Flag, Offset). On supposera que la valeur initiale de l'**Identifiant** du paquet est **71**.

**1.2** Justifiez le fait que la régénération des paquets fragmentés n'a lieu que sur la station destinataire et non sur les routeurs.

### 2. Routage des paquets IP

Soit le réseau de la Figure 1 (question 1 ci-dessus). A chaque liaison, supposée symétrique, est associée une distance égale à 1. On supposera que les routeurs mettent en œuvre un protocole de routage de type vecteur de distance avec l'algorithme Bellman-Ford.

**2.1** Quelle est la longueur maximale d'une route dans un réseau IP utilisant le protocole de routage RIP ? Comment s'assurer qu'un paquet IP ne bouclera pas sans fin en raison d'une table erronée ?

**2.2** On supposera que le réseau vient d'être mis en service par l'administrateur et que chaque routeur n'a qu'une connaissance locale de la topologie du réseau (il ne connaît que ses routeurs voisins et ses sous-réseaux voisins). Donner les tables de routage initiales des différents routeurs telles que configurées par l'administrateur, en suivant le format de table ci-dessous.

Adresse IP du réseau destination	adresse IP du prochain routeur	numéro de l'interface	Métrique (Hop Count) nbre de sauts

**2.3** Donner le vecteur de distance du routeur R1, que l'on notera VR1.

**2.4** On considérera la séquence d'échange de vecteurs de distance suivante:

<i>Instant</i>	<i>Evénement</i>
T <sub>1</sub>	R <sub>2</sub> , R <sub>4</sub> reçoivent VR <sub>1</sub> (vecteur de distance de R <sub>1</sub> )
T <sub>2</sub>	R <sub>1</sub> , R <sub>3</sub> , R <sub>5</sub> reçoivent VR <sub>2</sub>

Donnez la table de routage des routeurs suite à l'échange des vecteurs de distances VR<sub>1</sub> et VR<sub>2</sub>.

**2.5** Pourquoi et au bout de combien de temps une route est elle considérée comme non valide puis supprimée de la table d'un routeur ?.

**2.6** Travaux pratiques : RIP

2.6.1 - Visualiser la table de routage sur le serveur et sur le client, avec la commande : **route**

2.6.2 – Sur le serveur ajouter une route vers le client au moyen de la commande ci-dessous, et visualiser la nouvelle table:

**route add -net 192.168.119.0 netmask 255.255.255.0 eth0**

**Exercice 3. Table de routage IP : Jeu des 7 erreurs**

**Rappels :**

- On peut adresser directement tout poste connecté physiquement à son réseau et partageant la même adresse réseau au niveau IP.
- Un poste qui émet un paquet à destination d'un autre réseau IP, utilise une passerelle (routeur) qui se trouve sur son réseau.
- Les adresses de diffusion générale (255.255.255.255) ne passent pas les routeurs.
- L'adresse 0.0.0.0 est l'adresse par défaut (default sous unix), elle signifie "ailleurs", dans le sens où on utilise cette ligne de table pour router les paquets dont l'adresse de destination ne correspond à aucune adresse de la table de routage.
- L'adresse 127.0.0.1 est l'adresse de loopback, elle permet à un poste de "s'auto adresser".
- Chaque ligne de la table de routage se lit de la façon suivante : Pour atteindre l'adresse réseau (colonne 2) de masque réseau (colonne 3), je passerai par la passerelle (colonne 4) en utilisant la carte réseau d'adresse IP (colonne 5).

Question : huit erreurs se sont glissées dans la table de routage suivante, saurez vous les retrouver ?

Numéro de ligne	Adresse réseau	Masque réseau	Adresse passerelle	Interface
1	0.0.0.0	0.0.0.0	127.0.0.1	127.0.0.1
2	200.100.40.0	255.255.255.0	200.100.40.2	200.100.40.1
3	200.100.40.1	255.255.255.255	127.0.0.1	127.0.0.1
4	200.100.40.255	255.255.255.255	200.100.40.2	200.100.40.1
5	200.100.50.0	255.255.255.0	200.100.50.1	200.100.50.1
6	200.100.50.1	255.255.255.255	127.0.0.1	127.0.0.1
7	200.100.50.255	255.255.255.255	200.100.50.1	200.100.50.1
8	200.100.60.0	255.255.255.0	200.100.60.1	200.100.40.1
9	200.100.70.0	255.255.255.0	200.100.40.2	200.100.40.1
10	200.100.80.0	255.255.255.0	200.100.40.2	200.100.50.1
11	200.100.90.32	255.255.255.224	200.100.40.2	200.100.40.1
12	200.100.90.64	255.255.255.224	200.100.40.2	200.100.40.1
13	200.100.90.128	255.255.255.0	200.100.40.2	200.100.40.1
14	201.0.91.0	255.255.255.0	200.100.50.2	200.100.50.1
15	201.0.91.255	255.255.255.255	200.100.50.2	200.100.50.1
16	255.255.255.255	255.255.255.255	200.100.50.2	200.100.50.1

### Equipements d'interconnexion et Virtualisation de réseaux locaux

#### Exercice 1. : Equipements d'Interconnexion de réseaux locaux

L'interconnexion de réseaux locaux d'architectures différentes nécessite l'utilisation d'un équipement d'interconnexion. Citer les différents équipements que vous connaissez en précisant le niveau (couche du modèle OSI) où ils opèrent.

#### Exercice 2. : Les Commutateurs/Ponts (Switch/bridges) Ethernet Transparents

Soit un réseau d'entreprise constitué d'un commutateur Ethernet Transparent (à auto-apprentissage) et de 6 stations (A à F) réparties sur 3 segments tel qu'indiqué par la table ci-dessous.

Port 1	Port 2	Port 3
Station A	Station B	Station D
	Station C	Station E
		Station F

1. Faire un schéma du réseau local.
2. Indiquer l'évolution de la table du pont au fur et à mesure de l'émission des trames suivantes et de l'apprentissage des adresses MAC.

N°d'ordre	1	2	3	4	5	6
Source	A	B	C	A	E	F
Destination	B	A	B	B	B	D
Temps	T1	T2	T3	T4	T5	T6

<b>MAC address de</b>	<b>Port n°</b>	<b>Aging time</b>

#### Exercice 3 : Réseaux locaux Virtuels (VLAN pour Virtual Local Area Networks)

1. Dans le réseau local de l'exercice 2, on décide de remplacer l'équipement d'interconnexion « pont » par un « commutateur » Ethernet administrable. Quel est l'intérêt d'une telle modification ?
2. On choisit de restreindre les échanges de trames entre les stations A, B et F d'une part (VLAN\_1), et les stations C, D et E d'autres parts (VLAN\_2) au moyen de la technique des réseaux locaux virtuels (RLV ou VLAN en anglais). Citez deux modes de configuration de ces deux VLAN ?
3. En utilisant la technique de VLAN par port physique, proposer une configuration de la table de commutation du commutateur.

4. En utilisant la technique de VLAN par adresse MAC, proposer une configuration de la table de commutation du commutateur.

#### **Exercice 4. : Les Ponts et boucles**

Deux réseaux locaux Ethernet 10 base2 sont reliés par deux ponts (A et B) de telle sorte qu'elle forme une boucle. On supposera qu'il y a 3 stations par segment, soit 6 stations au total.

- A. Représenter graphiquement le réseau.
- B. Quel est l'intérêt d'avoir une topologie en boucle ?
- C. Que se passe-t-il au niveau des trames échangées sur le réseau quand la station A émet une trame vers la station B ?
- D. Une station souhaite transmettre une même trame vers toutes les autres stations du réseau local. Comment procède t elle ? Quelle est la valeur de l'adresse de diffusion (broadcast) MAC ?
- E. Que se passe-t-il quand une station émet une trame de diffusion (broadcast) sur ce type réseau local ?
- F. Comment éviter ce problème de bouclage à l'infini ?

#### **Exercice 5 : Interconnexion de LAN via des commutateurs ou ponts Ethernet redondants (Protocole de l'arbre couvrant)**

##### **Rappel :**

Pour construire un arbre couvrant, Les commutateurs ou ponts Ethernet s'échangent périodiquement des trames de configuration (**appelées des BPDU - Bridge Protocol Data Unit**) pour construire un arbre couvrant en invalidant les chemins multiples susceptibles de créer des boucles au sein du segment Ethernet. L'arbre couvrant regroupe l'ensemble des plus courts chemins entre chacun des commutateurs et un commutateur élu appelé commutateur-racine (Switch Root). Ce chemin est établi en fonction de la somme des coûts des liens entre les commutateurs, ce coût étant basé sur la vitesse des ports. Aussi, un chemin sans boucle suppose que certains ports des commutateurs soient bloqués et pas d'autres.

Le processus de création de l'arbre spanning tree peut durer plusieurs dizaines de secondes. Ce temps est, en réalité, proportionnel au nombre de commutateurs. Pendant cette phase, aucun commutateur ne traite de trame au cours des 15 premières secondes (valeur par défaut) ; le réseau s'arrête donc de fonctionner chaque fois qu'un commutateur est allumé ou éteint quelque part dans le réseau.

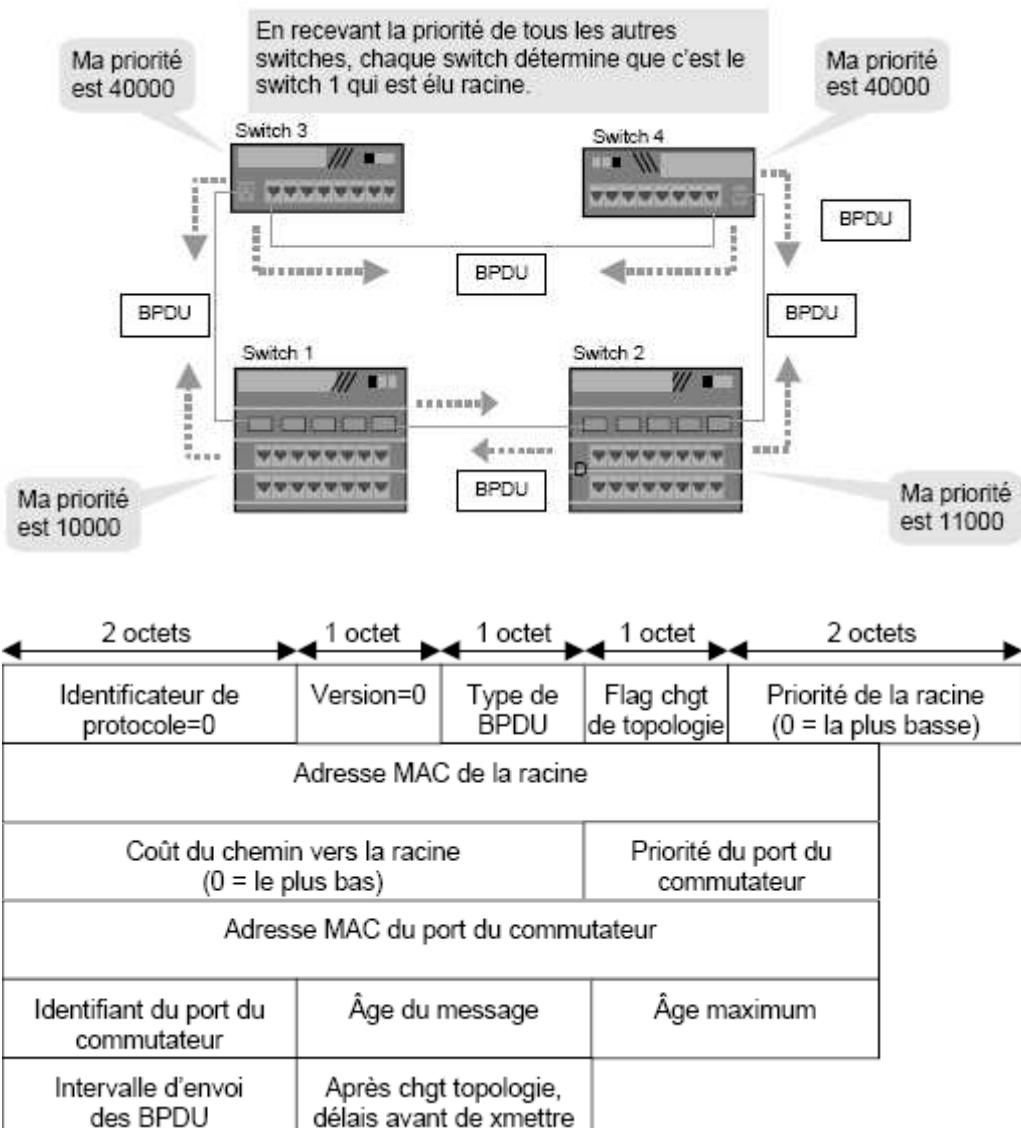
L'arbre couvrant est construit en 3 étapes :

1. Sélection d'un **Switch Root (Commutateur Racine)**
2. Sélection d'un **port Root pour les Switch non-Root (Port Racine)**
3. Sélection d'un **port désigné pour chaque segment (Port désigné)**

La première étape de ce processus consiste à élire un **commutateur racine** qui sera le point central de l'arbre couvrant: c'est celui qui aura l'**ID** la plus faible ou autrement dit celui qui possède la **priorité** la plus basse ou, en cas d'égalité, celui dont l'adresse MAC est la plus basse. Pour y parvenir, chaque commutateur émet un **BPDU** (Bridge Protocol Data Unit) de configuration contenant son **ID** sur tous ses ports. Inversement, il retransmet tous les **BPDU** (éventuellement en les modifiant) qui lui arrivent, et ainsi de suite jusqu'à ce que les **BPDU** échangés contiennent tous la même valeur. À son initialisation, chaque commutateur se désigne racine, puis compare les identifiants des autres **BPDU** qui arrivent. A la fin du processus d'élection, les commutateurs identifient et mémorisent le commutateur possédant le plus petit **ID** et le désigne comme **Switch Root**. Par la suite, le commutateur racine émet régulièrement (toutes les deux secondes par défaut) des **BPDU** pour maintenir l'état du spanning tree.

L'**ID** d'un commutateur comporte deux parties, d'une part, la priorité (2 octets) assigné par l'administrateur du réseau et, d'autre part, l'adresse MAC (6 octets). La priorité 802.1d est d'une valeur de 32768 par défaut (sur

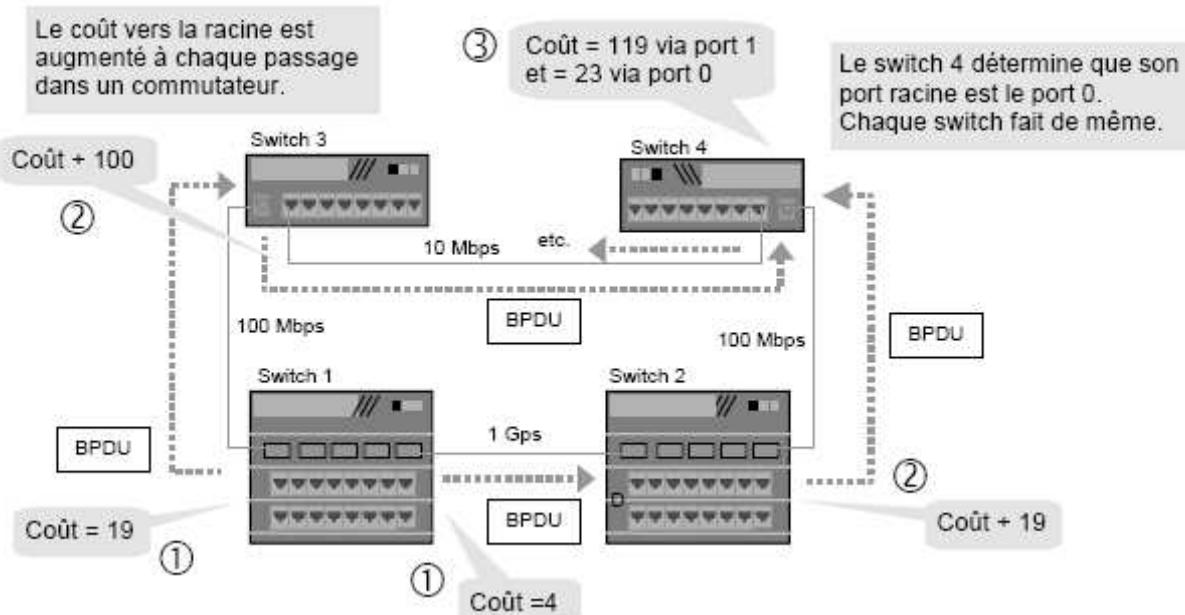
16 bits). Par exemple, un switch avec une priorité par défaut de 32768 (8000 Hex) et une adresse MAC 00:A0:C5:12:34:56, prendra l'ID 8000:00A0:C512:3456.



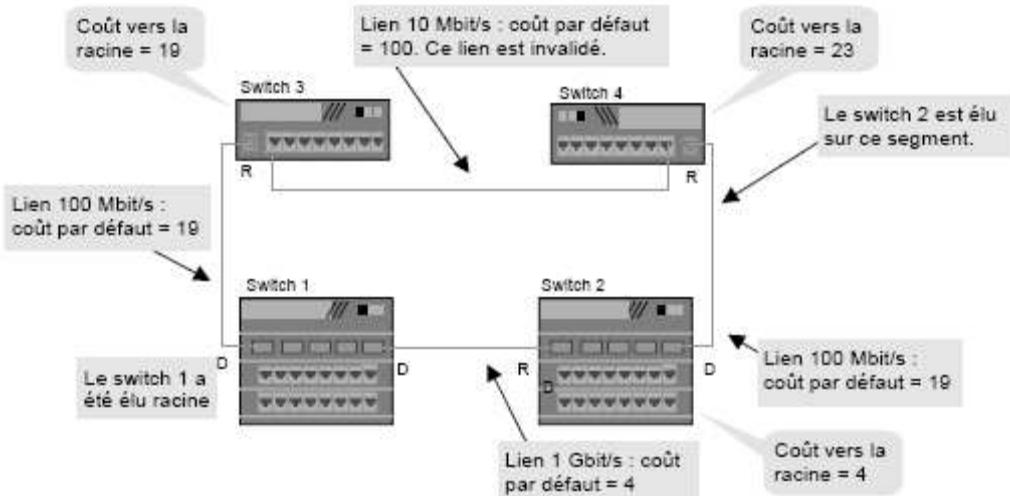
- Structure d'une trame BPDU -

La seconde étape consiste pour chaque switch non-Root à déterminer son **port racine** — c'est celui par lequel un BPDU émis par la racine arrive. S'il y en a plusieurs, le port choisi est celui qui a le **coût** de chemin vers la racine le plus bas. Le coût est déterminé par la somme des coûts des ports situés entre le commutateur et la racine. Le coût d'un port est inversement proportionnel à la vitesse du port (voir table ci-dessous). En cas d'égalité, le port choisi est celui qui a la priorité la plus basse; en cas de nouvelle égalité, c'est celui qui a l'adresse MAC la plus basse.

Vitesse du lien	Coût
4Mbps	250
10Mbps	100
16Mbps	62
100Mbps	19
1Gbps	4
10Gbps	2



La dernière étape consiste à déterminer sur chaque segment Ethernet, un **commutateur désigné** dont le port racine possède le coût de chemin vers la racine le plus bas. Ce port racine sera appelé **port désigné**. En cas d'égalité, c'est celui qui a la priorité la plus basse et, en cas de nouvelle égalité, celui qui a l'adresse MAC la plus basse. En définitive, sur chaque segment Ethernet, un seul chemin vers le commutateur racine sera calculé. Les commutateurs non-Root désactivent tous leurs ports qui ne sont ni racines ni désignés. Sur un switch Root, **tous les ports sont des ports désignés**, autrement dit, ils sont en état « *forwarding* », il envoient et reçoivent le trafic



#### Questions :

1. Rôle et définitions :
  - 1.1 Quel est le rôle de l'algorithme Spanning Tree ?
  - 1.2 Quel(s) est (sont) les équipements qui le mettent en œuvre ?
2. Construire l'arbre couvrant (Spanning Tree) du réseau de la figure 1 ci-dessous. En répondant aux questions suivantes :
  - 2.1 Déterminer le Pont Racine
  - 2.2 Déterminer les ports racines
  - 2.3 Déterminer les ports désignés

2.4 Déterminer les ports désactivés

2.5 En supposant que l'algorithme de *spanning tree* est stabilisé, donnez les tables relatives au *spanning tree* de chaque Pont.

2.6 Dessiner l'arbre couvrant reliant chacun des Ponts