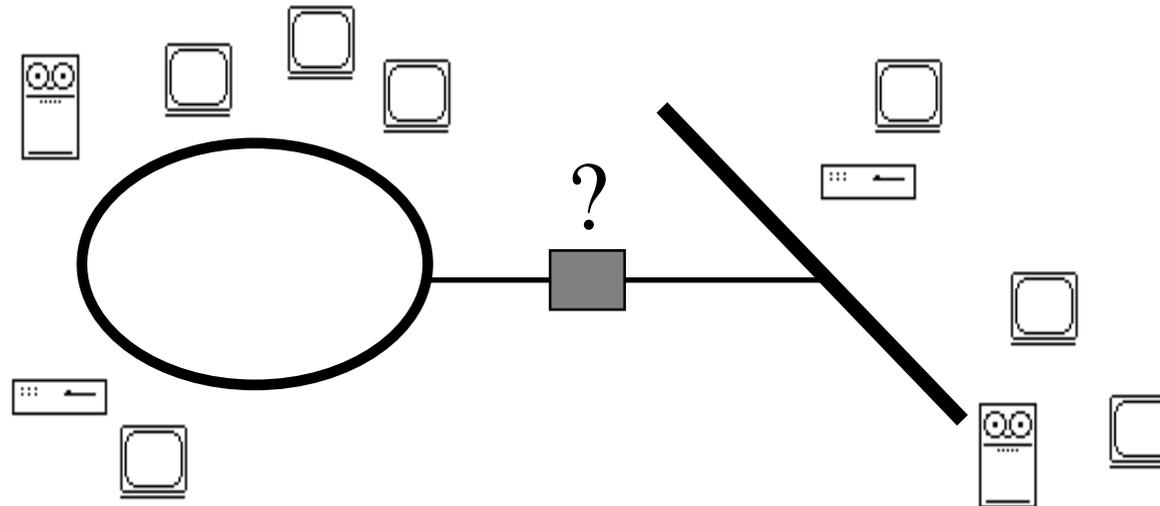


# Interconnexion de réseaux locaux

par  
Bernard COUSIN

(/home/kouna/d01/adp/bcousin/Fute/Cours/Reseaux-locaux/10-Pont.fm- 28 Juillet 1998 18:23)

**www.Mcours.com**  
Site N°1 des Cours et Exercices Email: [contact@mcours.com](mailto:contact@mcours.com)



# PLAN

- . Présentation
- . Architecture générale
- . Répéteurs - concentrateurs
- . Ponts
- . Transparent bridging
- . Source routing
- . Conclusion

# 1. Présentation

La supposition l'existence d'un monde homogène est fausse :

. Réseaux grandes distances :

- 20000 réseaux SNA (IBM) en 1990.
- 25000 sous-réseaux IP (en sept. 94).
- non OSI !

. Réseaux locaux :

- >1M.
- plusieurs méthodes d'accès incompatibles.

L'hétérogénéité : temporaire ou permanente ?

- évolution de la localisation et du nombre de stations
- émergence de nouveaux standards
- technologies radicalement différentes (RL, satellite, RG, etc)

Solution :

- un équipement d'interconnexion (répéteur, pont, routeur, passerelle).
- relié aux (2) réseaux à interconnecter.

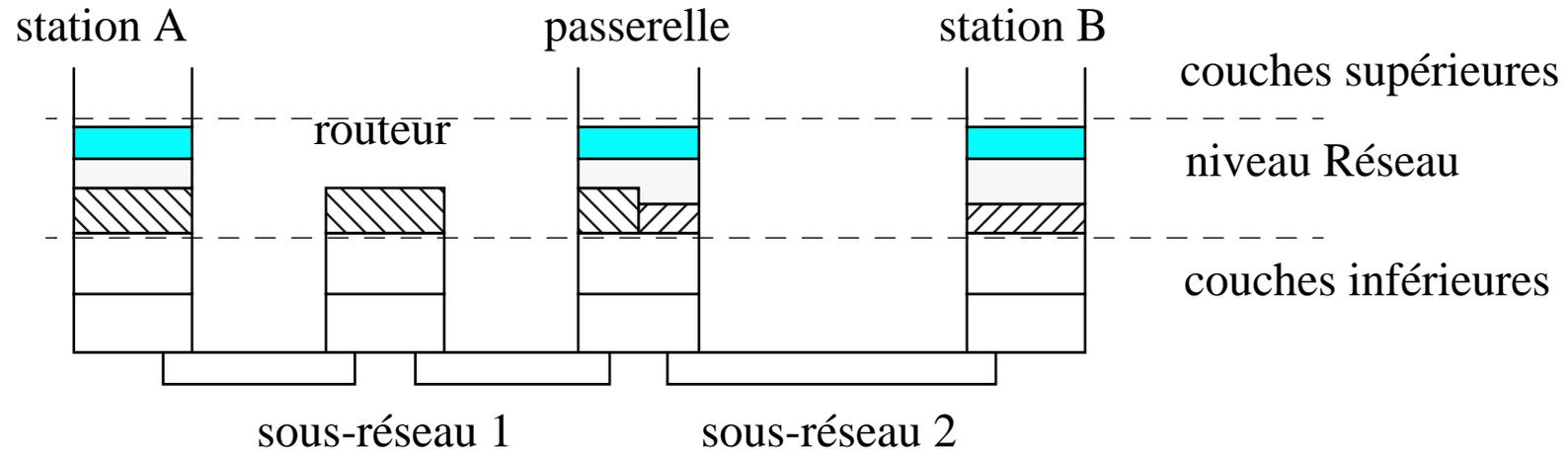
## 2. L'architecture générale

### 2.1. L'architecture d'interconnexion de l'OSI

De niveau Réseau !

3 sous-couches :

- a) d'accès au sous-réseau 
- b) de mise à niveau 
- c) d'interconnexion 



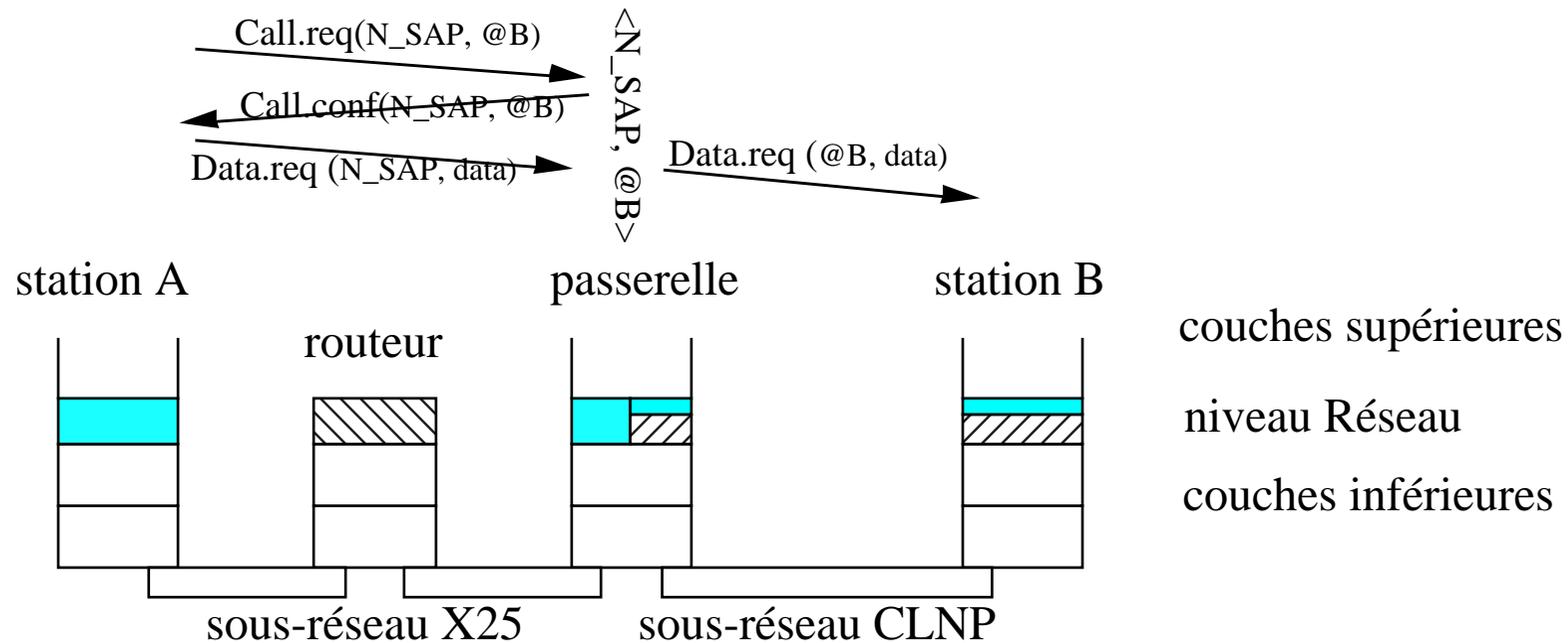
## 2.2. Exemple d'interconnexion

Interconnexion d'un sous-réseau en mode connecté à un sous-réseau en mode non-connecté, par exemple X25.3 et CLNP ("connectionless network protocol").

Définir la couche d'homogénéisation :

- soit mode connecté
- soit mode non connecté !
- soit autre !

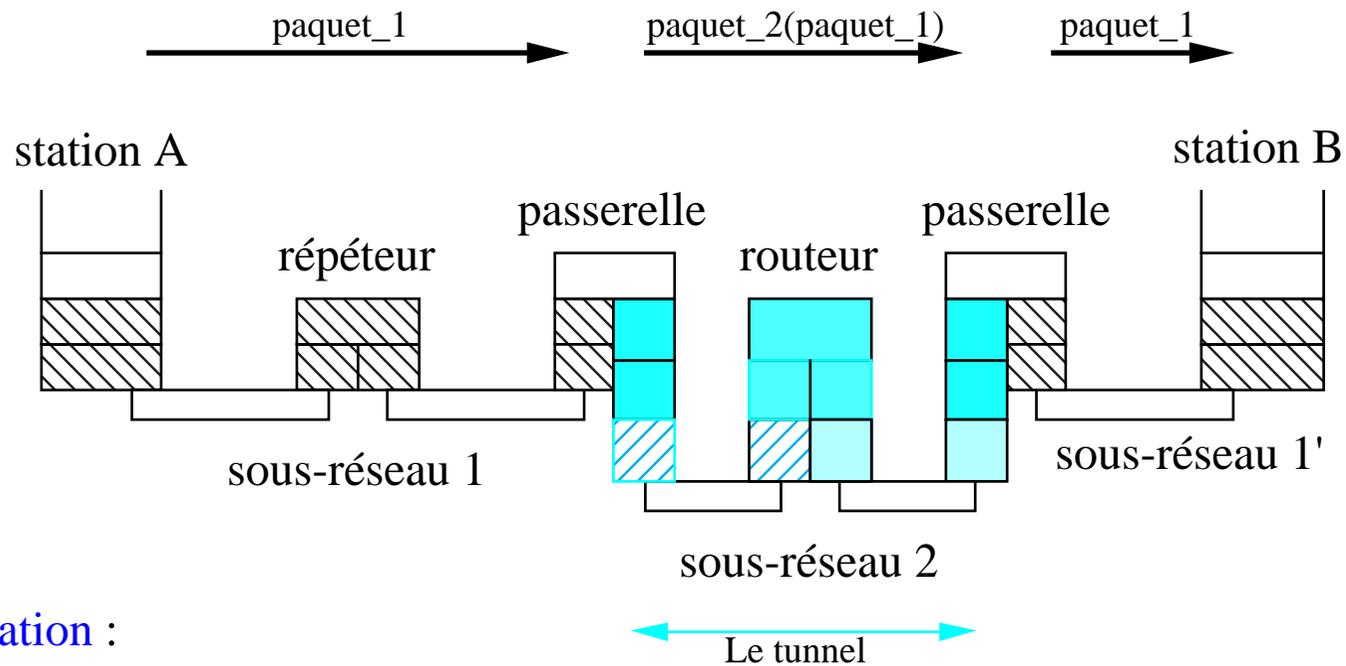
Quelque problèmes : l'adressage, l'existence de certains paquets, etc.



### 2.3. Le “Tunnelling”

**Interconnexion** de 2 (sous-)réseaux de type identique à travers un réseau d'un autre type :

- les messages des 2 sous-réseaux d'extrémité sont transportés tels quels par le réseau d'interconnexion.
- exemple : 2 réseaux Ethernet interconnectés par Transpac; ou le mbone.



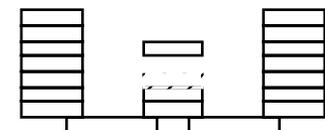
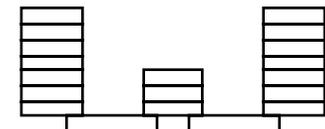
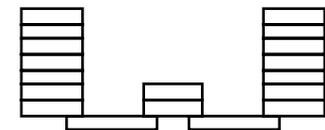
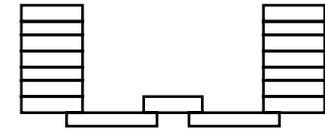
**Encapsulation :**

chaque paquet\_1 est placé en tant que données (encapsulé) dans un paquet\_2.

## 2.4. Les 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**
  - . conversion entre différentes méthodes d'accès
  - . par ex.: interconnexion de réseaux locaux
- couche 3 (Réseau) : **routeur**
  - . prévue pour !
- couches supérieures : **passerelle**, (relai, convertisseur de protocoles)
  - . interopérabilité de niveau applicatif
  - . par ex.: messagerie SMTP<-> X400



## 3. Modems, répéteurs et concentrateurs

### 3.1. Introduction

Traitement au niveau du signal dû aux limitations physiques,  
Pas de traitement sémantique (pas de notion de bit ou de trame)

#### Modem :

- rôle :
  - . adaptation entre deux supports physiques de communication différents

#### Répéteur :

- rôle :
  - . relie deux brins Ethernet (au sein d'un seul réseau Ethernet)
  - . extension de la couverture du réseau
  - . compatibilité entre deux segments utilisant des supports de type différents

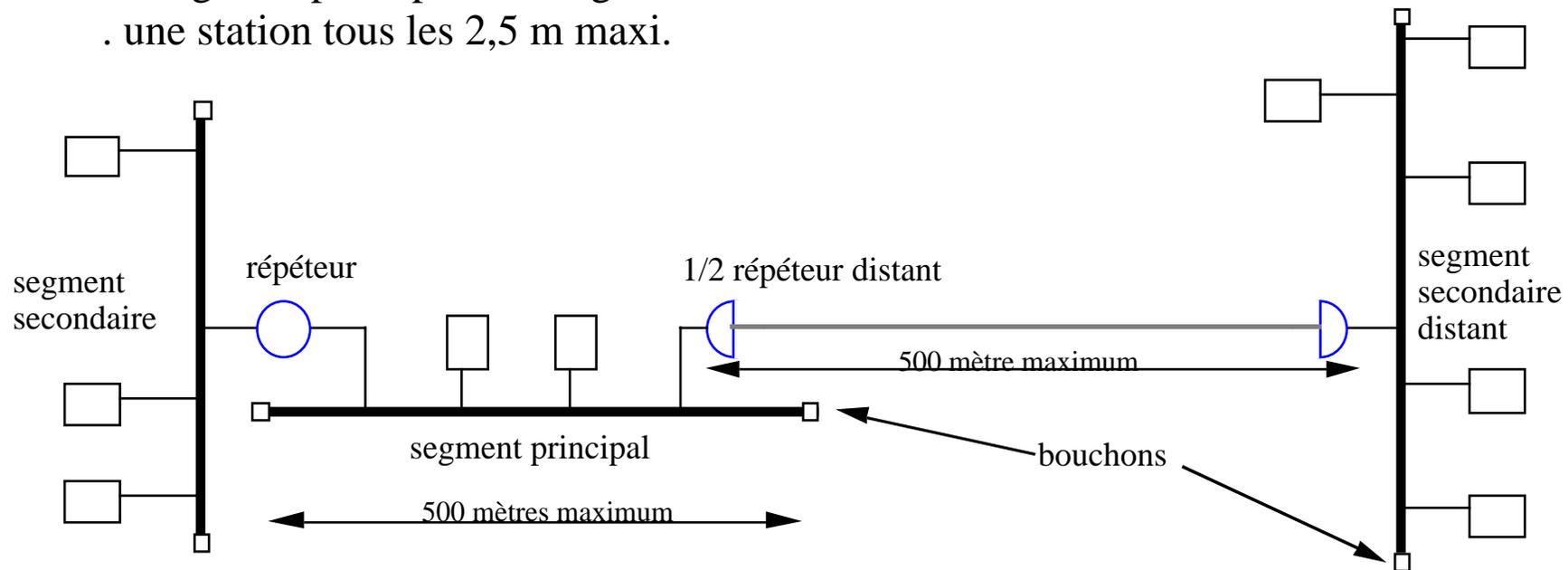
#### Concentrateur :

- rôle :
  - . permet le partage d'un seul point d'accès au réseau local entre plusieurs stations.
  - . +/- intelligent (**hub**)
- limitations :
  - . distance minimale entre points d'accès
  - . difficulté de la pose des prises sur le câble coaxial

### 3.2. Répéteurs Ethernet

Le protocole Ethernet utilise des répéteurs pour interconnecter ses différents segments (brins).

- . amplificateur du signal, détecteur de collision, générateur de brouillage
- . répéteur local ou distant (la liaison intermédiaire peut être en f.o.)
- . 1 segment principal, des segments secondaires
- . une station tous les 2,5 m maxi.



Ne pas confondre :

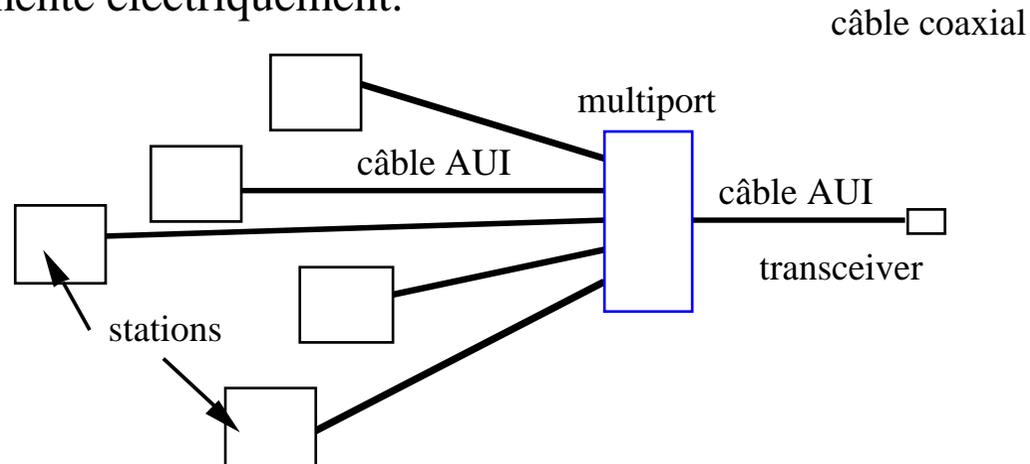
- . un segment d'un réseau local,
- . un réseau local (formé de plusieurs segments),
- . un ensemble de réseaux locaux interconnectés (par des ponts).

### 3.3. Concentrateurs



**Fan-out** (éventail), multiport :

- . distance minimale entre deux connecteurs,
- . topologie centrale (facilite la gestion),
- . équipement alimenté électriquement.



**Hub** (moyeu)

- . même principe (nouvelle terminologie),
- . architecture hiérarchique possible,
- . plus intelligent :
  - détection et contournement des pannes,
  - configurable à distance.
- . **Mediumless !**
- . empilable
- . **gérable à distance (pile SNMP)**

## 4. Les Ponts

### 4.1. Introduction

Interconnexion au niveau de la couche **Liaison de données**

- . Le réseau étant formé de deux réseaux locaux distincts, les informations passant de l'un à l'autre doivent obtenir le droit d'accès au médium du deuxième réseau.
- . Le pont fonctionne comme un récepteur sur le premier réseau et comme un émetteur sur le deuxième (et vice-versa).

#### Adressage

- . @ IEEE 802 dite universelle : 6 octets
- . l'unicité de l'adresse attribuée à chaque station est assurée

#### Transparence

- . le pont est insensible aux protocoles employés par les couches supérieures

#### Service

- . Le service assuré par l'interconnexion est le minimum commun

Augmentation de l'**étendue** couverte :

- . potentiellement infinie.
- . IRLE (Interconnexion de réseau local d'entreprise), "LAN\_E"

Augmentation de la **bande passante** :

- . le trafic local à un sous-réseau reste local
- . il y a deux (ou plusieurs) sous-réseaux : les débits sont potentiellement additionnés

Augmentation de la **sécurité** :

- . isolation : le trafic d'un sous-réseau ne pénètre pas dans l'autre sous-réseau
- . pont filtrant, "firewall"

## 4.2. Les types de ponts

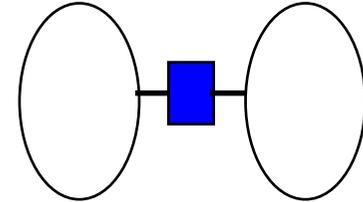
Un pont peut interconnecter deux RL identiques ou de types différents.

Homogène :

. les deux réseaux étant de type identique :

- le format des trames sont compatibles.
- les adresses sont compatibles

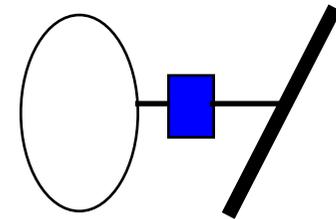
. simple : une seule entité de la sous-couche MAC est nécessaire dans le pont.



Hétérogène :

. le pont fonctionne comme un traducteur (traducteur) :

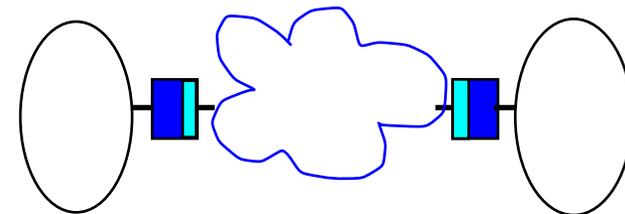
- de formats de trames (longueur max. des trames ?)
- de services (niveau de priorité ?)



L'interconnexion peut être directe ou indirecte.

Indirecte :

- . le réseau intermédiaire fonctionne en "tunnelling".
- . les trames originelles sont encapsulées dans les trames du réseau intermédiaire.



### 4.3. Les ponts filtrants

Il existe deux types de fonctionnement : les ponts simples et les ponts filtrants.

Pont simple :

- . Réémet toutes les trames reçues d'un réseau sur l'ensemble des autres réseaux.
- . Très peu utilisé, car coûteux.

Pont filtrant :

**Agit alors comme un routeur**

- . En fonction de l' **adresse de destination**
- . Connaît le sous-réseau sur lequel est connectée chaque station.
- . Seules sont réémises les trames dont on sait qu'elles sont à destination d'une station connectée à un autre sous-réseau que le sous-réseau de la station émettrice.
- . Les transmissions locales restent locales : le trafic est optimisé.

**Agit comme un “fire-wall”**

- . En fonction de l' **adresse d'émission**
- . Seules les trames issues de certaines stations sont traitées.

## 5. Le “Transparent bridging”

### 5.1. Introduction

Utilisé généralement pour Ethernet

Principe :

- . auto-apprentissage
- . diffusion par défaut
- . optimisation par “caching”

Fonctionne en **mode “promiscuous”** :

- le routeur traite toutes les trames circulant sur les sous-réseaux auxquels il est connecté.
- le routeur doit être puissant.
- le routeur est un point critique de sécurité.

Chaque routeur possède une **table** (de pontage) :

- qui associe à chaque adresse le sous-réseau vers lequel le routeur doit l’acheminer (cela peut ne pas être le sous-réseau sur lequel est connectée directement la station)
- la table est gérée comme une mémoire cache

**Adressage** : sur 8 octets

- 2 octets : priorité (0= +prioritaire), gestion la redondance par l'administrateur
- 6 octets : @IEEE 802

## 5.2. L'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

### 5.3. Conséquences

Le nombre d'adresses peut être grand :

- grand espace de stockage !
- le temps de recherche est important :
  - ⇒ utilisation d'un procédé de hachage
- récupération des entrées les moins utilisées (LRU : Least recently used)

Utilisation d'un temporisateur :

- . détection des entrées inactives ( environ 20 secondes)
- . suppression des entrées périmées ( déplacement ou disparition des stations)

Par défaut l'algorithme fonctionne :

- . par inondation totale
- . robuste mais peu efficace

Un ensemble des sous-réseaux peuvent être interconnectées par plusieurs ponts :

- . Assure une certaine redondance (volontaire ou par mégarde).
- . Le réseau peut présenter alors des cycles :
  - ⇒ Une même trame peut être recopiée et reçue indéfiniment !

## 5.4. Le “Spanning tree”

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).

Structure des messages de configuration [valeur lors de l'initialisation]:

- . Identité de la racine (proposée) [l'émetteur].
- . Coût de la liaison entre l'émetteur et la racine [0].
- . Identité de l'émetteur.
- . Numéro du port de l'émetteur.

## 5.5. L'algorithme du "Spanning tree"

A l'initialisation, chaque routeur **diffuse** sur tous ses ports un message de configuration contenant la valeur initiale.

Chaque pont mémorise la meilleure valeur reçue sur chacun de ses ports.

Lorsqu'un pont reçoit un message de configuration :

Il compare les valeurs courantes locales avec la valeur reçue :

- Si cette dernière est meilleure (+ petite)
  - . Le port de réception devient le chemin vers la racine
  - . Un nouveau message de configuration est calculé :
    - le coût est incrémenté
  - . Puis diffusé sauf vers le port de réception.
- Tout port qui a reçu une meilleure configuration que la configuration qui vient d'être calculée est désactivé.

La désactivation d'un port, et le calcul d'une nouvelle reconfiguration sont retardés :

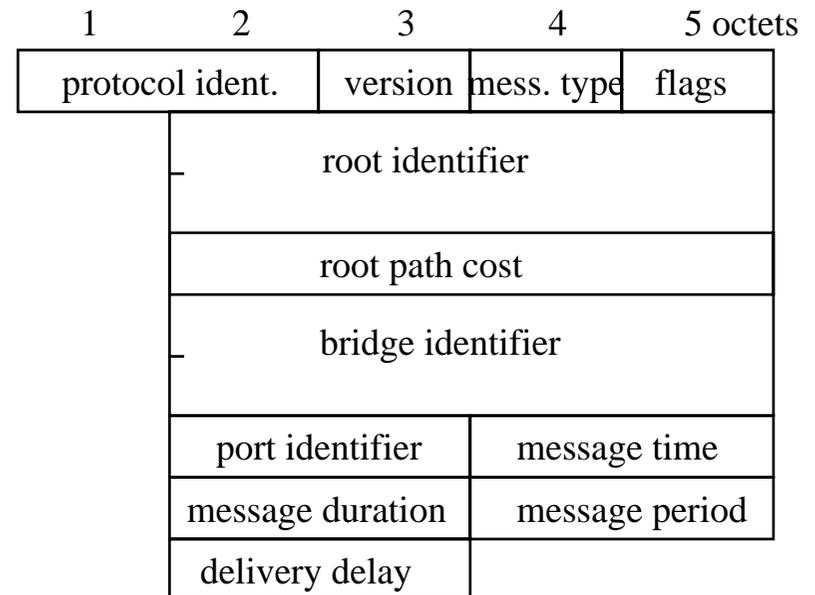
- . afin que chaque pont du réseau est le loisir d'arriver dans un état stable (délai de délivrance : 15s)

Les liaisons doivent être surveillées :

- . pour détecter toutes pannes.
- . un message de configuration est émis périodiquement (toutes les 4 secondes) par la racine et propagé par les ponts.
- . chaque message est numéroté, pour détecter les pertes.

## 5.6. Le format des messages

- . protocol identification : [0]
- . version : [0]
- . message type :
  - 0 = message de configuration
  - 128 = message de changement de topologie
- . flags :
  - Topology change bit
  - Topology change acknowledgment bit
- . root identifier : le numéro de la racine
- . path cost : le coût du chemin entre l'émetteur et la racine
- . bridge identifier : le numéro de l'émetteur du message
- . port identifier : le numéro du port d'émission
- . message time : heure d'émission du message (1/256s)
- . message duration : durée de vie du message (1/256s)
- . message period : fréquence d'envoi de message de surveillance (1/256s)
- . delivery delay : retard d'émission des messages de reconfiguration (1/256s)



## 5.7. Conclusion

L'algorithme du "Transparent bridging" est un algorithme simple.

Dans un environnement d'interconnexion non-contrôlé, il nécessite la mise en oeuvre d'un algorithme complémentaire : le "Spanning tree".

Il est particulièrement utilisé pour réaliser l'interconnexion des réseaux locaux Ethernet.

Il fonctionne par défaut par inondation,

et utilise l'auto-apprentissage pour optimiser l'acheminement des trames.

Le Spanning tree supprime la redondance,

et ne permet donc pas de l'utiliser pour étaler le trafic.

## 6. Le “Source Routing”

### 6.1. Introduction

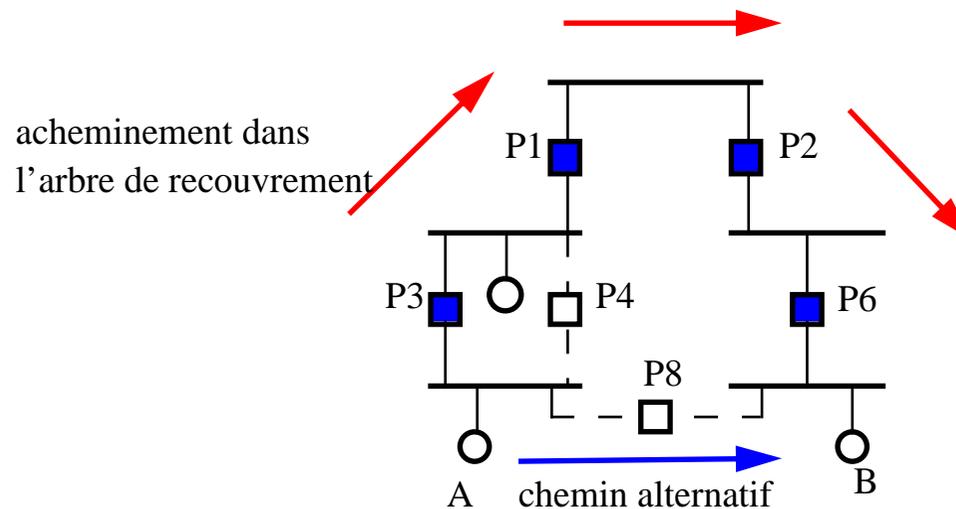
Proposé par IBM pour le Token Ring.

Problèmes du “Transparent bridging” :

- . les ponts redondants ne sont pas utilisés
- . un seul pont (un seul chemin) entre 2 points du réseau
- . les chemins optimaux ne sont pas suivis

Les ponts en mode “promiscuous” travaillent beaucoup :

☞ les ponts “Source routing” n'explorent que les trames marquées



## 6.2. Constitution des trames

Principe :

Chaque station connaît le meilleur chemin à suivre pour parvenir à l'émetteur

- . pour les trames destinées à une station située sur le même réseau local : émission normale (directe)

- . pour les trames destinées à des stations distantes :

- elles sont marquées (bit de poids fort de l'adresse d'émission à 1)

- en tête du champ de données : le chemin à suivre, c'est-à-dire :

-  Liste de couples  $\langle n^{\circ} \text{ pont}, n^{\circ} \text{ RL} \rangle$  :

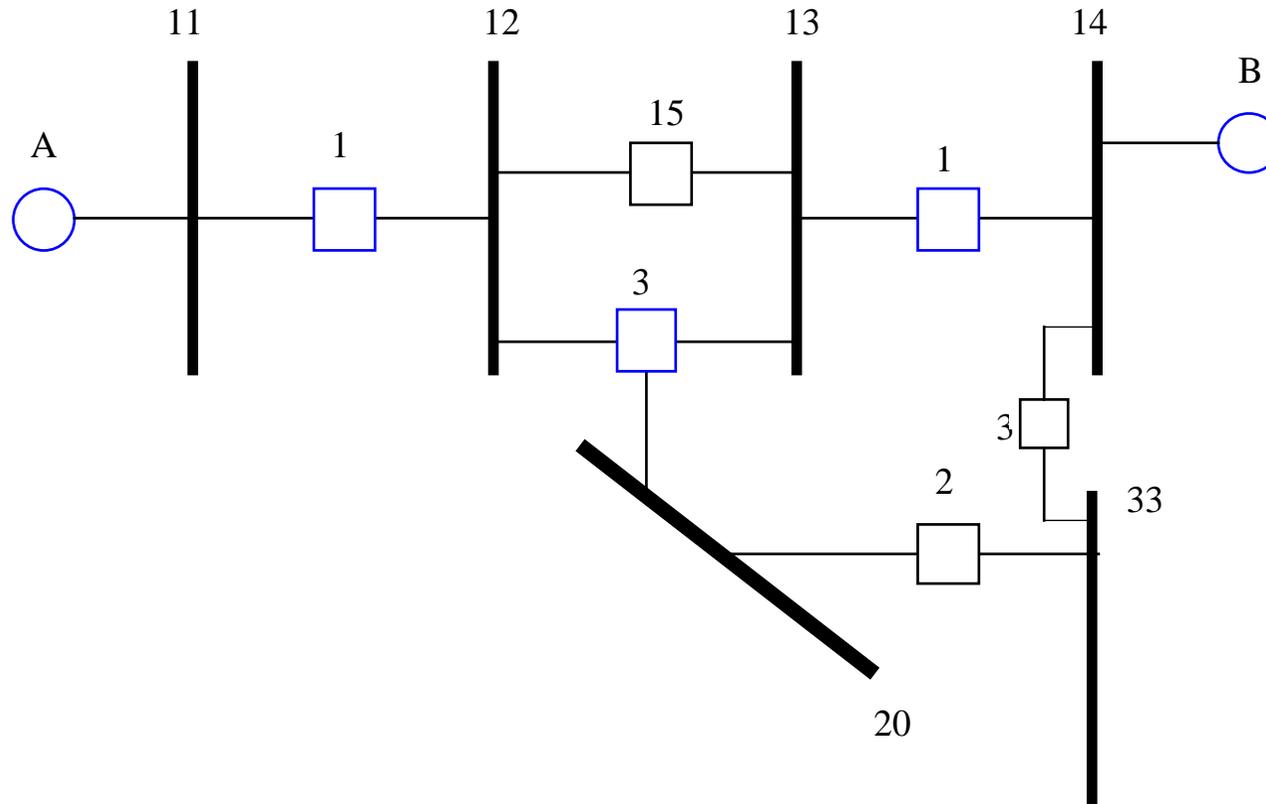
- .  $n^{\circ} \text{ RL}$  identifie de manière unique le réseau local (12 bits)

- .  $n^{\circ} \text{ pont}$  identifie le pont dans son réseau local (4 bits)

note : deux ponts situés sur des sous-réseaux différents peuvent avoir même numéro.

### 6.3. Exemple de “Source routing”

$A \Rightarrow B : (11,1; 12,3; 13,1; 14,0)$



## 6.4. L'acheminement par le "Source routing"

Les ponts ne s'intéressent qu'aux trames marquées.

Ils analysent le chemin présent dans la trame

s'il contient le n° de RL d'où provient la trame

- . si ce n° est suivi du n° du pont
- . alors il retransmet la trame vers le RL dont le n° suit
  - . s'il existe
  - . sinon le chemin est incorrect.

S'il n'existe pas de composant spécialisé, la scrutation de la marque se fait par logiciel en mode "promiscuous" !

## 6.5. Algorithme de recherche d'un chemin

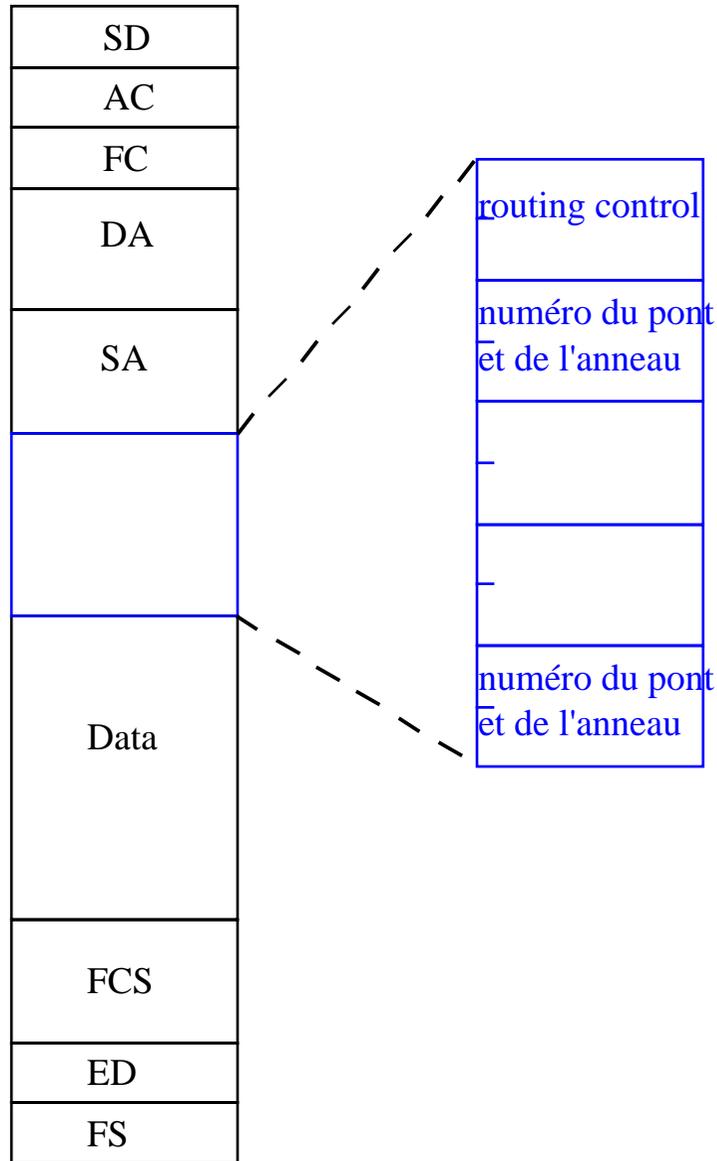
### Principe :

- . Lorsque qu'une destination est inconnue,
- . L'émetteur diffuse (partout!) une trame de recherche avec comme adresse de destination l'adresse de la station dont le chemin est recherché !
  - cette trame est munie initialement d'une liste vide.
- . Chaque pont rencontré recherche son numéro dans la trame :
  - s'il n'y figure pas, il place son identificateur et celui du réseau local d'où elle provient dans la trame,
  - sinon il détruit la trame (qui boucle).
- . Le destinataire retourne la trame de recherche à l'émetteur (**inversion du chemin**)
- . L'émetteur choisit parmi toutes les trames retournées le meilleur chemin :
  - généralement celui de la première retournée

### Optimisation :

- diffusion globale
- diffusion sur une arbre

## 6.6. Le format des trames



### Routing control (16 bits)

- Type de la trame (3 bits) :
  - . 000 = trame de données
  - . 010 = Spanning Tree explorer
  - . 100 = All path explorer
- Longueur de la liste en octets (5 bits)
- Sens de lecture de la liste (1 bit)
- Longueur maximale des trames (3 bits):
  - . code des longueurs les + courantes (MTU)

## 6.7. Conclusion

Le “source routing” est adapté au Token Ring.

La phase d'apprentissage est coûteuse :

- . par défaut nécessité une diffusion totale
- . par inondation
- . le nombre de duplications est fonction du nombre de ponts

Robuste et très précis :

- . trouve toujours le meilleur chemin

Optimisation :

- . chaque station mémorise les chemins trouvés
- . cette mémoire “cache” est consultée lors de chaque émission
- . ces données ont une durée de vie limitée

## 7. Conclusion

Equipements d'interconnexion : répéteurs, concentrateurs et ponts.  
Quelques techniques d'interconnexion : le Tunnelling.

### Source routing versus Transparent bridging

. intelligence dans les stations ou dans les routeurs

Approches complémentaires :

Si le bit de l'adresse source est positionné  $\Rightarrow$  Source routing

sinon  $\Rightarrow$  Transparent bridging

Les deux techniques utilisent la diffusion :

. adaptées aux réseaux locaux

Les deux techniques :

. ne permettent pas d'adapter la taille des trames (segmentation)

. sont peu efficaces s'il y a un trop de stations,

. ne gèrent pas les erreurs.

 [Le routage et les routeurs \(IP\)](#)