

# Enoncé du TP 1 Réseaux

## Couches physique et liaison de données — Ethernet (Spanning Tree et VLAN)

C. Pain-Barre

INFO - IUT Aix-en-Provence

version du 22/2/2013

**i** Pour ce TP (en particulier pour utiliser le simulateur), vous devez travailler sur Windows.

## Première partie

# Introduction

## 1 Vitesse de transmission : débit et délai

Le **débit**  $D$  d'une ligne de transmission détermine le temps  $Te(M, D)$  qu'il faut à un équipement pour émettre sur la ligne (ou en extraire) une certaine quantité d'information de  $M$  bits :

$$Te(M, D) = M/D \text{ secondes}$$

Par exemple, il faut 0,1 seconde (100 ms) à une carte émettrice pour "injecter" (émettre) 1 Mbit sur une liaison de débit 10 Mbit/s. De même, dès que le premier bit parvient au récepteur, sa carte mettra 0,1 seconde pour les recevoir tous.

**i** Rappels :

- pour les débits en transmission, on utilise le système international avec des puissances de 10 :  
1 kbit/s =  $10^3$  bit/s ; 1 Mbit/s =  $10^3$  kbit/s ; etc.
- pour les quantités de données en informatique, on utilise des puissances de 2 :  
1 octet = 8 bits ; 1 Ko =  $2^{10}$  octets ; 1 Mo =  $2^{10}$  Ko ; etc.  
Notons que pour être conforme à la nouvelle normalisation des préfixes binaires (de 1998), il faudrait remplacer dans la ligne précédente Ko et Mo par Kio et Mio...

## Exercice 1 (Temps d'émission selon le débit)

Calculer le temps d'émission d'un message de 100 Mo sur une liaison à 100 Mbit/s.

[\[Corrigé\]](#)

Pour parvenir au récepteur, le premier bit doit parcourir la distance qui le sépare de l'émetteur. Le temps nécessaire à ce parcours s'appelle le **délai**. Il dépend du support utilisé (câbles, fibres, air, ...) et de la nature de l'onde qui s'y propage (lumière, électricité, ...). Ce temps peut être négligeable sur de courtes distances selon le support (avec une vitesse de propagation qui peut atteindre 200 000 km/s, soit 2/3 de la vitesse de la lumière). Dans le cas d'une transmission entre la Terre et un satellite en orbite géostationnaire à 36 000 km, il est de l'ordre de 0,15 secondes.

Le temps total nécessaire à la transmission d'un message entre deux équipements sur une liaison est donc la somme du temps d'émission du message (ou de son extraction en réception mais pas les deux) et du délai d'acheminement d'un bit.

## Exercice 2 (Comparaison du débit et du délai)

*Cet exercice néglige bien des aspects de la communication entre ordinateurs mais en dépit de sa bizarrerie, il explique pourquoi nombre d'entreprises font convoyer leurs sauvegardes par des coursiers/transporteurs plutôt que par le réseau. D'autre part, il illustre la différence entre le délai et le débit. Il est proposé dans le livre "Réseaux. Cours et Exercices" d'Andrew Tanenbaum.*

Supposons qu'un Saint-Bernard, équipé d'une boîte de 3 cartouches magnétiques de 7 Go chacune à la place d'un tonnelet de rhum (sacrilège !), soit entraîné à effectuer l'aller-retour entre deux points quelconques. Celui-ci va à la vitesse de 18 km/h. Jusqu'à quelle distance le chien possède-t-il une plus grande vitesse de transmission qu'une liaison (d'un réseau) ATM à 155 Mbit/s ? On néglige ici le temps de propagation des bits sur le réseau ATM (quelques microsecondes). Seul compte le débit. Pour le chien, c'est l'inverse. On ne tient compte que du délai de propagation (du chien), et on néglige le temps qu'il faut pour l'équiper de la boîte de cartouches.

[\[Corrigé\]](#)

## Deuxième partie

# Couche Physique

## 2 Rappels

Rappels sur les logarithmes :

- $\log_b(x) = \frac{\log(x)}{\log(b)}$
- $x = b^{\log_b(x)}$
- $y = \log_b(x) \iff b^y = x$

Rappels sur les débits :

- Le débit binaire d'une ligne de transmission est :

$$D = R \times \log_2(V) = R \times v$$

où  $R$  est la rapidité de modulation ;  $V$  est le nombre d'états (symboles) possibles du signal ;  $v = \log_2(V)$  est la valence ;

- (Nyquist, 1924) Débit maximal théorique d'une ligne de transmission :

$$D_{max} = 2 \times H \times \log_2(V) = 2 \times H \times v$$

où  $H$  est la largeur de la bande passante (en Hz) ;

- On en déduit que la rapidité de modulation maximale d'une ligne est le double de sa bande passante, i.e. :

$$R_{max} = 2 \times H$$

- (Shannon, 1948) Débit maximal sur une ligne bruitée :

$$C_{max} = H \times \log_2(1 + S/N)$$

où  $S/N$  est le rapport signal bruit, généralement exprimé en décibel (dB). Pour un rapport de  $x$  dB, on a :  $x = 10 \times \log_{10}(S/N)$  et donc  $S/N = 10^{x/10}$

### Exercice 3 (Débit maximal d'une ligne de transmission)

1. On dispose d'un câble possédant une plage de fréquences utilisable de [ 10 kHz, 20 kHz ] et l'on souhaite transmettre l'information par une modulation quelconque. Quel doit être le rapport signal/bruit minimal de l'environnement (c.-à-d. sa qualité) afin d'atteindre le débit de 100 kbit/s ?
2. Quel est le débit binaire d'un modem ayant une rapidité de modulation de 2 400 baud et utilisant une modulation de 4 phases et 4 amplitudes ?

[\[Corrigé\]](#)

## 3 Codages

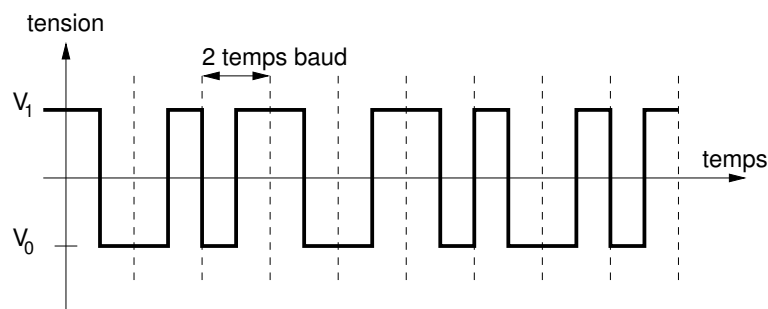
### 3.1 Codages Manchester

On rappelle que le codage Manchester transmet le signal d'horloge en même temps que l'information binaire. La valence de ce signal est 0,5 car il faut 2 bauds pour transmettre un bit ; le signal changeant toujours de polarité au milieu du temps bit. Ce codage nécessite ainsi le double de bande que le codage NRZ pour le même débit.

Pour le Manchester (normal), un signal partant de  $V_1$  au début du temps bit et finissant en  $V_0$  correspond à un bit à 1, s'il part de  $V_0$  pour terminer en  $V_1$ , il correspond à un bit à 0.

### Exercice 4

Soit le signal suivant, où la fin d'un bit est marquée par une barre verticale hachée :



1. En supposant qu'il s'agit d'un codage Manchester (normal), quelle séquence de bits représente-t-il ?
2. En supposant que la ligne admet une rapidité de modulation maximale de 8 000 bauds. Quel est le débit binaire maximal obtenu avec le codage Manchester ?
3. Supposons que l'on dispose d'un câble en paire torsadée possédant une plage de fréquences utilisables de [10 kHz, 25 kHz]. Est-il adapté pour fournir un débit de 20 kbit/s si on utilise le codage Manchester ?

[\[Corrigé\]](#)

### 3.2 Codage 4B/5B

Le codage 4B/5B vise à empêcher de longues suites de 0 ou de 1. En émission, tout groupe de 4 bits est remplacé par un groupe de 5 bits d'après une table de correspondance. En réception, la traduction inverse des groupes de 5 bits restitue les bits d'origine. La table de correspondance de ce codage est la suivante :

4B	5B	4B	5B
0000	11110	1000	10010
0001	01001	1001	10011
0010	10100	1010	10110
0011	10101	1011	10111
0100	01010	1100	11010
0101	01011	1101	11011
0110	01110	1110	11100
0111	01111	1111	11101

#### Exercice 5

1. Quelle séquence de bits sera émise effectivement pour transmettre la séquence 0000000111111110 ?
2. Une carte reçoit la séquence 01110110011001010101. Que peut-elle en déduire ?

[\[Corrigé\]](#)

### 3.3 Embrouillage

L'embrouillage vise aussi empêcher de longues suites de 0 ou de 1 mais sans surcharger la transmission. Cependant, c'est une méthode stochastique (aléatoire) dont le succès n'est pas assuré. L'émetteur et le récepteur sont d'accord sur une suite de nombres pseudo-aléatoires. Prenons au hasard (sur un octet) la suite : 173 (10101101), 150 (10010110), 77 (01001101), etc.

Quand l'émetteur commence sa transmission, il effectue un XOR entre le  $i^{\text{e}}$  octet à transmettre et le  $i^{\text{e}}$  octet de la suite pseudo-aléatoire. Le récepteur effectuera la même opération pour retrouver les bits d'origine : XOR entre le  $i^{\text{e}}$  octet reçu et le  $i^{\text{e}}$  octet de la suite.

Rappel : Table du XOR :

XOR	0	1
0	0	1
1	1	0

#### Exercice 6

1. Quelle séquence de bits sera émise effectivement pour transmettre la séquence 0000000010101010 ?
2. Refaire les mêmes XOR avec ce que vous avez obtenu à la question précédente. Que constatez-vous ?

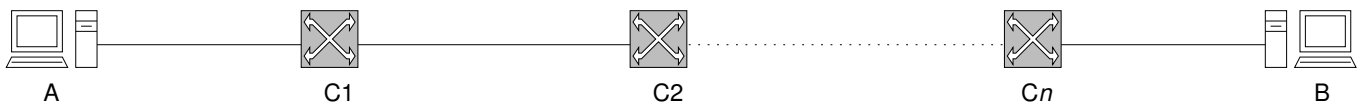
[\[Corrigé\]](#)

## Troisième partie

# Couche Liaison de données

### 3.4 Comparaisons des commutations de message et de paquet

En introduction du TD, nous avons vu le temps nécessaire à la transmission d'un message sur une liaison. Pour une communication entre un émetteur et un récepteur, mettant en jeu une succession de liaisons et nœuds de transfert, ce temps est augmenté par les temps de commutation/routage sur ces nœuds. Si les nœuds fonctionnent en mode *store and forward*, ils doivent recevoir la totalité du message avant de le réémettre sur une sortie. Il faut alors ajouter le temps de réémission du message sur les liaisons par les nœuds. Ainsi, si deux hôtes A et B sont reliés en réseau par  $n + 1$  liaisons et  $n$  commutateurs :



alors le temps total de transmission d'un message de  $M$  bits de A à B est (au minimum) :

$$Te_{A-B}(M) = \sum_{i=1}^{n+1} (Te(M, D_i) + Tp_i) + \sum_{i=1}^n Tc_i$$

où  $Te(M, D_i)$  est le temps d'émission de  $M$  bits sur la  $i^e$  liaison dont le débit est  $D_i$  ;  $Tp_i$  est le temps de propagation du signal sur la liaison  $i$  ;  $Tc_i$  est le temps de commutation du message sur le  $i^e$  commutateur. C'est un temps minimum qui suppose que les commutateurs ne sont concernés que par cette unique transmission, et peuvent donc retransmettre immédiatement après avoir commuté le message/paquet/trame.

Si A envoie à B plusieurs messages en rafale, ils se suivront sur les liaisons et un commutateur peut recevoir un message en entrée tout en émettant le précédent sur une sortie. Si toutes les liaisons ont le même débit (ou qu'il est croissant sur le chemin), alors aucun message n'a besoin d'être placé en attente sur un commutateur. En revanche, si le débit est irrégulier ou décroissant sur le chemin, des commutateurs devront mettre en attente des messages car ils ne pourront les réémettre aussi rapidement qu'ils les reçoivent. En supposant que tous les messages ont la même taille, et en négligeant le temps de propagation, alors le temps total d'émission de  $k$  messages en rafale de A à B est :

$$Te_{A-B}(k \times M) = \sum_{i=1}^{n+1} Te(M, D_i) + \sum_{i=1}^n Tc_i + (k - 1) \times \max_{1 \leq i \leq n+1} Te(M, D_i)$$

Son calcul est basé sur le dernier message : les deux premiers membres de l'addition sont son temps de transmission/traversée sur le réseau. La dernière partie de l'addition est le temps total d'attente de ce message avant émission sur les liaisons. C'est le temps d'émission des messages précédents sur la liaison de plus bas débit...

#### Exercice 7

Supposons que deux hôtes A et B sont placés en réseau et sont séparés par 3 lignes de transmission et 2 commutateurs C1 et C2 ainsi que le montre le schéma ci-dessous :



1. En supposant que les 3 liaisons proposent un débit de 10 000 bit/s chacune, et que le temps de commutation est de 100 ms sur chaque commutateur, calculer le temps de total d'envoi d'un message de 20 000 bits de A à B, dans le cas de la commutation par message et dans le cas de la commutation par paquets de 1 000 bits, où chaque paquet peut être commuté dès qu'il est reçu. On néglige le temps de propagation du signal : un bit émis est supposé immédiatement reçu.

2. Même question mais en supposant que :

- la liaison  $A \longleftrightarrow C1$  a un débit de 5 000 bit/s ;
- celle  $C1 \longleftrightarrow C2$  a un débit de 10 000 bit/s ;
- celle  $C2 \longleftrightarrow B$  a un débit de 20 000 bit/s.

[\[Corrigé\]](#)

## 4 Exploitation d'une liaison half-duplex

### Exercice 8

Soit A et B, deux stations reliées par une ligne de transmission offrant un débit de 1 Mbit/s exploitée en half-duplex. Le temps de propagation du signal sur la liaison est 10 ms. Comparer la quantité d'information que peuvent s'échanger A et B en une seconde en transmettant des paquets de 10 kbit dans chacun des cas suivants :

1. A et B transmettent à tour de rôle chacun un paquet en alternance
2. A transmet d'abord ses paquets puis laisse la liaison à B pour que celui-ci puisse commencer à émettre les siens à la demi-seconde. À la fin de la seconde, A et B doivent avoir transmis le même nombre de paquets.

[\[Corrigé\]](#)

## 5 Transmission de trames et synchronisation

Dans ces exercices, on va encapsuler (et décapsuler) des messages pour les transmettre dans des trames, en les encadrant par des délimiteurs.

On rappelle que l'intérêt des délimiteurs est de permettre au récepteur de se synchroniser s'il a perdu la synchronisation à cause d'une erreur de transmission ou de son propre fait.

### 5.1 Délimiteurs et caractères de transparence

Dans cette partie, une trame est composée de caractères. Une trame doit obligatoirement commencer par les caractères délimiteurs de début **DLE** et **STX**, et se terminer par la séquence **DLE ETX** (délimiteurs de fin).

#### Encapsulation pour l'émission

Pour encapsuler le message en l'encadrant par les délimiteurs afin d'émettre une trame, le message doit être traité pour enlever toute ambiguïté entre le contenu du message et les caractères délimiteurs : devant tout caractère **DLE** du message, il faut insérer un **DLE** supplémentaire (caractère de transparence).

 On utilisera le caractère '**&**' pour **DLE**, '**<**' pour **STX** et '**>**' pour **ETX**.

## Décapsulation en réception

L'extraction de trames (et de messages) à partir d'un flot de caractères reçus est une opération relativement simple lorsqu'il n'y a pas eu d'erreur de transmission et que le récepteur reste synchronisé : il lui suffit de rechercher les délimiteurs marquant le début de la trame en ignorant les caractères situés avant. Les caractères qui suivent sont traités pour extraire le message. Il faut supprimer les caractères de transparence et rechercher les délimiteurs de fin : si un **DLE** est suivi d'un autre **DLE** alors ce dernier est éliminé sinon ce doit être **ETX** (ou alors, une erreur est survenue).

### Exercice 9

1. Encapsuler le message suivant en vue de sa transmission (les guillemets ne sont pas à émettre...) :  
"&<A\_&\_TCHAO&<&!&>!&&>"
2. Quel(s) message(s) une station peut-elle extraire de la séquence de caractères suivante, reçue pendant un laps de temps ?  
"&<&&OLA&&&>YEP&>&<HOP&&&&<&>"

[\[Corrigé\]](#)

## 5.2 Fanions et bits de transparence

Dans cette partie, une trame est composée de bits. Elle doit obligatoirement commencer et se terminer par le fanion **01111110**.

### Encapsulation pour l'émission

Pour l'encapsuler en l'encadrant par le **fanion 01111110** afin d'émettre une trame, le message doit être traité pour enlever toute ambiguïté entre le contenu du message et les fanions : à la suite d'une série de 5 bits à 1 dans le message, il faut insérer un bit à 0 (bit de transparence).

### Décapsulation en réception

L'extraction de trames (et de messages) à partir d'un flot de bits reçus est elle aussi une opération relativement simple lorsqu'il n'y a pas eu d'erreur de transmission et que le récepteur reste synchronisé : il lui suffit de rechercher le fanion marquant le début de la trame en ignorant les bits situés avant. Les bits qui suivent sont traités pour extraire le message. Il faut supprimer les bits de transparence et rechercher le fanion de fin : si un 0 est suivi de cinq 1 consécutifs suivis d'un 0, celui-ci est un bit de transparence (à éliminer). Si les 1 sont suivis d'un autre 1, le bit qui suit devrait être un '0' (ou alors, une erreur est survenue).

### Exercice 10

1. Encapsuler le message suivant en vue de sa transmission (les guillemets ne sont pas à émettre...) :  
"101111111111101"
2. Quel(s) message(s) une station peut-elle extraire de la séquence de bits suivante, reçue pendant un laps de temps ?  
"01110111111011011111001111110001111111011111101101111101111101111110"

[\[Corrigé\]](#)

## Quatrième partie

# Simulateur : travail sur Ethernet et les vlans

## 6 Introduction

Le simulateur de réseaux que nous allons utiliser au cours de ce TP est accessible dans le dossier *Simulateur* sur le *Bureau* de votre poste de travail. Il a été développé par Pierre Loisel et est disponible gratuitement sur le [site du réseau CERTA](#) (une version plus aboutie mais payante existe aussi).

Il permet de créer un réseau et de simuler son comportement au niveau **Physique**, **Liaison de données**, **IP**, et **Transport**. On peut sauvegarder le réseau réalisé sous la forme d'un fichier XML, et le charger ultérieurement. Il permet aussi d'exporter l'image représentant le réseau.

Vous pouvez consulter la [documentation sur le simulateur](#) (disponible aussi sur le *Bureau* de votre poste de travail).

### 6.1 Création/mise à jour du réseau

Le simulateur permet de modifier un réseau du niveau physique jusqu'au transport. Chaque niveau correspond à l'un des 4 modes suivants (accessibles depuis le menu *Mode*) :

- **Conception réseau** : (raccourci F2) permet d'ajouter des stations, câbles, hubs, switchs, Internet. Dans ce mode, on peut modifier le nombre de cartes réseaux des stations, leur ajouter une carte d'accès distant (modem), modifier le type d'un câble, modifier le nombre de ports des hubs, switchs, choisir le mode de fonctionnement d'un switch (vlan, store and forward, ...), etc. C'est donc principalement un mode qui s'occupe du matériel ;
- **Ethernet** : (F3) permet d'émettre une trame à partir d'une carte à destination d'une autre carte (ou en *broadcast*), éteindre un matériel, etc.
- **IP** : (F4) permet de configurer les matériels au niveau IP, notamment les adresses IP et les tables de routage, activer le routage sur du matériel possédant plusieurs cartes et simuler l'utilisation de la commande **ping**. Permet aussi d'observer l'émission et le traitement de requêtes/réponses ARP, datagrammes IP, messages ICMP relatifs à **ping** ;
- **Transport** : (F5) permet d'envoyer des messages. Dans ce mode, on peut aussi faire du NAT/PAT et établir des règles de filtrage (fonction *firewall*).

### 6.2 Déroulement des simulations

Les modes *Ethernet*, *IP* et *Transport* proposent des simulations correspondant à leur niveau. La simulation peut être *manuelle*, *pas à pas* ou *automatique*. Le mode *Ethernet* propose aussi *trame réelle*, et les modes IP et Transport proposent *pas de démonstration* :

- **simulation manuelle** : cette simulation requiert de l'utilisateur qu'il choisisse lui-même les (bonnes) étapes qui permettent d'accomplir la simulation. Ces étapes sont diverses : consultation du cache ARP, émission d'une requête ARP, consultation de la table de routage, etc. Pour chaque étape, il y a un bouton *stop* qui demande d'annuler le mode manuel pour cette station et la simulation continue comme si l'utilisateur avait fait les bons choix.

Certaines étapes nécessitent de remplir correctement des informations. Lorsque toutes les bonnes étapes ont été choisies (quel que soit l'ordre), l'utilisateur peut tenter d'envoyer une trame (datagramme, message) en choisissant l'étape ultime correspondante. Les étapes ultimes ont une description qui se termine



par une flèche. Si l'utilisateur a fait les bons choix et a fourni les bonnes informations, un succès est affiché.

- **pas à pas** : pour cette simulation, c'est le simulateur qui choisit lui-même l'étape à réaliser et l'affiche. L'utilisateur est invité à cliquer sur le bouton *suivant* entre chaque étape.
- **automatique** : cette simulation ne sollicite pas l'utilisateur. Les étapes sont exécutées automatiquement et peuvent être visualisées à l'écran ;
- **trame réelle** : permet un réglage plus fin au niveau Ethernet. Permet notamment de provoquer des collisions de trames ;
- **pas de démonstration** : dans ce cas, les étapes n'apparaissent pas.

## 7 Exercices



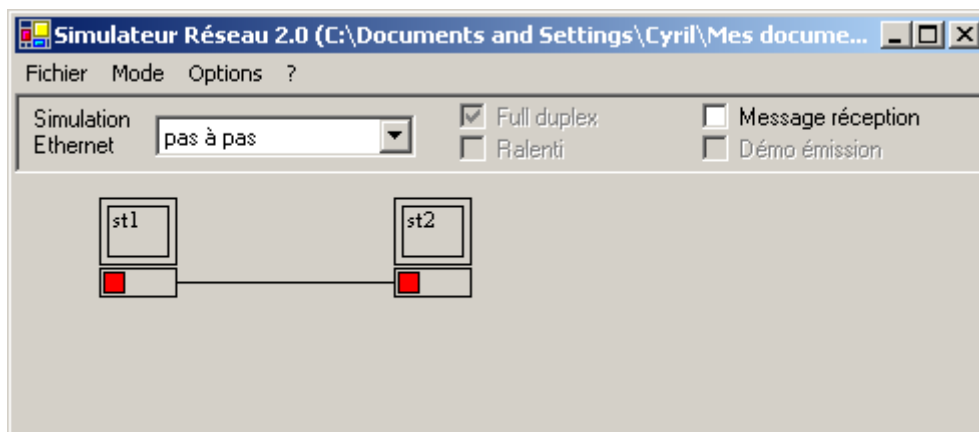
**Sauvegardez régulièrement votre configuration car le simulateur n'est pas très stable !!**

### Exercice 11 (création du réseau)

En mode *Conception réseau* (F2), créer un réseau composé de 2 ordinateurs reliés par un **câble** comme ci-dessous :



Passer en mode *Ethernet* (F3). Une carte affichée en rouge signifie qu'il y a un problème de connectique (cartes incompatibles ou mauvais câble), que nous allons régler à l'exercice suivant :



[\[Corrigé\]](#)

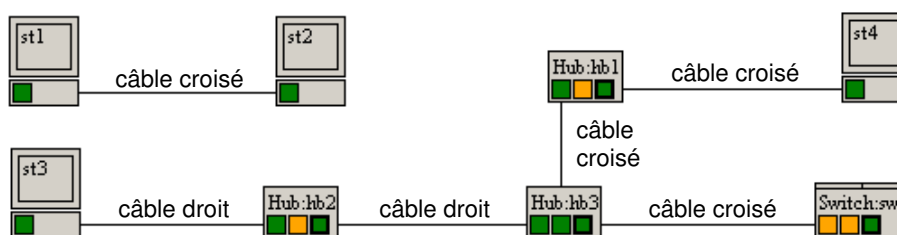
## Exercice 12 (correction du câblage)

Revenir en mode *Conception réseau* pour modifier le câble, en choisissant un câble croisé.

- ❗ On accède à un menu permettant de modifier le type de câble en effectuant un clic droit sur une des cartes qu'il relie. En effectuant un clic droit sur une station, on accède à un menu permettant de modifier son nom, son nombre de cartes réseau, et la présence d'une carte d'accès distant (modem). Le modem permettra de se connecter à (un fournisseur d'accès à) Internet via une ligne téléphonique.

✍ On rappelle qu'il existe principalement deux types d'interface (carte réseau) pour RJ-45 : MDI et MDI-X. La différence se situe sur les paires de fils utilisées en émission et en réception. Les stations et les routeurs ont généralement des interfaces MDI, alors que les Hubs et les Switchs ont généralement des interfaces (ports) MDI-X, et éventuellement un (rarement plusieurs) port MDI, dit **port uplink** (ou en cascade). Pour raccorder une interface MDI à une interface MDI-X, on utilise un câble droit. Pour raccorder deux interfaces de même type, on utilise un câble croisé.

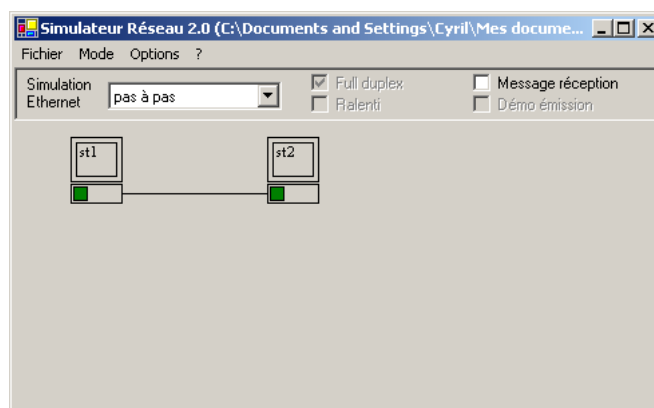
Le schéma ci-dessous présente quelques câblages valides (les ports uplink figurant en gras), où l'on peut interchanger les hubs et les switchs :



Les 2 raccordements les plus à droite n'ont généralement aucune raison d'être utilisés.

Notons que certaines interfaces sont commutables MDI/MDI-X via un bouton. D'autres sont Auto MDI-X, c'est à dire qu'elles basculent automatiquement en MDI ou en MDI-X selon ce qui est nécessaire au bon fonctionnement de la liaison.

Passer en mode *Ethernet* (F3). Cette fois, les cartes réseaux doivent être affichées en vert, signifiant que la liaison est correcte :



[Corrigé]

### Exercice 13 (émission d'une trame)

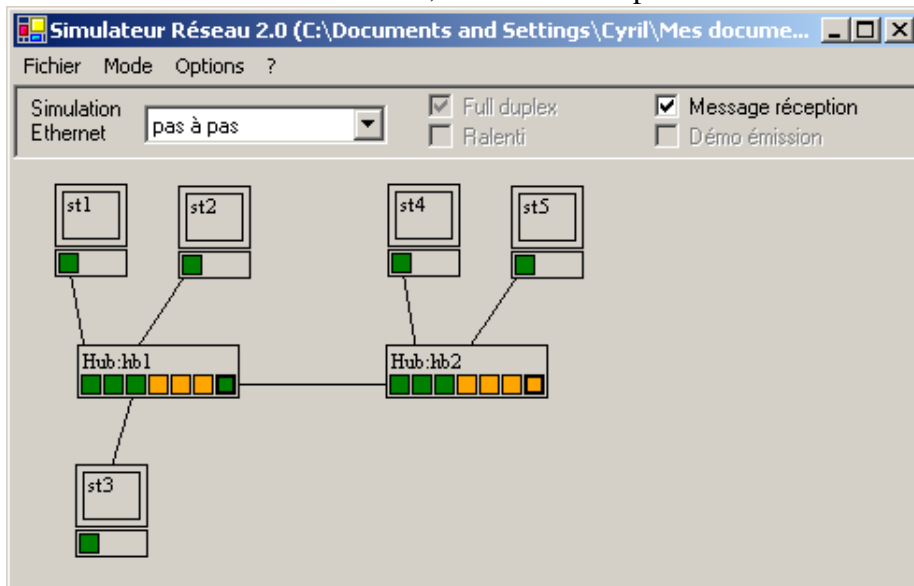
- ❗ Par défaut le simulateur fonctionne en mode pas à pas, le laisser dans ce mode afin de bien voir toutes les étapes des échanges de messages qu'il est conseillé de lire tout au long du TP.  
Cocher aussi *Message réception* de façon à avoir une popup lorsqu'un message arrive correctement à destination.

Émettre une trame en **unicast** depuis st1 vers st2. L'émission d'une trame se demande en faisant un clic droit sur la carte émettrice. Pour une émission en unicast (destinataire unique), il faut cliquer sur la carte du destinataire.

[\[Corrigé\]](#)

### Exercice 14 (extension du réseau)

Avec un réseau de 2 PCs nos tests seraient limités, on va l'étoffer pour obtenir :



Pour cela, en mode *Conception réseau* :

- supprimer le câble entre st1 et st2 (il suffit de cliquer sur une carte et de cliquer sur une région vide)
- ajouter 2 hubs et 3 PCs
- connecter st1, st2 et st3 au hub hb1 en utilisant un câble droit
- connecter st4 et st5 à hb2
- connecter le **port uplink** (la carte en gras) de hb1 à un port normal de hb2 en utilisant un câble droit

- ❗ On rappelle que lorsqu'une trame arrive sur un port d'un hub, celui-ci la retransmet sur tous ses ports (sauf celui d'où vient la trame) sans s'occuper de la destination de la trame. De plus, un hub ne fonctionne pas en mode *store and forward* et commence la retransmission de la trame dès qu'elle lui parvient, sans attendre d'en recevoir la fin.

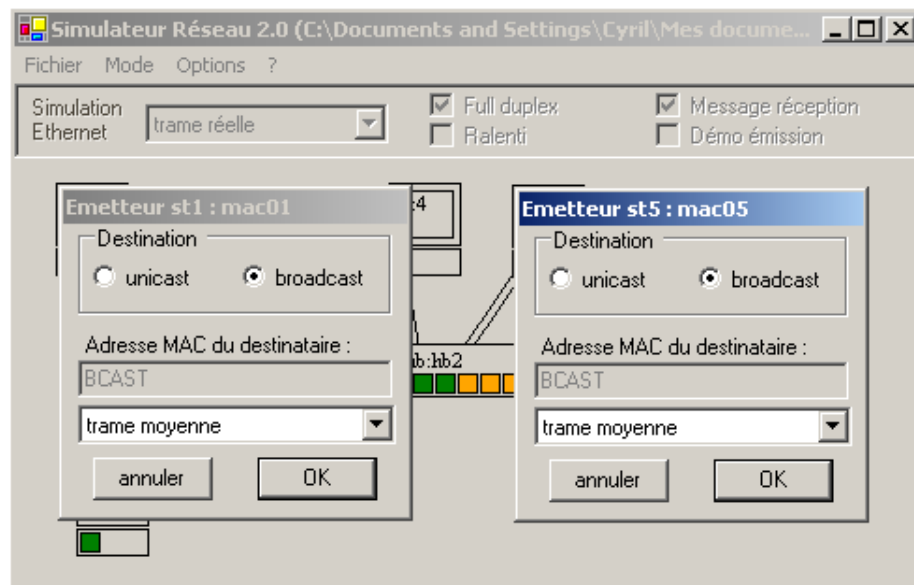
Passer en mode *Ethernet*. Aucune carte ne doit être rouge. Tester les connexions en envoyant :

- une trame en broadcast depuis st1
- une trame en unicast depuis st1 à destination de st2

[\[Corrigé\]](#)

### Exercice 15 (provocation d'une collision)

Passer en simulation trame réelle pour simuler une collision, puis effectuer un clic droit sur la carte réseau de st1 pour demander l'émission d'une trame en broadcast, mais avant de cliquer sur Ok, effectuer un clic droit sur la carte de st5 pour demander l'émission d'une trame en broadcast :



Cliquer sur Ok pour les deux émissions, et observer se qui se passe.

[\[Corrigé\]](#)

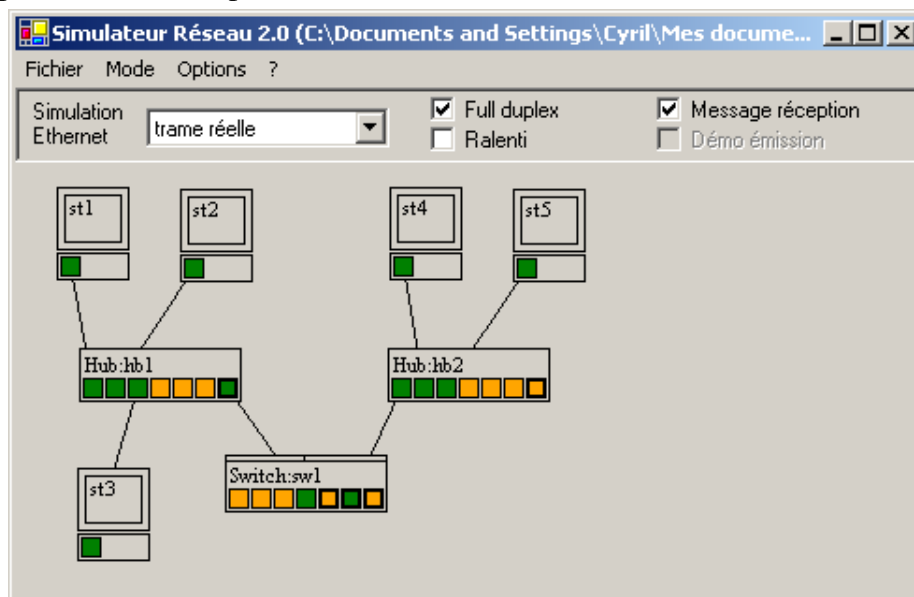
### Exercice 16 (boucle sur le réseau)

Ajouter un câble droit entre le port uplink de hb2 et un port normal de hb1. Que se passe-t-il si vous voulez émettre une trame ?

[\[Corrigé\]](#)

### Exercice 17 (ajout d'un switch)

Supprimer les connexions entre hb1 et hb2, ajouter un switch, et relier le port uplink de hb1 à un port normal de sw1, et un port uplink de sw1 à un port normal de hb2 :



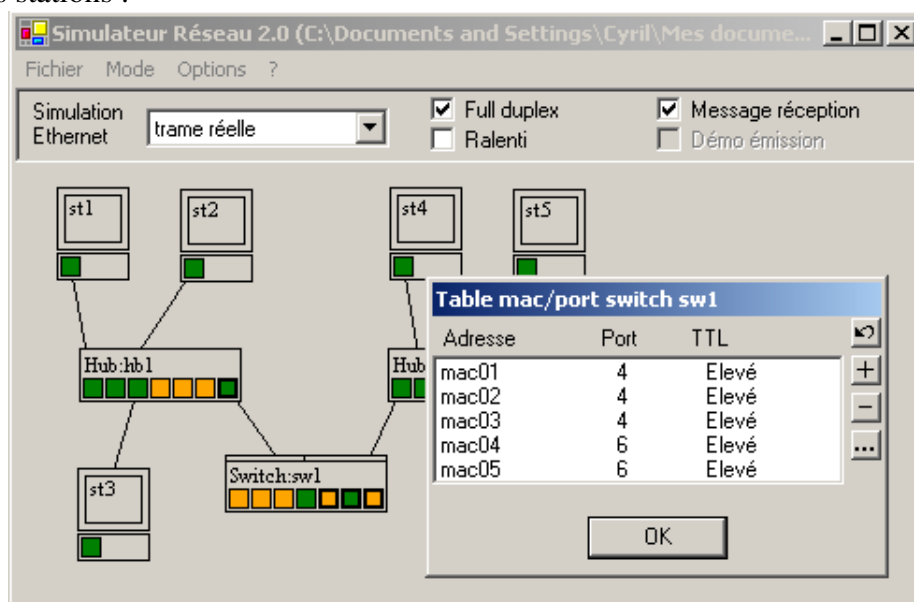
- ❗ On rappelle que les switchs font de l'**auto-apprentissage** : lorsqu'une trame arrive sur un port d'un switch, celui-ci examine son adresse source et enregistre dans sa table mac/port que l'hôte (interface) possédant cette adresse est accessible via ce port. Lorsqu'il recevra une trame à destination de cette adresse, elle ne sera retransmise que sur ce port. S'il ne connaît pas "l'emplacement" de la destination, il opérera comme un hub en retransmettra la trame sur tous ses ports (sauf celui d'où vient la trame). Une autre différence importante entre un hub et un switch est qu'un switch fonctionne en général en mode *store and forward* : il attend de recevoir la totalité de la trame avant de la retransmettre, et se charge de gérer les collisions éventuelles qu'elle subira. Pour terminer, notons qu'un switch ne garde pas indéfiniment les associations (*mac, port*) : chaque association établie a une durée de vie (TTL), par exemple de 300 secondes, et sera retirée à l'expiration du TTL si aucune trame ayant pour adresse source *mac* n'est entre-temps reçue via ce *port*.

[Corrigé]

### Exercice 18 (réinitialisation de la table mac/port)

- ❗ Quand on ajoute un switch dans le simulateur il connaît déjà toutes les machines du réseau : sa table mac/port est automatiquement renseignée. Dans la vraie vie, elle est construite au fur et à mesure.

Effectuer un clic droit sur sw1 pour éditer la table mac/port de sw1, et observer que celui-ci connaît l'emplacement de toutes les stations :



Cliquer sur ok puis vider la table mac/port de sw1. Vérifier qu'elle est vide en l'éditant à nouveau.

[Corrigé]

### Exercice 19 (comportement d'un switch pour une destination inconnue)

Selon vous, si st1 veut envoyer une trame vers st2, la trame sera-t-elle retransmise par sw1 vers hb2 ?

Envoyer une trame depuis st1 vers st2 et observer le comportement de sw1.

[Corrigé]

**Exercice 20 (apprentissage du réseau)**

Selon vous, la table mac/port de sw1 a-t-elle changé ?

L'éditer pour voir son contenu. Envoyer une trame depuis st2 vers st1, puis de nouveau depuis st1 vers st2. Observer le comportement du switch.


[\[Corrigé\]](#)

**Exercice 21 (tentative de collision)**

Émettre de nouveau 2 trames simultanées en broadcast depuis st1 et st5. Cette fois, il n'y a pas de collision.

- i** Dans la réalité, une collision peut toujours se produire au niveau des hubs... D'ailleurs, en s'y prenant bien, on parvient à provoquer une collision en demandant d'un côté l'émission d'une trame courte et de l'autre l'émission d'une trame longue, et en n'envoyant effectivement la trame longue que lorsque la trame courte est arrivée sur le switch.

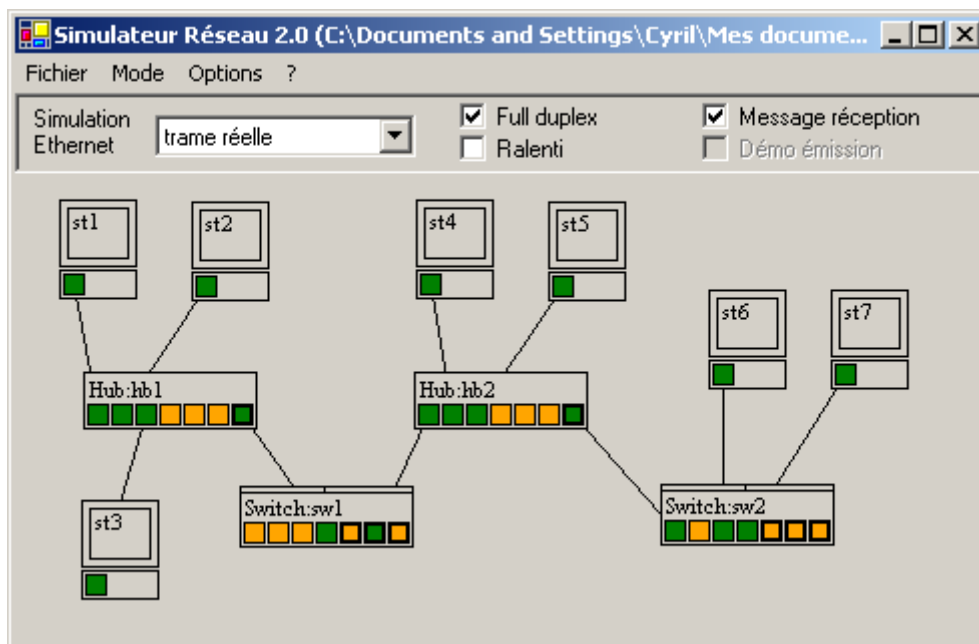
Essayer à nouveau de provoquer une collision.

-  On pourra constater que dans ce simulateur, les switches ne gèrent pas les collisions alors qu'ils le doivent...

[\[Corrigé\]](#)

**Exercice 22 (extension du réseau)**

Ajouter un switch et 2 stations comme sur le schéma ci-dessous :

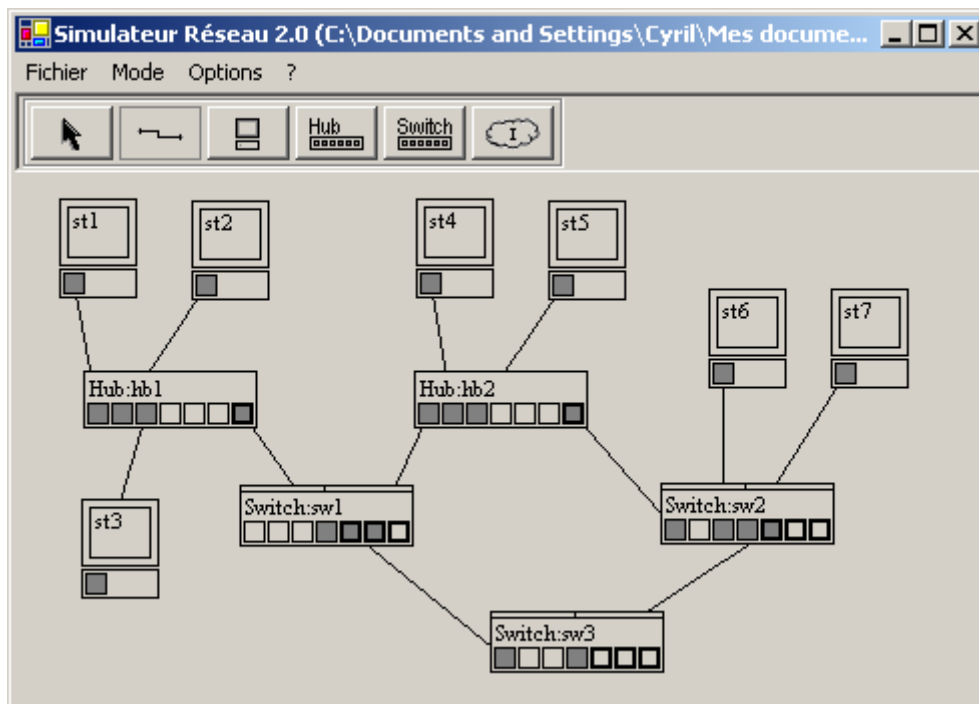


Consulter sa table mac/port.

[\[Corrigé\]](#)

### Exercice 23 (boucle et spanning tree)

Ajouter un 3<sup>e</sup> switch connecté aux 2 switches déjà présents comme sur le schéma ci-dessous :



Cela crée une boucle dans le réseau. Selon vous, celle-ci pose-t-elle un problème, sachant que ces switches gèrent par défaut le spanning tree ?

[\[Corrigé\]](#)

### Exercice 24 (émission de trames)

Envoyer une trame en broadcast depuis st1 puis envoyer une trame depuis st1 vers st4.

[\[Corrigé\]](#)

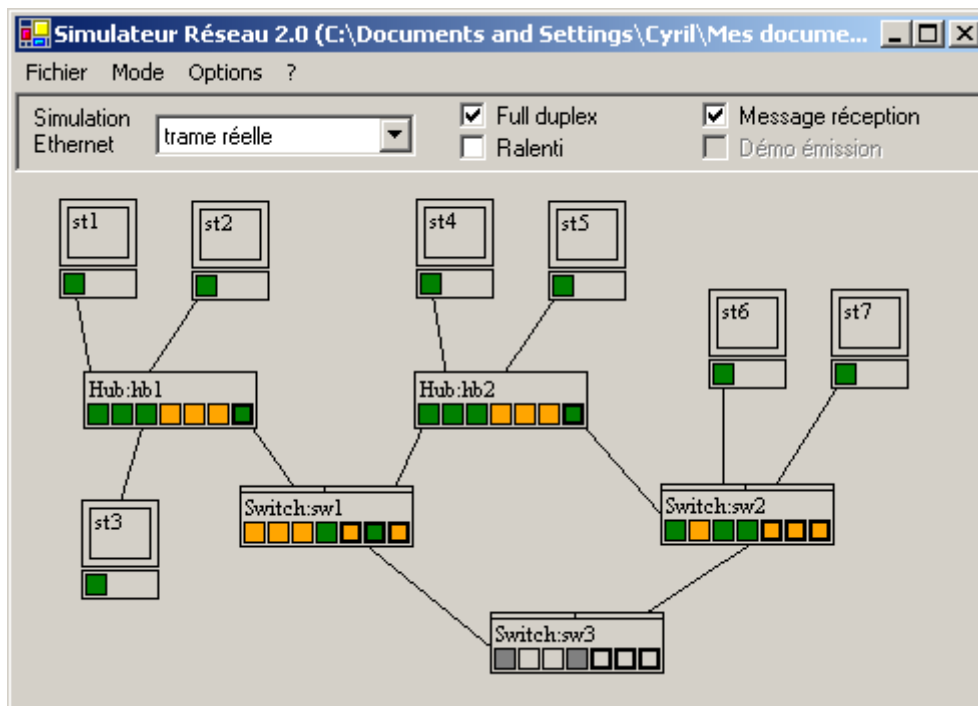
### Exercice 25 (désactivation du spanning tree)

En mode *Conception réseau*, effectuer un clic droit sur chaque switch pour supprimer la gestion du spanning tree. Essayer d'envoyer une trame depuis st1 vers st4.

[\[Corrigé\]](#)

## Exercice 26 (extinction d'un switch)

En mode *Conception réseau*, remettre la gestion du spanning tree sur les switches. En mode *Ethernet*, éteindre sw3 :



Envoyer une trame depuis st1 vers st4. Que se passe-t-il ? Consulter la table mac/port de sw1 pour comprendre ce qui s'est passé.

[\[Corrigé\]](#)

## Exercice 27 (découverte d'une route)

Envoyer une trame depuis st4 vers st1. Elle devrait arriver (mais passer par une route différente de celle qui aurait été prise si sw3 était allumé).

Selon vous, est-ce que l'émission d'une trame de st1 vers st4 échouera encore ? Essayer.

- i** De manière générale, il suffit de cliquer sur *Découvrir le réseau* sur tous les switches pour que ceux-ci connaissent l'emplacement de toutes les stations. On n'utilisera pas cette facilité pour le moment.

[\[Corrigé\]](#)

## Exercice 28 (remise en service de sw3)

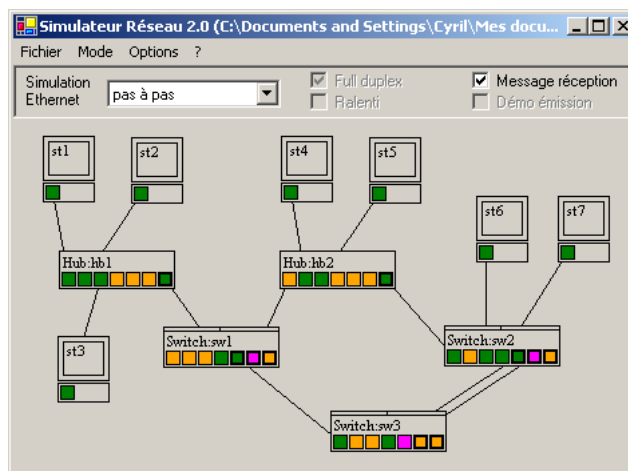
Rallumer sw3 et envoyer encore une fois une trame depuis st1 vers st4.

[\[Corrigé\]](#)



**Exercice 29 (redondance de liaison)**

Ajouter un second câble entre sw2 et sw3 :



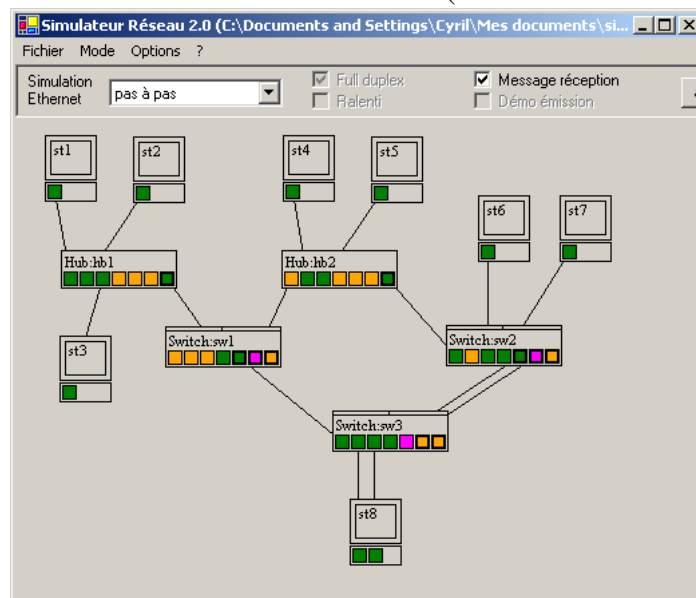
puis envoyer une trame en broadcast depuis st1.

**i** Dans ce simulateur, si on repasse en mode *Conception réseau*, alors les tables des switches sont remises à jour, comme si on avait demandé une découverte générale du réseau...

[Corrigé]

**Exercice 30 (ajout d'une station)**

Ajouter une station (st8) connectée à sw3 avec deux câbles (st8 doit donc avoir deux cartes réseau) :

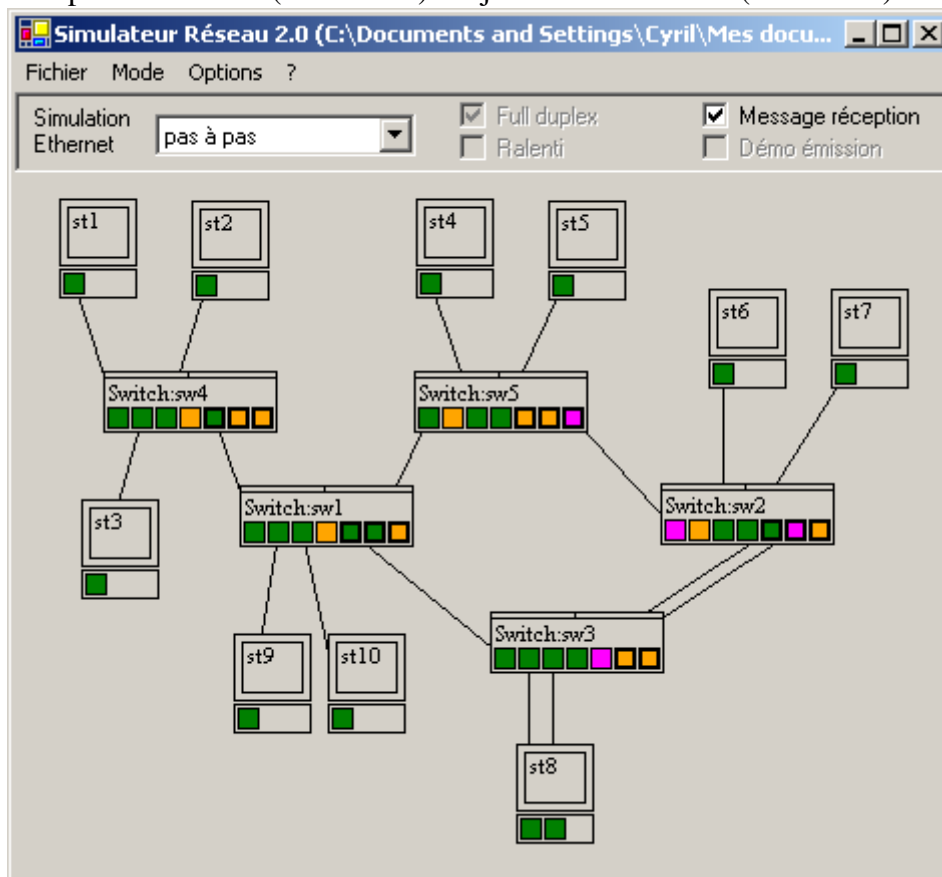


[Corrigé]

## Exercice 31 (remplacement des hubs et ajout de stations)

**i** Les hubs sont du matériel en voie de disparition...

Remplacer les hubs par des switches (sw4 et sw5) et ajouter deux stations (st9 et st10) sur sw1 :



[\[Corrigé\]](#)

## 7.1 Utilisation des vlans de niveau 1

**i** On rappelle que lorsqu'un switch gère les vlans de niveau 1, on peut placer ses ports dans différents vlans (portant des numéros). Lorsqu'une trame arrive sur un port d'un switch appartenant à un certain vlan, celle-ci ne sera éventuellement retransmise que sur des ports appartenant au même vlan.

## Exercice 32 (gestion des vlans sur sw4)

En mode *Conception réseau*, ajouter la gestion des vlan de niveau 1 sur sw4. En mode *Ethernet*, éditer la table port/vlan de sw4 pour :

- placer st1 et st3 dans le vlan 2
- placer st2 dans le vlan 3
- laisser les autres ports dans le vlan 1

[\[Corrigé\]](#)

### Exercice 33 (circulation de trames dans les vlans)

Selon vous :

1. si st1 émet une trame en broadcast, quelles sont les stations qui la recevront ?
2. si st2 émet une trame en broadcast, quelles sont les stations qui la recevront ?
3. si st1 envoie une trame à st2, celle-ci la recevra-t-elle ?
4. si st1 envoie une trame à st3, celle-ci la recevra-t-elle ?
5. si st4 émet une trame en broadcast, quelles sont les stations qui la recevront ?

Après avoir répondu à ces questions, vérifier en émettant les différentes trames.

[\[Corrigé\]](#)

### Exercice 34 (modification des vlans de sw4)

Modifier la table port/vlan de sw4 afin de placer le port qui le relie à sw1 dans le vlan 3.



**Ne pas activer la gestion des vlans dans sw1.**

Selon vous :

1. si st1 émet une trame en broadcast, quelles sont les stations qui la recevront ?
2. si st2 émet une trame en broadcast, quelles sont les stations qui la recevront ?
3. si st4 émet une trame en broadcast, quelles sont les stations qui la recevront ?

Après avoir répondu à ces questions, vérifier en émettant les différentes trames.

[\[Corrigé\]](#)

### Exercice 35 (gestion des vlans sur sw1)

Ajouter la gestion des vlans de niveau 1 sur sw1 et garder la configuration par défaut (tous ses ports dans le vlan 1). Selon vous :

1. si st1 émet une trame en broadcast, quelles sont les stations qui la recevront ?
2. si st2 émet une trame en broadcast, quelles sont les stations qui la recevront ?
3. si st4 émet une trame en broadcast, quelles sont les stations qui la recevront ?

Après avoir répondu à ces questions, vérifier en émettant les différentes trames.

[\[Corrigé\]](#)

### Exercice 36 (vlans de sw1 pour st9 et st10)

Éditer la table port/vlan de sw1 et :

- placer st9 dans le vlan 2
- placer st10 dans le vlan 3
- laisser les autres ports dans le vlan par défaut (1)

Selon vous :

1. si st1 émet une trame en broadcast, quelles sont les stations qui la recevront ?

2. si st2 émet une trame en broadcast, quelles sont les stations qui la recevront ?
3. si st4 émet une trame en broadcast, quelles sont les stations qui la recevront ?

Après avoir répondu à ces questions, vérifier en émettant les différentes trames.

[\[Corrigé\]](#)

### Exercice 37 (vlans de sw1 et un port de liaison)

Toujours dans sw1, placer son port de liaison vers sw4 dans le vlan 3.



**Ne pas modifier la configuration de sw4.**

Selon vous :

1. si st1 émet une trame en broadcast, quelles sont les stations qui la recevront ?
2. si st2 émet une trame en broadcast, quelles sont les stations qui la recevront ?
3. si st4 émet une trame en broadcast, quelles sont les stations qui la recevront ?

Après avoir répondu à ces questions, vérifier en émettant les différentes trames.

[\[Corrigé\]](#)

### Exercice 38 (modification sur sw1 du vlan du port vers sw4)

Toujours dans sw1, modifier le vlan de son port de liaison vers sw4 pour le placer dans le vlan 2.



**Ne pas modifier la configuration de sw4.**

Selon vous :

1. si st1 émet une trame en broadcast, quelles sont les stations qui la recevront ?
2. si st2 émet une trame en broadcast, quelles sont les stations qui la recevront ?
3. si st4 émet une trame en broadcast, quelles sont les stations qui la recevront ?

Après avoir répondu à ces questions, vérifier en émettant les différentes trames.

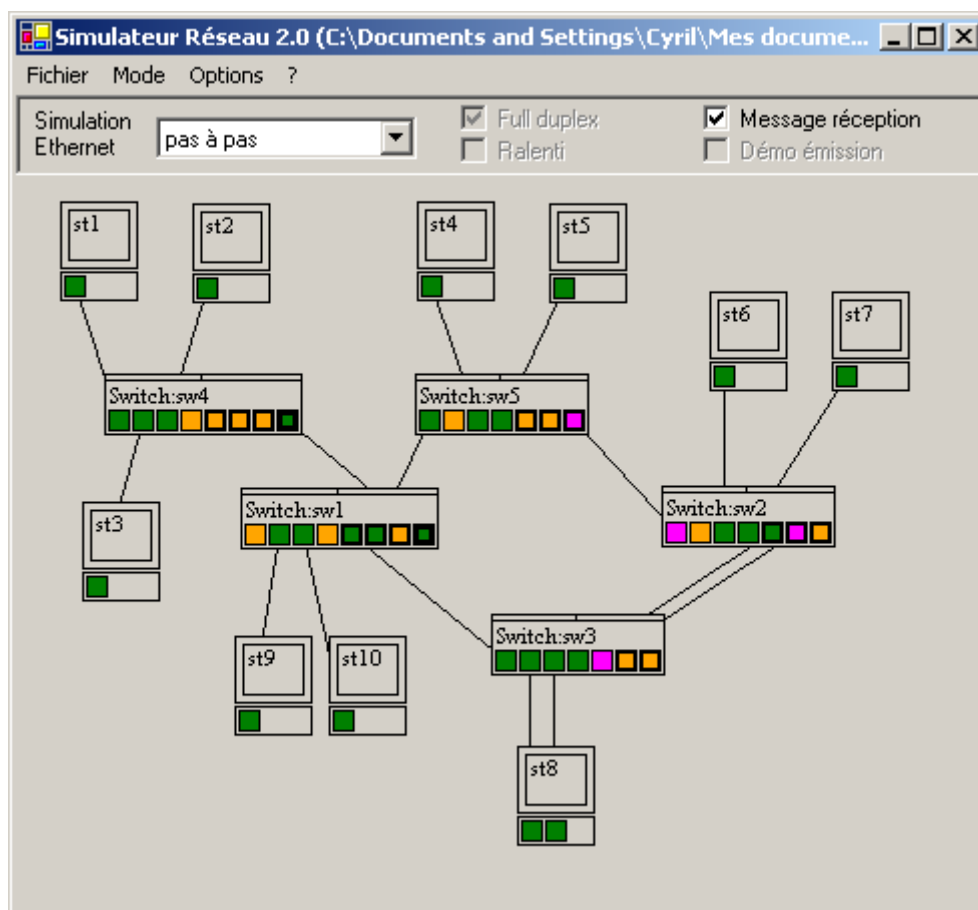
[\[Corrigé\]](#)

### Exercice 39 (utilisation de 802.1q entre sw1 et sw4)



On rappelle que le protocole 802.1q permet à deux switches de partager leurs vlans. Pour cela, lorsqu'une trame est transmise d'un switch à l'autre, celle-ci inclut le vlan concerné, et le switch récepteur ne la retransmettra éventuellement que sur des ports appartenant à ce vlan.

- Supprimer la liaison entre sw1 et sw4.
- Ajouter un port 802.1q sur les deux switches sw1 et sw4
- Utiliser ces ports pour connecter sw1 à sw4



Selon vous :

1. si st1 émet une trame en broadcast, quelles sont les stations qui la recevront ?
2. si st2 émet une trame en broadcast, quelles sont les stations qui la recevront ?
3. si st4 émet une trame en broadcast, quelles sont les stations qui la recevront ?

Après avoir répondu à ces questions, vérifier en émettant les différentes trames.

[\[Corrigé\]](#)

## 7.2 Utilisation des vlans de niveau 2

- i** Pour les vlans de niveau 2, ce n'est plus les ports qu'on associe à un vlan mais les adresses MAC des stations. Cela permet notamment de connecter une station sur n'importe quel port d'un switch tout en la laissant dans le même vlan. De plus, la station n'a pas besoin d'être connectée directement sur le switch pour qu'elle fasse partie d'un vlan géré par ce switch.

### Exercice 40 (vlan de niveau 2 dans sw4)

Sur sw4 :

- passer à la gestion de vlan de niveau 2
- éditer la table **mac/vlan** pour :
  - ◊ placer st1 et st3 dans le vlan 5
  - ◊ placer st2 dans le vlan 6

- consulter la table **port/vlan** : pouvez-vous expliquer son contenu ?

[\[Corrigé\]](#)

### Exercice 41 (apprentissage dans le vlan de niveau 2)

1. Selon vous :
  - (a) si st1 émet une trame en broadcast, quelles sont les stations qui la recevront ?
  - (b) si st2 émet une trame en broadcast, quelles sont les stations qui la recevront ?
2. Après avoir répondu aux questions précédentes :
  - (a) émettre une trame en broadcast depuis st1
  - (b) consulter la table port/vlan de sw4
  - (c) émettre une trame en broadcast depuis st2
  - (d) consulter la table port/vlan de sw4
  - (e) émettre une trame en broadcast depuis st3
  - (f) consulter la table port/vlan de sw4
  - (g) émettre une trame en broadcast depuis st1

[\[Corrigé\]](#)

### Exercice 42 (cohabitation de vlans de niveaux différents)

Sur sw1, tout en laissant une gestion des vlans au niveau 1 :

- placer st9 dans le vlan 5
- placer st10 dans le vlan 6

Selon vous :

1. si st1 émet une trame en broadcast, quelles sont les stations qui la recevront ?
2. si st2 émet une trame en broadcast, quelles sont les stations qui la recevront ?
3. si st4 émet une trame en broadcast, quelles sont les stations qui la recevront ?

Après avoir répondu à ces questions, vérifier en émettant les différentes trames.

[\[Corrigé\]](#)

### Exercice 43 (modification des vlans dans sw1)

Sur sw1 :

- placer le port connecté à sw5 dans le vlan 5
- placer le port connecté à sw3 dans le vlan 6

Selon vous :

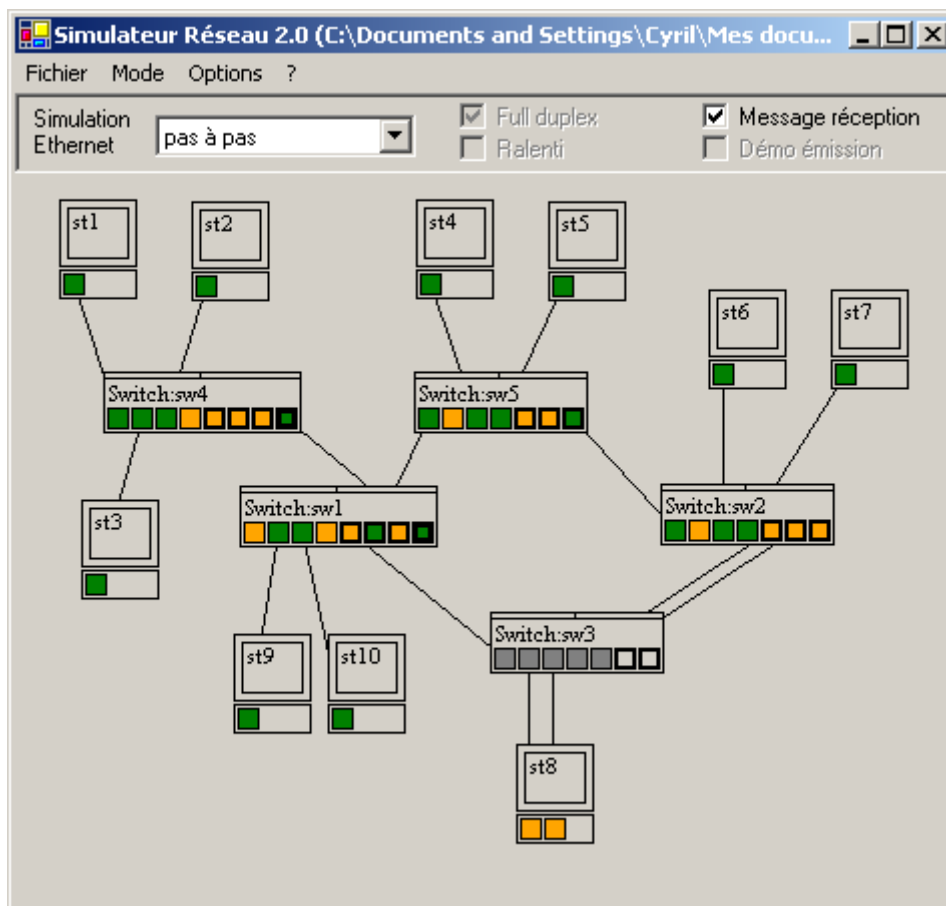
1. si st1 émet une trame en broadcast, quelles sont les stations qui la recevront ?
2. si st2 émet une trame en broadcast, quelles sont les stations qui la recevront ?
3. si st4 émet une trame en broadcast, quelles sont les stations qui la recevront ?

Après avoir répondu à ces questions, vérifier en émettant les différentes trames.

[\[Corrigé\]](#)

## Exercice 44 (extinction de sw3)

Éteindre sw3 :



Selon vous :

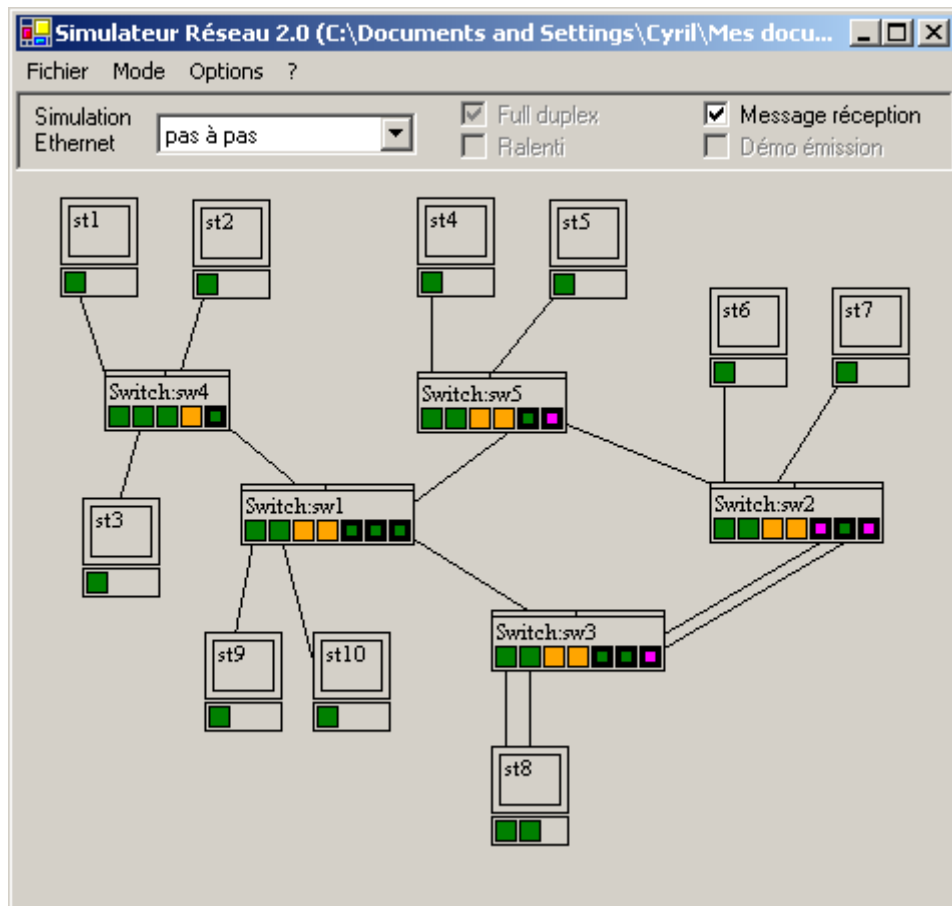
1. si st1 émet une trame en broadcast, quelles sont les stations qui la recevront ?
2. si st2 émet une trame en broadcast, quelles sont les stations qui la recevront ?
3. si st4 émet une trame en broadcast, quelles sont les stations qui la recevront ?

Après avoir répondu à ces questions, vérifier en émettant les différentes trames.

[\[Corrigé\]](#)

## Exercice 45 (gestion générale de vlan de niveau 2)

- rallumer sw3
- passer tous les switches en gestion des vlans de niveau 2
- remplacer toutes les liaisons inter-switchs par des trunks 802.1q
- placer toutes les machines impaires dans le vlan 5
- placer toutes les machines paires dans le vlan 6
- placer une carte de st8 dans le vlan5, et l'autre dans le vlan 6
- faire découvrir le réseau sur chaque switch



Tester l'émission d'une trame en broadcast depuis st1, st2, st4, st6 et st7

[\[Corrigé\]](#)

### Exercice 46 (déplacement de st1)

Débrancher st1 de sw4 et le brancher sur sw1, sans rien configurer. Selon vous, si st1 émet une trame en broadcast, quelles sont les stations qui la recevront ?

Après avoir répondu à la question, vérifier en émettant la trame.

[\[Corrigé\]](#)

### Exercice 47 (prise en compte du déplacement de st1)

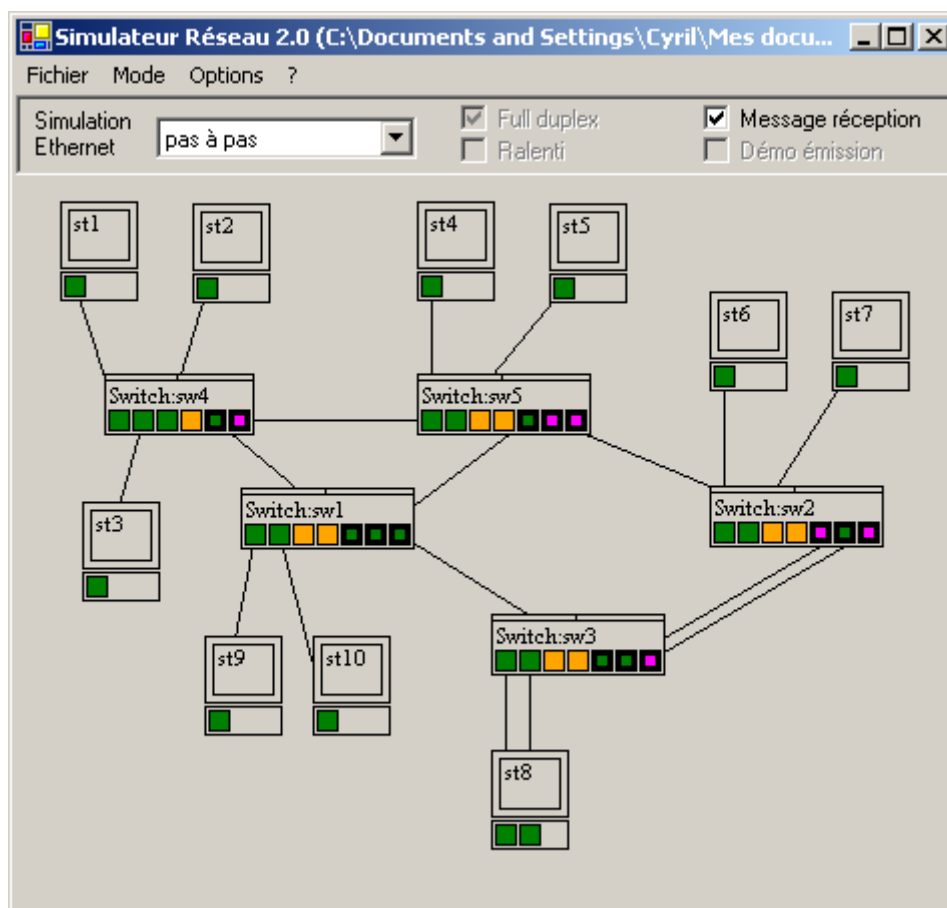
Configurer sw1 pour faire passer st1 dans le vlan 5, puis tester l'émission d'une trame en broadcast depuis st1.

[\[Corrigé\]](#)

### Exercice 48 (câblage final)

- rebrancher st1 sur sw4
- ajouter une liaison entre sw4 et sw5





[\[Corrigé\]](#)

### Exercice 49 (configuration pour la migration des stations)

Faire en sorte qu'on puisse brancher n'importe quelle machine sur n'importe quel switch tout en la laissant dans son vlan.

[\[Corrigé\]](#)

## Cinquième partie

# Ethernet : Diamètre, auto-apprentissage et spanning tree

## 8 Détection des collisions dans l'Ethernet partagé


On parle d'Ethernet partagé lorsque toutes les stations du réseau se partagent le même canal de communication. Cela comprend les topologies en bus, ou utilisant uniquement des répéteurs et/ou des hubs. Les stations doivent alors appliquer le CSMA/CD. Avant d'émettre une trame, une station doit écouter le canal pour vérifier qu'il est libre. Si deux stations émettent en même temps, il se produit une collision qui doit être détectée afin de retransmettre ces trames.

Pour pouvoir détecter les collisions, on limite la taille du réseau en fonction du temps de retournement (*Round Trip Delay*) de la trame minimum (64 octets) et du débit (1, 10, 100 ou 1 000 Mbit/s). En effet, **une station** qui

émet une trame **ne vérifie s'il y a collision que pendant l'émission de la trame**. Prenons le scénario suivant où les stations A et B sont deux stations situées aux extrémités du réseau (les stations les plus éloignées du réseau) :

1. A commence l'émission d'une trame de taille minimale (64 octets, soit 512 bits) : cette émission dure un temps  $e$
2. Les bits de la trame se propagent sur le réseau à une vitesse dépendant de la nature de la liaison physique utilisée. Soit  $t$  le temps mis par le premier bit de la trame pour arriver jusqu'à B
3. Juste avant que ce bit ne lui parvienne, B pense que la liaison est inoccupée et commence l'émission d'une trame
4. À l'instant  $t$ , B s'aperçoit qu'il y a collision, envoie un JAM, et cesse d'émettre
5. Cependant, le premier bit de la trame de B va mettre lui aussi un temps  $t$  pour parvenir à A
6. Au moment où ce bit parvient à A, celui-ci doit être encore en train d'émettre, sinon il ne s'apercevra pas qu'il y a eu collision

Ainsi, il faut que  $e \geq 2t$  pour qu'une collision soit toujours détectée par l'émetteur de la trame.

 Pour résumer, il faut que le temps d'émission de 512 bits soit supérieur au temps d'un aller-retour du signal sur le réseau (*Round Trip Delay*).

Or,  $e$  est dépendant du débit du réseau, et  $t$  est dépendant de la vitesse de propagation du signal et de la distance séparant A et B. Pour un débit ( $d$ ) et une liaison donnée (de vitesse de propagation  $v$ ), on peut donc en déduire la distance ( $D$ ) maximale devant séparer n'importe quelles stations du réseau, c'est à dire son **diamètre maximal**.

On a :  $e = 512/d$  et  $t = D/v$ , donc :


$$D \leq \frac{512 \times v}{2 \times d} \quad (\text{en mètres})$$

### Exercice 50 (calcul du diamètre maximal)

Sachant que la vitesse de propagation du signal est de l'ordre  $2 \times 10^7$  m/s sur de la paire torsadée et de  $2 \times 10^8$  m/s sur de la fibre optique, déterminez le diamètre maximal des réseaux suivants :

1. 10baseT (réseau à paire torsadée)
2. 100baseTX (réseau à paire torsadée)
3. 100baseFX (réseau à fibre optique)

[\[Corrigé\]](#)

 Afin d'augmenter le diamètre d'un réseau Ethernet partagé, il est nécessaire d'utiliser des switch ou d'augmenter la taille de la trame. Ainsi, pour appliquer CSMA/CD en présence de hubs, Ethernet 1 Gbit/s utilise une trame minimale de 512 octets, donnant un diamètre de 400 m.

## Ethernet commuté et les collisions

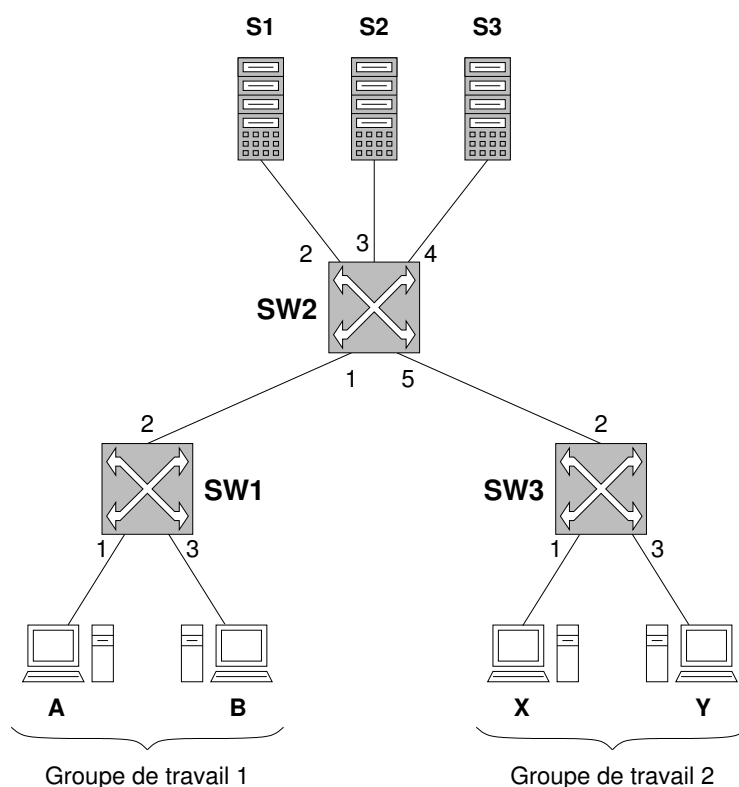
L'utilisation de commutateurs Ethernet (*switch*) permet d'augmenter le diamètre d'un réseau Ethernet parce qu'ils séparent des domaines de collision. Une fois qu'un switch reçoit une trame, il en devient responsable et doit gérer lui même les éventuelles collisions quand il la retransmettra sur une liaison. La station à l'origine de la

trame n'a qu'à se soucier de sa bonne transmission jusqu'au switch. Il n'y a plus que cette distance qui compte. Pour le switch, ce sera la distance qui le sépare de la station destinataire ou du prochain switch à traverser.

✍ Dans l'Ethernet commuté, où il n'y a pas de hubs mais uniquement des switch avec des liaisons full duplex, il n'y a pas de collision. Le diamètre du réseau peut être quelconque.

## 9 Auto-apprentissage et filtrage dans les commutateurs Ethernet

On considère le réseau Ethernet suivant :



Il est composé de :

- trois commutateurs (switchs) Ethernet SW1, SW2, et SW3 ;
- deux groupes de travail composés d'ordinateurs :
  - ◇ le groupe de travail 1 contenant notamment les ordinateurs A et B directement connectés à SW1 ;
  - ◇ le groupe de travail 2 contenant notamment les ordinateurs X et Y directement connectés à SW3 ;
- trois serveurs S1, S2 et S3 connectés à SW2.

Les numéros indiqués sur le schéma autour des commutateurs sont les numéros de leurs ports. Les stations et les serveurs ont des adresses MAC que l'on appellera mac(A), mac(B), etc. Les deux groupes de travail communiquent avec les trois serveurs, mais communiquent peu entre eux.

Ces commutateurs constituent leur table de commutation par **auto-apprentissage**.

- ❶ On rappelle que l'auto-apprentissage conduit le switch à maintenir une table de commutation qui se présente ainsi :

adresse MAC	port	durée de vie (TTL)
...	...	...

où chaque ligne est un enregistrement (*mac*, *port*, *ttl*) qui indique que le switch a appris que l'hôte (interface) d'adresse *mac* est accessible via son port *port*, et que cette information est considérée valide pendant *ttl* secondes. Le *ttl* est remis à sa valeur maximale, par exemple 300 secondes, chaque fois que le switch reçoit en entrée sur le port *port* une trame de source *mac*, qui confirme l'information.

D'autre part, ces commutateurs peuvent réaliser des opérations de filtrage :

- filtrage par liste d'accès (Access Control List) : bloquer une trame sur un port en entrée ou en sortie selon son contenu comme l'adresse source et/ou destination ;
- filtrage par vlan de niveau 1 ou 2

### Exercice 51 (Conditions de l'apprentissage)

Supposons que les tables de commutation des commutateurs soient vierges et que A vient de démarrer :

1. Comment SW1 peut-il apprendre l'existence (et l'emplacement) de A ?
2. En apprenant l'emplacement de A, qu'écrit-il dans sa table de commutation ?
3. Quelles sont les possibilités pour que SW2 et SW3 apprennent l'existence de A ?
4. En apprenant l'existence de A, inscrivent-ils quelque chose dans leur table ? Si oui, quoi ? Sinon, pourquoi ?

[\[Corrigé\]](#)

On supposera par la suite qu'une station qui démarre commence par envoyer une trame en broadcast<sup>1</sup>.

### Exercice 52 (État des tables de commutation)

1. Si les stations A, B, X et Y démarrent en même temps, quelles seront les tables des commutateurs après leur démarrage ?
2. En supposant que ces stations ont toutes contacté les trois serveurs qui leur ont répondu. Quelles sont alors les tables des commutateurs ?

[\[Corrigé\]](#)

### Exercice 53 (Durée de vie des associations)

Supposons que pendant une heure chaque groupe de travail communique très souvent (disons toutes les minutes) avec les trois serveurs, mais qu'il n'y a aucune communication directe entre les deux groupes de travail. Par ailleurs, pendant cette heure, les stations n'émettent aucune trame en diffusion. Quelles sont alors les tables de commutation des commutateurs au bout de cette heure ?

[\[Corrigé\]](#)

---

1. On verra plus tard qu'une station qui démarre commence généralement par envoyer en broadcast une requête DHCP demandant l'obtention d'une adresse IP et des informations complémentaires.

### Exercice 54 (Filtrage des communications)

1. On veut maintenant empêcher la station A de communiquer directement avec la station Y (quitte à empêcher aussi les communications de Y avec les stations et serveurs). Quelle(s) solution(s) proposez vous pour l'en empêcher ?
2. On veut maintenant isoler les deux groupes de travail en empêchant les stations d'un groupe de communiquer avec les stations de l'autre groupe. Quelle(s) solution(s) proposez-vous pour cela ?

[\[Corrigé\]](#)

### Exercice 55 (Connexions multiples)

Supposons que A ne soit pas directement connecté à SW1 mais à un hub, lui-même connecté à SW1. Sur ce hub est aussi connecté une station A' qui a démarré en même temps que A et a aussi envoyé une trame (on ignore dans cet exercice les autres stations et serveurs) :

1. Le hub possède-t-il une table de commutation ?
2. Quelle est la table de commutation de SW1 ?
3. Si A envoie une trame en unicast à A', quel est(sont) le(s) chemin(s) suivi(s) par cette trame ?

[\[Corrigé\]](#)

## 10 Spanning Tree

### Rappels

On rappelle que l'algorithme du spanning tree (802.1d) permet de supprimer les boucles en recouvrant la topologie physique par un arbre logique sans boucle (qui permet l'accès à tous les commutateurs et tous les équipements connectés). Les switches échangent des trames (BPDU) 802.3 en utilisant une adresse multicast. Chaque commutateur possède un identifiant (*Bridge ID*) composé d'une priorité (choisie par l'administrateur) et d'une adresse MAC (fixe pour un commutateur donné). Si l'administrateur ne choisit pas la priorité, une valeur par défaut identique pour tous les switches est utilisée, ce qui a comme conséquence que seule l'adresse MAC permet d'identifier le commutateur racine.

Chaque port d'un switch possède un identifiant (*Port ID*) composé d'une priorité (choisie par l'administrateur) et d'un numéro de port (fixe). De plus chaque port possède un coût de réception configuré par l'administrateur. Par défaut ce coût est de 1 000/débit en Mbps.

L'algorithme du spanning tree commence par élire un commutateur racine, ce commutateur sera celui qui a le *Bridge ID* le plus petit.

Sur chaque commutateur un port est choisi comme port racine. Ce port sera celui qui a le coût (cumulé) pour relier le commutateur racine le moins élevé. En cas d'égalité le port ayant le plus faible *Port ID* est choisi.

Sur chaque segment<sup>2</sup> un commutateur est choisi comme commutateur désigné : c'est celui qui a le plus faible coût pour atteindre le commutateur racine. En cas d'égalité le commutateur ayant le plus faible *Bridge ID* est choisi.

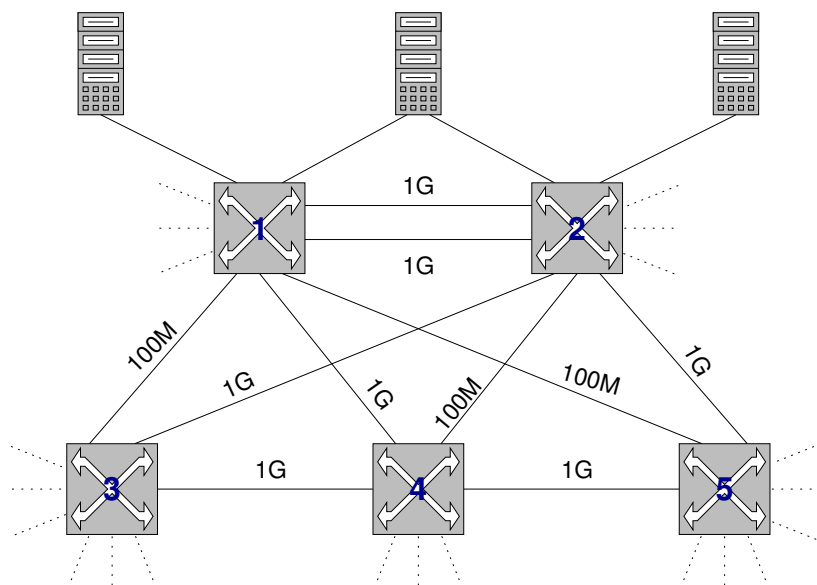
Les ports qui ne sont ni désignés, ni racines sont en standby (c'est à dire désactivés) !

---

2. Un segment est une liaison inter-switchs.

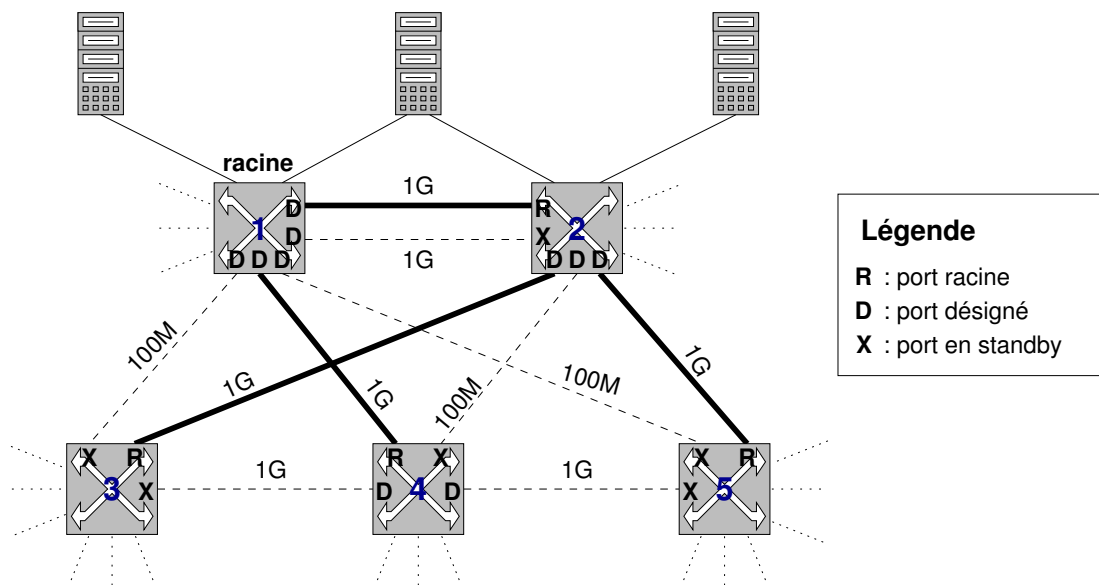
## Exemple

Soit la topologie suivante, où les numéros sur les commutateurs sont les *Bridges IDs*, les liaisons (segments) ont des débits de 1 Gbps (marquées 1G) et 100 Mbps (marquées 100M) :



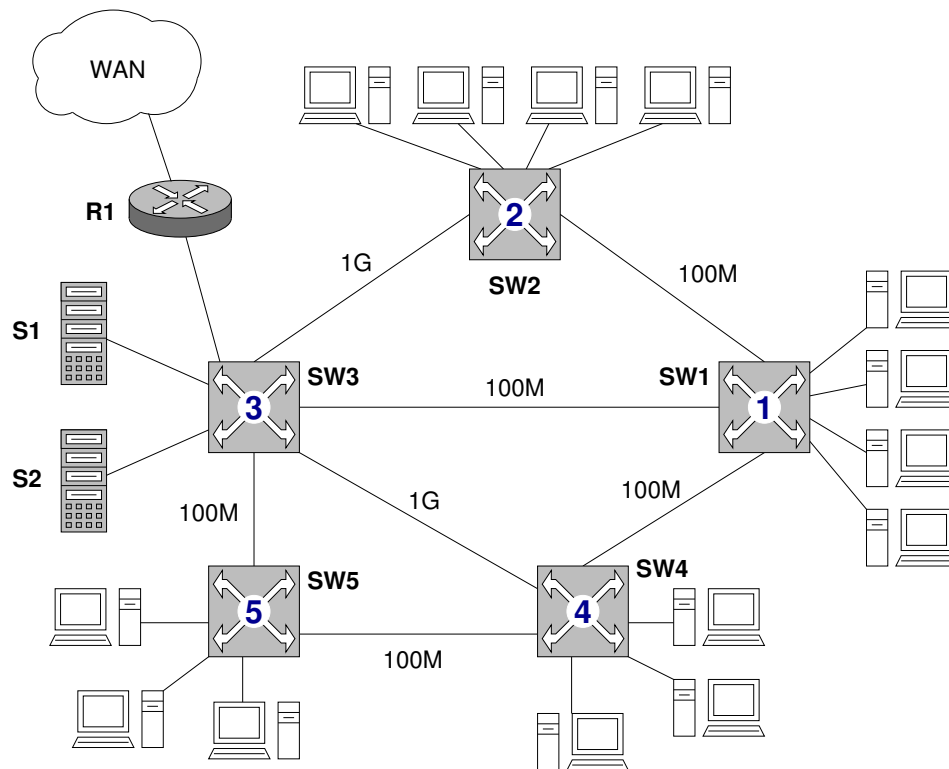
Pour ne pas surcharger le schéma, les *Ports IDs* ne sont pas indiqués. On prendra comme *Port ID* le numéro du switch qu'il relie (si l'on doit choisir entre plusieurs ports reliant le même switch au même coût, on en choisira un au hasard).

Le commutateur racine est le numéro 1. Le spanning tree est alors :



## Exercice 56 (construction d'un spanning tree optimal)

On considère le réseau Ethernet suivant, dans lequel toutes les liaisons sont des liaisons Ethernet à 100 Mbps (marquées 100M sur le schéma) ou à 1 Gbps (marquées 1G sur le schéma). Chaque commutateur a une adresse MAC qui lui sert d'identifiant. Les identifiants sont dans l'ordre 1 (le plus petit identifiant), 2, 3, 4, 5 (le nombre indiqué sur chaque commutateur) :



Les serveurs S1 et S2 sont utilisés par tous les postes clients (les postes clients sont représentés sur le schéma comme des PC. S1 et S2 sont raccordés à 1 Gbps. Les postes clients sont raccordés à 10/100 Mbps (selon les PC). R1 est un routeur permettant de connecter le LAN vers l'extérieur (notamment les autres LAN de l'entreprise et Internet). R1 est raccordé à son commutateur par une liaison 100 Mbps.

- Supposons que ces commutateurs soient configurés de façon à garder toutes les liaisons actives. Est-ce que ce réseau est utilisable en l'état ? Pourquoi ?
- Quels sont les ports concernés par le protocole STP (Spanning Tree Protocol) qu'il faut activer pour corriger le problème ?
- Sur une feuille, dessiner les switches et leurs liaisons, et indiquer les ports racine (R), les ports désignés (D) et les ports en standby (X). Mettre alors en évidence les liaisons maintenues actives et celles inactives.
- Est-ce que le résultat (la topologie active obtenue) vous paraît optimum ? Pourquoi ?
- S'il n'est pas optimum, que proposez vous de faire pour que le résultat soit optimum ? (sans changer les débits, qui sont supposés imposés).
- Quelle est alors la topologie active optimum (la dessiner) ?
- Supposons maintenant que les commutateurs SW1, SW2, SW4 et SW5 soient identiques (le même modèle du même constructeur) : ils disposent de 16 ports 10/100 et un seul port 1 Gbps. Le commutateur SW3 est un autre modèle, et dispose de 16 ports 10/100/1000. Faire un schéma indiquant la façon dont vous proposez d'utiliser ces commutateurs pour améliorer les performances de ce réseau :
  - Quel sera le commutateur racine (par défaut) dans le réseau que vous proposez ?
  - Quelle sera la topologie active (indiquer les ports racines, désignés et standby) ?
  - Y a-t-il une meilleure topologie en changeant de commutateur racine ?

[Corrigé]