

UE 503

L3 MIAGE

TD5 : Comprendre le fonctionnement d'un  
réseau

A. Belaïd

[abelaid@loria.fr](mailto:abelaid@loria.fr)

<http://www.loria.fr/~abelaid/>

Année Universitaire 2011/2012



# TD5 : Comprendre le fonctionnement d'un réseau local



- Pourquoi mettre en place un réseau local?
  - Pratique pour connecter ensemble plusieurs ordinateurs (entreprise, maison...)
  - LAN (abréviation de Local Area Network)
  - Avantages
    - Transfert de fichiers
    - Partage de ressources (partage de la connexion à internet, partage d'imprimante, disques partagés, etc.)
    - Mobilité (dans le cas d'un réseau sans fil)
    - Discussion (essentiellement lorsque les ordinateurs sont distants)
    - Jeu en réseau...



# TD5 : Comprendre le fonctionnement d'un réseau local



- Les deux types de réseaux locaux
  - Les réseaux filaires, basés sur la technologie Ethernet, représentant la quasi-totalité des réseaux locaux
    - Étant donné que les réseaux Ethernet utilisent généralement des câbles RJ45, on parle souvent de réseaux RJ45
  - Les réseaux sans fil, utilisant généralement la technologie WiFi

# TD5 : Comprendre le fonctionnement d'un réseau local



## ■ Matériel nécessaire

- Pour créer un réseau local en RJ45 sous Windows, il suffit de :
  - Plusieurs ordinateurs tournant sous Windows
  - Des cartes Ethernet, sur port PCI ou ISA (comportant une prise RJ45) ou intégrées à la carte mère
  - Des câbles RJ45, dans le cas de réseaux filaires
  - Un concentrateur, boîtier auquel il est possible de connecter les câbles RJ45 provenant des différents ordinateurs du réseau
  - ou un commutateur (switch) ou bien de façon alternative un câble croisé si l'on désire connecter uniquement deux ordinateurs



# TD5 : Comprendre le fonctionnement d'un réseau local



## ■ Architecture du réseau

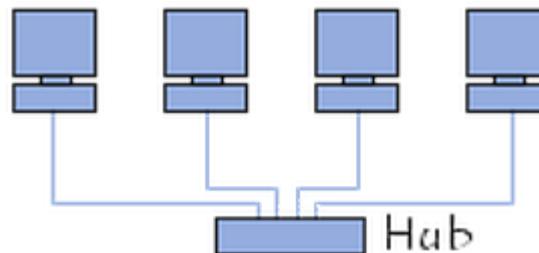
- Pour créer un réseau local en RJ45, il est recommandé d'adopter une structure dite « en étoile », dans laquelle les ordinateurs sont chacun connecté au hub (concentrateur ou collecteur) par l'intermédiaire d'un câble RJ45
- Un hub (**répéteur**) est un boîtier chargé d'acheminer les données d'un ordinateur à un autre
- Le choix du hub se fera donc en fonction du nombre d'ordinateurs connectés afin d'avoir assez de prises (appelées « ports ») sur celui-ci

# TD5 : Comprendre le fonctionnement d'un réseau local



## ■ Architecture du réseau (suite)

- Si le réseau est important, remplacer le concentrateur par un **commutateur**
- **Avantage du commutateur**
  - Ne diffuse les paquets qu'aux ordinateurs concernés, alors que le concentrateur envoie systématiquement les paquets à tous les ordinateurs connectés
  - La structure d'un tel réseau ressemble à ceci :



# TD5 : Comprendre le fonctionnement d'un réseau local

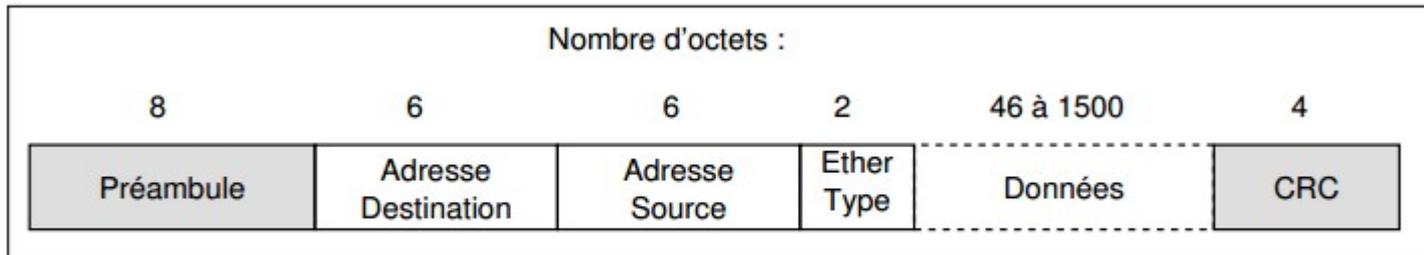


- Plus sur la commutation, centrale dans un réseau local
  - Un commutateur ne diffuse pas les trames. Il met en relation les seuls postes concernés par l'échange
  - Avant de réémettre les trames, le commutateur vérifie que le support de communication est libre
  - Un commutateur évite donc les collisions au contraire d'un concentrateur
  - A chaque fois qu'un message lui parvient, le commutateur associe le port par lequel arrive la trame à l'adresse matérielle (adresse MAC) de l'émetteur de la trame
  - Ainsi après un certain nombre de trames, le commutateur connaît « l'emplacement » (c'est à dire le port de rattachement) des postes sur le réseau et peut les mettre en relation deux à deux

# TD5 : Comprendre le fonctionnement d'un **réseau local**



- Plus sur la commutation, centrale dans un réseau local
  - Format de la trame Ethernet V2
  - Les adresses sont des adresses MAC



**Format de la trame Ethernet V2**

- Cette association adresse MAC / port est gérée dans des tables d'association présentes dans chaque commutateur et construites progressivement par apprentissage

- Plus sur la commutation, centrale dans un réseau local
  - Si une trame contient une adresse de destination qui n'est pas présente dans la table, cette trame est transmise sur tous les ports du commutateur à l'exception du port émetteur de la trame

## ■ On distingue deux modes de fonctionnement du commutateur :

### – *Store and forward* :

- il stocke les trames entièrement avant de les réémettre. Il ne réémet donc pas les trames erronées (CRC "Control Redundancy Check" innatendu) ou en collision

### – *On the fly*, à la volée :

- les commutateurs réémettent immédiatement après lecture de l'adresse MAC destinataire, c'est plus rapide mais on propage les trames erronées

## ■ Ethernet commuté

- L'actualité des architectures réseau est l'Ethernet entièrement commuté et donc la disparition progressive des concentrateurs
- En utilisant uniquement des commutateurs, il n'y a plus de collision possible
- Chaque port forme un mini-segment composé du commutateur et d'une carte ou aucune collision ne peut se produire
- Dans ce cas, pendant l'émission d'une trame, la paire de réception n'est plus monopolisée par la détection de collision et on peut recevoir en même temps, c'est à dire travailler en mode bidirectionnel (*full duplex*)



# TD5 : Comprendre le fonctionnement d'un réseau local

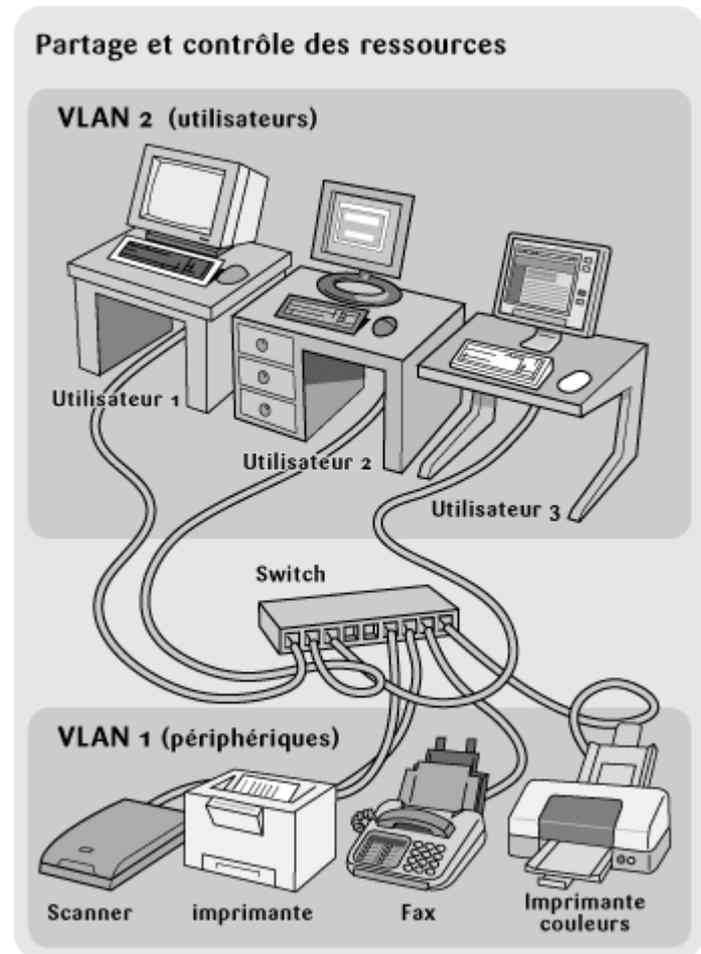


## ■ VLAN (Virtual Local Area Network)

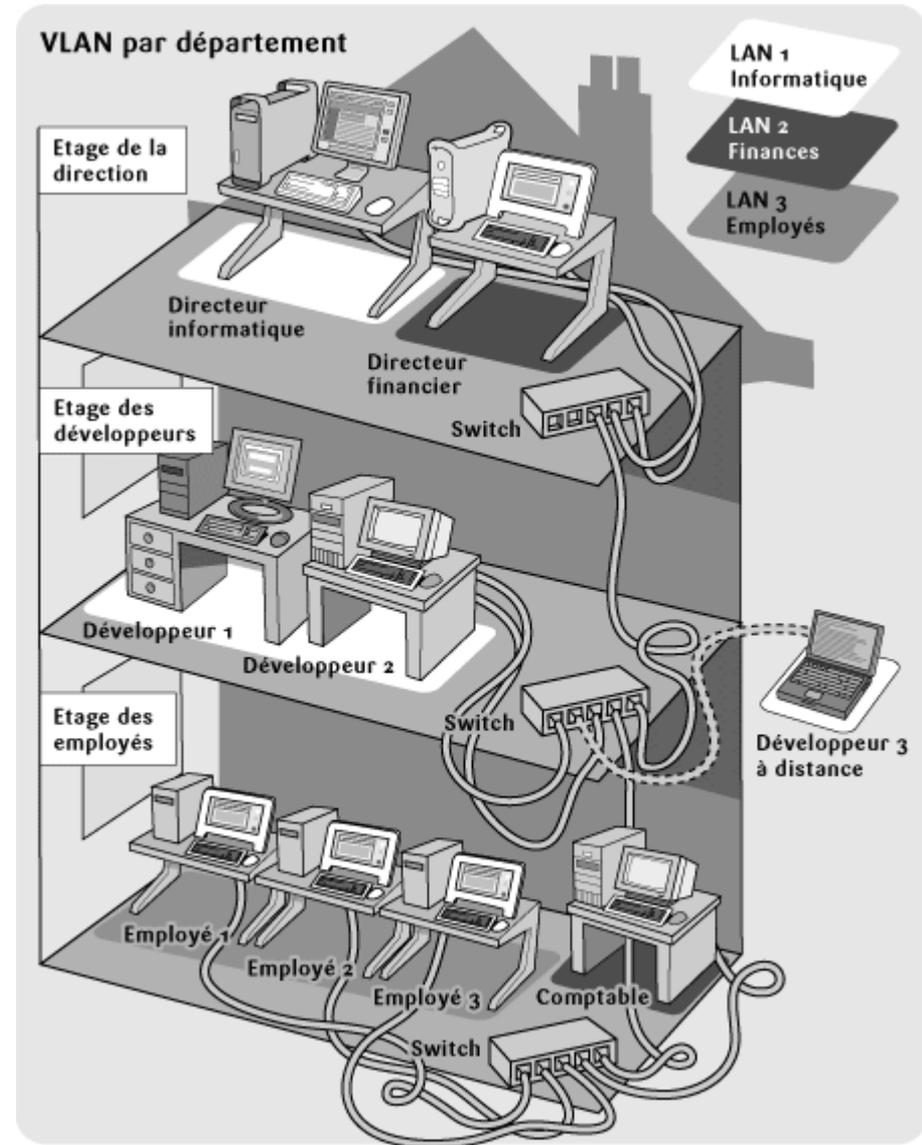
- Les réseaux Ethernet sont sujets à divers problèmes affectant les performances du réseau, à savoir :
  - Les collisions
  - La latence des équipements réseaux
  - La remise de données de type broadcast
- Afin d'augmenter les performances du réseau, on segmente le réseau en VLAN
- Le but est
  - D'obtenir une réduction de la taille des domaines de collision afin d'économiser la bande passante disponible, améliorer la sécurité, diminuer la taille des réseaux

## ■ Applications courantes du VLAN

- La création de deux VLAN regroupant d'une part les périphériques et, d'autre part, les utilisateurs permet de gérer individuellement les droits et priorités d'accès des utilisateurs
- Par exemple, l'utilisateur 3 est le seul à avoir accès à l'imprimante couleur



- La création de VLAN au sein d'un bâtiment complexe (nombreux étages, recoins etc.) permet de regrouper les utilisateurs par "centre d'intérêt" alors qu'ils sont situés à des endroits différents.
- Le VLAN 1 regroupe tous les informaticiens répartis entre deux étages ainsi qu'un travailleur à distance

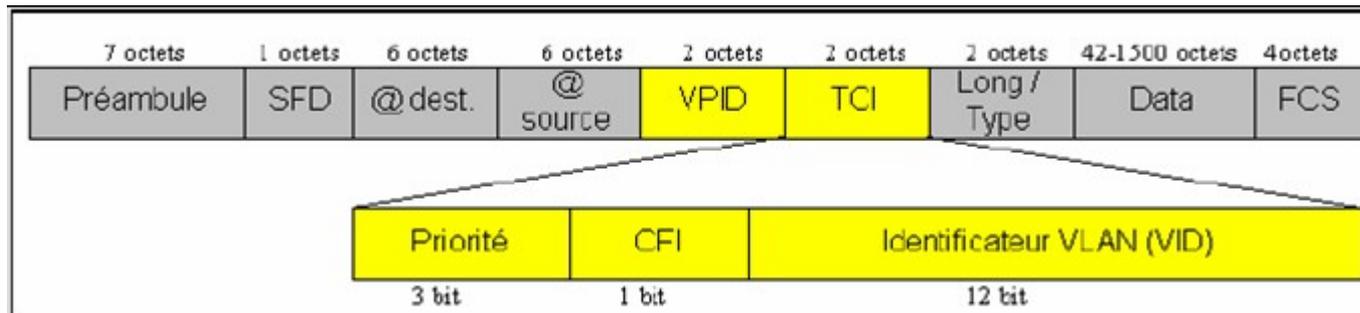


## ■ Les types de VLAN

- Les VLAN permettent de créer des domaines de diffusion gérés par des commutateurs
- Une trame ne peut être associée qu'à un VLAN et diffusée que sur les ports du commutateur associé à ce VLAN
- Il existe 3 façons principales d'associer des ports à un VLAN
  - VLAN de niveau 1 ou VLAN par port
    - Chaque port du commutateur est affecté à un VLAN, donc chaque carte réseau est affectée à un VLAN en fonction de son port de connexion
  - VLAN de niveau 2 ou VLAN d'adresses MAC
    - Chaque adresse MAC est affectée à un VLAN, donc chaque port du commutateur se voit affecter dynamiquement à un VLAN en fonction de l'adresse MAC de la carte réseau qui y est connectée
  - VLAN de niveau 3 ou VLAN d'adresse IP
    - Chaque carte réseau est affectée à un VLAN en fonction de son adresse IP, donc chaque port du commutateur se voit affecter dynamiquement à un VLAN en fonction de l'adresse IP de la carte réseau qui y est connectée

## ■ Gestion des VLAN

- Chaque VLAN peut être géré par un ou plusieurs commutateurs
- Un commutateur peut gérer plusieurs VLAN
- Les commutateurs identifient le VLAN auquel appartient une trame grâce au protocole 802.1q



- Ajout d'octet pour le transport des VLAN sur le réseau
- Permettre à deux machines d'un même Vlan de communiquer au travers un nombre non défini d'équipement réseau
- En pratique, un port de commutateur ne sera associé qu'à un seul VLAN (à l'exception des ports d'interconnexion)

## ■ Comment fonctionnent les VLAN?

- Les VLAN qui s'étendent sur plusieurs commutateurs sont catégorisés suivant deux types:
- les VLAN implicites: lorsqu'un message Ethernet passe d'un commutateur à un autre, une table indique au commutateur suivant à quel VLAN le message doit être communiqué;
- les VLAN explicites: une étiquette (tag) d'appartenance à un VLAN est apposée sur chaque message Ethernet.
- Le commutateur fait suivre les règles indiquées par le standard et fait en sorte de distribuer les messages aux portes réseaux adéquates (point de connexion d'une station sur le réseau). Cela reste valable même si des stations sont connectées sur des portes réseaux situées dans un autre commutateur
- C'est grâce à l'étiquette apposée sur les messages Ethernet ainsi qu'à l'ensemble des règles que l'administrateur a défini que le commutateur est capable de déterminer où le message doit être envoyé, diffusé ou bien encore si le message doit être filtré

# TD5 : Comprendre le fonctionnement d'un réseau local



## ■ Objectifs

- Ce TD prend appui sur un « simulateur réseau », réalisé par le CERTA, disponible à l'adresse
  - <http://www.reseaucerta.org/outils/simulateur>
- A travers plusieurs fiches, nous allons étudier de près :
  - Le rôle du concentrateur dans l'interconnexion de postes de travail via leurs cartes réseau
  - Le rôle du commutateur
  - La création de VLAN



# Simulateur CERTA

<http://www.reseaucerta.org/outils/simulateur/>



## ■ Téléchargement :

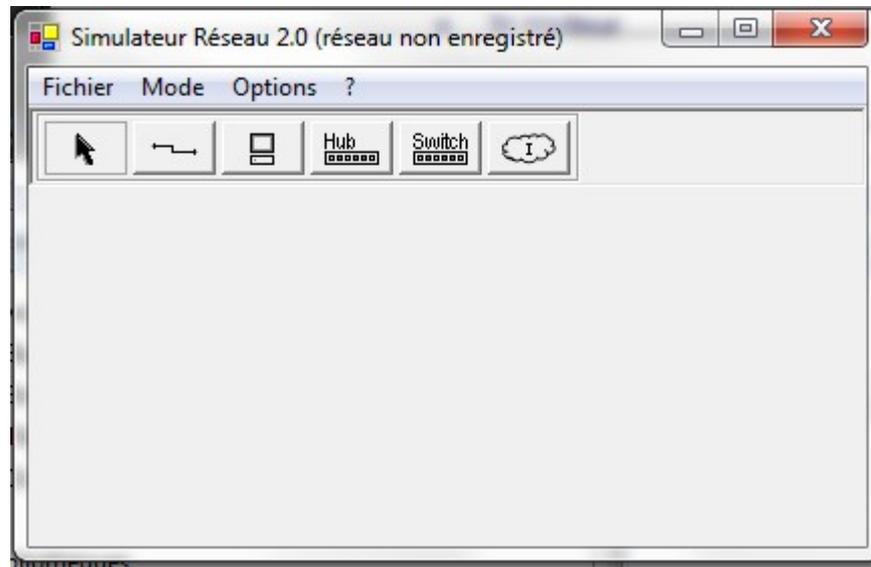
- Nécessité de télécharger une version du framework Microsoft dotNet version 1.1 ou supérieure
  - <http://www.reseaucerta.org/outils/simulateur/>
- et le simulateur :
  - Interface :
    - <http://www.reseaucerta.org/docs/outils/simulateur.zip>
  - Documentation :
    - <http://www.reseaucerta.org/docs/outils/simulateur.pdf>

## ■ Lancer le simulateur

– Cette interface s'affiche, elle permet de

- Créer un réseau composé de stations de travail, de hubs, de switchs et de câbles
- Simuler le fonctionnement de ce réseau au niveau

➤ **Ethernet, IP et transport**

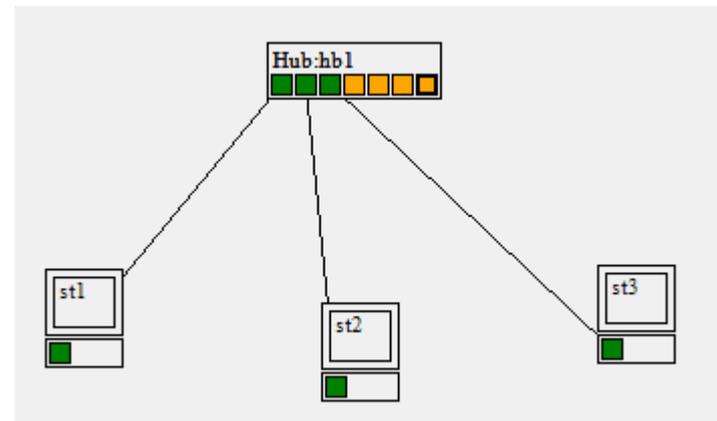


# Tâche 0

Prise de contact avec le simulateur

## ■ A faire

- Utiliser les icônes pour créer un réseau local personnel
  - Créer 3 stations
  - Créer un concentrateur (répéteur ou hub)
    - les petits carrés sont ceux de la carte réseau
  - Créer des liens en cliquant dans les carrés et joindre les carrés des cartes réseaux
    - de st1 vers Hub et de Hub vers st2 et st3
  - Passer en mode : Ethernet pour voir si les connexions sont valides
  - Un point rouge indique que non
  - En survolant les points verts, on lit les adresses provisoires :
    - mac01, mac02 et mac03





# Tâche 1



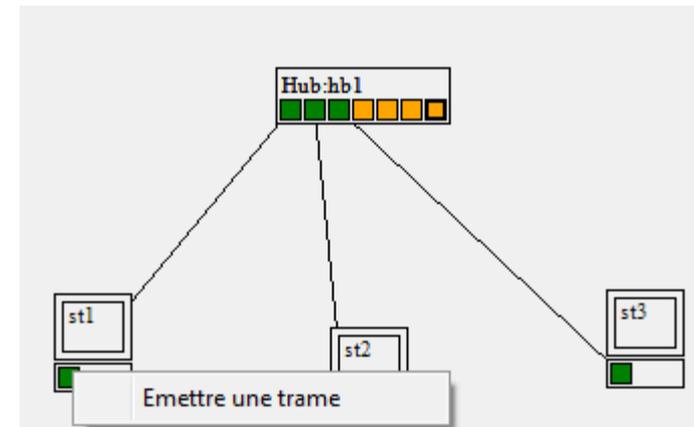
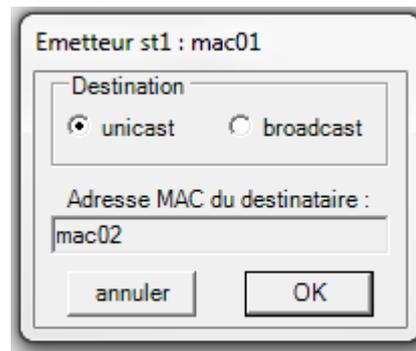
## 1. Concentrateur et adresse d'une carte réseau

### – Objectifs

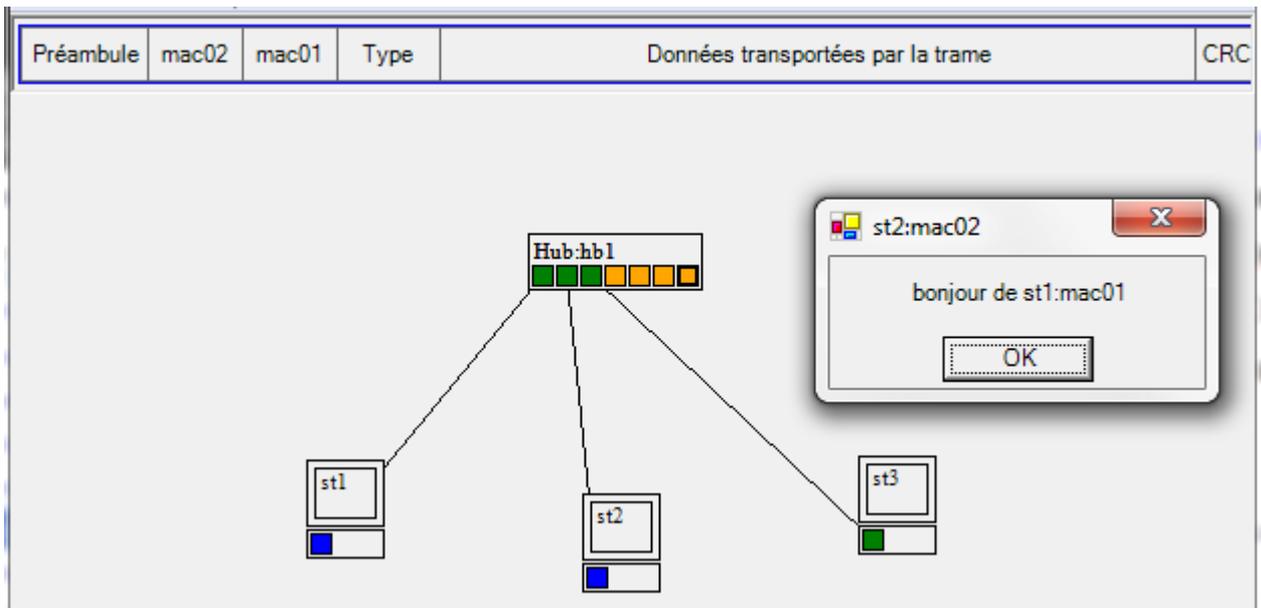
- Montrer le rôle de chaque élément, la circulation sur le câble et l'envoi à toutes les stations connectées (le concentrateur diffuse)
- Montrer la différence entre la réception de la trame par les cartes et la lecture de la trame (toutes les cartes reçoivent, mais la carte ne lit la trame que si elle reconnaît son adresse ou si il s'agit d'une trame de broadcast)
  - On peut activer le message de réception ou non pour bien montrer que la carte a lu la trame

## ■ Tâche 1 : créer un concentrateur (suite)

- Utiliser unhub.xml
- Se mettre en mode Internet
- Démo pas à pas
- Cliquer droit sur la carte réseau d'une station
- Faire envoi de trame en broadcast (à tous)
  - On voit la trame se diffuser de st1 vers le Hub, puis vers st2 et st3
- Si on essaie de lancer la trame en unicast,
  - On demande de cliquer sur la carte réseau
  - Cliquer sur le carré vers de st2
    - mac02 s'affiche
    - st2 et st3 reçoivent la trame mais seul st2 la lit



- **Tâche 1 : créer un concentrateur (suite)**
  - Un message s'affiche
  - Les carrés des cartes réseaux de st1 et st2 se mettent en bleu
  - Un réseau de l'action produite se met en haut



## ■ Tâche 1 : créer un concentrateur (suite)

### – À regarder :

- Relever les destinataires d'une trame
- Relever les lecteurs d'une trame
- Pourquoi une carte réseau lit ou ne lit pas une trame ?
- Que se passe-t-il si un poste est éteint alors qu'on lui envoie une trame ?
  - (le simulateur permet d'éteindre les postes : clic droit sur le poste)
- Que se passe-t-il si le concentrateur est éteint ?

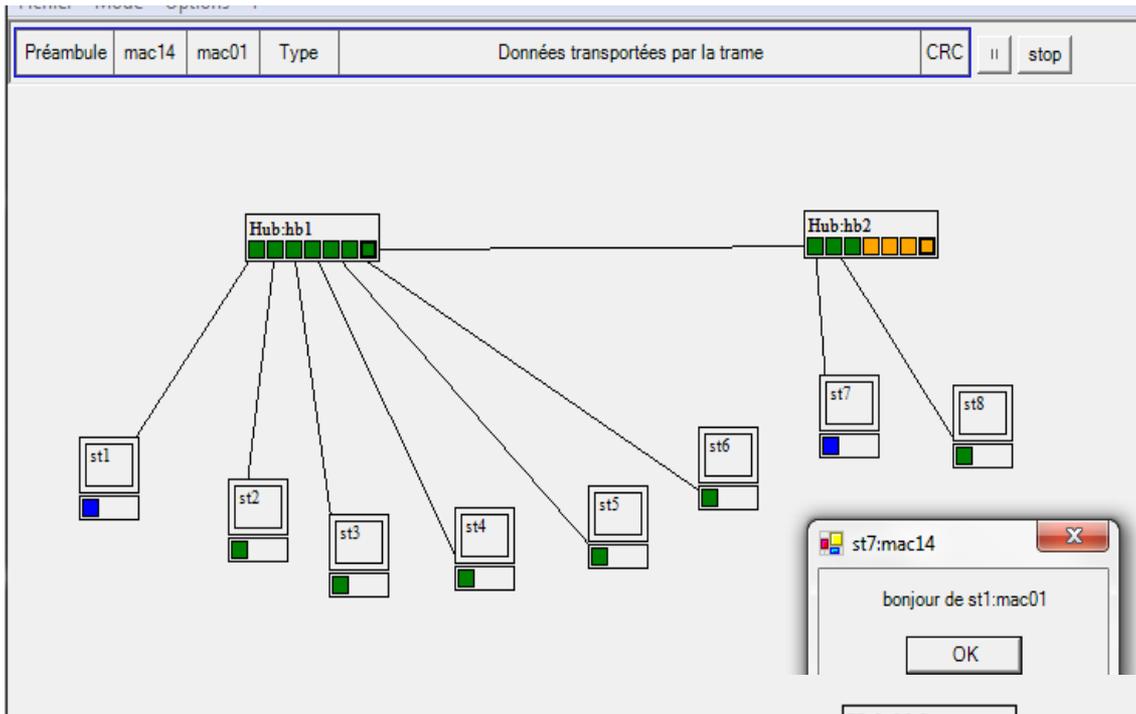
- Tâche 1 : créer un concentrateur (suite 2)
  - Interconnexion de concentrateurs
  - Attention à l'interconnexion des hubs
    - Elle se fait sur un port spécial (**uplink**) qui permet de croiser les paires
    - Petit point de cours : pourquoi faut-il croiser les paires entre deux hubs ?
      - car un concentrateur répète sur les paires de réception ce qu'il reçoit sur une paire d'émission
      - donc entre deux hubs la paire de réception sur lequel doit répéter un concentrateur doit être la paire d'émission pour l'autre

## ■ Tâche 1 : créer un concentrateur (suite 2)

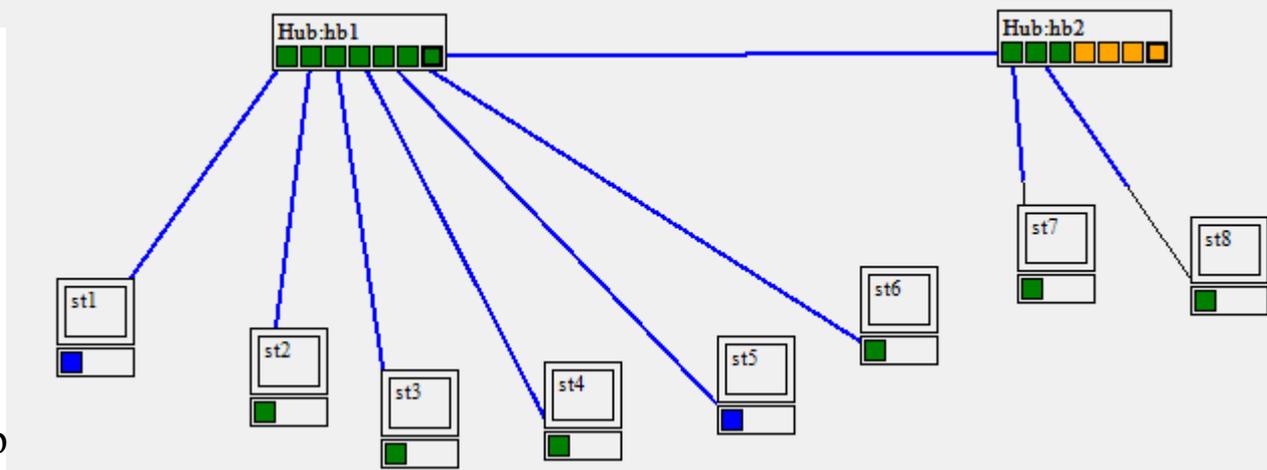
### – Manip

- On utilise **DeuxHub.xml**
- Il s'agit de montrer la propagation des messages à travers les concentrateurs
  - On envoie d'abord une trame entre deux stations connectées à deux concentrateurs différents
  - et on observe que ce message est diffusé à tous
  - On envoie ensuite une trame entre deux stations connectées à un même concentrateur et on observe que les stations connectées au second concentrateur reçoivent toujours le message

## ■ Tâche 1 : créer un concentrateur (suite 2)



Première manip



Deuxième manip

## ■ Tâche 1 : créer un concentrateur (suite 2)

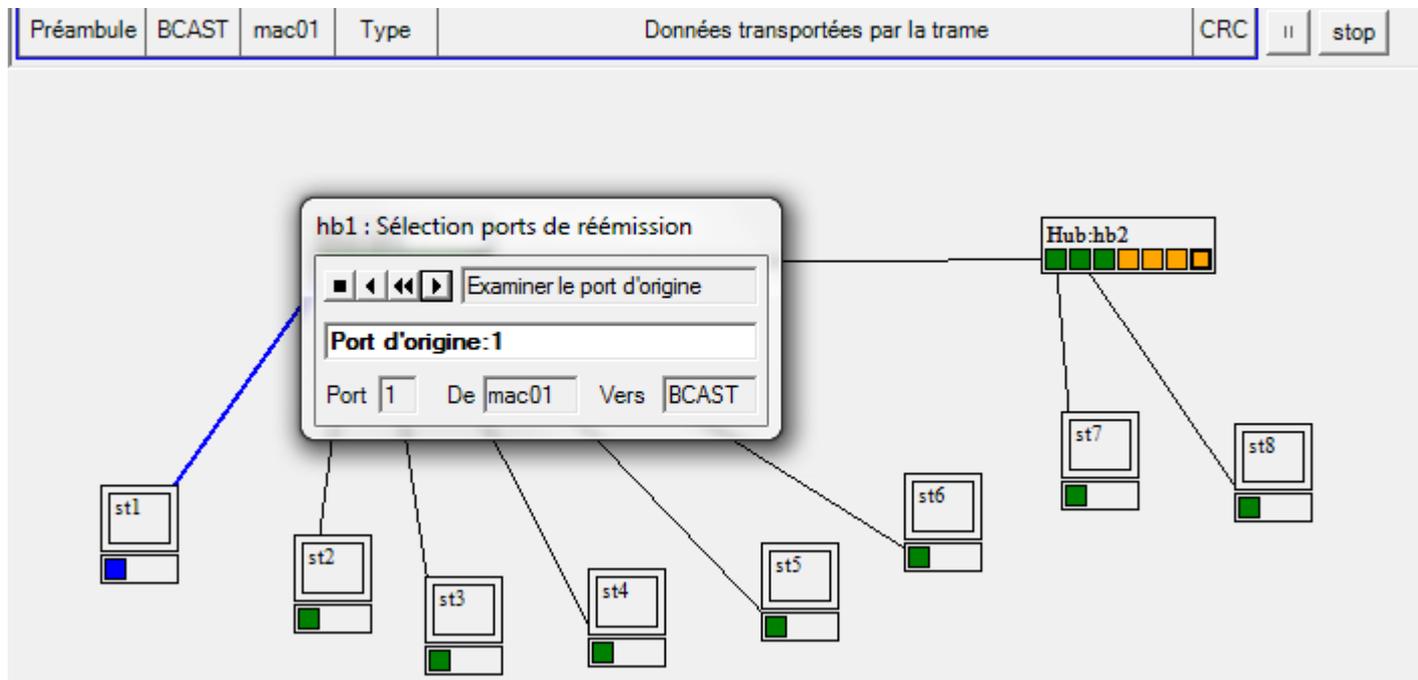
### – À faire :

- Relever les destinataires d'une trame
- Relever les lecteurs d'une trame
- Une trame adressée est-elle transmise à tous les postes ?
- Que se passe-t-il si un des concentrateurs est éteint ?
- On peut ajouter en mode conception un poste à hb1, puis un poste à hb2, puis un autre concentrateur pour bien illustrer le mécanisme

## ■ Tâche 1 : créer un concentrateur (suite 2)

### – A faire (suite)

- Tracer tous les nœuds
- Utiliser toujours le mode pas à pas
- On verra mieux le traçage du passage de la trame entre les stations





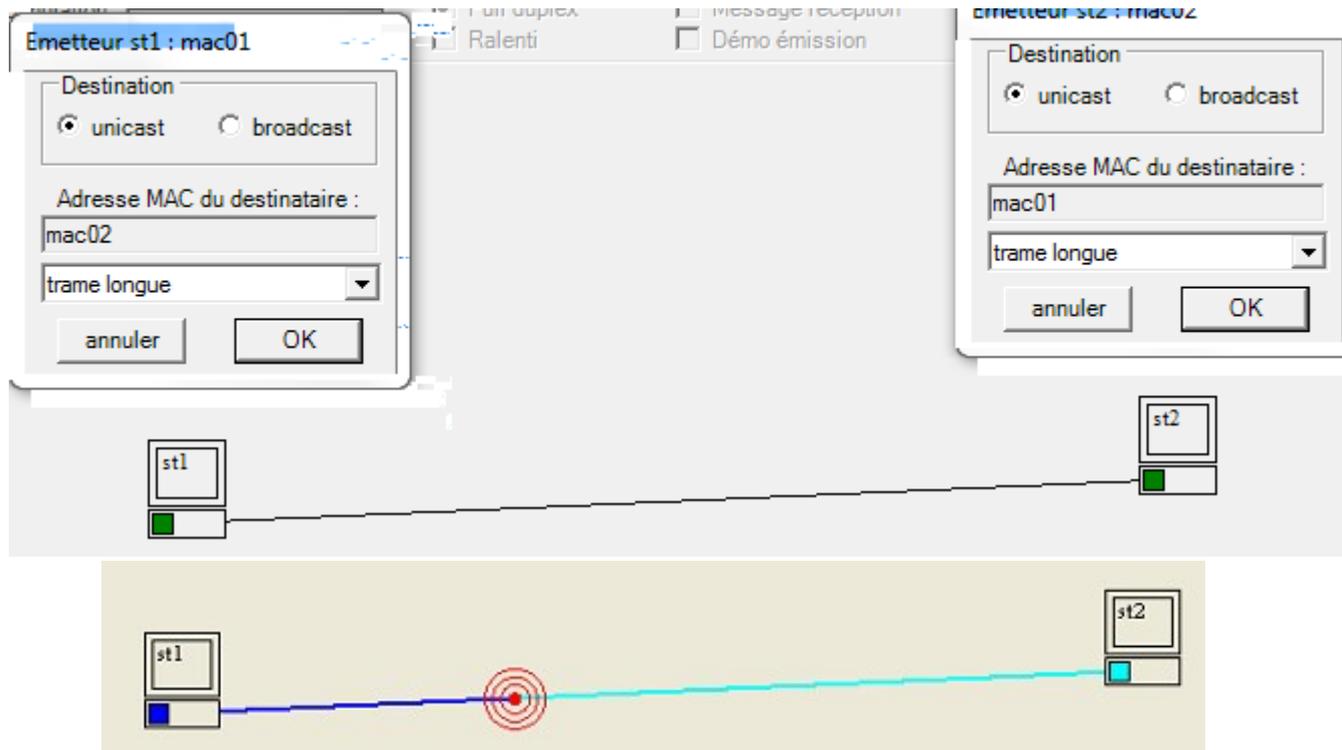
# Fiche 2



- La méthode d'accès CSMA/CD
  - On s'intéresse ici au partage du média d'accès
  - Dans Ethernet les stations émettent si le câble est libre
  - Si deux stations émettent au même moment alors que le câble est libre, on parle de **collision**
  - Le simulateur permet d'illustrer ce phénomène

# 1. La collision et la méthode d'accès CSMA/CD

- Utiliser le fichier *comprendre-reseaux/sim.xml/2postescablecoaxial*
- Utiliser « trame réelle »
- Envoyer une trame de st1 vers st2 et en même temps une trame de st2 vers st1
- Choisir trame longue pour avoir le temps de faire les deux envois





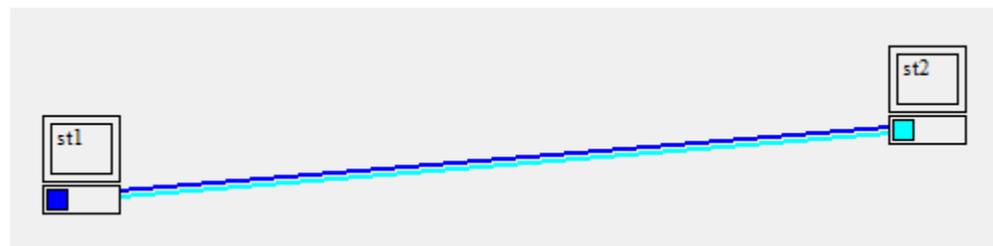
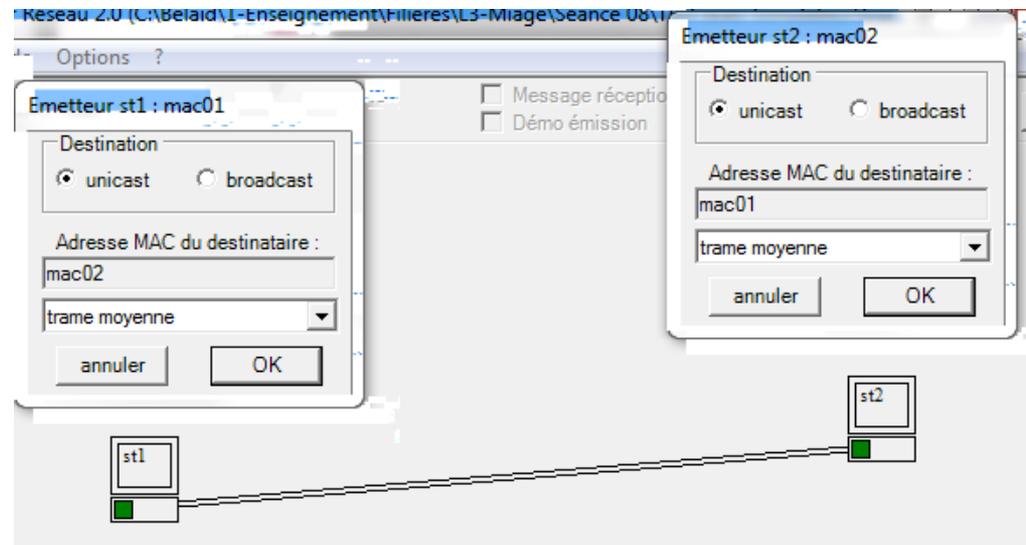
## ■ Questions

- Que s'est-il passé?
- Les postes s'en sont-ils rendu compte ?
- Qu'ont fait les postes ?



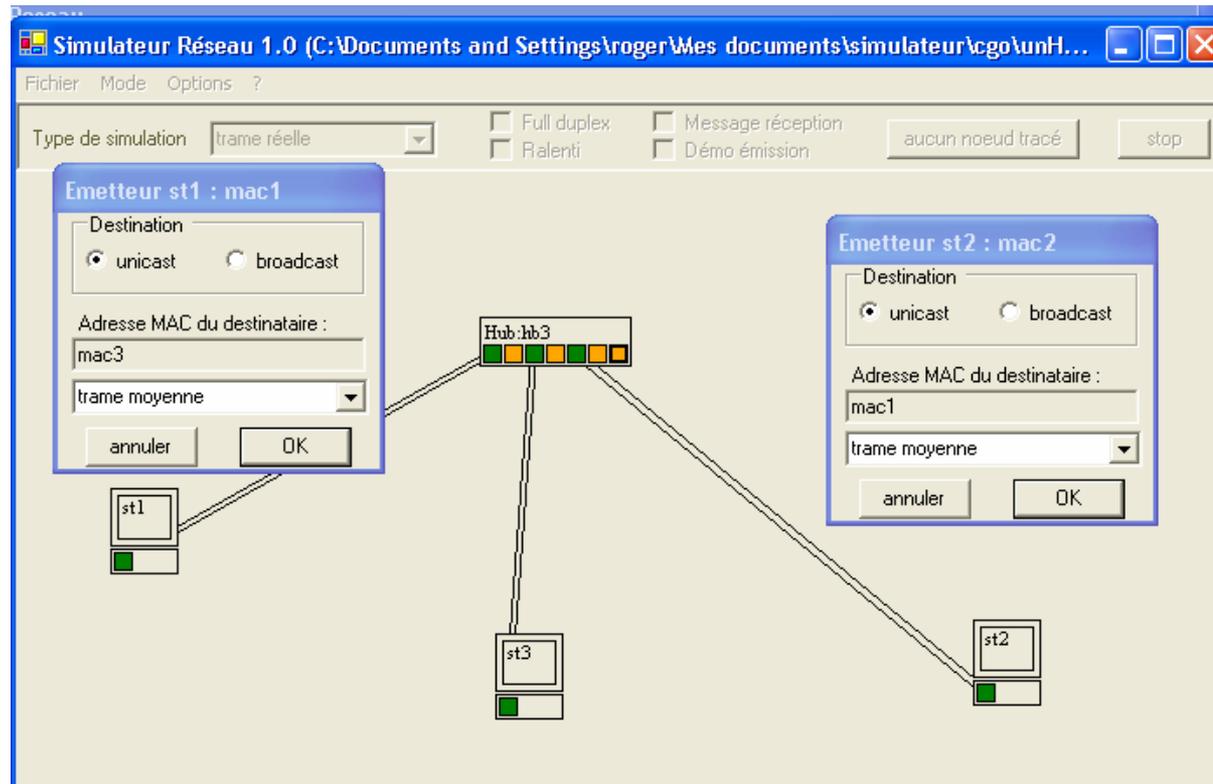
## ■ Le faire en paire torsadée croisée

- Pour obtenir la paire, configurer le câble en paire torsadée croisée, lancer une trame de chaque poste, alors la paire se double
- On voit qu'il n'y a pas de collision



## ■ Le faire en paire torsadée croisée avec un concentrateur

- On utilise le fichier unhub.xml
- On ne configure pas les cables, car déjà fait
- On montre la collision et la ré-émission



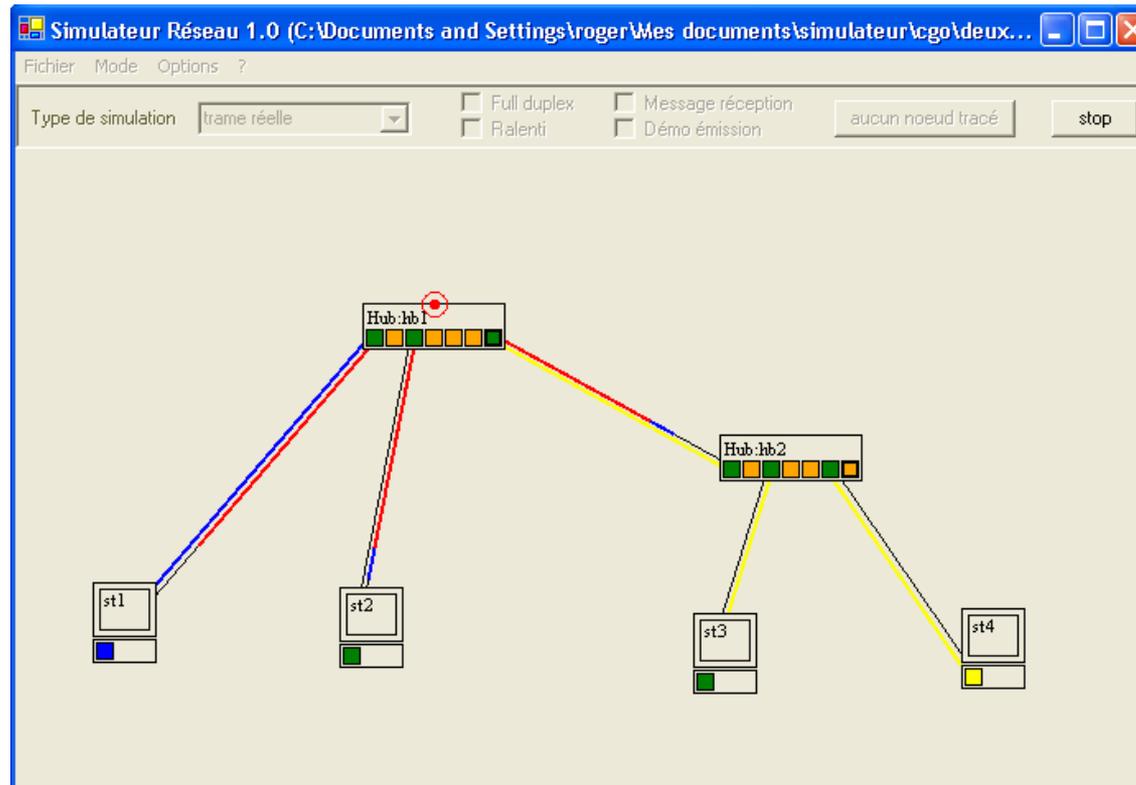
- On alternera les démonstrations « tracées » et « non tracées »
  - Envoyer une trame unicast à partir de st1 pour st3
  - Envoyer en même temps une trame unicast à partir de st2 pour st1

## ■ Questions

- Où se produit la collision ?
- Quels sont les postes qui détectent la collision ?
- Quels sont les postes qui réémettent une trame ?
- Quand les postes réémettent-ils ?
- Peut-il y avoir de nouveau collision après réémission ?

## 2. *La propagation de la collision entre les concentrateurs*

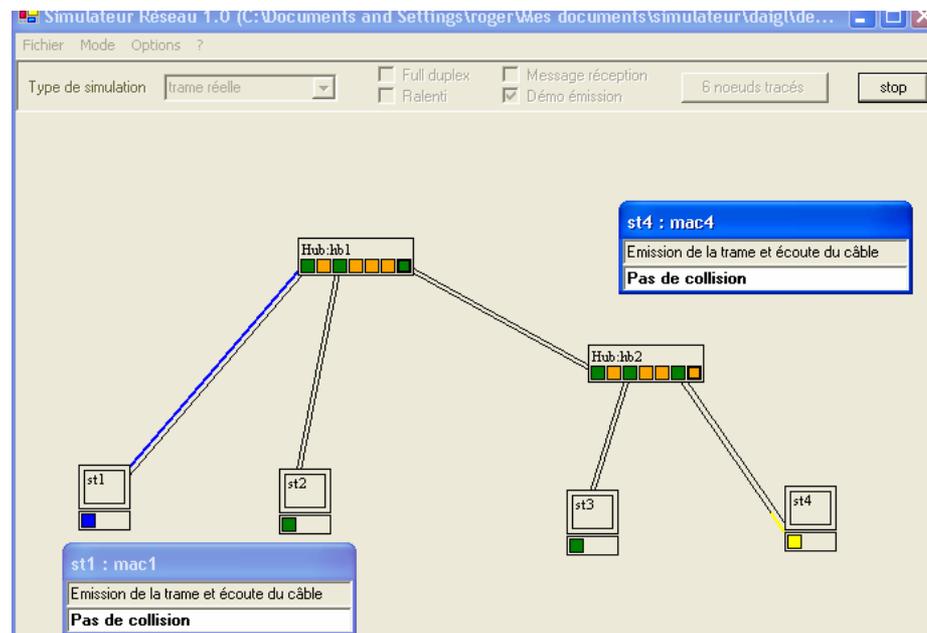
- On utilise le fichier **deuxhub.xml**
- On envoie d'abord une trame entre deux stations connectées à deux concentrateurs différents et on montre que la collision est diffusée à tous
- On envoie ensuite une trame entre deux stations connectées à un même concentrateur et on montre que la collision se propage aussi au second concentrateur
  - Envoyer une trame unicast à partir de st1 pour st3
  - Envoyer en même temps une trame broadcast à partir de st2 pour st1
  - Envoyer une trame de broadcast à partir de st4
  - Envoyer en même temps une trame broadcast à partir de st1



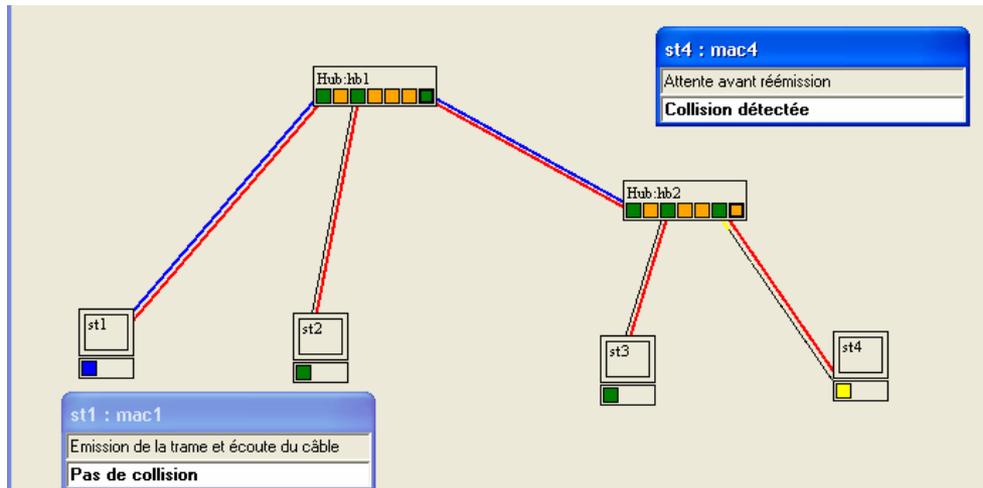
## ■ Question

- Une trame de broadcast a-t-elle plus de risque de provoquer une collision qu'une trame unicast ?
  - *Non, les deux trames sont propagées de la même façon par le concentrateur*

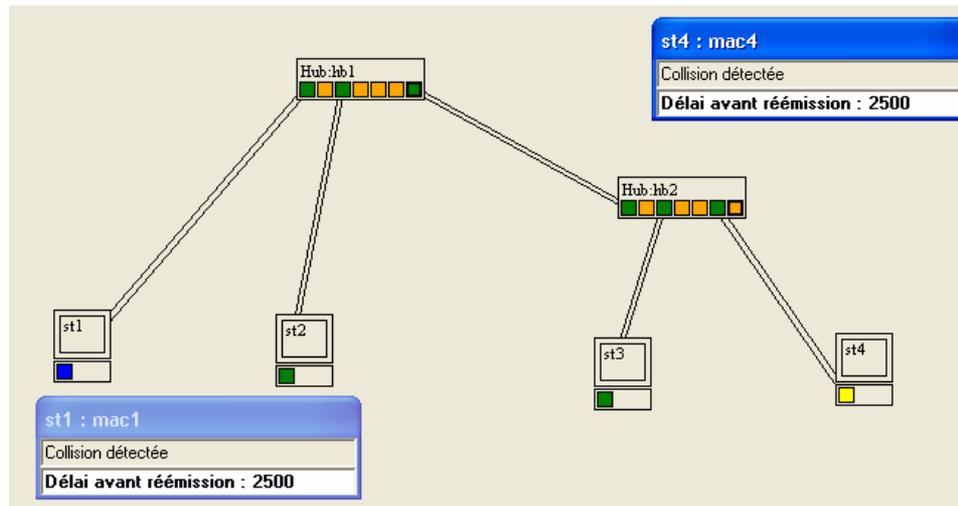
- Pour parfaire la démonstration, on peut utiliser le mode « trame réelle » en démo émission
- Ceci s'obtient en demandant le traçage de différents éléments
- On montre ainsi dynamiquement la gestion des collisions
- St1 et st4 ont envoyé une trame
- Les deux cartes restent à l'écoute sur la paire de réception, pour l'instant elles ne détectent pas de collision



- Une collision s'est produite et se propage par les paires de réception, elle est parvenue à st4 qui l'a détecté mais pas encore à st1.

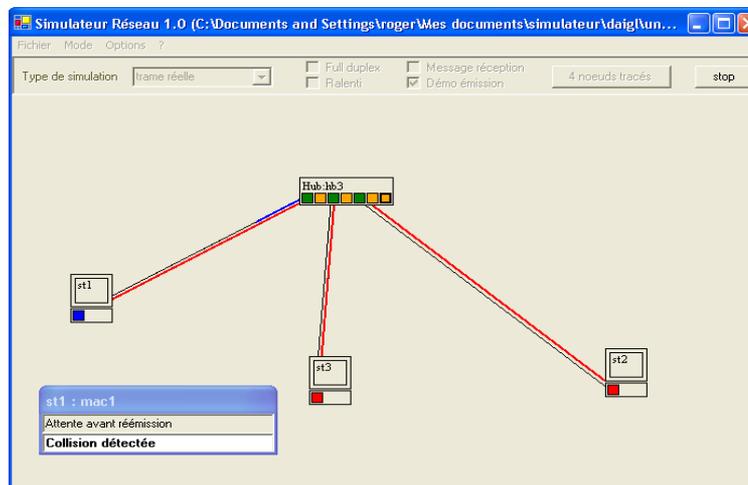


- Les deux cartes ont détecté la collision et ont calculé un délai avant réémission (pas de chance dans cet exemple elles ont calculé le même, il y aura donc une nouvelle collision)



### 3. Expérimenter le round trip delay

- On utilise le fichier unhub.xml
- On travaille toujours en mode **trame réelle** et démo-émission (des nœuds doivent être tracés)
- On va illustrer le problème du « round trip delay » avec des trames trop courtes dans un premier temps puis avec des câbles trop longs dans un second temps, enfin avec un nombre de répéteurs trop important , et montrer que la collision n'est pas détectée
  - Envoyer une trame unicast à partir de st1 pour st2. *choisir trame moyenne*
  - Envoyer en même temps une trame broadcast à partir de st2 pour st1, choisir trame courte

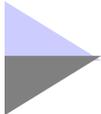


## ■ Questions

- Expliquer le comportement de st2
- Refaire cette démonstration avec des trames moyennes mais en affectant 115 comme longueur de câble à st1

## 4. Construire un réseau composé de cinq concentrateurs

- On peut utiliser [cinqhub.xml](#) ou construire son propre réseau
- Montrer les limites de la diffusion (surcharge du réseau) multiplication des collisions, ré-émission incertaine, etc.
- Montrer que lorsque les postes aux extrémités du réseau émettent en même temps (mode trame réelle) la collision n'est pas détectée, indépendamment des longueurs de câbles et de trames, cette fois, c'est le nombre de répéteurs traversés qui pose problème
- **D'où la nécessité des commutateurs**
- Enfin, montrer qu'entre deux postes avec un câble croisé, les collisions sont impossibles (introduction à l'Ethernet commuté puis au full duplex) : fichier [2postescablecroise.xml](#) et comparer avec le câble coaxial (fichier [2postescablecoaxial.xml](#))



# Fiche 3



## ■ La commutation

- On veut montrer l'absence de **collision** et l'absence de diffusion quand on transmet des trames avec des adresses **unicast** à travers des commutateurs
- Rappel
  - Un commutateur ne diffuse pas les messages à tous les postes
  - Il lit l'adresse du destinataire du message et envoie le message sur le port où est connecté le destinataire

# 1. La commutation et l'utilisation de l'adresse MAC de l'émetteur

- Utiliser unswitch.xml
- On montre l'apprentissage de la connaissance des adresses MAC des postes connectés
- On montre la différence de traitement entre adressage unicast et adressage multicast
- Enfin on montre l'absence de collision

- Phase d'apprentissage du commutateur (mode automatique)
  - Vider la table des commutateurs
  - Chaque station envoie une fois
  - On envoie une trame de st1 vers st3. On constate que tous les postes reçoivent
  - On envoie une trame de st3 vers st1. Seul st1 reçoit
  - On renvoie une trame de st1 vers st3. Seul st3 reçoit
  - On envoie une trame de st1 vers st2. st2 et st3 reçoivent
  - St2 envoie une trame à st3. st3 reçoit
  - St1 envoie une trame à st2. Seul st2 reçoit. A ce stade le commutateur a associé à chacun de ses ports une adresse de carte réseau
  - On peut continuer à envoyer des trames unicast des différents postes et donc bien insister sur la différence avec un concentrateur
  - St1 envoie une trame de broadcast. Tous les postes reçoivent. Le commutateur diffuse les trames de diffusion et les trames pour lesquelles il n'a pas encore associé de port à l'adresse destinataire de la trame

## ■ A faire

- Relever les destinataires d'une trame
- Relever les lecteurs d'une trame

## ■ Questions

- Pourquoi lors du deuxième envoi, seul st1 a été destinataire de la trame ?
- Que faut-il pour que le commutateur sache où se trouvent tous les postes ?
- Le commutateur peut-il ignorer la position d'une station ?
- Que fait un commutateur quand il reçoit une trame dont il ne connaît pas l'adresse du destinataire ?
- Que fait un commutateur quand il reçoit une trame de diffusion ?

## ■ Questions-réponses

- Pourquoi lors du deuxième envoi, seul st1 a été destinataire de la trame ?
  - car lorsque st1 a envoyé la trame précédente, le commutateur a associé à son port de réception de la trame l'adresse de l'émetteur de la trame
- Que faut-il pour que le commutateur sache où se trouvent tous les postes ?
  - toutes les stations doivent avoir émis au moins une trame
- Le commutateur peut-il ignorer la position d'une station ?
  - oui si celle-ci n'émet jamais de trames (ce qui est pratiquement impossible, une station destinataire d'une trame répond généralement à cette trame)
- Que fait un commutateur quand il reçoit une trame dont il ne connaît pas l'adresse du destinataire ?
  - il diffuse à tous les ports sur lequel se trouvent des connexions (il agit comme un concentrateur dans ce cas).
- Que fait un commutateur quand il reçoit une trame de diffusion ?
  - idem

- On utilise le mode pas à pas et trace le commutateur
  - *St1 émet une trame vers st3.* On montre la décision prise par le switch

The screenshot shows a network simulator window titled "Simulateur Réseau 1.0". The interface includes a menu bar with "Fichier", "Mode", and "Options". Below the menu bar, there are several controls: a dropdown menu for "Type de simulation" set to "pas à pas", checkboxes for "Full duplex", "Message réception", "Ralentir", and "Démonstration", and buttons for "1 noeud tracé" and "stop".

The main workspace displays a network diagram with three nodes: "st1", "Switch:sw1", and "st2". "st1" is connected to "Switch:sw1" by a blue line, and "Switch:sw1" is connected to "st2" by a black line. The switch node has five colored ports (green, green, green, orange, orange).

A configuration window for "sw1" is open, showing a table for "Destinataire port 3". The table has columns for "Adresse", "Port", and "TTL". The first row is "mac1" on port 1 with "Maximum" TTL. The second row is "mac3" on port 3 with "Moyen" TTL. The second row is highlighted in blue.

Adresse	Port	TTL
mac1	1	Maximum
mac3	3	Moyen

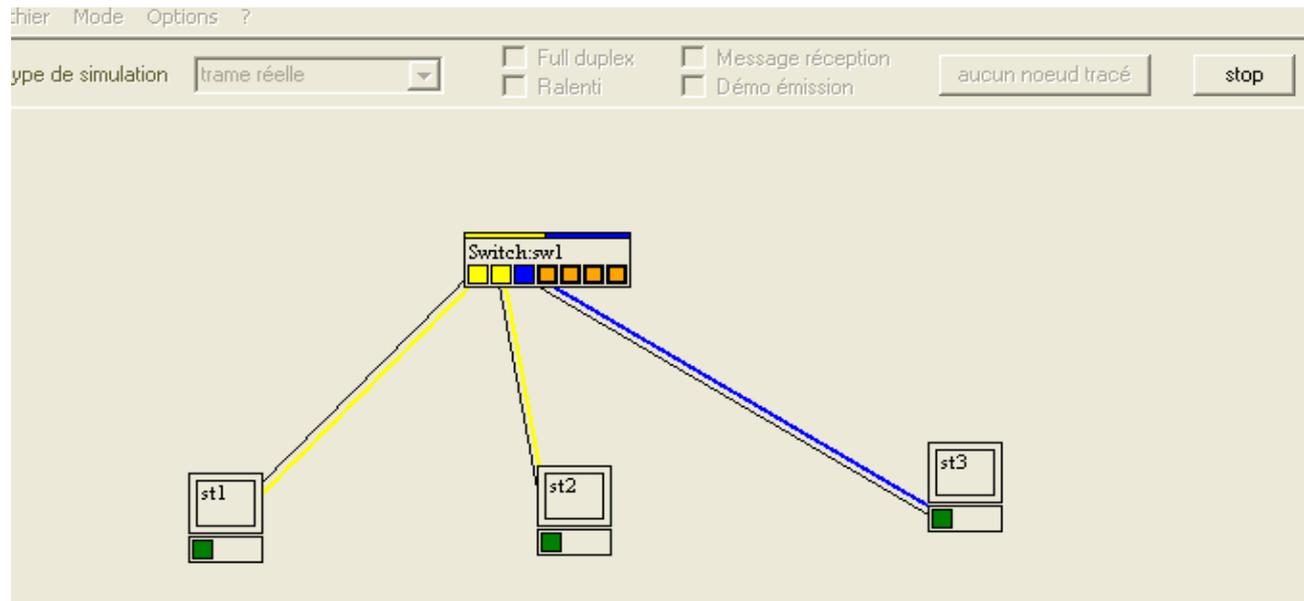
## ■ Questions

- Comment est construite la table ?
- Qu'est ce que le TTL ?
- A quoi sert le TTL ?
- Que se passe-t-il lorsque le TTL est égal à zéro ?
- Quand le TTL est-il mis à jour ?

## ■ Absence de collisions

### – On utilise le mode trame réelle

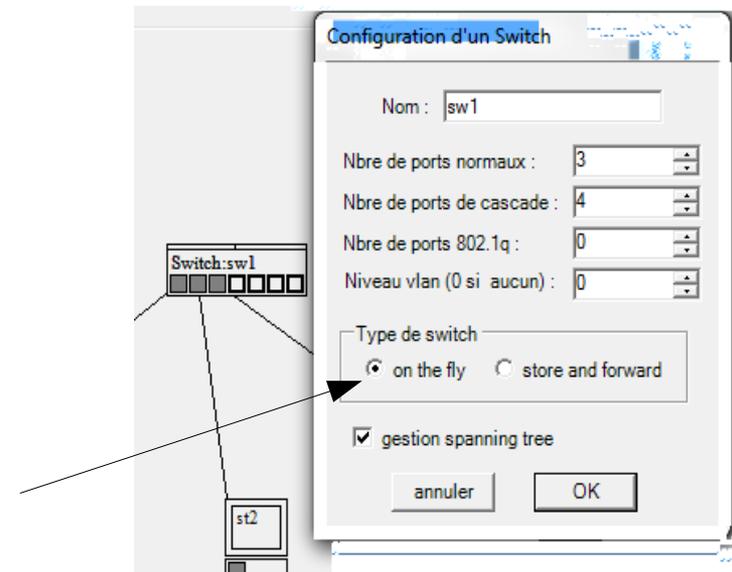
- St1 envoie une trame et st2 envoie une autre trame en même temps (mode manuel), unicast ou broadcast peu importe
- Il n'y a pas de collision
- On voit au-dessus du commutateur les bandes utilisées pour stocker les trames, de manière à éviter leur collision



- Questions

- Pourquoi n'y a-t-il pas de collisions ?

- On configure le commutateur en type « on the fly » et on passe en mode « automatique »
- St1 envoie une trame à st2
  - Il n'y a pas de stockage de la trame, la trame est commutée dès que le commutateur lit l'adresse du destinataire dans la trame
- On configure le commutateur en type « on the fly » et on revient en mode « trame réelle » en traçant tous les nœuds et en cochant «démon émission»
- *St1 envoie une trame à st3. st3 envoie une trame au même moment.* Bien qu'il n'y ait pas de stockage de la trame, il y a écoute du câble avant réémission de la trame



- **Interconnexion de commutateurs (fiche 3)**
  - *On utilise deuxswitvh.xml en mode automatique*
  - *Il s'agit de montrer la gestion de la commutation entre deux commutateurs*
  - *On distinguera encore une fois trame unicast et trame de broadcast.*
    - *On envoie une trame de st1 à st2 connecté au même commutateur. St2 et st3 reçoivent car sw1 a transmis à sw2*
    - *On envoie une trame de st2 à st1 seul st1 reçoit*
    - *On envoie une trame de st1 vers st3. St2 et st3 reçoivent*
    - *On envoie une trame de st2 vers st3. St1 et st2 reçoivent*
    - *On envoie une trame de st3 vers st2. Seul st2 reçoit*
    - *On envoie une trame de st1 vers st3. Seul st3 reçoit. A ce stade les deux commutateurs ont associé à chacun de leur port une adresse de carte réseau*
    - *Envoyer une trame de broadcast à partir de st3*
    - *Envoyer une trame de broadcast à partir de st1*

## ■ *Questions*

- Pourquoi sw1 transmet-il la trame à sw2 lors du premier envoi ?
- Pourquoi sw1 ne transmet rien à sw2 lors du deuxième envoi ?
- Pourquoi st2 et st3 reçoivent-ils lors du 3eme envoi alors que st2 a déjà été destinataire d'une trame ?



# Fiche 4



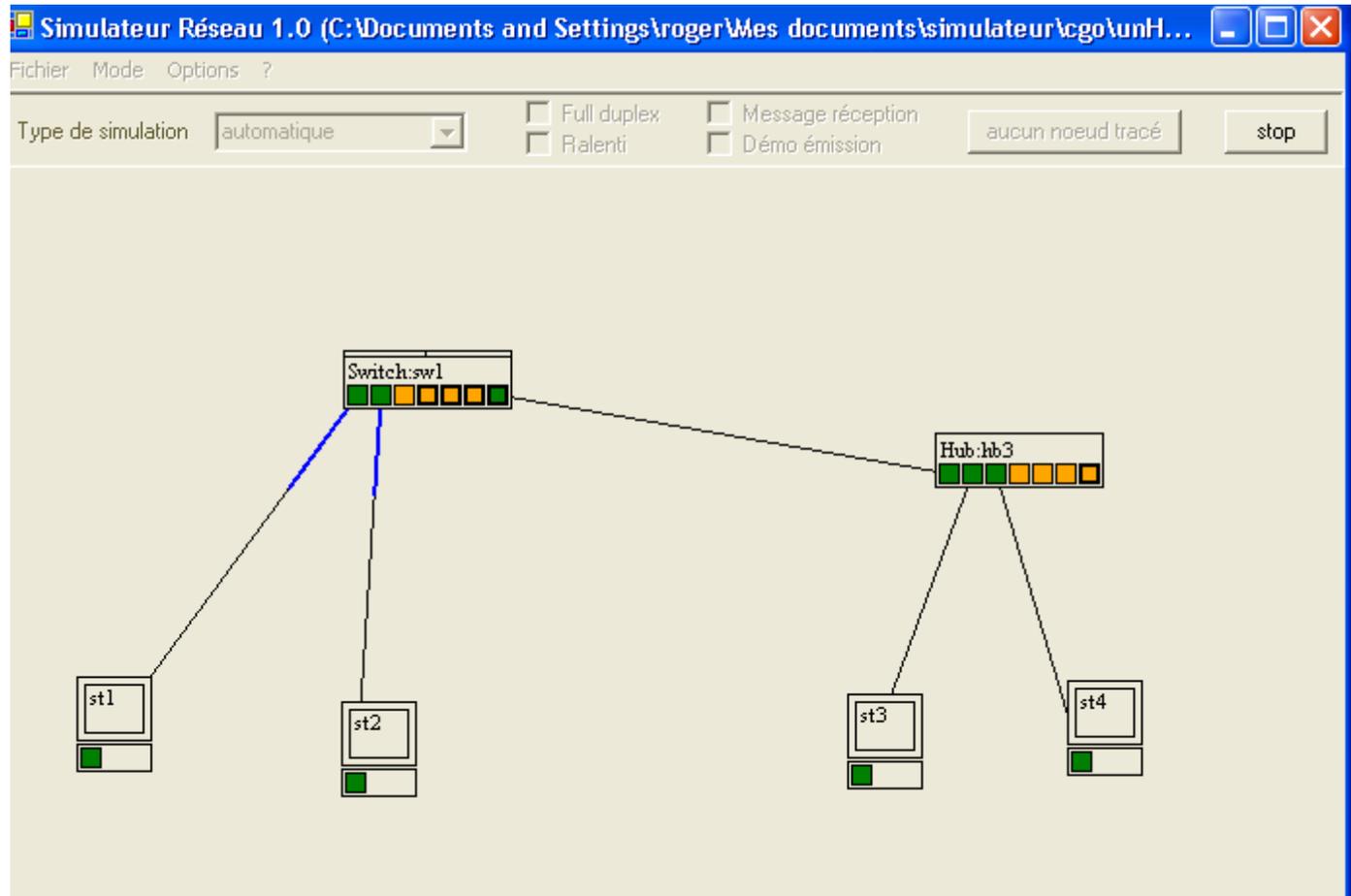
- **Le commutateur comme matériel fédérateur**
  - Il s'agit de montrer la construction d'une architecture complexe en utilisant le commutateur comme matériel fédérateur
  - On veut monter l'interconnexion de commutateurs avec des concentrateurs

## ■ 1. L'interconnexion des commutateurs et des concentrateurs

- On utilise le fichier unhubunswitch.xml
- On montre la diffusion dans le concentrateur et la commutation dans le commutateur.
- Phase d'apprentissage du commutateur
  - On envoie une trame de st1 à st2 connecté au même commutateur. *St3 et st4 reçoivent car sw1 a transmis à hb3*
  - On envoie une trame de st2 à st1 *seul st1 reçoit.*
  - On envoie une trame de st1 vers st3. *St2 st3 et st4 reçoivent.*
  - On envoie une trame de st2 vers st3. *St3 et st4 reçoivent.*
  - On envoie une trame de st3 vers st2. *St2 et st4 reçoivent.*
  - On envoie une trame de st4 vers st1. *St3 et st1 reçoivent.*
  - On envoie une trame de st4 vers st3. *st3 reçoit. Mais hb3 transmet bien au commutateur. A ce stade le commutateur connaît toutes les adresses.*

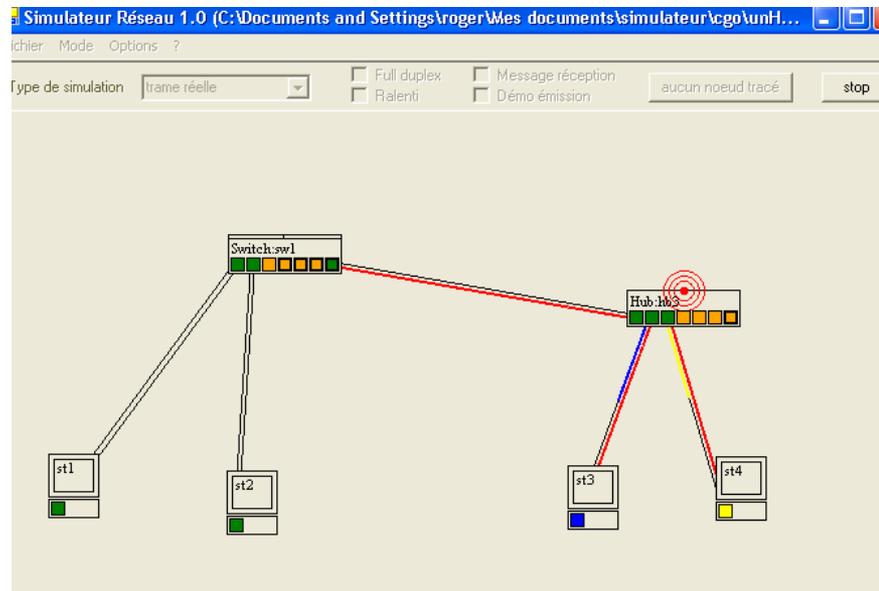
# ■ 1. L'interconnexion des commutateurs et des concentrateurs

## – Schéma



# ■ 1. L'interconnexion des commutateurs et des concentrateurs

- On utilise le mode trame réelle pour provoquer une collision dans le concentrateur et observer comment réagit le commutateur
  - On envoie une trame de st3 vers st1
  - On envoie en même temps une trame de st4 vers st2. *Il n'y a pas de propagation de la collision par le commutateur (en réalité c'est parce que le commutateur est de type « store and forward » ce qui est généralement le cas).*



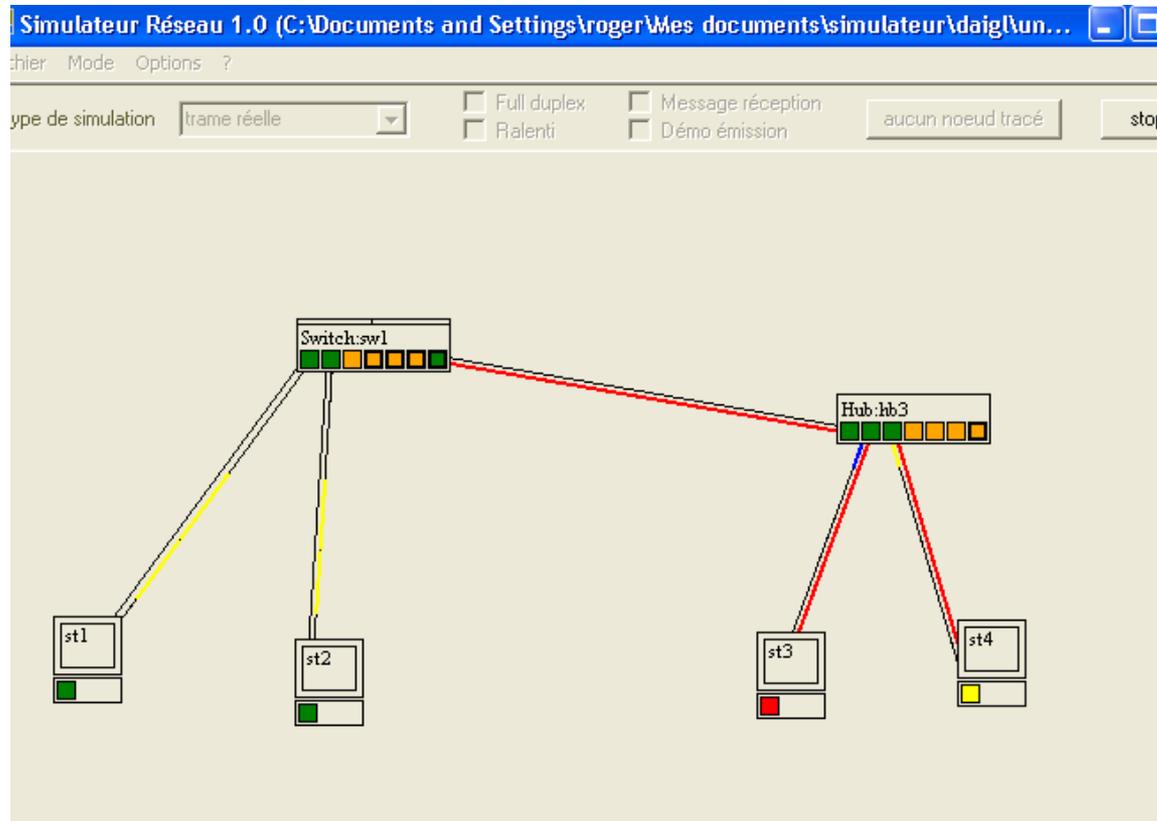
## ■ 1. L'interconnexion des commutateurs et des concentrateurs

### – A faire

- Relever les destinataires de la trame
- Relever les lecteurs de la trame
- Anticiper ou expliquer les résultats. *A ce stade, vous devez pouvoir prévoir le comportement des deux matériels*

- 1. L'interconnexion des commutateurs et des concentrateurs
  - On configure le commutateur en type « on the fly »
  - On reste en mode trame réelle pour provoquer une collision dans le concentrateur et observer comment réagit le commutateur
    - On envoie une trame de st3 vers st1
    - On envoie en même temps une trame de st4 vers st2
      - *Il n'y a pas de propagation de la collision par le commutateur par contre on peut constater que la trame a commencé à être retransmise alors qu'elle a provoqué une collision*
      - *La trame ne sera pas envoyée totalement et sera donc erronée (CRC faux) et rejetée par les cartes réceptrices. Elle sera ensuite réémise*

- 1. L'interconnexion des commutateurs et des concentrateurs
  - Schéma

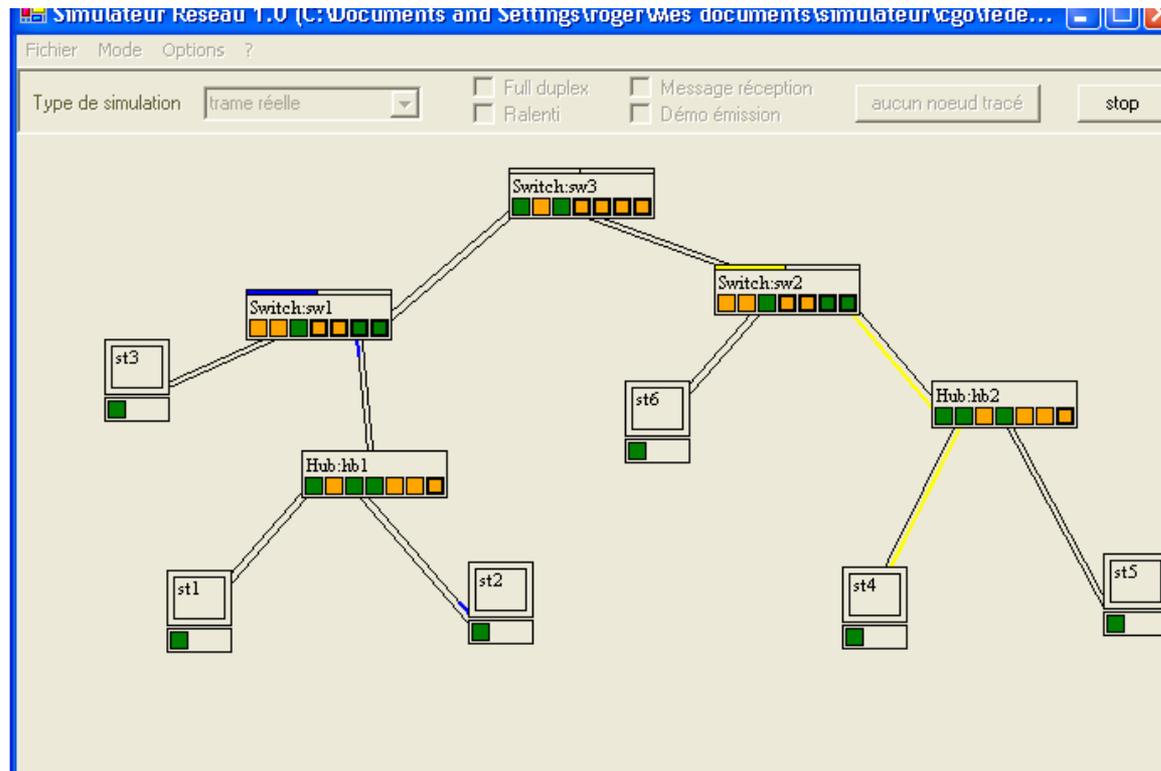


## ■ *Questions*

- La collision est-elle propagée ?
- Une trame en collision est-elle commutée ?
- Qu'arrive-t-il à une trame incomplètement transmise ?

## ■ 2. Architecture fédérée autour des commutateurs

- On utilise fédéswitch.xml
- Il s'agit de noter le résultat des actions suivantes
  - Quels sont les destinataires de la trame ?
  - Quels sont les lecteurs de la trame ?



## ■ 2. Architecture fédérée autour des commutateurs

### – Manipulations

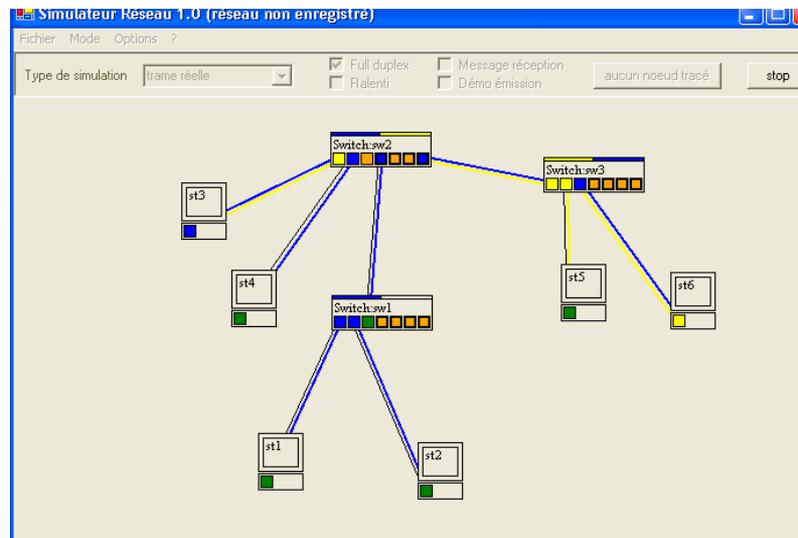
- On envoie une trame de st1 vers st2 .
- On envoie une trame de st2 vers st1.
- On sélectionne le type de simulation « trame réelle ».
- On envoie une trame de st1 vers st2 et en même temps on envoie une trame de st2 vers st1.

### – Toujours en « trame réelle »

- On envoie une trame de broadcast de st1 vers st2 et en même temps une trame de broadcast de st2 vers st1
- *Malgré l'utilisation d'adresses de broadcast, la collision ne se propage pas*
- *On observe aussi que le commutateur ne stocke que les trames correctes et non les trames « en collision »*
- St3 envoie une trame à st1.
- St6 envoie une trame à st1.
- St4 envoie une trame à st1.
- St6 envoie une trame à st5.
- St3 envoie une trame à st6.
- St3 envoie une trame à st4.
- St1 et st5 envoient simultanément une trame de broadcast.

### ■ 3. Architecture entièrement commutée

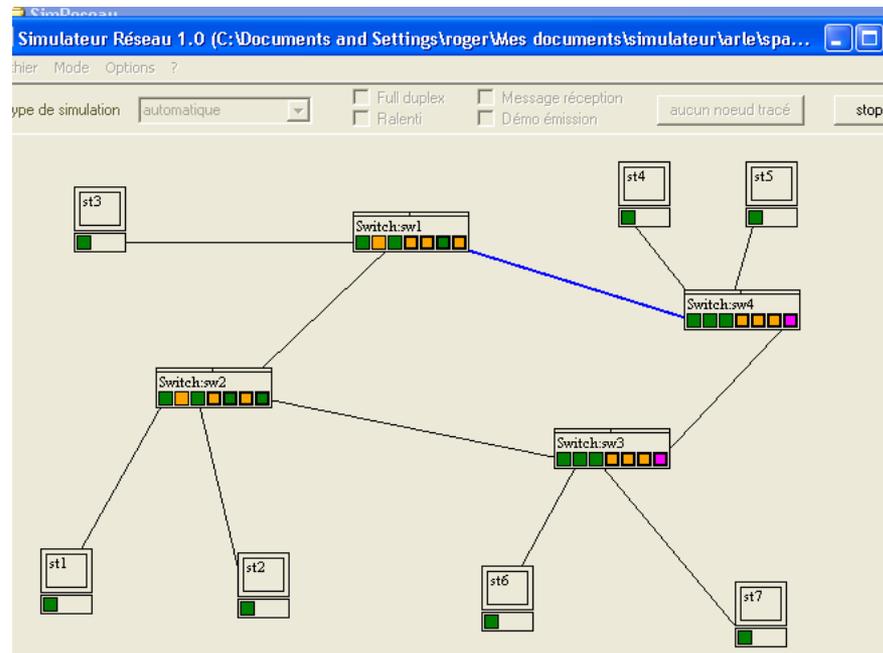
- On utilise le fichier full.xml
- Dans une architecture entièrement commutée,
  - il n'y a plus de collision on peut donc travailler en mode full-duplex car la paire de réception n'est plus monopolisée pendant l'envoi pour détecter la collision
- On peut utiliser aussi le type « on the fly »
  - Attention « on the fly » et « full duplex » sont des techniques indépendantes, on peut être à la fois en mode « store and forward » et en mode « full duplex »



## ■ Questions

- Quels sont les avantages et les inconvénients du type « on the fly » ?
- Quel est l'intérêt du « full duplex » ?
- Peut-on être en full-duplex sur le commutateur et pas sur la carte réseau ?

- 4. Ethernet commuté : 802.1d spanning tree (arbre de recouvrement)
  - Pour gérer la tolérance aux pannes des commutateurs, on introduit des circuits redondants qui sont inactifs en fonctionnement normal et activés en cas de panne
  - Pour cela chaque commutateur doit activer le protocole spanning tree (802.1d)
  - Les commutateurs doivent être configurés en mode conception réseau

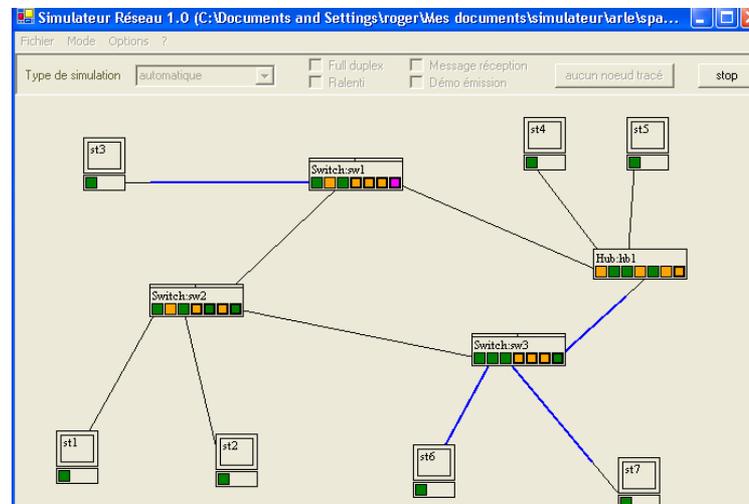


- 4. Ethernet commuté : 802.1d spanning tree (arbre de recouvrement)
  - Manipulation
    - On utilise le fichier span1.xml
    - On envoie de broadcast à partir de st1
  - Questions
    - Pourquoi un port est-il invalidé sur les switch3 et le switch4 (ces ports apparaissent par défaut en rose) ?
    - Quelle serait la conséquence d'une boucle sur la transmission de la trame de broadcast ?
    - Quelle serait la conséquence pour une trame *unicast* dont les commutateurs n'ont pas d'association pour l'adresse destinataire ?

## ■ Questions-réponses

- Pourquoi un port est-il invalidé sur les switch3 et le switch4 (ces ports apparaissent par défaut en rose) ?
  - *Pour éviter une boucle (un circuit)*
- Quelle serait la conséquence d'une boucle sur la transmission de la trame de broadcast ?
  - *la trame serait retransmise indéfiniment par les commutateurs*
- Quelle serait la conséquence pour une trame *unicast* dont les commutateurs n'ont pas d'association pour l'adresse destinataire ?
  - *Cette trame unicast serait diffusée sur tous les ports des commutateurs et poserait le même problème que la trame de broadcast.*

- 4. Ethernet commuté : 802.1d spanning tree (arbre de recouvrement)
  - Manipulation (suite)
    - On utilise maintenant le fichier span2.xml
    - Ici on a un concentrateur connecté à deux commuateurs. Bien que les concentrateurs ne gèrent pas le protocole spanning tree, ils diffusent (niveau 1 oblige) les trames 802.1d et permettent donc l'invalidation d'un circuit y compris à travers eux
    - On envoie une trame à partir de st1. *Le circuit entre sw1 et hb1 a été invalidé, il n'y a pas de boucle*



## ■ Questions

- Comment les deux commutateurs ont-ils pu invalider le circuit à travers le concentrateur ?



# Fiche 5

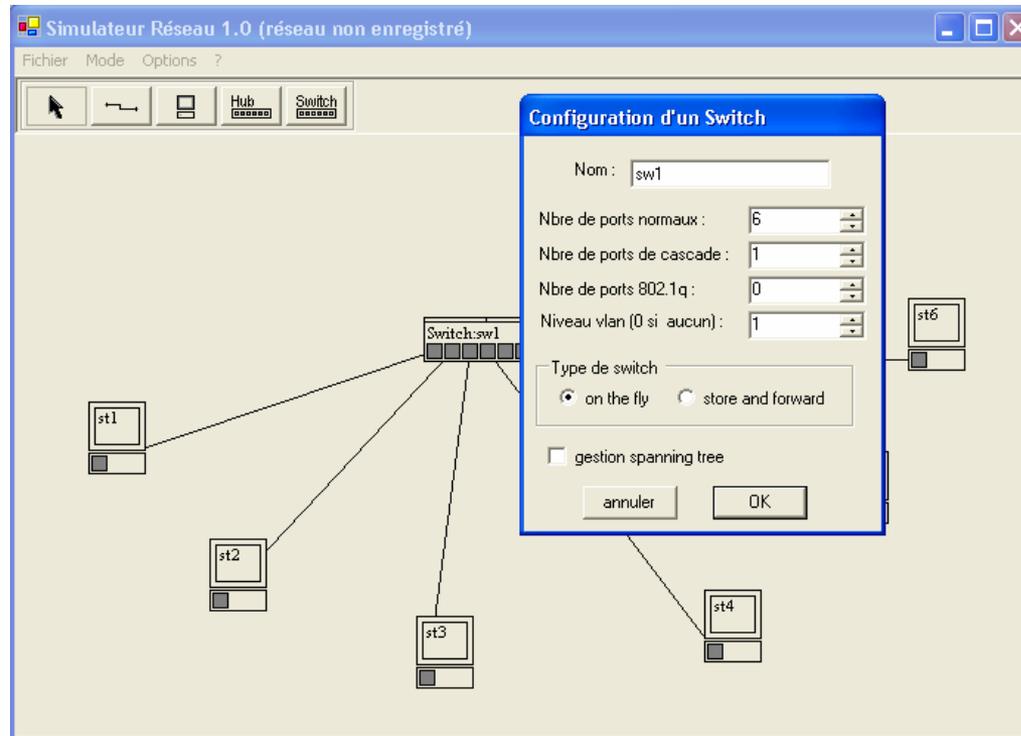


## ■ Objectifs

- Montrer le fonctionnement des VLAN
- Montrer l'interconnexion de commutateurs gérant des VLAN de niveau 1 en utilisant des ports 802.1q

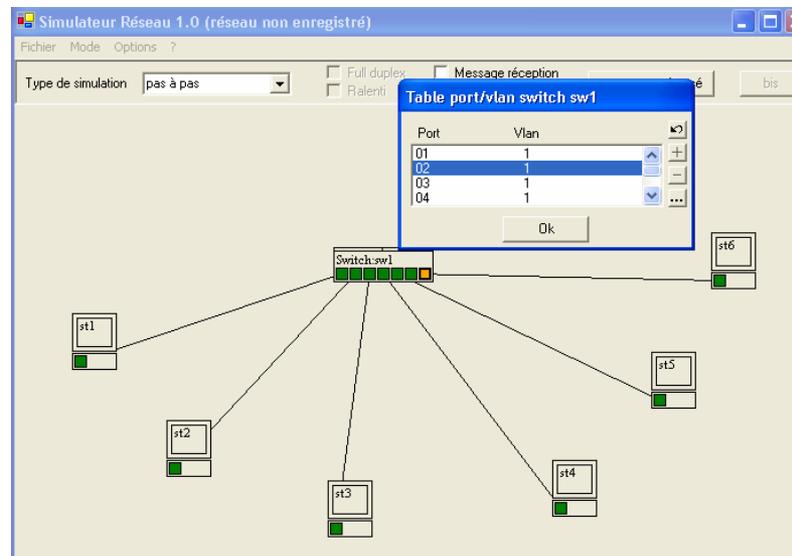
# 1. Comportement d'un commutateur gérant des VLAN de niveau 1

- *On utilise le fichier unswitchvlan1.xml*
- Définition du niveau de VLAN
  - En mode « conception ethernet » choisir 1 comme niveau de VLAN



## – Affectation des ports au VLAN

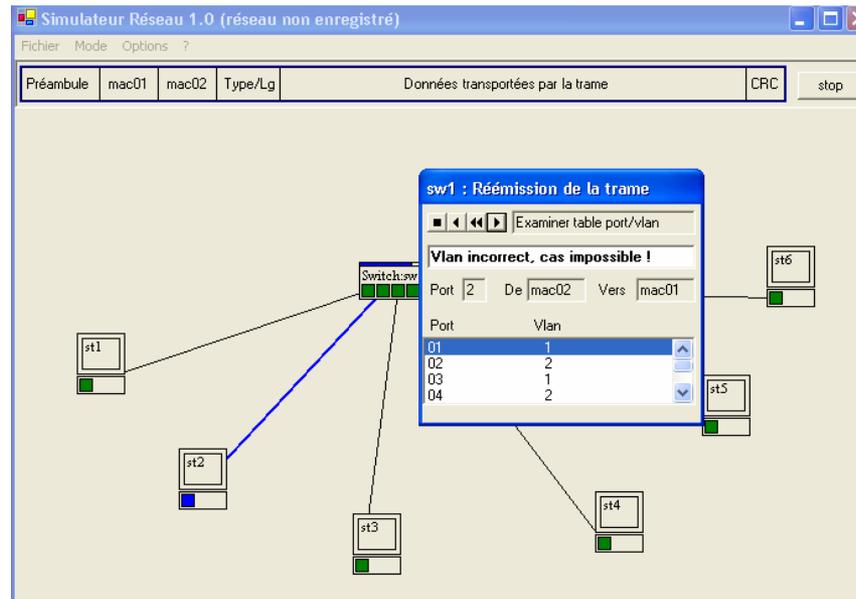
- On utilise le mode Ethernet
- Clic droit sur le commutateur, **Edition de la table port / Vlan**
  - On constate que tous les ports appartiennent au même VLAN, en l'absence de définition explicite de VLAN
- On sélectionne le port que l'on veut affecter et on clique sur le bouton avec les 3 points
- On affecte les ports 2 4 et 6 au VLAN 2; les ports 1 3 et 5 restent donc affectés au VLAN 1
- On considère que le **VLAN 1** est le **VLAN invité**
  - C'est à dire le VLAN par défaut lorsque des ports ne sont pas affectés de façon explicite à un VLAN



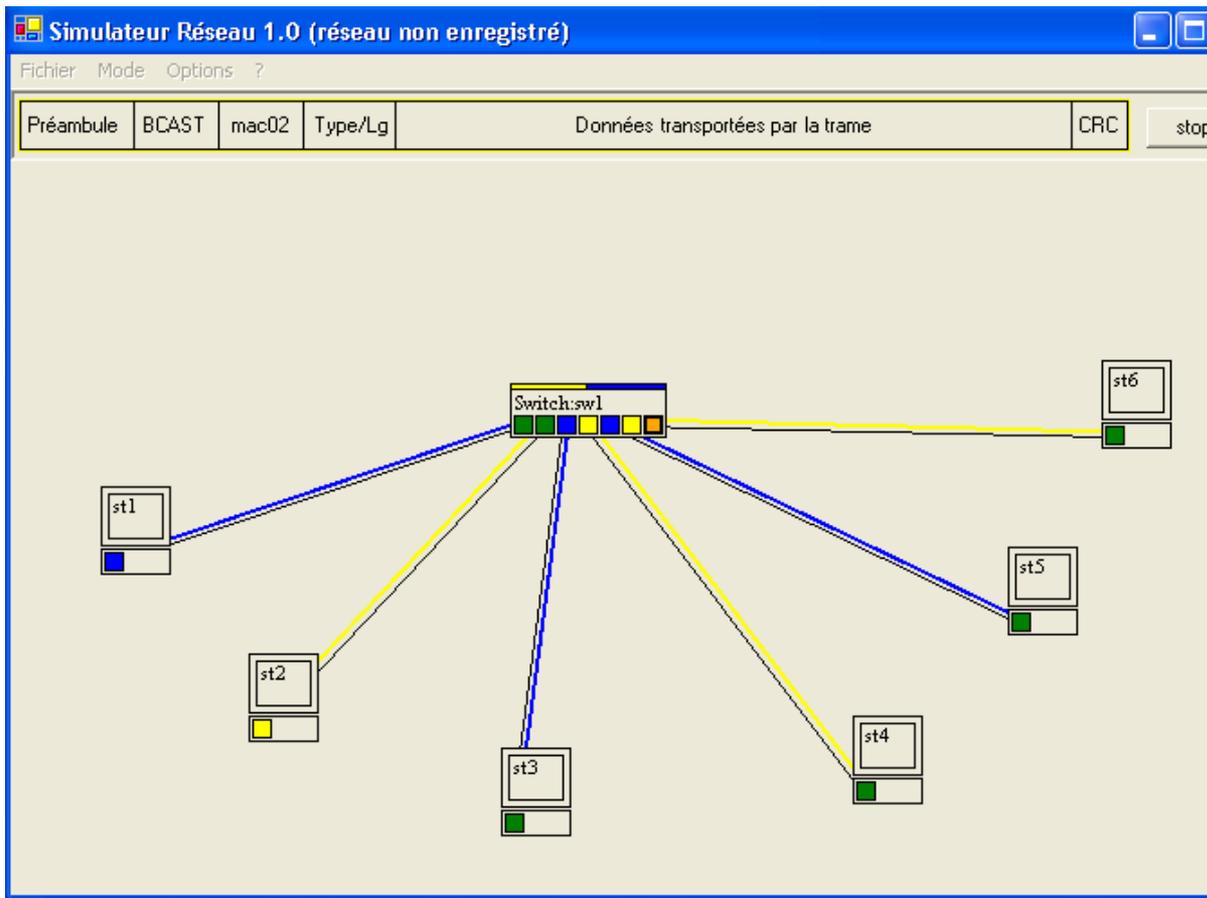
- Fonctionnement du commutateur avec le VLAN de niveau 1.
  - On envoie une trame de broadcast à partir de st1
    - *st3 et st5 reçoivent*
    - *Le commutateur a reçu une trame sur un port affecté au VLAN 1 il la transmet sur tous les ports affectés au VLAN 1*
  - On envoie une trame unicast de st5 vers st3.
    - *st3 et st1 reçoivent car le commutateur ne sait pas où se trouve st3 qui n'a pas encore émis de trame*
    - *Le commutateur a associé st1 au port numéro 1 mais il ne présuppose pas qu'il n'y a qu'un seul poste connecté au port numéro 1 (il pourrait y avoir un concentrateur ou un autre commutateur)*
  - On envoie une trame unicast de st1 vers st2
    - *St3 et st5 reçoivent la trame*
    - *St2 est sur un port affecté au VLAN 2 mais le commutateur l'ignore car st2 n'a pas encore émis de trame*
    - *Ce cas est impossible dans la réalité car un poste qui démarre enverra au minimum une trame « arp gratuit » pour tester l'unicité de son adresse IP*

## – Fonctionnement (suite)

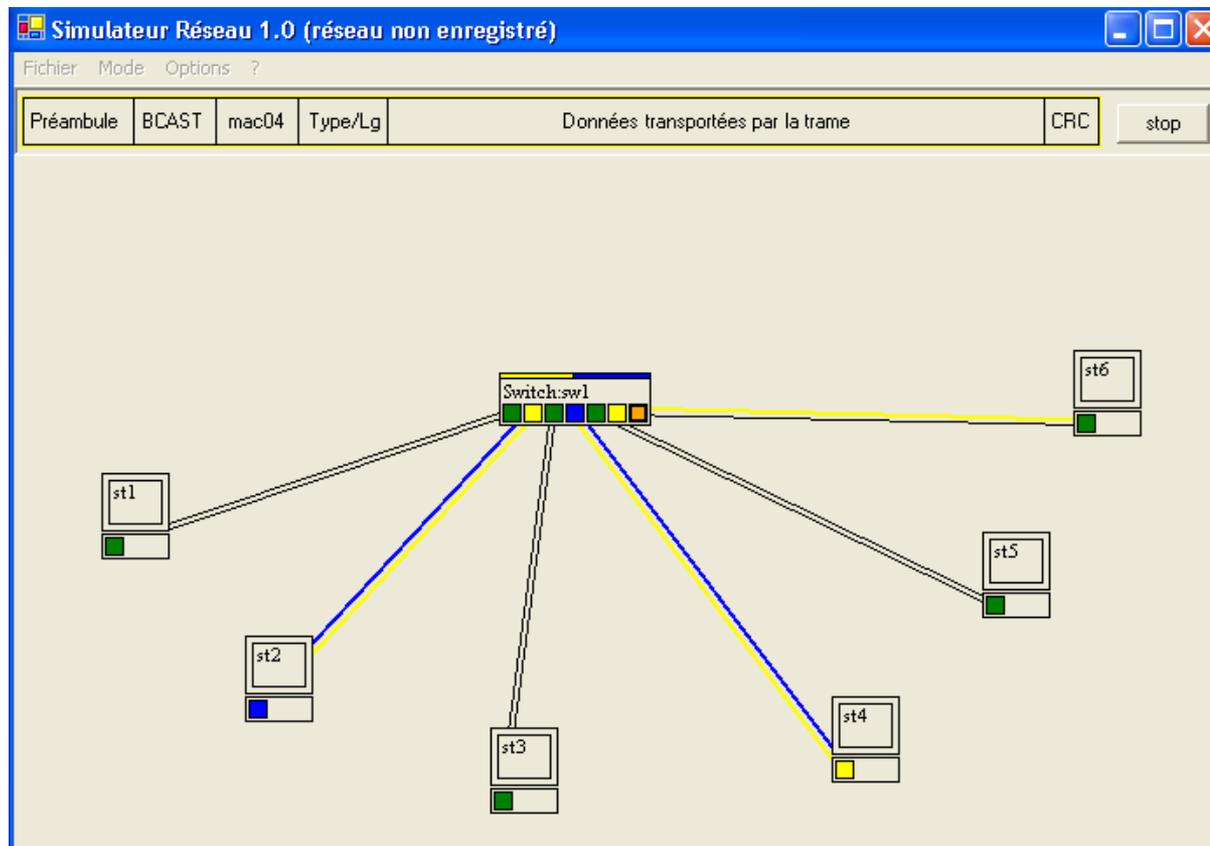
- On coche l'affichage des messages de réception
- St2 envoie une trame vers st1
  - *La trame est perdue*
  - *Elle n'est pas réémise par le commutateur car le commutateur a associé st1 au port 1 qui est affecté au VLAN 1 alors que la trame lui est parvenue par le port 2 affecté au VLAN 2*
  - *Ce cas est impossible dans la réalité car pour obtenir une adresse MAC la carte réseau a du procéder au préalable à un échange (ARP par exemple)*
- Recommencer cet envoi en demandant le « traçage » du switch. Ne pas oublier de choisir le type de simulation « pas à pas »



- Fonctionnement (suite 2)
- *Montrer les domaines de diffusion*
  - Se mettre en mode « trame réelle »
  - On envoie une trame de broadcast à partir de st1 et de St2
    - *On voit bien apparaître les deux domaines de diffusion*



- *Toujours en mode « frame réelle » , on coche la case « full duplex. »*
- *On envoie une frame de broadcast à partir de st2 et de st4*
  - *Les deux broadcast se croisent sur les paires st2 et st4 puis se succèdent sur les paires st6*
  - *St1 st3 et st5 ne sont pas concernés par l'échange*

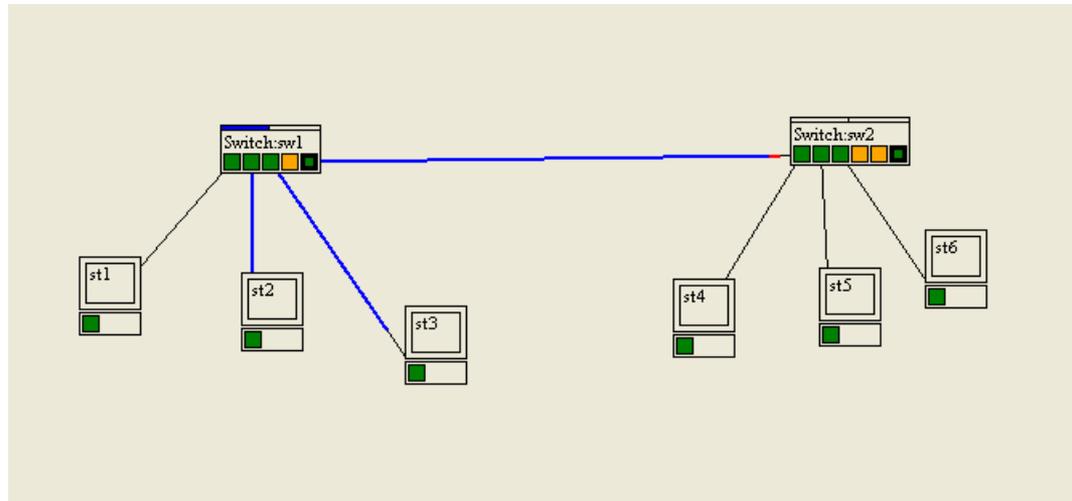


## ■ *Questions*

- *Relever les destinataires de la trame*
- *Relever les lecteurs de la trame*
- *Anticiper ou Expliquer les résultats*
- *Comment le commutateur sait-il à quel VLAN est raccordée une station ?*
- *Que se passe-t-il si un port n'est pas associé à un VLAN ?*
- *Qui crée la table Vlan/ port ?*
- *Qui crée la table Mac / port ?*

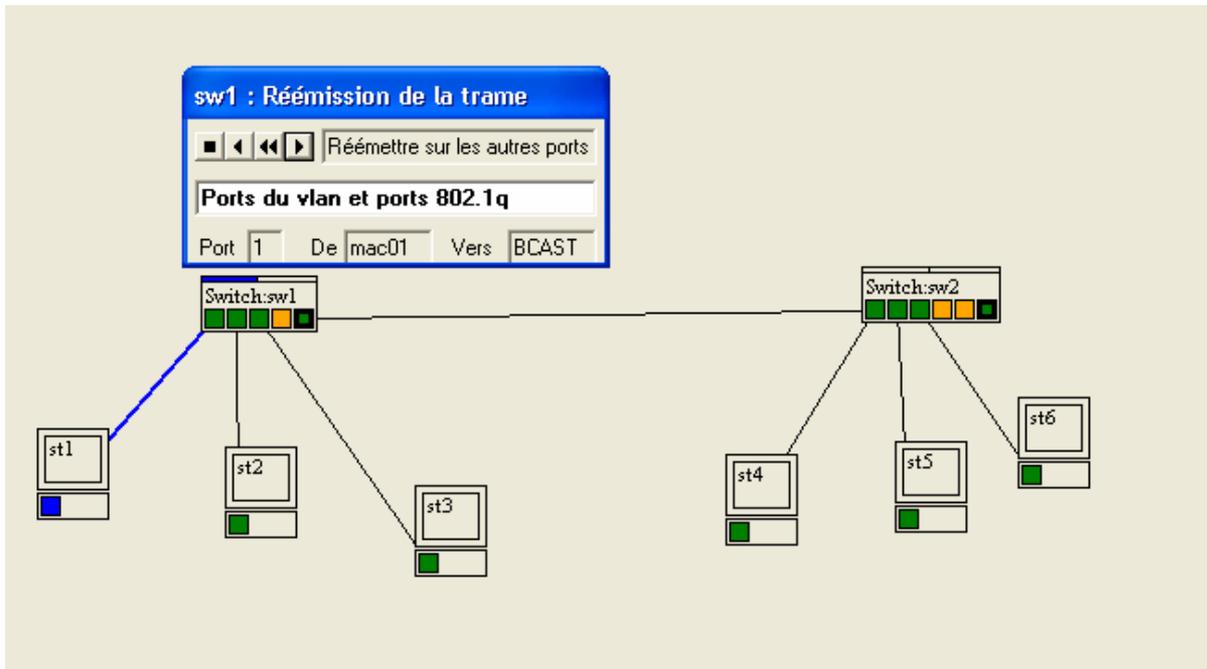
## 2. Gestion de deux VLAN de niveau 1 sur deux commutateurs avec 802.1q.

- On utilise le réseau deuxswitchvlan1.xml
  - Il s'agit de montrer le comportement de VLANs associés à un seul commutateur avec l'utilisation du port 802.1q
  - Le switch1 gère le VLAN1
  - Le switch2 gère le VLAN2
  - Les deux commutateurs sont interconnectés par un port 802.1q

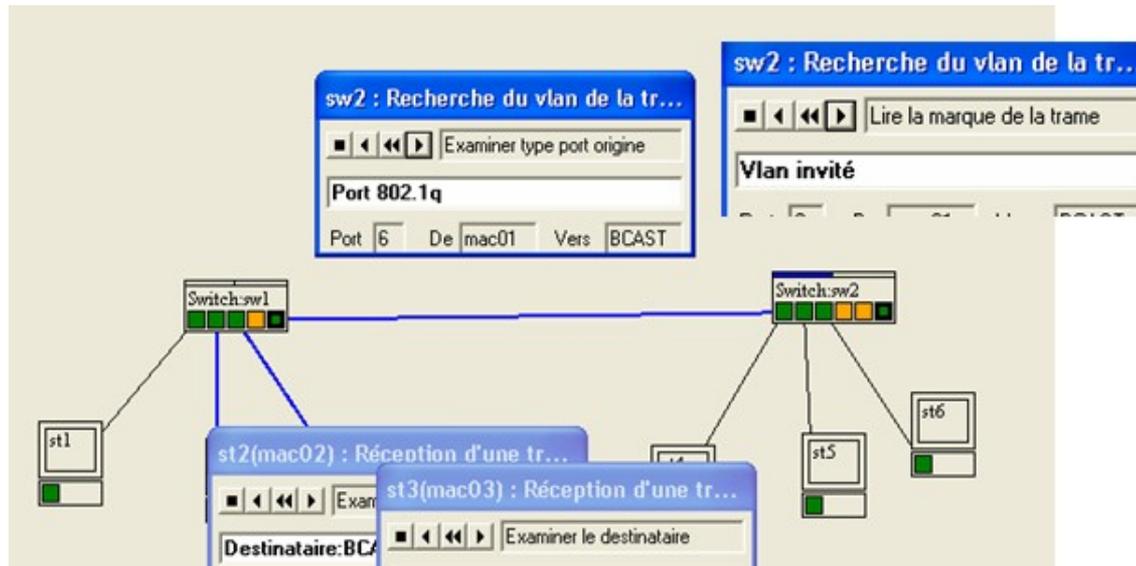


- *st1 envoie une trame de diffusion*
- *St2 et st3 reçoivent*
- *Sw2 reçoit aussi mais ne transmet pas à st4 st5 et st6*
- *Le type de simulation est automatique*
- *St2 et st3 reçoivent*
- *Sw1 transmet à sw2 par son port 802.1q en marquant la trame (petite marque rouge en début du tracé de la trame)*
- *sw2 ne sait pas par quel port la trame a été émise sur sw1 la seule façon pour lui d'associer cette trame à un VLAN est la marque (le tag) qui a été placée par le port 802.1q sur la trame*
- *En associant cette trame à un VLAN sw2 peut décider de ne pas la diffuser vers st4 st5 et st6, plus précisément il la diffuse aux ports qui sont associés au même VLAN que la marque*

- Rejouons la séquence en « traçant » les deux commutateurs. Et choisissons le type de simulation « pas à pas »

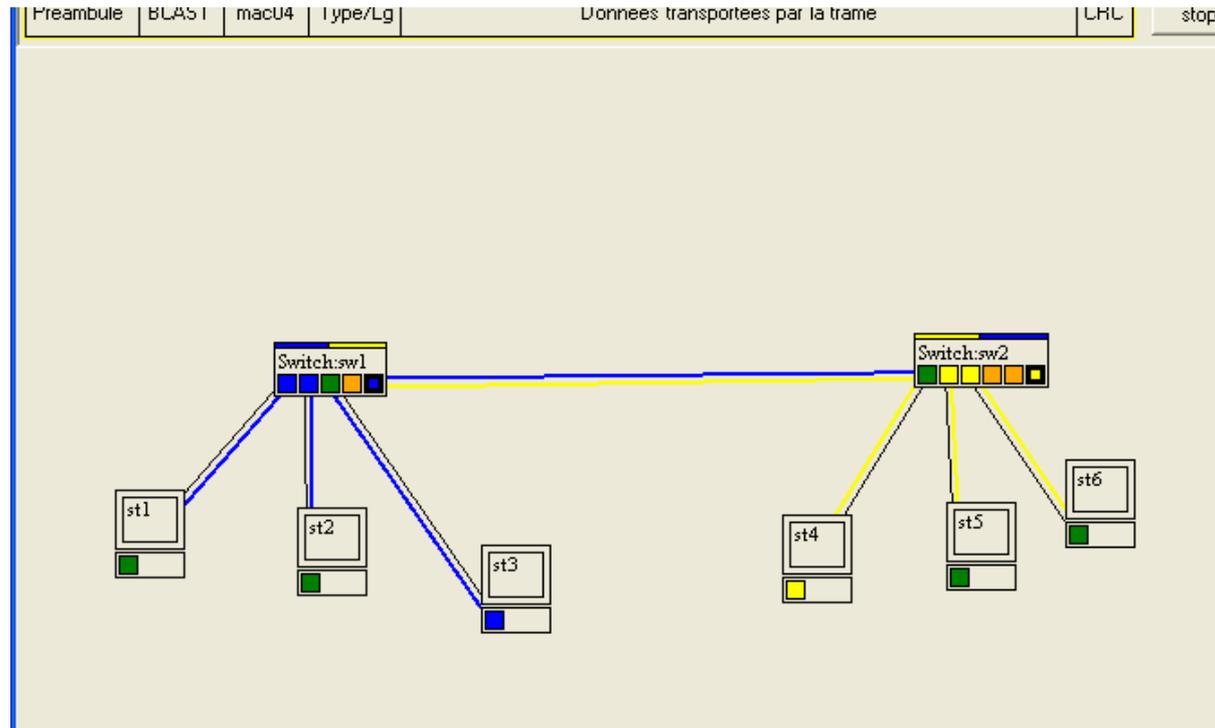


- on constate que le commutateur sw1 marque la trame et que le commutateur sw2 lit la marque de cette trame.



– Mettons nous maintenant en type de simulation « *trame réelle* ».

- st1 et st4 envoie simultanément une trame de broadcast.  
*Les trames sont diffusées dans chaque VLAN et sur les ports 802.1q après avoir été marquées*
- *Chaque commutateur associe ainsi la trame à un VLAN et peut prendre la bonne décision concernant son acheminement*

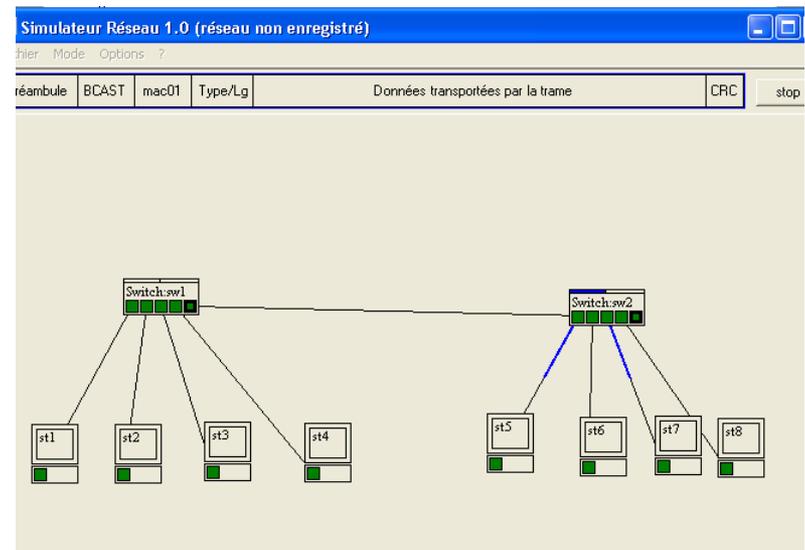


## ■ Questions

- Relever les destinataires de la trame
- Relever les lecteurs de la trame
- Anticiper ou Expliquer les résultats
- Un port 802.1q est-il affecté à un VLAN ?
- Qu'est-ce qu'un lien « trunk » (trunk link) ?
- Pourquoi le marquage de la trame est-il indispensable ?
- Les cartes réseaux des postes lisent-elles la « marque » 802.1q ?
- Une carte réseau peut-elle se trouver sur un lien « trunk link » ?

## ■ Répartition d'un VLAN de niveau 1 sur plusieurs commutateurs

- On utilise le fichier deuxswitchvlan1réparti.xml
- Il s'agit de montrer le comportement de VLAN répartis sur plusieurs commutateurs.
  - Ici les VLAN 1 et 2 sont répartis sur sw1 et sw2
  - Sur sw1 les stations 1 et 3 sont sur le VLAN 1 et les stations 2 et 4 sur le VLAN 2
  - Sur sw2 les stations 5 et 7 sont sur le VLAN 1 et les stations 6 et 8 sur le VLAN 2
  - Cela donne respectivement st1 st3 st5 et st7 sur le VLAN 1.
  - St2 st4 st6 et st8 sont donc sur le VLAN 2



## – Manipulation

- On se met tout d'abord en « type de simulation automatique »
- st1 envoie une trame de broadcast
  - *St3 st5 et st7 reçoivent la trame*
  - *Sw1 a transmis une trame marquée à sw2. sw2 a lu la marque et a commuté la trame vers les bons ports*
- St5 envoie une trame unicast vers st1
  - *st1 reçoit la trame*
  - *Sw2 a transmis uniquement sur le port 802.1q auquel il a associé st1 (on peut éditer la table mac/port) Il marque la trame avec le VLAN du port émetteur soit le VLAN 1*
- *sw1 récupère la trame, lit la marque, lit l'adresse du destinataire et commute sur le bon port à l'aide de sa table mac/port*
- St6 envoie une trame unicast vers st1
  - *Sw2 ne connaît pas le VLAN du port sur lequel est connecté st1, il transmet donc à sw1 après avoir marqué la trame avec le VLAN de l'émetteur soit le VLAN 2*

## ■ Questions

- Comment le deuxième commutateur associe-t-il la trame à un VLAN ?
- Les deux commutateurs s'échangent-ils les tables port / vlan ?