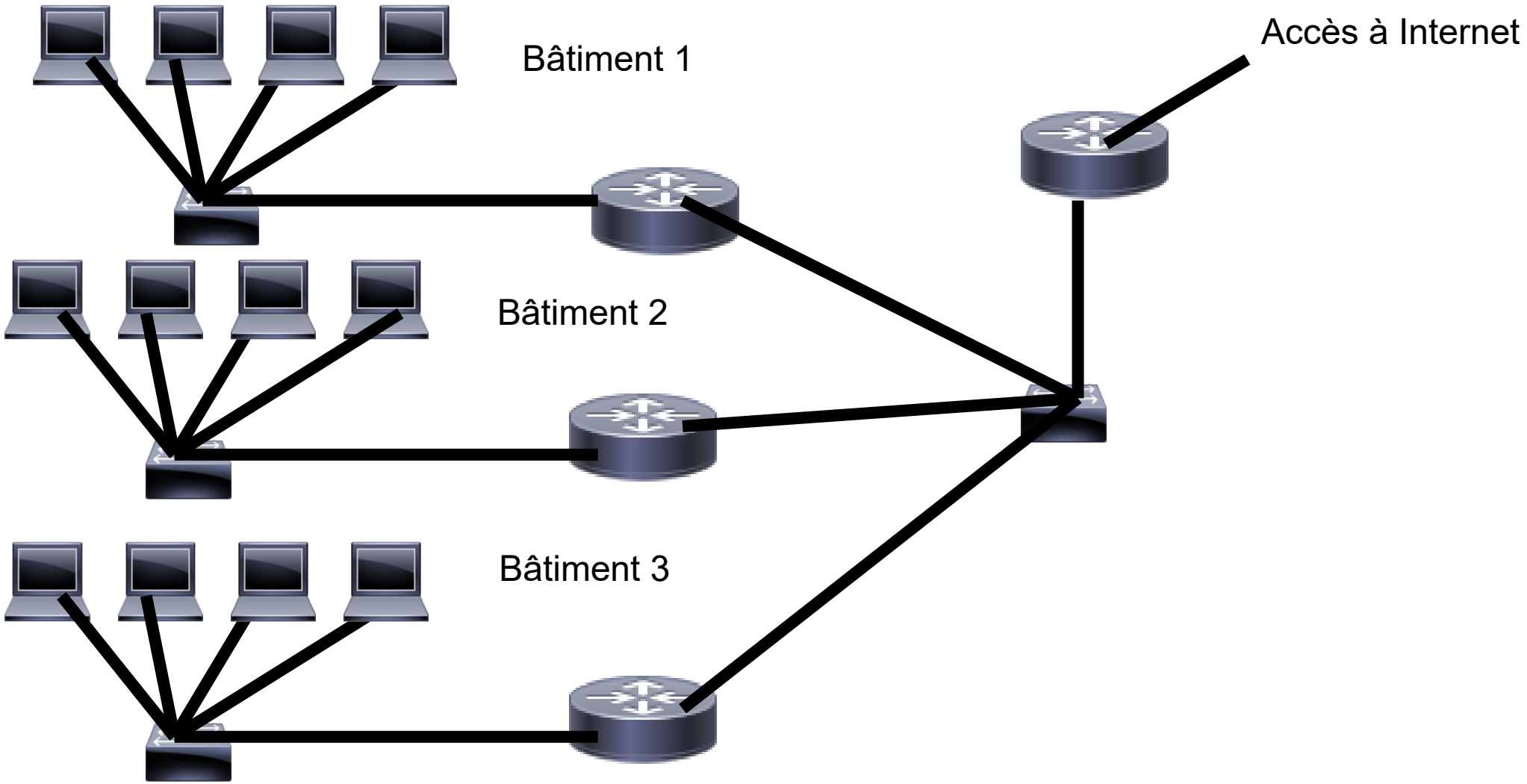
La commutation et les VLANs

Xavier Merrheim

Les gros réseaux et le routage

- Imaginons une structure comme un campus universitaire : une centaine de bâtiments répartis sur quelques km^2.
- Dans chaque bâtiment, des réseaux informatiques sont apparus.
- Une épine dorsale a été créée pour relier entre eux tous les bâtiments et pour connecter tous ces réseaux entre eux et à Internet.
- Historiquement parlant , ce sont des routeurs IP qui ont relié entre eux ces bâtiments

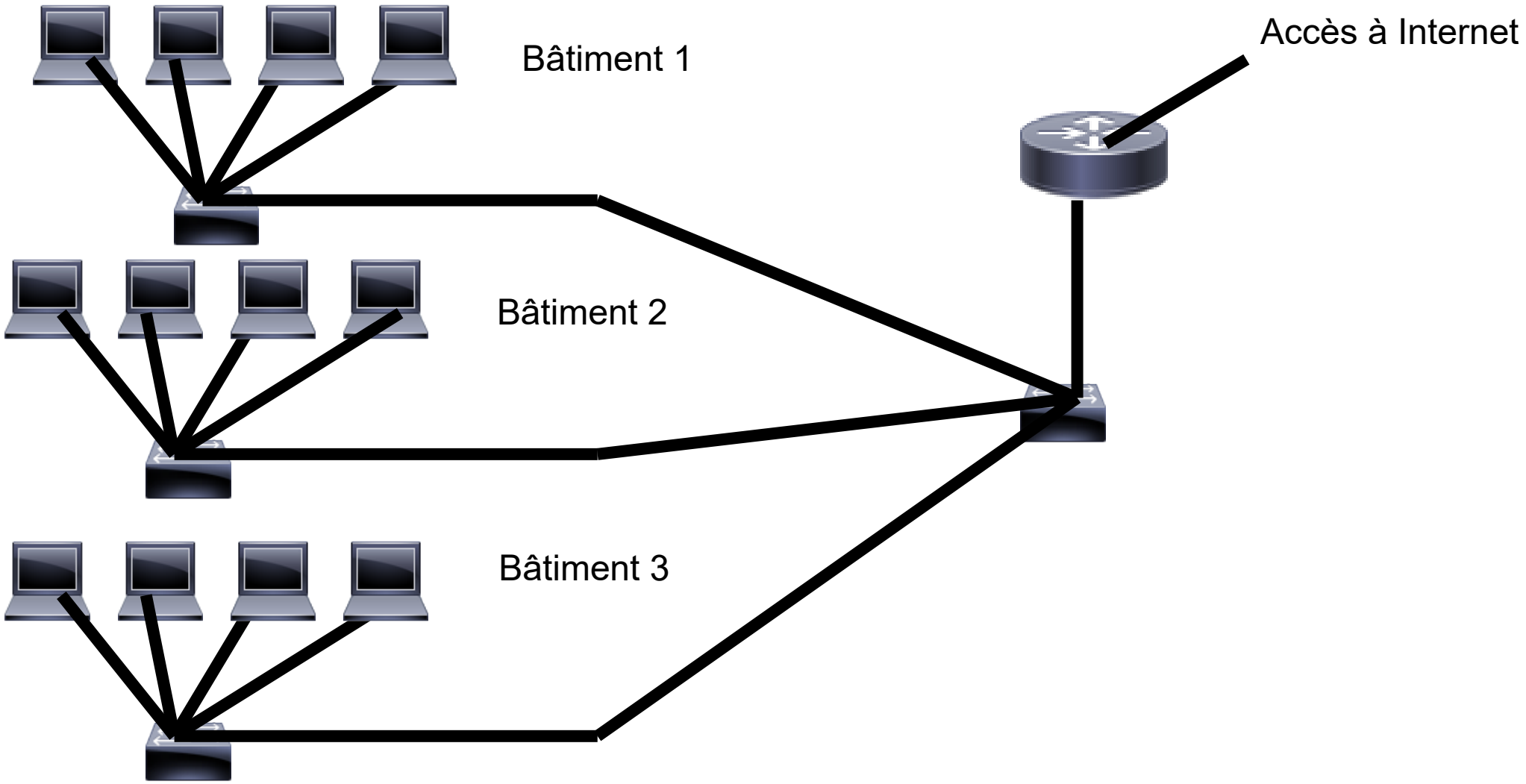
Architecture backbone



Problème

- Il est vite apparu un problème, la commutation est beaucoup plus rapide que le routage IP.
- On préfère donc connecter entre eux des commuteurs et n'avoir qu'un seul routeur pour accéder à Internet.
- Là où c'était possibles, on a relié des commutateurs entre eux.
- La situation a peu à peu évolué vers une nouvelle architecture.

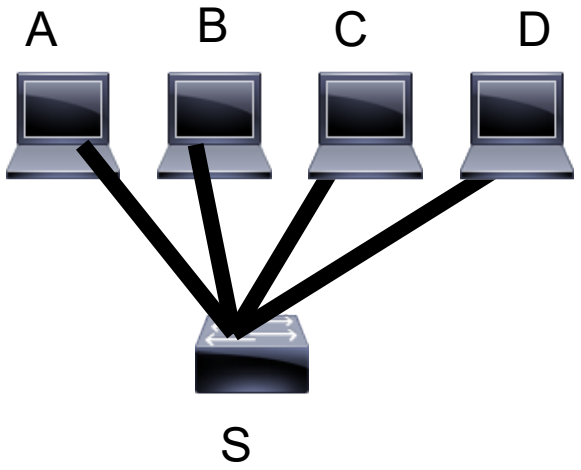
Interconnection de switchs



Interconnection de switchs

- Nous allons donc étudier la commutation.
- Ensuite viendra l'interconnection de switchs
- Puis les problèmes liés à la résistance aux pannes.
- Et enfin la nécissité de construire des VLANs.

Commutation éthernet



A, B,C et D sont 4 machines connectées à un switch S. Elles ont chacune une adresse MAC sur 48 bits écrite sur leur carte éthernet.

Que se passe-t-il si A envoie une trame éthernet à C ?

A envoie une trame éthernet à C

- A envoie une trame éthernet avec comme adresse mac expéditeur sa propre adresse mac et comme adresse mac destinataire l'adresse mac de C.
- La carte éthernet de A envoie cette trame à S.
- S ne sait pas où est la machine C ?

Que fait S ?

- S va envoyer cette trame à tout le monde sauf à l'expéditeur A. S envoie donc la trame à B, C et D.
- B va donc recevoir cette trame, s'apercevoir en lisant l'adresse mac destinataire qu'elle ne lui est pas destinée.
- C va donc recevoir cette trame, s'apercevoir en lisant l'adresse mac destinataire qu'elle est pour lui et l'envoyer à son système d'exploitation.
- D va donc recevoir cette trame, s'apercevoir en lisant l'adresse mac destinataire qu'elle ne lui est pas destinée.

Oui mais S est malin ...

- Le commutateur S va essayer de répérer où sont situées les machines en fonction de leur adresse MAC.
- Les ports éthernet d'un commutateur sont numérotés. Imaginons que :
  - A soit sur le port 1
  - B soit sur le port 2
  - C soit sur le port 3
  - D soit sur le port 4

Optimisation

- S se dit qu'il a intérêt à répérer où sont les machines.
- Il va essayer de remplir sa table des adresses MAC. C'est une table que le commutateur remplit automatiquement et qui indique les adresses MAC des machines et les numéros de ports éthernet associés.
- Lorsque S reçoit la trame éthernet en provenance de A, il sait que A est sur le port 1.
- Il va donc mettre dans sa table des adresses MAC.

MAC de A – port 1

A quoi ça sert ?

- Imaginons maintenant que C réponde à A. C va envoyer une trame éthernet avec comme expéditeur MAC de C et comme destinataire MAC de A.
- La trame va être envoyée à S sur le port éthernet 3.
- S va lire sa table des adresses MAC et va s'apercevoir que A est sur le port 1 du switch.
- S va envoyer cette trame uniquement sur le port 1 à A.
- Il va va en profiter pour mettre dans sa table des adresses MAC.

MAC de C – port 3

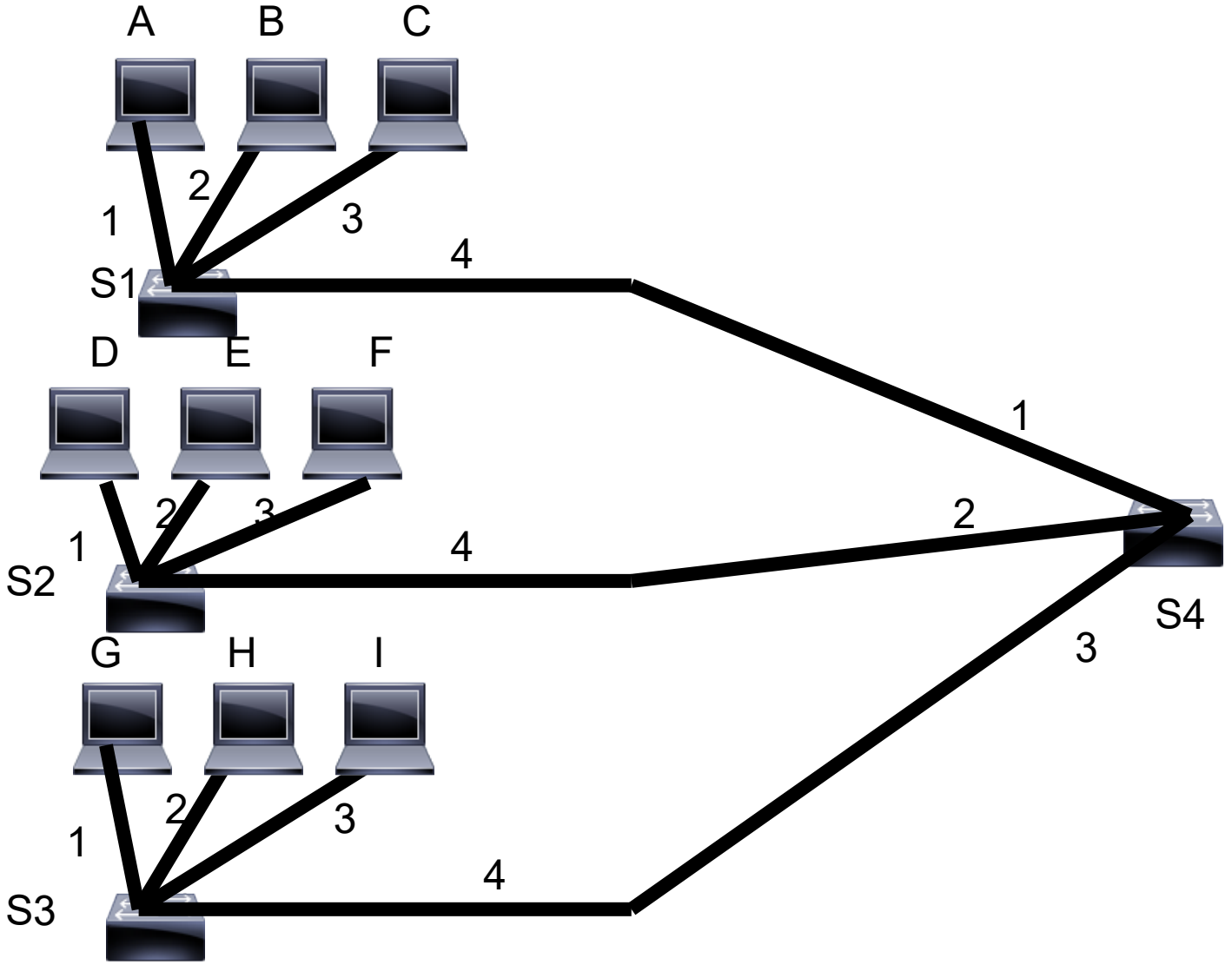
Avantage de cet algorithme de commutation

- L'administrateur n'a rien à confiigurer, c'est auto-apprenant.
- Dès qu'une machine a communiqué une fois sur le réseau, elle se trouve dans la table des adresses MAC.
- C'est très efficace à implémenter de façon hardware !

Remarque

- Les trames de diffusion éthernet dont l'adresse MAC destinataire est FFFF FFFF FFFF sont envoyées à tout le monde sauf à l'expéditeur.

Et maintenant on interconnecte des switchs entre eux



Tables des adresses MAC

- Chacun des 4 switchs S1, S2, S3 et S4 ont leur propre table des adresses MAC.
- Deux adresses MAC différentes peuvent être sur le même port d'un switch !
- Et le réseau va apprendre rapidement où sont les différentes machines.

Exercice

- On imagine que :
  - A, B, C et S1 soient sur les port 1, 2, 3, 4 de S1
  - D, E, F et S2 soient sur les port 1, 2, 3, 4 de S2
  - G, H, I et S3 soient sur les port 1, 2, 3, 4 de S3
  - S1, S2 et S3 soient s:r les port 1, 2, 3 de S4
- Dans le scénario suivant :
  - A envoie une trame éthernet à D
  - B envoie une trame éthernet à C
  - C envoie une trame éthernet à A
  - D envoie une trame éthernet à A
  - G envoie une trame éthernet à B
- Décrivez l'évolution des tables des adresses mac, pour chaque trame et chaque switch concerné indiquez sur quels ports il envoient la trame et au final la liste des machine la recevant.

Trames	S1	S2	S3	S4	Machines
A -> D	macA-1				
	S1->2,3,4			macA-1	B,C
				S4->2,3	
		macA-4	macA-4		
		S2->1,2,3	S3->1,2,3		D,E,F,G,H,I
B -> C	macB-2				
	S1->1,3,4			macB-1	A,C
				S4->2,3	
		macB-4	macB-4		
		S2->1,2,3	S3->1,2,3		D,E,F,G,H,I
C -> A	macC-3				
	S1->1				A

Trames	S1	S2	S3	S4	Machines
D -> A		macD-1 S2->4			
				macD-2 S4->1	
	macD-4 S1-> 1				A
G -> B			macG-1 S3->4		
				macG-3 S4->1	
	macG-4 S1-> 2				B

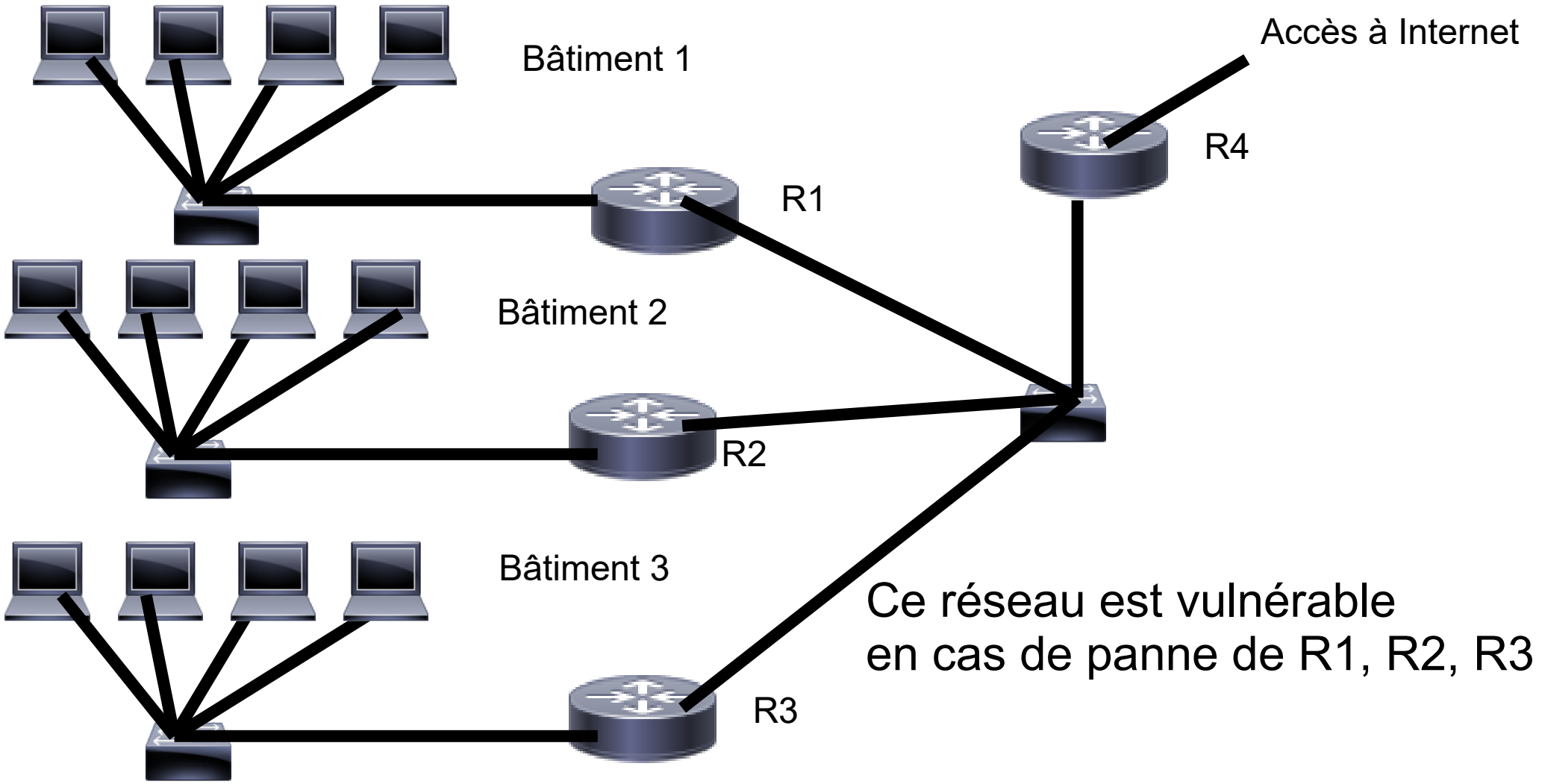
Interconnection

- Rapidement, les différents switchs remplissent leur tables des adresses MAC et savent commutés les trames éthernet correctement.
- Ceci fonction parfaitement tant que l'interconnexion de switchs est un arbre.

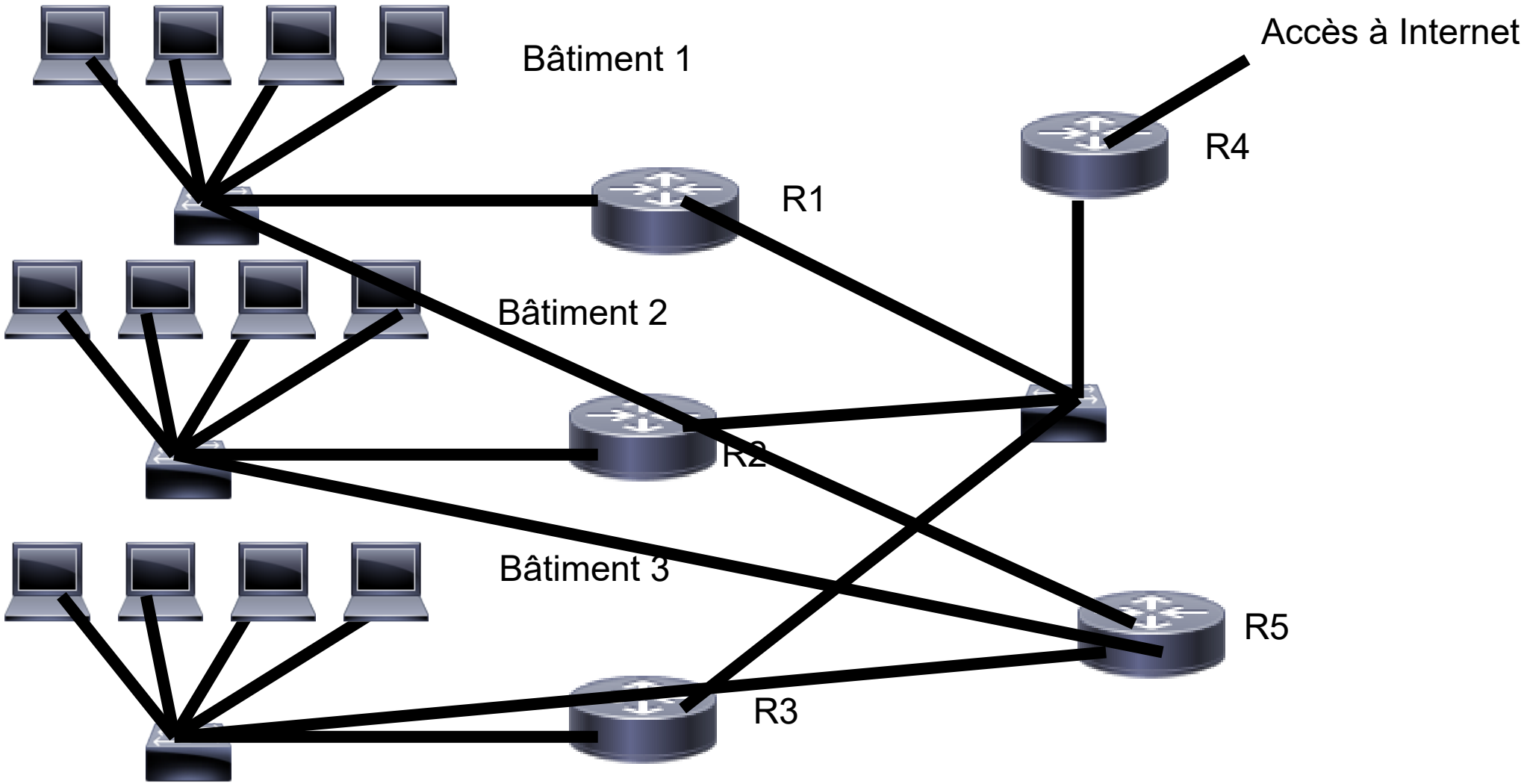
Oui mais ...

- De plus en plus, on veut des réseaux résistants aux pannes.
- En effet, le coût d'une panne devient de plus en plus important !
- Voyons tout d'abord comment cela s'est traduit pour le routage.

Architecture backbone



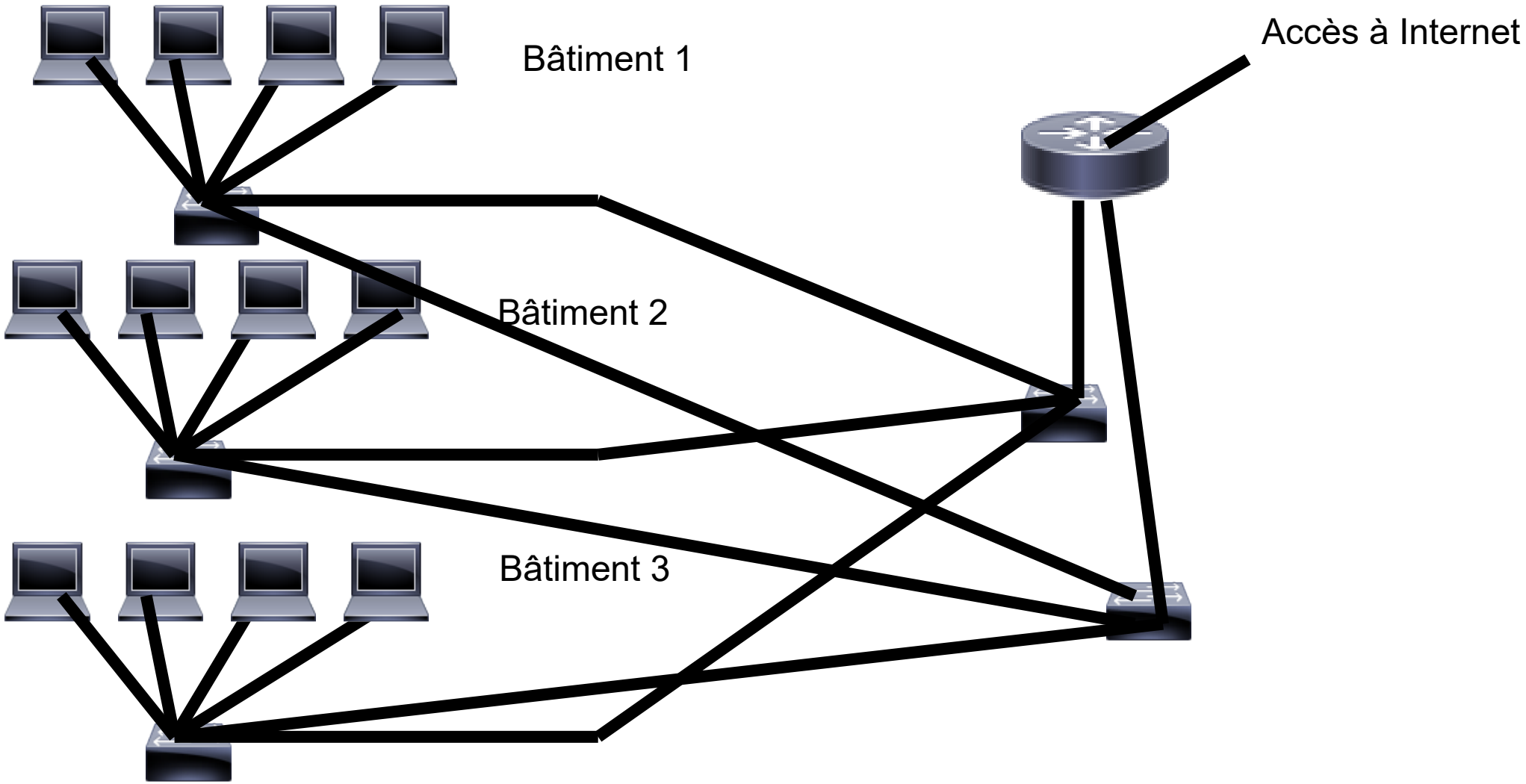
Solution améliorée



Principe de base pour améliorer la résistance aux panne du routage

- On va rajouter des routeurs redondants sur le réseau.
- On a rajouté un routeur R5 à notre réseau.
- Il faut maintenant configurer les routeurs pour qu'ils utilisent un algorithme de routage dynamique comme RIP et le réseau sera meilleur.

Interconnection de switchs améliorée



Commutation avec des switchs redondants

- L'interconnection de switchs est désormais un graphe et non un arbre. Il y a des boucles dans l'interconnection de switchs.
- Il est facile de voir qu'en utilisant l'algorithme précédent sur un tel graphe certains trames vont être dupliquées.
- Il y aura 2 trames puis 4,8,16, ... jusqu'à l saturation complète du réseau.
- Tel quel l'algorithme de commutation n'est utilisabe uniquement sur un arbre. Dès qu'il y a une boucle, c'est la catastrophe !

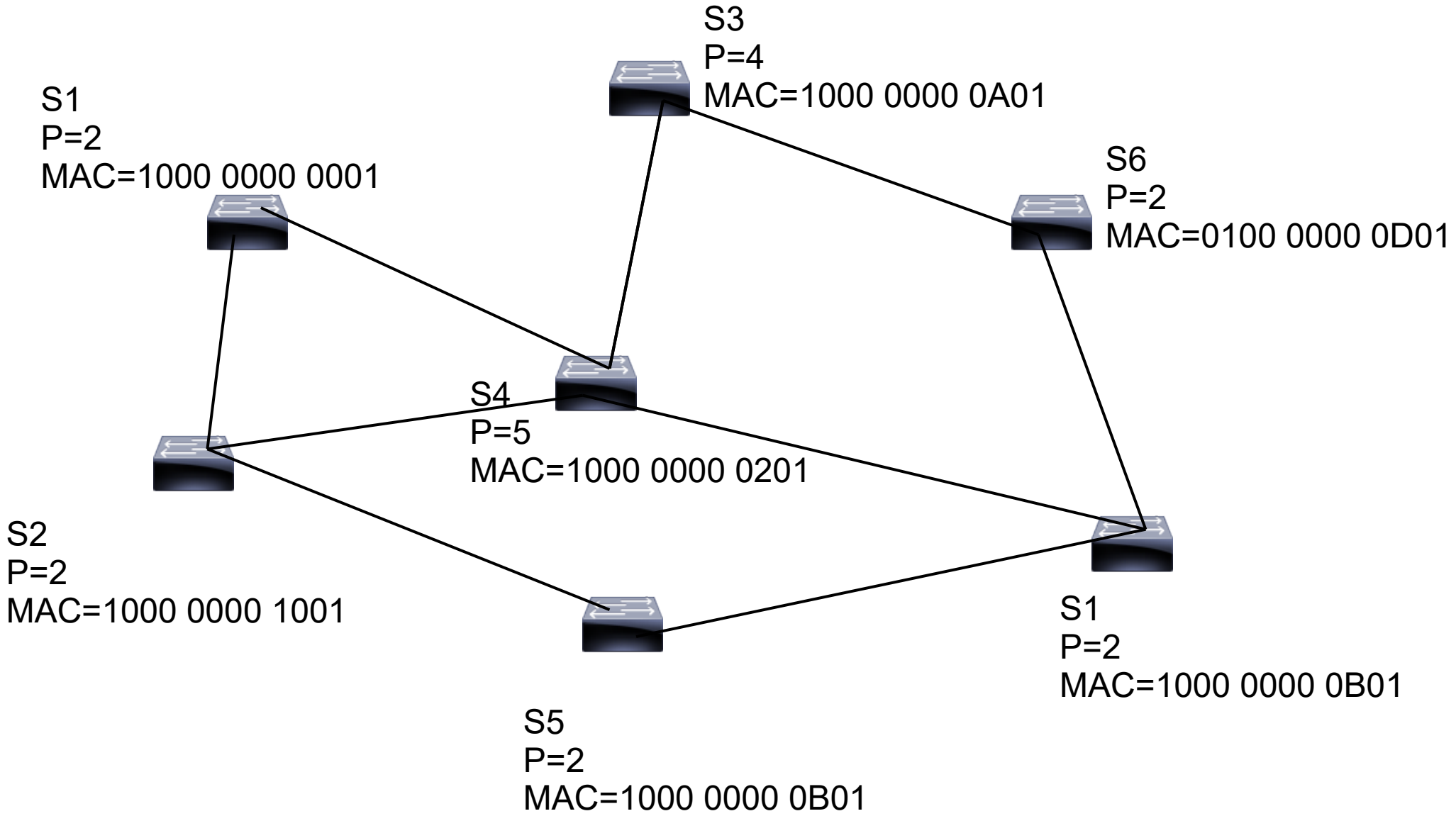
Solution

- Heureusement, une solution existe. On va lancer sur chaque switchs un algorithme appelé STP : Spanning Tree Protocol. Cet algorithme permet à partir d'une interconnection de switchs quelconque de désactiver certains liens pour en extraire un arbre.
- Sur l'arbre obtenu, la commutation fonctionnera parfaitement.
- On a améliorer la résistance aux pannes.
- En cas de paanne, STP se relancera automatiquement et trouvera un nouvel arbre !

Algorithme du spanning tree étape 1

- On attribue à chaque switch une adresse MAC et l'administrateur fixe une priorité sur chaque switch entre 0 et 65535. Plus on a une priorité faible plus on est prioritaire.
- Première étape : les switchs s'échangent des trames pour élire le commutateur Root. On va donc élire le commutateur avec la plus petit priorité et, en cas d'égalité, la plus petit adresse MAC.

Exemple : qui va devenir le commutateur root ?



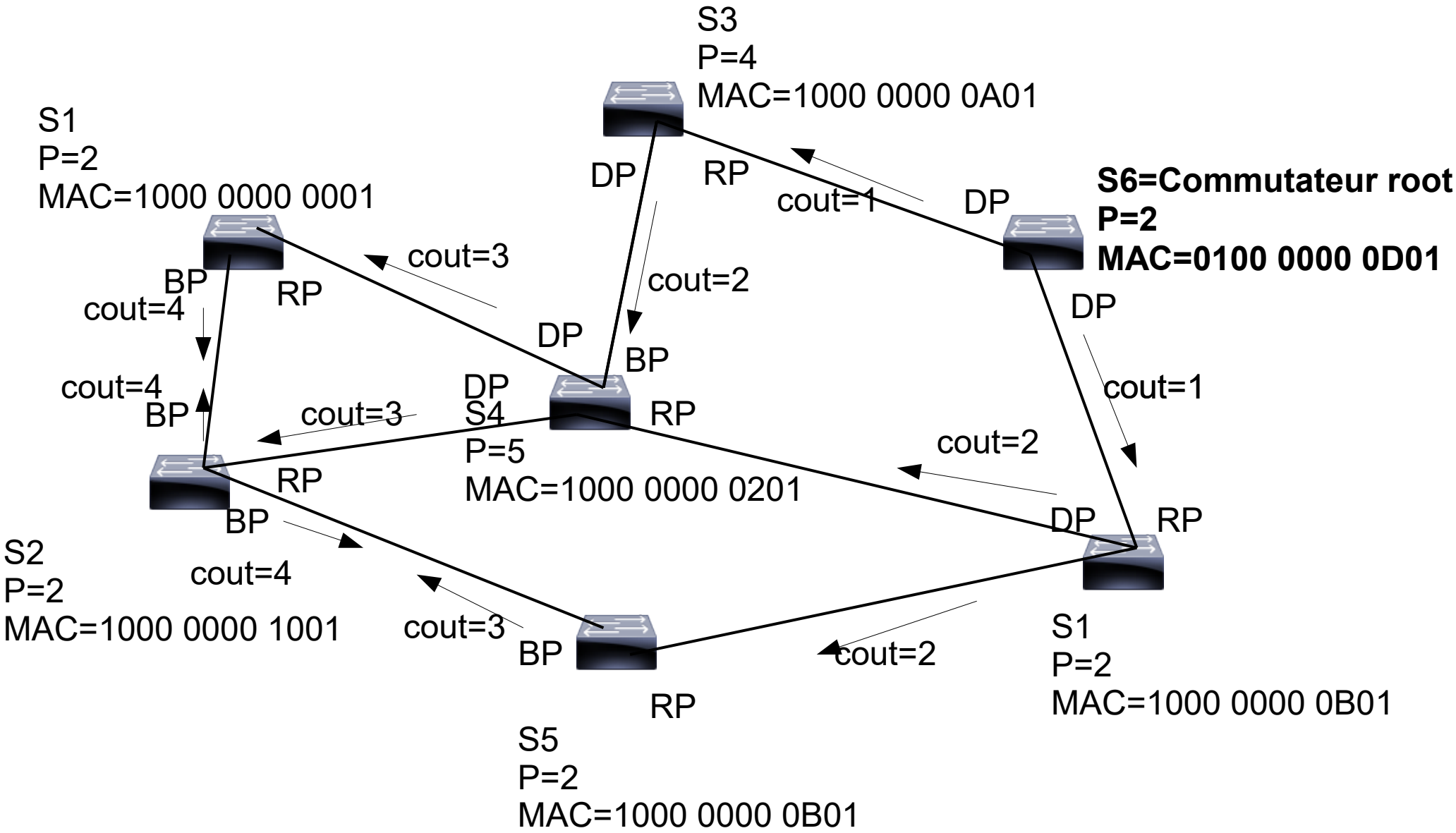
Algorithme du spanning tree étape 2 (1)

- Chaque port éthernet de chaque commutateur va être mis dans un des 3 états suivants :
  - RP : Root Port : ce port désigne le port du commutateur menant au commutateur root avec un chemin de moindre coût. Le coût étant le nombre de segments éthernet traversés.
  - DP : Designated port : c'est un port faisant partie de l'arbre final qui n'est pas un port RP
  - BP : Blocked Port. C'est un port qui a été désactivé

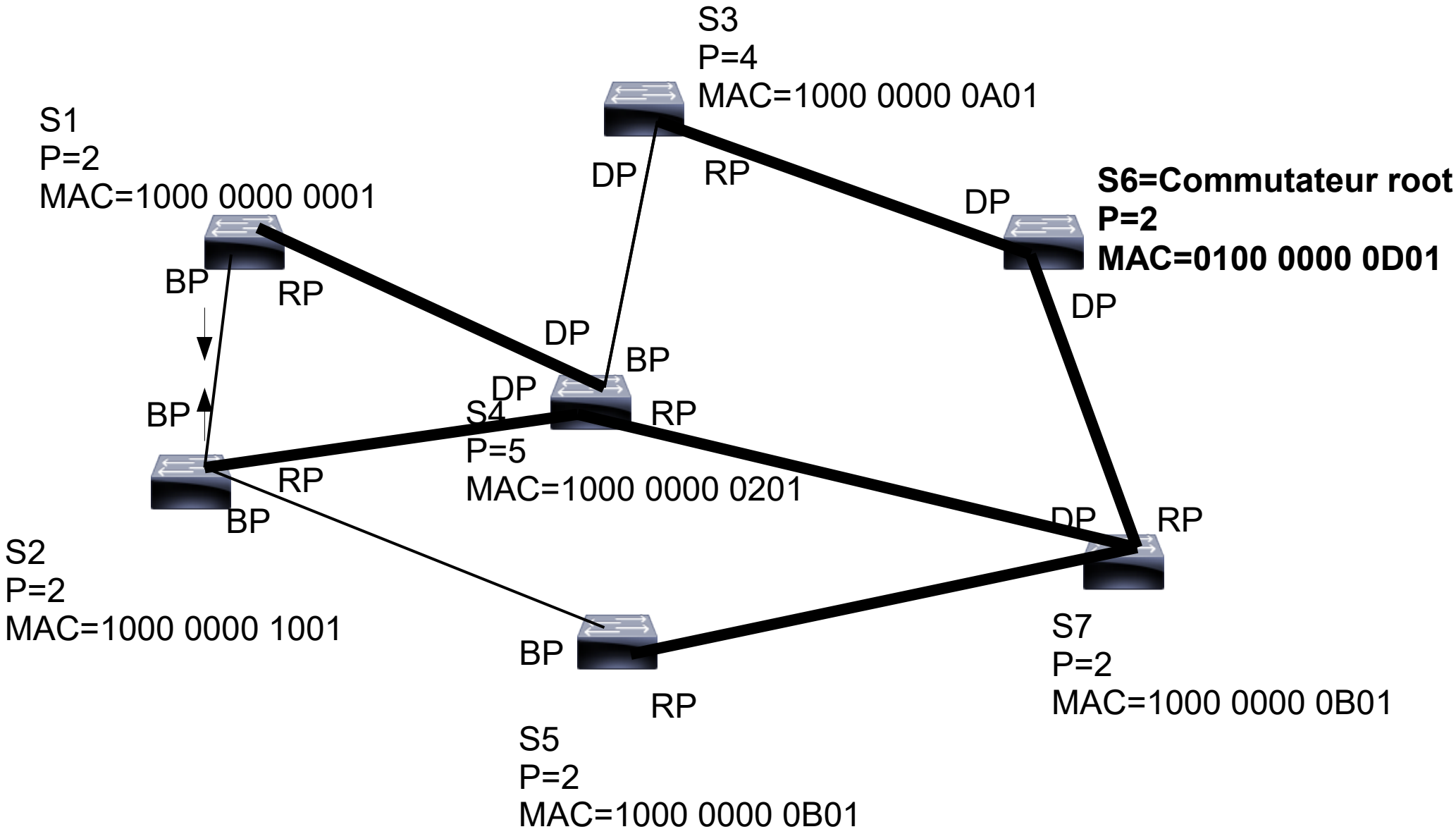
Algorithme du spanning tree étape 2 (2)

- Le commutateur Root va mettre tous ses ports en DP et il va envoyer une trame avec un coût de 1.
- Lorsqu'un commutateur reçoit des trames avec des coûts, il va choisir comme port RP celui de plus petit coût. En cas d'égalité, il choisira le commutateur de plus petite priorité, puis en cas de nouvelle égalité de plus petite adresse MAC. Les autres ports sur lesquels il a reçu une trame avec un coût seront mis en BP.
- Le commutauer va mettre ses autres port en DP et va à son tour envoyer une trame avec un coût N+1

Application



Arbre obtenu après Spanning Tree



Spanning

- On teste régulièrement chacun des liens entre commutateurs.
- En cas de panne d'un lien on relance complètement le Spanning Tree pour obtenir un nouvel arbre.

Regroupement de réseaux éthernet

- On peut donc créer de gros réseaux éthernet où on fera de la commutation.
- C'est beaucoup plus performant que le pur routage IP.
- On peut avoir des liens redondants en utilisant l'algorithme du spanning tree, ce qui améliorera la résistance aux pannes.
- Mais ils reste quelques soucis ....

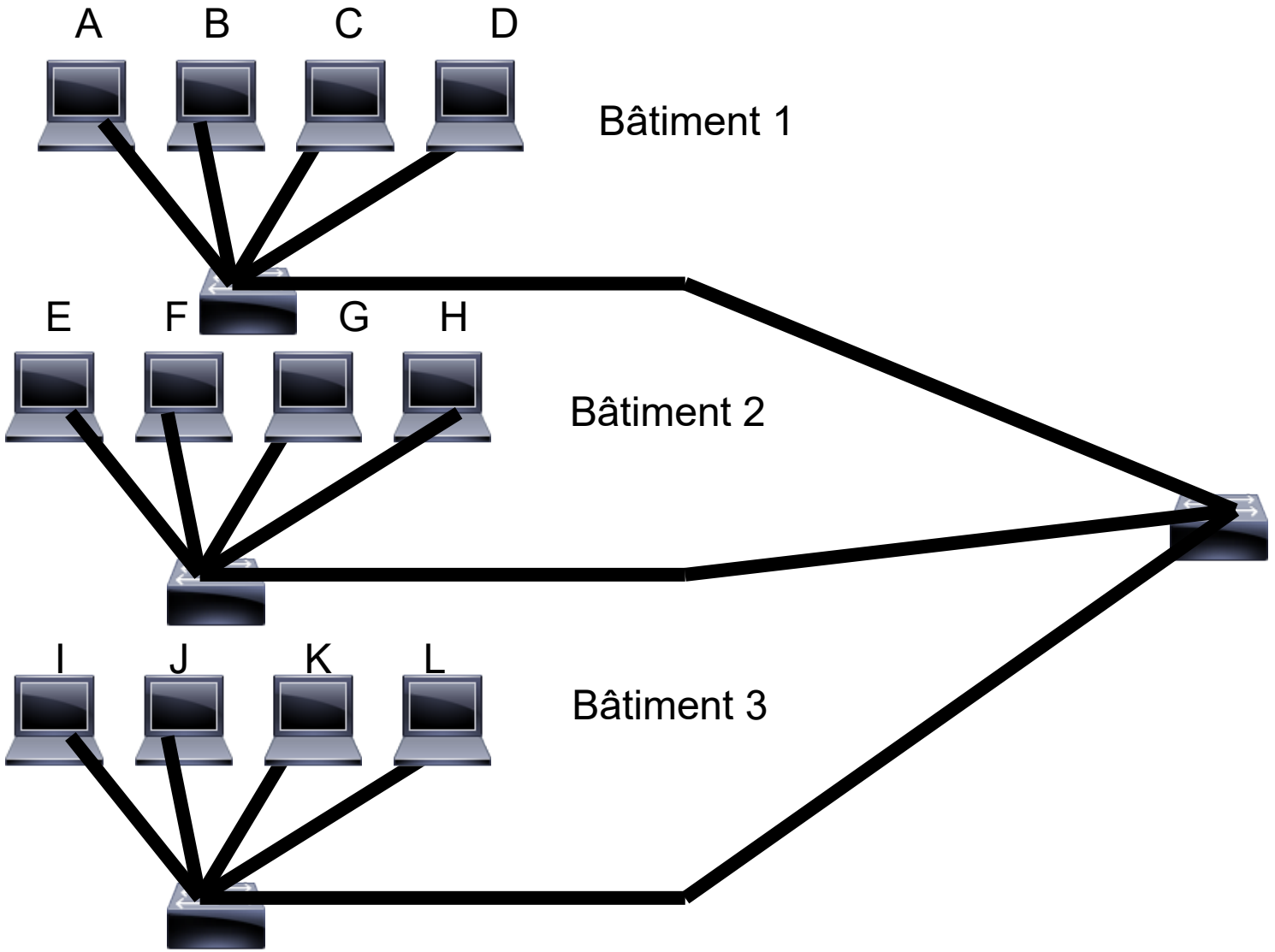
Problème des gros réseaux éthernet

- Imaginons un réseaux de commutation éthernet de plusieurs milliers de machines.
- En cas de trames de diffusions, le domaine de diffusion est trèès grands :
  - Problème de performance
  - Problème de sécurité

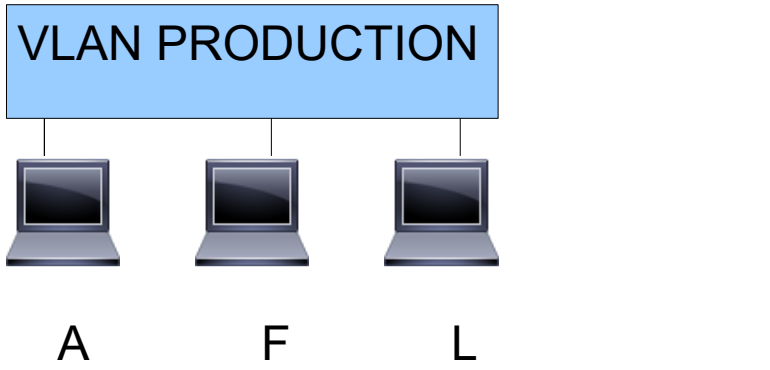
Découpage en VLAN

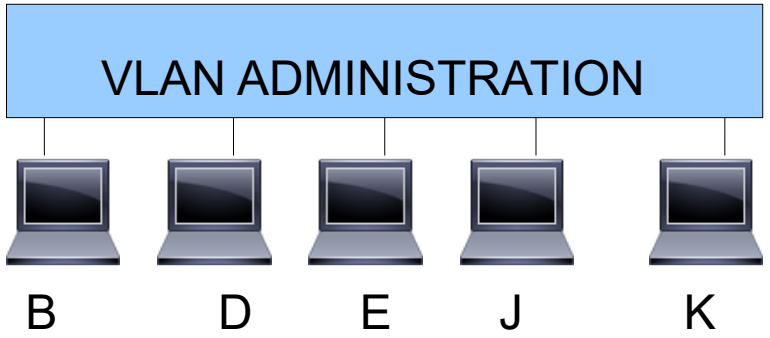
- On va découper notre réseau éthernet en différents VLAN (Virtual LAN).
- Chaque VLAN va se comporter comme un réseau éthernet indépendant.
- Le domaine de diffusion sera réduit à un VLAN.

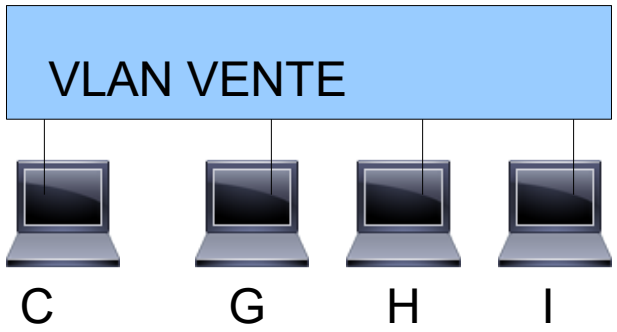
Exemple architecture physique



Exemple architecture logique



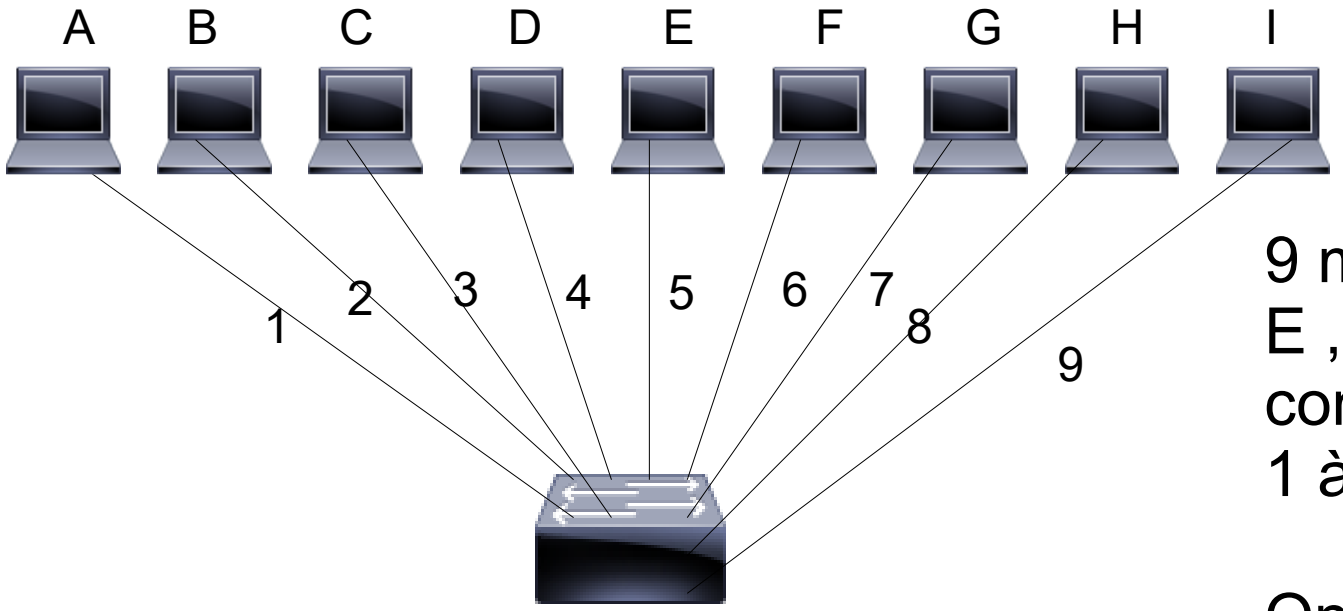




Souplesse VLAN

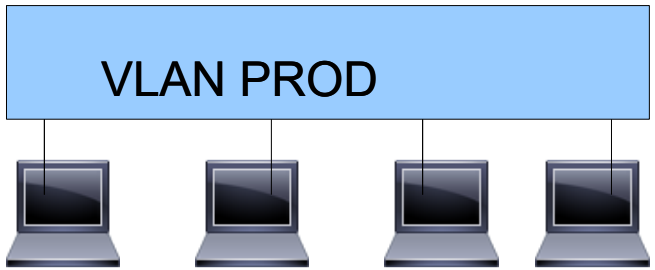
- Il y a indépendante totale entre architecture physique et logique du réseau.
- Le domaine de diffusion est restreint.
- On a gagné en sécurité.

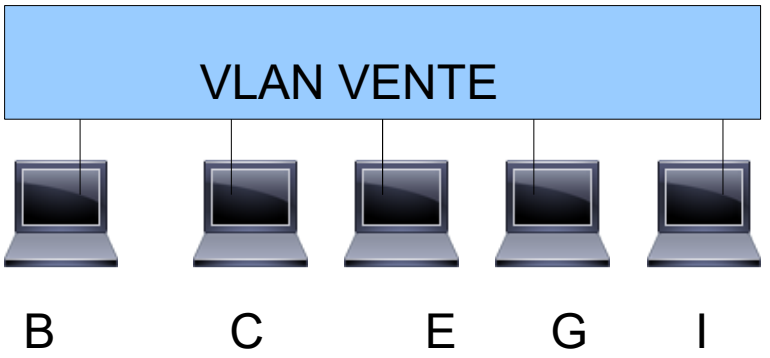
Les VLAN sur des switchs CISCO
Exemple 1



9 machines A, B, C, D, E , F, G, H, et I sont connectées sur les ports 1 à 9 d'un switch CISCO

On veut créer 2 vlans nommés PROD et VENTE





Switch CISCO

- Nous utiliserons des switch CISCO à 24 innterfaces nommées de fastethernet 0/1 à fastethernet 0/24.
- Par défaut, il y a un vlan numéroté 1 sur le switch CISCO et toutes les interface de 1 à 24 en font partie.

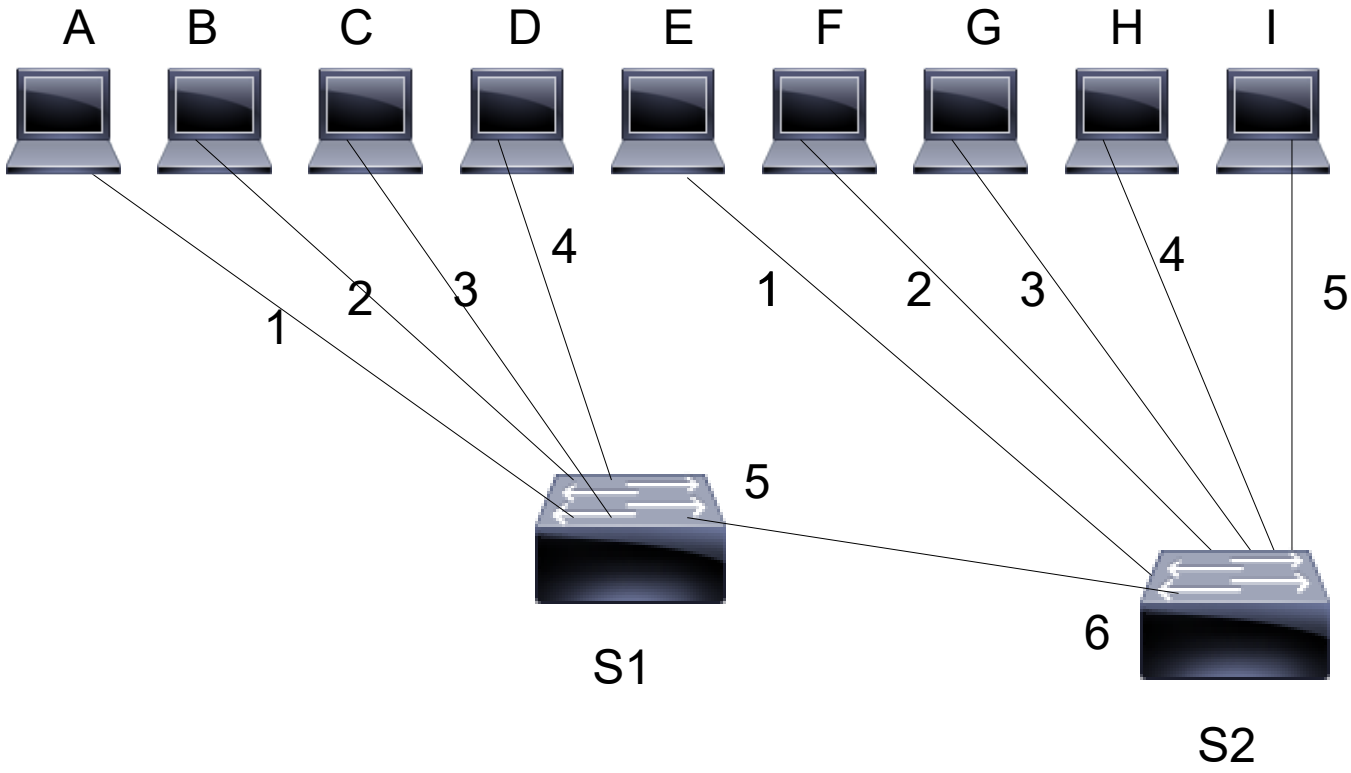
Quelques commandes

- **configure terminal** : permet de mettre le switch en mode config
- **vlan database** : permet de mettre le switch en mode vlan
- **vlan** num **name** nom_vlan : crée le vlan numéro num ayant pour nom nom_vlan
- **interface** nom_interface : permet de passer en mode config-if pour l'interface nom_interface
- **switchport mode** access : mets l'interface du vlan dans le mode access
- **switchport access vlan** num : mets l'interface dans le vlan numéro num

Configuration du Switch CISCO

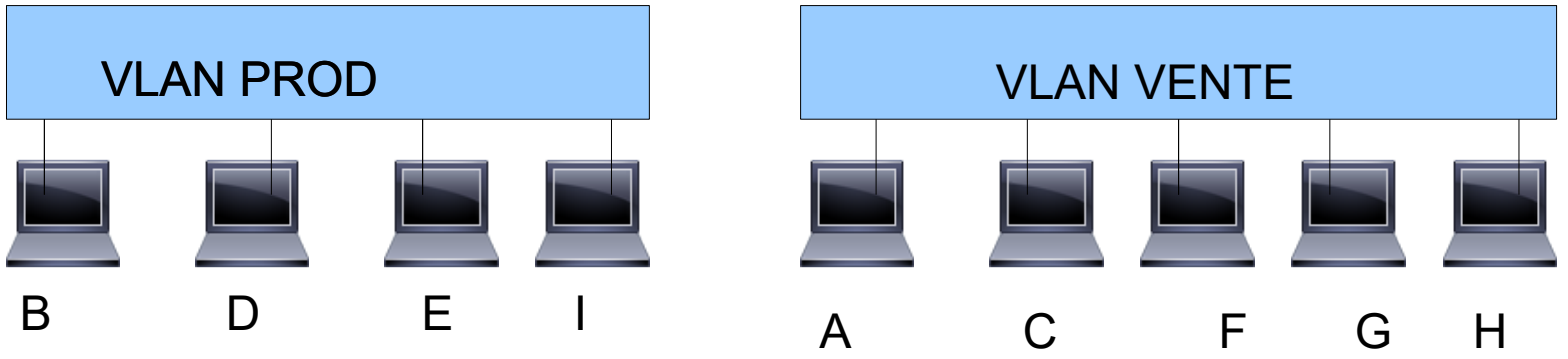
```
enable
configure terminal
vlan database
vlan 2 name PROD
vlan 3 name VENTE
exit
interface fastethernet 0/1
switchport mode access
switchport access vlan 2
exit
interface fastethernet 0/2
switchport mode access
switchport access vlan 3
exit
interface fastethernet 0/3
switchport mode access
switchport access vlan 3
exit
interface fastethernet 0/4
switchport mode access
switchport access vlan 2
exit
interface fastethernet 0/5
switchport mode access
switchport access vlan 3
exit
interface fastethernet 0/6
switchport mode access
switchport access vlan 2
exit
interface fastethernet 0/7
switchport mode access
switchport access vlan 3
exit
interface fastethernet 0/8
switchport mode access
switchport access vlan 2
exit
interface fastethernet 0/9
switchport mode access
switchport access vlan 3
exit
exit
```

Les VLAN sur des switchs CISCO
Exemple 2



9 machines A, B, C, D, E , F, G, H, et I sont connectées sur deux switchs CISCO S1 et S2

On veut créer 2 vlans nommés PROD et VENTE



Connection entre les 2 switchs

- La connection entre les 2 switchs S1 et S2 s'appelle un trunk. Nous utiliserons le protocole VTP : S1 sera le serveur VTP et S2 le client VTP. Le protocole assurera que les VLANs créés sur S1 soient connus de S2.
- **switchport mode** trunk : place une interface dans le mode trunk
- **switchport trunk allowed vlan** num1,..., num indique la liste des vlans autorisés à traverser le trunk
- **vtp server** : place le switch en tant que server VTP
- **vtp client** : place le switch en tant que client VTP

Configuration de S1

```
enable
configure terminal
vlan database
vtp server
vtp domain TEST
vlan 2 name PROD
vlan 3 name VENTE
exit
interface fastethernet 0/5
switchport mode trunk
switchport trunk allowed vlan 2,3
exit
interface fastethernet 0/1
switchport mode access
switch access vlan 3
exit
interface fastethernet 0/2
switchport mode access
switch access vlan 2
exit
interface fastethernet 0/3
switchport mode access
switch access vlan 3
exit
interface fastethernet 0/4
switchport mode access
switch access vlan 2
exit
exit
```

Configuration de S2

```
enable
configure terminal
vlan database
vtp client
vtp domain TEST
exit
interface fastethernet 0/6
switchport mode trunk
switchport trunk allowed vlan 2,3
exit
interface fastethernet 0/1
switchport mode access
switch access vlan 2
exit
interface fastethernet 0/2
switchport mode access
switch access vlan 3
exit
interface fastethernet 0/3
switchport mode access
switch access vlan 3
exit
interface fastethernet 0/4
switchport mode access
switch access vlan 3
exit
interface fastethernet 0/5
switchport mode access
switch access vlan 2
exit
exit
```