

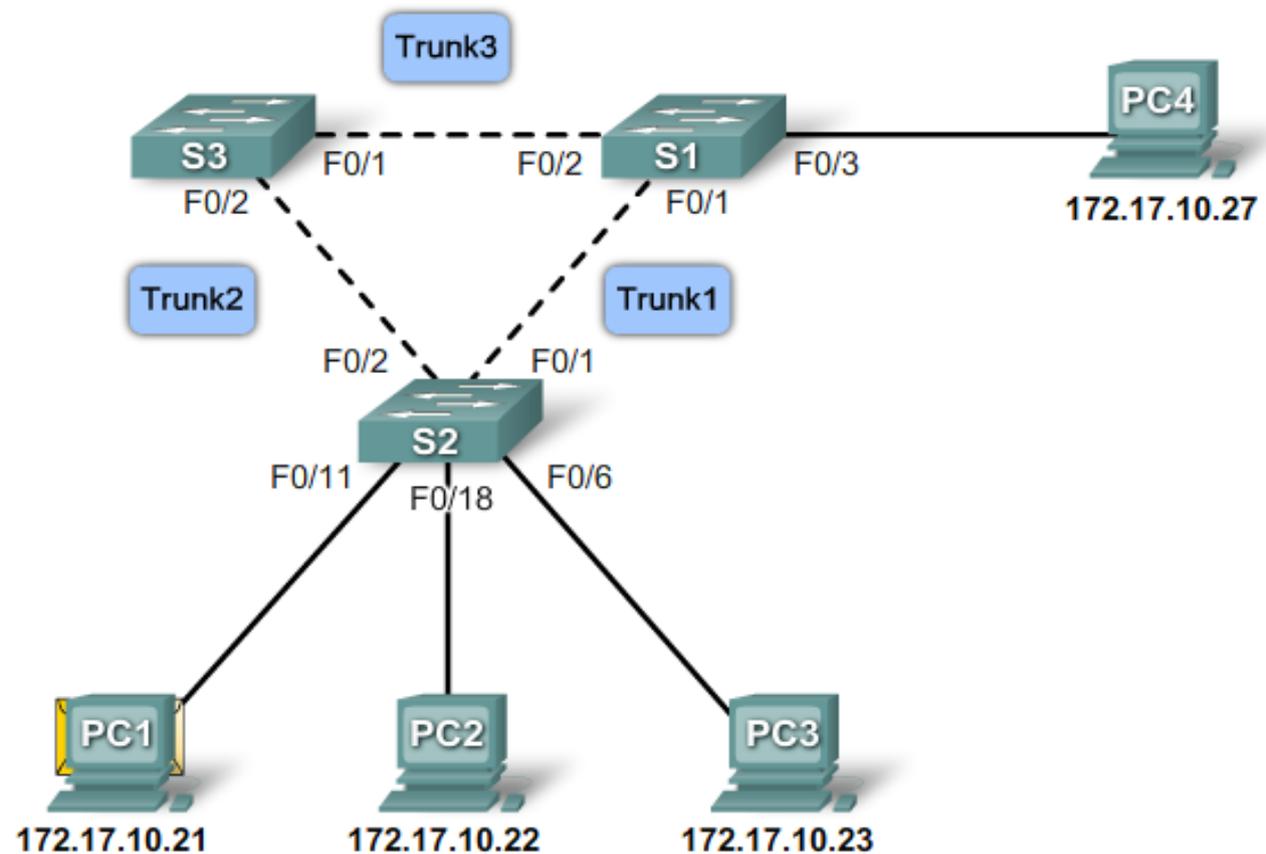
# Des protocoles de Spanning Tree

# Des protocoles de Spanning Tree

La redondance ?

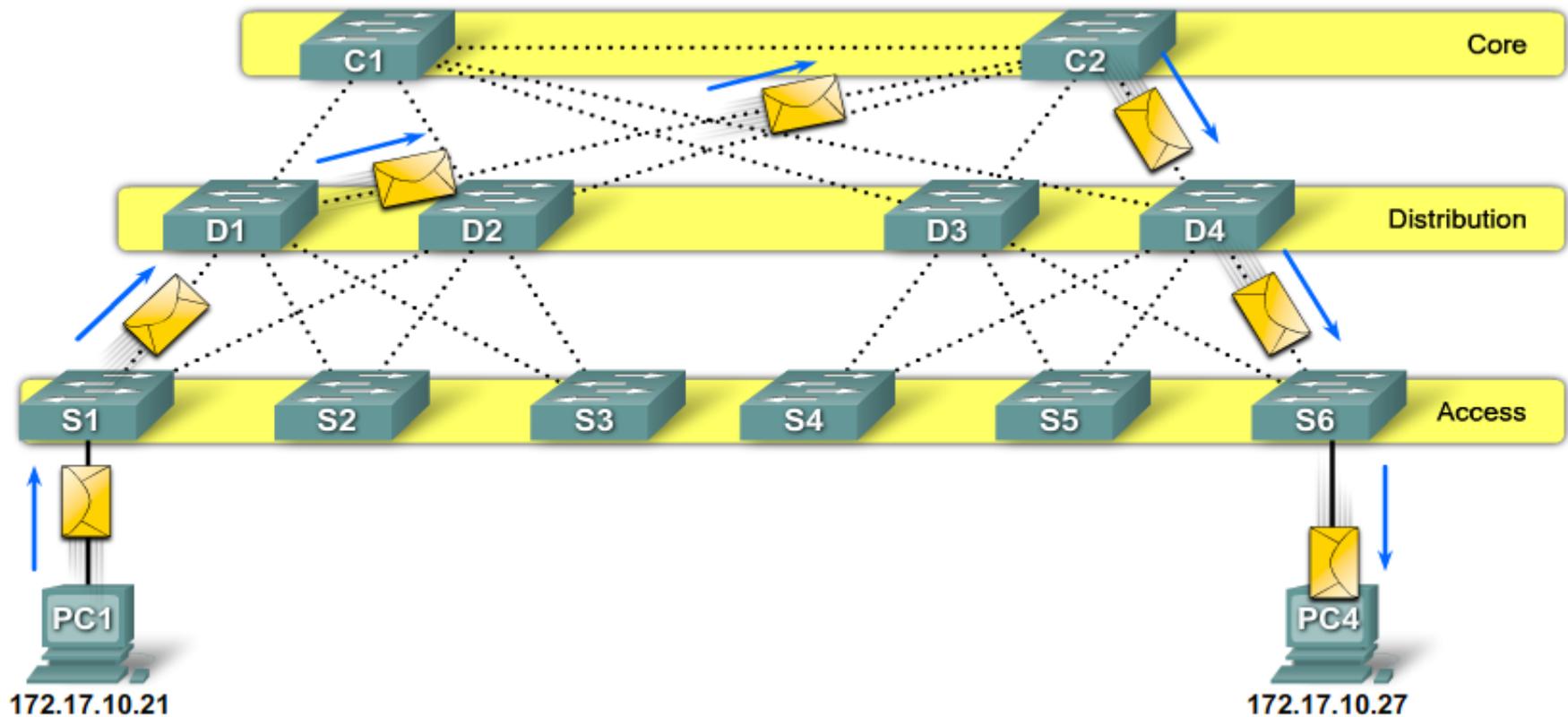
# La redondance sur les liens

- ➔ Permet d'avoir plusieurs chemins en cas d'une défaillance d'un lien
- ➔ Problème : un broadcast provenant de PC1 ne pourra jamais être arrêté !



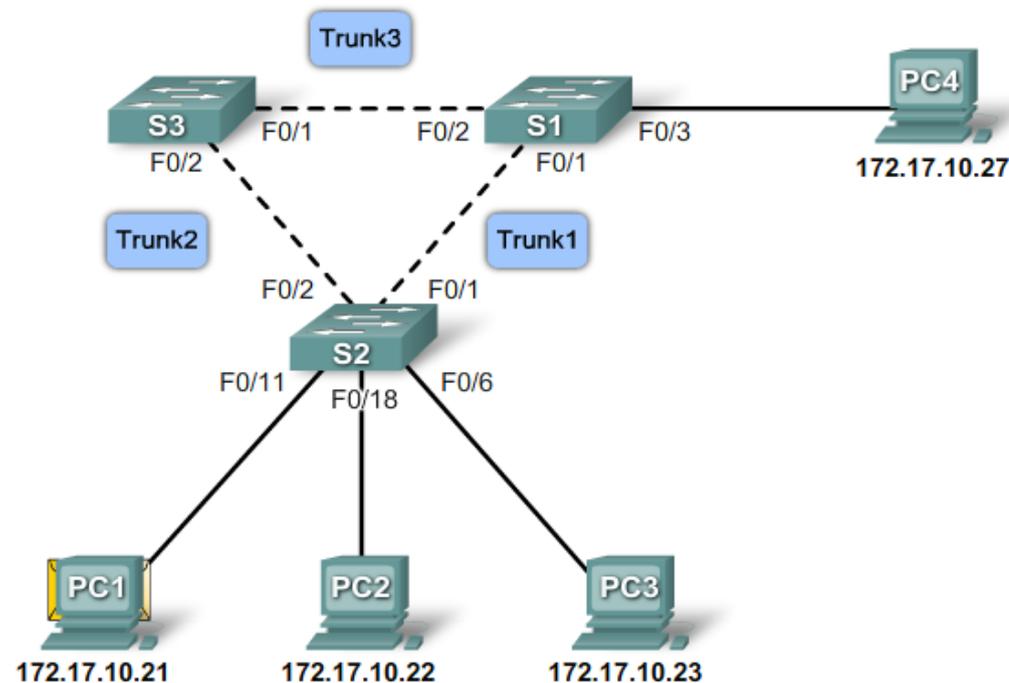
## Autre type de redondance

- ➔ Permet de supporter la perte d'un lien ou d'un équipement
- ➔ Toujours le problème du broadcast



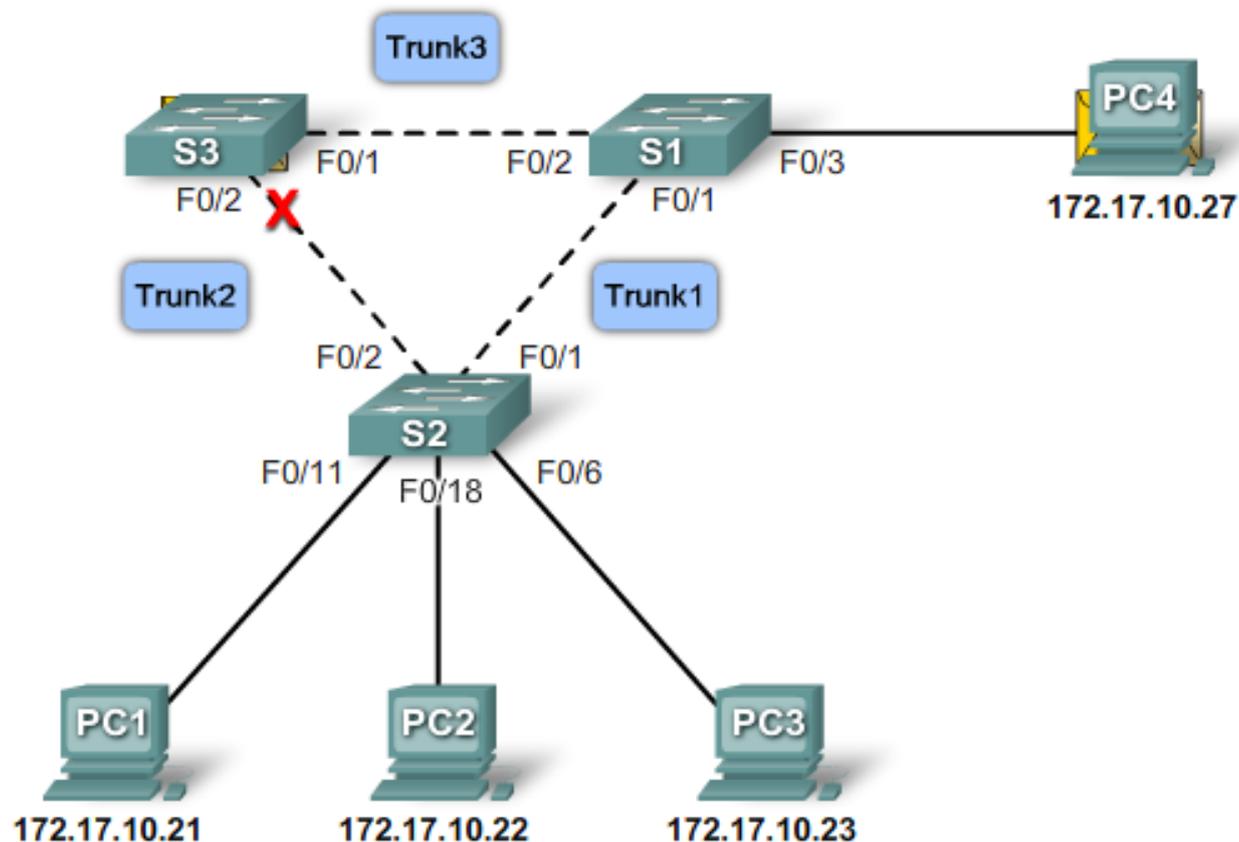
# Les problèmes

- ➔ Le broadcast
  - ➔ Généré par exemple par les requêtes ARP
  - ➔ Comme il n'existe aucun moyen de supprimer les trames Ethernet, comme sur IP avec le TTL, cela encombre les liens et génère des broadcast storm
- ➔ Duplication de trames unicast
  - ➔ A cause des boucles, un PC peut recevoir plusieurs fois une même trame !



# Une solution : Spanning Tree Protocol

- Les ports qui génèrent des boucles sont automatiquement « désactivés »



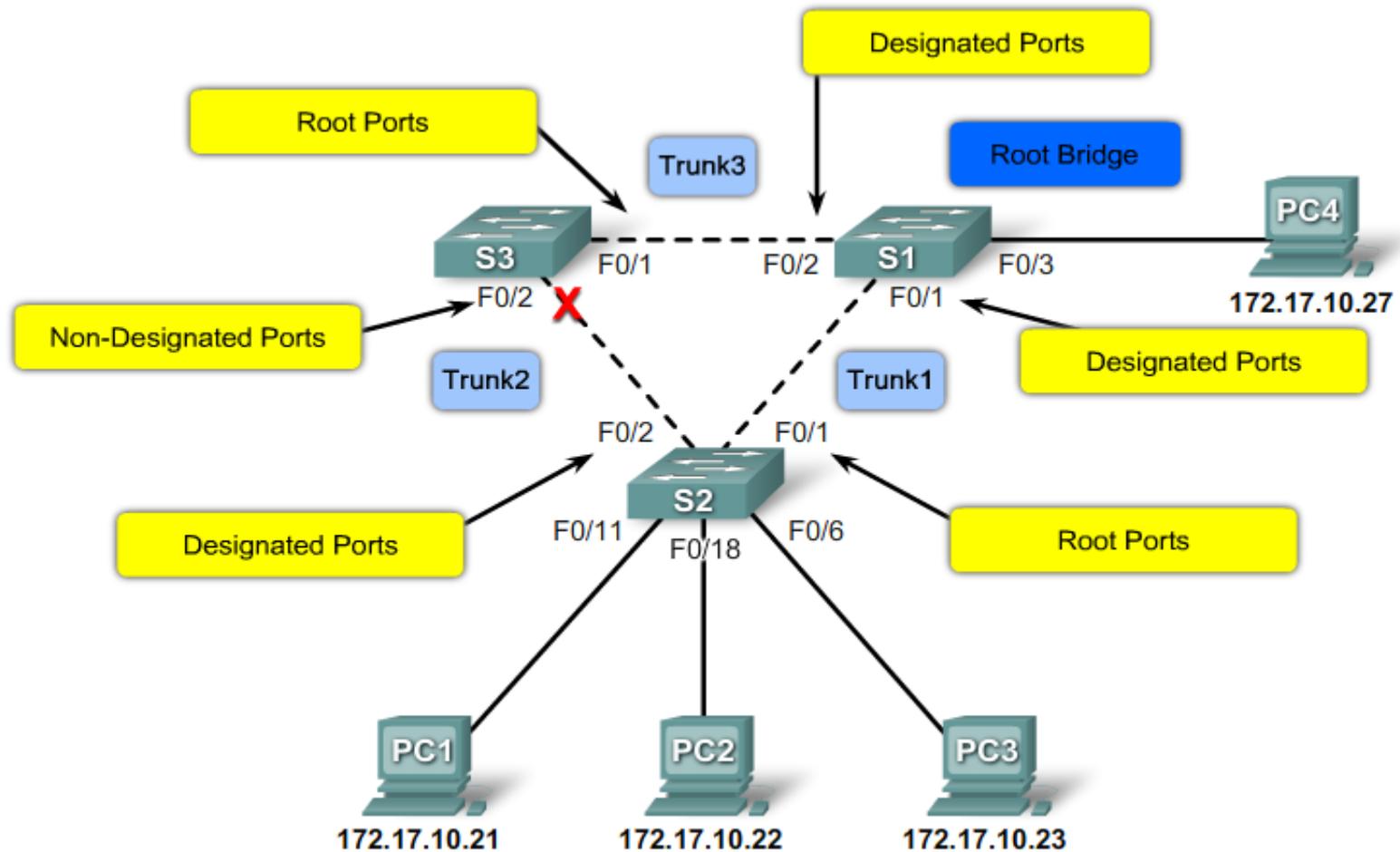
# Des protocoles de Spanning Tree

Le plus simple : STP ou IEEE 802.1D

# Le Spanning Tree Algorithm 802.1D

- ➔ Le switch qui possède la plus petite Bridged Identity (BID) est élu Root
  - ➔ Le BID est la concaténation
    - ➔ d'une priorité comprise entre 1 et 65536, par défaut 32768
    - ➔ de l'adresse MAC du switch
- ➔ Toutes les 2 secondes, chaque switch broadcast son BID ainsi que le BID du Root Bridge dont il a la connaissance
- ➔ Les ports des switchs sont appelés
  - ➔ Root Ports s'ils sont connectés au Root Bridge (ou Root Switch)
  - ➔ Designated Ports s'ils ne sont pas Root Port et s'ils autorisent le trafic à circuler
  - ➔ Non-designated Ports s'ils bloquent le trafic
- ➔ Pendant l'exécution de cet algorithme, les messages échangés entre les switchs sont appelés des BPDU

# Exemple pour le STP

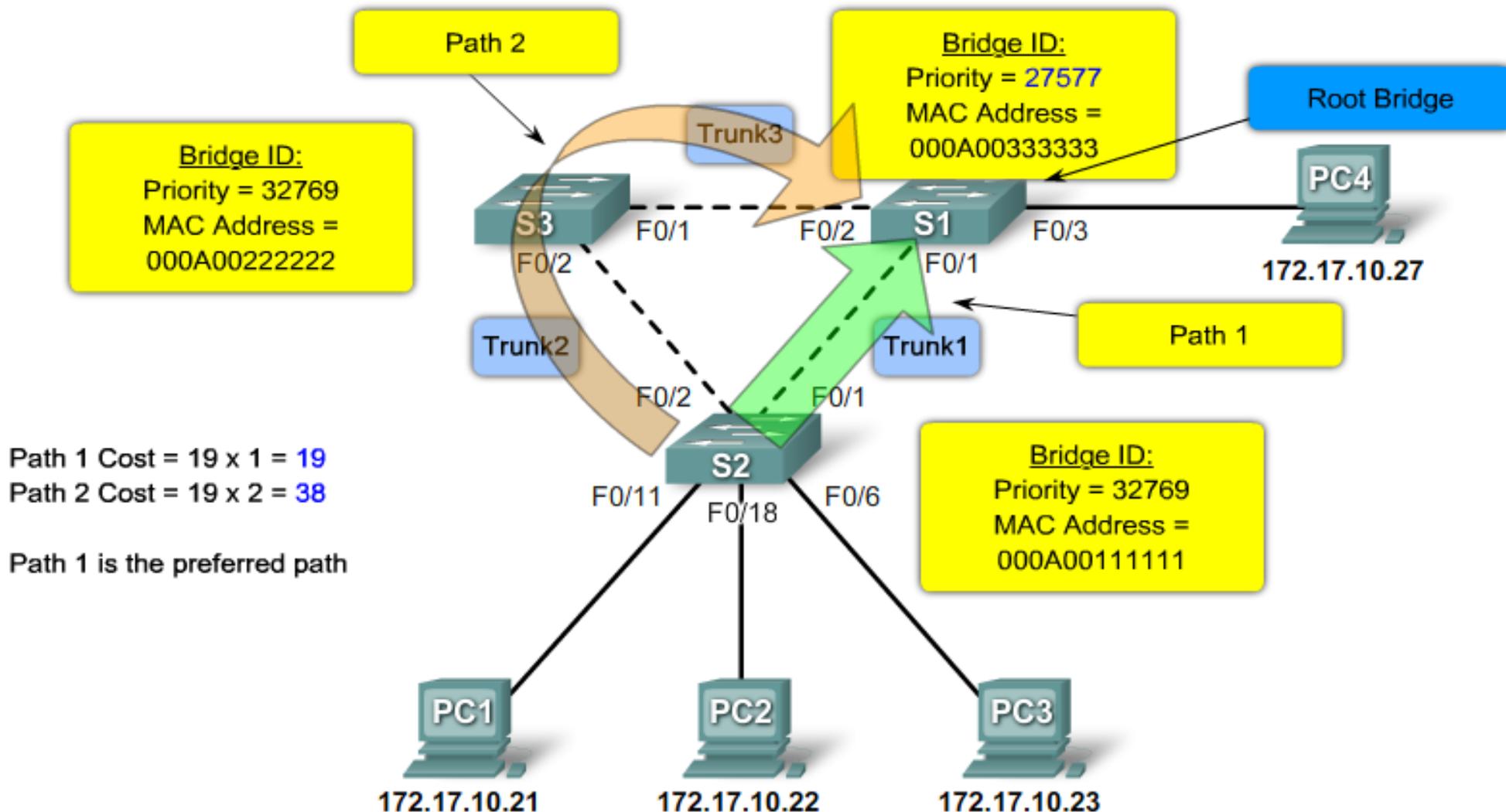


# Sélection des meilleurs chemins

- Sur chaque port du switch, le coût de chaque chemin vers la racine va être évalué

Link Speed	Cost (Revised IEEE Specification)	Cost (Previous IEEE Specification)
10 Gb/s	2	1
1 Gb/s	4	1
100 Mb/s	19	10
10 Mb/s	100	100

# Calcul des couts des chemins



## En cas d'égalité ?

- ➔ En cas de liens parallèles entre 2 switchs ?
- ➔ Quel est le chemin le plus court ?
- ➔ En cas d'égalité (coût du chemin et BID), le port ID est utilisé. Le port ID le plus petit forward les trames et les autres les bloquent

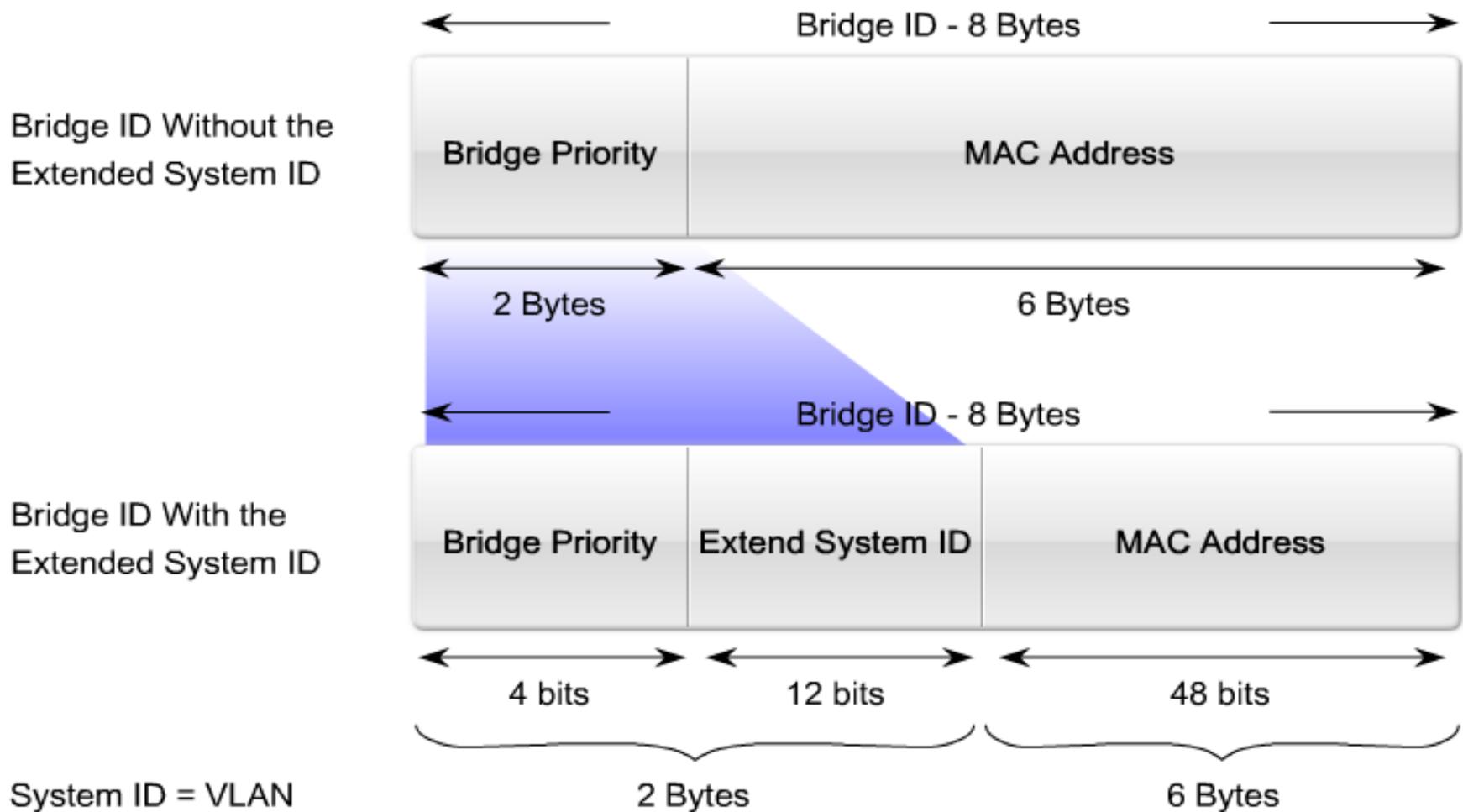
# Configuration

- ➔ Changer le cout d'un port
  - ➔ `Switch(config-if)#spanning-tree cost number`
- ➔ Revenir à cout par défaut
  - ➔ `Switch(config-if)#no spanning-tree cost`
- ➔ Vérifier l'état du spanning-tree sur un switch
  - ➔ `Switch#show spanning-tree`
  - ➔ Permet de connaître le Root ID, le BID du switch et le cout de chaque port vers le Root Bridge

# Les instances STP

- ➔ Une instance de spanning tree est définie de la façon suivante :
  - ➔ Quand tous les switches participant aux échanges de BPDU et à la négociation du spanning tree sont associés à une unique racine
- ➔ Si c'est fait pour tous les VLAN, on parle de Common Spanning Tree (CST), sinon, quand c'est effectué par VLAN, on parle de Per VLAN Spanning Tree (PVST)
  - ➔ Avec PVST, nous avons donc une racine par VLAN
- ➔ Par VLAN, cela nécessite d'avoir l'information de l'identifiant du VLAN concerné (le VID) dans la trame !
  - ➔ Le BID est donc étendu pour pouvoir contenir le VID

# Le BID



## Le BID

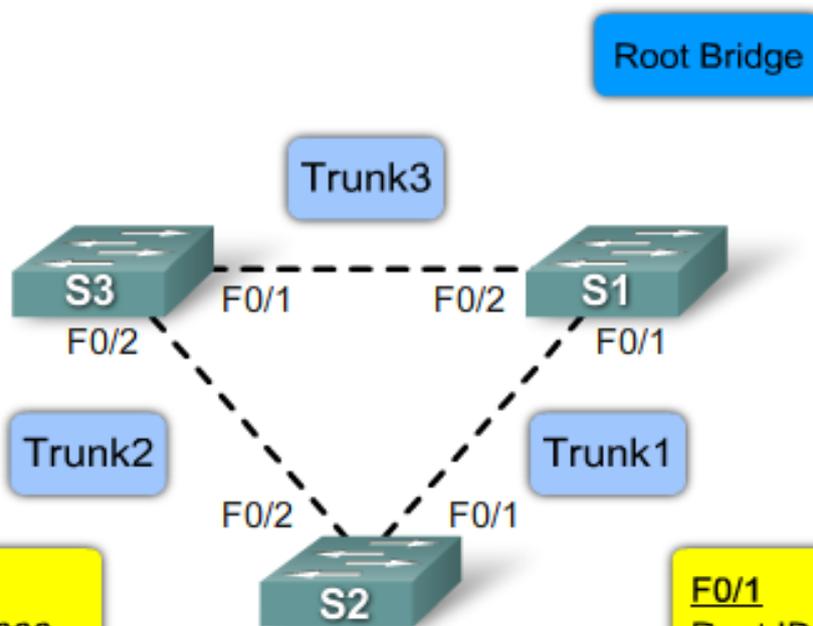
- ➔ Avant l'apparition des VLAN, la priorité était codé sur 16 bits
- ➔ Depuis l'existence des VLAN, la priorité est codé sur 4 bits auquel on ajoute 12 bits pour l'identifiant du VLAN sur lequel se construit le Spanning Tree
  - ➔ La priorité ne peut donc prendre que des valeurs multiples de 4096 ( $2^{12}$ )
- ➔ Pour configurer cette priorité 2 solutions
  - ➔ `Switch(config)#spanning-tree vlan 1 root primary`
    - ➔ Fixe la priorité à la valeur de 24576
    - ➔ puis éventuellement
    - ➔ `Switch(config)#spanning-tree vlan 1 root secondary`
      - ➔ Fixe la priorité à la valeur de 28672
  - ➔ `Switch(config)#spanning-tree vlan 1 priority number`

# Exemple

**F0/1**  
 Root ID = 24577.000A00333333  
 Bridge ID = 32769.000A00222222  
 Path Cost = 19

**F0/2**  
 Root ID = 24577.000A00333333  
 Bridge ID = 32769.000A00222222  
 Path Cost = 38

**F0/2**  
 Root ID = 24577.000A00333333  
 Bridge ID = 32769.000A00111111  
 Path Cost = 38



**F0/1**  
 Root ID = 24577.000A00333333  
 Bridge ID = 24577.000A00333333  
 Path Cost = 0

**F0/2**  
 Root ID = 24577.000A00333333  
 Bridge ID = 24577.000A00333333  
 Path Cost = 0

**F0/1**  
 Root ID = 24577.000A00333333  
 Bridge ID = 32769.000A00111111  
 Path Cost = 19

# Exemple

**F0/1**

Root ID = 24577.000A00333333  
 Bridge ID = 32769.000A00222222  
 Path Cost = 19

**F0/2**

Root ID = 24577.000A00333333  
 Bridge ID = 32769.000A00222222  
 Path Cost = 38

Root Ports

**F0/2**

Root ID = 24577.000A00333333  
 Bridge ID = 32769.000A00111111  
 Path Cost = 38

Root Bridge

**F0/1**

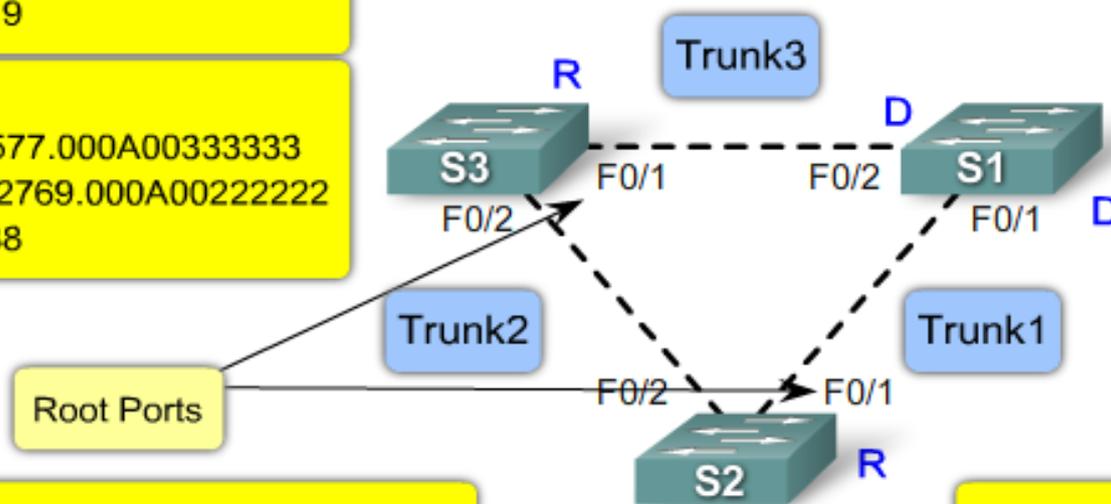
Root ID = 24577.000A00333333  
 Bridge ID = 24577.000A00333333  
 Path Cost = 0

**F0/2**

Root ID = 24577.000A00333333  
 Bridge ID = 24577.000A00333333  
 Path Cost = 0

**F0/1**

Root ID = 24577.000A00333333  
 Bridge ID = 32769.000A00111111  
 Path Cost = 19



# Exemple

**F0/1**

Root ID = 24577.000A00333333  
 Bridge ID = 32769.000A00222222  
 Path Cost = 19

**F0/2**

Root ID = 24577.000A00333333  
 Bridge ID = 32769.000A00222222  
 Path Cost = 38

**F0/2**

Root ID = 24577.000A00333333  
 Bridge ID = 32769.000A00111111  
 Path Cost = 38

Root Bridge

**F0/1**

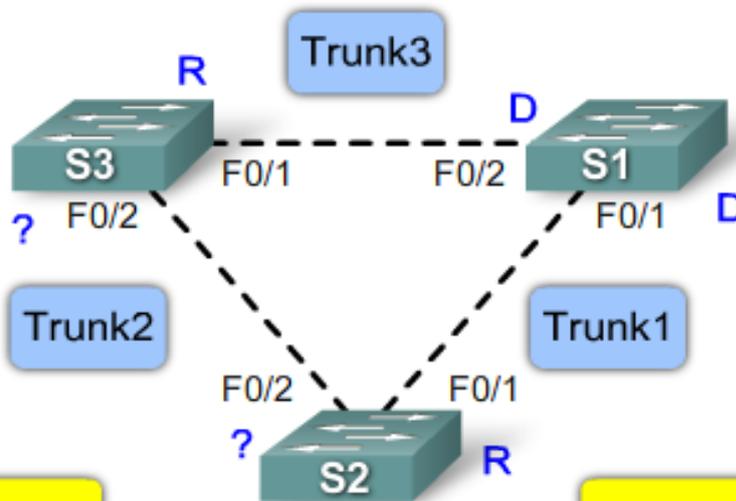
Root ID = 24577.000A00333333  
 Bridge ID = 24577.000A00333333  
 Path Cost = 0

**F0/2**

Root ID = 24577.000A00333333  
 Bridge ID = 24577.000A00333333  
 Path Cost = 0

**F0/1**

Root ID = 24577.000A00333333  
 Bridge ID = 32769.000A00111111  
 Path Cost = 19



En cas de cout égaux vers le root, le plus petit BID gagne.  
 F0/2 de S2 sera donc designated et F0/2 de S3 Non designated

# Exemple

**F0/1**

Root ID = 24577.000A00333333  
 Bridge ID = 32769.000A00222222  
 Path Cost = 19

**F0/2**

Root ID = 24577.000A00333333  
 Bridge ID = 32769.000A00222222  
 Path Cost = 38

**F0/1**

Root ID = 24577.000A00333333  
 Bridge ID = 24577.000A00333333  
 Path Cost = 0

**F0/2**

Root ID = 24577.000A00333333  
 Bridge ID = 24577.000A00333333  
 Path Cost = 0

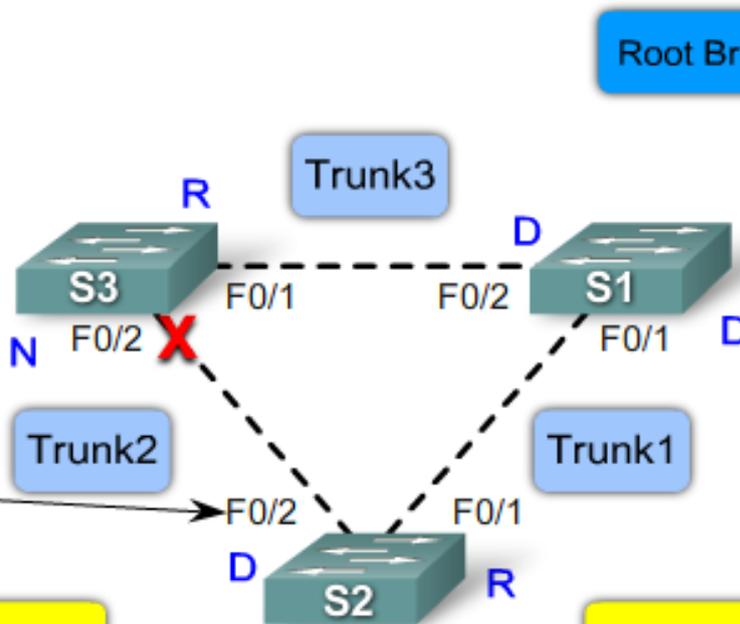
Designated Port

**F0/2**

Root ID = 24577.000A00333333  
 Bridge ID = 32769.000A00111111  
 Path Cost = 38

**F0/1**

Root ID = 24577.000A00333333  
 Bridge ID = 32769.000A00111111  
 Path Cost = 19



# L'état des ports sur STP

- ➔ Un port de switch peut prendre 5 états différents pendant le processus de fabrication de l'arbre couvrant via le protocole STP
  - ➔ Blocking : port non-designated. Ne fait juste qu'envoyer des BPDU et les analyser
    - ➔ Reste dans cet état pendant 20 secondes max. (max age)
  - ➔ Listening : reçoit les BPDU et transmet les siens également
    - ➔ Reste dans cet état pendant 15 secondes max. (forward delay)
  - ➔ Learning : le port ne forward toujours pas les trames mais apprend les adresses MAC sources contenues dans celles-ci
    - ➔ Reste dans cet état pendant 15 secondes max. (forward delay)
  - ➔ Forwarding : toutes les trames sont transmises
  - ➔ Disable : port administrativement désactivé (shutdown)
- ➔ Chaque switch envoie toutes les 2 secondes ses BPDU
  - ➔ Configurable entre 1 et 10 secondes
- ➔ Toutes les valeurs de timer ont été optimisées pour un réseau de diamètre 7
- ➔ Existence de 2 types de BPDU : configuration ou Topologie Change Notification (TCN)

# Le Bridge Protocol Data Unit (BPDU)

Bytes	Field
2	Protocol ID
1	Version
1	Message type
1	Flags
8	Root ID
4	Cost of path
8	Bridge ID
2	Port ID
2	Message age
2	Max age
2	Hello time
2	Forward delay

- Echange de BPDU entre les switches
- Message Type : Configuration ou TCN
- Root ID : le BID le plus petit
- Cost of Path : coût de tous les liens du switch expéditeur vers le root bridge
- BID : BID de l'expéditeur
- Port ID : ID du port qui expédie le BPDU
- STP timers values : Max age, hello time et forward delay
- Flags : dépende du protocole

# Des protocoles de Spanning Tree

Améliorations et variantes de STP

# Le Cisco PortFast

- ➔ C'est une solution propriétaire Cisco qui fait passer un port instantanément de Blocking à Forwarding
- ➔ A faire sur les switchs d'access layer sur lesquels sont connectés directement des workstations ou serveur
  - ➔ Ceci est à faire sur un réseau avec support du DHCP. Sans cela, une station va envoyer sa requête et en raison du temps de convergence du STP, n'obtiendra pas d'adresse IP
- ➔ Pour l'activer, sur l'interface correspondante :
  - ➔ `spanning-tree portfast`
- ➔ Pour l'activer sur toutes les interfaces non trunk
  - ➔ `spanning-tree portfast default`
- ➔ Vérification par
  - ➔ `show running-config interface fastethernet ...`

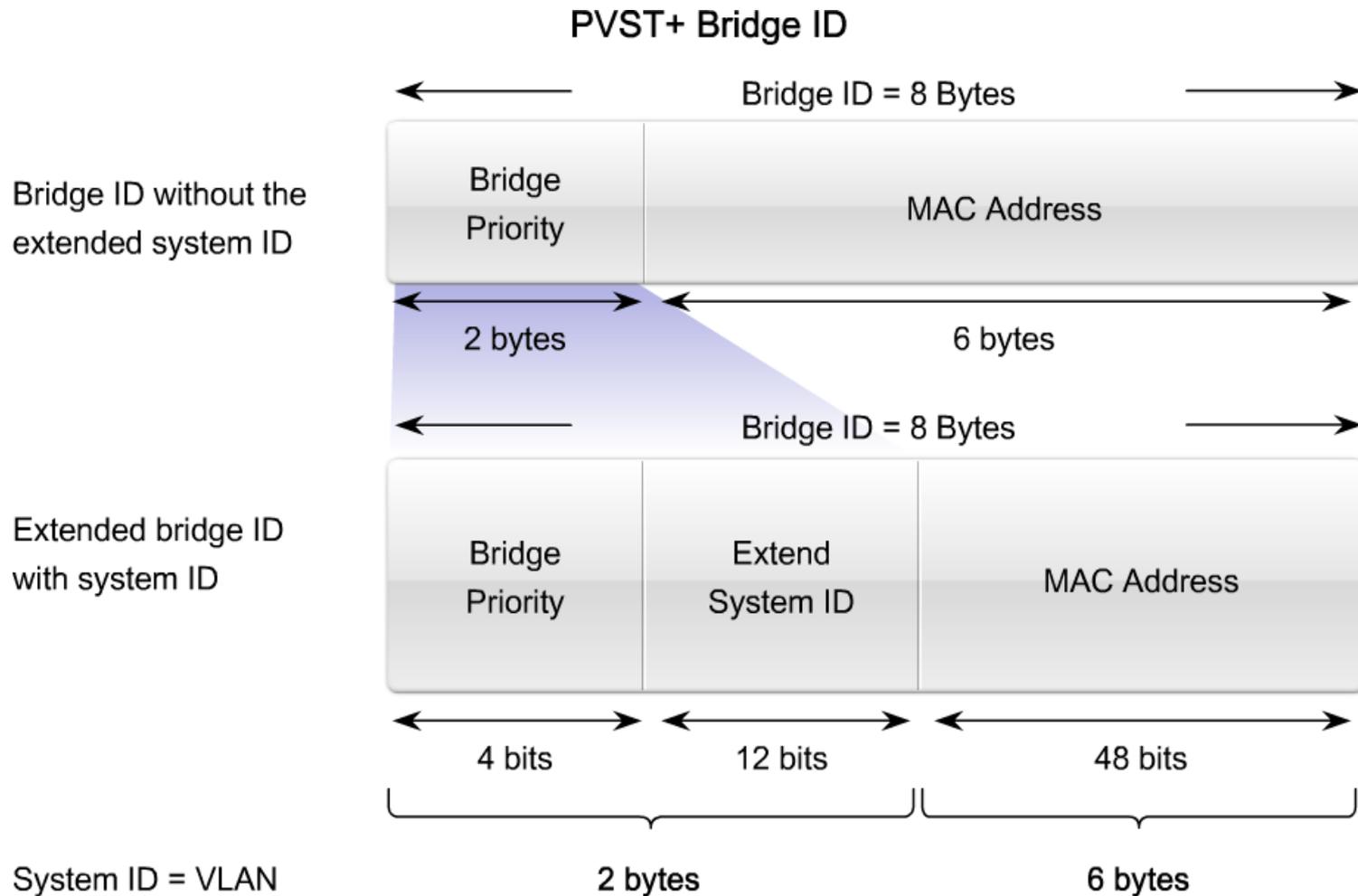
# Les variantes

Cisco Proprietary	<b>PVST</b> <ul style="list-style-type: none"> <li>• Uses the Cisco proprietary ISL trunking protocol</li> <li>• Each VLAN has an instance of spanning tree</li> <li>• Ability to load balance traffic at layer-2</li> <li>• Includes extensions BackboneFast, UplinkFast, and PortFast</li> </ul>
	<b>PVST+</b> <ul style="list-style-type: none"> <li>• Supports ISL and IEEE 802.1Q trunking</li> <li>• Supports Cisco proprietary STP extensions</li> <li>• Adds BPDU guard and Root guard enhancements</li> </ul>
	<b>rapid-PVST+</b> <ul style="list-style-type: none"> <li>• Based on IEEE802.1w standard</li> <li>• Has faster convergence than 802.1D</li> </ul>
IEEE Standard	<b>RSTP</b> <ul style="list-style-type: none"> <li>• Introduced in 1982 provides faster convergence than 802.1D</li> <li>• Implements generic versions of the Cisco proprietary STP extensions</li> <li>• IEEE has incorporated RSTP into 802.1D, identifying the specification as IEEE 802.1D-2004</li> </ul>
	<b>MSTP</b> <ul style="list-style-type: none"> <li>• Multiple VLANs can be mapped to the same spanning-tree instance</li> <li>• Inspired by the Cisco Multiple Instances Spanning Tree Protocol (MISTP),</li> <li>• IEEE 802.1Q-2003 now includes MSTP</li> </ul>

## PVST+

- ➔ Protocole propriétaire Cisco
- ➔ Il est possible d'avoir un process STP pour chaque VLAN
  - ➔ Un port peut donc être en forwarding pour un VLAN et blocking pour un autre
  - ➔ Un switch peut même être root pour un VLAN et un autre switch pour un autre VLAN
- ➔ L'identification des VLAN se fait grâce une modification de l'interprétation du champ Bridge ID

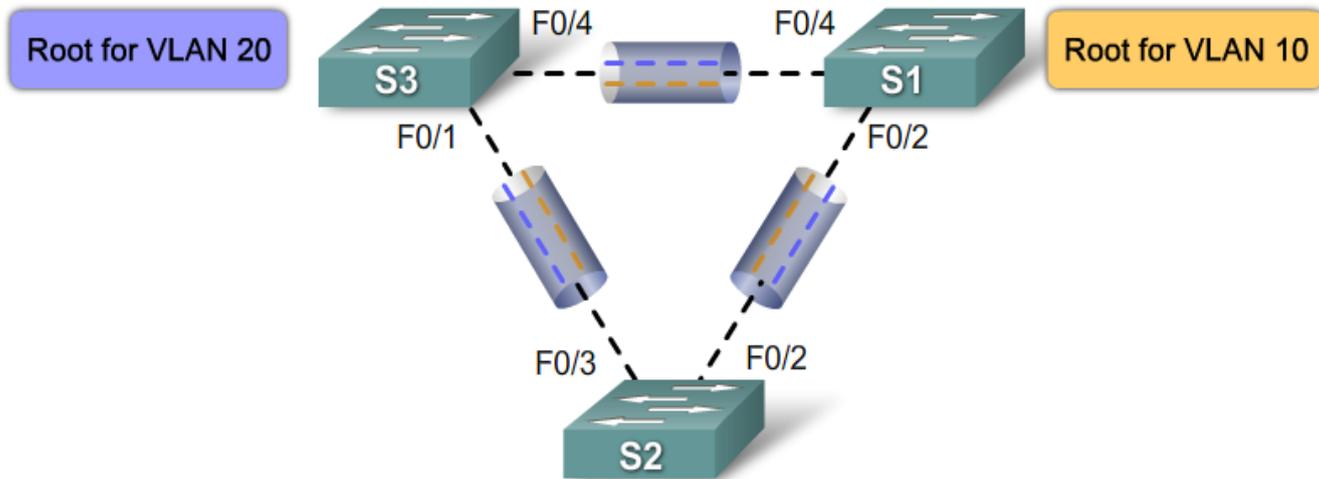
# Le Bridge ID sur PVST+



# Exemple sur PVST+

## PVST+ Bridge ID

Priority	+	VLAN ID + MAC Address	= BID
32768	+	10 + 000A00333333	= 32778.000A00333333
32768	+	20 + 000A00333333	= 32788.000A00333333



VLAN 10 - - - - -  
 VLAN 20 - - - - -

# Configuration par défaut

- ➔ Par défaut, sur un switch Cisco Catalyst 2960

Feature	Default Setting
Enable state	Enabled on VLAN 1
Spanning-tree mode	PVST+ (Rapid PVST+ and MSTP are disabled.)
Switch priority	32768
Spanning-tree port priority (configurable on a per-interface basis)	128
Spanning-tree port cost (configurable on a per-interface basis)	1000 Mb/s: 4, 100 Mb/s: 19, 10 Mb/s: 100
Spanning-tree VLAN port priority (configurable on a per-VLAN basis)	128
Spanning-tree VLAN port cost (configurable on a per-VLAN basis)	1000 Mb/s: 4, 100 Mb/s: 19, 10 Mb/s: 100
Spanning-tree timers	Hello time: 2 seconds Forward-delay time: 15 seconds Maximum-aging time: 20 seconds Transmit hold count: 6 BPDUs

# RSTP

- ➔ Rapid STP, normalisé par IEEE sous le numéro 802.1w
- ➔ Évolution du 802.1D
- ➔ Convergence plus rapide
- ➔ Notion de Edge port qui reprend la notion de PortFast de Cisco
  - ➔ Utilisé uniquement sur des ports qui ne seront jamais en liaison avec d'autres swiths
  - ➔ Port qui passe immédiatement à l'état forwarding

# Les états des ports

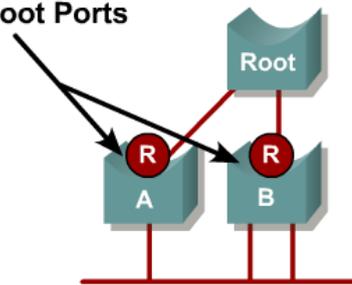
- ➔ 3 états uniquement pour les ports dans RSTP
  - ➔ **Discarding**
    - ➔ Aucune communication de paquet
  - ➔ **Learning**
    - ➔ Apprentissage des adresses MAC provenant des trames reçues
  - ➔ **Forwarding**
    - ➔ Toutes les trames sont transmises

# Comparaison STP / RSTP

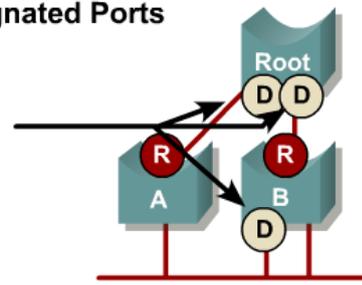
Operational Port State	STP Port State	RSTP Port State
Enabled	Blocking	Discarding
Enabled	Listening	Discarding
Enabled	Learning	Learning
Enabled	Forwarding	Forwarding
Disabled	Disabled	Discarding

# RSTP port Roles

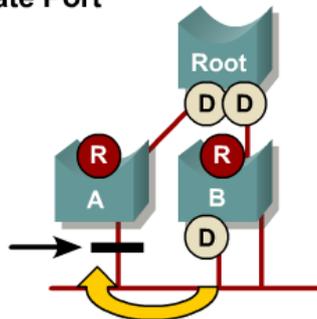
Root Ports



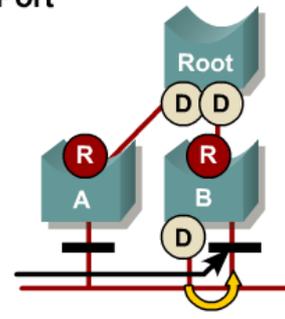
Designated Ports



Alternate Port



Backup Port

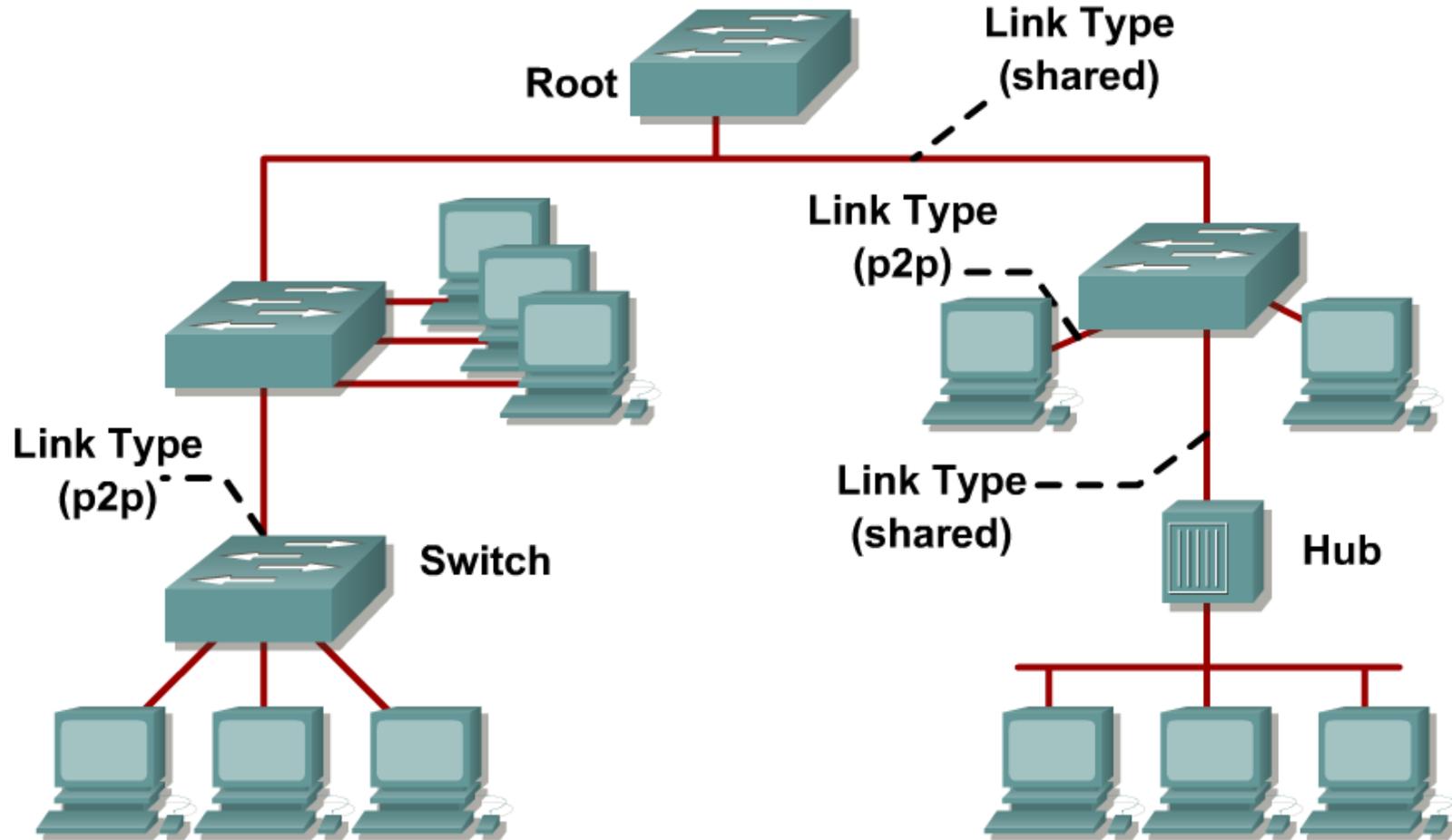


- ➔ Alternate Port : Port en discarding pendant la configuration stable et active. Permet d'obtenir un autre chemin sur un switch non designated si le chemin du designated switch est en échec
- ➔ Backup Port : Port en discarding pendant la configuration stable et active. Permet d'obtenir un autre port vers un même segment sur un designated switch. Port avec un ID plus élevé que le designated port du switch

## Edge port et type de lien

- ➔ Edge ports : ports qui ne seront jamais connectés à un autre équipement actif. Ce sont des ports qui seront automatiquement dans l'état forwarding
  - ➔ Contrairement au PortFast, si un edge port reçoit un BPDU, il redevient automatiquement un port normal de spanning tree
- ➔ Type de lien : distinction entre les liens p2p et shared
  - ➔ Shared link : fonctionne en half-duplex, suppose que le média est partagé par plusieurs switches
  - ➔ Point-to-point (p2p) link : fonctionne en full-duplex : le port est supposé être connecté à un unique switch, à l'autre extrémité du lien
- ➔ Le type de lien est utilisé pour pour déterminer si un port pourra ou pas passer immédiatement en forwarding port

# Type de lien



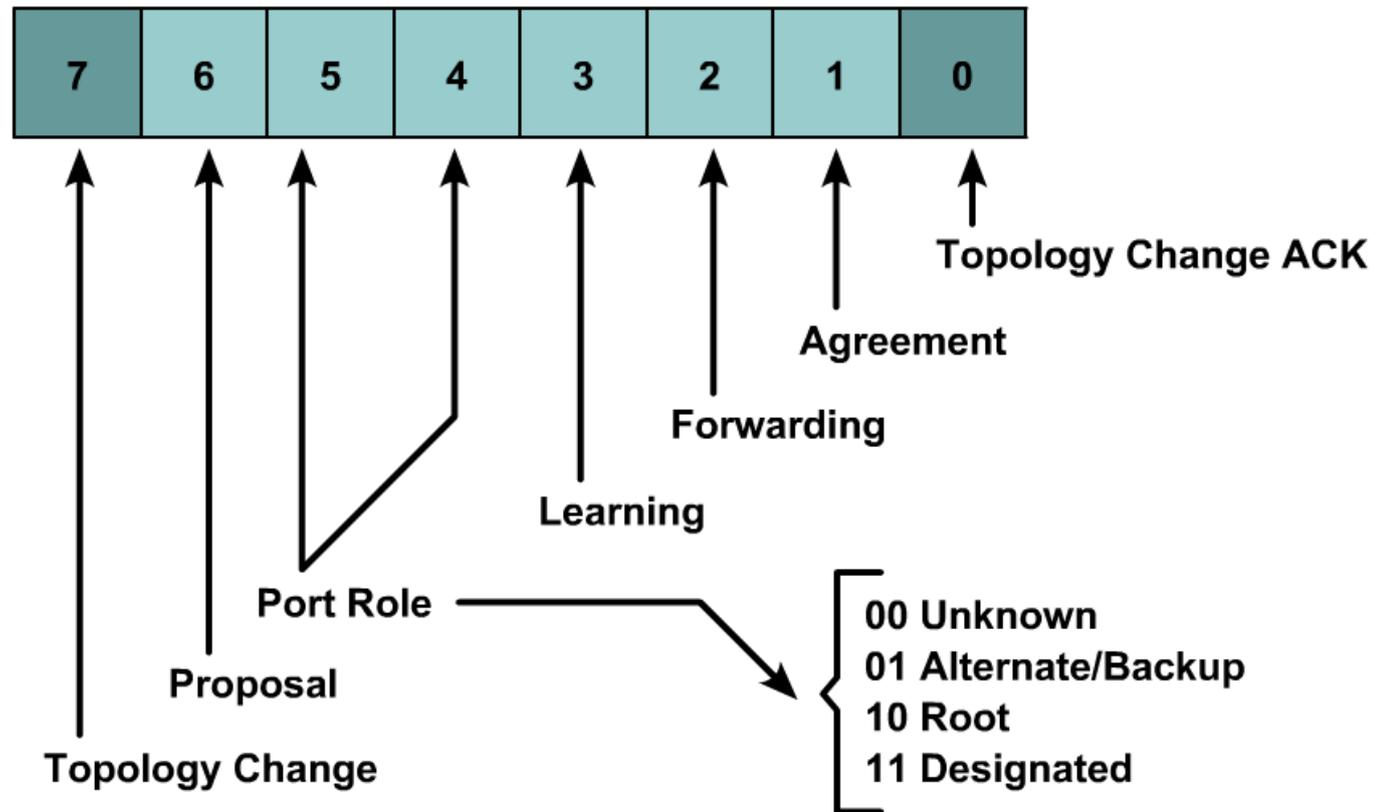
## Utilisation du type de lien

- ➔ Root ports ne s'en servent pas
- ➔ Alternate et Backup, dans la majorité des cas, non plus
- ➔ Designated port utilise le plus souvent ce paramètre : type de lien
- ➔ Transition rapide vers le forwarding state, pour un designated port, si le type est point-to-point

## RSTP BPDU

- ➔ RSTP (802.1w) utilise des BPDU de type 2, version 2
- ➔ Chaque switch BPDU envoie des BPDU à chaque intervalle hello (2 secondes par défaut). Même s'il n'a rien reçu du root bridge
  - ➔ Dans STP (802.1D), le BPDU n'est envoyé que par le root bridge
  - ➔ Dans RSTP (802.1w), chaque switch envoie des BPDU
- ➔ Les informations du protocole sont périmées si au bout de 3 hello times ou du age time, rien n'est reçu
  - ➔ Cela permet de faire une détection rapide de panne
  - ➔ On considère, qu'au bout de 3 hellos perdus, le root bridge, les switches voisins et le designated bridge ne peuvent plus communiquer entre-eux

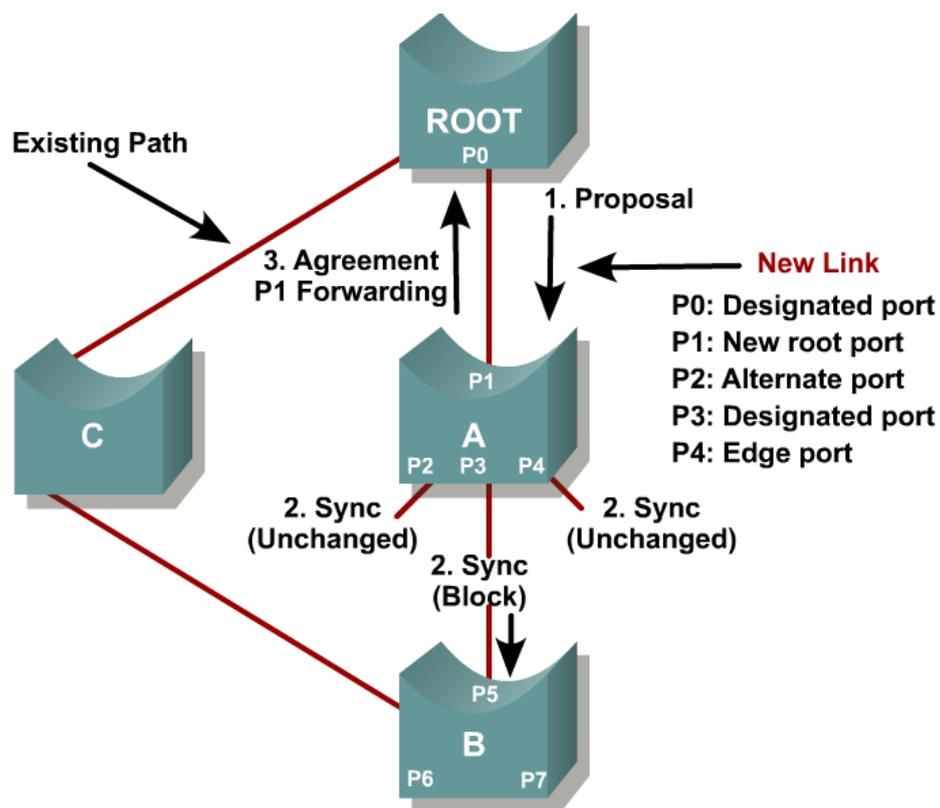
# RSTP BPDUs



# Fonctionnement

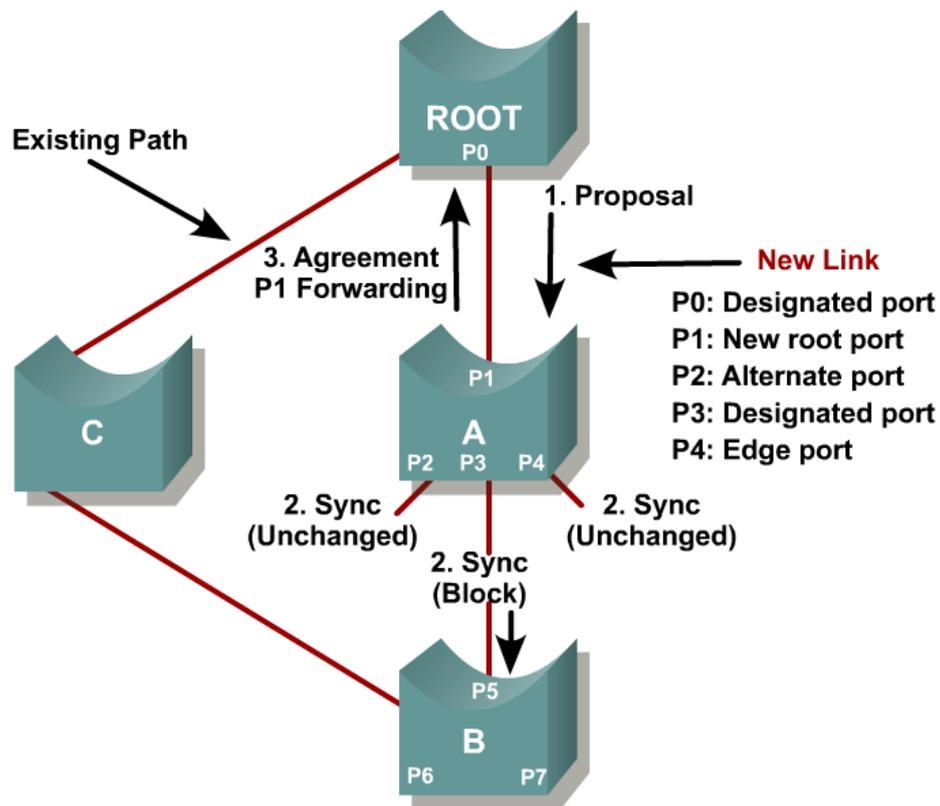
- ➔ Avec STP (802.1D), quand un port devient « designated port », il faut attendre 2 forward time pour être en forwarding port
- ➔ Avec RSTP, la transition ne se fait pas à expiration d'un compteur
- ➔ La transition rapide vers le forwarding port ne se fait que sur des Edge ports et sur des point-to-point links
- ➔ Quand un switch reçoit un BPDU supérieur d'un voisin, (donc avec une priorité plus petite, ou un chemin moins coûteux) (dit supérieur car les informations contenues donnent plus d'importance à ce BPDU), ce port passe en root port
  - ➔ Avec RSTP, les non-edge port passent en proposal-agreement process
  - ➔ Uniquement les non edge en discarding ou learning

# RSTP Proposal et Agreement process (1/2)



- ➔ Switch A a un chemin vers Root par B et C
- ➔ Les 2 ports P0 et P1 sont en designated blocking state
- ➔ Quand un designated port est en discarding ou learning, il met le Proposal bit du BPDU à 1 et envoie ce BPDU
- ➔ A met alors tous les autres non-edge designated port en blocking : cela évite les boucles pendant le proposal-agreement process
  - ➔ Ce opération est dite être l'opération sync
- ➔ Switch A envoie un agreement à Root et met le P1 en Forwarding

# RSTP Proposal et Agreement process (2/2)



- ➔ Ensuite A va envoyer un proposal à B qui va faire de même
- ➔ Mettre ses non-edge designated port, autre que P5, en sync, envoyer un agreement à A et mettre P5 en forwarding
- ➔ Le processus continue avec les autres switches, « sous » B

# Changement topologique

- ➔ En 802.1D, chaque changement génère un BPDU de type Topologie Change Notification (TCN)
- ➔ Quand un switch constate un changement topologique, il envoie au Root Bridge un TCN
- ➔ Le Root Bridge met le flag TC à 1 et diffuse à tous les switches l'information
  - ➔ A la réception d'un BPDU avec le flag TC à 1, toutes les adresses MAC associées aux non edge ports sont vidées afin de refaire un apprentissage

# Mécanismes après un TCN

Action	Notes
The RSTP bridge starts the TC-while timer.	RTSP sets the TC-while timer with a value equal to twice the hello time for all its non-edge designated ports and the root port, if necessary.
The RSTP bridge flushes the MAC addresses associated with all these ports.	
The TC flag bit is set on all outbound BPDUs.	BPDUs are sent on the root port as long as the While timer is active.
The bridge receives a BPDU with the TC bit set from a neighbor and clears the MAC addresses on all ports.	The port that received the TC BPDU retains learned MAC addresses.
The bridge starts the TC-while timer and sends BPDUs with a TC bit set out of all its designated ports and root port.	RSTP does not use the specific TCN BPDU, unless a legacy bridge needs to be notified.

# Des protocoles de Spanning Tree

## La configuration de RSTP

# La configuration

- ➔ Par défaut, les switchs Cisco implémente PVST+
  - ➔ Une instance STP 802.1D par VLAN
- ➔ Il est possible de mettre en place RSTP
  - ➔ Le mode est alors Rapid PVST+ (RPVST+)
  - ➔ On a alors un RSTP par VLAN

# Les commandes

Command	Description
Switch(config) # <b>spanning-tree mode rapid-pvst</b>	Sets spanning tree mode to Rapid PVST+
Switch# <b>show spanning-tree</b> <i>vlan vlan-number</i> [ <b>detail</b> ]	Shows commands that are VLAN-based rather than instance-based
Switch# <b>debug spanning-tree pvst+</b>	Debugs PVST+ events
Switch# <b>debug spanning-tree switch state</b>	Debugs port state changes

# Des protocoles de Spanning Tree

## Le protocole MSTP

# Objectif et avantages de MSTP

- ➔ Réduire le nombre d'instance de spanning tree et diminuer le charge CPU
- ➔ Avec des spanning tree par VLAN, on peut avoir jusque 4094 instances de spanning tree en simultanée !
- ➔ MSTP permet d'avoir une convergence plus rapide que PVST+
  - ➔ Permet également d'avoir une compatibilité avec 802.1D STP, 802.1w RSTP et Cisco PVST+ architecture
- ➔ Permet de faire des spanning tree à travers les trunks en regroupant des VLAN ensemble et en les associant dans des instances
- ➔ Améliore la tolérance aux pannes car si une instance est touchée, les autres ne le sont pas, comparativement au Common Spanning Tree qui fait une unique instance pour tous les VLAN
- ➔ Interconnecter des switchs qui partagent les mêmes informations MSTP permet de définir des régions

# Les régions MSTP

- ➔ Les VLAN peuvent être combinés en instances logiques de spanning tree
- ➔ Cela augmente le problème de savoir comment regrouper les VLAN en instance
  - ➔ Mais hors de propos avec 802.1D qui fait du CSTP
  - ➔ Et dans PVST+, nous avons une instance par VLAN et donc un BPDU par VLAN
- ➔ Pour faire ce regroupement, les switchs utilisent 3 paramètres
  - ➔ Un nom de configuration (32 octets)
  - ➔ Un numéro de révision (2 octets)
  - ➔ Un table de 4096 entrées qui associe les 4096 potentiels VLAN avec une instance : appelé VLAN-to-instance
- ➔ Chaque switch qui partage ces mêmes informations forment une région
- ➔ Ces paramètres doivent être administrer manuellement, rien n'est automatique

## Reconnaître une région ?

- ➔ Comment un switch peut savoir s'il est dans la même région qu'un voisin ?
- ➔ Dans le BPDU, un digest est envoyé, calculé à partir du nom de la configuration et de la table d'association VLAN-to-Instance
- ➔ A la réception d'un BPDU, un switch, par comparaison, sait donc s'il est dans la même région ou pas.

## Extended System ID

- ➔ Comme dans PVST, le bridge ID est sur 8 octets et inclus l'extended System ID
- ➔ Dans PVST, on inclut le numéro du VLAN, derrière la priorité du switch
- ➔ Dans MSTP, le 12 bit Extended System ID contient le numéro de l'instance

# Interaction entre 802.1D et MSTP région

- ➔ D'après la spécification IEEE 802.1s, un switch MSTP doit être en mesure de manipuler au moins un internal spanning tree (IST).
  - ➔ Dans une région MSTP, un IST est donc fonctionnel, qui peut être vu comme un CST, confiné dans une région
- ➔ Une région MSTP apparaît comme un unique switch virtuel aux switches voisins en CST ou aux autres régions MSTP
  - ➔ La région MSTP utilise les mêmes rôles et opérations pour ses ports que RSTP
- ➔ Dans chaque région, un spanning tree a lieu, appelé MSTP instance
  - ➔ L'IST est numéroté 0 et les MSTP instance de 1 à 15
  - ➔ Chaque MSTP instance est locale à sa région
- ➔ Dans le BPDU d'une instance MSTP figure un champ M-Record
  - ➔ Il contient le root bridge et sender bridge priority parameters

# Configuration MSTP

```
Switch(config)#spanning-tree mst configuration
```

- Enters MST configuration submode

```
Switch(config-mst)#name name
```

- Sets the MST region name

```
Switch(config-mst)#revision rev_num
```

- Sets the MST configuration revision number

```
Switch(config-mst)#instance inst vlan range
```

- Maps the VLANs to an MST instance

```
Switch(config-mst)#spanning-tree mst instance_number root  
primary|secondary
```

- Establishes primary and secondary roots for MST instance

# Vérification configuration MSTP

```
Switch#show spanning-tree mst configuration
```

- Displays MSTP configuration information

```
Switch#show spanning-tree mst configuration
```

```
Name      [cisco]
```

```
Revision  1
```

```
Instance  Vlans mapped
```

```
-----
```

```
0         11-4094
```

```
1         1-10
```

```
-----
```

# Vérification configuration MSTP

```
Switch#show spanning-tree mst instance_number
```

- Displays configuration information for a specific MSTP instance

```
Switch#show spanning-tree mst 1
```

```
##### MST01          vlans mapped: 1-10
Bridge      address 00d0.00b8.1400  priority 32769 (32768 sysid 1)
Root       this switch for MST01
```

Interface	Role	Sts	Cost	Prio.Nbr	Status
Fa4/4	Back	BLK	1000	240.196	P2p
Fa4/5	Desg	FWD	200000	128.197	P2p
Fa4/48	Boun	FWD	200000	128.240	P2p Bound(STP)

```
Switch#clear spanning-tree detected-protocols [interface interface-id]
```

- Forces renegotiation with neighboring switches during migration process

# Vérification de configuration

- ➔ Vérification des informations MSTP d'une interface
  - ➔ `show spanning-tree mst interface fastethernet 4/4`
- ➔ Vérification des informations MSTP d'une instance donnée et d'une interface
  - ➔ `show spanning-tree mst 1 interface fastethernet 4/4`
- ➔ Vérification des informations MSTP d'une instance donnée en détail
  - ➔ `show spanning-tree mst 1 detail`

# Des protocoles de Spanning Tree

QUESTIONS ?