upna
Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Spanning Tree Protocol

Area de Ingeniería Telemática
http://www.tlm.unavarra.es

Grado en Ingeniería en Tecnologías de Telecomunicación, 3º

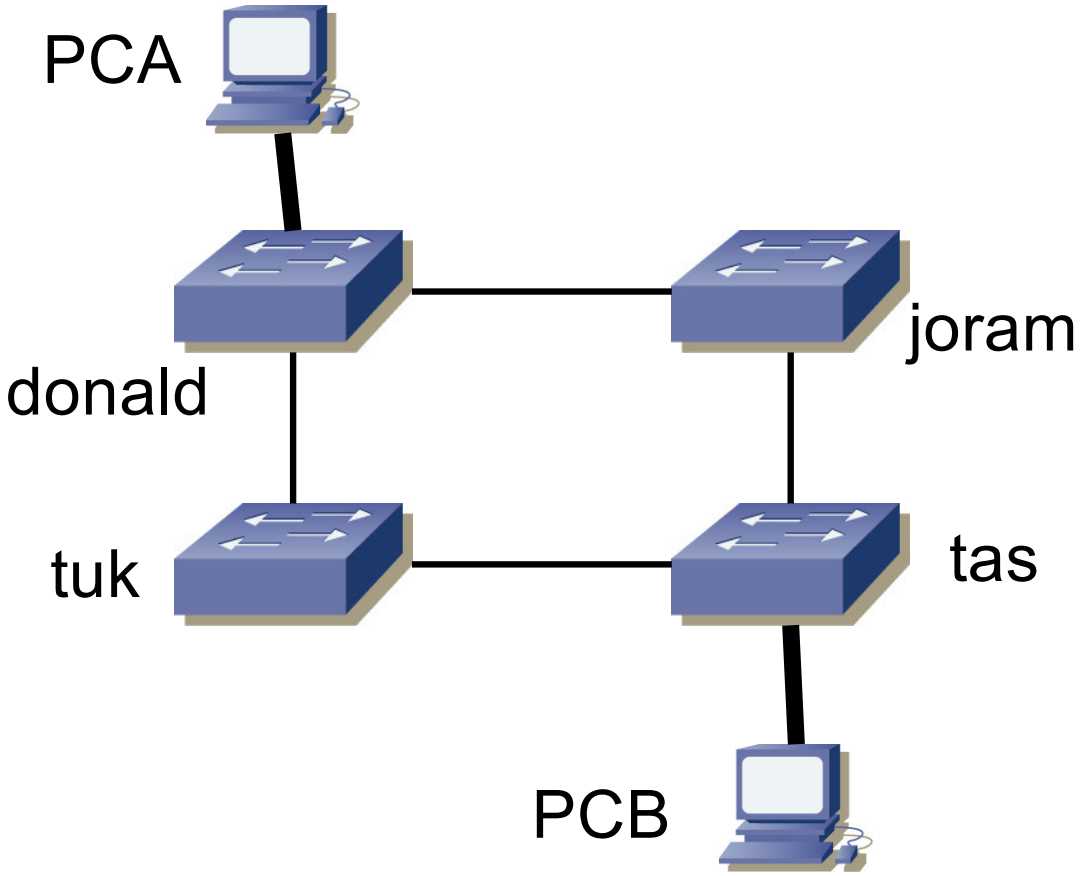
upna

Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Bucles en LANs con puentes

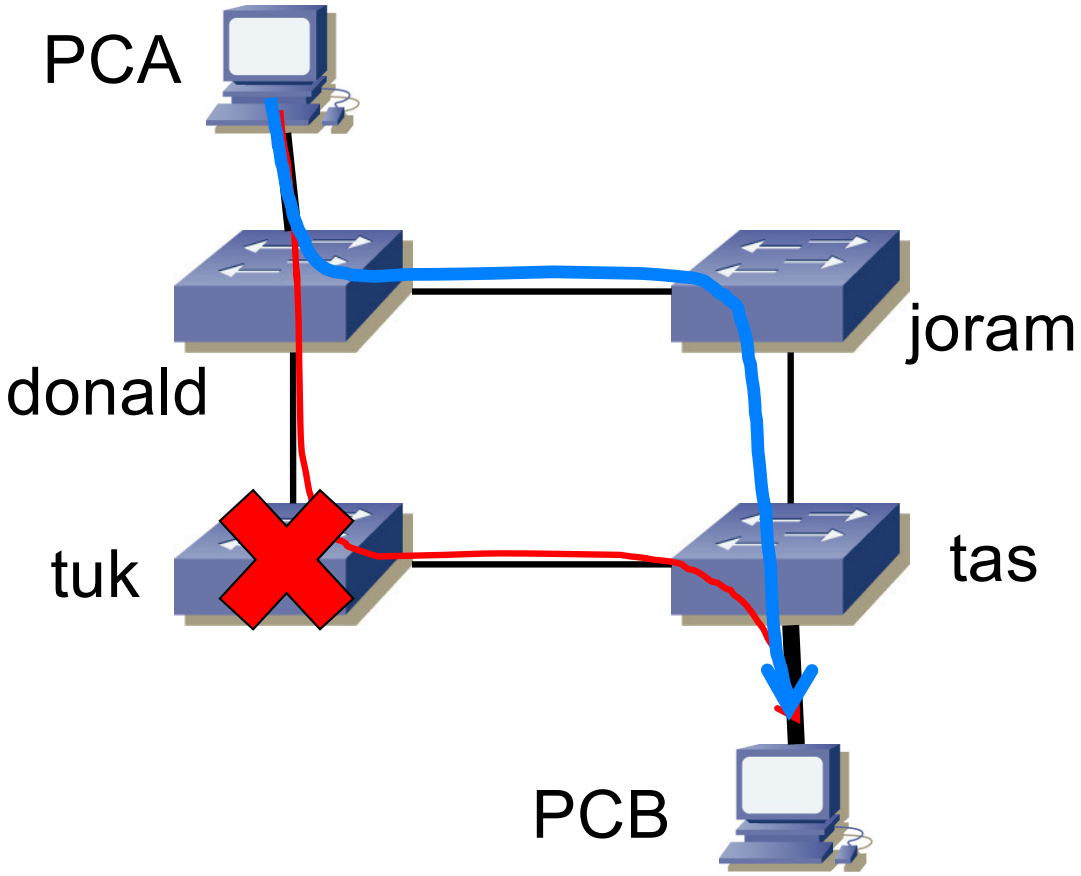
Caminos alternativos

- Ofrecerían la posibilidad de:
  - Reconfiguración ante fallos (...)
  - (...)



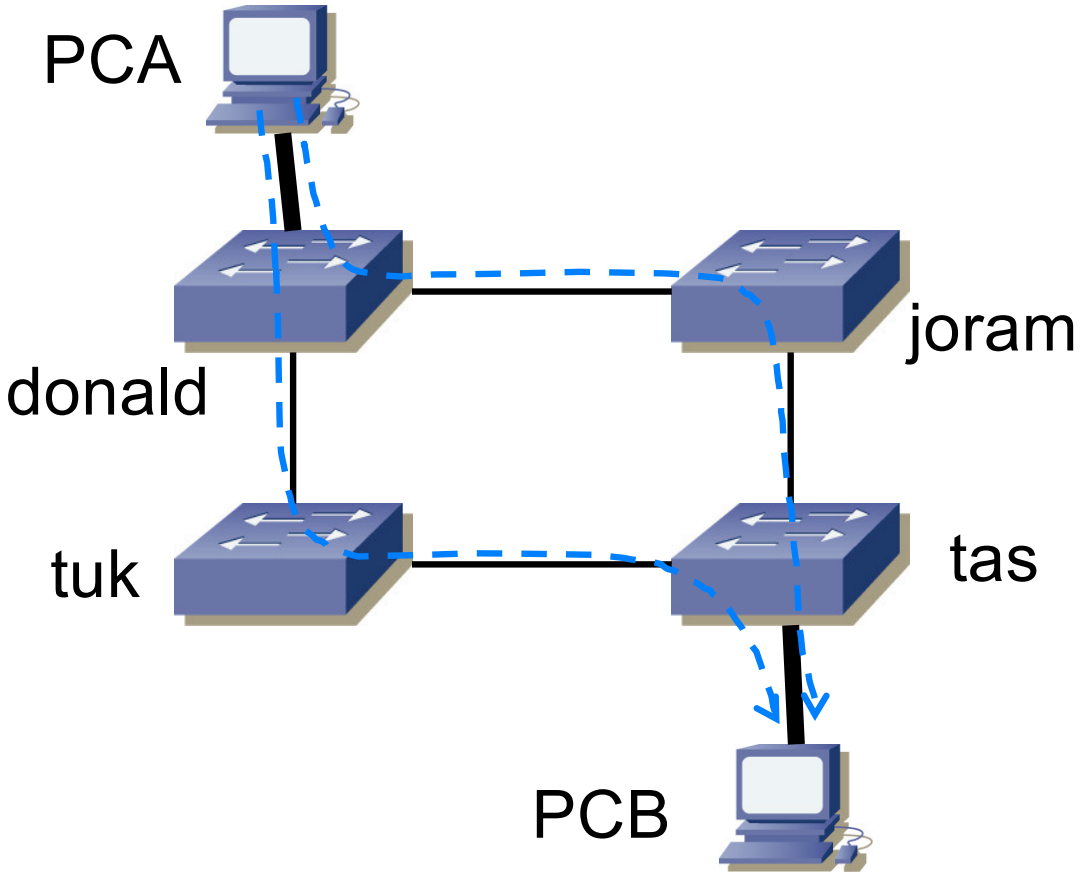
Caminos alternativos

- Ofrecerían la posibilidad de:
  - Reconfiguración ante fallos (...)
  - (...)



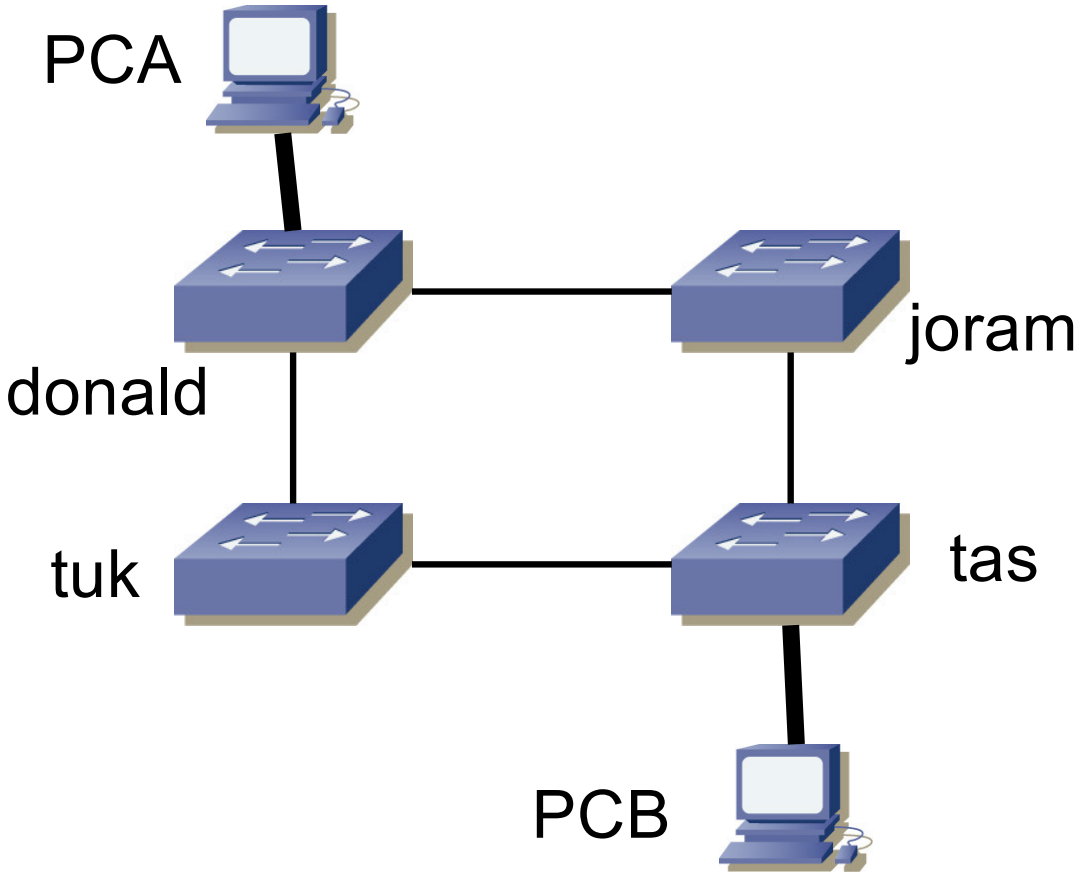
Caminos alternativos

- Ofrecerían la posibilidad de:
  - Reconfiguración ante fallos
  - Balanceo de carga
- Requiere tomar decisiones de encaminamiento



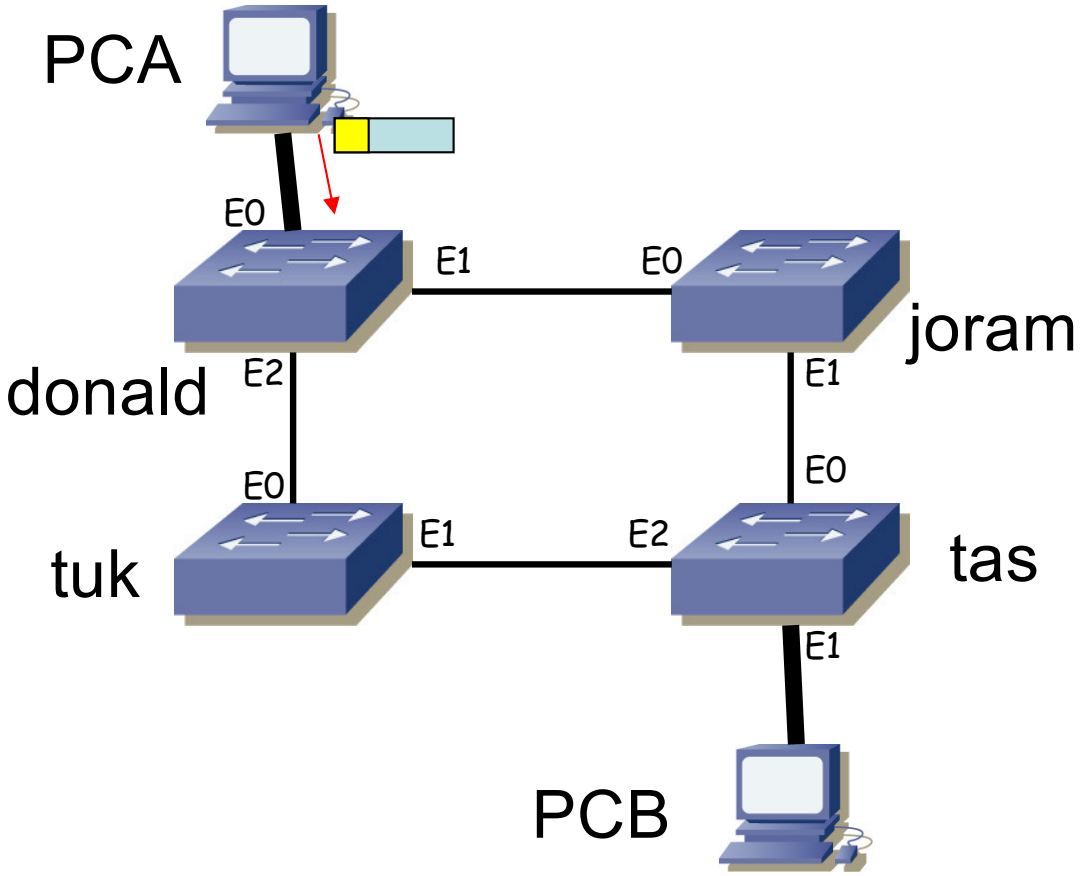
Pero...

- Ejemplo:
  - PCA envía una trama al PCB y los switches no han aprendido la MAC de PCB
  - O PCA envía una trama a broadcast
  - En ambos casos habrá inundación



Pero...

- Ejemplo:
  - PCA envía una trama al PCB y los switches no han aprendido la MAC de PCB
  - O PCA envía una trama a broadcast
  - En ambos casos habrá inundación



donald

If	MAC

tuk

If	MAC

joram

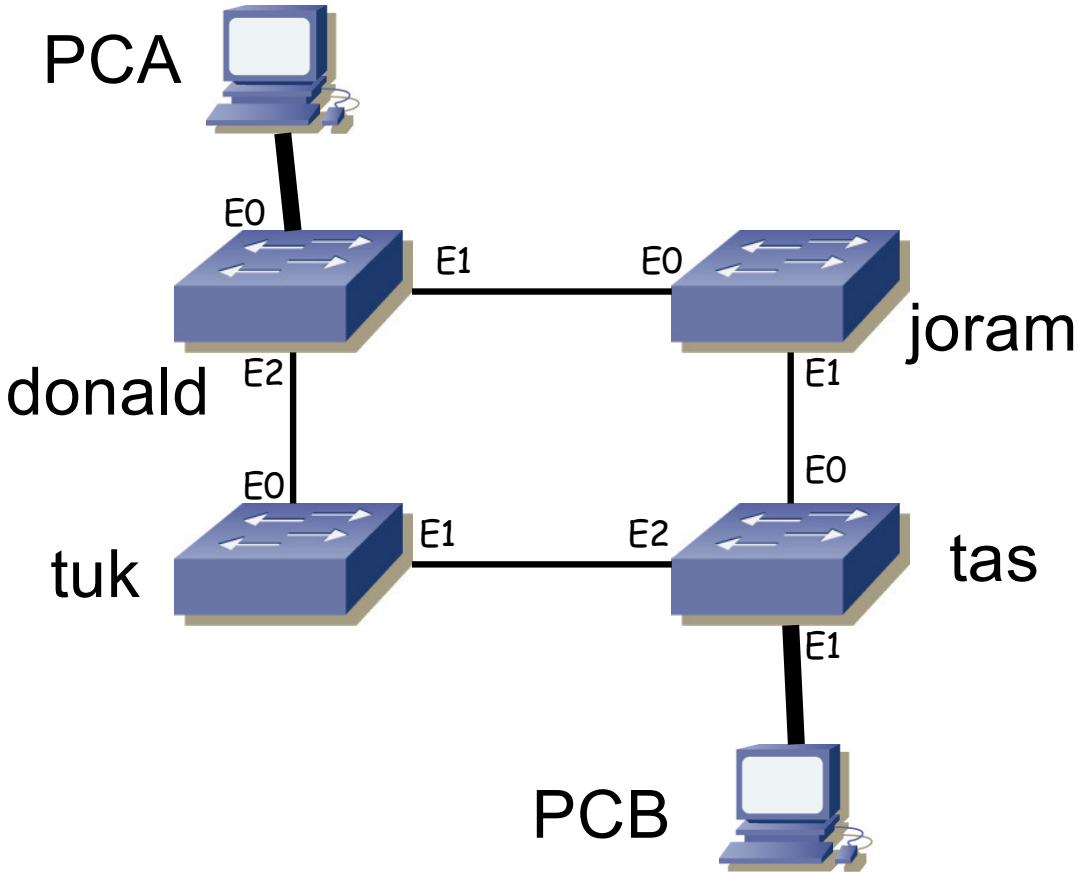
If	MAC

tas

If	MAC

Pero...

- Ejemplo:
  - PCA envía una trama al PCB y los switches no han aprendido la MAC de PCB
  - O PCA envía una trama a broadcast
  - En ambos casos habrá inundación



donald

If	MAC
E0	PCA

tuk

If	MAC

joram

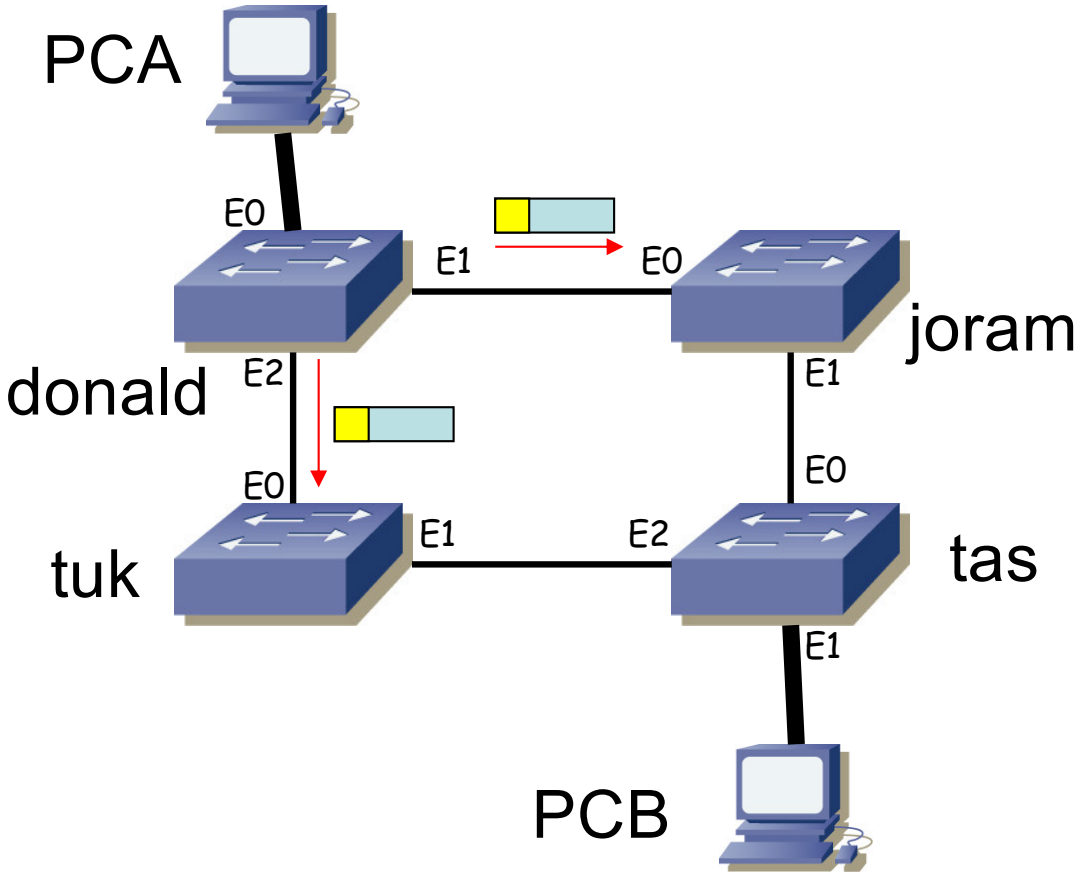
If	MAC

tas

If	MAC

Pero...

- Ejemplo:
  - PCA envía una trama al PCB y los switches no han aprendido la MAC de PCB
  - O PCA envía una trama a broadcast
  - En ambos casos habrá inundación



donald

If	MAC
E0	PCA

tuk

If	MAC

joram

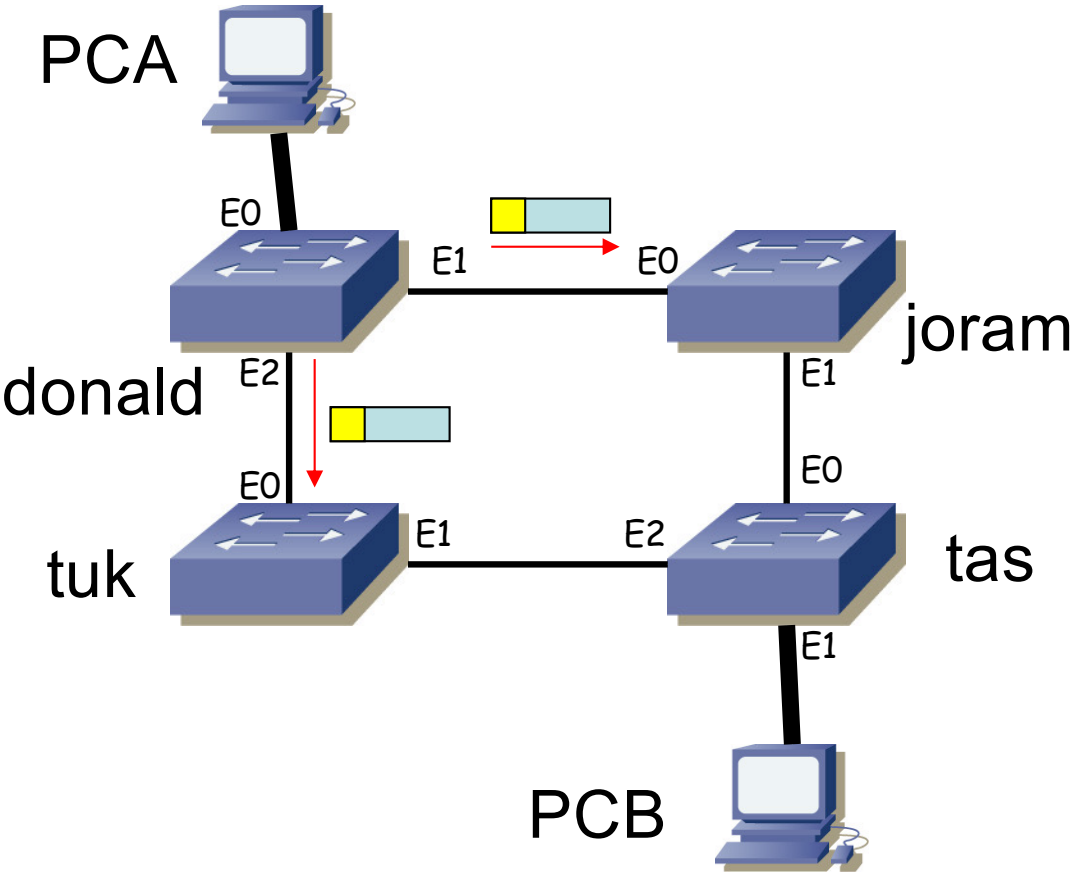
If	MAC

tas

If	MAC

Pero...

- Ejemplo:
  - PCA envía una trama al PCB y los switches no han aprendido la MAC de PCB
  - O PCA envía una trama a broadcast
  - En ambos casos habrá inundación



donald

If	MAC
E0	PCA

tuk

If	MAC
E0	PCA

joram

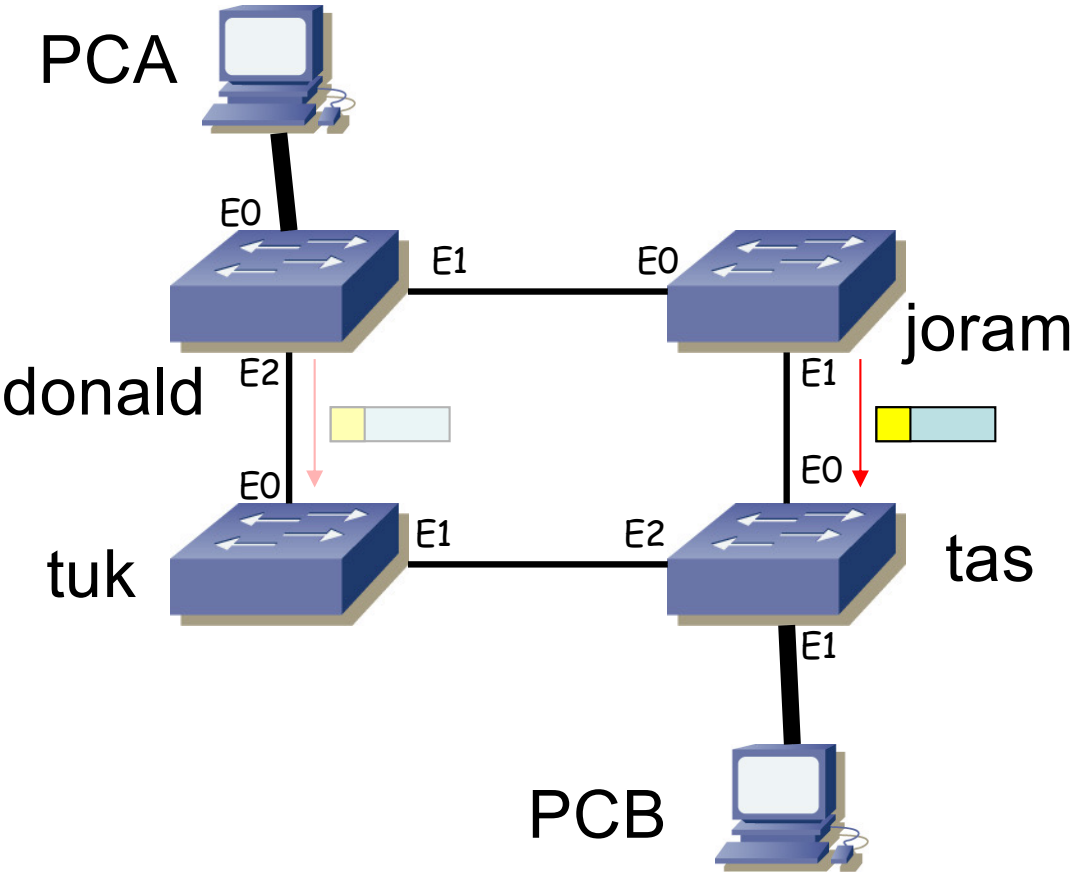
If	MAC
E0	PCA

tas

If	MAC

Pero...

- Ejemplo:
  - Suponemos que el paquete que viene de joram llega antes a tas que el que viene de tuk



donald

If	MAC
E0	PCA

tuk

If	MAC
E0	PCA

joram

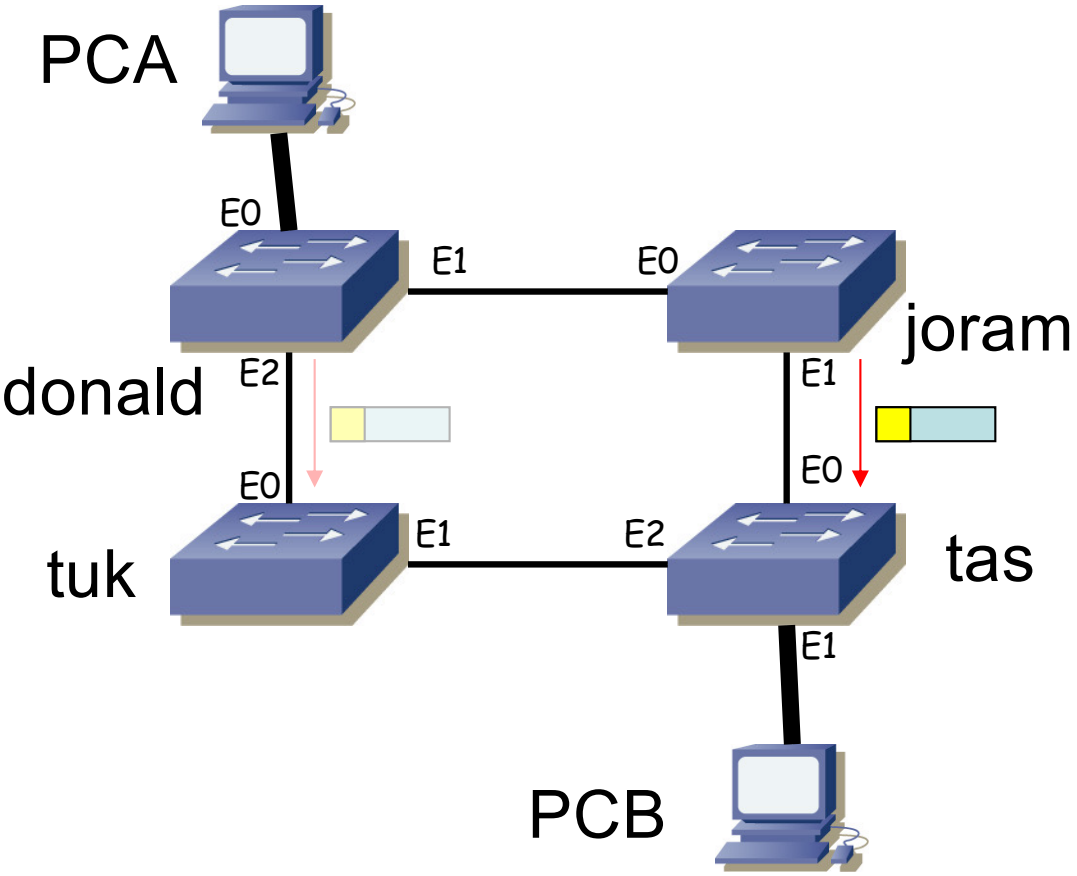
If	MAC
E0	PCA

tas

If	MAC

Pero...

- Ejemplo:
  - Suponemos que el paquete que viene de joram llega antes a tas que el que viene de tuk



donald

If	MAC
E0	PCA

tuk

If	MAC
E0	PCA

joram

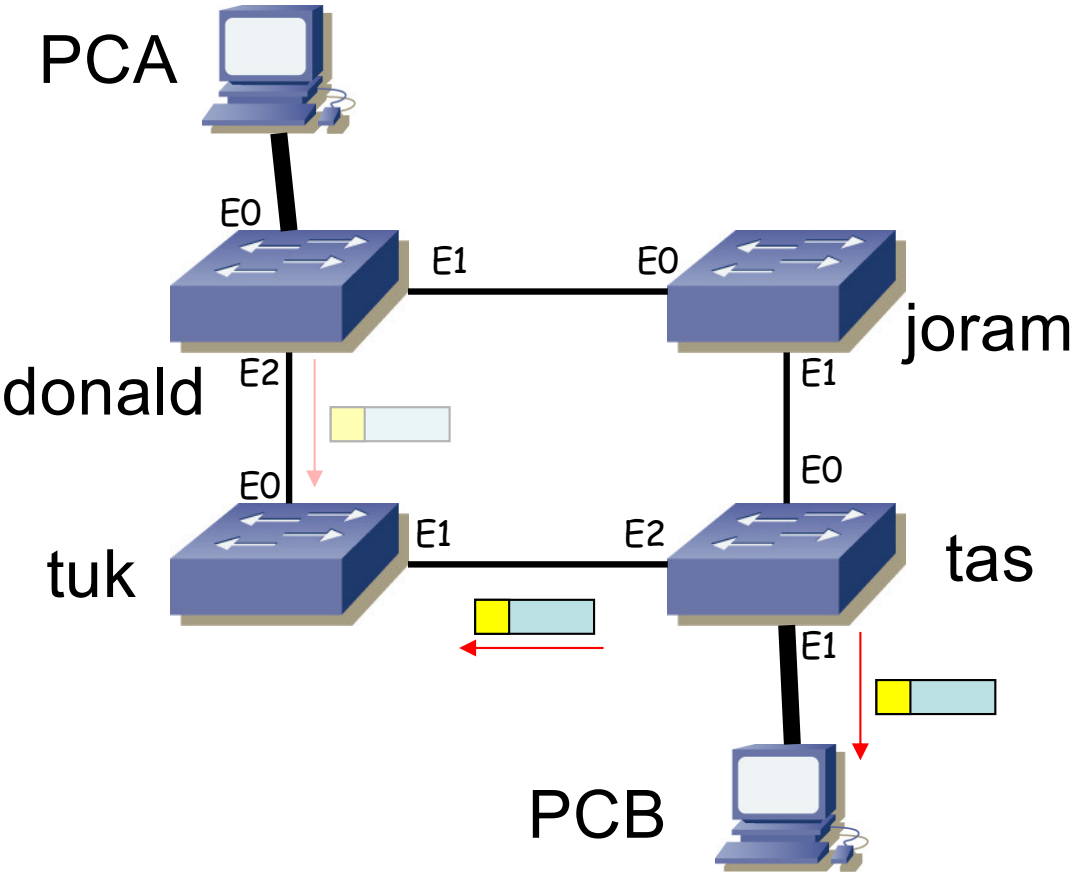
If	MAC
E0	PCA

tas

If	MAC
E0	PCA

Pero...

- Ejemplo:
  - Suponemos que el paquete que viene de joram llega antes a tas que el que viene de tuk



donald

If	MAC
E0	PCA

tuk

If	MAC
E0	PCA

joram

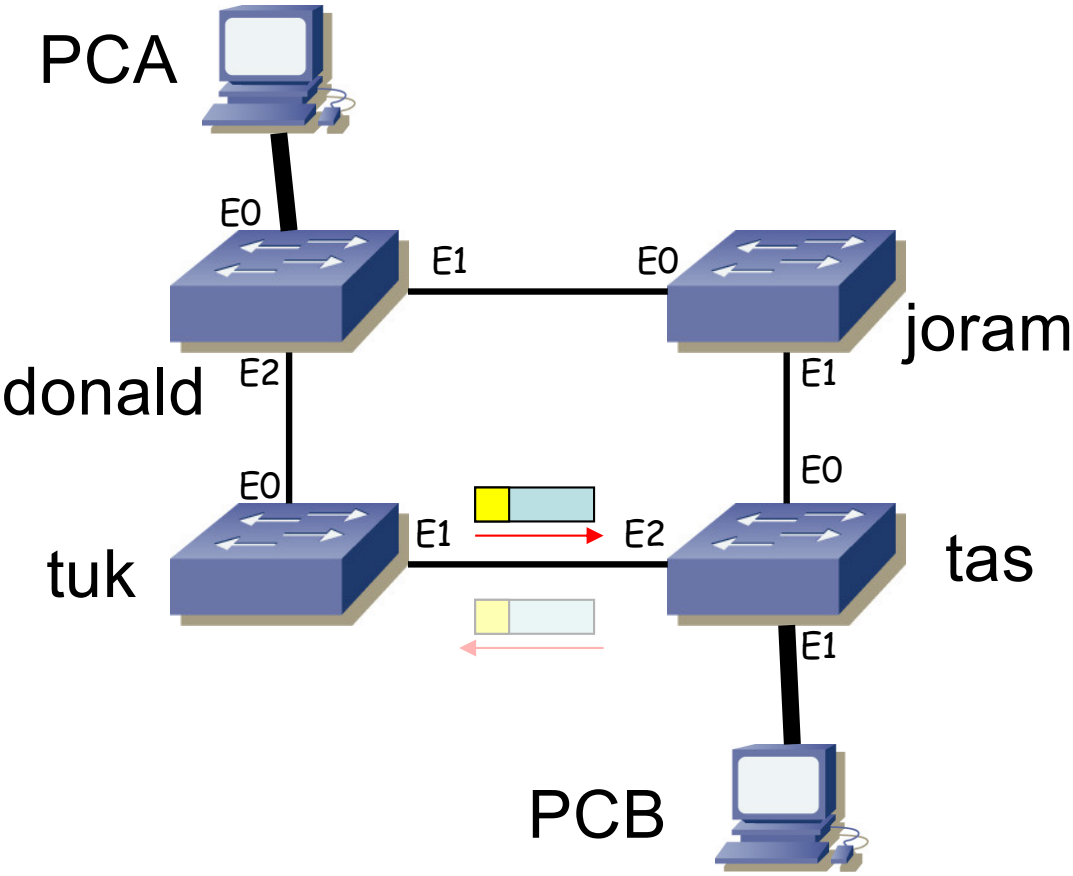
If	MAC
E0	PCA

tas

If	MAC
E0	PCA

Pero...

- Ejemplo:
  - Después llegará el que viene de tuk
  - Atención que aún no hemos terminado con el paquete que va de tas a tuk



donald

If	MAC
E0	PCA

tuk

If	MAC
E0	PCA

joram

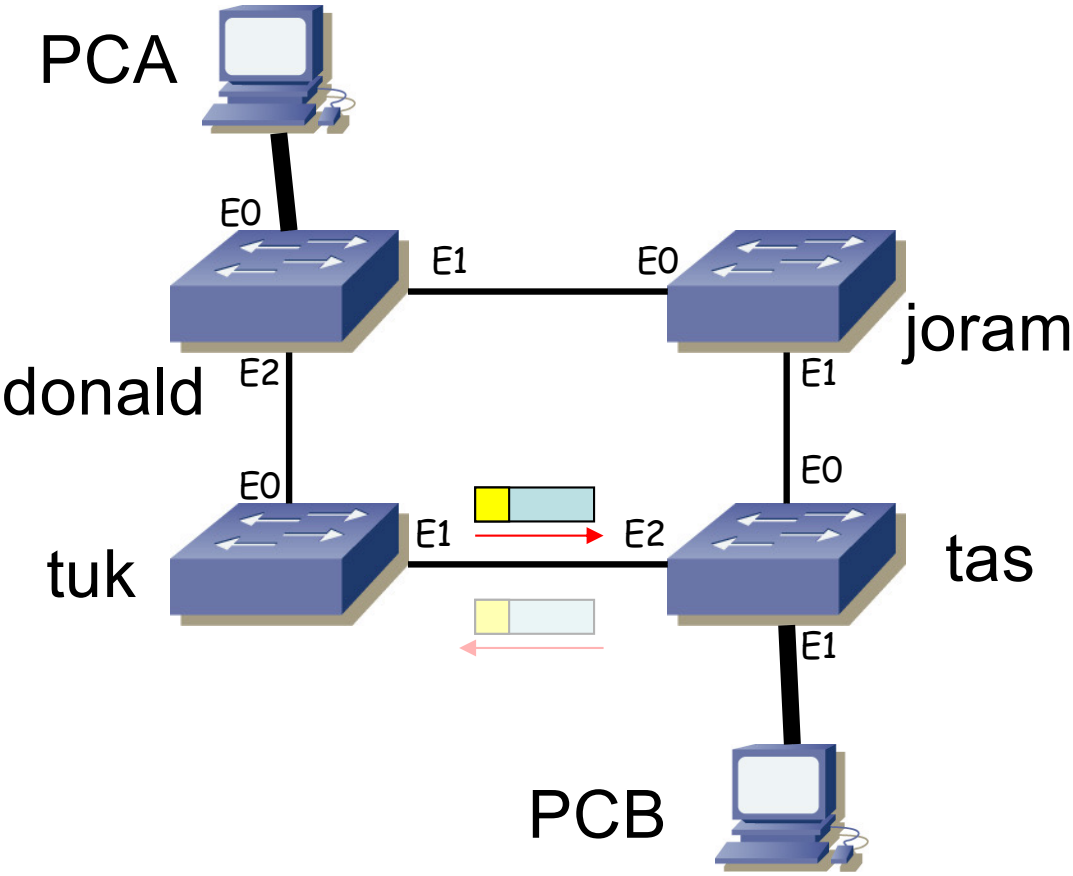
If	MAC
E0	PCA

tas

If	MAC
E0	PCA

Pero...

- Ejemplo:
  - Después llegará el que viene de tuk
  - Atención que aún no hemos terminado con el paquete que va de tas a tuk



donald

If	MAC
E0	PCA

tuk

If	MAC
E0	PCA

joram

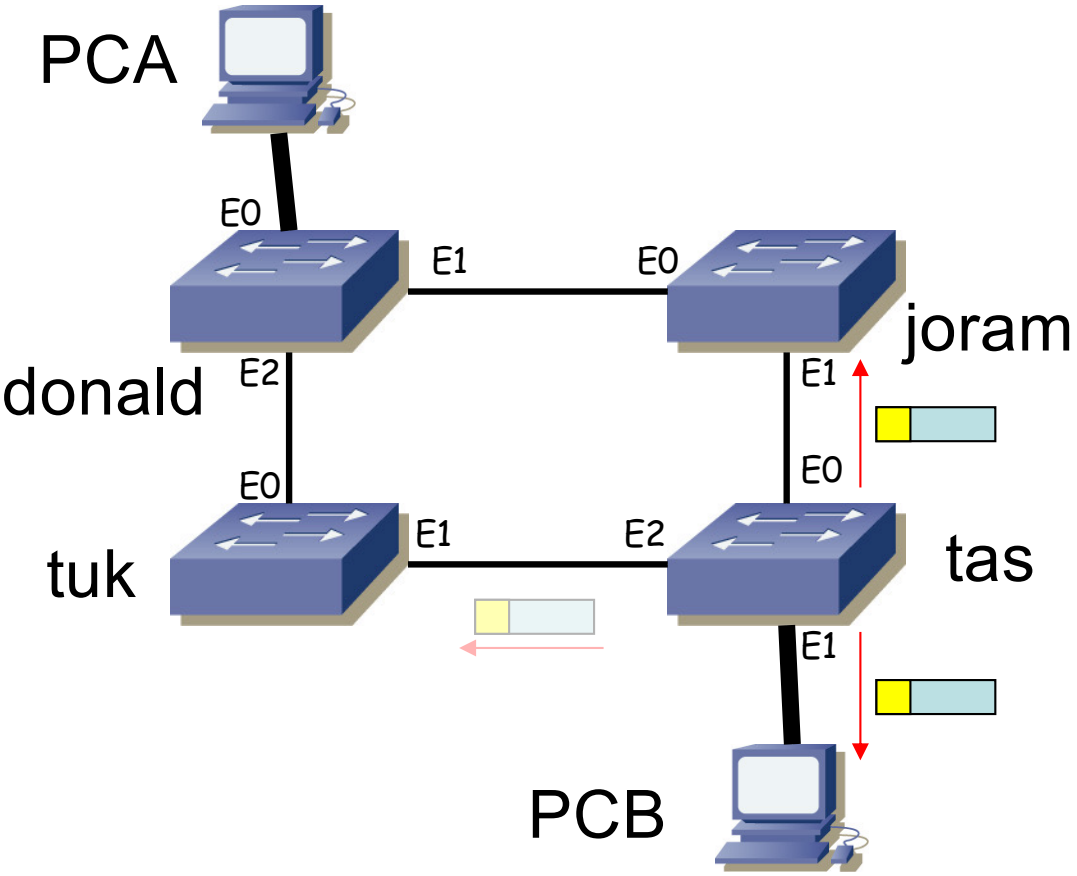
If	MAC
E0	PCA

tas

If	MAC
E2	PCA

Pero...

- Ejemplo:
  - Después llegará el que viene de tuk
  - Atención que aún no hemos terminado con el paquete que va de tas a tuk



donald

If	MAC
E0	PCA

tuk

If	MAC
E0	PCA

joram

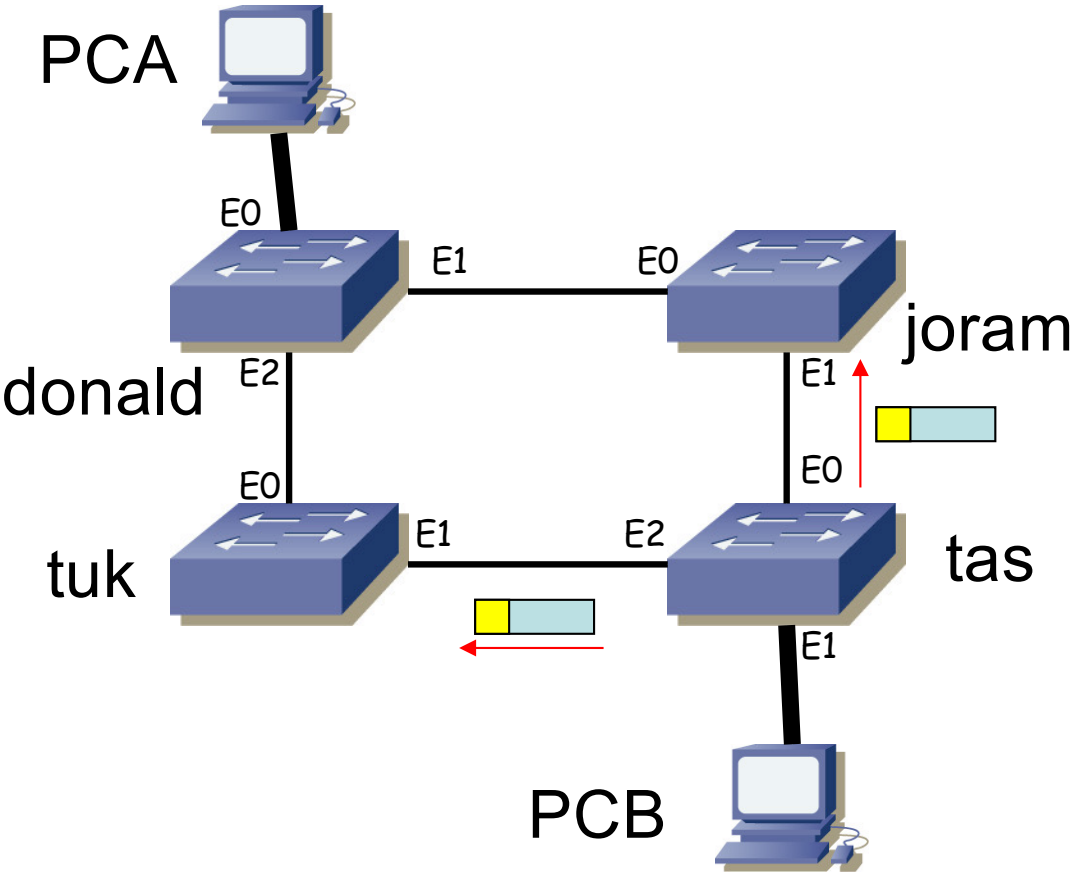
If	MAC
E0	PCA

tas

If	MAC
E2	PCA

Pero...

- Ejemplo:
  - Han llegado 2 paquetes idénticos a PCB
  - Tenemos 2 paquetes que han salido de tas, uno en cada sentido del anillo
  - Seguramente uno de ellos un poco antes que el otro



donald

If	MAC
E0	PCA

tuk

If	MAC
E0	PCA

joram

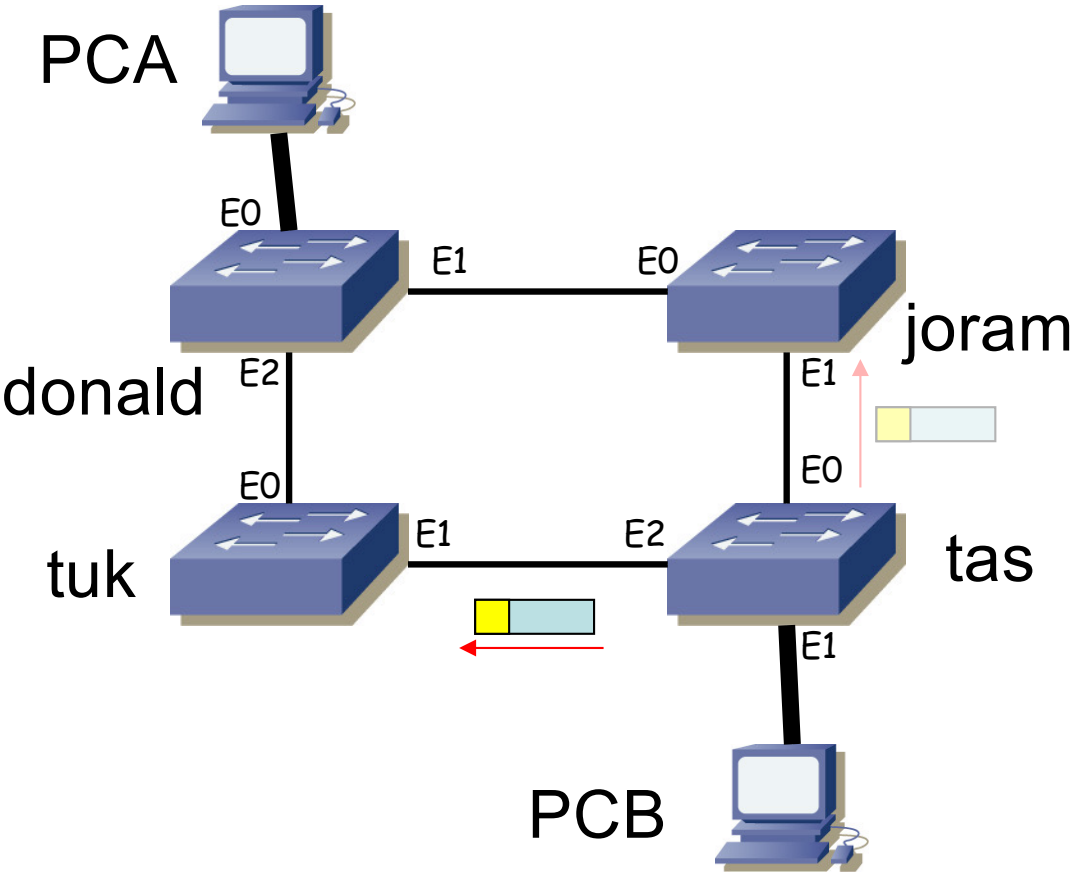
If	MAC
E0	PCA

tas

If	MAC
E2	PCA

Pero...

- Ejemplo:
  - Supongamos primero el que va de tas a tuk



donald

If	MAC
E0	PCA

tuk

If	MAC
E0	PCA

joram

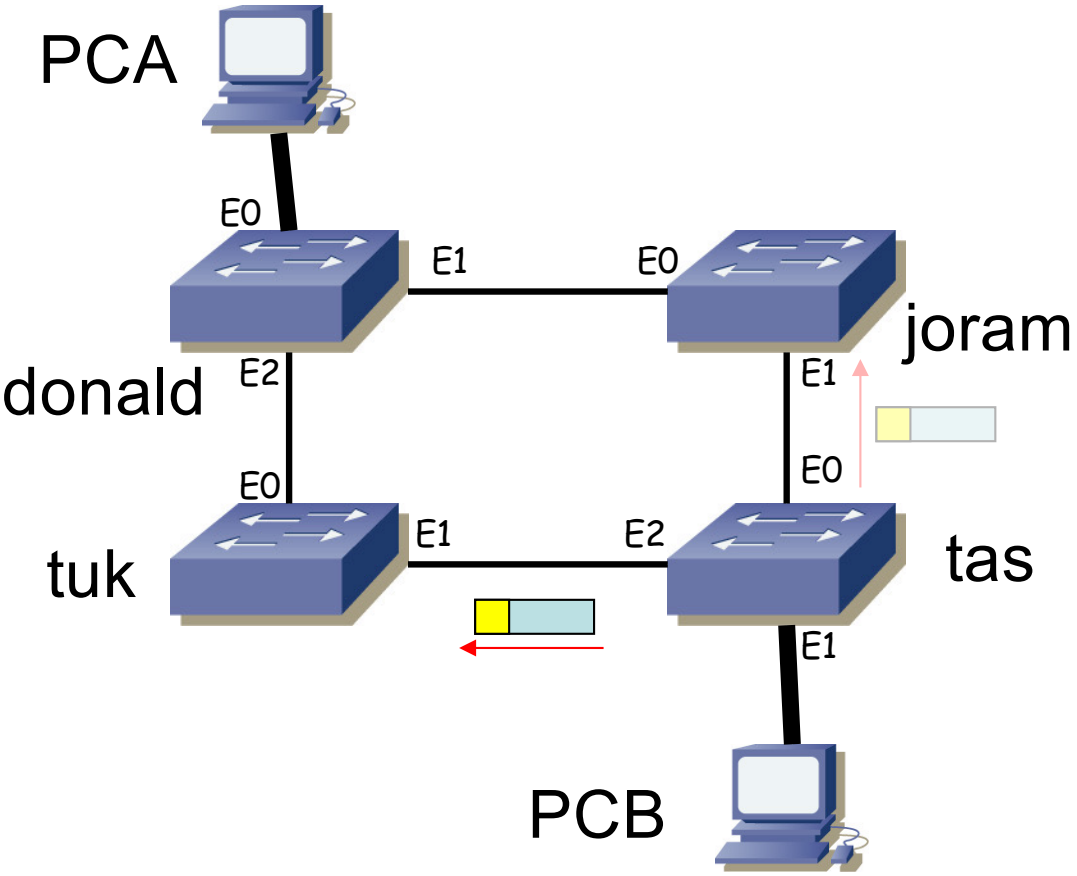
If	MAC
E0	PCA

tas

If	MAC
E2	PCA

Pero...

- Ejemplo:
  - Supongamos primero el que va de tas a tuk



donald

If	MAC
E0	PCA

tuk

If	MAC
E1	PCA

joram

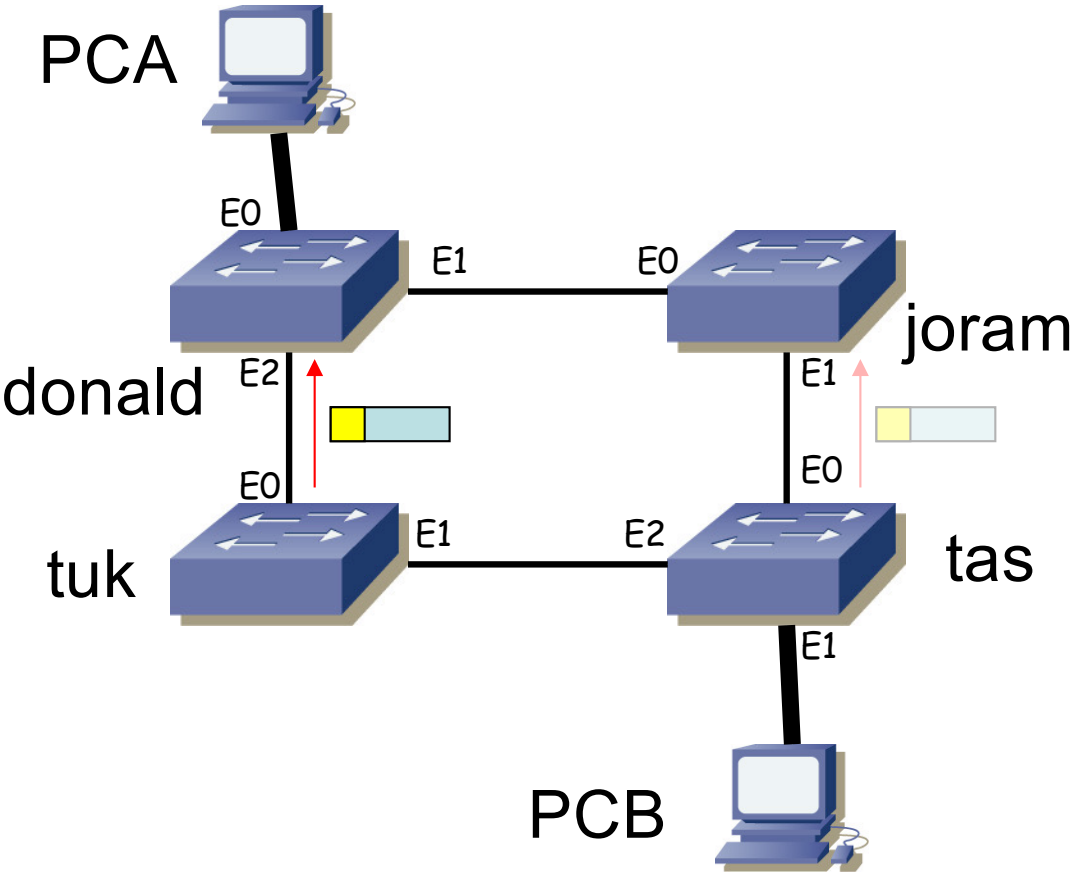
If	MAC
E0	PCA

tas

If	MAC
E2	PCA

Pero...

- Ejemplo:
  - Supongamos primero el que va de tas a tuk



donald

If	MAC
E0	PCA

tuk

If	MAC
E1	PCA

joram

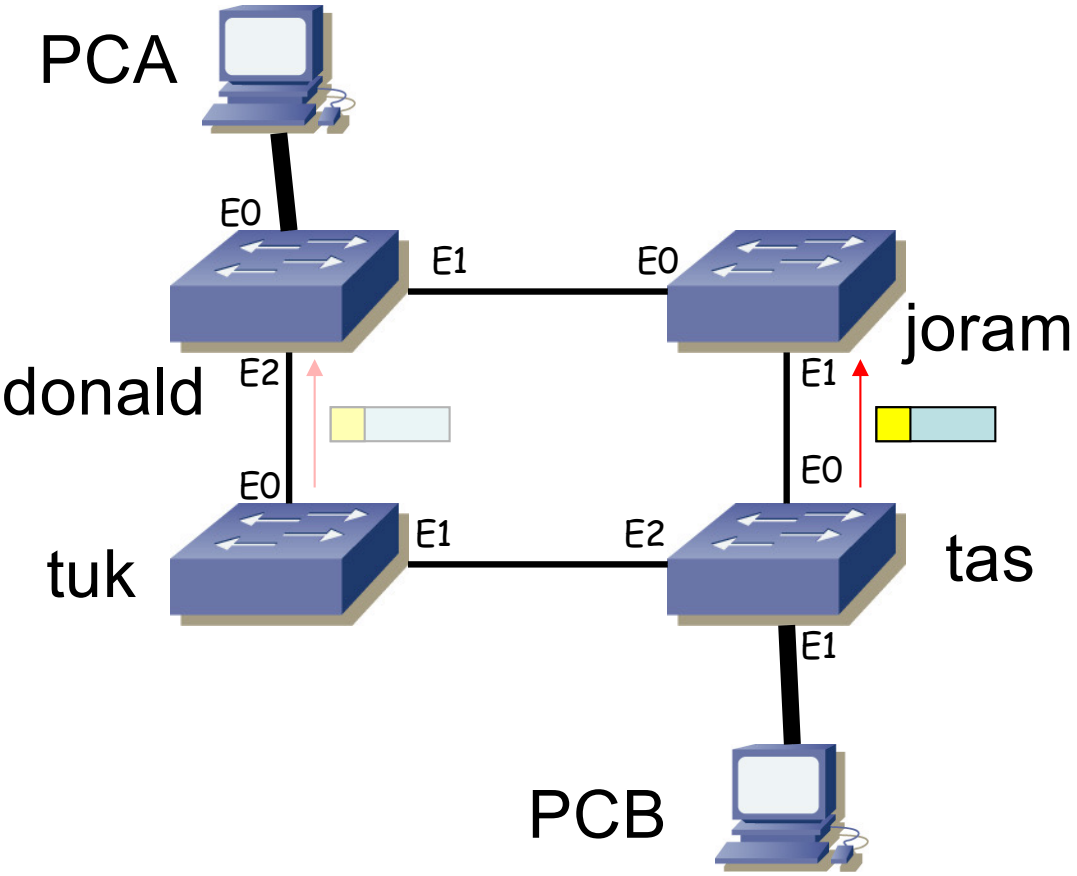
If	MAC
E0	PCA

tas

If	MAC
E2	PCA

Pero...

- Ejemplo:
  - Y ahora el paquete que va de tas a joram



donald

If	MAC
E0	PCA

tuk

If	MAC
E1	PCA

joram

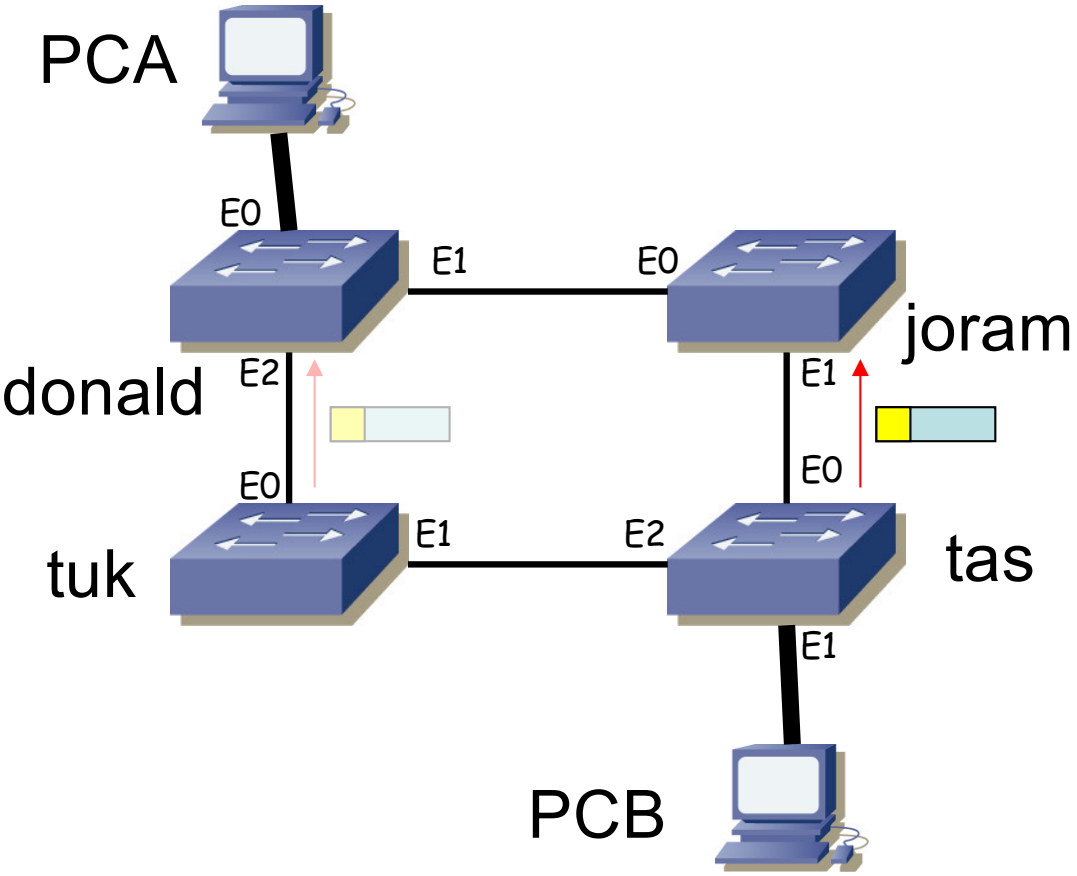
If	MAC
E0	PCA

tas

If	MAC
E2	PCA

Pero...

- Ejemplo:
  - Y ahora el paquete que va de tas a joram



donald

If	MAC
E0	PCA

tuk

If	MAC
E1	PCA

joram

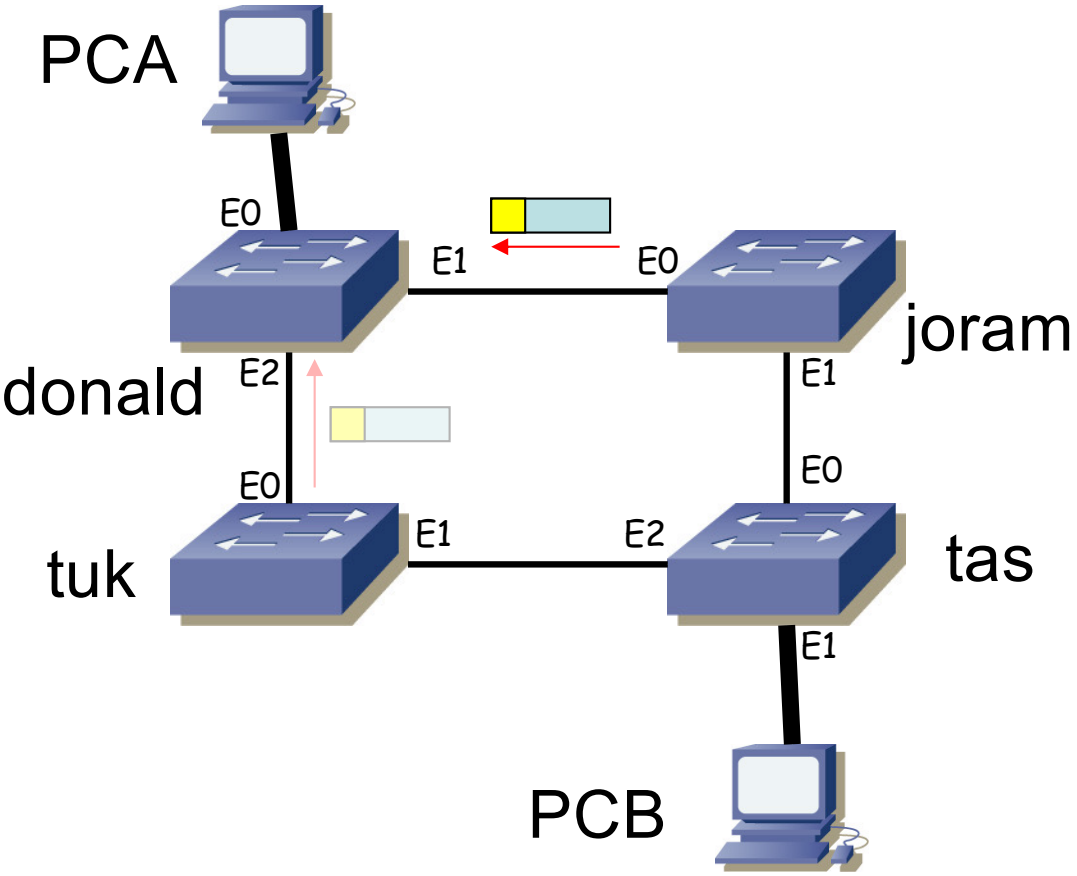
If	MAC
E1	PCA

tas

If	MAC
E2	PCA

Pero...

- Ejemplo:
  - Y ahora el paquete que va de tas a joram



donald

If	MAC
E0	PCA

tuk

If	MAC
E1	PCA

joram

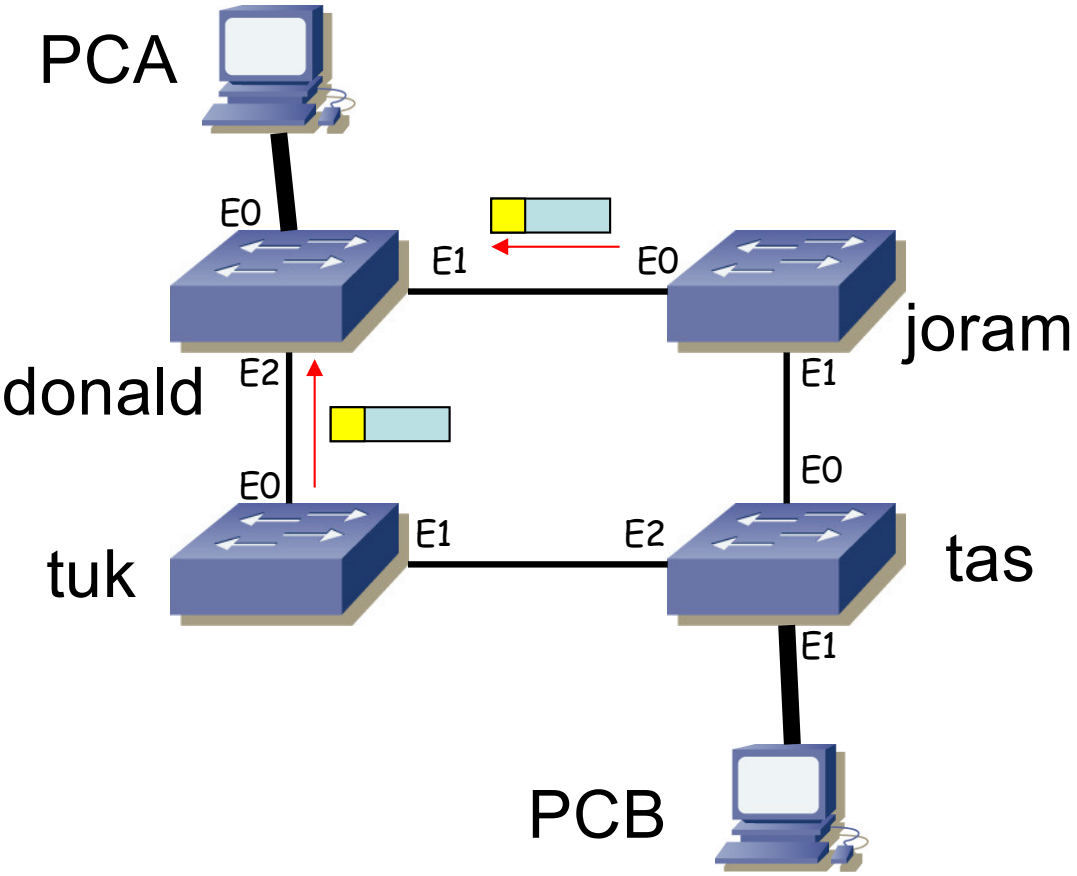
If	MAC
E1	PCA

tas

If	MAC
E2	PCA

Pero...

- Ejemplo:
  - Dos paquetes llegan a donald



donald

If	MAC
E0	PCA

tuk

If	MAC
E1	PCA

joram

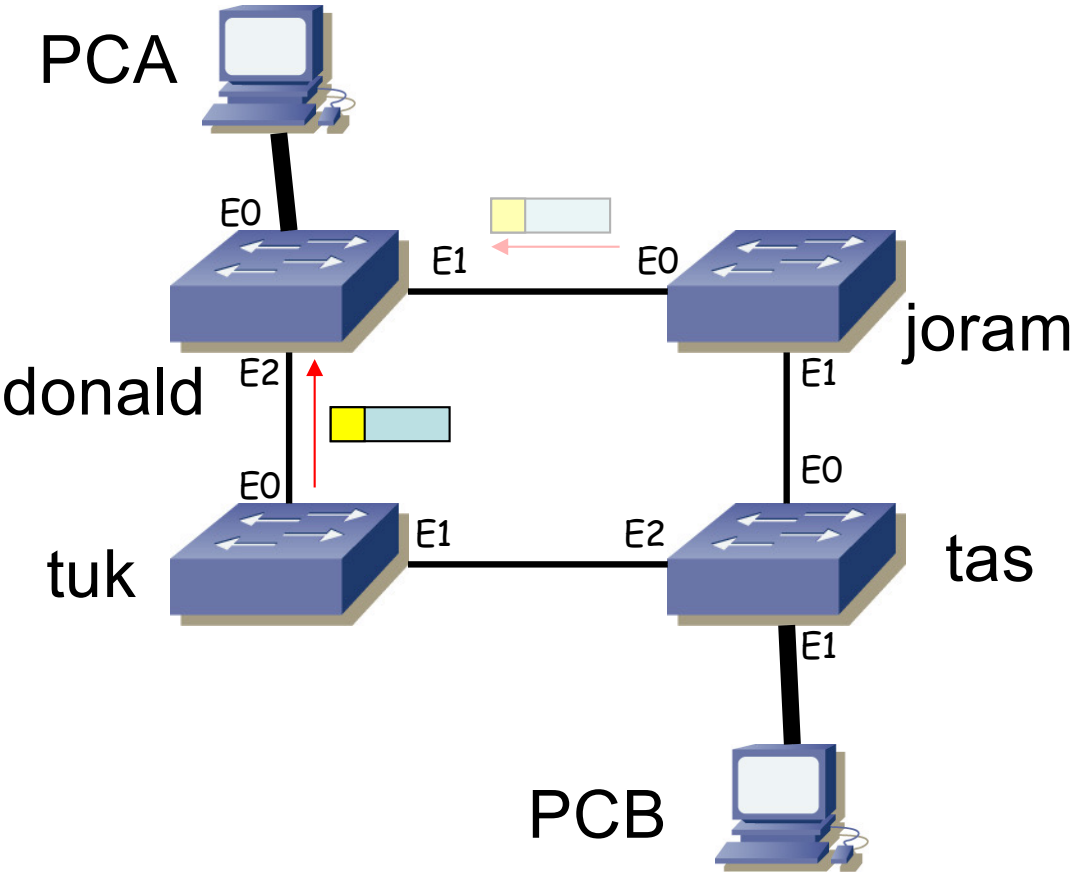
If	MAC
E1	PCA

tas

If	MAC
E2	PCA

Pero...

- Ejemplo:
  - Dos paquetes llegan a donald
  - Supongamos que llega primero el que viene de tuk



donald

If	MAC
E0	PCA

tuk

If	MAC
E1	PCA

joram

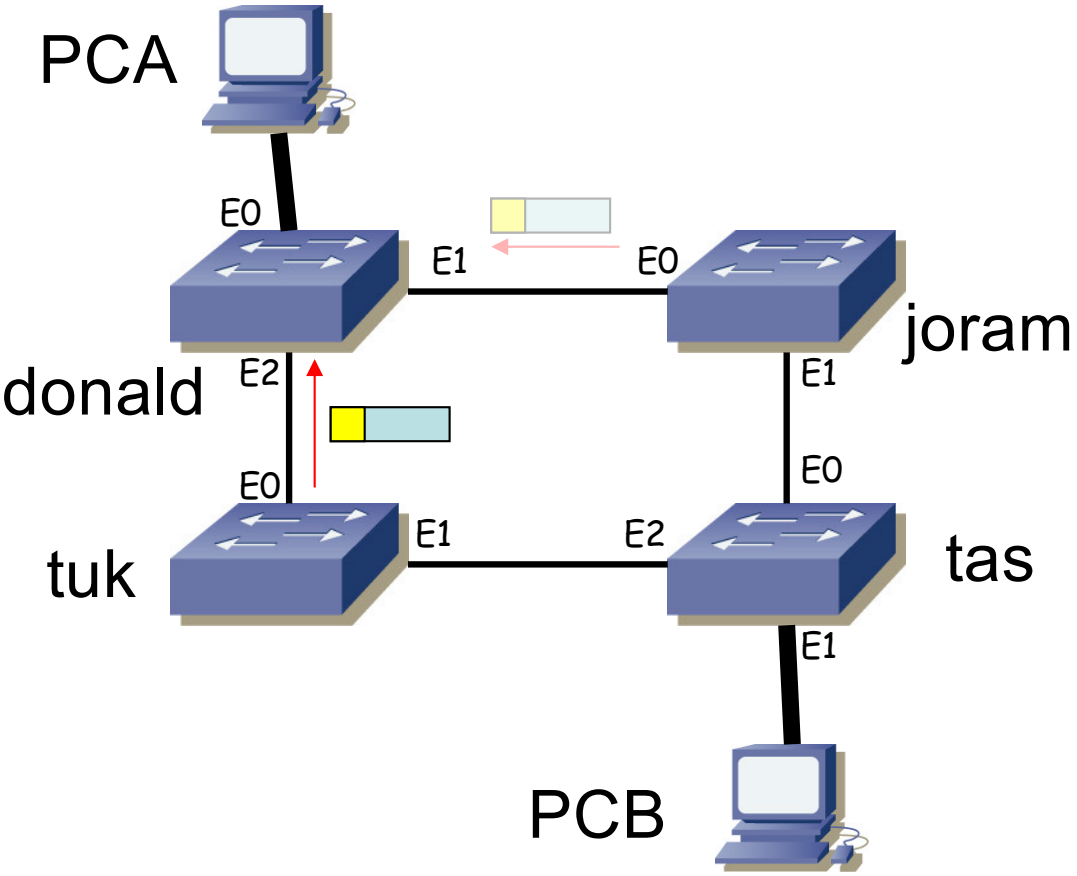
If	MAC
E1	PCA

tas

If	MAC
E2	PCA

Pero...

- Ejemplo:
  - donald cambia lo que había aprendido y reenvía el paquete



donald

If	MAC
E2	PCA

tuk

If	MAC
E1	PCA

joram

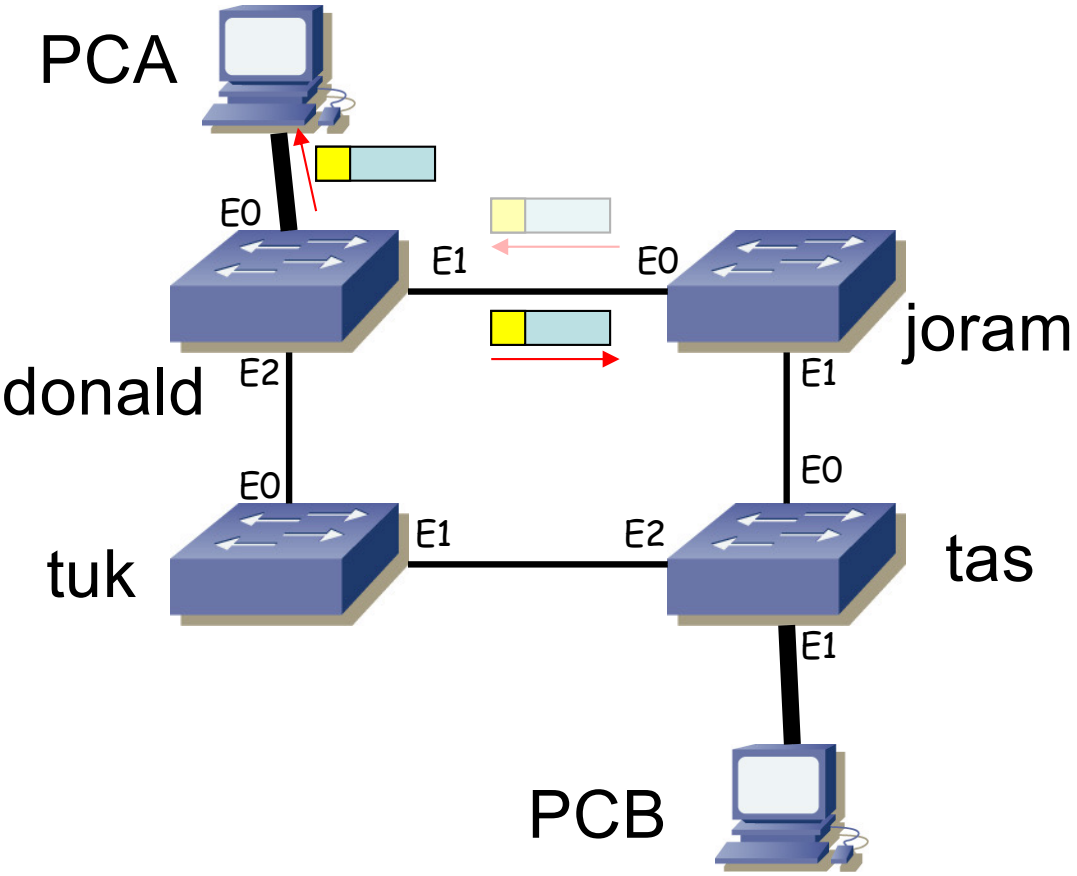
If	MAC
E1	PCA

tas

If	MAC
E2	PCA

Pero...

- Ejemplo:
  - donald cambia lo que había aprendido y reenvía el paquete



donald

If	MAC
E2	PCA

tuk

If	MAC
E1	PCA

joram

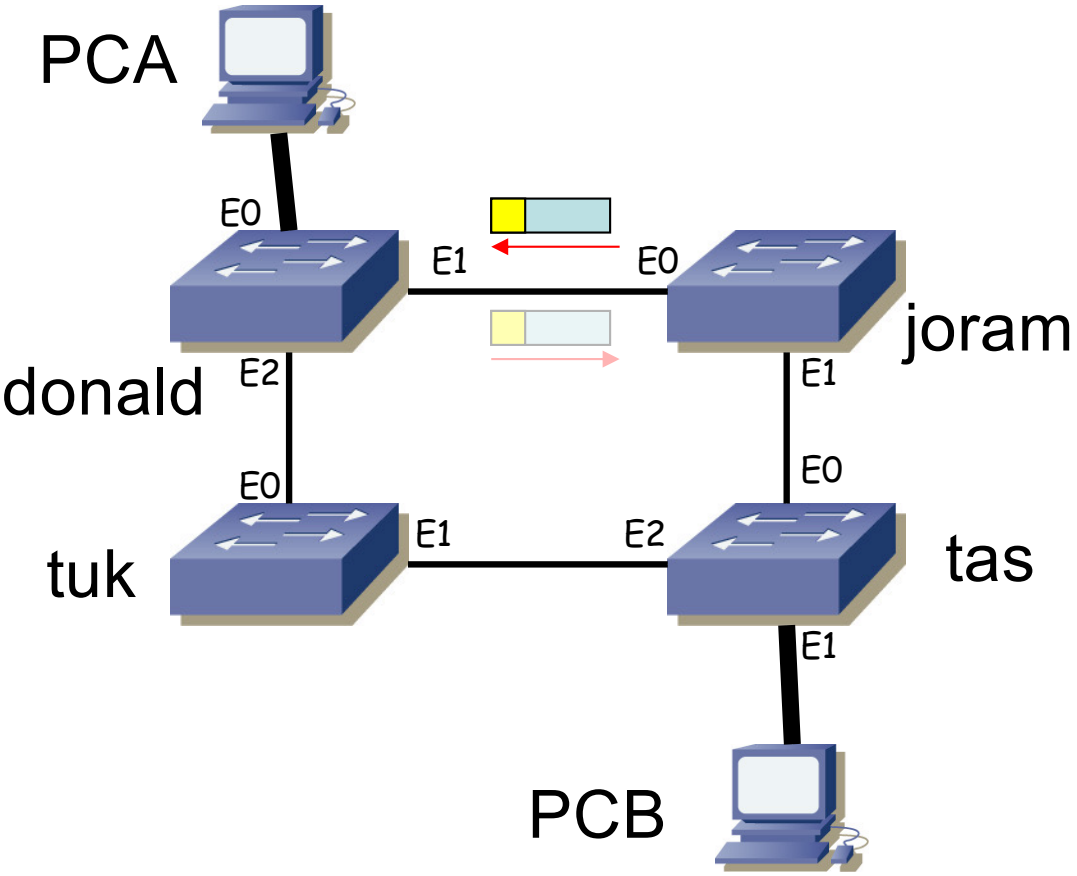
If	MAC
E1	PCA

tas

If	MAC
E2	PCA

Pero...

- Ejemplo:
  - Y ahora veamos con el que viene de joram



donald

If	MAC
E2	PCA

joram

If	MAC
E1	PCA

tuk

If	MAC
E1	PCA

tas

If	MAC
E2	PCA

Pero...

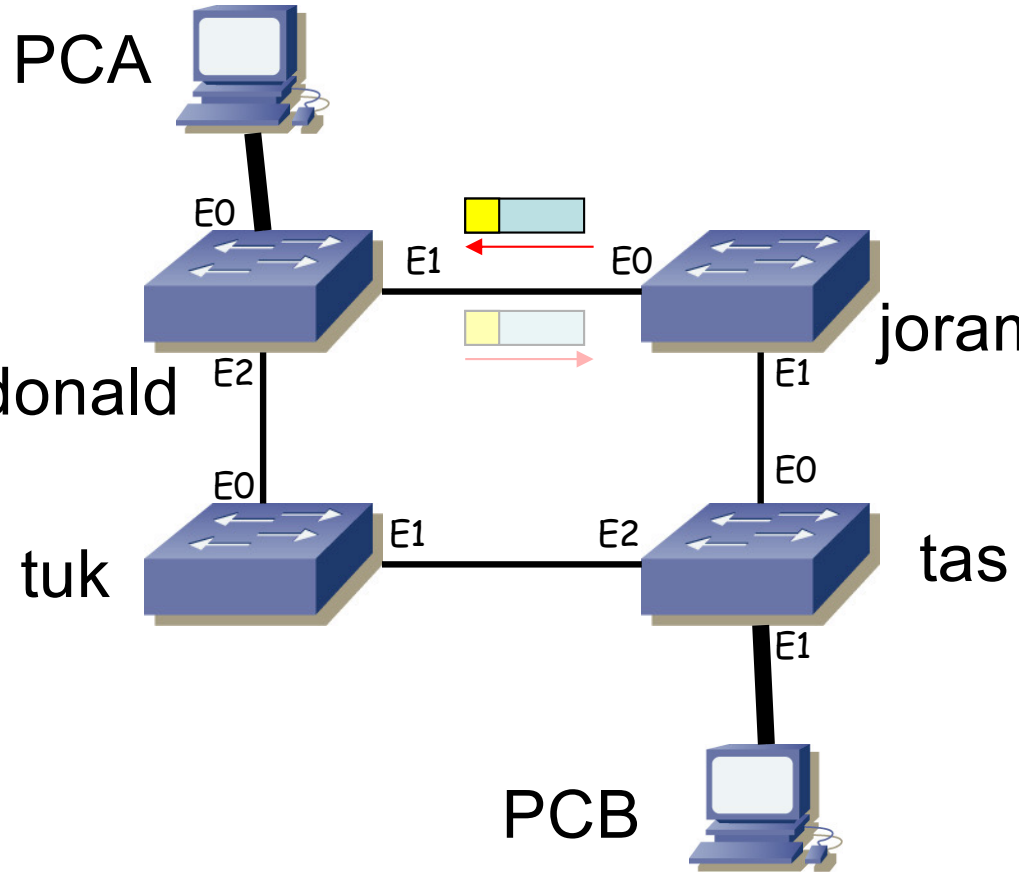
- Ejemplo:
  - Y ahora veamos con el que viene de joram
  - Vuelve a cambiar lo aprendido y reenvía

donald

If	MAC
E1	PCA

tuk

If	MAC
E1	PCA



joram

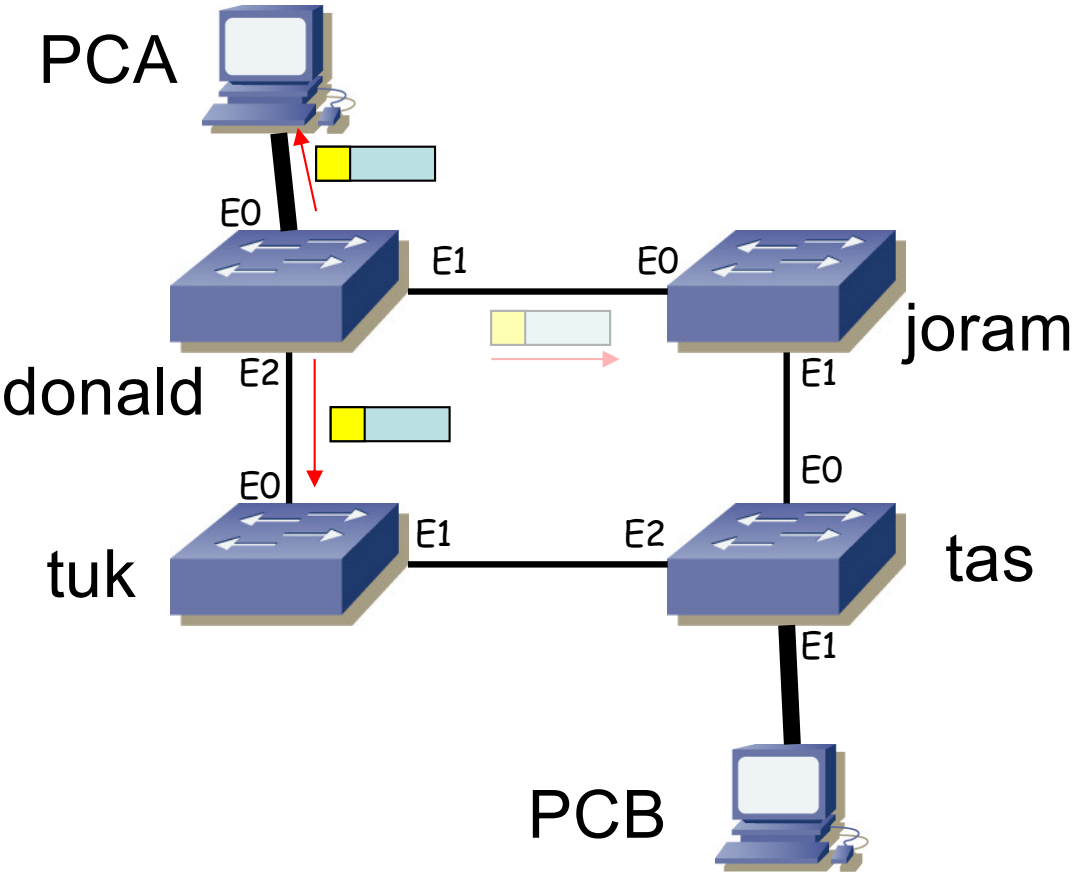
If	MAC
E1	PCA

tas

If	MAC
E2	PCA

Pero...

- Ejemplo:
  - Y ahora veamos con el que viene de joram
  - Vuelve a cambiar lo aprendido y reenvía



donald

If	MAC
E1	PCA

tuk

If	MAC
E1	PCA

joram

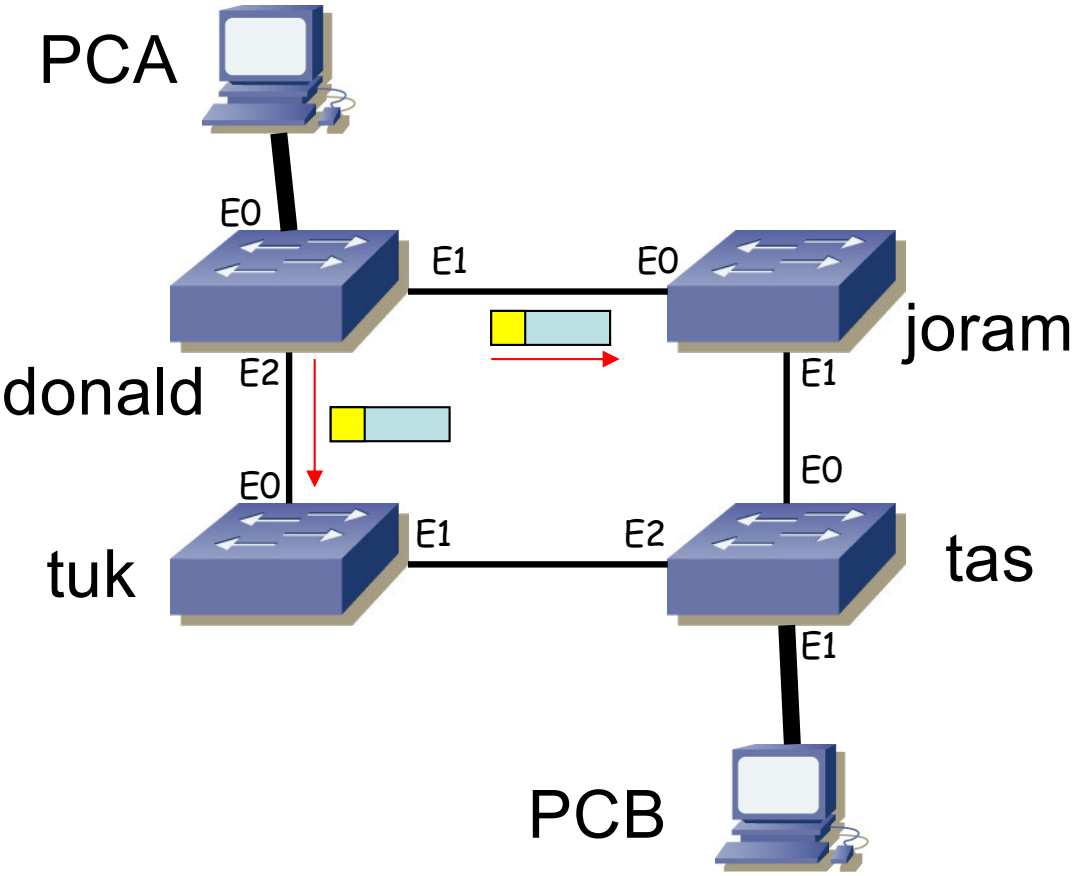
If	MAC
E1	PCA

tas

If	MAC
E2	PCA

Pero...

- Ejemplo:
  - Seguimos con los dos paquetes, uno en cada sentido del anillo



donald

If	MAC
E1	PCA

tuk

If	MAC
E1	PCA

joram

If	MAC
E1	PCA

tas

If	MAC
E2	PCA

Pero...

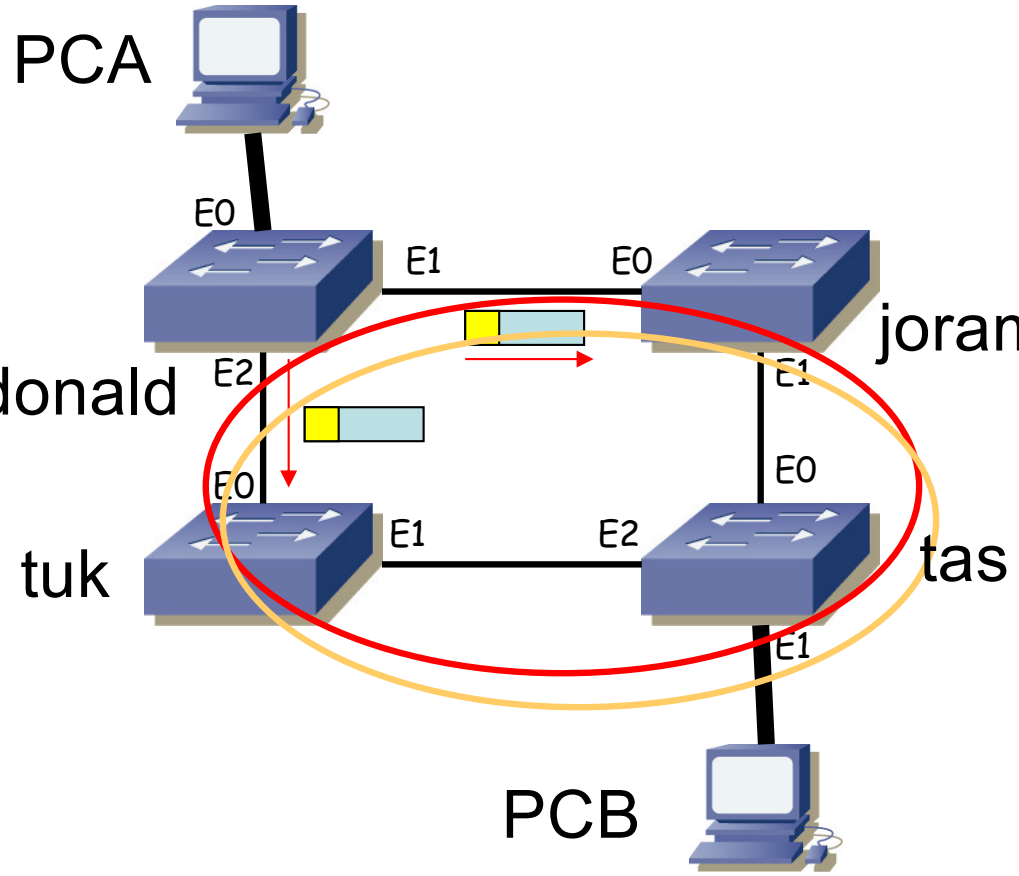
- Ejemplo:
  - Seguimos con los dos paquetes, uno en cada sentido del anillo
  - Esto no tiene fin
  - (...)

donald

If	MAC
E1	PCA

tuk

If	MAC
E1	PCA



joram

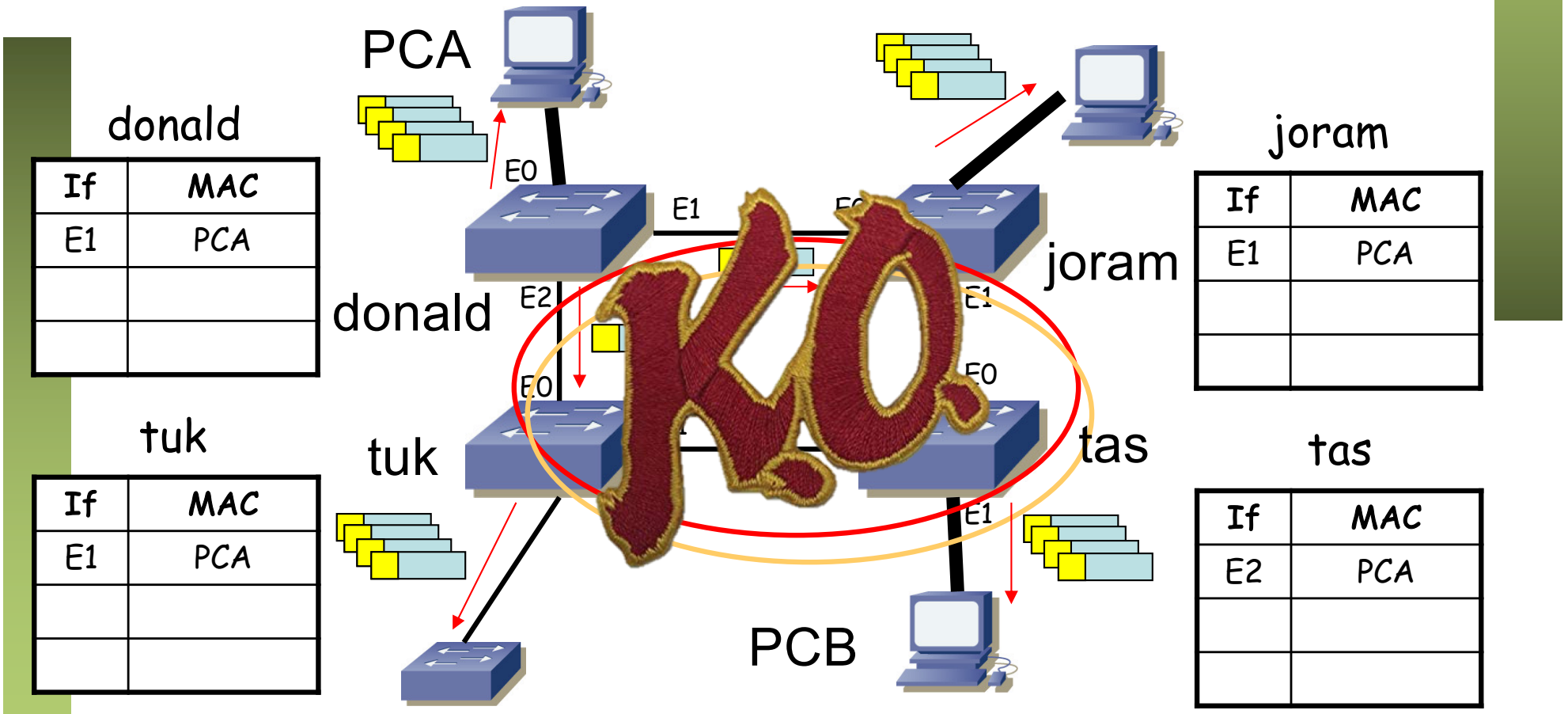
If	MAC
E1	PCA

tas

If	MAC
E2	PCA

Pero...

- Ejemplo:
  - Seguimos con los dos paquetes, uno en cada sentido del anillo
  - Esto no tiene fin
  - Además los paquetes salen por todos los puertos hacia hosts y otros switches



Soluciones

- ¡ Emplear una topología sin ciclos !
- Vale, resuelve el problema pero no conseguimos la protección
- Yo no he dicho que fuera la mejor solución
- ¿ Otras alternativas ? (¿Ideas?)

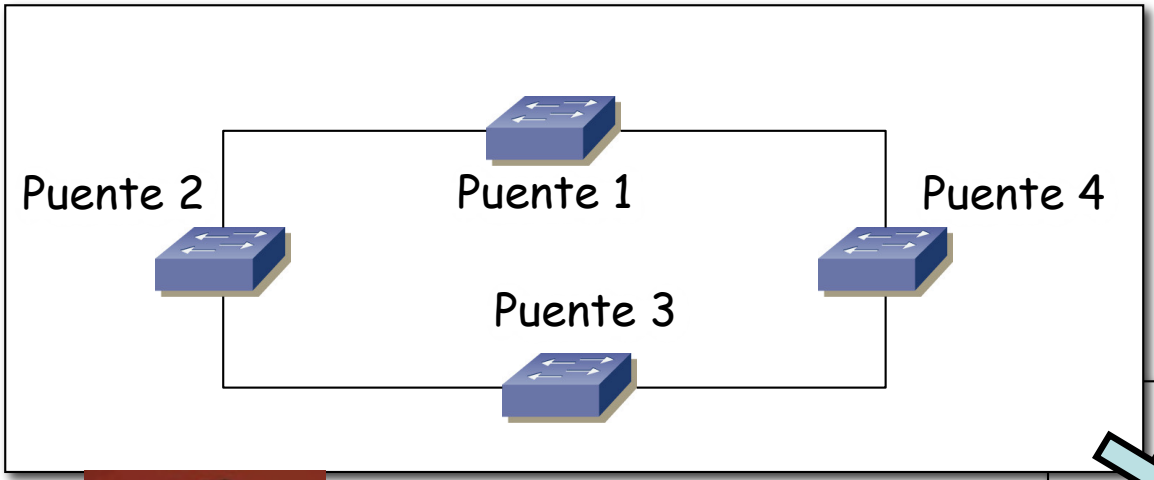
upna

Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Spanning-Tree Protocol

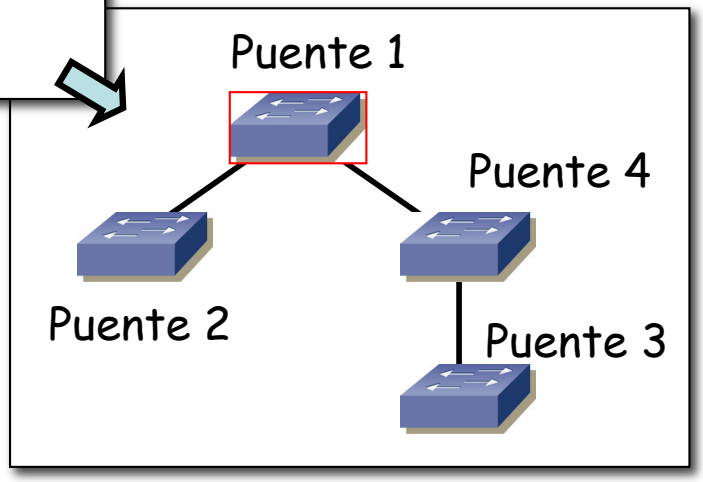
Spanning-Tree Protocol (STP)

- Calcula una topología libre de ciclos
- A partir del grafo de la topología crea un árbol
- ¿Cómo? Seleccionan un puente como “raíz” y desde él calculan un árbol
- Bloquean puertos de enlaces que no están en el árbol



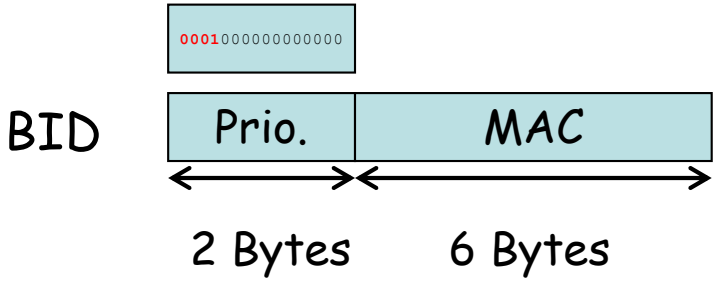
IEEE 802.1D

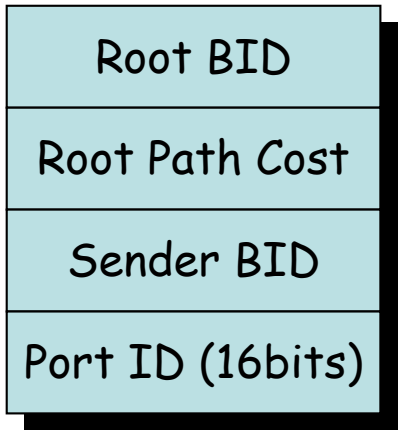
Radia Perlman



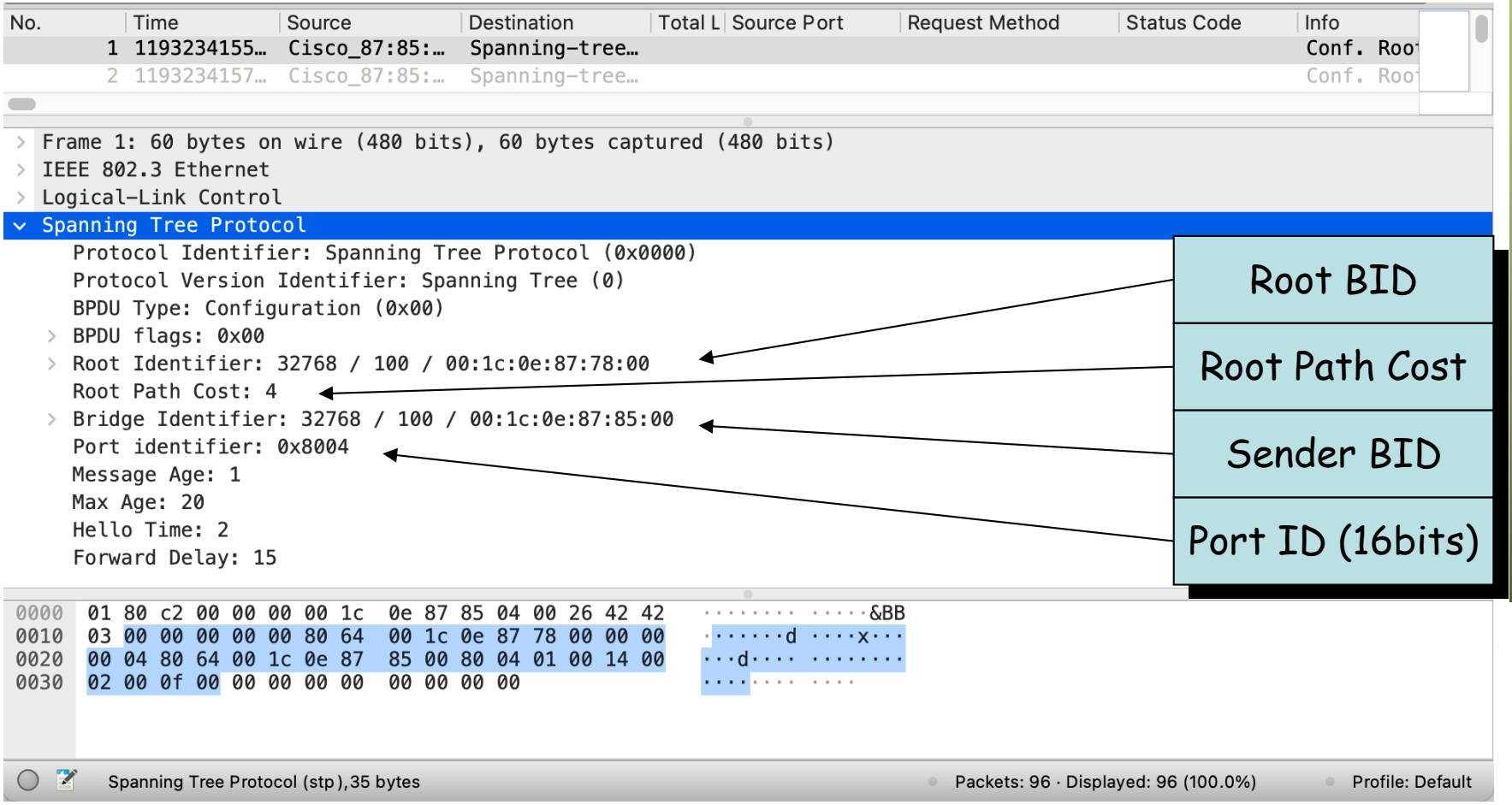
STP: BPDUs

- Bridge Protocol Data Units
- Enviadas periódicamente por los puentes, por todos sus puertos
- Destino 01:80:C2:00:00:00 (Bridge Group Address)
- No son reenviadas
- BID = Bridge ID = { Prioridad, MAC }
- Prioridad (default: 32768) en múltiplos de 4096
- En la BPDU se anuncia el coste que este switch cree tener hasta el puente raíz





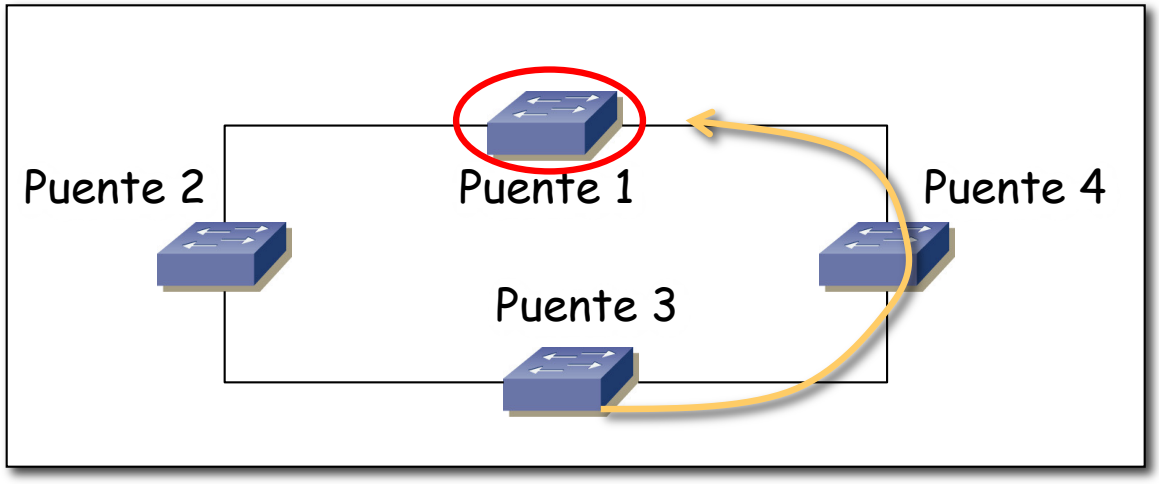
STP: BPDUs


No. Time Source Destination Total L Source Port Request Method Status Code Info
1 1193234155… Cisco_87:85:… Spanning-tree… Conf. Roo
2 1193234157… Cisco_87:85:… Spanning-tree… Conf. Roo
Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
IEEE 802.3 Ethernet
Logical-Link Control
Spanning Tree Protocol
Protocol Identifier: Spanning Tree Protocol (0x0000)
Protocol Version Identifier: Spanning Tree (0)
BPDU Type: Configuration (0x00)
BPDU flags: 0x00
Root Identifier: 32768 / 100 / 00:1c:0e:87:78:00
Root Path Cost: 4
Bridge Identifier: 32768 / 100 / 00:1c:0e:87:85:00
Port identifier: 0x8004
Message Age: 1
Max Age: 20
Hello Time: 2
Forward Delay: 15
0000 01 80 c2 00 00 00 00 1c 0e 87 85 04 00 26 42 42 &BB
0010 03 00 00 00 00 00 80 64 00 1c 0e 87 78 00 00 00 d x
0020 00 04 80 64 00 1c 0e 87 85 00 80 04 01 00 14 00 d
0030 02 00 0f 00 00 00 00 00 00 00 00 00
Spanning Tree Protocol (stp),35 bytes
Packets: 96 · Displayed: 96 (100.0%)
Profile: Default
Root BID
Root Path Cost
Sender BID
Port ID (16bits)


Dest Addr
Src Addr
Longitud
SSAP (42)
DSAP (42)
CNTL (3)
BPDU
CRC

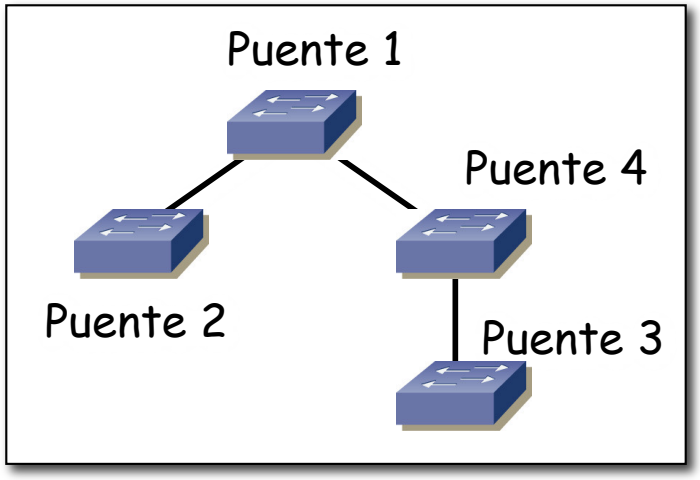
STP: BPDUs

- Se pueden “comparar” entre sí y decidir si una BPDU recibida por un puerto es “mejor” que otra
- “Mejor” en el sentido de “mejor” camino a la raíz
- Para ello cada enlace tendrá un coste y se suman
- El coste de un enlace se configura por defecto en función de su velocidad pero **debe ser configurable**



Dos detalles importantes

- Los conmutadores siguen enviando BPDUs tras calcular árbol
  - Un conmutador no tiene forma de saber que se ha alcanzado una topología estable
  - Seguir enviando y aceptando BPDUs permite recalcular el árbol ante fallos
- El plano de datos del conmutador no cambia
  - Es decir, los conmutadores siguen reenviando los paquetes de usuario en función de sus tablas
  - Y siguen aprendiendo igual
  - Simplemente algunos puertos descartan las tramas que reciben y no se reenvían tramas de usuario por ellos
  - Para decidir por dónde va una trama necesitamos saber qué puertos están bloqueados pero por lo demás sigue la misma lógica de siempre



upna

Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

STP/RSTP: Mecanismos

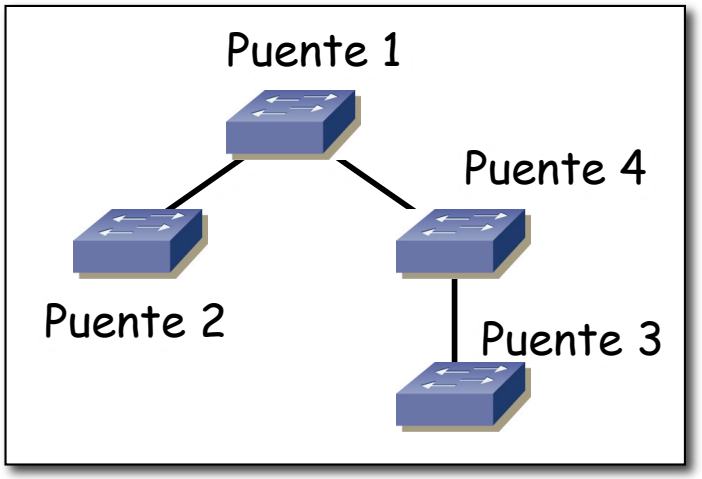
upna

Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Root bridge

Mecanismos

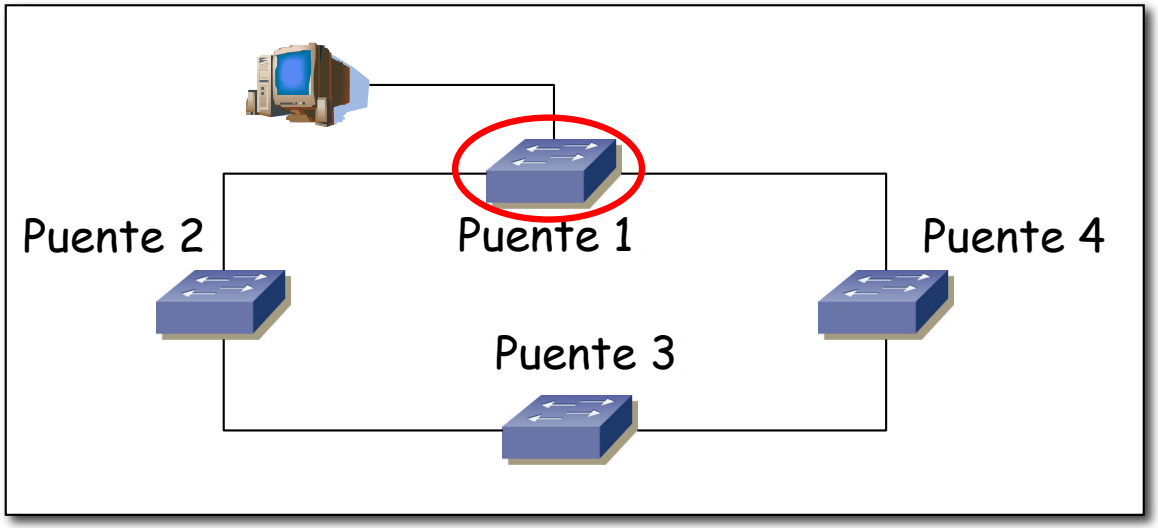
- Vamos a intentar entender cómo se calcula el árbol
- No nos interesa el “transitorio” sino solamente el estado final
- El mecanismo es básicamente *distance-vector* (Bellman-Ford distribuido)

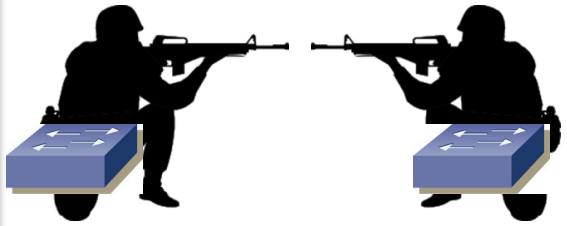


STP: Root Bridge

Selección de un *Root Bridge* (Root War !!!)

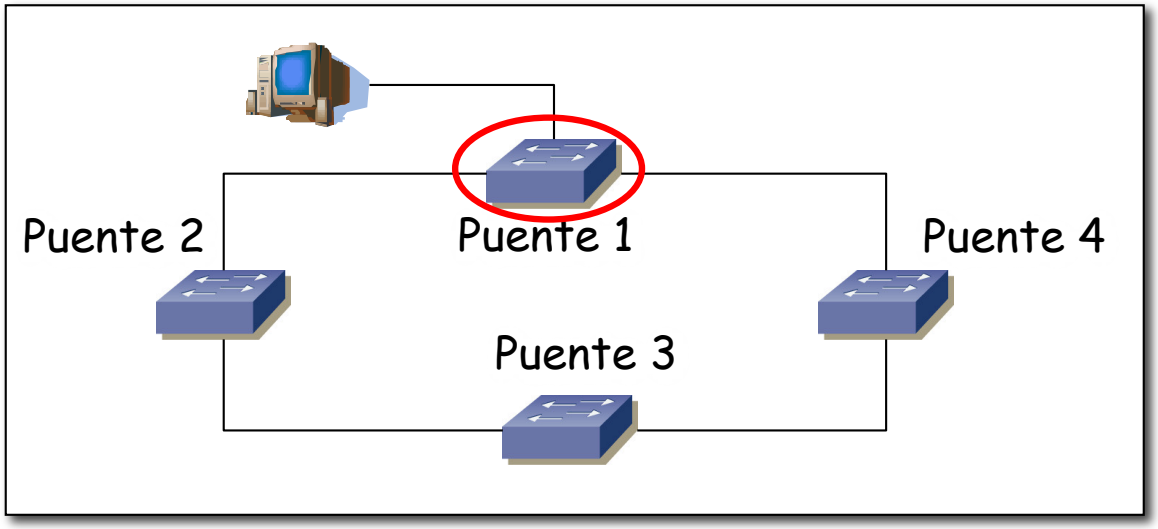
- Raíz para el árbol
- No es un primer paso sino que para cualquier BPDU que se recibe se decide si anuncia mejor root
- A partir de un valor de prioridad y una MAC del puente
  - Vienen en las BPDU
  - Puente de prioridad más baja (def. 0x8000 = 32768)
  - MAC más baja en caso de empate

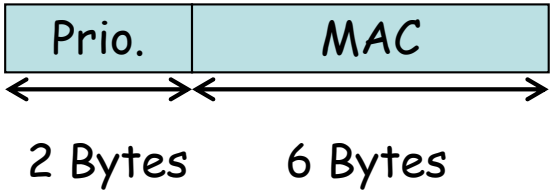




STP: Root Bridge

- Por defecto todos los puentes la misma prioridad
- Gana el de dirección MAC más baja
- Primeros 3 bytes de la MAC son el OUI
- ¡ Luego el ganador depende del fabricante !
- Cuidado pues puede ser el conmutador más lento
- Selección manual con el campo de prioridad





upna

Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Path cost

STP: Path Cost

- Asociado a cada LAN; según su velocidad **o administrativamente**
- Se va agregando en un camino creando el *Root Path Cost*
- En un switch se calcula a partir de los anuncios recibidos en cada puerto, sumando el coste del puerto por el que han llegado
- 802.1D-1998:

Table 8-5—Path Cost Parameter Values

Parameter	Link Speed	Recommended value	Recommended range	Range
Path Cost	4 Mb/s	250	100–1000	1–65 535
Path Cost	10 Mb/s	100	50–600	1–65 535
Path Cost	16 Mb/s	62	40–400	1–65 535
Path Cost	100 Mb/s	19	10–60	1–65 535
Path Cost	1 Gb/s	4	3–10	1–65 535
Path Cost	10 Gb/s	2	1–5	1–65 535

- ¿Y si tuviéramos un enlace de 2x10Gbps? ¿Coste 1?
- ¿Y si tuviéramos un enlace de 4x10Gbps?

STP: Path Cost

- Estos valores recomendados cambian en 802.1t
- 802.1D-2004 (desde 802.1t-2001):
  - Recomendado $2x10^7$/Link_Speed_en_Mbps
  - Podemos acumular al menos 20 saltos del peor coste sin exceder el máximo de un contador de 32bits

Table 17-3—Port Path Cost values

Link Speed	Recommended value	Recommended range	Range
<=100 Kb/s	200 000 000*	20 000 000–200 000 000	1–200 000 000
1 Mb/s	20 000 000[a]	2 000 000–200 000 000	1–200 000 000
10 Mb/s	2 000 000[a]	200 000–20 000 000	1–200 000 000
100 Mb/s	200 000[a]	20 000–2 000 000	1–200 000 000
1 Gb/s	20 000	2 000–200 000	1–200 000 000
10 Gb/s	2 000	200–20 000	1–200 000 000
100 Gb/s	200	20–2 000	1–200 000 000
1 Tb/s	20	2–200	1–200 000 000
10 Tb/s	2	1–20	1–200 000 000

*Bridges conformant to IEEE Std 802.1D, 1998 Edition, i.e., that support only 16-bit values for Path Cost, should use 65 535 as the Path Cost for these link speeds when used in conjunction with Bridges that support 32-bit Path Cost values.