

# Camada de Ligação (Ethernet)

**Redes e Serviços**

**Licenciatura em Engenharia Informática  
DETI-UA**



universidade de aveiro

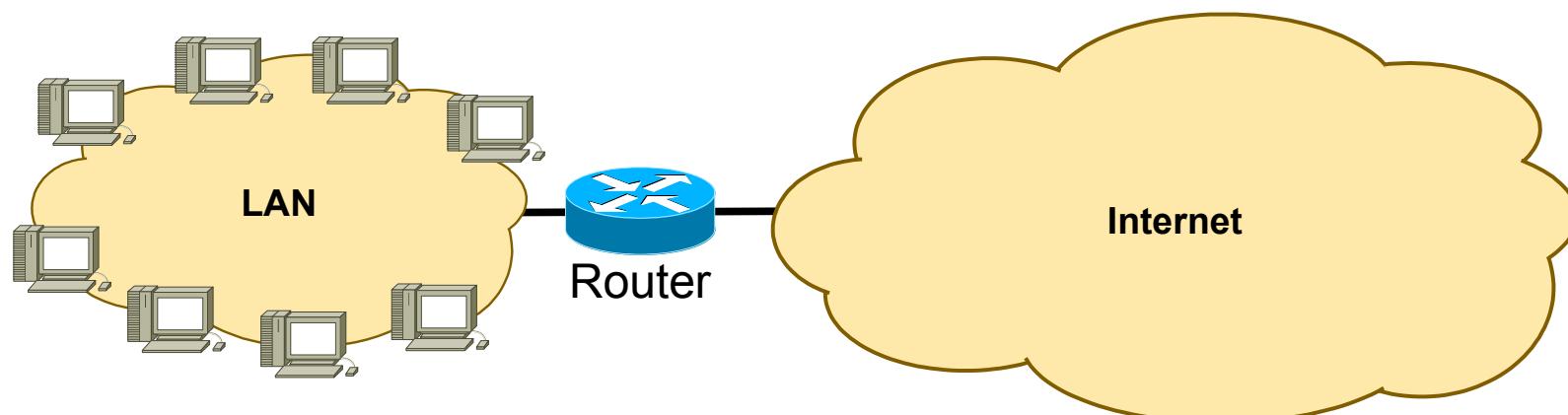
**deti.ua.pt**

# Redes Locais (LAN)

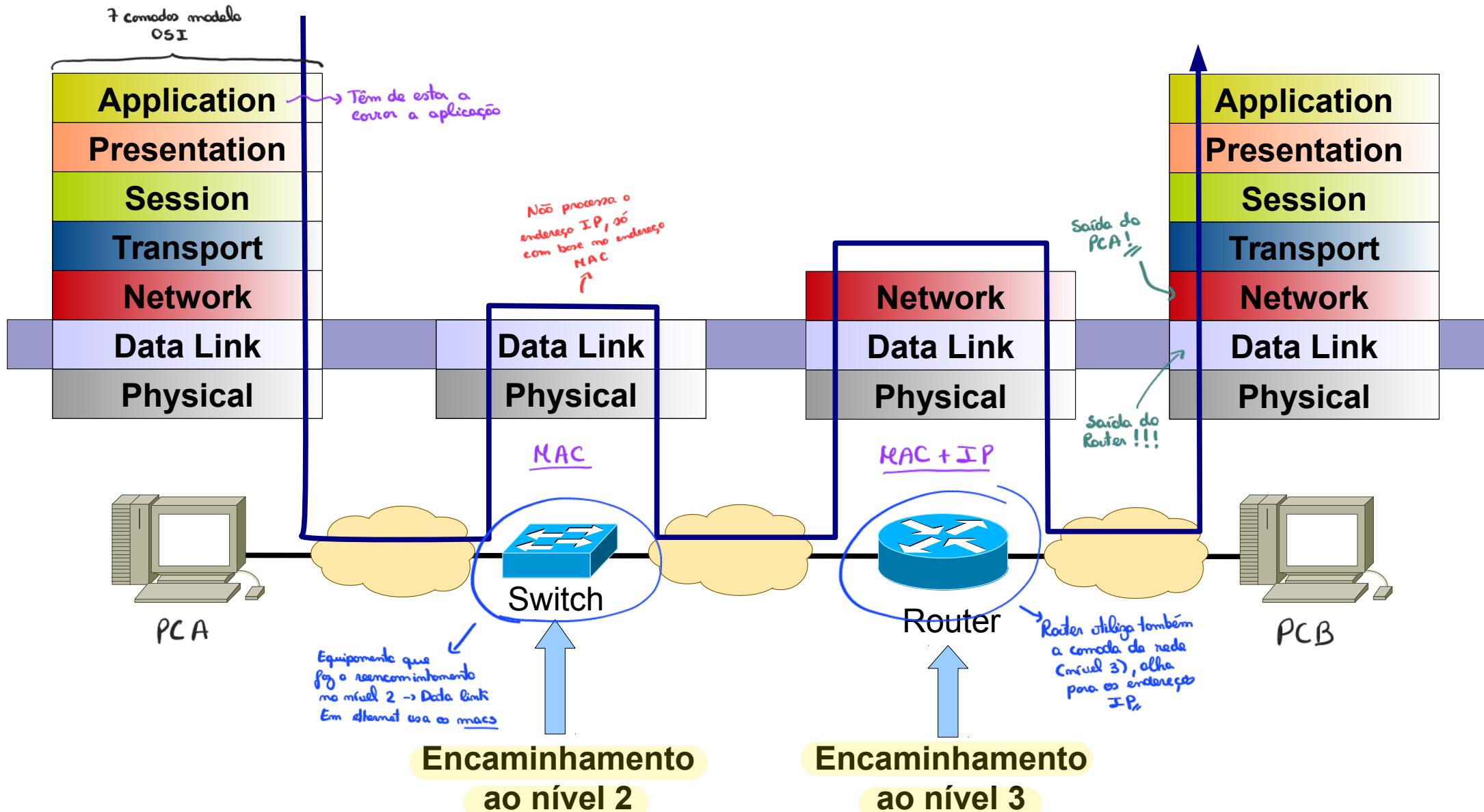
- Rede local
- Local Area Network

- Interligam estações relativamente “próximas” através de ligações partilhadas
- Tecnologias
  - ◆ Ethernet, Token Ring, 802.11, FDDI, ...
    - Simples e barata

Podem estar ligados  
por cabo ou sem  
fios → ethernet



# Camada de Ligação



# Equipamentos de Interligação (resumo)

Já está ultra passado...

- Hub/Repetidor: → Repetição de um sinal elétrico para os outros entradas

- - ◆ Opera no nível físico (camada OSI 1)
  - ◆ Regenera os sinais
  - ◆ Hub = repetidor com múltiplas portas
  - ◆ Não é usado nas redes locais actuais

- Switch/Bridge: → Comuta/ troca os pacotes entre vários portos (não precisa do IP)

- - ◆ Tipo store-and-forward → Memória e depois é reenviado para a porta específica
  - ◆ Opera ao nível da camada da ligação (camada OSI 2)
  - ◆ Interliga dois ou mais domínios de colisões
  - ◆ Comuta com base nos endereços MAC
  - ◆ Switch = bridge com múltiplas portas

De que mac o pacote veio?  
Para que mac o pacote deve ir?

- Router:

- - ◆ Tipo store-and-forward
  - ◆ Opera no nível de rede (camada OSI 3)
  - ◆ Comuta com base nos endereços de nível 3 (ex. IP, IPX, AppleTalk)

← Recebe os pacotes  
guarda-os e depois  
ve para quem vai monos

{ nível 3 → IP



# Ethernet (802.3)

~> Camada 2  
Data Link

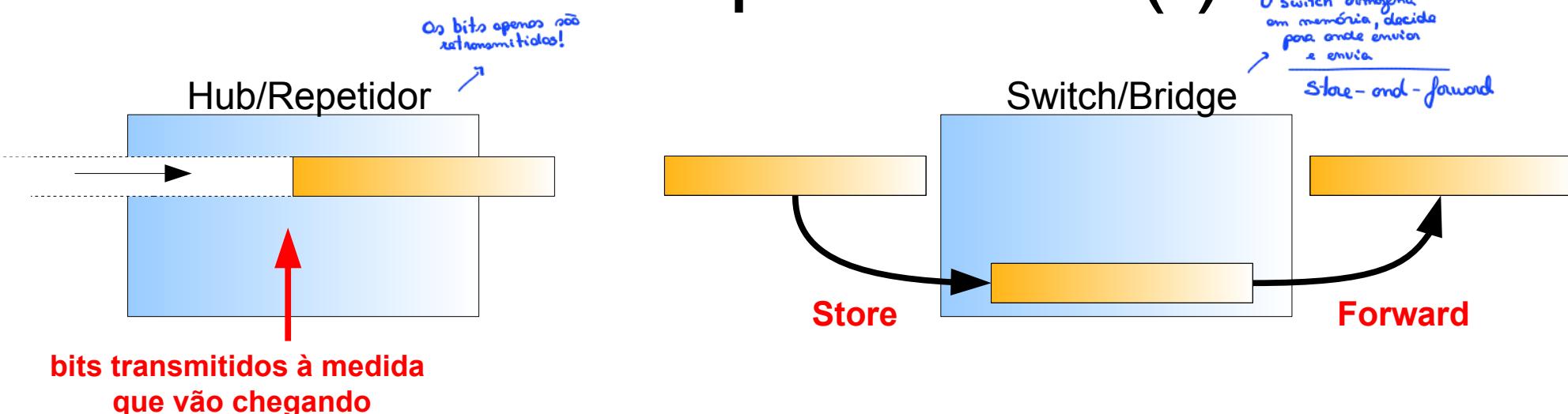
- Tecnologia LAN mais bem sucedida
- Inventada pela Xerox Palo Alto Research Center (PARC)
- A Xerox, DEC e Intel definiram em 1978 a norma para a Ethernet 10Mbps
  - Só podia haver 1 acesso de cada vez
  - Algoritmo que resolviu o problema!
- Usa “Carrier Sense/Multiple Access” with “Collision Detect” (CSMA/CD)
  - Carrier Sense: os nós conseguem perceber se o canal de comunicação está ocupado.
  - Multiple Access: múltiplos nós podem aceder simultaneamente.
  - Collision Detect: os nós “escutam” a rede enquanto transmitem, podem detetar colisões.

Garantir que ninguém manda 2 mensagens no mesmo cabo

No caso de colisão ... => As estações param de transmitir,  
esperam um tempo aleatório e  
voltam a começar



# Switches/Bridges versus Hubs/Repetidores (I)



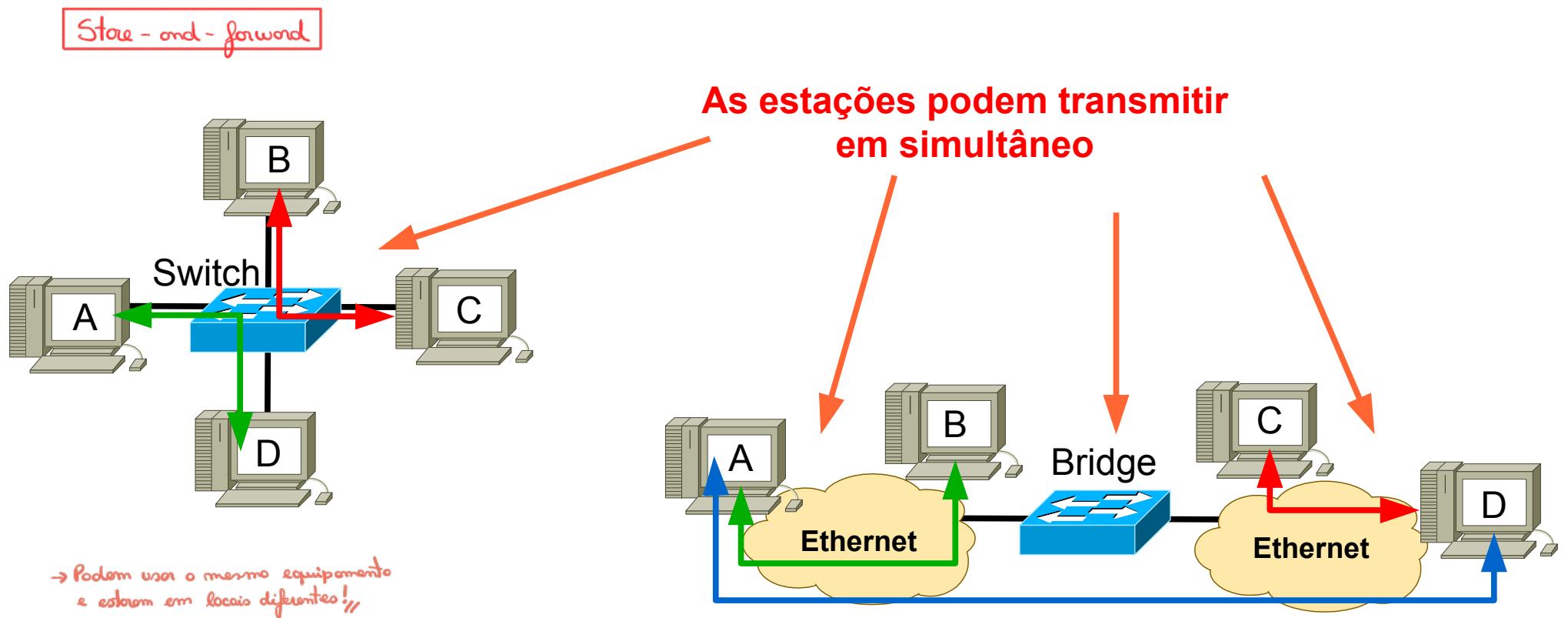
- Os repetidores/hubs interligam segmentos de LANs
- As bridges/switches interligam diferentes LANs
- Funções adicionais das bridges/switches:
  - ◆ Store & Forward + Filtragem
    - ◆ Em vez de expedir os pacotes para todas as portas pode expedir apenas para a porta da estação destino
    - ◆ As portas podem operar a diferentes taxas de transmissão

*↳ Pois o switch pode demorar, mas temos de ter alguns cuidados!*

# Switches/Bridges versus Hubs/Repetidores (II)

- Com Switches/Bridges

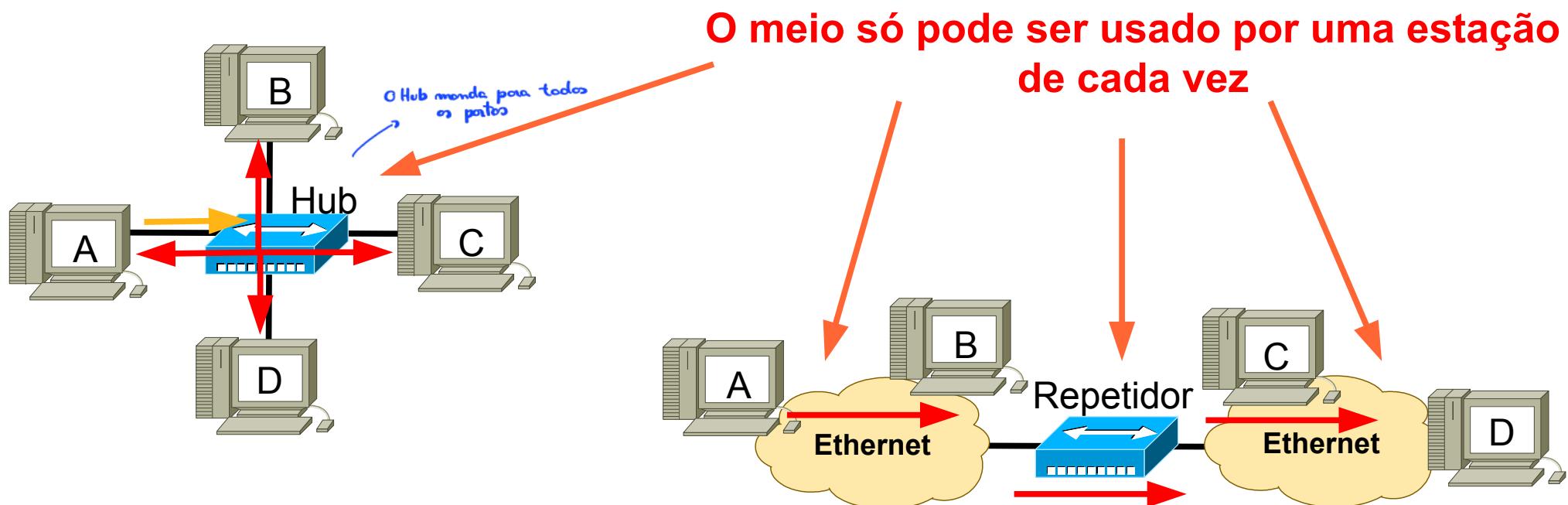
- ◆ As colisões deixam de ser um problema
- ◆ A interligação é feita ao nível da ligação (camada OSI 2)



# Repetidores/Hubs definem Domínios de Colisão

- Os hubs/repetidores interligam segmentos do mesmo tipo de LANs
  - Interligação apenas ao nível físico (camada OSI 1)
- Os hubs já (praticamente) não existem nas redes locais actuais

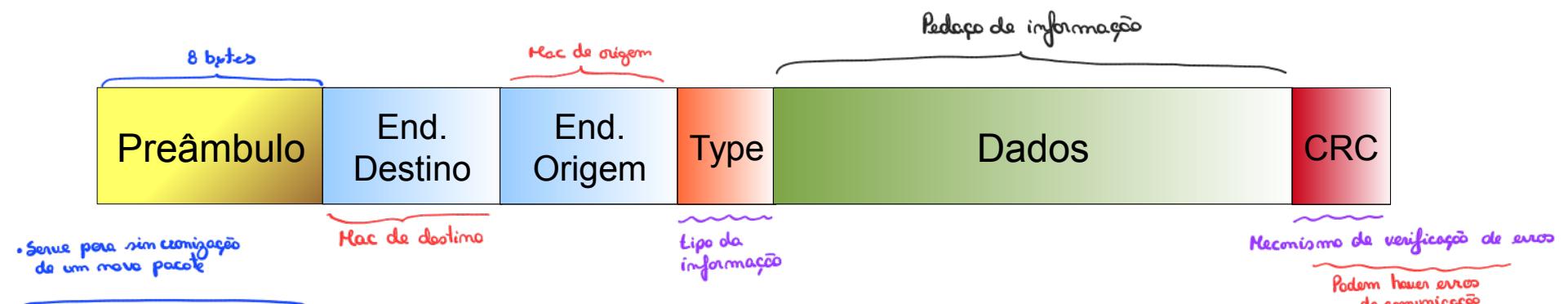
↳ Ele não sabe para que máquina enviar / Sem software!



# Trama Ethernet (1)

//Nível 2

O adaptador de rede do emissor encapsula o datagrama IP (ou outro protocolo da camada de rede) na trama Ethernet



- Preâmbulo:

- 7 bytes com o padrão 10101010 seguidos de um byte com o padrão 10101011
- É usado para sincronizar os relógios de emissão (e recepção) do emissor e do receptor.

↳ Para detectar se vai posso 1 pacote:  
ex:  
posso 10101010, 10101011  
(...)

Aqui ele tem de estar pronto para receber

# Trama Ethernet (2)

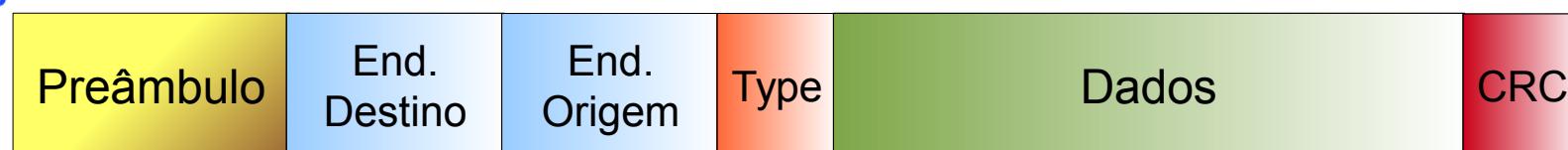
Broadcast é um pacote enviado para todos



Se for igual ao meu eu tento de o ler pois é meu !!!

- Endereços: 6 bytes
  - ◆ Se o adaptador de rede receber uma trama com um endereço de destino igual ao seu, ou se for o endereço de broadcast, passa os dados da trama ao protocolo da camada de rede
  - ◆ Senão, o adaptador descarta a trama
- Type: Indica qual o protocolo da camada de rede (tipicamente IP, mas pode ser IPX, AppleTalk, etc...)
- CRC: Usado para detecção de erros de transmissão
  - ◆ Se for detectado um erro no receptor a trama é descartada

D) código matemático que detecta o erro da transmissão



[64 bytes, 1518 bytes]



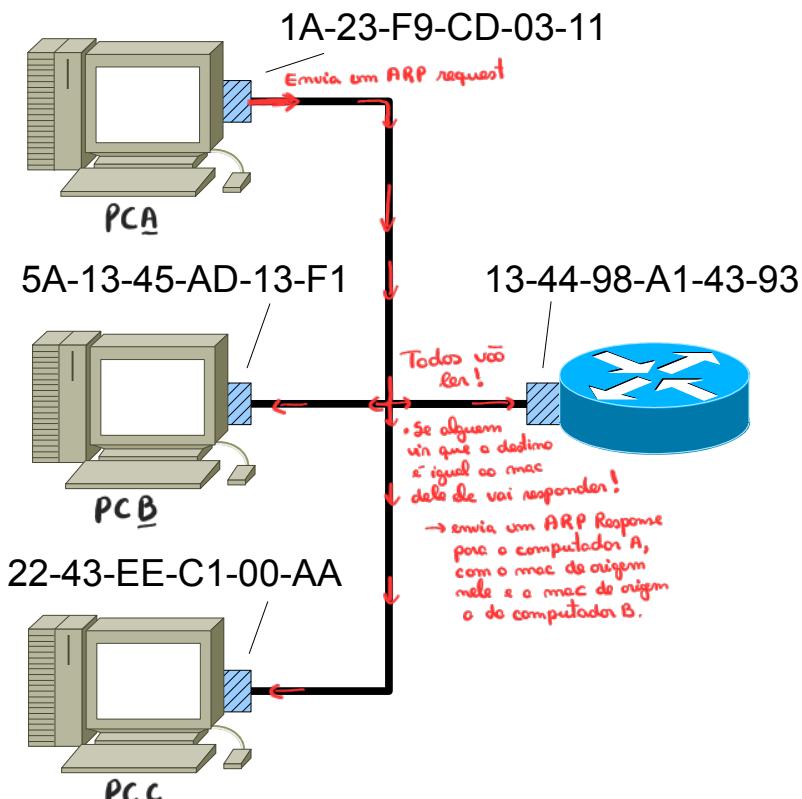
# Endereços MAC

## • Endereço MAC (LAN ou físico ou Ethernet):

- Função: permitir a transmissão de tramas entre interfaces ligados fisicamente (que estão na mesma rede)
- Tem **48 bits** (para a maioria das LANs)
  - Embutido de origem na ROM da placa de rede
  - Algumas placas permitem a sua alteração por software
- Notação hexadecimal
- Cada placa/adaptador de rede tem um endereço MAC ☺
- Endereço de broadcast: FF-FF-FF-FF-FF-FF

Como é que descobrir o mac do outro?

→ Protocolo ARP  
Pedido a nível ip!  
→ de sobre o ip  
mas não sabe o MAC } Quem tem o ip xxx?  
(mundo em broadcast)  
FF-FF-...



Todo 11111111-11111111-...  
Significa que quer enviar um pacote para todos lerem! „



# Operações básicas de um switch

- Os switches possuem uma tabela de encaminhamento (camada 2)

Ele guarda sempre porta X  
( pelo mac (..))  
Se o destino tiver na tabela  
monda apenas  
para a porta  
da tabela  
→ Se não tiver ele procura  
para todos os portos  
???

Store-and-forward

Não quer saber do IP.

- Quando um switch recebe uma trama MAC numa porta:

MAC	Porta
00:11:11:11:11:11	1
00:22:22:22:22:22	1
A1:33:33:33:33:33	2
44:44:44:44:44:44	3
55:55:55:00:00:55	3

- Regista na sua tabela de encaminhamento a porta em que recebeu a trama e o endereço MAC de origem da trama
- Procura o endereço MAC de destino da trama na tabela de encaminhamento para reencaminhar a trama

- Mecanismo de forwarding:

Encomendar ...

- Quando o endereço MAC destino da trama existe na tabela de encaminhamento, o switch envia a trama apenas pela porta registada na sua tabela

- Mecanismo de flooding:



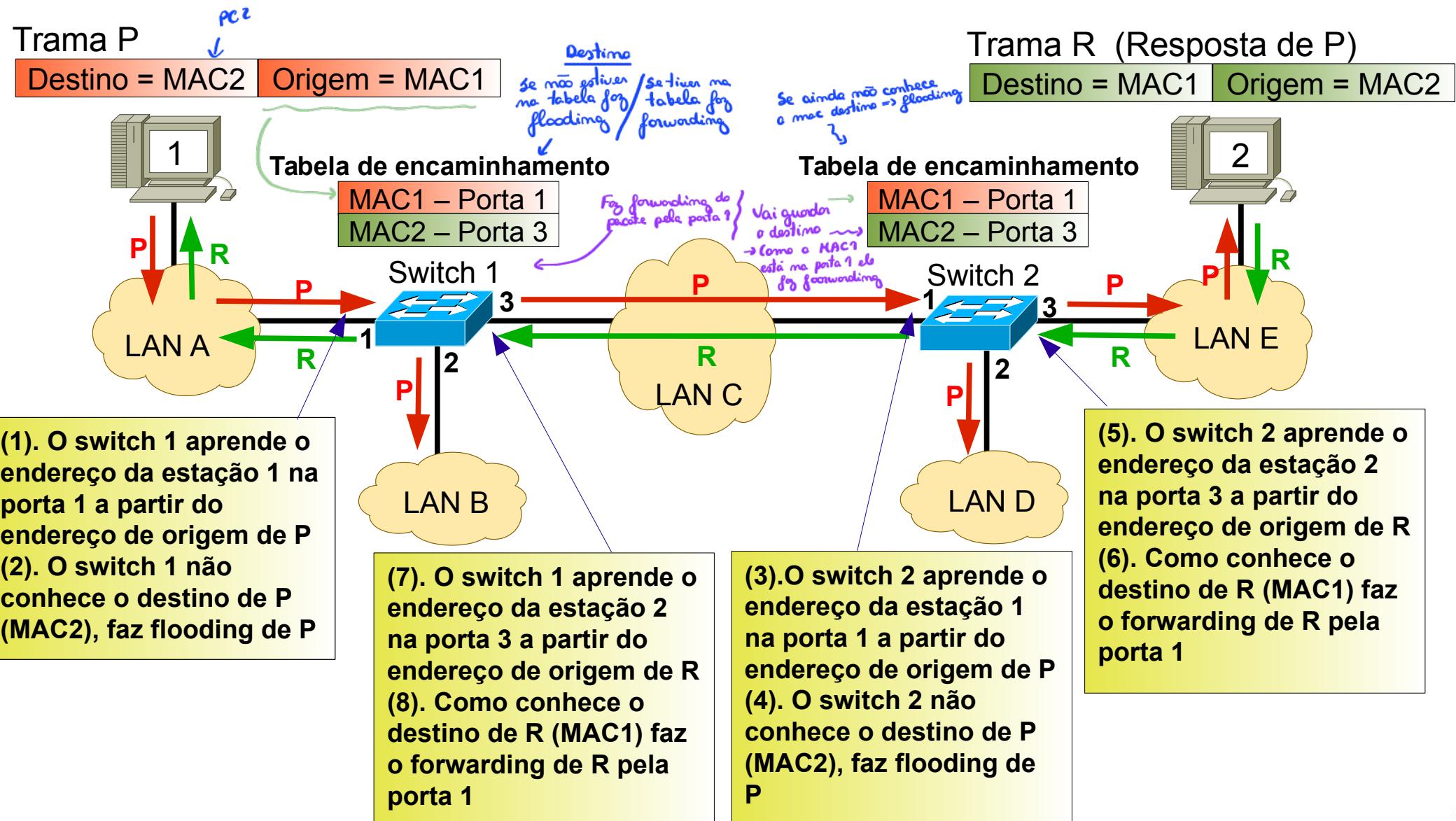
broadcast: quero na origem enviar para toda a rede!

- Quando o endereço MAC destino da trama não existe na tabela de encaminhamento, o switch envia a trama por todas as portas exceto pela porta em que recebeu a trama

Switch escolhe a porta para enviar o pacote!



# Aprendizagem de endereços em switches

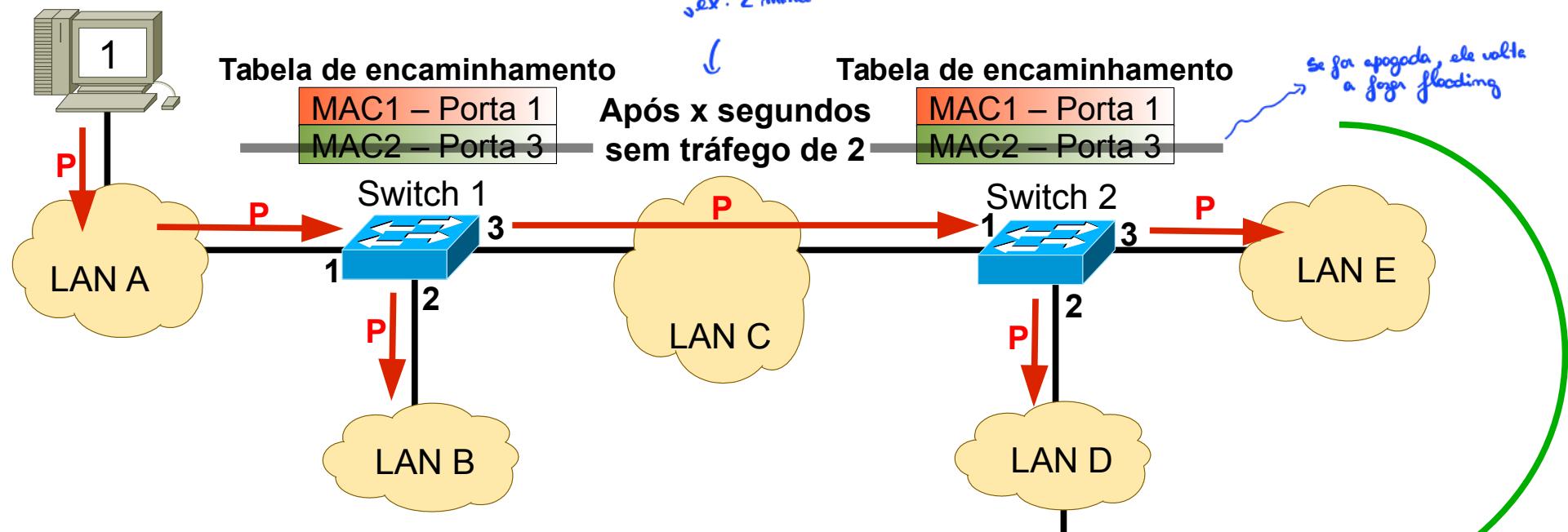


# Tempo de vida das entradas das tabelas de encaminhamento

Passado um tempo  
elas vão sendo  
apagadas

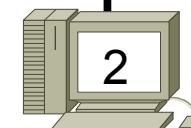
Trama P

Destino = MAC2 | Origem = MAC1



Quando a estação 2 muda de LAN (E para D)  
não recebe pacotes até:

- Transmitir dados para a rede
- Expirar a sua entrada nas tabelas de encaminhamento do switches

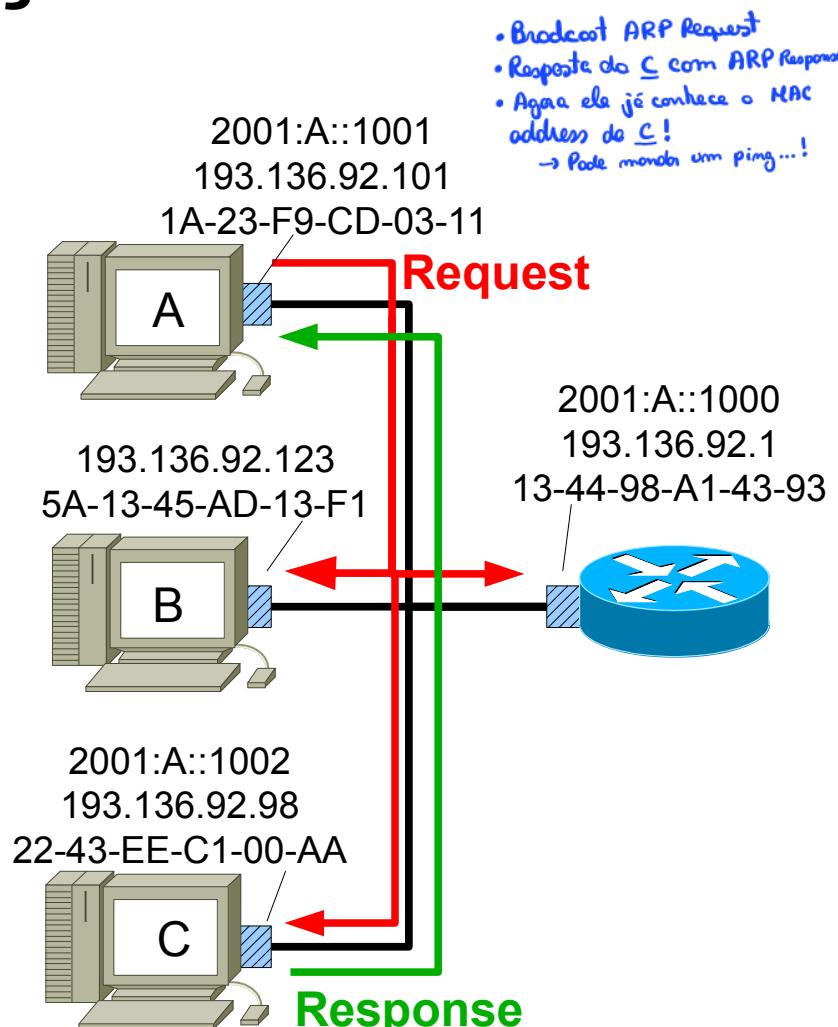


- Tempo grande:
  - ⊕ → não precisa de sótar sempre a fazer flooding...
  - ⊖ → mas se mudarmos um pc de rede o pacote não vai lá chegar, pois está a fazer forwarding...



# Resolução de Endereços Físicos

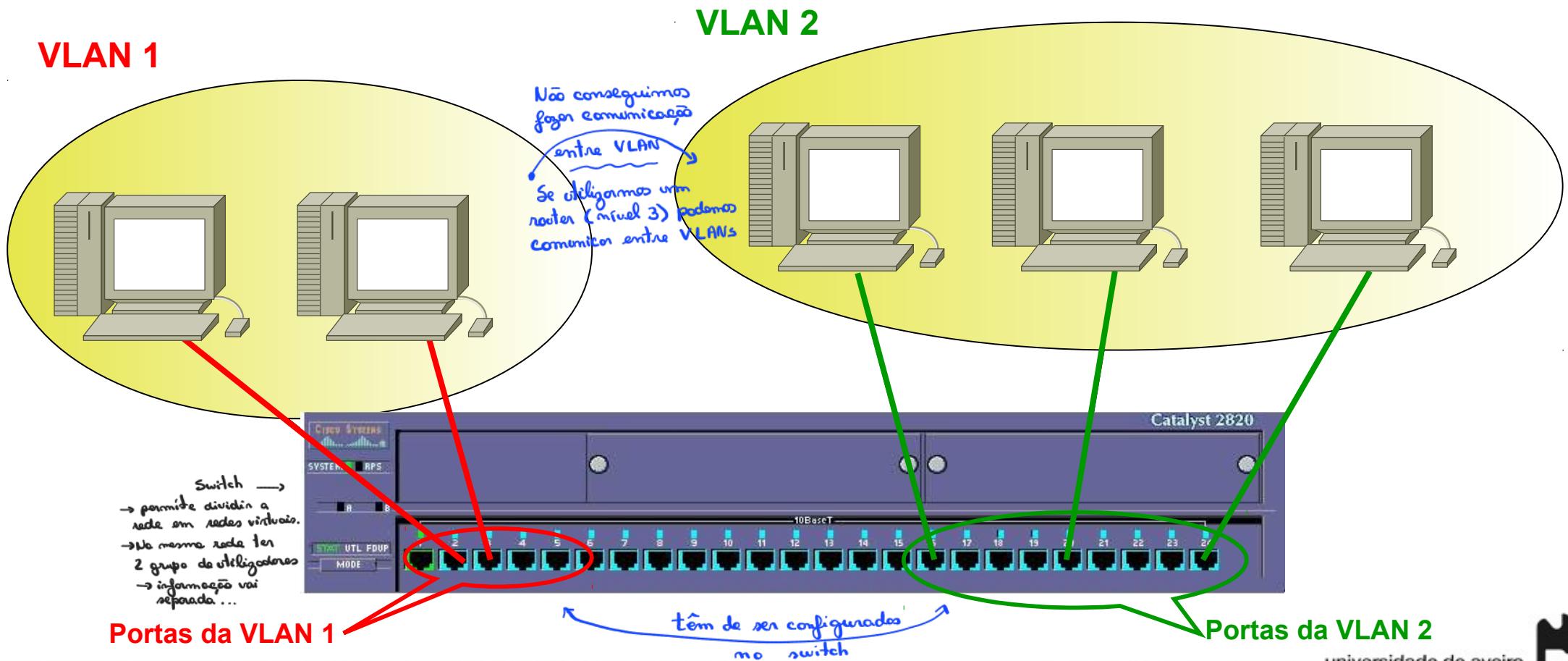
- Quando “A” precisa de contactar “C” por IP
  - “A” precisa do endereço MAC de “C”
    - Só sabe o endereço IPv4 ou IPv6
  - Se o endereço de “C” não estiver na tabela de endereços físicos conhecidos
    - IPv4: Tabela ARP → *Address Resolution Protocol*
    - IPv6: Tabela de vizinhos (NDP)
  - “A” envia um pedido para a rede com o endereço IP de “C” e a pedir o endereço MAC de C
    - IPv4: pacote ARP, enviado em broadcast (para todos)
    - IPv6: pacote ICMPv6, enviado em multicast (para um grupo restrito)
  - “C” verifica que o seu IP vem no pedido e responde diretamente para “A” (MAC destino=MAC de A) com a indicação do seu MAC.
    - “A” atualiza a sua tabela ARP/NDP
- A resolução de endereços físicos apenas existe dentro da mesma rede local (mesmo domínio de broadcast)



• Cada VLAN é uma rede isolada no nível do switch

# LANs virtuais (VLAN)

- Uma LAN virtual (VLAN) é um grupo de máquinas/utilizadores com um conjunto de requisitos/características comuns que comunicam no mesmo domínio de broadcast.
  - ◆ Independente da sua localização física (switch a que estão ligados).
- Máquinas em VLAN diferentes não comunicam ao nível da camada 2.
  - ◆ Para comunicar têm de o fazer ao nível da camada 3.



# Exemplo – VLANs

Ping enviado por 10.0.0.1

A → B ✓

```
# ping 10.0.0.2
```

Pinging 10.0.0.2 with 32 bytes of data:

```
Reply from 10.0.0.2: bytes=32 time<10ms TTL=128
```

Ping statistics for 10.0.0.2:

```
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
  Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

# ping 10.0.0.5 A → C X

Pinging 10.0.0.5 with 32 bytes of data:

Request timed out.  
Request timed out.  
Request timed out.  
Request timed out.

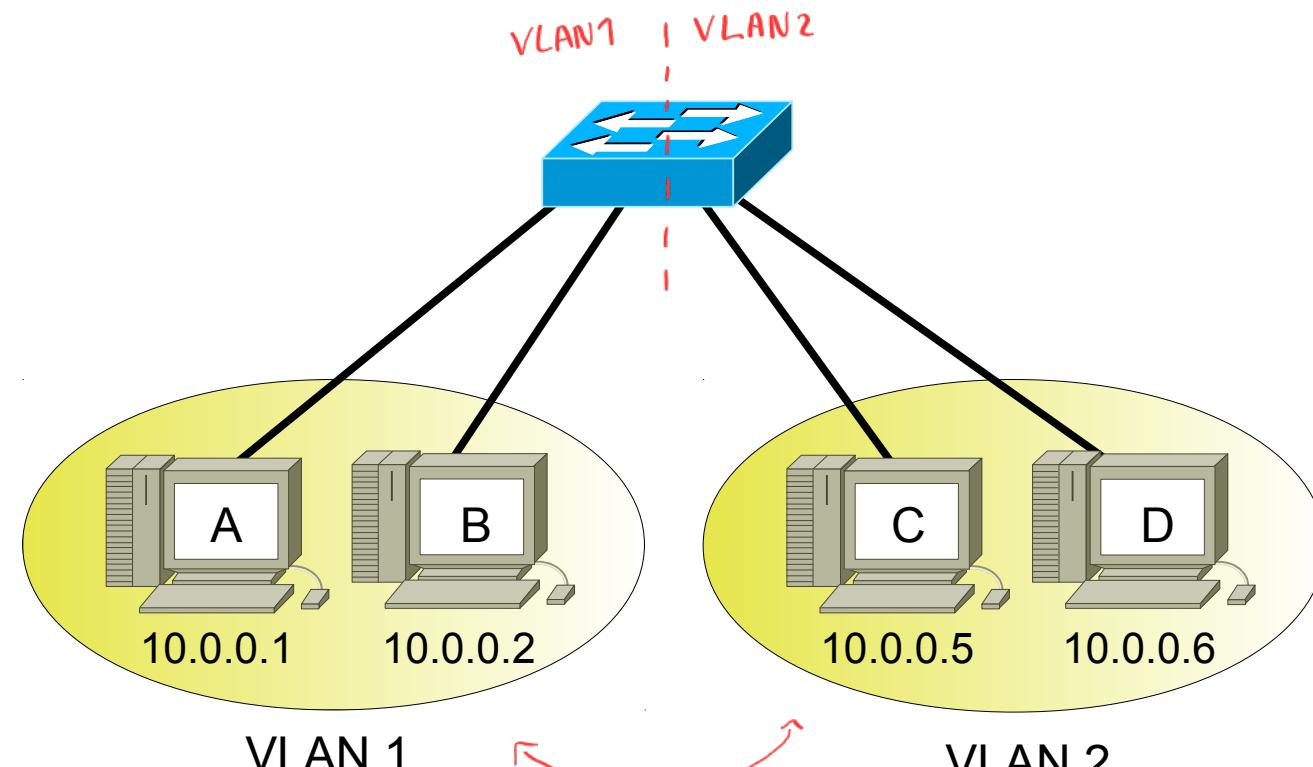
Ping statistics for 10.0.0.5:

```
  Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
  Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

# ping 10.0.0.6 A → D X

Pinging 10.0.0.6 with 32 bytes of data:

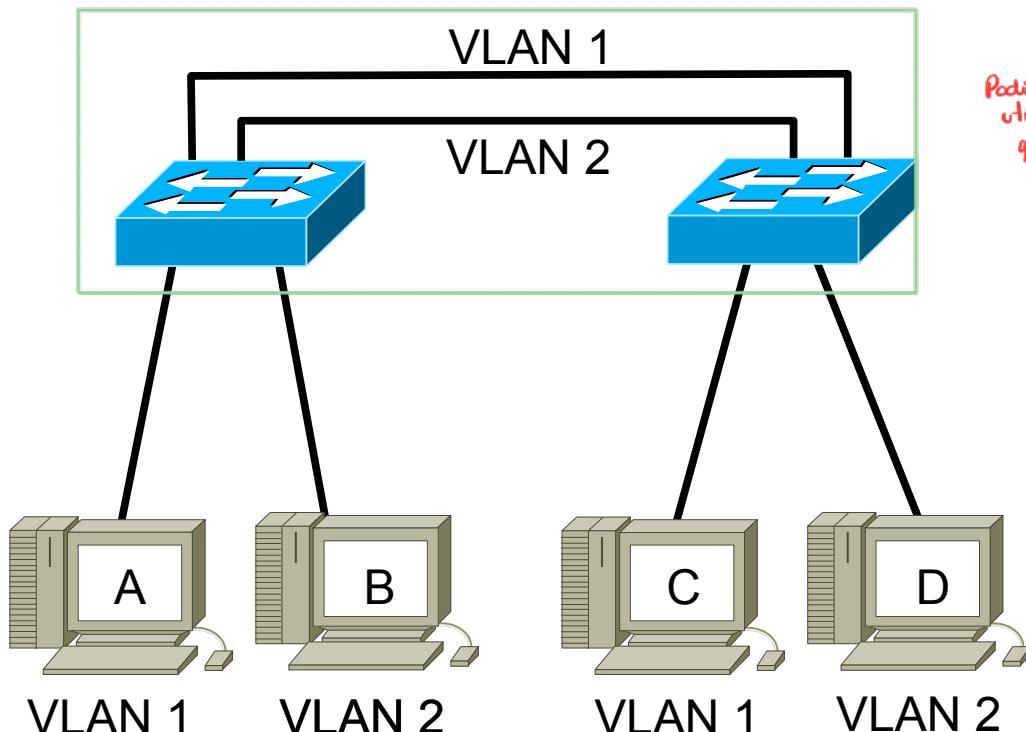
Request timed out.  
Request timed out.  
Request timed out.  
Request timed out.



# Interligação de switches

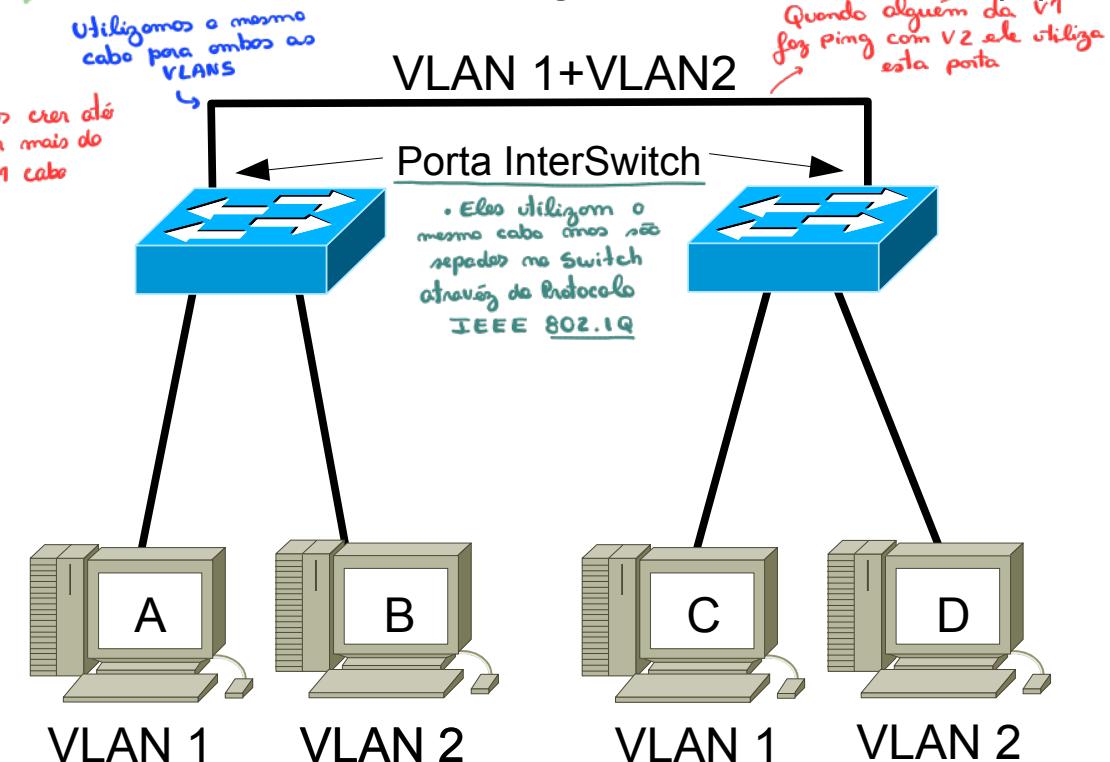
- Usando uma ligação física por VLAN

Para podermos distinguir entre VLANs assim, MAS...



- Usando porta(s) InterSwitch

- Também designadas por **trunk(s)**



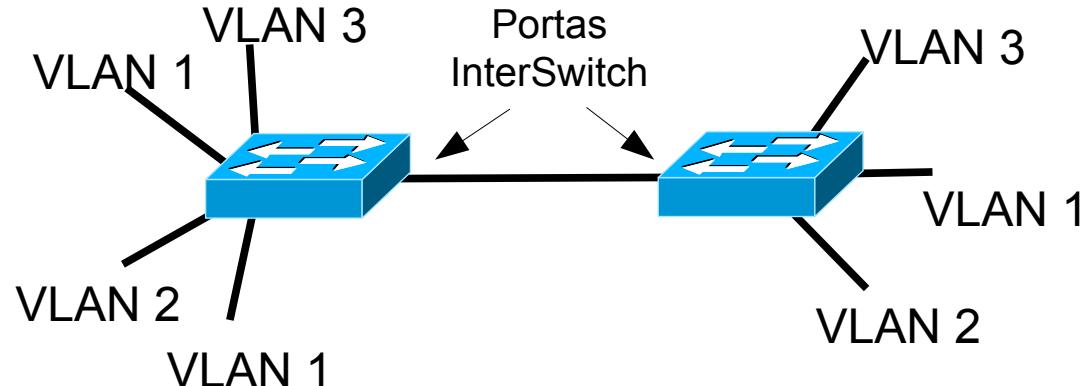
**É necessária uma forma de distinguir os pacotes de diferentes VLANs**

Protocolo 802.1Q



# Norma IEEE802.1Q

*É utilizado entre switch!*



Pacote Ethernet sem etiqueta VLAN

6	6	2		
destination	source	type	data	

+4 Bytes } • Para definir:  
                  → prioridade  
                  → ...  
                  → VLAN id (Identificador da VLAN)  
Assim o Switch consegue reencaminhar para a VLAN certa!

Pacote Ethernet com etiqueta VLAN

6	6	4	2						
destination	source	TAG	type	data					
				16bits 8100h	3bits priority	1bit CFI	12bits VLAN ID		

*Diferença os VLAN's*

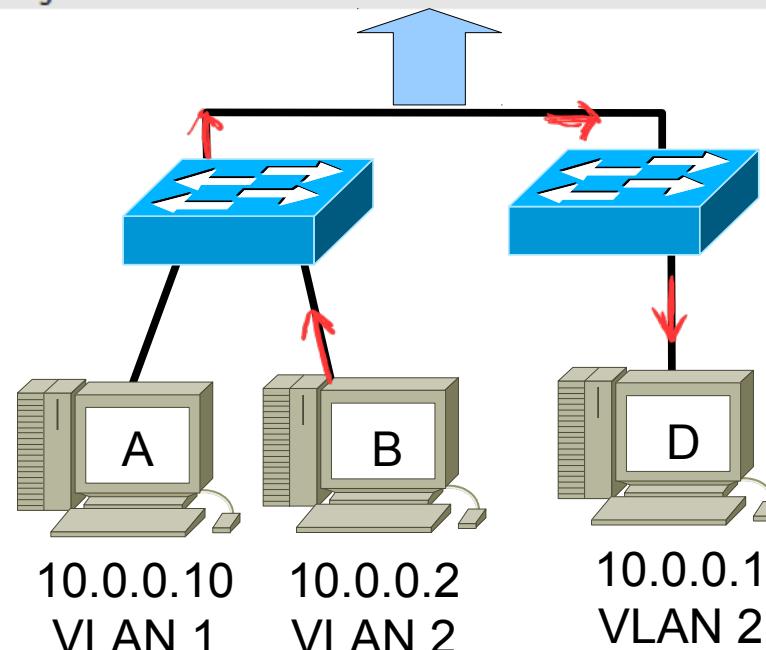
- Priority: Prioridade do tráfego de acordo com a norma 802.1q (valores 0 a 7)
- CFI: usado para compatibilidade com tecnologias mais antigas (sempre zero na Ethernet)
- VLAN ID: identificados da VLAN



# Exemplo – portas InterSwitch/Trunk

Filter: icmp		Expression...		Clear	Apply
No..	Time	Source	Destination	Protocol	Info
23	11.535990	10.0.0.2	10.0.0.1	ICMP	Echo (ping) request
24	11.536995	10.0.0.1	10.0.0.2	ICMP	Echo (ping) reply
27	12.538443	10.0.0.2	10.0.0.1	ICMP	Echo (ping) request
28	12.539186	10.0.0.1	10.0.0.2	ICMP	Echo (ping) reply

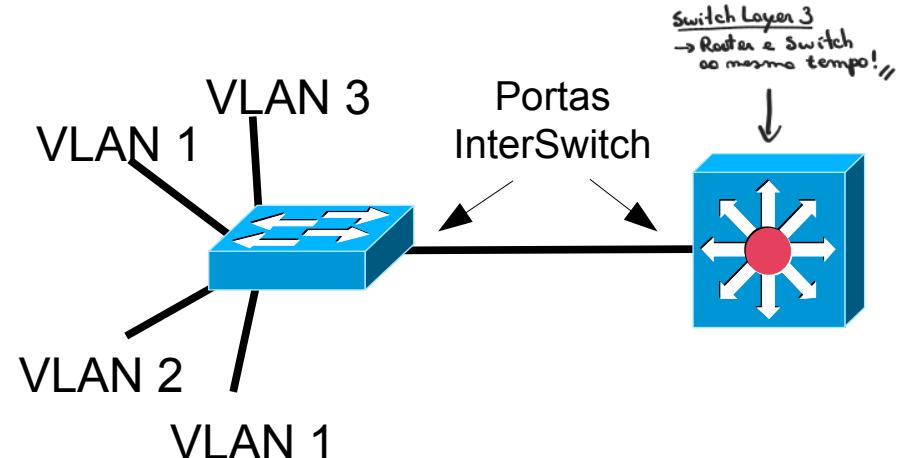
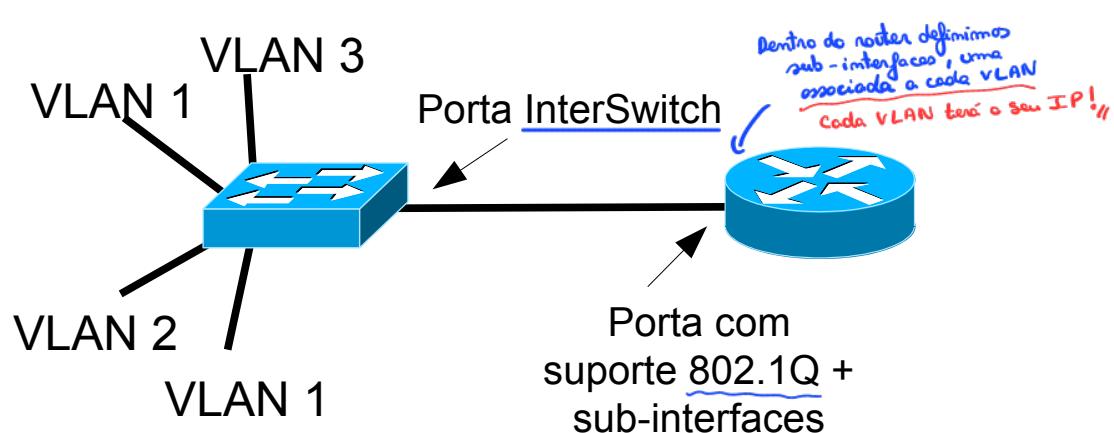
Frame 23 (102 bytes on wire, 102 bytes captured)  
Ethernet II, Src: 00:aa:00:53:7c:00 (00:aa:00:53:7c:00), Dst: 00:aa:00:fa:67:00 (00:aa:00:fa:67:00)  
802.1Q Virtual LAN, PRI: 0, CFI: 0, ID: 2  
000. .... .... .... = Priority: 0  
... 0 .... .... .... = CFI: 0  
.... 0000 0000 0010 = ID: 2  
Type: IP (0x0800) *Is Vazio da VLAN 2*  
Internet Protocol, Src: 10.0.0.2 (10.0.0.2), Dst: 10.0.0.1 (10.0.0.1)  
Internet Control Message Protocol



De vez em quando temos  
de comunicar entre  
VLANs

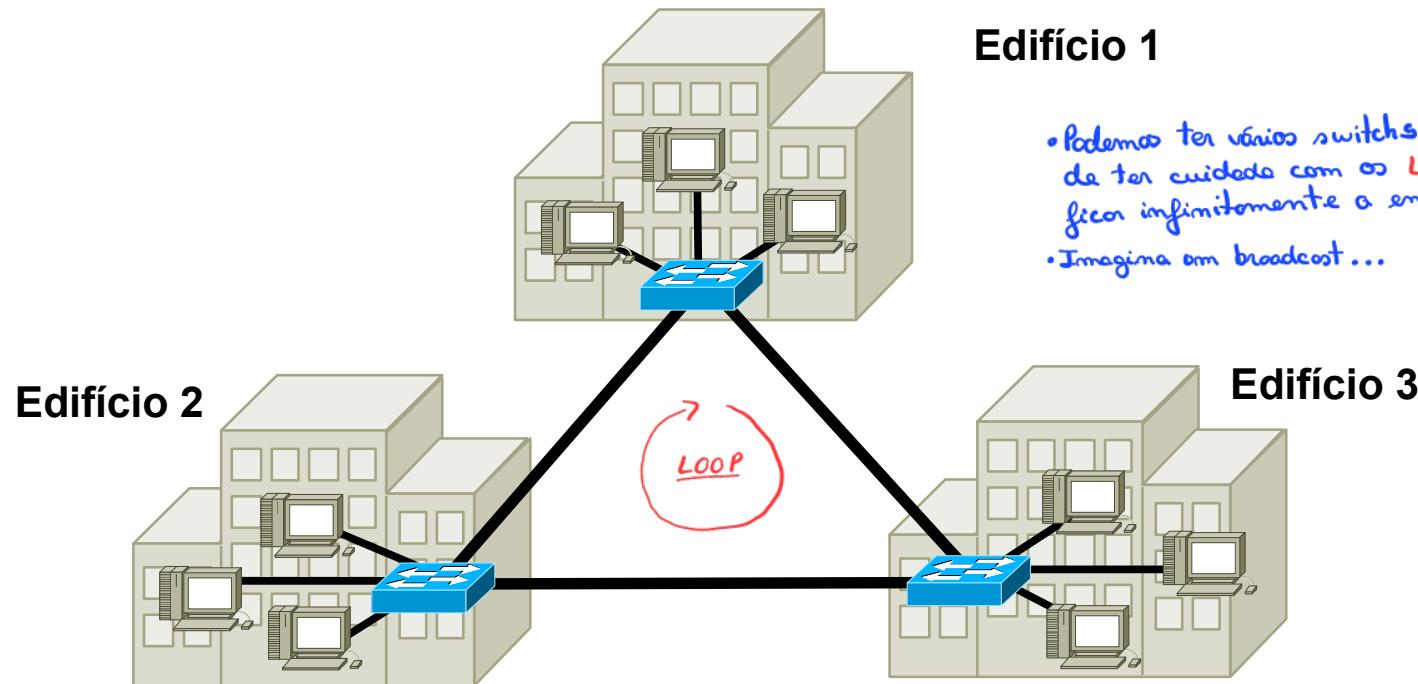
# Encaminhamento entre VLANs

- Para comunicar entre VLAN diferentes é necessário **um router**.
  - ◆ É equivalente à comunicação entre duas LAN distintas.
- As soluções mais comuns são:
  - ◆ Usar um router com suporte do protocolo **802.1Q**,
    - ◆ Ligar uma porta de inter-switch ao interface físico de um router,
      - O interface físico é sub-dividido em sub-interfaces, um para cada VLAN.
  - ◆ Usar um switch L3
    - ◆ Faz encaminhamento entre as diferentes VLAN.



# Interligação de Switches com redundância

↑ Princípios básicos...



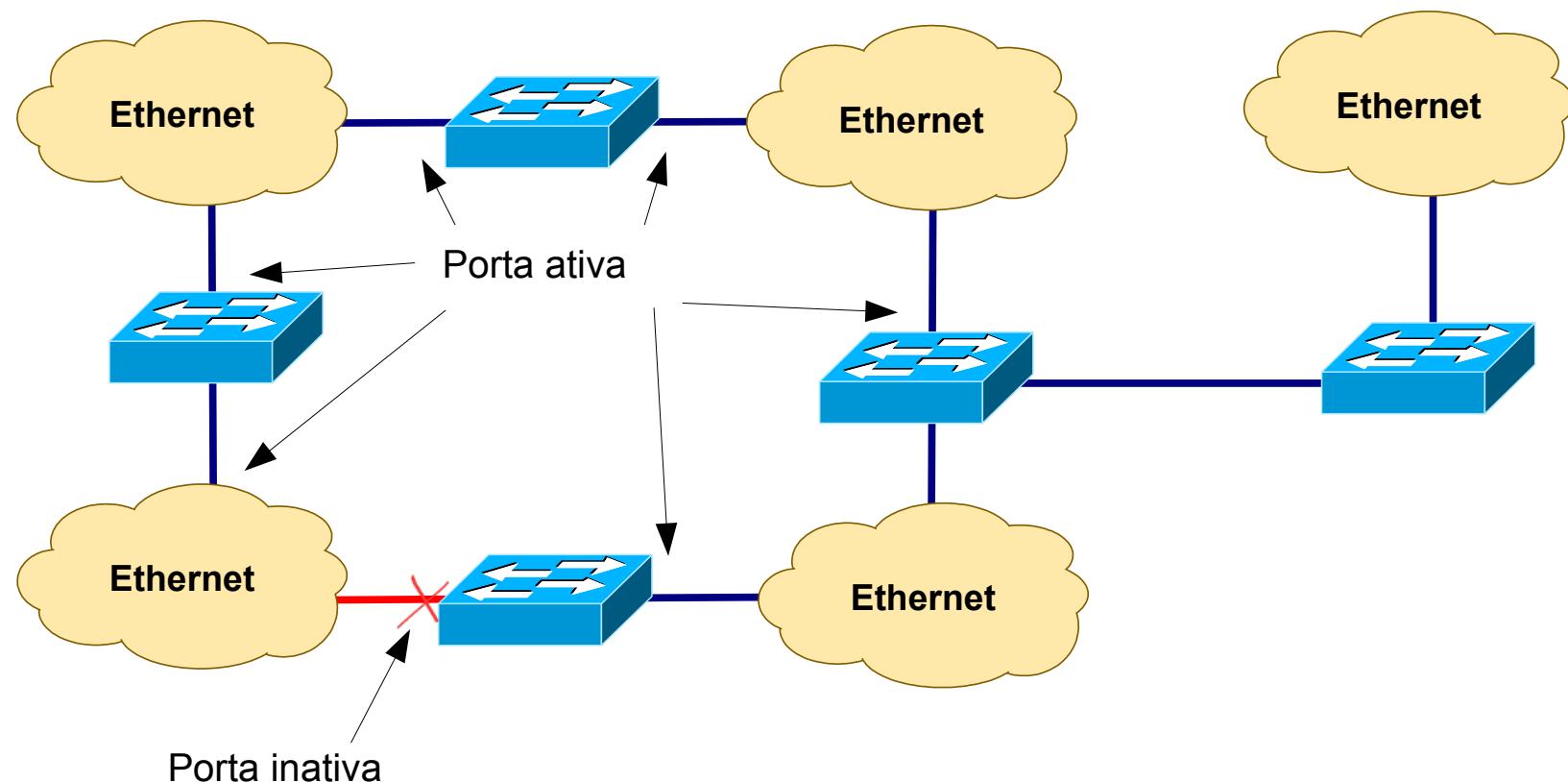
- Podemos ter vários switches mas temos de ter cuidado com os LOOP ele iria ficar infinitamente a enviar ....
- Imagina um broadcast ...

- Objectivo: proporcionar capacidade de recuperação a falhas na rede
- Problema: redundância lógica provoca colapso de comunicações devido ao envio de tramas MAC para o endereço de broadcast

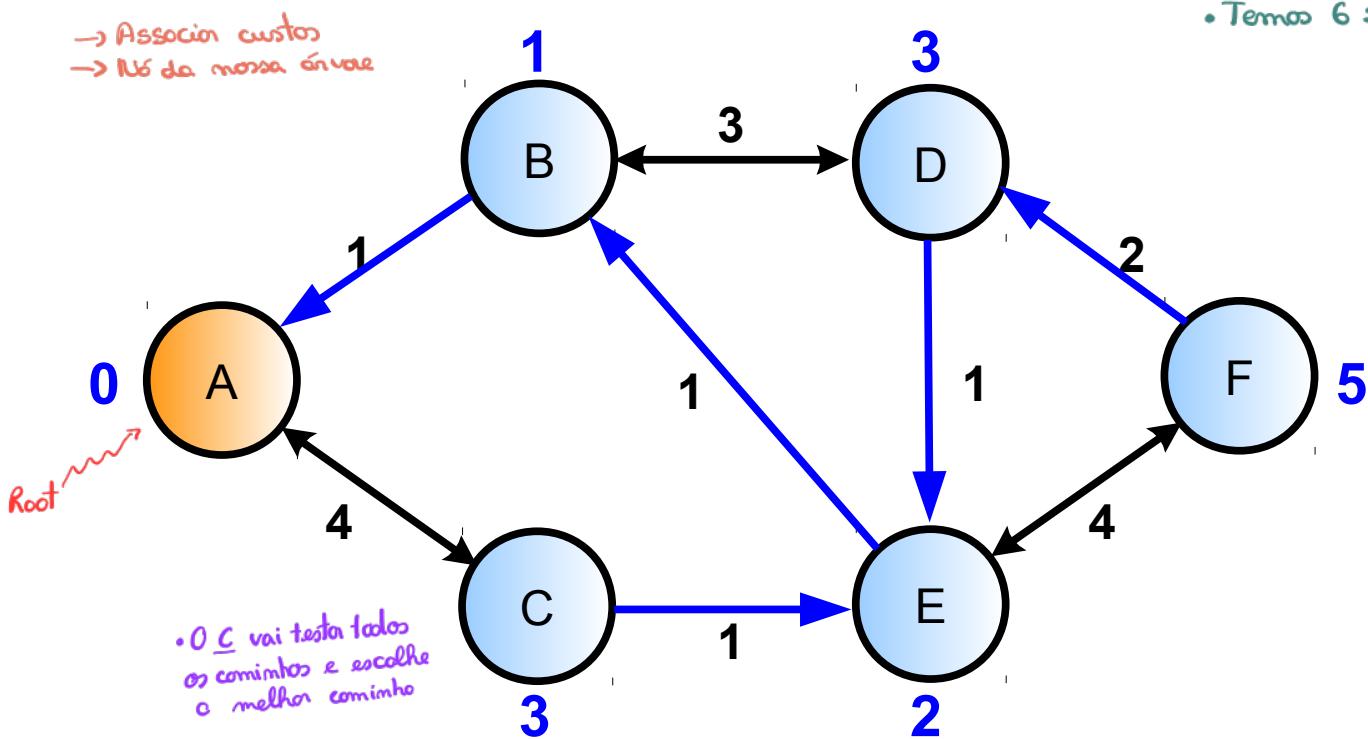
# Spanning tree numa rede de LANs (LAN estendida)

↓  
Protocolo por  
Software! //

• Precisamos desativar uma das portas!



# Equações de Bellman



- Quando os custos das ligações são não negativos, então:  
**Comprimento do percurso mínimo de um nó para A**

=

Comprimento do arco que une esse nó ao nó que se lhe segue no percurso mínimo

+

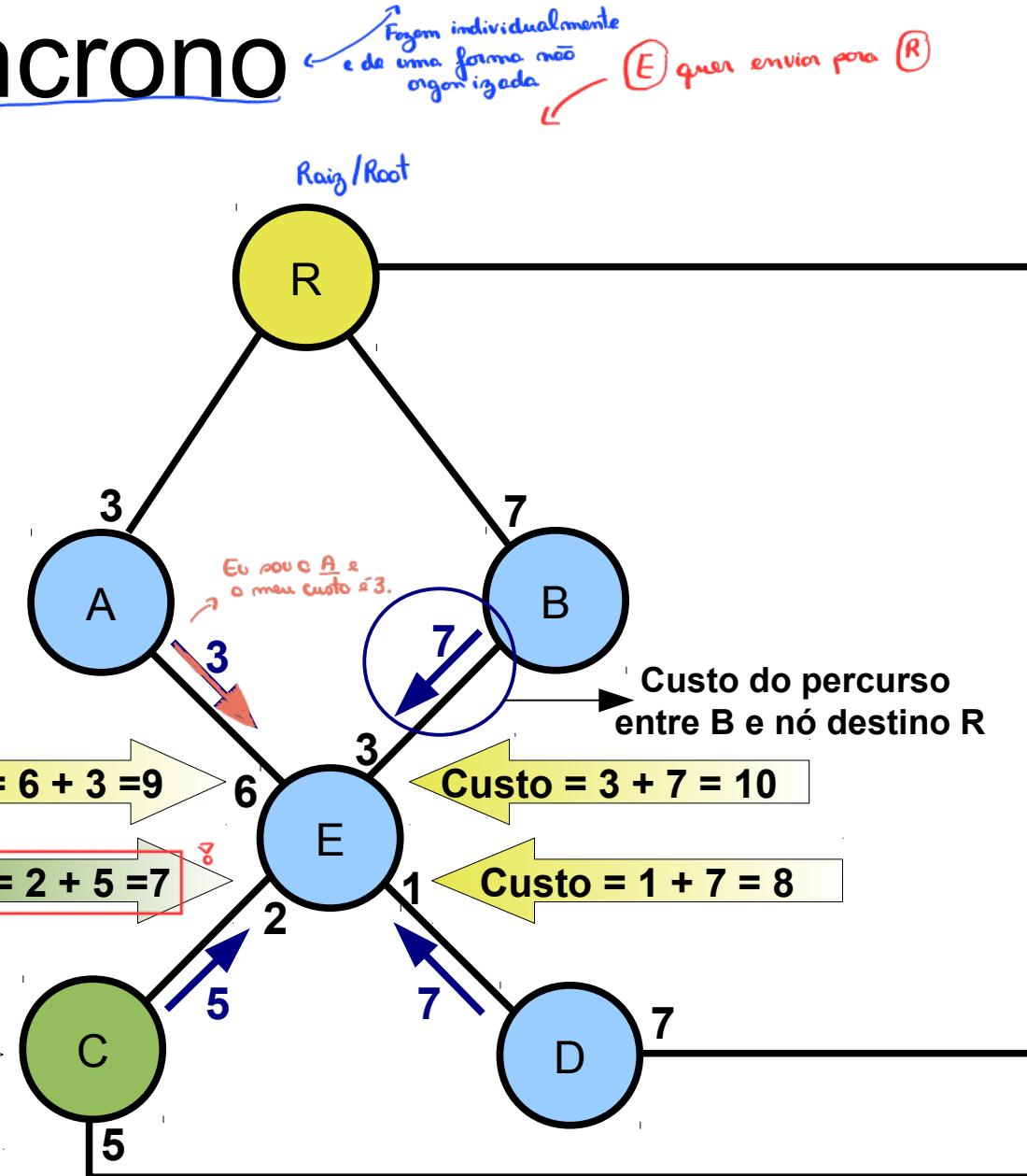
Comprimento do percurso mínimo desse nó para o nó A



# Algoritmo de Bellman-Ford distribuído e assíncrono

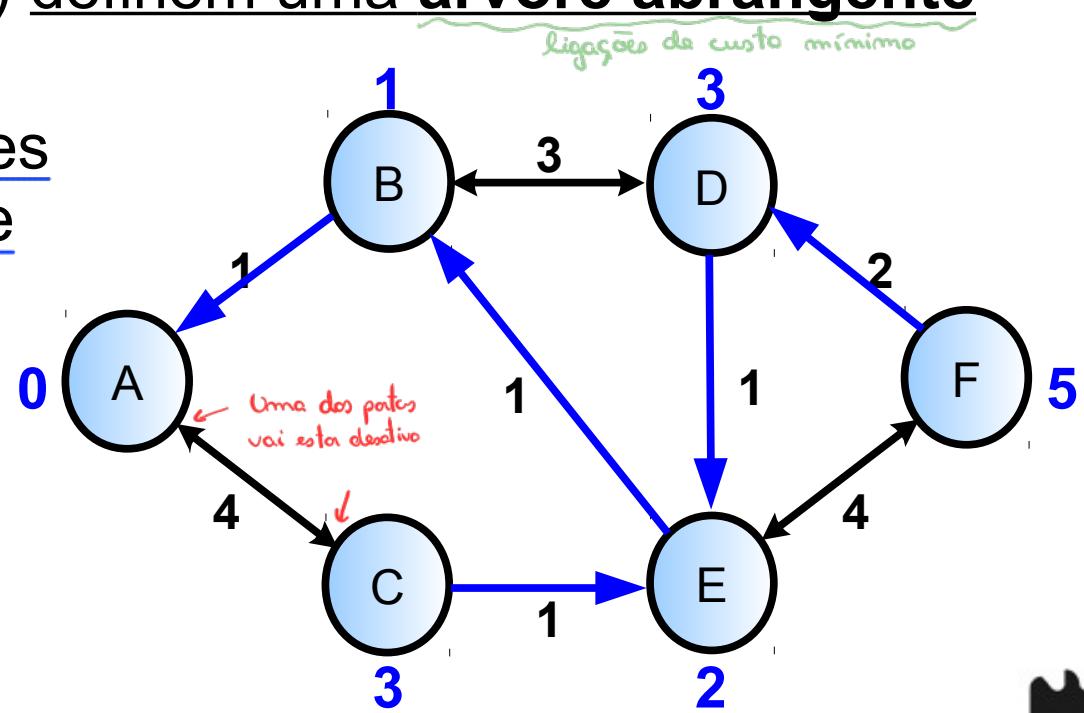
- Cada switch irá estor a correr um algoritmo baseado nos egrupos de Bellman-Ford

- Cada nó transmite de tempos a tempos a sua estimativa do custo do percurso entre ele e o nó destino
  - Cada nó recalculate o estimativa do custo até ao nó destino:
    - Somando as estimativas recebidas dos vizinhos ao custo da ligação/porta por onde receberam o anúncio do vizinho
    - Escolhem o menor valor

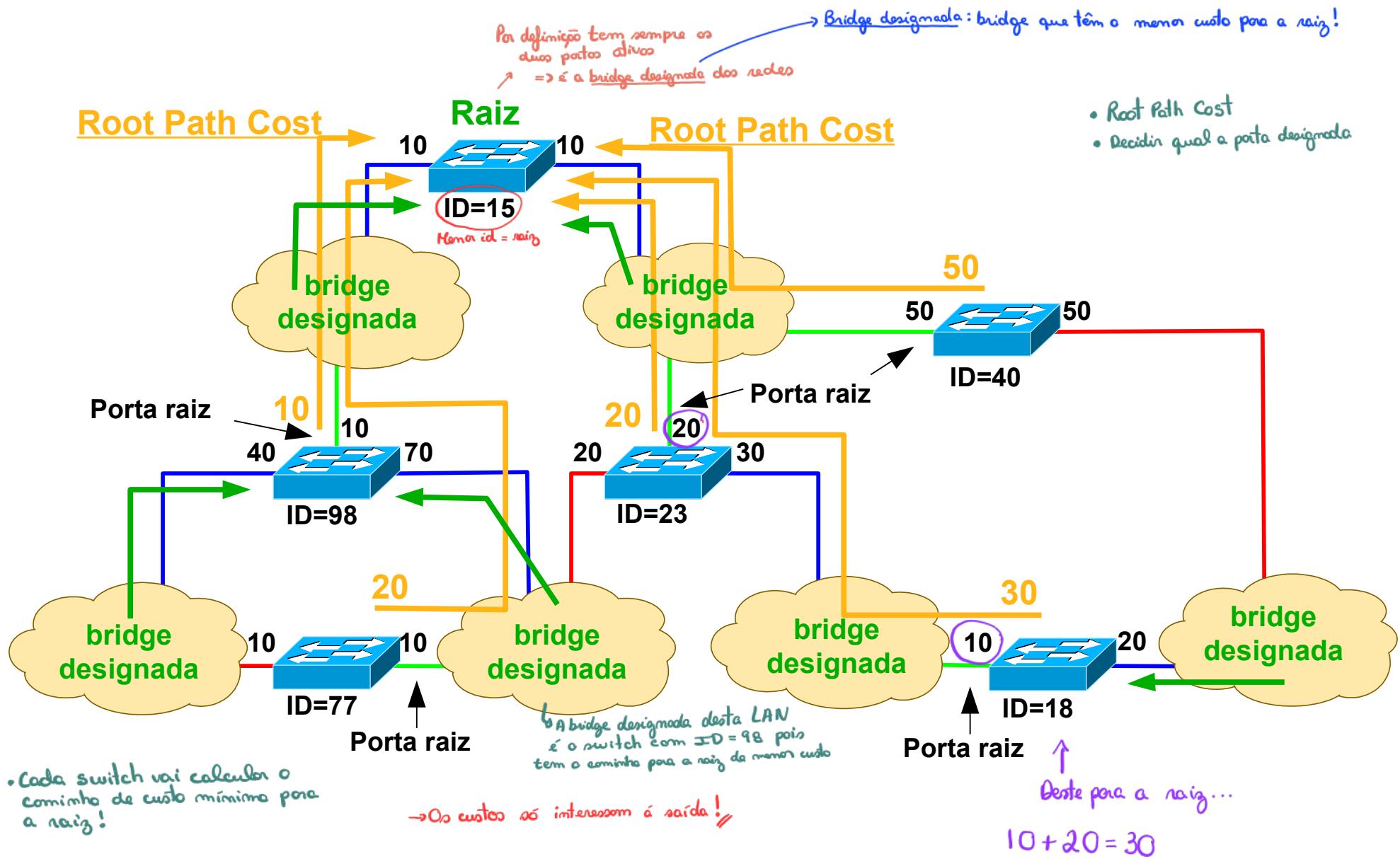


# Encaminhamento baseado em Spanning Trees

- É escolhido um switch como **nó origem/raiz**
- Os outros switches usam o **algoritmo de Bellman-Ford assíncrono e distribuído** para calcular o vizinho no percurso de custo mínimo para o nó origem/raiz
- As ligações compostas pelos percursos de custo mínimo (de todos os outros switches para a origem/raiz) definem uma árvore abrangente (spanning tree)
- As portas ativas são as das ligações que compõem a árvore abrangente
- É necessário um critério para desempatar quando há múltiplos percursos de custo mínimo

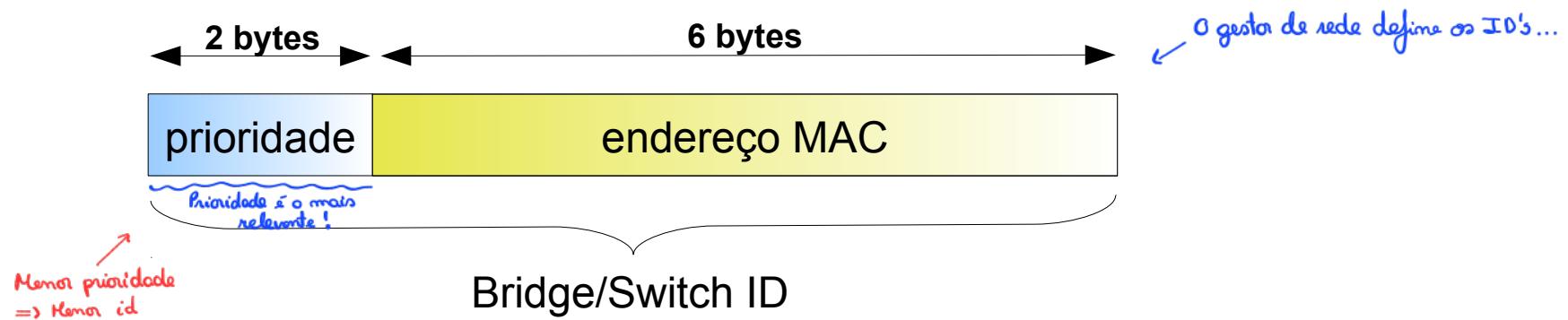


# Conceitos básicos spanning tree (I)



# Conceitos básicos spanning tree (II)

- Bridge/Switch ID – cada switch é identificado por um endereço que contém:
  - ◆ 2 octetos de prioridade, configurável pelo gestor da rede
  - ◆ 6 octetos fixos (um dos endereços MAC das portas do switch, ou qualquer outro endereço único de 48 bits)
  - ◆ A prioridade tem precedência sobre o campo de octetos fixos



- Switch/bridge raiz (Root) – switch que está na raiz da spanning tree
  - ◆ Switch com menor ID
- Path cost – custo associado a cada porta do switch
  - ◆ Pode ser configurado pelo gestor da rede

# Conceitos básicos spanning tree (III)

*Switch que está no caminho para a raiz*

- Bridge designada (Designated Bridge) – bridge que, numa LAN, é responsável pelo envio de pacotes da LAN para a raiz e vice-versa

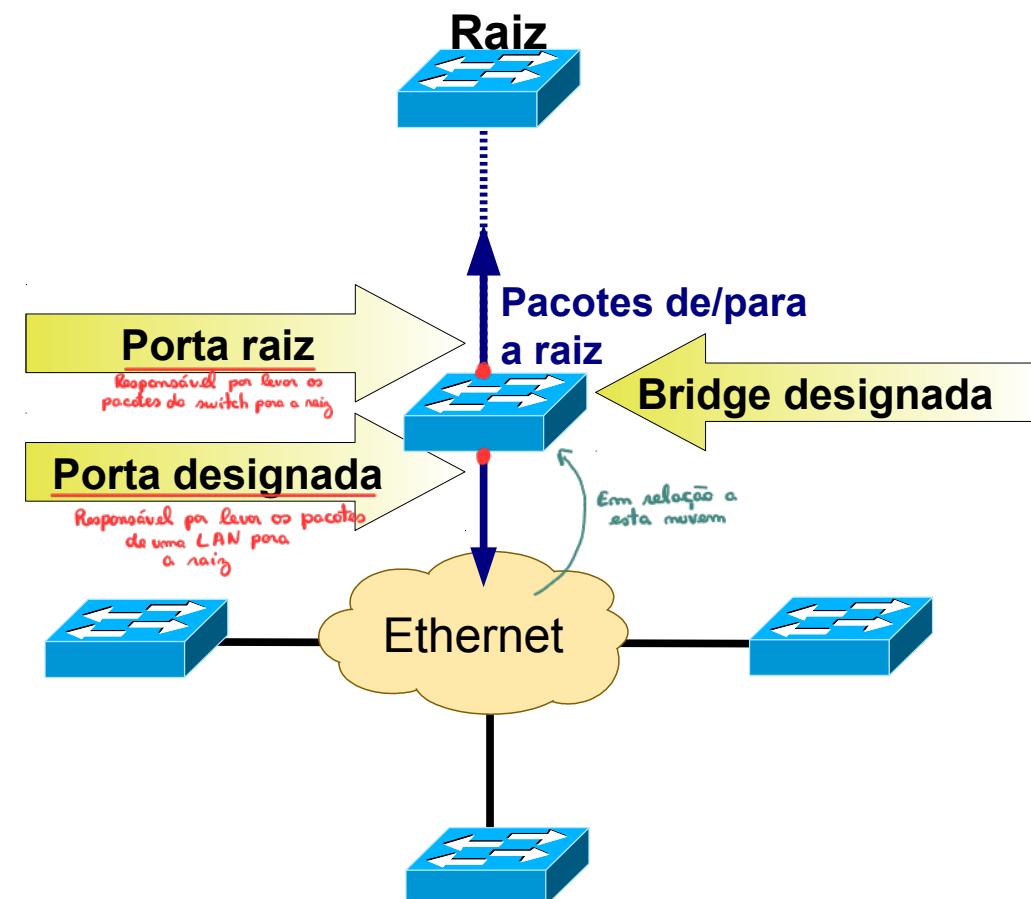
- ◆ A bridge raiz é a bridge designada em todas as LANs a que está ligada

*Porta do switch que liga à LAN*

- Porta designada (Designated Port) – porta que, numa LAN, é responsável pelo envio de pacotes da LAN para a raiz e vice-versa (uma das portas da bridge designada)

*Porta de comutação é Raiz*

- Porta raiz (Root Port) – porta responsável pela recepção/transmissão de pacotes de/para a bridge raiz



# Conceitos básicos spanning tree (IV)

- Cada bridge tem associado um custo do percurso para a raiz (Root Path Cost), igual à soma dos custos das portas que recebem pacotes enviados pela raiz (portas raiz) no percurso de menor custo para a bridge  
*→ Soma dos custos dos portas até chegar à raiz*
- A porta raiz é, **em cada bridge**, a porta que fornece o melhor percurso (de menor custo) para a raiz
- A porta designada é, **em cada LAN**, a porta que fornece o melhor percurso para a raiz
- As portas activas em cada bridge são a porta raiz + as portas designadas
  - As restantes portas ficam inactivas (blocking)  
*• Não estão desligadas!  
→ Apenas não contribuem...*



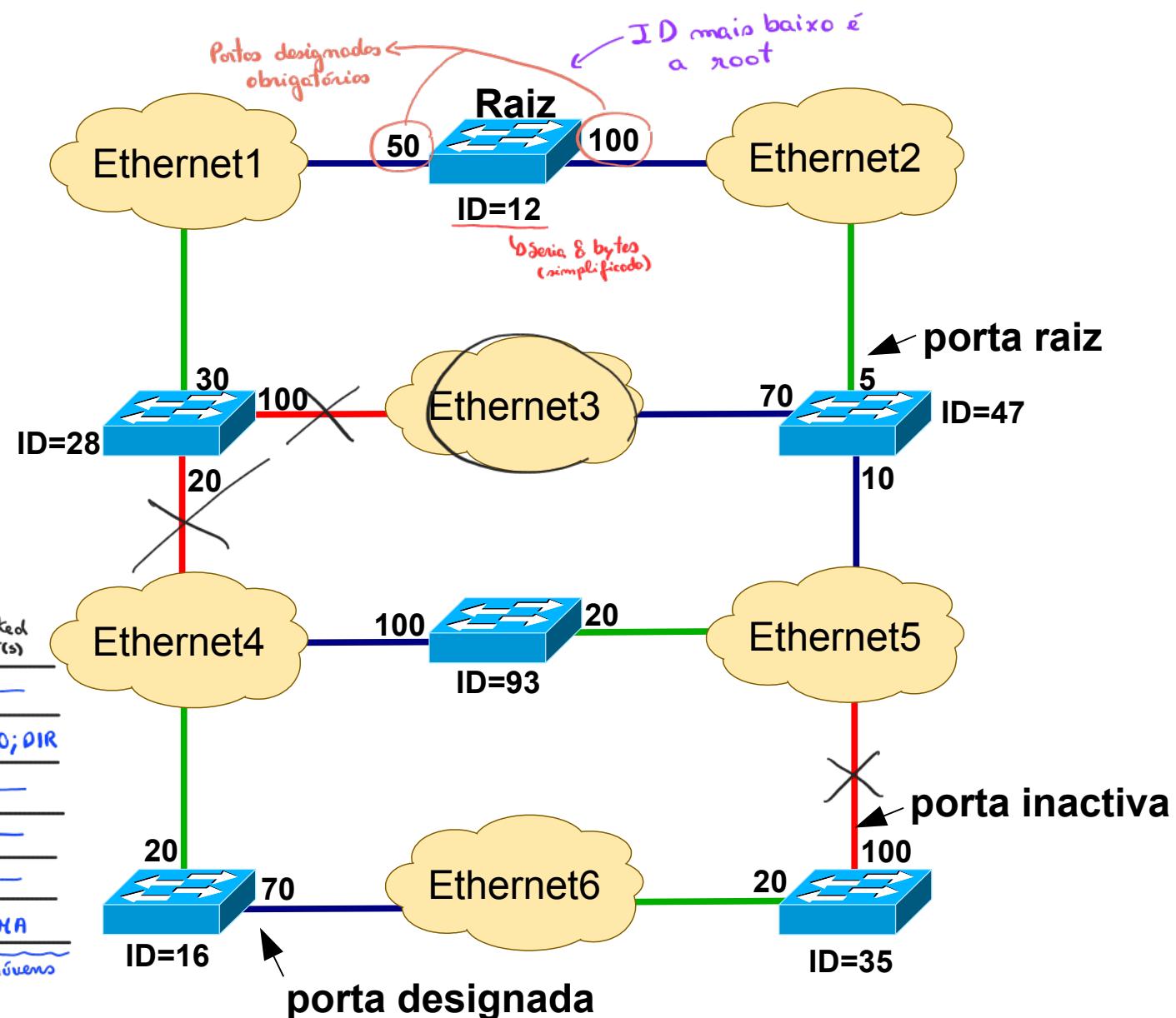
# Exemplo – spanning tree (I)

## Bridges designadas

Eth1	12
Eth 2	12
Eth 3	47
Eth 4	93
Eth 5	47
Eth 6	16

Switch ID	Root Path cost	Root Port	Designated Port(s)	Blocked Port(s)
Raiz → 12	0	—	ESQ; DIR	—
28	30	CIMA	—	BAIXO; DIR
47	5	CIMA	BAIXO; ESQ	—
93	25	DIR	ESQ	—
16	45	CIMA	DIR	—
35	65	ESQ	—	CIHA

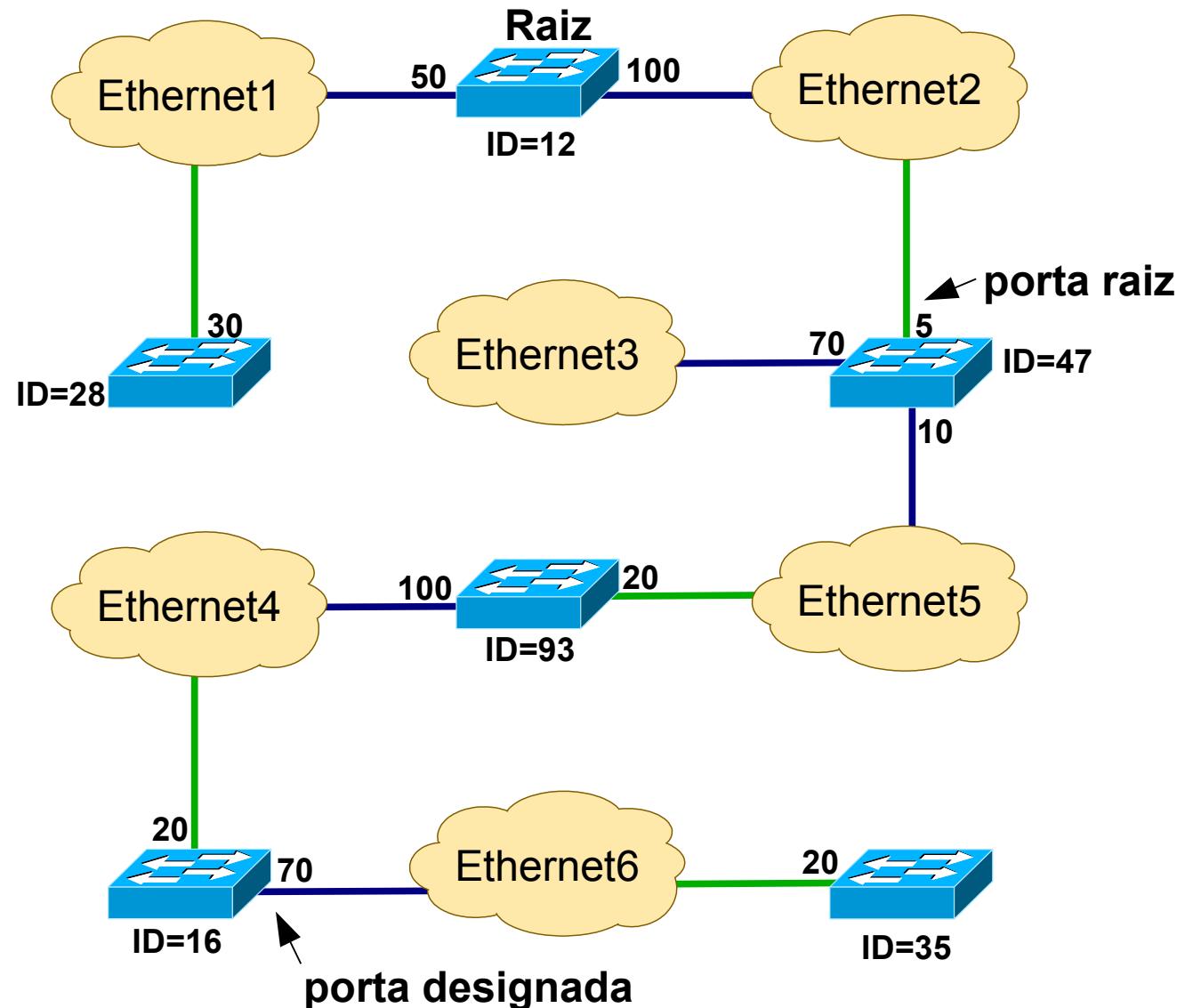
Com base nos switches      Com base nos máximos



# Exemplo – spanning tree (II)

## Bridges designadas

Eth1	12
Eth 2	12
Eth 3	47
Eth 4	93
Eth 5	47
Eth 6	16



Como na realidade  
funcionam  $\Rightarrow$

# Protocolo IEEE 802.1D

## BPDUs (Bridge Protocol Data Units)

## Nivel 2 - MAC Address

- Para construir e manter a spanning tree as bridges trocam mensagens especiais entre si, designadas por Bridge Protocol Data Units (BPDUs)
  - Existem dois tipos: *Configuration* e *Topology Change Notification*

# IEEE 802.3 Ethernet

Destination: 01:80:c2:00:00:00 (01:80:c2:00:00:00)

Source: 00:16:e0:9a:c3:92 (00:16:e0:9a:c3:92)

Length: 39

## Logical-Link Control

DSAP: Spanning Tree BPDU (0x42)

#### SSAP: Spanning Tree BPDU (0x42)

Control field: U, func=UI (0x03)

## Spanning Tree Protocol

Protocol Identifier: Spanning Tree Protocol (0x0000)

#### Protocol Version Identifier: Spanning Tree (0)

BPDU Type: Configuration (0x00) → Configuration or Topology Change Notification

Root ID: 32768 / 00:05:1a:4e:fd:58 => Identifica que é a Raiz!

Root Path Cost: 200004

Bridge ID: 32768 / 00:16:e0:9a:c3:80  $\Rightarrow$  IP do switch que enviou o pacote

Port ID: 0x8012 SW SW => Para distinguir entre 2 ligações

## Part IV: exec12

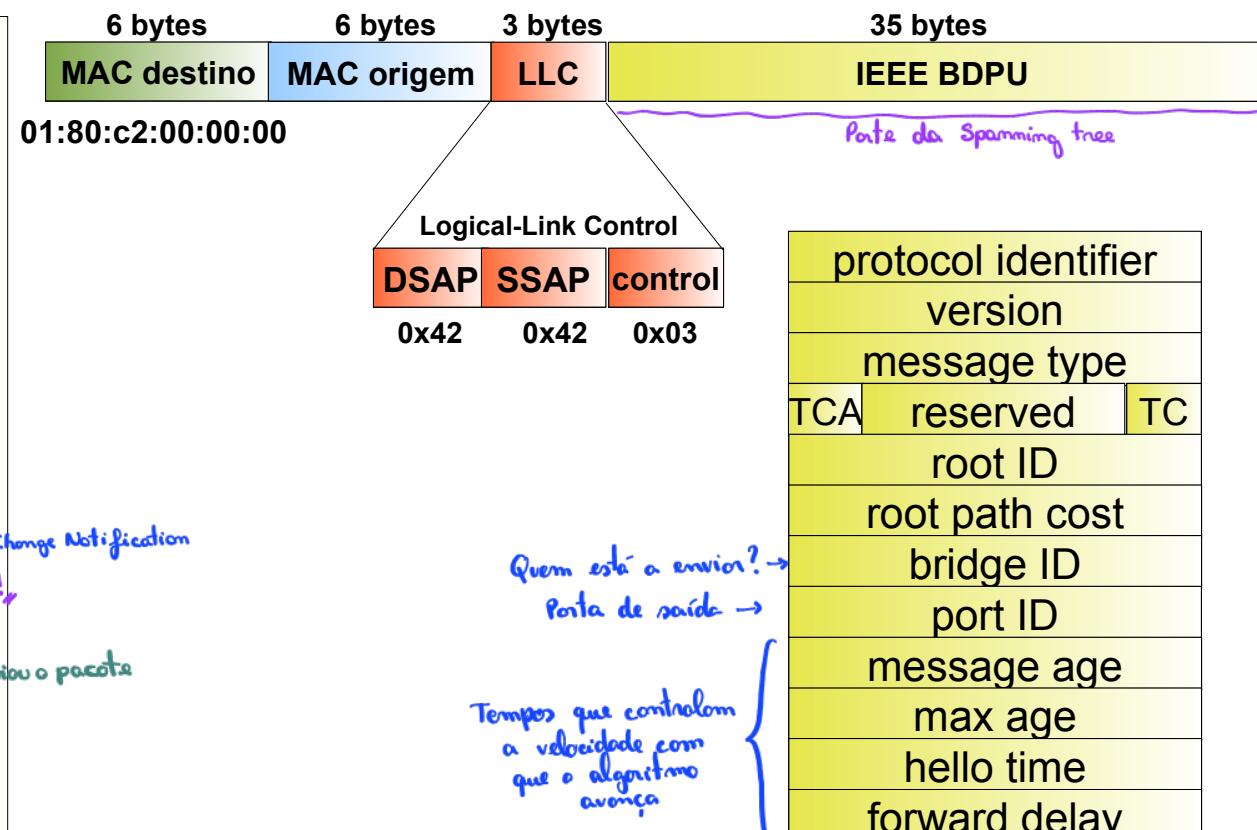
## Message Age

Max Age: 25  
Hello Time: 3

## Hello Time 2

**[SN]** **[SW]** => Para distinguir entre 2 ligações entre dois switches!

#### Environmental temperature



# Configuration BPDUs

- A configuração da Spanning Tree é feita pelas Conf - BPDUs (mensagens de configuração)

## IEEE 802.3 Ethernet

Destination: 01:80:c2:00:00:00 (01:80:c2:00:00:00)  
Source: 00:16:e0:9a:c3:92 (00:16:e0:9a:c3:92)  
Length: 39

## Logical-Link Control

DSAP: Spanning Tree BPDU (0x42)  
SSAP: Spanning Tree BPDU (0x42)  
Control field: U, func=UI (0x03)

## Spanning Tree Protocol

Protocol Identifier: Spanning Tree Protocol (0x0000)

Protocol Version Identifier: Spanning Tree (0)

BPDU Type: Configuration (0x00)

Root ID: 32768 / 00:05:1a:4e:fd:58 ← Root?

Root Path Cost: 200004 ← custo?

Bridge ID: 32768 / 00:16:e0:9a:c3:80 ← Quem enviou?

Port ID: 0x8012 ← Porta?

Message Age: 1

Max Age: 20

Hello Time: 2

Forward Delay: 15

Com base nos pacotes e com base  
na sua informação e utilizando  
os Algoritmos de Bellman - Ford



- Campos mais importantes:
  - Root ID: estimativa actual do endereço da bridge raiz
  - Root Path Cost: estimativa actual do custo para a bridge raiz
  - Bridge ID: endereço da bridge que envia a mensagem de configuração
  - Port ID: identificação da porta que envia a mensagem de configuração
    - Prioridade (1 byte) + Número da porta

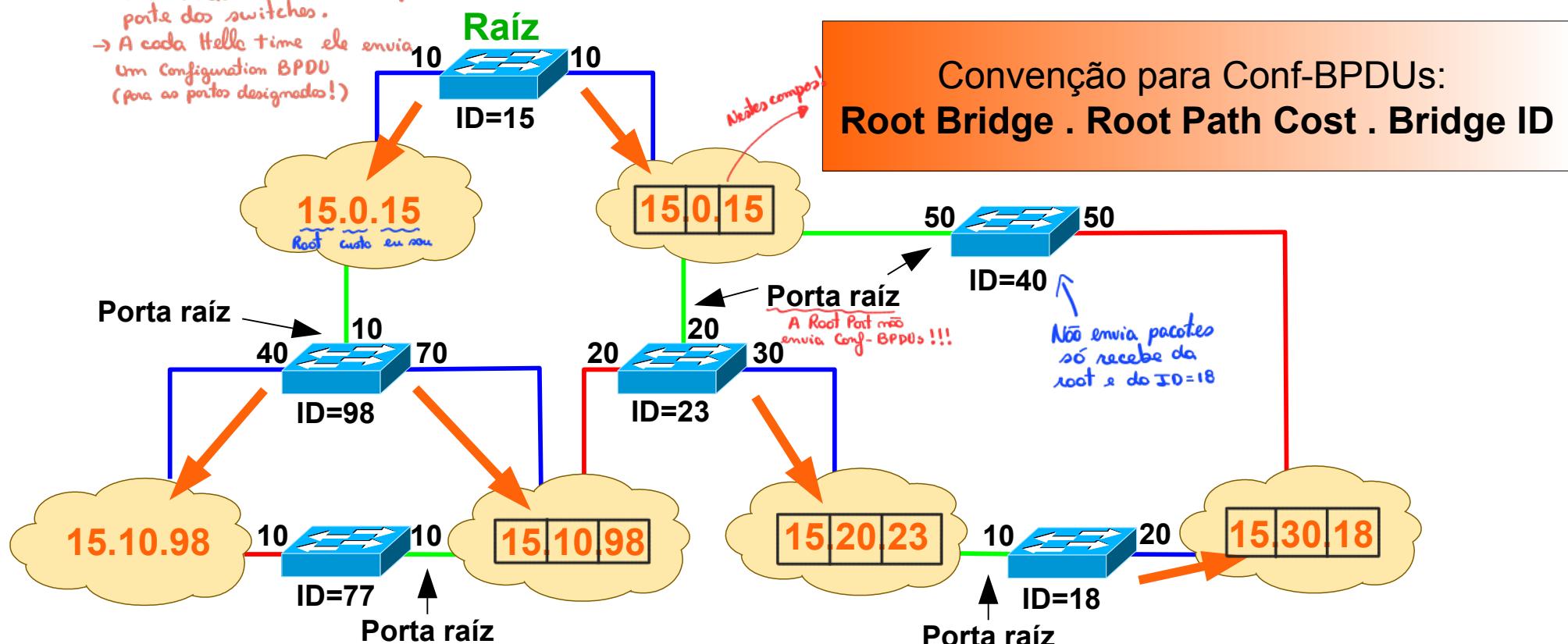


# Manutenção da spanning tree

- Periodicamente as bridges enviam pelas portas designadas Conf-BPDUs
  - Muito importante: os Conf-BPDUs são enviados pelos portos designados!

- Periodicidade das mensagens Conf-BPDUs = hello time
  - Hello time recomendado: 2 segundos

→ Periodicidade do envio de pacotes por parte dos switches.  
→ A cada Hello Time ele envia um Configuration BPDU (para as portas designadas!)



# Ordenação das mensagens de configuração

Pacotes em portos diferentes de switches diferentes

- Uma mensagem de configuração C1 diz-se melhor que outra C2 se:

- o Root ID de C1 for inferior ao de C2
- sendo os Root ID idênticos, o Root Path Cost de C1 for inferior ao de C2
- sendo idênticos o Root ID e o Root Path Cost, o Bridge ID de C1 for inferior ao de C2
- Desempate → sendo idênticos o Root ID, o Root Path Cost e o Bridge ID, o Port ID de C1 for inferior ao de C2

Mensagem preferencial ! →

Root ID	Root Path Cost	Bridge ID	Port ID
18	27	32	2
18	27	32	4
18	27	43	1
18	35	23	3
23	31	45	2

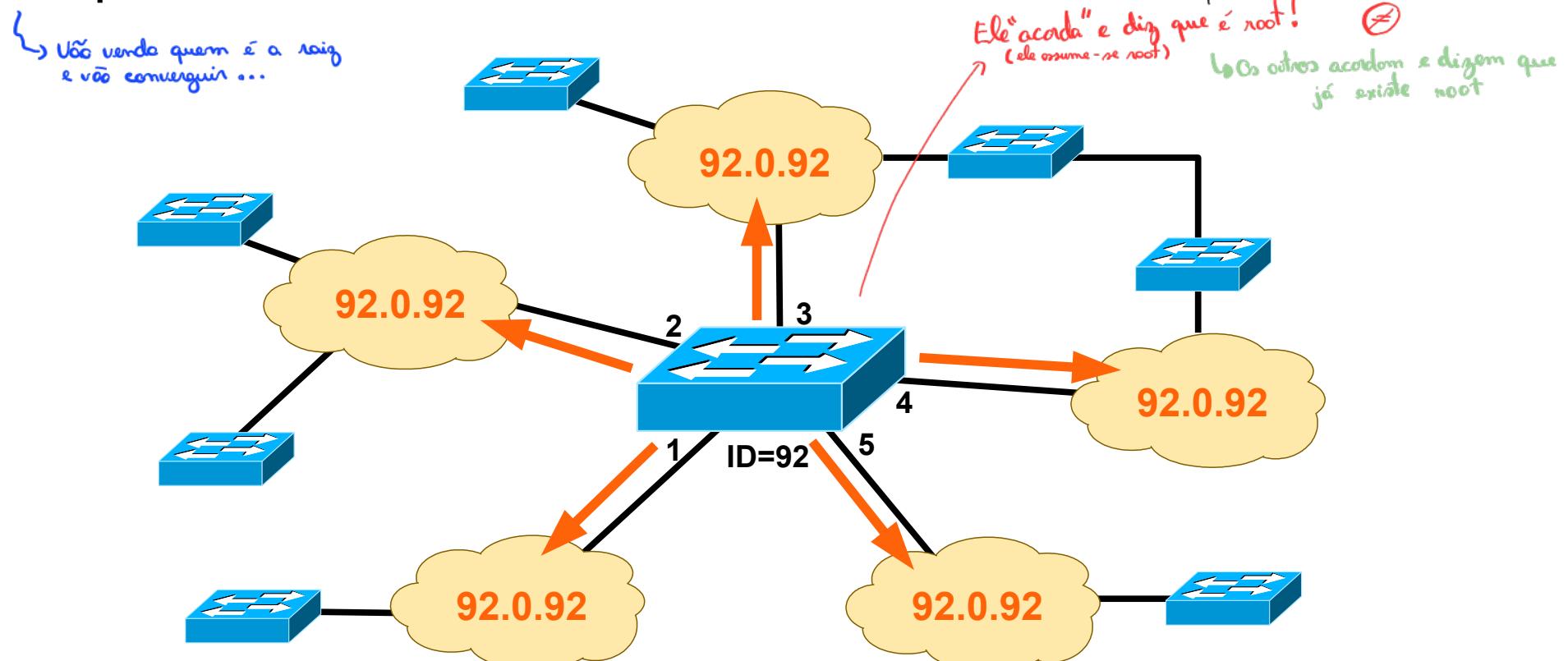
Vêm do mesmo Switch mas de portos diferentes



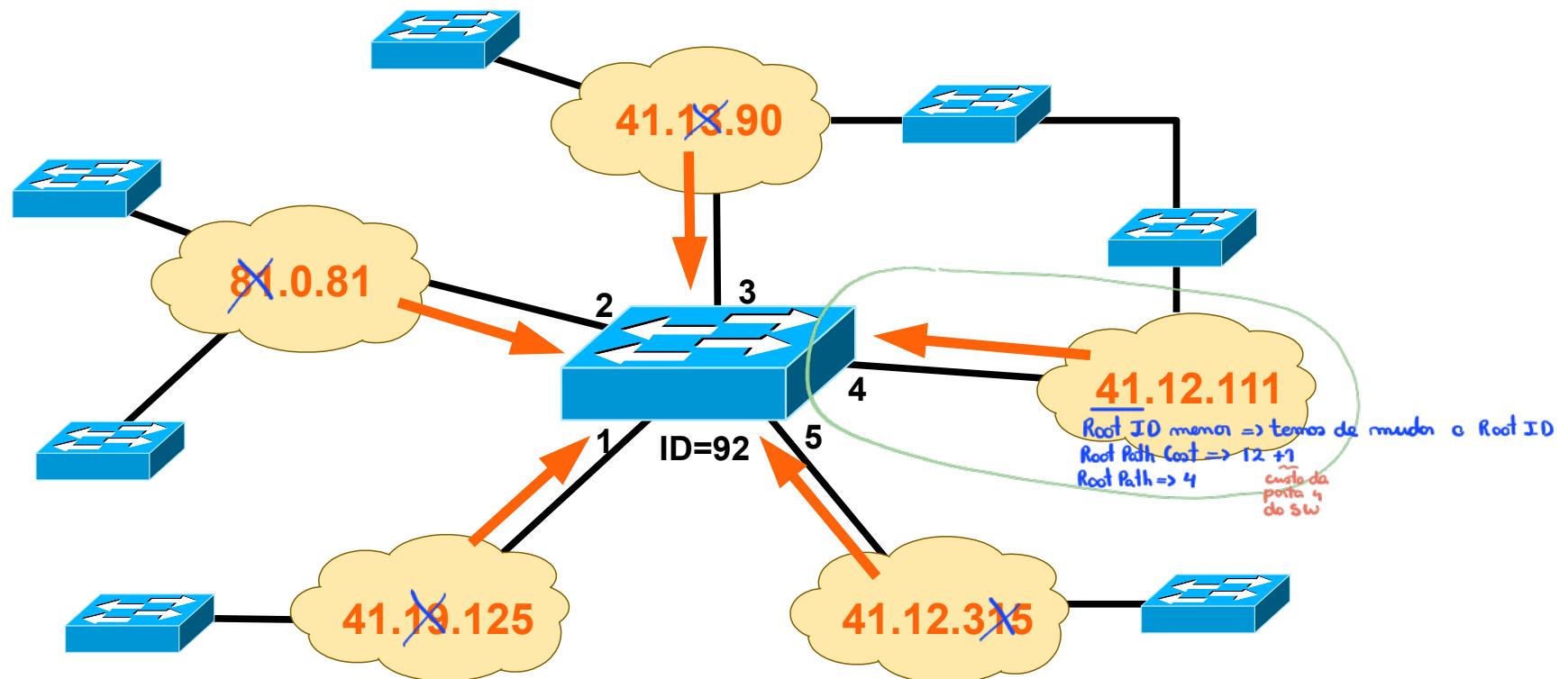
# Construção da spanning tree (I)

No início → Começam a enviar pacotes uns para os outros (Conf-BPDUs)

- Cada bridge assume inicialmente que é a bridge raiz (faz Root Path Cost = 0); envia mensagens de configuração em todas as suas portas



# Construção da spanning tree (II)



Melhores mensagens recebidas na Bridge 92 até um dado instante

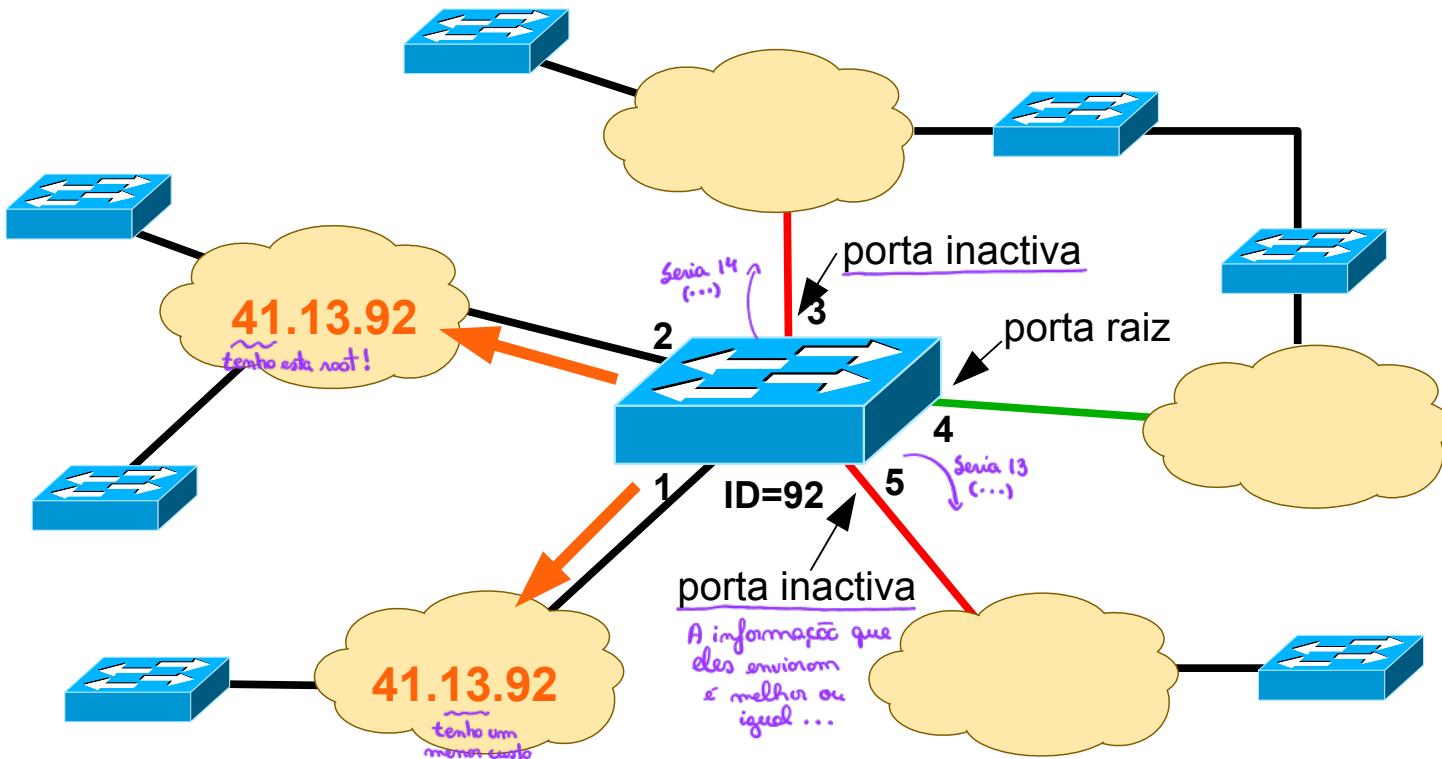
Estimativas da Bridge 92 (assumindo que os custos das portas são unitários)

Bridge raiz = 41  
Porta raiz = 4  
Custo para a raiz = 12 + 1

*Custo dos portos do switch são 1 //*



# Construção da spanning tree (III)



Mensagens enviadas pela Bridge 92 - 41.13.92

O Vai enviar para os portos que estão "piores" que ele

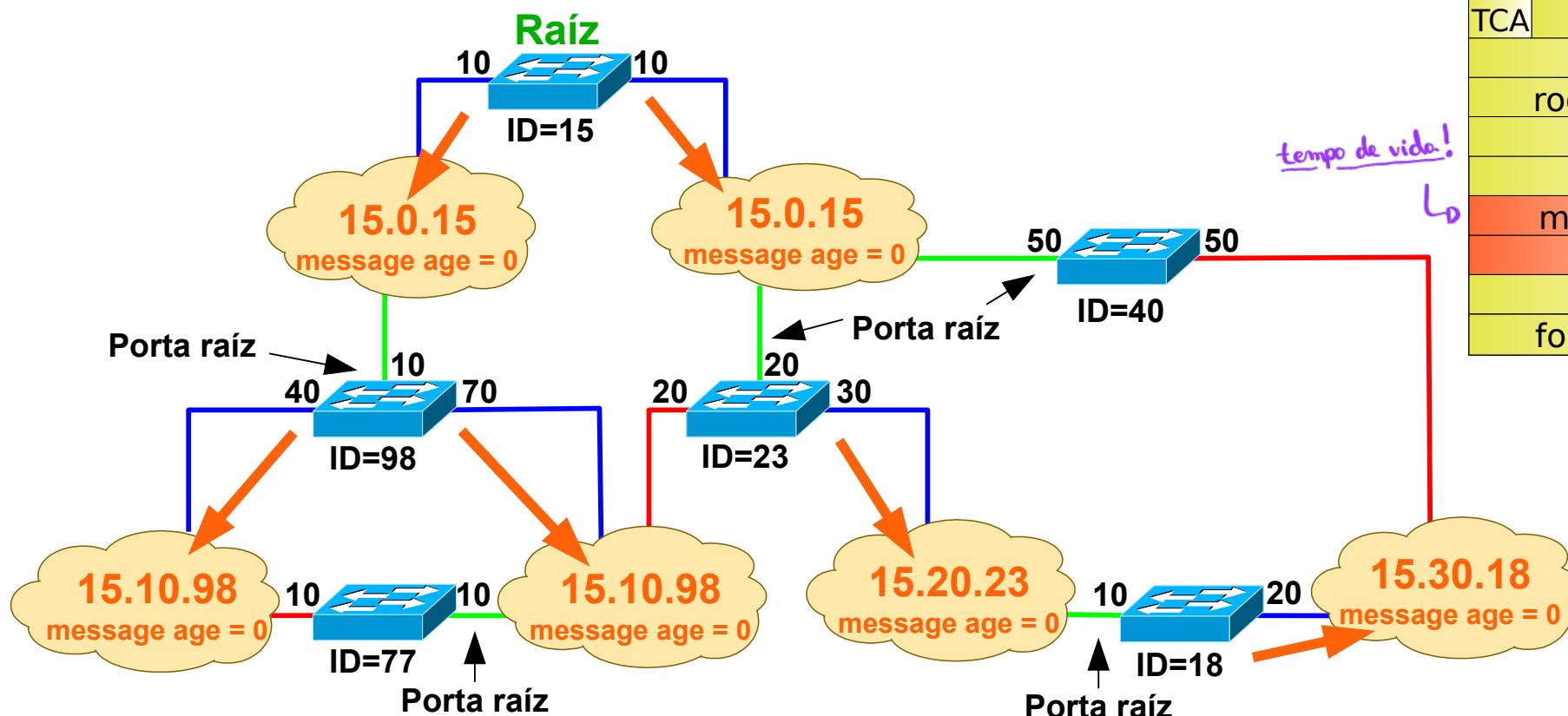
ex: O do lado esquerdo não sabia a Root...



# Avarias nas bridges ou nas LANs (I)

• Depois de convergirem

→ Eles vão enviando conforme o Hello Time

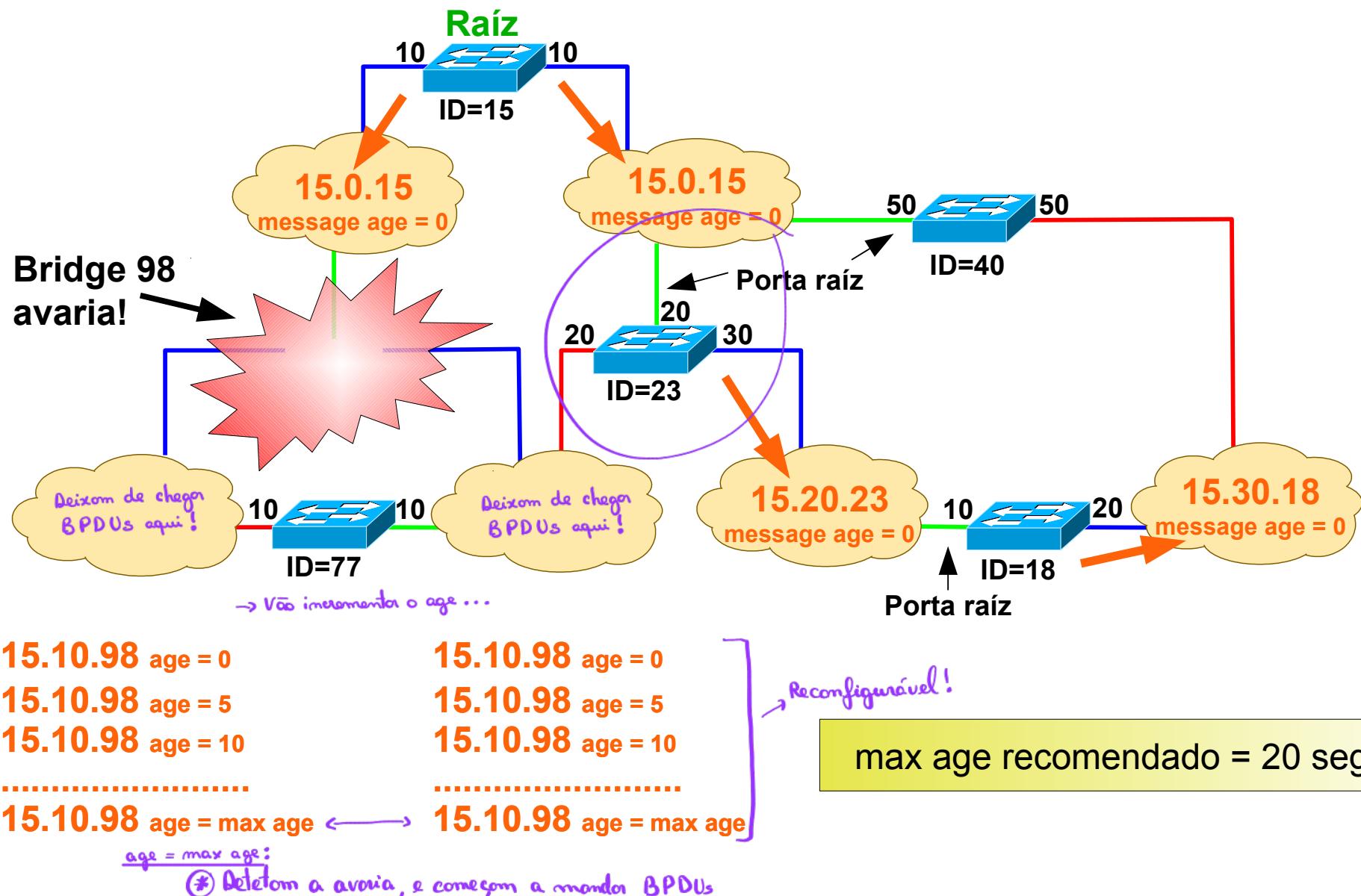


Conf-BPDU

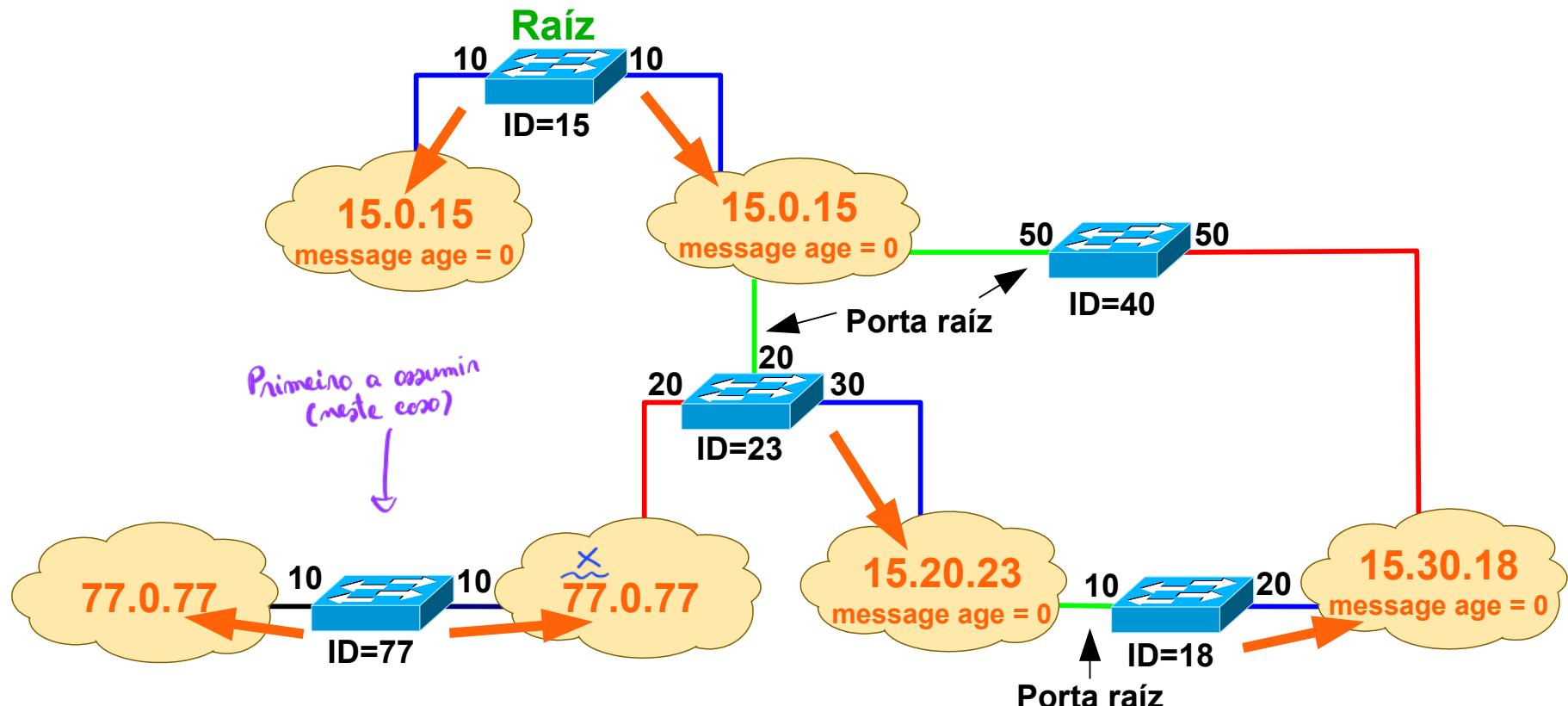
protocol identifier	TC	
version		
message type		
TCA	reserved	TC
root ID		
root path cost		
bridge ID		
port ID		
message age		
max age		
hello time		
forward delay		



# Avarias nas bridges ou nas LANs (II)



# Avarias nas bridges ou nas LANs (III)

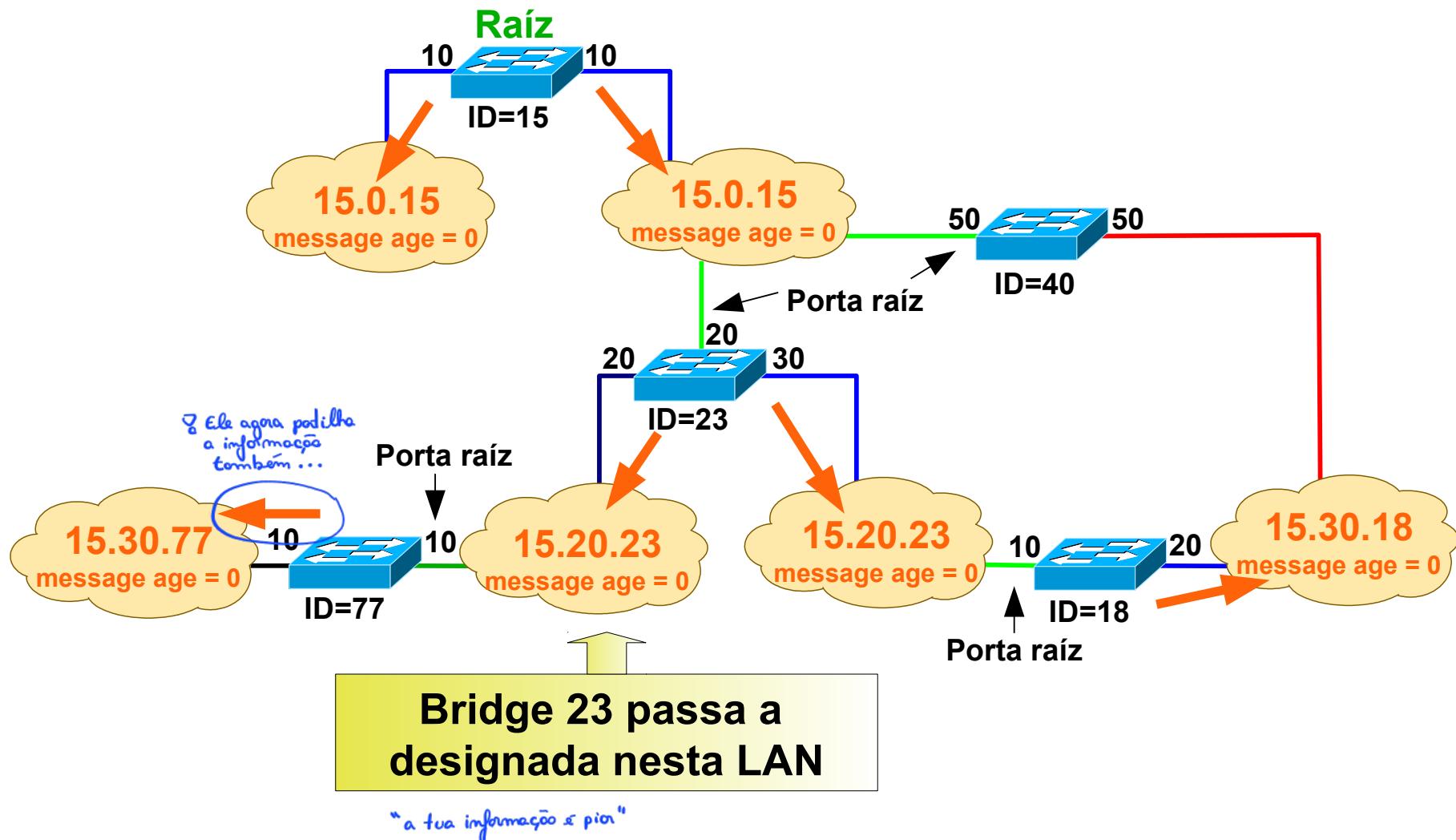


Bridge 77 assume ser a raiz

Começa tudo do zero para ele...



# Avarias nas bridges ou nas LANs (IV)



# Existência de ciclos temporários

- Após alteração da topologia da rede: → Mudança de estados...
  - ◆ Pode existir perda temporária de conectividade se uma porta que estava inativa na topologia antiga ainda não se apercebeu que deverá estar ativo na nova topologia  
↳ termos de cinco estados intermédios
  - ◆ Podem existir ciclos temporários se uma porta que estava ativa na topologia antiga ainda não se apercebeu que deverá estar inativa na nova topologia
- Para minimizar a probabilidade de se formarem ciclos temporários as *bridges* são obrigadas a esperar algum tempo antes de permitirem que uma das suas portas passe do estado inativo para o estado ativo; o tempo de espera é função do parâmetro forward delay  
↳ Esperar para mudar de ativo ⇔ inativo



Demora a passar de Blocking → Forwarding  
 $2 \times \text{forwarding delay}$   
~ 40 segundos



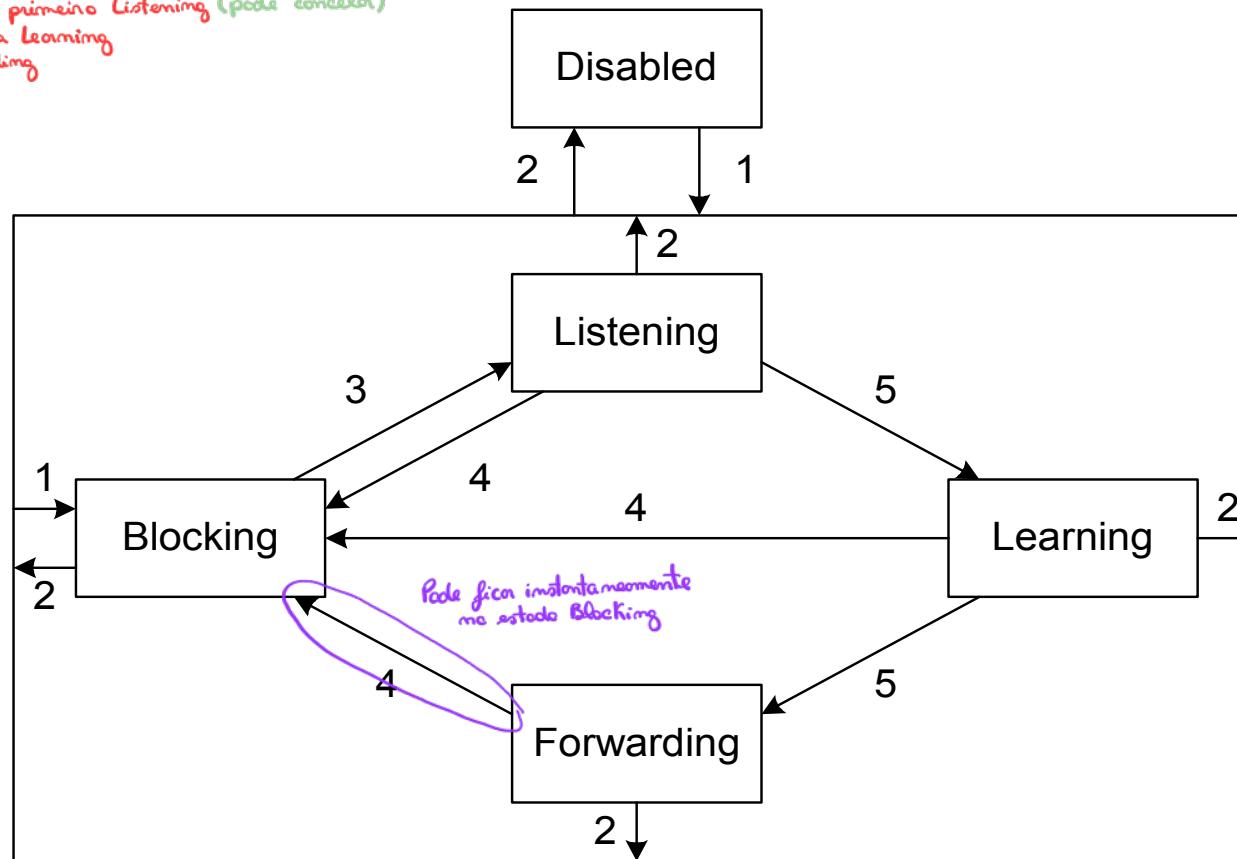
# Estados das portas da bridge

- Nem sequer aprende MAC's, nem faz forwarding / flooding.
- Estado **blocking**: os processos de aprendizagem e de expedição de pacotes estão inibidos; recebe e processa mensagens de configuração → Nos têm de estar atentos a mudanças!
  - Estado **listening**: os processos de aprendizagem e de expedição de pacotes estão inibidos; transita para o estado learning após um tempo de permanência neste estado igual a forward delay; recebe e processa mensagens de configuração → Espera simplesmente ...
  - Estado **learning**: o processo de aprendizagem está activo mas o processo de expedição de pacotes está inibido; transita para o estado forwarding após um tempo de permanência neste estado igual a forward delay; recebe e processa mensagens de configuração → Espera simplesmente ...
  - Estado **forwarding**: é o estado activo; tanto o processo de aprendizagem com o processo de expedição de pacotes estão activos; recebe e processa mensagens de configuração → Os MAC's já são Store-and-forward.
  - Estado **disabled**: os processos de aprendizagem e de expedição de pacotes estão inibidos; não participa no algoritmo de spanning tree  
Desliga a porta ...



# Diagrama de estados das portas

- Não existe: Blocking → Forwarding
  - ele fica primeiro Listening (pode cancelar)
  - vai para Learning
  - Forwarding



- 1 - Porta activada por gestão ou por inicialização
- 2 - Porta desactivada, por gestão ou falha
- 3 - Algoritmo selecciona como sendo porta designada ou porta raiz
- 4 - Algoritmo selecciona como não sendo porta designada ou porta raiz
- 5 - Forwarding timer expira



# Tempo de vida das entradas das tabelas de encaminhamento

Na reconfiguração os tabelos de encaminhamento pode ficar totalmente errada ... } tempo de vida vai variando, conforme a árvore ...

- Tempo de vida demasiado longo – pode haver um número exagerado de pacotes perdidos quando a estação muda de localização. (Se uma porta mudar de estado)
- Tempo de vida demasiado curto – o tráfego na rede pode ser exagerado devido ao processo de **flooding**  
↳ Rede inundada de pacotes
- Existem dois tempos de vida:
  - ◆ **Longo**: usado por defeito (valor recomendado = 5 minutos)
  - ◆ **Curto**: usado quando a spanning tree está em reconfiguração (valor recomendado = 15 segundos) - exige processo de notificação de alterações da topologia da rede



# Notificação de alterações da topologia

Conf (Configuration) BPDU

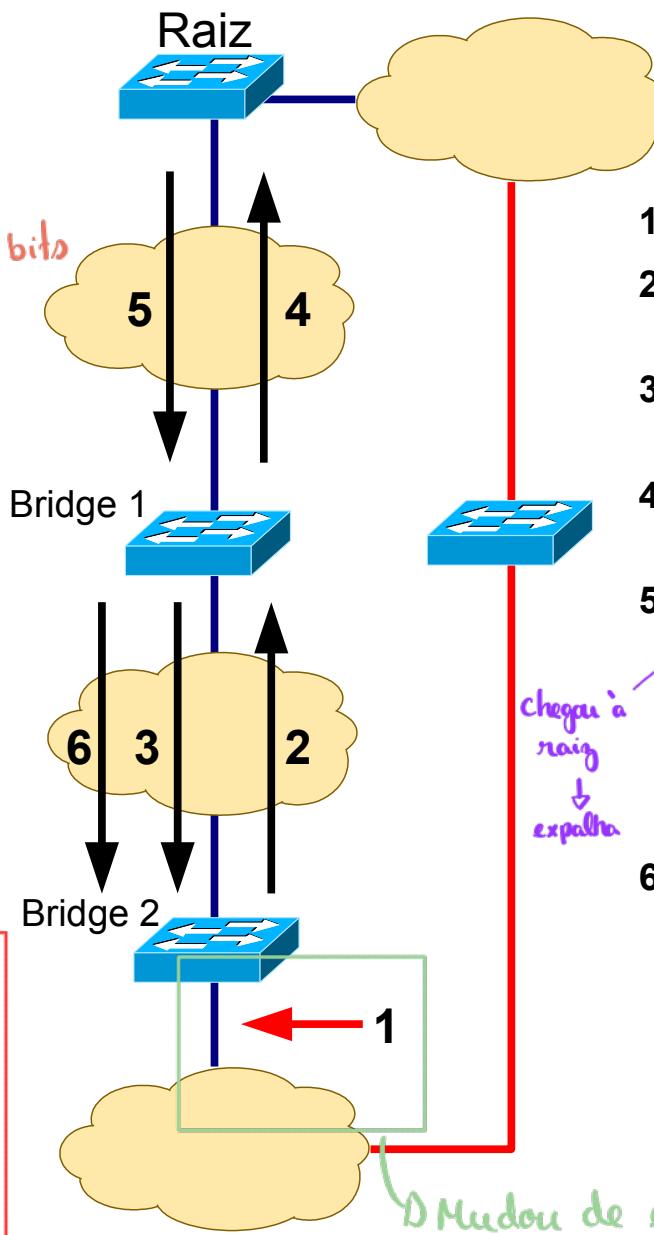
protocol identifier
version
message type = 0
TCA
reserved
TC
root ID
root path cost
bridge ID
port ID
message age
max age
hello time
forward delay

TCA - flag Topology Change Acknowledgment

TC - flag Topology Change

TCN (Topology Change Notification)  
BPDU

protocol identifier
version
message type = 1



Topology Change Notification  
→ entrou no estado Blocking  
→ avisa todos os switches para mudar o tempo de vida longo para curto (mudaram a TE)



1. Porta passa ao estado blocking
2. Envia TCN-BPDU (periodicidade = hello time) (envia)
3. Envia Conf-BPDU com TCA = 1 até deixar de receber TCN-BPDU (confirma)
4. Envia TCN-BPDU (periodicidade = hello time) (continua a enviar)
5. Envia Conf-BPDU com TCA = 1 até deixar de receber TCN-BPDU e com TC=1 durante um tempo forward delay + max age
  - Aviso para apagarem os tabelos e voltarem a reconstruir!

Bridge raiz passa a usar o tempo de vida curto nas suas tabelas de encaminhamento durante este período
6. Repete Conf-BPDU com TC=1

Bridge 1 passa a usar o tempo de vida curto nas suas tabelas até voltar a ouvir TC=0

Bridge 2 passa a usar o tempo de vida curto nas suas tabelas até voltar a ouvir TC=0

↓ Mudou de estado!  
(para blocking)

} BPDU Topology Change Notification  
→ envia para a sua porta raiz

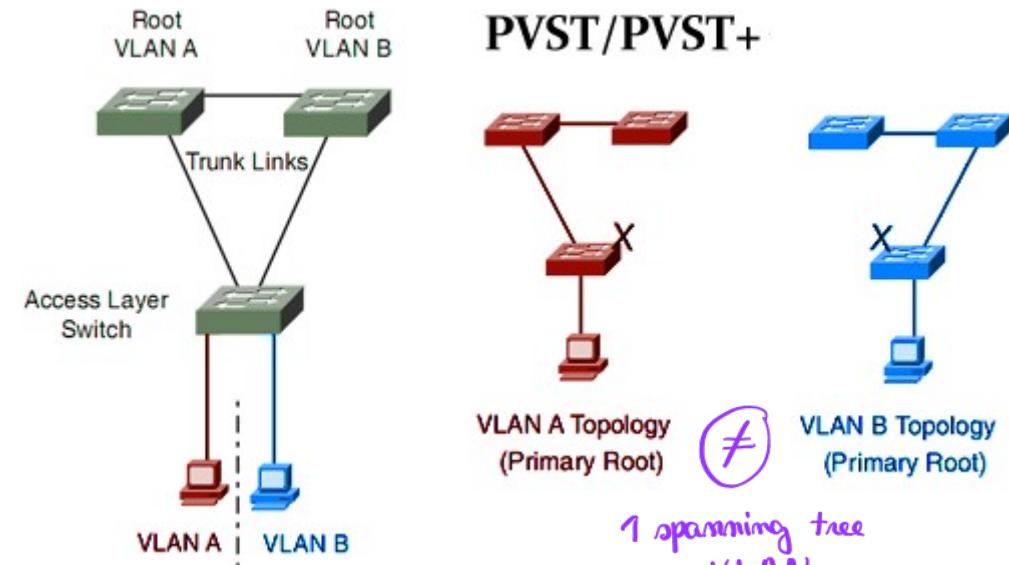
universidade de aveiro



# Outros Protocolos

(variantes mais rápidas)  
(...)

- As versões proprietárias do STP da Cisco designam-se por:
  - Per-VLAN Spanning Tree (PVST).
  - Per-VLAN Spanning Tree Plus (PVST+). *↳ pacote leva o VLAN ID!*
- Criam uma spanning tree distinta para cada VLAN.
  - Diferentes raízes, custos, portas bloqueadas, etc...
- As pacotes BPDU de diferentes VLAN são distinguidos (nos trunks):
  - PVST: usando o ISL (protocolo proprietário da Cisco equivalente ao 802.1Q).
  - PVST+: usando o 802.1Q.



Ethernet II, Src: c2:00:05:7f:f1:01 (c2:00:05:7f:f1:01), Dst: PVST+ (01:00:0c:cc:cc:cd)  
802.1Q Virtual LAN, PRI: 0, CFI: 0, ID: 1  
000. .... .... .... = Priority: 0  
...0 .... .... .... = CFI: 0  
.... 0000 0000 0001 = ID: 1

Length: 50  
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 / 0 / c2:00:05:7f:00:00  
Root Path Cost: 0  
Bridge Identifier: 32768 / 0 / c2:00:05:7f:00:00  
Port identifier: 0x802a  
Message Age: 0  
Max Age: 20  
Hello Time: 2

**Identificador da VLAN**



# Outros Protocolos

## IEEE 802.1p

- É uma extensão do IEEE 802.1Q  $\Rightarrow$  Portos InterSwitch
- Providencia qualidade de serviço com base em prioridades
- Define o campo *User Priority* (3 bits) que permite 8 níveis de prioridade.
  - Streaming > Partilha de dados
- A norma recomenda
  - Prioridade 7 : tráfego crítico da gestão de rede
  - Prioridades 5–6 : tráfego sensível ao atraso (voz e vídeo)
  - Prioridades 1–4 : tráfego sensível à variação do atraso (*streaming*)
  - Prioridade 0 : restante tráfego (transferência de ficheiros)

Prioridades entre pacotes  
ex:  
- vídeos  
- televisão



# Outros Protocolos

Mais rápida a reconfiguração!

## IEEE 802.1w Rapid Spanning Tree Protocol

- É uma extensão do IEEE 802.1D
- Acelera os tempos de convergência da Spanning Tree em caso de alteração da topologia de rede
  - Altera os estados e as funções de cada porta
  - Introduz um mecanismo de negociação entre bridges
  - Usa os bits entre as flags TCA e TC

Sem os estados intermédios  
Listening e Learning

### Conf (Configuration) BPDU

protocol identifier		
version		
message type = 0		
TCA	reserved	TC
root ID		
root path cost		
bridge ID		
port ID		
message age		
max age		
hello time		
forward delay		

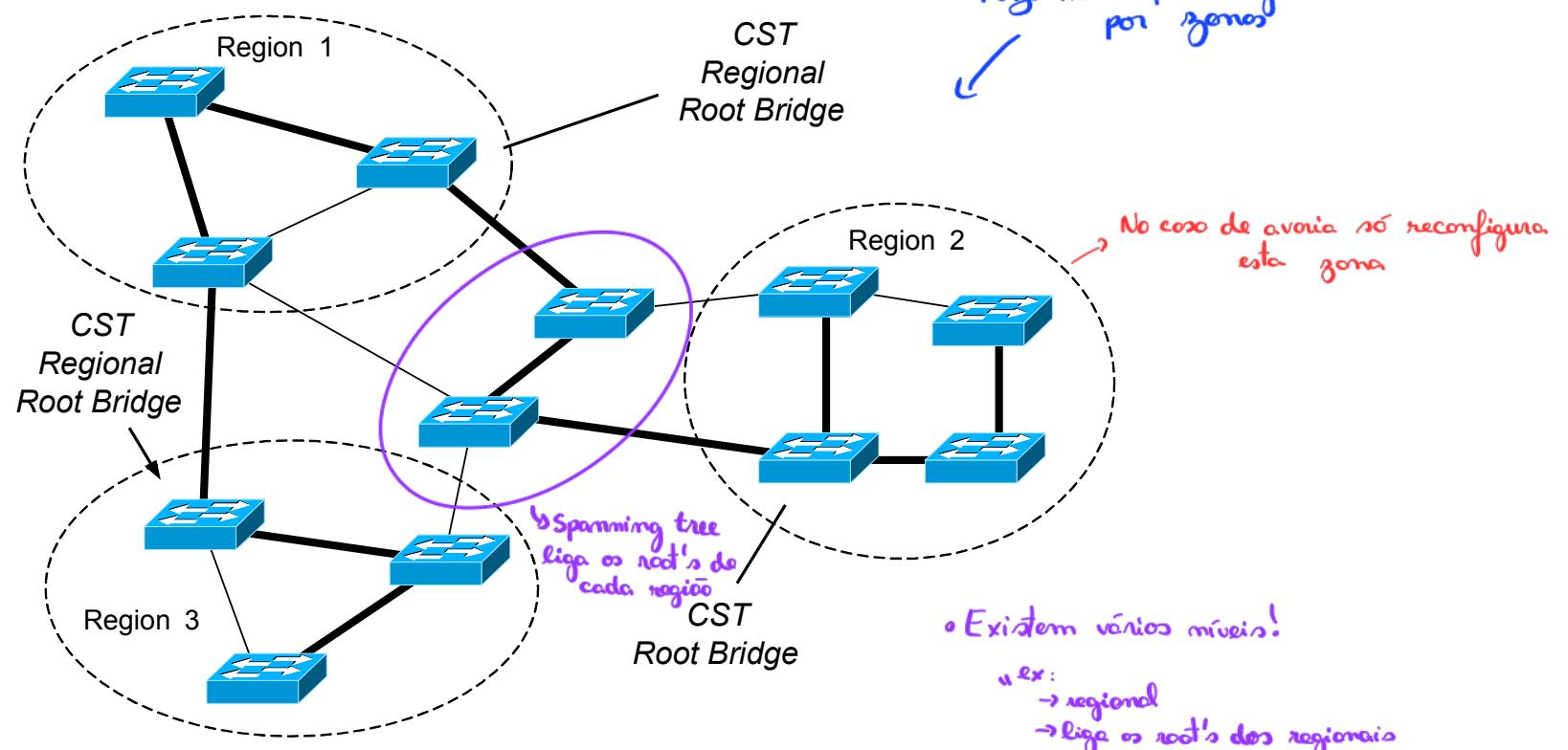


# Outros Protocolos

Para redes muito grandes!

## IEEE 802.1s Multiple Spanning Tree Protocol

- Permite criar múltiplas Spanning Trees e atribuir cada VLAN a uma das Spanning Trees criadas
- Usa os protocolos IEEE 802.1W (VLANs) e IEEE 802.1w (RSTP)
- Permite criar múltiplas regiões
  - Fora das regiões funciona como o protocolo IEEE 802.1D



# Referências

- IEEE 802.1w Rapid Spanning Tree Protocol
  - ◆ “Understanding Rapid Spanning Tree Protocol (802.1w)”
  - ◆ <http://www.cisco.com/warp/public/473/146.html>
- IEEE 802.1s Multiple Spanning Tree Protocol
  - ◆ “Understanding Multiple Spanning Tree Protocol (802.1s)”
  - ◆ <http://www.cisco.com/warp/public/473/147.html>

