

# Redes de Computadores 2

## Parte 05 – camada de enlace – comutador e VLAN

Prof. Kleber Vieira Cardoso



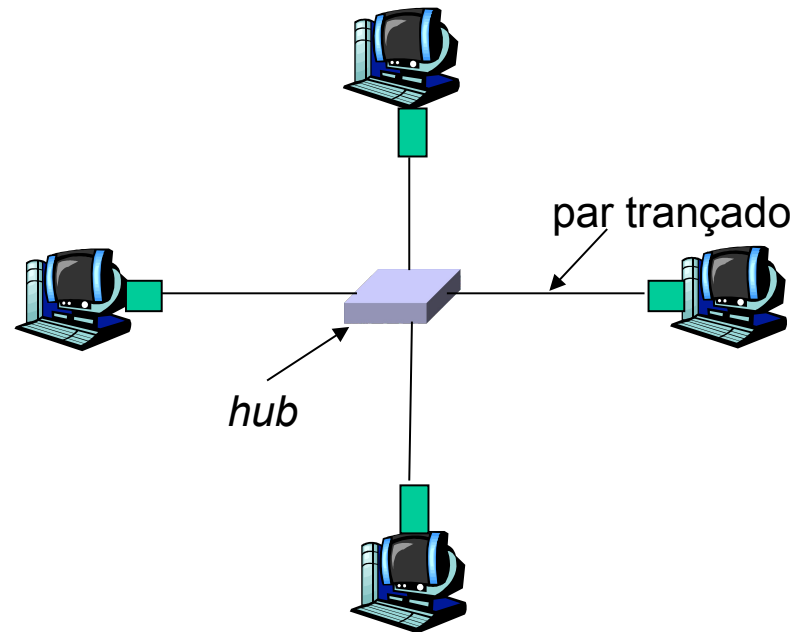
INSTITUTO DE  
INFORMÁTICA  
UFG

# Tópicos

- Computadores
  - Características
  - Tabela de comutação e autoaprendizagem
- VLAN (*Virtual Local Area Network*)
  - IEEE 802.1Q

# Hub (concentrador)

- Repetidor (“burro”) da camada física:
  - bits vindos de um dado enlace saem por *todos* os outros enlaces na mesma taxa
  - todos os nós conectados ao *hub* podem colidir uns com os outros
  - não há armazenamento temporário (“bufferização”) de quadros
  - não há CSMA/CD no *hub*: interfaces dos *hosts* detectam as colisões

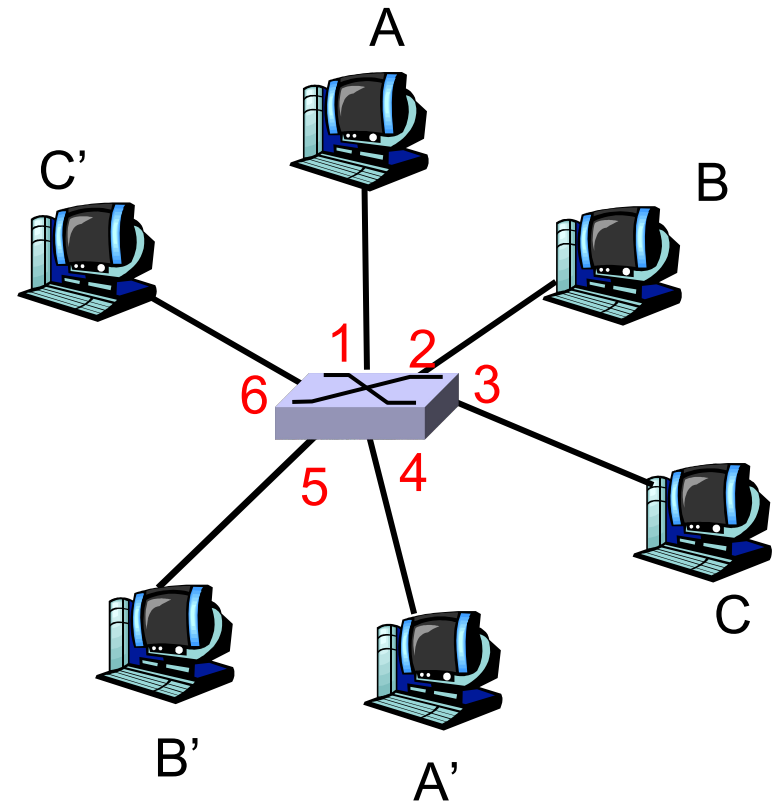


# Comutador (*switch*)

- Dispositivo da camada de enlace: mais “inteligente” que o *hub*, i.e. tem papel ativo
  - armazena e retransmite quadros Ethernet
  - examina o cabeçalho do quadro e **seletivamente** encaminha o quadro para um ou mais enlaces de saída, usa o CSMA/CD para acessar o segmento
- **transparente**
  - *hosts* ignoram a presença do *switch*
  - quadros continuam sendo trocados entre os *hosts*
- ***plug-and-play, self-learning (autodidata)***
  - o comutador não precisa ser configurado

# Switch: permite *múltiplas* transmissões simultâneas

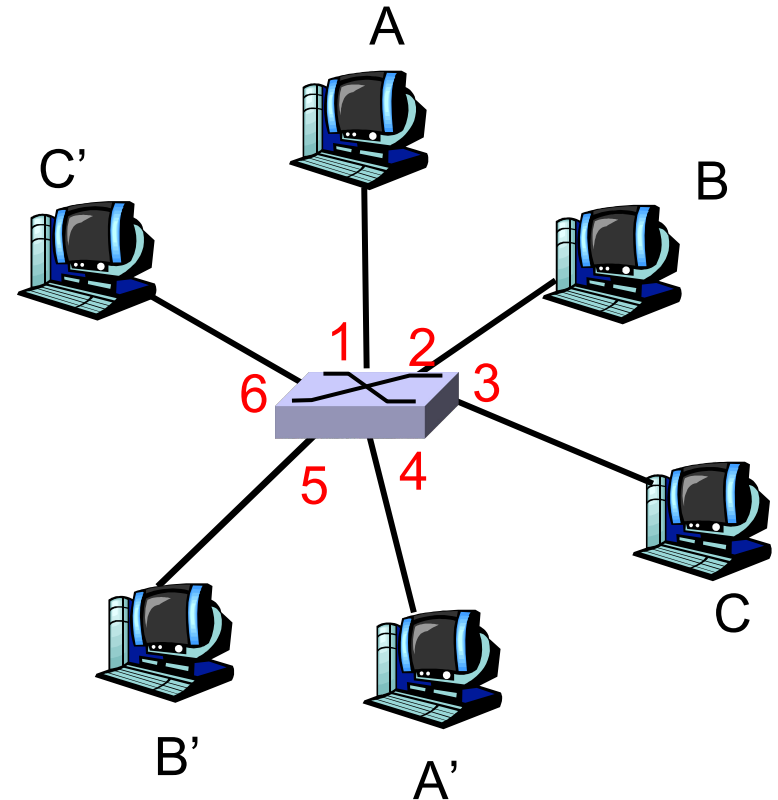
- *Hosts* têm conexão direta e dedicada para o *switch*
- Os *switches* armazenam quadros
- O protocolo Ethernet é usado em cada enlace de entrada e, em geral, não há colisões, pois são *full-duplex*
  - cada sentido do enlace é o seu próprio domínio de colisão
- **Exemplo de comutação:** A-para-A' e B-para-B' simultaneamente, sem colisões
  - isso não é possível com *hub*



*switch* com seis interfaces  
(1,2,3,4,5,6)

# Tabela de comutação

- P: como o *switch* sabe que A' é alcançável através da interface 4 e que B' é alcançável a partir da interface 5?
- R: cada *switch* possui uma **tabela de comutação** e cada entrada contém:
  - <endereço MAC do *host*, interface para alcançar o *host*, estampa de tempo>
- parece uma tabela de roteamento!
- P: como são criadas e mantidas as entradas na tabela de comutação?
  - há algo como um protocolo de roteamento?

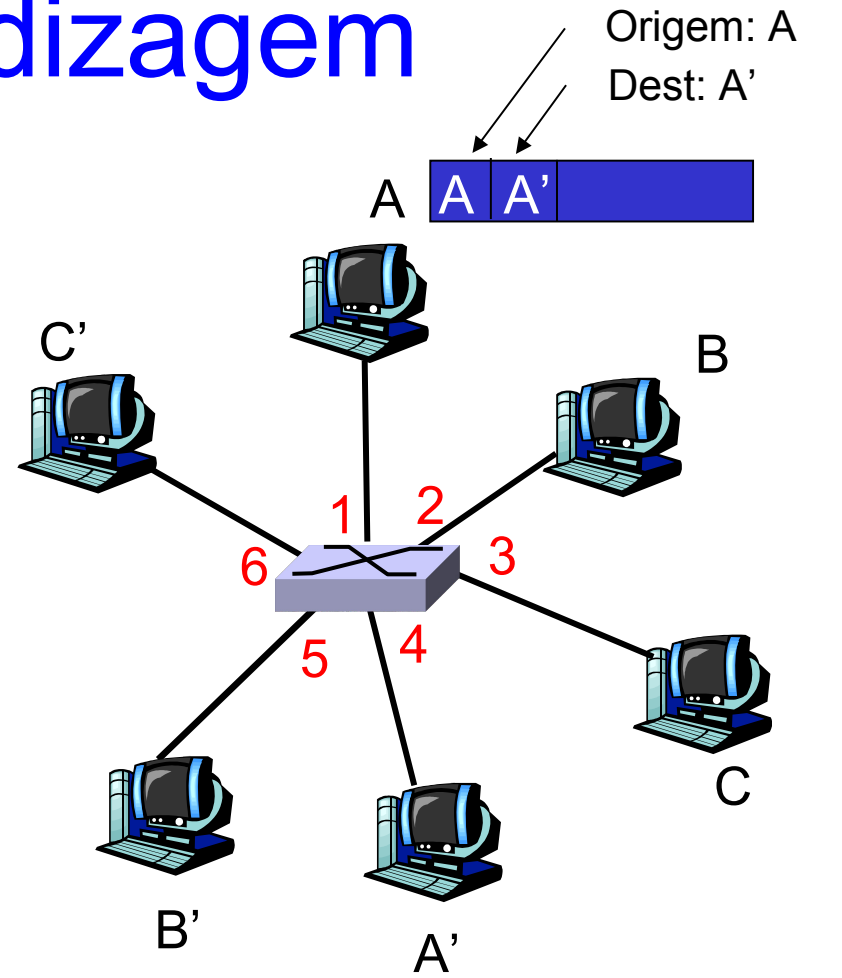


*switch* com seis interfaces  
(1,2,3,4,5,6)

# Switch: autoaprendizagem

- Switch “*aprende*” quais *hosts* podem ser alcançados através de quais interfaces
  - quando um quadro é recebido, o *switch* “aprende” a localização do transmissor: segmento de rede local de entrada
  - registra transmissor-interface-tempo na tabela de comutação

Tabela de comutação  
(inicialmente vazia)



End. MAC	interface	tempo
A	1	60

# Switch: Filtragem/Repasse de quadros

Quando um comutador recebe um quadro:

registra o enlace associado com o *host* transmissor  
procura o endereço MAC do destino na tabela de  
comutação

**se** entrada encontrada para o destino

**então {**


**se** destino estiver no segmento de onde veio o quadro

**então** descarta o quadro

**senão** repassa o quadro na interface indicada

**}**

**senão** usa inundação

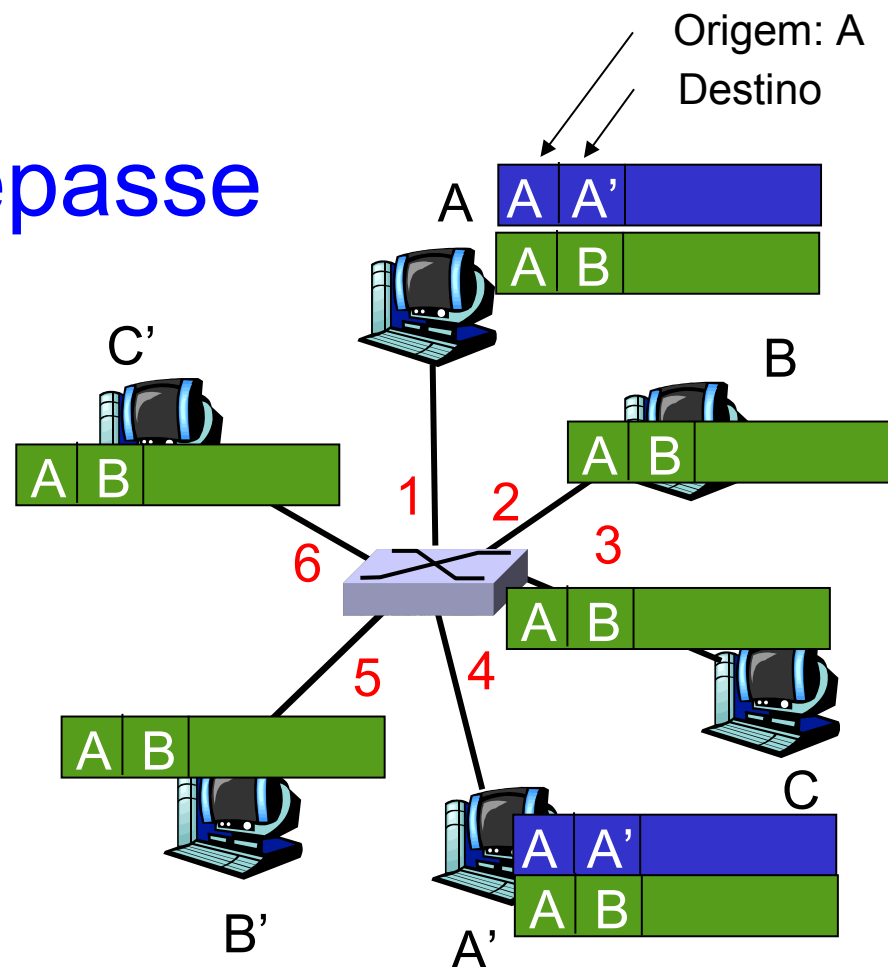


*Repassa o quadro para todas as demais  
interfaces, exceto aquela em que o  
quadro foi recebido*



# Exemplo de autoaprendizagem e repasse

- Destino B desconhecido: **inundação**
- Destino A' conhecido: **transmissão seletiva**

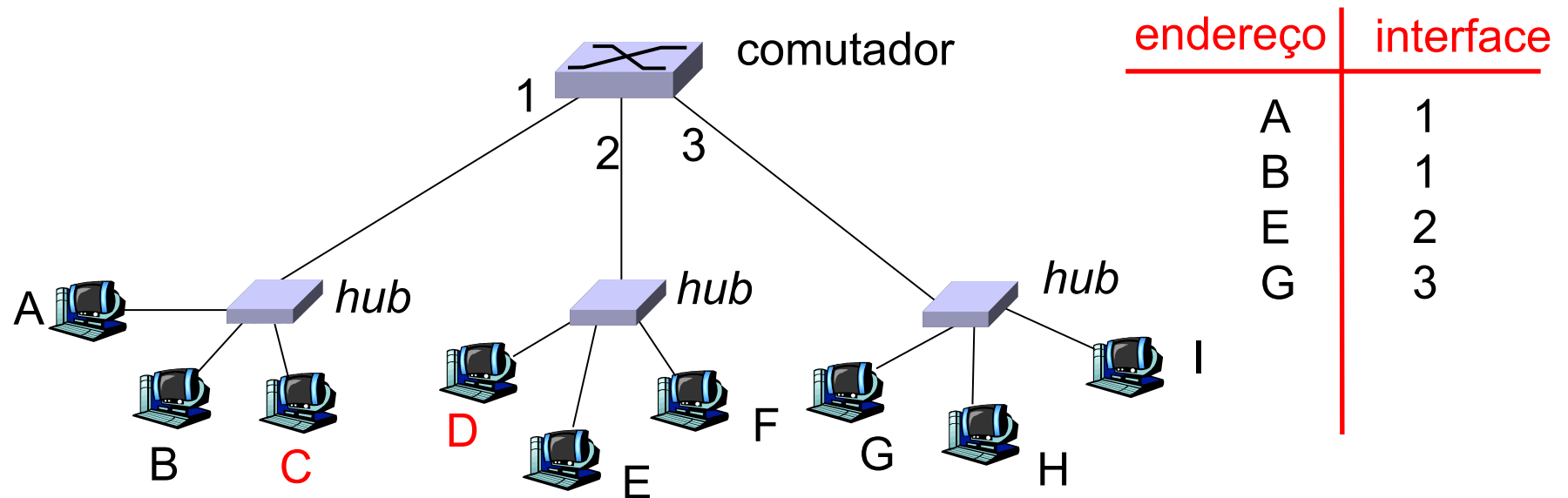


End. MAC	interface	TTL
A	1	60
A'	4	60

Tabela de comutação  
(inicialmente vazia)

# Exemplo com comutador+*hubs*

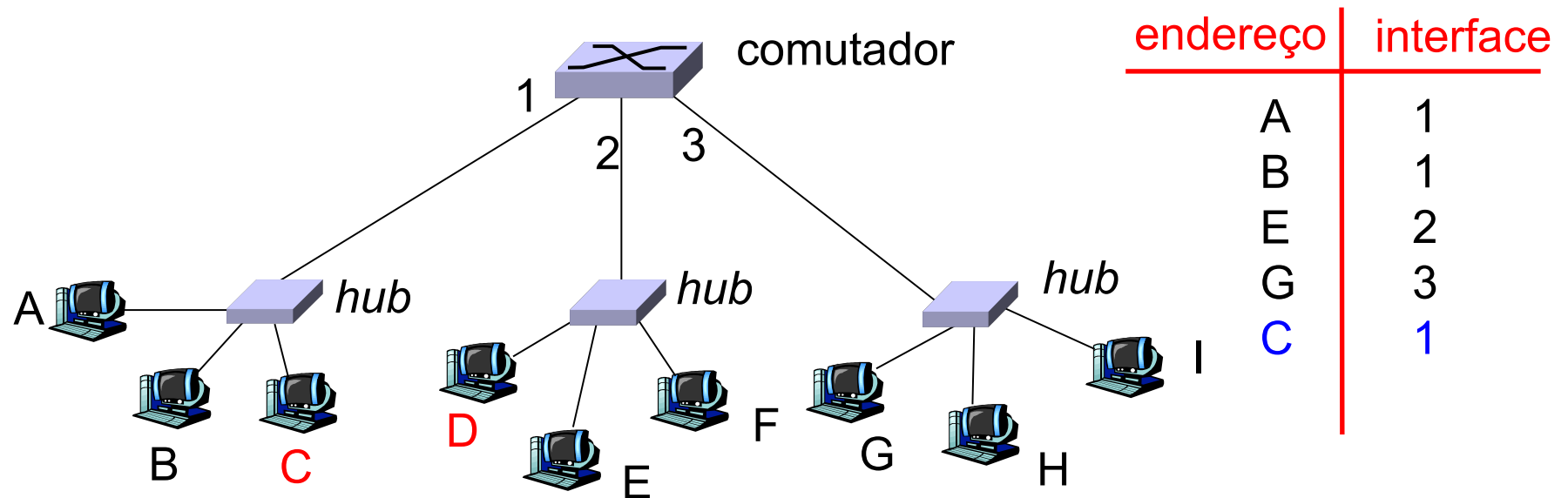
Suponha que C envia quadro para D



- Comutador recebe o quadro vindo de C
  - anota na tabela de comutação que C está na interface 1
  - supondo que D não se encontra na tabela, encaminha o quadro para as demais interfaces: 2 e 3
- Quadro é recebido por D

# Exemplo com comutador+*hubs* (cont.)

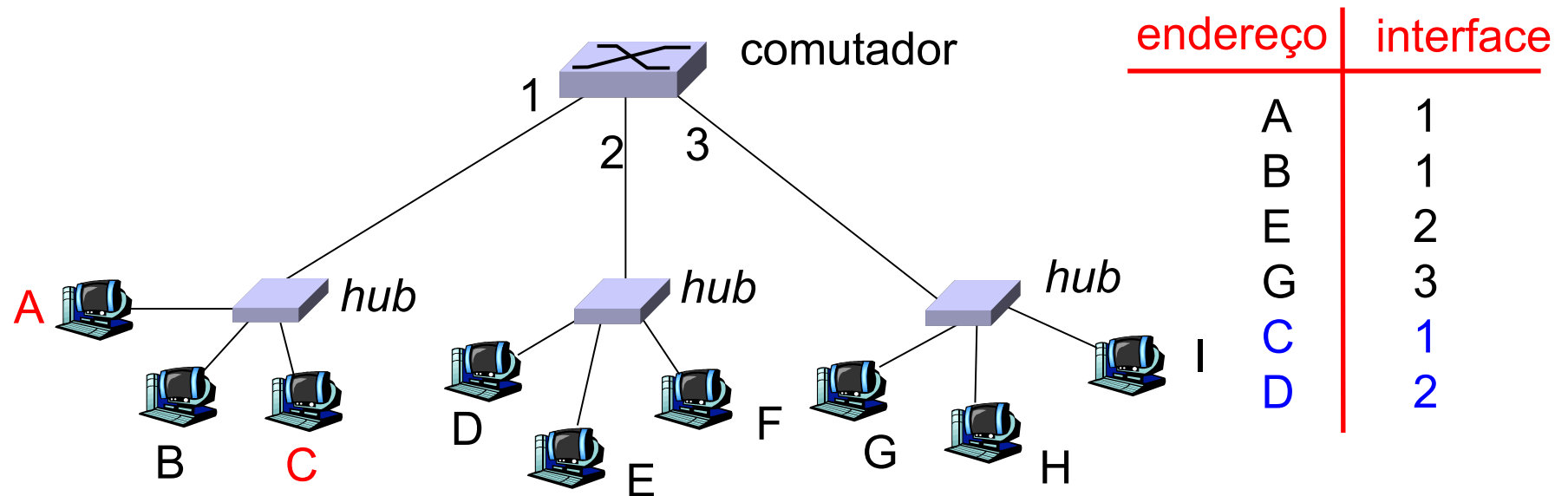
Suponha que D responde com um quadro para C



- comutador recebe o quadro vindo de D
  - anota na tabela de comutação que D está na interface 2
  - dado que C está na tabela, encaminha o quadro apenas na interface 1
- quadro é recebido por C

# Exemplo com comutador+*hubs* (cont.)

Suponha que C envia quadro para A



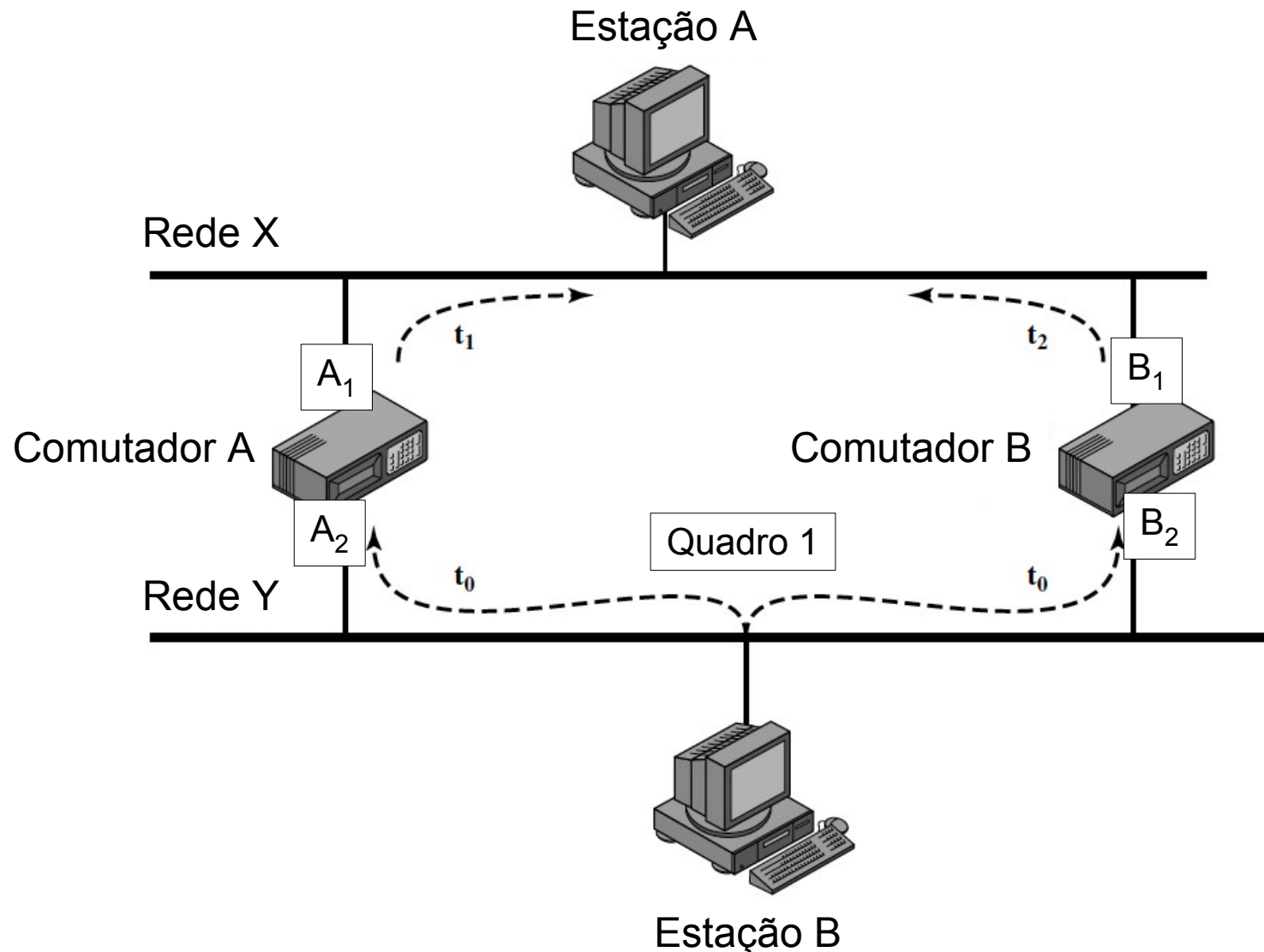
- Comutador recebe o quadro vindo de C
  - descarta, pois a interface de destino é a mesma de onde veio o quadro
- Quadro é recebido por A (se não houve colisão)

# Propriedades do comutador

- Elimina de colisões
- Integra enlace heterogêneos
- Facilita a gerência
  - Dificulta a captura de quadros na rede
  - Pode detectar uma NIC com problema e desconectá-la
  - Isola um segmento de rede com problema sem afetar o restante de rede

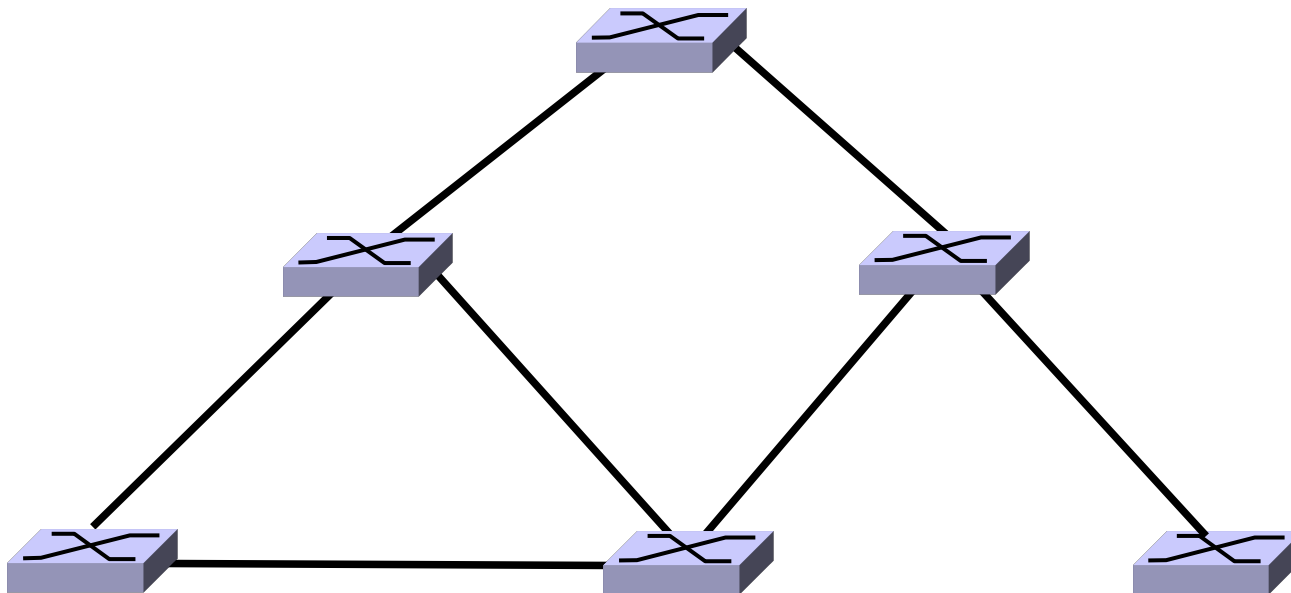
# Laços na rede

- São graves? Analise o exemplo. A qual interface cada comutador associou a estação B após o envio do Quadro 1? Assuma tabelas vazias



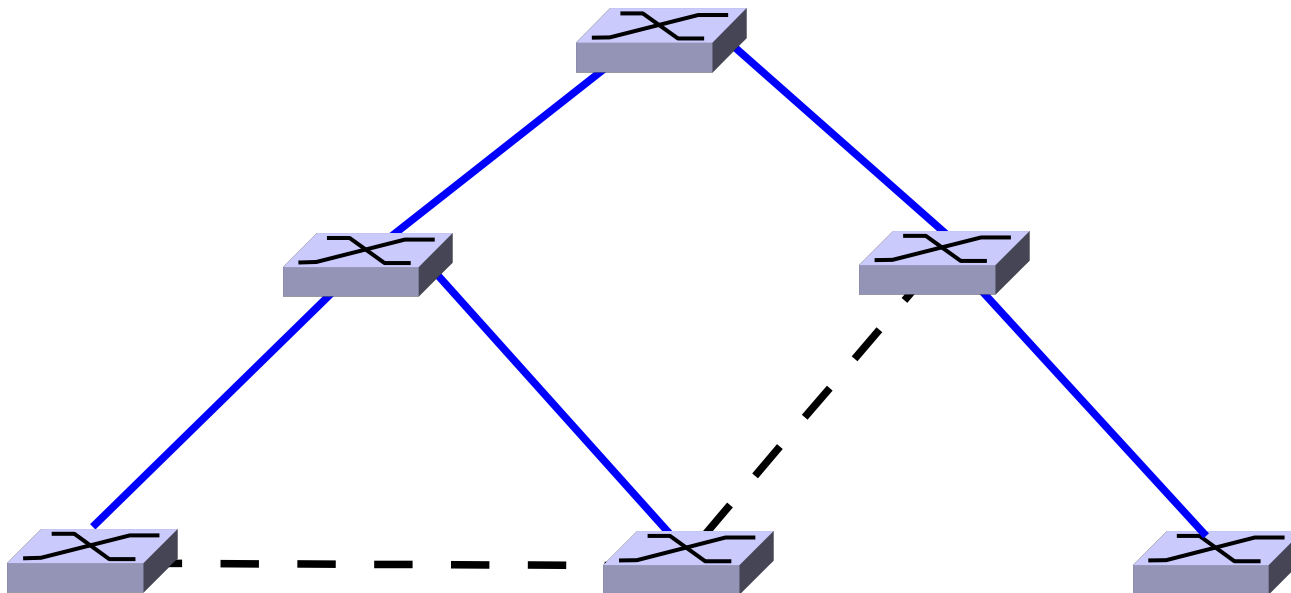
# Laços na rede (cont.)

- Como resolver?



# Laços na rede (cont.)

- Usando um algoritmo de **árvore geradora** (*spanning tree*)





# *Spanning tree*

- Algoritmo consiste de três mecanismos:
  - Aprendizado de endereços – funcionamento convencional
  - Encaminhamento de quadro – funcionamento convencional, porém com a seguinte alteração:
    - Quando o endereço de destino já está na tabela, antes de enviar é necessário verificar se a interface não está no estado bloqueado
  - Resolução de laço

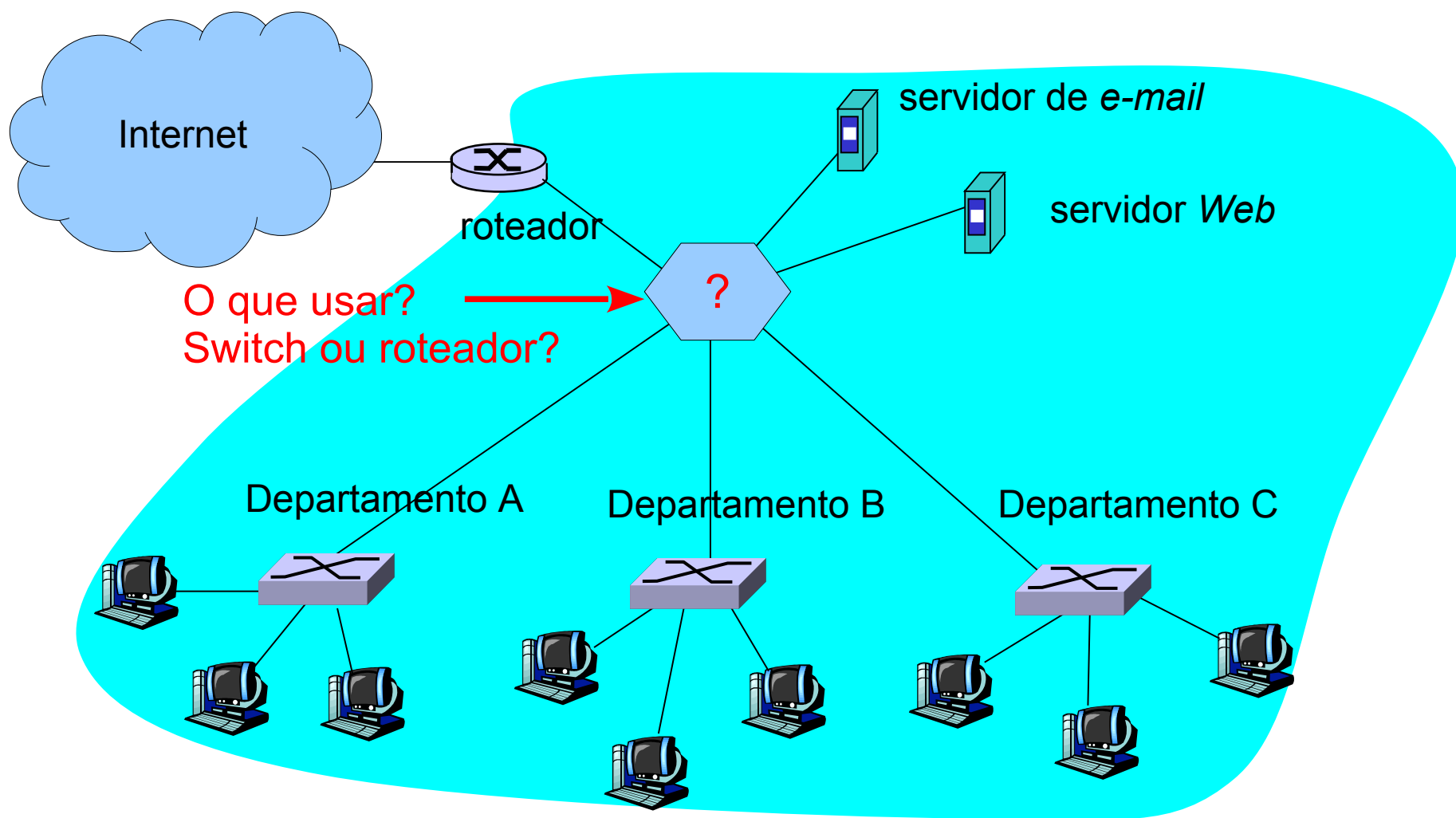
# *Spanning tree (cont.)*

- Resolução de laço:
  - Todas os comutadores divulgam seu ID único
    - O comutador com menor ID é escolhido como raiz
  - Algoritmo calcula o caminho com menor custo (ou mais curto, de acordo com alguma métrica) a partir da raiz para todos os comutadores (i.e. a *spanning tree*)
  - A árvore é difundida para todos os comutadores para identificar como as portas devem ser configuradas
    - Determinadas portas devem permanecer no estado de encaminhamento, enquanto outras devem passar para o estado bloqueado

# *Spanning tree (cont.)*

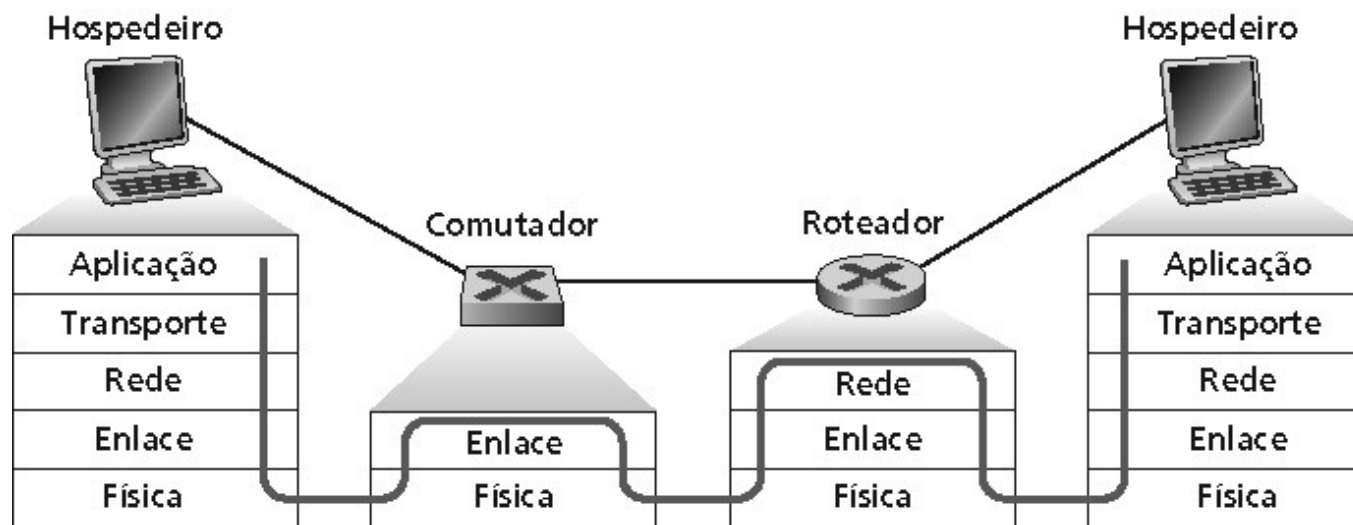
- Resolução de laço – resultado simples de teoria dos grafos:
  - Para cada grafo conectado, composto de vértices e arestas, as quais conectam pares de vértices, há uma árvore geradora que mantém o grafo conexo, mas não contém laços
- IEEE 802.1 – padrão para: LAN/MAN, interligação de LANs/MANs e correlatos (segurança, gerência, etc.)
  - Define o STP (*Spanning Tree Protocol*) e RSTP (*Rapid Spanning Tree Protocol*)
    - Cada comutador (ou ponte) deve ter um identificador único e um custo associado a cada porta (exemplo, pode ser o mesmo custo ou com base na velocidade)
    - Um comutador é escolhido como raiz e uma árvore geradora mínima é encontrada
    - Sempre que há uma mudança na topologia, os comutadores recalculam a árvore geradora

# Rede Institucional/corporativa



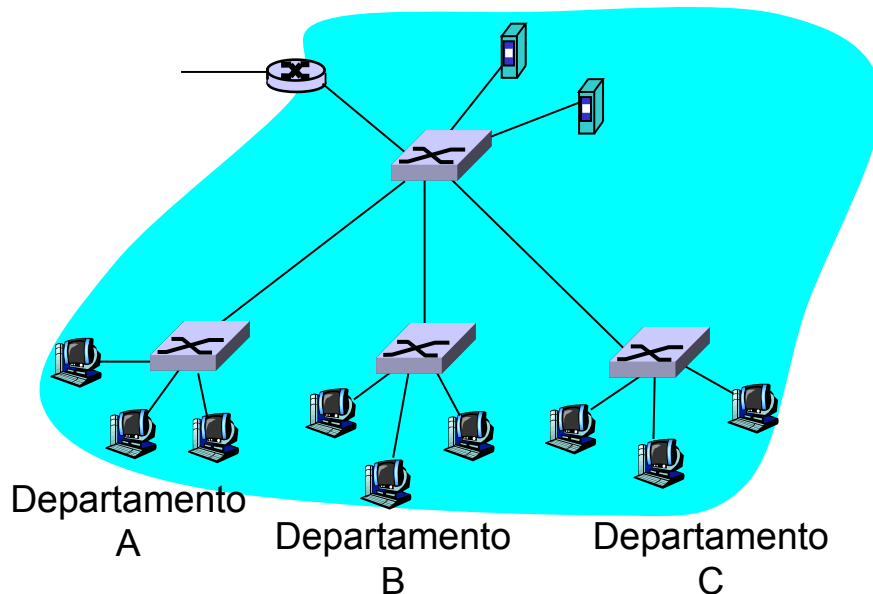
# Comutadores vs. Roteadores

- Ambos são dispositivos do tipo armazena-e-repassa
  - Roteadores: dispositivos da camada de rede (examinam os cabeçalhos da camada de rede) – separam domínios de difusão
  - Comutadores: dispositivos da camada de enlace – separam domínio de colisão (ex.: solicitação ARP inunda a rede)
- Roteadores mantêm tabelas de roteamento, implementam algoritmos de roteamento
- Comutadores mantêm tabelas de comutação, implementam filtragem, algoritmos de aprendizagem



# VLANs: motivação

*Quais as desvantagens dessa configuração?*



- Único domínio de difusão
  - todo tráfego de difusão da camada 2 (ARP, DHCP, etc.) passa por toda a rede (questões de eficiência, segurança/privacidade)
- Uso ineficiente dos comutadores
  - Exemplo: comutador com 12 portas, mas em uso apenas 3 (na figura)

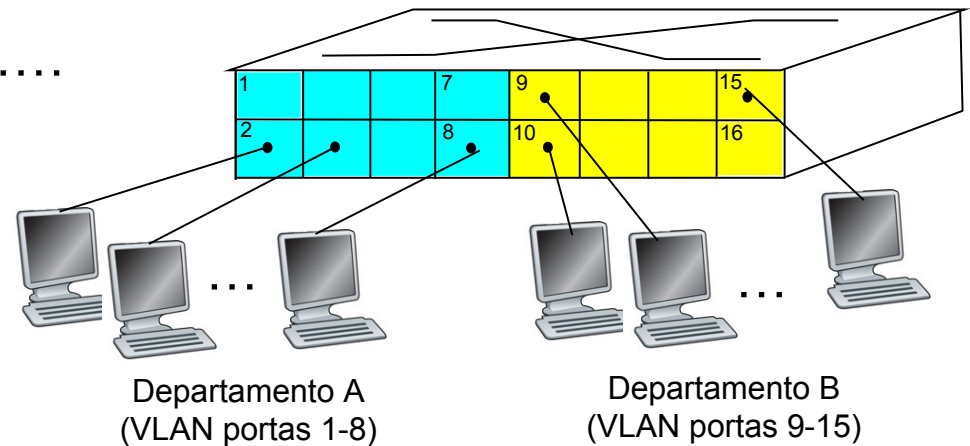
# VLANs

VLAN (baseada em porta):

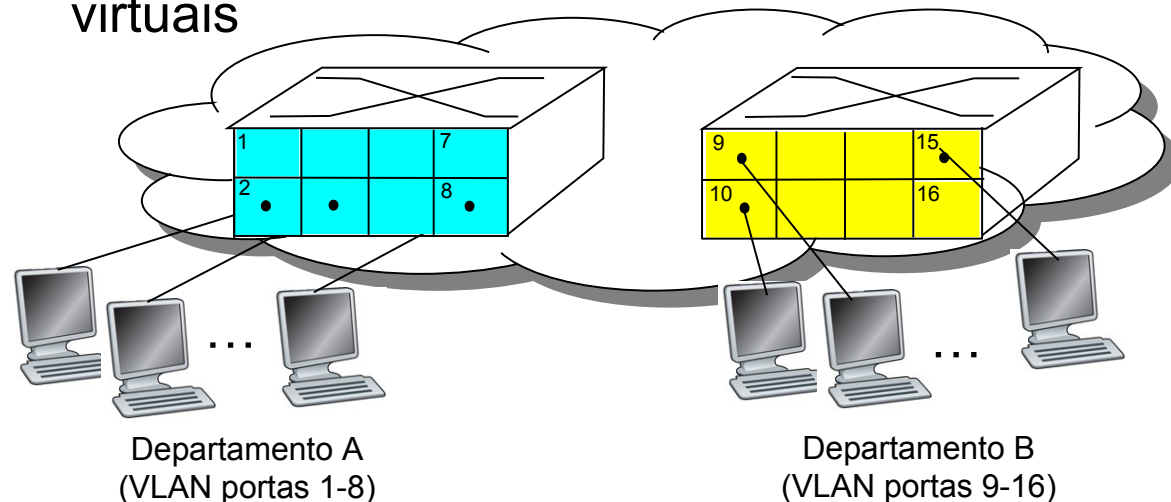
portas de comutador agrupadas  
(por software de gerenciamento do  
comutador) para que *único*  
comutador físico .....

## *Virtual Local Area Network*

Comutadores com  
capacidade de VLAN  
podem ser configurados  
para definir múltiplas  
redes locais *virtuais*  
usando uma única  
infraestrutura de rede  
física

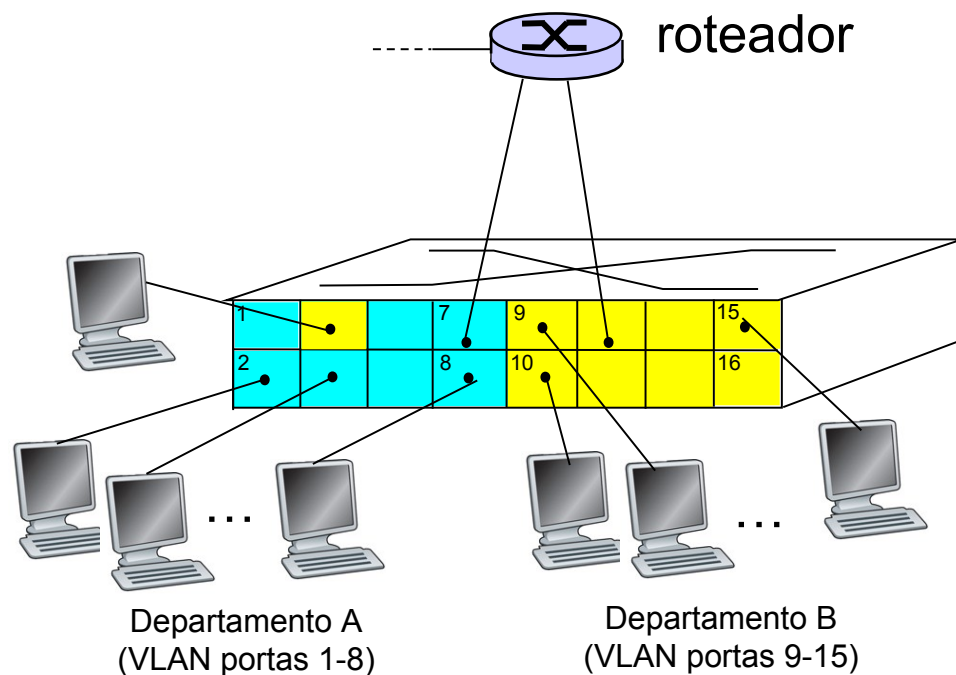


... opere como *múltiplos* comutadores  
virtuais



# VLAN baseada em porta

- **Isolamento de tráfego:** quadros de/para portas 1-8 só podem alcançar portas 1-8
  - É possível definir VLAN com base em endereços MAC das extremidades ou outros critérios (tipo de camada de rede, endereço de rede, grupo *multicast*, política, etc.), porém exige equipamentos mais sofisticados => mais caros

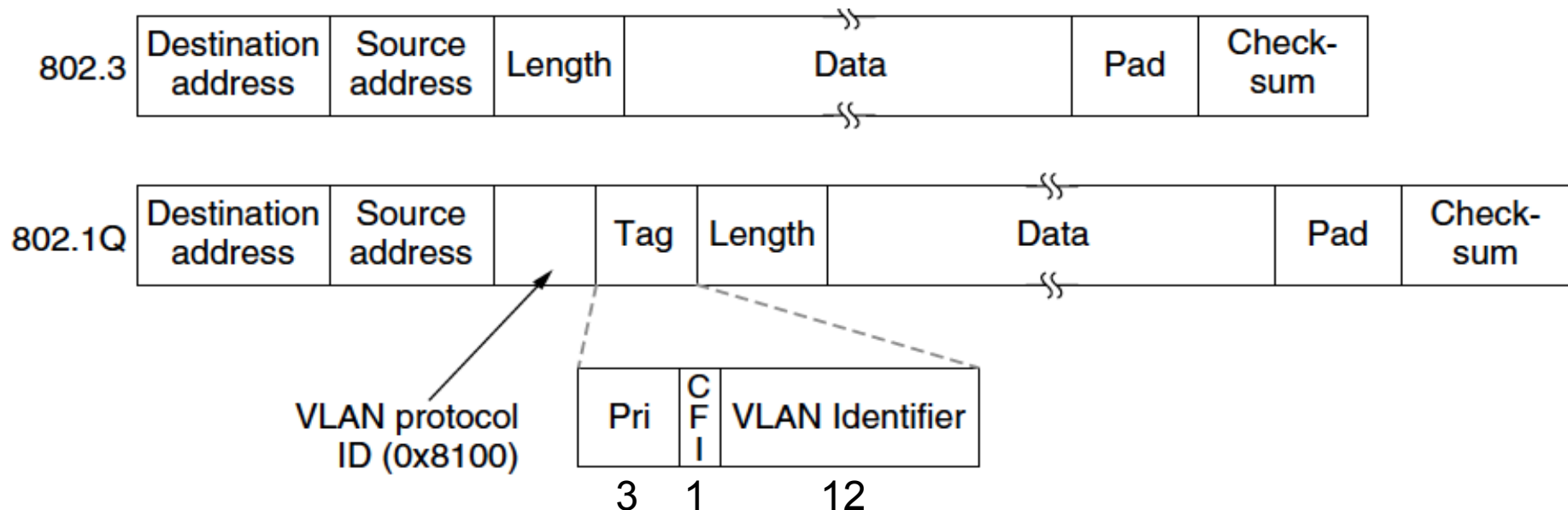


- **Reencaminhamento entre VLANs:** feito por roteamento (assim como em comutadores separados)
  - Na prática, é comum encontrar equipamentos que são uma combinação de comutador e roteador (comutador de nível 3 – *layer-3 switch*)



# IEEE 802.1Q

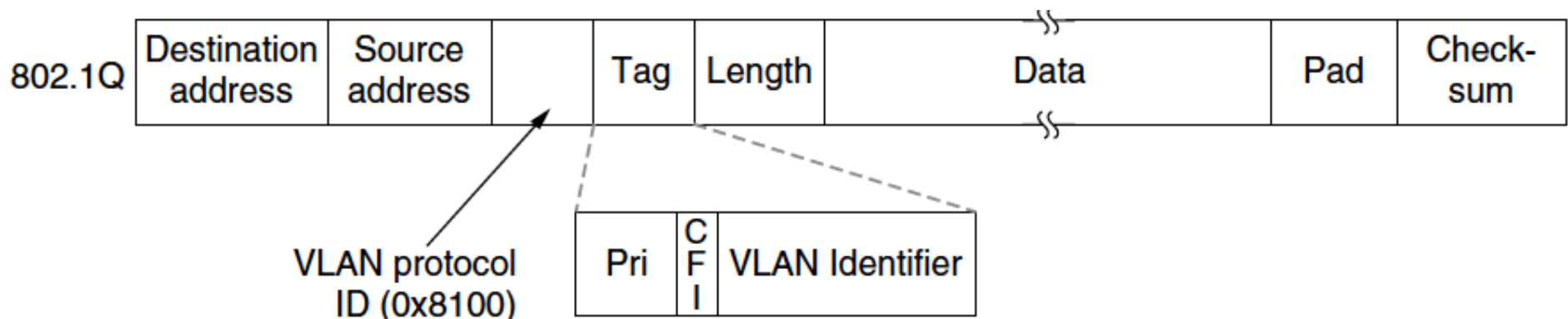
- Introduz um rótulo de VLAN no cabeçalho Ethernet



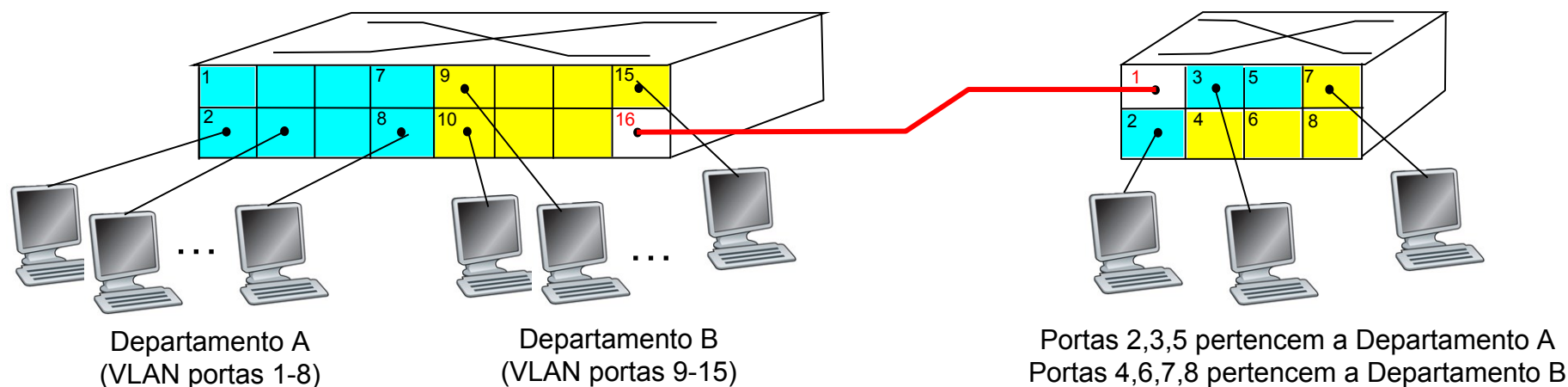
- Dúvidas:** como lidar com o novo cabeçalho? Descartar NICs antigas? Como tratar quadros que já possuem tamanho máximo?
- Solução:** apenas comutadores cientes de VLAN precisam tratar o novo cabeçalho. Antes de enviar um quadro para NIC ou comutador legado, retirar o rótulo de VLAN

# IEEE 802.1Q (cont.)

- Todos os quadros com VLAN usam o identificador de protocolo VLAN (0x8100)
- Os 3 bits de prioridade são definidos por outro padrão: IEEE 802.1p
- O campo de identificação de VLAN é o que permite associar cada quadro a uma VLAN



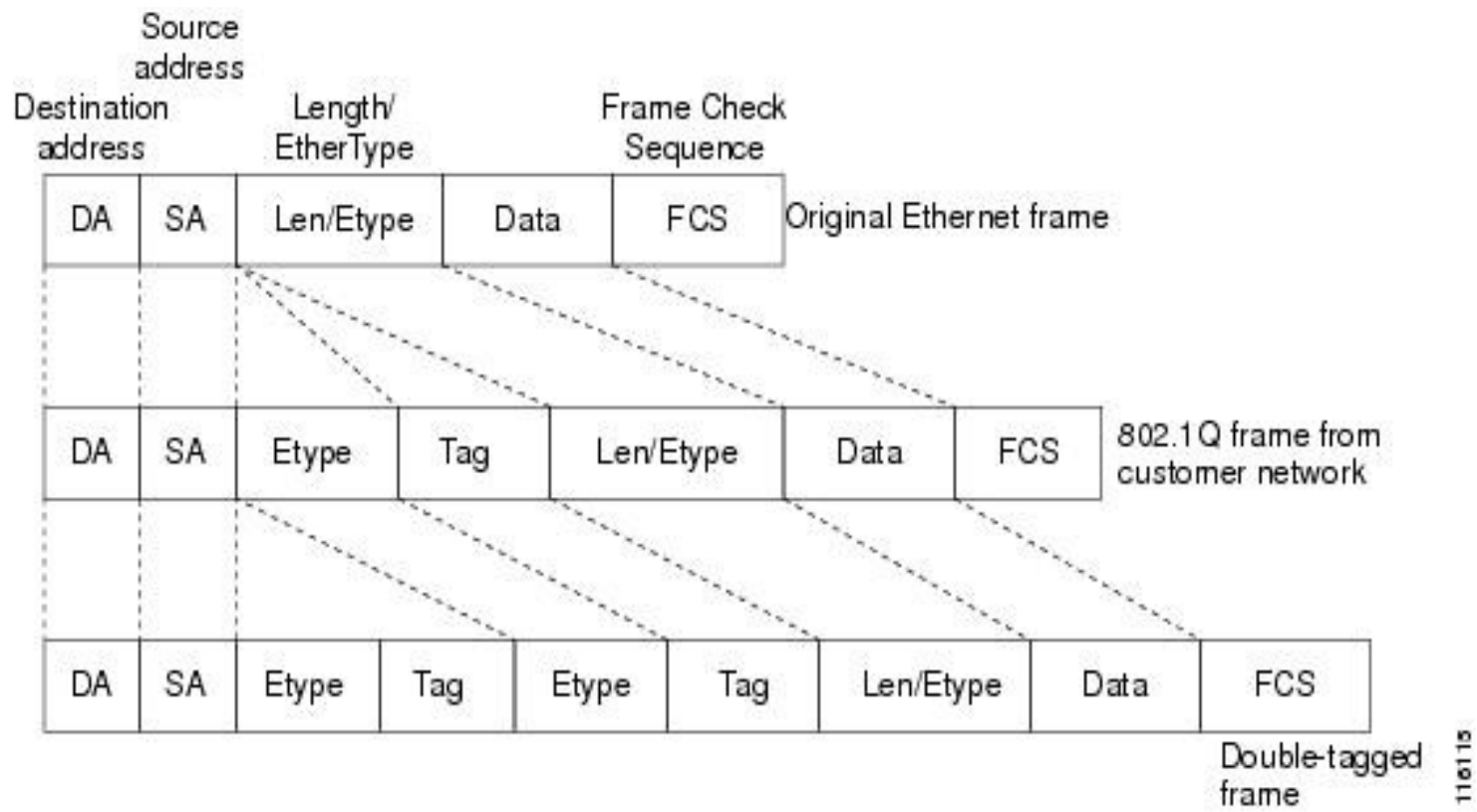
# VLANs sobre múltiplos comutadores



- **Porta de tronco:** carrega quadros entre VLANs definidas sobre vários comutadores físicos
  - Oferece grande flexibilidade
  - No entanto, quadros repassados dentro da VLAN entre comutadores não podem ser quadros comuns, precisam de identificador de VLAN ID

# Q-in-Q – IEEE 802.1d

- Permite transportar VLANs sobre VLANs
- Amplia o número de redes virtuais: 4096x4096
- Útil em redes metropolitanas para encapsular tráfego de diferentes clientes



# Q-in-Q – IEEE 802.1d (cont.)

- Exemplo

