

Capítulo 5: A Camada de Enlace

Nossos objetivos:

- Entender os princípios por trás dos serviços da camada de enlace de dados:
 - detecção e correção de erros
 - compartilhamento de canal de *broadcast*: acesso múltiplo
 - endereçamento da camada de enlace
 - redes locais (LANs): Ethernet, VLANs
- instanciação e implementação de diversas tecnologias de camada de enlace

Camada de Enlace

5.1 Introdução e serviços

5.2 Detecção e correção de erros

5.3 Protocolos de acesso múltiplo

5.4 Redes Locais

- Endereçamento, ARP

- Ethernet

- Switches

- VLANs

5.5 Virtualização do enlace: MPLS

5.6 Redes de centros de dados

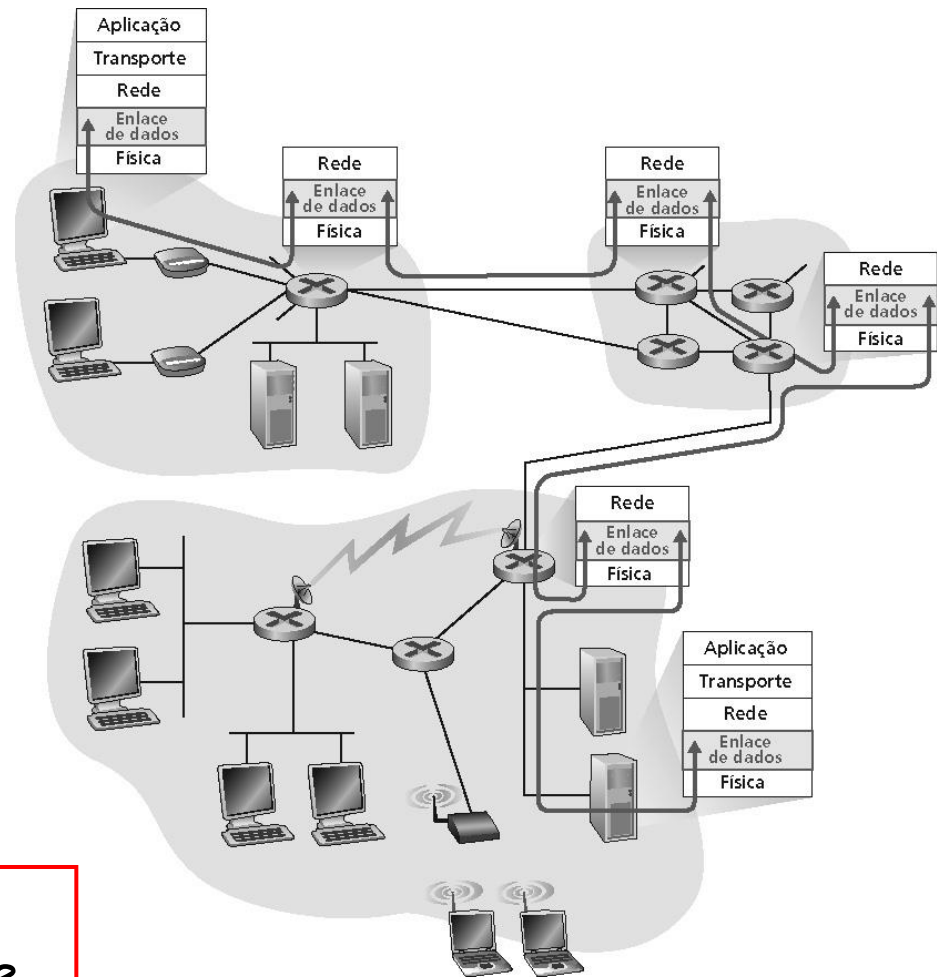
5.7 Um dia na vida de uma solicitação de página Web

Camada de Enlace: Introdução

Terminologia:

- ☐ hospedeiros e roteadores são **nós**
- ☐ canais de comunicação que conectam nós adjacentes ao longo de um caminho de comunicação são **enlaces** (*links*)
 - enlaces com fio (cabeados)
 - enlaces sem fio (não cabeados)
 - LANs
- ☐ Pacote da camada 2 é um **quadro** (*frame*), encapsula o datagrama

a camada de enlace é responsável por transferir os datagramas entre nós fisicamente adjacentes através de um enlace



Camada de Enlace: Contexto

- um datagrama é transferido por diferentes protocolos de enlace em diferentes enlaces:
 - Ex.: Ethernet no primeiro enlace, frame relay em enlaces intermediários e 802.11 no último enlace
- cada protocolo de enlace provê diferentes serviços
 - ex.: pode ou não prover transporte confiável de dados através do enlace

Analogia com um sistema de transporte

- Viagem de Princeton até Lausanne
 - taxi: Princeton até JFK
 - avião: JFK até Genebra
 - Trem: Genebra até Lausanne
- turista = **datagrama**
- segmento de transporte = **enlace de comunicação**
- modo de transporte = **protocolo da camada de enlace**
- agente de viagens = **algoritmo de roteamento**

Serviços da Camada de Enlace

□ enquadramento (delimitação do quadro) e acesso ao enlace:

- encapsula datagrama num quadro adicionando cabeçalho e cauda (*trailer*).
- implementa acesso ao canal se meio for compartilhado,
- 'endereços físicos (MAC)' são usados nos cabeçalhos dos quadros para identificar origem e destino de quadros em enlaces multiponto
 - Diferente do endereço IP!

□ entrega confiável entre nós adjacentes:

- já aprendemos como fazer isto (Capítulo 3)
- raramente usada em canais com baixas taxas de erro (fibra óptica, alguns tipos de pares trançados)
- Canais sem fio: altas taxas de erros
 - P: para que confiabilidade na camada de enlace e fim-a-fim?

Serviços da Camada de Enlace (mais)

□ *controle de Fluxo:*

- compatibilizar taxas de produção e consumo de quadros entre remetentes e receptores

□ *detecção de Erros:*

- erros são causados por atenuação do sinal e por ruído
- receptor detecta presença de erros
 - receptor sinaliza ao remetente para retransmissão, ou simplesmente descarta o quadro em erro

□ *correção de Erros:*

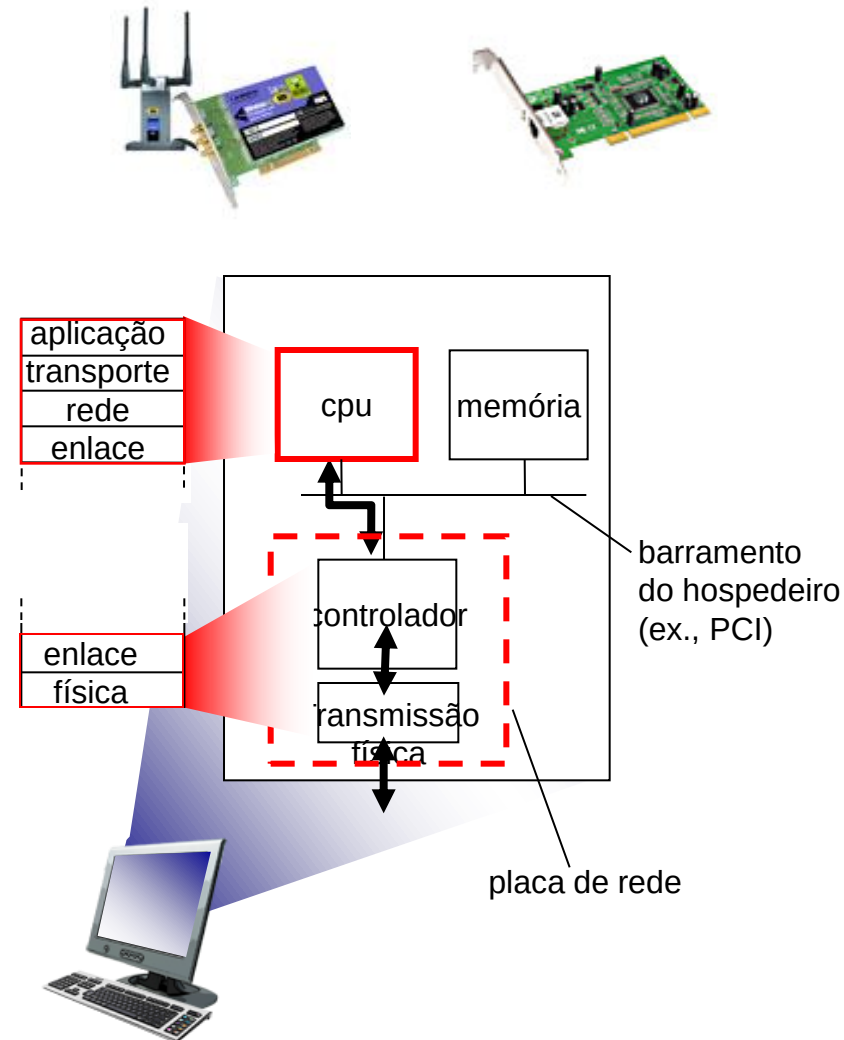
- mecanismo que permite que o receptor localize *e corrija* o(s) erro(s) sem precisar da retransmissão

□ *half-duplex e full-duplex*

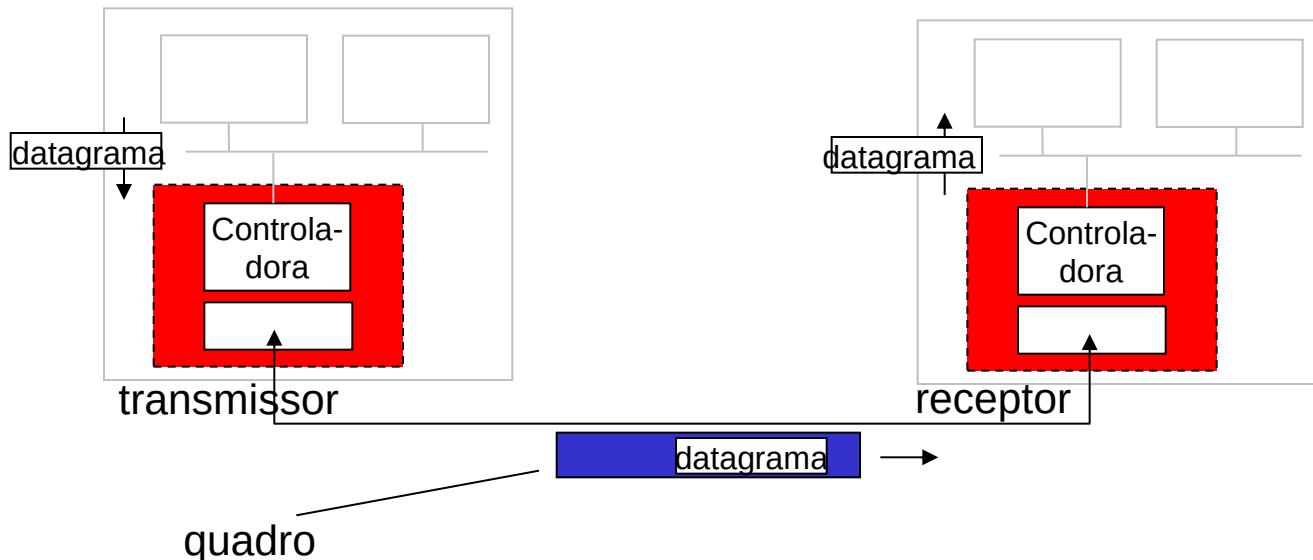
- com half duplex um nó não pode transmitir e receber pacotes ao mesmo tempo

Onde a camada de enlace é implementada?

- ❑ em cada um dos hospedeiros
- ❑ camada de enlace implementada no "adaptador" (NIC - *Network Interface Card*)
 - placa Ethernet, placa 802.11; chipset Ethernet
 - implementa as camadas de enlace e física
- ❑ conecta ao barramento do sistema hospedeiro
- ❑ combinação de hardware, software e firmware



Comunicação entre Adaptadores



❑ lado transmissor:

- Encapsula o datagrama em um quadro
- Adiciona bits de verificação de erro, transferência confiável de dados, controle de fluxo, etc.

❑ lado receptor

- verifica erros, transporte confiável, controle de fluxo, etc.
- extrai o datagrama, passa-o para o nó receptor

Camada de Enlace

5.1 Introdução e serviços

5.2 Detecção e correção de erros

5.3 Protocolos de acesso múltiplo

5.4 Redes Locais

- Endereçamento, ARP

- Ethernet

- Switches

- VLANs

5.5 Virtualização do enlace: MPLS

5.6 Redes de centros de dados

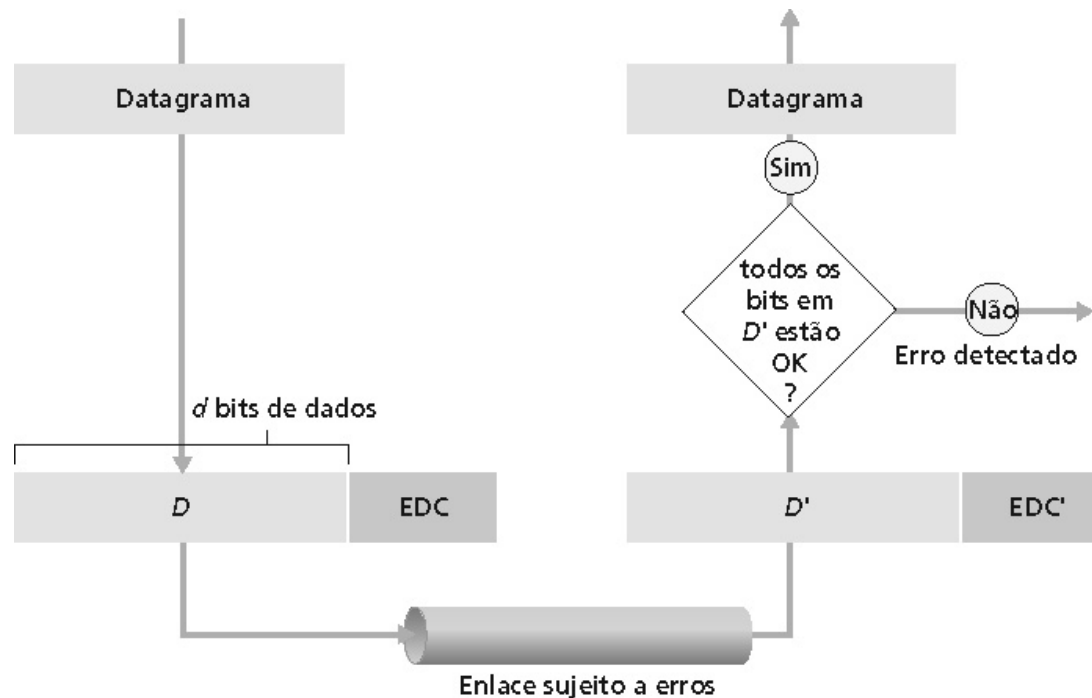
5.7 Um dia na vida de uma solicitação de página Web

Detecção de Erros

EDC= bits de Detecção e Correção de Erros (redundância)

D = Dados protegidos por verificação de erros,
podem incluir alguns campos do cabeçalho

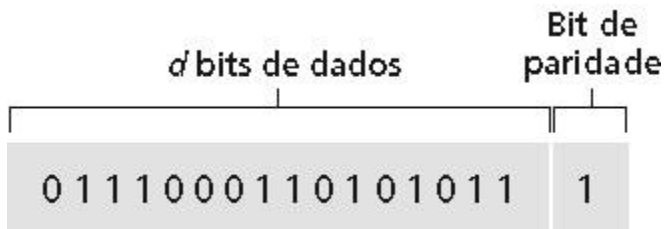
- a detecção de erros não é 100% confiável;
 - protocolos podem deixar passar alguns erros, mas é raro
 - quanto maior for o campo EDC, melhor será a capacidade de detecção e correção de erros



Verificações de Paridade

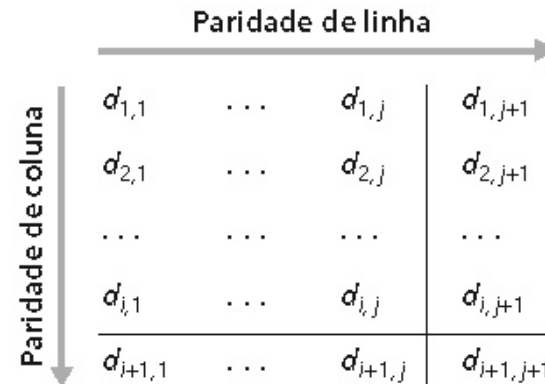
Paridade de 1 Bit:

Detecta erros em um único bit



Paridade Bidimensional:

Detecta e corrige erro em um único bit



Nenhum erro

1	0	1	0	1	1
1	1	1	1	0	0
0	1	1	1	0	1
0	0	1	0	1	0

Erro de bit
único corrigível

1	0	1	0	1	1
1	0	1	1	0	0
0	1	1	1	0	1
0	0	1	0	1	0

Erro de
paridade

Erro de
paridade

Soma de verificação da Internet

Objetivo: detectar "erros" (ex., bits trocados) no segmento transmitido (nota: usado apenas na camada de transporte)

Transmissor:

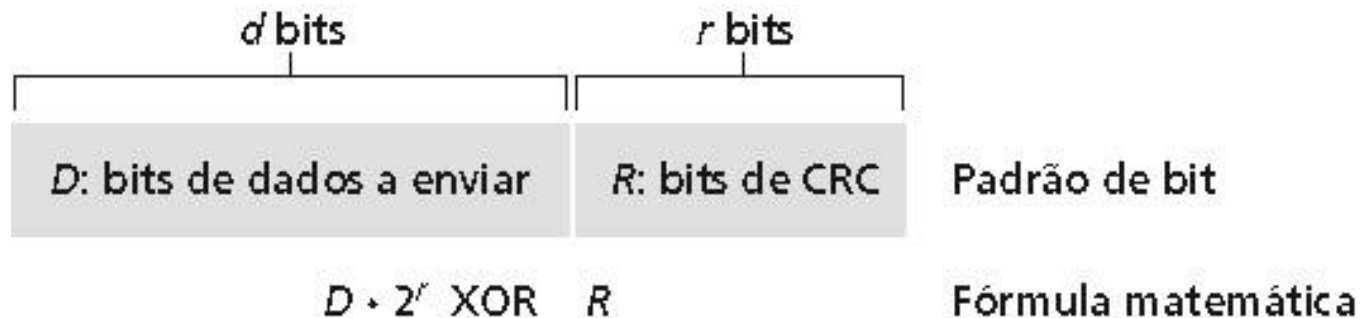
- ☐ trata o conteúdo do segmento como uma sequência de inteiros de 16 bits
- ☐ Soma de verificação: adição (complemento de 1 da soma) do conteúdo do segmento
- ☐ transmissor coloca o valor do *checksum* no campo de *checksum* UDP

Receptor:

- ☐ calcula a soma de verificação do segmento recebido
- ☐ verifica se o resultado é igual ao valor do campo da soma de verificação:
 - NÃO - erro detectado
 - SIM - nenhum erro foi detectado. *Mas ainda pode conter erros?*

Verificação de redundância cíclica (CRC)

- ❑ codificação para detecção de erro mais poderosa
- ❑ vê os bits de dados, **D**, como um número binário
- ❑ é escolhido um polinômio Gerador, (**G**), ($\Rightarrow r+1$ bits)
- ❑ objetivo: escolher r bits CRC, **R**, de modo que
 - $\langle D, R \rangle$ seja exatamente divisível por G (módulo 2)
 - receptor conhece G , divide $\langle D, R \rangle$ por G . Caso o resto seja diferente de zero: detectado erro!
 - Pode detectar todos os erros em rajadas menores do que $r+1$ bits
- ❑ Largamente usado na prática (Ethernet, 802.11 WiFi, ATM)



Exemplo de CRC

Queremos:

$$D \cdot 2^r \text{ XOR } R = nG$$

de forma equivalente:

$$D \cdot 2^r = nG \text{ XOR } R$$

de forma equivalente :

se dividirmos $D \cdot 2^r$ por G , queremos o resto R que satisfaça:

$$R = \text{resto} \left(\frac{D \cdot 2^r}{G} \right)$$

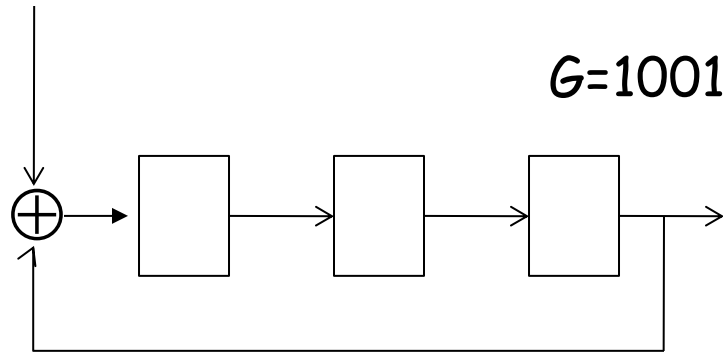
$$r = 3$$

Handwritten binary division showing the calculation of the CRC remainder R . The dividend D is 101110000 and the divisor G is 1001. The remainder R is 011.

$$\begin{array}{r} \overline{101110000} \\ 1001 \\ \hline 1010 \\ 1001 \\ \hline 1100 \\ 1001 \\ \hline 1010 \\ 1001 \\ \hline 011 \\ \hline \end{array}$$

Implementação em Hardware

$D \cdot 2^r = 101110000$



Camada de Enlace

5.1 Introdução e serviços

5.2 Detecção e correção de erros

5.3 Protocolos de acesso múltiplo

5.4 Redes Locais

- Endereçamento, ARP

- Ethernet

- Switches

- VLANs

5.5 Virtualização do enlace: MPLS

5.6 Redes de centros de dados

5.7 Um dia na vida de uma solicitação de página Web

Enlaces e Protocolos de Acesso Múltiplo

Dois tipos de enlaces:

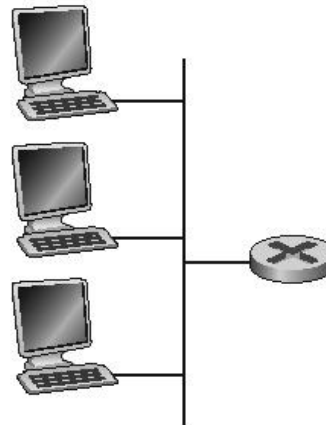
□ Ponto-a-ponto:

- PPP para acesso discado
- Conexão entre switch Ethernet e hospedeiro

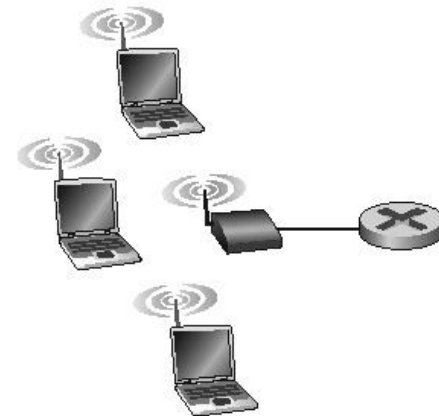
□ *broadcast* (cabo ou meio compartilhado):

- Ethernet tradicional
- Upstream HFC
- 802.11 LAN sem fio
- Satélite
- Etc.

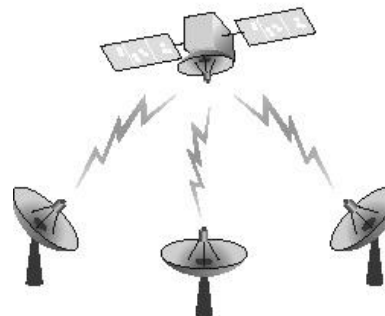
Compartilhado com fio
(por exemplo, Ethernet)



Compartilhado sem fio
(por exemplo, Wi-Fi)



Satélite



Coquetel



Protocolos de Acesso Múltiplo

- canal de comunicação único de *broadcast*
- interferência: quando dois ou mais nós transmitem simultaneamente
 - **colisão** se um nó receber dois ou mais sinais ao mesmo tempo

Protocolo de acesso múltiplo

- algoritmo distribuído que determina como os nós compartilham o canal, isto é, determina quando um nó pode transmitir
- comunicação sobre o compartilhamento do canal deve usar o próprio canal!
 - não há canal fora da faixa para coordenar a transmissão

Protocolo Ideal de Acesso Múltiplo

Para um canal de *broadcast* com taxa de R bps:

1. Quando apenas um nó tem dados para enviar, esse nó obtém uma vazão de R bps.
2. Quando M nós têm dados para enviar, cada um desses nós poderá transmitir em média a uma taxa de R/M bps.
3. Completamente descentralizado
 - nenhum nó especial (mestre) para coordenar as transmissões
 - nenhuma sincronização de relógios ou slots
4. Simples para que sua implementação seja barata

Taxonomia dos Protocolos MAC

Três categorias gerais:

□ **divisão de Canal**

- divide o canal em pequenos "pedaços"
(*slots*/compartimentos de tempo, frequência, código)
- aloca pedaço a um dado nó para uso exclusivo deste

□ **acesso Aleatório**

- canal não é dividido, podem ocorrer colisões
- "recuperação" das colisões

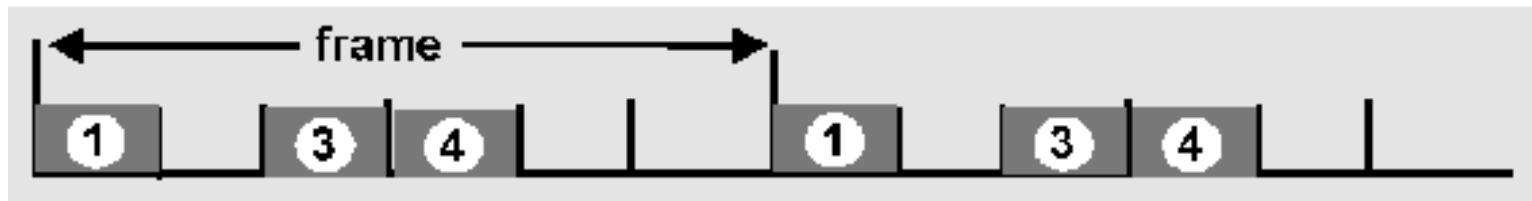
□ **revezamento**

- Nós se alternam em revezamento, mas um nó que possui mais dados a transmitir pode demorar mais quando chegar a sua vez

Protocolos MAC de divisão de canal: TDMA

TDMA: Acesso Múltiplo por Divisão de Tempo

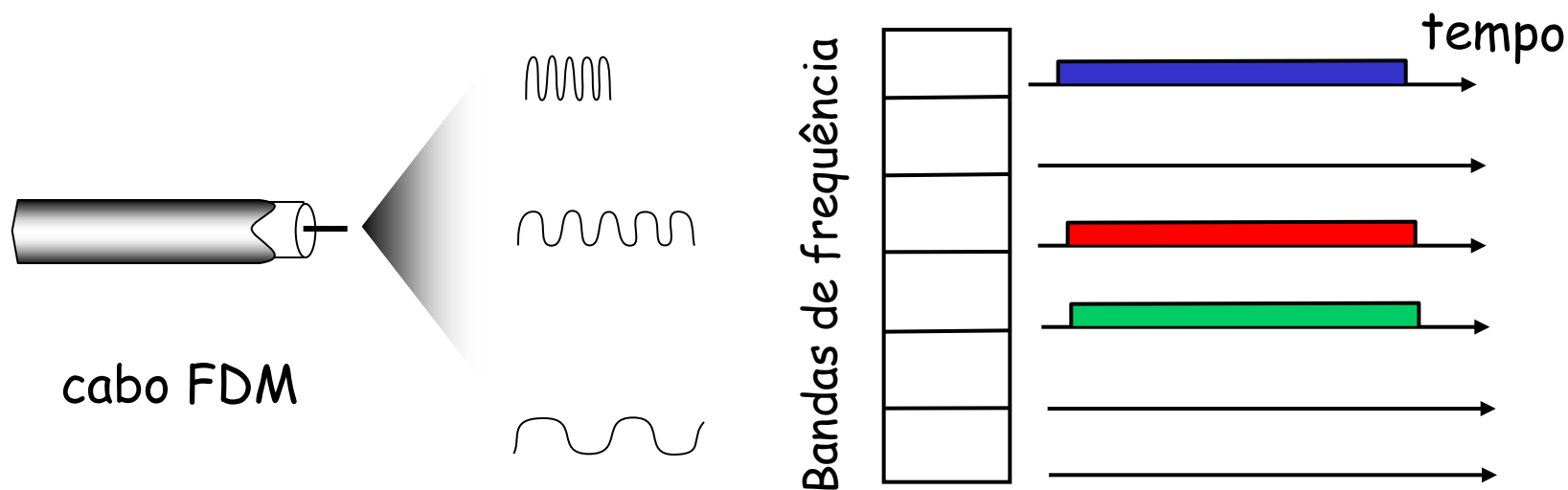
- acesso ao canal em "turnos"
- cada estação recebe um comprimento fixo de compartimento (comprimento = tempo de tx do pacote) em cada turno
- compartimentos não usados permanecem ociosos
- Exemplo: LAN com 6 estações: compartimentos 1, 3 e 4 com pacotes, compartimentos 2, 5 e 6 ociosos



Protocolos MAC de divisão de canal: FDMA

FDMA: Acesso Múltiplo por Divisão de Frequência

- espectro do canal dividido em bandas de frequência
- a cada estação é atribuída uma banda fixa de frequência
- tempo de transmissão não usado nas bandas permanecem ociosos
- exemplo: LAN com 6 estações, 1,3,4 com pacotes, bandas 2,5,6 ociosas



Protocolos de Acesso Aleatório

- Quando nó tem um pacote para transmitir
 - transmite na taxa máxima R .
 - nenhuma coordenação *a priori* entre os nós
- dois ou mais nós transmitindo → "colisão",
- O **protocolo MAC de acesso aleatório** especifica:
 - como detectar colisões
 - como se recuperar delas (através de retransmissões retardadas, por exemplo)
- Exemplos de protocolos MAC de acesso aleatório:
 - slotted ALOHA
 - ALOHA
 - CSMA, CSMA/CD, CSMA/CA

Slotted ALOHA

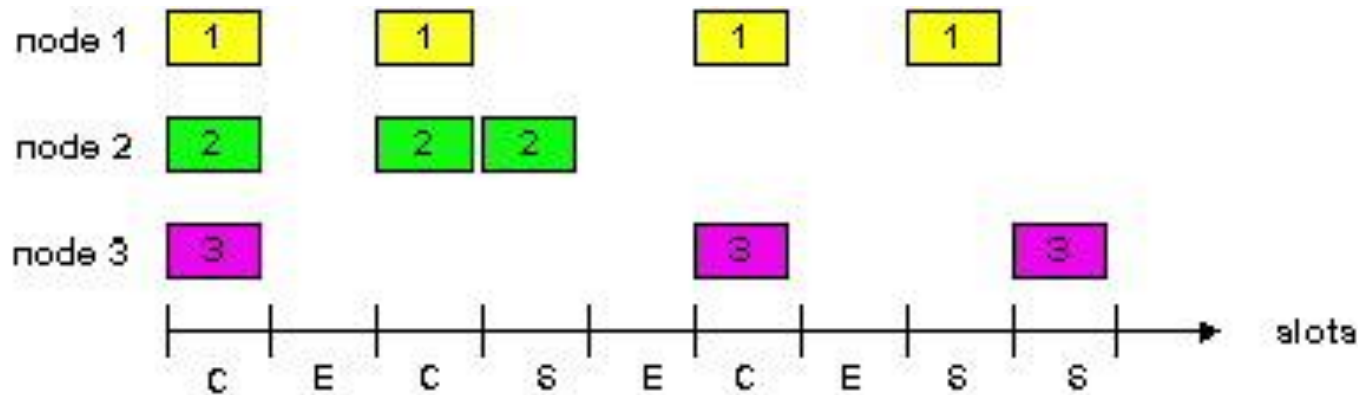
Hipóteses

- todos os quadros têm o mesmo tamanho (L bits)
- tempo é dividido em slots de tamanho igual, tempo para transmitir 1 quadro (L/R seg)
- nós começam a transmitir quadros apenas no início dos intervalos (*slots*)
- nós são sincronizados
- se 2 ou mais nós transmitirem num *slot*, todos os nós detectam a colisão

Operação

- quando o nó obtém um novo quadro, ele espera até o início do próximo *slot* e transmite o quadro inteiro
- Se não houver colisão, nó poderá enviar um novo quadro no próximo *slot*
- caso haja uma colisão (detectada antes do final do intervalo), nó retransmite o quadro em cada intervalo subsequente com probabilidade p até obter sucesso

Slotted ALOHA



Vantagens

- ☐ único nó ativo pode transmitir continuamente na taxa máxima do canal
- ☐ Altamente descentralizado: apenas *slots* nos nós precisam estar sincronizados
- ☐ simples

Desvantagens

- ☐ colisões: *slots* desperdiçados
- ☐ *slots* ociosos (desperdício)
- ☐ nós podem ser capazes de detectar colisões num tempo inferior ao da transmissão do pacote
- ☐ sincronização dos relógios

Eficiência do Slotted Aloha

Eficiência é a fração de longo prazo de *slots* bem sucedidos quando há muitos nós cada um com muitos quadros para transmitir

- Assuma N nós com muitos quadros para enviar, cada um transmite num *slot* com probabilidade p
- probabilidade que nó 1 tenha sucesso em um *slot* = $p(1-p)^{N-1}$
- probabilidade que qualquer nó tenha sucesso = $Np(1-p)^{N-1}$

- Para eficiência máxima com N nós, encontre p^* que maximiza $Np(1-p)^{N-1}$
- Para muitos nós, faça limite para $Np^*(1-p^*)^{N-1}$ quando N tende a infinito, dá

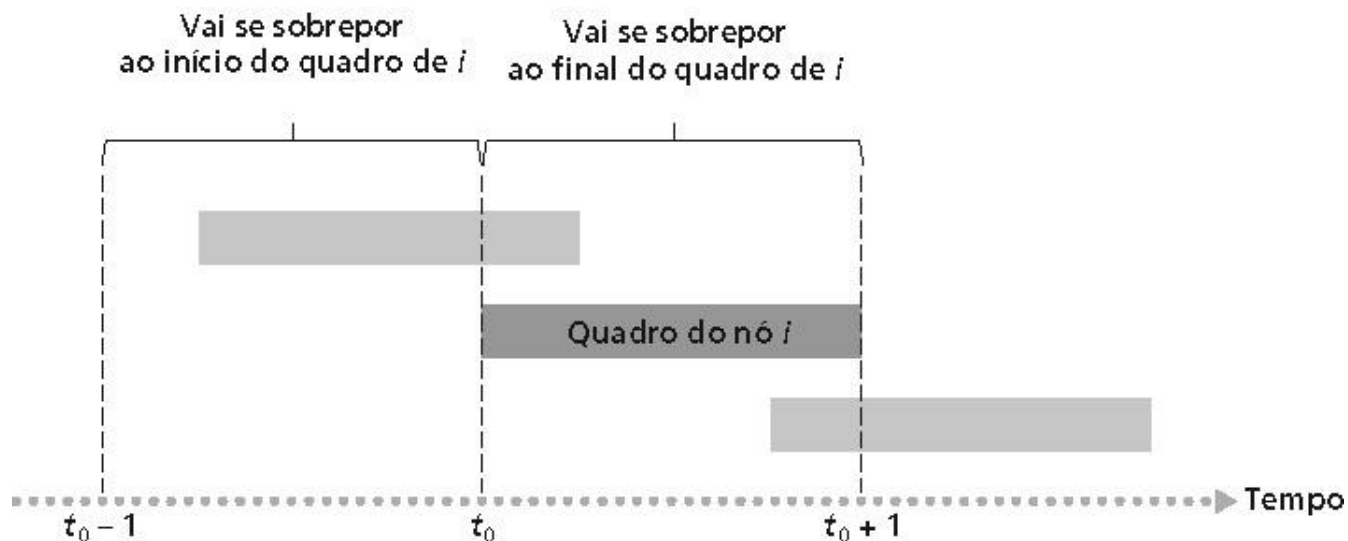
eficiência máxima = $1/e = 0,37$

Melhor caso: canal usado para transmissões úteis em apenas 37% do tempo!



ALOHA Puro (sem slots)

- Aloha puro (sem slots): mais simples, sem sincronização
- Ao chegar um quadro no nó
 - transmite imediatamente
- Probabilidade de colisão aumenta:
 - quadro enviado em t_0 colide com outros quadros enviados em $[t_0-1, t_0+1]$



Eficiência do Aloha puro

$P(\text{sucesso por um dado nó}) = P(\text{nó transmite}) \cdot$

$P(\text{nenhum outro nó transmite em } [t_0-1, t_0] \cdot$

$P(\text{nenhum outro nó transmite em } [t_0, t_0+1])$

$$= p \cdot (1-p)^{N-1} \cdot (1-p)^{N-1}$$

$$= p \cdot (1-p)^{2(N-1)}$$

... escolhendo o valor ótimo de p e deixando $n \rightarrow$ infinito ...

$$= 1/(2e) = 0,18$$

Ainda pior do que o Slotted Aloha!

CSMA (Acesso múltiplo com detecção de portadora)

CSMA (Carrier Sense Multiple Access):

Escuta antes de transmitir (detecção de portadora):

- Se o canal estiver livre: transmite todo o quadro
- Se o canal estiver ocupado, adia a transmissão

□ Analogia humana: não interrompa outros!

Colisões no CSMA

colisões ainda *podem*
acontecer:

atraso de propagação significa
que dois nós podem não ouvir
a transmissão do outro

colisão:

todo o tempo de
transmissão é desperdiçado

nota:

papel da distância e atraso de
propagação na determinação da
probabilidade de colisão

Disposição espacial dos nós



t_0
time
↓

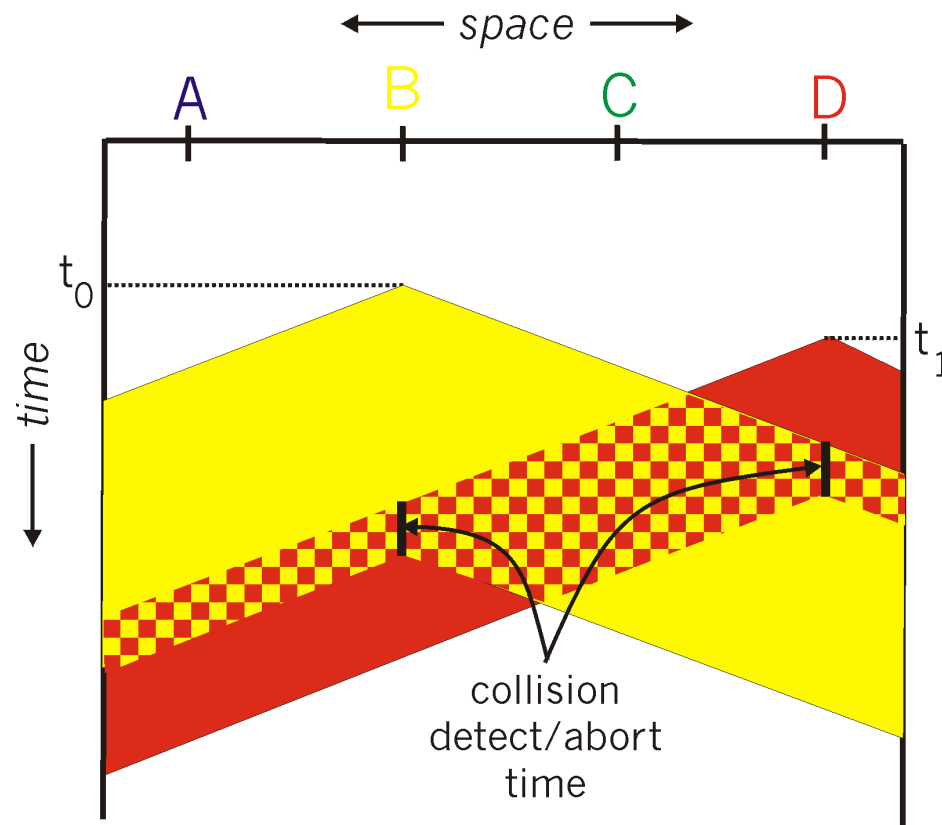
t_1

CSMA/CD (Detecção de Colisões)

CSMA/CD: detecção da portadora, adia a transmissão como no CSMA

- As colisões são *detectadas* em pouco tempo
- Transmissões que sofreram colisões são abortadas, reduzindo o desperdício do canal
- Detecção de colisões:
 - Fácil em LANs cabeadas: mede a potência do sinal, compara o sinal recebido com o transmitido
 - Difícil em LANs sem fio: o receptor é desligado durante a transmissão
- Analogia humana: bate papo educado!

Detecção de colisões em CSMA/CD



Algoritmo CSMA/CD do Ethernet

1. Adaptador recebe datagrama da camada de rede e cria um quadro
2. Se o adaptador percebe que o canal está ocioso, começa a transmitir o quadro. Se percebe que o canal está ocupado, espera que o canal fique livre e transmite
3. Se o adaptador transmitir todo o quadro sem detectar outra transmissão, o adaptador concluiu a operação com o quadro!
4. Se o adaptador detectar outra transmissão enquanto estiver transmitindo, aborta e envia sinal de **reforço de colisão** de 48 bits
5. Após o aborto, o adaptador entra na **retirada (backoff) exponencial binária**:
 - após a m -ésima colisão, o adaptador escolhe um K aleatoriamente entre $\{0, 1, 2, \dots, 2^m - 1\}$. O adaptador espera $K \cdot 512$ tempos de bit e retorna ao Passo 2
 - quanto mais colisões houver, mas longos serão os intervalos de retirada.

Eficiência do CSMA/CD

- t_{prop} = atraso máximo de prop. entre 2 nós na LAN
- t_{trans} = tempo para transmitir quadro de tamanho máximo

$$\text{eficiência} = \frac{1}{1 + 5t_{prop} / t_{trans}}$$

- Eficiência vai para 1 à medida que:
 - t_{prop} vai para 0
 - t_{trans} vai para infinito
- Muito melhor do que ALOHA, e ainda é descentralizado, simples, e barato!

Protocolos MAC de "revezamento"

Protocolos MAC de divisão de canal:

- Compartilha o canal eficientemente e de forma justa em altas cargas
- Ineficiente em baixas cargas: atraso no canal de acesso, alocação de $1/N$ da largura de banda mesmo com apenas 1 nó ativo!

Protocolos MAC de acesso aleatório:

- eficiente em baixas cargas: um único nó pode utilizar completamente o canal
- Altas cargas: overhead com colisões

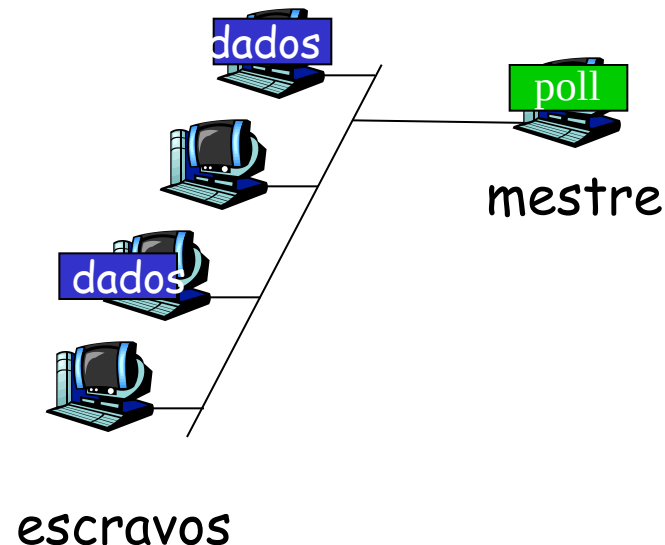
Protocolos de revezamento:

Procura oferecer o melhor dos dois mundos!

Protocolos MAC de "revezamento"

Seleção (Polling):

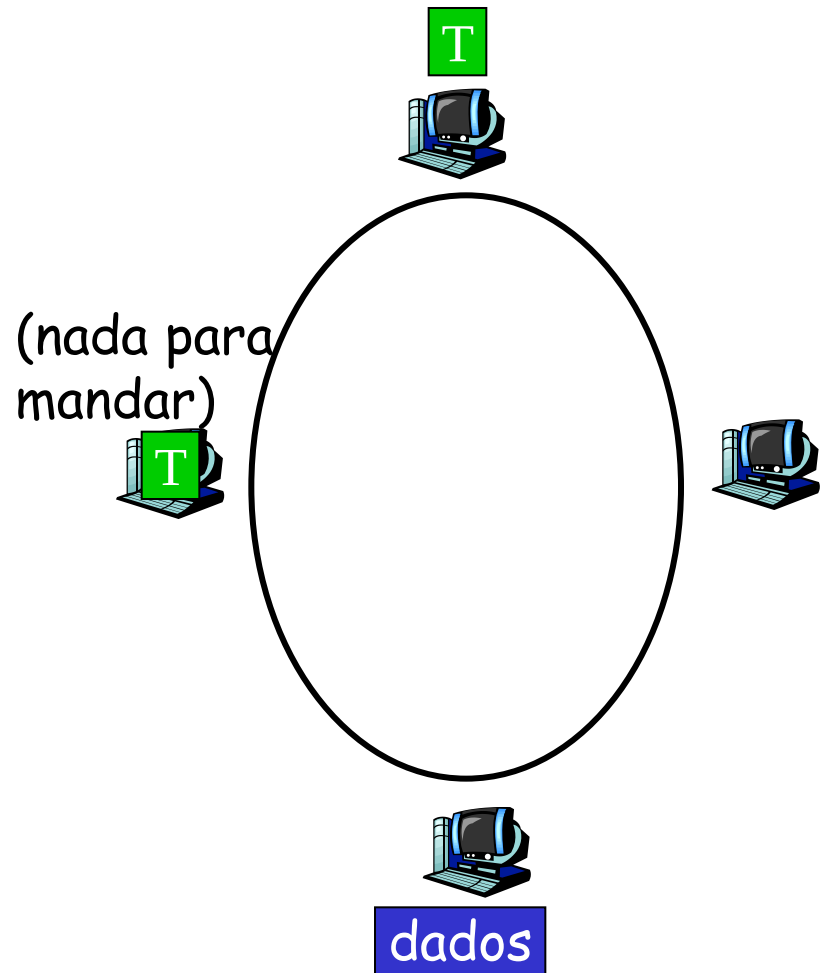
- ☐ Nó mestre "convida" nós escravos a transmitir em revezamento
- ☐ Usado tipicamente com dispositivos escravo burros.
- ☐ Preocupações:
 - Overhead com as consultas (*polling*)
 - Latência
 - Ponto único de falha (mestre)



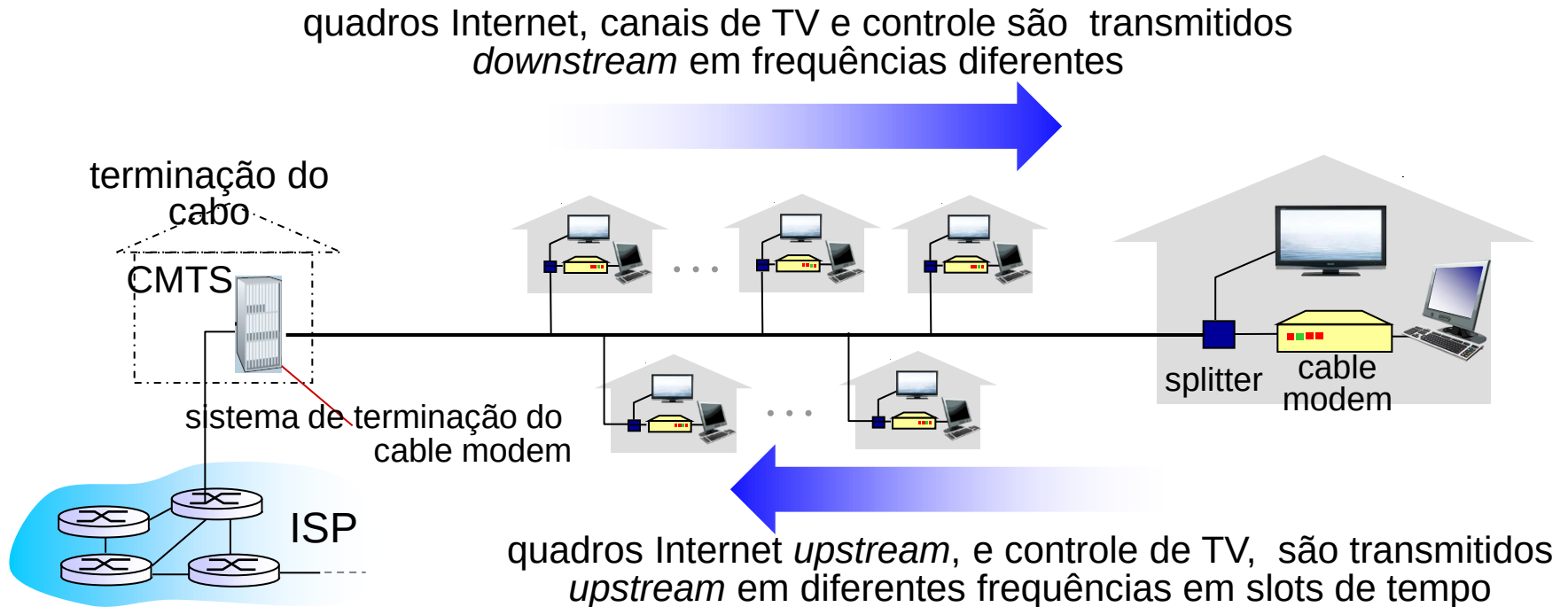
Protocolos MAC de "revezamento"

Passagem de permissão (token):

- ☐ controla **permissão** passada de um nó para o próximo de forma sequencial.
- ☐ mensagem de passagem da permissão
- ☐ preocupações:
 - *overhead* com a passagem de permissão
 - latência
 - Ponto único de falha (permissão)

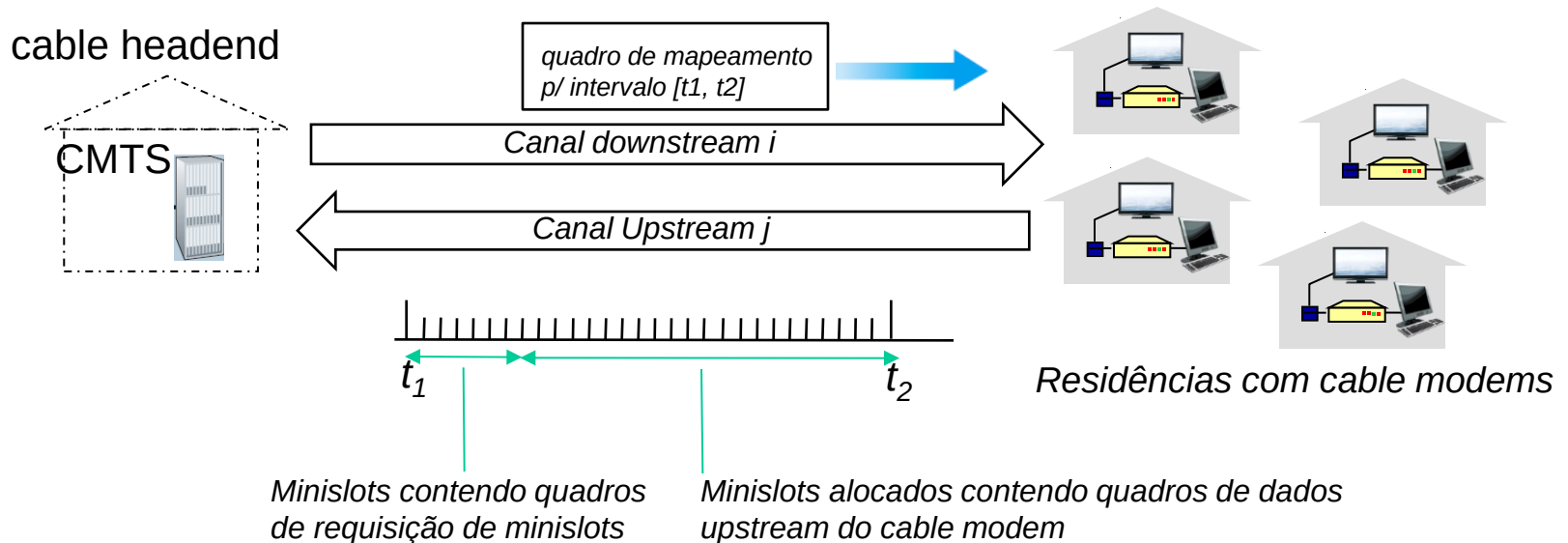


Rede de acesso a cabo



- ❑ **Múltiplos** canais de difusão *downstream* de 40Mbps
 - Um único CMTS transmite para os diversos canais
- ❑ **Múltiplos** canais *upstream* de 30 Mbps
 - **Acesso múltiplo:** todos os usuários disputam alguns slots de tempo em canais *upstream* (outros são pré-alocados)

Rede de acesso a cabo



- **DOCSIS:** espec. da interface de serviço de dados sobre cabo
 - FDM sobre as frequências dos canais *up* e *downstream*
 - TDM *upstream*: alguns slots são alocados, outros têm disputa
 - quadro de mapeamento *downstream*: aloca slots *upstream*
 - requisição de slots *upstream* (e dados) são transmitidos através de acesso aleatório (retirada binária) em slots selecionados

Resumo dos protocolos MAC

- ❑ **divisão do canal** por tempo, frequência ou código
 - Divisão de Tempo, Divisão de Frequência
- ❑ **acesso aleatório** (dinâmico):
 - ALOHA, S-ALOHA, CSMA, CSMA/CD
 - escutar a portadora: fácil em algumas tecnologias (cabeadas), difícil em outras (sem fio)
 - CSMA/CD usado na Ethernet
 - CSMA/CA usado no 802.11
- ❑ **Revezamento**
 - Seleção (*polling*) a partir de um ponto central, passagem de permissões
 - Bluetooth, FDDI, Token Ring

Camada de Enlace

5.1 Introdução e serviços

5.2 Detecção e correção de erros

5.3 Protocolos de acesso múltiplo

5.4 Redes Locais

- Endereçamento, ARP

- Ethernet

- Switches

- VLANs

5.5 Virtualização do enlace: MPLS

5.6 Redes de centros de dados

5.7 Um dia na vida de uma solicitação de página Web

Endereços MAC e ARP

□ Endereço IP de 32 bits:

- endereços da camada de rede para a interface
- usado pelo repasse da camada 3 (rede)

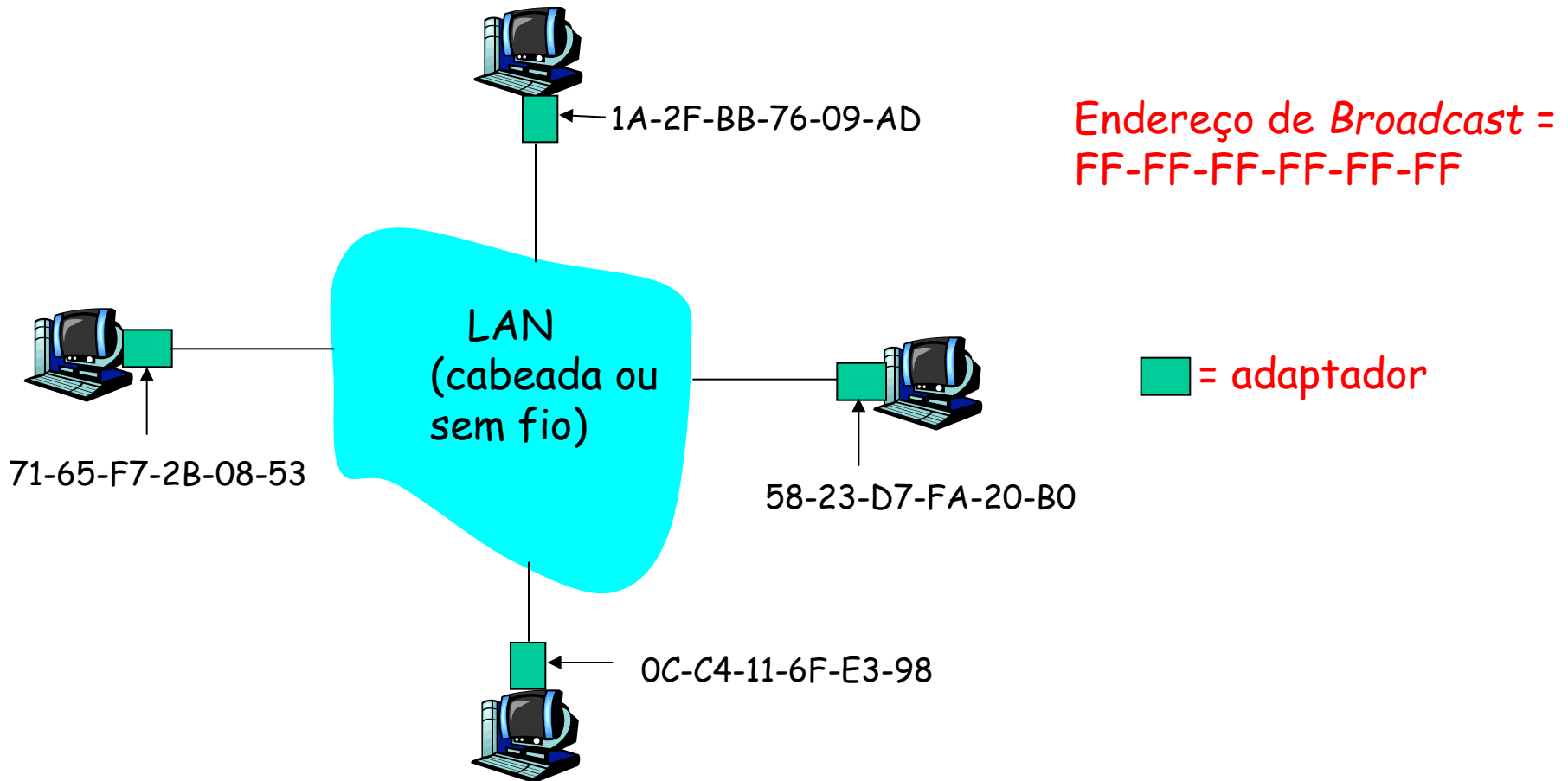
□ Endereço MAC (ou LAN, ou físico, ou Ethernet):

- Função: usado 'localmente' para levar o quadro de uma interface até outra interface conectada fisicamente (na mesma rede, no sentido do endereçamento IP)
- Endereço MAC de 48 bits (para a maioria das redes) gravado na ROM do adaptador, ou configurado por software
- Ex: 1A-2F-BB-76-09-AD

notação hexadecimal (base 16)
(cada “número” representa 4 bits)

Endereços MAC e ARP

cada adaptador na LAN possui um endereço **MAC** único



Endereço MAC (cont)

- Alocação de endereços MAC gerenciada pelo IEEE
- Um fabricante compra uma parte do espaço de endereços (para garantir unicidade)
- Analogia:
 - (a) endereço MAC: como número do CPF
 - (b) endereço IP: como endereço postal (CEP)
- endereço MAC tem estrutura linear => portabilidade
 - Pode mover um cartão LAN de uma LAN para outra
- endereço IP hierárquico NÃO é portátil (requer IP móvel)
 - Depende da subrede IP à qual o nó está conectado

ARP: Address Resolution Protocol (Protocolo de Resolução de Endereços)

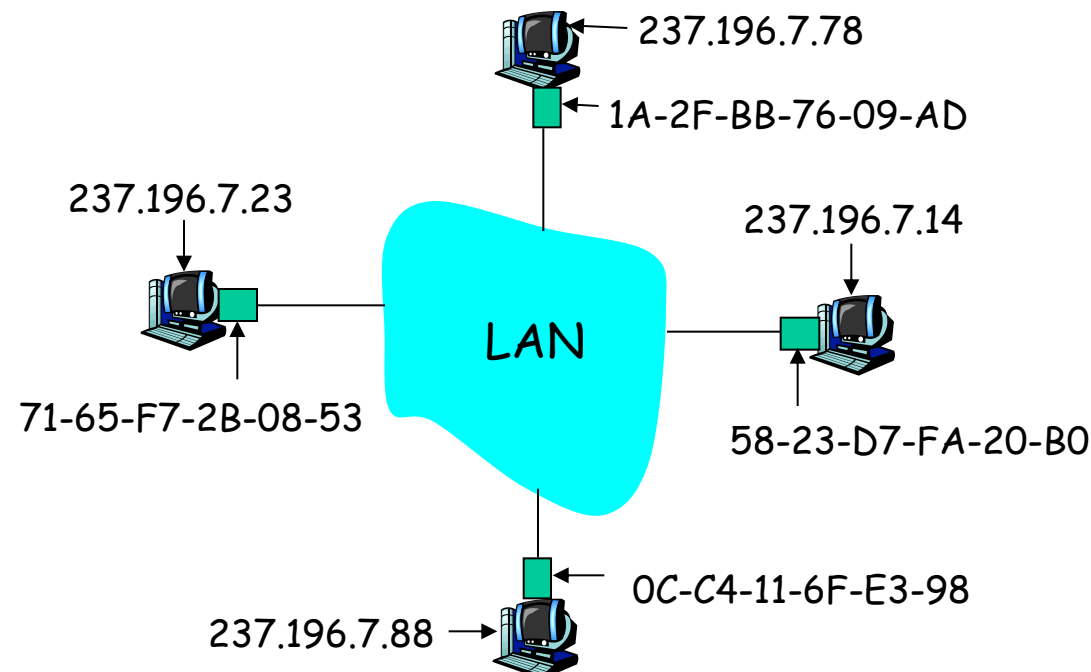
Pergunta: como obter o endereço MAC de B a partir do endereço IP de B?

□ Cada nó IP (Host, Roteador) de uma LAN possui tabela **ARP**

□ Tabela ARP: mapeamento de endereços IP/MAC para alguns nós da LAN

< endereço IP; endereço MAC; TTL >

- TTL (*Time To Live*): tempo a partir do qual o mapeamento de endereços será esquecido (valor típico de 20 min)

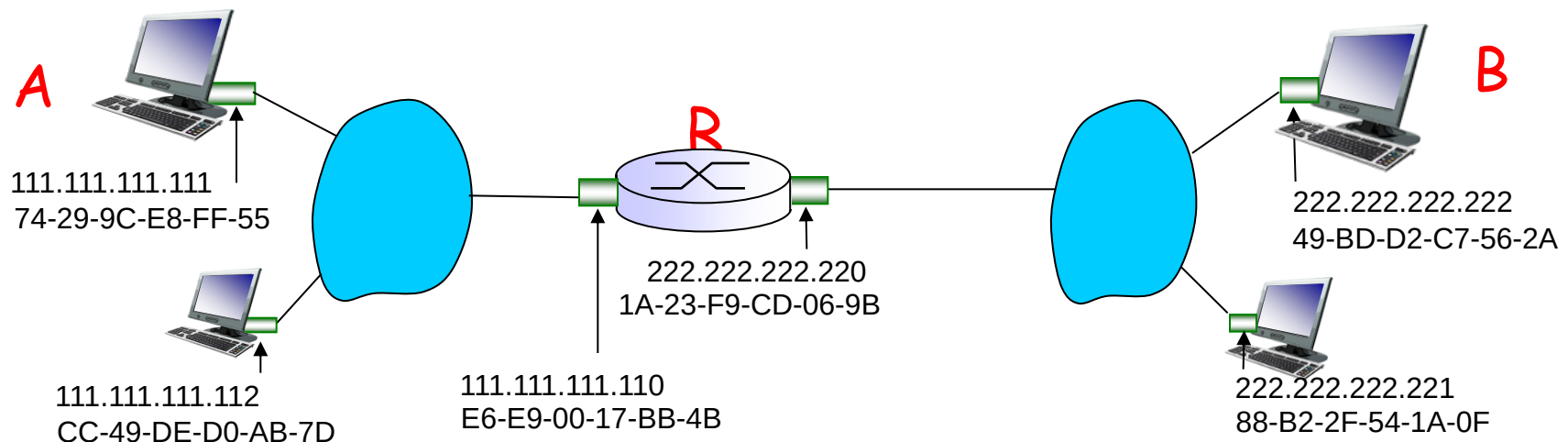


Protocolo ARP: mesma LAN (rede)

- A deseja enviar datagrama para B, e o endereço MAC de B não está na tabela ARP.
- A **difunde** o pacote de solicitação ARP, que contém o endereço IP de B
 - Endereço MAC destino = FF-FF-FF-FF-FF-FF
 - todas as máquinas na LAN recebem a consulta do ARP
- B recebe o pacote ARP, responde a A com o seu (de B) endereço MAC
 - Quadro enviado para o endereço MAC (*unicast*) de A
- Uma cache (salva) o par de endereços IP-para-MAC na sua tabela ARP até que a informação fique antiquada (*expire*)
 - 'soft state': informação que expira (vai embora) a menos que seja renovada
- ARP é "*plug-and-play*":
 - os nós criam suas tabelas ARP sem a intervenção do administrador da rede

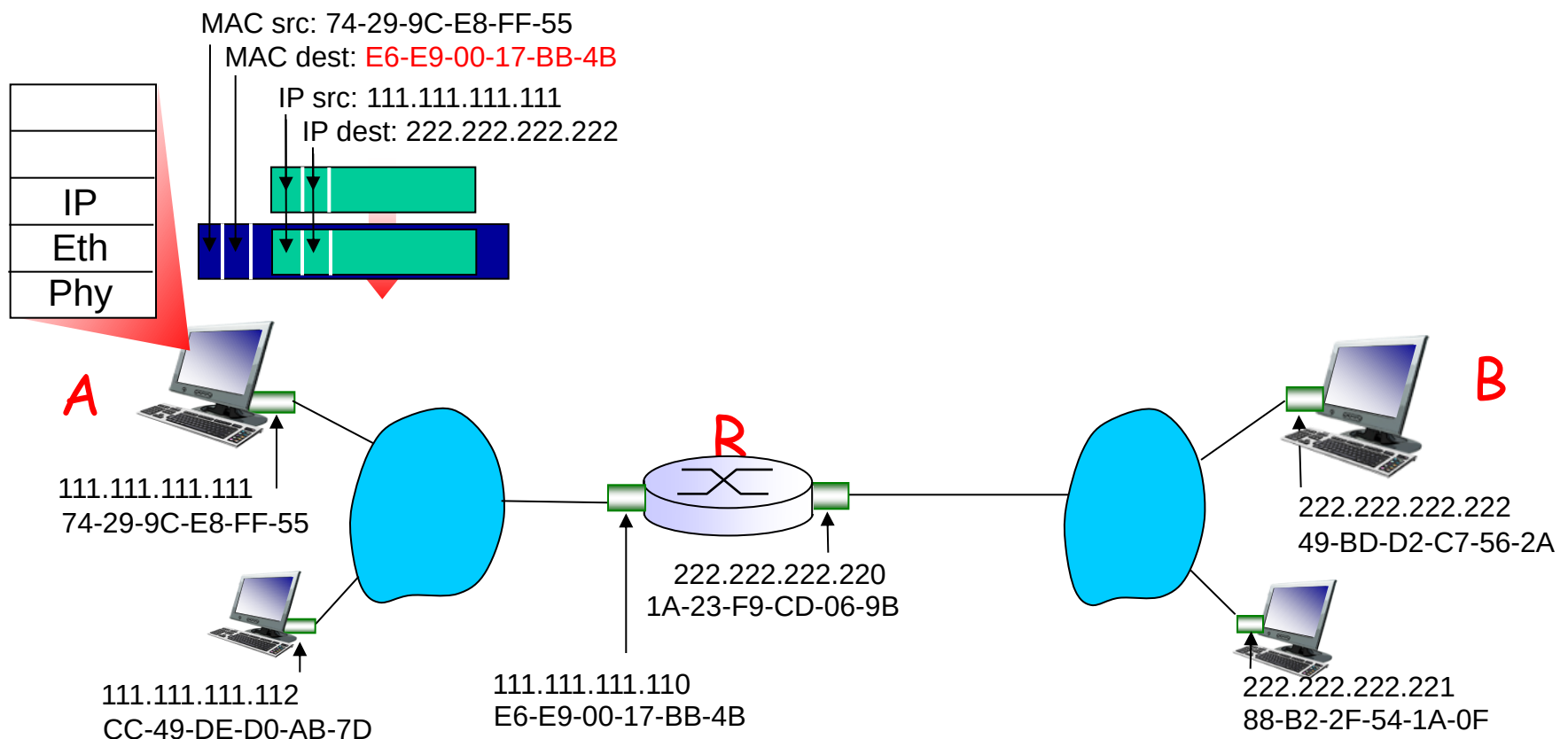
Endereçamento: repassando para outra LAN

- encaminhamento: **envio de datagrama de A para B via R**
 - foco no endereçamento - nas camadas IP (datagrama) e MAC (quadro)
 - assume que A conhece o endereço IP de B
 - assume que A conhece o endereço IP do próximo roteador, R (como?)
 - assume que A conhece o endereço MAC de R (como?)



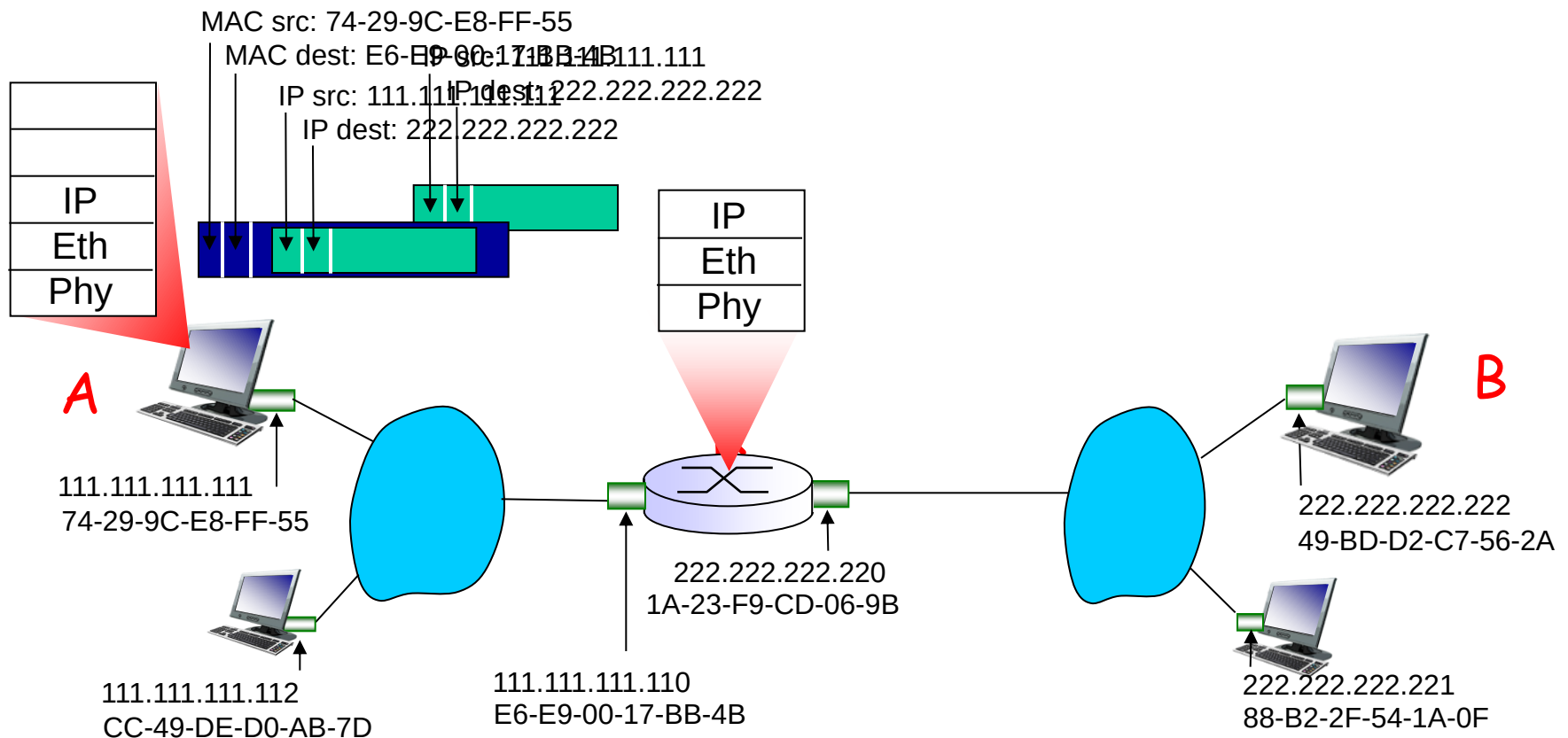
Endereçamento: repassando para outra LAN

- ❖ A cria datagrama IP com IP origem A, destino B
- ❖ A cria quadro da camada de enlace com endereço MAC de R como destino. quadro contém o datagrama IP de A-para-B



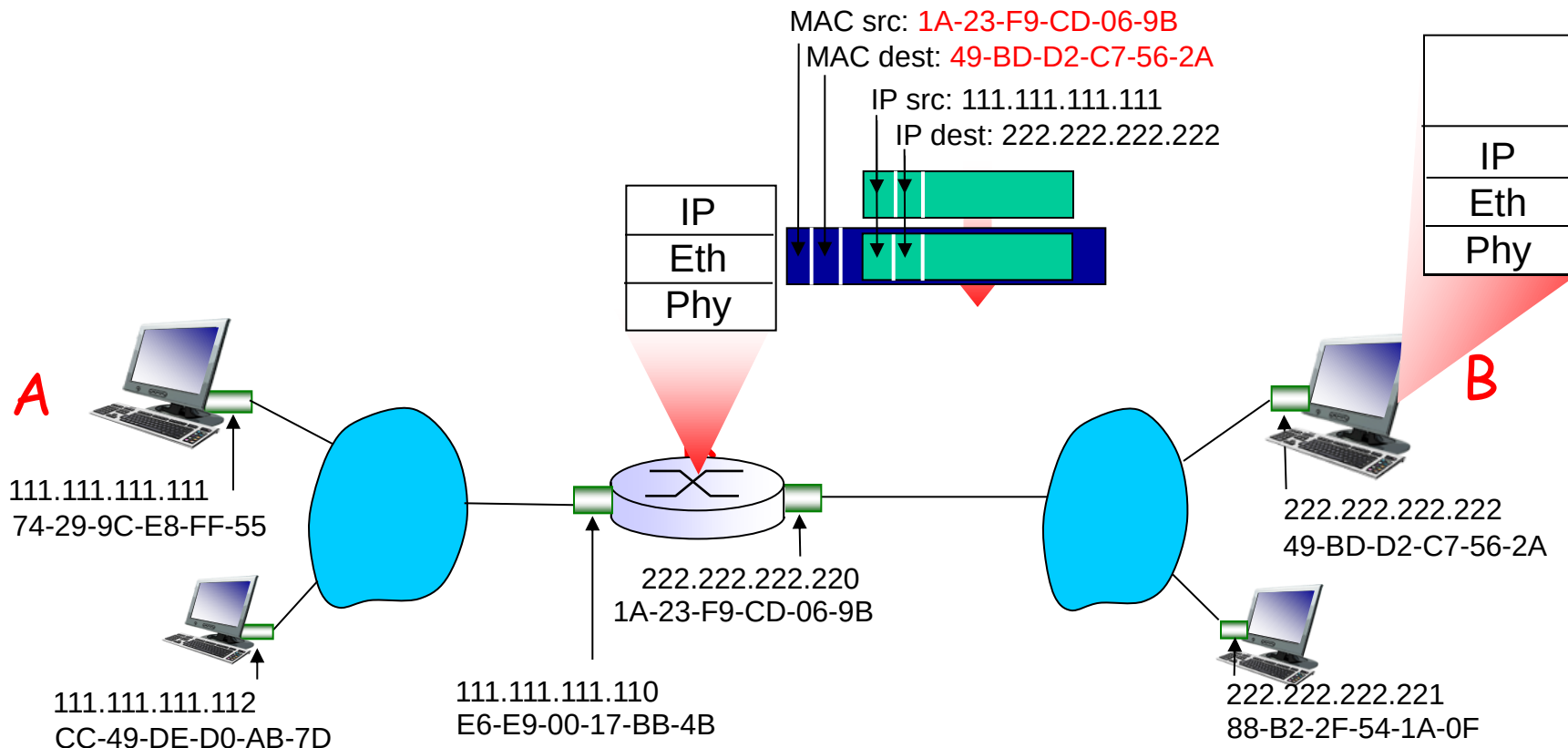
Endereçamento: repassando para outra LAN

- ❖ quadro enviado de A para R
- ❖ quadro recebido em R, datagrama removido, passado para o IP



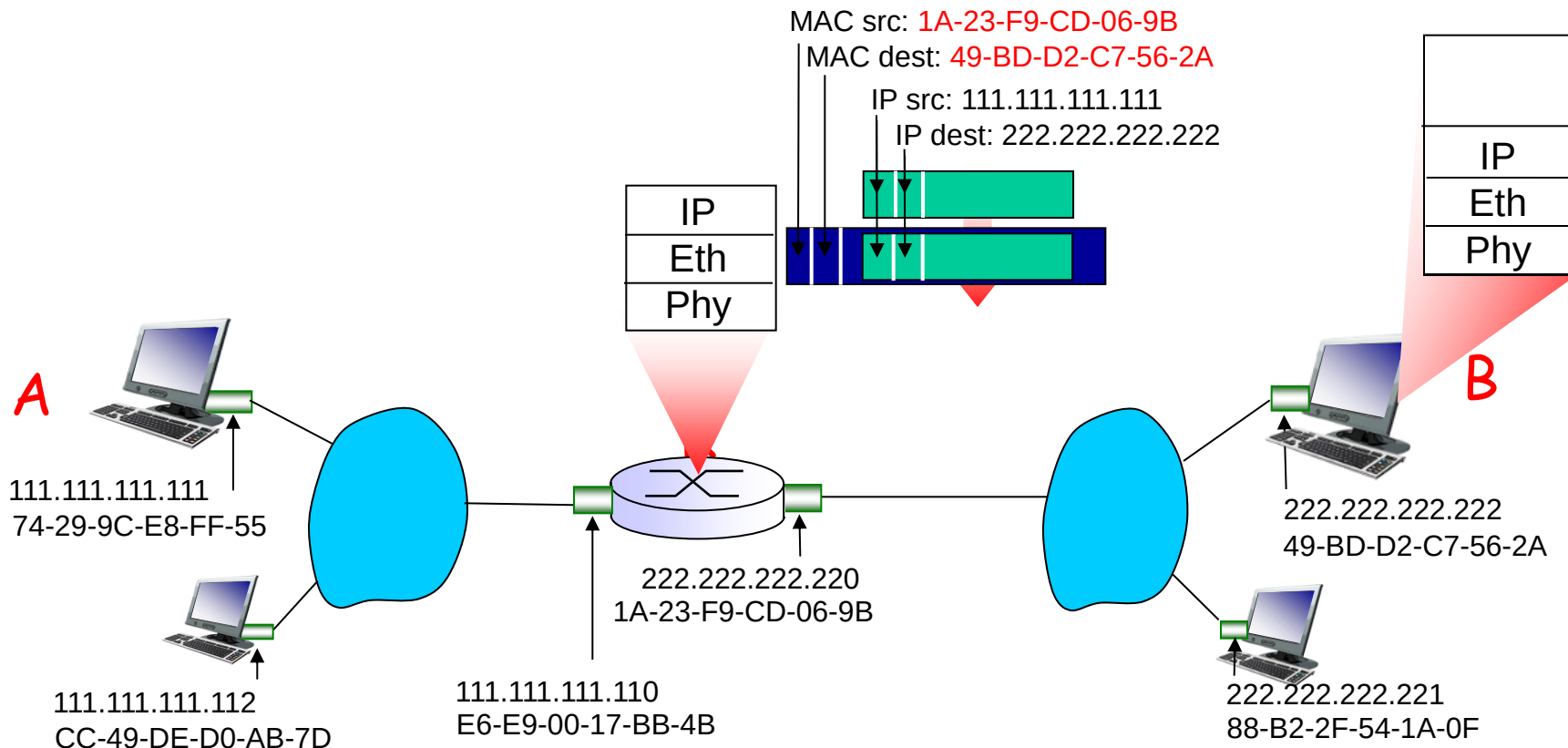
Endereçamento: repassando para outra LAN

- ❖ R repassa o datagrama com origem IP A, destino B
- ❖ R cria quadro da camada de enlace com endereço MAC de B como destino. quadro contém o datagrama IP de A para B



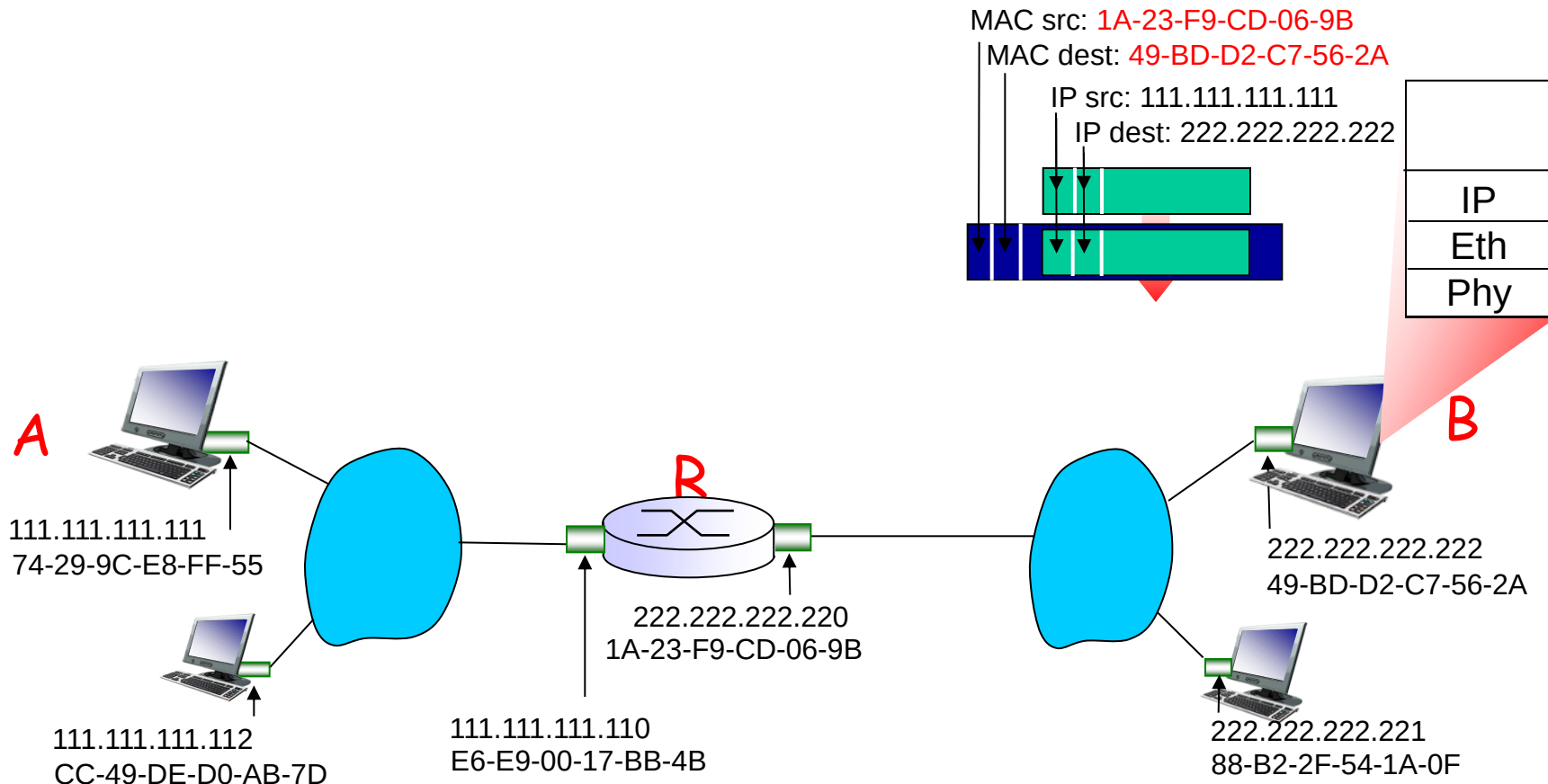
Endereçamento: repassando para outra LAN

- ❖ R repassa o datagrama com origem IP A, destino B
- ❖ R cria quadro da camada de enlace com endereço MAC de B como destino. quadro contém o datagrama IP de A para B



Endereçamento: repassando para outra LAN

- ❖ R repassa o datagrama com origem IP A, destino B
- ❖ R cria quadro da camada de enlace com endereço MAC de B como destino. quadro contém o datagrama IP de A para B



Camada de Enlace

5.1 Introdução e serviços

5.2 Detecção e correção de erros

5.3 Protocolos de acesso múltiplo

5.4 Redes Locais

- Endereçamento, ARP

- Ethernet

- Switches

- VLANs

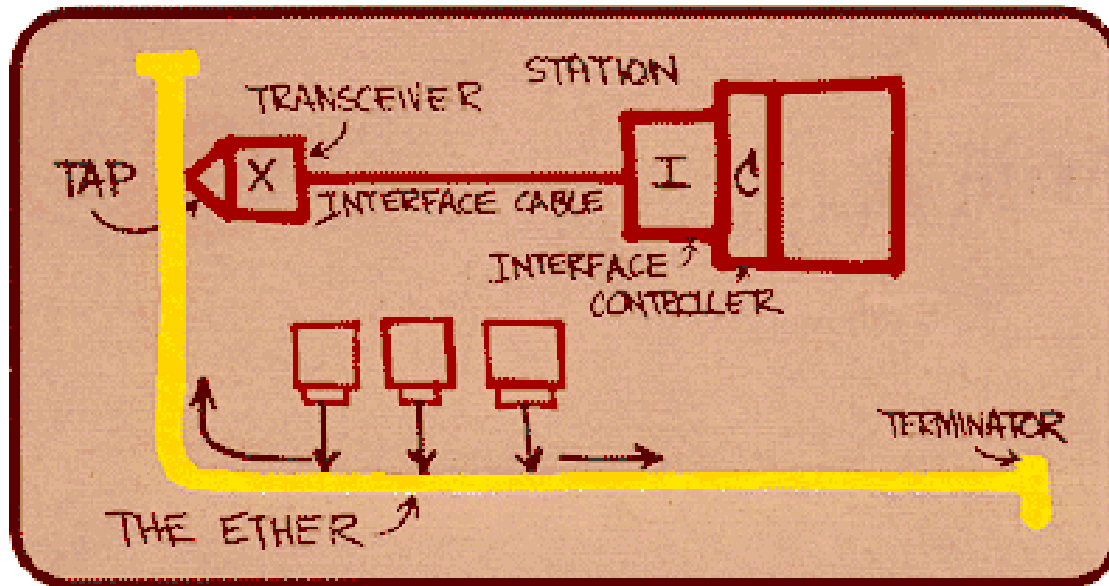
5.5 Virtualização do enlace: MPLS

5.6 Redes de centros de dados

5.7 Um dia na vida de uma solicitação de página Web

Ethernet

- MUITÍSSIMO difundida porque:
 - Muito barata! R\$50 para placas 10/100/1000Mbps!
 - A mais antiga das tecnologias de rede local (meados da década de 70)
 - Mais simples e menos cara que redes usando fibra ou ATM
 - Acompanhou o aumento de velocidade: 10 Mbps - 10 Gbps



Rascunho de Metcalfe
sobre o Ethernet

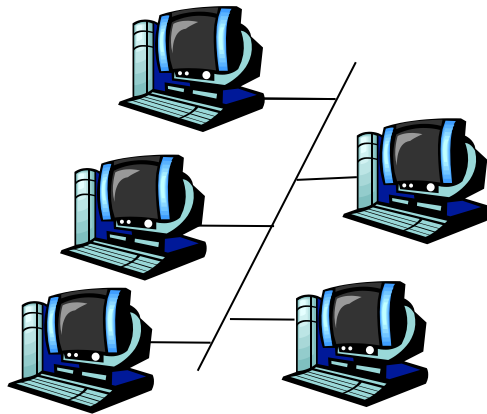
Ethernet: topologia física

□ **barramento** popular até meados dos anos 90

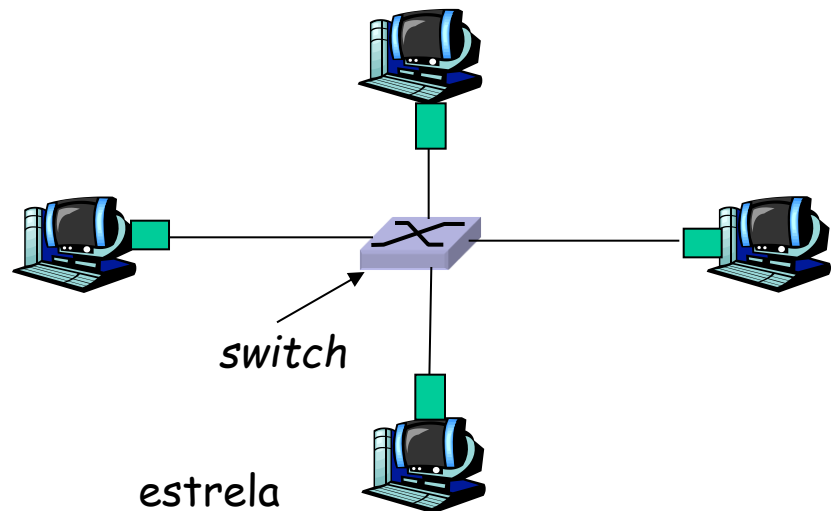
- Todos os nós no mesmo domínio de colisão (podem colidir um com o outro)

□ **estrela**: prevalência hoje

- Comutador (*switch*) ativo no centro
- Cada porta roda o protocolo Ethernet separadamente (os nós não colidem uns com os outros)



barramento: cabo coaxial



Estrutura do Quadro Ethernet

Adaptador remetente encapsula datagrama IP (ou pacote de outro protocolo da camada de rede) num **quadro Ethernet**



preâmbulo:

- ☐ 7 bytes com o padrão 10101010 seguidos por um byte com o padrão 10101011
- ☐ usado para sincronizar receptor ao relógio do remetente (relógios nunca são exatos, é muito provável que exista algum desvio entre eles)

Estrutura de Quadro Ethernet (cont)

- **Endereços:** 6 bytes para cada endereço MAC
 - se o adaptador recebe um quadro com endereço destino igual ao seu, ou com endereço de *broadcast* (ex., pacote ARP), ele passa os dados do quadro para o protocolo da camada de rede
 - caso contrário, o adaptador descarta o quadro
- **Tipo (2 bytes):** indica o protocolo da camada superior, usualmente IP, mas existe suporte para outros (tais como IPX da Novell e AppleTalk)
- **CRC (4 bytes):** verificado pelo receptor: se for detectado um erro, o quadro será descartado



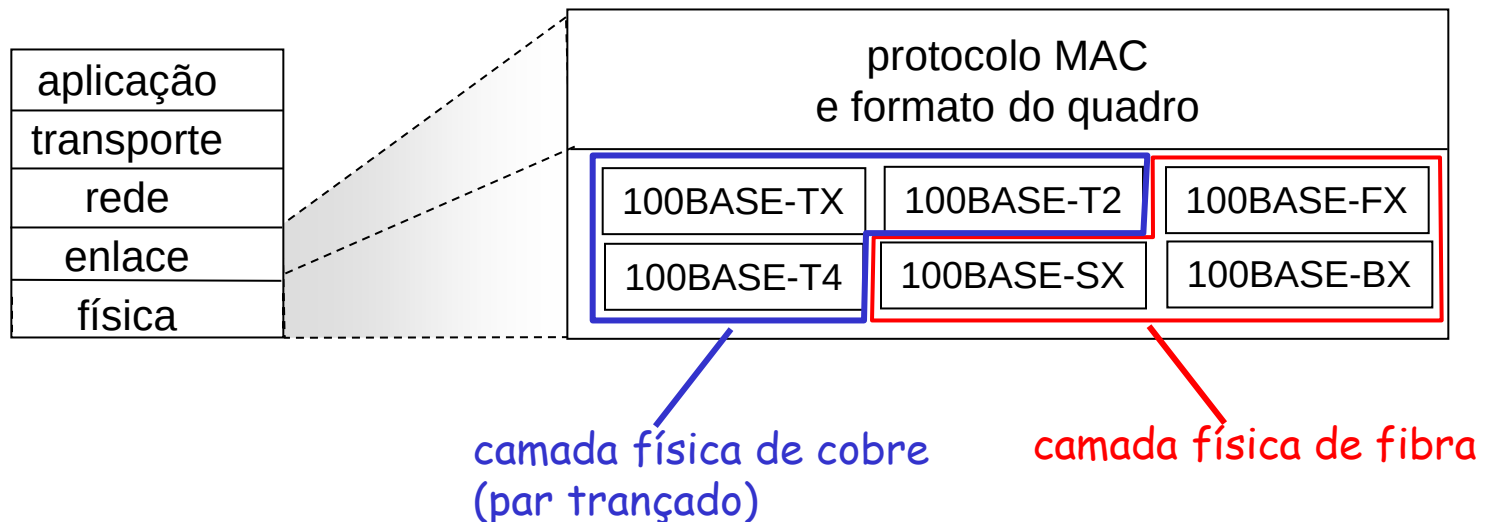
Ethernet: não confiável e sem conexão

- **sem conexão:** não há estabelecimento de conexão (saudação) entre os adaptadores transmissor e receptor.
- **Não confiável:** o adaptador receptor não envia ACKs ou NACKs para o adaptador transmissor
 - dados em quadros descartados são recuperados apenas se o transmissor usar transferência confiável em camadas mais altas (ex. TCP), caso contrário os dados estarão perdidos
- Protocolo MAC do Ethernet: **CSMA/CD com retirada binária**

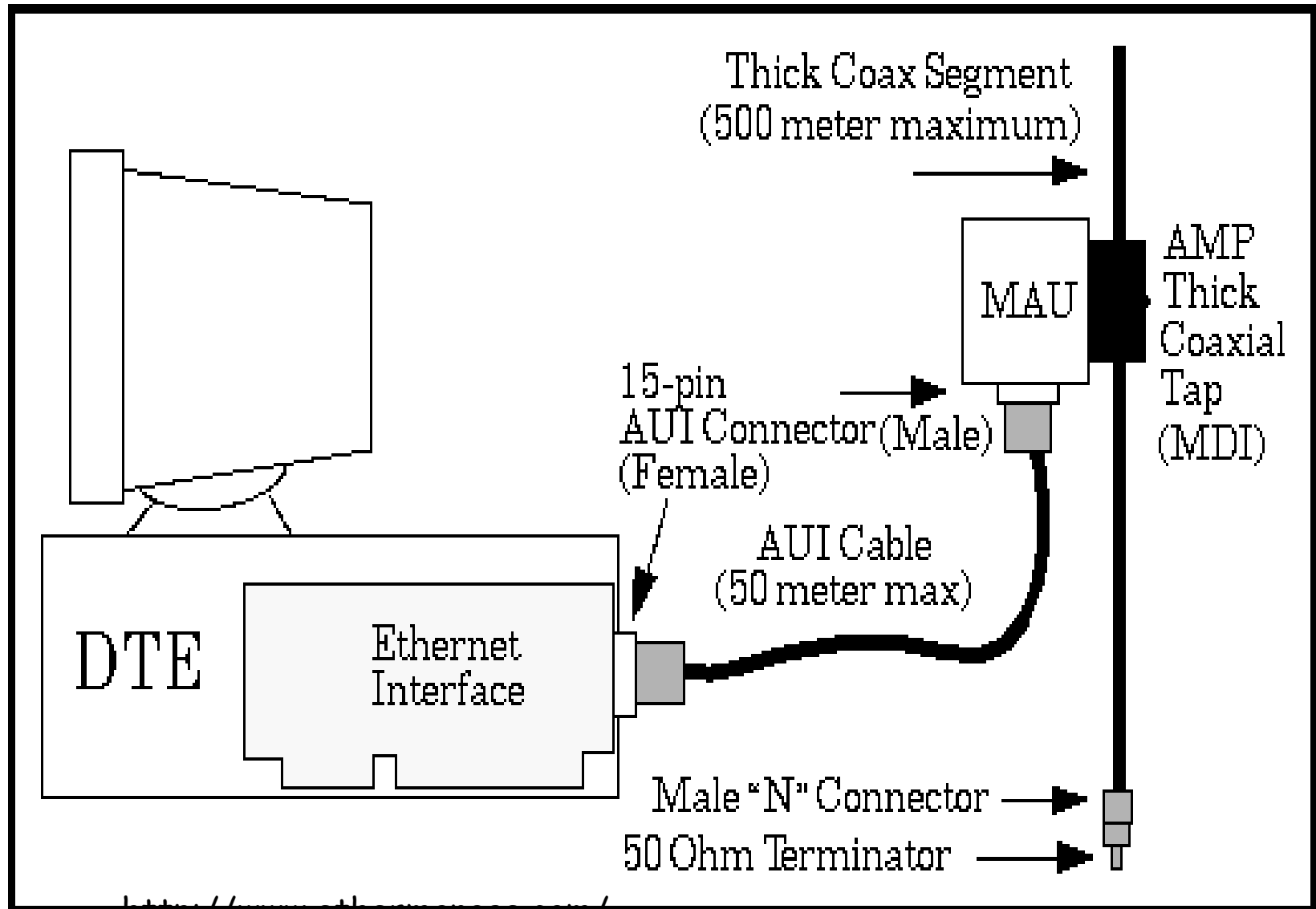
Padrões Ethernet 802.3: Camadas de Enlace e Física

□ diversos padrões Ethernet diferentes

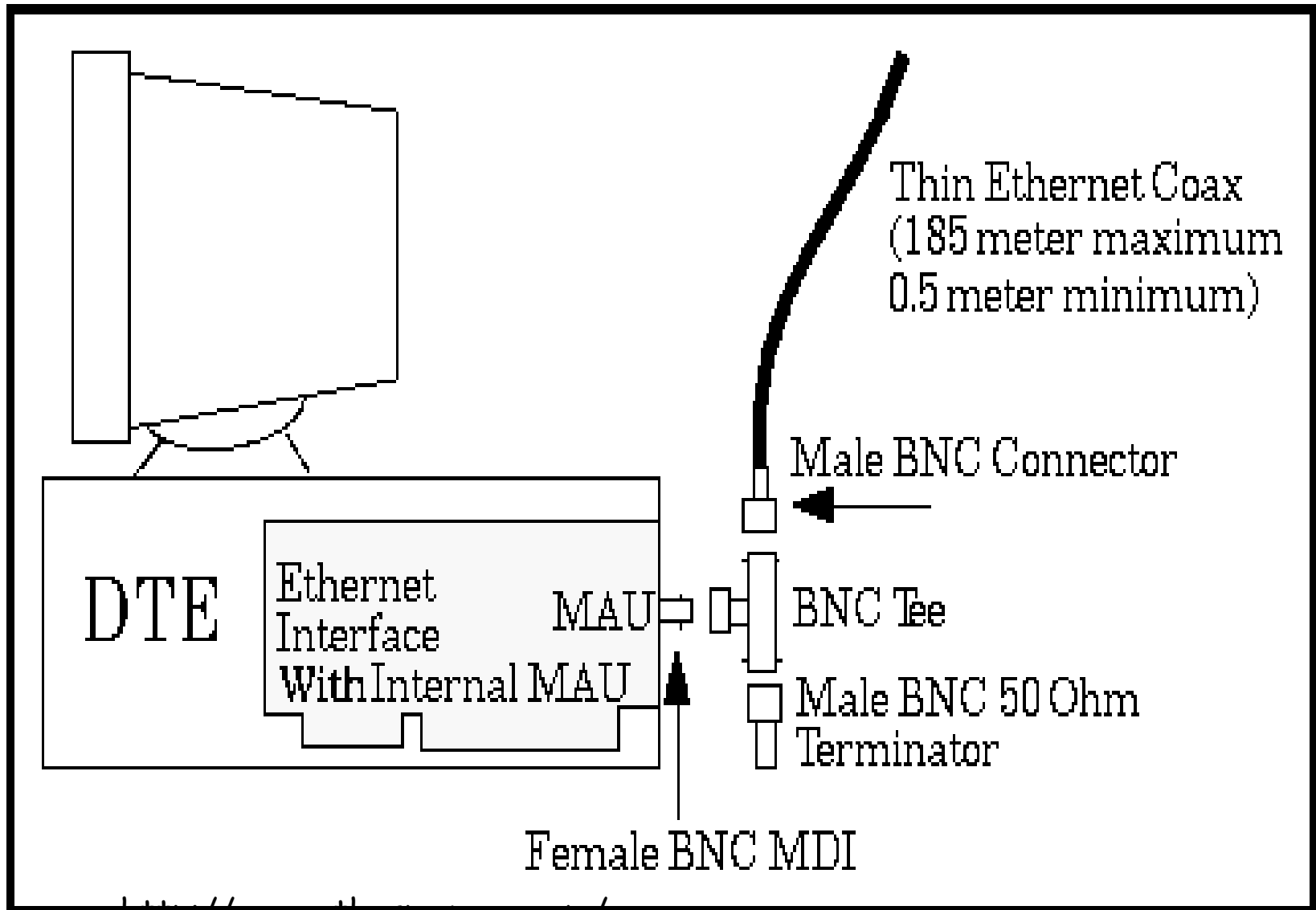
- têm em comum o protocolo MAC e o formato do quadro
- diferentes velocidades: 2Mbps, 10Mbps, 100Mbps, 1Gbps, 10Gbps
- diferentes meios da camada física: fibra, cabo



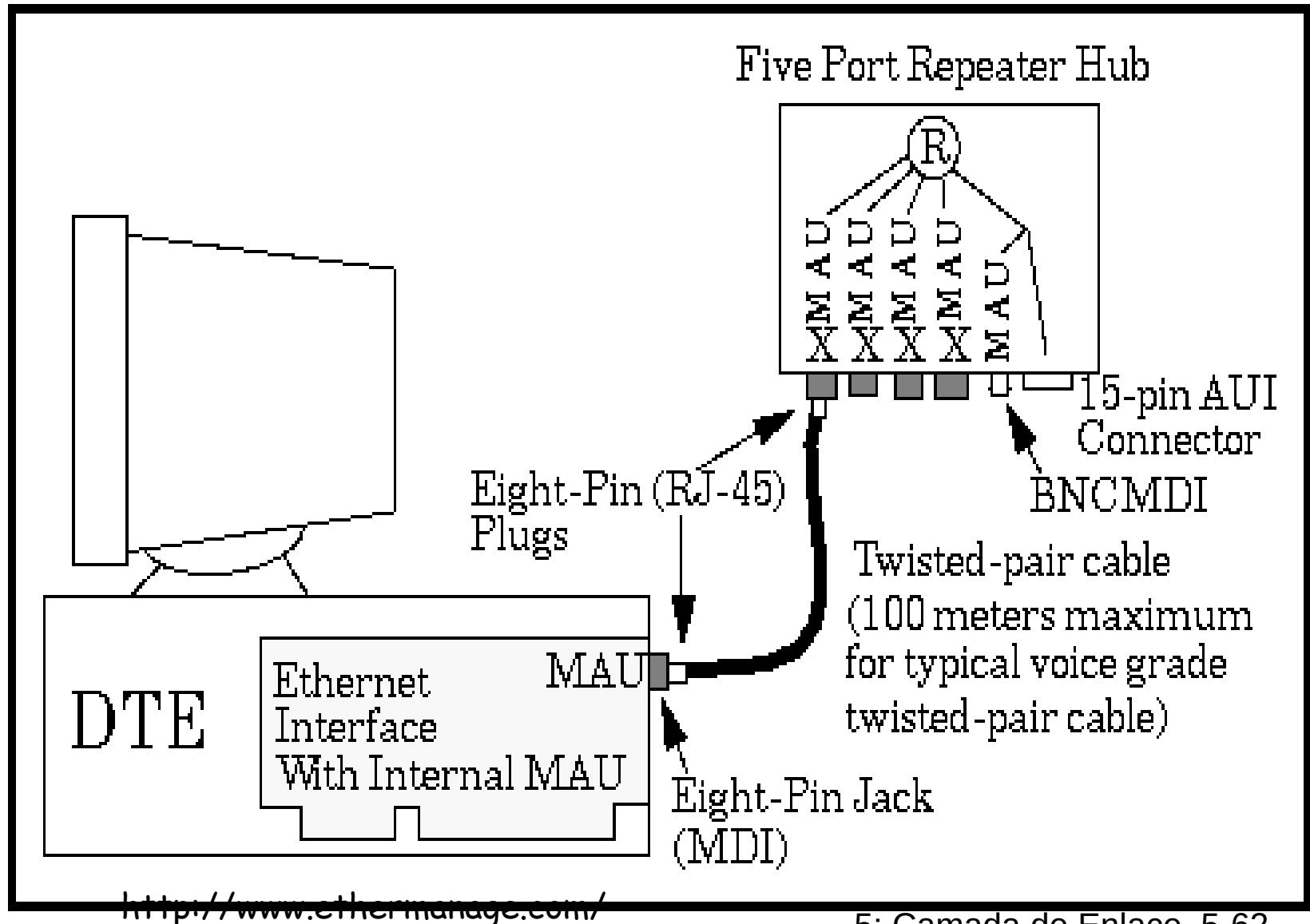
10Base5



10Base2



10BaseT

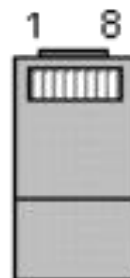


Conector RJ 45



Pin 1

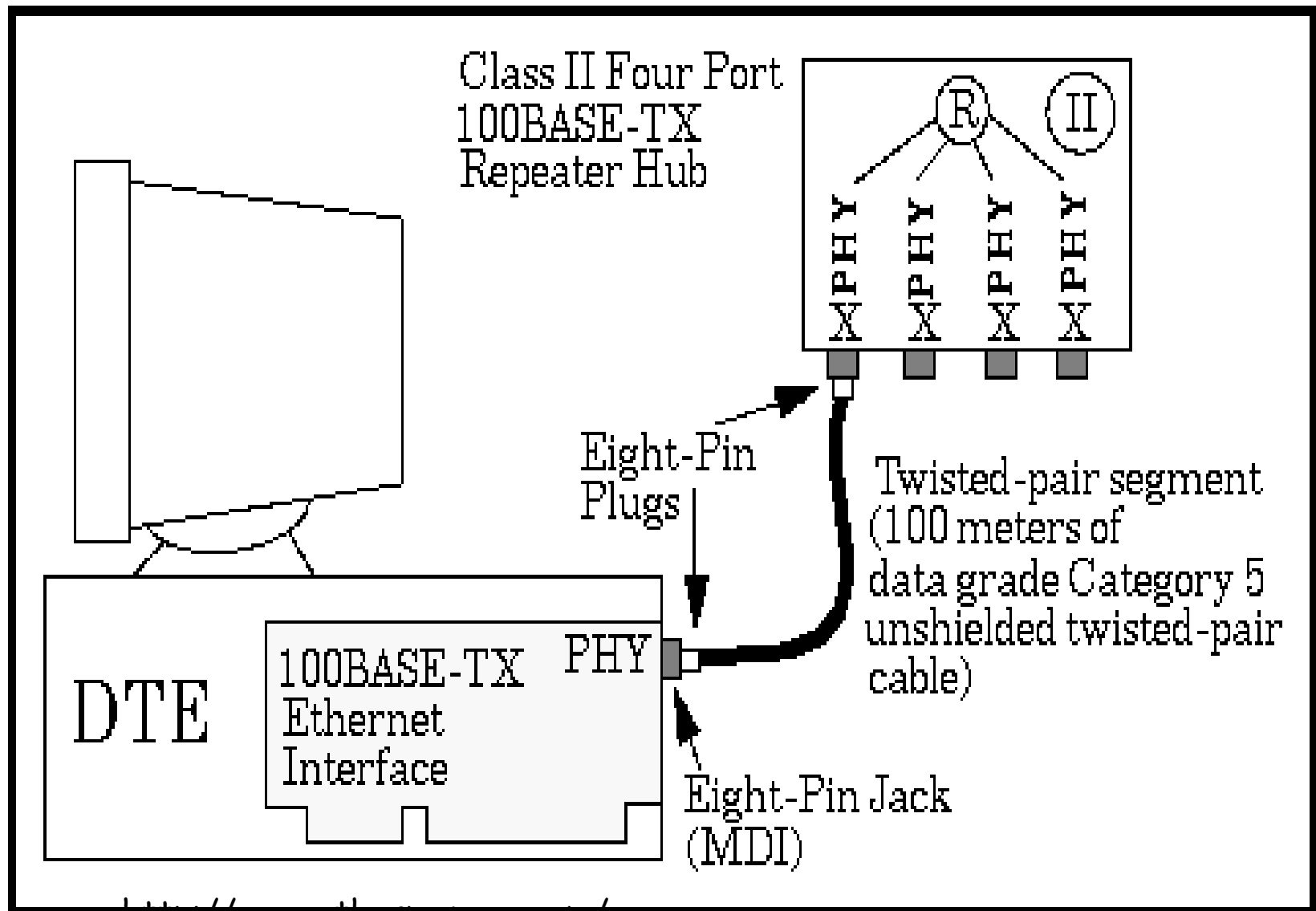
TOP:



FRONT:

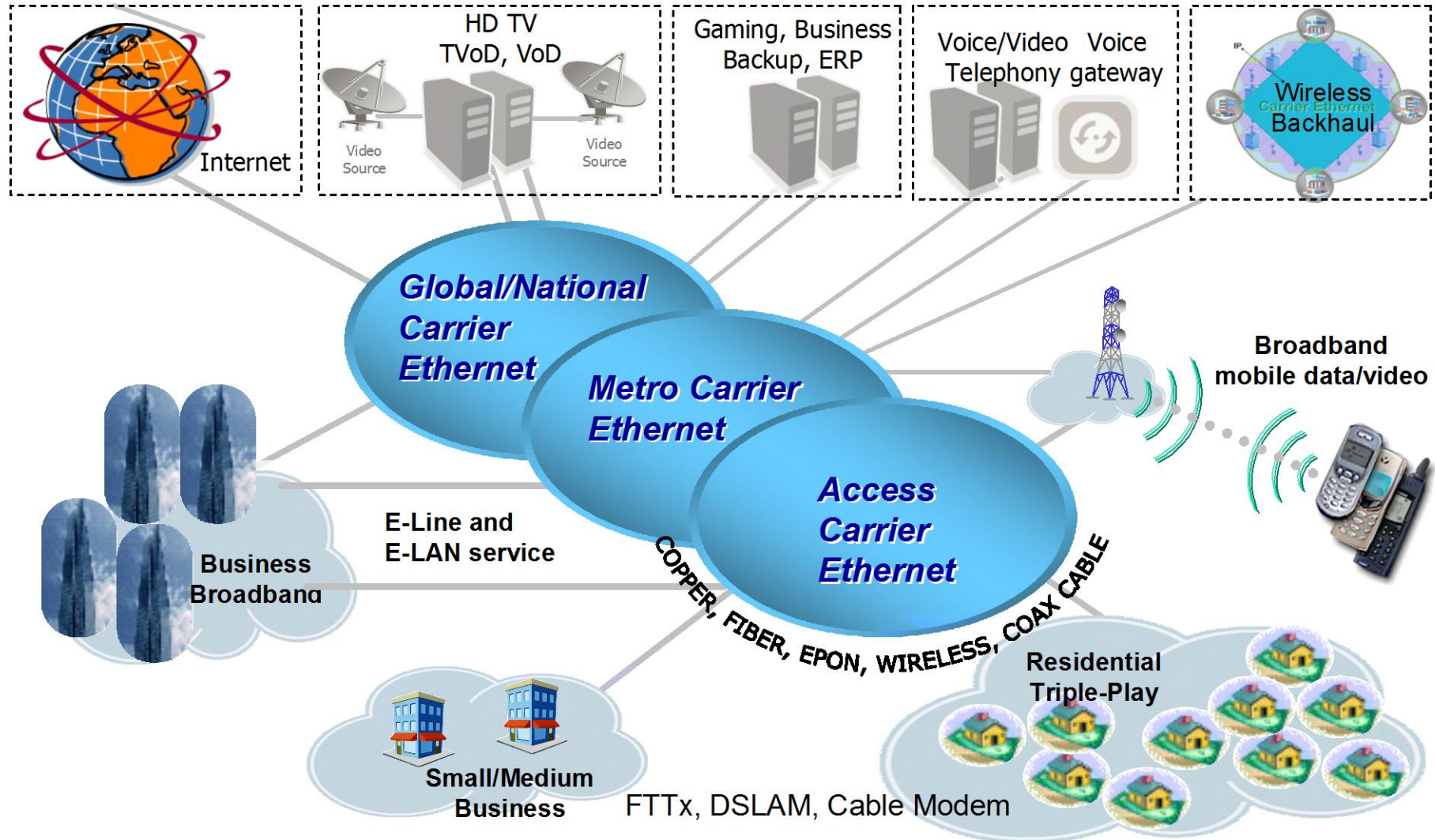


100Base-TX



Carrier Ethernet Access Expansion

Bringing vastly extended scalability for business and residential users



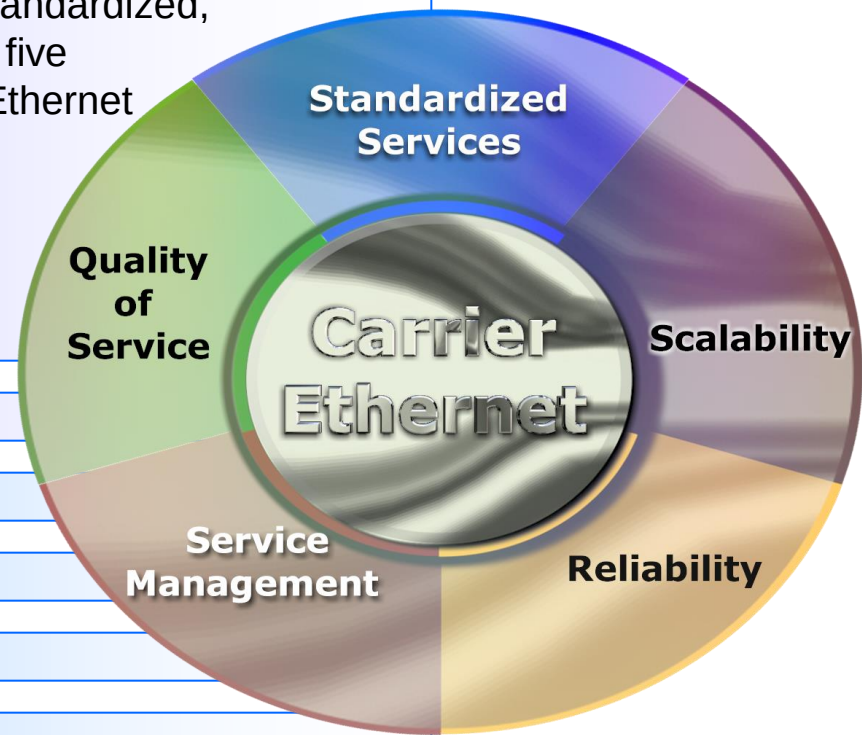
Carrier Ethernet Defined

Carrier Ethernet

- Carrier Ethernet is a ubiquitous, standardized, carrier-class **SERVICE** defined by five attributes that distinguish Carrier Ethernet from familiar LAN based Ethernet
- It brings the compelling business benefit of the Ethernet cost model to achieve significant savings

Carrier Ethernet Attributes

- **Standardized Services**
- **Scalability**
- **Service Management**
- **Reliability**
- **Quality of Service**



Camada de Enlace

5.1 Introdução e serviços

5.2 Detecção e correção de erros

5.3 Protocolos de acesso múltiplo

5.4 Redes Locais

- Endereçamento, ARP

- Ethernet

- Switches

- VLANs

5.5 Virtualização do enlace: MPLS

5.6 Redes de centros de dados

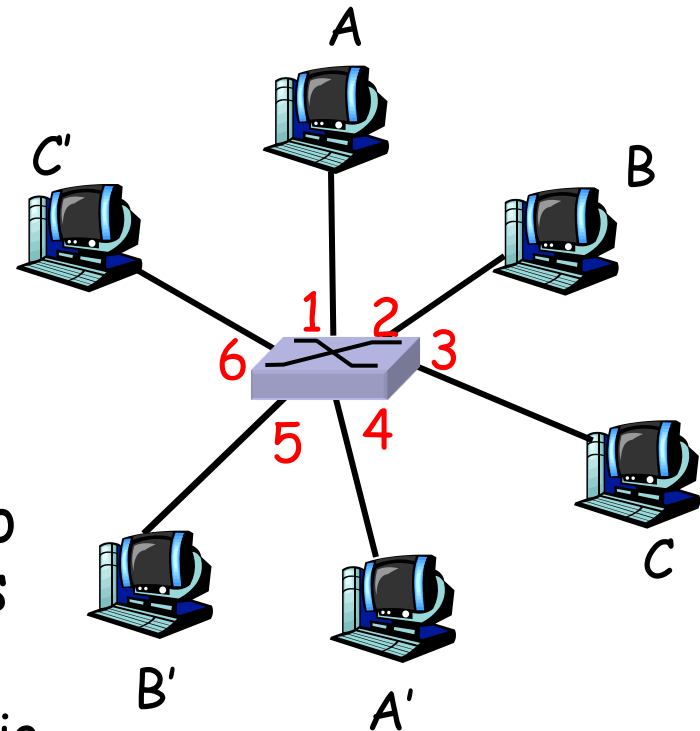
5.7 Um dia na vida de uma solicitação de página Web

Comutador (switch) Ethernet

- Dispositivo da camada de enlace: têm papel ativo
 - armazena e retransmite quadros Ethernet
 - examina o cabeçalho do quadro e **seletivamente** encaminha o quadro para um ou mais links de saída, usa o CSMA/CD para acessar o segmento
- transparente
 - hospedeiros ignoram a presença dos *switches*
- *plug-and-play, self-learning (autodidatas)*
 - os comutadores não necessitam ser configurados

Switch: permite múltiplas transmissões simultâneas

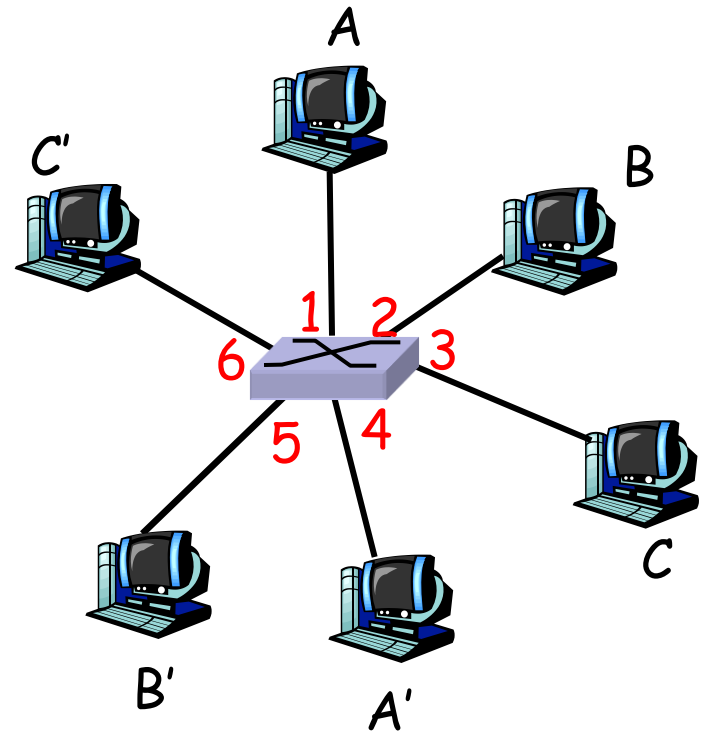
- hospedeiros têm conexão direta e dedicada para o switch
- os switches armazenam quadros
- o protocolo Ethernet é usado em cada link de entrada, mas não há colisões; full duplex
 - cada link é o seu próprio domínio de colisão
- **comutação:** A-para-A' e B-para-B' simultaneamente, sem colisões



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

Tabela de repasse do switch

- P: como é que 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**, cada entrada contém:
 - (endereço MAC do hospedeiro, interface para alcançar o hospedeiro, carimbo de tempo)
 - parece uma tabela de repasse!
- P: como são criadas e mantidas as entradas na tabela de comutação?
 - há algo como um protocolo de roteamento?



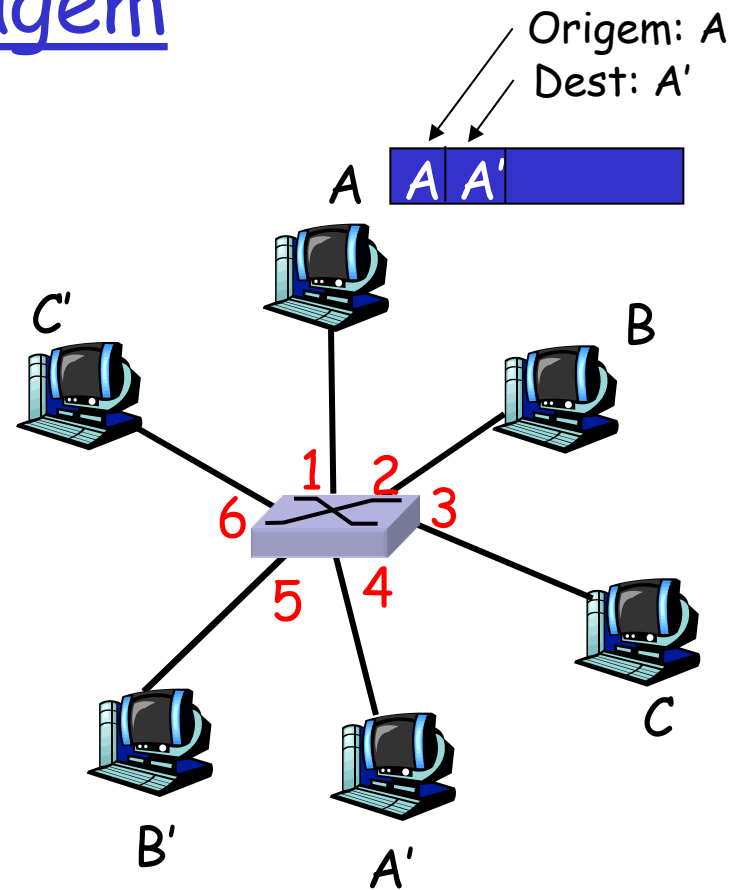
switch with six interfaces
(1,2,3,4,5,6)

Switch: autoaprendizagem

□ switch **aprende** quais hospedeiros podem ser alcançados através de quais interfaces

- quando um quadro é recebido, o switch "aprende" a localização do transmissor: segmento LAN de entrada
- registra o par transmissor/localização na tabela de comutação

Tabela de comutação
(inicialmente vazia)



end MAC	interface	TTL
A	1	60

Switch: filtragem/repassse de quadros

Quando o quadro é recebido em um comutador:

registra o link de entrada, endereço MAC do hospedeiro
transmissor

indexa a tabela de comutação usando o endereço MAC do destino

if entrada encontrada para o destino

then{

if dest estiver no segmento de onde veio o quadro

then descarta o quadro

else repassa o quadro na interface indicada

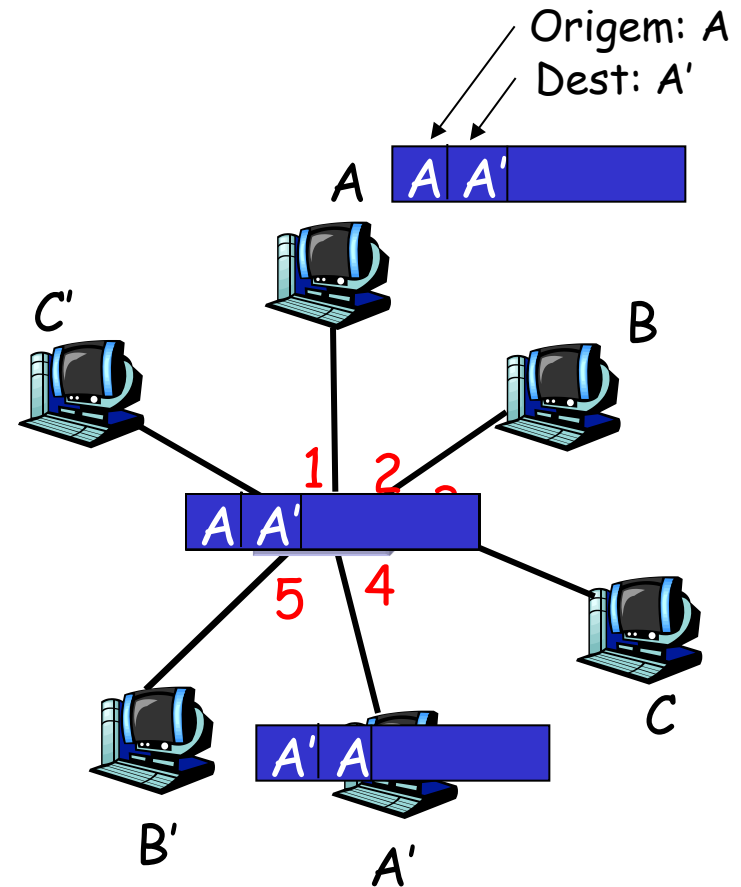
}

else usa inundação

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

Exemplo de auto aprendizagem e repassse

- destino do quadro, A', com localização desconhecida: **inundação**
- ? local do destino A conhecido: **transmissão seletiva**

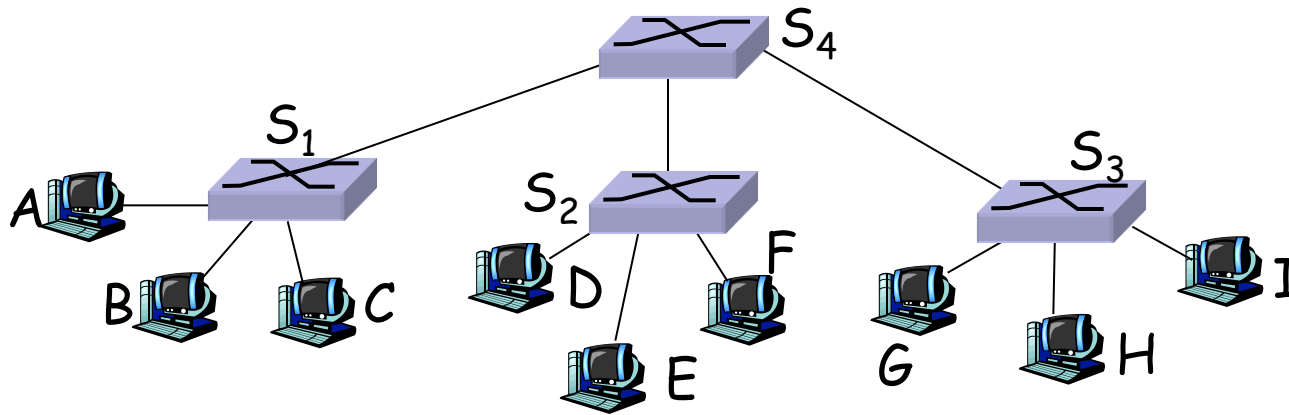


end MAC	interface	TTL
A	1	60
A'	4	60

Tabela de comutação
(inicialmente vazia)

Interligação de comutadores

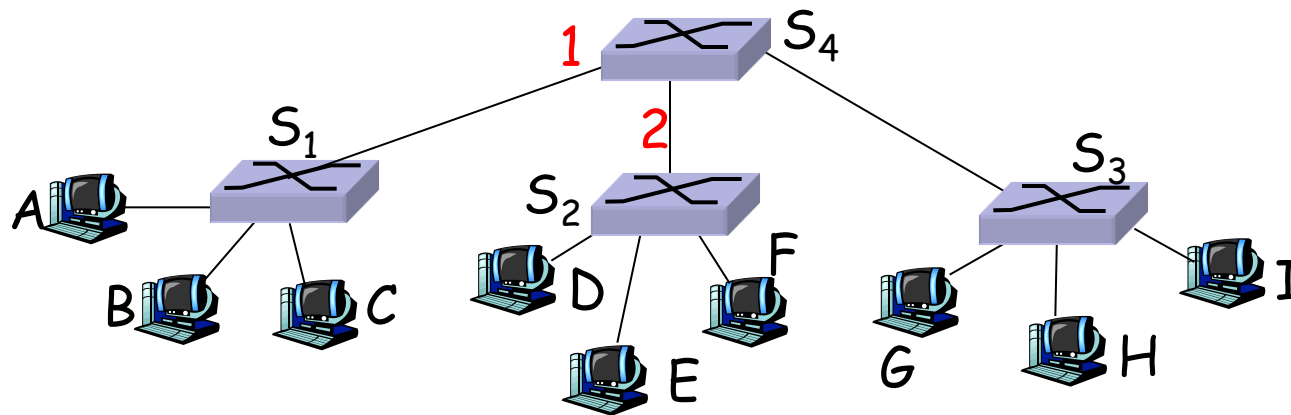
□ Podemos interligar comutadores (*switches*)



- ? **P:** ao transmitir de A para G - como S₁ sabe que deve repassar o quadro destinado a G via S₄ e S₃?
- ? **R:** autoaprendizado! (funciona exatamente da mesma forma do que no caso de um único comutador!)

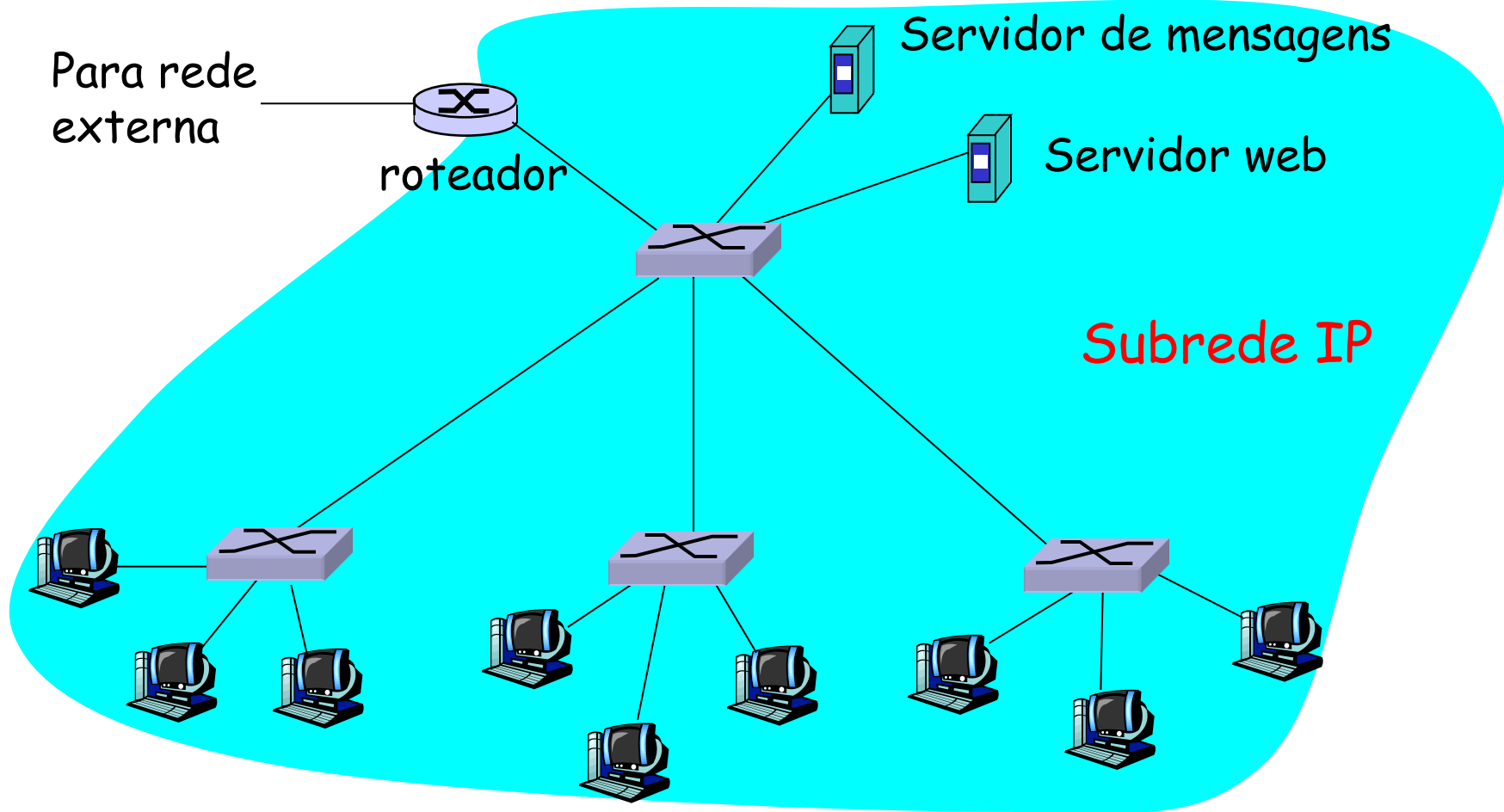
Exemplo de autoaprendizado com múltiplos comutadores

Suponha que C envia quadro para I, I responde para C



- ❓ P: mostre as tabelas de comutação e repasse de pacotes em S₁, S₂, S₃ e S₄

Rede institucional



Comutadores x Roteadores

- ambos são dispositivos do tipo armazena-e-repassa

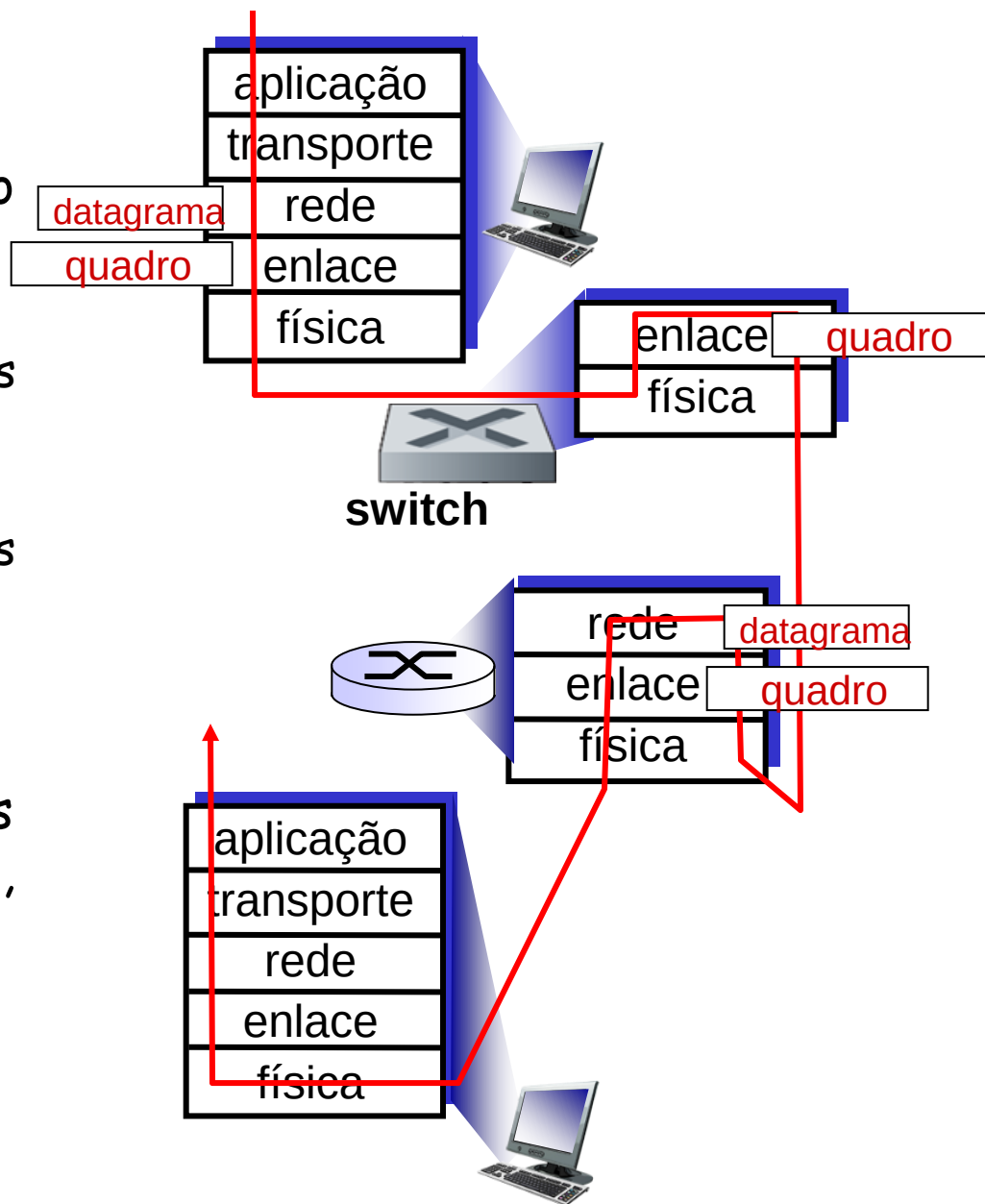
- **roteadores:** dispositivos da camada de rede (examinam os cabeçalhos da camada de rede)

- **comutadores:** são dispositivos da camada de enlace

- ambos possuem tabelas de repasse:

- **roteadores:** obtém tabelas usando alg. de roteamento, endereços IP

- **comutadores:** obtém tabela usando inundação, aprendizado, endereços MAC



Camada de Enlace

5.1 Introdução e serviços

5.2 Detecção e correção de erros

5.3 Protocolos de acesso múltiplo

5.4 Redes Locais

- Endereçamento, ARP

- Ethernet

- Switches

- VLANs

5.5 Virtualização do enlace: MPLS

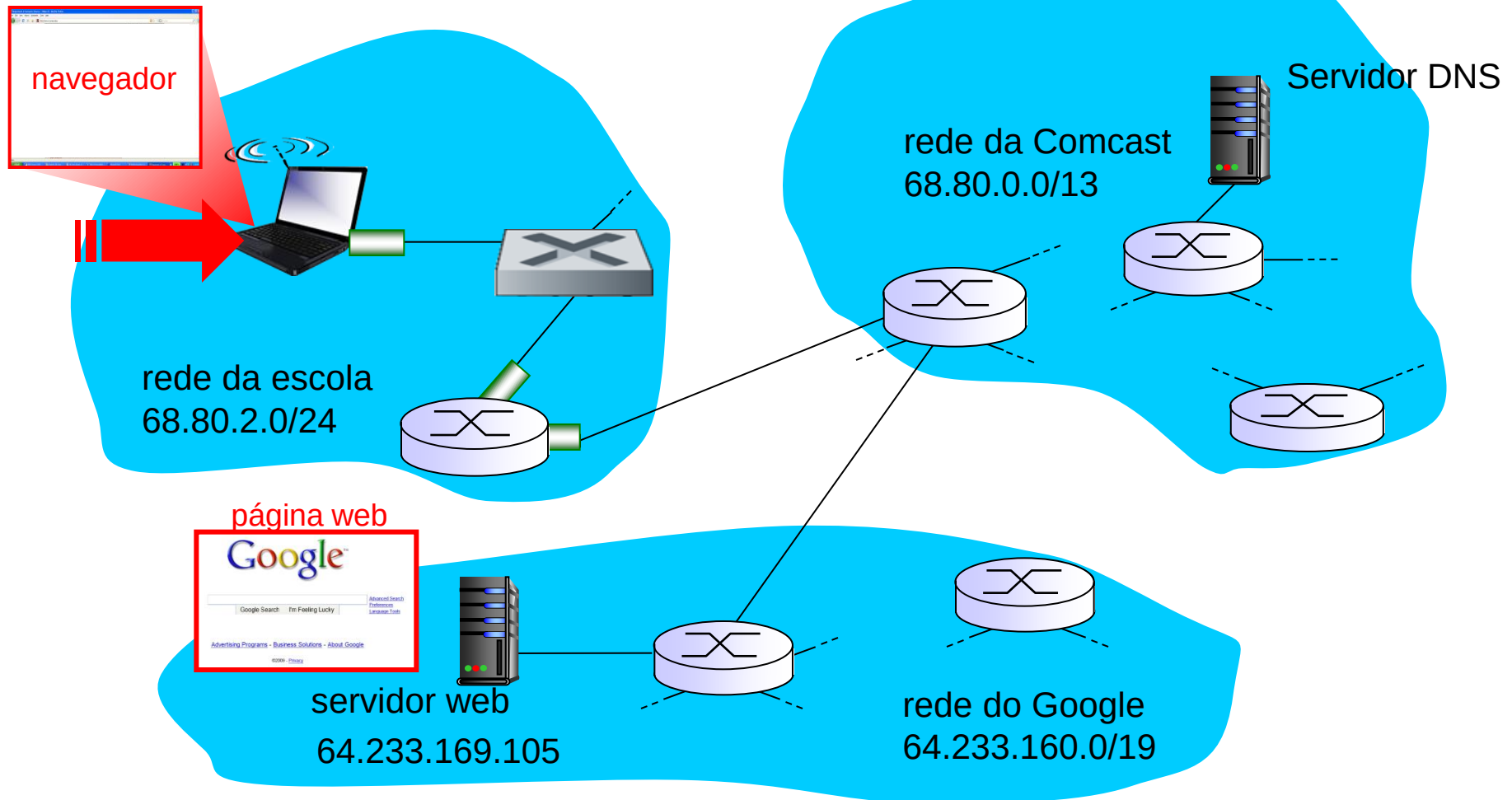
5.6 Redes de centros de dados

5.7 Um dia na vida de uma solicitação de página Web

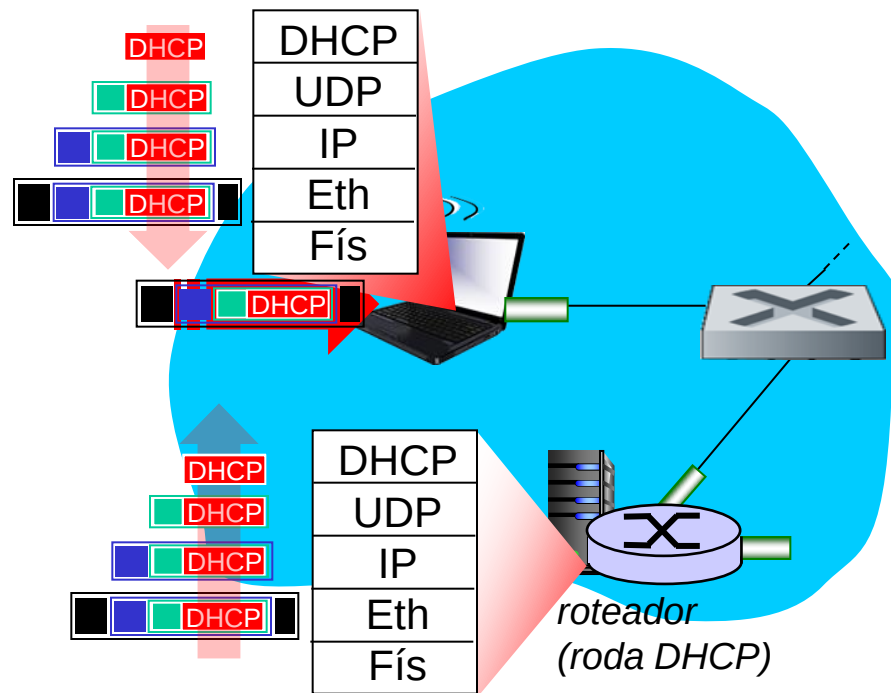
Síntese: um dia na vida de um pedido web

- ❖ jornada completa atravessando toda a pilha de protocolos!
 - aplicação, transporte, rede, enlace
- ❖ colocando tudo junto: síntese!
 - *objetivo*: identificar, revisar, entender os protocolos (em todas as camadas) envolvidos em um cenário aparentemente simples: solicitação de uma página web
 - *cenário*: estudante conecta laptop à rede do campus, solicita/recebe www.google.com

Um dia na vida: cenário

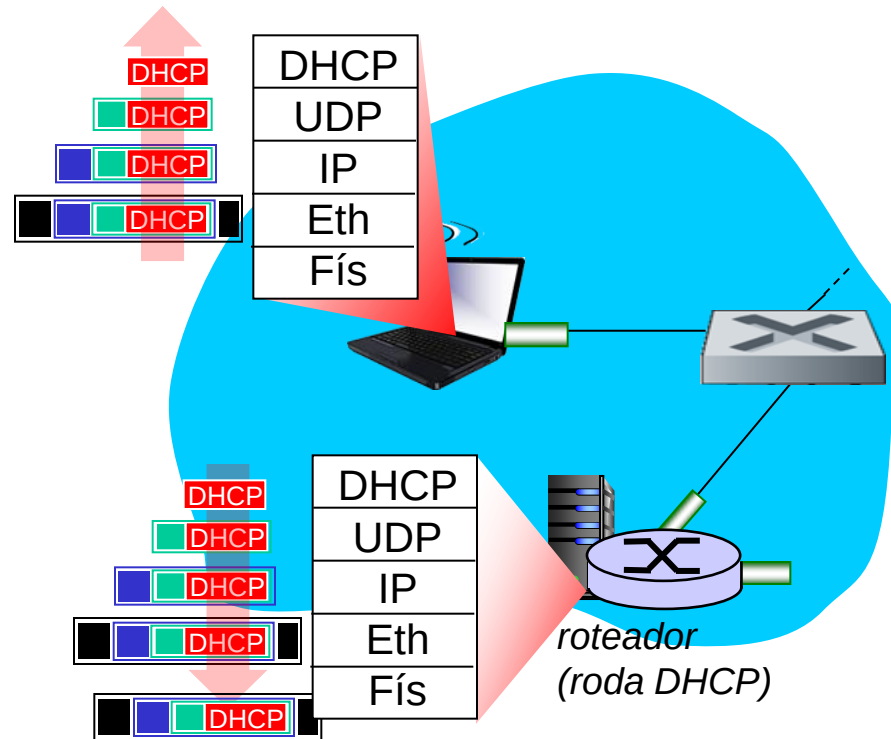


Um dia na vida... conectando à Internet



- ❖ o laptop necessita obter o seu endereço IP, o endereço do primeiro roteador e endereço do servidor DNS: usa o **DHCP**.
- ❖ Solicitação DHCP **encapsulada** em **UDP**, encapsulada no **IP**, encapsulada no **802.3 Ethernet**
- ❖ quadro Ethernet **difundido** (dest: FFFFFFFFFFFFFFFF) na LAN, é recebido pelo roteador que executa o servidor **DHCP**
- ❖ Ethernet **demultiplexado** para IP, demultiplexado para UDP e demultiplexado para DHCP

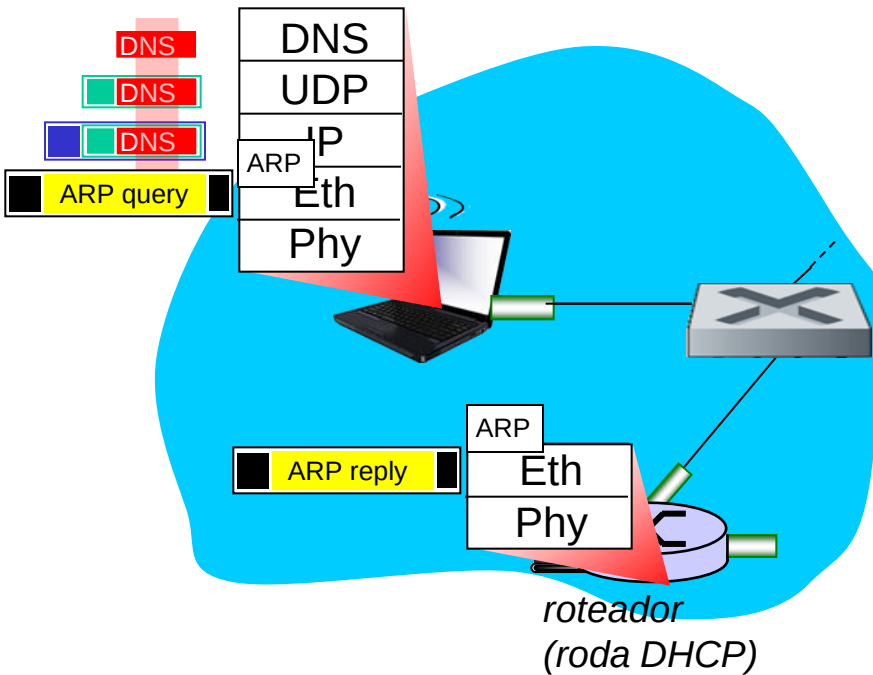
Um dia na vida... conectando à Internet



- servidor DHCP prepara **ACK DHCP** contendo endereço IP do cliente, endereço IP do primeiro roteador, nome e endereço IP do servidor DNS
- ❖ encapsulamento no servidor DHCP, quadro repassado (**aprendizado do switch**) através da LAN, demultiplexação no cliente
- ❖ cliente DHCP recebe a resposta ACK DHCP

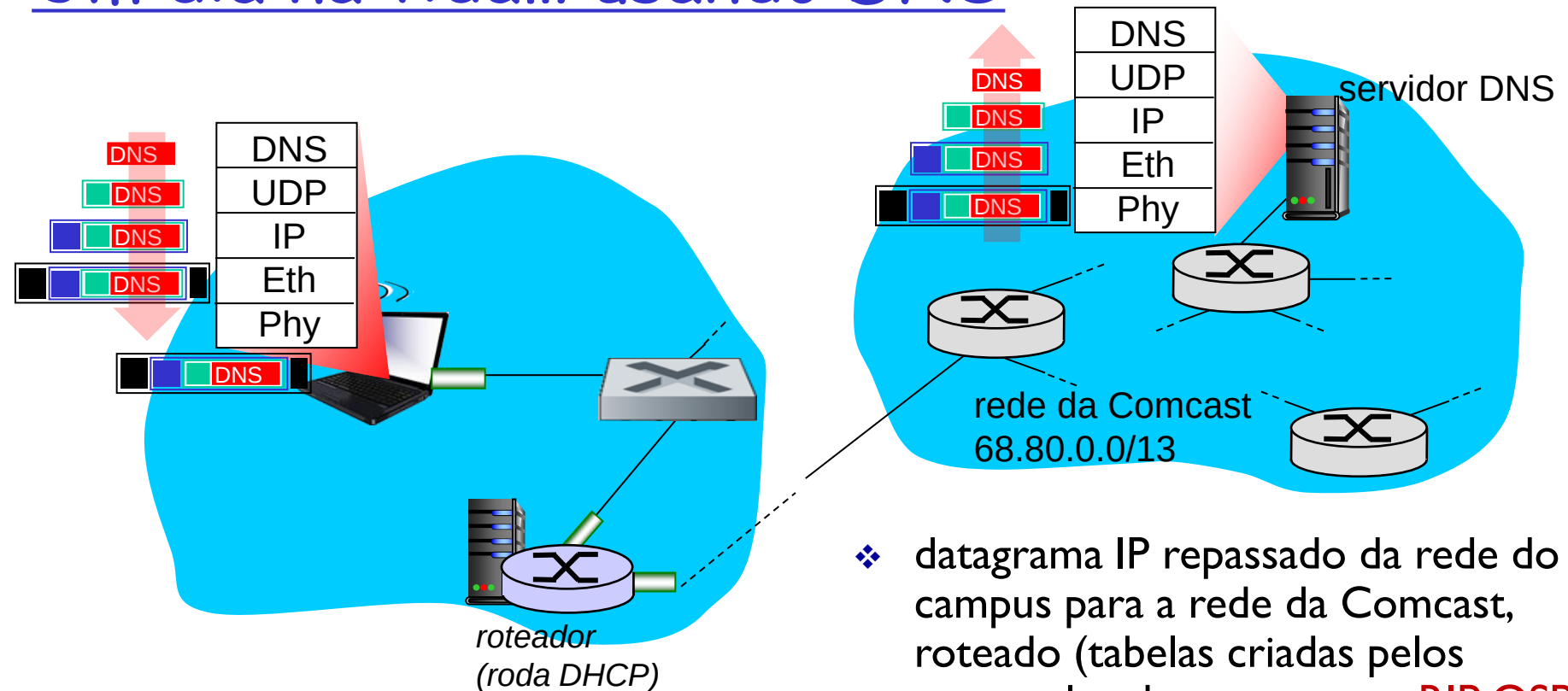
Cliente agora possui um endereço IP, conhece o nome e end. do servidor DNS, e o endereço IP do seu primeiro roteador

Um dia na vida... ARP (antes do DNS, antes do HTTP)



- ❖ antes de enviar pedido **HTTP**, necessita o endereço IP de **www.google.com**: **DNS**
- ❖ consulta DNS criada, encapsulada no UDP, encapsulada no IP, encapsulada no Eth. Para enviar quadro ao roteador, necessita o endereço MAC da interface do roteador: **ARP**
- ❖ **consulta ARP** difundida, recebida pelo roteador, que responde com uma **ARP reply** dando o endereço MAC da interface do roteador
- ❖ o cliente agora conhece o endereço MAC do primeiro roteador; podendo agora enviar o quadro contendo a consulta DNS

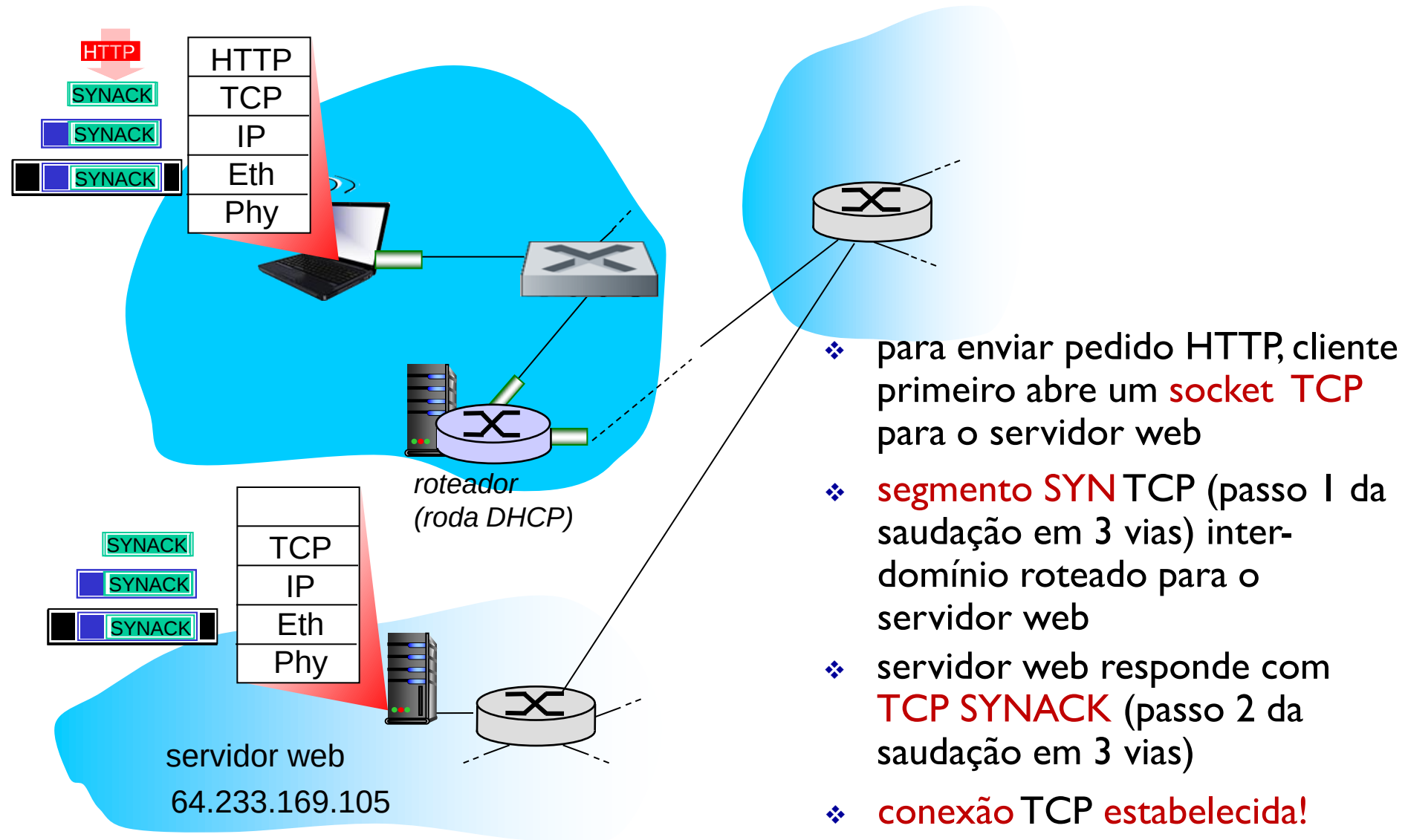
Um dia na vida... usando DNS



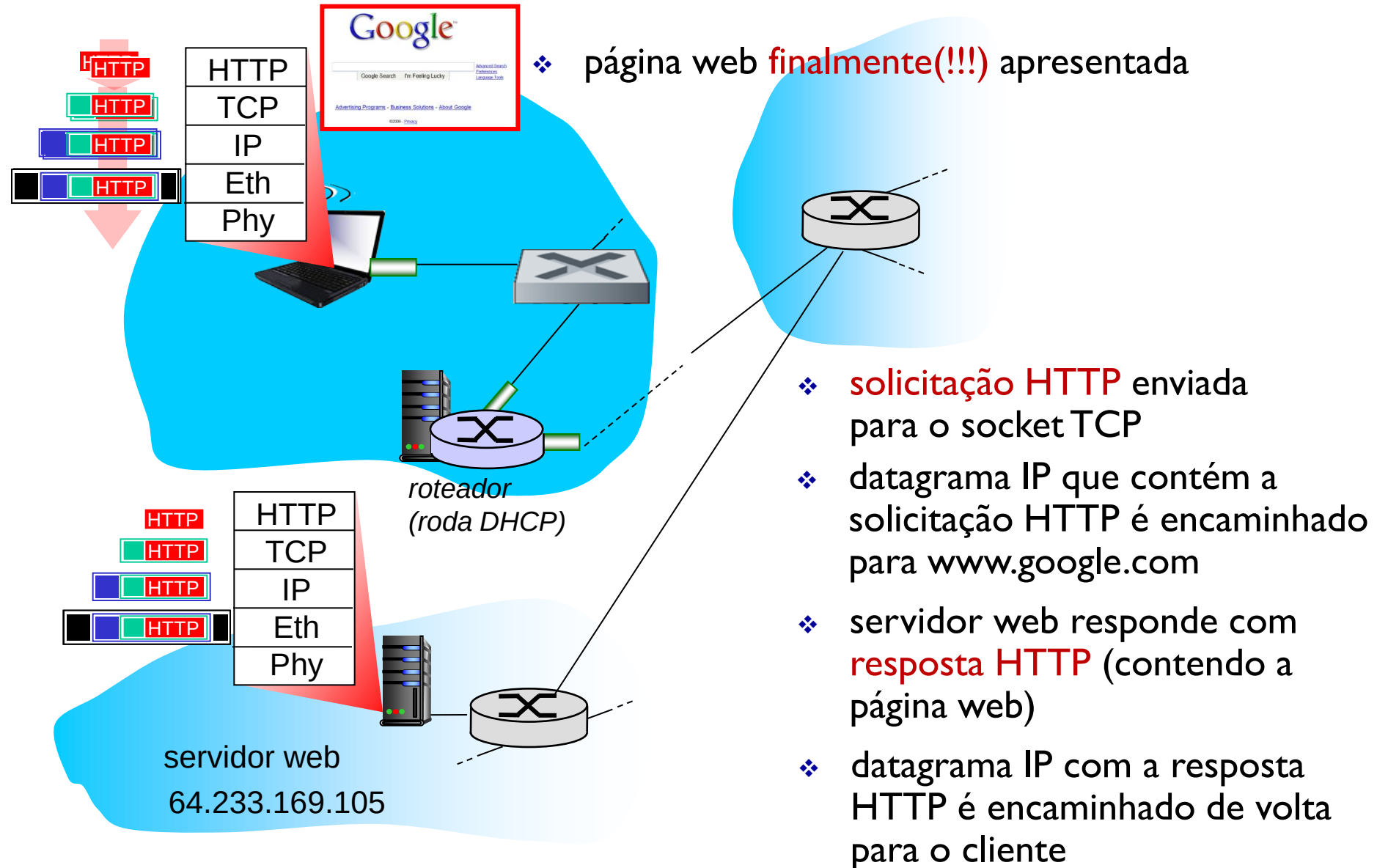
- ❖ datagrama IP contém consulta DNS encaminhada através do switch LAN do cliente até o primeiro roteador

- ❖ datagrama IP repassado da rede do campus para a rede da Comcast, roteado (tabelas criadas pelos protocolos de roteamento **RIP**, **OSPF**, **IS-IS** e/ou **BGP**) para o servidor DNS
- ❖ demultiplexado pelo servidor DNS
- ❖ servidor DNS responde ao cliente com o endereço IP de www.google.com

Um dia na vida... conexão TCP transportando HTTP



Um dia na vida... solicitação/resposta HTTP



Capítulo 5: Resumo

- princípios por trás dos serviços da camada de enlace de dados:
 - detecção, correção de erros
 - compartilhamento de canal de difusão: acesso múltiplo
 - endereçamento da camada de enlace
- instanciação e implementação de diversas tecnologias de camada de enlace
 - Ethernet
 - LANs comutadas, VLANs
 - redes virtualizadas como camada de enlace: MPLS
- síntese: um dia na vida de uma solicitação web