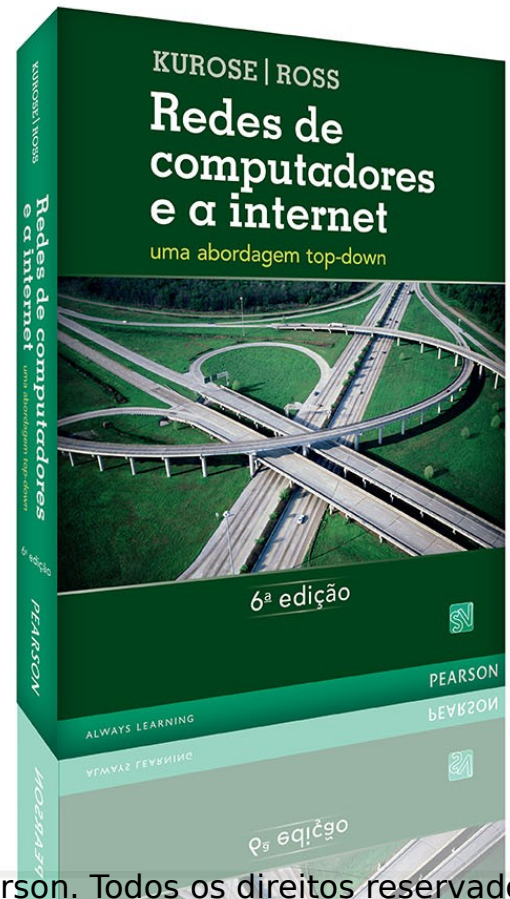


# Capítulo 5

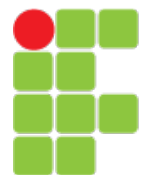
## Camada de enlace: enlaces, redes de acesso e redes locais



# Os serviços fornecidos pela camada de enlace

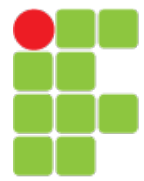
Entre os serviços que podem ser oferecidos por um protocolo da camada de enlace, estão:

- Enquadramento de dados.
- Acesso ao enlace (MAC – *medium access control*).
- Entrega confiável.
- Detecção e correção de erros.



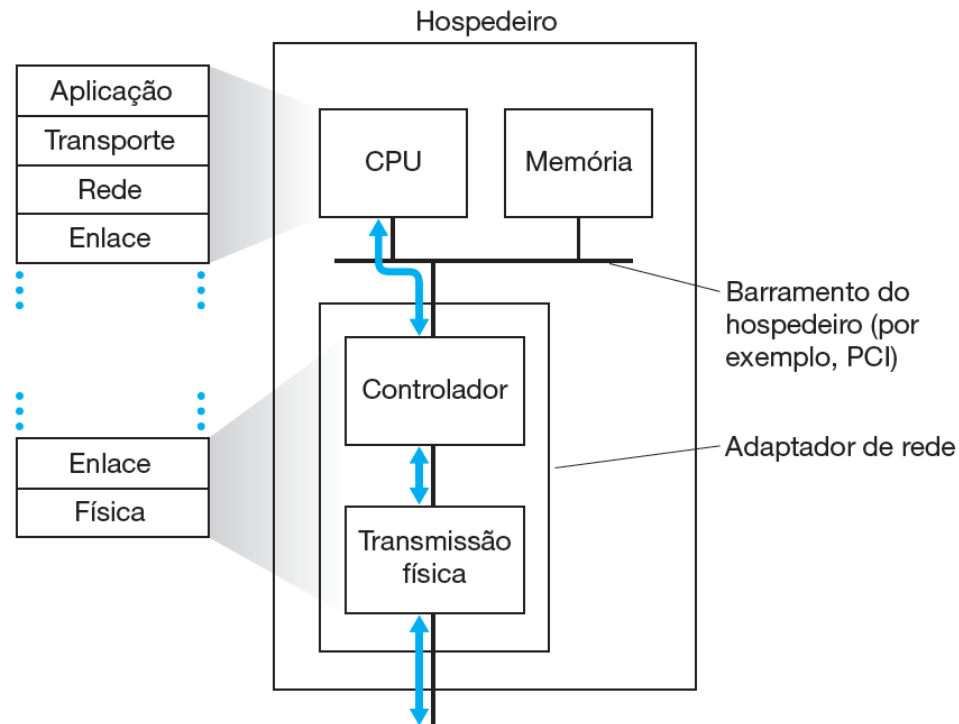
# Onde a camada de enlace é implementada?

- A figura a seguir mostra a arquitetura típica de um hospedeiro.
- Na maior parte, a camada de enlace é implementada em um **adaptador de rede**, às vezes também conhecido como **placa de interface de rede (NIC)**.
- No núcleo do adaptador de rede está o controlador da camada de enlace que executa vários serviços da camada de enlace.
- Dessa forma, muito da funcionalidade do controlador da camada de enlace é realizado em hardware.



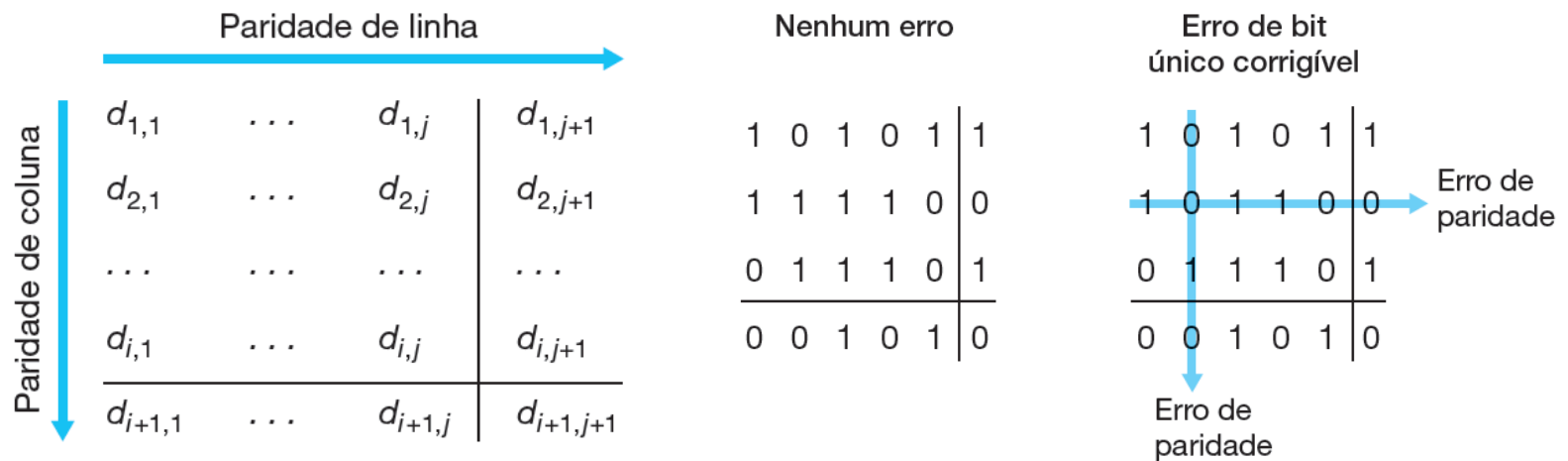
# Onde a camada de enlace é implementada?

- Adaptador de rede: seu relacionamento com o resto dos componentes do hospedeiro e a funcionalidade da pilha de protocolos



# Verificações de paridade

- Talvez a maneira mais simples de detectar erros seja utilizar um único bit de paridade.
- A figura abaixo mostra uma generalização bidimensional do esquema de paridade de bit único.

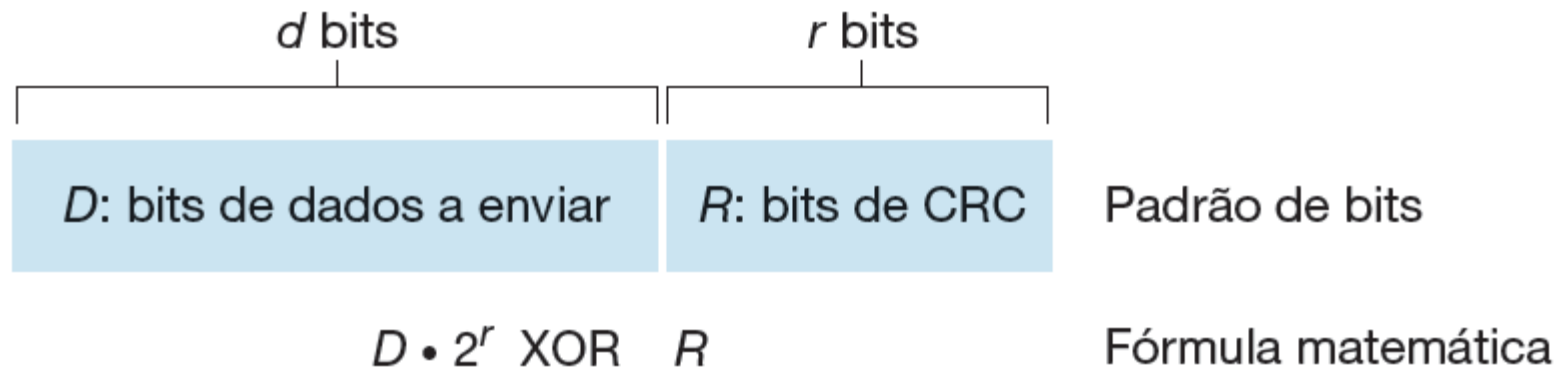


# Métodos de soma de verificação

- Um método simples de soma de verificação é somar os inteiros de  $k$  bits e usar o total resultante como bits de detecção de erros.
- O complemento de 1 dessa soma forma, então, a soma de verificação da Internet, que é carregada no cabeçalho do segmento.
- No IP, a soma de verificação é calculada sobre o cabeçalho IP.
- Métodos de soma de verificação exigem relativamente pouca sobrecarga no pacote.

# Verificação de redundância cíclica (CRC)

- Uma técnica de detecção de erros muito usada nas redes de computadores de hoje é baseada em **códigos de verificação de redundância cíclica (CRC)**.
- Códigos de CRC também são conhecidos como **códigos polinomiais**.



# Enlaces e protocolos de acesso múltiplo

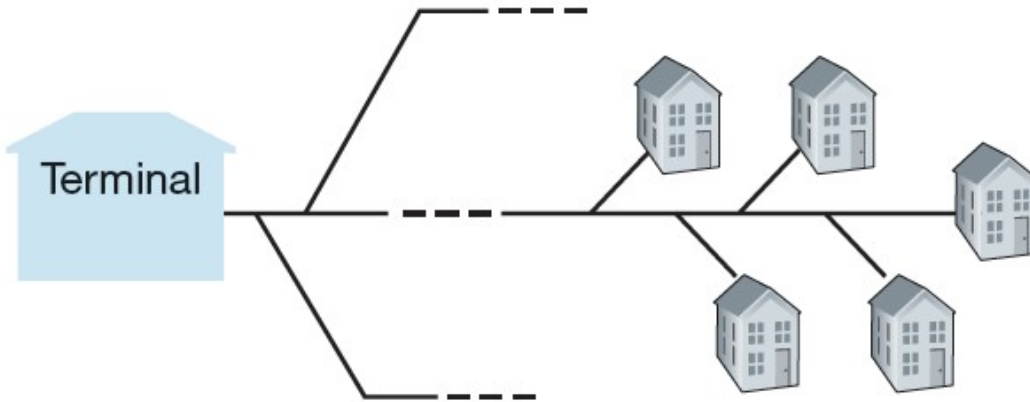
- Um **enlace ponto a ponto** consiste em um único remetente em uma extremidade do enlace e um único receptor na outra. **Exemplo?**
- O **enlace de difusão**, pode ter vários nós remetentes e receptores, todos conectados ao mesmo canal de transmissão único e compartilhado. **Exemplo?**
- **Protocolos de acesso múltiplo** — através dos quais os nós regulam sua transmissão pelos canais de difusão compartilhados.



# Enlaces e protocolos de acesso múltiplo

- Vários canais de acesso múltiplo

Compartilhado com fio  
(por exemplo, rede de acesso a cabo)

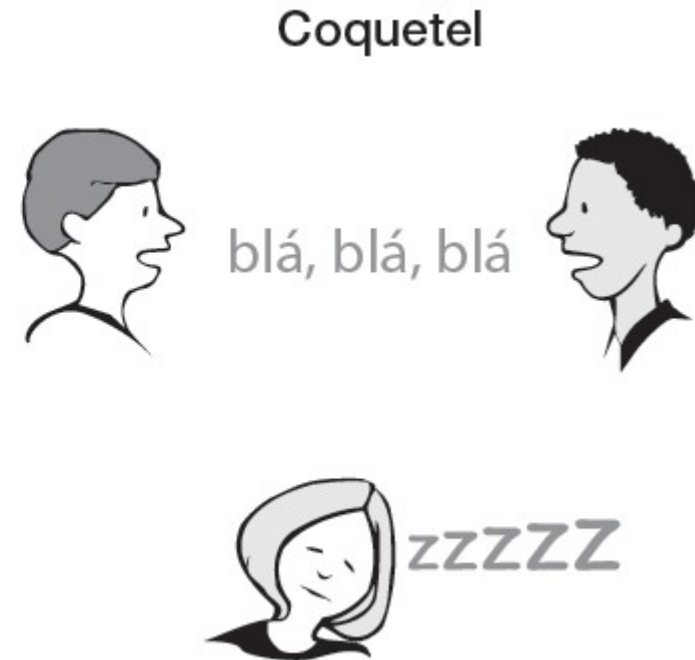
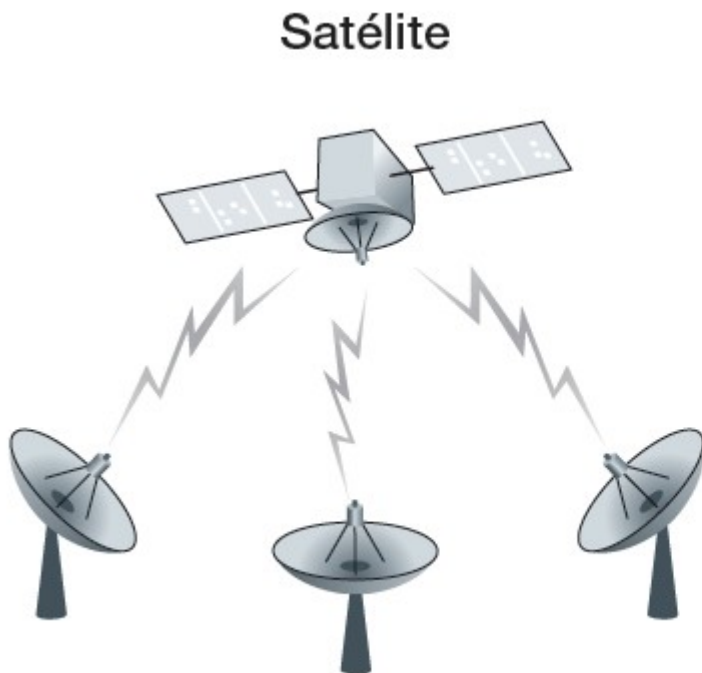


Compartilhado sem fio  
(por exemplo, Wi-Fi)



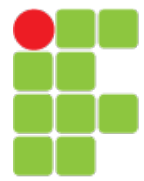
# Enlaces e protocolos de acesso múltiplo

- Vários canais de acesso múltiplo



# Enlaces e protocolos de acesso múltiplo - características desejáveis:

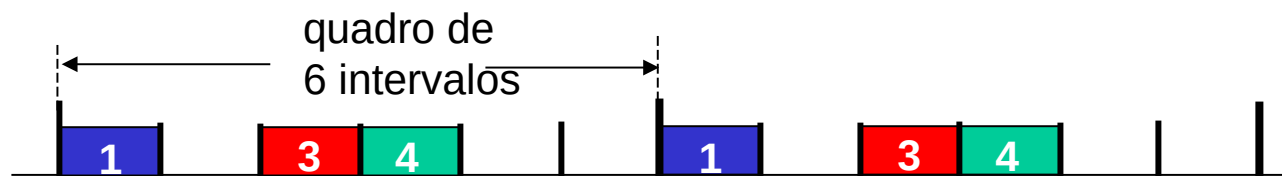
- Quando apenas um nó tem dados a enviar, envia à taxa máxima
- Quando  $M$  nós tem dados a enviar, divide-se igualmente a taxa média
- O protocolo é descentralizado, isto é, não há um mestre que represente um único ponto de falha
- O protocolo é simples para que sua implementação seja barata



# Protocolos MAC de particionamento de canal: TDMA

## TDMA: Time Division Multiple Access

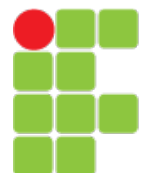
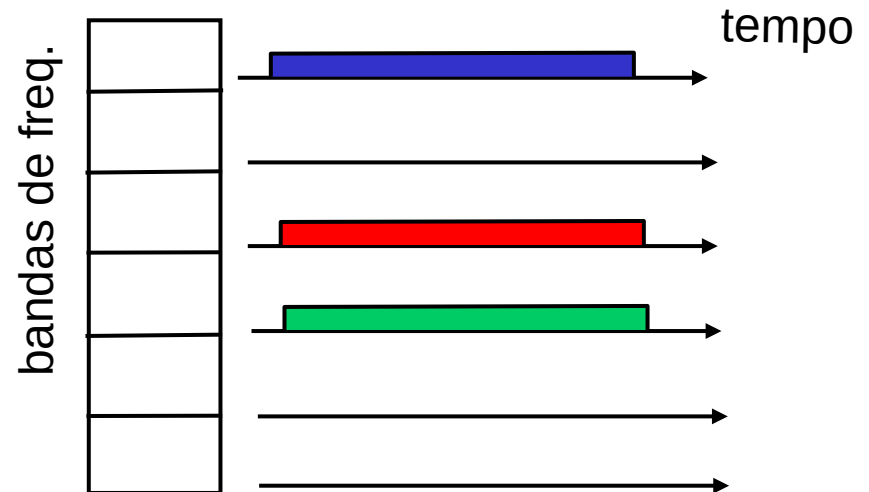
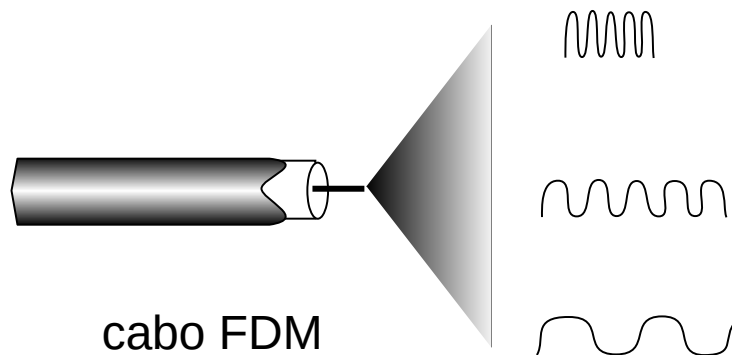
- acesso ao canal em “rodadas”
- cada estação recebe intervalo de tamanho fixo (tamanho = tempo transm. pacote) a cada rodada
- intervalos não usados ficam ociosos
- exemplo: LAN de 6 estações, 1, 3, 4 têm pacote, intervalos 2, 5, 6 ociosos



# Protocolos MAC de particionamento de canal: FDMA

## FDMA: Frequency Division Multiple Access

- espectro do canal dividido em bandas de frequência
- cada estação recebe banda de frequência fixa
- tempo de transmissão não usado nas bandas de frequência fica ocioso
- exemplo: LAN de 6 estações, 1, 3, 4 têm pacote, bandas de frequência 2, 5, 6 ociosas



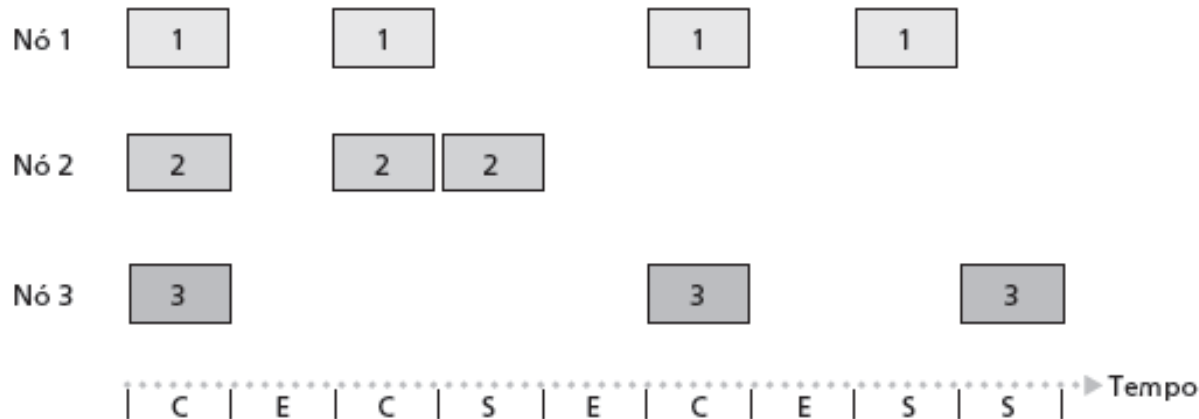
# Protocolos de divisão de canal

- O **protocolo FDM** divide o canal de  $R$  bits/s em frequências diferentes e reserva cada frequência a um dos  $N$  nós, criando, desse modo,  $N$  canais menores de  $R/N$  bits/s a partir de um único canal maior de  $R$  bits/s.
- O **protocolo de acesso múltiplo por divisão de código (CDMA)** atribui um código diferente a cada nó.
- Se os códigos forem escolhidos com cuidado, as redes CDMA terão a maravilhosa propriedade de permitir que nós diferentes transmitam simultaneamente.

# Protocolos de acesso aleatório

- Com um protocolo de acesso aleatório, um nó transmissor sempre transmite à taxa total do canal, isto é,  $R$  bits/s.
- O que é: compartimentado, percebe colisões.
- O *slotted ALOHA* é altamente (não completamente) descentralizado.
- Funciona bem quando há apenas um nó ativo.
- Com múltiplos nós ativos a eficiência máxima do protocolo é dada por  $1/e = 0,37$  (37 %).

# Slotted ALOHA



## Prós

- único nó ativo pode transmitir continuamente na velocidade plena do canal
- altamente descentralizado: somente intervalos nos nós precisam estar em sincronismo
- simples

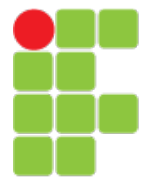
## Contras

- colisões, intervalos desperdiçados
- intervalos ociosos
- nós podem ser capazes de detectar colisão em menos tempo do que para transmitir pacote
- sincronismo de clock



# CSMA (acesso múltiplo com detecção de portadora - *Carrier Sense Multiple Access* )

- Especificamente, há duas regras importantes que regem a conversação educada entre seres humanos:
- *Ouçã antes de falar.* Se uma pessoa estiver falando, espere até que ela tenha terminado. No mundo das redes, isso é denominado **detecção de portadora** — um nó ouve o canal antes de transmitir.
- *Se alguém começar a falar ao mesmo tempo que você, pare de falar.* No mundo das redes, isso é denominado **detecção de colisão** (portadora) — um nó que está transmitindo ouve o canal enquanto transmite.



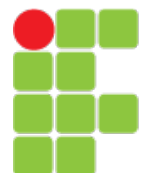
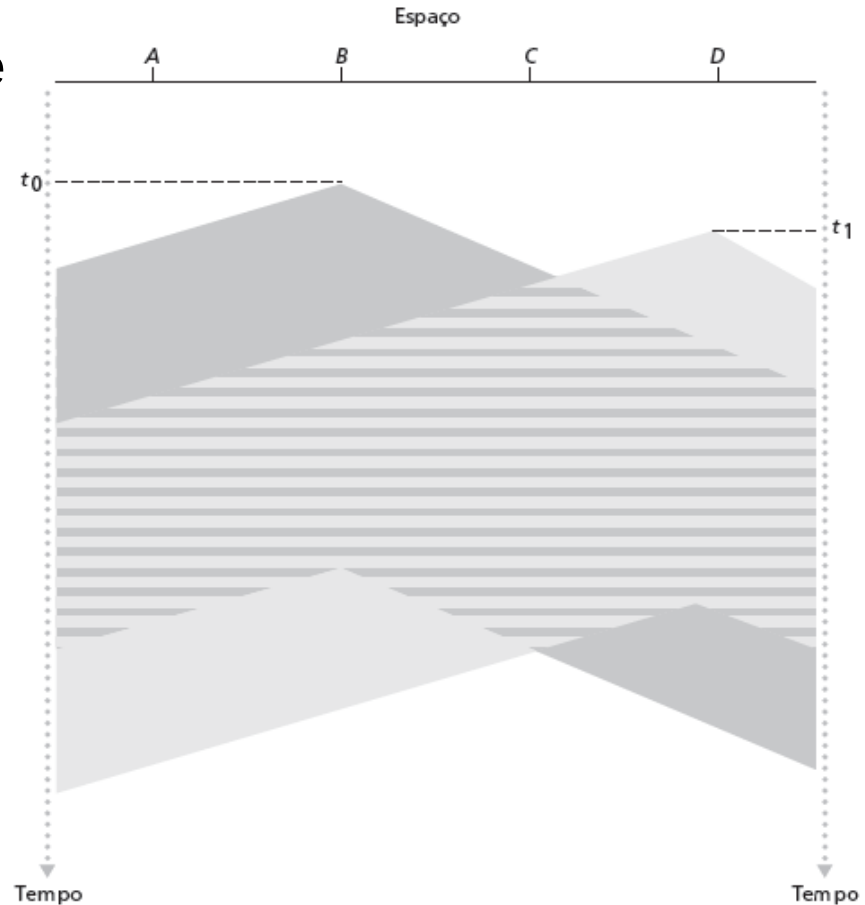
# Colisões CSMA

colisões ainda *podem* ocorrer:  
 atraso de propagação significa que dois nós podem não ouvir a transmissão um do outro

colisão:  
 tempo de transmissão de pacote inteiro desperdiçado

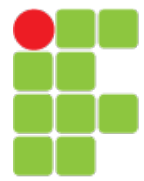
nota:  
 papel da distância & atraso de propagação determinando probabilidade de colisão (comprimento máximo de cabo)

layout espacial dos nós



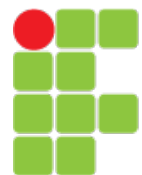
# CSMA (acesso múltiplo com detecção de portadora)

- Essas duas regras estão incorporadas na família de protocolos de **acesso múltiplo com detecção de portadora** (CSMA – *Carrier Sense Multiple Access*) e **CSMA com detecção de colisão** (CSMA/CD).
- Se todos os nós realizam detecção de portadora, por que ocorrem colisões?
- A resposta a essa pergunta pode ser ilustrada utilizando diagramas espaço/tempo.

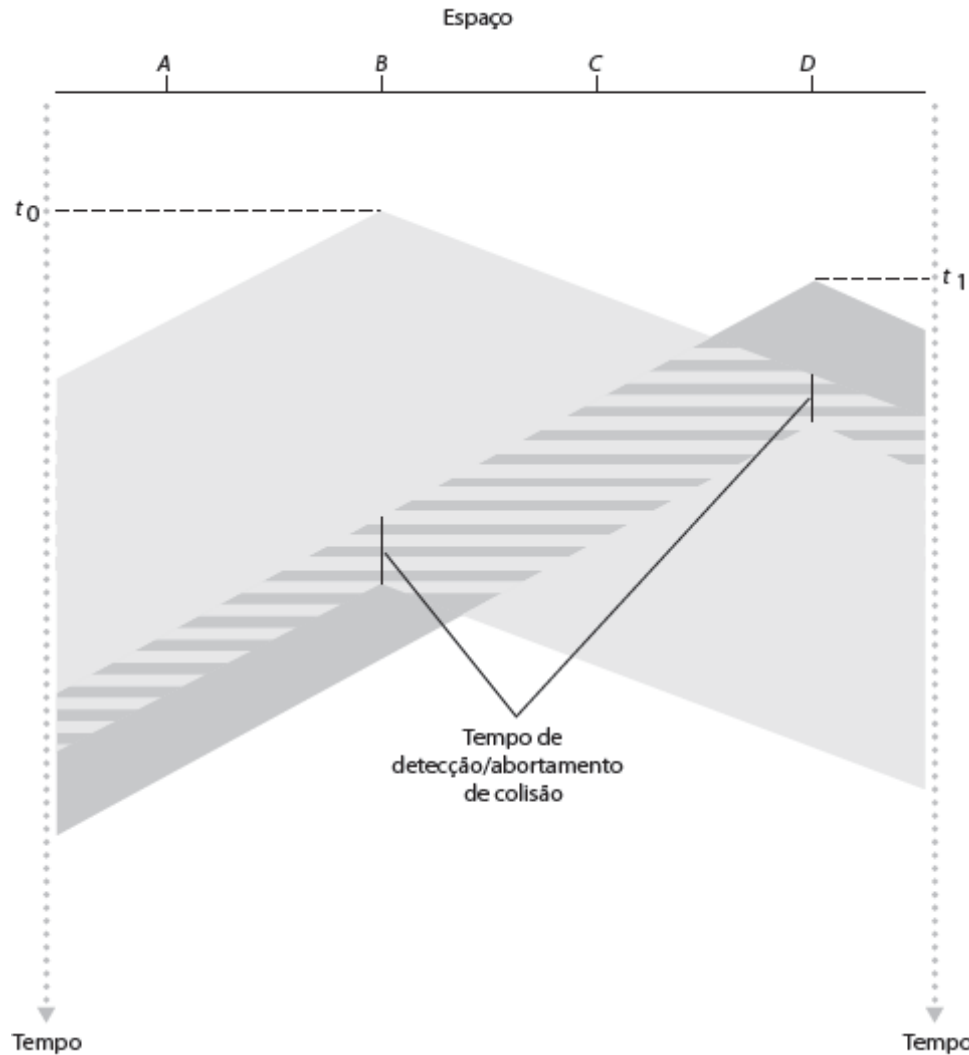


# CSMA/CD (*Collision Detection*)

- CSMA/CD:** detecção de portadora, adiada como no CSMA
- colisões *detectadas* dentro de pouco tempo
  - transmissões colidindo abortadas, reduzindo desperdício do canal
  - detecção de colisão:
    - fácil em LANs com fio: mede intensidades de sinal, compara sinais transmitidos, recebidos
    - difícil nas LANs sem fio: intensidade do sinal recebido abafada pela intensidade da transmissão local
  - analogia humana: o interlocutor educado

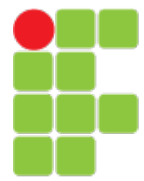


# Detecção de colisão CSMA/CD



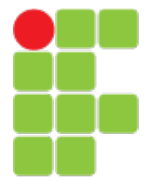
# Protocolos de revezamento

- O **protocolo de *polling*** elimina as colisões e os intervalos vazios que atormentam os protocolos de acesso aleatório, e isso permite que ele tenha uma eficiência muito maior, mas tem o nó mestre: único ponto de falha.
- No **protocolo de passagem de permissão** não há nó mestre.
- Um pequeno quadro de finalidade especial conhecido como uma **permissão** (*token*) é passado entre os nós obedecendo a uma determinada ordem fixa.



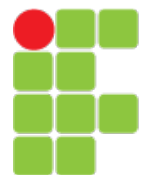
# Endereçamento MAC e ARP

- Endereço IP de 32 bits:
  - endereço da *camada de rede*
  - usado para obter datagrama até sub-rede IP de destino
- Endereço MAC (ou LAN ou físico ou Ethernet) :
  - função: *levar quadro de uma interface para outra interface conectada fisicamente (na mesma rede)*
  - *Endereço MAC de 48 bits (para maioria das LANs)*
    - queimado na ROM da NIC, às vezes também configurável por software
  - MAC – *Media Access Control* – Controle de acesso ao meio
  - ARP – *Address Resolution Protocol* – Protocolo de resolução de endereços: **MAC X IP** (rede local)



# Ethernet

- A Ethernet praticamente tomou conta do mercado de LANs com fio.
- Há muitas razões para o sucesso da Ethernet:
  1. Ela foi a primeira LAN de alta velocidade amplamente disseminada.
  2. Token ring, FDDI e ATM são tecnologias mais complexas e mais caras do que a Ethernet, o que desencorajou ainda mais os administradores na questão da mudança.

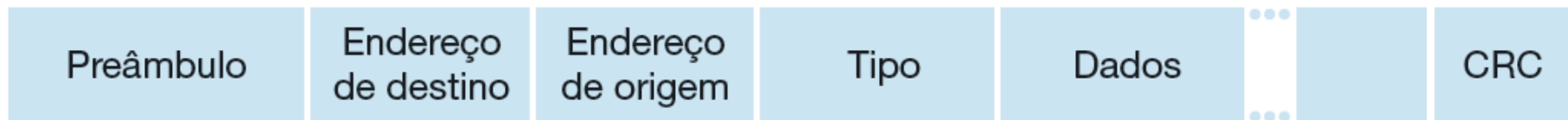




# Ethernet

3. A Ethernet sempre produziu versões que funcionavam a velocidades iguais, ou mais altas.
4. O hardware para Ethernet passou a ser mercadoria comum, de custo muito baixo.

- Estrutura do quadro Ethernet



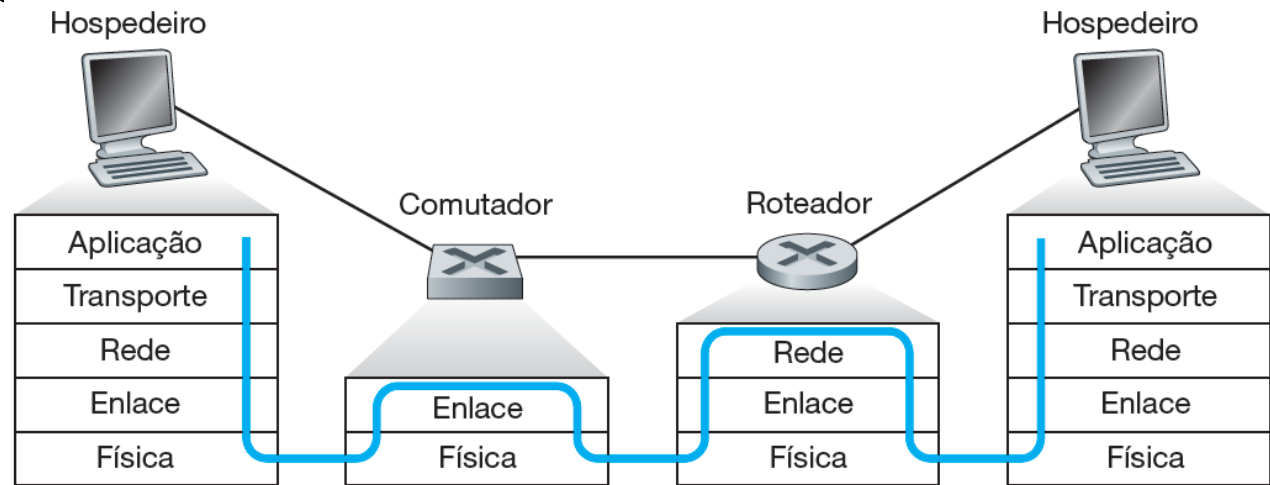
- Preâmbulo: 7 x 10101010 + 10101011: despertam e sincronizam os receptores (10 Mb/s, 100 Mb/s e 1Gb/s).
- Tipo: indica o protocolo da camada de rede: IP, ARP, IPX...
- Dados: min 46 bytes e máximo 1500 bytes.
- CRC: erro ==> nenhum aviso a ninguém!

# Comutadores da camada de enlace

Podemos identificar diversas vantagens no uso de comutadores:

- Eliminação de colisões.
- Enlaces heterogêneos.
- Gerenciamento.

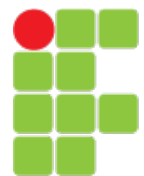
Processamento de pacotes em comutadores, roteadores e hospedeiros:



- ambos dispositivos de armazenamento e repasse
  - roteadores: dispositivos da camada de rede
  - comutadores são dispositivos da camada de enlace
- roteadores mantêm tabelas de roteamento, implementam algoritmos de roteamento
- switches mantêm tabelas de comutação, implementam filtragem, algoritmos de aprendizagem

# Redes do datacenter

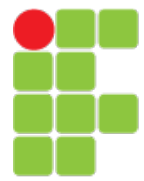
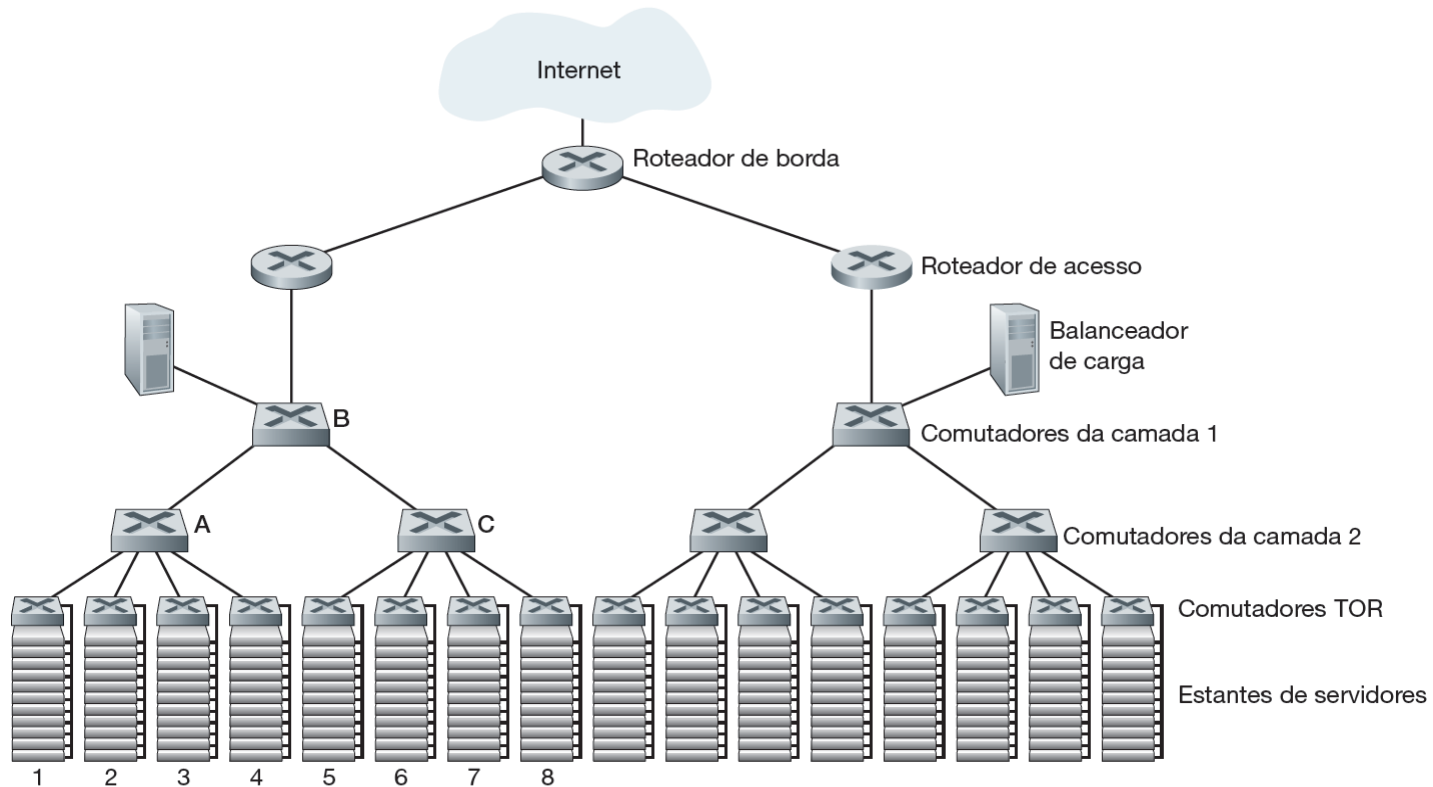
- Nos últimos anos, empresas de Internet como Google, Microsoft, Facebook e Amazon construíram datacenters maciços.
- Cada datacenter tem sua própria **rede do datacenter** que interconecta seus hospedeiros e liga o datacenter à Internet.
- O custo de um grande datacenter é imenso, ultrapassando US\$ 12 milhões por mês para um datacenter de 100 mil hospedeiros [Greenberg, 2009a].
- A figura a seguir mostra um exemplo de uma rede do datacenter.



# Redes do datacenter

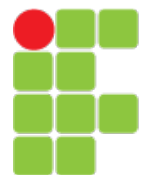
- Uma rede do datacenter com uma topologia hierárquica

TOR: *top-of-rack*

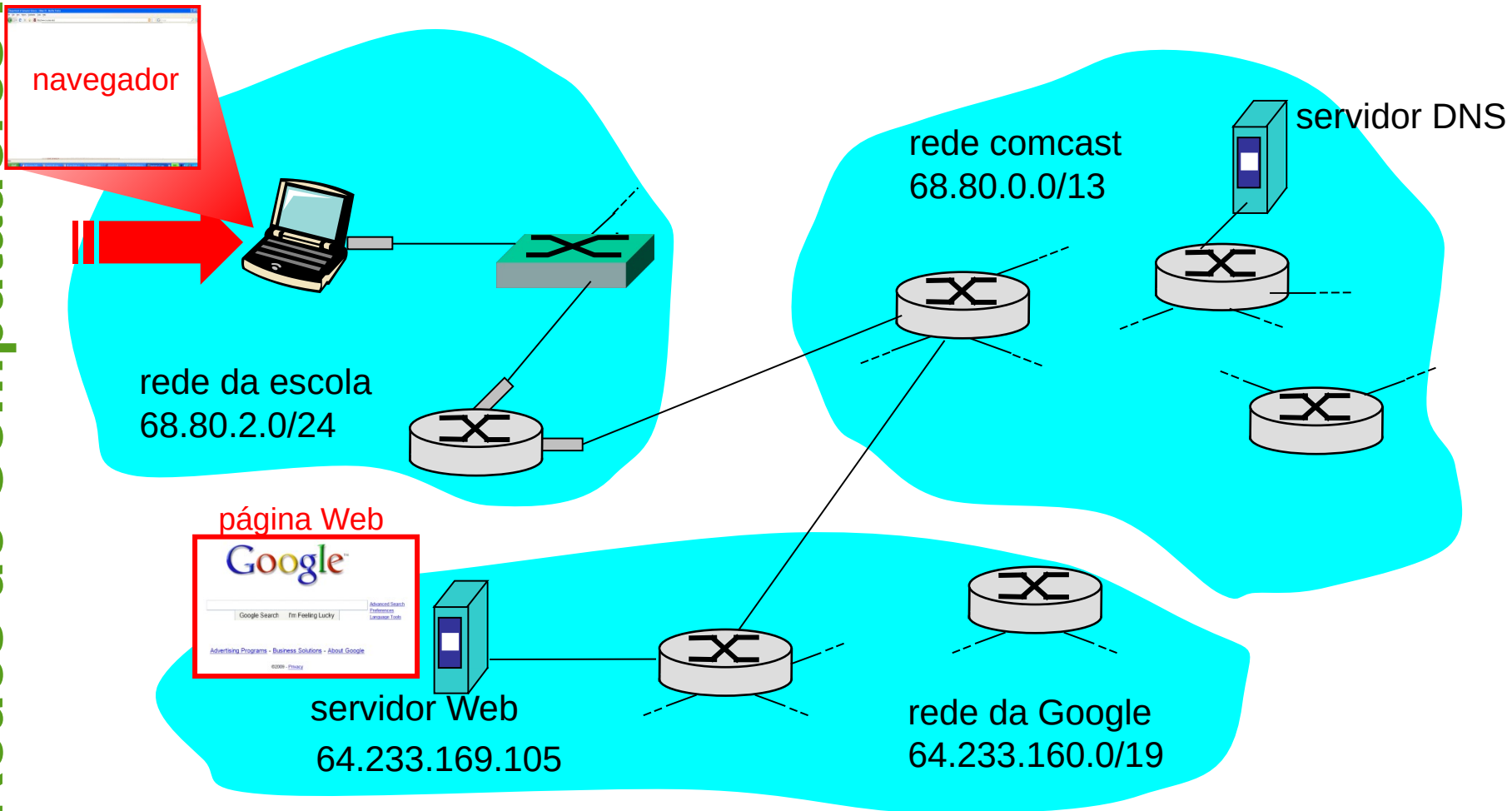


## Síntese: um dia na vida de uma solicitação Web

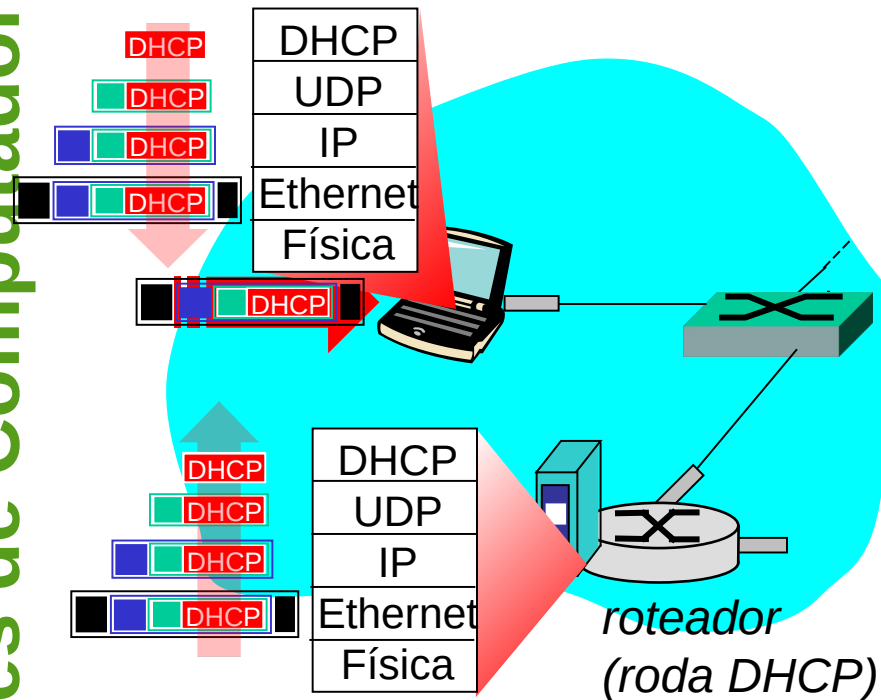
- viagem pela pilha de protocolos completa!
  - aplicação, transporte, rede, enlace
- juntando tudo: síntese!
  - *objetivo*: identificar, analisar, entender os protocolos (em todas as camadas) envolvidos no cenário aparentemente simples: solicitar página WWW
  - *cenário*: aluno conecta laptop à rede do campus, solicita/recebe `www.google.com`



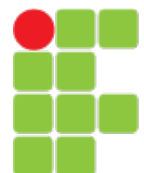
# Um dia na vida: cenário

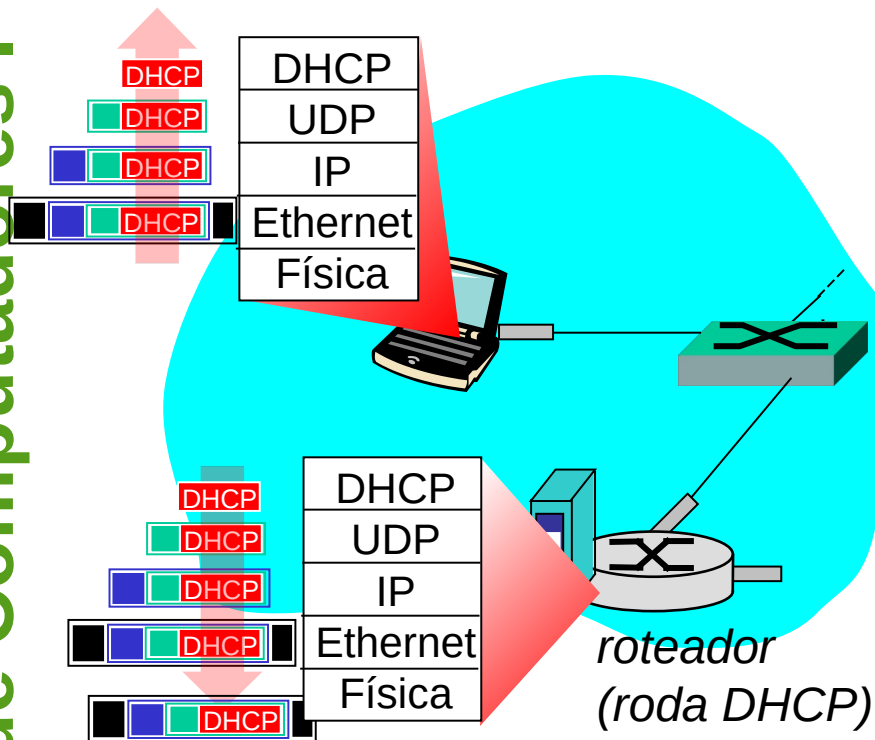


# Um dia na vida... conectando à Internet



- o laptop conectando precisa obter seu próprio endereço IP, end. do roteador do 1º salto e do servidor DNS: use **DHCP**
- Solicitação DHCP **encapsulada** no **UDP**, encapsulada no **IP**, encapsulada na Ethernet **802.1**
- Quadro Ethernet enviado por **broadcast** (dest.: FFFFFFFFFFFFFFFF) na LAN, recebido no roteador rodando servidor **DHCP**
- Ethernet **demultiplexado** para IP demultiplexado, UDP demultiplexado para DHCP



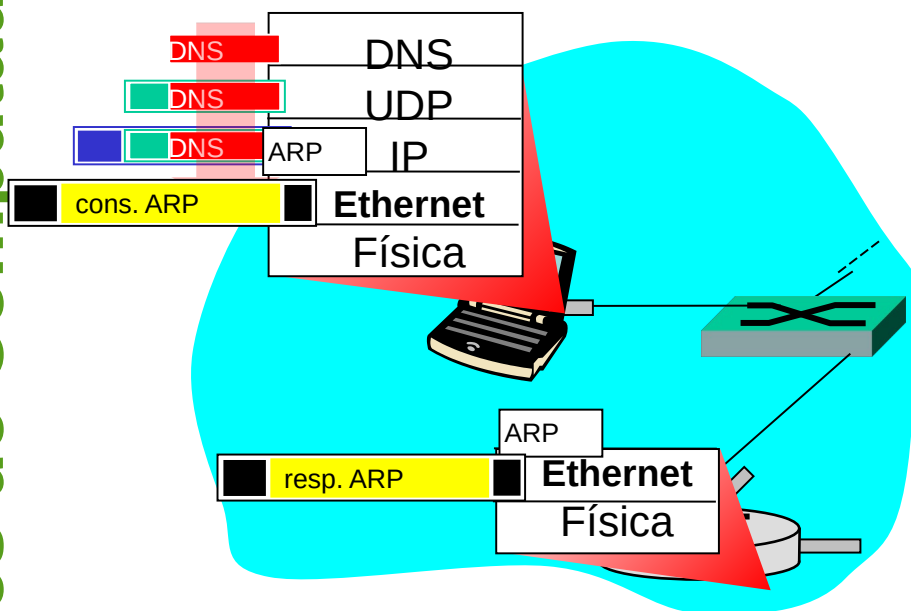


- Servidor DHCP formula **ACK DHCP** contendo endereço IP do cliente, IP do roteador no 1º salto para cliente, nome & endereço IP do servidor DNS
- Encapsulamento no servidor DHCP, quadro repassado (*aprendizagem do comutador*) através da LAN, demultiplexando no cliente
- Cliente DHCP recebe resposta ACK do DHCP

*Cliente agora tem endereço IP, sabe nome e endereço do servidor DNS, endereço IP do seu roteador no primeiro salto*

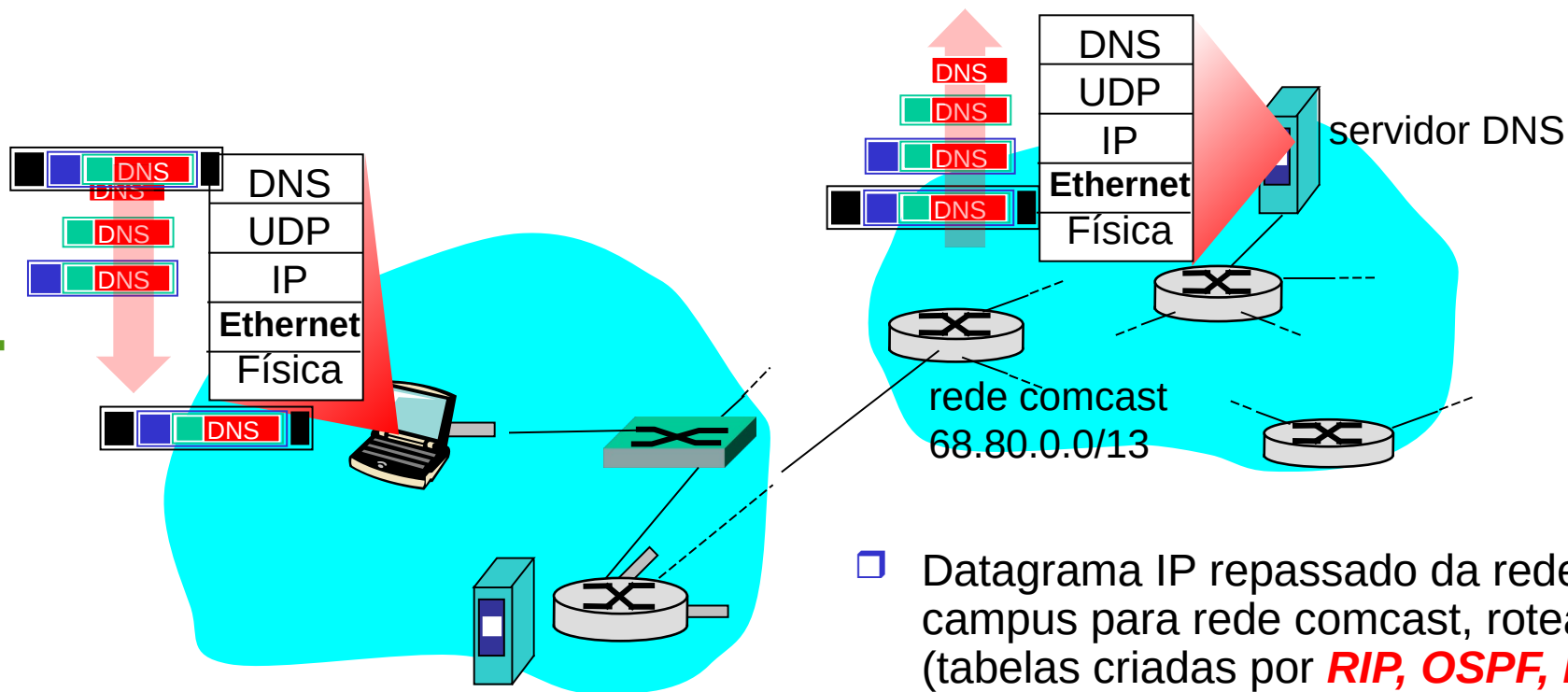


# Um dia na vida... ARP (antes do DNS, antes do HTTP)



- Antes de enviar solicitação **HTTP**, precisa de endereço IP de **www.google.com**: **DNS**
- Consulta DNS criada, encaps. no UDP, no IP, na Ethernet. Para enviar quadro ao roteador, precisa de endereço MAC da interface do roteador: **ARP**
- Broadcast da **consulta ARP**, recebido pelo roteador, que responde com **resposta ARP** dando endereço MAC da interface do roteador
- cliente agora sabe endereço MAC do roteador no 1º salto, e agora pode enviar quadro contendo consulta DNS

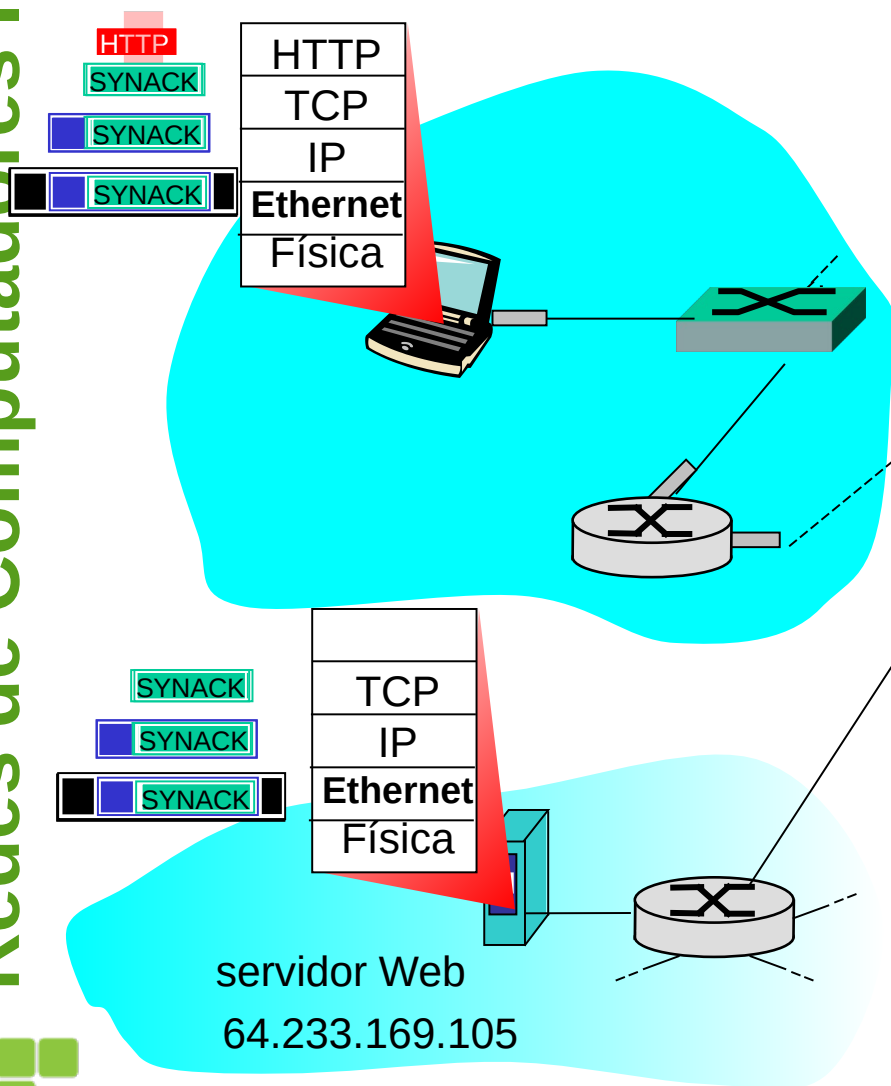
# Um dia na vida... usando DNS



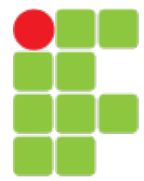
- Datagrama IP contendo consulta DNS repassada via comutador da LAN do cliente ao roteador do 1º salto

- Datagrama IP repassado da rede do campus para rede comcast, roteado (tabelas criadas por **RIP, OSPF, IS-IS** e/ou protocolos de roteamento **BGP**) ao servidor DNS
- demultiplexado ao servidor DNS
- Servidor DNS responde ao cliente com endereço IP de [www.google.com](http://www.google.com)

# Um dia na vida... conexão TCP transportando HTTP



- para enviar solicitação HTTP, cliente primeiro abre **socket TCP** com servidor Web
- **segmento SYN** TCP (etapa 1 na apresentação de 3 vias) **roteado interdomínio** com servidor Web
- servidor Web responde com **SYNACK TCP** (etapa 2 na apresentação de 3 vias)
- **Conexão TCP estabelecida.**



# Um dia na via... solicitação/ resposta HTTP

