

# Aula 4 - Métricas de Desempenho, Modelos em Camadas

Diego Passos

Universidade Federal Fluminense

Redes de Computadores I

Material adaptado a partir dos slides  
originais de J.F Kurose and K.W. Ross.

# Revisão da Última Aula...

- **Comutação de pacotes.**

- Informação dividida em unidades chamadas pacotes.
- Cada pacote tratado isoladamente.
- Quando transmitido, pacote usa toda a capacidade do enlace.

- **Store-and-Forward.**

- Armazenar e encaminhar.
- Comutador só pode começar transmissão no próximo enlace, quando recepção estiver **completa**.
- Transmissões do pacote são sequenciais pela rota.

- **Comutação de circuitos.**

- Estabelecimento prévio de chamada.
- Recursos pré-alocados, não compartilhados.
- Possibilidade de **ociosidade**.

- **Multiplexação.**

- Dados de vários usuários “combinados” em um mesmo meio.
- FDM, TDM.

- Multiplexação estatística.

- Consequência da comutação de pacotes.
- Ordem intercalada, aleatória de pacotes.
- **Pode melhorar utilização dos recursos.**

- Enfileiramento.

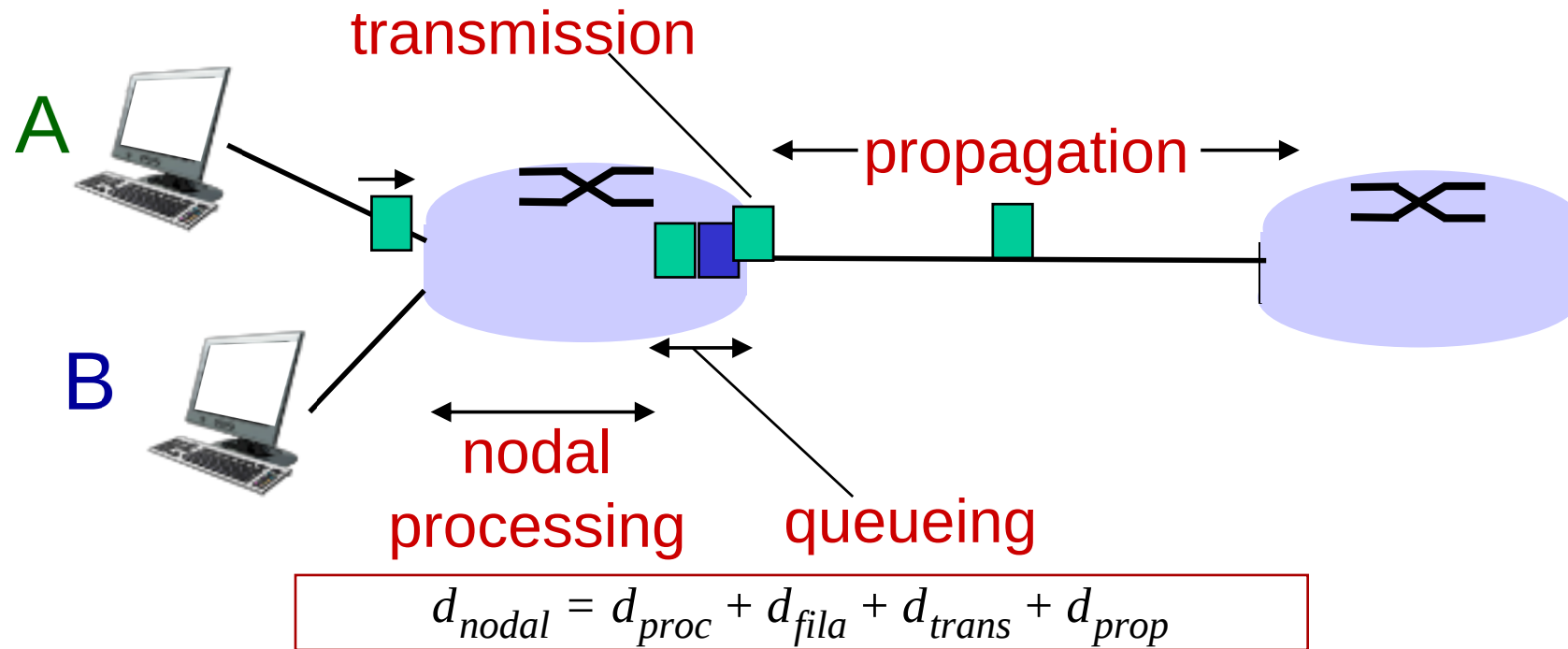
- Consequência do *store-and-forward*.
- Ocorre quando chegada supera a capacidade de saída.
- Descarte de pacotes, atrasos.

# Métricas de Desempenho

# Métricas de Desempenho em Redes

- Há **quatro** métricas clássicas de desempenho em redes:
  - Atraso: tempo que pacote leva para sair de um ponto do sistema até outro.
  - Perda de pacotes: fração dos pacotes transmitidos que são descartados.
  - Vazão: o quão rapidamente dados são transmitidos entre uma origem e um destino.
  - *Jitter*: variação do atraso (tópico de Redes II).
- Fim-a-fim vs. por salto:
  - Métricas podem ser definidas a cada salto/enlace.
  - Mas também podem ser calculadas considerando a comunicação fim-a-fim.
    - Desde o nó de origem até o nó de destino.

# Quatro Fontes de Atraso (I)



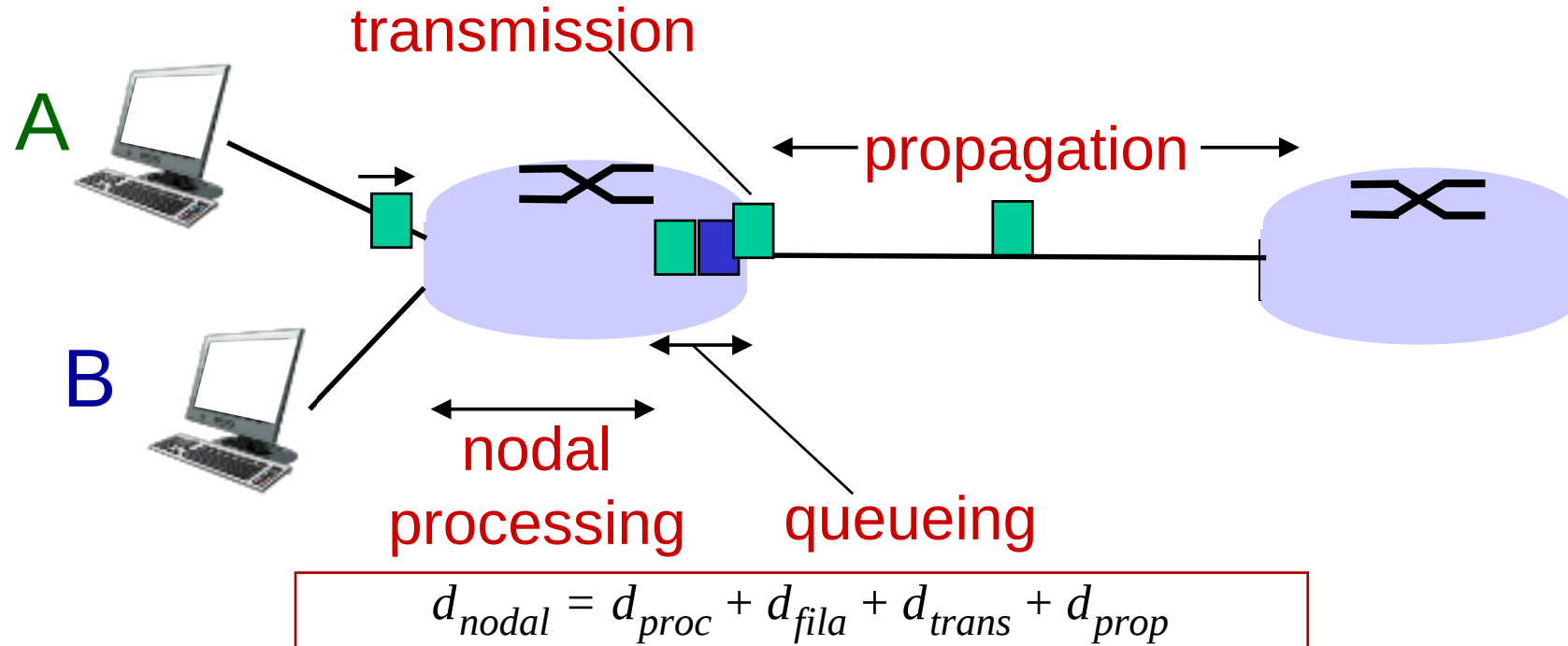
- **$d_{proc}$ : atraso de processamento.**

- Verificar integridade.
- Determinar enlace de saída.
- ...
- Geralmente, < ms

- **$d_{fila}$ : atraso de enfileiramento.**

- Tempo que pacote aguarda a fila para ser transmitido.
- Depende do nível de congestionamento do roteador.

# Quatro Fontes de Atraso (II)



- **$d_{trans}$ : atraso de transmissão.**

- L: comprimento do pacote (bits).
- R: capacidade do enlace (b/s).
- $d_{trans} = \frac{L}{R}$ .

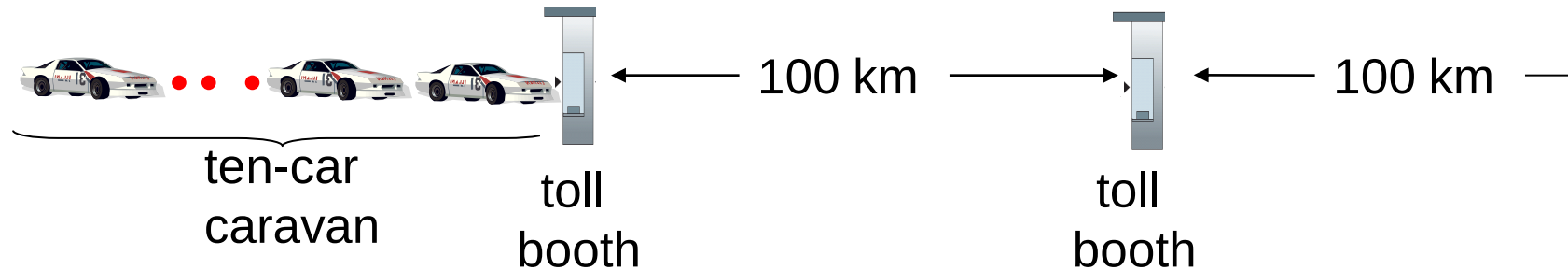
- **$d_{prop}$ : atraso de propagação.**

- d: comprimento do meio físico.
- s: velocidade de propagação do sinal no meio.
- $d_{prop} = \frac{d}{s}$ .

$d_{trans}$  e  $d_{prop}$  são  **muito diferentes!**

# Atraso de Transmissão vs. Atraso de Propagação (I)

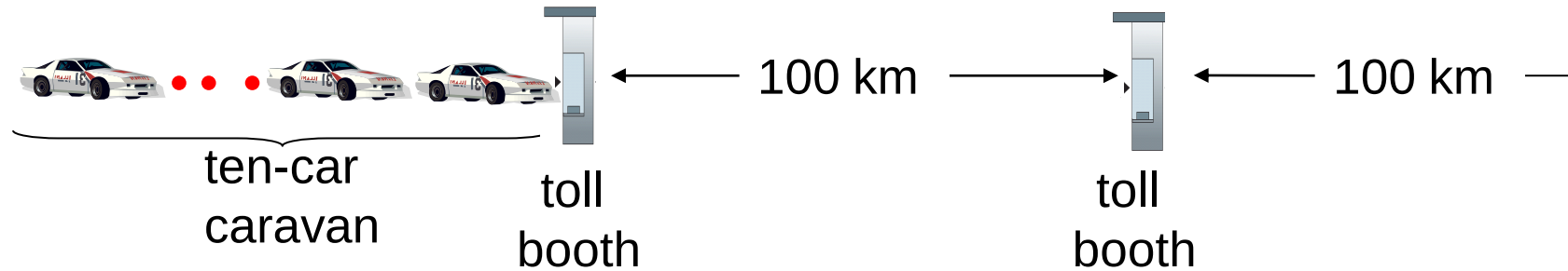
- Analogia de caravana:



- Carros “se propagam” a 100 Km/h.
- Cabine de pedágio leva 12 segundos para servir carro.
  - Tempo de “transmissão de um bit”.
  - “Taxa de transmissão”:  $\frac{1}{12}$  carros/s.
- Carro ~ bit, caravana ~ pacote.
- **Pergunta: quanto tempo até a caravana estar toda enfileirada na segunda cabine?**
- Tempo para “colocar” toda a caravana na estrada =  $12 \cdot 10 = 120$  segundos.
- Tempo para que o último carro se propague da primeira para a segunda cabine =  $\frac{100 \text{ km}}{100 \text{ km/h}} = 1 \text{ hora}$ .
- **Ou 62 minutos.**

# Atraso de Transmissão vs. Atraso de Propagação (II)

- Analogia de caravana:



- Suponha que agora carros propaguem a 1000 km/h.
- E cabines agora levam um minuto para servir cada carro.
- **Pergunta: um ou mais carros chegarão à segunda cabine antes que todos sejam servidos pela primeira?**
  - **Sim!** Depois de 7 minutos, primeiro carro chega à segunda cabine.
  - Mas três ainda serão servidos pela primeira.



# Atraso de Transmissão vs. Atraso de Propagação (III)

Atraso de Propagação (ms)	<input type="text" value="200"/>
Taxa de Transmissão (kb/s)	<input type="text" value="120"/>
Tamanho do Pacote (B)	<input type="text" value="1000"/>
<input type="button" value="Transmitir"/>	

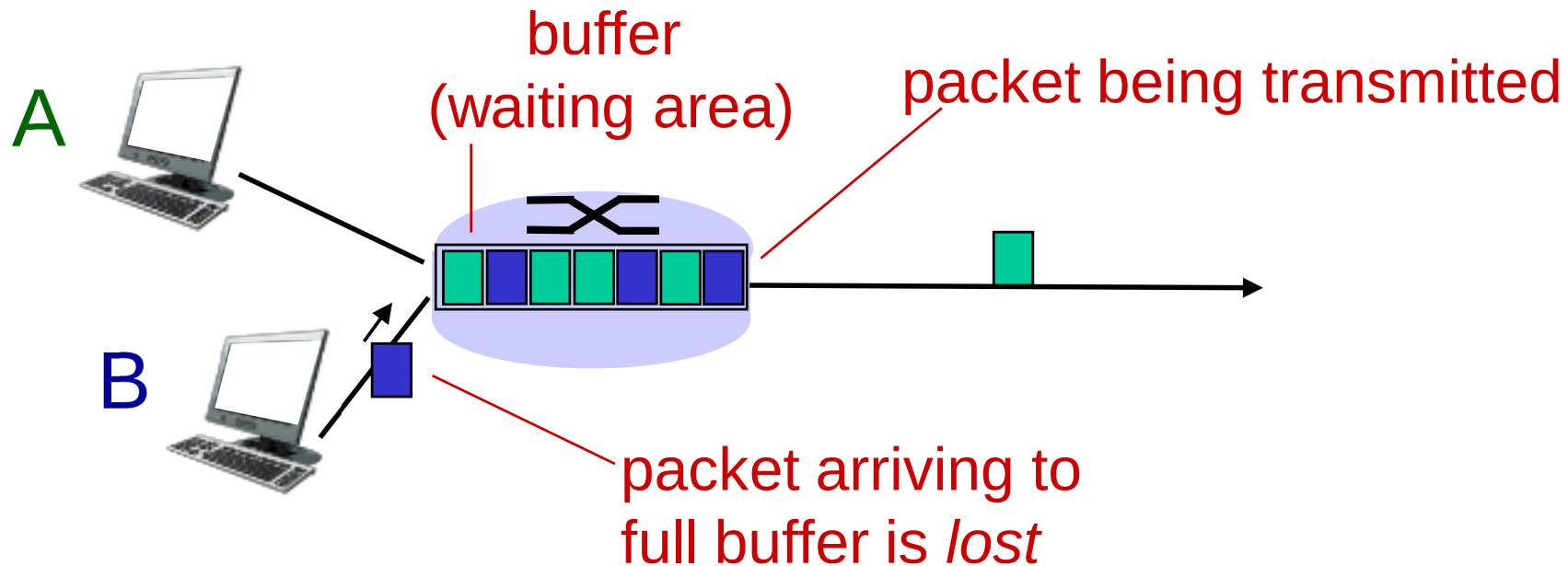


$t = 0$  ms

Último bit é colocado no canal em  $t =$   
Primeiro bit chega ao receptor em  $t =$   
Último bit chega ao receptor em  $t =$

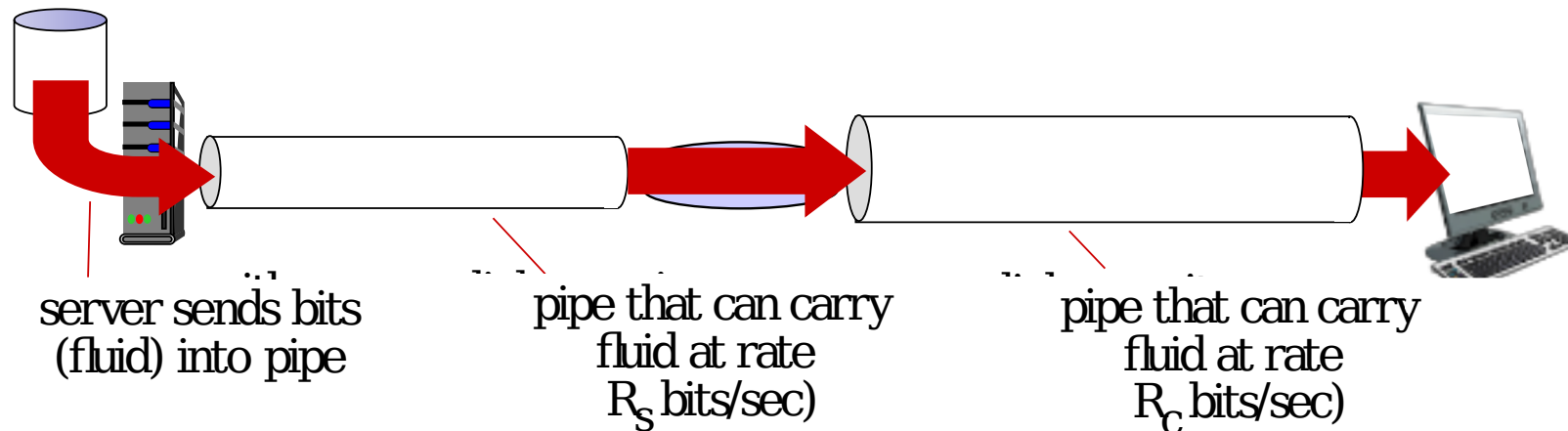
# Perdas de Pacotes

- Filas (ou *buffers*) têm capacidade finita.
- Pacote que chega a um *buffer* causa descarte.
  - Do pacote recém chegado ou de outro.
  - Também conhecido como **perda**.
- Pacotes perdidos **podem** ser retransmitidos.
  - Pelo nó anterior (menos comum).
  - Pela origem (mais comum).



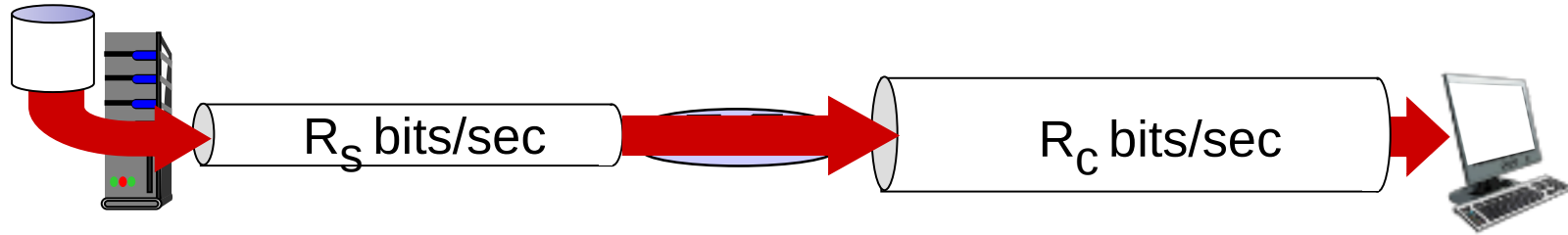
# Vazão (I)

- **Vazão:** taxa (bits/unidade de tempo) na qual bits são transferidos entre origem e destino.
  - **Instantânea:** vazão em um ponto no tempo.
  - **Média:** taxa considerando um período mais longo.
- Vazão do fluxo vs. taxa de transmissão do fluxo.
  - Aplicação pode gerar pacotes a uma taxa alta e vazão ser limitada pela rede.
  - Rede pode ter capacidade alta, mas aplicação pode gerar fluxo com baixa taxa.

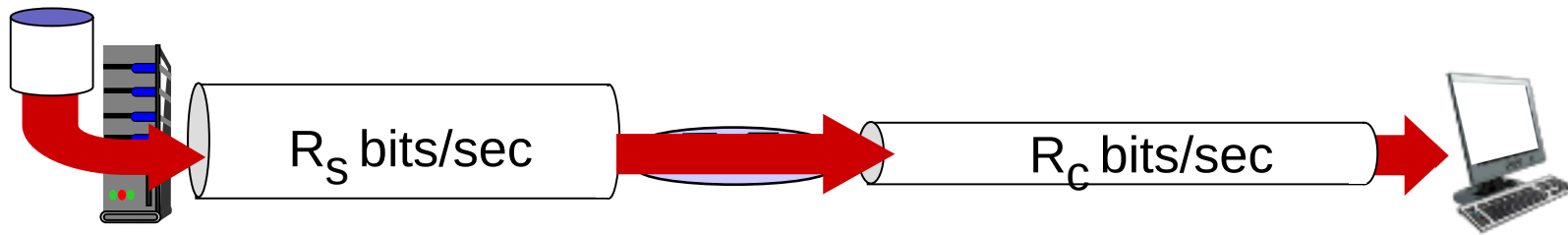


## Vazão (II)

$R_s < R_c$  Qual é a vazão média fim-a-fim?



$R_s > R_c$  Qual é a vazão média fim-a-fim?



### Enlace de Gargalo

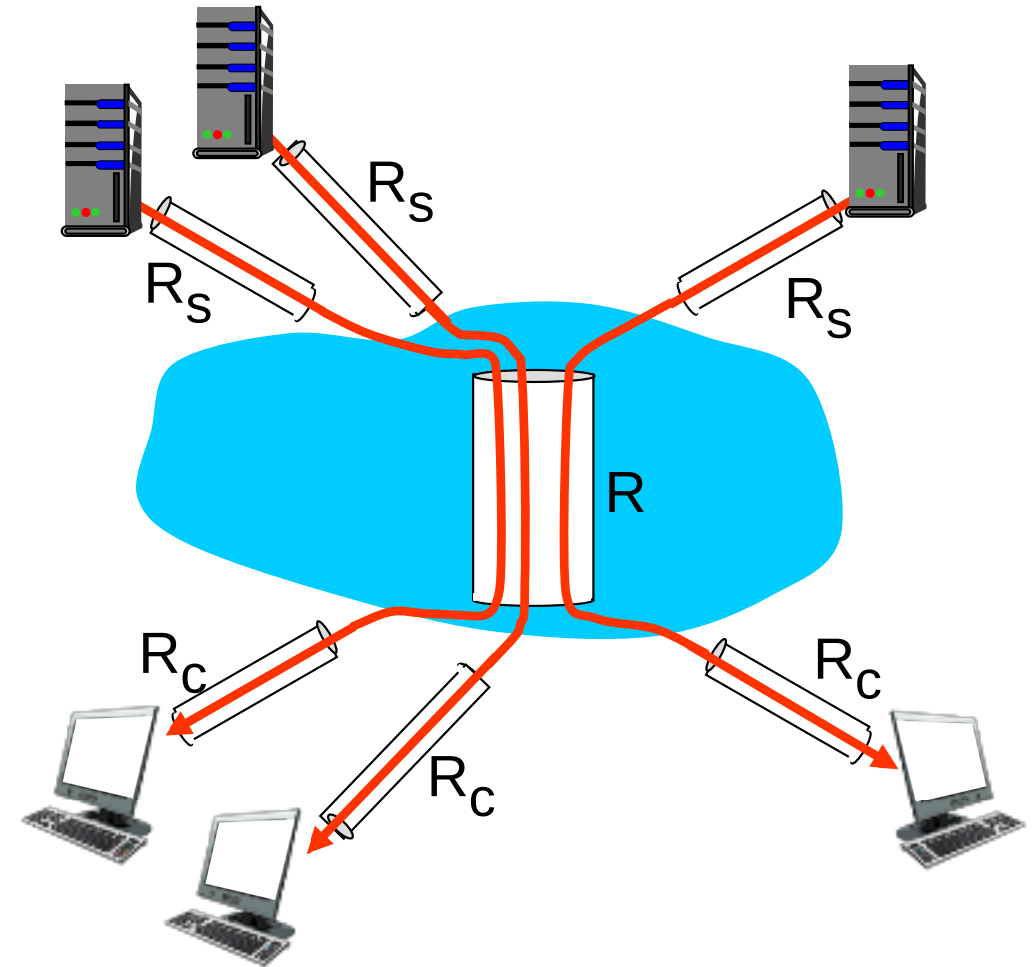
Enlace no caminho fim-a-fim que **restringe** vazão fim-a-fim.

# Cenário de Vazão na Internet

- Vazão fim-a-fim por conexão:

$$\min \left( R_c, R_s, \frac{R}{10} \right)$$

- Na prática:  $R_c$  ou  $R_s$  são normalmente os gargalos.



10 conexões compartilham (de forma justa) o enlace de gargalo no backbone

# Modelos em Camadas

# “Camadas” de Protocolos

- **Redes são sistemas complexos, com várias “partes”.**

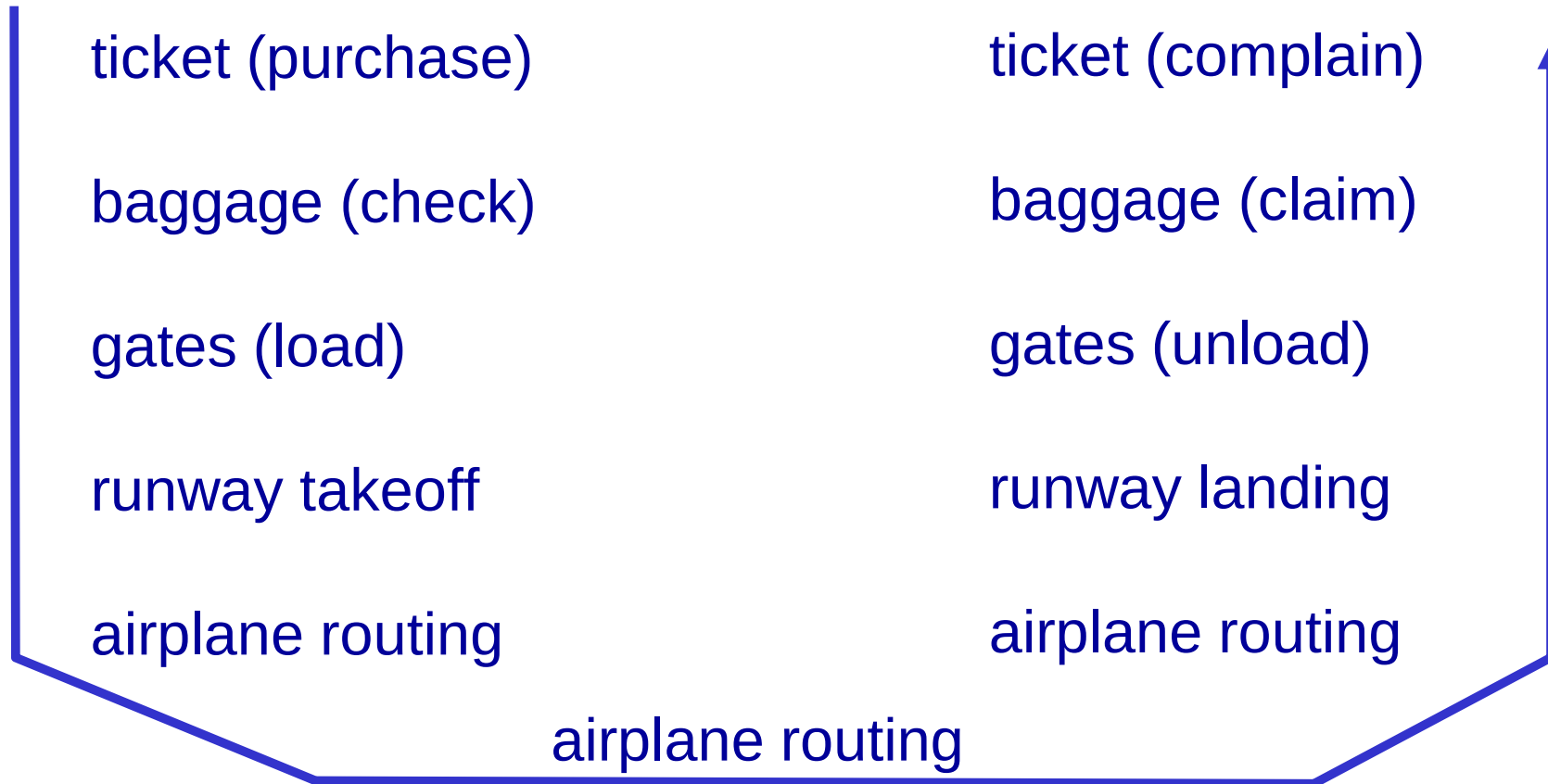
- *Hosts.*
- *Roteadores.*
- *Enlaces (de vários tipos).*
- *Aplicações.*
- *Protocolos.*
- *Hardware, software.*

## Pergunta

Alguma chance de organizar a estrutura da rede?

... ou ao menos nossa discussão sobre redes?

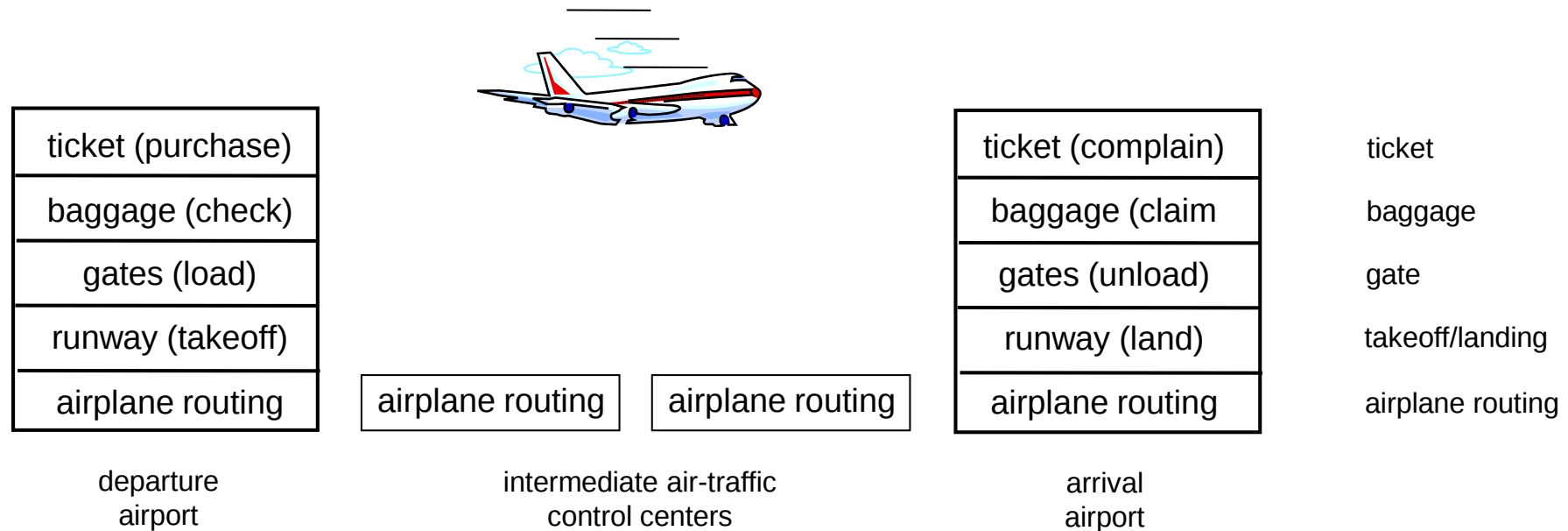
# Analogia: Organização de uma Viagem Aérea (I)



- Uma série de passos envolvidos.



# Organizando Funcionalidades da Cia. Aérea em Camadas



- **Camadas:** cada camada implementa um serviço.
  - Através de ações internas da própria camada.
  - Dependendo de serviços providos pela camada abaixo.

# Por Que uma Organização em Camadas?

- Lidar com sistemas complexos:
  - Estrutura explícita permite identificar relações entre os pedaços do sistema.
    - **Modelo de referência** em camadas para discussão.
  - Modularização facilita a manutenção, atualização do sistema.
    - Alterar a implementação de uma camada é transparente para o resto do sistema.
    - *e.g.*, mudança no processo de embarque não afeta o resto do sistema da cia. aérea.
- “Layering considered harmful”?

# Pilha de Protocolos da Internet (I)

- Ou suíte de protocolos da Internet.
- Ou modelo TCP/IP.
- Define organização dos vários protocolos usados na Internet em camadas.
- Define **responsabilidades, serviços** de cada camada.
- Dividida em 4 ou 5 camadas, dependendo do autor.
  - Nomes das camadas também podem variar.  
item
  - Nesta disciplina, **seguiremos o modelo do livro-texto**.

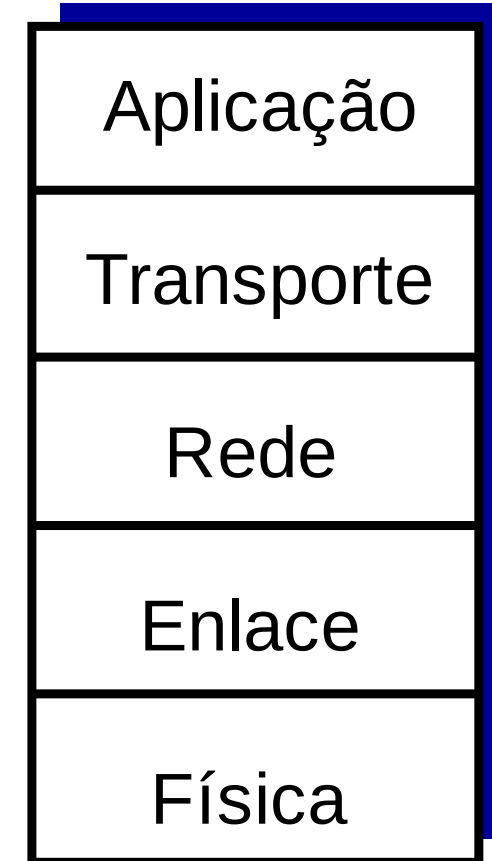
# Pilha de Protocolos da Internet (II)

- **Aplicação:** suporte à aplicações de rede.
  - Definem formato, ordem, semântica das mensagens trocadas pelas aplicações.
  - Exemplo da web:
    - Cliente (*browser*) gera mensagem de requisição de conteúdo.
    - Servidor envia mensagem de resposta.
    - Cada mensagem tem seus campos específicos (próximo capítulo).
  - Exemplos de protocolos:
    - HTTP, FTP, SMTP, ...



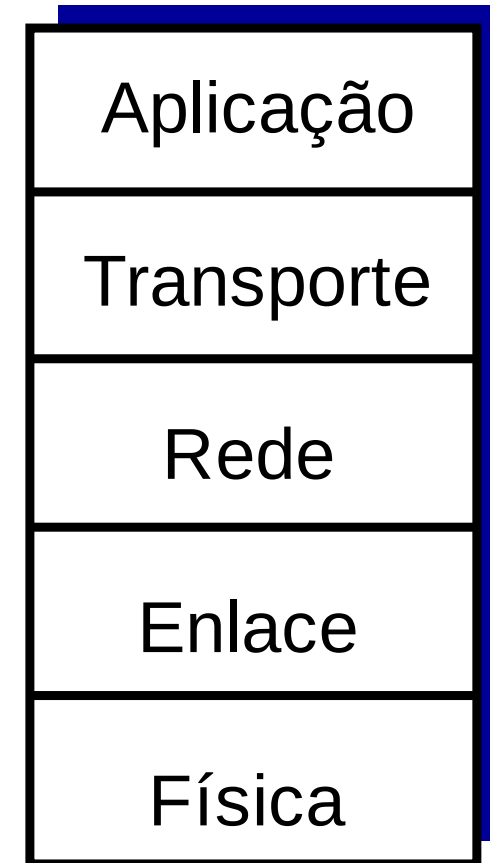
# Pilha de Protocolos da Internet (III)

- **Transporte:** comunicação entre processos em computadores (potencialmente) diferentes.
  - Transfere dados de um processo para o outro.
    - Potencialmente, em computadores diferentes.
  - Dois exemplos clássicos de protocolos:
    - UDP e TCP.
  - Protocolos diferentes proveem **modelos de serviço diferentes**.
    - TCP provê garantias mais fortes.
  - **Responde a pergunta: para qual processo devemos enviar o conteúdo de um pacote que chegou a este host?**



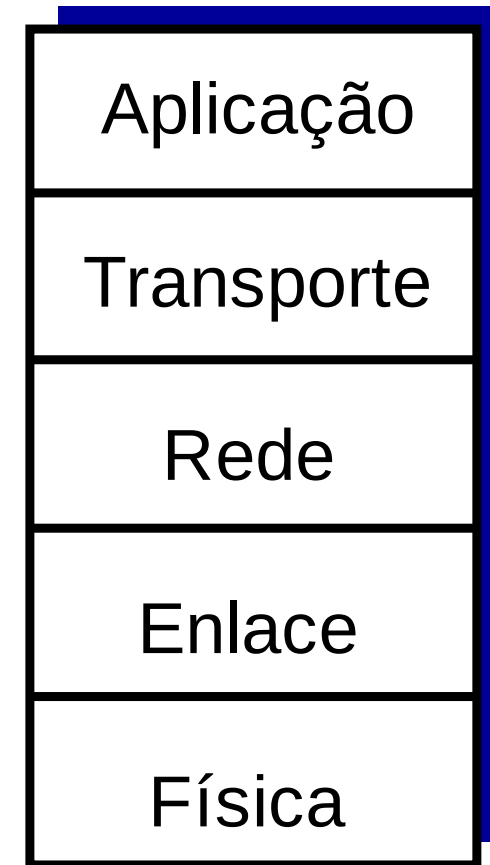
# Pilha de Protocolos da Internet (IV)

- **Rede:** comunicação entre *hosts*.
  - Transfere dados de um *host* para outro.
  - Diferença (aparentemente) sutil em relação à camada de transporte.
    - Aquela comunicava processos específicos, diferenciando-os.
    - Esta comunica *hosts* indiscriminadamente.
  - Provê o serviço de roteamento dos pacotes.
    - Entre outros.
  - **Responde a pergunta: por qual caminho devemos enviar este pacote para que chegue ao *host* de destino?**



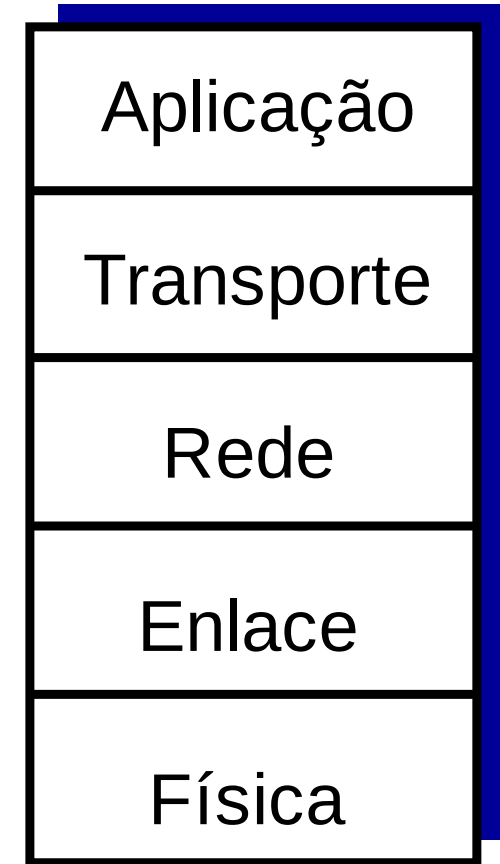
# Pilha de Protocolos da Internet (V)

- **Enlace:** comunicação nós (*hosts*, computadores) diretamente conectados por um enlace.
  - *i.e.*, nós “vizinhos”.
  - Diferente da camada de rede, preocupação apenas com **próximo salto**.
    - Aspectos relacionados à transmissão do pacote pelo enlace.
  - Exemplo de serviço: integridade.
    - Em enlaces susceptíveis a falhas, verifica se houve bits errados na recepção pelo enlace.
  - **Responde a pergunta: como envio o pacote para o próximo dispositivo no caminho?**



# Pilha de Protocolos da Internet (VI)

- **Física:** representação do pacote no meio físico de transmissão.
  - Traduz bits para sinais.
    - Pulsos elétricos, ondas acústicas, pulsos de luz, ...
  - Se preocupa com a forma mais eficiente de representação.
    - *e.g.*, como transmitir mais bits em menos tempo no canal?
    - *e.g.*, como minimizar os erros na transmissão?





# Pilha de Protocolos: Uma Visão Bottom-Up

- Cada camada assume que camadas inferiores resolvem “uma parte do problema”.
  - Sobre a solução existente, são adicionadas novas funcionalidades.
- Exemplo (simplificado):
  - Camada de enlace assume que sabemos enviar bits por um enlace.
    - Preocupa-se em verificar se bits chegaram corretos.
  - Camada de rede assume que sabemos enviar pacote entre nós diretamente conectados.
    - Preocupa-se em encontrar e usar caminhos de múltiplos saltos entre origem e destino.
  - Camada de transporte assume que sabemos enviar pacote entre origem e destino.
    - Preocupa-se em separar pacotes que chegam entre os vários processos do *host*.
  - Camada de aplicação assume que sabemos enviar pacotes entre processos em computadores diferentes.
    - Preocupa-se em gerar mensagens que permitam o funcionamento da **aplicação distribuída**.

# Pilha de Protocolos: Uma Visão *Top-Down* (I)

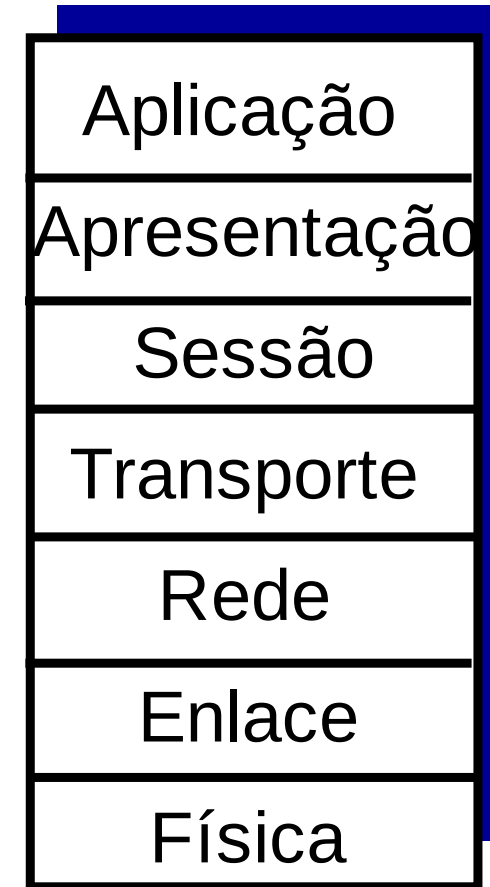
- Estratégia utilizada pelo livro-texto.
- Estudar as redes de computadores (Internet, em particular) percorrendo camadas de cima para baixo.
  - Aplicação – Capítulo 2.
  - Transporte – Capítulo 3.
  - Rede – Capítulo 4.
  - Enlace – Capítulo 5.
- No restante desta disciplina, estudaremos as várias camadas da pilha TCP/IP.

# Pilha de Protocolos: Uma Visão Top-Down (II)

- Estratégia utilizada pelo livro-texto.
- Estudar as redes de computadores (Internet, em particular) percorrendo camadas de cima para baixo.
  - **Aplicação – Capítulo 2.**
  - **Transporte – Capítulo 3.**
  - **Rede – Capítulo 4.**
  - Enlace – Capítulo 5.
- No restante desta disciplina, estudaremos as várias camadas da pilha TCP/IP.
  - Aplicação, Rede e Transporte em Redes I (camadas 5, 4 e 3).
  - Camada de enlace (e outros tópicos) em Redes II.

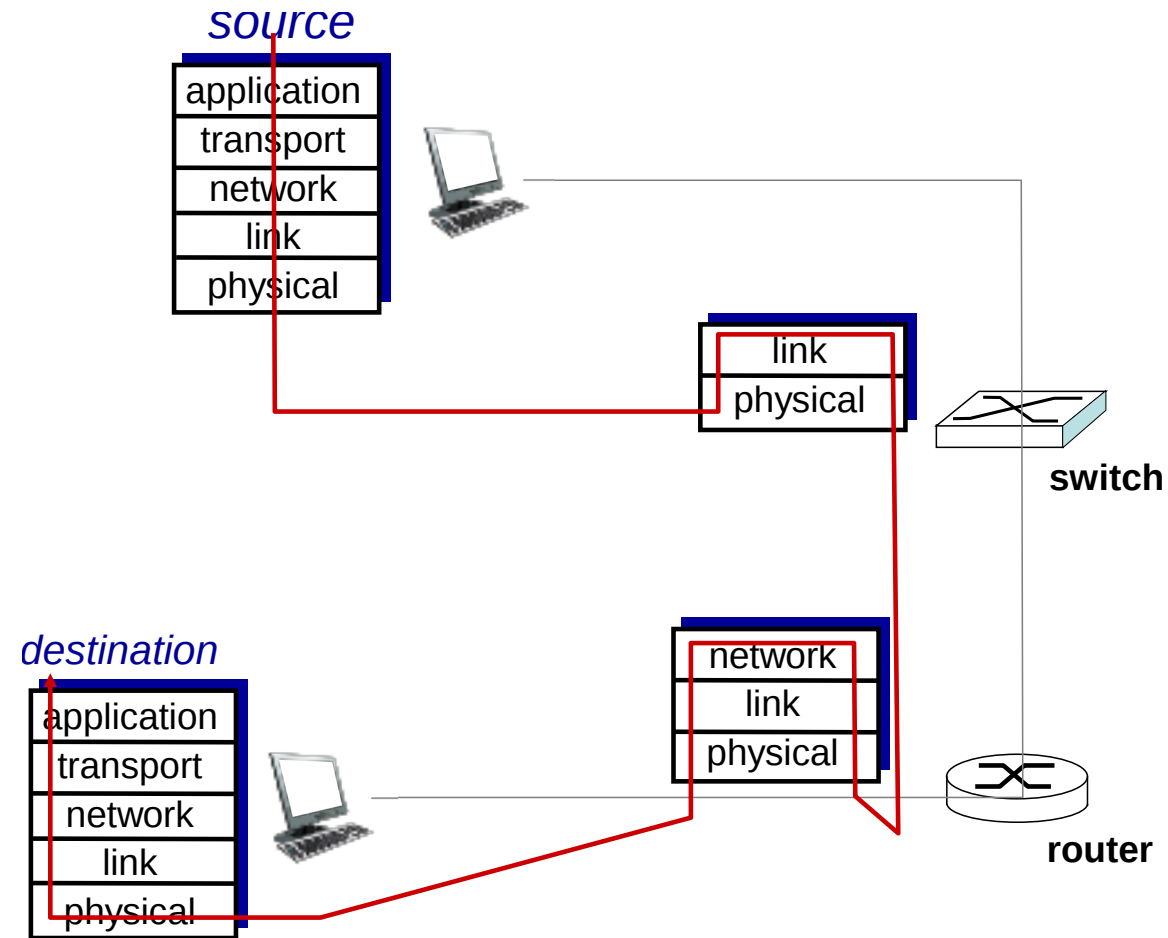
# O Modelo de Referência ISO/OSI

- Modelo TCP/IP não é o único, absoluto.
- Uma alternativa: modelo OSI.
  - Sete camadas, ao invés de 5/4.
  - **Apresentação**: “traduz” formato dos dados entre a aplicação e a rede.
    - e.g., criptografia, compressão, *endianess*.
  - **Sessão**: provê funcionalidades como sincronização, *checkpoints*, recuperação de dados.
- Como a Internet lida com a ausência destas camadas?
  - Funcionalidades implementadas na aplicação, **se necessárias**.
  - São necessárias?



# Inteligência nas Bordas e Camadas

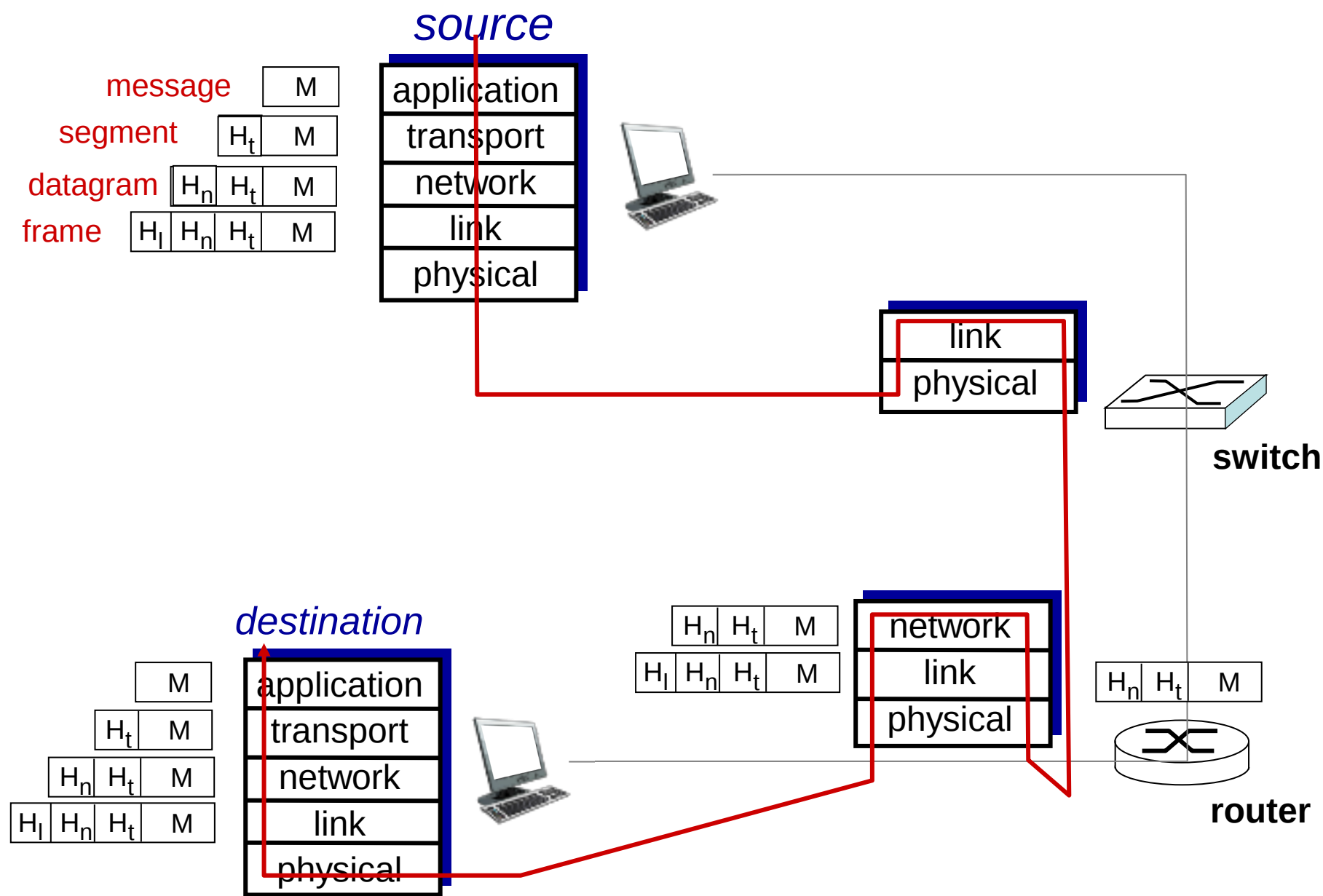
- Lembrando: na Internet, procura-se manter a inteligência nas bordas.
  - *i.e.*, na medida do possível, funcionalidades são mantidas nas bordas, e não no núcleo.
  - Argumento fim-a-fim.
- Isto se manifesta (idealmente) na organização em camadas.
  - Hosts implementam todas as camadas.
  - Comutadores implementam apenas as 2 ou 3 mais baixas.
    - Switches até a camada 2.
    - Roteadores até a camada 3.



# Encapsulamento e Cabeçalhos (I)

- Pacotes são gerados (normalmente) na aplicação e descem pelas demais camadas.
- Cada nova camada pode precisar adicionar informações ao pacote para cumprir suas responsabilidades.
  - e.g., camada de transporte adiciona um identificador ao pacote para que receptor saiba para qual processo este é destinado.
- Informações adicionais são colocadas em posições bem definidas do pacote.
  - Em **cabeçalhos**.
- Cada camada pode adicionar (e geralmente adiciona) seu cabeçalho à mensagem.
  - Com suas informações relevantes.

# Encapsulamento e Cabeçalhos (II)



# Segurança



# Segurança em Redes

- **Campo que estuda:**

- Como atacantes podem gerar problemas para a rede/computadores.
- Como podemos nos defender destes ataques.
- Como projetar a arquiteturas de redes imunes a ataques.

- Internet **não foi** originalmente pensada com (muita) segurança em mente.

- Visão original: “grupo de usuários que confiam uns nos outros conectados a uma rede transparente”.
- Projetistas de protocolos da Internet estão sempre “correndo atrás”.
- Considerações de segurança aparecem em todas as camadas!

# Atacantes: Inserção de *malware* nos *hosts* via Internet

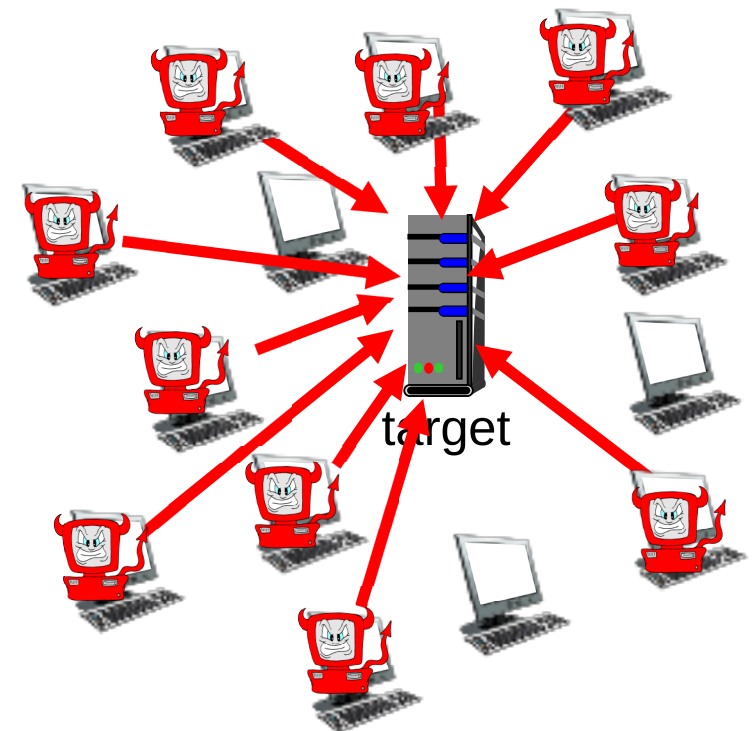
- *Malware* pode infectar um *host* de algumas formas.
  - **Vírus**: *software* que se auto-replica através da recepção/execução de objetos pela rede (e.g., anexo em e-mail).
  - **Worm**: *software* que se auto-replica através do recebimento/execução **passivos** (e.g., automáticos) de um objeto.
- Um *malware* do tipo **spyware** pode gravar ações do usuário (e.g., teclas pressionadas, páginas visitadas) e enviar para servidor do atacante.
- Host infectado pode ser controlado como parte de uma **botnet**.  
item

# Ataques a Servidores, Infraestrutura de Rede

- **Ataque de Negação de Serviço:**

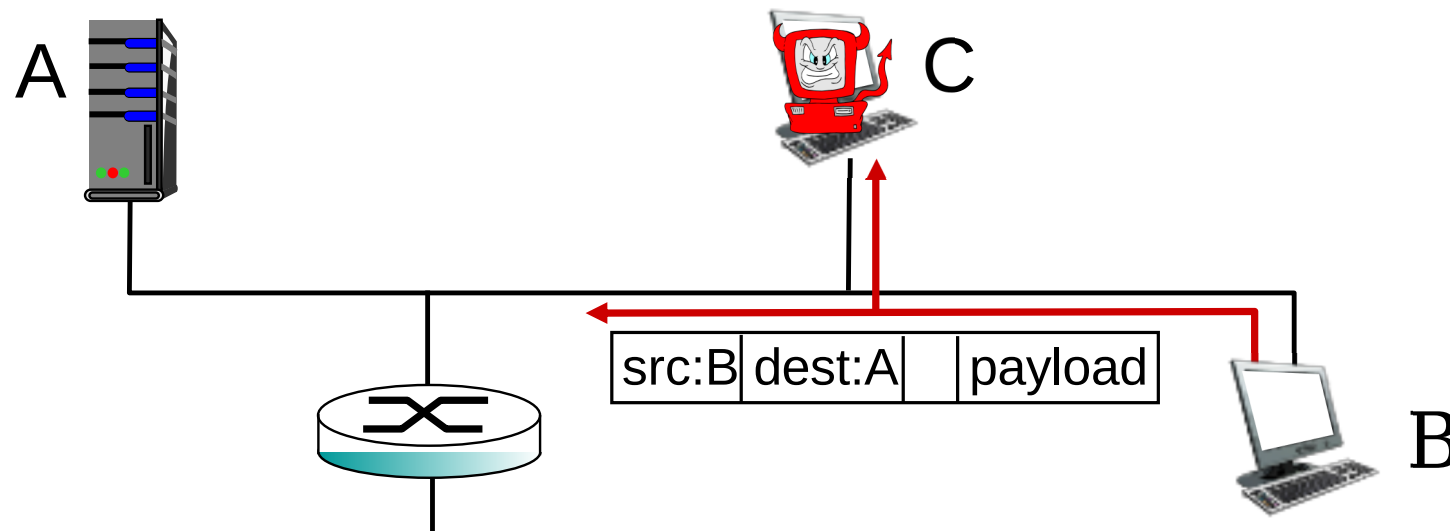
- Denial of Service (DoS).
- Atacante faz com que recurso (servidor, banda) fique indisponível para os usuários legítimos.
- Normalmente, se baseia na geração de tráfego artificial em grandes volumes, ocupando os recursos.

1. Selecionar o alvo.
2. Comprometer *hosts* pela rede (*i.e.*, criar uma *botnet*).
3. Enviar pacotes para o alvo a partir dos *hosts* comprometidos.
  - Mais de um *host* → DDoS.



# Sniffing de Pacotes

- Normalmente, interfaces só passam para as camadas superiores quadros destinados ao próprio dispositivo.
- Mas, fisicamente, interface muitas vezes recebe quadros para outros nós.
  - e.g., enlaces compartilhados.
- Softwares especiais podem capturar e exibir estes pacotes.
  - Sniffers de pacotes, como o Wireshark.



- Também utilizados para fins legítimos!

# Histórico

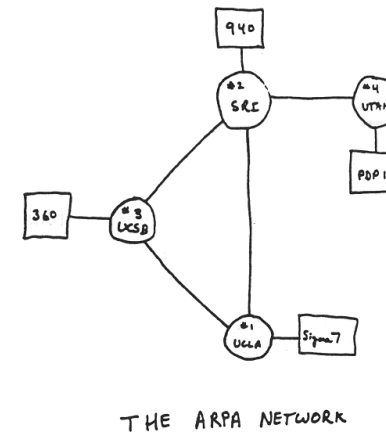
# História da Internet (I)

1961 a 1972

Estabelecimento dos princípios da comutação de pacotes

- **1961:** Kleinrock mostra através de teoria das filas a efetividade da comutação de pacotes.
- **1964:** Comutação de pacotes é aplicada a redes militares.
- **1967:** Concepção da ARPAnet pela ARPA.
- **1969:** primeiro nó operacional da ARPAnet.

- **1972:**
  - Primeira demonstração pública da ARPAnet.
  - NCP (Network Control Protocol).
  - Primeiro programa de e-mail.
  - ARPAnet chega a 15 nós.



# História da Internet (II)

1972 a 1980

Intercomunicação entre redes, redes novas e proprietárias

- **1970:** ALOHAnet no Havaii.
- **1974:** Cerf e Kahn definem arquitetura para intercomunicação de redes.
- **1976:** Ethernet é criado na Xerox PARC.
- **Final de 1970:** Arquiteturas proprietárias (DECnet, SNA, XNA).
- **Final de 1970:** Comutação de pacotes de comprimento fixo (precursor do ATM).
- **1979:** ARPAnet com 200 nós.

## Princípios de intercomunicação de redes segundo Cerf e Kahn:

- Minimalismo, autonomia: evitar mudanças nas redes interconectadas.
- Modelo de serviço de **melhor esforço**.
- Roteadores **stateless**.
- Controle descentralizado.

**Definem a arquitetura da Internet atual**

# História da Internet (III)

1980 a 1990

Novos protocolos, proliferação de redes

- **1983:** Implantação do TCP/IP.
- **1982:** Protocolo de e-mail SMTP definido.
- **1983:** Protocolo DNS é definido para tradução de nomes.
- **1985:** Protocolo FTP é definido.
- Novas redes nacionais surgem nos EUA: Csnet, BITnet, NSFnet, Minitel.
- 100000 *hosts* conectados a confederação de redes.



# História da Internet (IV)

1990 a 2005

Comercialização, a web, novas aplicações

- **Início de 1990:** ARPAnet é desativada.
- **1991:** NSF retira restrições de uso comercial da NSFnet.
- **Início de 1990:** Surgimento da Web.
  - Hipertexto.
  - HTML e HTTP: Berners-Lee.
  - 1994: Mosaic, mais tarde Netscape.
  - Fim da década 1990: comercialização da web.
- **Fim de 1990 a meados de 2000:**
  - Mais aplicações populares: mensagens instantâneas, transferência de arquivos via P2P.
  - Foco maior em segurança.
  - Estimativas de 50 milhões de *hosts*, mais de 100 milhões de usuários.
  - *Links de backbone* a Gb/s.

# História da Internet (V)

## 2005 – presente

Expansão, aumento de escala, multimídia, dispositivos móveis.

- Aproximadamente 900 milhões de *hosts* (2012).
  - *Tablets, smartphones, ...*
  - Implantação agressiva de acesso banda larga.
  - Acesso à Internet sem fio se tornando ubíquo.
  - Surgimento das redes sociais.
    - Facebook: 1,65 bilhões de usuários (2016).
  - Servidores de conteúdo criam suas próprias redes.
    - “Evitam” Internet pública, entregam conteúdo de forma “instantânea”: buscas, e-mail, vídeos.
  - Empresas, universidades, comércio eletrônico rodando seus serviços na “nuvem” (*e.g.*, EC2 da Amazon).

# Resumo da Aula...

- Medidas de desempenho em redes:
  - Perda de pacotes.
  - Atraso.
  - Vazão.
- Perda de pacotes ocorre por:
  - Erros na transmissão por enlaces (menos comum na Internet).
  - **Descartes por buffers cheios** (mais comum).
- Atraso é acumulado a cada salto:
  - Processamento.
  - Enfileiramento.
  - Transmissão.
  - Propagação.
- Vazão é diferente da taxa de transmissão do fluxo.
  - Depende da capacidade da rede.
  - Limitada pelo **enlace de gargalo**.
  - Também afetada por competição entre fluxos.
- Pilha de Protocolos TCP/IP.
  - Modelo em camadas.
  - Organiza protocolos na Internet.
  - Define **responsabilidades**.
  - Alternativa: modelo OSI.
  - **Encapsulamento**: camadas/protocolos adicionam cabeçalhos.
- Segurança:
  - Ataque de negação de serviço.
- Modelo de serviço de **melhor esforço**.

# Conceitos Importantes do Capítulo 1

- Hosts, sistemas finais, comutadores, enlaces.
- ISPs.
- Núcleo *vs.* borda da rede.
- Protocolos.
- Taxa de transmissão/capacidade de um enlace.
- Descarte de pacotes.
- Atraso, perda, vazão.
- Inteligência nas bordas, argumento fim-a-fim, KISS.
- Comutação de pacotes *vs.* comutação de circuitos.
- *Store-and-forward*.
- Multiplexação estatística.
- Enfileiramento.
- Melhor esforço.
- Pilha de protocolos TCP/IP.
- Encapsulamento.
- Cabeçalhos.

# Próxima Aula...

- Começamos um novo capítulo: Camada de Aplicação.
  - Conceitos básicos de aplicações distribuídas.
  - Arquiteturas (Cliente-Servidor *vs.* P2P).
  - Serviços da camada de aplicação.
  - Início da discussão sobre o protocolo HTTP.