

# Aula 12 - Camada de Rede, Circuitos vs. Datagramas, Roteadores

Diego Passos

Universidade Federal Fluminense

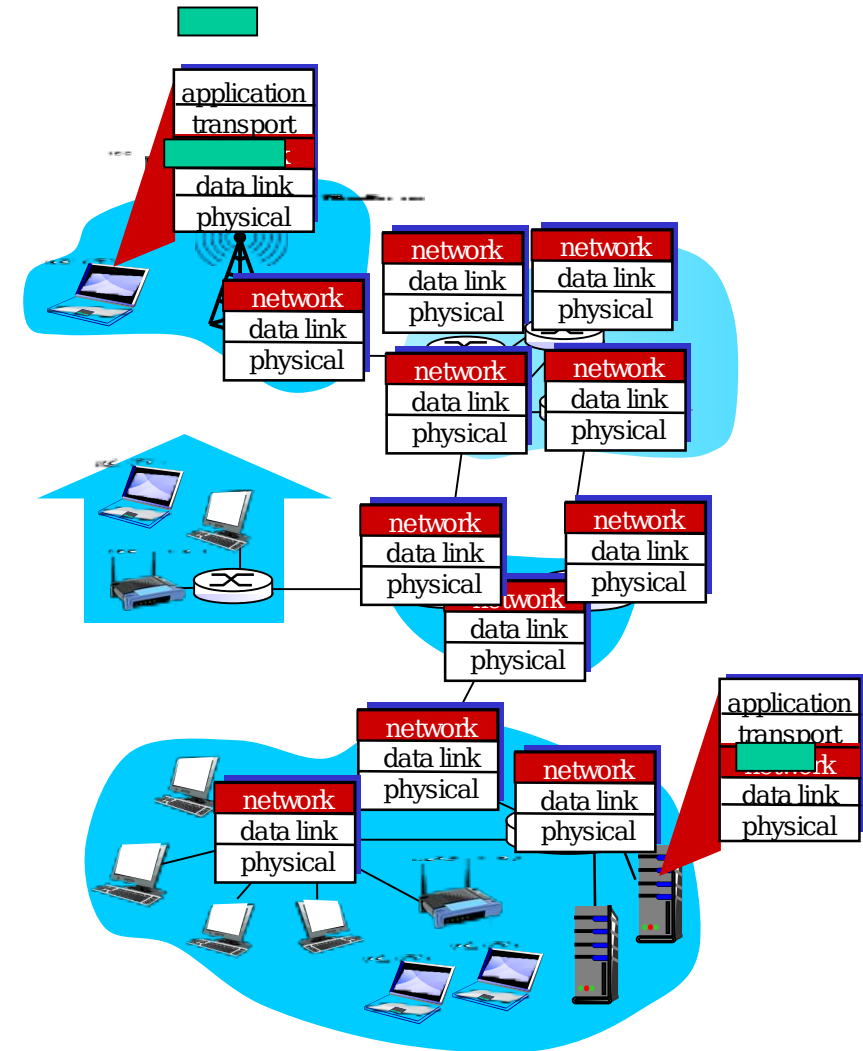
Redes de Computadores

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

# Camada de Rede: Conceitos Básicos

# Camada de Rede

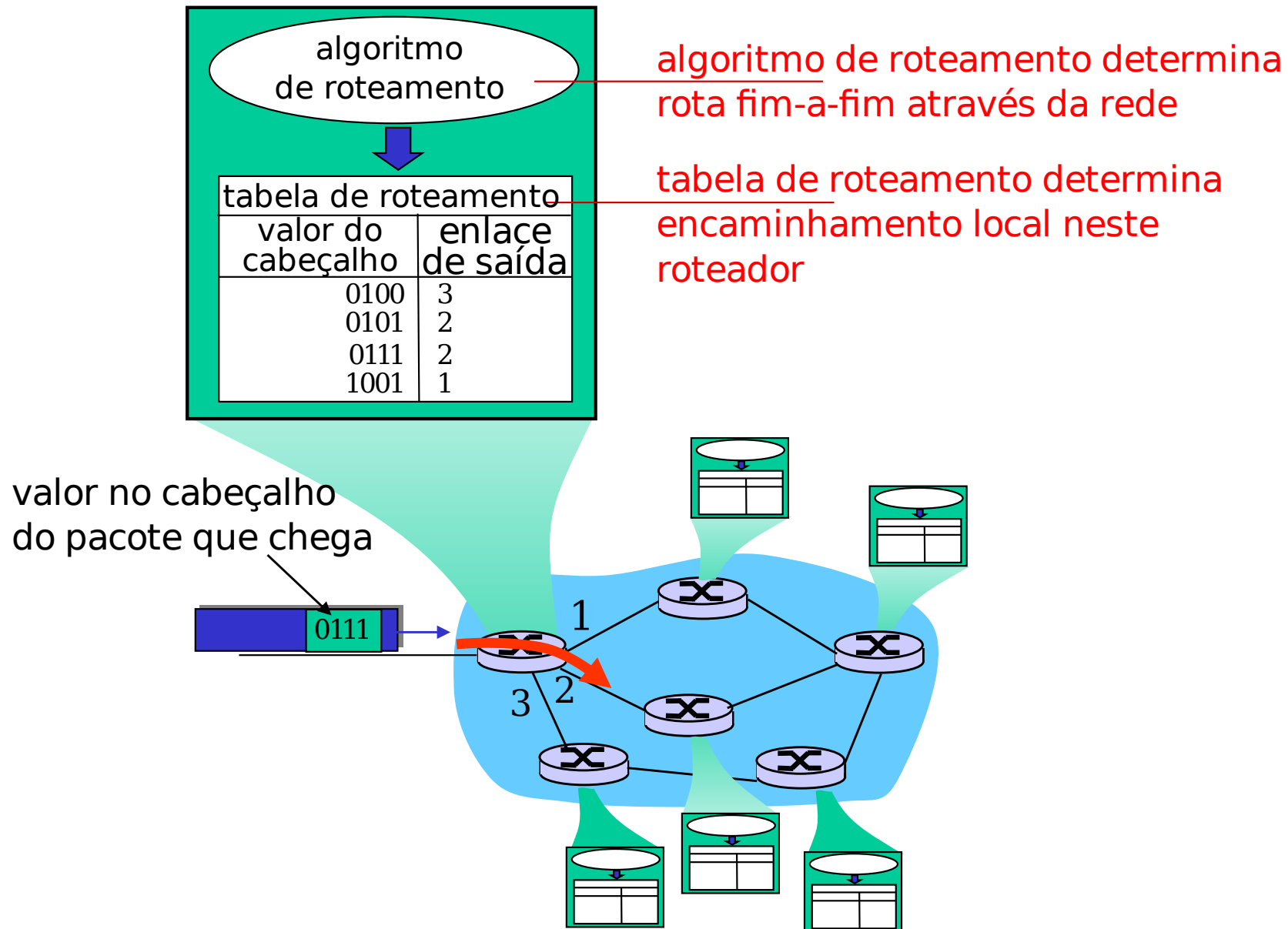
- Transporta segmento do *host* de origem ao *host* de destino.
- No lado transmissor, encapsula segmentos em **datagramas**.
- No lado receptor, entrega segmentos à camada de transporte.
- Protocolos de camada de rede atuam em **todos os nós** (roteadores, *hosts*).
- Roteador examina campos de cabeçalho em todos os datagramas IP que passam por ele.



# Duas Funções Chave da Camada de Rede

- **Encaminhamento:** mover pacotes da entrada para a saída de um roteador.
- **Roteamento:** determina rota usada por pacote da origem ao destino.
  - **Algoritmos de roteamento.**
- **Analogia:**
  - **Roteamento:** processo de planejar uma viagem da origem ao destino.
  - **Encaminhamento:** processo de realizar um trecho da viagem.

# Sinergia entre Roteamento e Encaminhamento



# Estabelecimento de Conexão

- Terceira função importante em **algumas** redes.
  - *e.g.*, ATM, *frame relay*, X.25.
- Antes de iniciarem o fluxo de dados, *hosts* e roteadores intermediários estabelecem uma conexão virtual.
  - **Roteadores participam do processo.**
- Serviço orientado a conexão na camada de rede *vs.* na camada de transporte:
  - **Rede:** conexão entre dois *hosts* (pode também envolver roteadores intermediários em caso de circuitos virtuais).
  - **Transporte:** entre dois processos.

# Modelo de Serviço da Rede

- **Pergunta:** qual o **modelo de serviço** para o “canal” que transporta datagramas entre origem e destino?
- **Exemplos de serviço para datagramas individuais:**
  - Garantia de entrega.
  - Garantia de entrega com menos de 40 ms de atraso.
- **Exemplos de serviço para um fluxo de datagramas:**
  - Entrega ordenada.
  - Garantia de vazão mínima para o fluxo.
  - Restrições sobre alterações no espaçamento entre pacotes.

# Modelos de Serviço da Rede: Exemplos

| Arquitetura da Rede | Modelo de Serviço | Garantias?       |       |           |        | Aviso de Congestionamento |
|---------------------|-------------------|------------------|-------|-----------|--------|---------------------------|
|                     |                   | Banda            | Perda | Ordenação | Atraso |                           |
| Internet            | Melhor Esforço    | Não              | Não   | Não       | Não    | Não (inferida via perdas) |
| ATM                 | CBR               | Taxa Constante   | Sim   | Sim       | Sim    | Não há                    |
| ATM                 | VBR               | Taxa Garantida   | Sim   | Sim       | Sim    | Não há                    |
| ATM                 | ABR               | Mínima Garantida | Não   | Sim       | Não    | Sim                       |
| ATM                 | UBR               | Nenhuma          | Não   | Sim       | Não    | Não                       |



# Redes de Circuitos Virtuais

# Serviços Orientados e Não-Orientados a Conexão

- Redes de **datagramas** proveem serviço não-orientado a conexão na camada de rede.
- Redes de **circuitos virtuais** proveem serviço orientado a conexão na camada de rede.
- Análogo aos serviços do TCP/UDP com e sem conexão na camada de transporte, mas:
  - **Serviço:** *host a host*.
  - **Não há escolha:** rede provê um ou outro.
  - **Implementação:** no núcleo da rede.

# Circuitos Virtuais

- Caminho fim-a-fim se comporta de maneira similar a circuito telefônico.
  - Em termos de desempenho.
  - Ações da rede ao longo do caminho fim-a-fim.

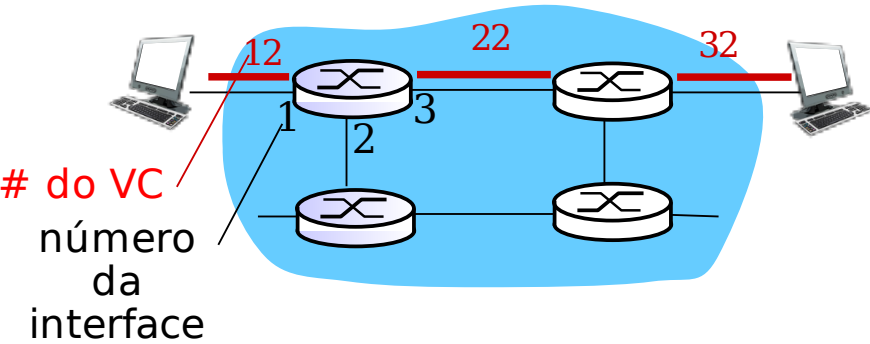
- Estabelecimento de chamada para cada conexão **antes** do fluxo de dados.
- Cada pacote carrega um **identificador de circuito virtual** (ou VC), ao invés de endereço do destinatário.
- Cada roteador no caminho fim-a-fim mantém “estado” para cada conexão passante.
- Recursos de enlaces, roteadores (banda, *buffers*) podem ser **alocados** para o VC.
  - Recursos dedicados = serviço previsível.

# Implementação de um Circuito Virtual

- Um VC consiste de:
  1. **Caminho** entre origem e destino.
  2. **Número(s) de identificação**, um para cada enlace no caminho.
  3. **Entradas nas tabelas de roteamento** nos roteadores do caminho.
- Pacote pertencente ao VC carrega o número do VC (ao invés do endereço do destinatário).
- Número do VC pode ser alterado a cada salto do caminho.
  - Novo número do VC vem da tabela de roteamento.

# Circuitos Virtuais: Tabela de Roteamento

tabela de roteamento  
no roteador de cima  
à esquerda:

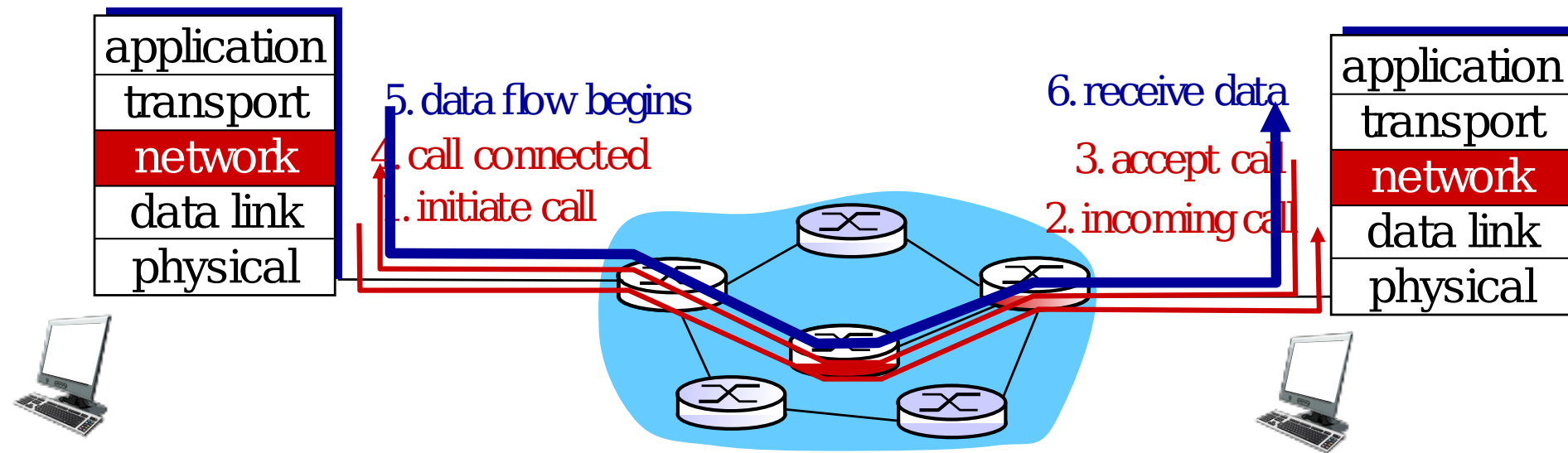


| Interface de entrada | # VC de chegada | Interface de saída | # VC de saída |
|----------------------|-----------------|--------------------|---------------|
| 1                    | 12              | 3                  | 22            |
| 2                    | 63              | 1                  | 18            |
| 3                    | 7               | 2                  | 17            |
| 1                    | 97              | 3                  | 87            |
| ...                  | ...             | ...                | ...           |

Em rede de circuitos virtuais, roteadores mantêm informação de estado da conexão!

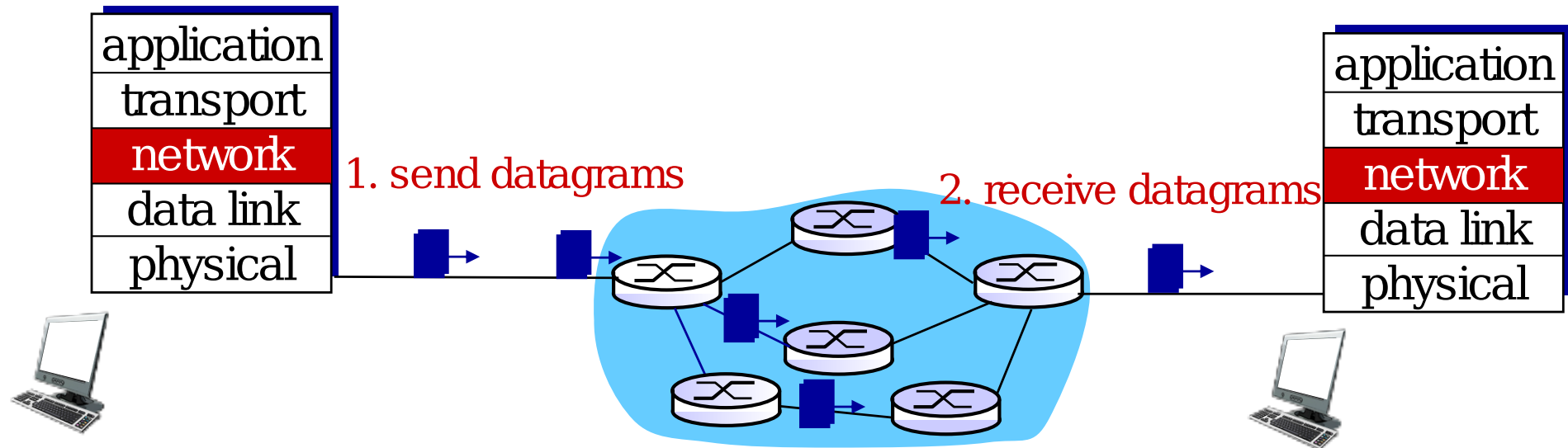
# Circuitos Virtuais: Protocolos de Sinalização

- Usados para estabelecimento, manutenção e finalização do VC.
- Usados em redes ATM, *frame relay*, X.25.
- Não são utilizados na Internet atual.

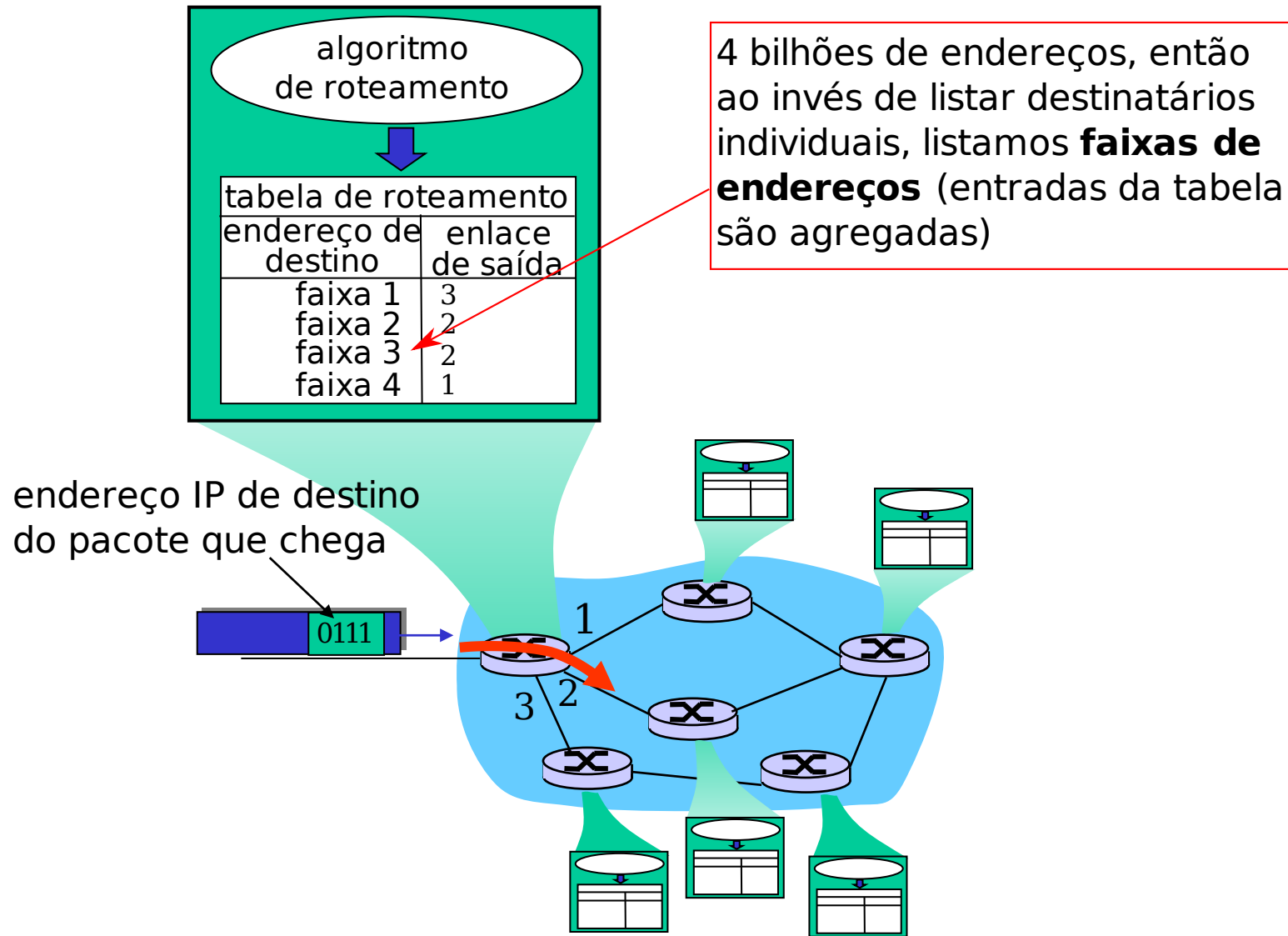


# Redes de Datagramas

- Sem estabelecimento de conexão na camada de rede.
- Roteadores: sem estado sobre conexões fim-a-fim.
  - Não há o conceito de “conexão” no nível da rede.
- Pacotes encaminhados usando o endereço de destino do *host*.



# Redes de Datagramas: Tabela de Roteamento (I)





# Redes de Datagramas: Tabela de Roteamento (II)

| Faixa de Endereços de Destino  | Enlace |
|--|--------|
| <b>11001000 00010111 00010000 00000000</b><br><b>até</b><br><b>11001000 00010111 00010111 11111111</b> | 0      |
| <b>11001000 00010111 00011000 00000000</b><br><b>até</b><br><b>11001000 00010111 00011000 11111111</b> | 1      |
| <b>11001000 00010111 00011001 00000000</b><br><b>até</b><br><b>11001000 00010111 00011111 11111111</b> | 2      |
| <b>Caso contrário</b>  | 3      |

- **Pergunta:** e se os endereços não são divididos de forma tão organizada?

# Casamento por Prefixo mais Longo

## Casamento por Prefixo mais longo

Ao procurar por uma entrada na tabela de roteamento para um destino, opte sempre pelo **prefixo mais longo** que casa com o endereço do destino.

| Faixa de Endereços de Destino    | Enlace |
|----------------------------------|--------|
| 11001000 00010111 00010*** ***** | 0      |
| 11001000 00010111 00011000 ***** | 1      |
| 11001000 00010111 00011*** ***** | 2      |
| Caso contrário                   | 3      |

- Exemplos:
  - Destino: 11001000 00010111 00010110 10100001. Qual interface?
  - Destino: 11001000 00010111 00011000 10101010. Qual interface?

# Datagrama ou Circuitos Virtuais: Por Quê?

- **Internet (datagrama):**

- Dados trocados entre computadores.
  - Serviço “elástico”, sem requisitos temporais estritos.
- Muitos tipos diferentes de enlaces.
  - Características variadas.
  - Difícil prover serviço uniforme.
- Dispositivos finais “inteligentes” (computadores).
  - Podem se adaptar, realizar controle, recuperação de erros.
  - **Núcleo simples, complexidade nas bordas.**

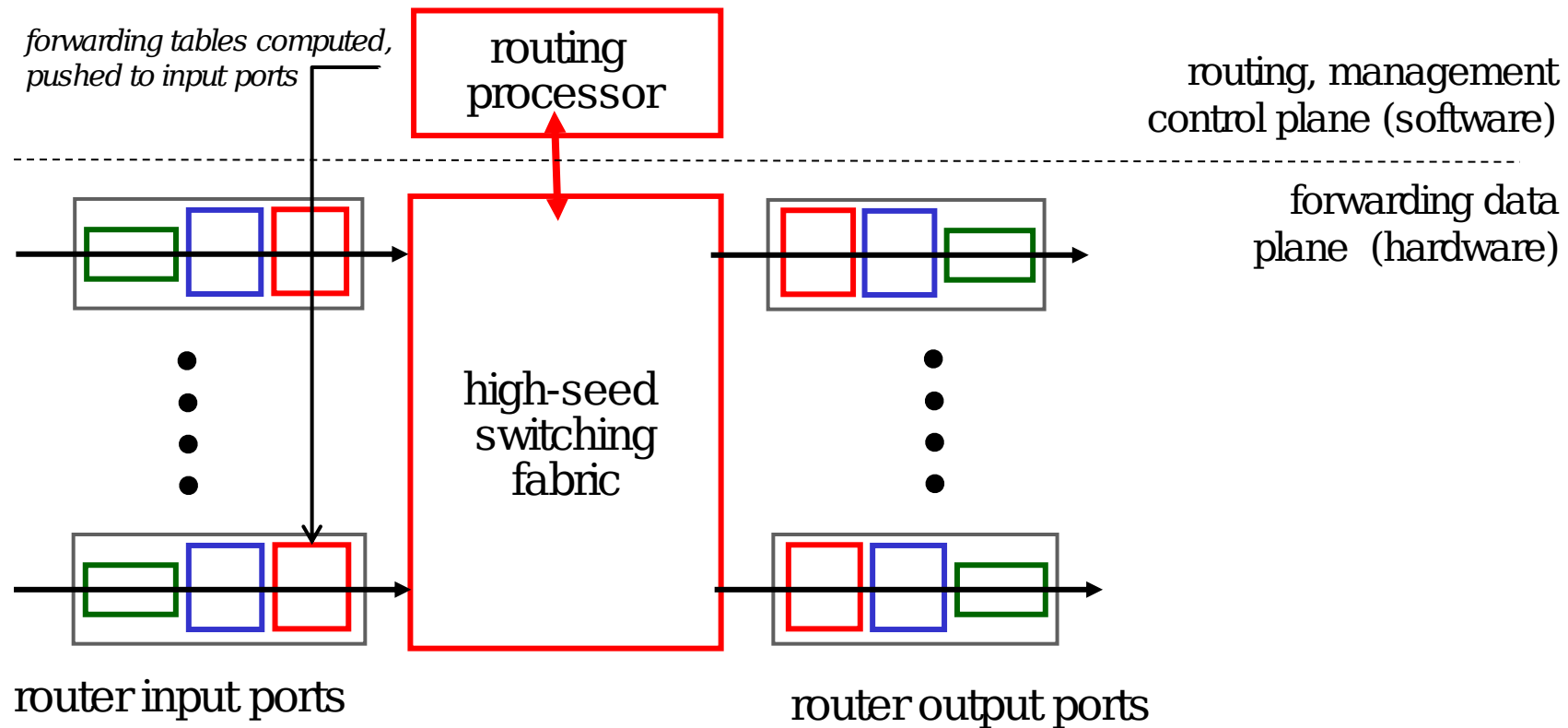
- **ATM (VC):**

- Evoluiu da telefonia.
- Conversação humana:
  - Requisitos temporais estritos.
  - Garantias de serviço necessárias.
- Sistemas finais “burros”.
  - Telefones.
  - **Complexidade no núcleo da rede.**

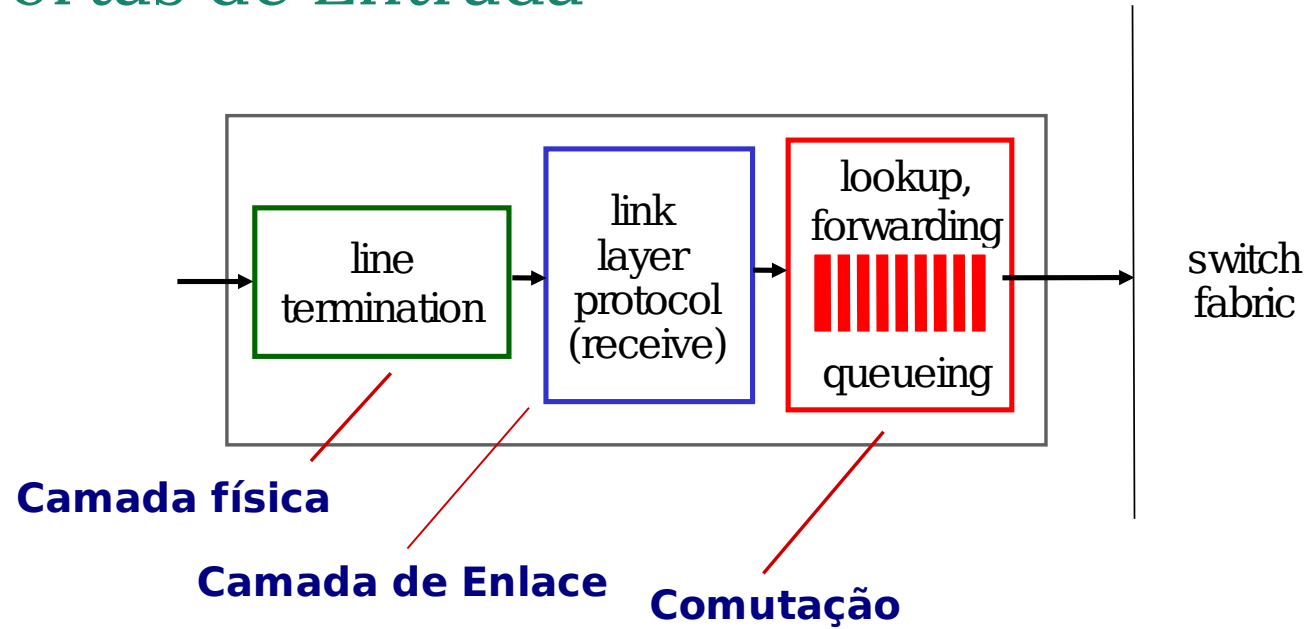
# Roteadores: Arquiteturas

# Aquiteturas de Roteadores: Visão Geral

- Duas funções chave em um roteador:
  - Execução de algoritmos/protocolos de roteamento (RIP, OSPF, BGP).
  - **Encaminhamento de datagramas de enlaces de entrada para enlaces de saída.**



# Funções das Portas de Entrada



- **Camada física:**

- Recepção no nível dos bits.

- **Camada de enlace:**

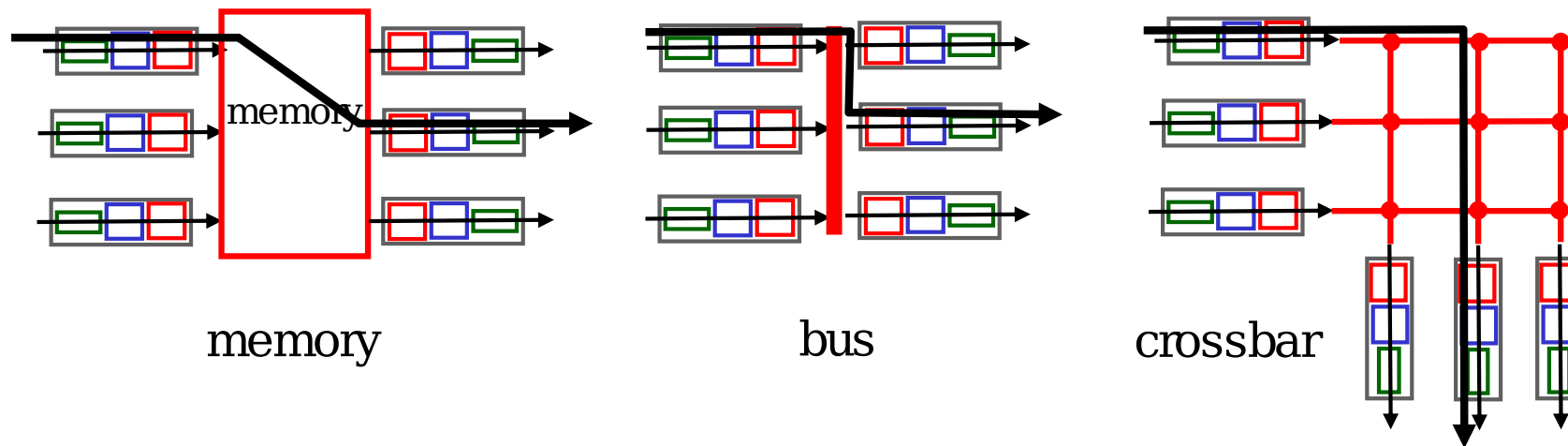
- e.g., Ethernet.
- Vide capítulo 5 (Redes II).

- **Comutação:**

- Dado destino do datagrama, procurar porta de saída usando tabela de roteamento em memória.
- Objetivo: completar processamento da porta de entrada na “velocidade de linha”.
- Enfileiramento: se datagramas chegam mais rápido que taxa de encaminhamento para dentro da malha de comutação.

# Malhas de Comutação

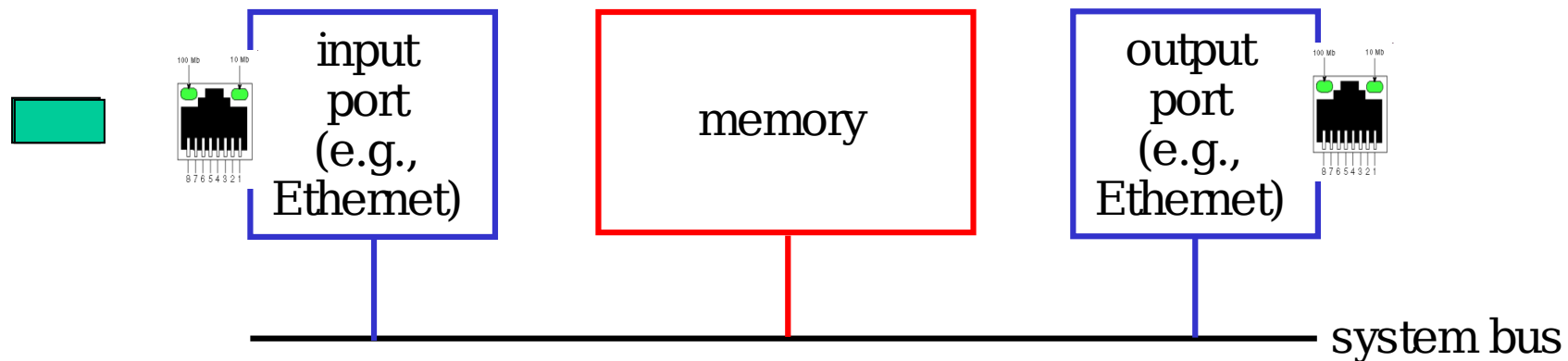
- Transferem pacotes do *buffer* da porta de entrada para o *buffer* da porta de saída apropriada.
- **Taxa de comutação:** taxa na qual pacotes podem ser transferidos das entradas para as saídas.
  - Comumente medida como um múltiplo da velocidade da linha das portas de entrada/saída.
  - N entradas: taxa de comutação desejada de N vezes a velocidade da linha.
- Três tipos básicos de malha de comutação:



# Comutação Através da Memória

- **Primeira geração de roteadores:**

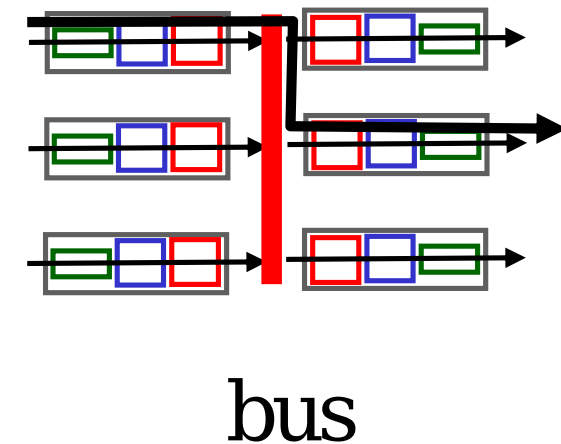
- Computadores comuns com comutação feita diretamente pelo processador.
- Pacotes copiados para a memória principal do sistema.
- Taxa limitada pela vazão da memória.
  - Dois usos do barramento por datagrama.





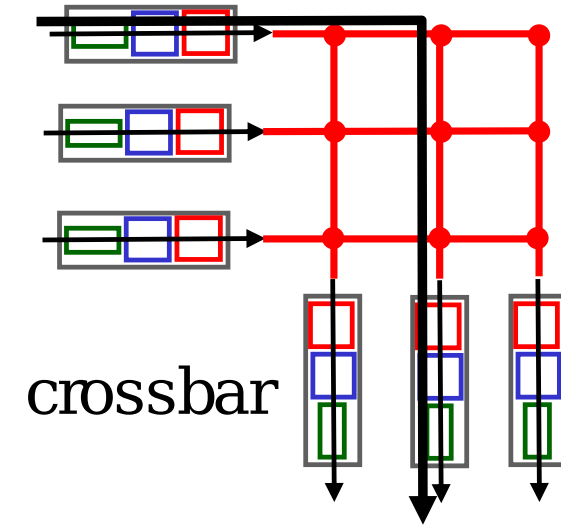
# Comutação Através de um Barramento

- Datagrama passa da porta de entrada para a porta de saída através de um barramento compartilhado.
- **Contenção no barramento:** taxa de comutação limitada pela banda do barramento.
- Barramento de 32 Gb/s, Cisto 5600: taxa suficiente para roteadores de acesso e empresariais.

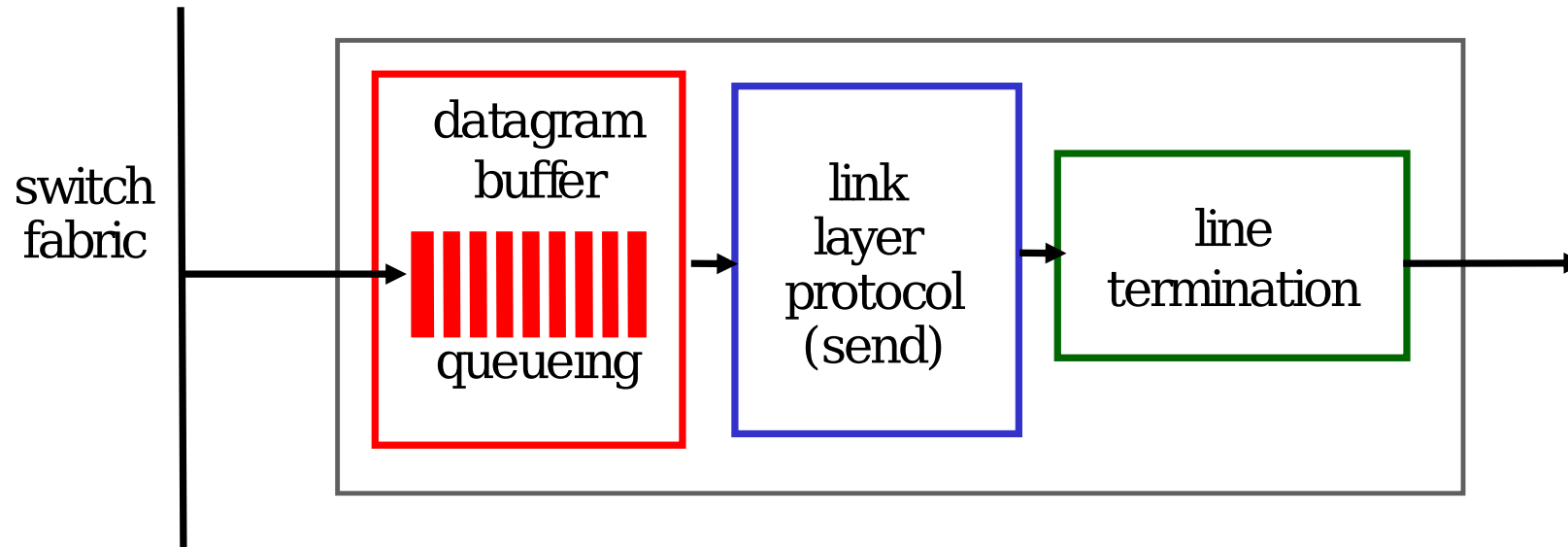


# Comutação Através de uma Rede de Interconexão

- Supera limitações de banda do barramento.
- Rede banyan, *crossbar*, outras redes de interconexão inicialmente desenvolvidas para conectar processadores em computadores multiprocessados.
- Projeto avançado: fragmentam datagramas em células de comprimento fixo comutadas pela malha.
- Cisco 12000: comuta 60 Gb/s através da rede de interconexão.

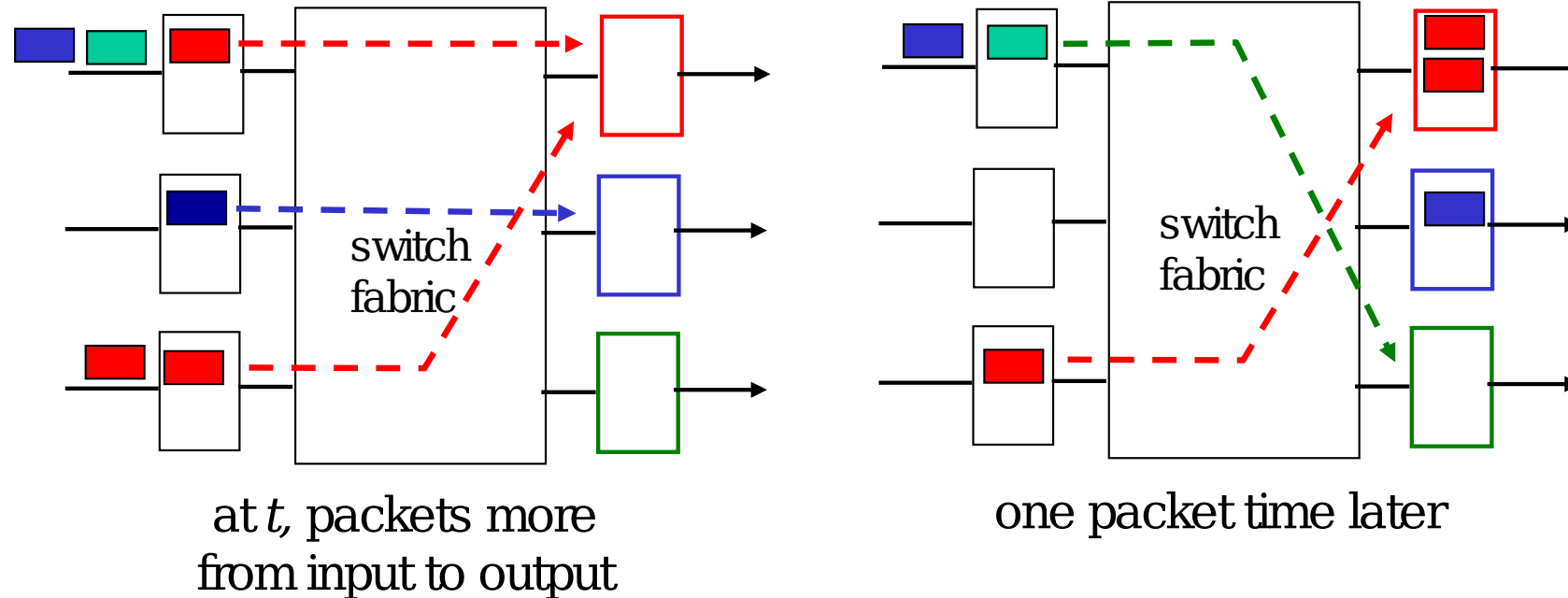


# Portas de Saída



- **Enfileiramento** necessário devido à taxa mais alta da malha de comutação.
  - Datagramas podem ser perdidos devido a congestionamento, falta de *buffer*.
- **Escalonamento** de datagramas.
  - Prioridades: quem recebe melhor desempenho? Neutralidade da rede?

# Enfileiramento na Porta de Saída



- Ocorre quando a taxa de chegada da malha de comutação excede velocidade da linha da porta de saída.
- **Enfileiramento (atraso) e perdas devido a overflow do buffer da porta de saída!**

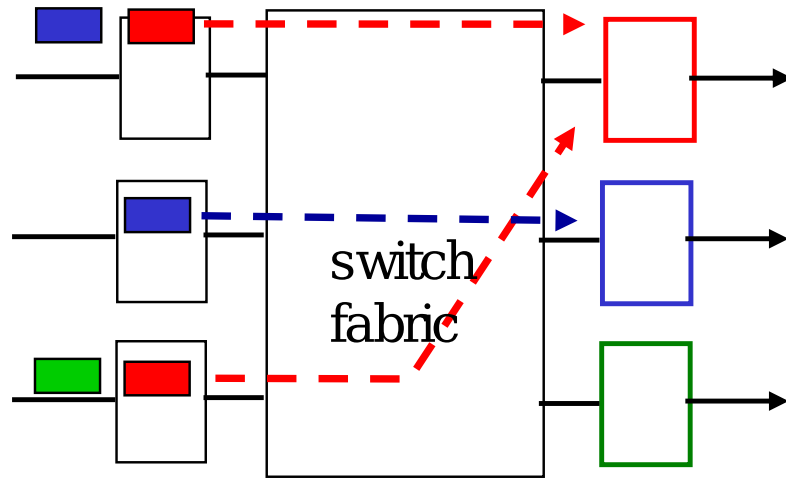
# Quanto Buffer Alocar?

- Recomendação da RFC 3439: *buffer* deve ser capaz de armazenar aproximadamente 250 ms de dados.
  - 250 ms: RTT “típico” da Internet.
  - Tamanho do *buffer*:  $RTT \times C$ .
    - Onde  $C$  é a capacidade do enlace.
    - e.g.,  $C = 10 \text{ Gb/s}$ ,  $RTT = 250 \text{ ms}$ , 2,5 Gb de *buffer*.
- Recomendação recente: com  $N$  fluxos, tamanho do *buffer* igual a:

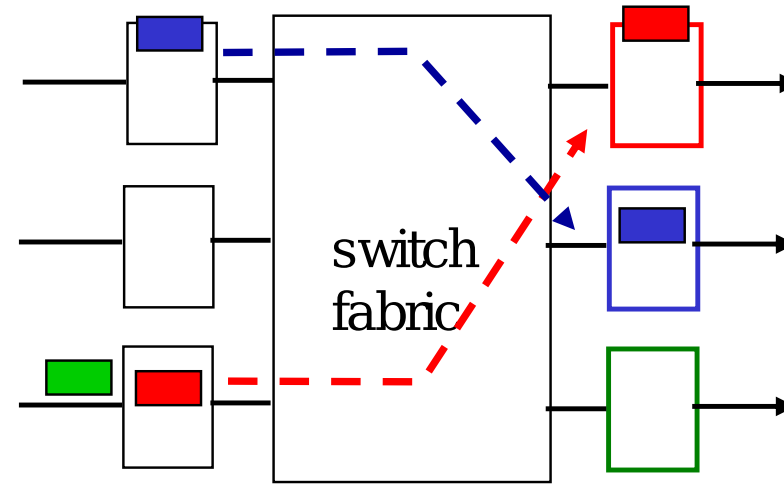
$$\frac{RTT \times C}{\sqrt{N}}$$

# Enfileiramento na Porta de Entrada

- Malha de comutação mais lenta que portas de entrada combinadas  $\Rightarrow$  enfileiramento pode ocorrer nas filas das portas de entrada.
  - **Enfileiramento (atraso) e perdas devido a overflow do buffer da porta de saída!**
- **Bloqueio de cabeça de linha (Head-of-line Blocking, ou HOL):** datagrama da frente da fila não permite que outros enfileirados sejam comutados.



output port contention:  
only one red datagram can be  
transferred.  
*lower red packet is blocked*



one packet time later:  
green packet  
experiences HOL  
blocking