

Redes de Computadores

Prof. Daniel Ludovico Guidoni

guidoni@ufop.edu.br

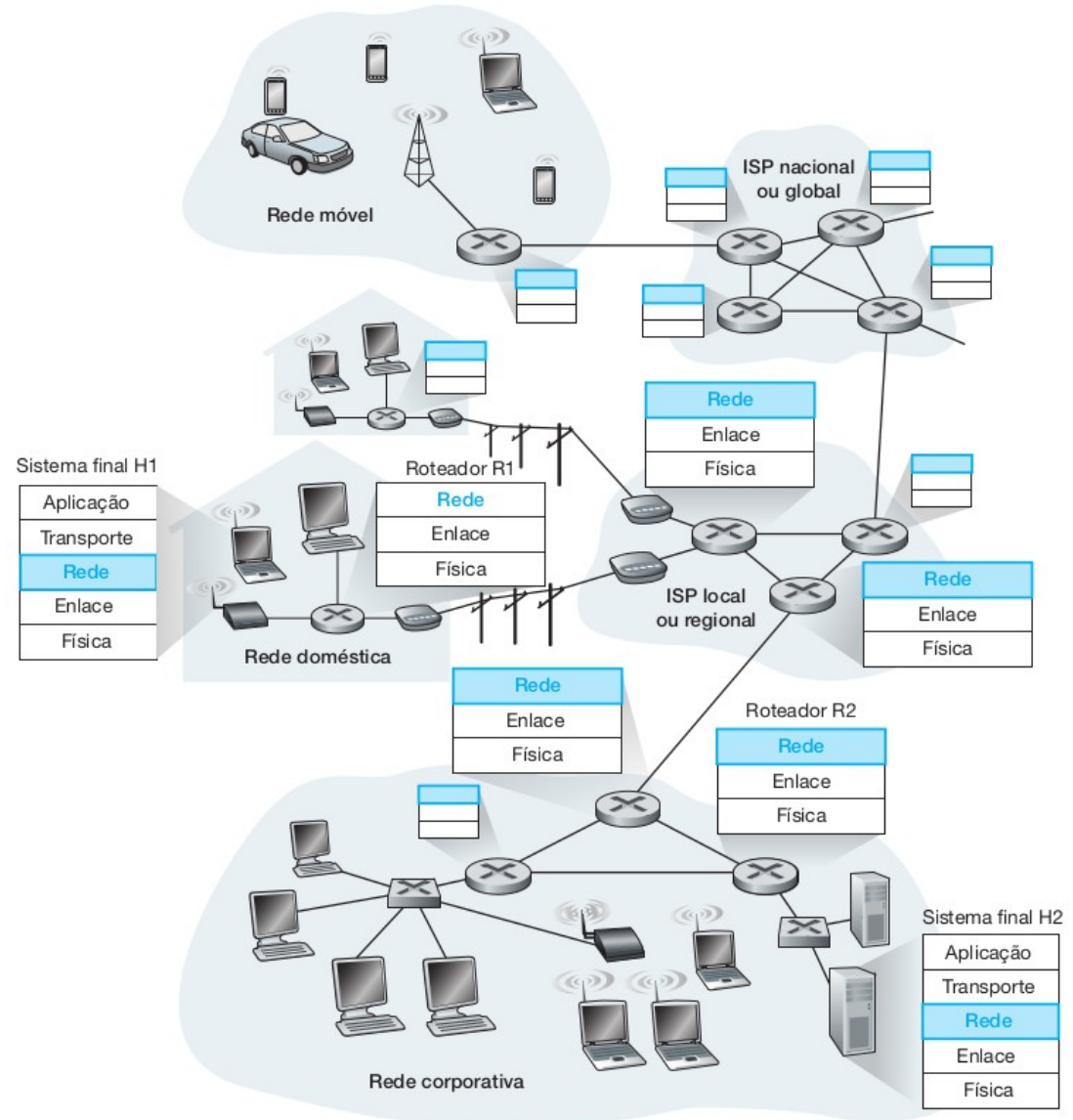
Redes de Computadores

Capítulo 4: Camada de rede

Introdução

A Camada de Rede

- Transporta segmentos da estação remetente à receptora
- No lado remetente, encapsula segmentos dentro de datagramas
- No lado receptor, entrega os segmentos para a camada de transporte
- Protocolos da camada de rede em todos os sistemas finais e roteadores
- Roteadores examinam campos de cabeçalho de todos os datagramas IP que passam por eles



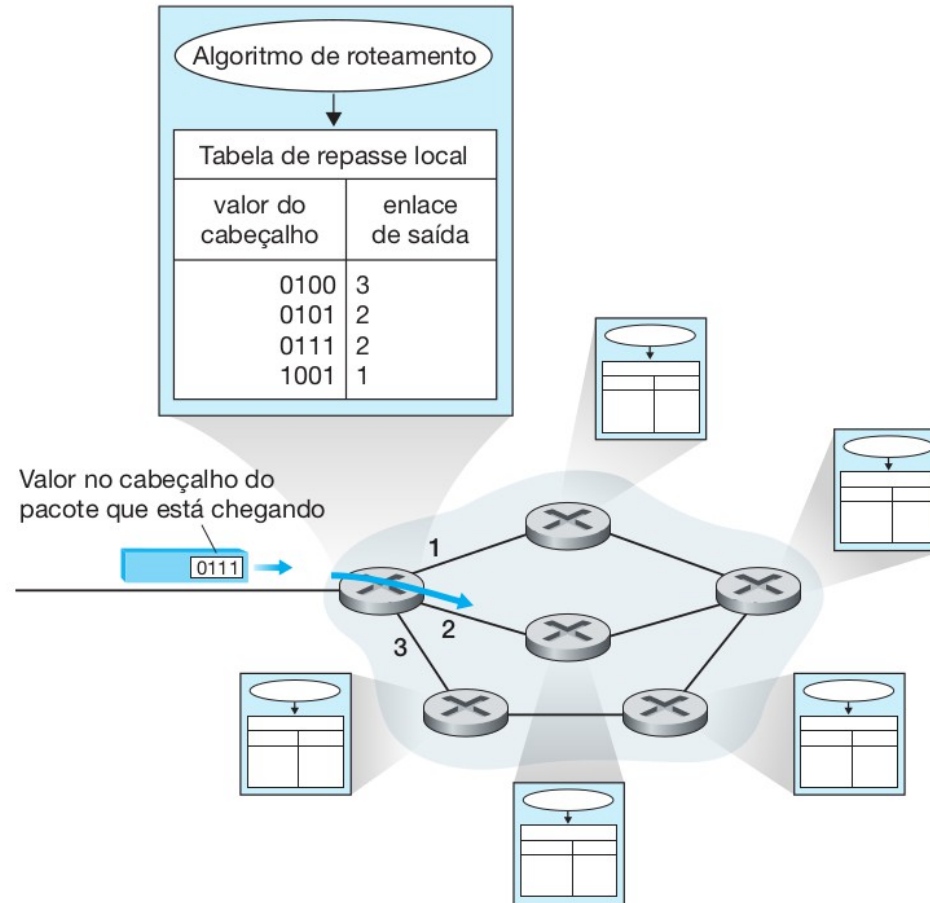
Funções da camada de rede

- Repasse: move pacotes de uma entrada do roteador para uma saída apropriada
- Roteamento: determina a rota a ser seguida pelos pacotes da fonte até o destino
 - Algoritmos de roteamento

Funções da camada de rede

- Cada roteador possui uma tabela de repasse
- O roteador examina o cabeçalho do pacote que está chegando e utiliza esse valor para indexar a tabela de repasse
- O resultado da tabela de repasse indica a interface de saída que o pacote deve ser repassado
- Modificações na tabela de repasse são realizadas pelos algoritmos de roteamento

Tabela de repasse e algoritmo de roteamento



Modelos de serviços

- Entrega garantida
- Entrega garantida com atraso limitado
- Entrega de pacotes na ordem
- Largura de banda mínima garantias
- *Jitter* máximo garantido (tempo entre duas trasnsmissões e recepções)
- ...

Modelo de serviço da camada de rede da Internet

- Melhor esforço

Redes de Computadores

Capítulo 4: A camada de rede

O que há dentro de um roteador

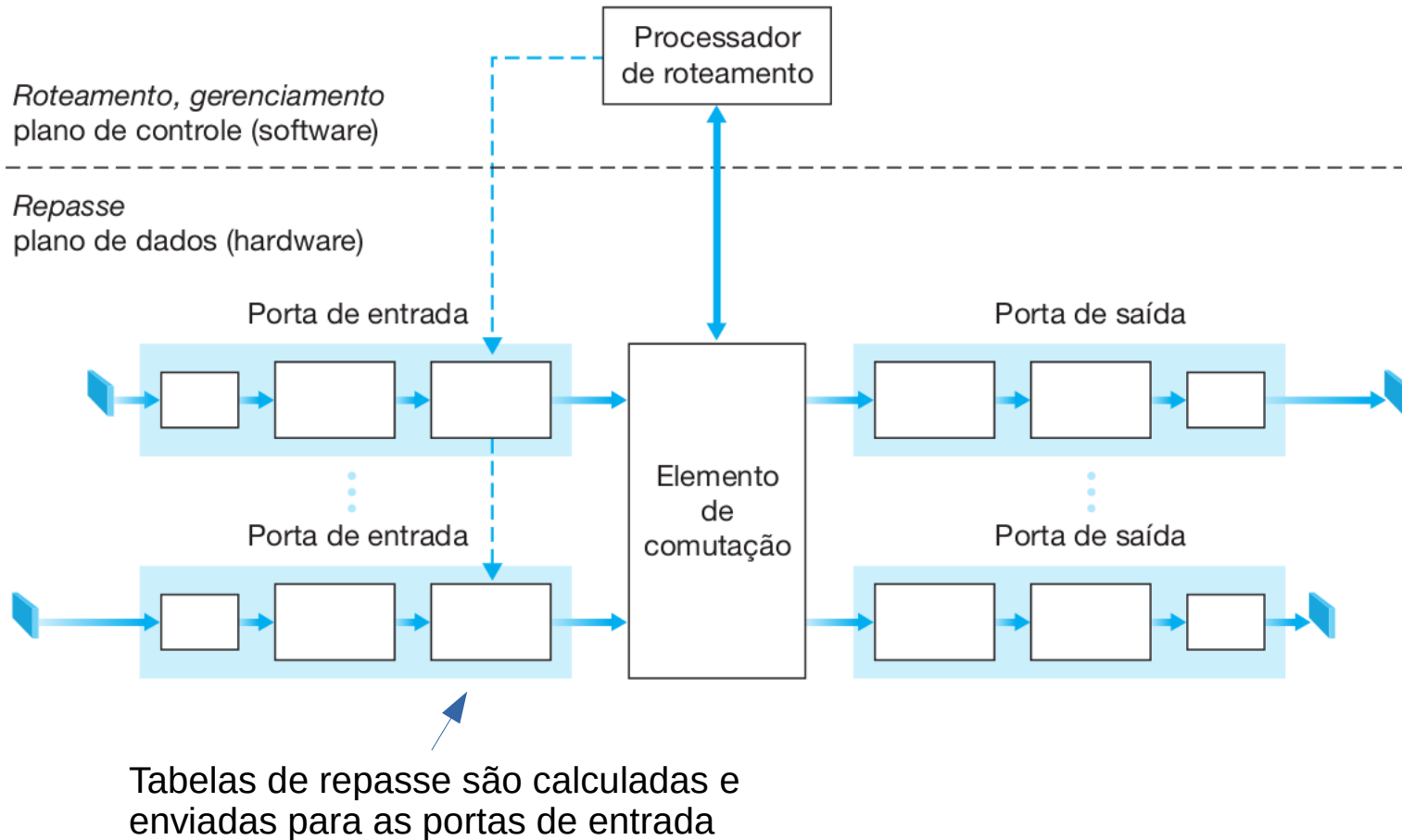
Roteadores



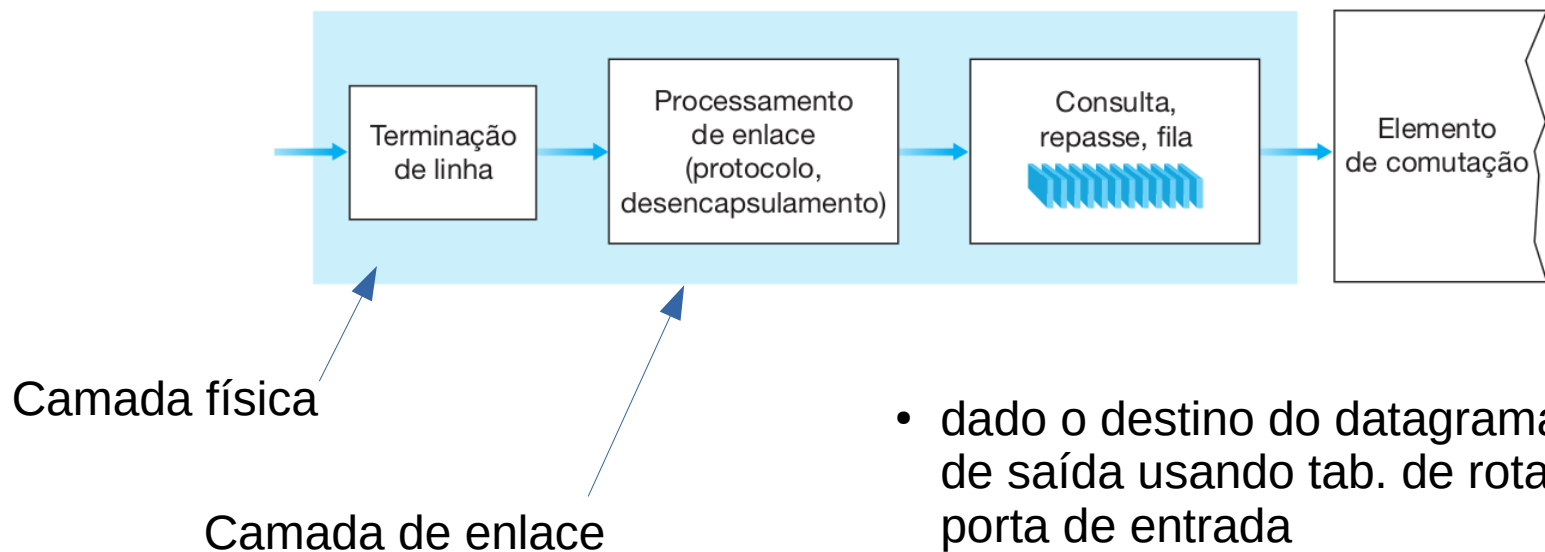
Roteadores

- Executam algoritmos/protocolos de roteamento (RIP, OSPF, BGP)
- Repassam datadragas do enlace de entrada para o enlace de saída utilizando a tabela de repasse

Arquitetura de roteadores



Funções das portas de entrada



- dado o destino do datagrama, procura porta de saída usando tab. de rotas na memória da porta de entrada
- **meta**: completar processamento da porta de entrada na 'velocidade da linha
- **filas**: se datagramas chegam mais rápido que taxa de re-envio para elemento de comutação

Repasse com base no destino

Faixa de endereços de destino	Interface de enlace
11001000 00010111 00010000 00000000 até 11001000 00010111 00010111 11111111	0
11001000 00010111 00011000 00000000 até 11001000 00010111 00011000 11111111	1
11001000 00010111 00011001 00000000 até 11001000 00010111 00011111 11111111	2
senão	3

Utilização do prefixo mais longo

Concordância do prefixo	Interface do enlace
11001000 00010111 00010*** *****	0
11001000 00010111 00011000 *****	1
11001000 00010111 00011*** *****	2
senão	3

Destino: 11001000 00010111 00010110 10100001

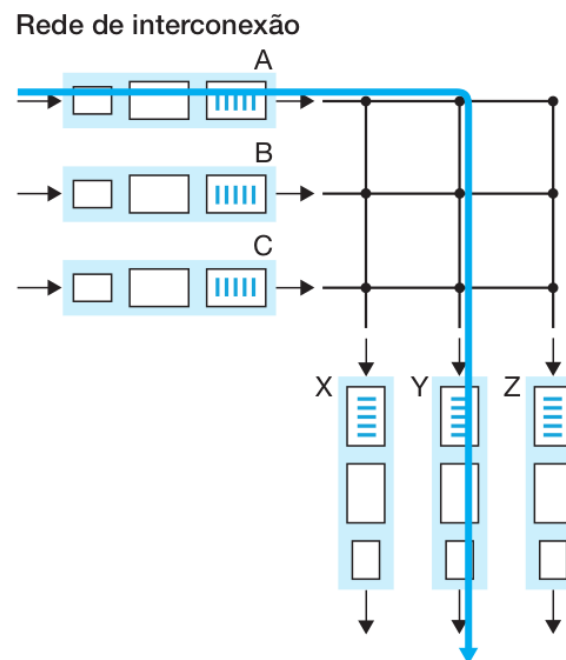
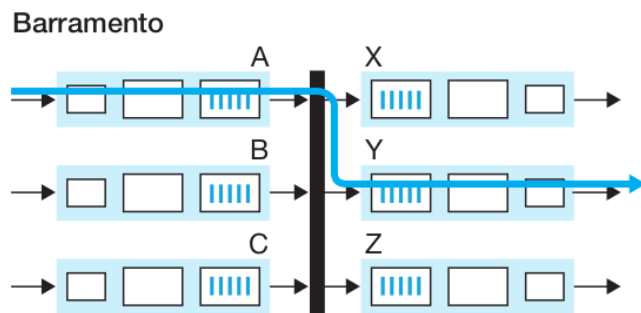
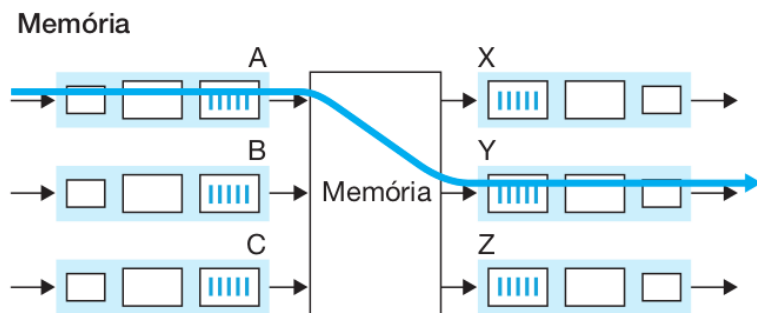
Destino: 11001000 00010111 00011000 10101010

qual interface?

Elemento de comutação

- Transfere pacotes do buffer de entrada para o buffer de saída apropriado
- Taxa de comutação: taxa na qual os pacotes podem ser transferidos das entradas para as saídas:
 - frequentemente medida como múltiplo das taxas das linhas de entrada/saída
 - N entradas: desejável taxa de comutação N vezes a taxa da linha.

Técnicas de comutação



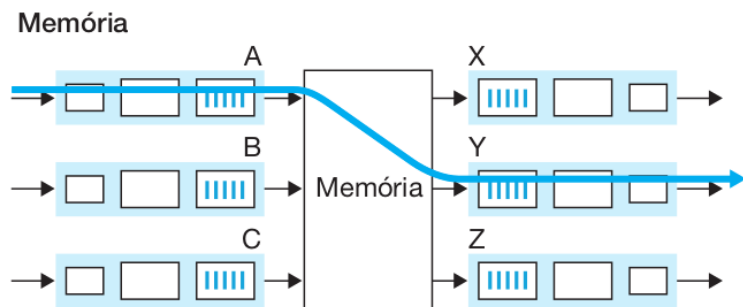
Legenda:

□ □ █ Porta de entrada █ □ □ Porta de saída

Comutação por memória

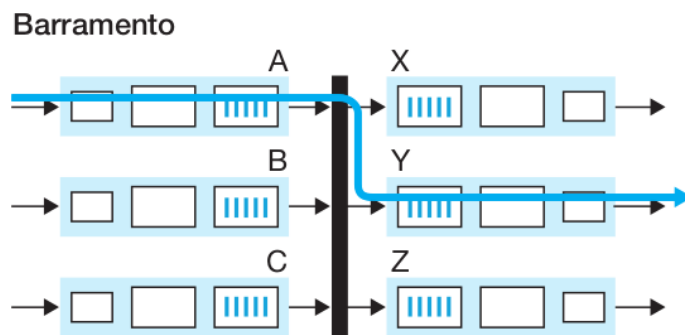
Roteadores da primeira geração:

- computadores tradicionais com comutação controlada diretamente pela CPU
- pacote copiado para a memória do sistema
- velocidade limitada pela largura de banda da memória (2 travessias do barramento por datagrama)



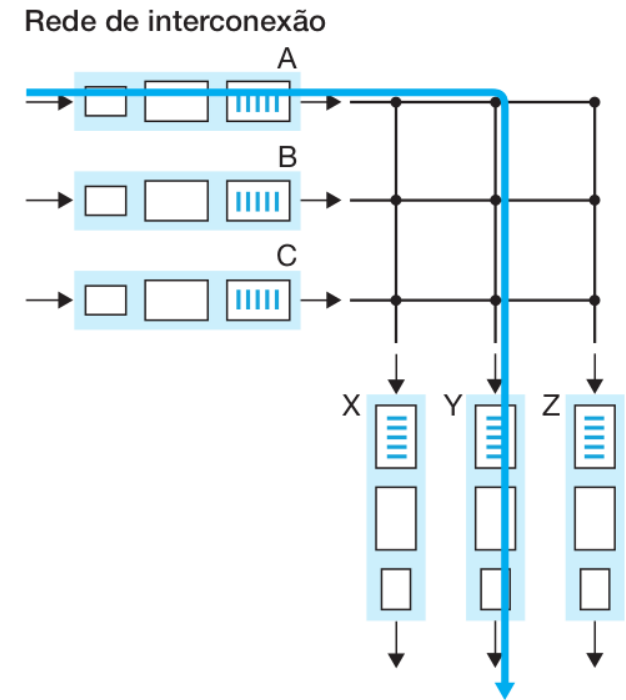
Comutação por um barramento

- Datagrama da memória da porta de entrada para a memória da porta de saída via um barramento compartilhado
- Disputa (contenção) pelo barramento: taxa de comutação limitada pela largura de banda do barramento
- Cisco 6500 usa barramento de 32 Gbps: velocidade suficiente para roteadores de acesso e corporativos.

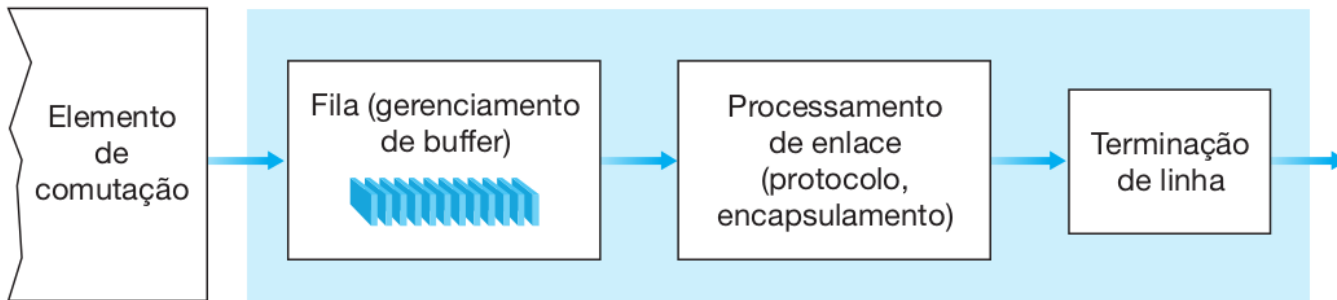


Comutação por rede de interconexão

- Supera limitações de banda dos barramentos
- Redes Banyan, outras redes de interconexão desenvolvidas inicialmente para interligar processadores num sistema multiprocessador
- Projeto avançado: fragmentar datagrama em células de tamanho fixo, comutar células através da matriz de comutação.
- Cisco 12000: comuta 60 Gbps pela rede de interconexão.

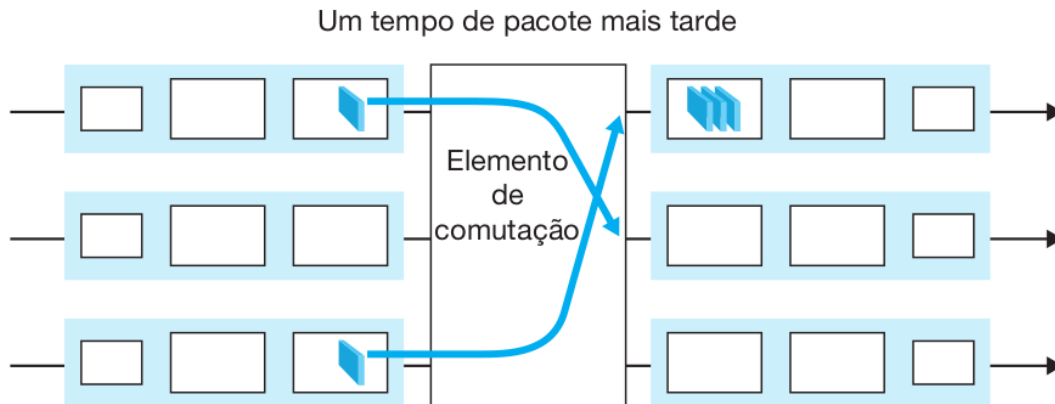
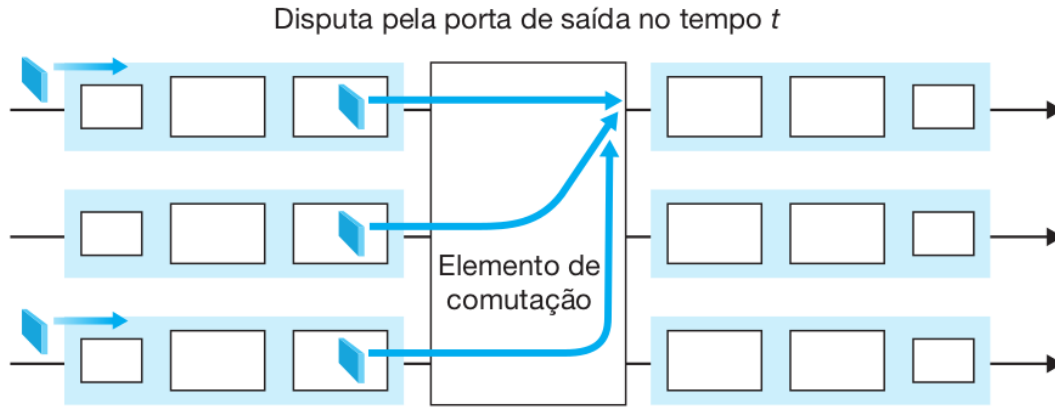


Portas de saídas



- **Enfileiramento** necessário quando datagramas chegam do elemento de comutação mais rapidamente do que a taxa de transmissão
- **Escalonamento** escolhe um dos datagramas enfileirados para transmissão

Filas nas portas de saída



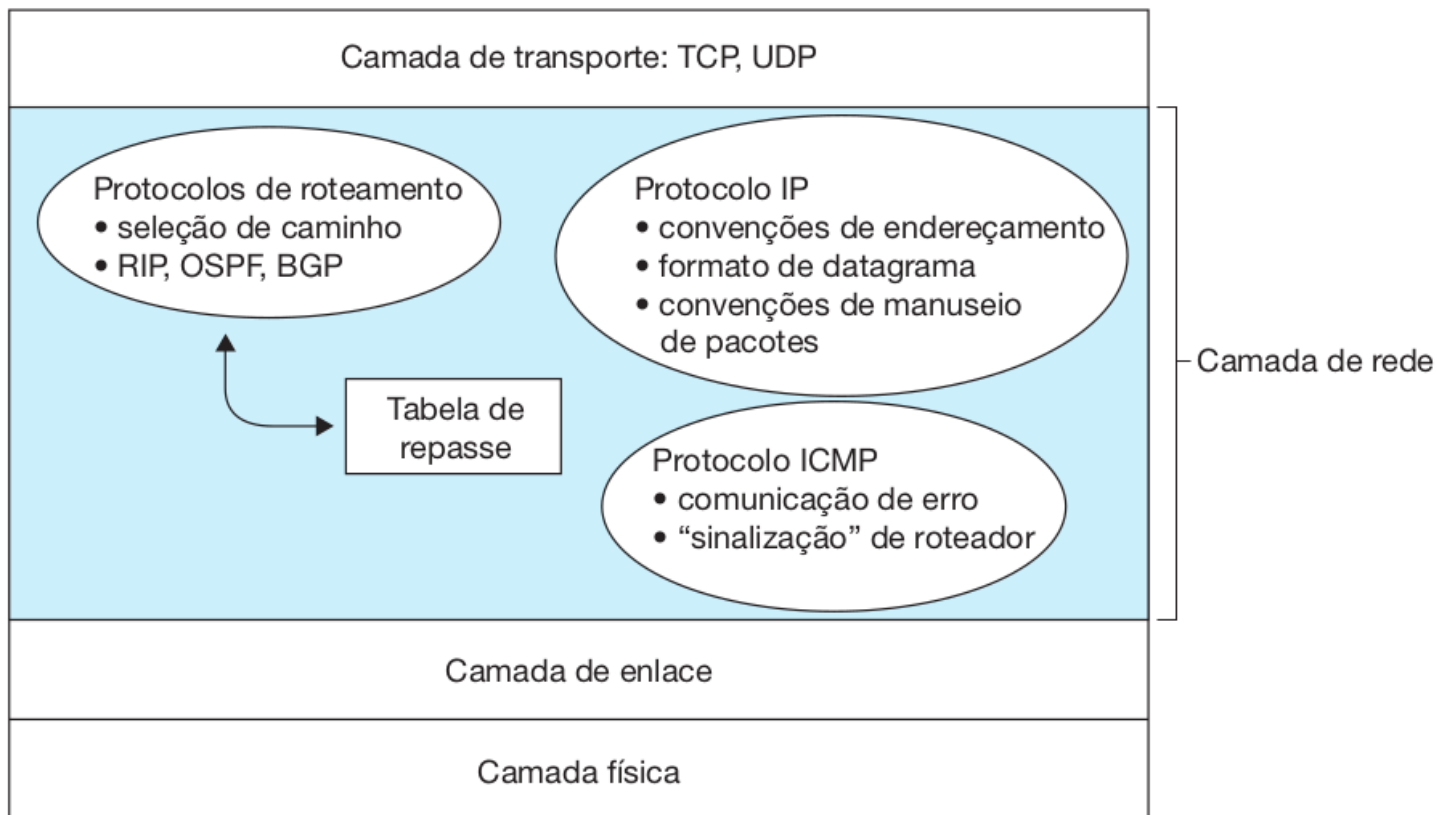
- Usa *buffers* quando taxa de chegada através do comutador excede taxa de transmissão de saída
- Enfileiramento (retardo), e perdas devidas ao transbordo do buffer da porta de saída!

Redes de Computadores

Capítulo 4: A camada de rede

O protocolo da internet (IP)

A camada de rede da Internet

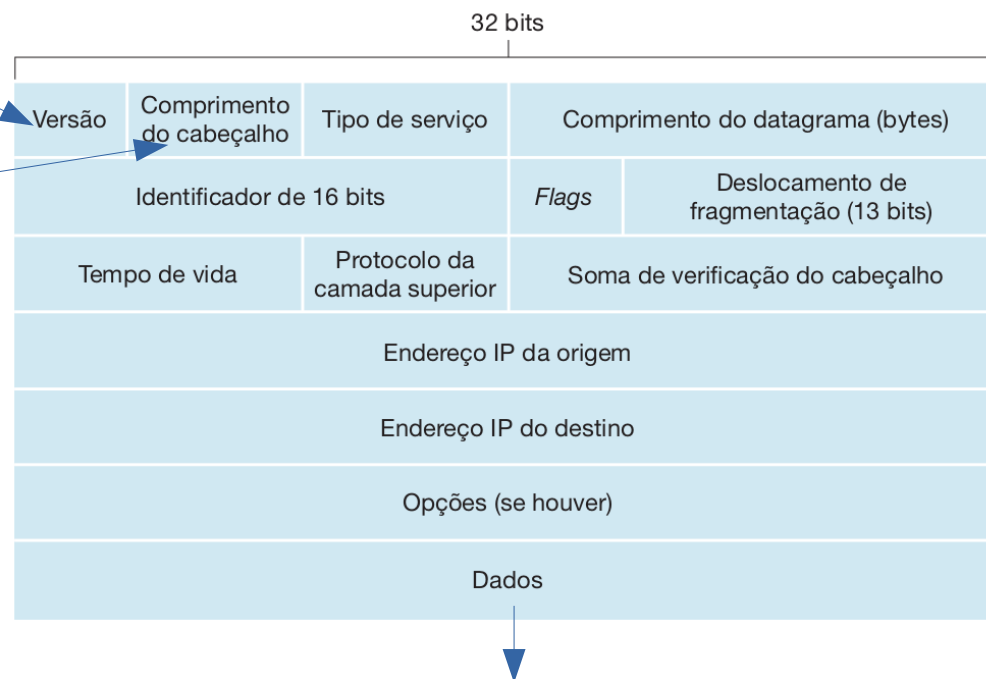


O formato do datagrama IP

Versão: número de versão do protocolo IP

Comprimento do cabeçalho em bytes

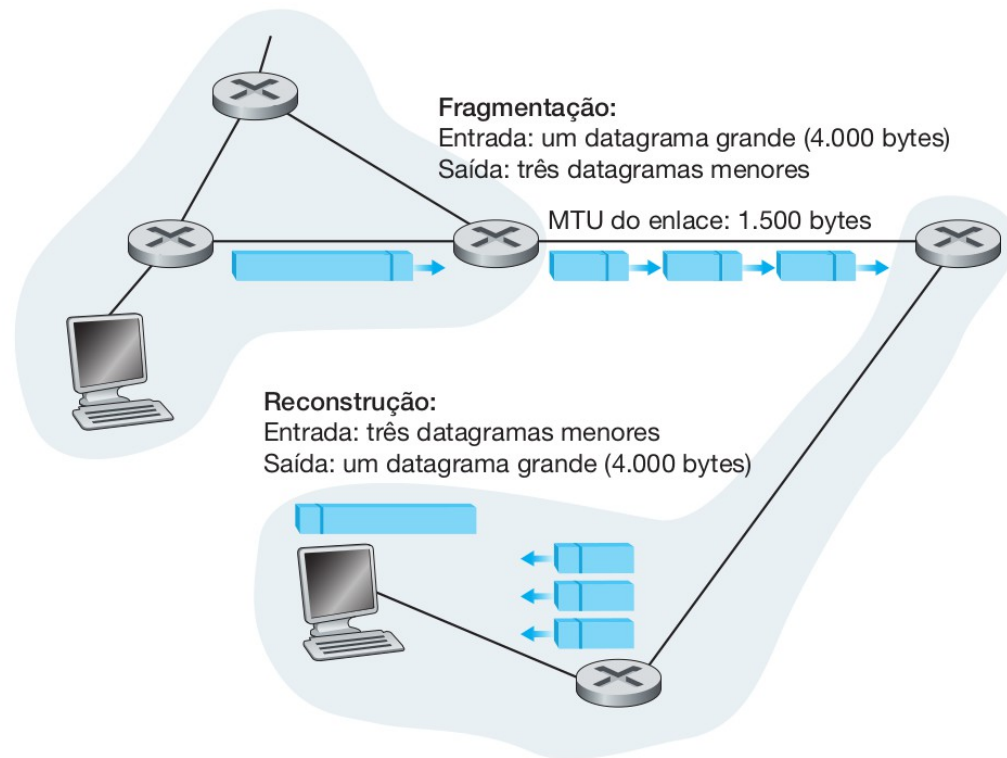
Número máximo de enlaces restantes



Tipicamente um segmento TCP ou UDP

IP: Fragmentação e remontagem

- Cada enlace de rede tem MTU (max. transmission unit)
 - maior tamanho possível de quadro neste enlace
 - tipos diferentes de enlace têm MTUs diferentes
- datagrama IP muito grande é dividido (“fragmentado”) dentro da rede
 - um datagrama vira vários datagramas
 - “remontado” apenas no destino final
 - bits do cabeçalho IP usados para identificar, ordenar fragmentos relacionados



IP: Fragmentação e remontagem

- Exemplo
 - Datagrama com 4000 bytes com identificador 777
 - MTU de 1500 bytes

Fragmento	Bytes	ID	Deslocamento	Flag
1º fragmento	1.480 bytes no campo de dados do datagrama IP	identificação = 777	0 (o que significa que os dados devem ser inseridos a partir do byte 0)	1 (o que significa que há mais)
2º fragmento	1.480 bytes de dados	identificação = 777	185 (o que significa que os dados devem ser inseridos a partir do byte 1.480. Note que $185 \times 8 = 1.480$)	1 (o que significa que há mais)
3º fragmento	1.020 bytes de dados (= $3.980 - 1.480 - 1.480$)	identificação = 777	370 (o que significa que os dados devem ser inseridos a partir do byte 2.960. Note que $370 \times 8 = 2.960$)	0 (o que significa que esse é o último fragmento)

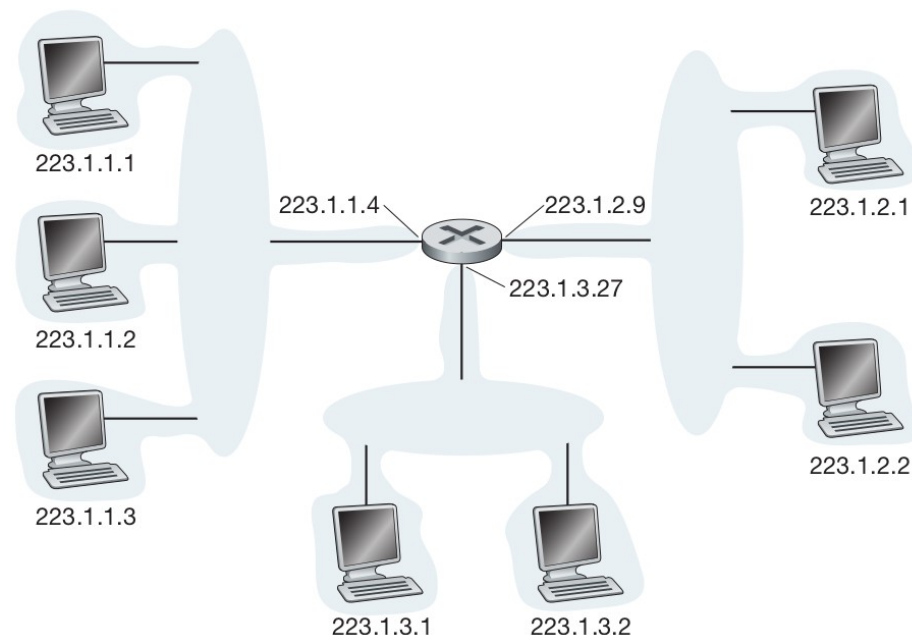
Redes de Computadores

Capítulo 4: A camada de rede

O protocolo da internet (IP)
continuação...

Endereçamento IP

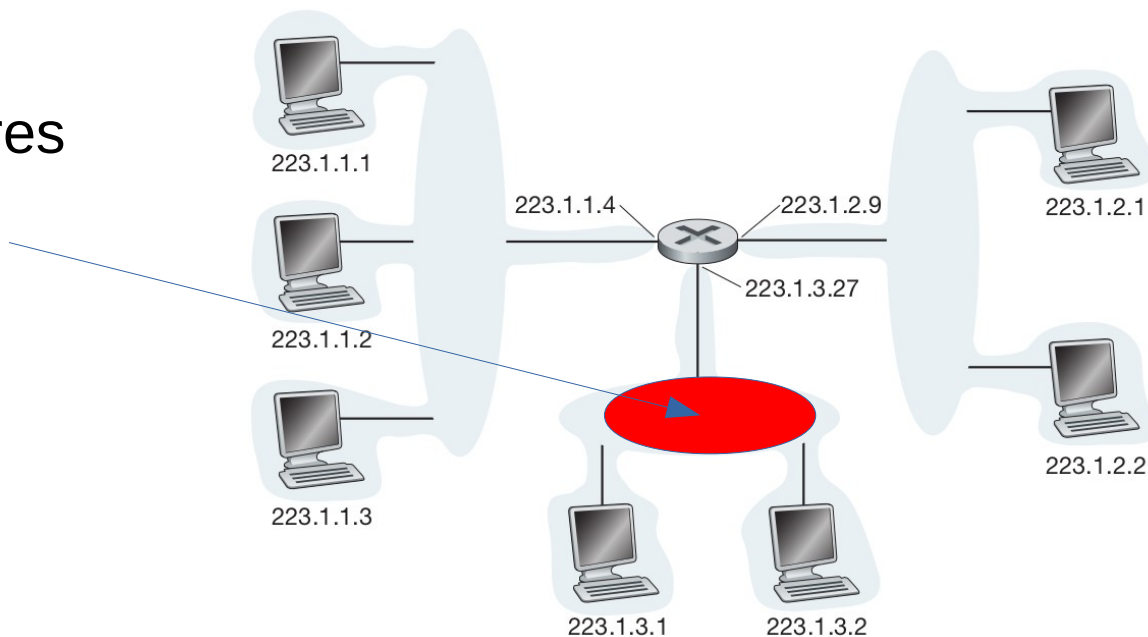
- **endereço IP:** ident. de 32-bits para interface de estação, roteador
- **interface:** conexão entre estação, roteador e enlace físico
 - roteador típico tem múltiplas interfaces
 - estação típica possui uma ou duas interfaces (ex.: Ethernet e Wi-fi)
 - **endereços IP** associados a cada interface



$$223.1.1.1 = \underbrace{11011111}_{223} \underbrace{00000001}_1 \underbrace{00000001}_1 \underbrace{00000001}_1$$

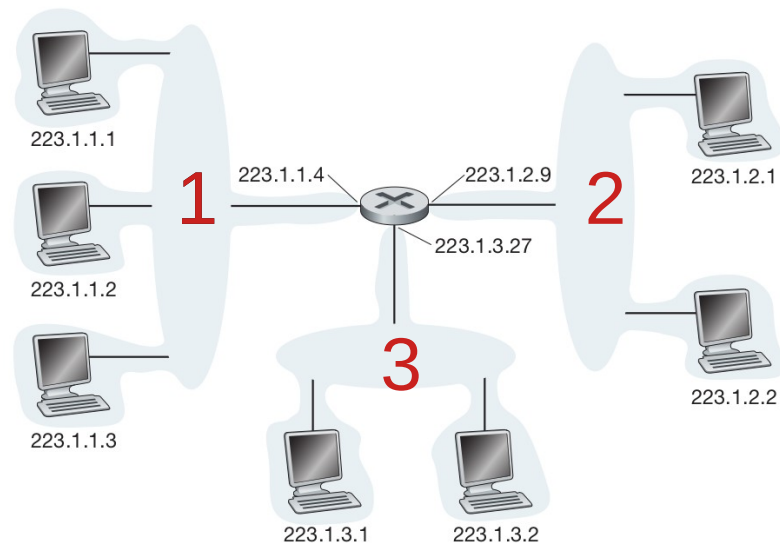
Endereçamento IP

- As interfaces são conectadas por roteadores ou *switchs*



Sub-redes

- endereço IP:
 - parte de rede (bits de mais alta ordem)
 - parte de estação (bits de mais baixa ordem)
- O que é uma subrede IP?
 - interfaces de dispositivos com a mesma parte de subrede nos seus endereços IP
 - podem alcançar um ao outro **sem passar por um roteador intermediário**

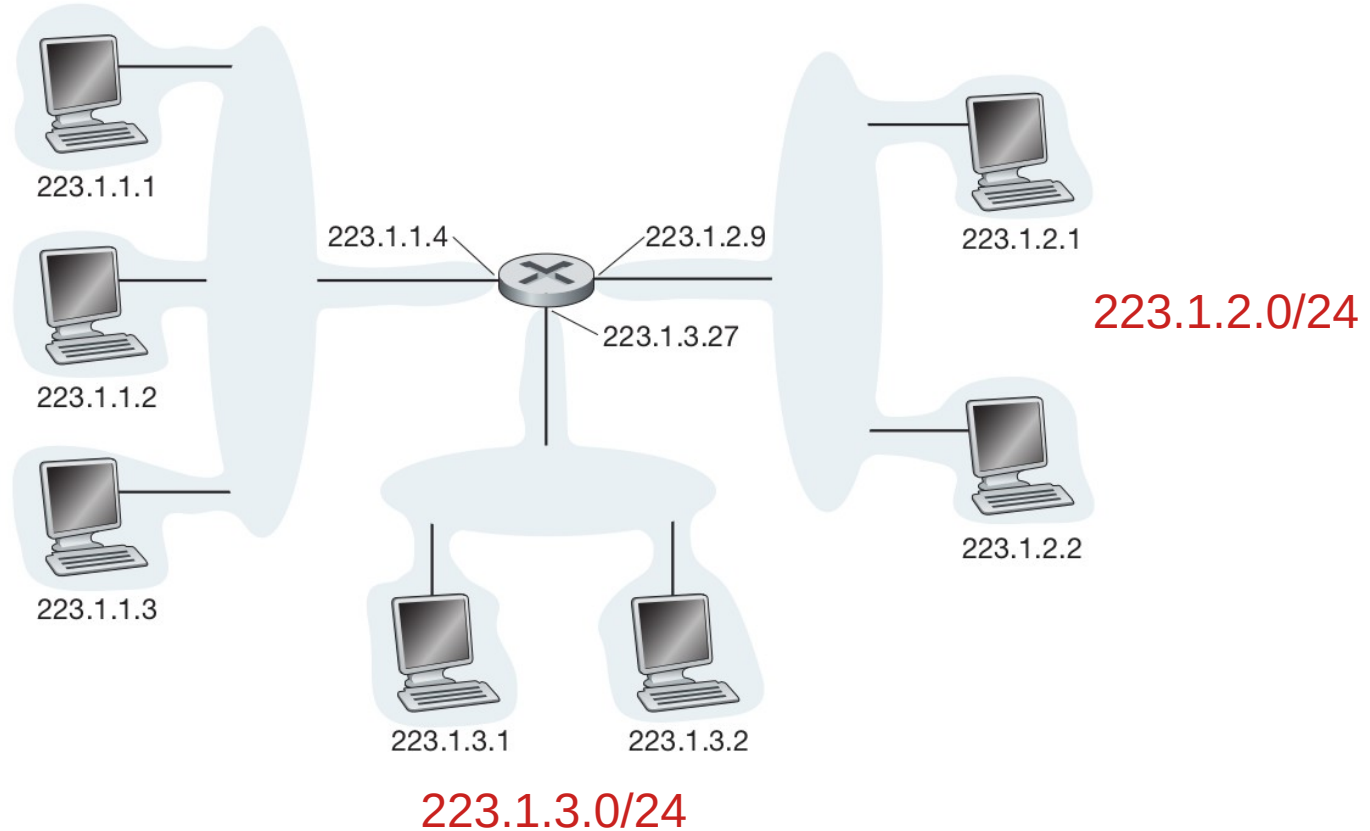


Rede composta por 3 sub-redes

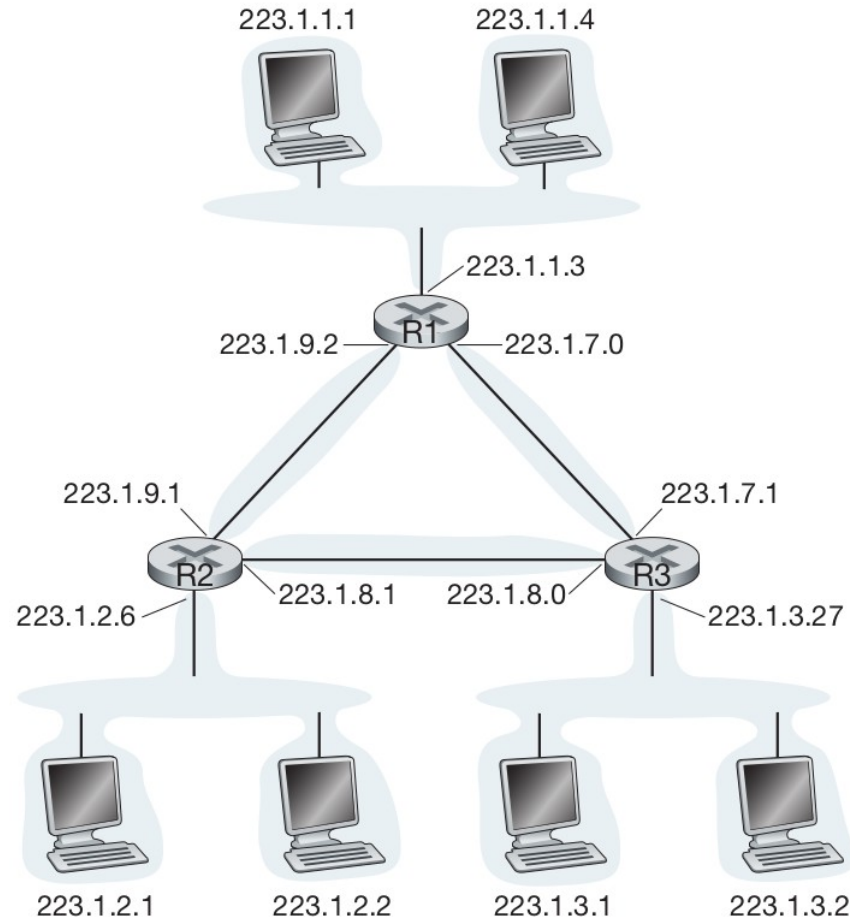
Sub-redes

Máscara de rede: /24

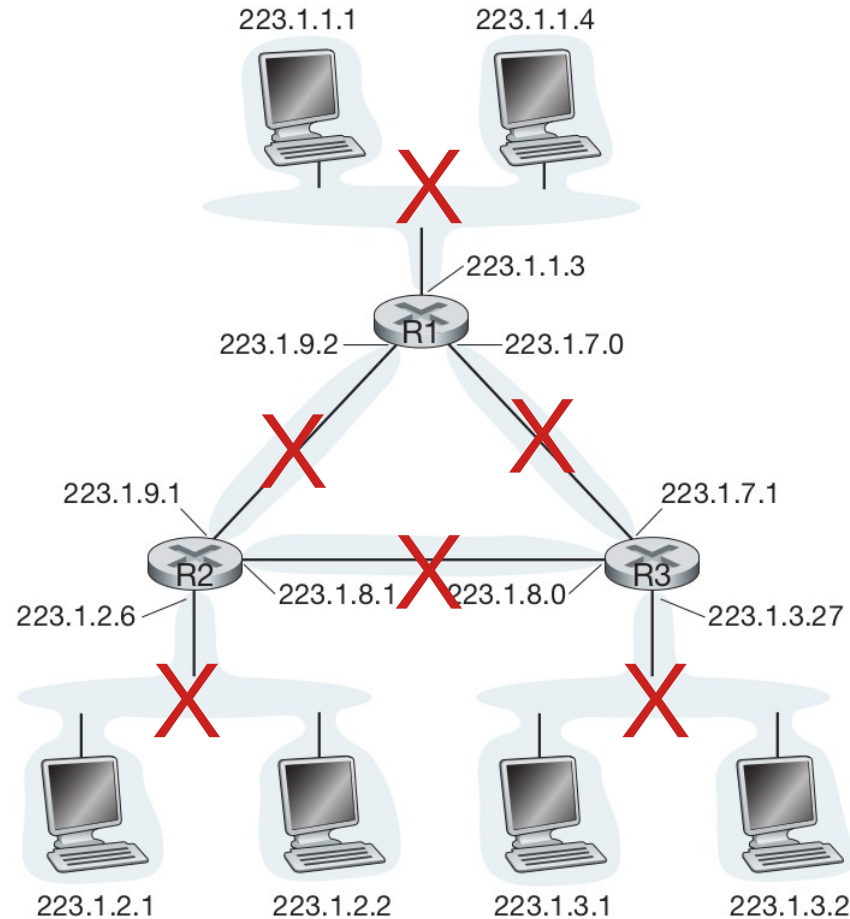
223.1.1.0/24



Quantas sub-redes existem na figura?

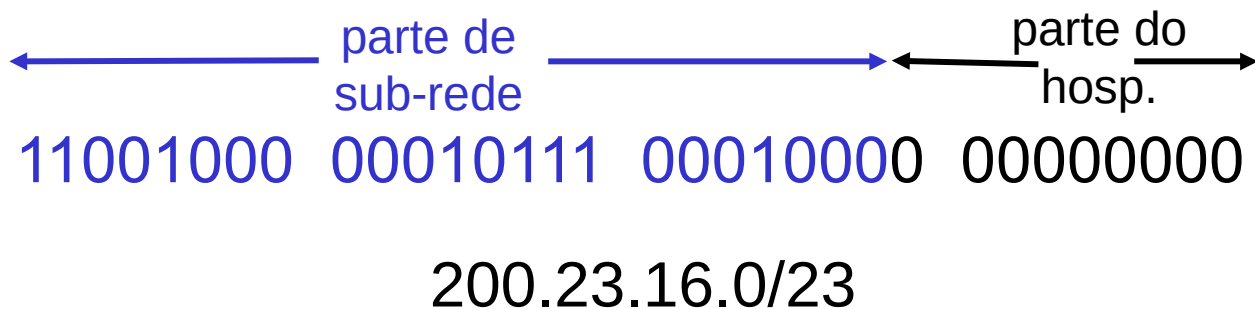


Quantas sub-redes existem na figura?



Endereçamento IP: CIDR

- CIDR: Classless Inter Domain Routing (Roteamento Interdomínio sem classes)
 - parte de rede do endereço de comprimento arbitrário
 - formato de endereço: a.b.c.d/x, onde x é o número de bits na parte de subrede do endereço



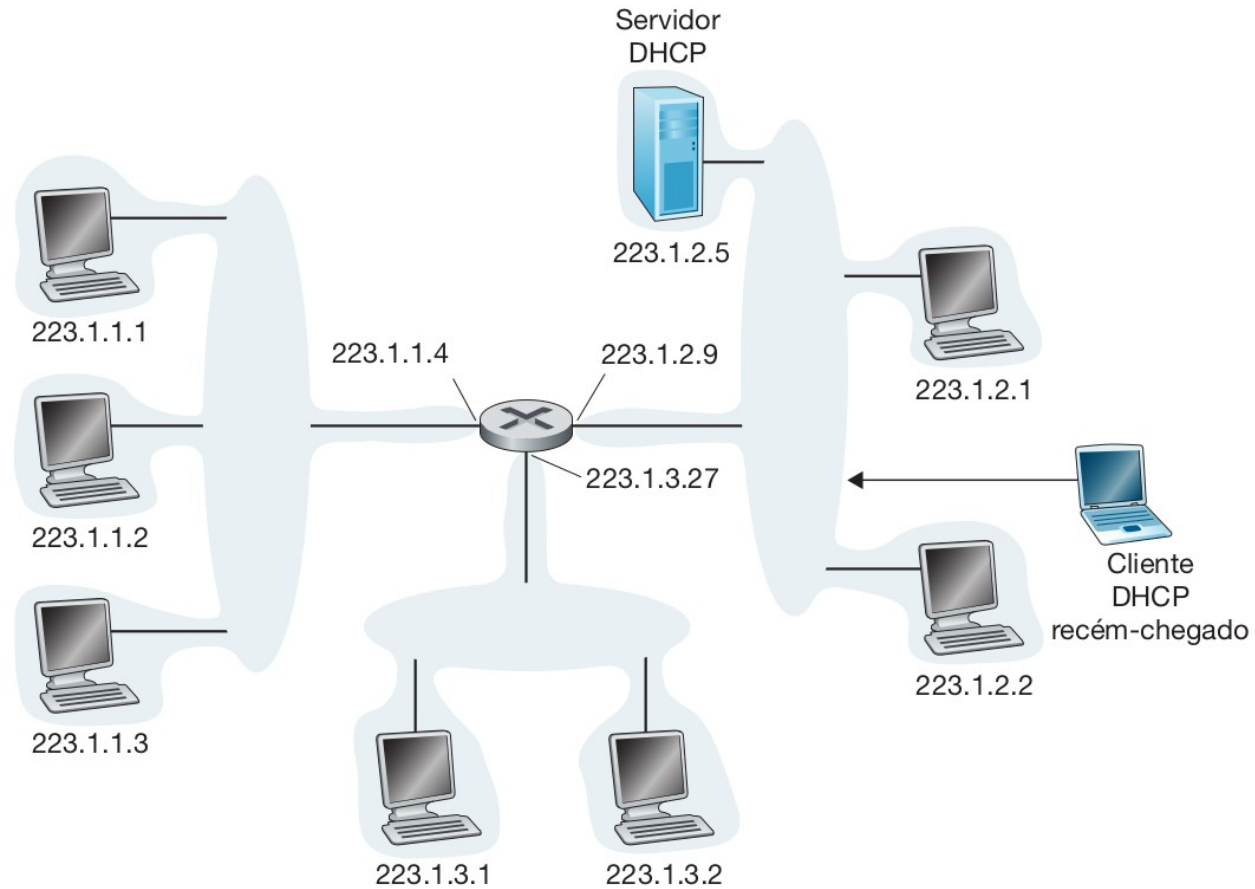
Endereço IP: como obter um?

- codificado pelo administrador em um arquivo no sistema operacional
- **DHCP: Dynamic Host Configuration Protocol:**
 - obtém endereço dinamicamente de um servidor
 - “plug-and-play”

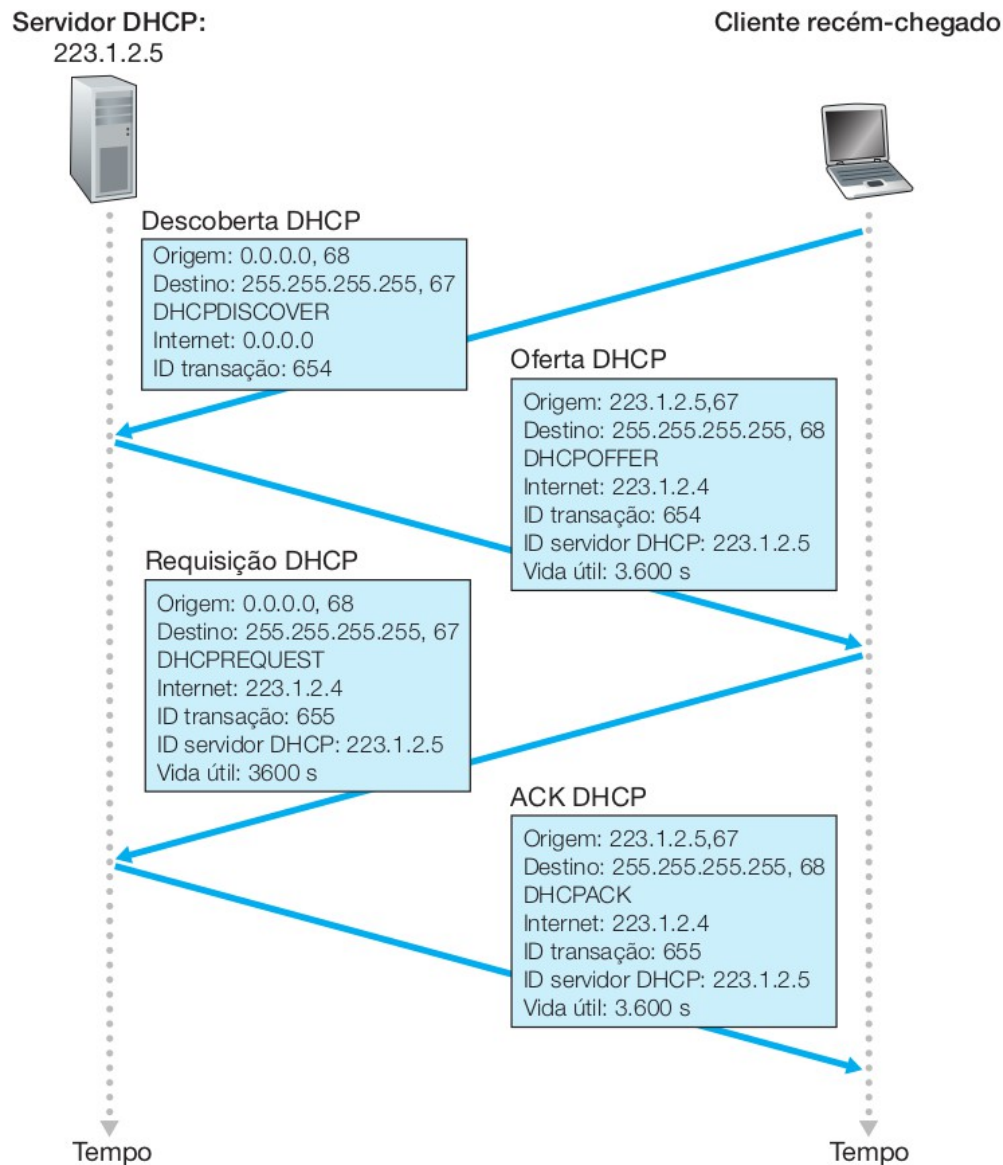
DHCP

- **Objetivo:** permitir que o hospedeiro obtenha dinamicamente seu endereço IP do servidor de rede quando se conectar à rede
 - pode renovar seu prazo no endereço utilizado
 - permite reutilização de endereços (só mantém endereço enquanto conectado e “ligado”)
 - aceita usuários móveis que queiram se juntar à rede (mais adiante)
- **Visão geral do DHCP:**
 - host broadcasts “DHCP discover” msg [optional]
 - servidor DHCP responde com msg “DHCP offer” [opcional]
 - hospedeiro requer endereço IP: msg “DHCP request”
 - servidor DHCP envia endereço: msg “DHCP ack”

DHCP: cliente-servidor



DHCP: interação cliente-servidor



DHCP: mais do que “servir” IPs

- DHCP pode retornar mais do que apenas o endereço IP alocado na sub-rede:
 - endereço do roteador do primeiro salto para o cliente
 - nome e endereço IP do servidor DNS
 - máscara de rede (indicando parte de rede versus hospedeiro do endereço)

Endereçamento IP

- Como um provedor IP consegue um bloco de endereços?
 - ICANN: Internet Corporation for Assigned Names and Numbers (www.icann.org.br)
 - aloca endereços
 - gerencia DNS
 - aloca nomes de domínio, resolve disputas

Redes de Computadores

Capítulo 4: A camada de rede

O protocolo da Internet (IP)
continuação....

Endereços IPs

- [1 byte].[1 byte].[1 byte].[1 byte]
 - 4 bytes ou 32 bits
- 1 byte pode representar de 0 a 255
- 1 byte pode representar de 00000000 a 11111111
- 0.0.0.0 a 255.255.255.255
- 00000000.00000000.00000000.00000000 a 11111111.11111111.11111111.11111111

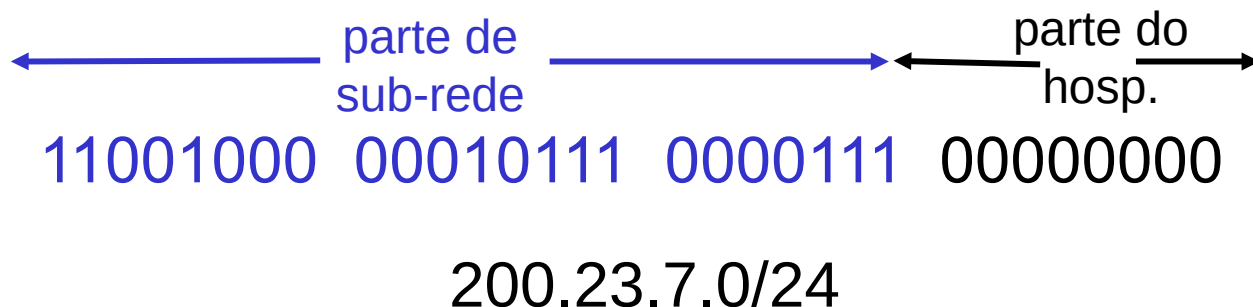
Endereços IPs

- Pode assumir qualquer variação dentro dos limites estabelecidos
- 15.0.230.64
- 200.240.180.27

Endereços IPs

- Endereço de broadcast
 - 255.255.255.255
- Localhost
 - 127.0.0.1

Endereços IPs

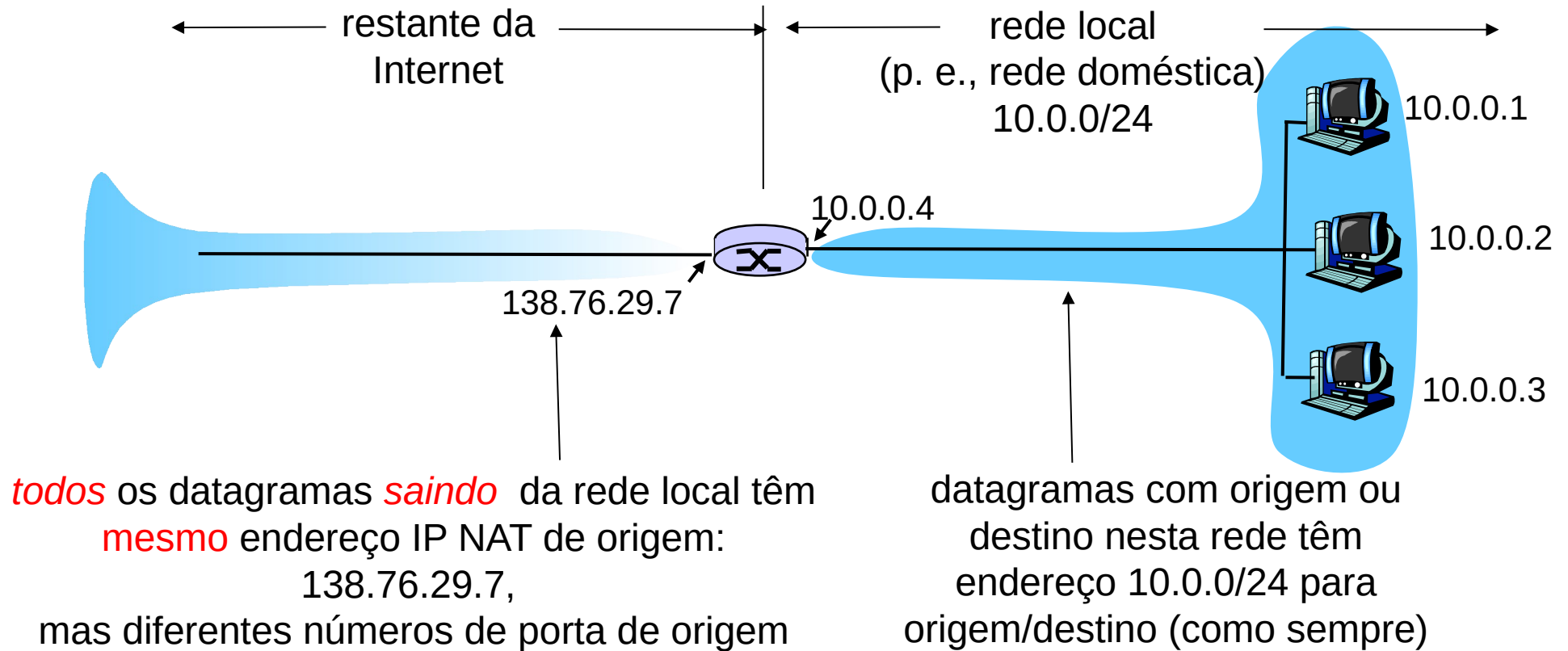


- 200.23.7.0 define o endereço da rede e não é utilizado por nenhuma interface
- 200.23.7.255 define o endereço de broadcast dentro da subrede e também não é utilizado por nenhuma interface

Endereços IPs

- Redes privadas
 - 10.0.0.0 até 10.255.255.255
 - 172.16.0.0 até 172.31.255.255
 - 192.168.0.0 até 192.168.255.255
 -
- Esses endereços **não funcionam na Internet**
 - Funcionam apenas em redes locais

Tradução de endereços na rede (NAT)



Tradução de endereços na rede (NAT)

- NAT: Network Address Translation
- **motivação**: rede local usa apenas um endereço IP no que se refere ao mundo exterior:
 - intervalo de endereços não necessário pelo ISP
 - para a Internet: apenas um endereço IP para todos os dispositivos
 - pode mudar os endereços dos dispositivos na rede local sem notificar o mundo exterior
 - pode mudar de ISP sem alterar os endereços dos dispositivos na rede local
 - dispositivos dentro da rede local não precisam ser explicitamente endereçáveis ou visíveis pelo mundo exterior (uma questão de segurança)

Tradução de endereços na rede (NAT)

Implementação: roteador NAT deve:

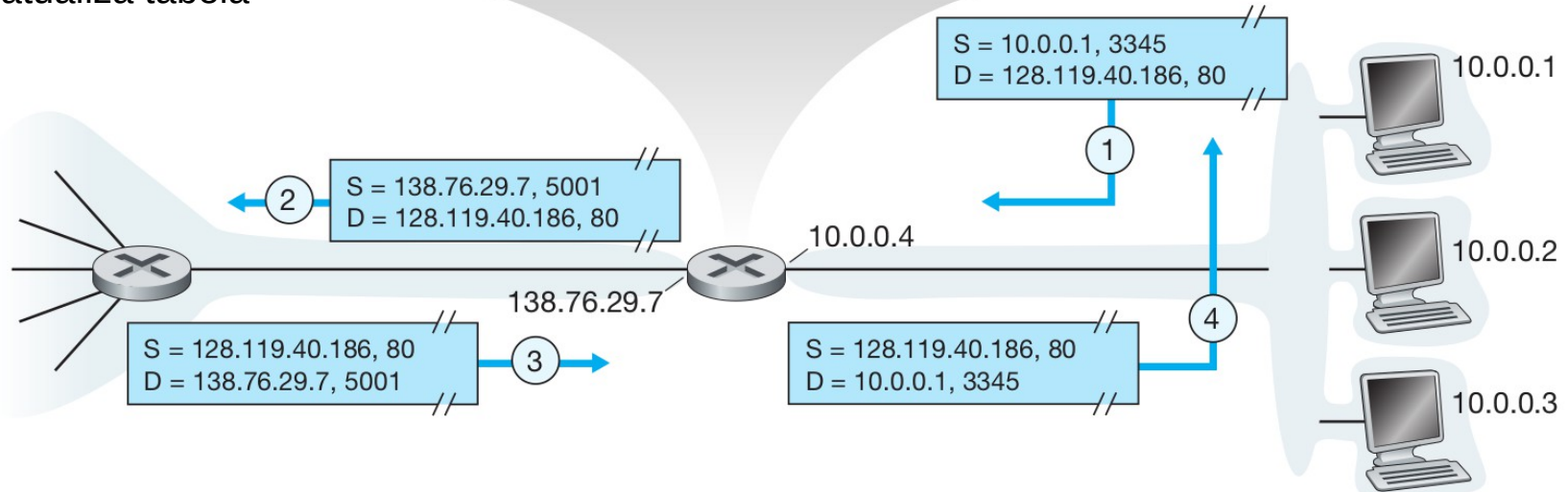
- **enviando datagramas: substituir** (endereço IP de origem, # porta) de cada datagrama saindo por (endereço IP da NAT, novo # porta)
 - . . . clientes/servidores remotos responderão usando (endereço IP da NAT, novo # porta) como endereço de destino
- **lembrar (na tabela de tradução NAT)** de cada par de tradução (endereço IP de origem, # porta) para (endereço IP da NAT, novo # porta)
- **recebendo datagramas: substituir** (endereço IP da NAT, novo # porta) nos campos de destino de cada datagrama chegando por (endereço IP origem, # porta) correspondente, armazenado na tabela NAT

Tradução de endereços na rede

2: roteador NAT muda endereço de origem do datagrama de 10.0.0.1, 3345 para 138.76.29.7, 5001, atualiza tabela

Tabela de tradução NAT	
Lado da WAN	Lado da LAN
138.76.29.7, 5001	10.0.0.1, 3345
...	...

1: hospedeiro 10.0.0.1, 3345 envia datagrama para 128.119.40.186, 80



3: Resposta chega endereço destino: 138.76.29.7, 5001

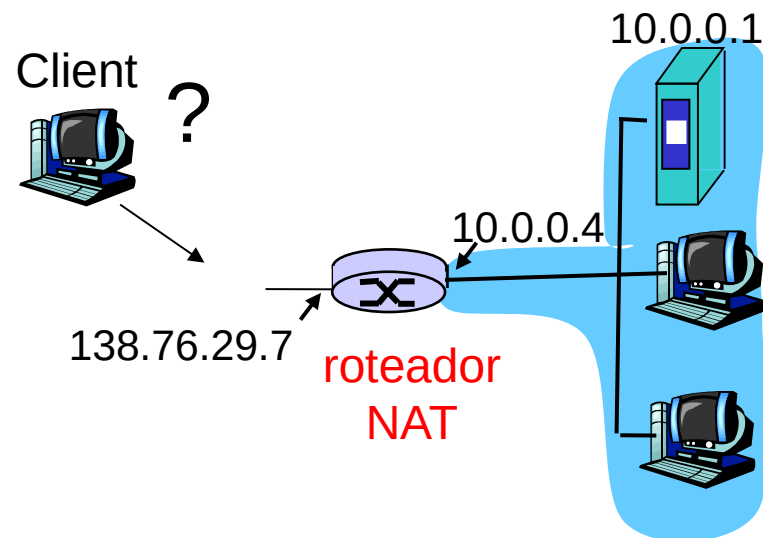
4: roteador NAT muda endereço de destino do datagrama de 138.76.29.7, 5001 para 10.0.0.1, 3345

Tradução de endereços na rede

- campo de número de porta de 16 bits:
 - 60.000 conexões simultâneas com um único endereço no lado da LAN!
- NAT é controverso:
 - roteadores só devem processar até a camada 3
 - viola argumento de fim a fim
 - a possibilidade de NAT deve ser levada em conta pelos projetistas da aplicação, p. e., aplicações P2P
- a falta de endereços deverá ser resolvida pelo IPv6

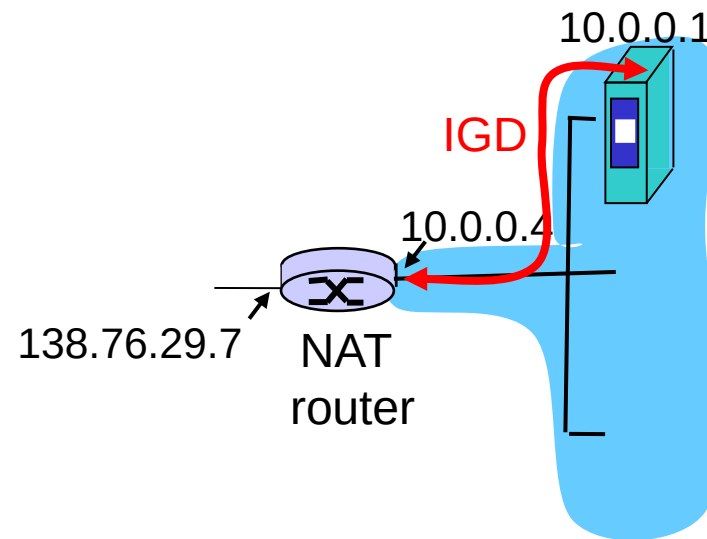
Problema de travessia do NAT

- cliente quer se conectar ao servidor com endereço 10.0.0.1
 - endereço do servidor 10.0.0.1 local à LAN (cliente não pode usá-lo como endereço destino)
 - apenas um endereço NAT visível externamente: 138.76.29.7
- **solução 1:** configure a NAT estaticamente para repassar as solicitações de conexão que chegam a determinada porta ao servidor
 - p. e., (123.76.29.7, porta 2500) sempre repassado para 10.0.0.1 porta 25000



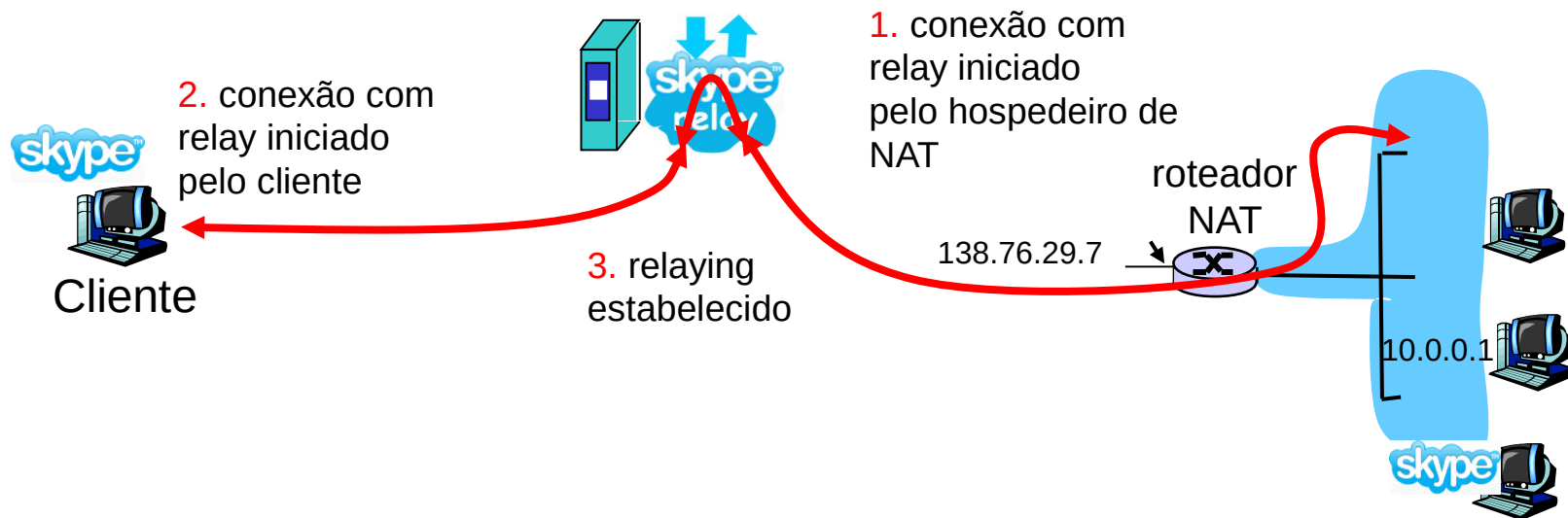
Problema de travessia do NAT

- **solução 2:** Universal Plug and Play (UPnP) Internet Gateway Device (IGD) Protocol. Permite que o hospedeiro com NAT:
 - descobrir endereço IP público (138.76.29.7)
 - incluir/remover mapeamentos de porta (com tempos de validade)
- automatizar configuração estática do mapa de porta NAT



Problema de travessia do NAT

- solução 3: repasse (usado no Skype)
 - cliente com NAT estabelece conexão com *relay*
 - cliente externo se conecta ao *relay*
 - *relay* liga pacotes entre duas conexões



ICMP: Internet Control Message Protocol

- usado por hospedeiros & roteadores para comunicar informações em nível de rede
 - relato de erro: hospedeiro, rede, porta, protocolo inalcançável
 - eco de solicitação/ resposta (usado por ping)
- camada de rede “acima” do IP:
 - msgs ICMP transportadas em datagramas IP
- **mensagem ICMP:** tipo, código mais primeiros 8 bytes do datagrama IP causando erro

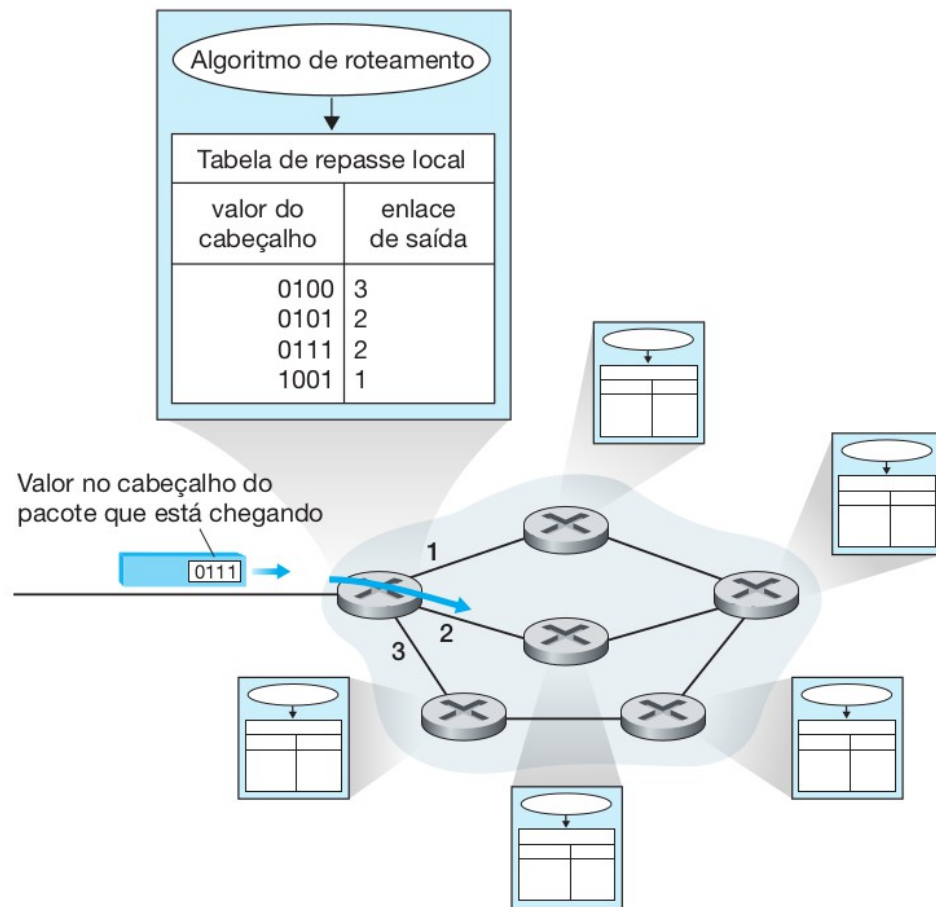
Tipo	Cód,	Descrição
0	0	resposta de eco (ping)
3	0	rede de destino inalcançável
3	1	hosp. de destino inalcançável
3	2	protocolo de destino inalcançável
3	3	porta de destino inalcançável
3	6	rede de destino desconhecida
3	7	hosp. de destino desconhecido
4	0	redução da fonte (controle de congestionamento – não usado)
8	0	solicitação de eco (ping)
9	0	anúncio de rota
10	0	descoberta do roteador
11	0	TTL expirado
12	0	cabeçalho IP inválido

Redes de Computadores

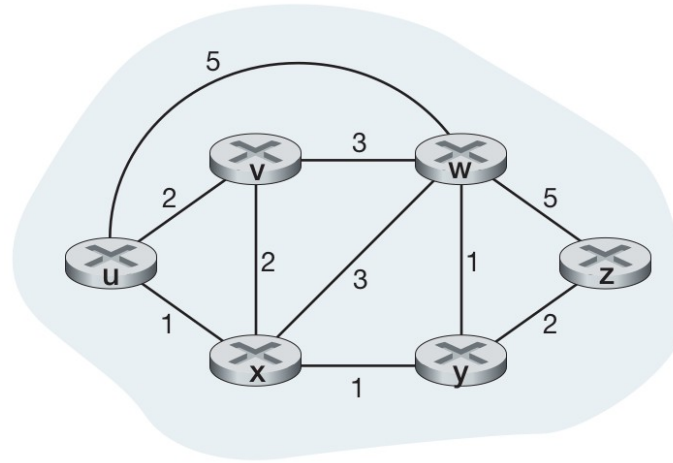
Capítulo 4: A camada de rede

Algoritmos de roteamento

Roteamento e repasse



Abstração da rede como um grafo



Grafo: $G = (N, E)$

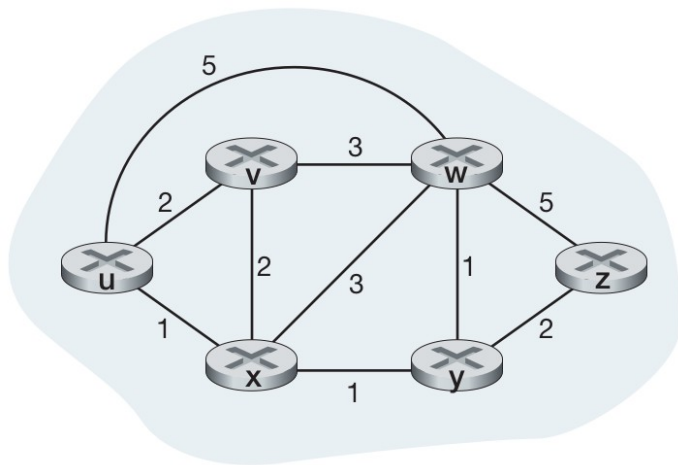
N = conjunto de roteadores = $\{u, v, w, x, y, z\}$

E = conjunto de enlaces = $\{(u,v), (u,x), (v,x), (v,w), (x,w), (x,y), (w,y), (w,z), (y,z)\}$

Comentário: Abstração de grafo é útil em outros contextos de rede

Exemplo: P2P, onde N é conj. de pares e E é conj. de conexões TCP

Grafos, custos e caminhos



- $c(x, x') = \text{custo do enlace } (x, x')$

- $c(w, z) = 5$

- custo poderia ser sempre 1, ou inversamente relacionado à largura ou inversamente relacionado ao congestionamento

Custo do caminho $(x_1, x_2, x_3, \dots, x_p) = c(x_1, x_2) + c(x_2, x_3) + \dots + c(x_{p-1}, x_p)$

Pergunta: Qual é o caminho de menor custo entre u e z?

algoritmo de roteamento: algoritmo que encontra o caminho de menor custo

Classificação de algoritmos de roteamento

Informação global ou descentralizada?

- global:
 - todos os roteadores têm topologia completa, informação de custo do enlace
 - algoritmos de “estado do enlace”
- descentralizada:
 - roteador conhece vizinhos conectados fisicamente, custos de enlace para vizinhos
 - processo de computação iterativo, troca de informações com vizinhos
 - algoritmos de “vetor de distância”

Classificação de algoritmos de roteamento

Estático ou dinâmico?

- **estático:**
 - rotas mudam lentamente com o tempo
- **dinâmico:**
 - rotas mudam mais rapidamente
 - atualização periódica em resposta a mudanças no custo do enlace

Estado de enlace

Algoritmo de Dijkstra

- nova topologia, custos de enlace conhecidos de todos os nós
 - realizado por “broadcast de estado do enlace”
 - todos os nós têm a mesma informação
- calcula caminhos de menor custo de um nó (“origem”) para todos os outros nós da rede
- iterativo: após k iterações, sabe caminho de menor custo para k destinos

Estado de enlace

notação:

- $c(x,y)$: custo do enlace do nó x até y ; $= \infty$ se não forem vizinhos diretos
- $D(v)$: valor atual do custo do caminho da origem ao destino v
- $p(v)$: nó predecessor ao longo do caminho da origem até v
- N' : conjunto de nós cujo caminho de menor custo é definitivamente conhecido

Estado de enlace

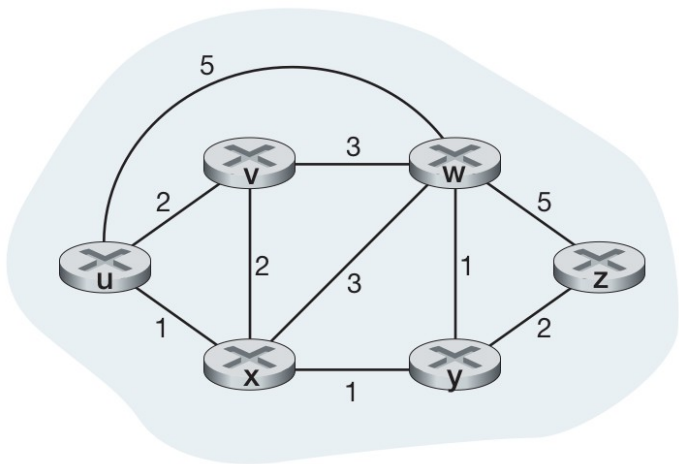
Algoritmo de estado de enlace para o nó de origem u

```
1  Inicialização
2       $N' = \{u\}$ 
3      para todos os nós  $v$ 
4          se  $v$  for um vizinho de  $u$ 
5              então  $D(v) = c(u,v)$ 
6              senão  $D(v) = \infty$ 
7
8  Loop
9      encontre  $w$  não em  $N'$  tal que  $D(w)$  é um mínimo
10     adicione  $w$  a  $N'$ 
11     atualize  $D(v)$  para cada vizinho  $v$  de  $w$  e não em  $N'$ :
12          $D(v) = \min( D(v), D(w) + c(w,v) )$ 
13     /* o novo custo para  $v$  é o velho custo para  $v$  ou
14        o custo do menor caminho conhecido para  $w$  mais o custo de  $w$  para  $v$  */
15 até  $N' = N$ 
```

Estado de enlace: exemplo

Algoritmo de estado de enlace para o nó de origem u

```
1  Inicialização
2     $N' = \{u\}$ 
3    para todos os nós  $v$ 
4      se  $v$  for um vizinho de  $u$ 
5        então  $D(v) = c(u,v)$ 
6      senão  $D(v) = \infty$ 
7
8  Loop
9    encontre  $w$  não em  $N'$  tal que  $D(w)$  é um mínimo
10   adicione  $w$  a  $N'$ 
11   atualize  $D(v)$  para cada vizinho  $v$  de  $w$  e não em  $N'$ :
12      $D(v) = \min( D(v), D(w) + c(w,v) )$ 
13   /* o novo custo para  $v$  é o velho custo para  $v$  ou
14     o custo do menor caminho conhecido para  $w$  mais o custo de  $w$  para  $v$  */
15  até  $N' = N$ 
```



$c(x,y)$: custo do enlace do nó x até y ; $= \infty$ se não forem vizinhos diretos

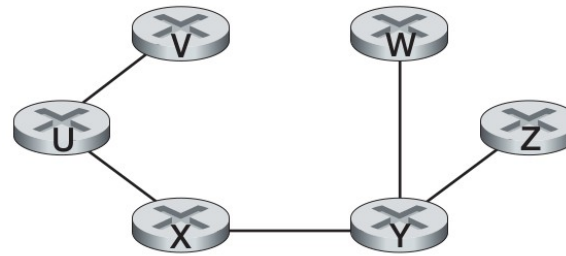
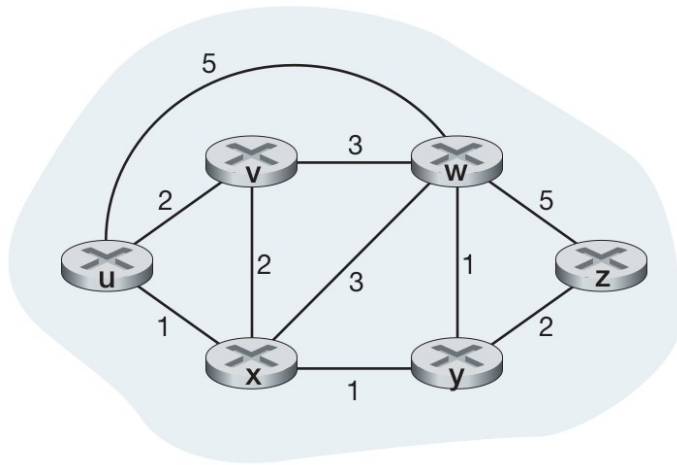
$D(v)$: valor atual do custo do caminho da origem ao destino v

$p(v)$: nó predecessor ao longo do caminho da origem até v

N' : conjunto de nós cujo caminho de menor custo é definitivamente conhecido

<i>Etap</i> a	N'	$D(v),p(v)$	$D(w),p(w)$	$D(x),p(x)$	$D(y),p(y)$	$D(z),p(z)$
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					

Estado de enlace: exemplo



Destino	Enlace
v	(u, v)
w	(u, x)
x	(u, x)
y	(u, x)
z	(u, x)

Caminho de menor custo

Redes de Computadores

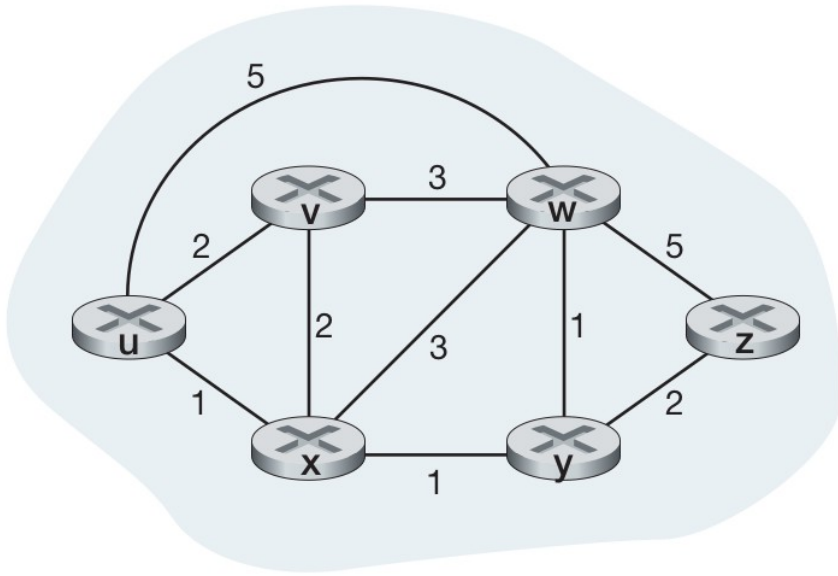
Capítulo 4: A camada de rede

Algoritmos de roteamento
continuação...

Algoritmos de vetor de distância

- Equação de Bellman-Ford (programação dinâmica)
- $d_x(y) :=$ custo do caminho de menor custo de x para y
- $d_x(y) = \min_v \{ c(x,v) + d_v(y) \}$
 - onde \min assume todos os vizinhos v de x

Algoritmos de vetor de distância



$$d_v(z) = 5, d_x(z) = 3, d_w(z) = 3$$

equação B-F diz:

$$\begin{aligned} d_u(z) &= \min \{ c(u,v) + d_v(z), \\ &\quad c(u,x) + d_x(z), \\ &\quad c(u,w) + d_w(z) \} \\ &= \min \{ 2 + 5, \\ &\quad 1 + 3, \\ &\quad 5 + 3 \} = 4 \end{aligned}$$

nó que alcança mínimo é o próximo salto
no caminho mais curto → tabela de
repassse

Algoritmos de vetor de distância

- $D_x(y)$ = estimativa do menor custo de x para y
- nó x sabe custo de cada vizinho v : $c(x,v)$
- nó x mantém vetor de distância $D_x = [D_x(y): y \in N]$
- nó x também mantém vetor de distância de seus vizinhos
 - para cada vizinho v , x mantém $D_v = [D_v(y): y \in N]$

Algoritmo de vetor de distâncias

ideia básica:

- de tempos em tempos, cada nó envia sua própria estimativa de vetor de distância aos vizinhos
- assíncrono
- quando um nó x recebe nova estimativa DV do vizinho, ele atualiza seu próprio DV usando a equação de B-F:

$$D_x(y) \leftarrow \min_v \{c(x,v) + D_v(y)\} \quad \text{para cada nó } y \in N$$

- sob condições modestas, naturais, a estimativa $D_x(y)$ converge para o menor custo real $d_x(y)$

Algoritmo de vetor de distâncias

iterativo, assíncrono: cada iteração local causada por:

- mudança de custo do enlace local
- mensagem de atualização do DV do vizinho

distribuído:

- cada nó notifica os vizinhos apenas quando seu DV muda
- vizinhos, então, notificam seus vizinhos, se necessário

Algoritmo de vetor de distâncias (DV)

Tabela do nó x

De	Custo até		
	x	y	z
x	0	2	7
y	∞	∞	∞
z	∞	∞	∞

Tabela do nó y

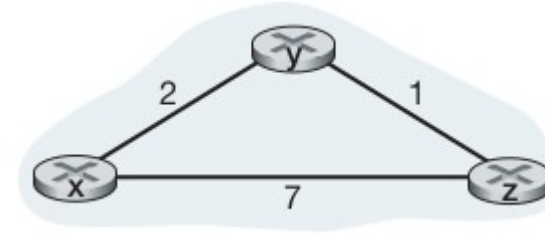
De	Custo até		
	x	y	z
x	∞	∞	∞
y	2	0	1
z	∞	∞	∞

Tabela do nó z

De	Custo até		
	x	y	z
x	∞	∞	∞
y	∞	∞	∞
z	7	1	0

$$D_x(y) \leftarrow \min_v \{c(x,v) + D_v(y)\}$$

para cada nó $y \in N$



Tempo

Algoritmo de vetor de distâncias (DV)

Tabela do nó x

		Custo até		
		x	y	z
De	x	0	2	7
	y	∞	∞	∞
	z	∞	∞	∞

Tabela do nó y

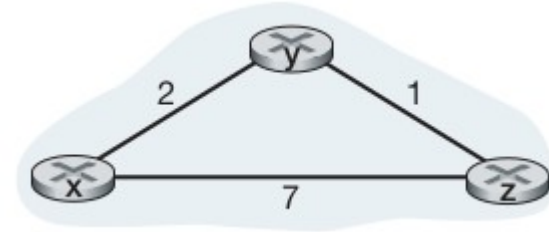
		Custo até		
		x	y	z
De	x	∞	∞	∞
	y	2	0	1
	z	∞	∞	∞

Tabela do nó z

		Custo até		
		x	y	z
De	x	∞	∞	∞
	y	∞	∞	∞
	z	7	1	0

$$D_x(y) = \min\{c(x,y) + D_y(y), c(x,z) + D_z(y)\} \\ = \min\{2+0, 7+1\} = 2$$

$$D_x(z) = \min\{c(x,y) + D_y(z), c(x,z) + D_z(z)\} \\ = \min\{2+1, 7+0\} = 3$$



Tempo

Algoritmo de vetor de distâncias (DV)

Tabela do nó x

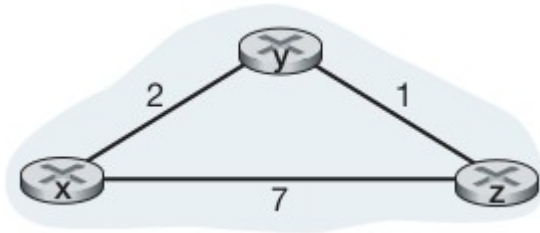
		Custo até		
		x	y	z
De	x	0	2	7
	y	∞	∞	∞
	z	∞	∞	∞

Tabela do nó y

		Custo até		
		x	y	z
De	x	∞	∞	∞
	y	2	0	1
	z	∞	∞	∞

Tabela do nó z

		Custo até		
		x	y	z
De	x	∞	∞	∞
	y	∞	∞	∞
	z	7	1	0



Tempo

Estado do enlace ou vetor de distâncias?

- Complexidade?
- Velocidade de convergência?
- Robustez?

Redes de Computadores

Capítulo 4: A camada de rede

Roteamento Hierárquico

Roteamento Hierárquico

Nosso estudo de roteamento até aqui – o ideal:

- todos os roteadores idênticos
- rede “achatada”
- ... *não* acontece na prática

escala: com 200 milhões de destinos:

- não pode armazenar todos os destinos nas tabelas de roteamento!
- troca de tabela de roteamento atolaria os enlaces!

autonomia administrativa

- Internet = rede de redes
- cada administrador de rede pode querer controlar o roteamento em sua própria rede

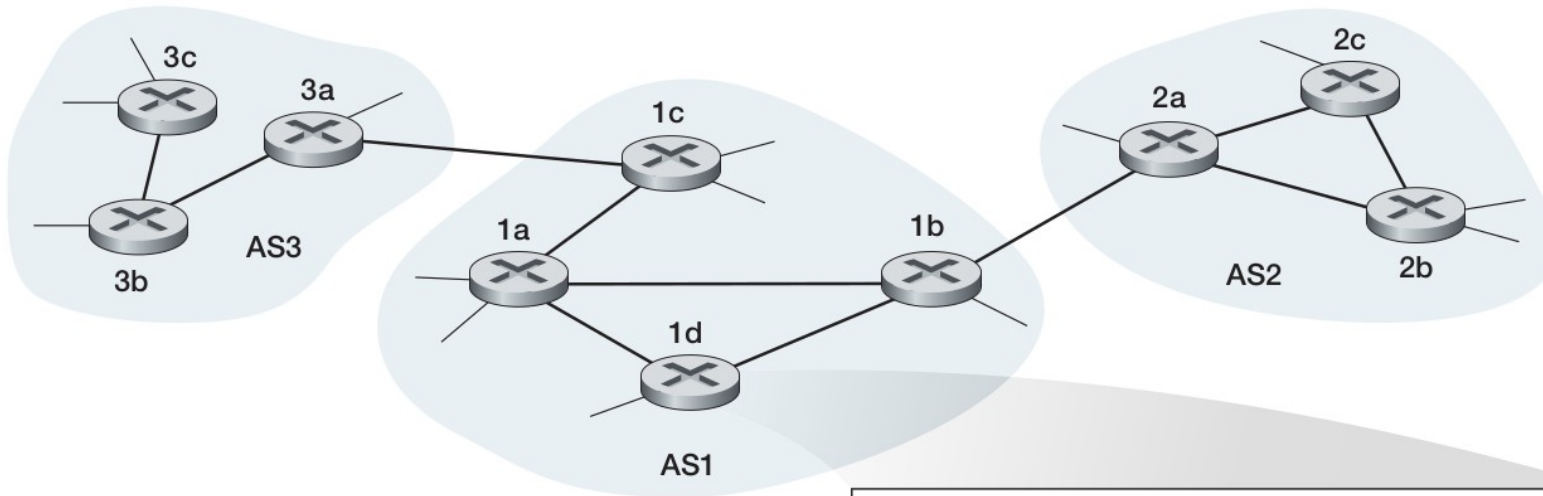
Roteamento Hierárquico

- Roteadores agregados em regiões, “sistemas autônomos” (AS)
- Roteadores no mesmo AS rodam o mesmo protocolo de roteamento
 - Protocolo de roteamento “intra-AS”
 - Roteadores em ASes diferentes podem executar protocolo de roteamento intra-AS diferente

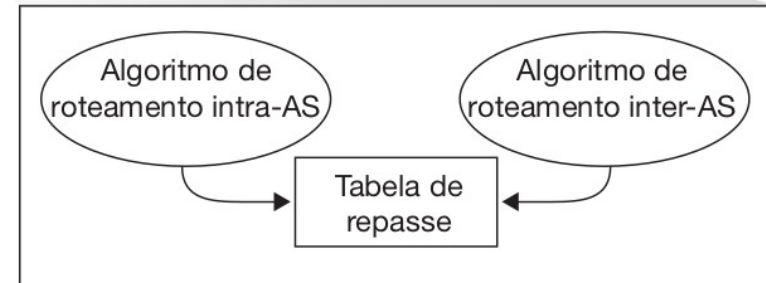
roteador de borda

- Enlace direto com roteador em outro AS

ASes interconectadas



- Tabela de repasse configurada por algoritmo de roteamento intra e inter-AS
 - intra-AS define entradas para destinos internos
 - inter-AS & intra-AS definem entradas para destinos externos



Redes de Computadores

Capítulo 4: A camada de rede

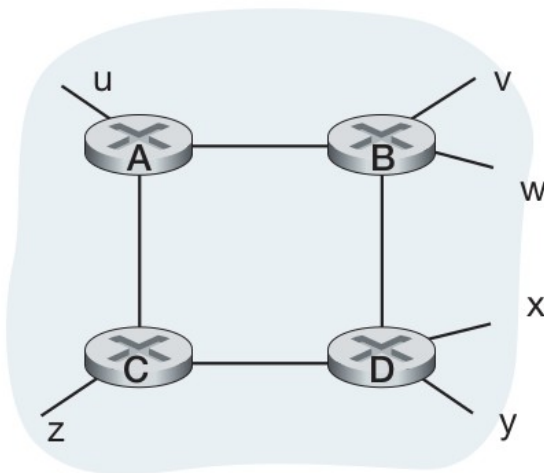
Roteamento na Internet

Roteamento intra-AS

- Também conhecido como **Interior Gateway Protocols (IGP)**
- Protocolos de roteamento intra-AS mais comuns:
 - RIP: Routing Information Protocol
 - OSPF: Open Shortest Path First
 - IGRP: Interior Gateway Routing Protocol (proprietário da Cisco)

RIP (Routing Information Protocol)

- Algoritmo de vetor de distância
- Incluído na distribuição BSD-UNIX em 1982
- Métrica de distância: # de saltos (máx. = 15 saltos)



Destino	Saltos
u	1
v	2
w	2
x	3
y	3
z	2

RIP (Routing Information Protocol)

- Vetores de distância: trocados entre vizinhos a cada 30s por meio de mensagem de resposta (também conhecida como anúncio)
- Cada anúncio: lista de até 25 sub-redes de destino dentro do AS

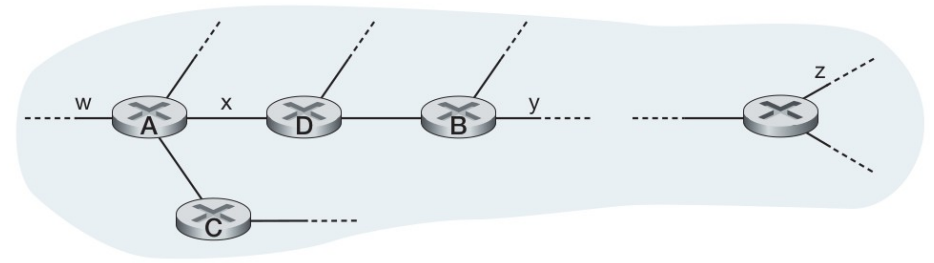


Tabela de roteamento do roteador D

Sub-rede de destino	Roteador seguinte	Número de saltos até o destino
w	A	2
y	B	2
z	B	7
x	—	1
....

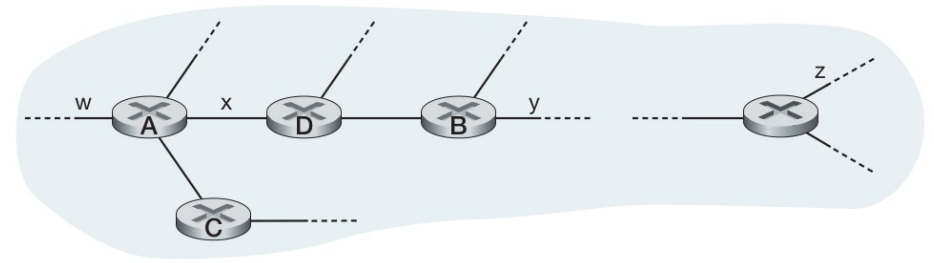


Tabela de roteamento do roteador D

Sub-rede de destino	Roteador seguinte	Número de saltos até o destino
w	A	2
y	B	2
z	B	7
x	—	1
....

Anúncio recebido do roteador A

Sub-rede de destino	Roteador seguinte	Número de saltos até o destino
z	C	4
w	—	1
x	—	1
....

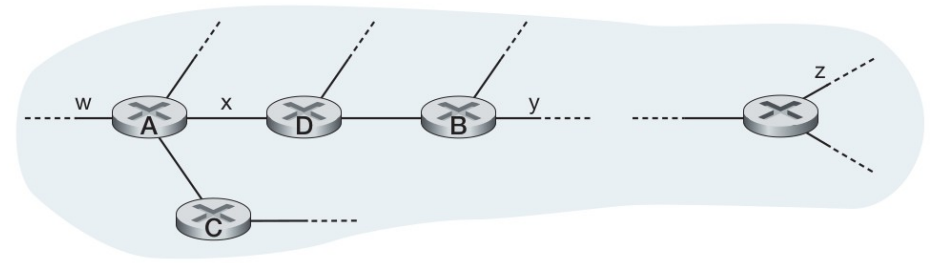


Tabela de roteamento do roteador D

Sub-rede de destino	Roteador seguinte	Número de saltos até o destino
w	A	2
y	B	2
z	B	7
x	—	1
....

Anúncio recebido do roteador A

Sub-rede de destino	Roteador seguinte	Número de saltos até o destino
z	C	4
w	—	1
x	—	1
....

Sub-rede de destino	Roteador seguinte	Número de saltos até o destino
w	A	2
y	B	2
z	A	5
....

Nova tabela de roteamento do roteador D

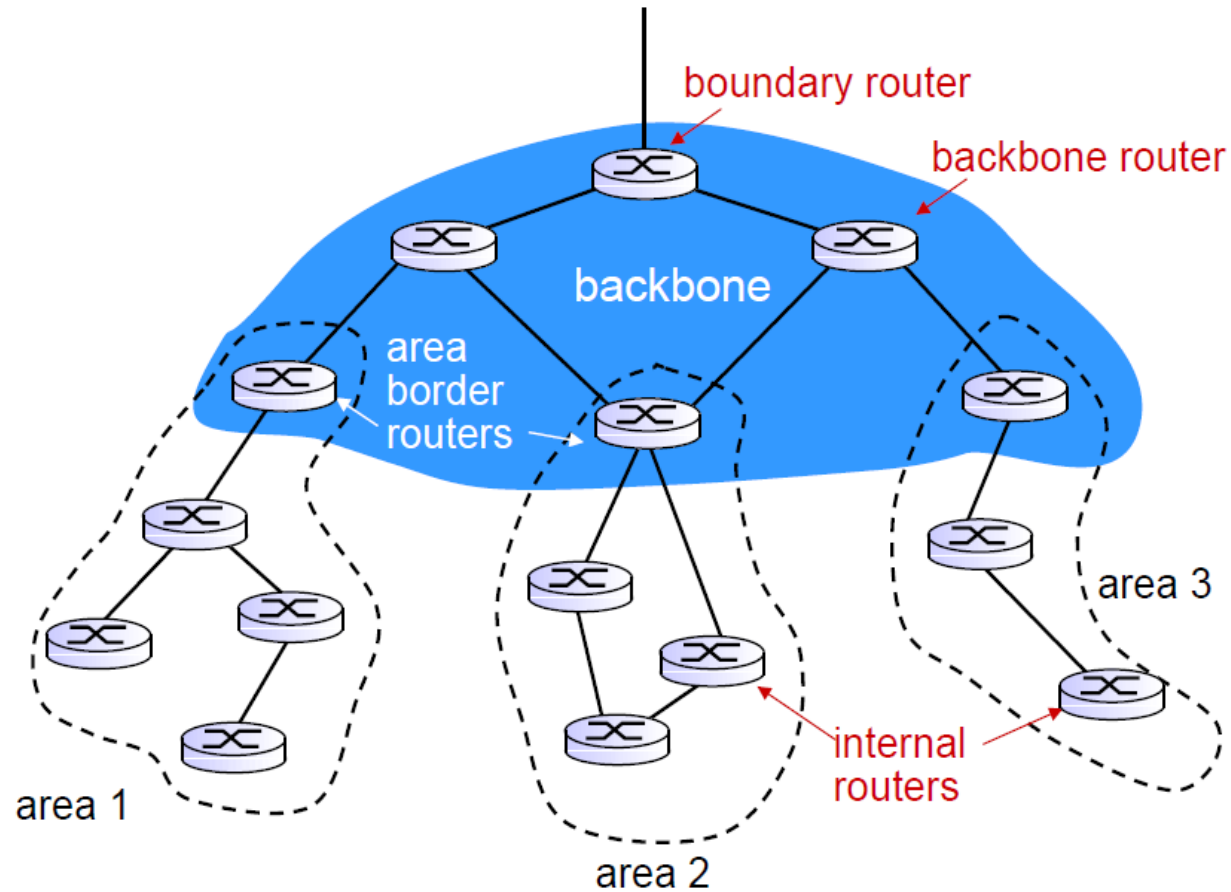
OSPF (Open Shortest Path First)

- “open”: publicamente disponível
- Usa algoritmo Link State (estado de enlace)
 - disseminação de pacote LS
 - mapa de topologia em cada nó
 - cálculo de rota usando algoritmo de Dijkstra
- Anúncio OSPF transporta uma entrada por roteador vizinho
- anúncios disseminados ao AS inteiro (com inundação)
 - transportados nas mensagens OSPF diretamente por IP (em vez de TCP ou UDP)

Recursos “avançados” do OSPF (não no RIP)

- **segurança**: todas as mensagens OSPF autenticadas (para impedir intrusão maliciosa)
- **múltiplos caminhos** de mesmo custo permitidos (apenas um caminho no RIP)
- para cada enlace, múltiplas métricas de custo para diferentes (p. e., custo de enlace de satélite definido “baixo” para melhor esforço; alto para tempo real)
- OSPF **hierárquico** em grandes domínios

OSPF hierárquico



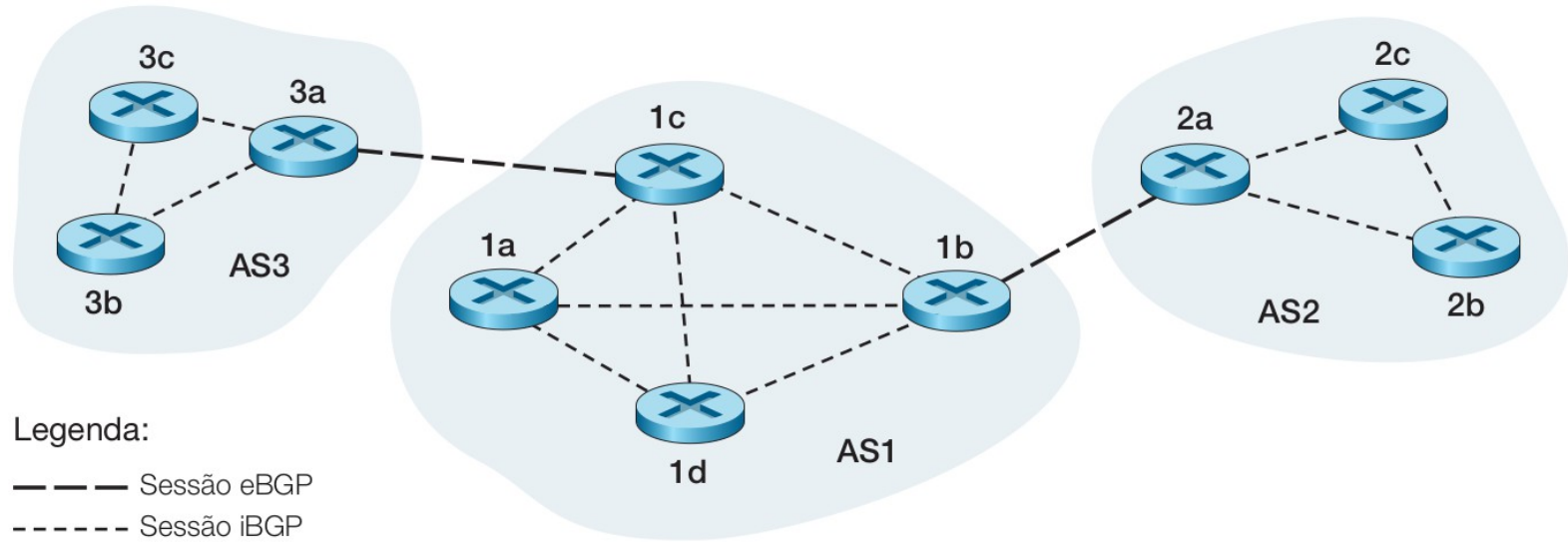
Roteamento inter-AS da Internet: BGP

- **BGP (Border Gateway Protocol):** o padrão de fato
- BGP oferece a cada AS um meio de:
 - Obter informação de acessibilidade da sub-rede a partir de ASs vizinhos.
 - Propagar informação de acessibilidade a todos os roteadores internos ao AS.
 - Determinar rotas “boas” para sub-redes com base na informação e política de acessibilidade.
- Permite que a sub-rede anuncie sua existência ao resto da Internet: “Estou aqui”

Fundamentos do BGP

- Pares de roteadores (pares BGP) trocam informações de roteamento nas conexões TCP semi-permanentes: sessões BGP
 - sessões BGP não precisam corresponder a enlaces físicos
- Quando AS2 anuncia um prefixo para AS1:
 - AS2 promete que repassará datagramas para esse prefixo
 - AS2 pode agregar prefixos em seu anúncio

Fundamentos do BGP



Mensagens BGP

- Mensagens BGP trocadas usando TCP.
- Mensagens BGP:
 - **OPEN**: abre conexão TCP com par e autentica remetente
 - **UPDATE**: anuncia novo caminho (ou retira antigo)
 - **KEEPALIVE**: mantém conexão viva na ausência de UPDATES; também envia ACK para solicitação OPEN
 - **NOTIFICATION**: informa erros na msg anterior; também usada para fechar conexão

Por que roteamento intra e inter-AS diferente?

política:

- inter-AS: admin deseja controle sobre como seu tráfego é roteado, quem roteia através de sua rede
- intra-AS: único admin, de modo que nenhuma decisão política é necessária

escala:

- roteamento hierárquico salva tamanho de tabela, tráfego de atualização reduzido

desempenho:

- intra-AS: pode focalizar no desempenho
- inter-AS: política pode dominar sobre desempenho