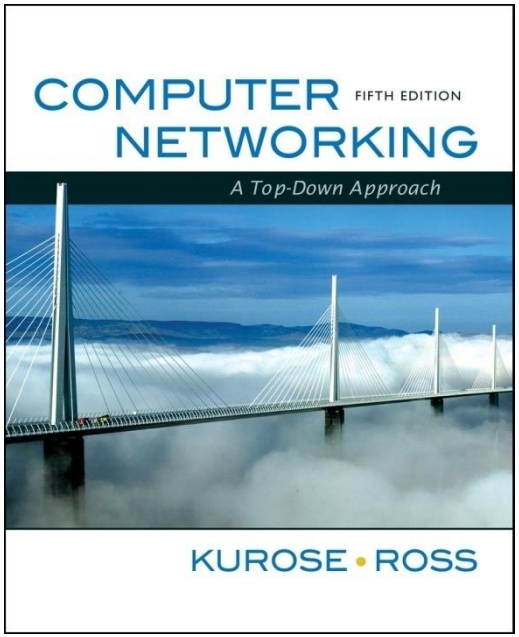
Camada de Redes

Prof Nelson Fonseca

http//:www.ic.unicamp.br/~nfonseca/redes/inf502

Chapter 4
Network Layer



A note on the use of these ppt slides:

We're making these slides freely available to all (faculty, students, readers). They're in PowerPoint form so you can add, modify, and delete slides (including this one) and slide content to suit your needs. They obviously represent a *lot* of work on our part. In return for use, we only ask the following:

- ❑ If you use these slides (e.g., in a class) in substantially unaltered form, that you mention their source (after all, we'd like people to use our book!)
- ❑ If you post any slides in substantially unaltered form on a www site, that you note that they are adapted from (or perhaps identical to) our slides, and note our copyright of this material.

Thanks and enjoy! JFK/KWR

Computer Networking: A Top Down Approach 5th edition.
Jim Kurose, Keith Ross
Addison-Wesley, April 2009.

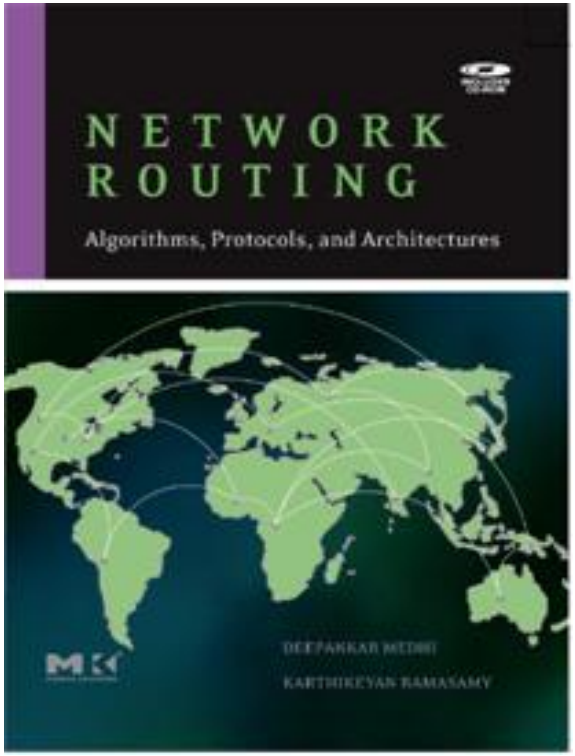
Outra fonte bibliográfica

- Alguns slides nesse arquivo foram gentilmente cedidos pelos autores do livro:
- Computer Networks: An Open Source Approach, Ying-Dar Lin, Ren-Hung Hwang, Fred Baker, published by McGraw Hill, Feb 2011

UNICAMP

Outra fonte bibliográfica

- Alguns slides nesse arquivo foram gentilmente cedidos pelos autores do livro:
- D. Medhi and K. Ramasamy, Network Routing: Algorithms, Protocols and Architectures, Morgan Kaufmann Publishers



Camada de Redes

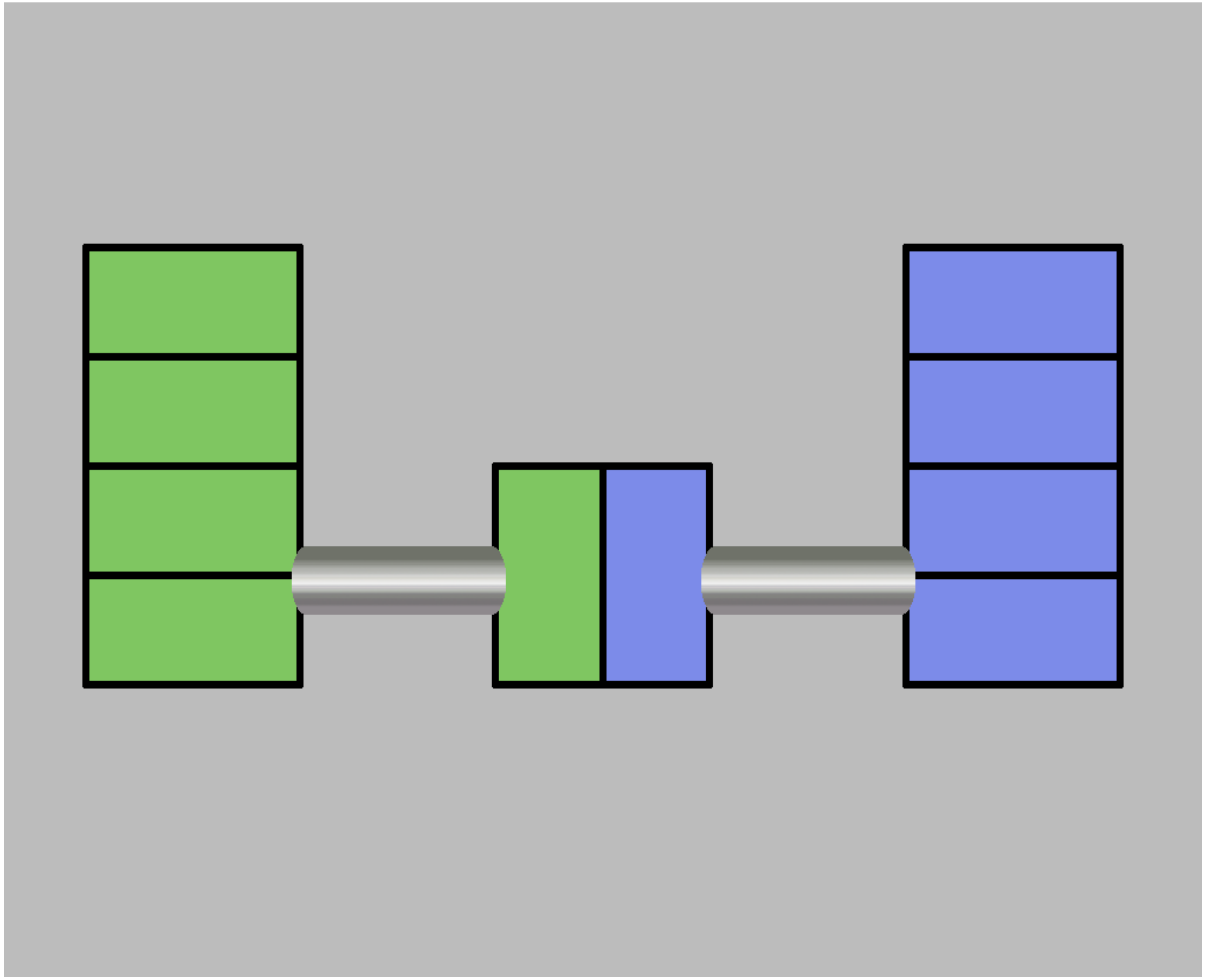
Objetivos do Capítulo:

- Entender os principais princípios do serviço da camada de redes :
  - ✓ Modelos de serviço da camada de redes
  - ✓ Encaminhamento e roteamento
  - ✓ Como um roteador funciona
  - ✓ Roteamento (seleção de caminhos)
  - ✓ Lidando com escala
  - ✓ IPv
- Implementações na Internet

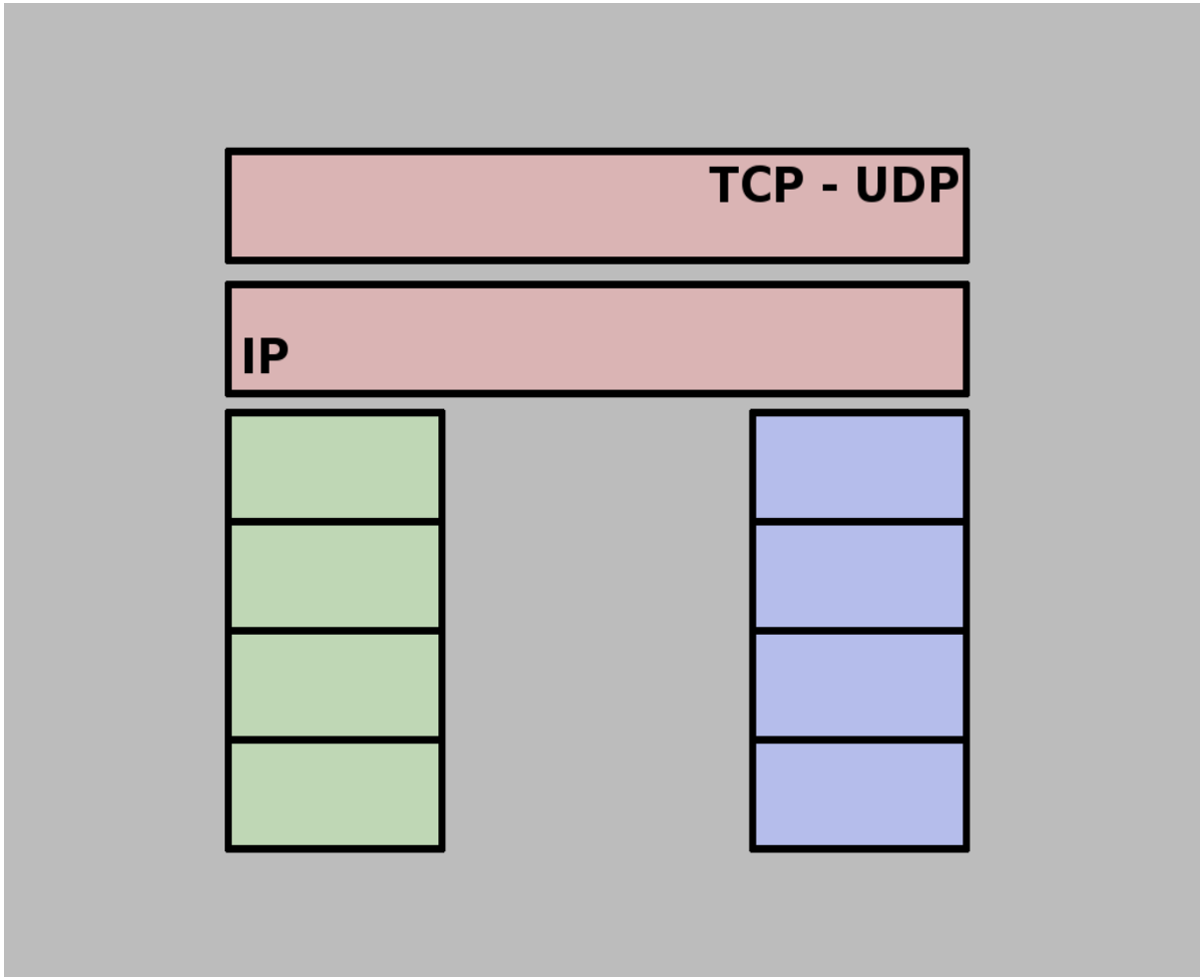
Roteiro

- 4. 1 Introdução
- 4.2 Circuitos virtuais x datagrama
- 4.3 Como é um roteador
- 4.4 Protocolo IP
  - ✓ Formato datagarma
  - ✓ endereçamento IPv4
  - ✓ ICMP
  - ✓ IPv6
- 4.5 Algoritmos de roteamento
  - ✓ Estado de enlace
  - ✓ Vetor distância
  - ✓ Roteamento hierarquico
- 4.6 Roteamento na Internet
  - ✓ RIP
  - ✓ OSPF
  - ✓ BGP
- 4.7 Roteamento Broadcast e multicast

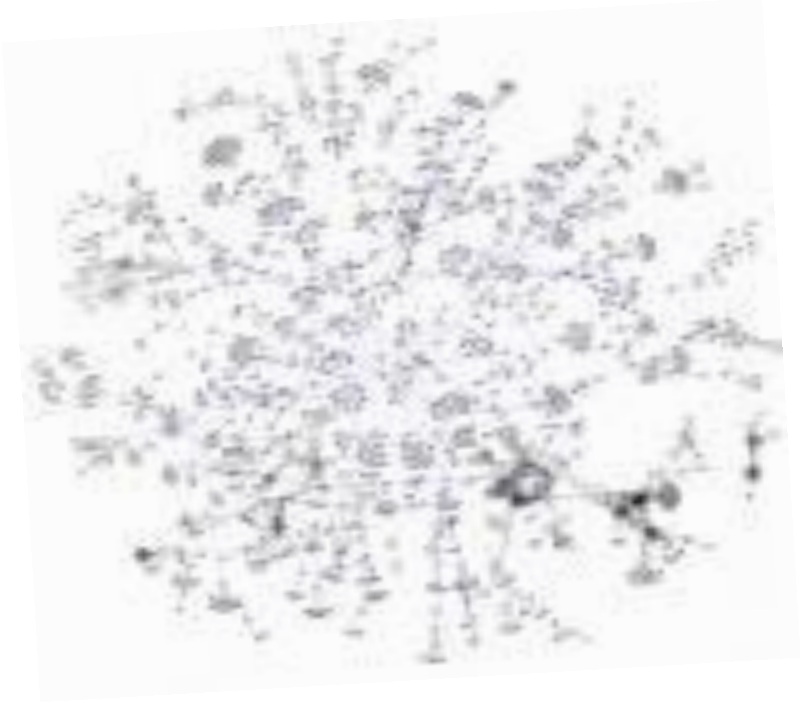
Interconexão de Redes



A Internet



A Internet



Estatísticas População da Internet

World Internet Users and 2017 Population Stats

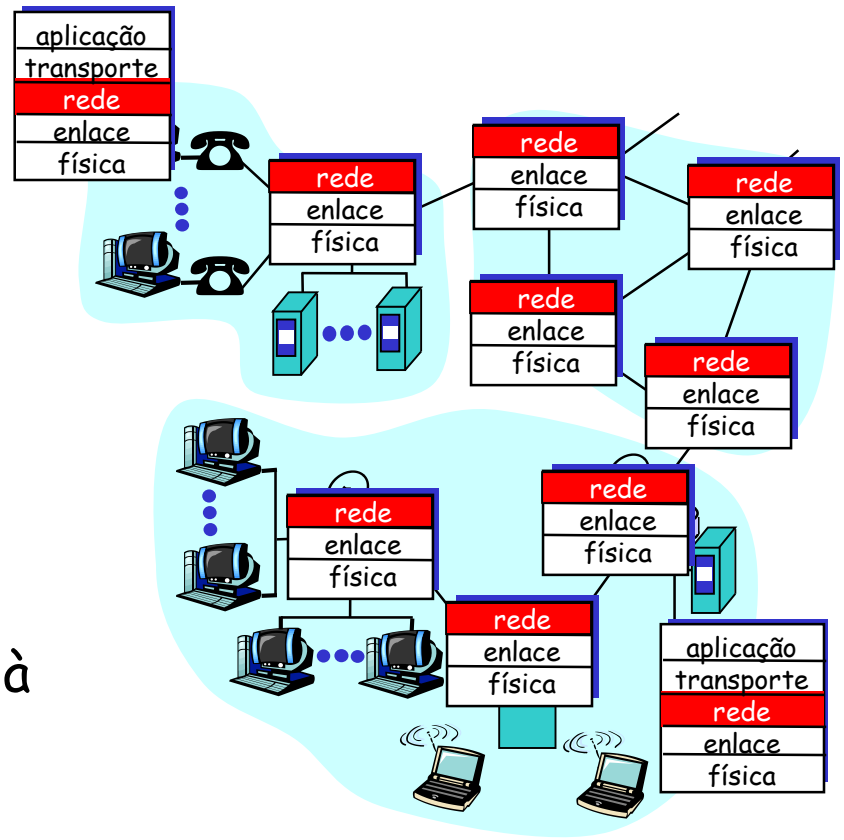
WORLD INTERNET USAGE AND POPULATION STATISTICS MARCH 25, 2017 - Update						
World Regions	Population (2017 Est.)	Population % of World	Internet Users 31 Mar 2017	Penetration Rate (% Pop.)	Growth 2000-2017	Users % Table
Africa	1,246,504,865	16.6 %	**345,676,501**	27.7 %	7,557.2%	9.3 %
Asia	4,148,177,672	55.2 %	**1,873,856,654**	45.2 %	1,539.4%	50.2 %
Europe	822,710,362	10.9 %	**636,971,824**	77.4 %	506.1%	17.1 %
Latin America / Caribbean	647,604,645	8.6 %	**385,919,382**	59.6 %	2,035.8%	10.3 %
Middle East	250,327,574	3.3 %	**141,931,765**	56.7 %	4,220.9%	3.8 %
North America	363,224,006	4.8 %	**320,068,243**	88.1 %	196.1%	8.6 %
Oceania / Australia	40,479,846	0.5 %	**27,549,054**	68.1 %	261.5%	0.7 %
WORLD TOTAL	**7,519,028,970**	**100.0 %**	**3,731,973,423**	**49.6 %**	**933.8%**	**100.0 %**

Funções da camada de rede

- transporta pacote da estação remetente à receptora
- protocolos da camada de rede em *cada* estação, roteador

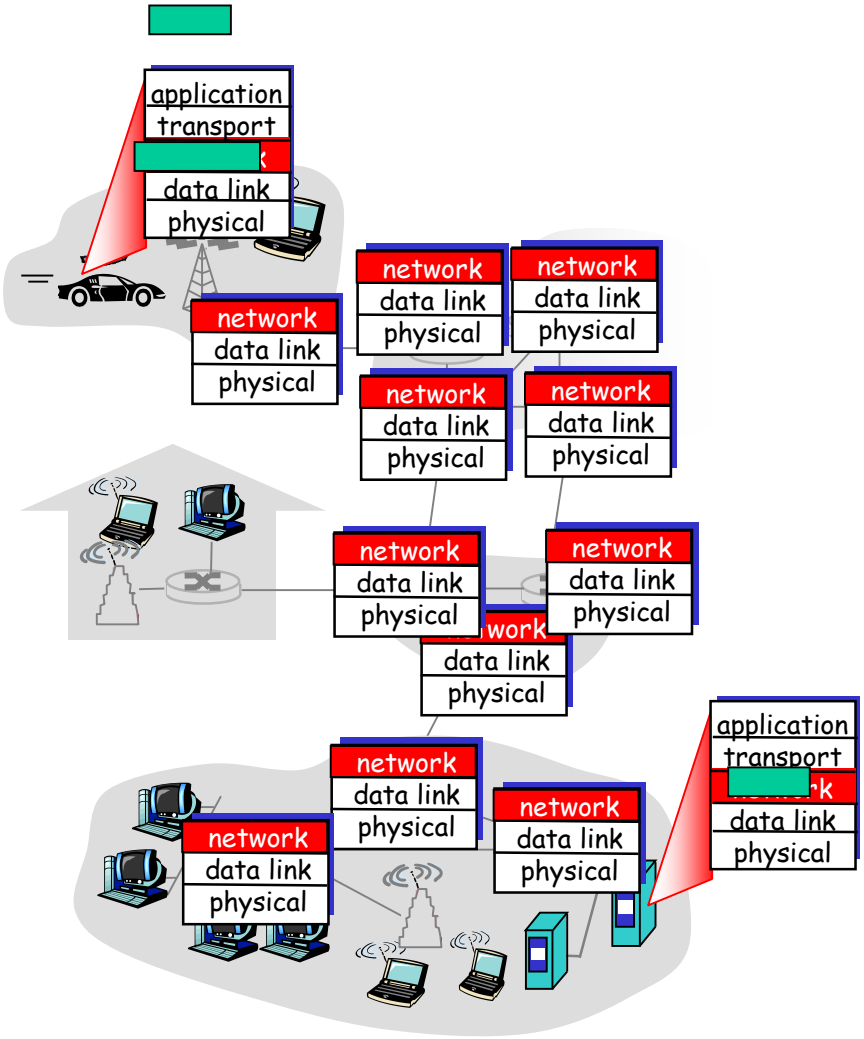
três funções importantes:

- *determinação do caminho:* rota seguida por pacotes da origem ao destino. *Algoritmos de roteamento*
- *comutação:* mover pacotes dentro do roteador da entrada à saída apropriada
- *estabelecimento da chamada:* algumas arquiteturas de rede requerem determinar o caminho antes de enviar os dados

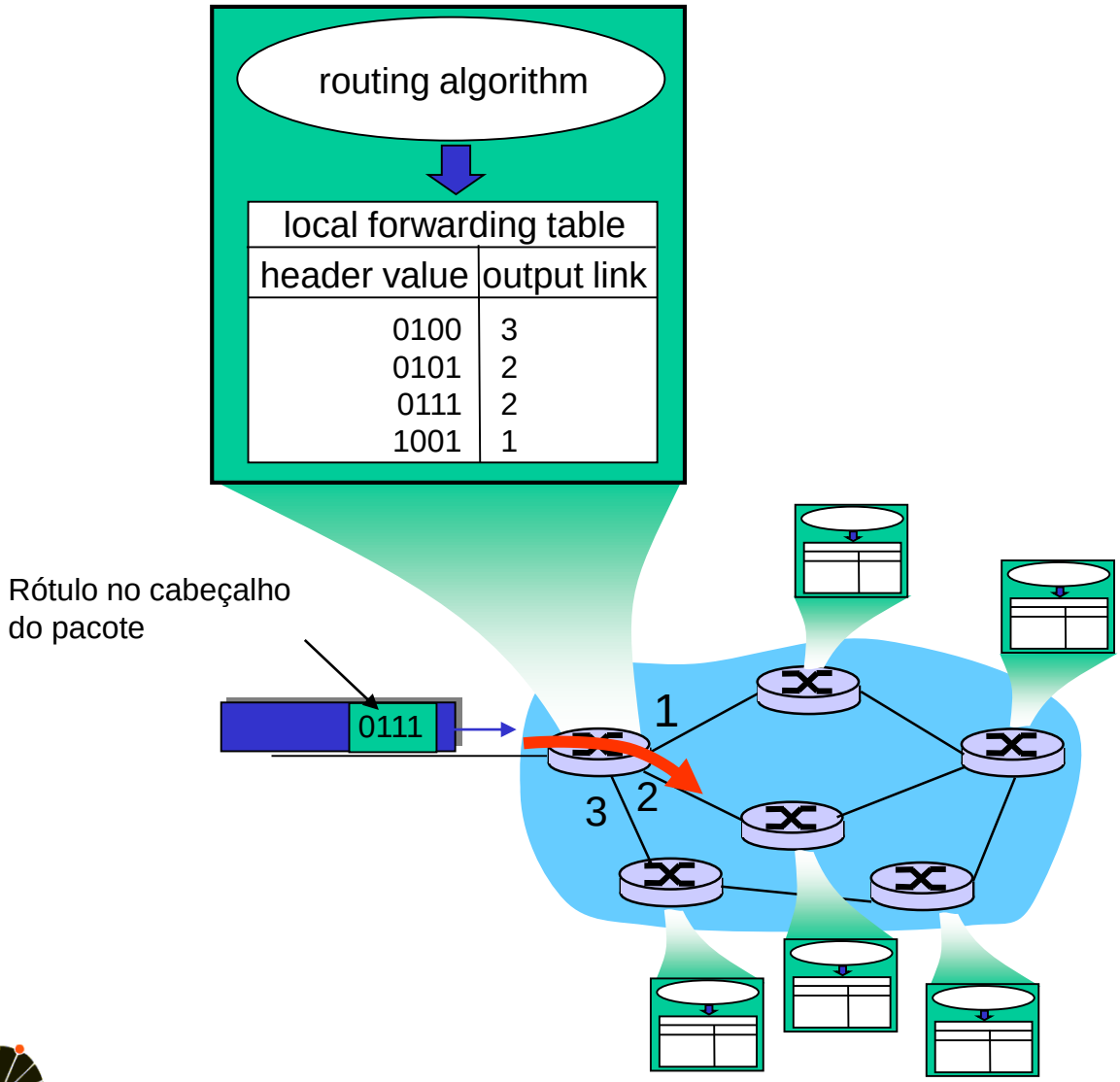


Network layer

- Transporta segmentos do transmissor ao repector
- Encapsula segmentos em datagramas
- No receptor, desencapsula e entrega a camada de transporte
- roteadores examina cabeçalho de todos os datagrams que passam pelo roteador



Roteamento e encaminhamento



Modelo de serviço de rede

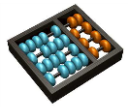
Q: Qual é o *modelo de serviço* para o "canal" que transporta pacotes do remetente ao receptor?

abstração do serviço

- largura de banda garantida?
- preservação de temporização entre pacotes (sem *jitter*)?
- entrega sem perdas?
- entrega ordenada?
- realimentar informação sobre congestionamento ao remetente?

A abstração mais importante provida pela camada de rede:

circuito virtual
ou
datagrama?



Rede de datagramas: o modelo da Internet

- não requer estabelecimento de chamada na camada de rede
- roteadores: não guardam estado sobre conexões fim a fim
  - não existe o conceito de "conexão" na camada de rede
- pacotes são roteados tipicamente usando endereços de destino
  - pacotes entre o mesmo par origem-destino podem seguir caminhos diferentes

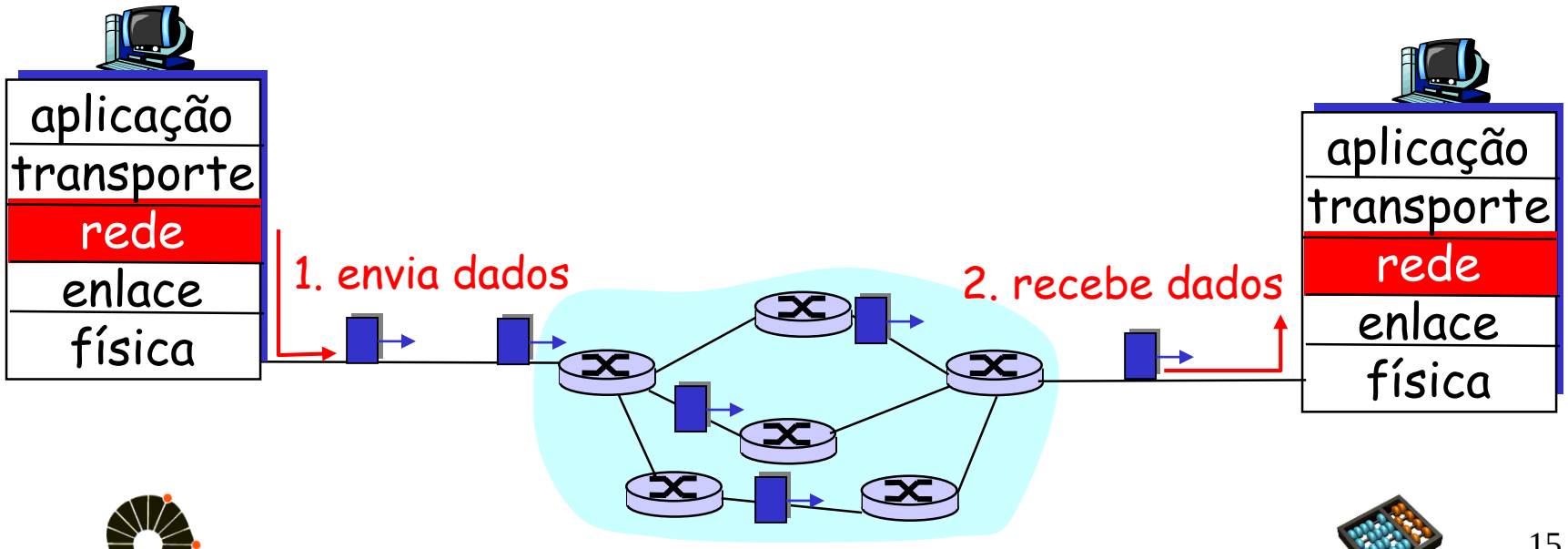


Tabela Encaminhamento

Faixa Endereço Destino	Interface 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
caso contrário	3

Casamento Prefixo mais longo (Longest prefix matching)

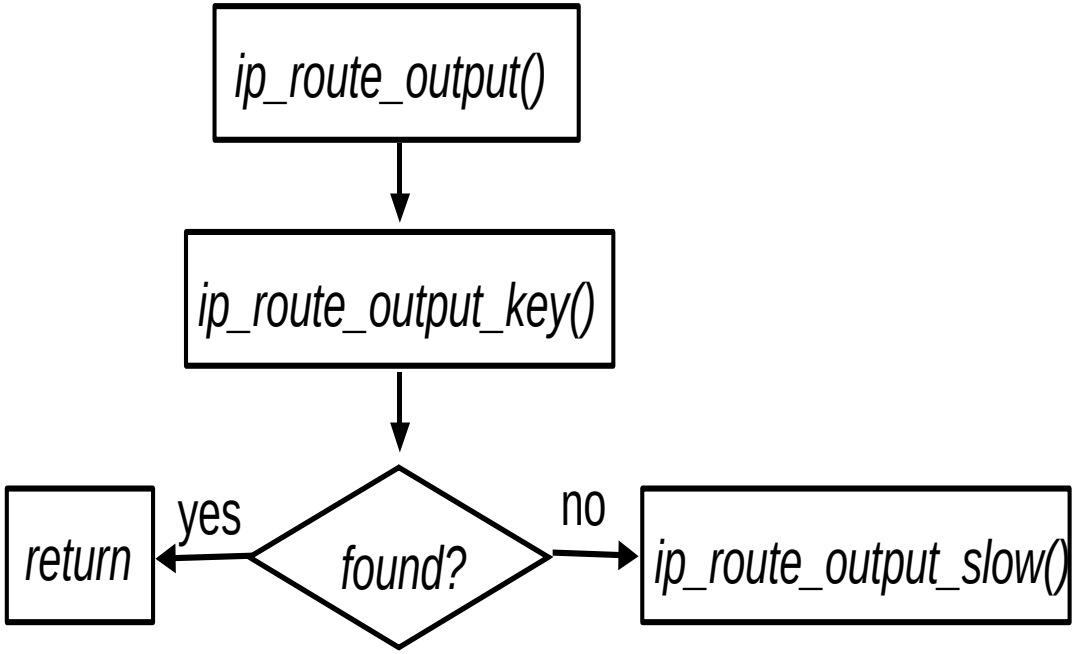
Prefix Match	Interface Enlace
11001000 00010111 00010	0
11001000 00010111 00011000	1
11001000 00010111 00011	2
caso contrario	3

Exemplos

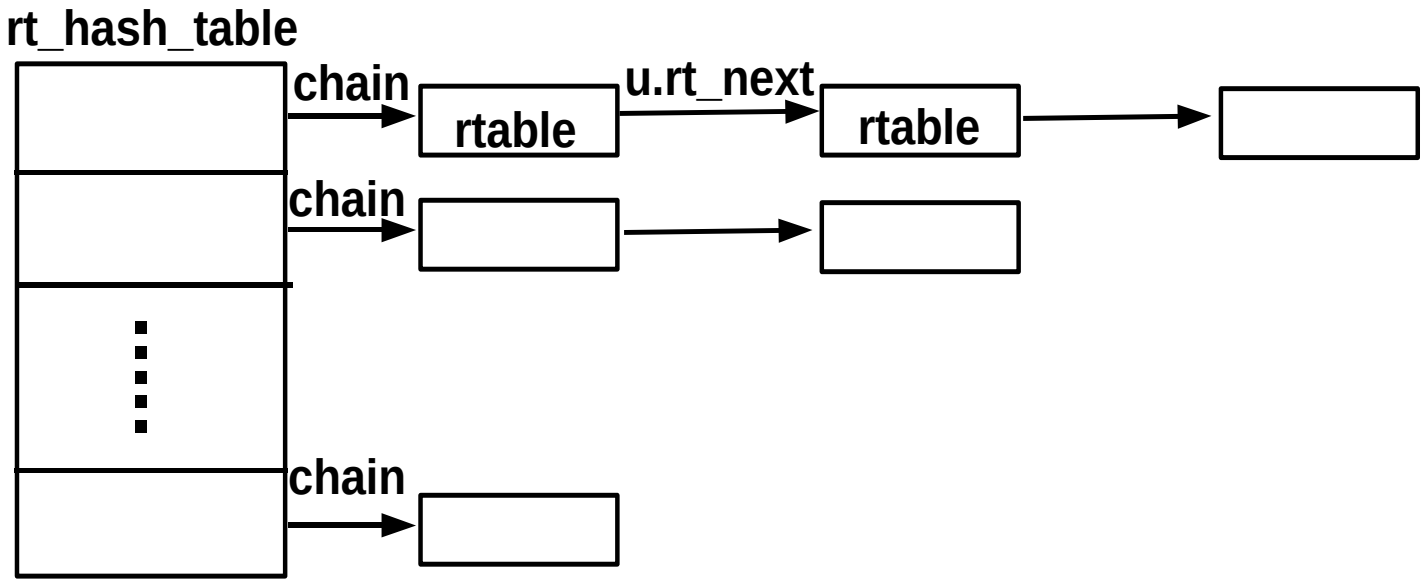
DA: 11001000 00010111 00010110 10100001

DA: 11001000 00010111 00011000 10101010

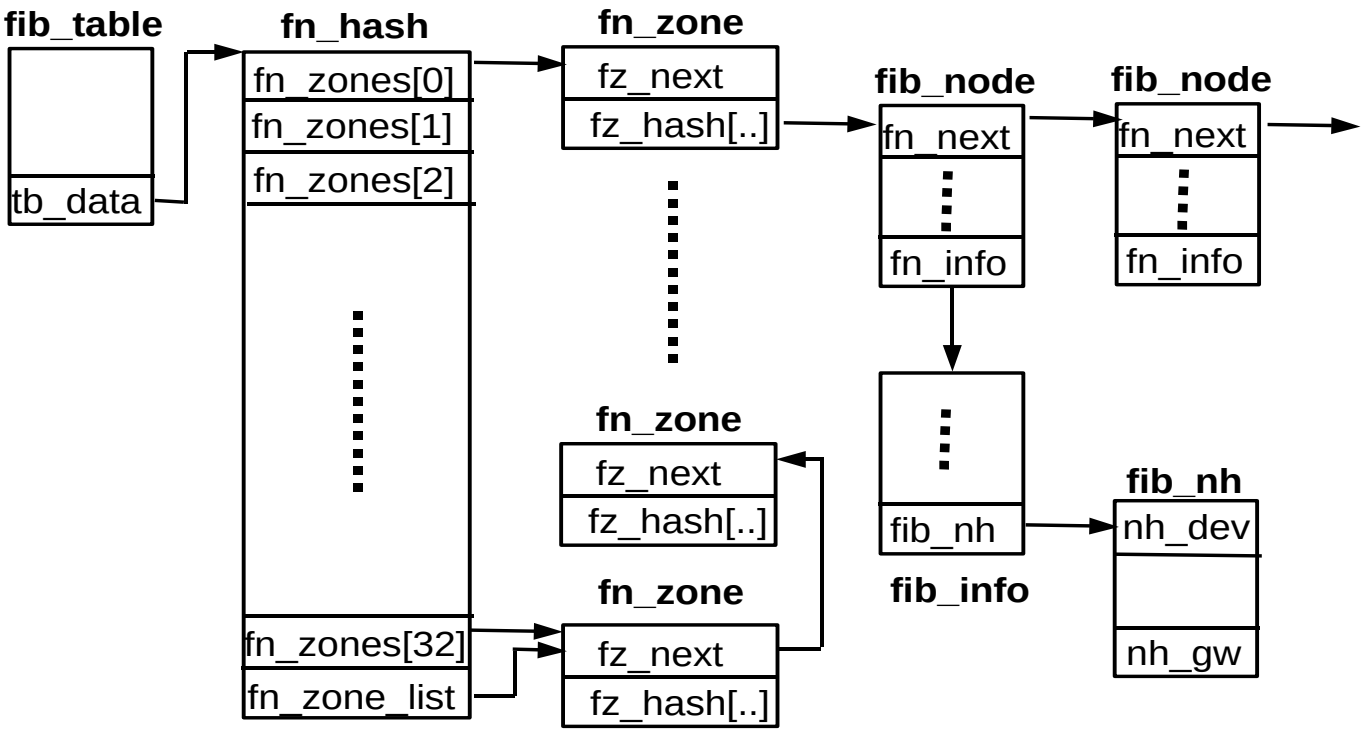
Consulta a Tabela no Linux



Cache para Acesso a Tabela



Estrutura da Tabela de Roteamento



Circuitos virtuais

"caminho da-origem-ao-destino se comporta como um circuito telefônico"

- ✓ em termos de desempenho
- ✓ em ações da rede ao longo do caminho da-origem-ao-destino

- estabelecimento de cada chamada *antes* do envio dos dados
- cada pacote tem ident. de CV (e não endereços origem/dest)
- *cada* roteador no caminho da-origem-ao-destino mantém "estado" para cada conexão que o atravessa
  - ✓ conexão da camada de transporte só envolve os 2 sistemas terminais
- recursos de enlace, roteador (banda, *buffers*) podem ser *alocados* ao CV
  - ✓ para permitir desempenho como de um circuito

Implementação CV

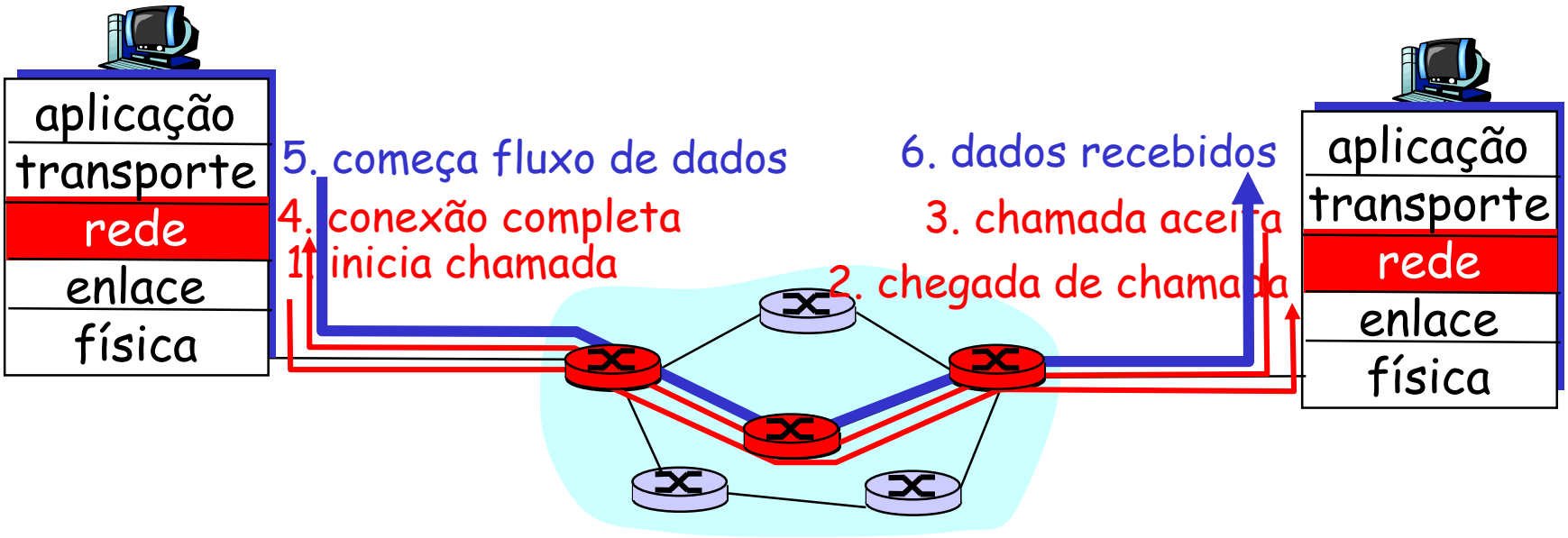
Um circuito virtual consiste de:

1. Caminho entre origem e destino
2. Identificador CV, um para cada enlace ao longo do caminho
3. Entradas nas tabelas de roteamento nos roteadores ao longo do caminho

- Pacote carrega identificador de CV ao invés de endereço destino
- Identificador de CV pode mudar a cada enlace
  - ✓ Novos identificadores de VC são gerados nas tabelas de roteamento

Circuitos virtuais: protocolos de sinalização

- usados para estabelecer, manter, destruir CV
- usados em ATM, frame-relay, X.25
- não usados na Internet de hoje



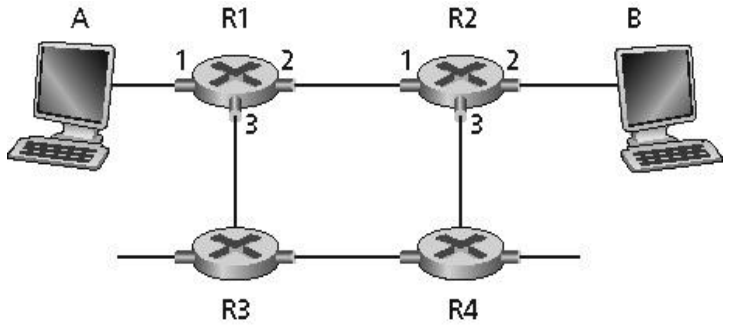


Tabela de comutação no roteador a noroeste:

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

Roteadores mantêm informações de estado de conexão

Modelos de serviço da camada de rede:

Arquitetura de Rede	Modelo de serviço	Garantias ?				Informa s/ congestion.?
		Banda	Perdas	Ordem	Tempo	
Internet	melhor esforço	nenhuma	não	não	não	não (inferido via perdas)
ATM	CBR	taxa constante	sim	sim	sim	sem congestion.
ATM	VBR	taxa garantida	sim	sim	sim	sem congestion.
ATM	ABR	mínima garantida	não	sim	não	sim
ATM	UBR	nenhuma	não	sim	não	não

- Modelo Internet está sendo estendido: Intserv, Diffserv

Rede de datagramas ou CVs: por quê?

Internet

- troca de dados entre computadores
  - ✓ serviço "elástico", sem reqs. temporais estritos
- sistemas terminais "inteligentes" (computadores)
  - ✓ podem se adaptar, exercer controle, recuperar de erros
  - ✓ núcleo da rede simples, complexidade na "borda"
- muitos tipos de enlaces
  - ✓ características diferentes

ATM

- evoluiu da telefonia
- conversação humana:
  - ✓ temporização estrita, requisitos de confiabilidade
  - ✓ requer serviço garantido
- sistemas terminais "burros"
  - ✓ telefones
  - ✓ complexidade dentro da rede

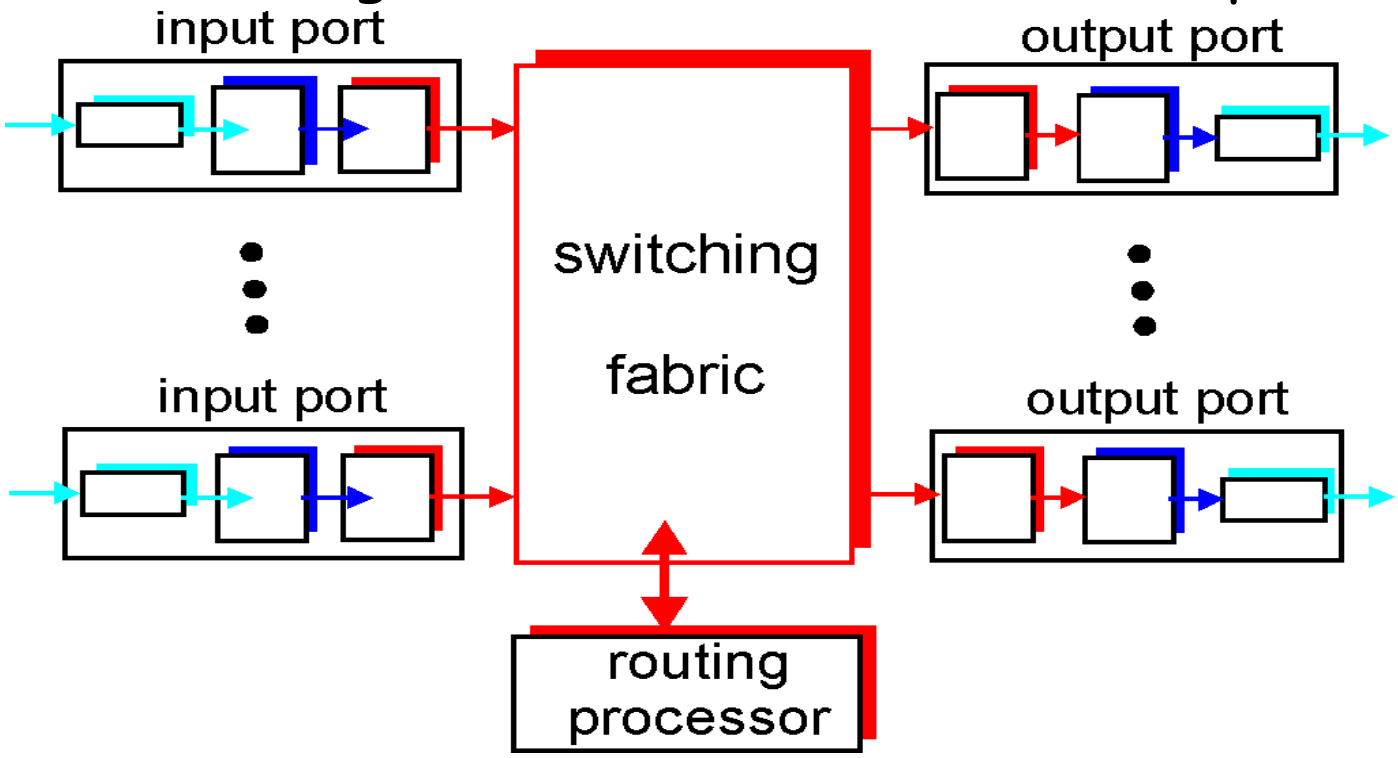
Roteiro

- 4. 1 Introdução
- 4.2 Circuitos virtuais x datagrama
- 4.3 Como é um roteador
- 4.4 Protocolo IP
  - Formato datagarma
  - endereçamento IPv4
  - ICMP
  - IPv6
- 4.5 Algoritmos de roteamento
  - Estado de enlace
  - Vetor distância
  - Roteamento hierarquico
- 4.6 Roteamento na Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Roteamento Broadcast e multicast

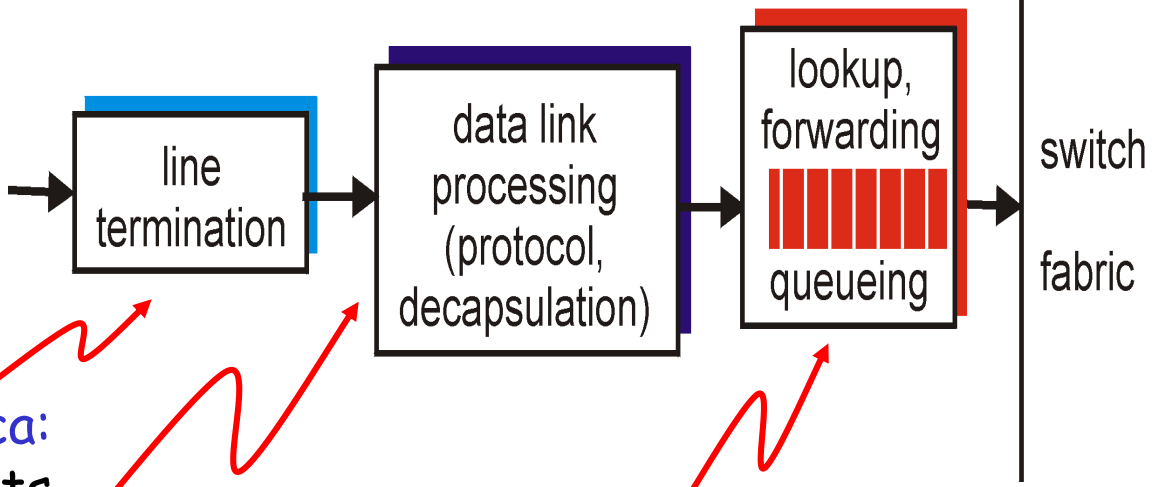
Sumário de Arquitetura de Roteadores

Duas funções chave de roteadores:

- usam algoritmos/protocolos de roteamento (RIP, OSPF, BGP)
- *comutam* datagramas do enlace de entrada para a saída



Funções da Porta de Entrada



Camada f'ísica:
recepção de bits

Camada de enlace:
p.ex., Ethernet
veja capítulo 5

Comutação descentralizada:

- dado o dest 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 matriz de comutação

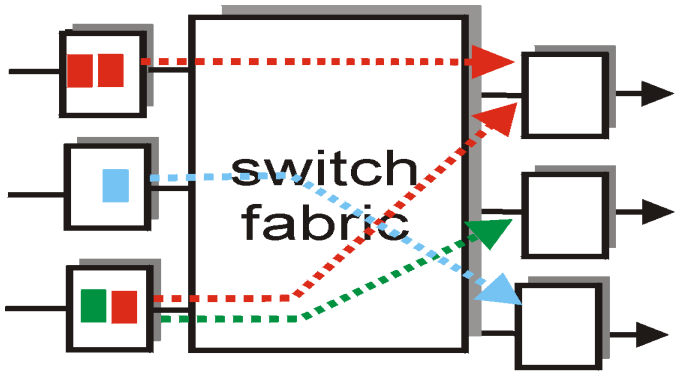
Tamanho do Buffer

- RFC 3439 recomenda que tamano médio do buffer deve corresponder a um RTT típico vezes a capacidade do enlace
  - ✓ 250 seg, C = 10 Gps link: 2.5 Gbit buffer
- Recomendação recente para N fluxos:

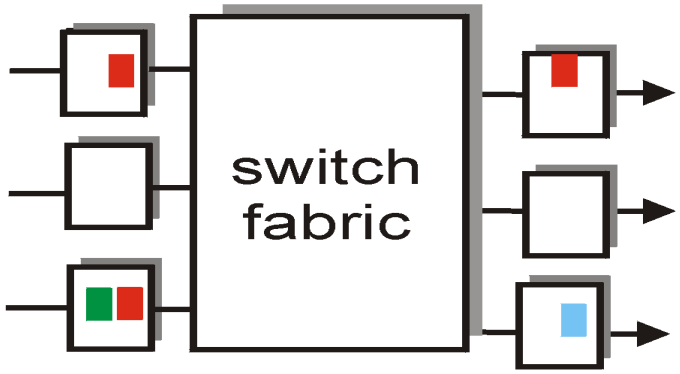
$$\frac{RTT \cdot C}{\sqrt{N}}$$

Filas na Porta de Entrada

- Se matriz de comutação for mais lenta do que a soma das portas de entrada juntas -> pode haver filas nas portas de entrada
- Bloqueio cabeça-de-linha (Head-of-the-Line - HOL): datagrama na cabeça da fila impede outros na mesma fila de avançarem
- *retardo de enfileiramento e perdas devido ao transbordo do buffer de entrada!*

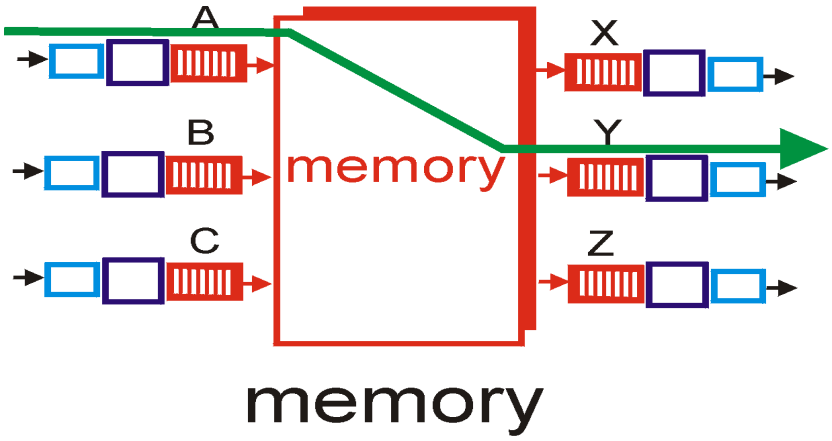


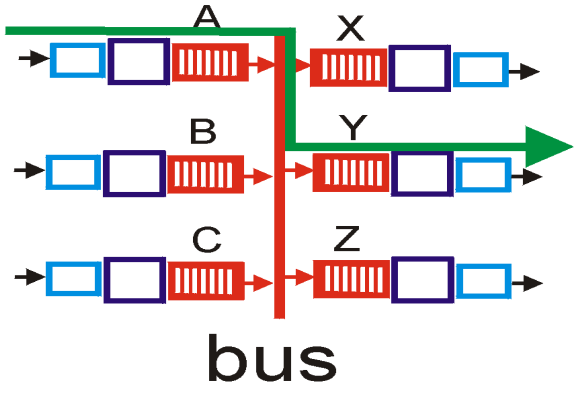
output port contention
at time t - only one red
packet can be transferred

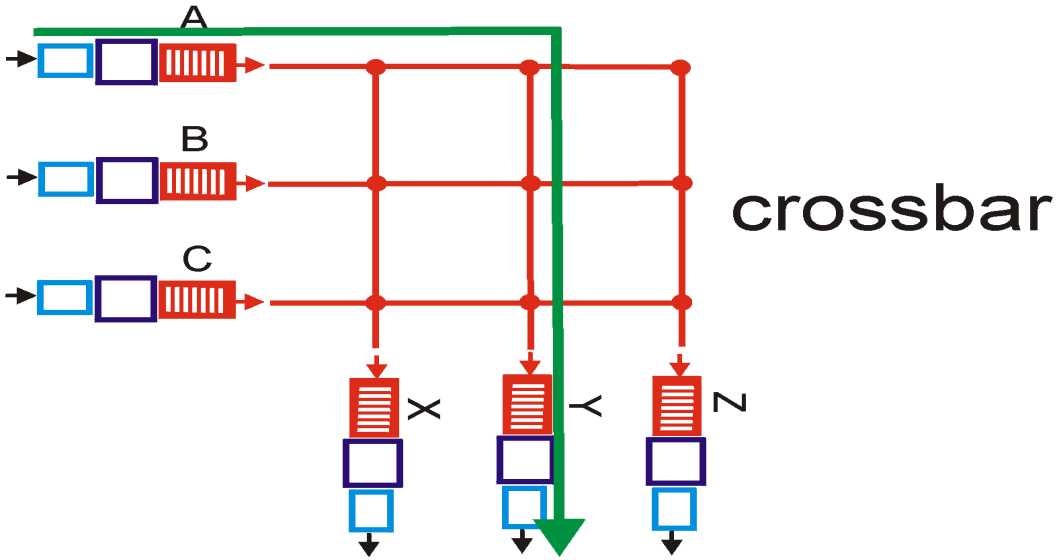


green packet
experiences HOL blocking

Três tipos de matriz de comutação



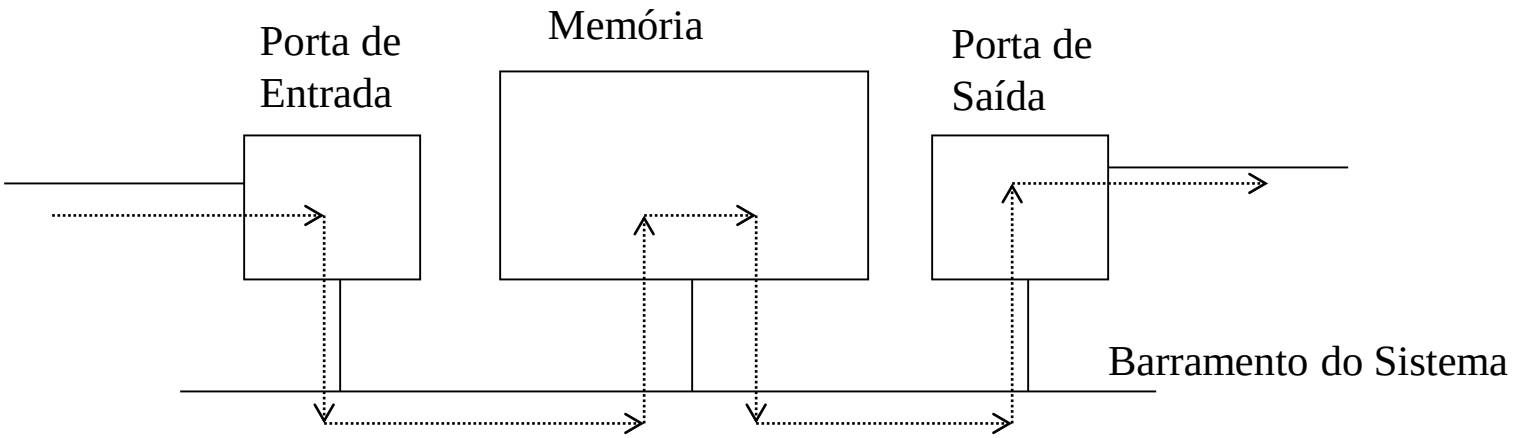




Comutação via Memória

Roteadores da primeira geração:

- pacote copiado pelo processador (único) do sistema
- velocidade limitada pela largura de banda da memória (2 travessias do barramento por datagrama)



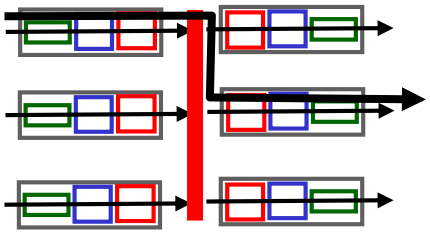
Roteadores modernos:

- processador da porta de entrada consulta tabela, copia para a memória
- Cisco Catalyst 8500

Switching via a bus

- datagram from input port memory

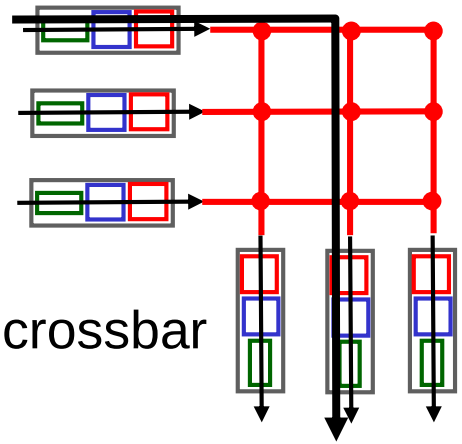
  to output port memory via a shared bus
- *bus contention:* switching speed limited by bus bandwidth
- 32 Gbps bus, Cisco 5600: sufficient speed for access and enterprise routers



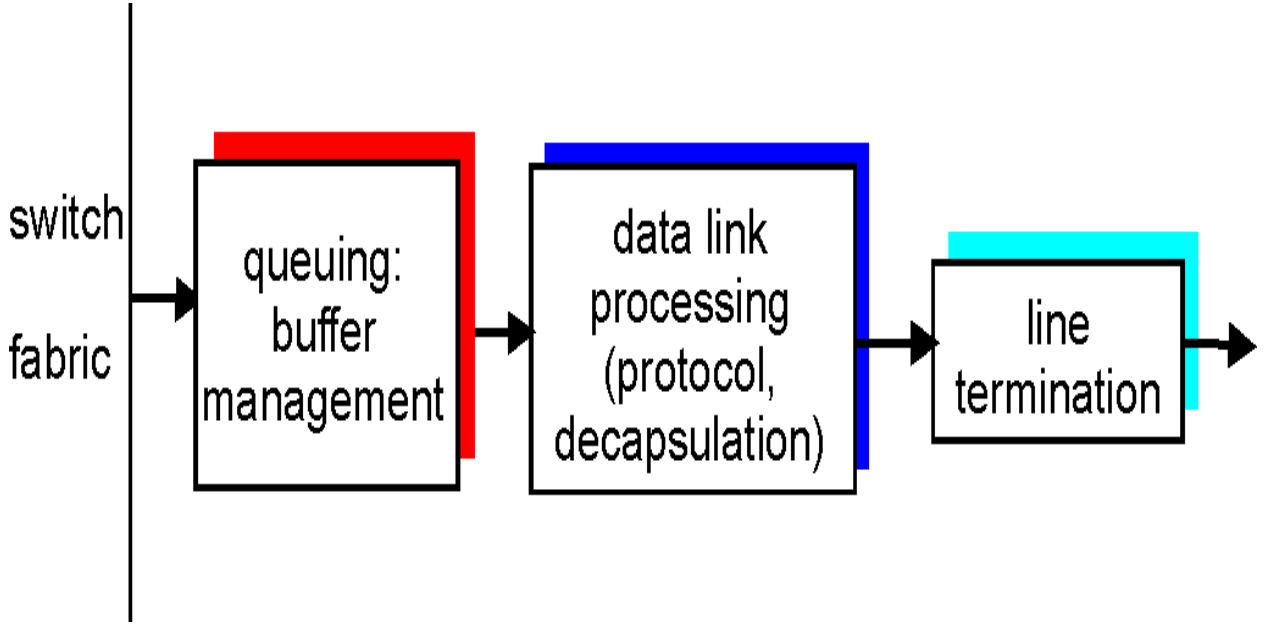
bus

Switching via interconnection network

- overcome bus bandwidth limitations
- banyan networks, crossbar, other interconnection nets initially developed to connect processors in multiprocessor
- advanced design: fragmenting datagram into fixed length cells, switch cells through the fabric.
- Cisco 12000: switches 60 Gbps through the interconnection network



Porta de Saída



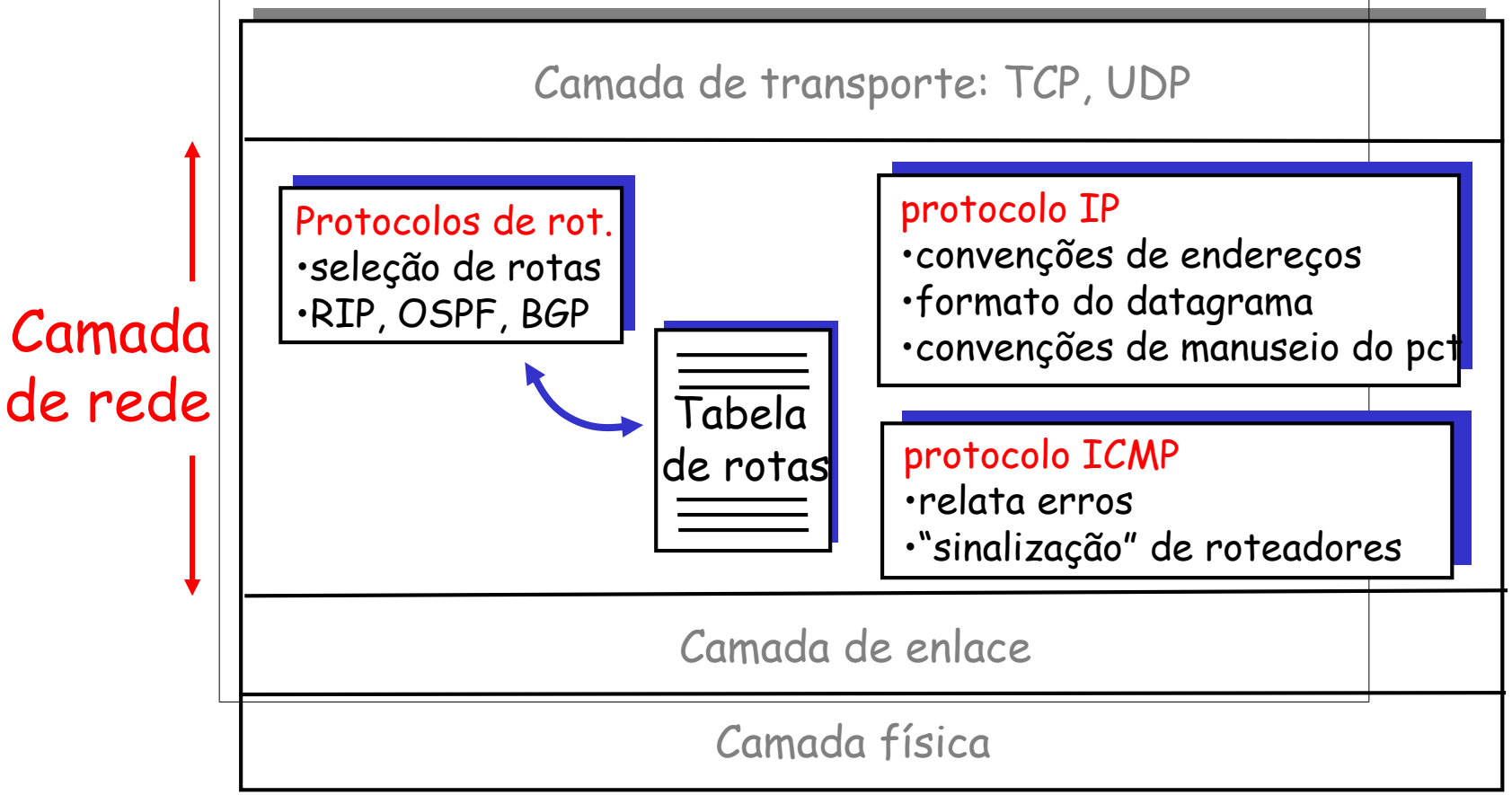
- *Buffers* necessários quando datagramas chegam da matriz de comutação mais rapidamente que a taxa de transmissão
- *Disciplina de escalonamento* escolhe um dos datagramas enfileirados para transmissão

Roteiro

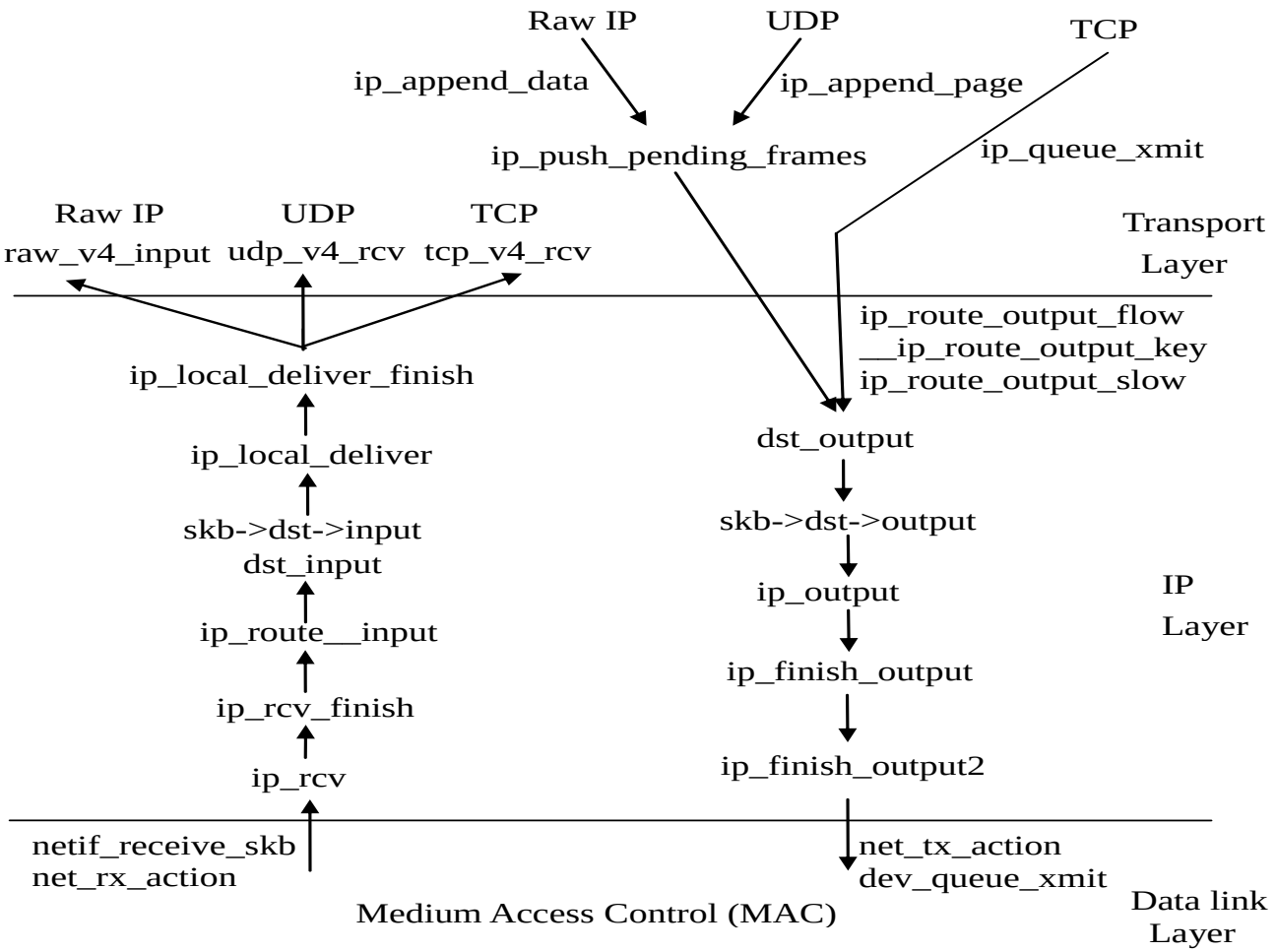
- 4. 1 Introdução
- 4.2 Circuitos virtuais x datagrama
- 4.3 Como é um roteador
- 4.4 Protocolo IP
  - Formato datagarma
  - endereçamento IPv4
  - ICMP
  - IPv6
- 4.5 Algoritmos de roteamento
  - Estado de enlace
  - Vetor distância
  - Roteamento hierarquico
- 4.6 Roteamento na Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Roteamento Broadcast e multicast

A Camada de Rede na Internet

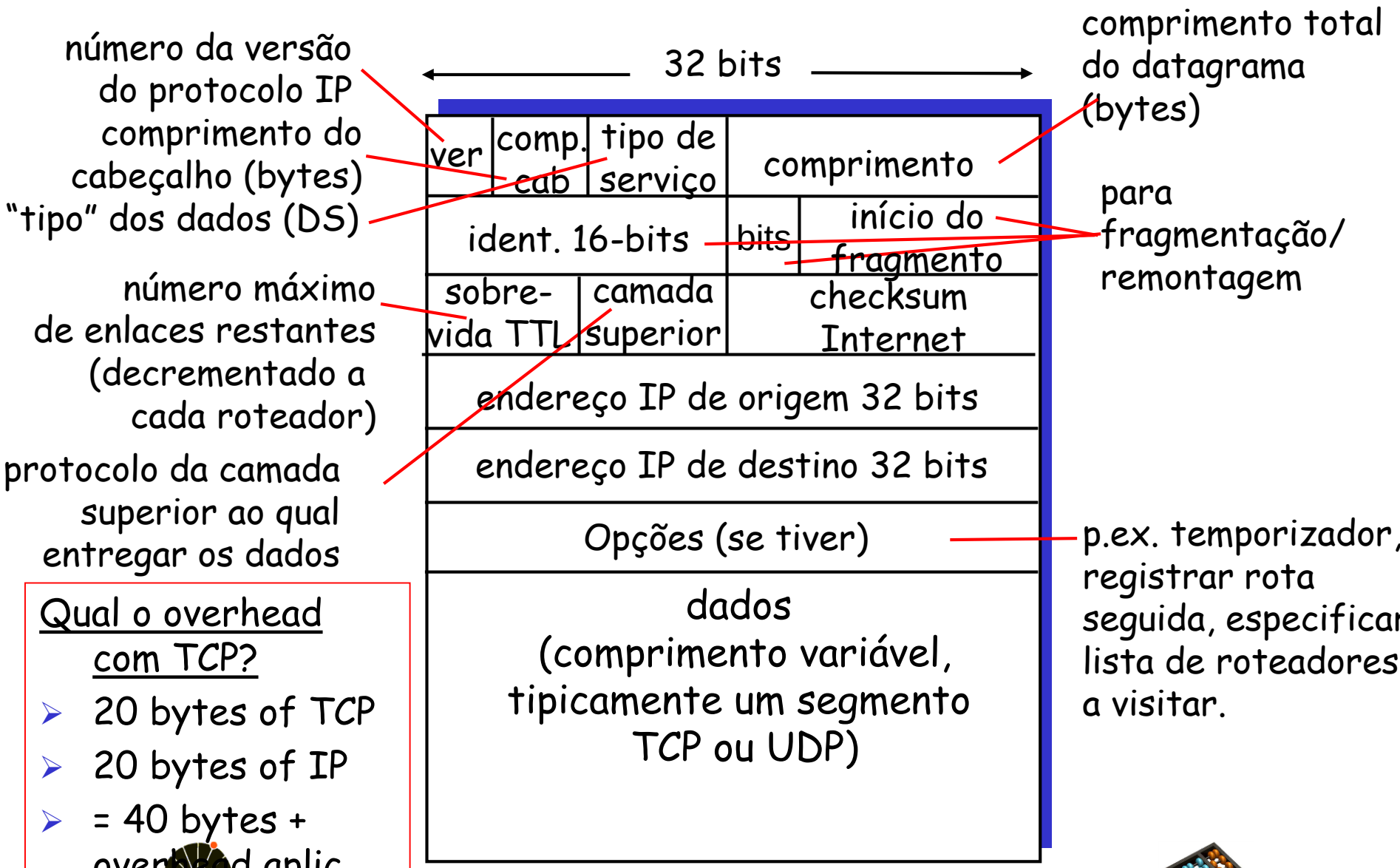
Funções da camada de rede em estações, roteadores:



Implementação IP - Linux



Formato do datagrama IP

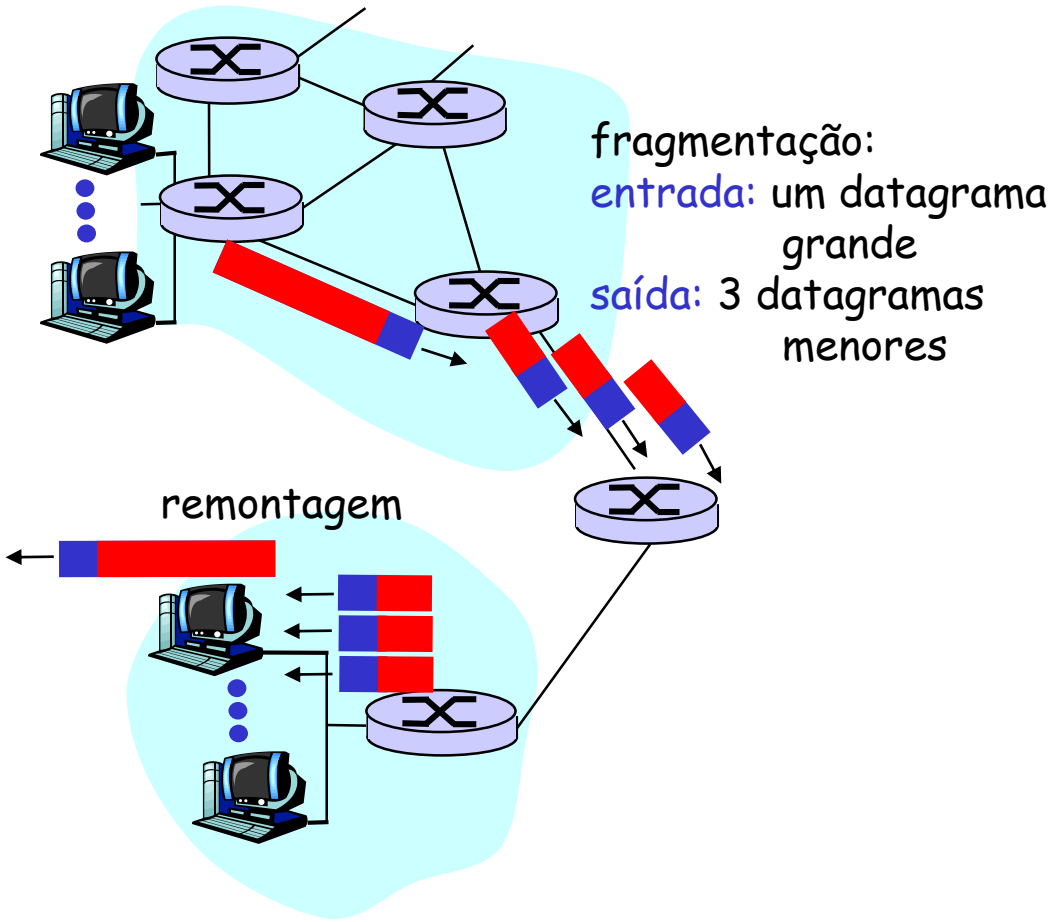


Qual o overhead com TCP?

- 20 bytes of TCP
- 20 bytes of IP
- = 40 bytes + overhead aplic.

IP: Fragmentação & Remontagem

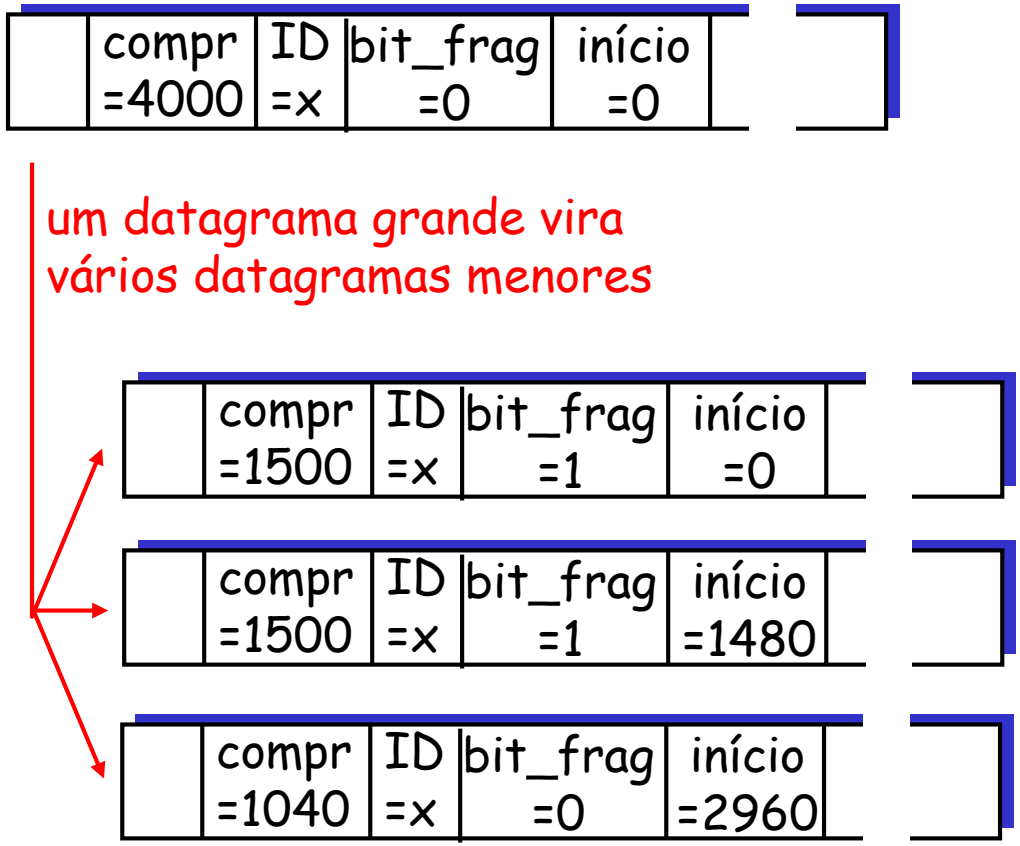
- cada enlace de rede tem MTU (max.transmission unit) - maior tamanho possível de quadro neste enlace.
  - tipos diferentes de enlace têm MTUs diferentes
- datagrama IP muito grande dividido ("fragmentado") dentro da rede
  - um datagrama vira vários datagramas
  - "remontado" apenas no destino final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



IP: Fragmentação & Remontagem

Exemplo

- Datagrama com 4000 bytes
- MTU = 1500 bytes

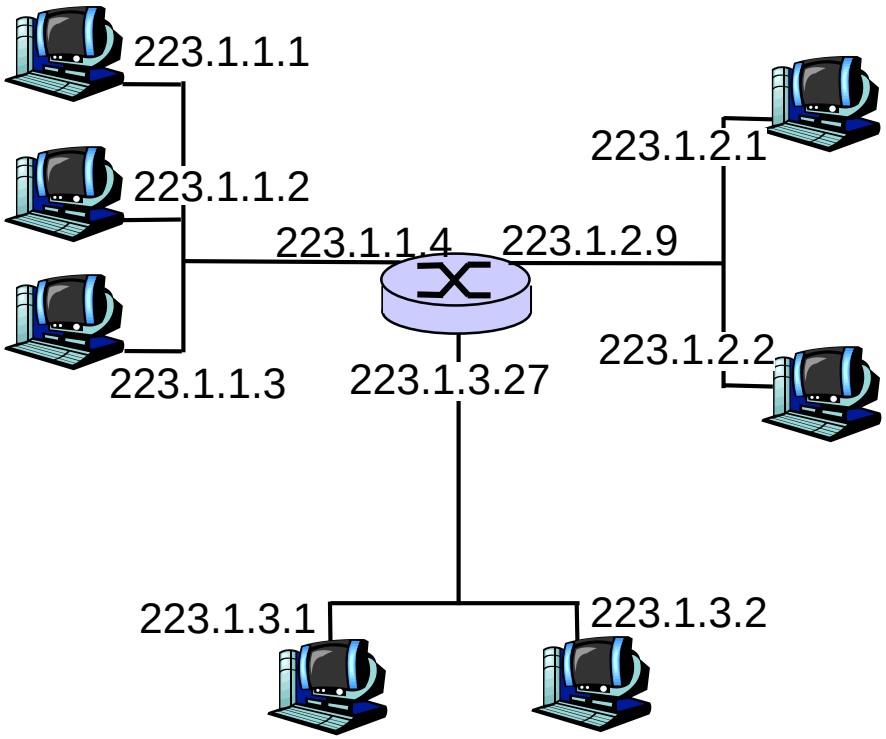


Roteiro

- 4. 1 Introdução
- 4.2 Circuitos virtuais x datagrama
- 4.3 Como é um roteador
- 4.4 Protocolo IP
  - ✓ Formato datagarma
  - ✓ endereçamento IPv4
  - ✓ ICMP
  - ✓ IPv6
- 4.5 Algoritmos de roteamento
  - ✓ Estado de enlace
  - ✓ Vetor distância
  - ✓ Roteamento hierarquico
- 4.6 Roteamento na Internet
  - ✓ RIP
  - ✓ OSPF
  - ✓ BGP
- 4.7 Roteamento Broadcast e multicast

Endereçamento IP: introdução

- 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 pode ter múltiplas interfaces
  - ✓ endereço IP associado à interface



223.1.1.1 = 11011111 00000001 00000001 00000001

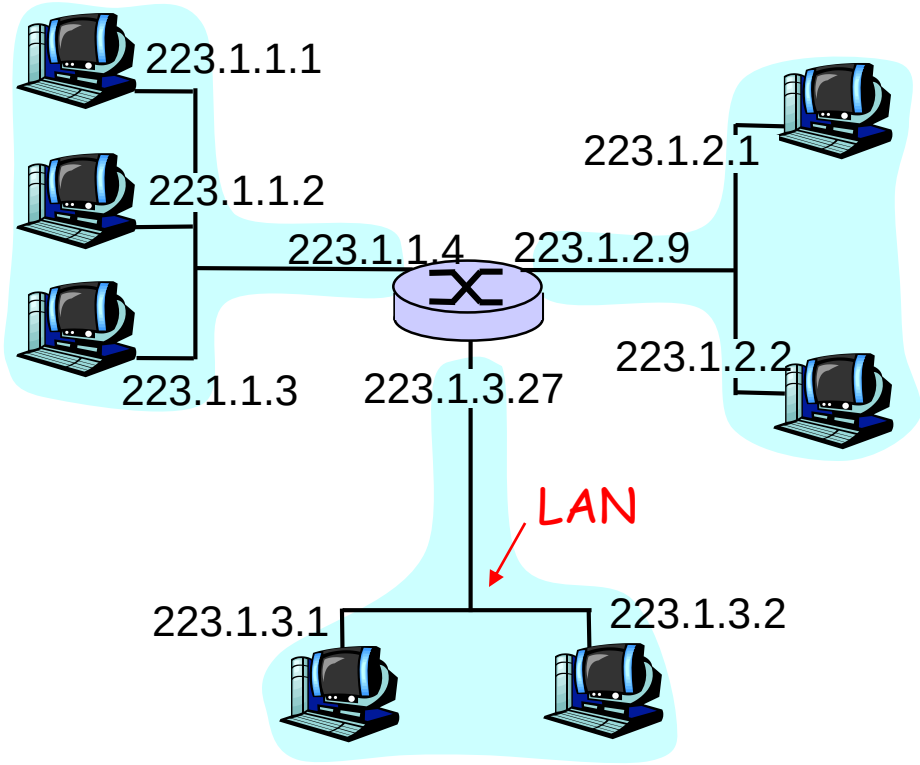
223 1 1 1

Endereçamento IP

- endereço IP:
  - ✓ parte de rede (bits de mais alta ordem)
  - ✓ parte de estação (bits de mais baixa ordem)
- *O que é uma rede IP?*

(da perspectiva do endereço IP)

  - ✓ interfaces de dispositivos com a mesma parte de rede nos seus endereços IP
  - ✓ podem alcançar um ao outro sem passar por um roteador



Esta rede consiste de 3 redes IP (para endereços IP começando com 223, os primeiros 24 bits são a parte de rede)

Enviando um datagrama da origem ao destino

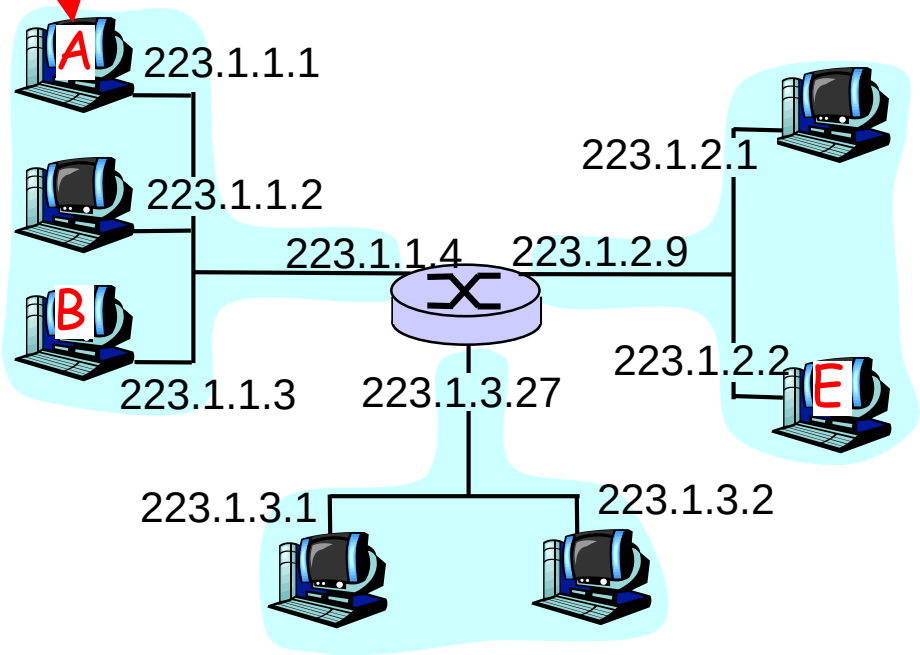
datagrama IP:

aampos misc	end. IP origem	end. IP dest	dados

- datagrama permanece inalterado, enquanto passa da origem ao destino
- campos de endereços de interesse aqui

tabela de rotas em *A*

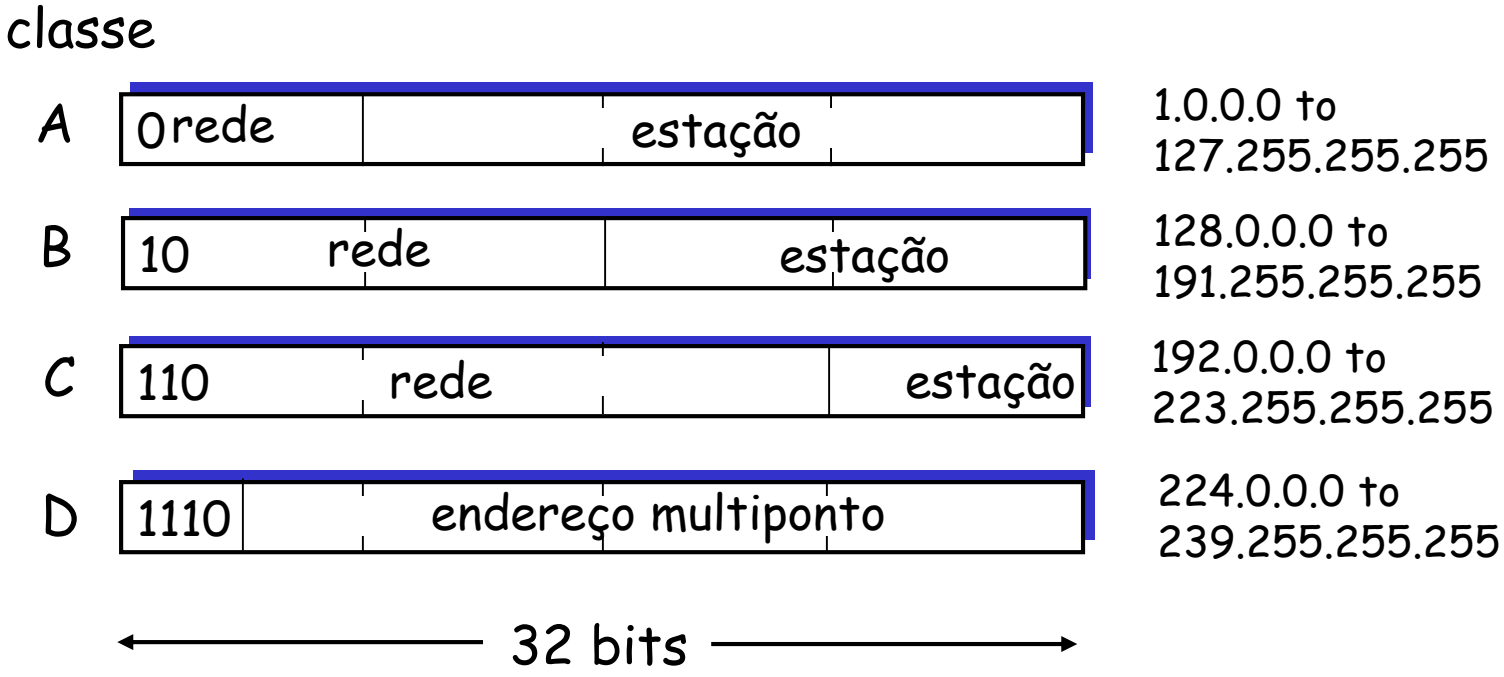
rede dest.	próx. rot.	Nenlaces
223.1.1		1
223.1.2	223.1.1.4	2
223.1.3	223.1.1.4	2



Endereços IP

dada a noção de "rede", vamos reexaminar endereços IP:

endereçamento "baseado em classes":



Endereçamento IP: CIDR

- Endereçamento baseado em classes:
  - uso ineficiente e esgotamento do espaço de endereços
  - p.ex., rede da classe B aloca endereços para 65K estações, mesmo se houver apenas 2K estações nessa rede
- CIDR: Classless InterDomain Routing
  - parte de rede do endereço de comprimento arbitrário
  - formato de endereço: a.b.c.d/x, onde x é no. de bits na parte de rede do endereço

200.23.16.0/23

Endereçamento IP: a última palavra...

P: Como um provedor IP consegue um bloco de endereços?

A: ICANN: Internet Corporation for Assigned Names and Numbers

- ✓ aloca endereços
- ✓ gerencia DNS
- ✓ aloca nomes de domínio, resolve disputas

(no Brasil, estas funções foram delegadas ao Comitê Gestor Internet BR)

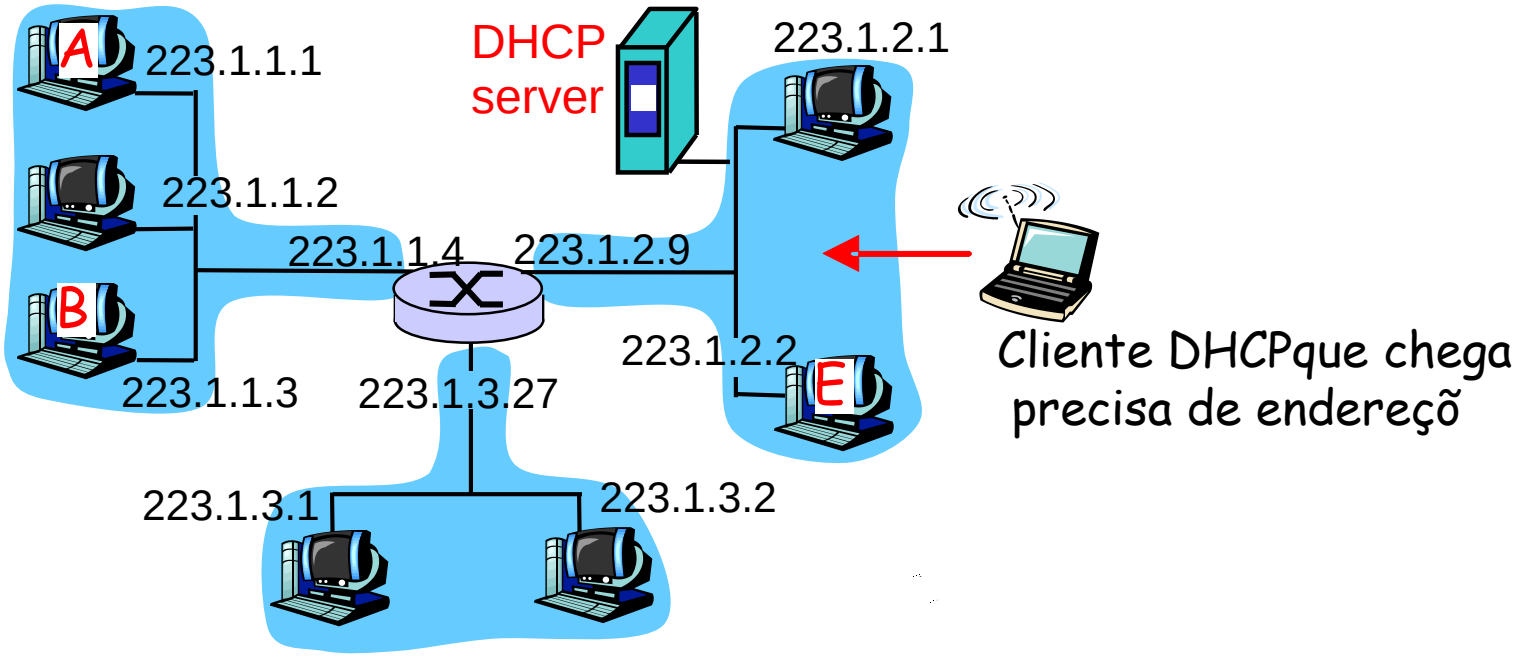
Endereços IP: como conseguir um?

Rede (parte de rede):

- conseguir alocação a partir do espaço de endereços do seu provedor IP

Bloco do provedor	11001000 00010111 00010000	00000000	200.23.16.0/20
Organização 0	11001000 00010111 0001000**0**	00000000	200.23.16.0/23
Organização 1	11001000 00010111 0001001**0**	00000000	200.23.18.0/23
Organização 2	11001000 00010111 0001010**0**	00000000	200.23.20.0/23
...	.....	....	....
Organização 7	11001000 00010111 0001111**0**	00000000	200.23.30.0/23

DHCP

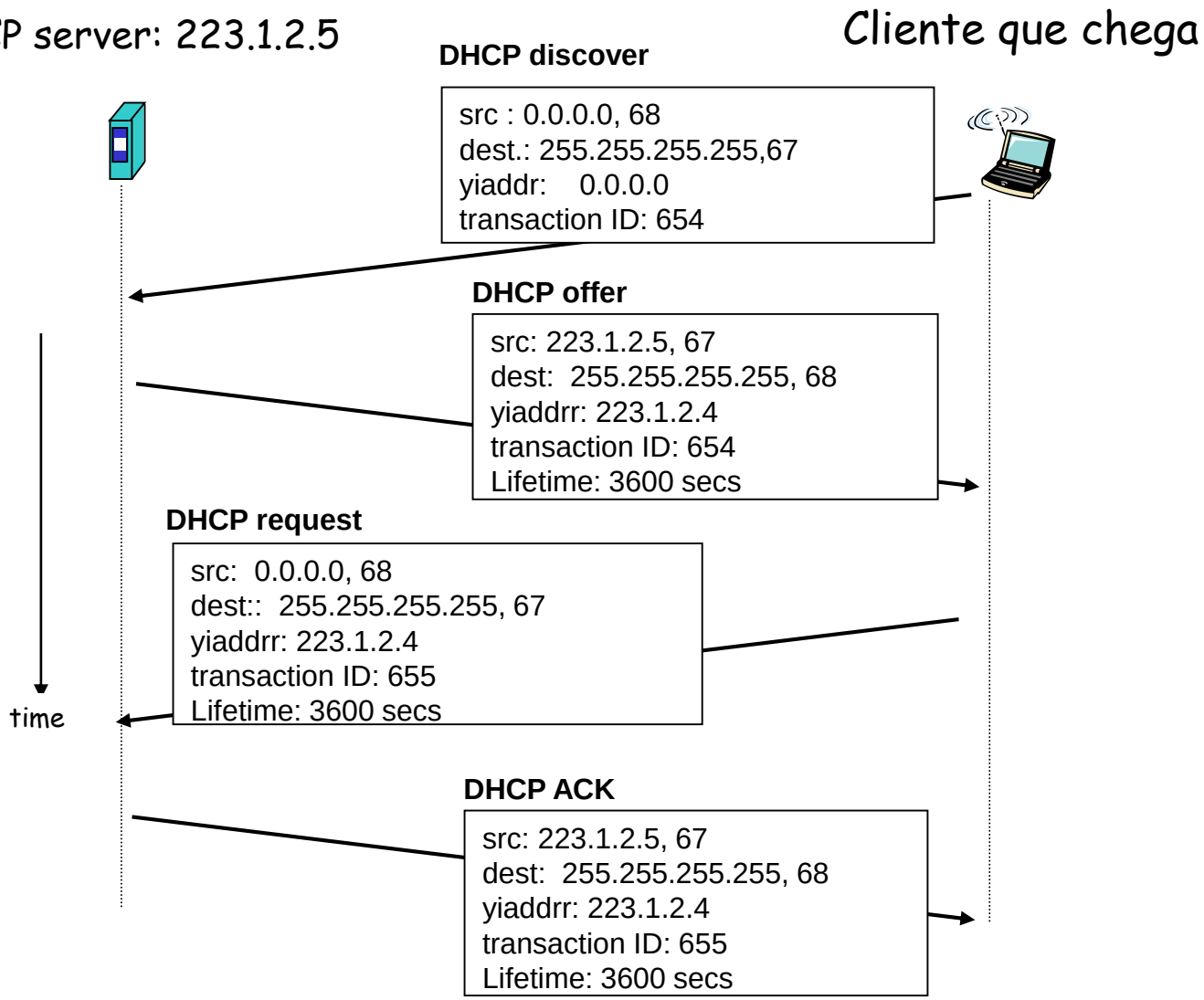

A
223.1.1.1
DHCP
server
223.1.2.1
223.1.1.2
223.1.1.4
223.1.2.9
B
223.1.2.2
E
223.1.1.3
223.1.3.27
Cliente DHCPque chega
precisa de endereçõ
223.1.3.1
223.1.3.2

Endereços IP: como conseguir um?

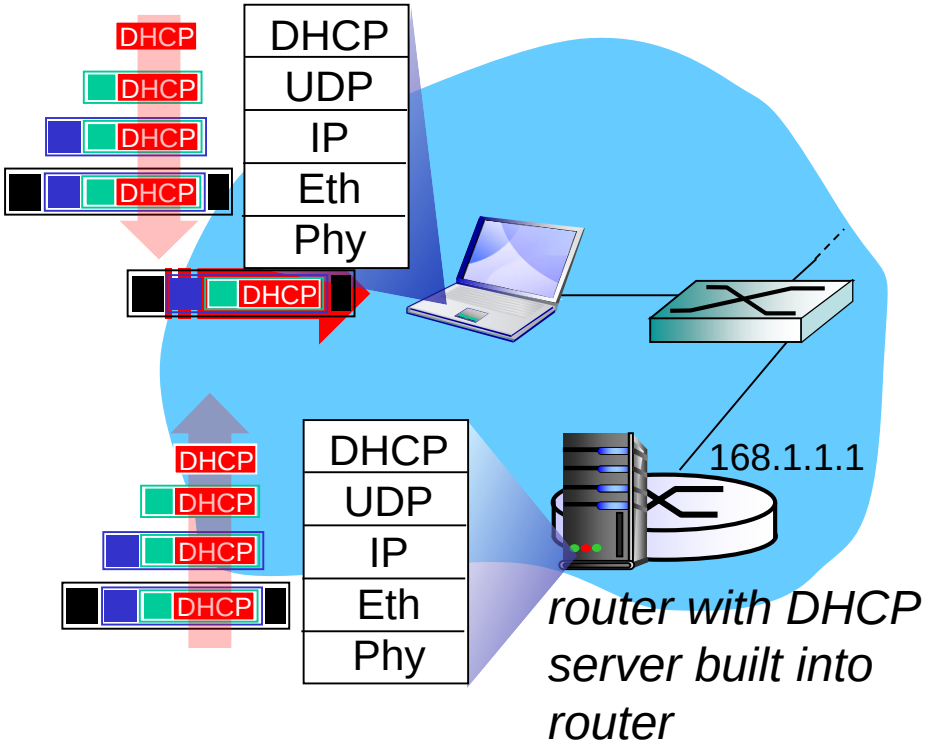
Estações (parte de estação):

- codificado pelo administrador num arquivo
  - Windows: control-panel->network->configuration->tcp/ip->properties
  - UNIX: /etc/rc.config
- DHCP: Dynamic Host Configuration Protocol: obtém endereço dinamicamente: "plug-and-play"
  - estação difunde mensagem "DHCP discover"
  - servidor DHCP responde com "DHCP offer"
  - estação solicita endereço IP: "DHCP request"
  - servidor DHCP envia endereço: "DHCP ack"

DHCP client-server scenario

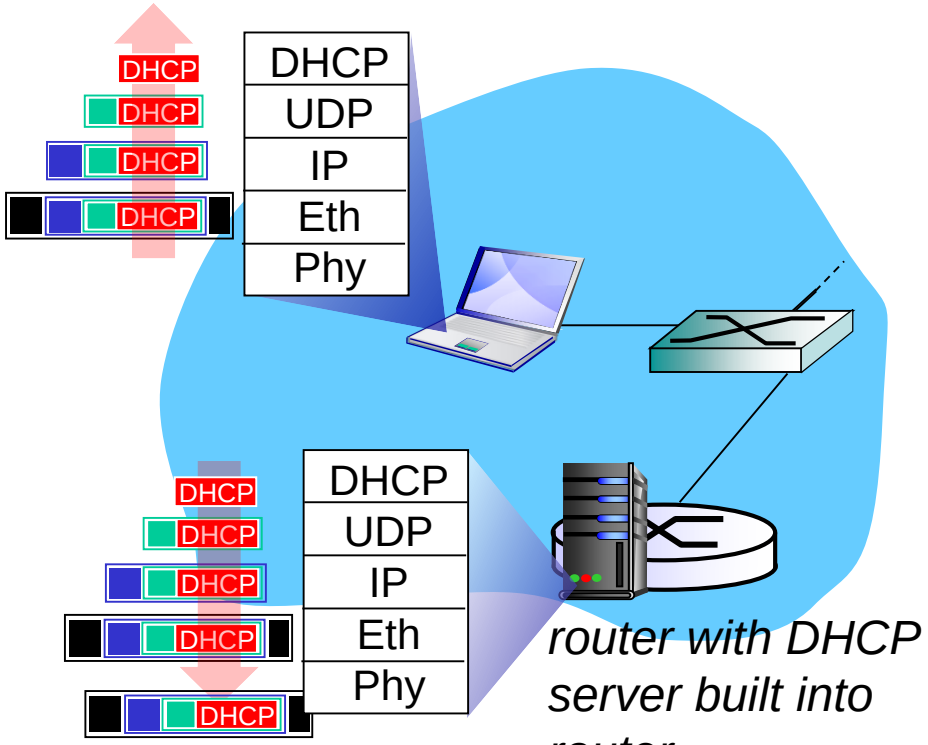


DHCP: example



- connecting laptop needs its IP address, addr of first-hop router, addr of DNS server: use DHCP
- DHCP request encapsulated in UDP, encapsulated in IP, encapsulated in 802.1 Ethernet
- Ethernet frame broadcast (dest: FFFFFFFFFFFF) on LAN, received at router running DHCP server
- Ethernet demuxed to IP demuxed, UDP demuxed to DHCP

DHCP: example



- DCP server formulates DHCP ACK containing client's IP address, IP address of first-hop router for client, name & IP address of DNS server
- encapsulation of DHCP server, frame forwarded to client, demuxing up to DHCP at client
- client now knows its IP address, name and IP address of DSN server, IP address of its first-hop router

DHCP: Wireshark output (home LAN)

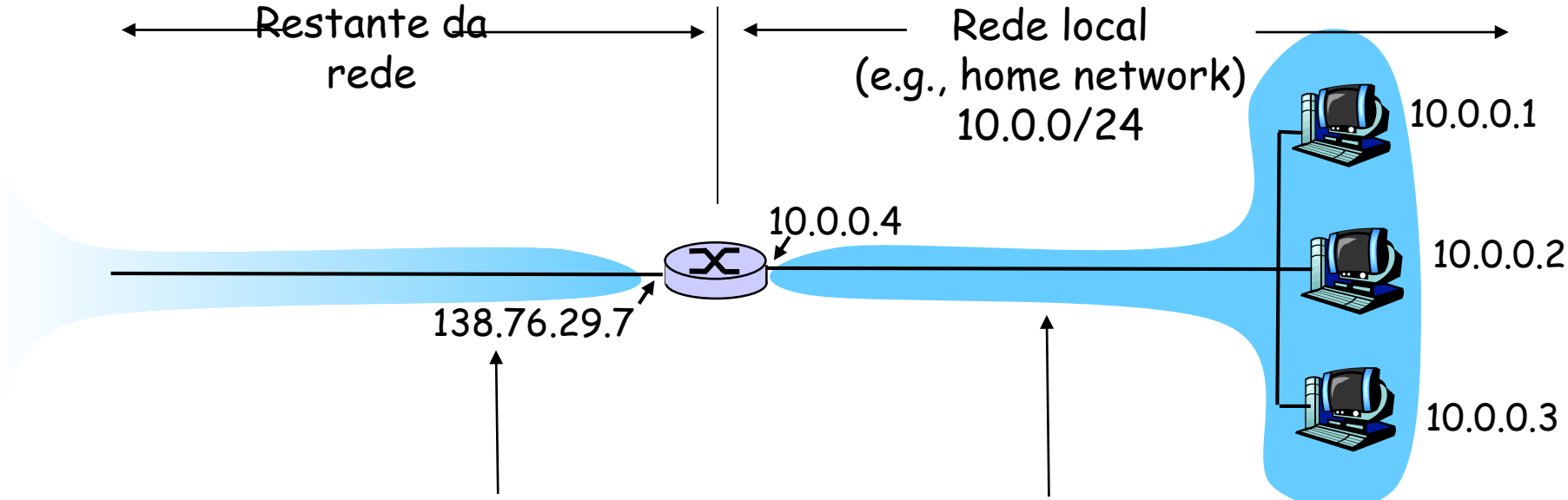
request

Message type: **Boot Request (1)**
Hardware type: Ethernet
Hardware address length: 6
Hops: 0
Transaction ID: 0x6b3a11b7
Seconds elapsed: 0
Bootp flags: 0x0000 (Unicast)
Client IP address: 0.0.0.0 (0.0.0.0)
Your (client) IP address: 0.0.0.0 (0.0.0.0)
Next server IP address: 0.0.0.0 (0.0.0.0)
Relay agent IP address: 0.0.0.0 (0.0.0.0)
Client MAC address: Wistron_23:68:8a (00:16:d3:23:68:8a)
Server host name not given
Boot file name not given
Magic cookie: (OK)
Option: (t=53,l=1) **DHCP Message Type = DHCP Request**
Option: (61) Client identifier
    Length: 7; Value: 010016D323688A;
    Hardware type: Ethernet
    Client MAC address: Wistron_23:68:8a (00:16:d3:23:68:8a)
Option: (t=50,l=4) Requested IP Address = 192.168.1.101
Option: (t=12,l=5) Host Name = "nomad"
Option: (55) Parameter Request List
    Length: 11; Value: 010F03062C2E2F1F21F92B
    1 = Subnet Mask; 15 = Domain Name
    3 = Router; 6 = Domain Name Server
    44 = NetBIOS over TCP/IP Name Server
    ……

reply

Message type: **Boot Reply (2)**
Hardware type: Ethernet
Hardware address length: 6
Hops: 0
Transaction ID: 0x6b3a11b7
Seconds elapsed: 0
Bootp flags: 0x0000 (Unicast)
Client IP address: 192.168.1.101 (192.168.1.101)
Your (client) IP address: 0.0.0.0 (0.0.0.0)
Next server IP address: 192.168.1.1 (192.168.1.1)
Relay agent IP address: 0.0.0.0 (0.0.0.0)
Client MAC address: Wistron_23:68:8a (00:16:d3:23:68:8a)
Server host name not given
Boot file name not given
Magic cookie: (OK)
Option: (t=53,l=1) DHCP Message Type = DHCP ACK
Option: (t=54,l=4) Server Identifier = 192.168.1.1
Option: (t=1,l=4) Subnet Mask = 255.255.255.0
Option: (t=3,l=4) Router = 192.168.1.1
Option: (6) Domain Name Server
    Length: 12; Value: 445747E2445749F244574092;
    IP Address: 68.87.71.226;
    IP Address: 68.87.73.242;
    IP Address: 68.87.64.146
Option: (t=15,l=20) Domain Name = "hsd1.ma.comcast.net."

NAT: Network Address Translation



Todos os datagramas *saindo* da rede local tem o mesmo endereço NAT IP: 138.76.29.7, diferentes números de portas fontes

Datagramas com origem ou destino nesta rede tem endereço 10.0.0/24 para fonte, e de destino o usual

NAT: Network Address Translation

- Motivação: rede local usa apenas um endereço IP:
  - ✓ Não há necessidade de alocar faixas de endereços de um ISP
    - apenas um endereço IP é usado para todos os dispositivos
  - ✓ Permite mudar o endereço dos dispositivos internos sem necessitar notificar o mundo externo;
  - ✓ Permite a mudança de ISPs sem necessitar mudar os endereços dos dispositivos internos da rede local
  - ✓ Dispositivos internos a rede, não são visíveis nem endereçaveis pelo mundo externo (melhora segurança);

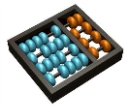
NAT: Network Address Translation

Implementação: roteador NAT deve;

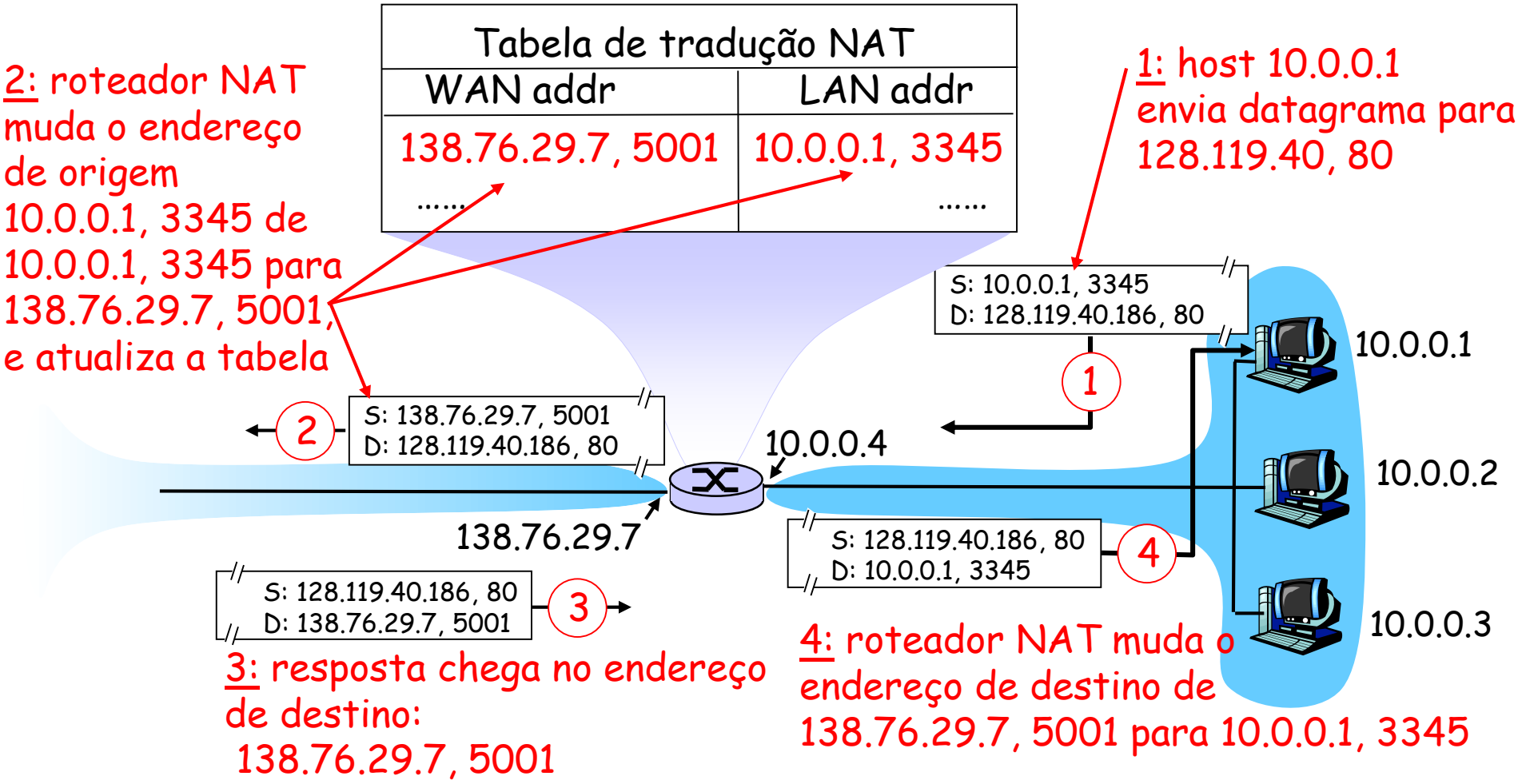
- ✓ *Datagramas que saem: trocar* (endereço IP fonte, porta #) de cada datagrama de saída para (endereço NAT IP, nova porta #)

  . . . clientes/servidores remotos irão responder usando (endereço NAT IP, nova porta #) como endereço destino.

- ✓ *guardar (na tabela de tradução de endereços NAT): os pares de tradução de endereços* (endereço IP fonte, porta #) para (endereços NAT IP, nova porta #)
- ✓ *Datagramas qeu chegam: trocar* (endereço NAT IP, nova porta #) no campo de destino de cada datagrama que chega com o correspondente (endereço IP fonte, porta #) armazenado na tabela NAT



NAT: Network Address Translation



NAT: Network Address Translation

- Campo de porta de 16-bit :
  - ✓ 60,000 conexões simultâneas com um único endereço de rede;
- NAT é controverso:
  - ✓ Roteadores devem fazer processamentos até no máximo a camada 3;
  - ✓ Viola o "conceito fim-a-fim"
    - A possibilidade de suporte a NAT deve ser levado em consideração pelos desenvolvedores de aplicações;
  - ✓ O problema de diminuição do número de endereços deveria ser tratada por IPv6;

Tipos NAT

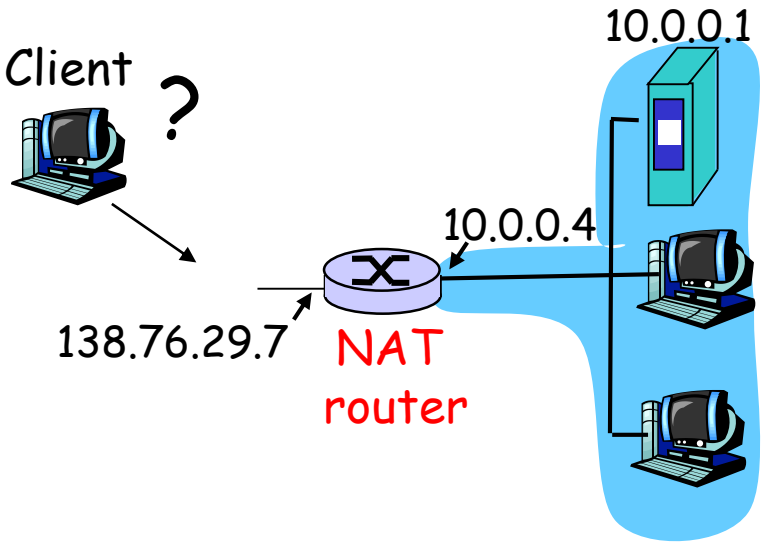
- Cone – mapeia sem restrições;
- Cone restrito: memoriza endereço IP de destinatário e verifica se pacotes que chegam são do destinatário;
- Cone restrito com porta inclusiva: verifica se IP e porta de pacote que chega é o mesmo do destinatário;
- Simétrico: identico ao anterior e além disso mapeamento fonte-destinatário é único.

NAT - Aplicações

- Aplicações que fornecem endereços IP ou de porta podem ter sua funcionalidade prejudicada, havendo necessidade de modificação do datagrama. Exemplos:
  - ✓ Comando PORT no FTP;
  - ✓ Mensagens de erro ICMP – necessita recomputar checksun após tradução.
- Solução: Application level gateway acompanhando NAT

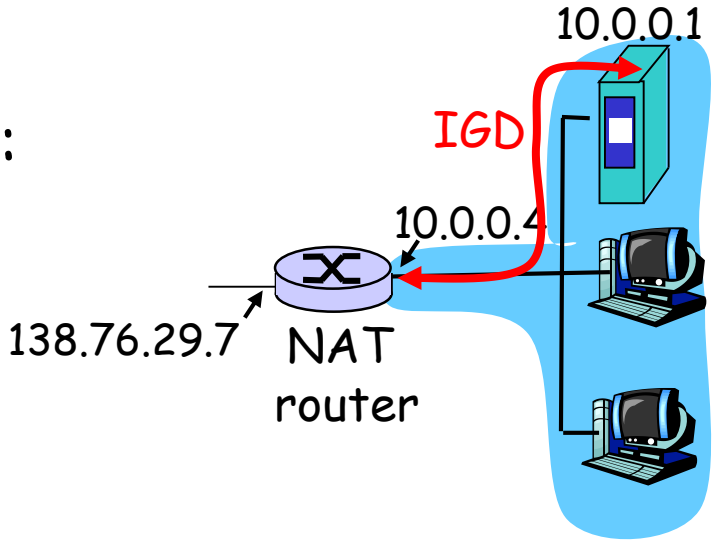
Problema NAT traversal

- Cliente deseja conectar-se ao servidor com endereço 10.0.0.1
  - ✓ Endereço servidor 10.0.0.1 na rede local (cliente não pode usar como endereço destino)
  - ✓ Somente um endereço (NATed) externamente visível: 138.76.29.7
- Uma solução: configurar NAT manualmente para encaminhar requisição de conexão a uma certa porta do servidor
- exemplo., (123.76.29.7, port 2500) sempre envia para 10.0.0.1 port 25000



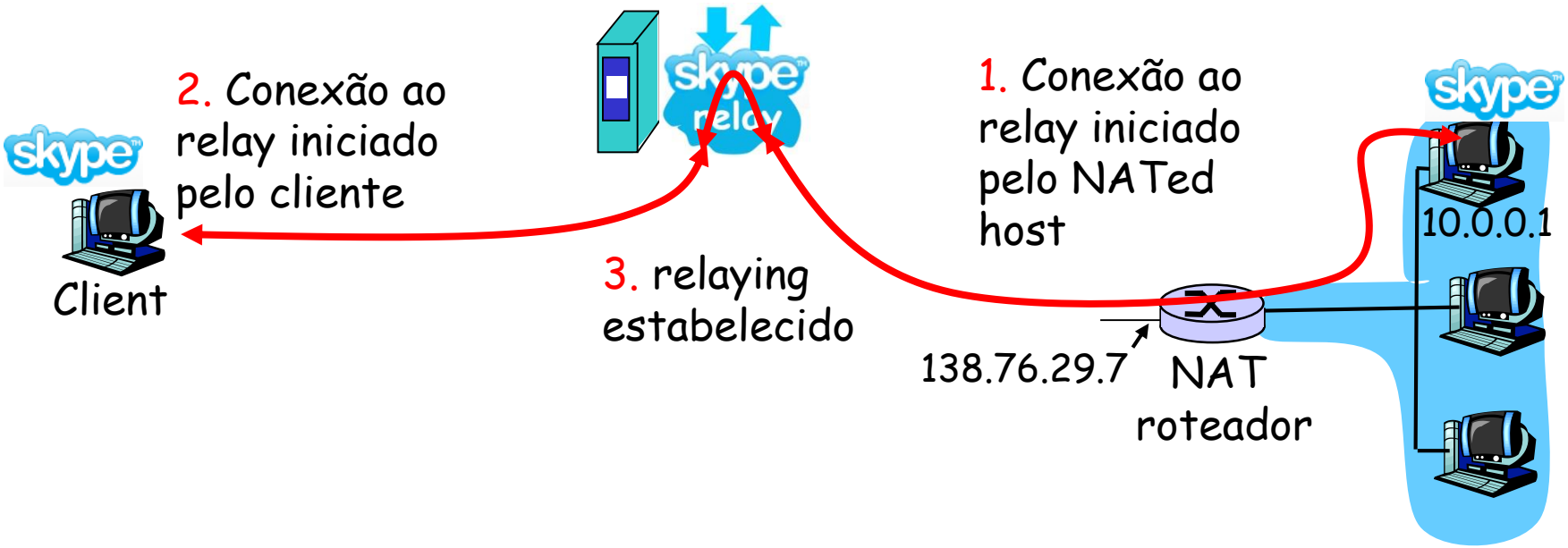
Problema NAT traversal

- Segunda solução: protocolo Universal Plug and Play (UPnP) Internet Gateway Device (IGD) permite NATted host to:
  - Aprende endereço IP publico (138.76.29.7)
  - Adiciona/remove mapeamento de porta (por tempo determinado) i.e., automatiza mapeamento de configuração de porta NAT



Problema NAT

- solução 3: relaying (usado no Skype)
  - ✓ Cliente NATed estabelece conexão com relay
  - ✓ Cliente externo conecta com relay
  - ✓ Relayinterconecta pacotes entre conexões



Roteiro

- 4. 1 Introdução
- 4.2 Circuitos virtuais x datagrama
- 4.3 Como é um roteador
- 4.4 Protocolo IP
  - Formato datagarma
  - endereçamento IPv4
  - ICMP
  - IPv6
- 4.5 Algoritmos de roteamento
  - Estado de enlace
  - Vetor distância
  - Roteamento hierarquico
- 4.6 Roteamento na Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Roteamento Broadcast e multicast

ICMP: Internet Control Message Protocol

- usado por estações, roteadores para comunicar informação s/ camada de rede
  - ✓ relatar erros: estação, rede, porta, protocolo inalcançáveis
  - ✓ pedido/resposta de eco (usado por ping)
- camada de rede "acima de" IP:
  - ✓ msgs ICMP transportadas em datagramas IP
- mensagem ICMP: tipo, código mais primeiros 8 bytes do datagrama IP causando erro

Tipo	Código	descrição
0	0	resposta de eco (ping)
3	0	rede dest. inalcançável
3	1	estação dest inalcançável
3	2	protocolo dest inalcançável
3	3	porta dest inalcançável
3	6	rede dest desconhecida
3	7	estação dest desconhecida
4	0	abaixar fonte (controle de congestionamento - ñ usado)
8	0	pedido eco (ping)
9	0	anúncio de rota
10	0	descobrir roteador
11	0	TTL (sobrevida) expirada
12	0	erro de cabeçalho IP

ICMP eTraceroute

- Fonte UDP envia uma série de segmentos
  - ✓ Primeiro TTL = 1
  - ✓ Segundo TTL=2, etc.
- Quando n-ésimo datagrama chega no nésimo roteador :
  - ✓ Roteador descarta datagrama
  - ✓ Envia para a fonte um datagrama ICMP (type 11, code 0)
  - ✓ Mensagem inclui nome de roteador e endereço IP
- Quando mensagem ICMP chega, finte calcula RTT
- Traceroute faz isso três vezes

Critério de parada

- Segmento UDP eventualmente chega ao destinatário
- Destinatário ICMP retorna mensagem "host unreachable" (type 3, code 3)
- Quando fonte recebe mensagem, ICMP para.

Roteiro

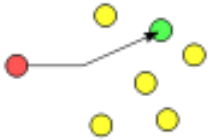
- 4. 1 Introdução
- 4.2 Circuitos virtuais x datagrama
- 4.3 Como é um roteador
- 4.4 Protocolo IP
  - Formato datagarma
  - endereçamento IPv4
  - ICMP
  - IPv6
- 4.5 Algoritmos de roteamento
  - Estado de enlace
  - Vetor distância
  - Roteamento hierarquico
- 4.6 Roteamento na Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Roteamento Broadcast e multicast

IPv6

- Motivação inicial: espaço de endereços de 32-bits completamente alocado até 2008.
- Motivação adicional :
  - ✓ formato do cabeçalho facilita acelerar processamento/re-encaminhamento
  - ✓ mudanças no cabeçalho para facilitar QoS
  - ✓ novo endereço "anycast": rota para o "melhor" de vários servidores replicados
- formato do datagrama IPv6:
  - ✓ cabeçalho de tamanho fixo de 40 bytes
  - ✓ não admite fragmentação

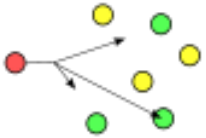
Anycast

- Unicast

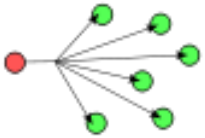


- Multicast

- Anycast



- Broadcast

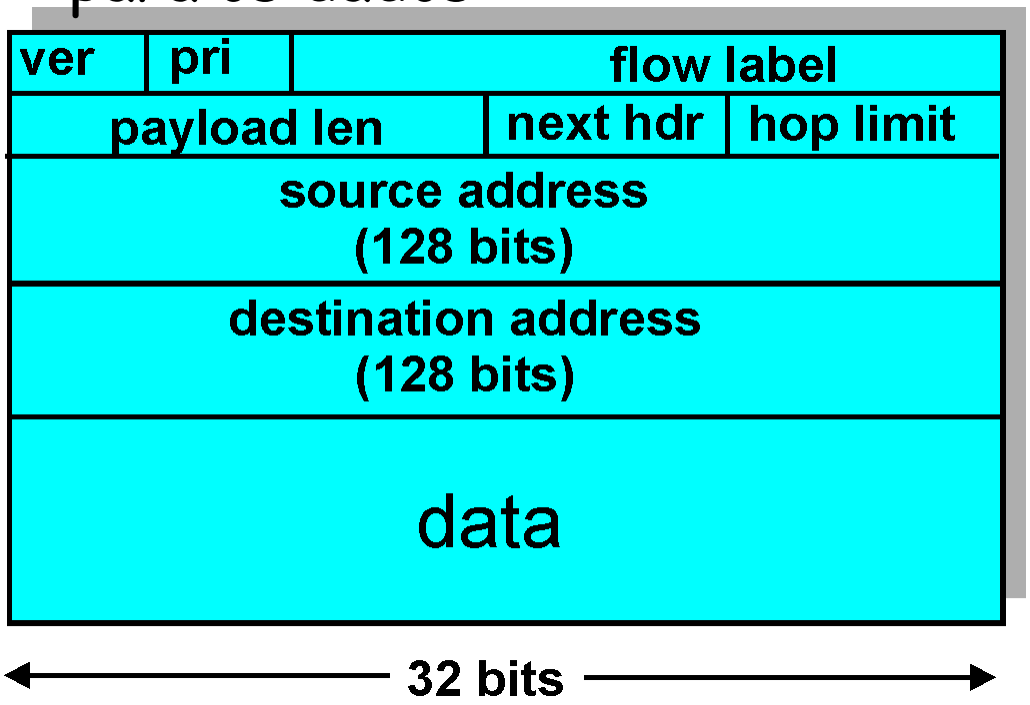


Cabeçalho IPv6

Prioridade: identifica prioridade entre datagramas no fluxo

Rótulo do Fluxo: identifica datagramas no mesmo "fluxo" (conceito de "fluxo" mal definido).

Próximo cabeçalho: identifica protocolo da camada superior para os dados

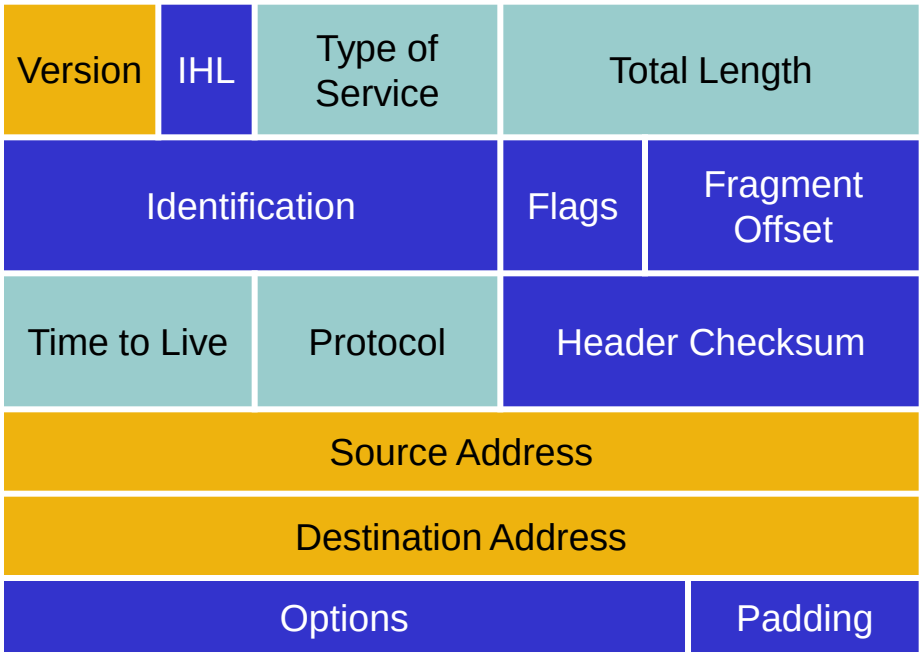


Outras mudanças de IPv4

- *Checksum:* removido completamente para reduzir tempo de processamento a cada roteador
- *Opções:* permitidas, porém fora do cabeçalho, indicadas pelo campo "Próximo Cabeçalho"
- *ICMPv6:* versão nova de ICMP
  - ✓ tipos adicionais de mensagens, p.ex. "Pacote Muito Grande"
  - ✓ funções de gerenciamento de grupo multiponto

Comparação IPv4 and IPv6

IPv4 Header

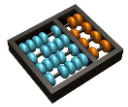


IPv6 Header

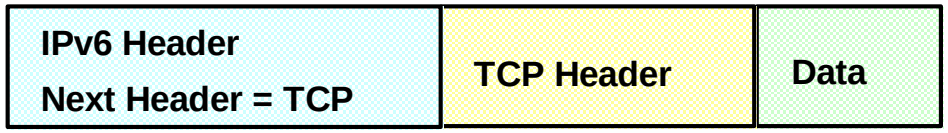
Legend

- Field's name kept from IPv4 to IPv6
- Fields not kept in IPv6
- Name and position changed in IPv6
- New field in IPv6

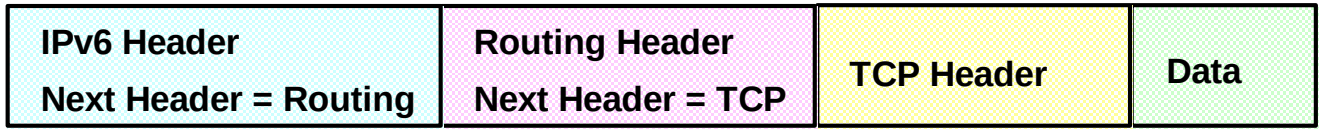
UNICAMP



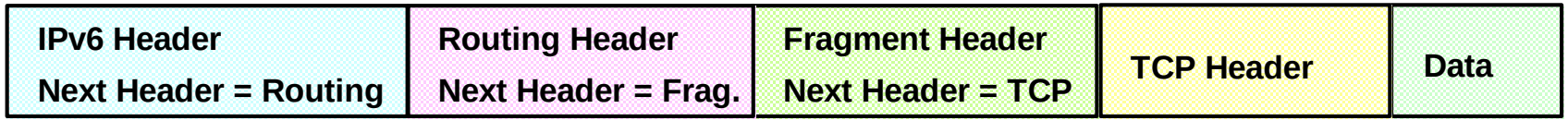
Extensão do cabeçalho - Opções em IPv6



(a) No extension header



(b) IPv6 header followed by a routing header



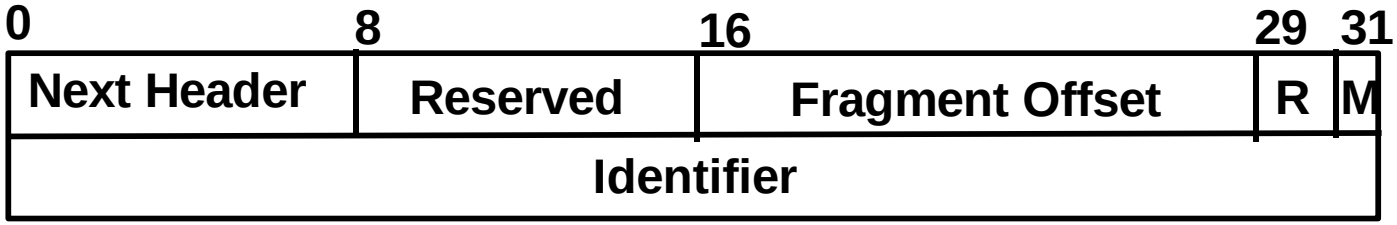
(c) IPv6 header followed by a routing header and a fragment header

Ordem das extensões de cabeçalho

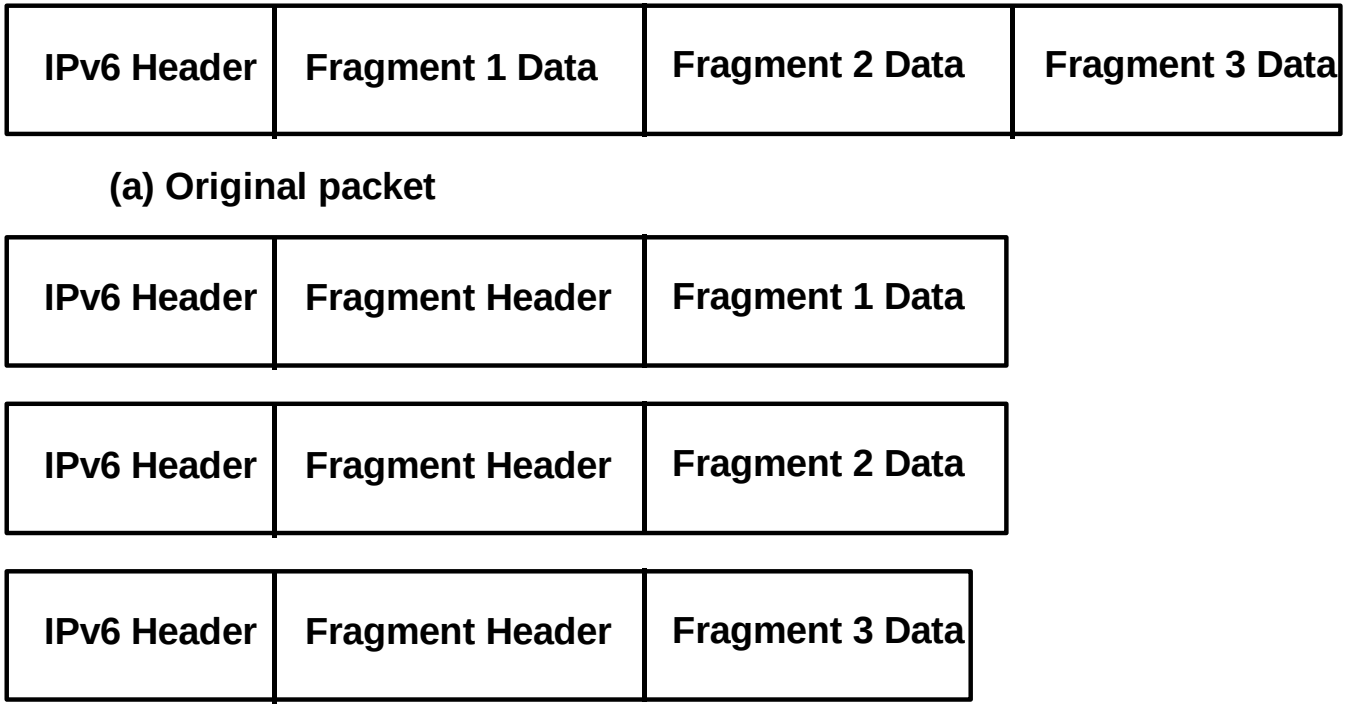
- IPv6 (41)
- Hop-By-Hop Options header (0)
- Destination Options header (60)
- Routing header (43)
- Fragment header (44)
- Authentication header (51)
- Encapsulating Security Payload header (50)
- Destination Options header (60)
- Upper-layer header
  - ICMPv6(58)
  - TCP(6), UDP(17), RSVP(46), SCTP(132)

Cabeçalho de fragmentação

- A Fragmentação só é realizada na fonte e não nos roteadores intermediários
- Campos semelhantes aos do



Exemplo de Fragmentação



Endereço IPv6

- 128 bits
- Notação hexadecimal separada por ":"

3FFD:3600:0000:0000:0302:B3FF:FE3C: C0DB

- Sequência de números de de 16 bits nulos separados por "::"

3FFD:3600:0:0:0:0:1:A =>3FFD:3600::1:A

Endereços reservados

Prefix	Address Type	Portion
0000 0000	Reserved (IPv4 compatibility)	1/256
0000 0001	Unassigned	1/256
0000 001	Reserved for NSAP	1/128
0000 010	Reserved for IPX	1/128
0000 011	Unassigned	1/128
0000 1	Unassigned	1/32
0001	Unassigned	1/16
001	Aggregatable Global Unicast Address	1/8
010	Unassigned	1/8
011	Unassigned	1/8
100	Unassigned	1/8
101	Unassigned	1/8
110	Unassigned	1/8
1110	Unassigned	1/16
1111 0	Unassigned	1/32
1111 10	Unassigned	1/64
1111 110	Unassigned	1/128
1111 1110 0	Unassigned	1/512
1111 1110 10	Link Local Unicast Address	1/1024
1111 1110 11	Site Local Unicast Address	1/1024
1111 1111	Multicast Address	1/256

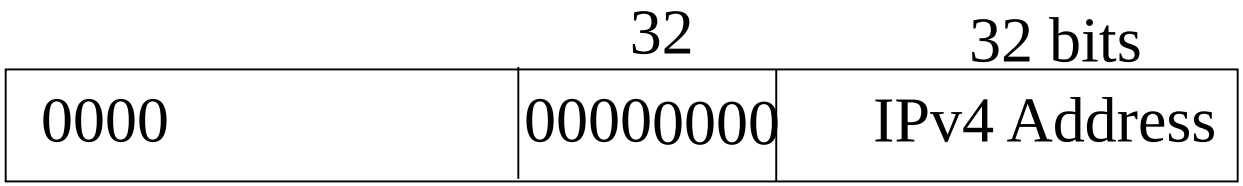
Endereço IPv6

- Unicast
  - IPv4 Compatible Adress
  - Global Unicast Adress
  - Link Local Unicast Adress

- Multicast
  - Começa com 11111111
- Anycast
  - Começa com prefixo de rede seguido de sequência de zeros

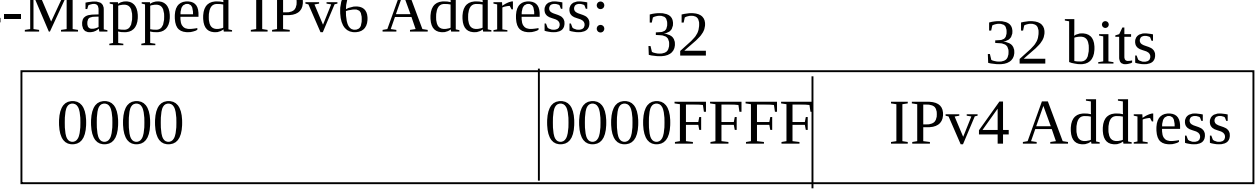
Endereços IPv6 compatíveis com IPv4

IPv4-compatible IPv6 Address: ::8C7B:65A0



::FFFF:8C7B:65A0

IPv4-Mapped IPv6 Address:



Endereços IPv6 Unicast

Unicast Address without Internal Structure:

Node Address

Unicast Address with Subnet:

Subnet Prefix	Interface ID

Unicast Unspecified Address:

0000	0000	0000

Unicast Loopback Address:

0000	0000	0001

ICMPv6

- Versão do protocolo ICMP utilizada pelo protocolo Ipv6, basicamente as mesmas funcionalidades do ICMP
  - ✓ Realizar diagnósticos
  - ✓ Relatar erros de processamento de pacotes
- Possui também funções adicionais
  - ✓ *Descoberta de vizinhança (antes providas pelo protocolo ARP)*
  - ✓ *Gerenciamento de grupos Multicast (antes providas pelo protocolo IGMP).*

Obtenção de endereço IPV6

- Duas opções:
  - Stateless autoconfiguration
  - Stateful DHCP

IPv6 Autoconfiguração

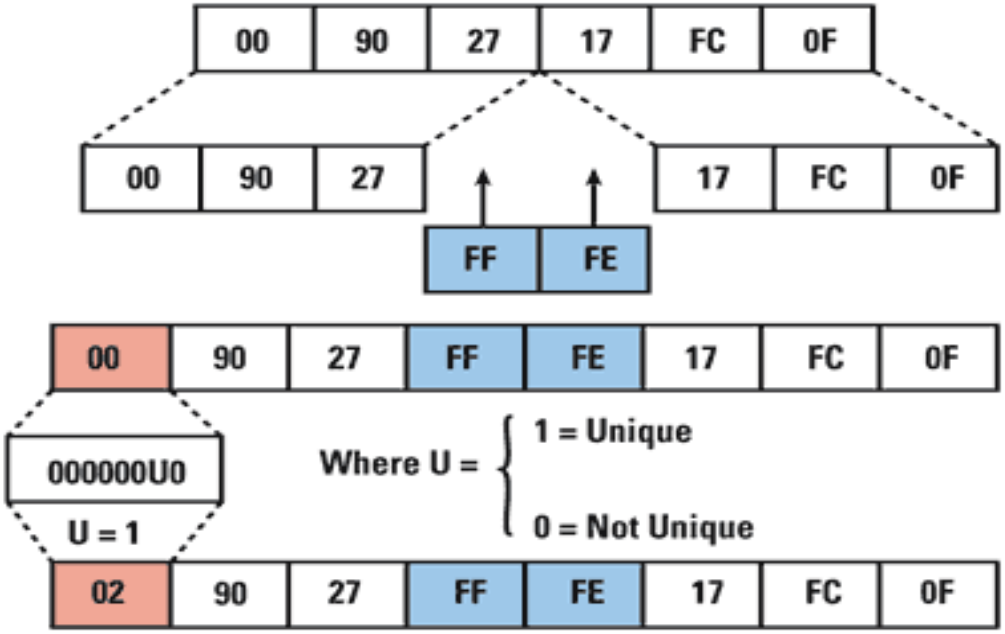
- Pode-se obter endereços IPv6 automaticamente (substitui DHCP IPv4)
- Endereços de enlace-local (link-local) são auto-configuráveis. Devem ser usados somente em enlaces locais (ex redes locais)
- Roteadores e servidores devem ser configurados manualmete

IPv6 Autoconfiguração

- Interface pode ter mais de um endereço Ipv6: um link-local e outro Global
- Endereço: identificador + token
- Identificador obtido através de mensagens Router Advertisement

IPv6 Auto-configuração

- Token pode ser obtido endereço MAC EUI-64
- Alternativa Randon Identifier (temporário)



IPv6 Autoconfiguração

- Após criar endereço em estado tentativa, verifica-se a unicidade do endereço através de envio de mensagem para outras interfaces no link (Dupliate Address Detection)

- Após confirmação endereço usado como permanente, uso só após virar permanente

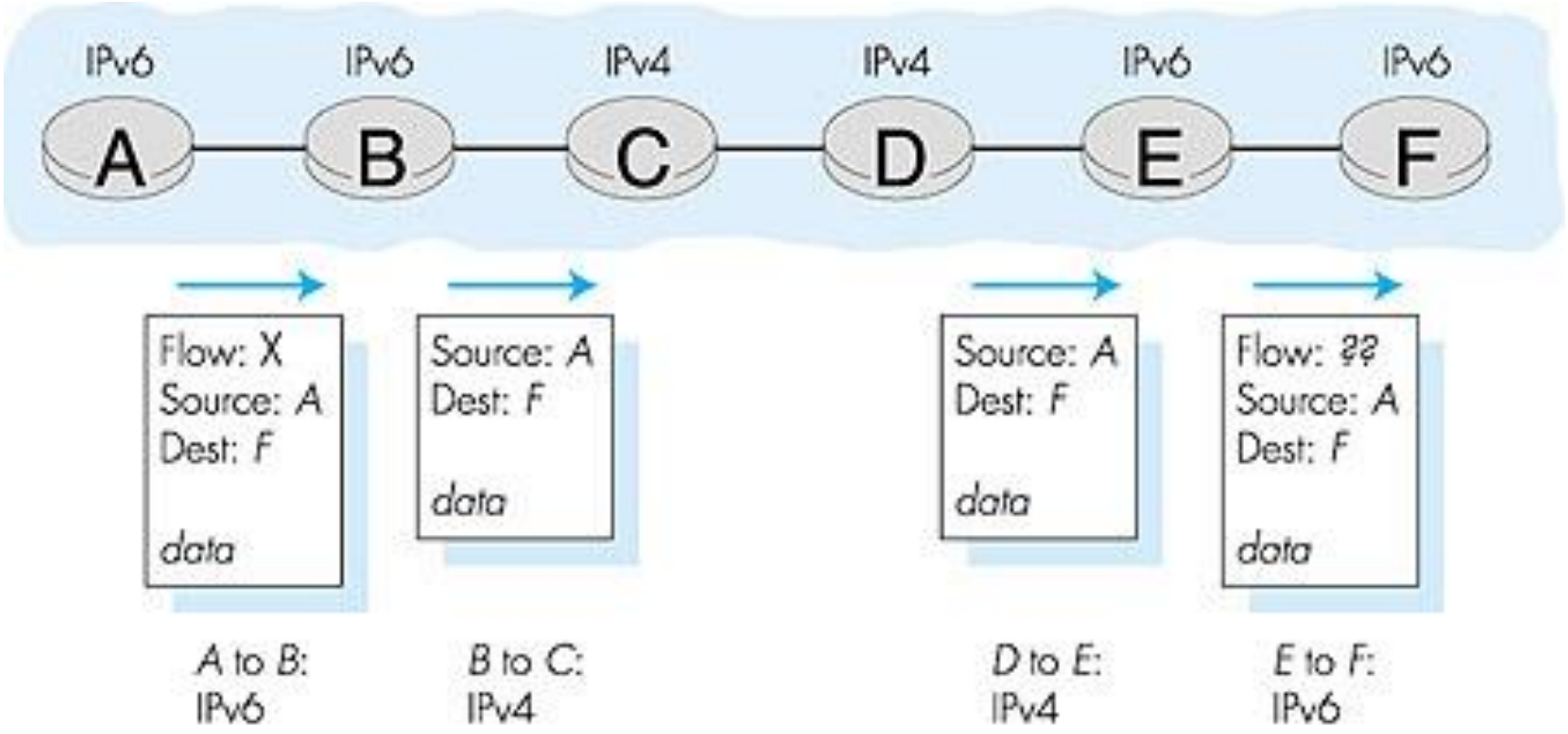
IPv6 DHCPv6

- Semelhante ao DHCP
- Opera em modo cliente – servidor
- Obtém endereço IPv6 e informações de configurações e segurança

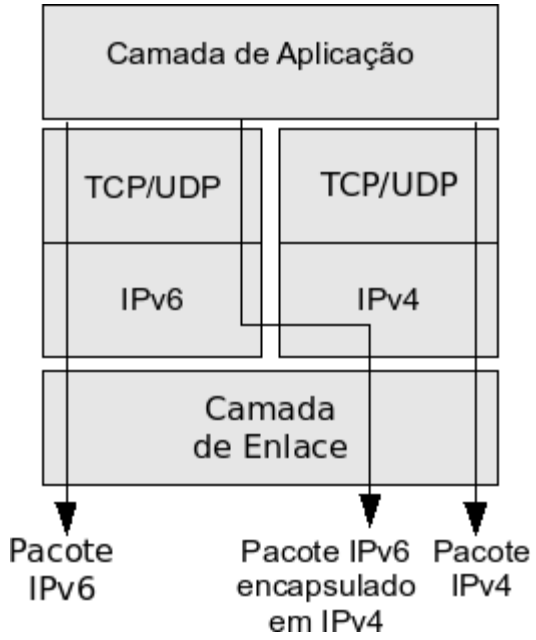
Transição de IPv4 para IPv6

- Nem todos roteadores podem ser atualizados simultaneamente
  - ✓ "dias de mudança geral" inviáveis
  - ✓ Como a rede pode funcionar com uma mistura de roteadores IPv4 e IPv6?
- Três abordagens propostas:
  - ✓ *Pilhas Duais*: alguns roteadores com duas pilhas (v6, v4) podem "traduzir" entre formatos
  - ✓ *Tunelamento*: datagramas IPv6 carregados em datagramas IPv4 entre roteadores IPv4
  - ✓ Tradutores de protocolos

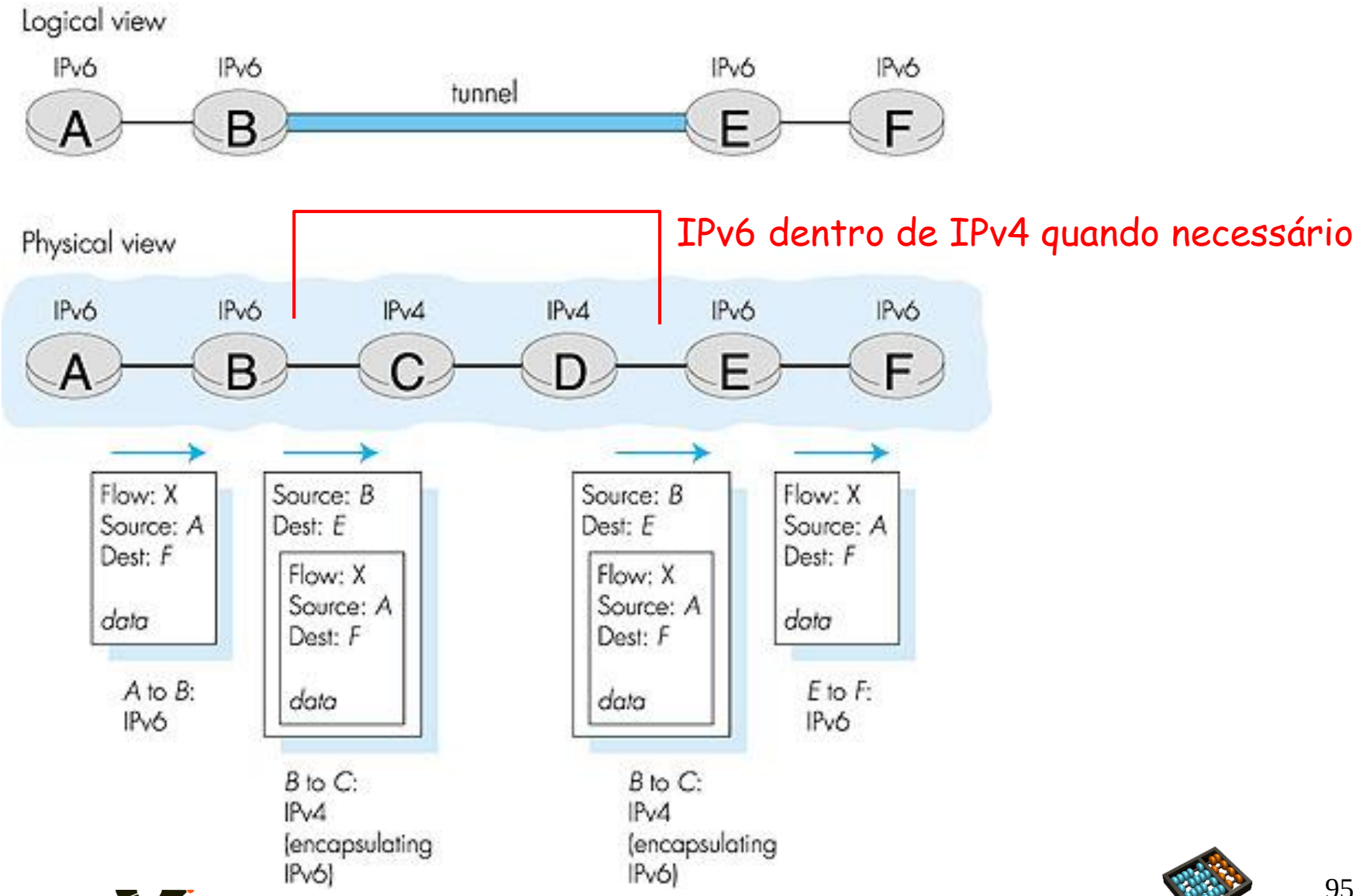
Abordagem de Pilhas Duais



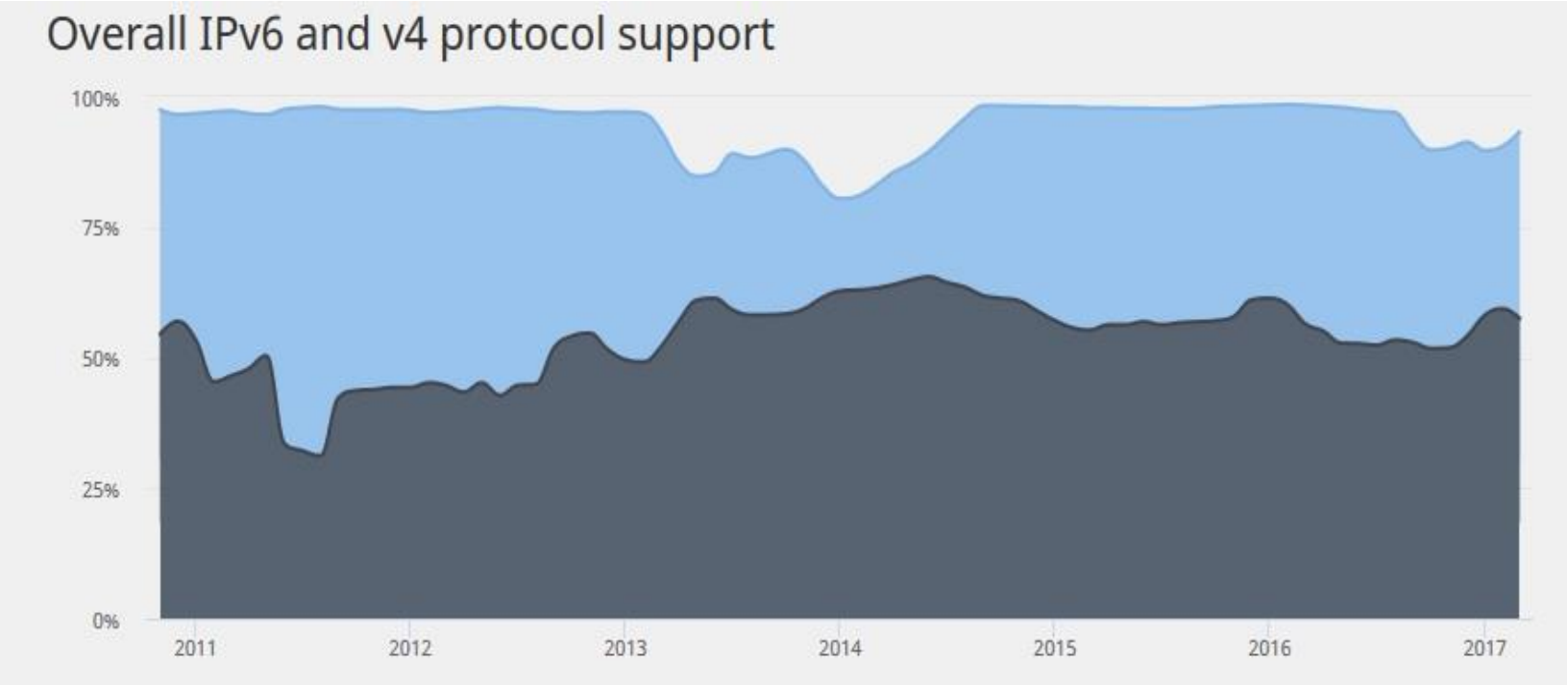
Pilhas Duais



Tunelamento



Estatística IPv6



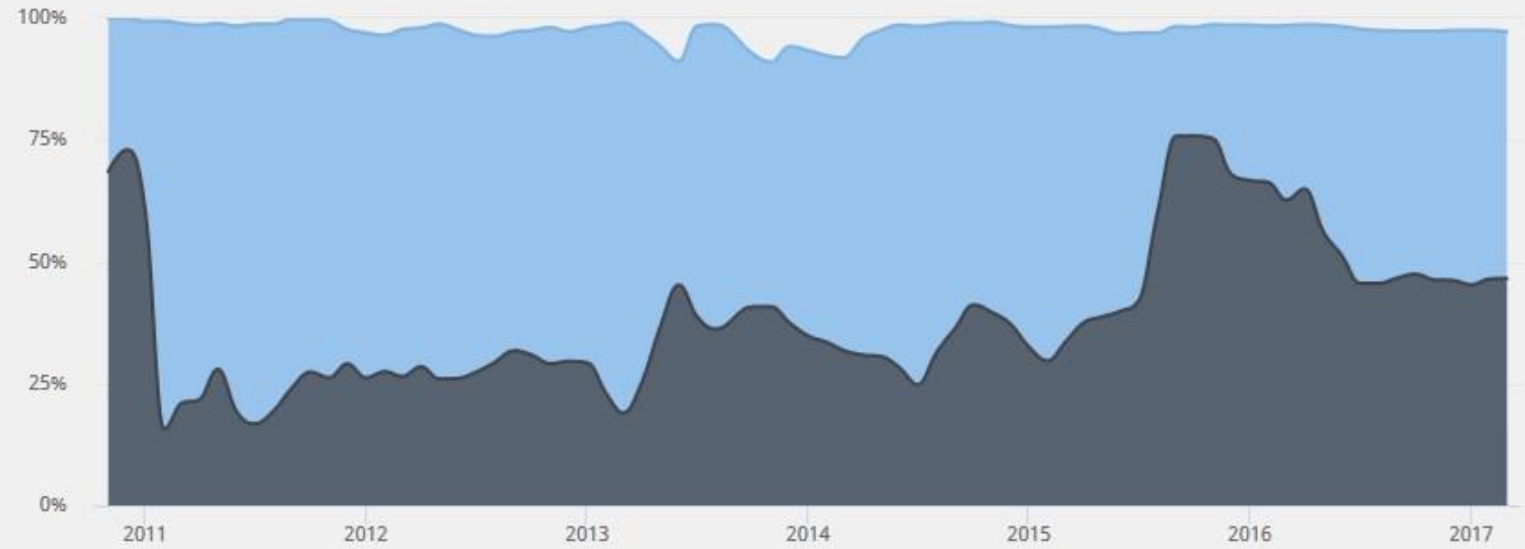
IPv6 na América do Sul

IPv6 in Brazil

These graphs show the evolution of default protocol, v6 address types, and average bandwidth in Brazil over time.

They are generated using the data collected by the ipv6-test.com connection test page, and are updated on a monthly basis.

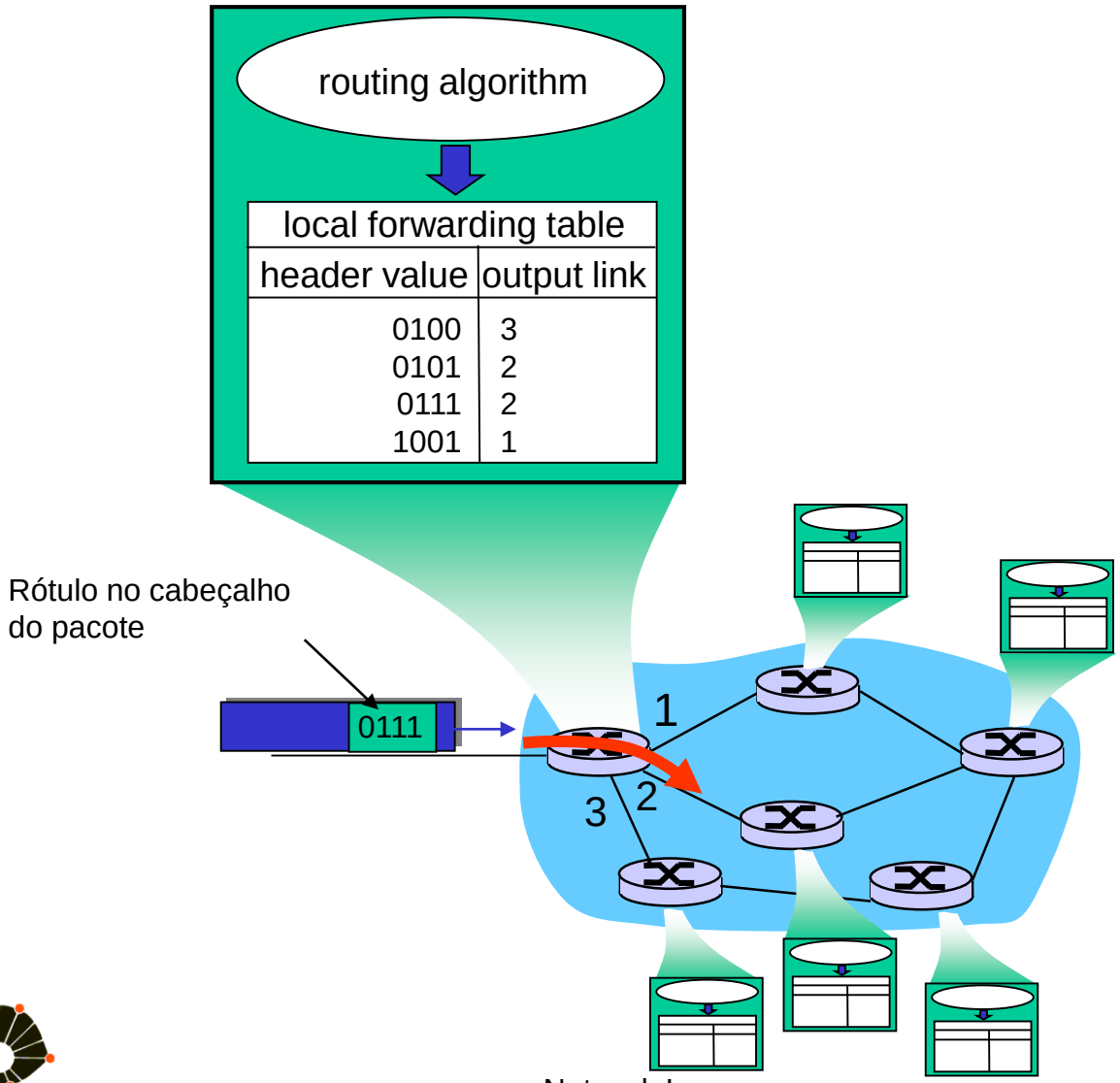
Overall IPv6 and v4 protocol support in Brazil



Roteiro

- 4. 1 Introdução
- 4.2 Circuitos virtuais x datagrama
- 4.3 Como é um roteador
- 4.4 Protocolo IP
  - ✓ Formato datagarma
  - ✓ endereçamento IPv4
  - ✓ ICMP
  - ✓ IPv6
- 4.5 Algoritmos de roteamento
  - ✓ Estado de enlace
  - ✓ Vetor distância
  - ✓ Roteamento hierarquico
- 4.6 Roteamento na Internet
  - ✓ RIP
  - ✓ OSPF
  - ✓ BGP
- 4.7 Roteamento Broadcast e multicast

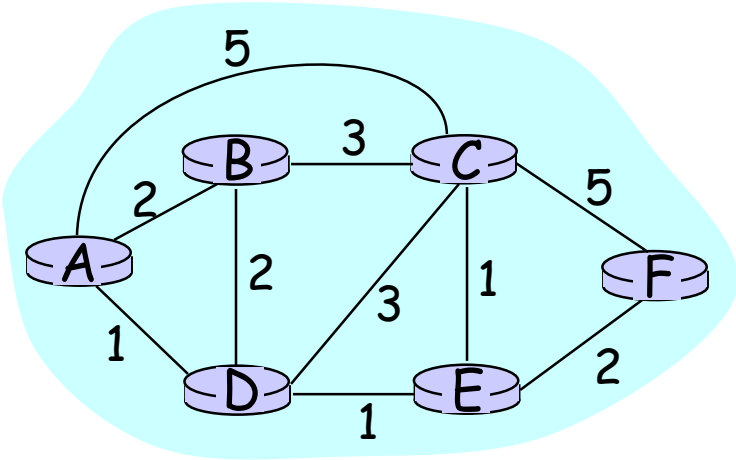
Roteamento


routing algorithm
local forwarding table
header value
output link
0100
3
0101
2
0111
2
1001
1
Rótulo no cabeçalho
do pacote
0111
1
2
3

Roteamento

protocolo de roteamento

meta: determinar caminho (seqüência de roteadores) "bom" pela rede da origem ao destino



Abstração de grafo para algoritmos de roteamento:

- nós do grafo são roteadores
- arestas do grafo são os enlaces físicos
  - custo do enlace: retardo, financeiro, ou nível de congestionamento

- caminho "bom":
  - tipicamente significa caminho de menor custo
  - outras definições são possíveis

Classificação de Algoritmos de Roteamento

Informação global ou descentralizada?

Global:

- todos roteadores têm info. completa de topologia, custos dos enlaces
- algoritmos "estado de enlaces"

Decentralizada:

- roteador conhece vizinhos diretos e custos até eles
- processo iterativo de cálculo, troca de info. com vizinhos
- algoritmos "vetor de distâncias"

Estático ou dinâmico?

Estático:

- rotas mudam lentamente com o tempo

Dinâmico:

- rotas mudam mais rapidamente
  - ✓ atualização periódica
  - ✓ em resposta a mudanças nos custos dos enlaces

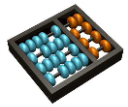
Um algoritmo de roteamento de "estado de enlaces" (EE)

Algoritmo de Dijkstra

- topologia da rede, custos dos enlaces conhecidos por todos os nós
  - ✓ realizado através de "difusão do estado dos enlaces"
  - ✓ todos os nós têm mesma info.
- calcula caminhos de menor custo de um nó ("origem") para todos os demais
  - ✓ gera tabela de rotas para aquele nó
- iterativo: depois de k iterações, sabemos menor custo p/ k destinos

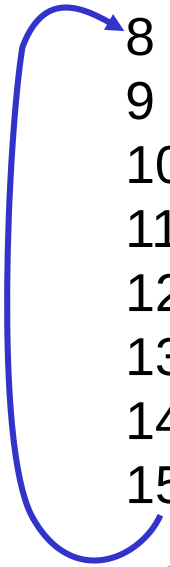
Notação:

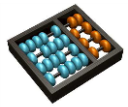
- c(i,j): custo do enlace do nó i ao nó j. custo é infinito se não forem vizinhos diretos
- D(V): valor corrente do custo do caminho da origem ao destino V
- p(V): nó antecessor no caminho da origem ao nó V, imediatamente antes de V
- N: conjunto de nós cujo caminho de menor custo já foi determinado



O algoritmo de Dijkstra

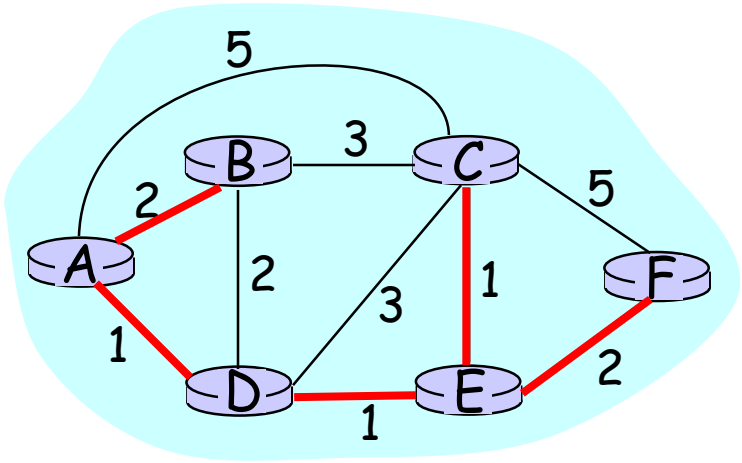
```
1  Inicialização:
2    N = {A}
3    para todos os nós V
4      se V for adjacente ao nó A
5        então D(V) = c(A,V)
6        senão D(V) = infinito
7
8   Repete
9     determina W não contido em N tal que D(W) é o mínimo
10    adiciona W ao conjunto N
11    atualiza D(V) para todo V adjacente ao nó W e ainda não em N:
12       D(V) = min( D(V), D(W) + c(W,V) )
13    /* novo custo ao nó V ou é o custo velho a V ou o custo do
14     menor caminho ao nó W, mais o custo de W a V */
15  até que todos nós estejam em N
```





Algoritmo de Dijkstra: exemplo

Passo	N inicial	D(B),p(B)	D(C),p(C)	D(D),p(D)	D(E),p(E)	D(F),p(F)
0	A	2,A	5,A	1,A	infinito	infinito
1	AD	2,A	4,D		2,D	infinito
2	ADE	2,A	3,E			4,E
3	ADEB		3,E			4,E
4	ADEBC					4,E
5	ADEBCF					



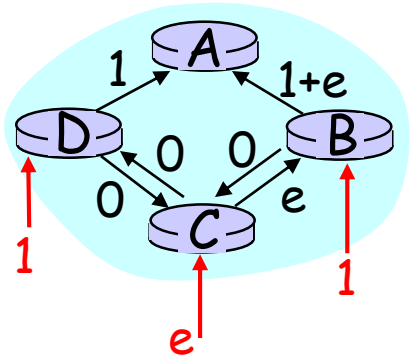
Algoritmo de Dijkstra, discussão

Complexidade algoritmica: n nós

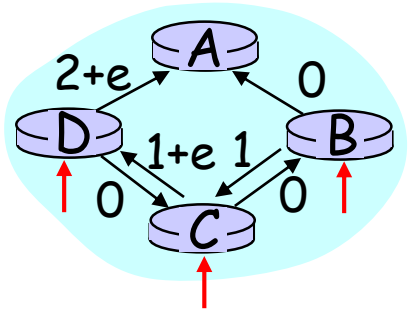
- a cada iteração: precisa checar todos nós, W, não em N
- n*(n+1)/2 comparações => O(n**2)
- implementações mais eficientes possíveis: O(nlogn)

Oscilações possíveis:

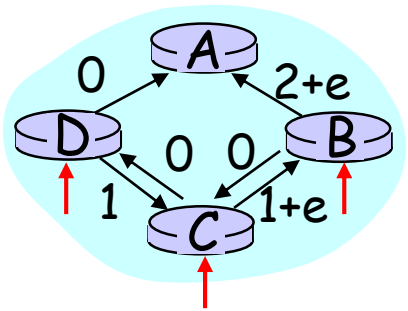
- p.ex., custo do enlace = carga do tráfego carregado



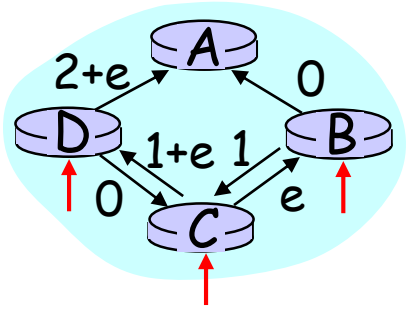
inicialmente



... recalcula rotas



... recalcula



... recalcula

Um algoritmo de roteamento de "vetor de distâncias" (VD)

iterativo:

- continua até que não haja mais troca de info. entre nós
- *se auto-termina*: não há "sinal" para parar

assíncrono:

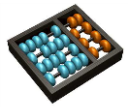
- os nós *não* precisam trocar info./iterar de forma sincronizada!

distribuído:

- cada nó comunica *apenas* com seus vizinhos diretos

Estrutura de dados: Tabela de Distâncias

- cada nós possui sua própria TD
- 1 linha para cada destino possível
- 1 coluna para cada vizinho direto



Algoritmo Vetor Distância

- $D_x(y)$ = estimativa do menor custo de x para y
- Nó x sabe o custo para seu vizinho v $c(x,v)$
- Nó x mantém vetor distância $\mathbf{D_x} = [D_x(y): y \in N]$
- Nó x mantém informação do vetor distância dos seus vizinhos
  - ✓ Para caa vizinho:
    $\mathbf{D_v} = [D_v(y): y \in N]$

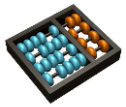
Algoritmo de Vetor Distância

Algoritmo Bellman-Ford

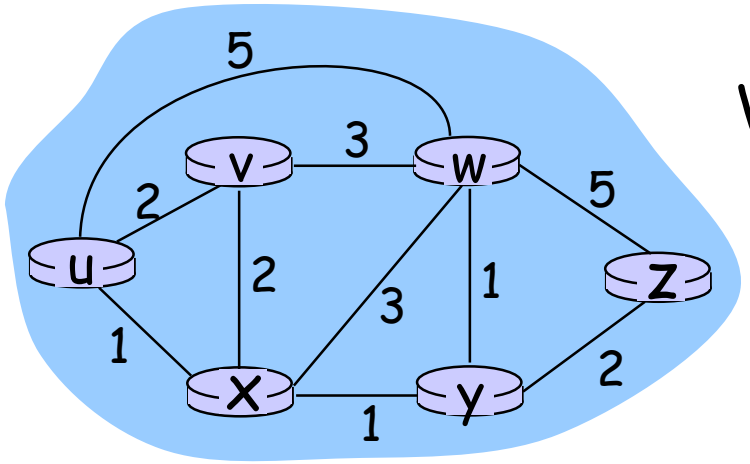
$d_x(y)$:= menor custo de se ir de x para y

$$d_x(y) = \min_v \{c(x,v) + d_v(y)\}$$

Mínimo entre todos os vizinhos



Exemplo



Vê-se que: $d_v(z) = 5$, $d_x(z) = 3$, $d_w(z) =$

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

Tabela de distâncias gera tabela de rotas

$D^E()$

destino \ custo ao destino via	A	B	D
A	1	14	5
B	7	8	5
C	6	9	4
D	4	11	2

destino	enlace de saída a usar, custo
A	A,1
B	D,5
C	D,4
D	D,4

Tabela de distâncias → Tabela de rotas

Roteamento vetor de distâncias: sumário

Iterativo, assíncrono: cada iteração local causada por:

- mudança do custo do enlace local
- mensagem do vizinho: mudança de caminho de menor custo para algum destino

Distribuído:

- cada nó avisa a seus vizinhos *apenas* quando muda seu caminho de menor custo para qualquer destino
  - ✓ os vizinhos então avisam a seus vizinhos, se for necessário

Cada nó:

espera (mudança no custo de mensagem do vizinho)

↓

recalcula tabela de distâncias

↓

se mudou o caminho de menor custo para qq. destino, *avisa* vizinhos

Algoritmo Vetor de Distâncias:

Em todos nós, X:

1 Inicialização:
2 para todos nós adjacentes V:
3 $D^X(*,V) = infinito$ /* o operador * significa "para todas linhas" */
4 $D^X(V,V) = c(X,V)$
5 para todos destinos, Y
6 envia $\min_w D^X(Y,W)$ para cada vizinho /* W indica vizinhos de X */

Algoritmo Vetor de Distâncias (cont.):

8 **repete**
9 **espera** (até observar mudança de custo do enlace ao vizinho V,
10 ou até receber atualização do vizinho V)
11
12 **se** (c(X,V) muda por d unidades)
13 /* altera custo para todos destinos através do vizinho V por d */
14 /* note: d pode ser positivo ou negativo */
15 para todos destinos Y: $D^X(Y,V) = D^X(Y,V) + d$
16
17 **senão, se** (atualização recebido de V para destino Y)
18 /* mudou o menor caminho de V para algum Y */
19 /* V enviou um novo valor para seu $\text{mín}_w D^V(Y,w)$ */
20 /* chamamos este novo valor de "val_novo" */
21 para apenas o destino Y: $D^X(Y,V) = c(X,V) + \text{val\_novo}$
22
23 **se** temos um novo $\text{mín}_w D^X(Y,W)$ para qq destino Y
24 envia novo valor de $\text{mín}_w D^X(Y,W)$ para todos vizinhos
25
26 **para sempre**

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

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

Tabela nó x

Custo para

De	x	y	z
x	0	2	7
y	∞	∞	∞
z	∞	∞	∞

Custo para

De	x	y	z
x	0	2	3
y	2	0	1
z	7	1	0

Tabela nó y node

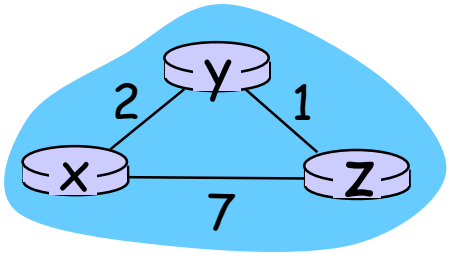
cost to

De	x	y	z
x	∞	∞	∞
y	2	0	1
z	∞	∞	∞

Tabela nó z

cost to

De	x	y	z
x	∞	∞	∞
y	∞	∞	∞
z	7	1	0



time

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

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

Tabela nó x

Custo para

De	x	y	z
x	0	2	7
y	∞	∞	∞
z	∞	∞	∞

De	x	y	z
x	0	2	3
y	2	0	1
z	7	1	0

De	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

Tabela nó y

Custo para

De	x	y	z
x	∞	∞	∞
y	2	0	1
z	∞	∞	∞

De	x	y	z
x	0	2	7
y	2	0	1
z	7	1	0

De	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

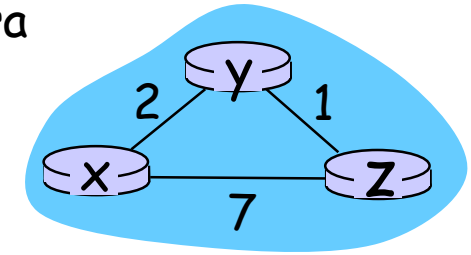
Tabela nó z

Custo para

De	x	y	z
x	∞	∞	∞
y	∞	∞	∞
z	7	1	0

De	x	y	z
x	0	2	7
y	2	0	1
z	3	1	0

De	x	y	z
x	0	2	3
y	2	0	1
z	3	1	0

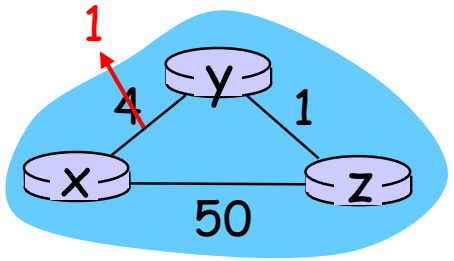


time

Dimuição no custo de enlace

Mudanca no estado do enlace:

- Nó detecta mudanca no custo do enlace
- Atualiza informação de roteamento, recalcula vetor
- Se distâncias alterada, notifica vizinhos



"Boas notícias Caminham rápido"

No tempo t_0, y detecta mudanca no custo e notifica vizinho

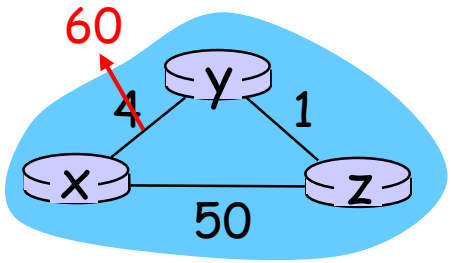
No tempo t_1, z recebe atualização de y e atualiza sua tabela , Computa um novo custo para x e envia para seus vizinhos a sua VD

No tempo t_2, y recebe a atualização de z e atualiza a sua tabela de distância, o menor custo de y não se altera e consequentemente não envia nenhuma mensagem para z

Aumento do custo de enlace

Mudanca de custo no enlace:

- Notícias ruins demoram a propagar, problema de "contagem até
- 44 interações até estabilizar



Reverso envenenado:

- Se Z rotea através de Y para chegar a X :
  - ✓ Z diz a Y que sua distância a x é infinita, de tal forma que Y não vai rotear através de Z

Problema de Convergência

- Soluções:
  - ✓ Horizonte dividido (Split horizon)
    - Anuncio de rotas para vizinho não deve conter rotas aprendidads por anuncios do próprio vizinho
  - ✓ Reverso envenenado (Poisson reverse)
    - Anuncia custo infinito para vizinho que faz parte do loop.
  - ✓ Temporização de retenção (Hold down timer)
    - Retem a informação de menor custo por um tempo igual ao período pré-estabelecido de temporização
    - Solução para loop envolvendo mais de dois
    - Aumenta tempo de convergência das atualizações das tabelas

Comparação dos algoritmos EE e VD

Complexidade de mensagens

- EE: com n nós, E enlaces, O(nE) mensagens enviadas
- VD: trocar mensagens apenas entre vizinhos
  - ✓ varia o tempo de convergência

Rapidez de Convergência

- EE: algoritmo O(n**2) requer O(nE) mensagens
  - ✓ podem ocorrer oscilações
- VD: varia tempo para convergir
  - ✓ podem ocorrer rotas cíclicas
  - ✓ problema de contagem ao infinito

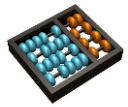
Robustez: o que acontece se houver falha do roteador?

EE:

- ✓ nó pode anunciar valores incorretos de custo de *enlace*
- ✓ cada nó calcula sua *própria* tabela

VD:

- ✓ um nó VD pode anunciar um custo de *caminho* incorreto
- ✓ a tabela de cada nó é usada pelos outros nós
  - erros se propagam pela rede



Roteamento Hierárquico

Neste estudo de roteamento fizemos uma idealização:

- todos os roteadores idênticos
- rede "não hierarquizada" ("flat")

… *não* é verdade, na prática

escala: 200 milhões de destinos:

- impossível guardar todos destinos na tabela de rotas!
- troca de tabelas de rotas afogaria os enlaces!

autonomia administrativa

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

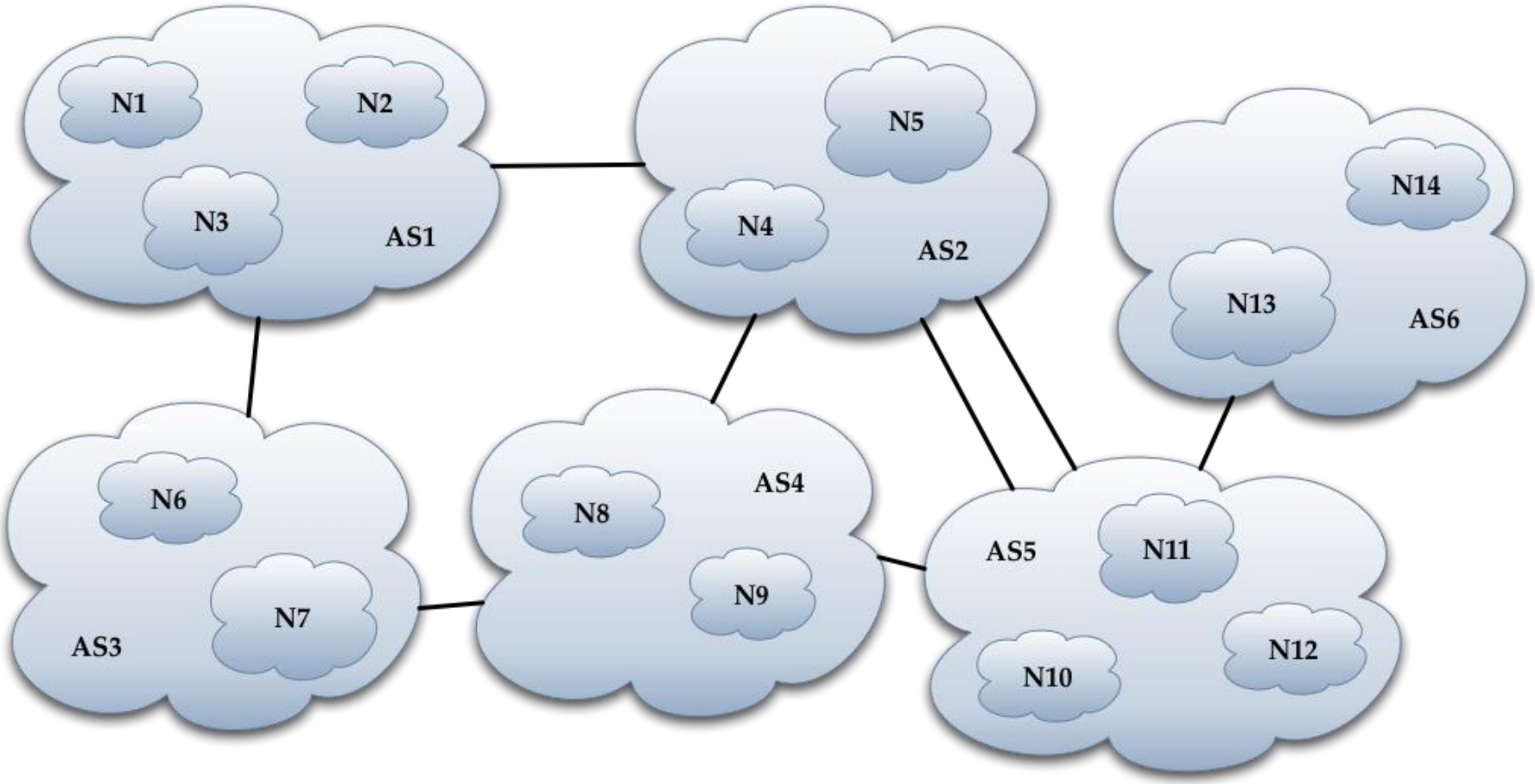
Roteamento Hierárquico

- agregar roteadores em regiões, "sistemas autônomos" (SAs)
- roteadores no mesmo SA usam o mesmo protocolo de roteamento
  - protocolo de roteamento "intra-SA"
  - roteadores em SAs diferentes podem usar diferentes protocolos de roteamento intra-SA

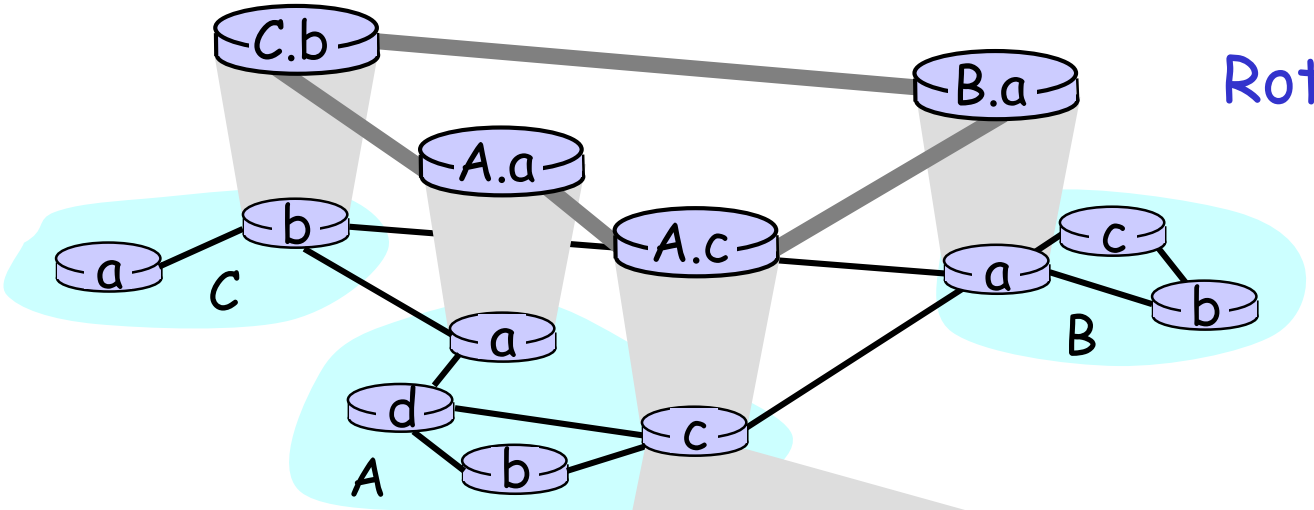
roteadores de borda

- roteadores especiais no SA
- usam protocolo de roteamento intra-SA com todos os demais roteadores no SA
- *também* responsáveis por rotear para destinos fora do SA
  - usam protocolo de roteamento "inter-SA" com outros roteadores de borda

Visão abstrata da Internet



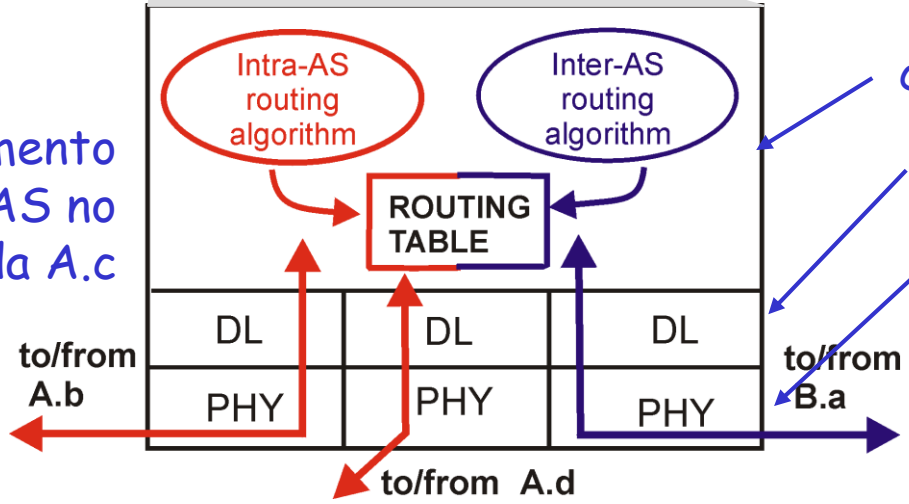
Roteamento Intra-SA e Inter-SA



Roteadores de borda:

- fazem roteamento inter-SA entre si
- fazem roteamento intra-SA com outros roteadores do seu próprio SA

Roteamento inter-AS, intra-AS no roteador de borda A.c

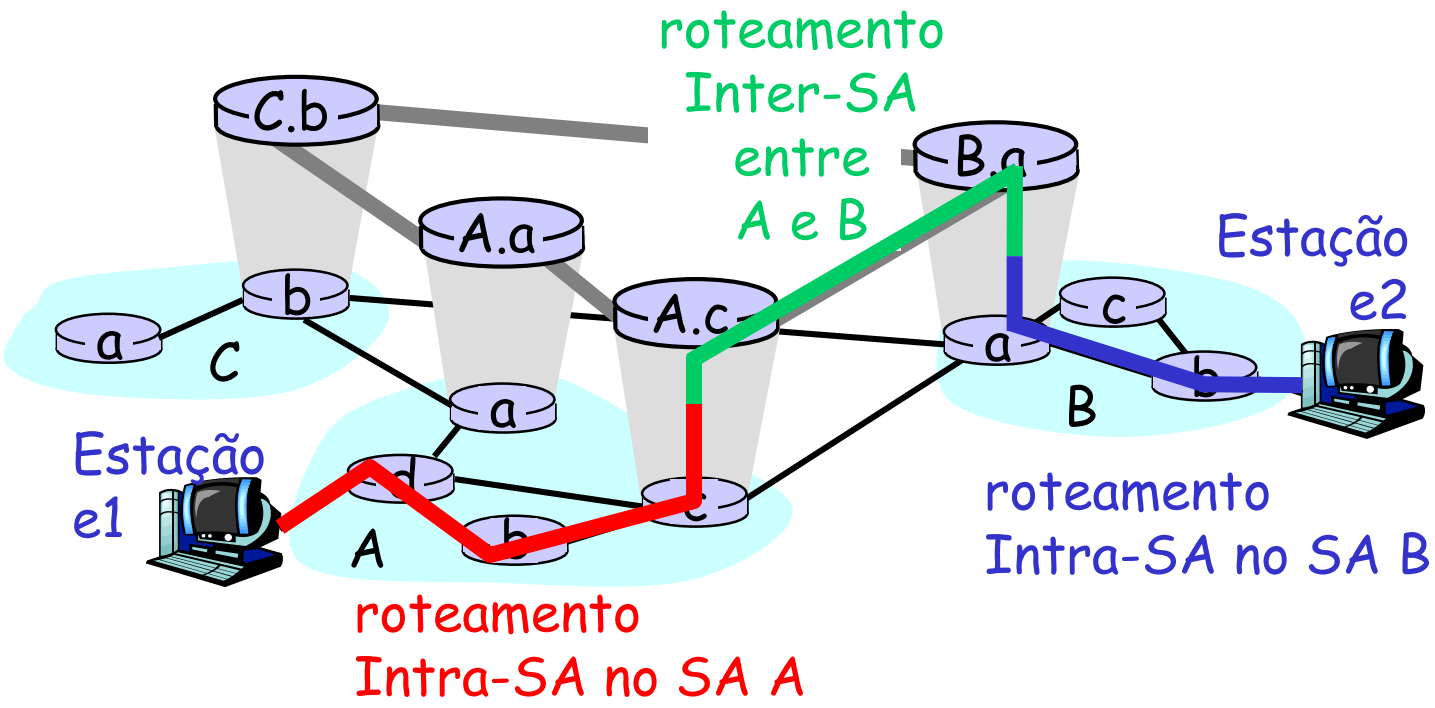


camada de rede

camada de enlace

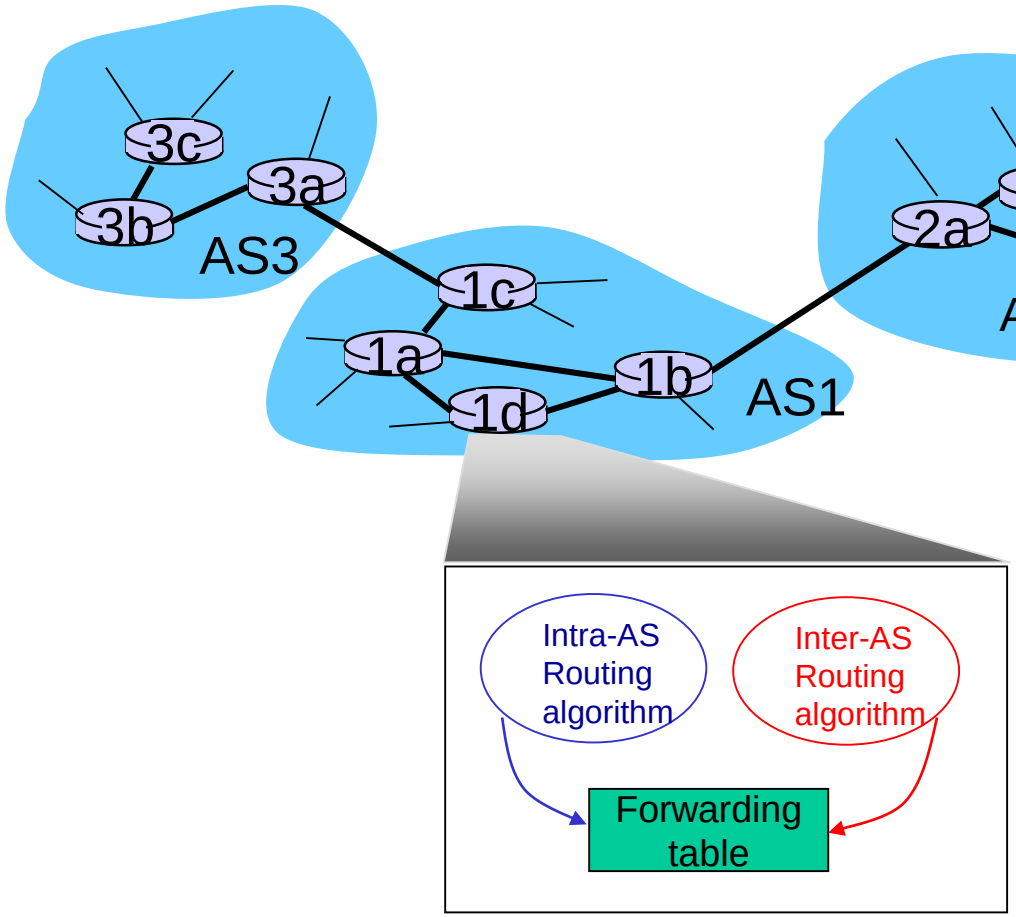
camada física

Roteamento Intra-SA e Inter-SA



- Em breve veremos protocolos de roteamento inter-SA e intra-SA específicos da Internet

Interconnected ASes



- forwarding table configured by both intra- and inter-AS routing algorithm
  - intra-AS sets entries for internal dests
  - inter-AS & intra-AS sets entries for external dests

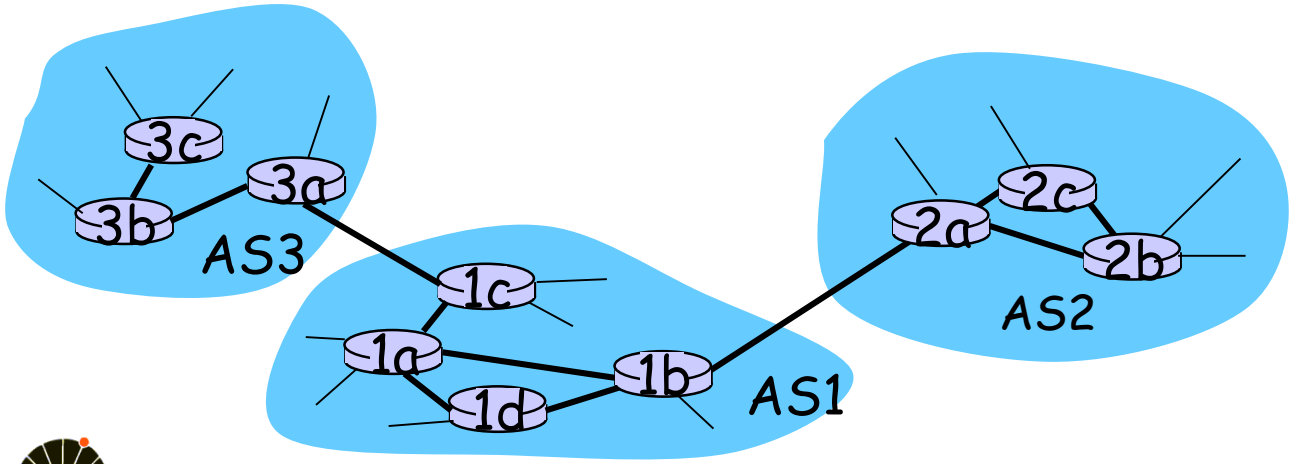
Tarefa Interna SA

- Roteador em AS1 recebe datagrama destinado para roteador em outro :
  - ✓ Para qual roteador deve enviar o pacote?

AS1 deve:

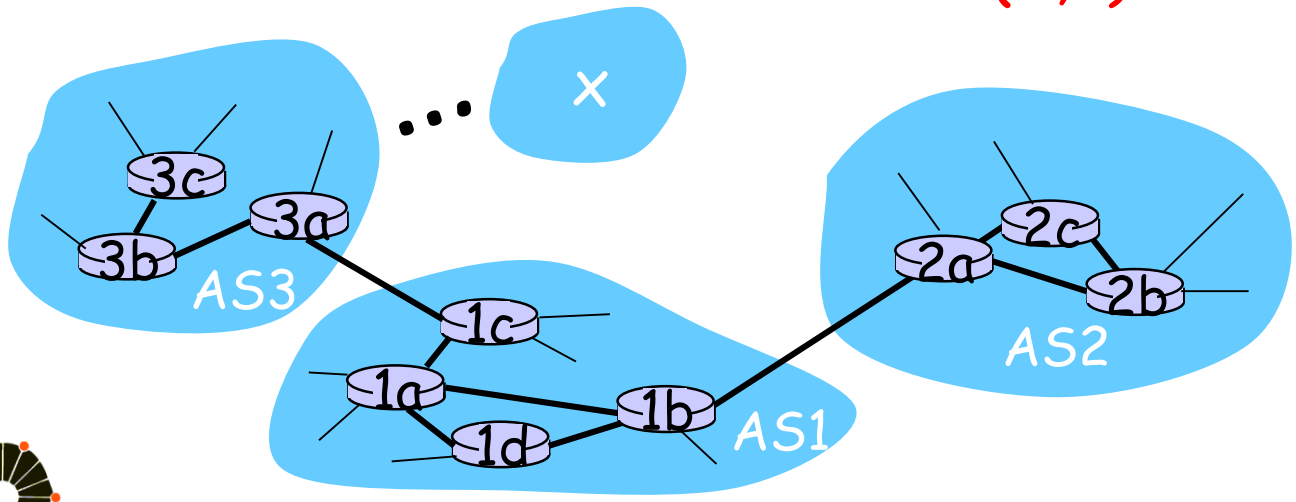
1. Propagar para qual destinatários pode-se chegar através de AS2 e de AS3
2. Propagar informação de alcancabilidade através de AS1

Tarefa de roteamento intra-domínio!



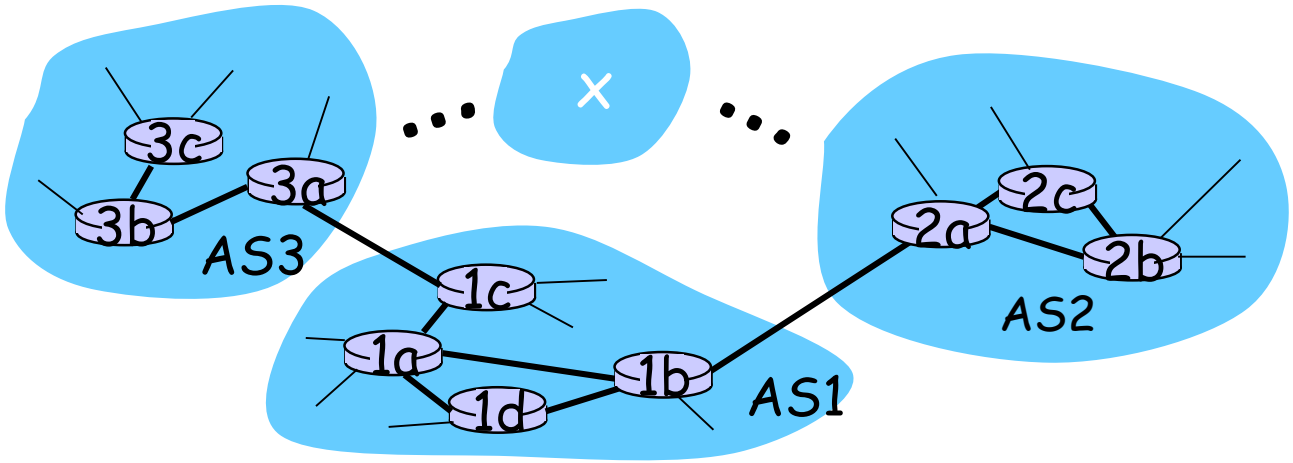
Exemplo: setando tabela de roteamento 1d

- Suponha que SA1 aprendeu (via protocolo inter-AS) que a submet *x* alcancável através de AS3 (gateway 1c) mas não via AS2.
- Protocolo inter-SA propoaga informação de alcancabilidade para todos roteadores internos
- Roteador 1d determina através de informação de roteamento intra-SA qual interface *I* está no menor caminho para 1c.
  - ✓ Instala tabela de encaminhamento *(x,I)*



Exemplo: escolha entre múltiplos SAs

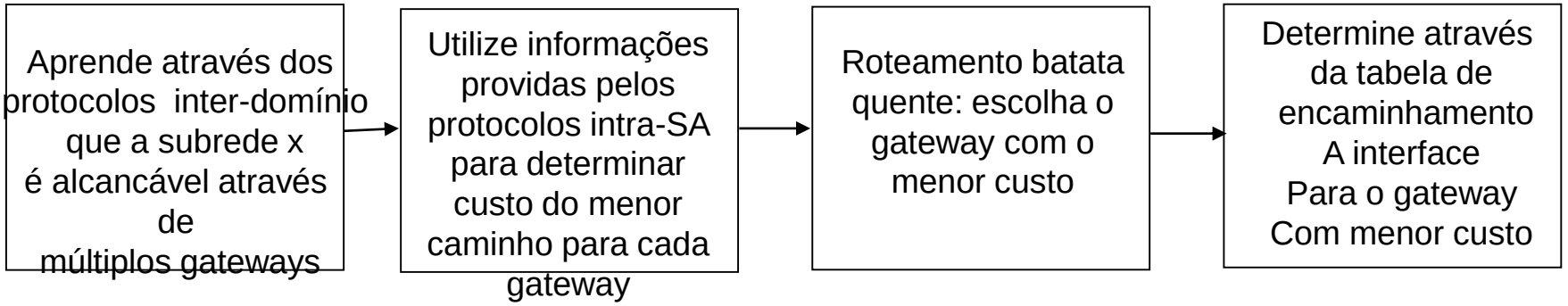
- Suponha agora que AS1 aprende através do protocolo inter-SA que a subrede x é alcancável de AS3 e de AS2.
- Para configyrar a tabela de encamihamento, o roteador Id tem que determinar através de qual gateway ele deve encaminhar pacote ao destinatário x.
  - ✓ Esta também é uma tarefa do protocolo inter-SA



Exemplo: escolha entre múltiplos ASs

- Suponha agora que AS1 aprende através do protocolo inter-domínio que a sub-rede *x* é alcancável de AS3 e deAS2.
- Para configurar a tabela, o roteador 1d deve determinar através de qual gateway, deve enviar os pacotes para o destinatário *x*.
  - ✓ Esta também é uma tarefa do protocolo inter-domínio

- Protocolo batata quente: envia pacote para a roteador mais próximo.



Roteiro

- 4. 1 Introdução
- 4.2 Circuitos virtuais x datagrama
- 4.3 Como é um roteador
- 4.4 Protocolo IP
  - Formato datagarma
  - endereçamento IPv4
  - ICMP
  - IPv6
- 4.5 Algoritmos de roteamento
  - Estado de enlace
  - Vetor distância
  - Roteamento hierarquico
- 4.6 Roteamento na Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Roteamento Broadcast e multicast

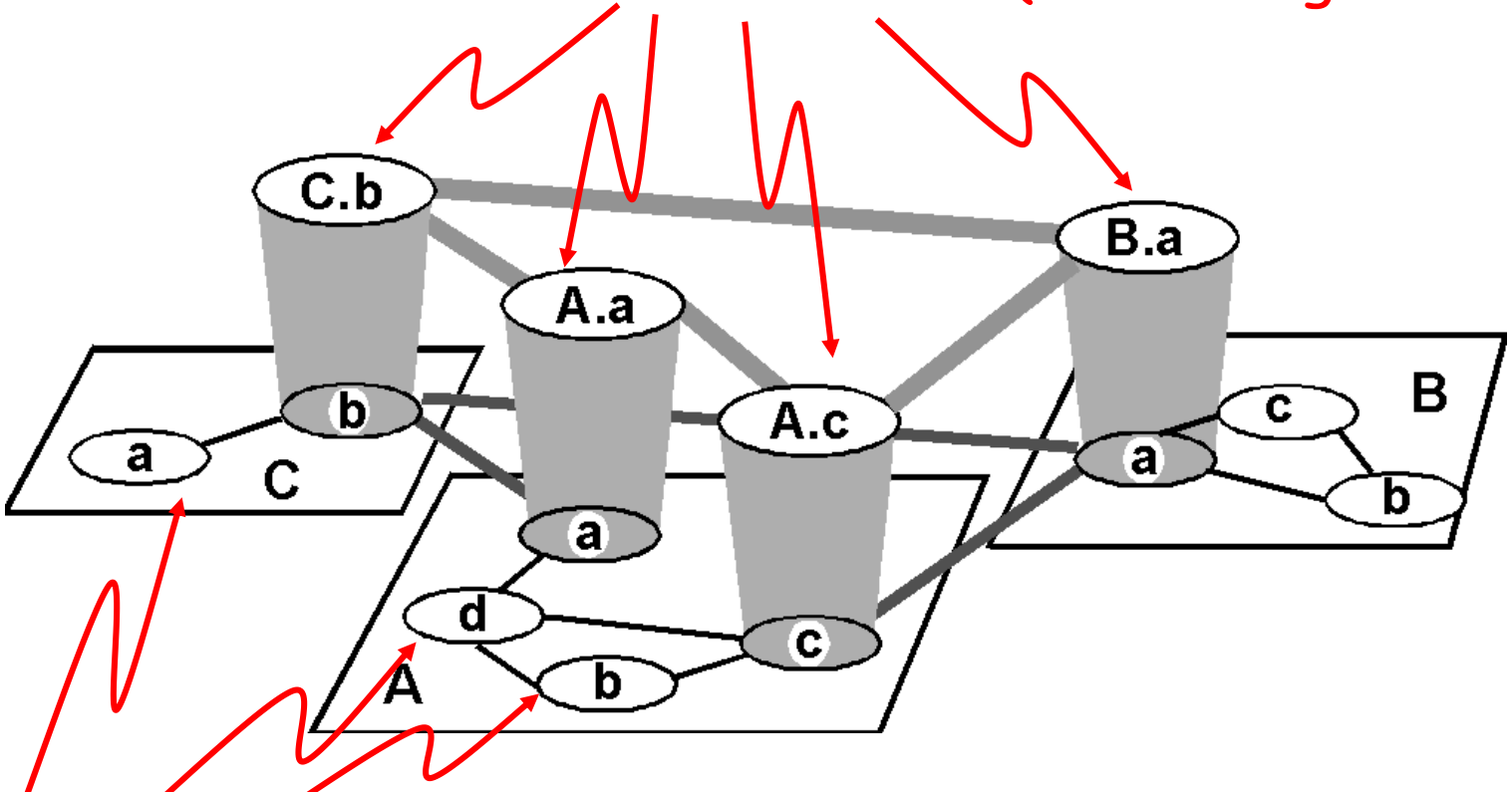
Roteamento na Internet

- A Internet Global consiste de Sistemas Autonômos (SAs) interligados entre si:
  - **SA Folha**: empresa pequena
  - **SA com Múltipla Conectividade**: empresa grande (sem trânsito)
  - **SA de Trânsito**: provedor

- Roteamento em dois níveis:
  - **Intra-SA**: administrador é responsável pela escolha
  - **Inter-SA**: padrão único

Hierarquia de SAs na Internet

Inter-SA: roteadores de fronteira (exterior gateways)



Intra-SA: roteadores internos (interior gateways)

Roteamento Intra-SA

- Também conhecido como Interior Gateway Protocols (IGP) (protocolos de roteamento interno)
- Os IGPs mais comuns são:
  - RIP: *Routing Information Protocol*
  - OSPF: *Open Shortest Path First*
  - IGRP: *Interior Gateway Routing Protocol* (proprietário da Cisco)

RIP (*Routing Information Protocol*)

- Algoritmo do tipo vetor de distâncias
- Incluído na distribuição do BSD-UNIX em 1982
- Métrica de distância: # de enlaces (máx = 15 enlaces)
- Vetores de distâncias: trocados a cada 30 seg via Mensagem de Resposta (tb chamada de **anúncio**)
- Cada anúncio: rotas para 25 redes destino

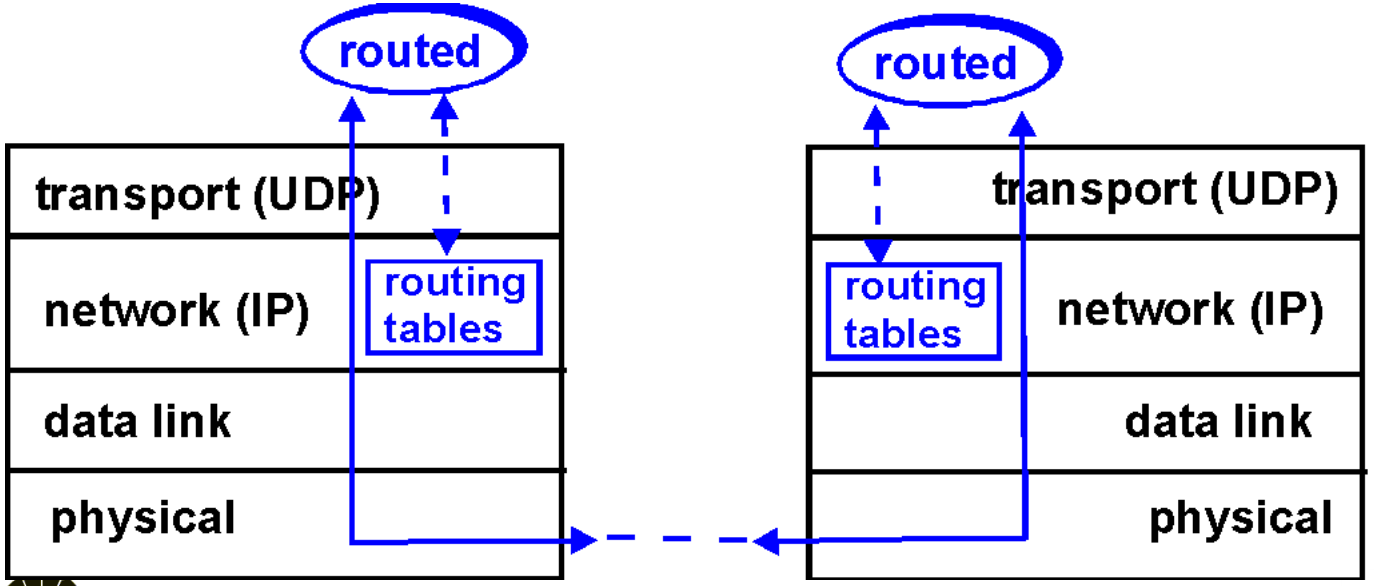
RIP: Falha e Recuperação de Enlaces

Se não for recebido anúncio novo durante 180 seg --> vizinho/enlace declarados mortos

- ✓ rotas via vizinho invalidadas
- ✓ novos anúncios enviados aos vizinhos
- ✓ na sua vez, os vizinhos publicam novos anúncios (se foram alteradas as suas tabelas)
- ✓ informação sobre falha do enlace rapidamente propaga para a rede inteira
- ✓ reverso envenenado usado para impedir rotas cíclicas (ping-pong) (distância infinita = 16 enlaces)

RIP: Processamento de tabelas

- Tabelas de roteamento RIP gerenciadas por processo de **nível de aplicação** chamado routed (routing daemon)
- anúncios enviados em pacotes UDP, repetidos periodicamente



RIP: exemplo de tabela de rotas (cont)

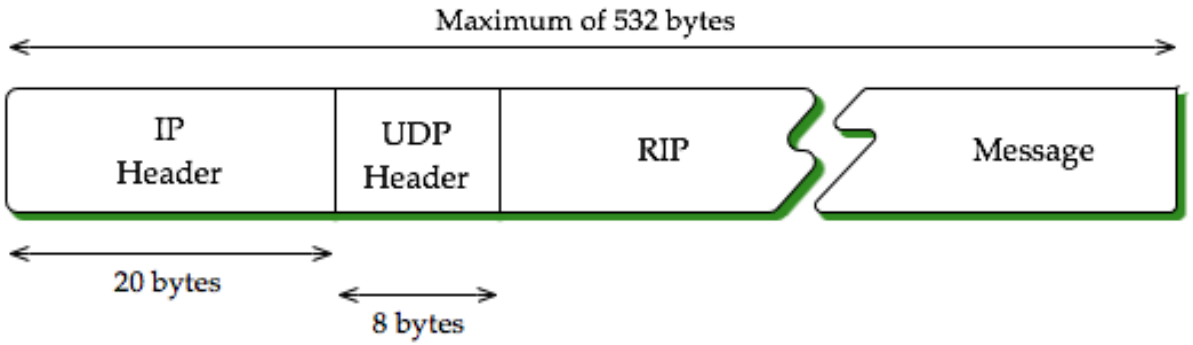
Router: *giroflee.eurocom.fr*

Destination	Gateway	Flags	Ref	Use	Interface
127.0.0.1	127.0.0.1	UH	0	26492	lo0
192.168.2.	192.168.2.5	U	2	13	fa0
193.55.114.	193.55.114.6	U	3	58503	le0
192.168.3.	192.168.3.5	U	2	25	qaa0
224.0.0.0	193.55.114.6	U	3	0	le0
default	193.55.114.129	UG	0	143454	

- Três redes vizinhas diretas da classe C (LANs)
- Roteador apenas sabe das rotas às LANs vizinhas
- Roteador "default" usado para "subir"
- Rota de endereço multiponto: 224.0.0.0
- Interface "loopback" (para depuração)

Formato Pacore RIP

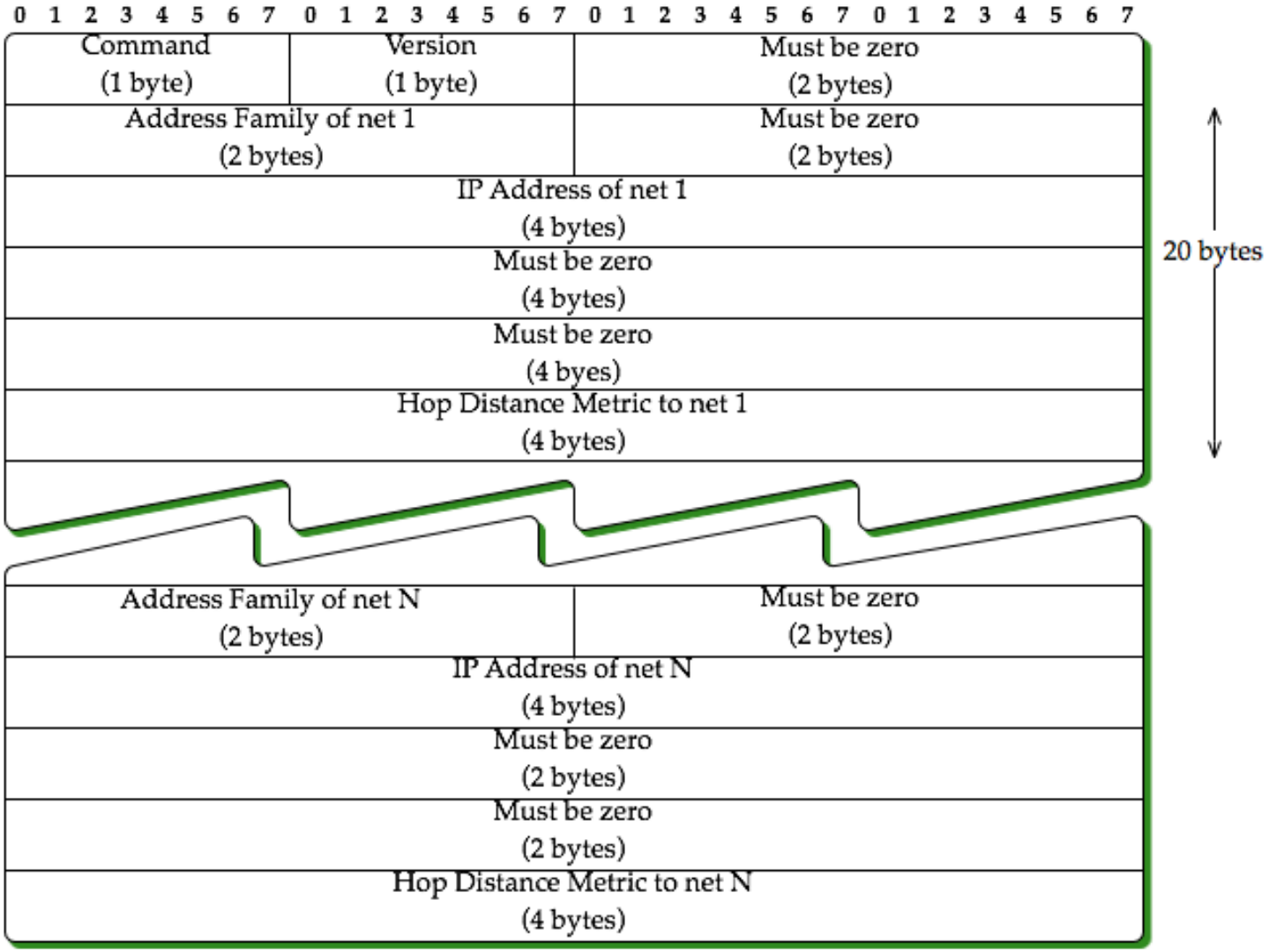
- RIP utiliza UDP

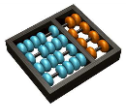


- 4 bytes de header + 20 bytes para cada bloco de endereço (rota)

- Pacote UDP limitação de 512 bytes, pode carregar no máximo 25 rotas

Formato Pacore RIPv1





IGRP (*Interior Gateway Routing Protocol*)

- Proprietário da CISCO; sucessor do RIP (anos 80)
- Vetor de Distâncias, como RIP
- Diversas métricas de custo (retardo, largura de banda, confiabilidade, carga, etc)
- usa TCP para trocar mudanças de rotas
- Roteamento sem ciclos via *Distributed Updating Algorithm* (DUAL) baseado em *computação difusa*

Métricas de Composta

- (E) IGRP utiliza métrica composta com parcelas relativas a: atraso, banda passante, carga na rede e confiabilidade compesos diferentes

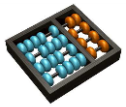
$$C = \begin{cases} (K_1 \times B + K_2 \times \frac{B}{256-L} + K_3 \times D) \times \left(\frac{K_5}{R+K_4}\right), & \text{if } K_5 \neq 0 \\ K_1 \times B + K_2 \times \frac{B}{256-L} + K_3 \times D, & \text{if } K_5 = 0. \end{cases}$$

Comparação Protocolos Vetor Distância

TABLE 5.2 Comparison of protocols in the distance vector protocol family.

Protocol	RIPv1	RIPv2	IGRP	EIGRP	RIPng
Address Family	IPv4	IPv4	IPv4	IPv4	IPv6
Metric	Hop	Hop	Composite	Composite	Hop
Information Communication	Unreliable, broadcast	unreliable, multicast	Unreliable, multicast	Reliable, multicast	Unreliable, multicast
Routing Computation	Bellman–Ford	Bellman–Ford	Bellman–Ford	Diffusing computation	Bellman–Ford
VLSM/CIDR	No	Yes	No	Yes	v6-based
Remark	Slow convergence; split horizon	Slow convergence; split horizon	Slow convergence; split horizon	Fast, loop-free convergence; chatty protocol	Slow convergence; split horizon

UNICAMP



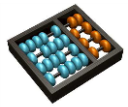
OSPF (*Open Shortest Path First*)

- "open" (aberto): publicamente disponível
- Usa algoritmo do Estado de Enlaces
  - ✓ disseminação de pacotes EE
  - ✓ Mapa da topologia a cada nó
  - ✓ Cálculo de rotas usando o algoritmo de Dijkstra

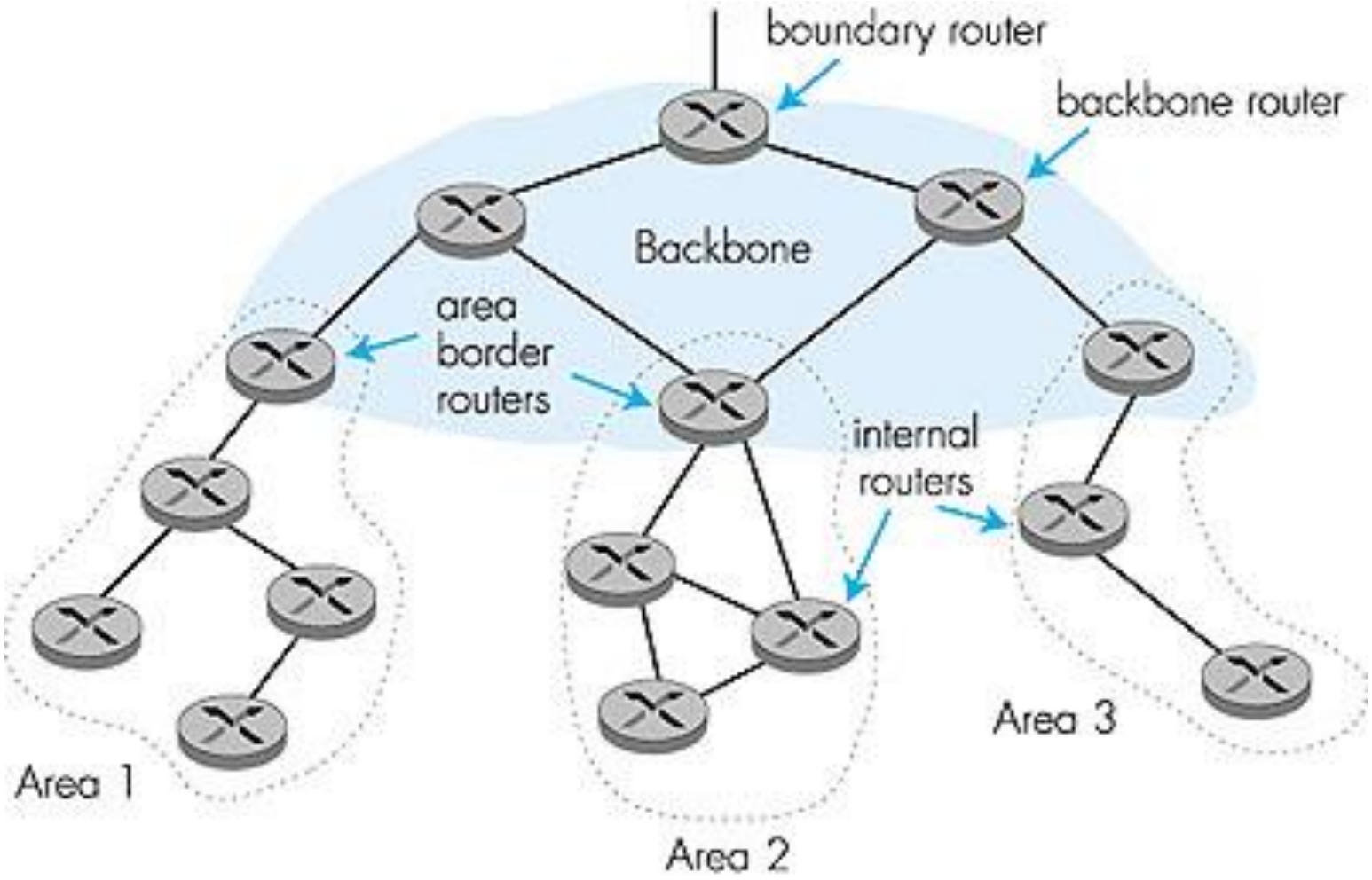
- Anúncio de OSPF inclui uma entrada por roteador vizinho
- Anúncios disseminados para SA inteiro (via inundação)

OSPF: características "avançadas" (não existentes no RIP)

- Segurança: todas mensagens OSPF autenticadas (para impedir intrusão maliciosa); conexões TCP usadas
- Caminhos Múltiplos de custos iguais permitidos (o RIP permite e usa apenas uma rota)
- Para cada enlace, múltiplas métricas de custo para TOS diferentes (p.ex, custo de enlace de satélite colocado como "baixo" para melhor esforço; "alto" para tempo real)
- Suporte integrado para ponto a ponto e multiponto:
  - ✓ OSPF multiponto (MOSPF) usa mesma base de dados de topologia usado por OSPF
- OSPF hierárquico em domínios grandes.

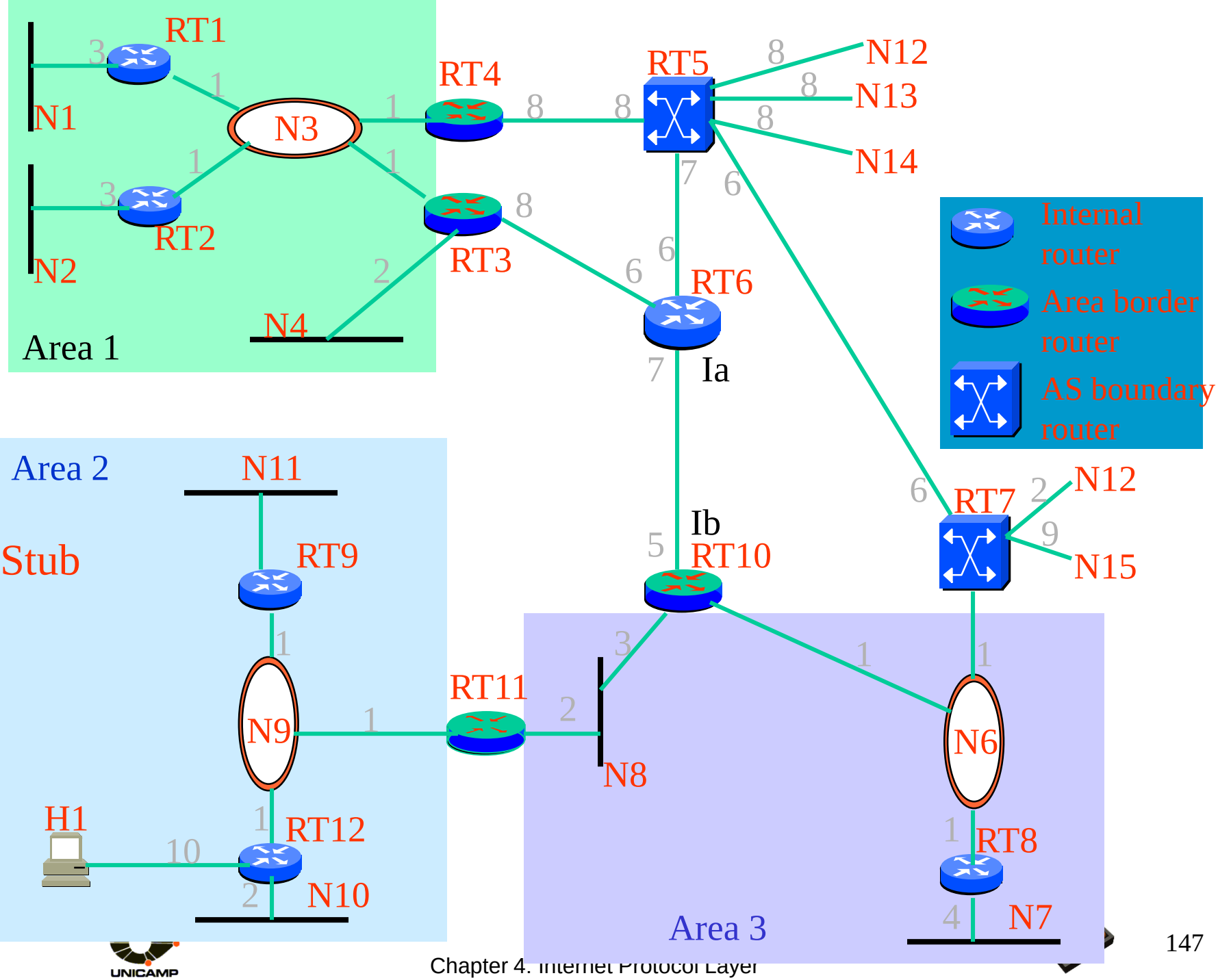


OSPF Hierárquico



OSPF Hierárquico

- **Hierarquia de dois níveis:** área local, backbone.
  - ✓ Anúncios de EE disseminados apenas na mesma área
  - ✓ cada nó possui topologia detalhada da área; apenas sabe a direção (caminho mais curto) para redes em outras áreas (alcançadas através do backbone).
- **Roteador de fronteira de área:** "sumariza" distâncias às redes na sua própria área, anuncia a outros roteadores de fronteira de área.
- **Roteadores do backbone:** realizam roteamento OSPF limitado ao backbone.
- **Roteadores de fronteira:** ligam a outros SAs.


RT1
RT2
RT3
RT4
RT5
RT6
RT7
RT8
RT9
RT10
RT11
RT12
N1
N2
N3
N4
N6
N7
N8
N9
N10
N11
N12
N13
N14
N15
H1
Area 1
Area 2
Area 3
Stub
Ia
Ib
Internal router
Area border router
AS boundary router

OSPF Example: Intra-area

- Informação sobre a área difundidada por RT3 and RT4 para o backbone.

Network	Cost advertised by RT3	Cost advertised by RT4
N1	4	4
N2	4	4
N3	1	1
N4	2	3

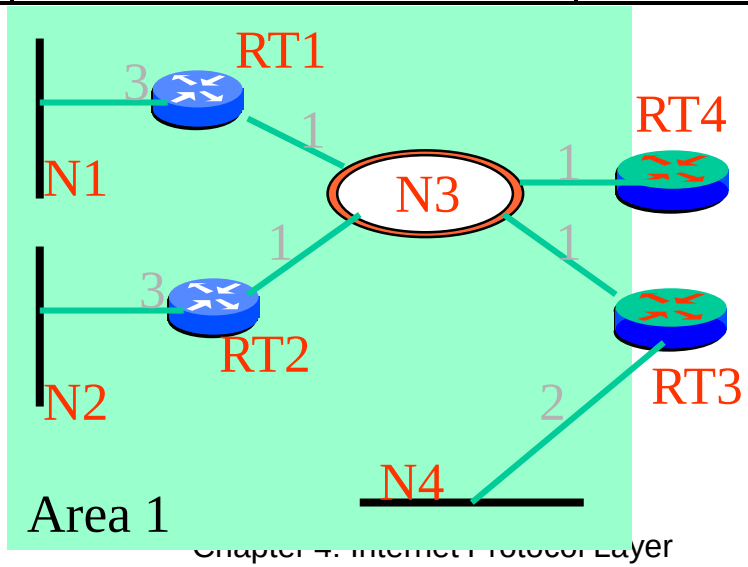
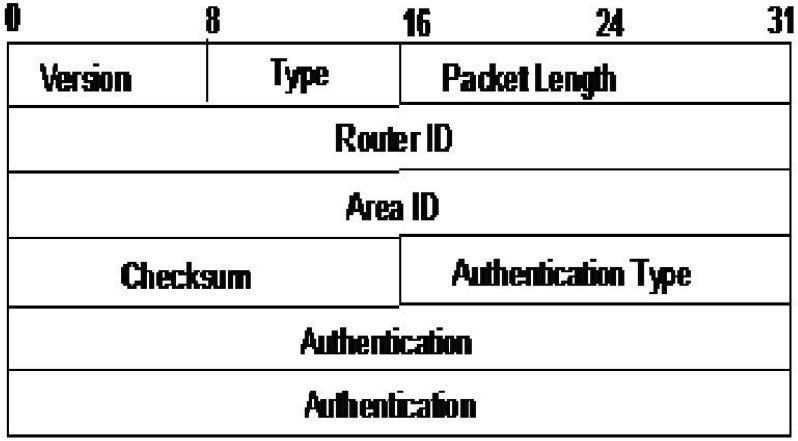


Tabela de roteamento - OSPF

- Tabela do roteador RT4

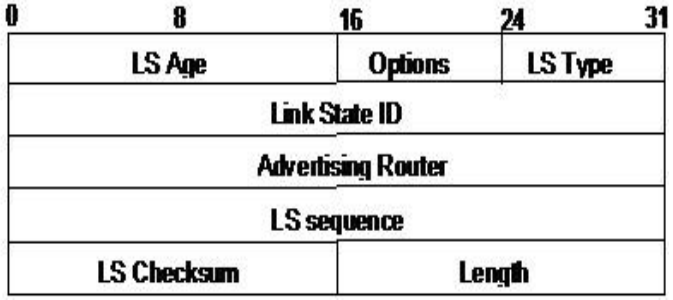
Destination	Path Type	Cost	Next Hop
N1	intra-area	4	RT1
N2	intra-area	4	RT2
N3	intra-area	1	direct
N4	intra-area	3	RT3
N6	Inter-area	15	RT5
N7	inter-area	19	RT5
N8	Inter-area	18	RT5
N9-N11	inter-area	36	RT5
N12	Type 1 external	16	RT5
N13	Type 1 external	16	RT5
N14	Type 1 external	16	RT5
N15	Type 1 external	23	RT5

Tipos de Mensagem OSPF



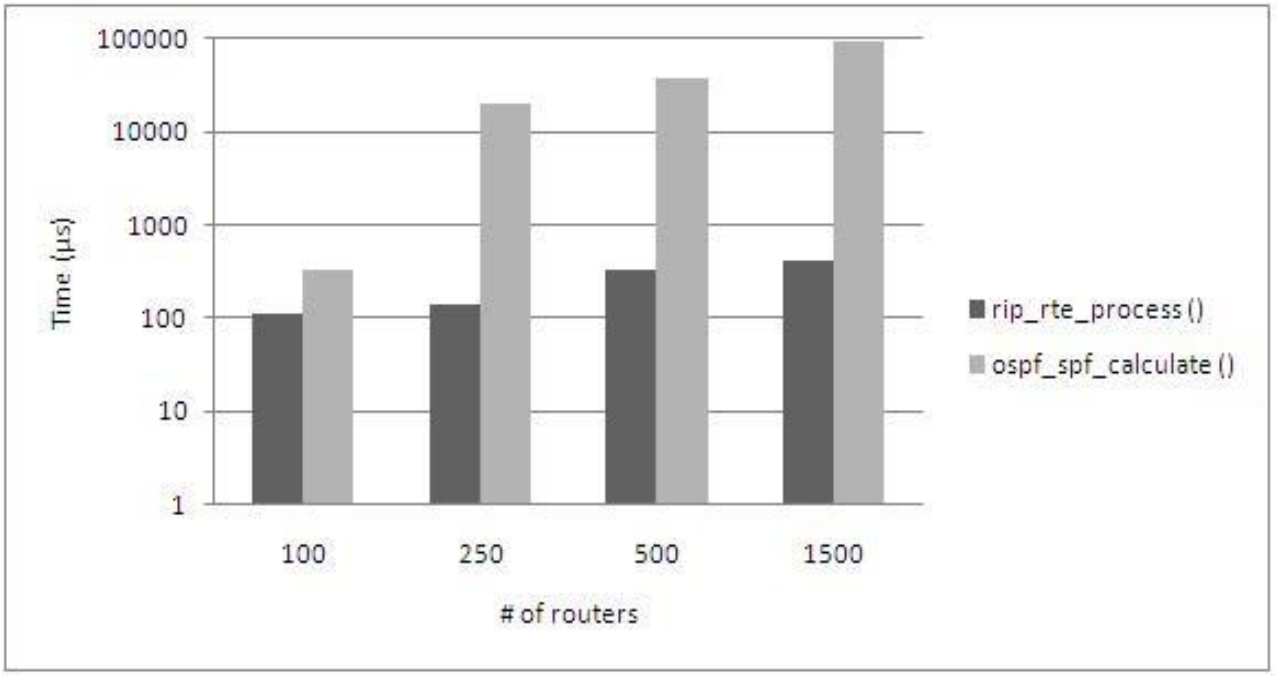
Type	Description
1	Hello
2	Database Description
3	Request
4	Update
5	Acknowledgment

Mensagens LS

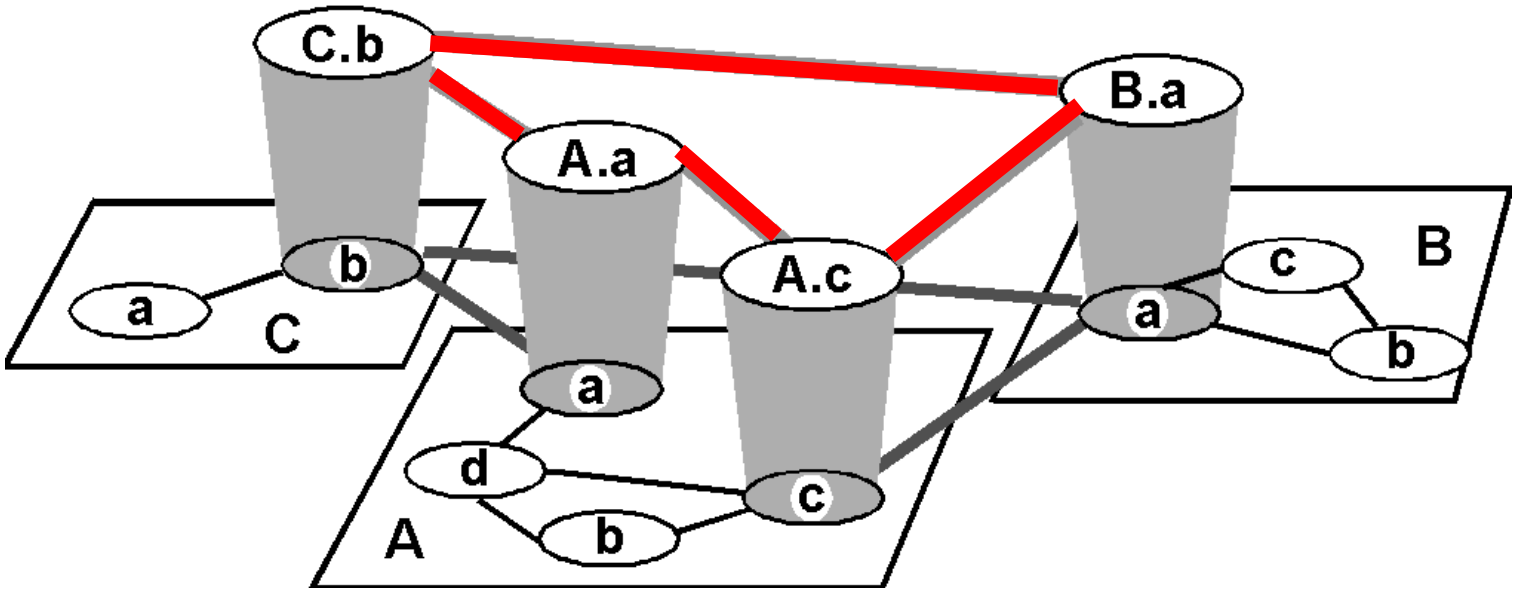


LS Type	LS Name	Originated by	Scope of Flood	Description
1	Router-LSAs	All routers	Area	Describes the collected states of the router's interfaces to an area.
2	Network-LSAs	Designated router	Area	Contains the list of routers connected to the network.
3	Summary-LSAs (IP network)	Area border router	Associated Areas	Describes routes to inter-area networks.
4	Summary-LSAs (ASBR)	Area border router	Associated Areas	Describes routes to AS boundary routers.
5	AS-external-LSAs	AS boundary router	AS	Describes routes to other Ass.

Comparativo Desempenho



Roteamento Inter-SA



Operação BGP

Q: O que um roteador BGP faz?

- Envia anúncio de rotas para seus vizinhos;
- Recebe e filtra anúncios de rotas dos seus vizinhos diretamente conectados
- Escolha da rota .
  - ✓ Para rotear para o destino X, qual caminho (entre tantos anunciados) deve ser seguindo?

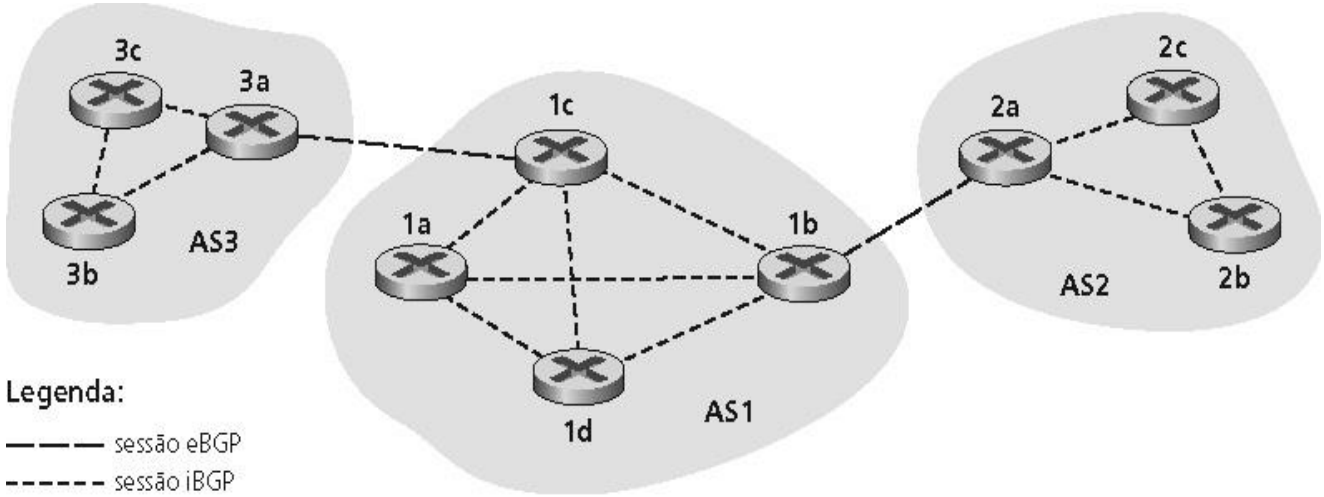
BGP

- BGP (Border Gateway Protocol): é o padrão de fato para uso na Internet
- BGP provê cada AS dos meios para:
  1. Obter informações de alcance de sub-rede dos Ass. Vizinhos
  2. Propagar informações de alcance para todos os roteadores internos ao AS
  3. Determinar “boas” rotas para as sub-redes baseado em informações de alcance e política
- Permite que uma subnet comunique sua existência para o resto da Internet: **“Estou aqui”**

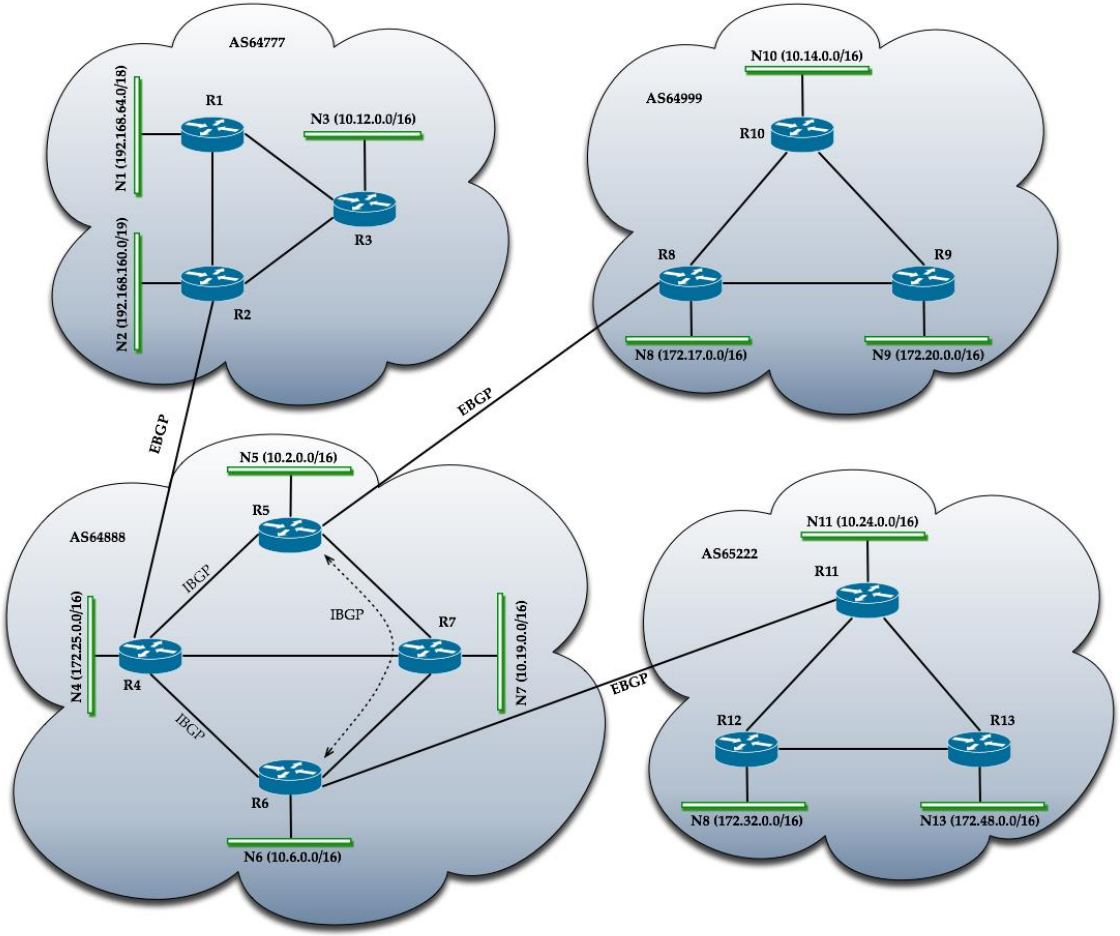
Difusão Informação - BGP

Pares de roteadores (BGP peers) trocam informações de roteamento por conexões TCP semipermanentes: sessões BGP

- Note que as sessões BGP não correspondem aos links físicos
- Quando AS2 comunica um prefixo ao AS1, AS2 está **prometendo** que irá encaminhar todos os datagramas destinados a esse prefixo em direção ao prefixo
  - AS2 pode agregar prefixos em seu comunicado



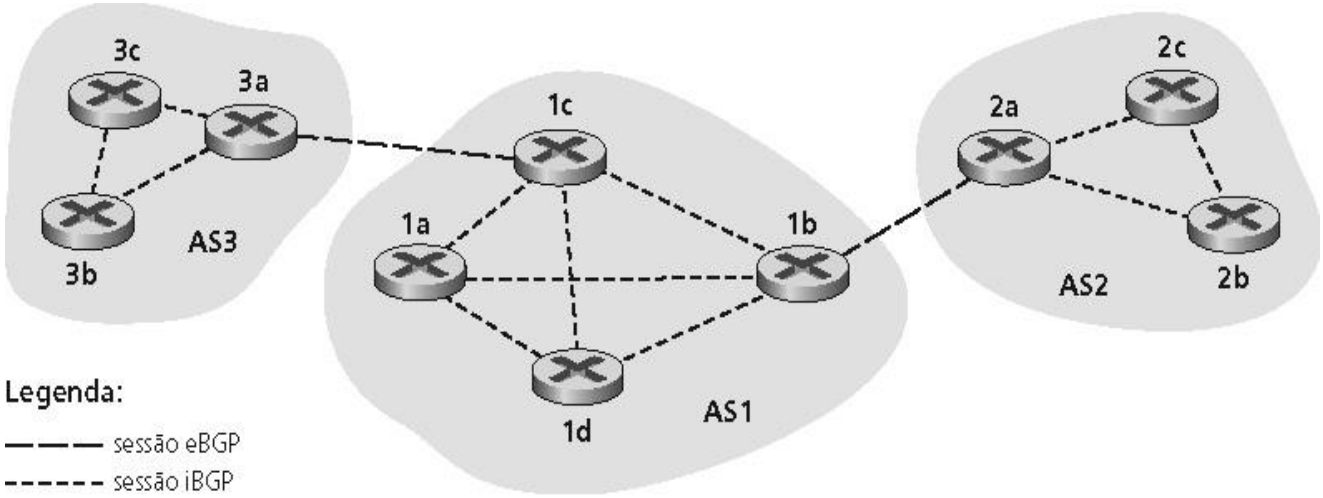
iBGP e eBGP


AS64777
N1 (192.168.64.0/18)
R1
N3 (10.12.0.0/16)
R3
N2 (192.168.160.0/19)
R2
AS64999
N10 (10.14.0.0/16)
R10
R8
R9
N8 (172.17.0.0/16)
N9 (172.20.0.0/16)
EBGP
EBGP
AS64888
N5 (10.2.0.0/16)
R5
IBGP
IBGP
IBGP
N4 (172.25.0.0/16)
R4
R7
N7 (10.19.0.0/16)
R6
N6 (10.6.0.0/16)
AS65222
N11 (10.24.0.0/16)
R11
EBGP
R12
R13
N8 (172.32.0.0/16)
N13 (172.48.0.0/16)

Difusão de Informação - BGP

Em cada sessão eBGP entre 3a e 1c, AS3 envia informações de alcance de prefixo para AS1.

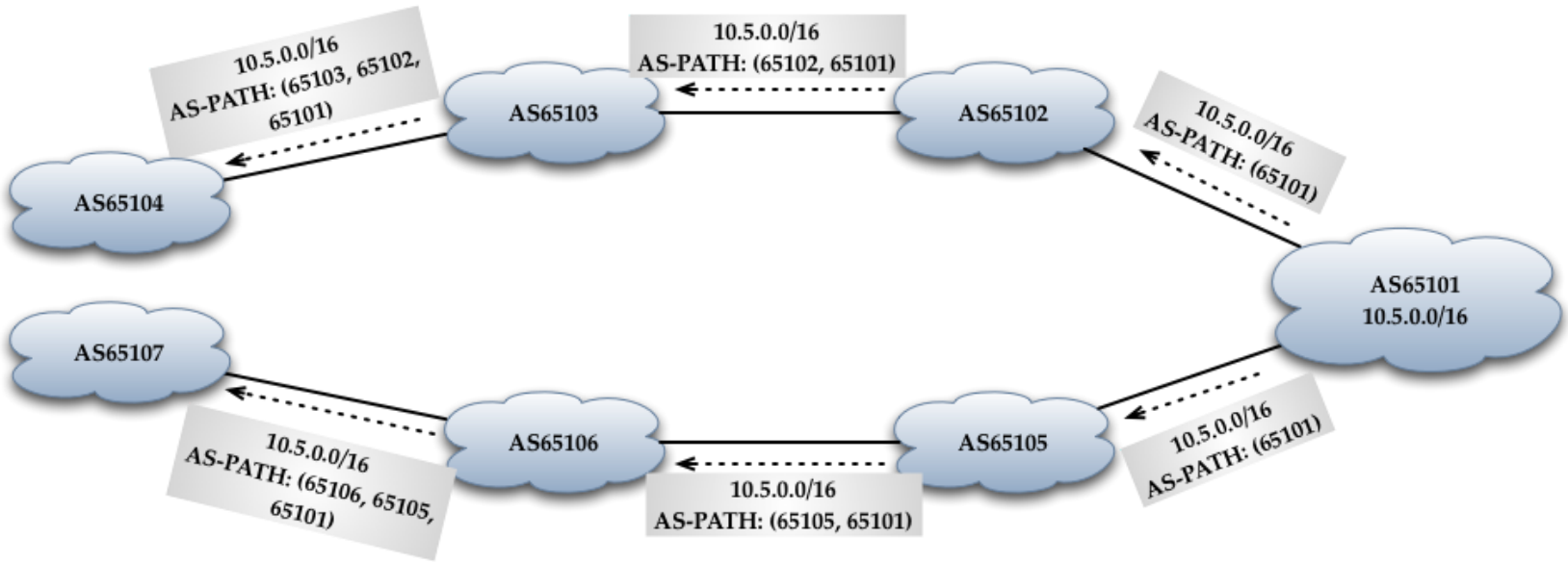
- 1c pode então usar iBGP para distribuir essa nova informação de alcance de prefixo para todos os roteadores em AS1
- 1b pode recomunicar essa nova informação para AS2 por meio da sessão eBGP 1b-para-2a.
- Quando um roteador aprende um novo prefixo, ele cria uma entrada para o prefixo em sua tabela de roteamento.



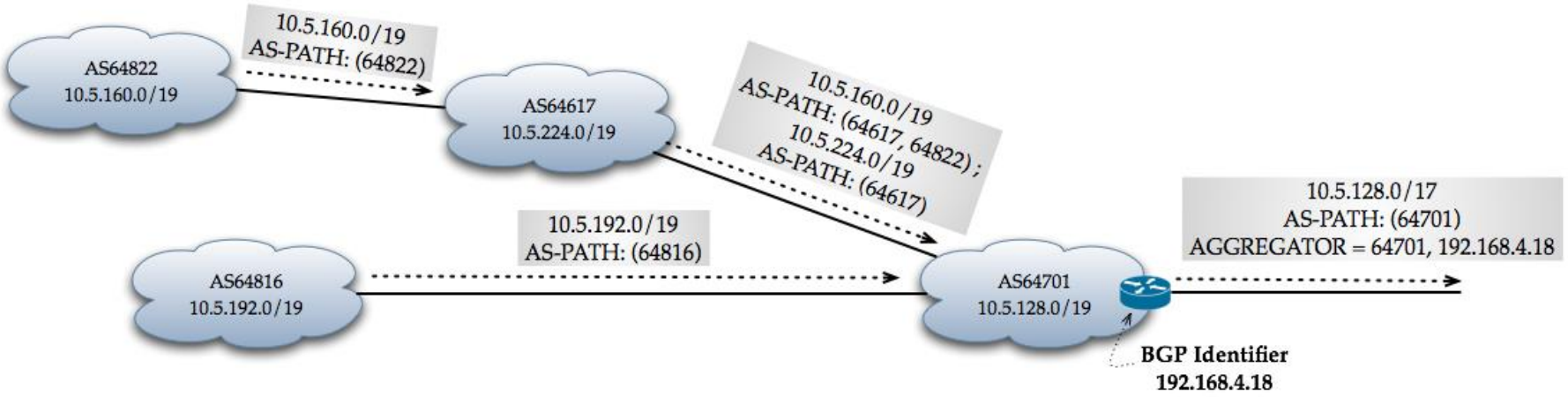
Rotas BGP

- Quando se comunica um prefixo, o comunicado inclui os atributos do BGP.
  - Prefixo + atributos = "rota"
- Dois atributos importantes:
- AS-PATH: contém os ASs pelos quais o comunicado para o prefixo passou: AS 67 AS 17
  - NEXT-HOP: Indica o roteador específico interno ao AS para o AS do próximo salto (next-hop). (Pode haver múltiplos links do AS atual para o AS do próximo salto.
- Quando um roteador gateway recebe um comunicado de rota, ele usa política de importação para aceitar/rejeitar.

Atributo AS_PATH



Agregação de Rotas



Seleção de Rotas - BGP

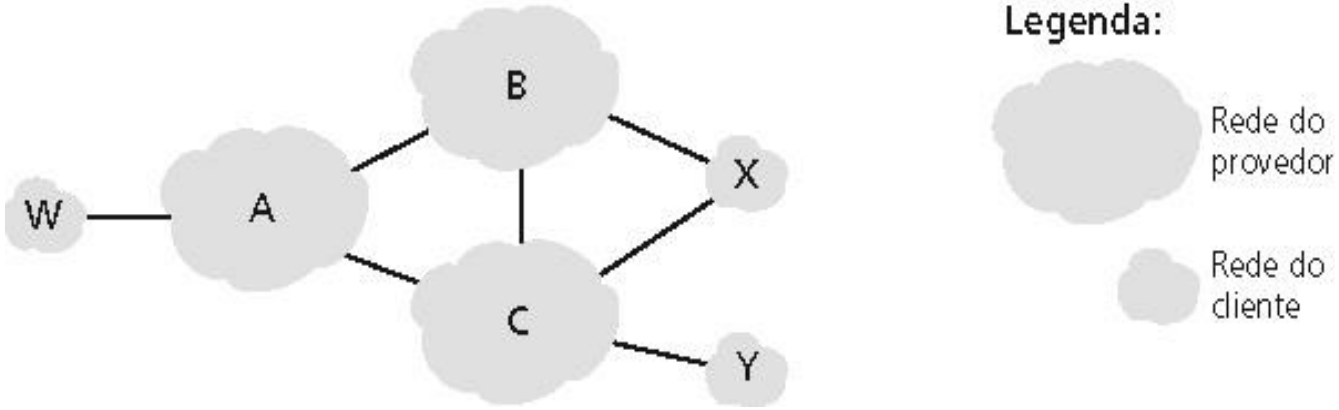
Um roteador pode aprender mais do que 1 rota para o mesmo prefixo. O roteador deve selecionar uma rota

- Regras de eliminação:
  - Atributo de valor de preferência local: decisão de política
  - AS-PATH (caminho) mais curto
  - Roteador do NEXT-HOP (próximo salto) mais próximo: roteamento da "batata quente"
  - Critérios adicionais

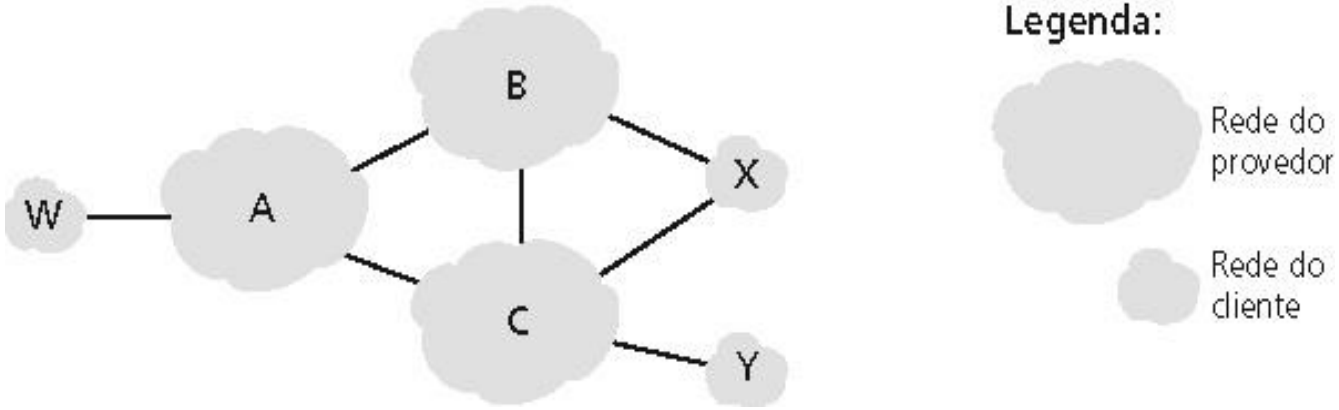
Selação de Rotas

Network	Next Hop	LOCAL_PREF	Weight	Best?	PATH	Origin
61.13.0.0/16	139.175.56.165		0	N	4780,9739	IGP
	140.123.231.103		0	N	9918,4780,9739	IGP
	140.123.231.100	0	0	Y	9739	IGP
61.251.128.0/20	139.175.56.165		0	Y	4780,9277,17577	IGP
	140.123.231.103		0	N	9918,4780,9277,17577	IGP
211.73.128.0/19	210.241.222.62		0	Y	9674	IGP
218.32.0.0/17	139.175.56.165		0	N	4780,9919	IGP
	140.123.231.103		0	N	9918,4780,9919	IGP
	140.123.231.106		0	Y	9919	IGP
218.32.128.0/17	139.175.56.165		0	N	4780,9919	IGP
	140.123.231.103		0	N	9918,4780,9919	IGP
	140.123.231.106		0	Y	9919	IGP

- Mensagens BGP são trocadas usando o TCP
- Mensagens BGP:
  - OPEN: abre conexão TCP para o peer e autentica o transmissor
  - UPDATE: comunica novo caminho (ou retira um antigo)
  - KEEPALIVE mantém a conexão ativa na ausência de atualizações (updates); também ACKs OPEN request
  - NOTIFICATION: reporta erros em mensagens anteriores; também usado para fechar a conexão



- A,B,C são redes do provedor
- X,W,Y são clientes (das redes do provedor)
- X é dual-homed: anexados a duas redes
  - X não quer rotear de B via X para C
  - … então X não comunicará ao B uma rota para C



- A comunica ao B o caminho AW
- B comunica ao X o caminho BAW
- B deveria comunicar ao C o caminho BAW?
  - De jeito nenhum! B não obtém nenhum "rendimento" em rotear CBAW pois nem W nem C são seus clientes
  - B quer forçar C a rotear para W via A
  - B quer rotear **somente** de/para seus clientes!

Porque protocolos Intra- e Inter-AS diferentes ?

Políticas:

- Inter-SA: administração quer controle sobre como tráfego roteado, quem transita através da sua rede.
- Intra-AS: administração única, logo são desnecessárias decisões políticas

Escalabilidade:

- roteamento hierárquico economiza tamanho de tabela de rotas, reduz tráfego de atualização

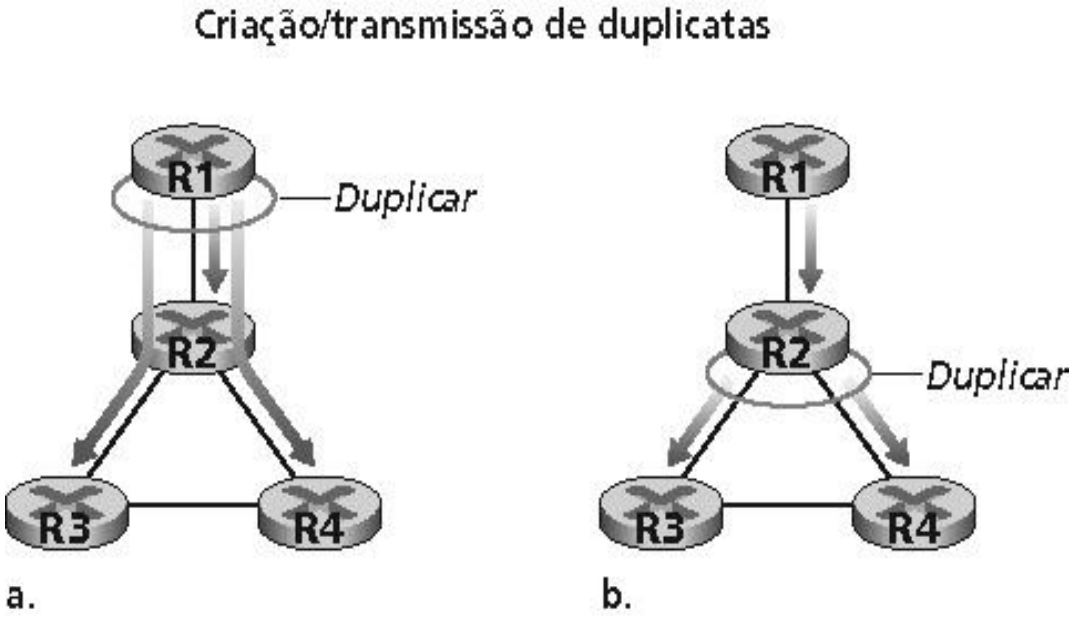
Desempenho:

- Intra-AS: pode focar em desempenho
- Inter-AS: políticas podem ser mais importantes do que desempenho

Roteiro

- 4. 1 Introdução
- 4.2 Circuitos virtuais x datagrama
- 4.3 Como é um roteador
- 4.4 Protocolo IP
  - ✓ Formato datagarma
  - ✓ endereçamento IPv4
  - ✓ ICMP
  - ✓ IPv6
- 4.5 Algoritmos de roteamento
  - ✓ Estado de enlace
  - ✓ Vetor distância
  - ✓ Roteamento hierarquico
- 4.6 Roteamento na Internet
  - ✓ RIP
  - ✓ OSPF
  - ✓ BGP
- 4.7 Roteamento Broadcast e multicast

Difusão (Broadcast)



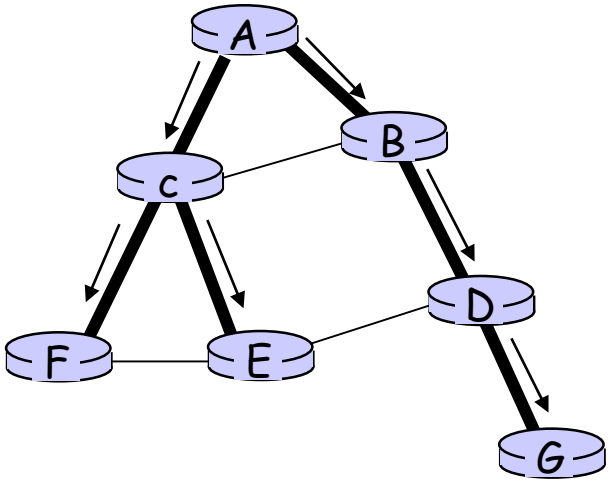
Duplicação na origem versus duplicação na rede.
(a) duplicação na origem, (b) duplicação na rede

Duplicação na rede

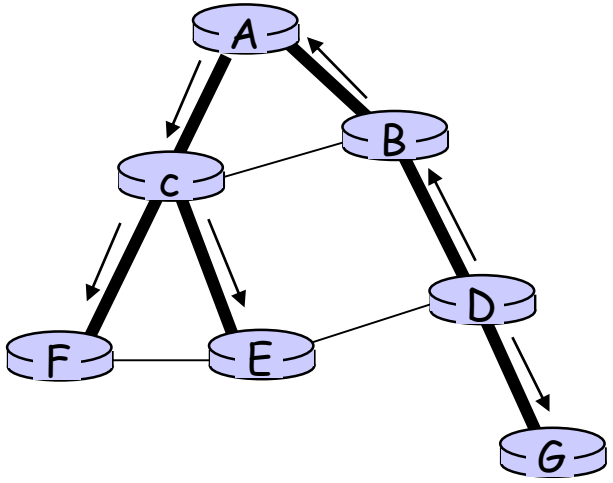
- Dilúvio: quando roteador recebe pacote, envia para todos
- Problemas de ciclo e tempestade
- Dilúvio controlado: só envia para vizinhos caso não tenha enviado anteriormente
- Ou controla informação em tabela ou faz encaminhamento caminho
- spanning tree
- Nós não recebem pacotes duplicatas

Spanning Tree

- Construção de spanning tree,
- Nós encaminham pela spanning tree



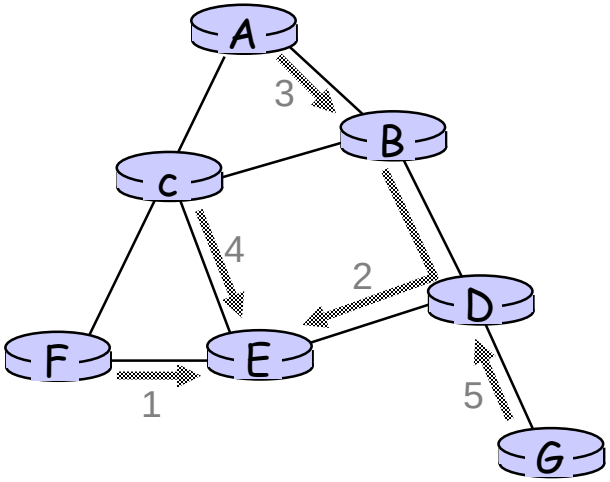
(a) Broadcast initiated at A



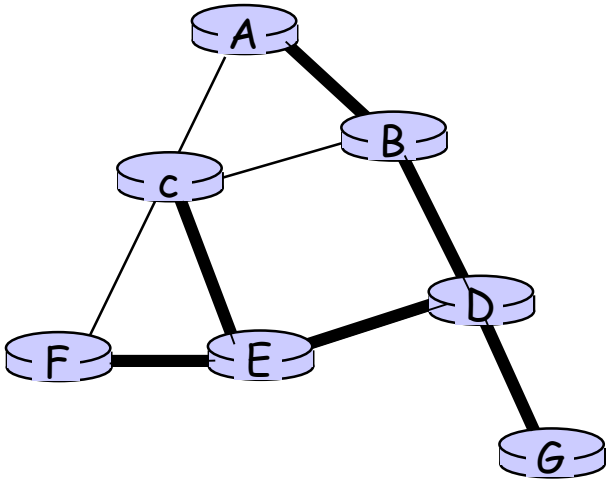
(b) Broadcast initiated at D

Criação da Spanning Tree

- Nó central
- Cada nó envia mensagem de enxerto para o centro
  - ✓ Mensagens são encaminhadas até encontrar algum nó na spanning tree



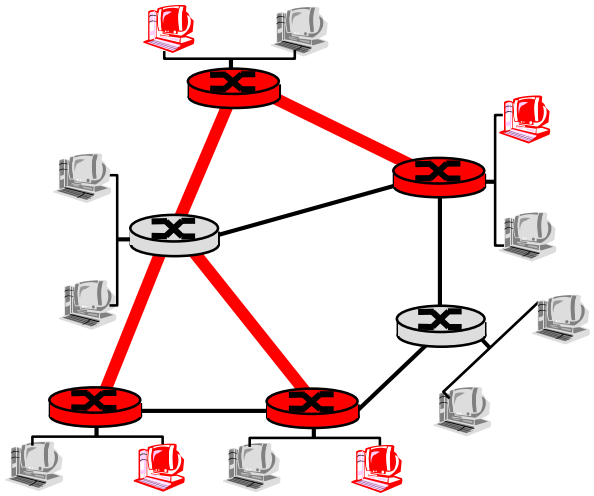
(a) Construção da spanning tree



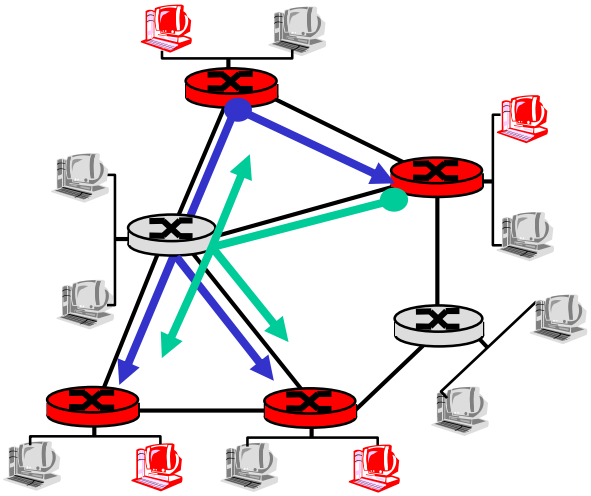
(b)Spanning tree constuída

Roteamento Multicast

- Objetivo: encontrar árvore find a tree (or trees) connecting routers having local mcast group members
  - ✓ Árvore: nem todas rotas são utilizadas
  - ✓ Baseada na fonte: diferentes árvores de cada transmissor para um receptor
  - ✓ *Árvore compartilhada:* mesma árvore usada por todos os membros do grupo



Shared tree



Source-based trees

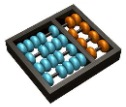
UNICAMP

Approaches para construção de árvore mcast

Approaches:

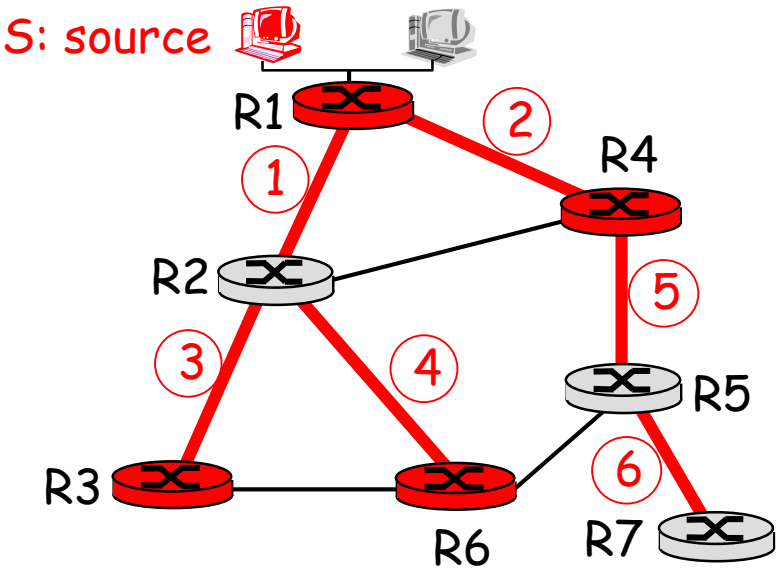
- Árvores baseadas na fonte: uma árvore por fontes
  - Árvores com menor caminho
  - reverse path forwarding
- Árvore compartilhadas por grupo: grupo usa uma única árvore
  - minimal spanning (Steiner)
  - Árvores baseadas no centro

UNICAMP



Árvores com camonhos mais curtos

- Árvores mcast: árvores com rotas de caminhos mais entre fonte e todos os recsptores curtos :
  - ✓ Algoritmo de Dijkstra



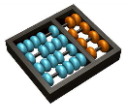
LEGEND

Roteadores com membros Dogrupo associados

Roteadores sem membros do grupo associado

i Enlaces usados para encaminhar
i indica ordem em que o enlace f
adicionado pelo algoritmo

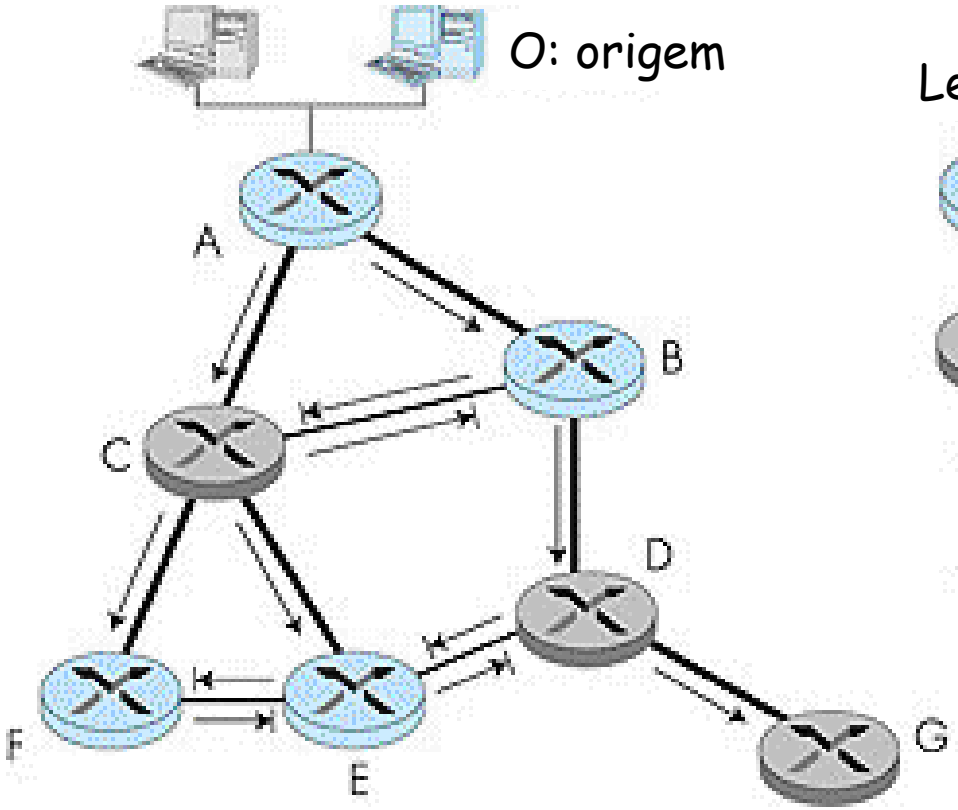
UNICAMP



Envio pelo Caminho Reverso

- Idéia simples, mas elegante.
- Quando um roteador recebe um pacote multicast, ele transmite o pacote em todos os seus enlaces de saída (exceto por aquele em que recebeu o pacote) apenas se o pacote tiver sido recebido através do enlace que está no seu caminho mais curto até o transmissor (origem).
- Note que o roteador não precisa conhecer o caminho mais curto até a origem, mas apenas o próximo roteador no seu caminho mais curto unicast até a origem.

Envio pelo Caminho Reverso



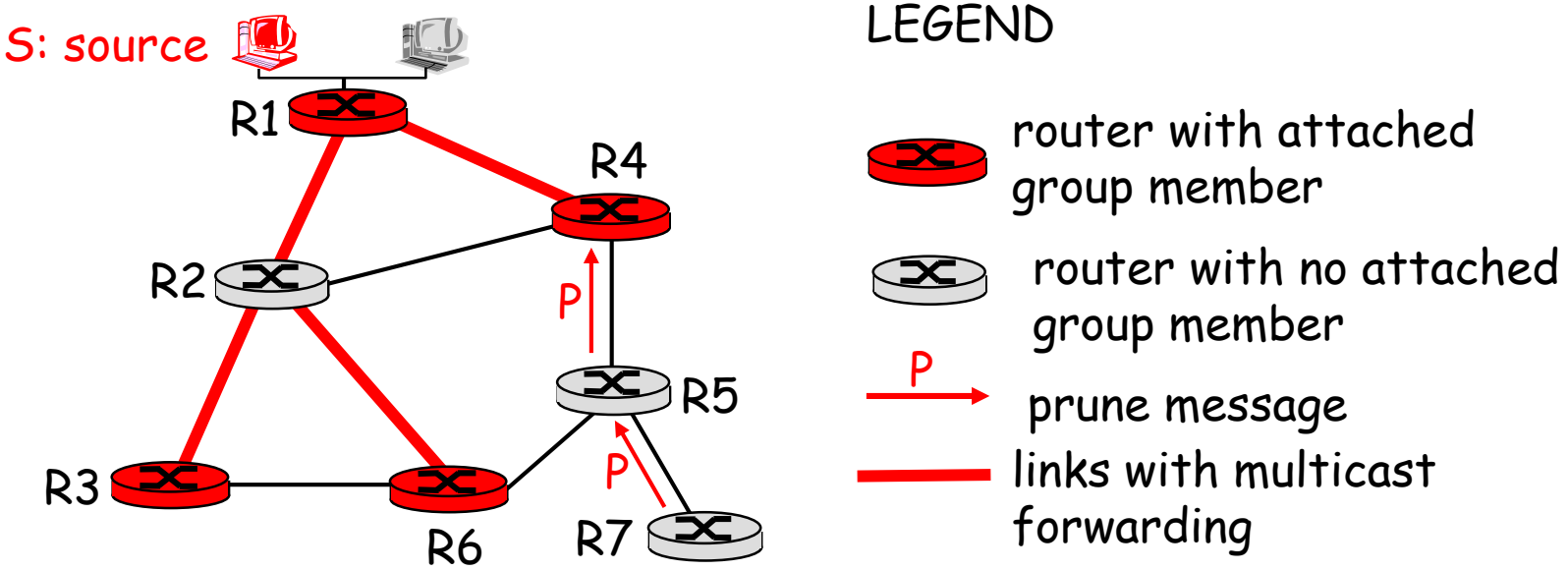
Legenda

- roteador com conexão a algum membro do grupo
- roteador sem conexão com nenhum membro do grupo
- pacote a ser enviado
- pacote que não será enviado além do roteador receptor

Problema: *G* e outros roteadores a partir dele receberiam pacotes multicast apesar de não terem conexão com nenhum *host* participante do grupo!
Solução: Podar a árvore!

Poda

- Mensagens de poda são enviadas "upstream", eliminação de subarvore sem elementos de grupo

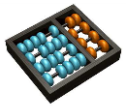


UNICAMP

Shared-Tree: Steiner Tree

- Steiner Tree: árvore de custo mínimo conectando todos roteadores do grupo
- problema NP-completo
- exelentes heuristicas
- Não é usada na pratica:
  - ✓ Computacionalmente complexa
  - ✓ Informação sobre toda rede necessária
  - ✓ Necessidade de re-execução a cada alteração de membros

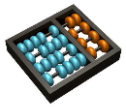
UNICAMP



Árvore baseada no centro

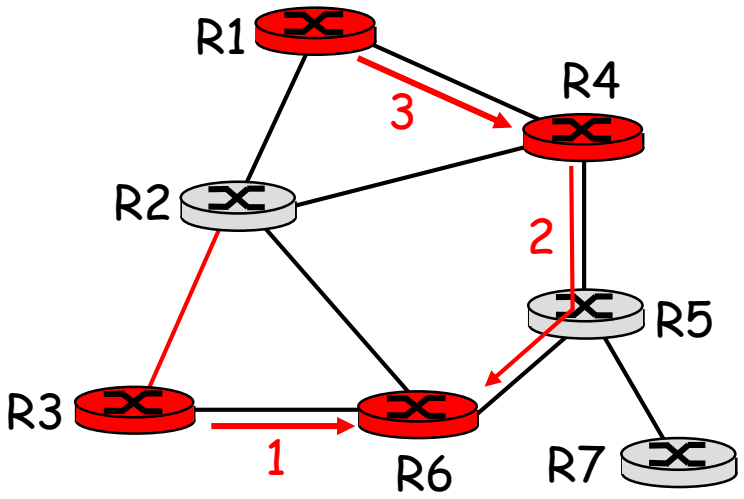
- Uma única árvore compartilhada por todos
- Um roteador identificado como centro da árvore
- Para enxertar nó:
  - ✓ *Roteador envia mensagemjoin-msg* para o roteador centro
  - ✓ *join-msg processada por roteadores intermediários e repassada ao centro*
  - ✓ *join-msg* ou chega a um nó da árvore ou ao centro
  - ✓ Caminho feito pela mensagem *join-msg* torna-se novo caminho para o roteador

UNICAMP



Exemplo: árvore baseada no centro

R6 é o centro:



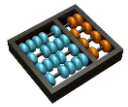
LEGENDA

Roteadores com membros associados

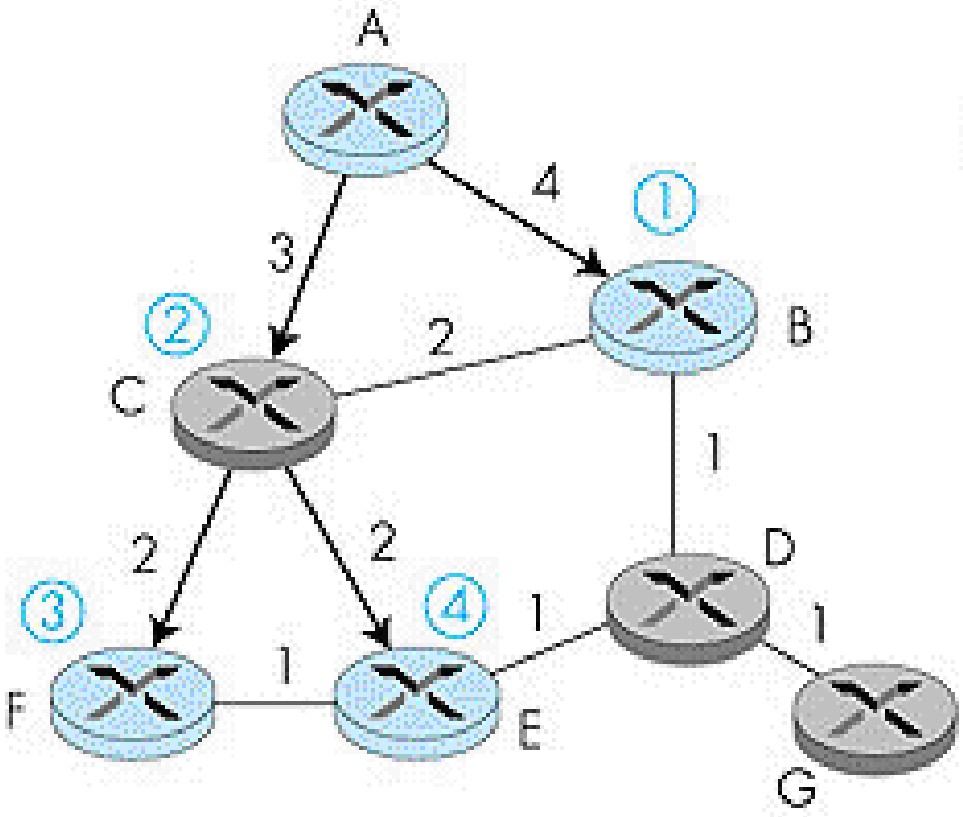
Roteadores sem membros associados

1 Ordem de geração da mensgem join

UNICAMP



Roteamento Multicast usando árvores baseadas nas origens

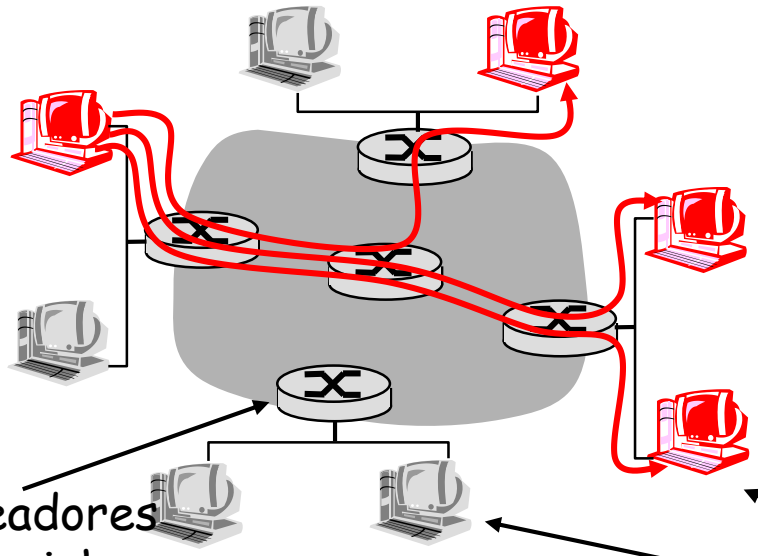


i-ésimo caminho
a ser adicionado

- Árvores de caminho mais curto a partir de cada origem.
- Este é um algoritmo de EE (cada roteador deve conhecer o estado de cada enlace na rede).
- Mais simples: envio pelo caminho reverso (RPF - *Reverse Path Forwarding*)

Multicast: um emissor para vários receptores

- Multicast: envia datagramas para múltiplos receptores com uma **única operação de transmissão**
  - ✓ analogia: um professor para vários estudantes,
  - ✓ alimentação de dados: cotações da bolsa de valores;
  - ✓ atualização de cache WWW;
  - ✓ ambientes virtuais interativos distribuídos, etc.
- Questão: como garantir multicast?



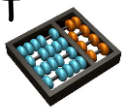
Roteadores encaminham datagramas unicast

Receptores multicast (vermelho)

Não é um receptor multicast

Multicast via unicast

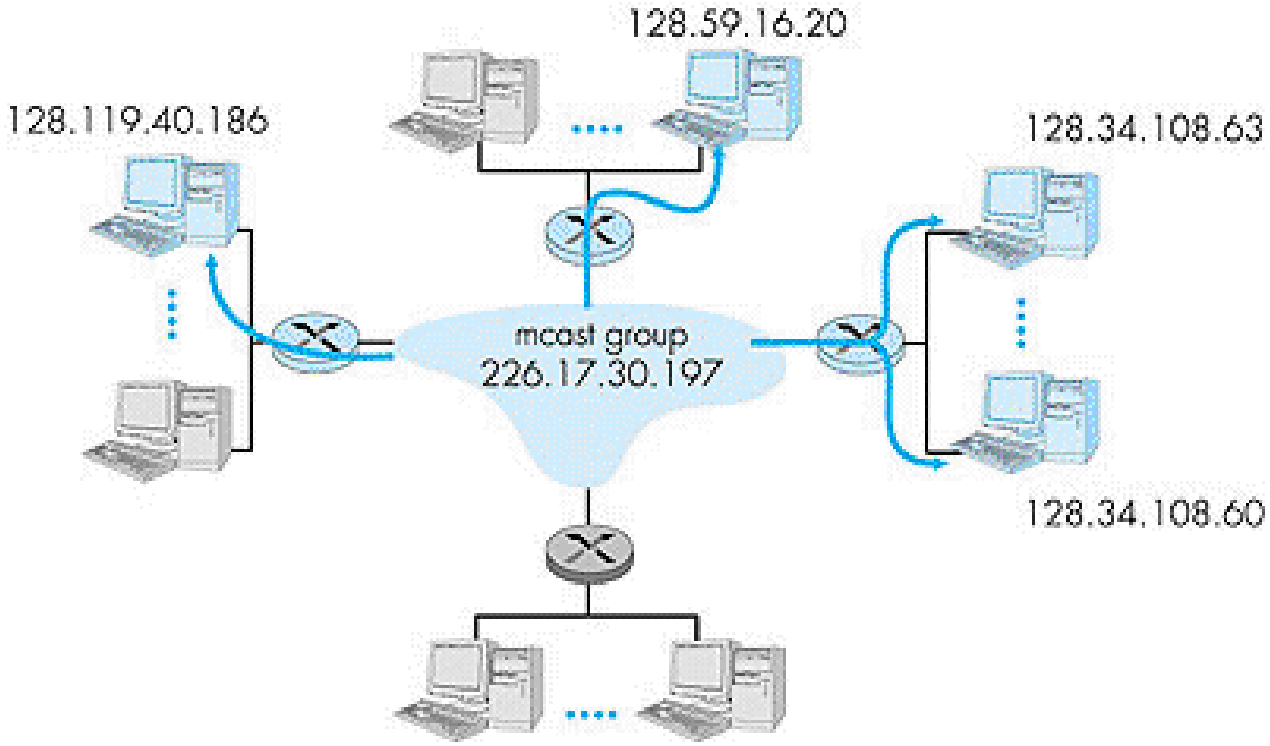
- Fonte envia N datagrams unicast, um para cada um dos N receptores



Desafios do Suporte a Multicast na Camada de Rede

- Como identificar os receptores de um datagrama multicast?
- Como endereçar um datagrama a ser enviado para estes receptores.
- Não dá para incluir o endereço IP de cada um dos destinos no cabeçalho do datagrama!
  - ✓ Não funciona para um grande número de receptores;
  - ✓ requer que o transmissor conheça a identidade e endereços de cada um dos destinatários.
- Endereço indireto: é usado um identificador único para um grupo de usuários.
- Grupo Multicast associado a um endereço classe D.

Modelo de Serviço Multicast da Internet

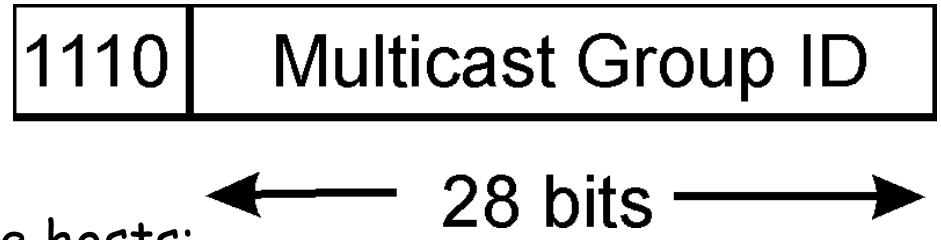


Conceito de grupo Multicast: uso de indireção

- ✓ Hosts endereçam os datagramas IP para o grupo multicast
- ✓ Roteadores encaminham os datagramas multicast para os hosts que se "juntaram" ao grupo multicast

Grupos Multicast

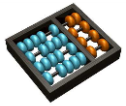
- Endereços classe D na Internet são reservados para multicast:



- Semântica de grupo de hosts:
  - ✓ qualquer um pode se "juntar" (receber) a um grupo multicast
  - ✓ qualquer um pode enviar para um grupo multicast
  - ✓ nenhuma identificação na camada de rede para os hosts membros
- *necessário:* infraestrutura para enviar datagramas multicast para todos os hosts que se juntaram ao grupo

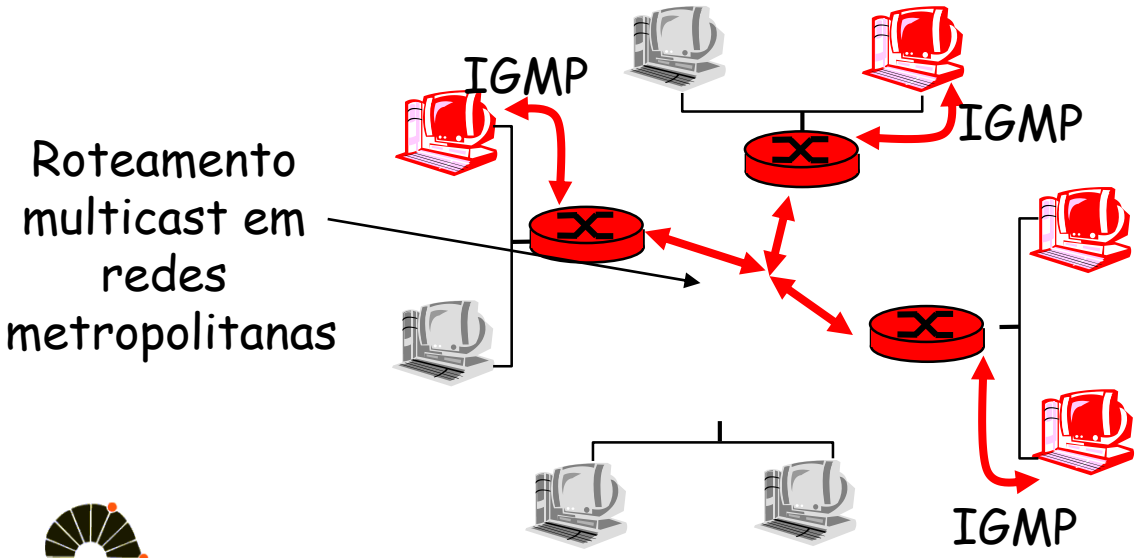
Grupos Multicast: questões

- Como um grupo é iniciado e como ele é encerrado?
- Como é escolhido o endereço do grupo?
- Como são adicionados novos *hosts* ao grupo?
- Qualquer um pode fazer parte (ativa) do grupo ou a participação é restrita?
- Caso seja restrita, quem determina a restrição?
- Os membros do grupo têm conhecimento das identidades dos demais membros do grupo na camada de rede?
- Como os roteadores interoperam para entregar um datagrama multicast a todos os membros do grupo?



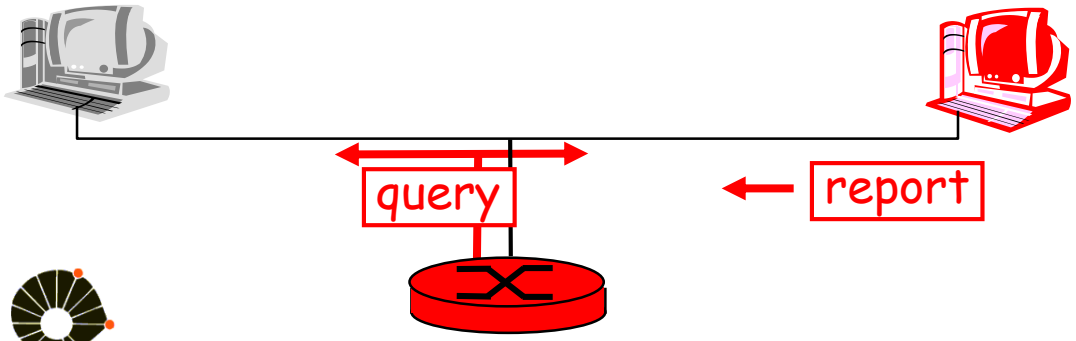
Juntando-se a um grupo Multicast: processo em dois passos

- *Rede local:* host informa ao roteador multicast local que deseja fazer parte do grupo: IGMP (Internet Group Management Protocol)
- *Rede metropolitanta:* roteador local interage com outros roteadores para receber os fluxos multicast
  - ✓ Vários protocolos (e.g., DVMRP, MOSPF, PIM)



IGMP: Internet Group Management Protocol – RFC 2236

- Opera entre o host e o roteador ao qual ele está conectado diretamente:
- *host:* envia notificação IGMP quando a aplicação se junta a um grupo multicast
  - ✓ IP_ADD_MEMBERSHIP opção de socket
  - ✓ host não necessita fazer uma notificação quando sai de um grupo
- *roteador:* envia requisição IGMP a intervalos regulares
  - ✓ host pertencente a um grupo multicast deve responder a requisição



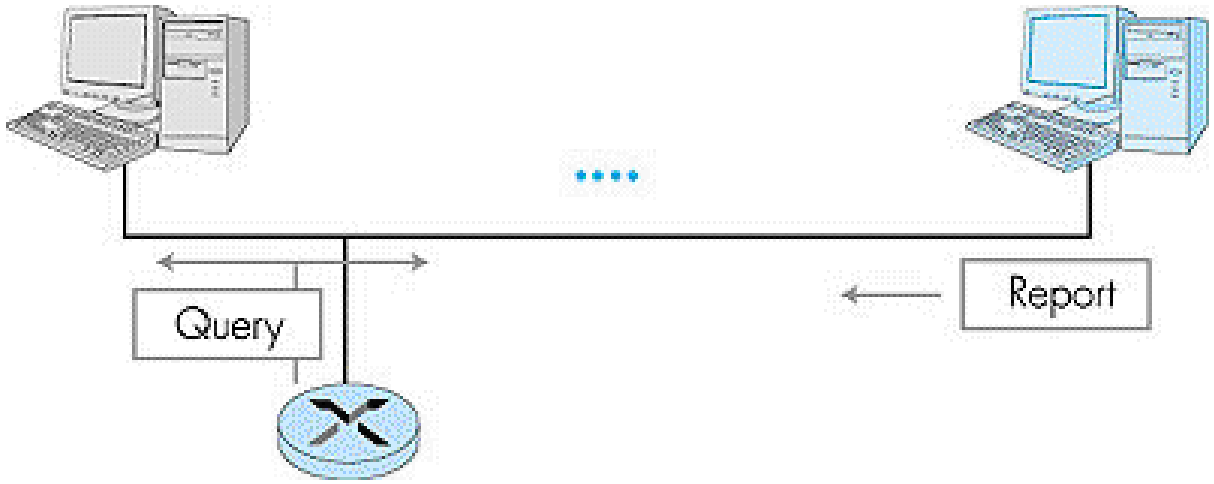
O Protocolo IGMP

- O IGMP fornece meios para que o host informe ao roteador ao qual está conectado que uma aplicação deseja ser incluída em um grupo multicast.
- Apesar do nome ele não é um protocolo que opera entre todos os *hosts* que tenham formado um grupo multicast.
- É necessário um outro protocolo para coordenar os roteadores multicast, de modo que os datagramas multicast sejam roteados até seus destinos: algoritmos de roteamento multicast da camada de rede.
  - ✓ Ex: PIM, DVMRP e MOSPF.

Tipos de Mensagens IGMP v2

Tipos das Mensagens IGMP	Enviada por	Finalidade
Consulta sobre participação em grupos:geral	Roteador	Consultar quais os grupos multicast em que os hosts associados estão incluídos.
Consulta sobre participação em grupos:específica	Roteador	Consultar se os hosts associados estão incluídos em um grupos multicast específico.
Relato de participação	Host	Relatar que o host quer ser ou já está incluído num dado grupo multicast.
Saída de grupo	Host	Relata que está saindo de um determinado grupo multicast.

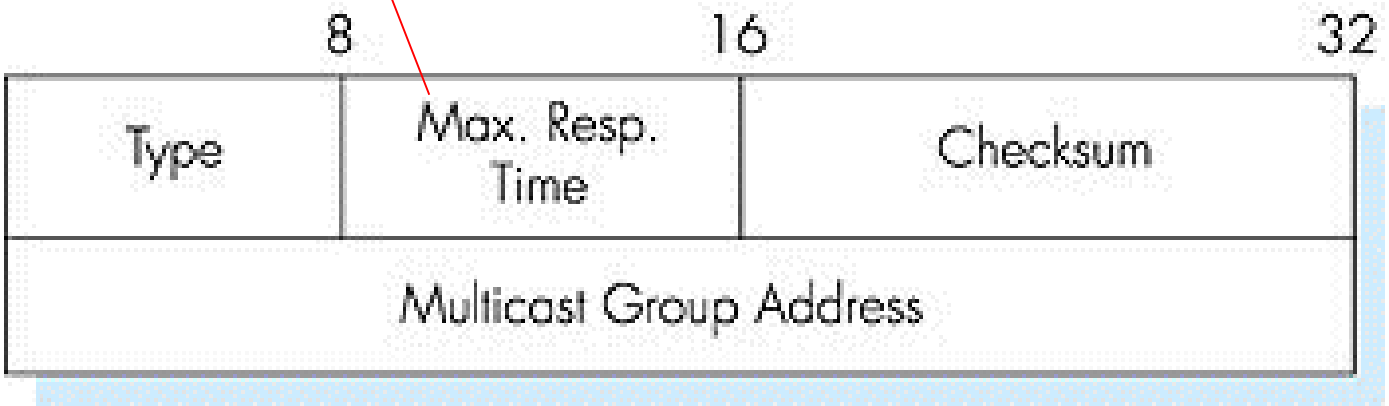
Consulta sobre participação e resposta



- As mensagens de relato também podem ser enviadas por iniciativa do host quando uma aplicação deseja ser incluída num grupo multicast.
- Para o roteador não importa quais nem quantos hosts fazem parte do mesmo grupo multicast.

Formato das Mensagens IGMP

Usado para suprimir relatos duplicados: cada host espera um tempo aleatório entre 0 e este valor máximo antes de enviar o seu relato. Se antes disto este host escutar o relato de algum outro host, ele descarta a sua mensagem.



Encapsuladas em datagramas IP com número de protocolo 2.

Modelo do Serviço Multicast da Internet

- Qualquer host pode ser incluído no grupo multicast na camada de rede.
  - O host simplesmente envia uma mensagem IGMP de relato de participação para o roteador ao qual está conectado.
- Em pouco tempo o roteador agindo em conjunto com os demais roteadores começará a entregar datagramas multicast para este host.
- Portanto, a adesão a um grupo é uma iniciativa do receptor.

Modelo do Serviço Multicast da Internet

- O transmissor não precisa se preocupar em adicionar receptores e nem controla quem é incluído no grupo.
- Também não há nenhum controle de coordenação a respeito de quem e quando pode transmitir para o grupo multicast.
- Não há nem mesmo uma coordenação na camada de rede sobre a escolha de endereços multicast: dois grupos podem escolher o mesmo endereço!
- Todos estes controles podem ser implementados na camada de aplicação. Alguns deles podem vir a ser incluídos na camada de rede.

Protocolos de Roteamento Multicast na Internet

- **DVMRP:** *Distance Vector Multicast Routing Protocol*
- **MOSPF:** *Multicast Open Shortest Path First*
- **PIM:** *Protocol Independent Multicast*

DVMRP – *Distance Vector Multicast Routing Protocol*

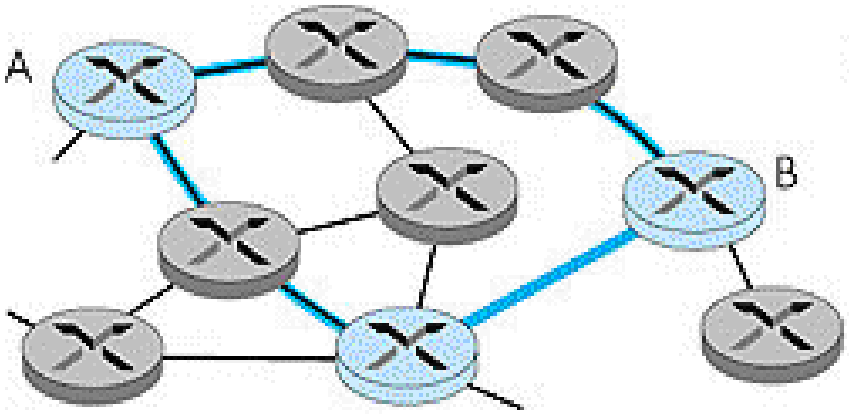
- Primeiro e o mais difundido.
- Implementa árvores baseadas nas origens com envio pelo caminho reverso, poda e enxerto.
- Utiliza o algoritmo de vetor de distância para permitir que o roteador calcule o enlace de saída que se encontra no caminho mais curto até cada uma das origens possíveis.
- Também calcula a lista dos roteadores que estão abaixo dele para questões de poda.

DVMRP – *Distance Vector Multicast Routing Protocol*

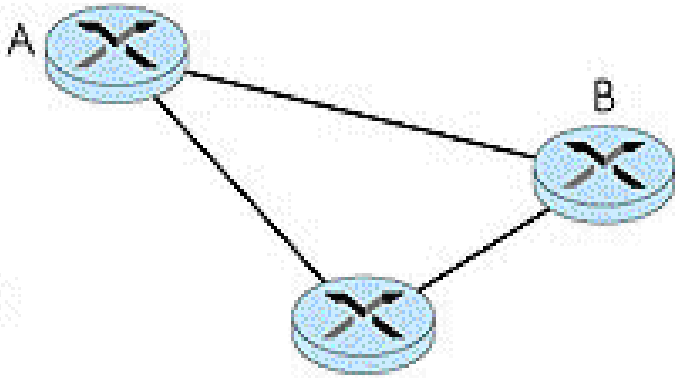
- A mensagem de poda contém a duração da poda (com valor default de 2 horas) após o qual o ramo é automaticamente enxertado na árvore.
- Uma mensagem de enxerto força a reinclusão de um ramo que tenha sido podado anteriormente da árvore multicast.

Implantação de roteamento Multicast na Internet

- O ponto crucial é que apenas uma pequena fração dos roteadores estão aptos ao Multicast.
- Tunelamento pode ser usado para criar uma rede virtual de roteadores com multicast.
  - ✓ Esta abordagem foi utilizada no Mbone



Topologia física



Topologia lógica

MOSPF - *Multicast Open Shortest Path First*

- É utilizado num Sistema Autônomo que utiliza o protocolo OSPF para o roteamento unicast.
- Os roteadores adicionam a informação dos grupos que devem atender junto com os anúncios dos estados dos enlaces.
- Com base nestas informações cada roteador do AS pode construir árvores de caminho mais curto, específicas para cada origem, já podadas para cada grupo multicast.

PIM - *Protocol Independent Multicast*

- Considera dois tipos de cenários:
  - **Modo denso**: os membros de um grupo estão concentrados numa dada região. A maior parte dos roteadores devem se envolver com o roteamento dos datagramas de multicast.
  - **Modo esparso**: os membros de um grupo estão muito dispersos geograficamente.
- Conseqüências:
  - No modo denso: todos os roteadores devem ser envolvidos com o multicast. Uma abordagem como a de encaminhamento pelo caminho reverso é adequada.
  - No modo esparso: o default é que o roteador não se envolva com multicast. Os roteadores devem enviar mensagens explicítas solicitando a sua inclusão.

Roteamento Multicast entre Sistemas Autônomos

- Cada SA pode utilizar um protocolo de roteamento multicast diferente.
- Ainda não existe um padrão para o roteamento multicast inter-SA.
- O padrão *de fato* tem sido o DVMRP que não é adequado por ser um protocolo do tipo modo denso, enquanto que os roteadores multicast atuais estão espalhados.

Fatores de avaliação de protocolos multicast

- Escalabilidade: como cresce a quantidade de info de estados com o crescimento do número de grupos e dos transmissores de um grupo?
- Dependência do roteamento unicast: Ex.: MOSPF x PIM.
- Recepção excessiva (não necessária) de tráfego.
- Concentração de tráfego: a árvore única concentra tráfego em poucos enlaces.
- Optimalidade dos caminhos de envio.

Capítulo 4: Resumo

- Iniciamos a nossa jornada rumo ao núcleo da rede.
- Roteamento dos datagramas: um dos maiores desafios da camada de rede.
  - ✓ Particionamento das redes em SAs.
  - ✓ Problema de escala pode ser resolvido com a hierarquização.
- Capacidade de processamento dos roteadores:
  - ✓ As tarefas dos roteadores devem ser as mais simples possíveis.
- Princípios dos alg. de roteamento:
  - ✓ Abordagem centralizada
  - ✓ Abordagem descentralizada
- Assuntos avançados:
  - ✓ IPv6
  - ✓ Roteamento multicast
  - ✓ Mobilidade
- Próximo capítulo:
  - ✓ Camada de Enlace: transferência de pacotes entre nós no mesmo enlace ou LAN.