

# Ch. 04 – Camada de Rede

## 4.1 – Introdução e Serviços

### 4.1.1 - Repasse e Roteamento

### 4.1.2 - Modelo de Serviço de Rede

## 4.2 – Redes Datagramas / Circuitos Virtuais

### 4.2.1 - Redes de Circuitos Virtuais

### 4.2.2 - Redes de Datagramas

## 4.3 – Comutador de Camada de Rede

### 4.3.1 – Portas de Entrada

### 4.3.2 – Elemento de Comutação

### 4.3.3 – Portas de Saída

### 4.3.4 - Repasse e Descarte de Pacotes

## ... Ch. 04 – Camada de Rede

### 4.4 – Protocolo IP

#### 4.4.1 – Formato do Pacote

#### 4.4.2 – Endereçamento IPv4

#### 4.4.3 – Protocolo ICMP

#### 4.4.4 – Protocolo IPv6

#### 4.4.5 – Segurança do IP

### 4.5 – Algoritmos de Roteamento

#### 4.5.1 – Algoritmo de Roteamento Link State

#### 4.5.2 – Algoritmos de Roteamento Distance Vector

#### 4.5.3 – Roteamento Hierárquico

## ... Ch. 04 – Camada de Rede

4.6 – Roteamento na Arq. TCP/IP

4.6.1 – Roteamento Intra-AS RIP

4.6.2 - Roteamento Intra-AS OSPF

4.6.3 – Roteamento InterASs BGP

4.7 – Roteamento Broadcast e Multicast

4.7.1 – Roteamento Broadcast

4.7.2 – Roteamento Multicast

# Referências Bibliográficas

- James F. Kurose; Keith W. Ross – Redes de Computadores e a Internet: Uma Abordagem Top-Down – Pearson São Paulo; 6th; 2014; ISBN: 978-85-430-1443-2
- ... Lectures dos autores James F. Kurose; Keith W. Ross  
“[https://gaia.cs.umass.edu/kurose\\_ross/eighth.htm](https://gaia.cs.umass.edu/kurose_ross/eighth.htm)”
- Notas de Aula do Prof. Maurício Magalhães e Eleri Cardozo da Faculdade de Engenharia Elétrica e de Computação (FEEC) da UNICAMP “[www.dca.feec.unicamp.br/~mauricio/~elerj](http://www.dca.feec.unicamp.br/~mauricio/~elerj)”.

## 4 - Camada de Rede / 4.0 – Objetivos e Motivações

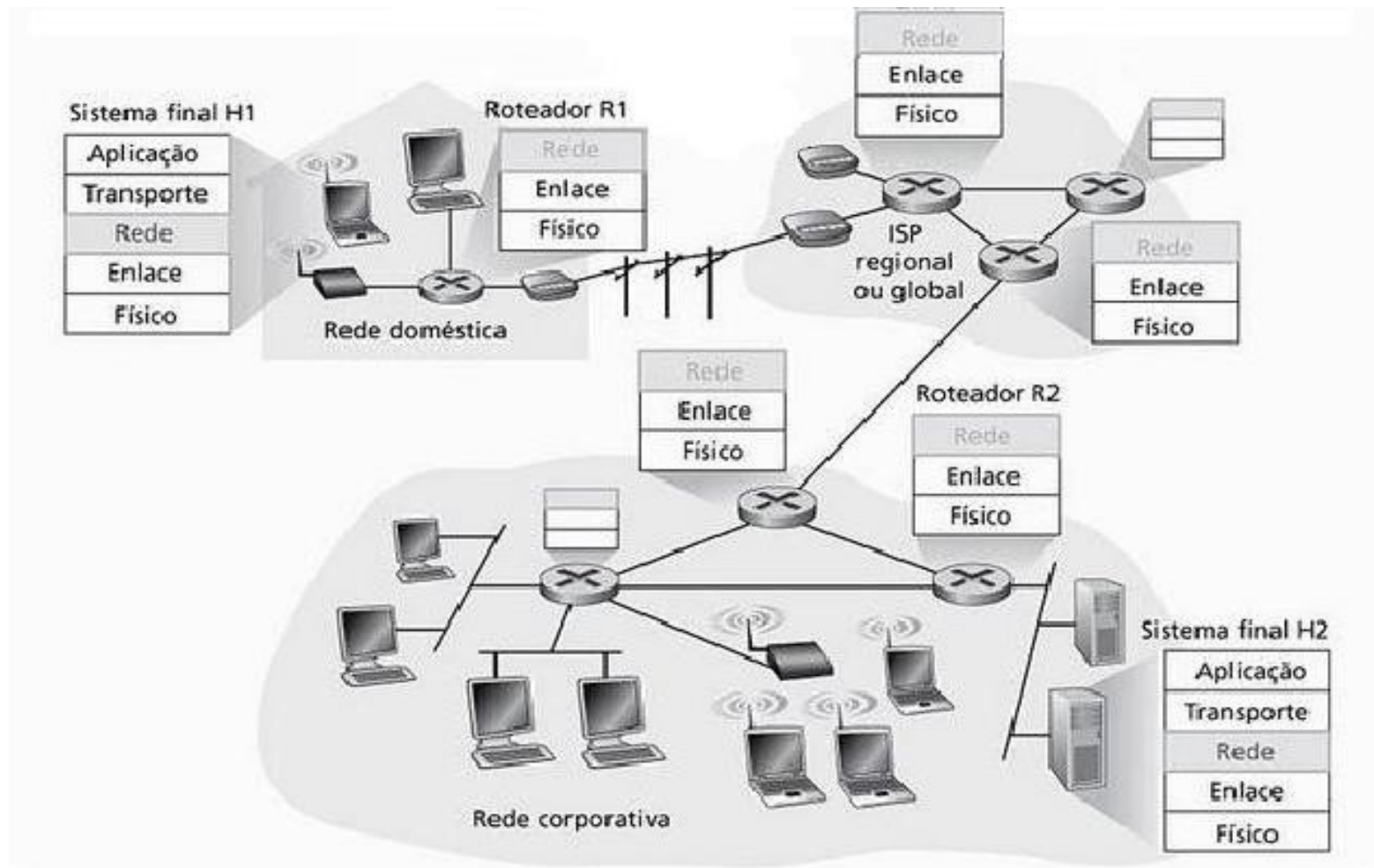
# Objetivos e Motivações

- “**princípios dos serviços**” .. da camada de rede:
- .. modelos de serviço com ou sem conexão;
- .. função de repasse versus função de roteamento;
- .. princípios do funcionamento de um roteador para acomodar uma das principais funções do mesmo, ou seja, roteamento de pacotes.
- .. como acomodar mecanismos que contemplam a escalabilidade, uma vez que o núcleo da rede contém 100s de 1000s de roteadores.
- “**objetivo**” .. entender os aspectos de implementação da camada de rede na Arquitetura TCP/IP ou Arquitetura Internet.
- .. abordar alguns tópicos avançados, p.ex., IPv6 e Mobilidade do Host (aspectos apenas introdutórios).

## 4 - Camada de Rede / 4.1 - Introdução e Serviços

### 4.1 – Introdução e Serviços

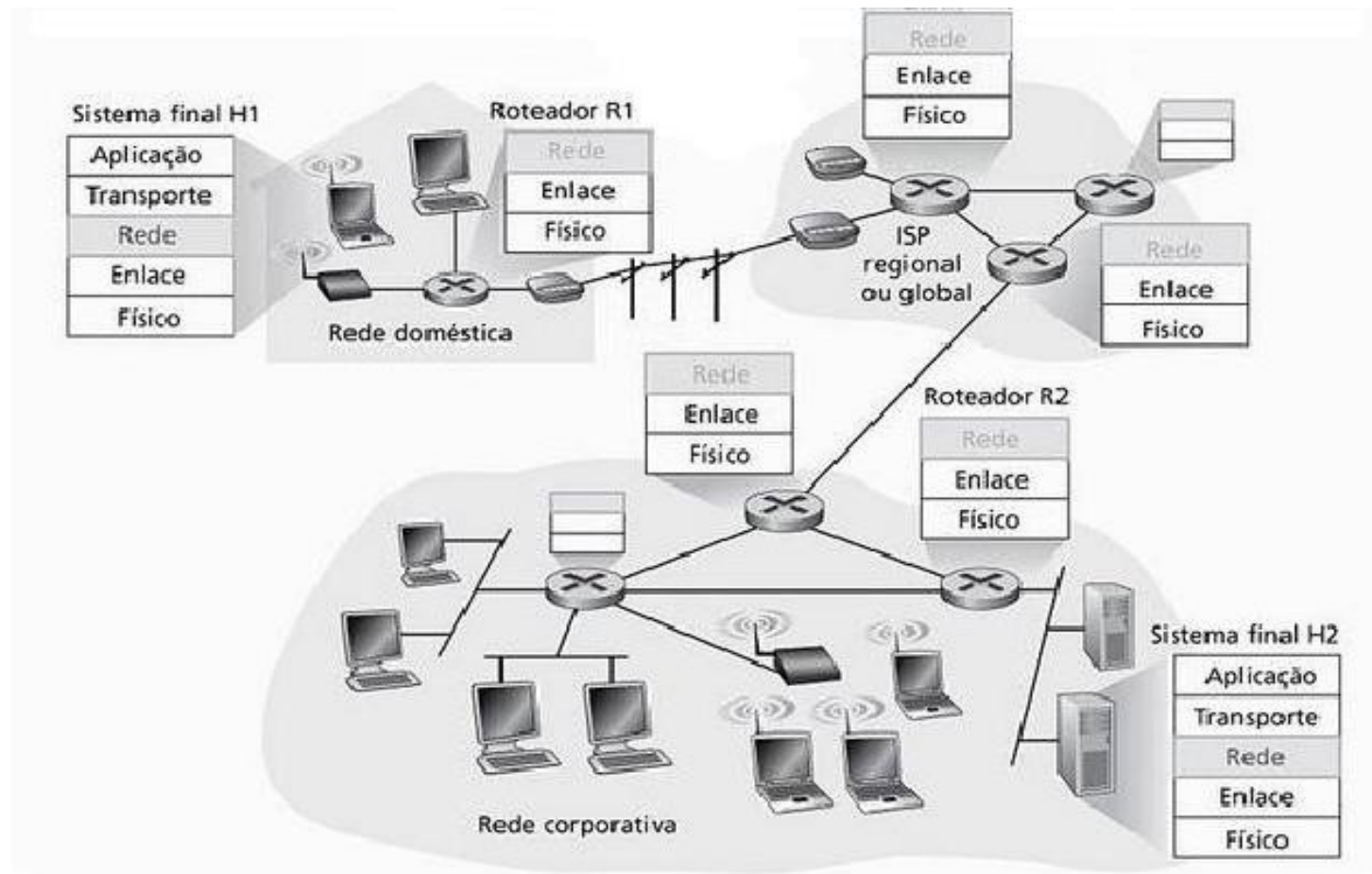
- e.g., seja a troca de dados entre “hosts” H1 e H2 e considere o papel da camada de rede nesses “hosts” e nos roteadores.



## 4 - Camada de Rede / 4.1 - Introdução e Serviços

### ... 4.1 – Introdução e Serviços

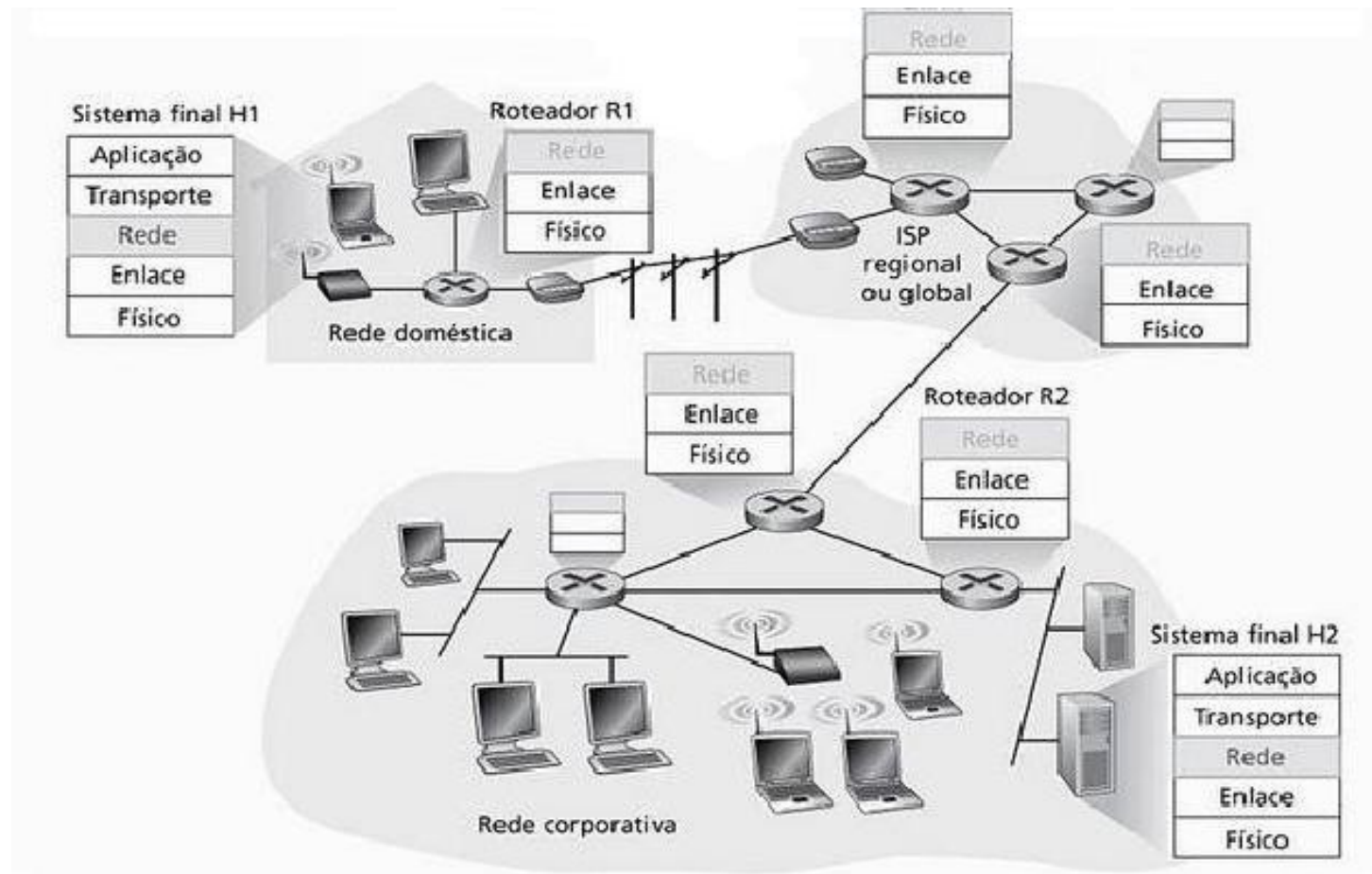
- “**lado remetente**” .. (1) emissor recebe dados do transporte; (2) emissor encapsula segmentos em datagramas e, na sequência, os repassa para a camada de enlace, que por sua vez, repassa a camada física.



## 4 - Camada de Rede / 4.1 - Introdução e Serviços

### ... 4.1 – Introdução e Serviços

- **“lado destinatário”** .. dados recebidos no destinatário são repassados à camada de rede, ou seja, (3) receptor recebe datagramas e extrai os segmentos; (4) .. entrega segmentos à camada de transporte.

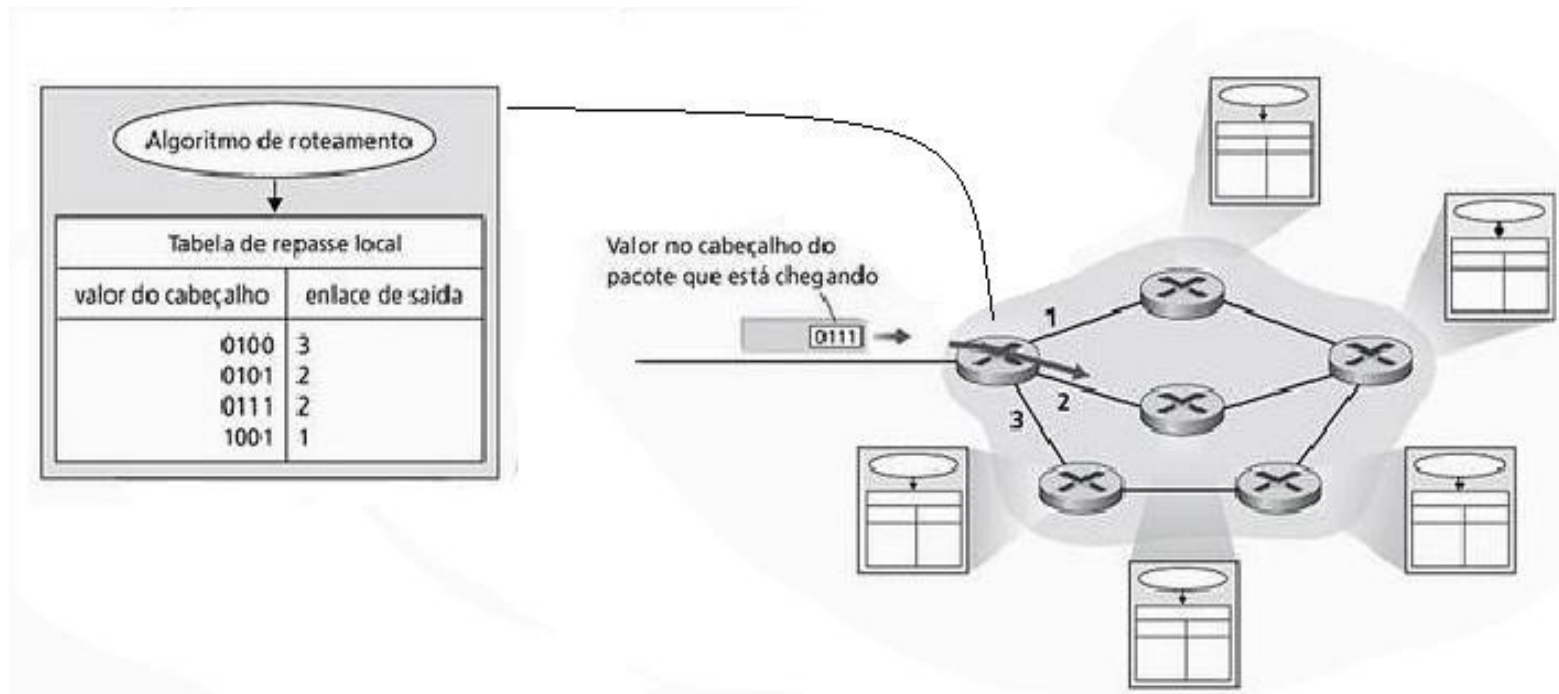




## 4 - Camada de Rede / 4.1- Introdução e Serviços

### 4.1.1 – Repasse e Roteamento

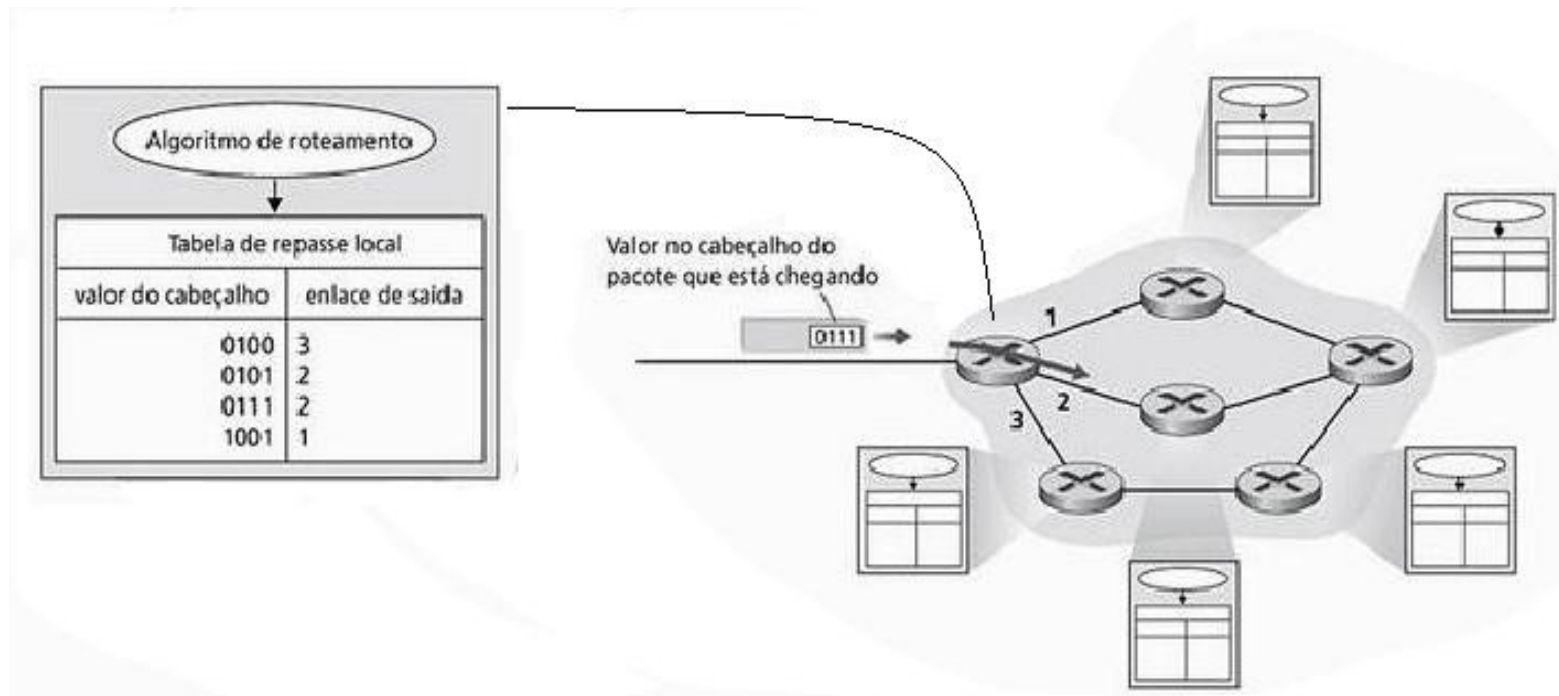
- “**funções**” .. são 02 as principais funções da camada de rede:
- “**repasse**” .. mover pacotes de algum enlace de entrada do roteador para algum enlace de saída do roteador.
- “**roteamento**” .. determinar a rota a ser seguida pelos pacotes desde o emissor até o receptor através dos algoritmos de roteamento.



## 4 - Camada de Rede / 4.1 - Introdução e Serviços

### ... 4.1 – Introdução e Serviços

- “**roteadores**” .. papel primordial de como comutadores de rede ..
- examinam campos de cabeçalho dos datagramas IP.
- executam protocolos da camada de rede em cada roteador.
- repassam datagramas de enlaces de entrada para enlaces de saída.



## 4 - Camada de Rede / 4.1- Introdução e Serviços

### ... 4.1.1 – Repasse e Roteamento

- “**roteador**” .. transfere um pacote de uma interface de enlace de entrada para uma interface de enlace de saída conforme o valor de campo específico no cabeçalho do pacote da camada de rede.
- “**comutador da camada de enlace**” ou “**switch**” .. transfere um quadro de um enlace de entrada para um de saída tendo por base o valor de um campo no cabeçalho do quadro da camada de enlace.
- “**algoritmo de roteamento**” .. centralizado ou descentralizado ..
- “**centralizado**” .. algoritmo executado em um local central e que descarrega informações de roteamento em cada um dos roteadores.
- “**descentralizado**” .. algoritmo executado localmente em cada um dos elementos de comutação de pacotes para gerar a tabela de rotas.

## 4 - Camada de Rede / 4.1- Introdução e Serviços

### ... 4.1.1 – Repasse e Roteamento

- Obs.: ... algumas camadas de redes podem acomodar uma 3ª função, ou seja, estabelecimento de conexão na camada de rede.
- e.g., Arquiteturas de Redes como ATM e Frame Relay exigem que roteadores ao longo de um caminho escolhido estabeleçam uma conexão antes da troca de dados entre emissor e receptor.
- ... para tanto, os roteadores trocam mensagens de controle para estabelecer a conexão antes que os pacotes de dados fluam pelo caminho previamente estabelecido.
- ... ou seja, ao longo de todo o caminho pelo qual os dados passam, há a garantia de que os recursos reservados são suficientes para suportar o fluxo de dados entre emissor e receptor.

## 4 - Camada de Rede / 4.1- Introdução e Serviços

### 4.1.2 – Modelo de Serviço de Rede

- “**modelo de serviço do enlace**” .. que modelo de serviço é o melhor para transportar datagramas do remetente ao destinatário !?
- “**modelo de serviço**” - define as características do transporte de dados fim a fim entre uma borda e a outra da rede, ou seja, entre sistemas finais remetente e destinatário.
- “datagramas individuais” ou “fluxo de datagramas” !?
- Alguns serviços oferecidos pela Camada de Rede ..
- “**entrega garantida**” - assegura que o pacote mais cedo ou tarde chegará ao seu destino independente do tempo.
- “**entrega garantida com atraso limitado**” - assegura a entrega de um pacote, mas o faz dentro de certas restrições de tempo.

## 4 - Camada de Rede / 4.1- Introdução e Serviços

### ... 4.1.2 – Modelo de Serviço de Rede

- Outros serviços oferecidos pela Camada de Rede:
- **“entrega de pacotes na ordem”** .. pacotes são entregues no destinatário na ordem em que foram enviados/transmitidos.
- **“largura de banda mínima garantida”** .. emula o comportamento de um enlace de transmissão com uma taxa específica, p.ex., 1 Mbps.
- **“jitter máximo garantido”** .. garante que a intervalo de tempo entre a transmissão de 02 pacotes sucessivos no remetente seja igual ao intervalo de tempo entre o recebimento dos 02 pacotes.
- **“serviços de segurança”** .. provê a codificação da carga útil de todos os datagramas trocados entre remetente e destinatário.
- **“Camada de Rede da Arq. TCP/IP”** .. **“melhor esforço”**, ou seja, não há a garantia de temporização, ordem ou entrega de pacotes.

## 4 - Camada de Rede / 4.1- Introdução e Serviços

### ... 4.1.2 – Modelo de Serviço de Rede

- Arquitetura de Rede ATM .. define e implementa modelos de serviço que vão além do serviço de melhor esforço da Arq. TCP/IP ..
- **“Constant Bit Rate” (CBR)** .. provê um fluxo de pacotes (células) em uma conexão virtual cujas propriedades são iguais às de um hipotético enlace de transmissão dedicado de largura de banda fixa.

Arquitetura da rede	Modelo de serviço	Garantia de largura de banda	Garantia contra perda	Ordenamento	Temporização	Indicação de congestionamento
Internet	Melhor esforço	Nenhuma	Nenhuma	Qualquer ordem possível	Não mantida	Nenhuma
ATM	CBR	Taxa constante garantida	Sim	Na ordem	Mantida	Não ocorrerá congestionamento
ATM	ABR	Mínima garantida	Nenhuma	Na ordem	Não mantida	Indicação de congestionamento

## 4 - Camada de Rede / 4.1- Introdução e Serviços

### ... 4.1.2 – Modelo de Serviço de Rede

- Arquitetura de Rede ATM .. define e implementa modelos de serviço que vão além do serviço de melhor esforço da Arq. TCP/IP ..
- “**Available Bit Rate**” .. provê um fluxo de células a uma taxa mínima (MCR - Minimum Cell Transmission Rate), mas não permite a reordenação das células embora as células possam ser perdidas.
- ... caracterizado como um serviço de melhor esforço ligeiramente melhorado em razão da possível perda de células / pacotes.

Arquitetura da rede	Modelo de serviço	Garantia de largura de banda	Garantia contra perda	Ordenamento	Temporização	Indicação de congestionamento
Internet	Melhor esforço	Nenhuma	Nenhuma	Qualquer ordem possível	Não mantida	Nenhuma
ATM	CBR	Taxa constante garantida	Sim	Na ordem	Mantida	Não ocorrerá congestionamento
ATM	ABR	Mínima garantida	Nenhuma	Na ordem	Não mantida	Indicação de congestionamento



## 4 - Camada de Rede / 4.2 - Rede de Datagramas / Circuitos Virtuais

### 4.2 – Redes de Datagramas / Circuitos Virtuais

- **“rede de datagrama”** .. serviço sem conexão na camada de rede, ou seja, os roteadores utilizam o endereço de destino presente no pacote para encaminhar o pacote ao longo de um caminho ..
- .. caminho contempla conjunto de enlaces mais nós intermediários (roteadores), estes últimos responsáveis pelo roteamento dos pacotes.
- .. pacotes distintos com mesmo endereço de destino podem seguir caminhos diferentes em função de mudanças nas tabelas de roteamento face a reconfiguração de enlaces e/ou roteadores.

## 4 - Camada de Rede / 4.2 - Rede de Datagramas / Circuitos Virtuais

### ... 4.2 – Redes de Datagramas / Circuitos Virtuais

- **“rede de circuitos virtuais”** .. serviço com conexão na camada de rede, ou seja, um caminho virtual é estabelecido ao longo dos vários caminhos disponíveis entre remetente e destinatário.
- ... caminho virtual acomoda conjunto de enlaces mais nós intermediários (roteadores), ou seja, os roteadores utilizam o nro. do circuito virtual que está associado a cada pacote para roteá-lo.

## 4 - Camada de Rede / 4.2 - Rede de Datagramas / Circuitos Virtuais

### ... 4.2 – Redes de Datagramas / Circuitos Virtuais

- **“serviços da camada de rede” vs “camada de transporte”**
- serviços na camada de rede são de “host” para “host”, enquanto serviços na camada de transporte são de aplicação para aplicação.
- em todas as arquiteturas importantes de redes de computadores, ou a camada de rede oferece serviço orientado a conexão ou serviço não orientado a conexão, mas muito raramente ambos os serviços.
- **“serviço orientado por conexão”** .. trata-se de um serviço fundamentalmente diferente na camadas de rede e de transporte, ...
- ... na camada de transporte é implementado nos “hosts” enquanto que na camada de rede é implementado nos roteadores do núcleo da rede bem como nos “hosts” finais (fundamentalmente diferentes) !!

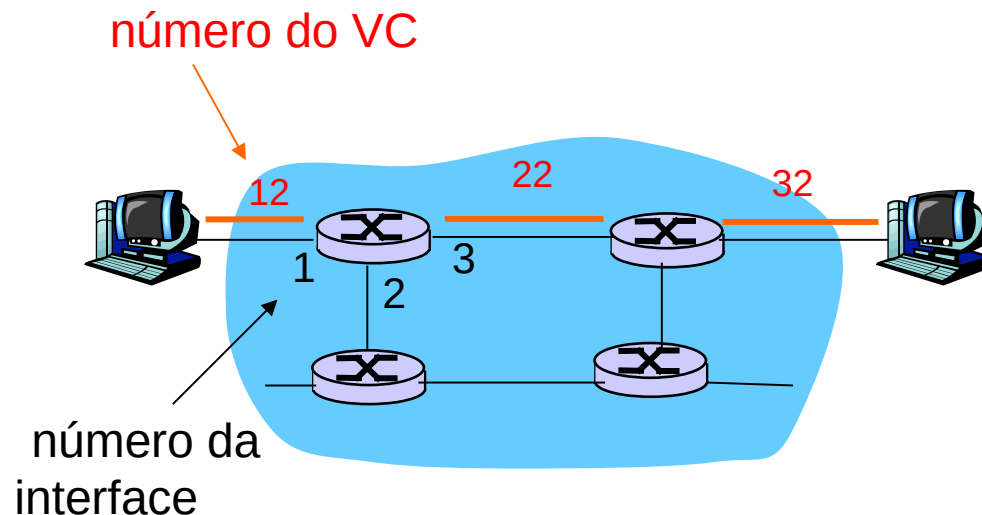
## 4 - Camada de Rede / 4.2 - Rede de Datagramas / Circuitos Virtuais

### 4.2.1 – Redes de Circuitos Virtuais

- **“circuito virtual”** - caminho da origem ao destino comporta-se como um circuito telefônico com respeito ao desempenho e necessita de ações da rede ao longo do caminho ..
- .. necessário estabelecer a conexão antes que os dados possam fluir, assim como fechar a conexão para liberar os recursos reservados.
- .. cada pacote carrega identificador do circuito virtual e não o endereço do “host” de destino para permitir que o roteador efetue o roteamento.
- .. cada roteador no caminho entre remetente e destinatário mantém “estado” para cada conexão na qual participa.
- .. recursos do enlace e roteador (largura de banda, buffers) são alocados ao circuito, oferecendo serviço de qualidade previsível, desde que os recursos reservados satisfaçam os requisitos da aplicação.

## 4 - Camada de Rede / 4.2 - Rede de Datagramas / Circuitos Virtuais ... 4.2.1 – Redes de Circuitos Virtuais

- “**circuito virtual**” - consiste de um caminho da origem ao destino; número de circuito virtual; número de cada enlace ao longo do caminho; e de entradas e tabelas de repasse nos roteadores.
- .. pacote de um VC carrega número do VC no cabeçalho;
- .. nro. do VC (Virtual Circuit) no pacote pode ser alterado por cada roteador em razão da alocação de identificadores ser “in-loco” e não alocação global (VC extraído da Tabela de Repasse).



## 4 - Camada de Rede / 4.2 - Rede de Datagramas / Circuitos Virtuais

### ... 4.2.1 – Redes de Circuitos Virtuais

- Como o Roteador determina o novo nro de CV para um pacote ?
- .. tabela de repasse de cada roteador inclui a tradução de nro. de CV, p.ex., tabela de repasse de  $R_1$  contempla CV de entrada e saída.
- .. para um pacote que chega com nro. de CV = 12, o mesmo terá o campo correspondente alterado para 22 (linha 1 da tabela).

Interface de entrada	Nº do CV de entrada	Interface de saída	Nº do CV de saída
1	12	2	22
2	63	1	18
3	7	2	17
1	97	3	87
...	...	...	...

## 4 - Camada de Rede / 4.2 - Rede de Datagramas / Circuitos Virtuais

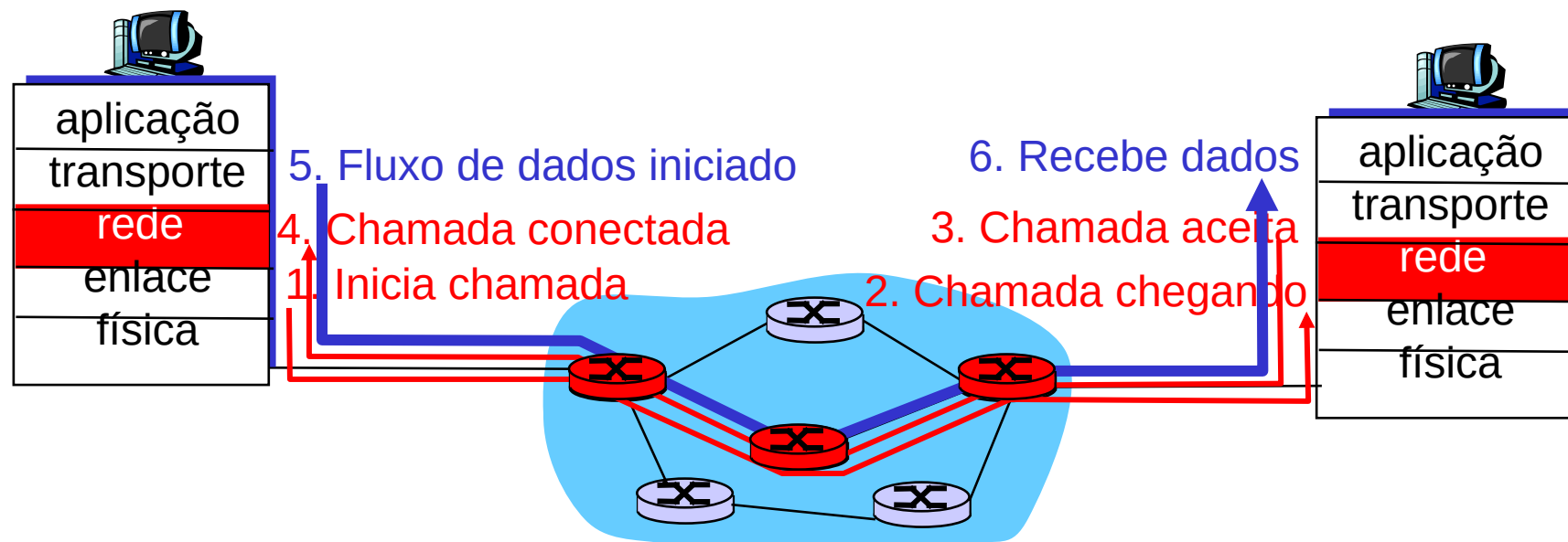
### ... 4.2.1 – Redes de Circuitos Virtuais

- .. roteadores mantêm “estado” para cada conexão, assim é possível associar nros. de CVs com os nros. das interfaces da saída.
- .. sem a manutenção de estado, a consistência entre nro. de CV e interface para a qual deve ser encaminhado um pacote é invalidada.

Interface de entrada	Nº do CV de entrada	Interface de saída	Nº do CV de saída
1	12	2	22
2	63	1	18
3	7	2	17
1	97	3	87
...	...	...	...

## 4 - Camada de Rede / 4.2 - Rede de Datagramas / Circuitos Virtuais ... 4.2.1 – Redes de Circuitos Virtuais

- São 03 as fases para identificação de um CV:
- (1) “**estabelecimento do CV**” .. camada de rede determina o caminho entre remetente e destinatário, ou seja, série de enlaces e roteadores pelos quais todos os pacotes do circuito virtual são transmitidos.
- (2) “**transferência de dados**” .. pacotes fluem ao longo do CV.

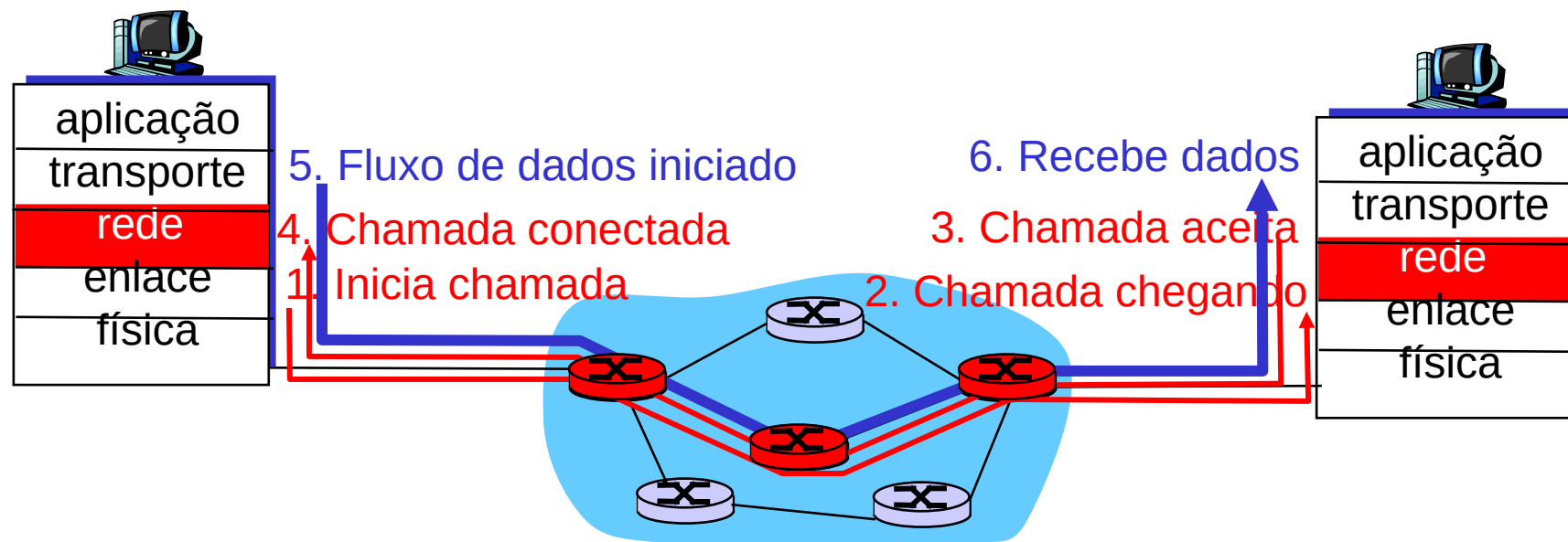




## 4 - Camada de Rede / 4.2 - Rede de Datagramas / Circuitos Virtuais

### ... 4.2.1 – Redes de Circuitos Virtuais

- São 03 as fases para identificação de um CV:
- (3) “**encerramento do CV**” .. camada de rede informa o destinatário o encerramento da conexão e procede com a atualização das tabelas de repasse de todos os roteadores do circuito virtual.



## 4 - Camada de Rede / 4.2 - Rede de Datagramas / Circuitos Virtuais

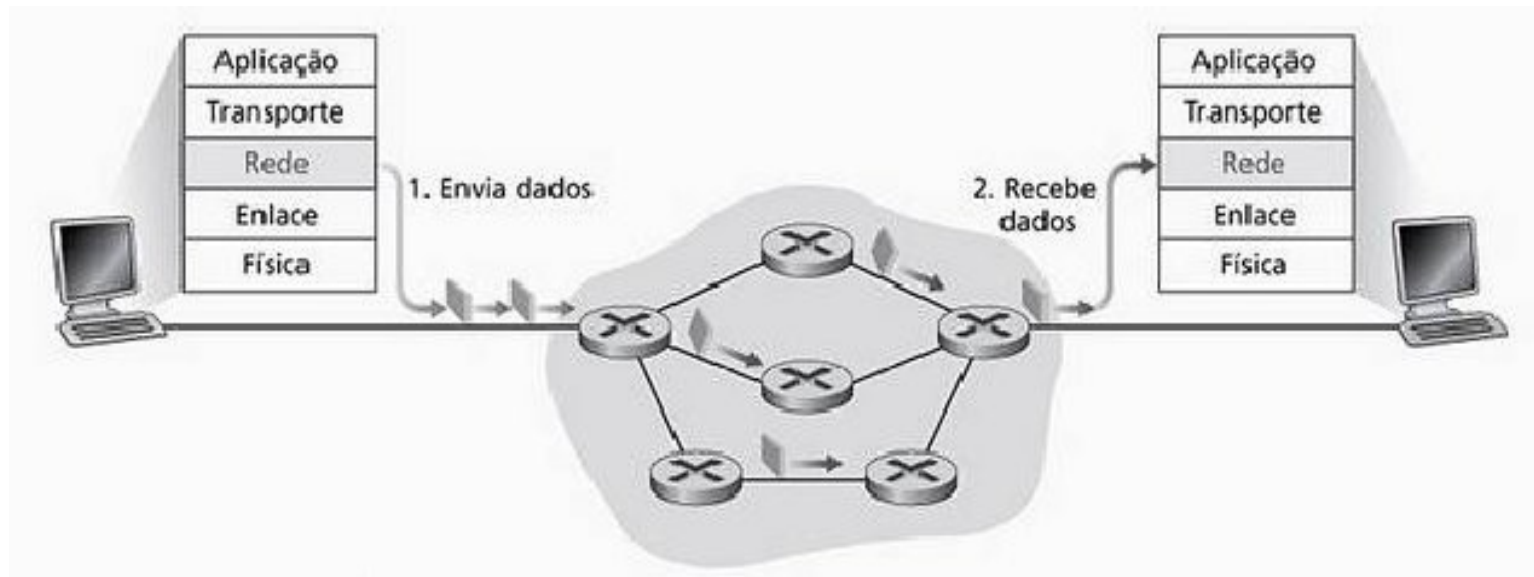
### ... 4.2.1 – Redes de Circuitos Virtuais

- **“CV na Camada de Rede”** .. roteadores ao longo do caminho entre 02 “hosts” estão envolvidos no estabelecimento do CV e cada roteador está ciente de todos os CV que por ele passam.
- ... mensagens trocadas entre os roteadores para estabelecer o circuito virtual são conhecidas como “mensagens de sinalização”.
- **“Conexão na Camada de Transporte”** .. na Arq. TCP/IP, uma conexão na camada de transporte envolve apenas os sistemas finais, ou seja, os roteadores no núcleo estão alheios ao fato.
- ... durante o estabelecimento de uma conexão na camada de transporte, os 02 “hosts” determinam os parâmetros de sua conexão.

## 4 - Camada de Rede / 4.2 - Rede de Datagramas / Circuitos Virtuais

### 4.2.2 – Redes de Datagramas

- “**rede de datagramas**” - serviço sem conexão na camada de rede, ou seja, os roteadores utilizam o endereço de destino presente no pacote para encaminhar o pacote ao longo de um caminho.
- .. ou seja, os roteadores não mantêm estado sobre conexões fim-a-fim uma vez que não estabelecem conexões fim-a-fim.
- .. pacotes enviados pelo remetente para o destinatário podem seguir caminhos diferentes em decorrência de alterações no roteamento.



## 4 - Camada de Rede / 4.2 - Rede de Datagramas / Circuitos Virtuais

### ... 4.2.2 – Redes de Datagramas

- “**tabela de repasse**” .. mapeia endereços de destino, p.ex., “110010 ... 00” para interfaces de enlace, p.ex., interface “0”.

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

## 4 - Camada de Rede / 4.2 - Rede de Datagramas / Circuitos Virtuais

### ... 4.2.2 – Redes de Datagramas

- “**tabela de repasse**” .. armazena o prefixo do endereço de destino, assim, caso haja concordância com o prefixo, o roteador transmite pacote para o enlace associado àquele prefixo.
- .. quando há várias concordâncias de prefixos nas entradas, o roteador utiliza a regra de concordância do “**prefixo mais longo**”.

Prefixo do endereço	Interface de enlace
11001000 00010111 00010	0
11001000 00010111 00011000	1
11001000 00010111 00011	2
senão	3

## 4 - Camada de Rede / 4.2 - Rede de Datagramas / Circuitos Virtuais

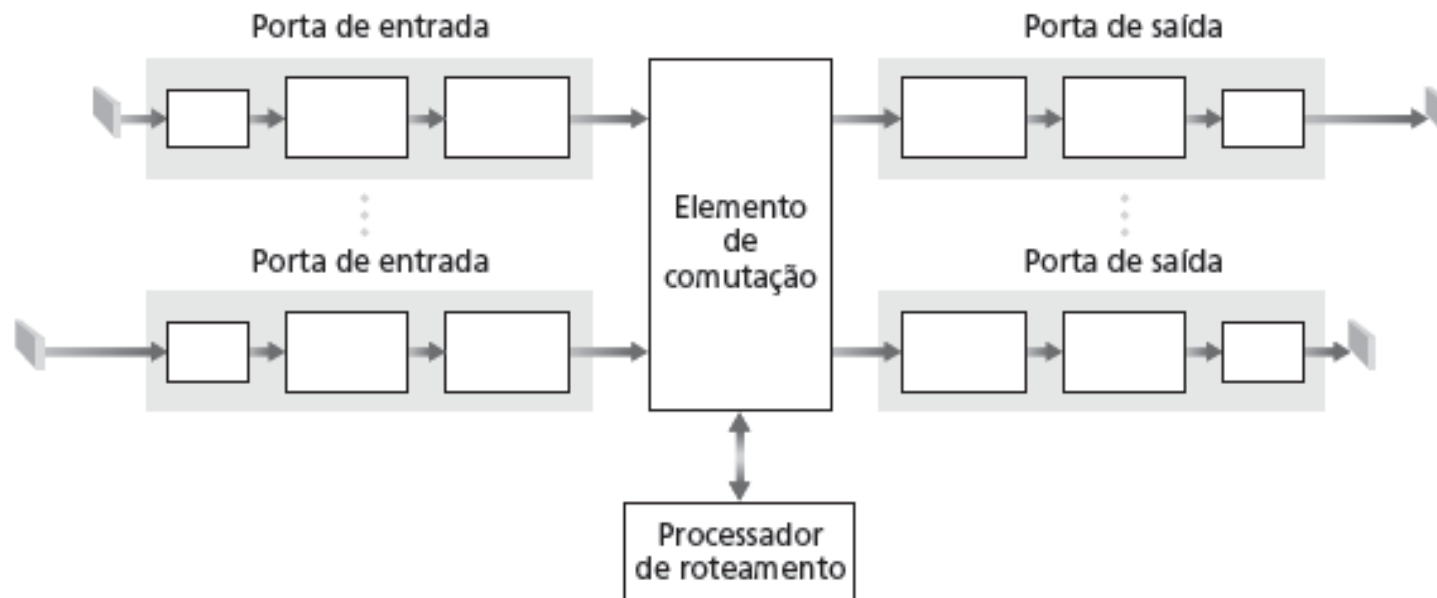
### ... 4.2.2 – Redes de Datagramas

- “**modificação das tabelas de repasse**” - algoritmos de roteamento normalmente atualizam a tabela de repasse em intervalos de tempo previamente definidos.
- .. embora redes de datagramas não mantenham nenhuma informação de estado de conexão, ainda assim, mantêm informação de estado de repasse em suas tabelas de repasse.
- .. escala temporal dentro do qual estas informações de estado são modificadas é relativamente lenta (10s minutos).

## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

### 4.3 – Comutador de Camada de Rede

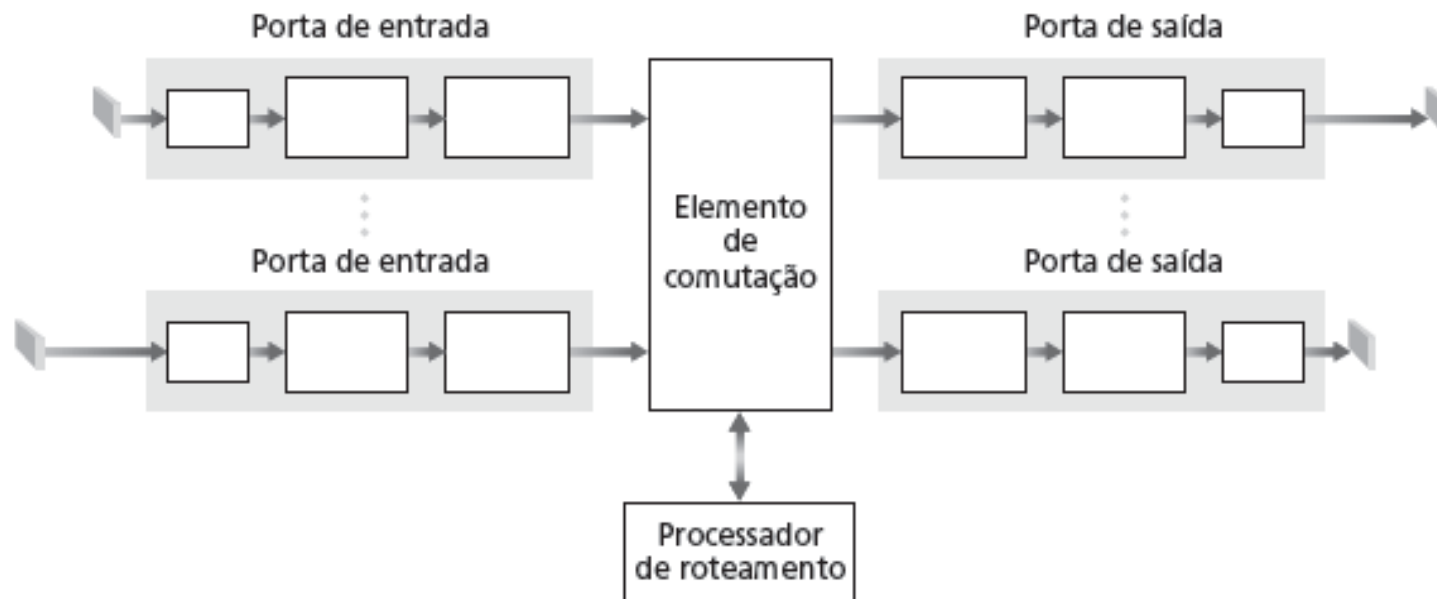
- 02 funções principais do Roteador:
- .. executar algoritmo/protocolo de roteamento (OSPF, RIP, BGP).
- .. repassar datagramas dos enlaces de entrada para os de saída.
- **“principais componentes”** da arquitetura de um roteador ..



## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

### ... 4.3 – Comutador de Camada de Rede

- **“portas de entrada”** ..
- .. desempenha funções da camada física de um enlace físico.
- .. realiza as funções da camada de enlace necessárias para interoperação com as funções da camada de enlace do lado destinatário.
- .. examina e repassa o pacote ao elemento de comutação do roteador de modo que o mesmo seja inserido na porta de saída.

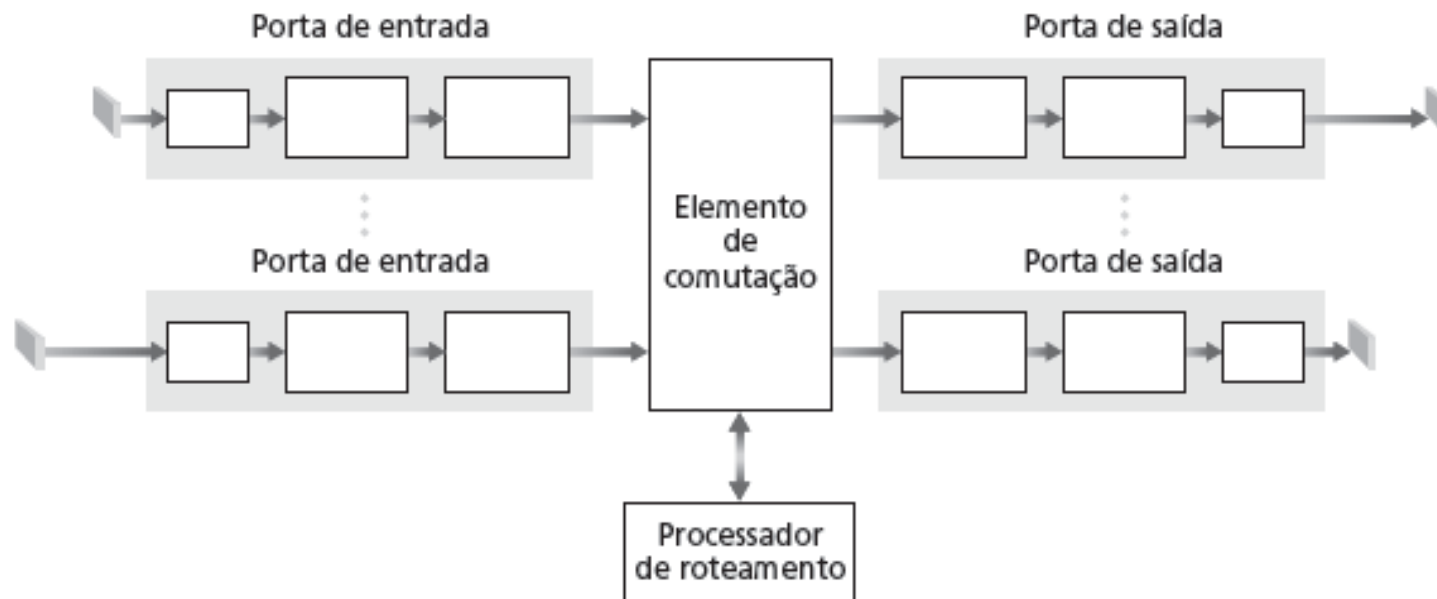




## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

### ... 4.3 – Comutador de Camada de Rede

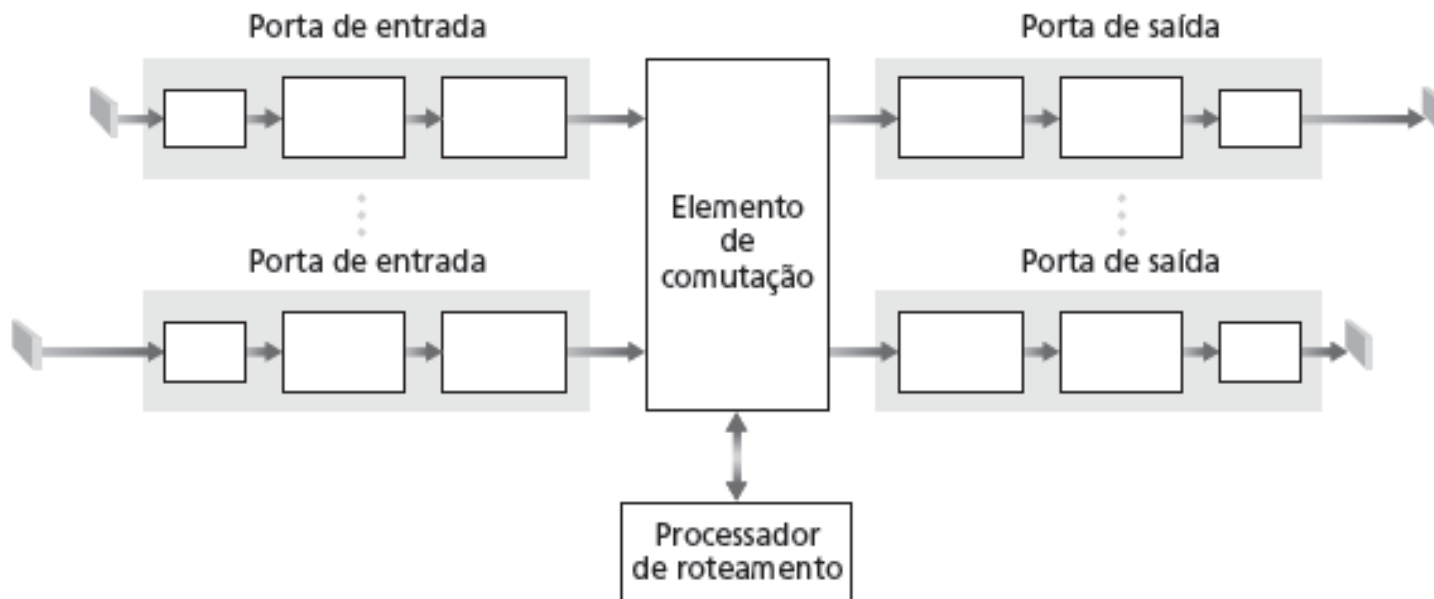
- **“elemento de comutação”** ..
- .. integralmente contido no interior do roteador, conecta as portas de entrada do roteador às suas portas de saída.
- .. responsável pelo repasse entre as portas de entrada e portas de saída segundo diferentes abordagens de comutação.



## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

### ... 4.3 – Comutador de Camada de Rede

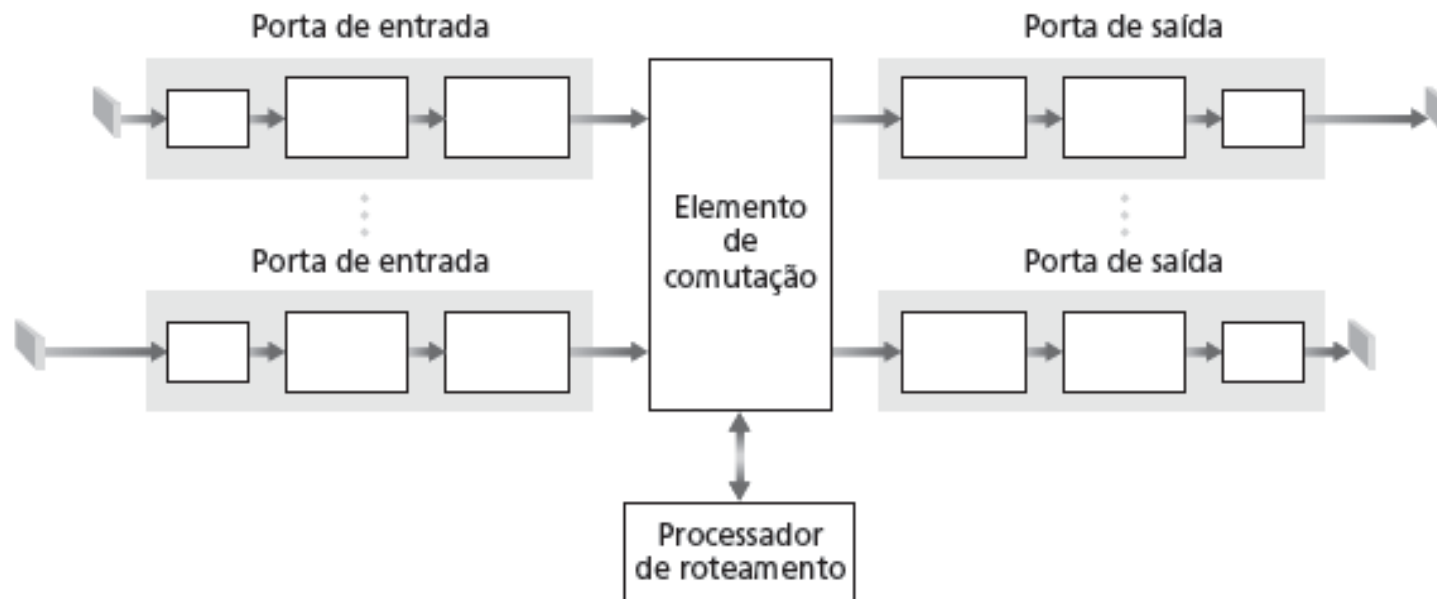
- “**processador do roteamento**” ..
- (1) .. executa os protocolos de roteamento.
- (2) .. executa funções de gerenciamento de rede.
- (3) .. mantém as informações de roteamento e tabelas de repasse.



## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

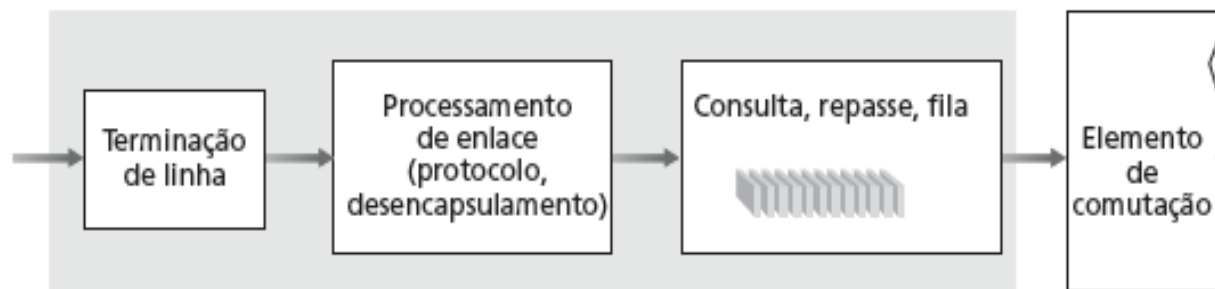
### ... 4.3 – Comutador de Camada de Rede

- “**portas de saída**” .. .. realiza o inverso da camada de enlace e da camada física da porta de entrada (discutido anteriormente).
- .. armazena os pacotes que foram repassados a ela através do elemento de comutação e, então, os transmite até o enlace de saída.
- .. no caso de enlace bidirecional, a porta de saída está emparelhada com a porta de entrada para aquele enlace na mesma placa de linha.



## 4.3.1 – Portas de Entrada

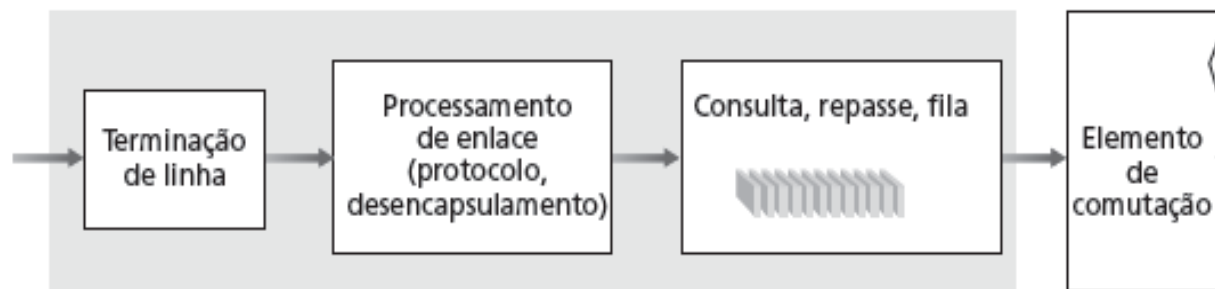
- “**portas de entrada**” .. implementam as camadas física e de enlace associadas a um enlace de entrada individual do roteador.
- ... em muitos roteadores é aqui que se determina a porta de saída para a qual o pacote será repassado pelo elemento de comutação, que se dá com base nas informações da tabela de repasse.
- ... tabela de repasse é calculada pelo processador de roteamento, mas a cópia da tabela de repasse para cada porta de entrada, permite que as decisões de repasse sejam tomadas localmente.
- ... repasse descentralizado evita a criação de um gargalo de processamento de repasse em 01 único ponto no roteador.



## 4 - Camada de Rede / 4.3 - Computador de Camada de Rede

### ... 4.3.1 – Portas de Entrada

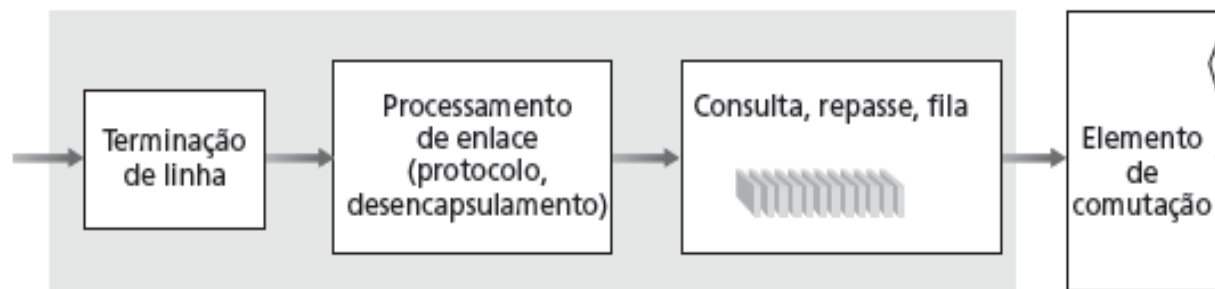
- e.g., seja um “desktop no papel de roteador” e no qual as portas de entrada não tem capacidade de processamento .. o que esperar ?
- “**usual**” .. porta de entrada repassa o pacote para o processador, que então, realiza o exame da tabela de repasse e transmite o pacote a porta de saída apropriada.
- ... conceitualmente a busca na tabela de repasse consiste em procurar o registro mais longo compatível com o endereço de destino.
- ... por questões práticas, é desejável que o processamento na porta de entrada possa operar na velocidade da linha, ou seja, o exame possa ser feito antes de receber o próximo pacote na porta.



## 4 - Camada de Rede / 4.3 - Computador de Camada de Rede

### ... 4.3.1 – Portas de Entrada

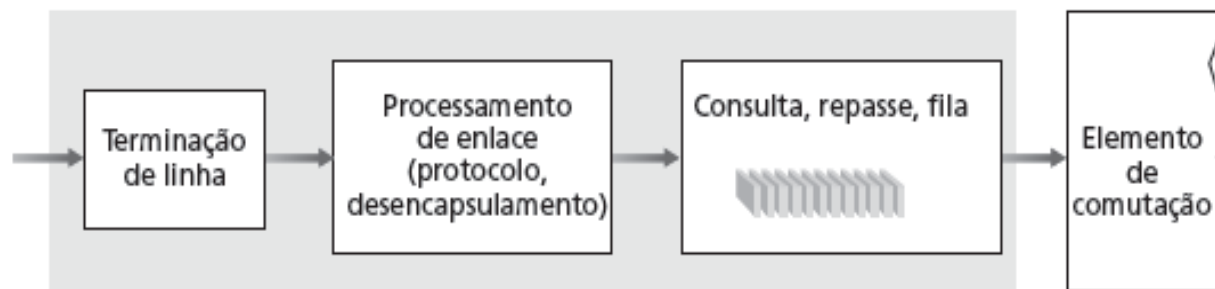
- e.g., considere um enlace OC48 de 2,5 Gbps e pacotes de 256 bytes, ou seja, são 1.220.000 pacotes/seg. chegando na porta de entrada, logo, são necessárias 1.220.000 consultas/segundo.
- $2,5 * 10^9 \text{ bps} / 8 \text{ bits} = 0.3125 * 10^9 \text{ bytes por seg.}$ , ou seja,  $0.3125 * 10^9 / 256 = 0.001220703 * 10^9 = 1.220.703 \text{ pcts por segundo}$
- “**problema**” .. impossível efetuar uma busca linear em uma tabela de repasse extensa, assim técnicas mas rápidas fazem-se necessárias.
- .. registros da tabela de repasse organizados em uma estrutura de árvore permite examinar o registro da tabela de repasse em “N” etapas, onde “N” é o nro. de bits do endereço cuja busca será realizada.



## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

### ... 4.3.1 – Portas de Entrada

- e.g., considere um enlace OC48 de 2,5 Gbps e pacotes de 256 bytes, ou seja, são 1.220.000 pacotes/seg. chegando a porta de entrada, logo são necessárias 1.220.000 consultas/segundo.
- ... mesmo assim, a velocidade de consulta por busca binária não é rápida o suficiente para os requisitos de roteamento de “backbone”
- ... p.ex., em memória com tempo de acesso de 40 nano seg., consegue-se 781.250 consultas/seg. >> bem menos que 1.220.000 de consultas de endereços ou pacotes por segundo.



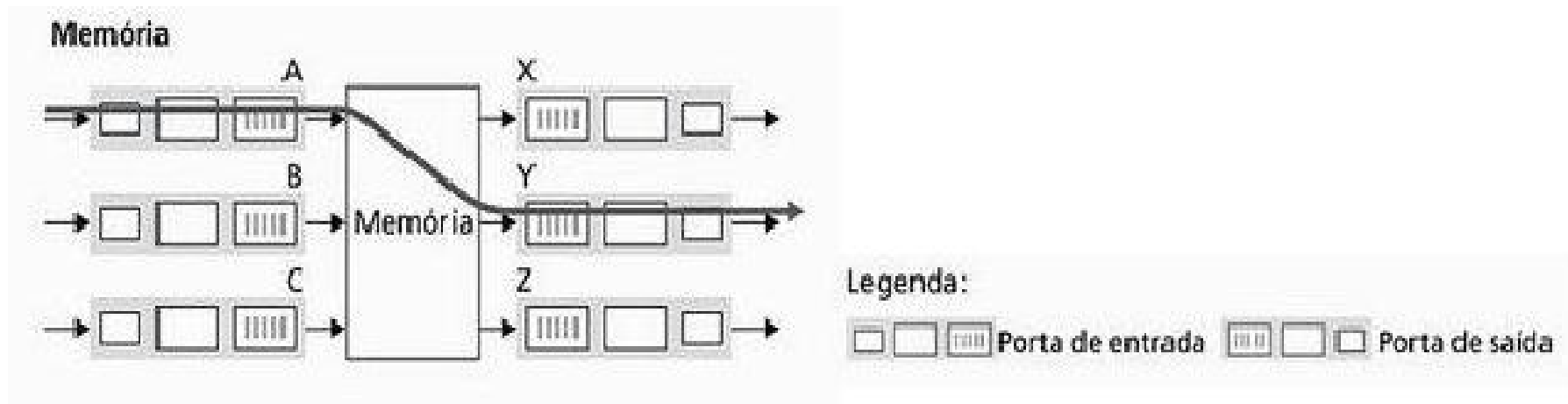
## ... 4.3.1 – Portas de Entrada

- e.g., em memória com tempo de acesso de 40 nano seg., consegue-se 781.250 consultas/seg. >> bem menos que 1.220.000 de consultas de endereços ou pacotes por segundo.
- .. memória de conteúdo endereçável permite que um endereço de 32 bits seja retornado da tabela em tempo essencialmente constante.
- .. Cisco 8500 contempla (CAM Content Addressable Memory) de 64K para cada porta de entrada do roteador.
- .. outra forma de acelerar a consulta é manter registros da tabela de repasse que foram acessados recentemente armazenados na “cache”.
- Obs.: ... assim que uma porta de saída é selecionada por meio da consulta, o pacote pode ser repassado para o elemento de comutação uma vez que o mesmo não esteja ocupado.



## 4.3.2 – Elemento de Comutação

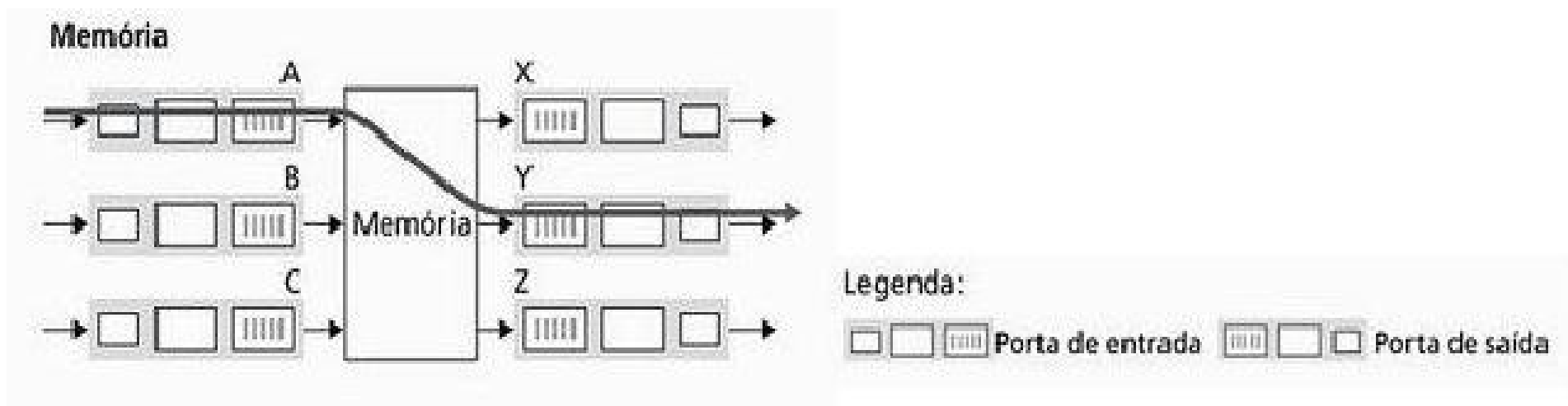
- **“elemento de comutação”** .. responsável pela comutação (repasse) do pacote de uma porta de entrada para uma porta de saída.
- **“comutação por memória”** .. portas de entrada e porta de saída funcionam como dispositivos tradicionais de entrada/saída de um sistema operacional tradicional.
- ... processador de roteamento copia pacote do “buffer” da porta de entrada para a memória e após consulta a tabela de repasse efetua a cópia da memória para o “buffer” da porta de saída.



## 4 - Camada de Rede / 4.3 - Computador de Camada de Rede

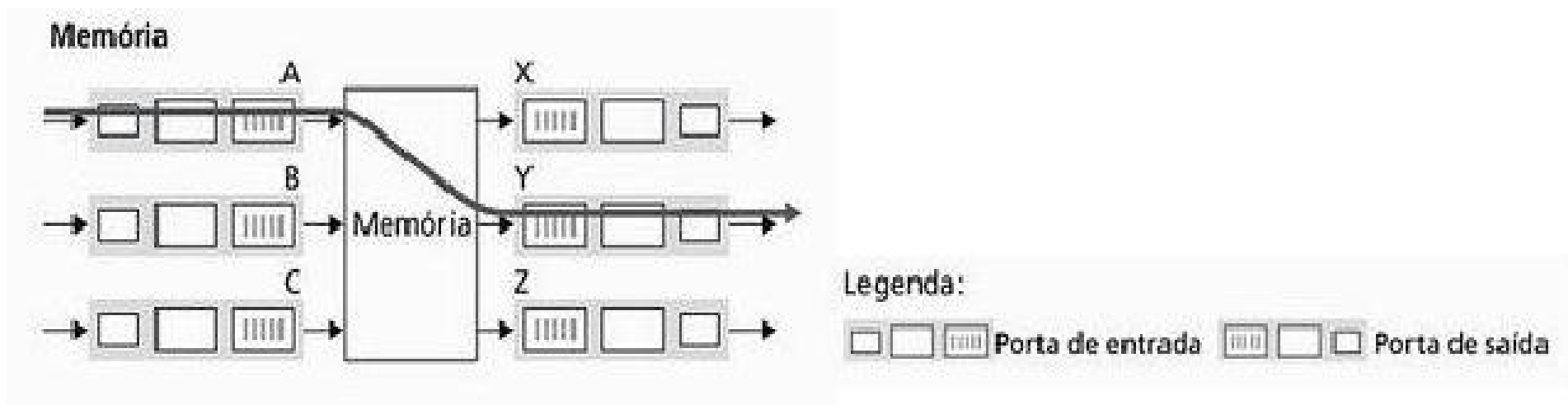
### 4.3.2 – Elemento de Comutação

- “**comutação por memória**” - portas de entrada e porta de saída funcionam como dispositivos tradicionais de entrada/saída de um sistema operacional tradicional.
- ... se “B” pacotes/seg. podem ser escritos ou lidos na memória, a vazão total de repasse com que os pacotes são transferidos de portas de entrada para portas de saída deve ser menor que “ $B/2$ ”.



## 4.3.2 – Elemento de Comutação

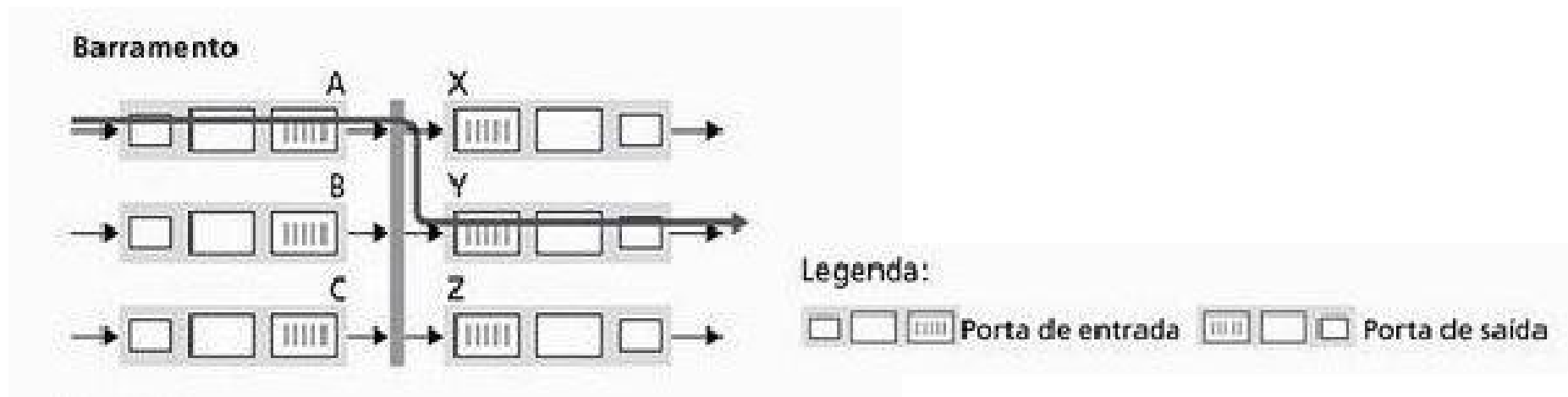
- “**comutação por memória**” - portas de entrada e porta de saída funcionam como dispositivos tradicionais de entrada/saída de um sistema operacional tradicional.
- ... roteadores modernos contemplam processadores nas placas de linha de entrada responsáveis por realizar a consulta do endereço bem como o armazenamento na memória.



## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

### ... 4.3.2 – Elemento de Comutação

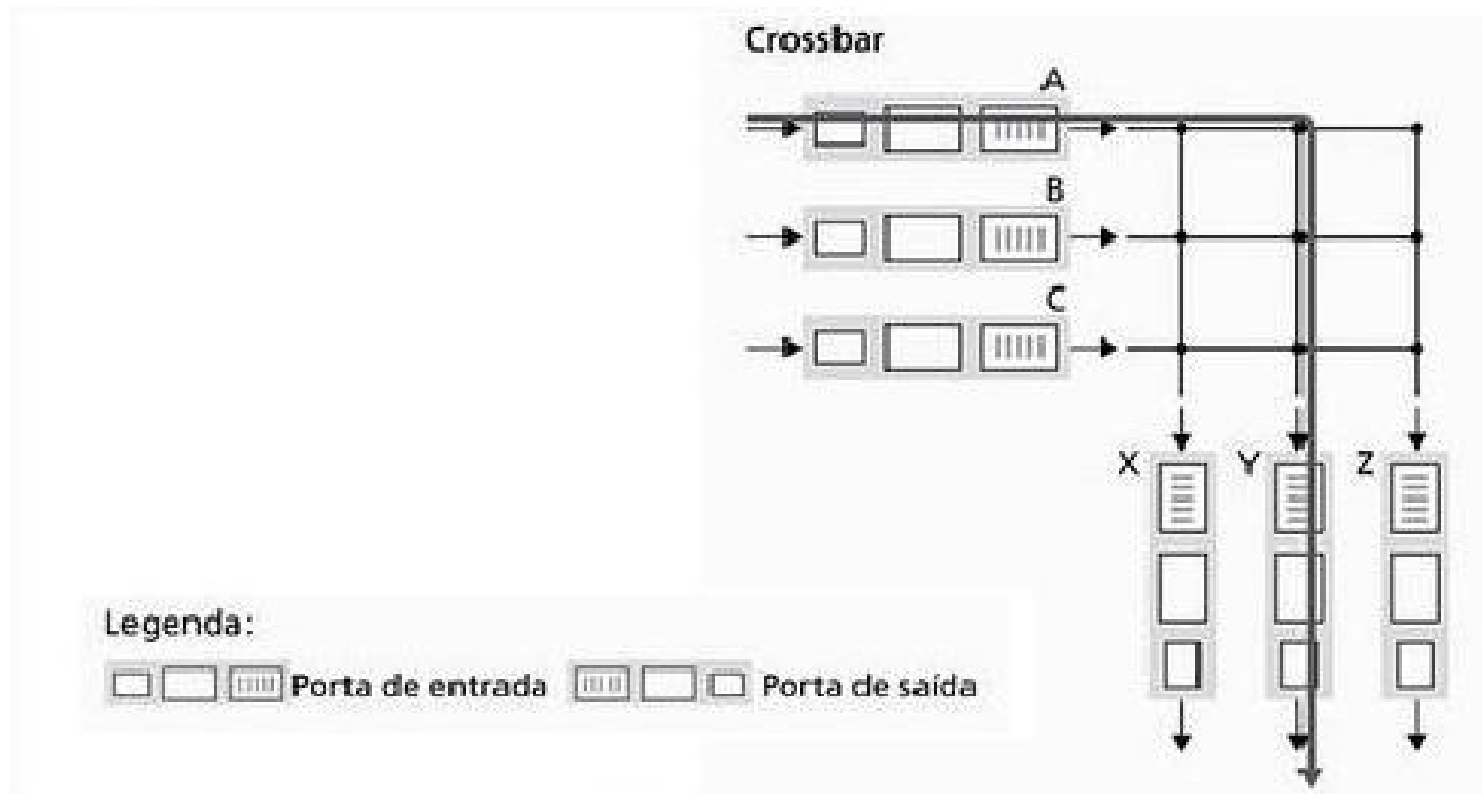
- “**comutação por barramento**” .. portas de entrada transferem um pacote diretamente para a porta de saída por um barramento compartilhado sem a intervenção do processador de roteamento.
- .. enlaces disputam acesso ao barramento >> velocidade da comutação é limitada pela largura de banda do barramento.
- e.g., Cisco 5600 comuta pacotes por um barramento de 32 Gbps - velocidade suficiente para roteadores de acesso e corporativos.



## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

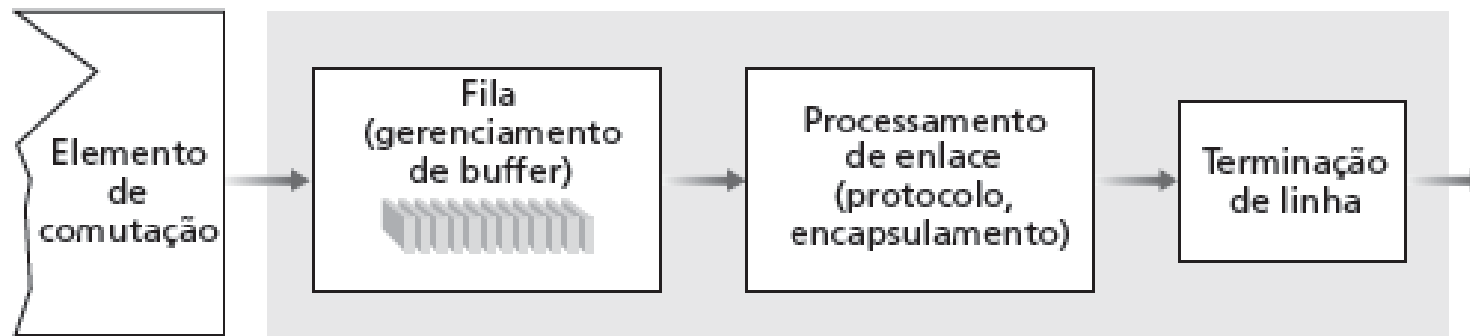
### ... 4.3.2 – Elemento de Comutação

- “**comutação crossbar**” .. rede de interconexão de “ $2*n$ ” barramentos conectam “ $n$ ” portas de entrada com “ $n$ ” portas de saída.
- e.g., CISCO 12000 utiliza rede de interconexão com até 60 Gbps pelo elemento de comutação e pode apresentar taxas ainda maiores quando os pacotes comutados apresentam mesmo tamanho.



### 4.3.3 – Portas de Saída

- “**portas de saída**” .. pega os pacotes que foram armazenados na memória da porta de saída e os transmite pelo enlace de saída.
- ... processamento de enlace e a terminação da linha são as funcionalidades de camada de enlace e de camada física do lado remetente que interagem com a porta de saída do enlace de saída.



## 4.3.4 – Repasse e Descarte de Pacotes

- Ao examinarmos as funcionalidades da porta de entrada e da porta de saída, verifica-se que filas de pacotes podem se formar tanto nas portas de entrada como nas portas de saída.
- ... perda de pacotes pode ocorrer a medida que as filas ficam maiores, ou seja, a falta de buffers disponíveis no roteador implica em descarte.
- ... local exato do descarte, seja nas filas das portas de entrada, seja nas filas das portas de saída dependerá da carga de tráfego, da velocidade relativa do elemento de comutação e da taxa de linha.
- ... quando se comenta que pacotes são perdidos ou descartados dentro da rede, nos referimos aos descartes / perdas nestas filas.
- **“taxa do elemento de comutação”** - taxa na qual o elemento de comutação movimenta pacotes de portas de entrada para de saída.

## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

### ... 4.3.4 – Repasse e Descarte de Pacotes

- e.g., considere um roteador no qual a taxa de linhas de entrada e taxa de linhas de saída sejam idênticas e no qual temos “n” portas de entrada e “n” portas de saída.
- ... suponha que a taxa do elemento de comutação seja no mínimo “n” vezes as taxas das linhas de entrada e de saída.
- ... considere o caso em que os pacotes que chegam a cada uma das “n” portas de entrada sejam destinados à mesma porta de saída.





## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

### ... 4.3.4 – Repasse e Descarte de Pacotes

- e.g., considere um roteador no qual a taxa de linhas de entrada e taxa de linhas de saída sejam idênticas e no qual temos “n” portas de entrada e “n” portas de saída.
- ... considerando que a porta de saída transmite 01 pacote por unidade de tempo, os “n” pacotes que chegarem terão que entrar na fila (esperar) para transmissão no enlace de saída.



## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

### ... 4.3.4 – Repasse e Descarte de Pacotes

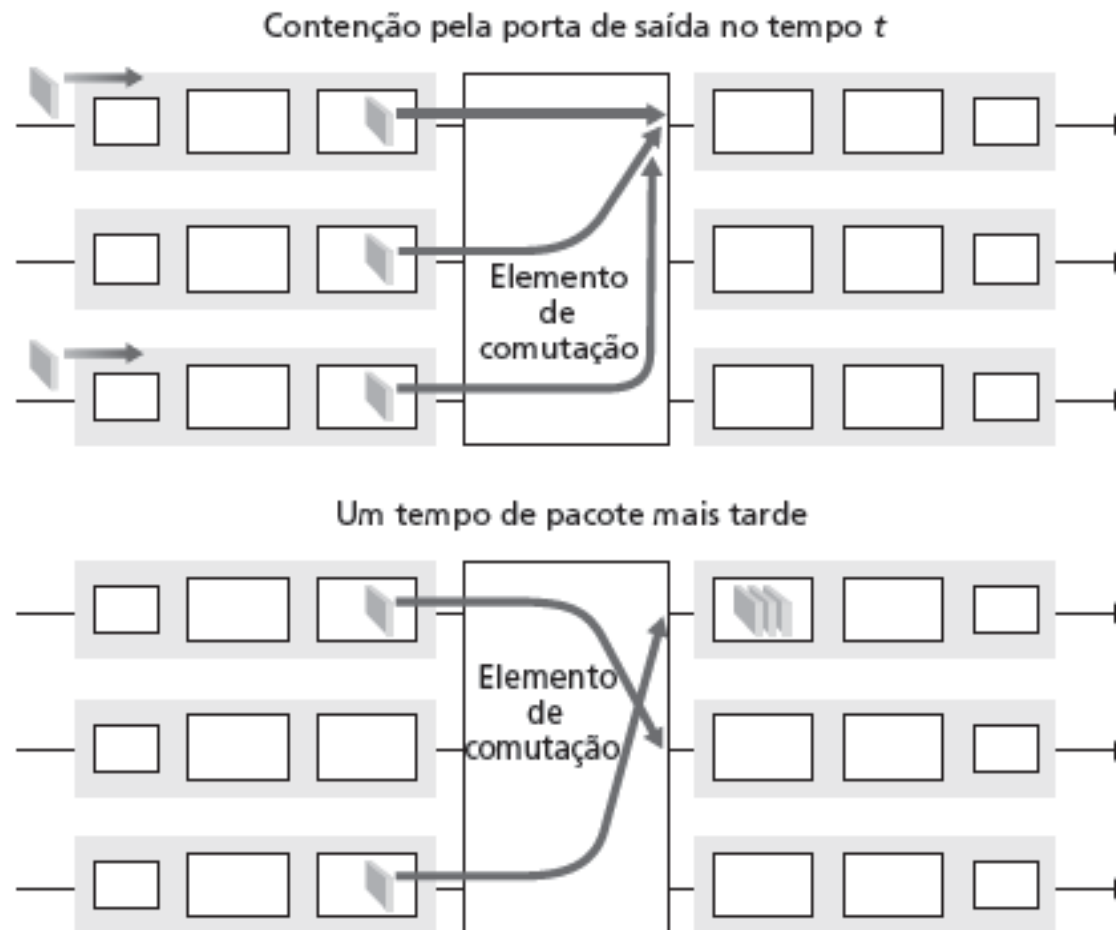
- e.g., considere um roteador no qual a taxa de linhas de entrada e taxa de linhas de saída sejam idênticas e no qual temos “n” portas de entrada e “n” portas de saída;
- ... se mais “n” pacotes chegarem no intervalo de tempo para transmitir um único pacote, a fila pode se tornar muito grande a ponto de exaurir o espaço de memória na porta de saída >> **“descarte de pacotes”**.



## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

### ... 4.3.4 – Repasse e Descarte de Pacotes

- “**buffering**” .. taxa de chegada via comutador ou taxa da linha de saída.
- enfileiramento e perda devidos a estouro de “buffer” na porta de saída!



## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

### ... 4.3.4 – Repasse e Descarte de Pacotes

- RFC 3439 .. tamanho médio do buffer deve ser igual à RTT “típico”, p.ex.,  $250 \text{ ms} * “C”$ , onde “C” é a capacidade do enlace em “bps”
- e.g., enlace de 10 Gbps, teremos “buffer” de 2,5 Gbits, ou seja, “buffer” suficiente para armazenar o que se transmite em 250 ms.
- **“recomendação recente”** .. com N fluxos, o “buffer” deve ser:
- $RTT * C / \text{square}(N)$  onde “N” é nro. de fluxos TCP.
- **“política de escalonamento”** .. desempenham papel crucial na garantia da qualidade de serviço da camada em questão.

## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

### ... 4.3.4 – Repasse e Descarte de Pacotes

- “**política de escalonamento**” .. desempenham papel crucial na garantia da qualidade de serviço da camada em questão.
- .. uma consequência da fila de porta de saída é que o escalonador de pacotes deve transmitir um pacote dentre os que estão na fila.
- .. políticas como FCFS (First Come First Served) ou WFQ (Weighted Fair Queuing) são exemplos de políticas para escolha do pacote.



## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

### ... 4.3.4 – Repasse e Descarte de Pacotes

- ... de modo semelhante, se não houver memória suficiente para armazenar um pacote que está chegando, será preciso descartar ou remover um ou mais pacotes já enfileirados.
- ... tais políticas são também conhecidas como políticas de descarte do final de fila ou algoritmos de gerenciamento ativo de fila (Active Queue Management – AQM).



## 4 - Camada de Rede / 4.3 - Computador de Camada de Rede

### ... 4.3.4 – Repasse e Descarte de Pacotes

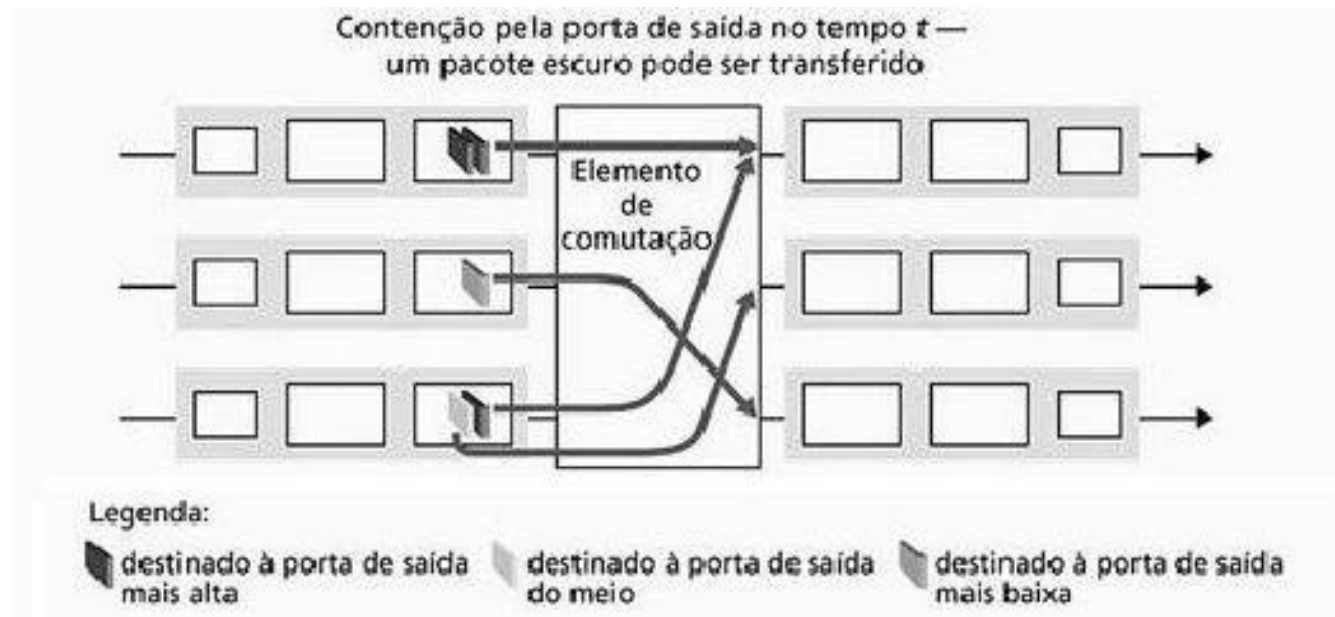
- ... se um elemento de comutação não for suficientemente veloz para transmitir sem atraso todos os pacotes que chegam através dele, então a fila nas portas de entrada poderá crescer.
- ... pois terão que esperar a sua vez para serem transferidas através do elemento de comutação até a porta de saída.



## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

### ... 4.3.4 – Repasse e Descarte de Pacotes

- e.g., considere um elemento de comutação “crossbar” onde:
- (1) velocidades dos enlaces são idênticas.
- (2) tempo para encaminhar um pacote da porta de entrada para porta de saída é o mesmo para receber um pacote na porta de entrada.
- (3) pacotes são transferidos de uma fila de entrada para uma fila de saída segundo a política FCFS (First Come First Served).

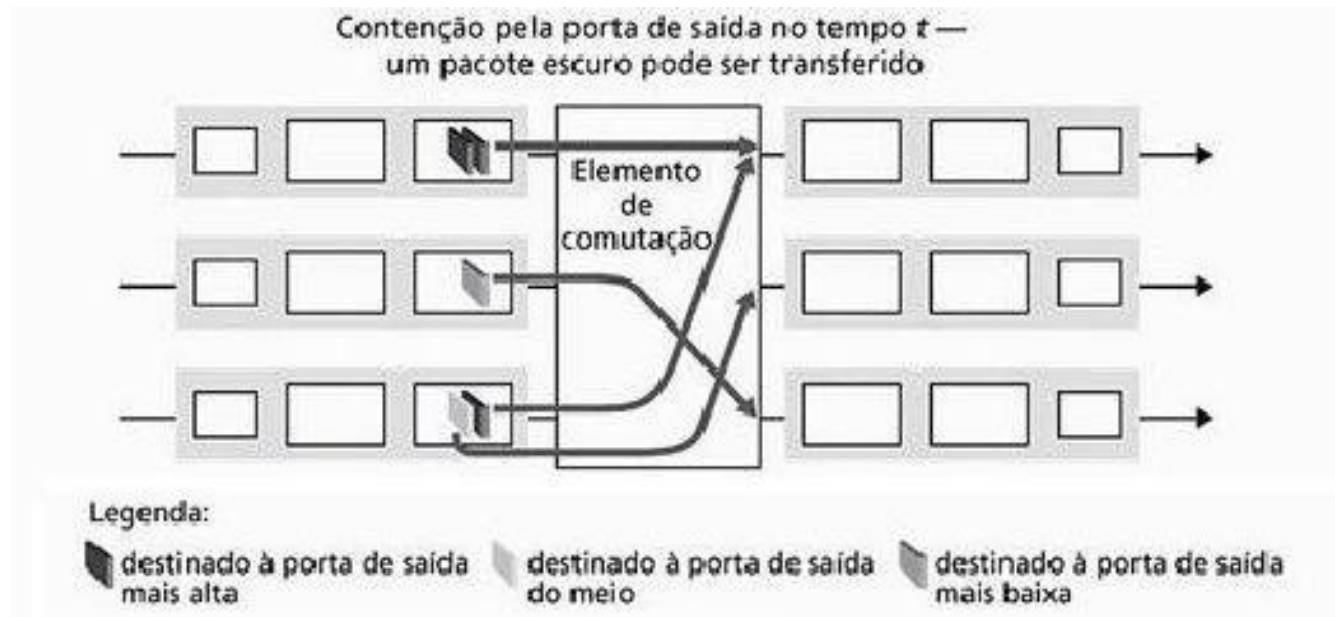




## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

### ... 4.3.4 – Repasse e Descarte de Pacotes

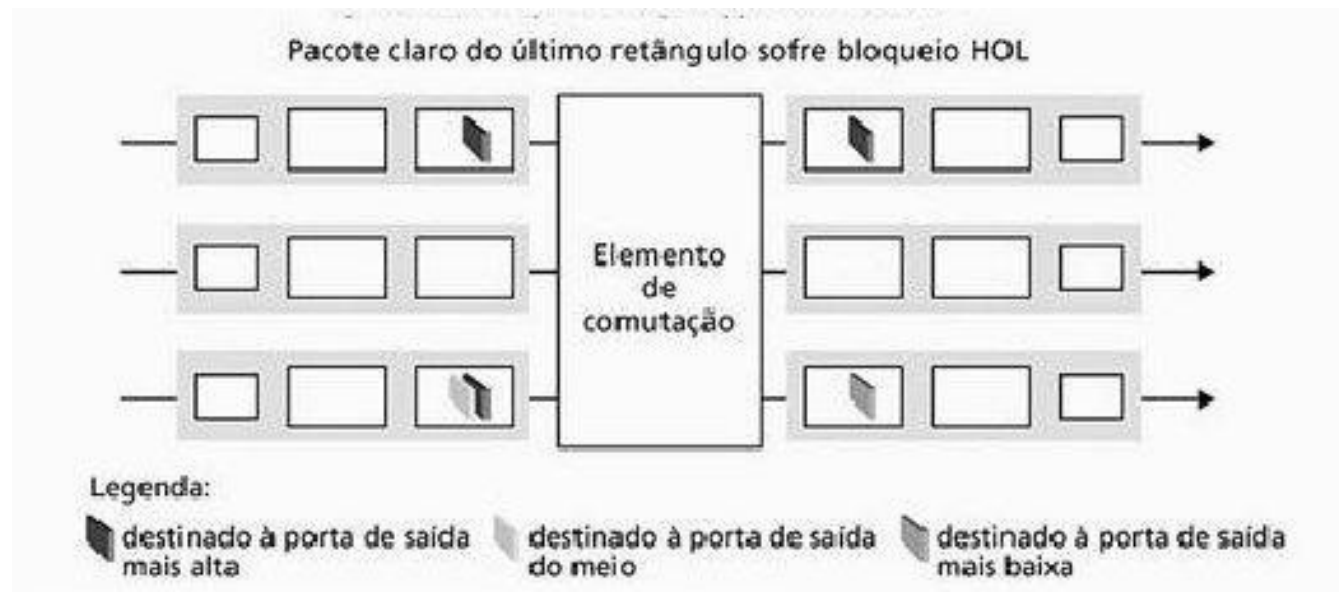
- ... se 02 pacotes que estão a frente das 02 filas de entrada forem destinados à mesma fila de saída, então um dos pacotes ficará bloqueado e terá de esperar na fila de entrada.
- ... suponha que o elemento de comutação escolha transferir o pacote que está à frente da fila mais alta à esquerda, assim o pacote mais escuro na fila mais baixa à esquerda tem que esperar.



## 4 - Camada de Rede / 4.3 - Comutador de Camada de Rede

### ... 4.3.4 – Repasse e Descarte de Pacotes

- “**bloqueio de cabeça de fila**” ou “Head Of the Line – HOL Blocking” ..
- .. pacote que está na fila de entrada deve esperar pela transferência através do elemento de comutação porque ele está bloqueado por um outro pacote na cabeça da fila.
- ... isto demonstra que o comprimento da fila de entrada cresce sem limites, ou seja, teremos significativas perdas de pacotes.



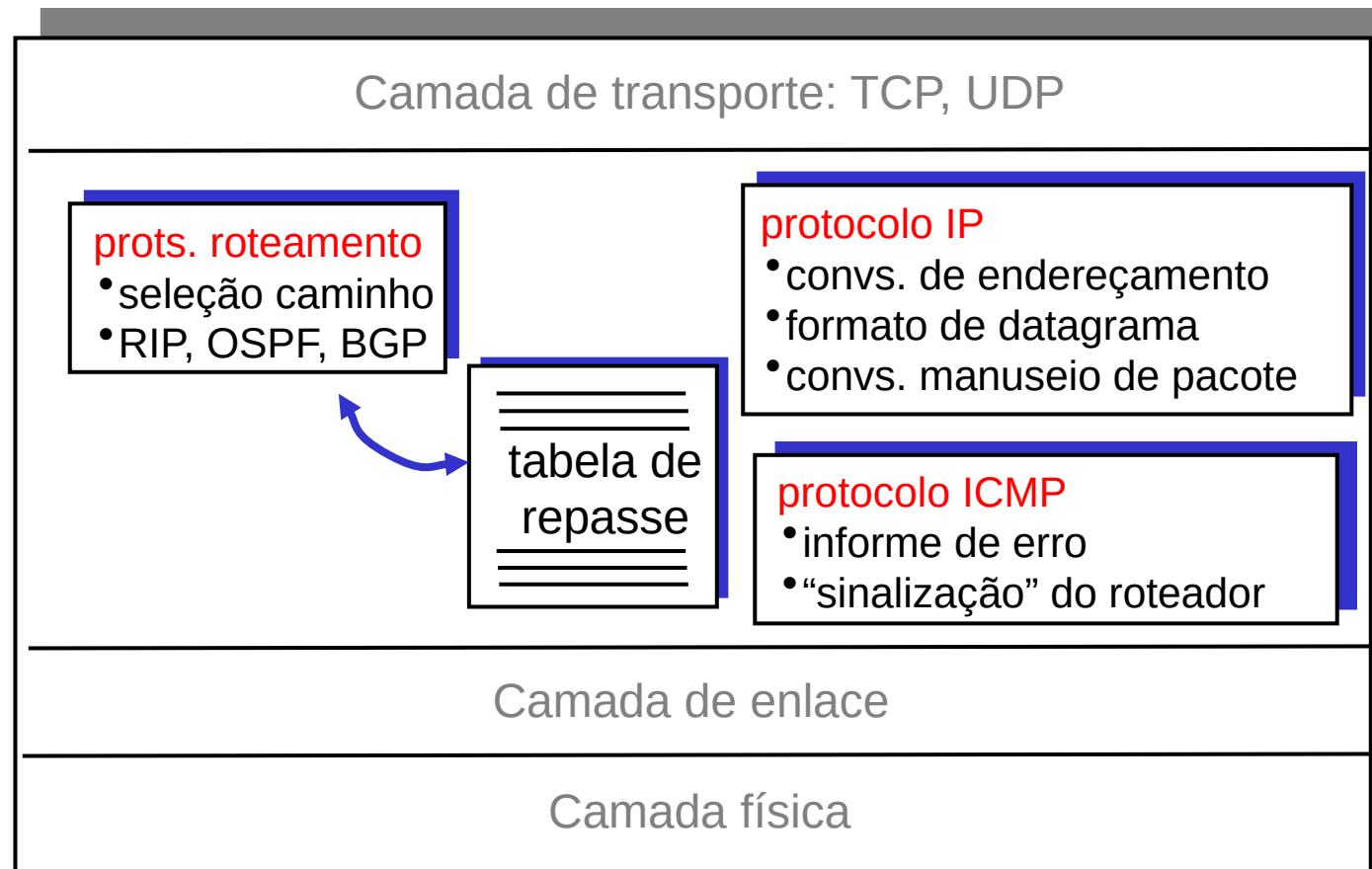
## 4.4 – Protocolo IP

- IPv4 [RFC 791] e IPv6 [RFC 2460; RFC 4291] .. 02 versões do IP !!
- “**principais componentes**” da Camada de Rede da Arq. TCP/IP:
- Internet Protocol (IP) .. conjunto de regras e procedimentos que governam a troca de datagramas IP entre dispositivos pares.
- “**componente de roteamento**”.. determina o caminho que o datagrama irá percorrer desde o remetente até o destinatário.
- “**dispositivo para comunicação de erros**” .. tem como principal elemento o Protocolo ICMP (Internet Control Message Protocol).

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4 – Protocolo IP

- “**principais componentes**” .. i) Protocolo IP (Internet Protocol); ii) RIP, OSPF, BGP (protocolos de roteamento); iii) Protocolo ICMP (protocolo para comunicação de erros de datagramas).



## 4 - Camada de Rede / 4.4 - Protocolo IP

### 4.4.1 – Formato do Datagrama IP

- “**packet**” .. também denominado de datagrama na camada de rede, desempenha um papel central na Arq. TCP/IP.
- “**Version**” (4 bits) - especifica a versão do protocolo do datagrama, assim ao analisar a versão do datagrama o roteador pode determinar como interpretar o restante do datagrama.

**IPv4 Header Format**

Octet	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Version				IHL				DSCP						ECN		Total Length															
32	Identification																Flags			Fragment Offset												
64	Time To Live								Protocol								Header Checksum															
96	Source IP Address																															
128	Destination IP Address																															
160	Options (if IHL > 5)																															

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.1 – Formato do Datagrama IP

- **“Internet Header Length”** (4 bits) – tamanho do cabeçalho em múltiplos de 32 bits, podendo variar de 20 bytes a 60 bytes.
- **“Differentiated Services Code Point”** (6 bits) – ToS (Type of Service) permite a diferenciação de datagramas IP que requerem baixo atraso, alta vazão, confiabilidade, datagrama tempo real, não tempo real.

**IPv4 Header Format**

Octet	0							1							2							3										
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Version			IHL				DSCP						ECN		Total Length																
32	Identification															Flags			Fragment Offset													
64	Time To Live							Protocol							Header Checksum																	
96	Source IP Address																															
128	Destination IP Address																															
160	Options (if IHL > 5)																															

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.1 – Formato do Datagrama IP

- **“Explicit Congestion Notification”** (2 bits) – possibilita a notificação fim-a-fim no congestionamento sem a perda de pacotes.
- **“Total Length”** (16 bits) – define o comprimento total do datagrama em bytes, ou seja, tamanho máximo teórico é de 65.535 bytes, contudo datagramas raramente são maiores do que 1.500 bytes.

**IPv4 Header Format**

Octet	0							1							2							3										
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Version			IHL				DSCP						ECN		Total Length																
32	Identification															Flags			Fragment Offset													
64	Time To Live							Protocol							Header Checksum																	
96	Source IP Address																															
128	Destination IP Address																															
160	Options (if IHL > 5)																															



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.1 – Formato do Datagrama IP

- **“Identification”** (16 bits) .. usado para identificação unívoca de um grupo de fragmentos pertencentes a um mesmo datagrama.
- **“Flags”** (3 bits) .. controle e identificação de fragmentos (bit 0 - Reservado; bit 1 – Don't Fragment ou DF; bit 2 – More Fragments ou MF).
- **“Fragment Offset”** (13 bits) .. relacionado com a fragmentação no IPv4, já o IPv6 não permite a fragmentação do datagrama IP.

**IPv4 Header Format**

Octet	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Version				IHL				DSCP						ECN		Total Length															
32	Identification																Flags			Fragment Offset												
64	Time To Live								Protocol								Header Checksum															
96	Source IP Address																															
128	Destination IP Address																															
160	Options (if IHL > 5)																															



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.1 – Formato do Datagrama IP

- **“Time To Live”** TTL (8 bits) – garante que datagramas não fiquem circulando para sempre na rede, pois é decrementado de uma unidade a cada vez que é processado por um roteador.
- **“Protocol”** (8 bits) – indica o protocolo da camada de transporte que o campo dados do datagrama IP está carregando, assim, permite a ligação entre as camadas de rede e de transporte.

**IPv4 Header Format**

Octet	0							1							2							3										
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Version			IHL				DSCP						ECN		Total Length																
32	Identification															Flags			Fragment Offset													
64	Time To Live							Protocol							Header Checksum																	
96	Source IP Address																															
128	Destination IP Address																															
160	Options (if IHL > 5)																															

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.1 – Formato do Datagrama IP

- **“Header Checksum”** (16 bits) – permite que o roteador detecte erros de bits em um datagrama ao recalcular este campo.
- .. soma verificação é obtida tratando cada 2 bytes do cabeçalho como número para então efetuar uma soma em complemento 1.
- .. caso o valor calculado não seja igual ao valor do campo de soma verificação do datagrama IP, o datagrama é descartado.

**IPv4 Header Format**

Octet	0							1							2							3										
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Version			IHL				DSCP						ECN		Total Length																
32	Identification															Flags			Fragment Offset													
64	Time To Live							Protocol							Header Checksum																	
96	Source IP Address																															
128	Destination IP Address																															
160	Options (if IHL > 5)																															

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.1 – Formato do Datagrama IP

- **“Source/Destination Address”** (32 bits) – endereços fonte e destino que representam as interfaces de rede do remetente e do destinatário.
- **“Options”** (0 a 40 bytes) – permite que o cabeçalho do datagrama IP seja ampliado, mas raramente é utilizado por questões de sobrecarga.
- Obs.: No IPv6 o campo de “opções” foi descartado.

**IPv4 Header Format**

Octet	0							1							2							3										
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Version			IHL				DSCP						ECN		Total Length																
32	Identification															Flags			Fragment Offset													
64	Time To Live							Protocol							Header Checksum																	
96	Source IP Address																															
128	Destination IP Address																															
160	Options (if IHL > 5)																															

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.1 – Formato do Datagrama IP

- “**dados**” (0 .. 65.515 com o cabeçalho de 20 bytes) .. contém o segmento da camada de transporte (TCP; UDP; SCTP; RTP .. etc.)
- ... note que o datagrama IP tem 20 bytes de cabeçalho se o campo de “opções” for 0 bytes sendo o restante o campo de “payload”.
- ... se a carga for um segmento TCP, tem-se 40 bytes de cabeçalho (IP + TCP) mais a mensagem da camada de aplicação.

**IPv4 Header Format**

Octet	0							1							2							3										
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Version			IHL				DSCP					ECN		Total Length																	
32	Identification															Flags			Fragment Offset													
64	Time To Live							Protocol							Header Checksum																	
96	Source IP Address																															
128	Destination IP Address																															
160	Options (if IHL > 5)																															

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.1 – Formato do Datagrama IP

- **“restrição quanto ao tamanho”** - decorre dos diferentes protocolos nas diferentes camadas ao longo de um caminho na rede.
- Quadro Ethernet (802.3) na camada de enlace aceita quadros de 64 bytes (mínimo) até 1518 bytes (máximo).
- Quadro Ethernet (802.1Q) na camada de enlace aceita quadros de 64 bytes (mínimo) até 1522 bytes (máximo).
- Quadro Ethernet (802.3) na camada física aceita quadros de 72 bytes (mínimo) até 1526 bytes (máximo) >> diferença de 8 bytes.
- Quadro Ethernet (802.1Q) na camada física aceita quadros de 72 bytes (mínimo) até 1530 bytes (máximo) >> diferença de 8 bytes.

## ... 4.4.1 – Formato do Datagrama IP

- Maximum Transmission Unit (MTU) .. quantidade máxima de dados que um quadro da camada de enlace pode carregar.
- “**problema**” .. ao longo de uma rota entre remetente e destinatário podemos encontrar diferentes enlaces, cada um com diferentes MTUs.
- “**problema**” .. MTU limita o comprimento máximo do datagrama IP, em rotas com diferentes enlaces, o que gera problemas no roteamento.
- “**solução**” . fragmentar o datagrama IP em 02 ou mais fragmentos IP e enviá-los através do enlace respeitando-se a MTU do enlace.
- ... datagramas precisam ser reconstruídos antes de serem entregues à camada de transporte no destino, isto porque a camada de transporte espera receber segmentos completos da camada de rede.

## ... 4.4.1 – Formato do Datagrama IP

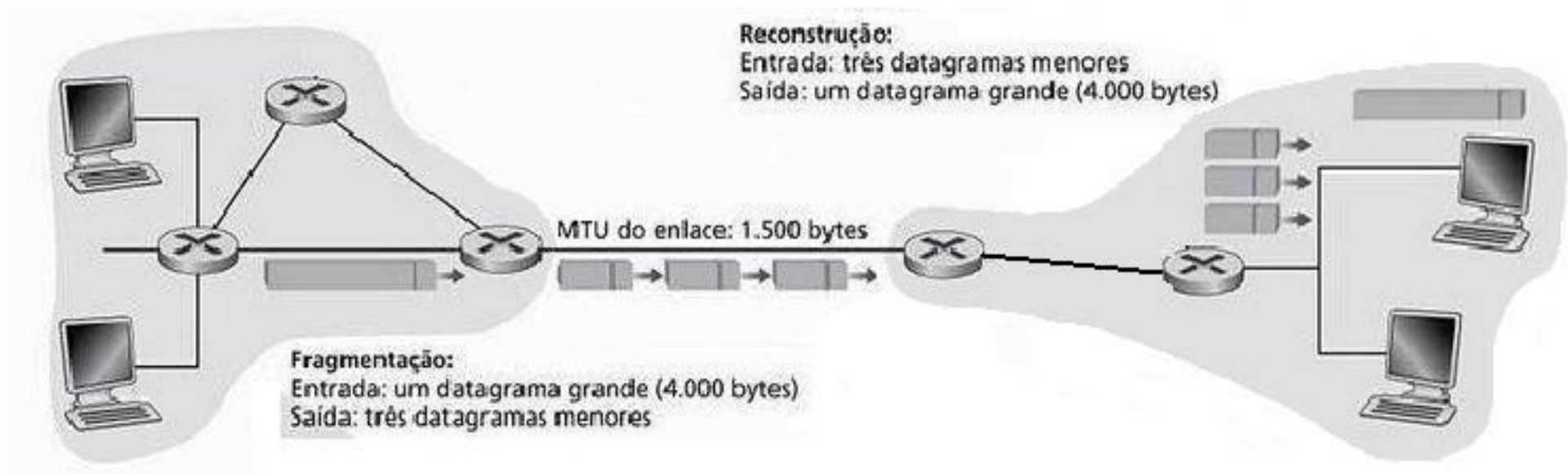
- “**identificador, flags, deslocamento de fragmentação**” - estão relacionados com a fragmentação do datagrama IPv4, já o IPv6 não permite a fragmentação do datagrama IP.
- ... datagrama recebe um único nro. de identificação quando é montado no remetente juntamente com os endereços fonte e destino.
- ... para cada novo datagrama com mesmo endereço fonte e destino, incrementa-se o “**fragment offset**” pelo remetente.
- ... destinatário ao receber datagramas de um mesmo remetente, pode examinar os nros. de identificação para determinar quais são fragmentos intermediários e qual é o último fragmento.
- ... datagrama fragmentado contém o “flag” ajustado para “1” para todos os fragmentos exceto para o último fragmento do datagrama original.



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.1 – Formato do Datagrama IP

- e.g., considere um datagrama de 4000 bytes (cabeçalho de 20 bytes + “payload” de 3980 bytes) chegando a um roteador o qual deve repassá-lo a um enlace com MTU de 1500 bytes.
- .. datagrama original cuja identificação é “777” precisa ser dividido em 03 fragmentos cujo tamanho deve ser múltiplo de 8 bytes.





## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.1 – Formato do Datagrama IP

- e.g., considere um datagrama de 4000 bytes (cabeçalho de 20 bytes + “payload” de 3980 bytes) chegando a um roteador o qual deve repassá-lo a um enlace com MTU de 1500 bytes.
- .. datagrama original cuja identificação é “777” precisa ser dividido em 03 fragmentos cujo tamanho deve ser múltiplo de 8 bytes.

**IPv4 Header Format**

Octet	0							1							2							3										
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Version			IHL				DSCP						ECN		Total Length																
32	Identification															Flags			Fragment Offset													
64	Time To Live							Protocol							Header Checksum																	
96	Source IP Address																															
128	Destination IP Address																															
160	Options (if IHL > 5)																															

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.1 – Formato do Datagrama IP

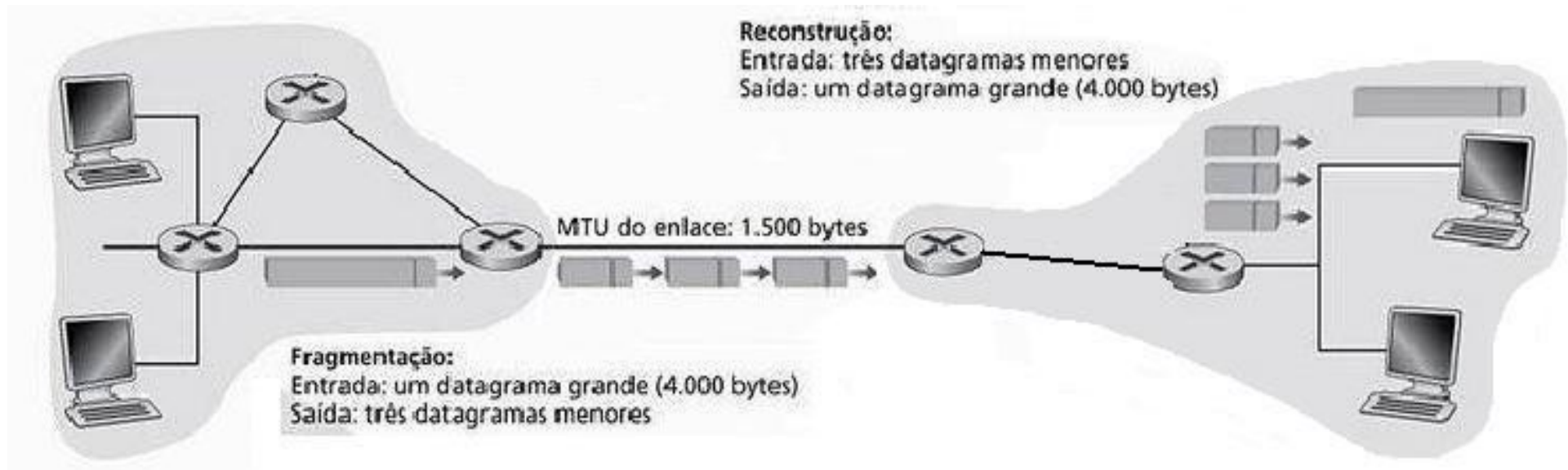
- .. no destino, a carga útil do datagrama é passada para a camada de transporte somente após a reconstrução do datagrama original.
- .. caso um dos datagramas não chegue ao destino, o datagrama será descartado e não será entregue à camada de transporte.
- **“fragmentação”** - papel importante na união das diversas tecnologias distintas da camada de enlace.

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

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.1 – Formato do Datagrama IP

- “**fragmentação**” .. exige que roteadores e “hosts” contemplem a fragmentação do datagrama bem como o reagrupamento.
- “**fragmentação**” .. pode ser usada para criar DoS fatais, uma vez que o atacante envia uma série de fragmentos estranhos e inesperados.
- .. sistemas operacionais de rede NÃO preparados para lidar com fragmentos podem simplesmente parar de operar.



## 4 - Camada de Rede / 4.4 - Protocolo IP

### 4.4.2 – Endereçamento IPv4

- Datagrama IP exige que cada interface tenha o seu próprio endereço IP, assim, 01 endereço IP está associado a 01 interface de rede.
- ... um roteador ligado a 02 ou mais enlaces tem como tarefa receber datagramas em um enlace e repassá-lo a algum outro enlace, assim, o roteador está ligado a dois ou mais enlaces.
- **“interface de rede”** .. fronteira entre o roteador e qualquer um dos enlaces que estão associados ao roteador.

## 4 - Camada de Rede / 4.4 - Protocolo IP

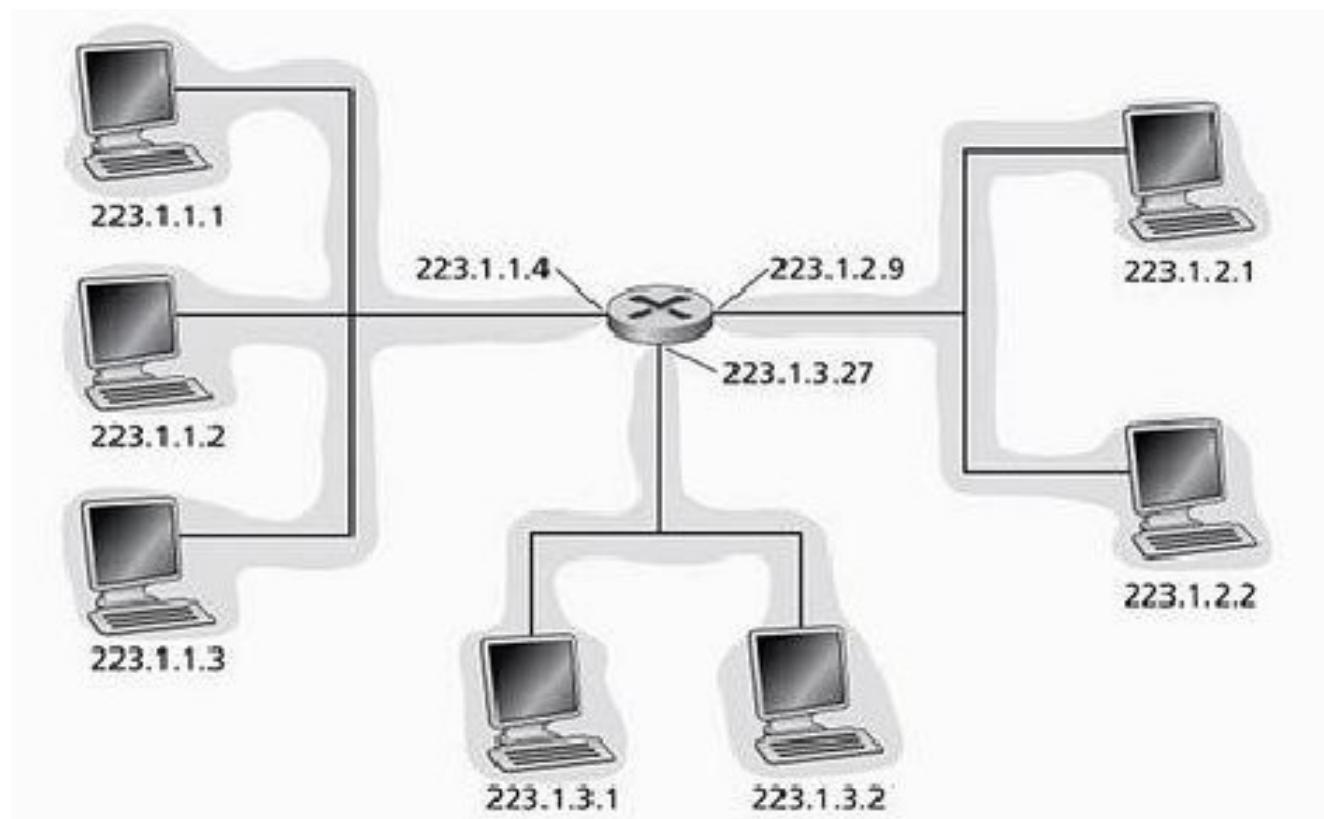
### ... 4.4.2 – Endereçamento IPv4

- **“interface de rede”** .. fronteira entre o roteador e qualquer um dos enlaces que estão associados ao roteador.
- .. a exigência de um endereço IP por interface advém do fato de que “hosts” e roteadores podem enviar e receber datagramas IP.
- .. cada endereço IP tem 32 bits (4 bytes), portanto há um total de  $2^{32}$  endereços IP possíveis de serem utilizados.
- **“notação decimal”** .. utilizada para representar os endereços, ou seja, cada byte é convertido para o decimal correspondente e na sequência são separados por pontos, p.ex., “193.32.216.9”

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

- ... cada interface em cada “host” ou roteador tem um endereço IP público, exceto as interfaces que estão atrás de NATs – nestes casos, utilizam normalmente endereços privados.



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

- ... cada interface em cada “host” ou roteador tem um endereço IP público, exceto as interfaces que estão atrás de NATs – nestes casos, utilizam normalmente endereços privados.
- A (Private Addresses) - 10.0.0.0 até 10.255.255.255
- B (Private Addresses) - 172.16.0.0 até 172.31.255.255
- C (Private Addresses) - 192.168.0.0 até 192.168.255.255

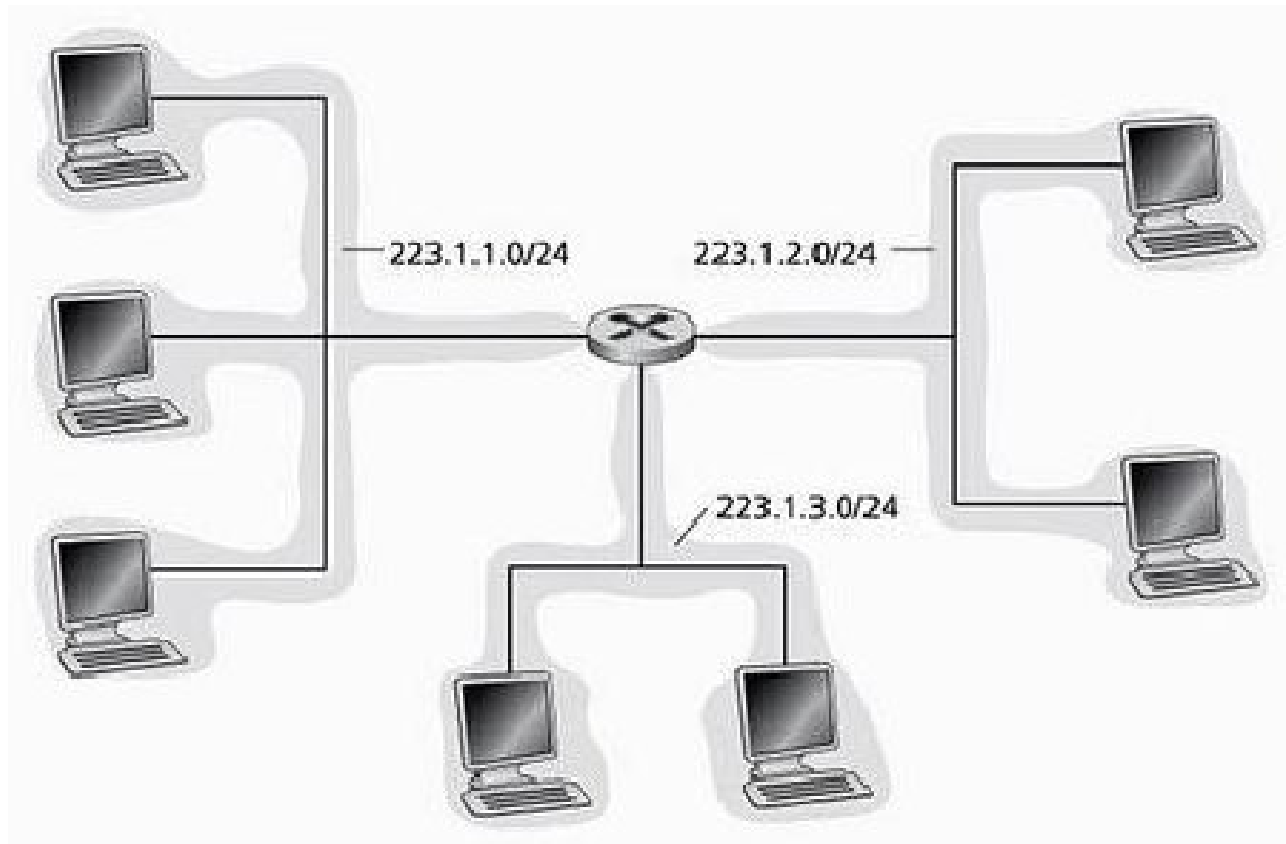
**Historical classful network architecture**

<b>Class</b>	<b>Leading bits</b>	<b>Size of <i>network number</i> bit field</b>	<b>Size of <i>rest</i> bit field</b>	<b>Number of networks</b>	<b>Addresses per network</b>	<b>Start address</b>	<b>End address</b>
<b>A</b>	0	8	24	128 ( $2^7$ )	16,777,216 ( $2^{24}$ )	0.0.0.0	127.255.255.255
<b>B</b>	10	16	16	16,384 ( $2^{14}$ )	65,536 ( $2^{16}$ )	128.0.0.0	191.255.255.255
<b>C</b>	110	24	8	2,097,152 ( $2^{21}$ )	256 ( $2^8$ )	192.0.0.0	223.255.255.255

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

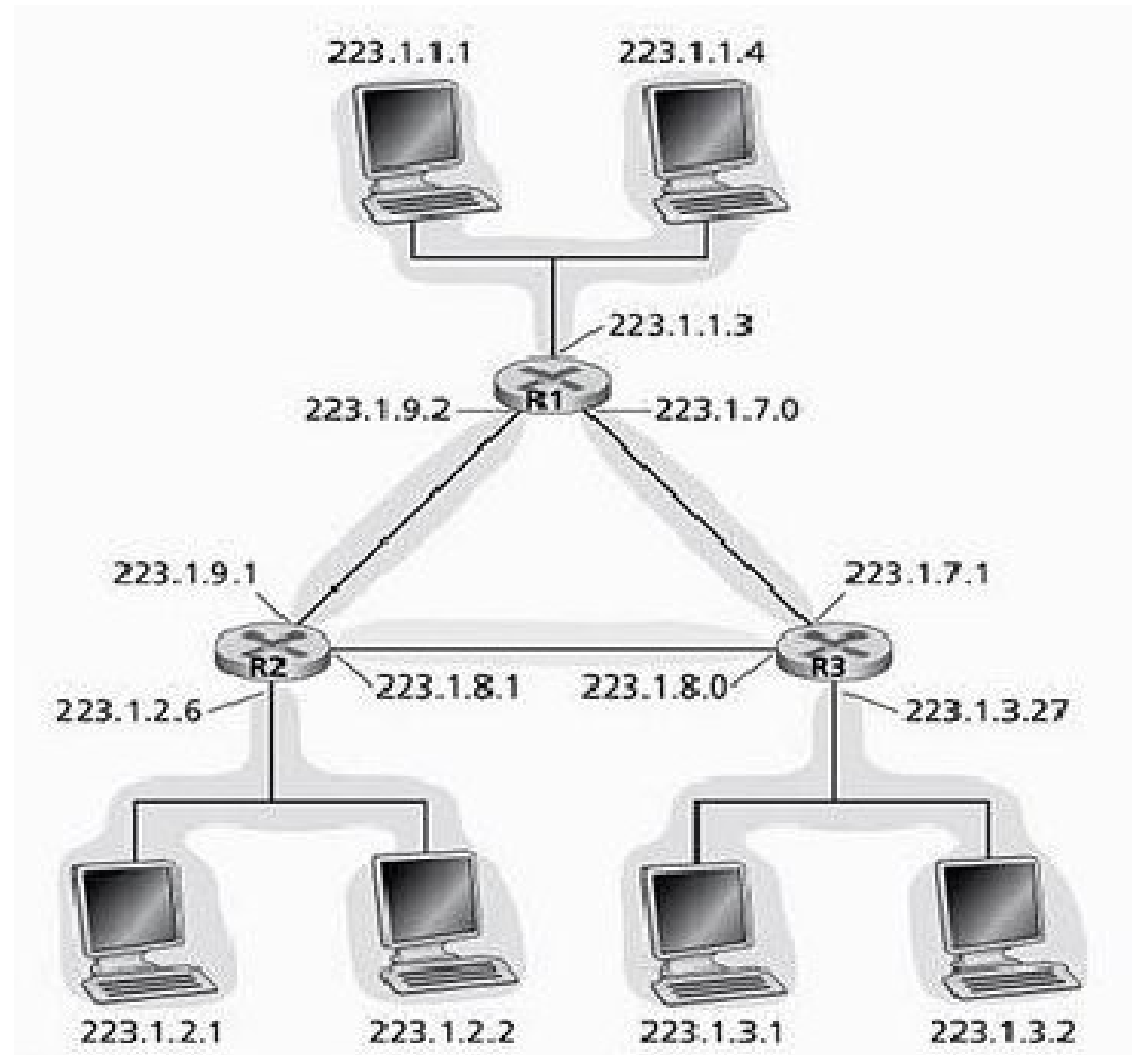
- “**sub-rede**” .. quando um dispositivo se conecta à mesma parte da sub-rede do endereço IP, o mesmo pode alcançar um outro dispositivo na mesma sub-rede sem a necessidade de roteador.





## 4 - Camada de Rede / 4.4 - Protocolo IP ... 4.4.2 – Endereçamento IPv4

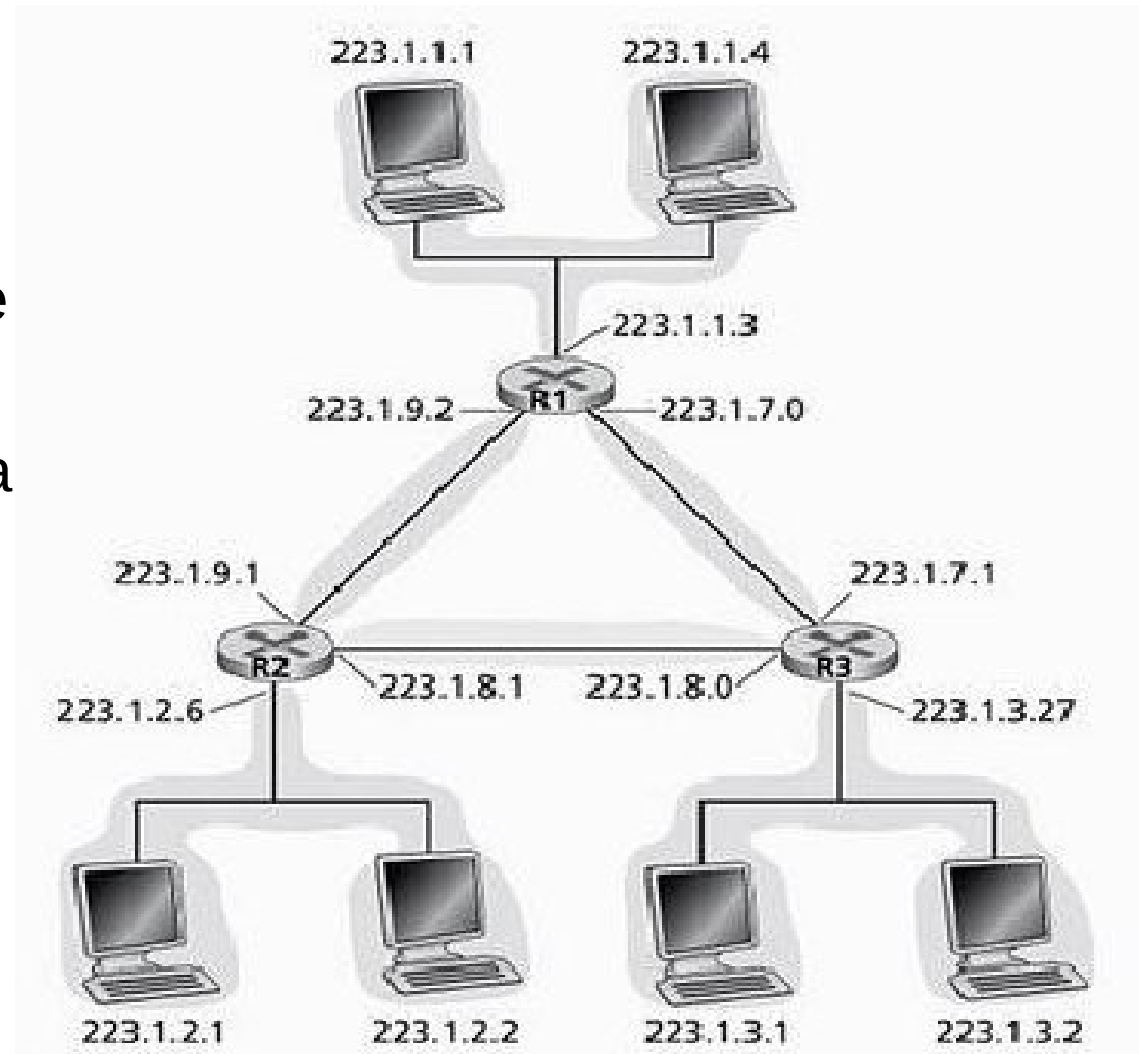
- Que sub-redes IP estão presentes na figura ao lado ?
- “**resposta**” .. 06 sub-redes !!
- 03 sub-redes .. 223.1.1.0/24; 223.1.2.0/24 e 223.1.3.0/24;
- 03 sub-redes adicionais .. 223.1.7.0/24; 223.1.8.0/24 e 223.1.9.0/24.



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

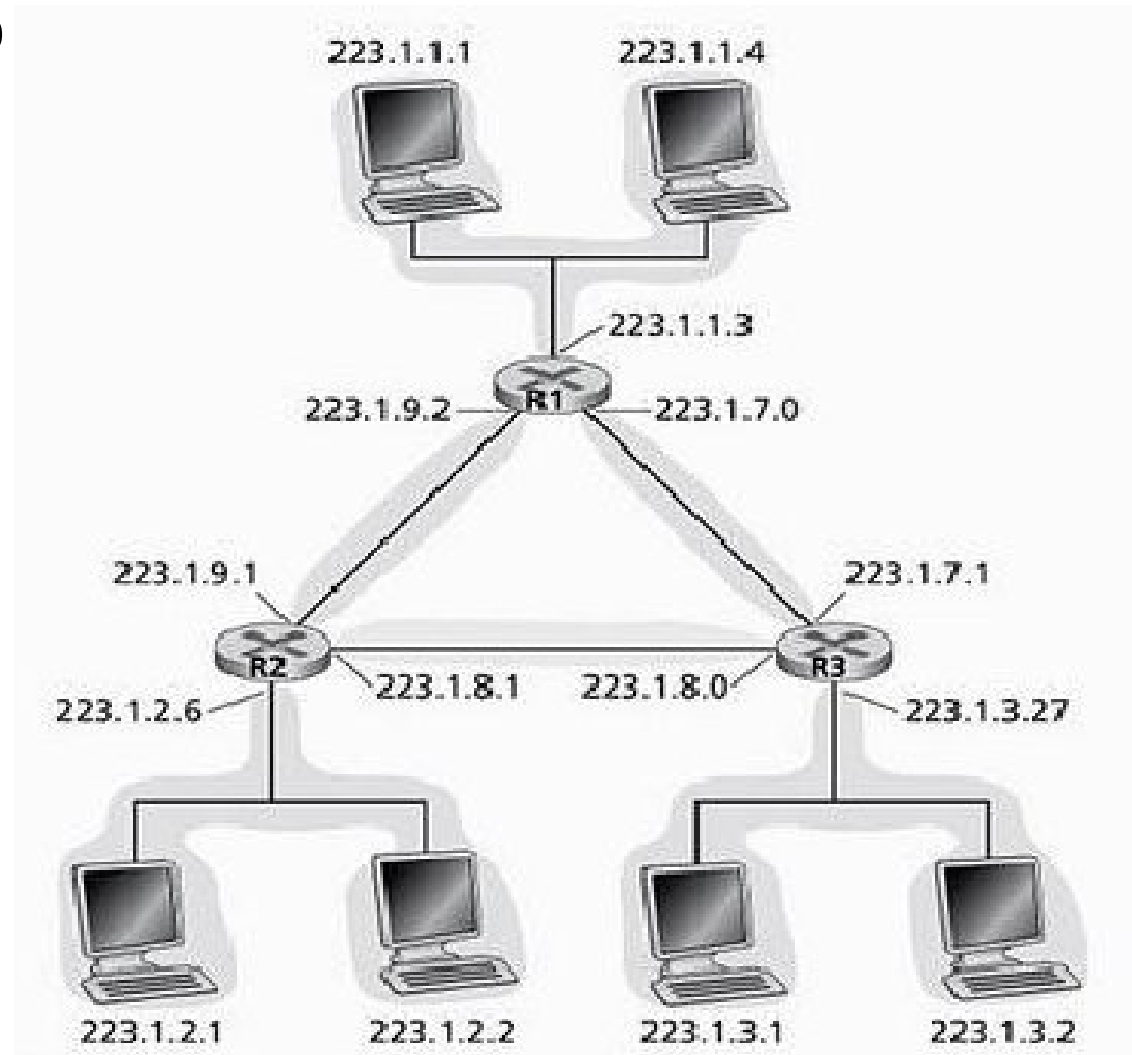
- **“definição da sub-rede”** ..  
destaque cada interface de “host” ou roteador e, na sequência, identifique a partir do seu endereço IP quais são os bits que se repetem.
- .. endereço mais significativo para o menos significativo .. mais sig. (+sig.) >> menos sig (-sig.)
- ... este conjunto de bits define o nro. da sub-rede bem como a máscara da sub-rede para todo endereço IP pertencente a ela.



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

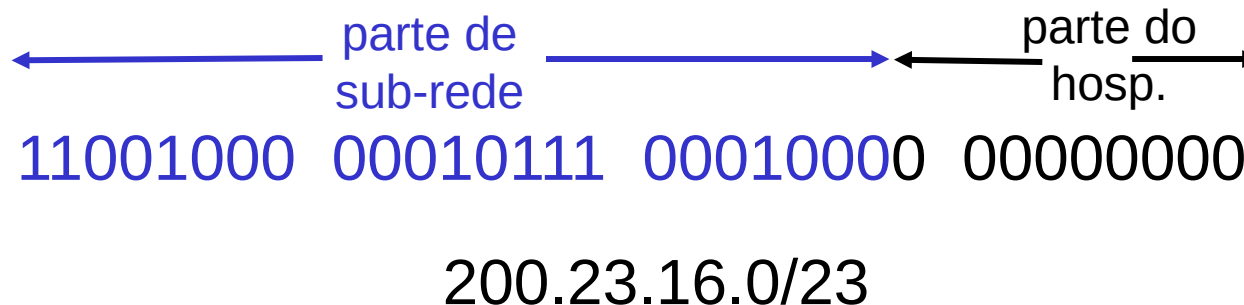
- ... este conjunto de bits define o nro. da sub-rede bem como a máscara da sub-rede para todo endereço IP pertencente a ela.
- sub-rede “223.1.1.0” e máscara “/24” ou sub-rede “223.1.1.0” e máscara “255.255.255.0”
- sub-rede “223.1.2.0” e máscara “/24” ou sub-rede “223.1.2.0” e máscara “255.255.255.0”
- .. mesmo procedimento para as demais sub-redes da figura !!



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

- **CIDR (Classless InterDomain Routing)** .. roteamento interdomínio sem classes generaliza a noção de endereçamento de sub-rede.
- parte da sub-rede do endereço de tamanho arbitrário - a.b.c.d/x, onde “x” é nro. de bits na parte correspondente a subrede do endereço IP.
- “x” bits mais significativos de um endereço constituem a parcela da rede do endereço IP e são denominados “prefixo de rede”.
- .. uma organização normalmente recebe um bloco de endereços contíguos, isto é, uma faixa de endereços com um prefixo comum.



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

- **“Dynamic Host Configuration Protocol” (DHCP)** .. permitir que o “host” obtenha dinamicamente endereço IP do servidor de rede.
- .. pode renovar seu prazo no endereço utilizado, bem como aceitar usuários móveis que queiram se juntar à rede (mais adiante).
- .. permite reutilização de endereços, embora só mantenha o endereço enquanto “host” está conectado e “ligado”.
- Protocolo DHCP .. automatiza aspectos relativos a conexão de “hosts” na rede » algo como o protocolo “plug-and-play”.
- .. utilizado largamente nas redes residenciais de acesso a Internet e em LANs “wireless” onde “hosts” entram e saem da rede.

## 4 - Camada de Rede / 4.4 - Protocolo IP

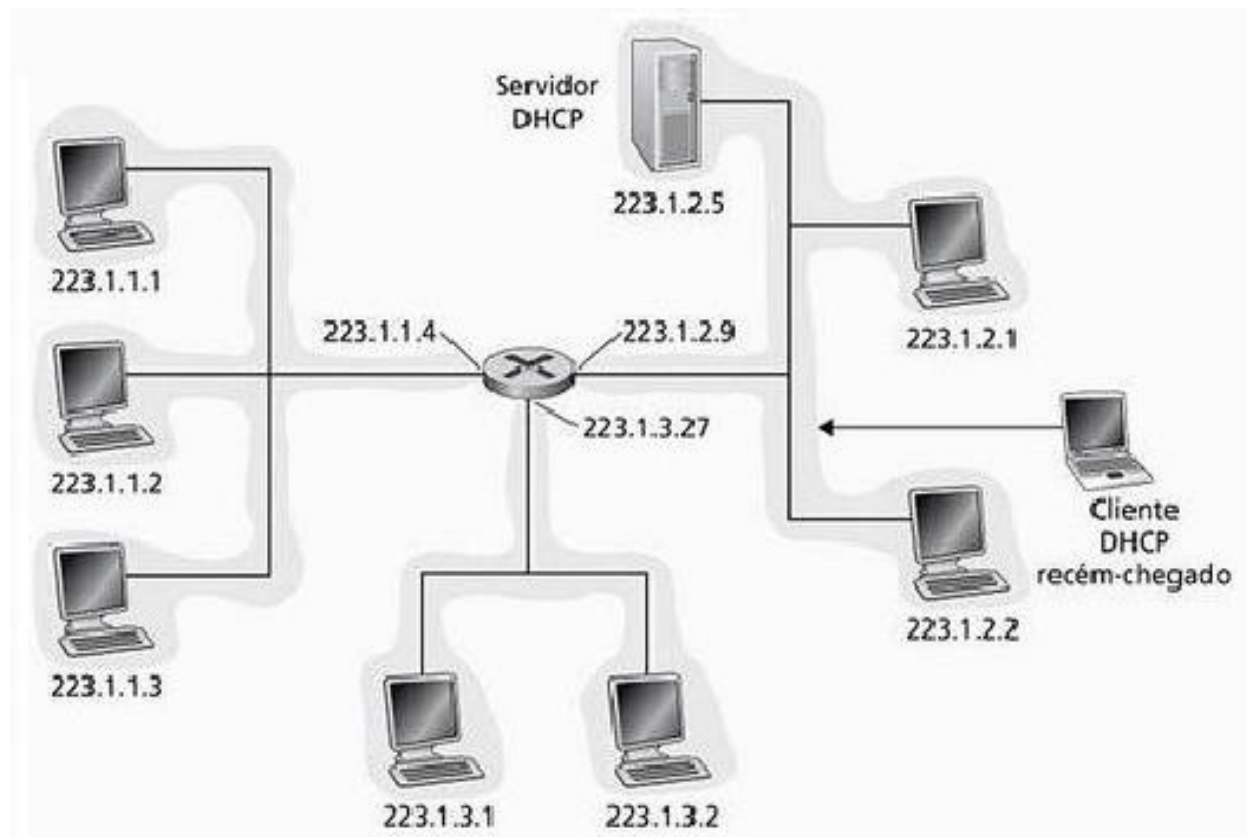
### ... 4.4.2 – Endereçamento IPv4

- Protocolo DHCP .. automatiza aspectos relativos a conexão de “hosts” na rede » algo como o protocolo “plug-and-play”.
- ... ..
- ... extremamente útil onde há muitos usuários em trânsito e os endereços são utilizados apenas por um tempo limitado.
- ... medida que “hosts” entram e saem da rede, servidor precisa atualizar a lista de endereços disponíveis.
- ... toda vez que um “host” se conecta à rede, designa-se um endereço arbitrário dentre os endereços disponíveis no reservatório.
- ... toda vez que um “host” sai da rede, o endereço é devolvido ao reservatório para ser reutilizado por um outro “host” ou pelo mesmo.

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

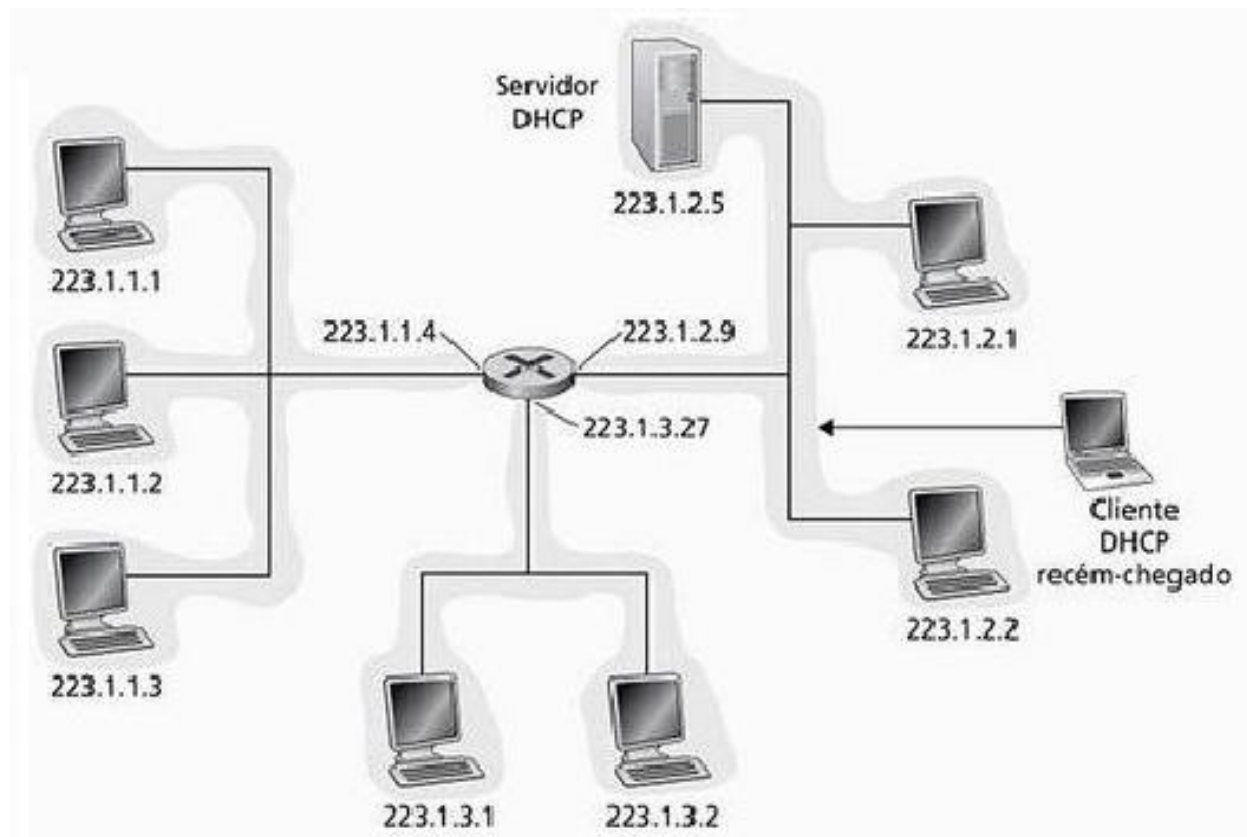
- Visão geral do DHCP
- .. “host” “broadcasts” msg. “DHCP DISCOVER” [opcional].
- .. servidor DHCP responde com msg “DHCP OFFER” [opcional].



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

- Visão geral do DHCP
- .. “host” requer endereço IP com a msg “DHCP REQUEST”.
- .. servidor DHCP envia endereço através da msg “DHCP ACK”.

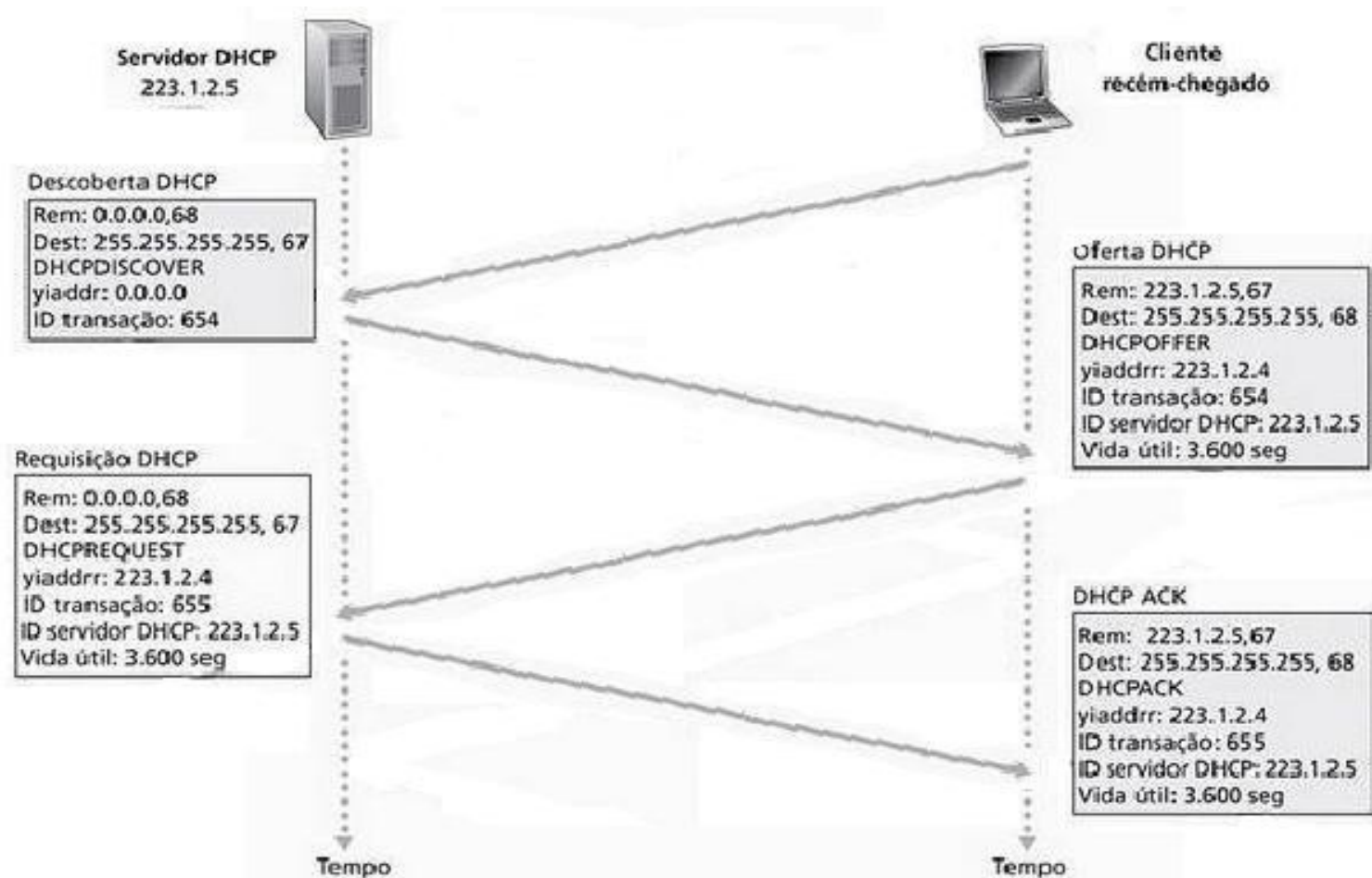




## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

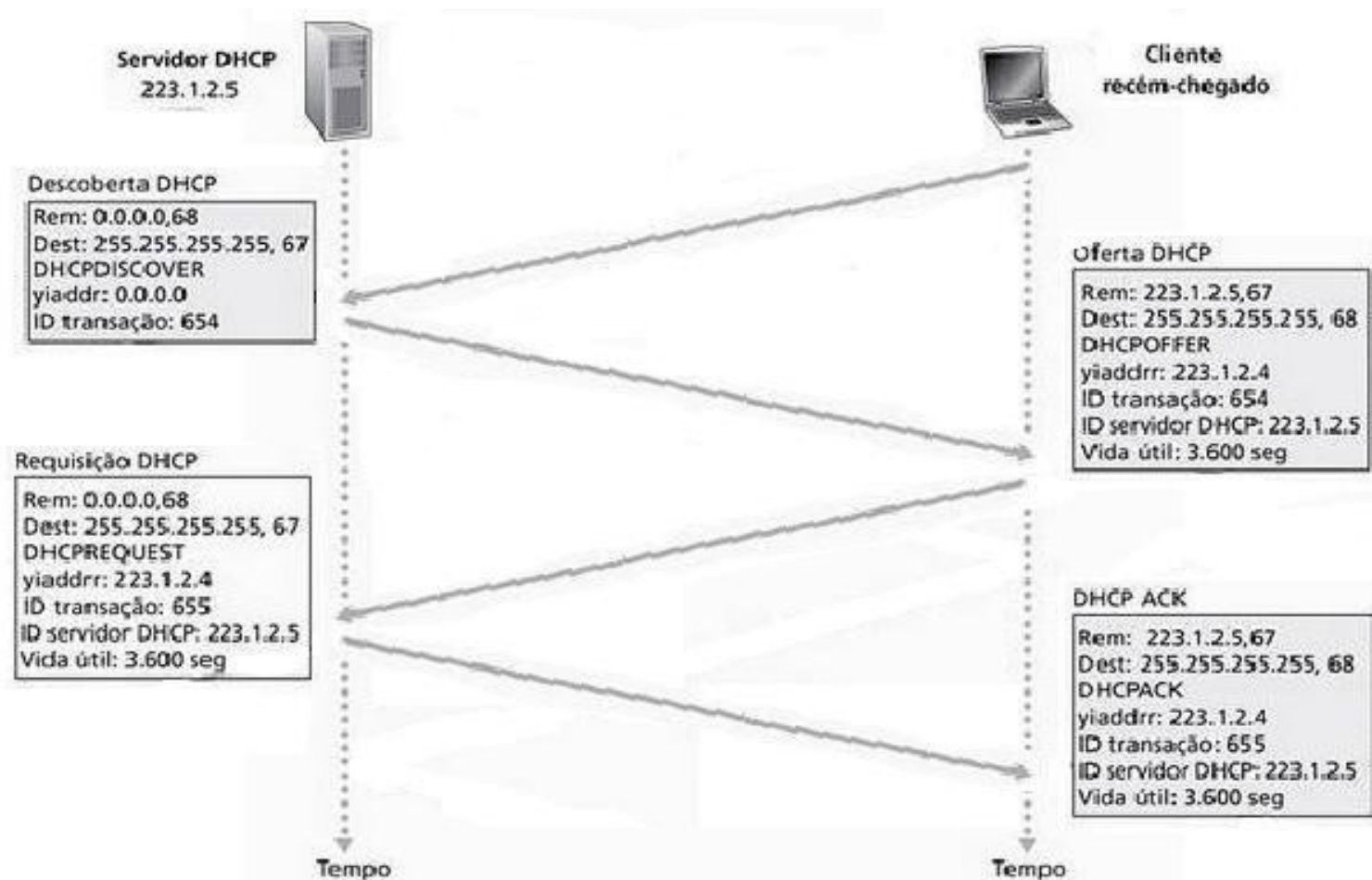
- “**cliente**” .. “broadcast” da mensagem de “Descoberta DHCP” com IP destino “255.255.255.255” e “port” “67”.



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

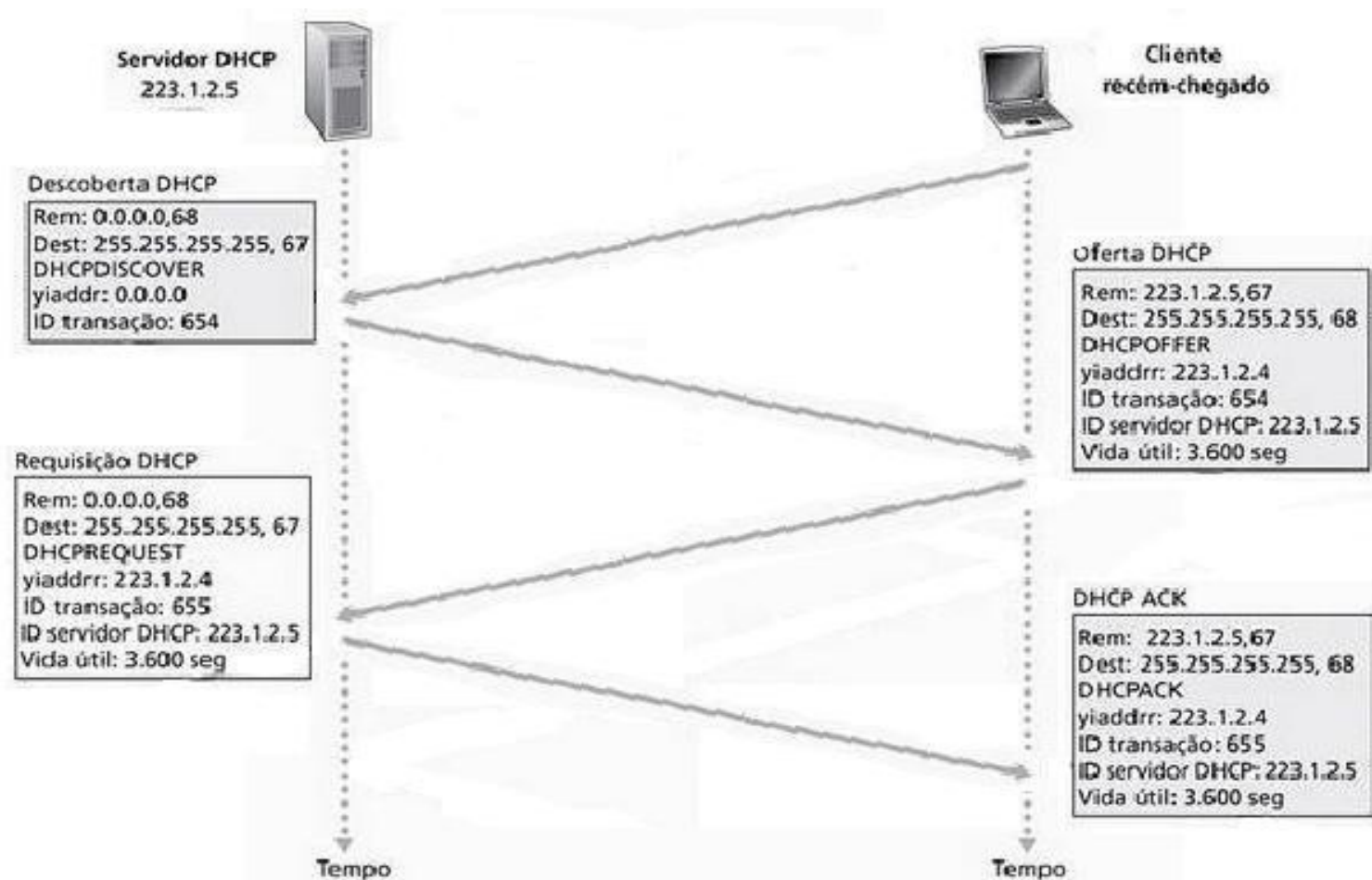
- “**oferta de servidor**” .. servidor DHCP ao receber msg. de descoberta, responde ao cliente com uma mensagem “Oferta DHCP”.



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

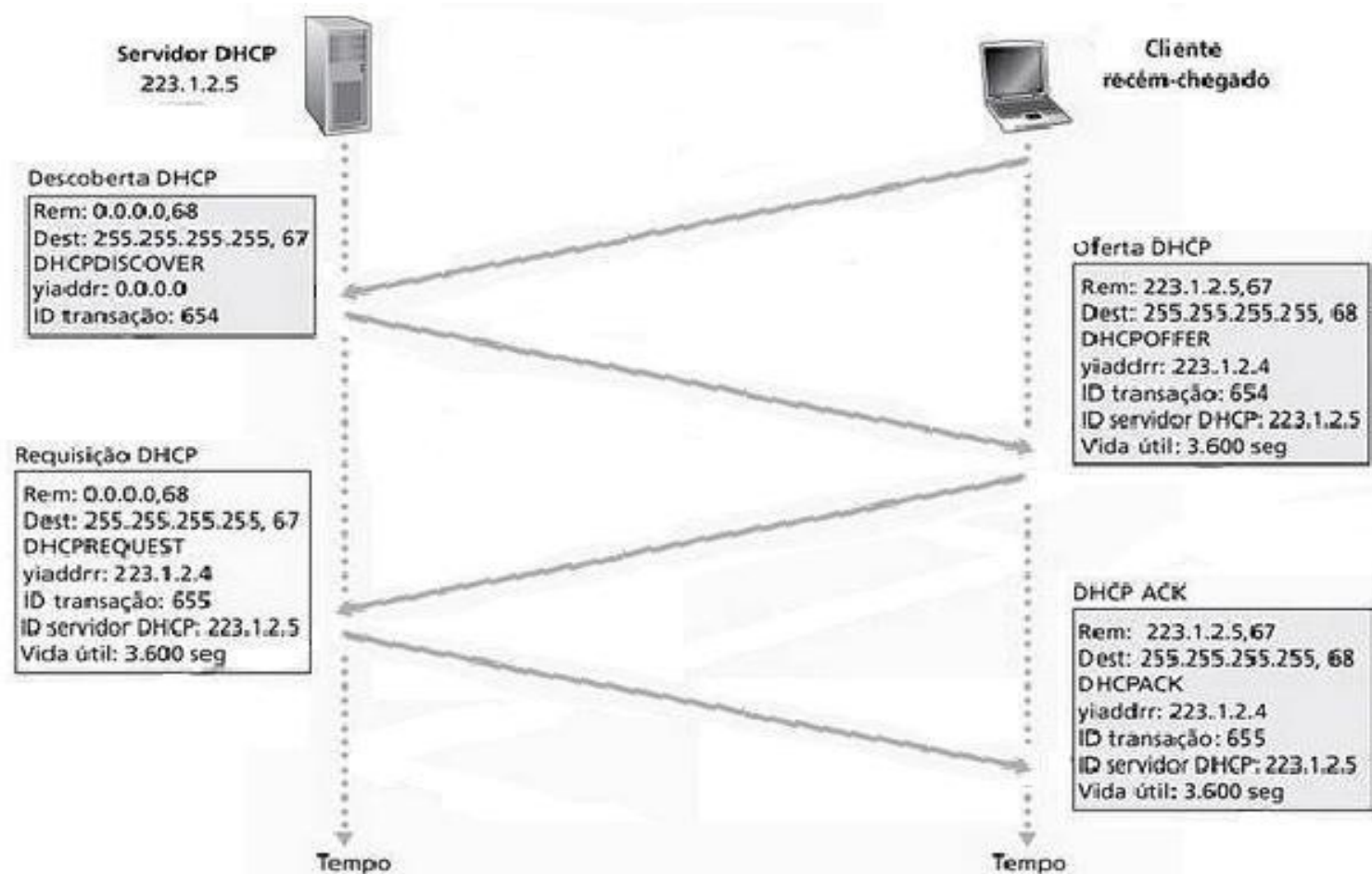
- “**solicitação**” .. cliente escolhe dentre 01 ou mais ofertas de servidores com uma mensagem “Solicitação DHCP”.



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

- **“DHCP Ack”** .. servidor responde a msg. de “Requisição DHCP” com a mensagem “DHCP Ack” ... confirmando os parâmetros.



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

- DHCP pode retornar mais do que o IP alocado na sub-rede:
- p.ex., endereço do roteador do 1º salto para o cliente.
- p.ex., nome e endereço IP do servidor DNS.
- p.ex., máscara de rede, ou seja, parte do endereço que representa a rede *versus* parte do endereço que representa o “host”.

## 4 - Camada de Rede / 4.4 - Protocolo IP

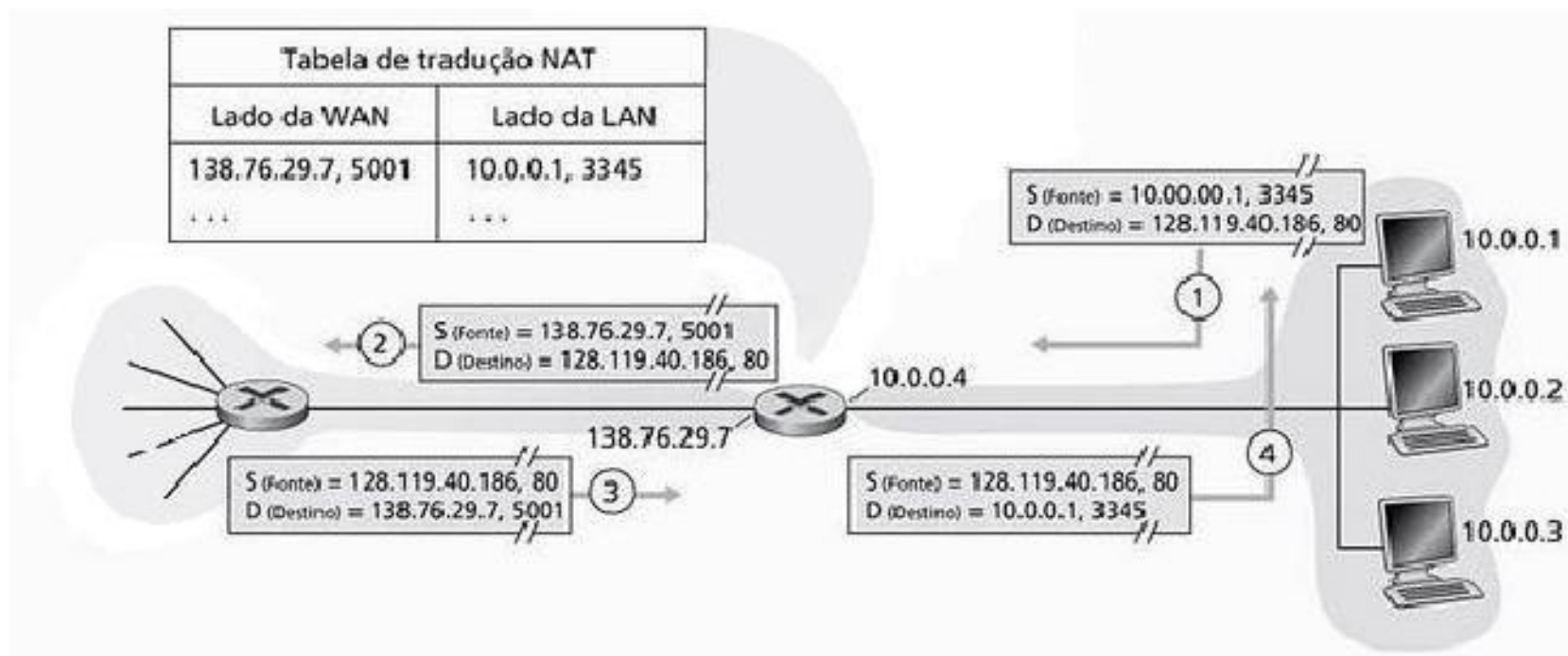
### ... 4.4.2 – Endereçamento IPv4

- **NAT (Network Address Translation)** .. promove a mudança de endereço IP no cabeçalho de uma datagrama enquanto o mesmo passa por um “roteador” ou por um “firewall”.
- .. “host” na rede interna passa a ter acesso à Rede Internet (Pública).
- RFC 2663; RFC 3022 – abordagem mais simples de alocação de endereços tem como princípio a utilização de porções reservadas do espaço de endereçamento IP (RFC 1918).
- A (Private Addresses) - 10.0.0.0 to 10.255.255.255
- B (Private Addresses) - 172.16.0.0 to 172.31.255.255
- C (Private Addresses) - 192.168.0.0 to 192.168.255.255

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

- RFC 2663; RFC 3022 .. abordagem mais simples de alocação de endereços tem como princípio a utilização de porções reservadas do espaço de endereçamento IP (RFC 1918).
- “**endereços privados**” .. endereços que somente tem significado para equipamentos pertencentes à rede em questão.

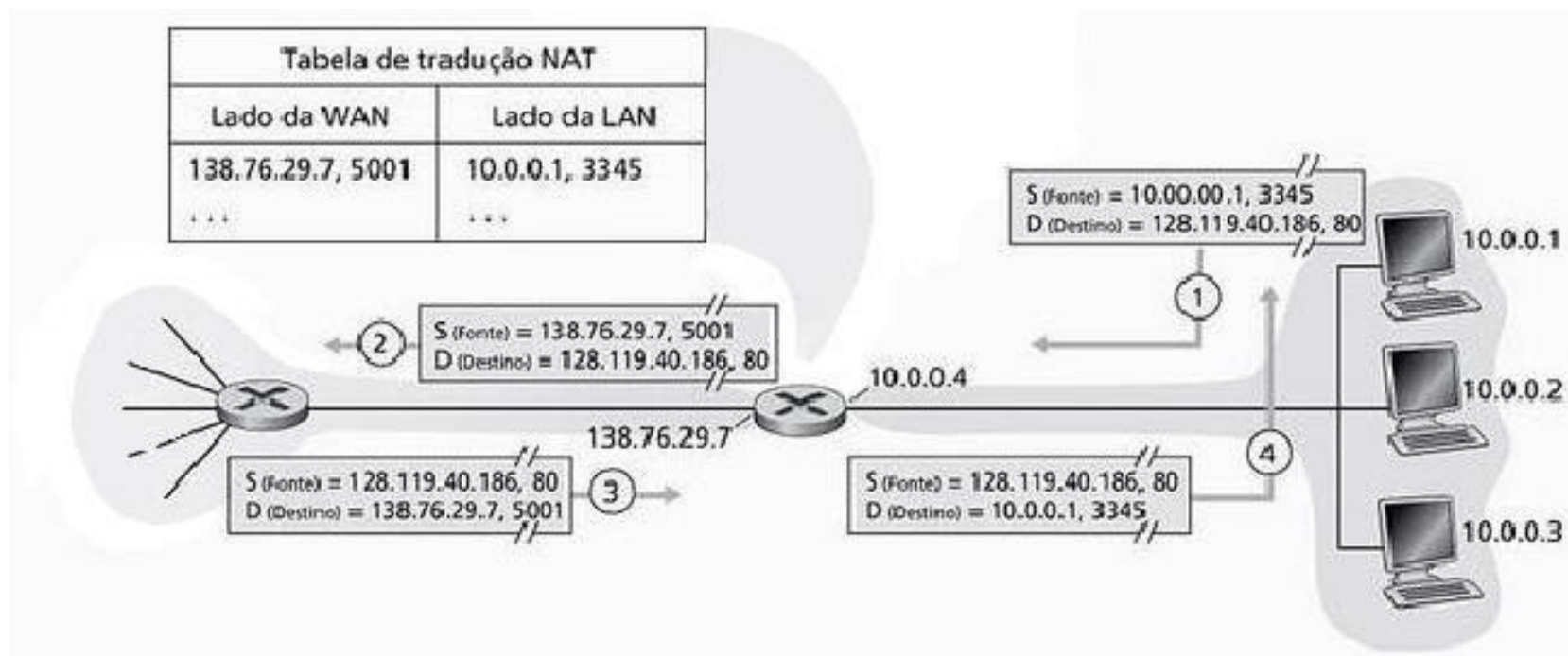




## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

- .. se todos os datagramas que chegam ao roteador NAT provenientes da WAN tiverem o mesmo endereço IP de destino, como o roteador NAT sabe para qual “host” repassar ?
- .. quando datagrama chega ao roteador NAT, o mesmo indexa a tabela de tradução usando “IP Destino” e “Nro. de Porta de Destino” para obter o IP “10.0.0.1” e nro. de porta “3345”.

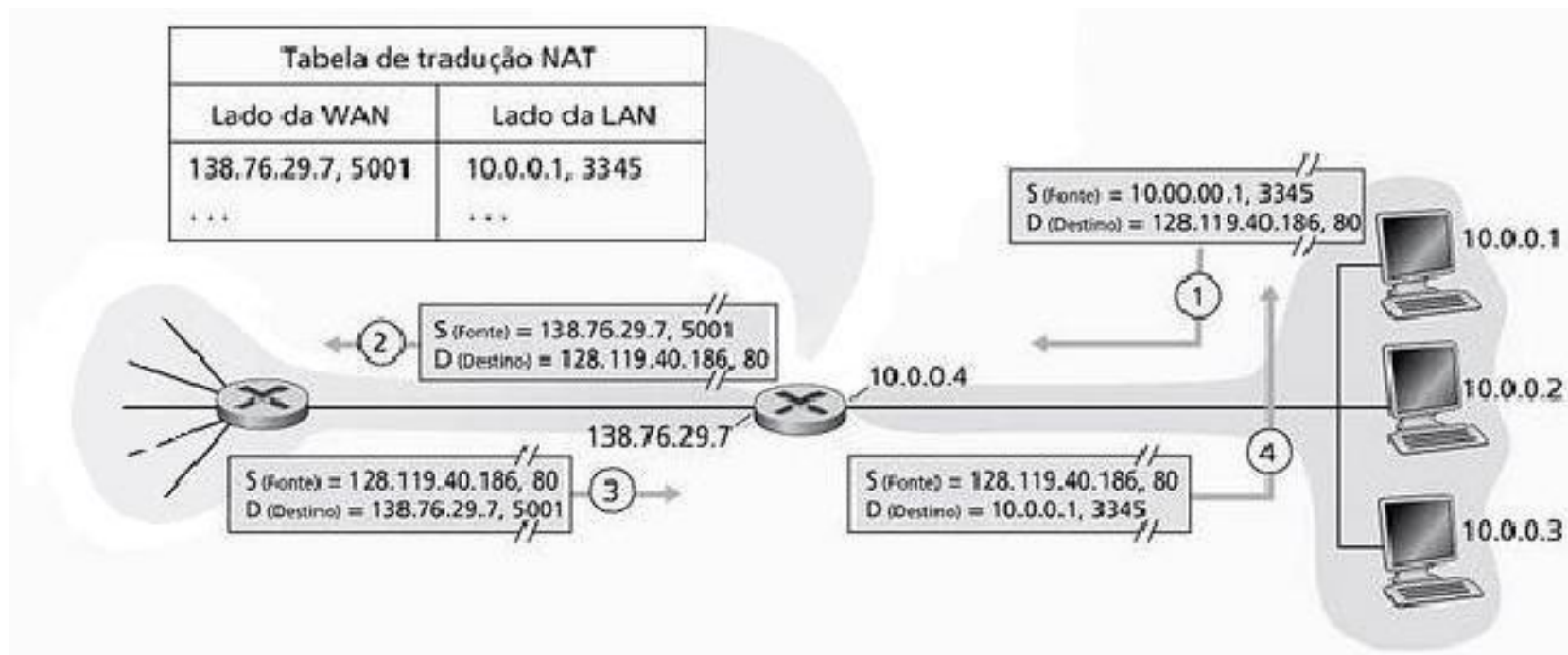




## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

- .. roteador reescreve endereço de destino e o nro. de porta de destino do datagrama e o repassa para a rede interna.
- .. comprimento do campo nro. de portas é de 16 bits » pode suportar até 65535 conexões simultâneas com 01 único endereço IP.



## 4 - Camada de Rede / 4.4 - Protocolo IP

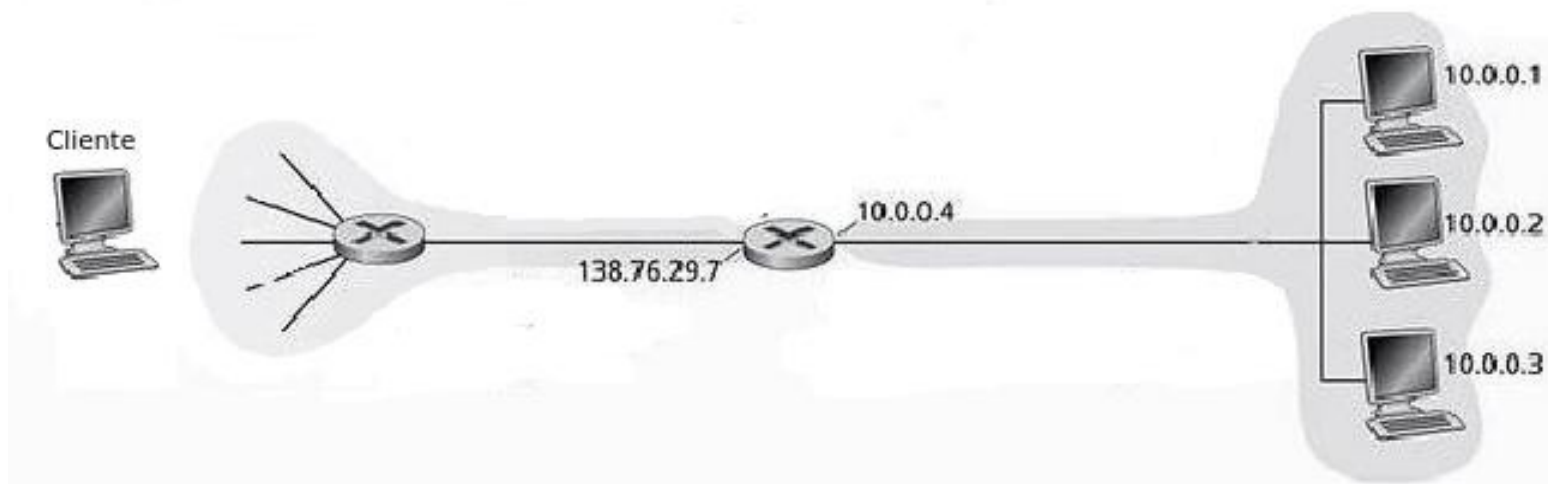
### ... 4.4.2 – Endereçamento IPv4

- Restrições do IETF ao NAT:
- .. argumenta-se que a finalidade dos nros. de portas é endereçar processos e não “hosts”, no entanto, o servidor NAT utiliza nro. de porta.
- .. roteadores devem processar pacotes somente até a 3ª camada e, portanto, não cabe ao mesmo o processamento para o nro. de porta.
- “**conclusão**” .. protocolo viola o argumento de comunicação “host-host” ou “fim-a-fim”, uma vez que há a interferência de nós que modificam endereços IP e nro. de portas após a indexação da tabela de tradução.
- Obs.: IPv6 já resolve este problema de escassez de endereços, no entanto, NAT é utilizado como uma solução temporária.

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

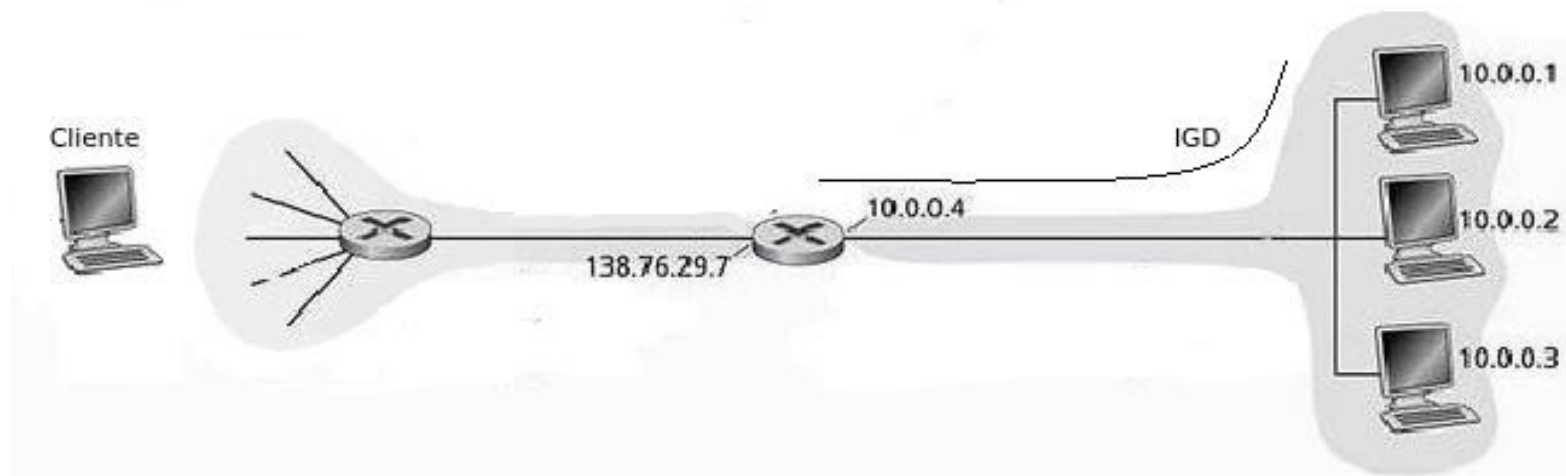
- e.g., considere “A” e “B” elementos de uma Aplicação P2P, onde “A” ou “B” podem iniciar uma conexão TCP com o outro.
- **“problema”** - ... se “B” estiver atrás de um Servidor NAT, não será possível aceitar conexões e agir como um servidor !?
- .. pergunta faz sentido pois “B” é identificado por um Endereço IP Privado, cuja validade faz sentido somente na sub-rede interna.



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

- “**solução**” - ... permitir que cliente descubra endereço IP público do servidor, ou seja, configuração automática do NAT.
- Universal Plug and Play (UPnP) Internet Gateway Device (IGD).
- .. descoberta do endereço IP público – 138.76.29.7.
- .. inclusão/remoção de mapeamentos de portas (com tempo de posse), ou seja, automatizar configuração estática do Mapa de Portas do NAT.



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.2 – Endereçamento IPv4

- “**solução**” - “A” se conecta primeiro com “B” através de um outro “host”, p.ex. “C”, que não está por trás de um NAT e com o qual “B” tenha estabelecido uma conexão TCP em curso.
- “A” pode solicitar a “B” por intermédio de “C” que inicie uma Conexão TCP diretamente com ele o que irá permitir a troca de informações.
- ... solução denominada “reversão de conexão”, pois é o servidor representado por “B” que inicia a conexão com um cliente - “A”.



## 4 - Camada de Rede / 4.4 - Protocolo IP

### 4.4.3 – Protocolo ICMP

- ICMP (Internet Control Message Protocol) .. usado por “hosts” e por roteadores para comunicar informações na camada de rede.
- .. relato de erro .. “host”, rede, porta, protocolo inalcançável, etc.
- .. eco de solicitação/resposta (usado por “ping”).
- ICMP é considerado parte do IP, mas em termos de arquitetura está logo acima do IP, assim como TCP e UDP.

ICMP Header Format

Octet	0							1							2							3										
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Type							Code							Checksum																	
32	Rest of Header																															

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.3 – Protocolo ICMP

- ICMP .. carregado como carga útil no IP assim como TCP e UDP, assim o “host” ao receber um datagrama IP com ICMP, demultiplexa o conteúdo do datagrama para ICMP.

**IPv4 Header Format**

Octet	0							1							2							3										
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Version			IHL				DSCP					ECN		Total Length																	
32	Identification															Flags			Fragment Offset													
64	Time To Live							Protocol							Header Checksum																	
96	Source IP Address																															
128	Destination IP Address																															
160	Options (if IHL > 5)																															



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.3 – Protocolo ICMP

- .. msg. ICMP contempla campo de tipo, código e os primeiros 08 bytes do datagrama IP que causou a msg. ICMP, assim, remetente pode determinar o datagrama que causou o erro.
- .. cabeçalho ICMP inicia após o cabeçalho IPv4 e é identificado pelo nro. de protocolo = “1”, ou seja, ICMP.
- .. todos os pacotes ICMP tem cabeçalho de 8 bytes e campo de dados de tamanho variável.
- .. os primeiros 4 bytes tem formato fixo, enquanto os 4 bytes restantes dependem do tipo/código do pacote ICMP.

ICMP Header Format

Octet	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Type								Code								Checksum															
32	Rest of Header																															



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.3 – Protocolo ICMP

- e.g., “ping” envia uma msg. ICMP do tipo “8” e código “0” para um “host” específico, que por sua vez ao receber a msg. de “eco” devolve uma resposta do tipo “0” e código “0”.

Tipo de mensagem ICMP	Código	Descrição
0	0	resposta de eco (para ping)
3	0	rede de destino inalcançável
3	1	hospedeiro de destino inalcançável
3	2	protocolo de destino inalcançável
3	3	porta de destino inalcançável
3	6	rede de destino desconhecida
3	7	hospedeiro de destino desconhecido
4	0	redução da fonte (controle de congestionamento)
8	0	solicitação de eco
9	0	anúncio do roteador
10	0	descoberta do roteador
11	0	TTL expirado
12	0	cabeçalho IP inválido

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.3 – Protocolo ICMP

- e.g., considere o comando de “traceroute”, onde o remetente envia uma série de segmentos UDP ao destino.
- .. 1º datagrama com campo TTL = 1; 2º datagrama com campo TTL = 2, 3º datagrama com campo TTL = 3 e, assim, por diante.
- ... para cada datagrama enviado, o remente recebe uma msg. de resposta e pode na sequência calcular o tempo de ida e volta.
- ... quando o n-ésimo datagrama chega no n-ésimo roteador, o roteador percebe que o datagrama acabou de expirar e envia uma msg. ICMP do tipo “11” e código “0” contendo o nome e IP do roteador.
- ... quando chega ao remetente, msg. obtém do temporizador o tempo de ida e volta da msg. ICMP, nome e IP do n-ésimo roteador.

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.3 – Protocolo ICMP

- “**dúvida**” .. quando o “traceroute” para de enviar segmentos UDP ?
- ... como o TTL é incrementado a cada datagrama enviado, um dos datagramas irá chegar ao destinatário.
- ... como este segmento UDP contém um nro. de porta improvável, “host” de destino devolve msg. ICMP informando que a porta não pode ser alcançada, ou seja, msg. tipo “3” e código “3”.
- ... quando esta msg. ICMP chega ao remetente, o mesmo sabe que não precisa mais enviar segmentos UDP, pois atingiu o destinatário.
- Obs.: “**traceroute**” envia um conjunto de 03 segmentos UDP, o que permite a obtenção de 03 resultados para TTL.

## 4 - Camada de Rede / 4.4 - Protocolo IP

### 4.4.4 – Protocolo IPv6

- “Datagrama IPv6” .. contém cabeçalho de 40 bytes de tamanho fixo e não suporta a fragmentação do datagrama.
- “**endereço**” .. expansão de 32 bits para 128 bits (16 bytes), contempla endereço “multicast”; “unicast” e “anycast”.

Octet	0								1								2								3											
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31				
0	Version				Traffic Class								Flow Label																							
32	Payload Length																Next Header								Hop Limit											
64	Source Address																																			
96																																				
128																																				
160																																				
192	Destination Address																																			
224																																				
256																																				
288																																				

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.4 – Protocolo IPv6

- “**endereçamento**” - expansão de 32 bits para 128 bits (16 bytes), contempla endereço “multicast”; “unicast” e “anycast”.
- “**unicast**” .. associação “um-para-um”, ou seja, cada endereço destino identifica unicamente um único receptor.
- “**multicast**” .. associação “um-para-muitos” (subgrupo do todo), ou seja, datagramas são roteados de um único remetente para múltiplos receptores selecionados em uma única vez em uma única transmissão.

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.4 – Protocolo IPv6

- “**endereçamento**” - expansão de 32 bits para 128 bits (16 bytes), contempla endereço “multicast”; “unicast” e “anycast”;
- “**broadcast**” - associação “um-para-muitos”, ou seja, datagramas são roteados de um único remetente para múltiplos receptores em uma única vez e em uma única transmissão.
- “**anycast**” - associação de “um-para-muitos” de modo que datagramas sejam roteados de um único remetente para um grupo de potenciais receptores que são identificados pelo mesmo endereço.

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.4 – Protocolo IPv6

- “**Datagrama IPv6**” .. contém cabeçalho de 40 bytes de tamanho fixo e não suporta a fragmentação do datagrama.
- “**cabeçalho**” .. comprimento fixo de 40 bytes para permitir processamento mais rápido do datagrama.

Octet	0								1								2								3											
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31				
0	Version				Traffic Class								Flow Label																							
32	Payload Length																Next Header								Hop Limit											
64	Source Address																																			
96																																				
128																																				
160																																				
192	Destination Address																																			
224																																				
256																																				
288																																				

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.4 – Protocolo IPv6

- “**versão**” (4 bits) .. nro. de versão do IP » “6” para IPv6.
- “**classe de tráfego**” (8 bits) .. pode ser usado para dar prioridade a certos datagramas dentro de um fluxo ou de certas aplicações.
- “**rótulo de fluxo**” (20 bits) .. permite identificar um fluxo de datagramas para o qual se deseja tratamento particular.

Octet	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Version				Traffic Class								Flow Label																			
32	Payload Length																Next Header								Hop Limit							
64	Source Address																															
96																																
128																																
160																																
192	Destination Address																															
224																																
256																																
288																																



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.4 – Protocolo IPv6

- **“Payload Length”** (16 bits) .. nro. de bytes do datagrama IPv6 que se segue ao cabeçalho, ou seja, aos 40 bytes de cabeçalho.
- **“Next Header”** (8 bits) .. especifica o tipo do próximo cabeçalho, mas também pode identificar o protocolo para o qual deve ser entregue a carga ou “payload” do datagrama.

Octet	0								1								2								3							
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	Version				Traffic Class								Flow Label																			
32	Payload Length																Next Header								Hop Limit							
64	Source Address																															
96																																
128																																
160																																
192	Destination Address																															
224																																
256																																
288																																

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.4 – Protocolo IPv6

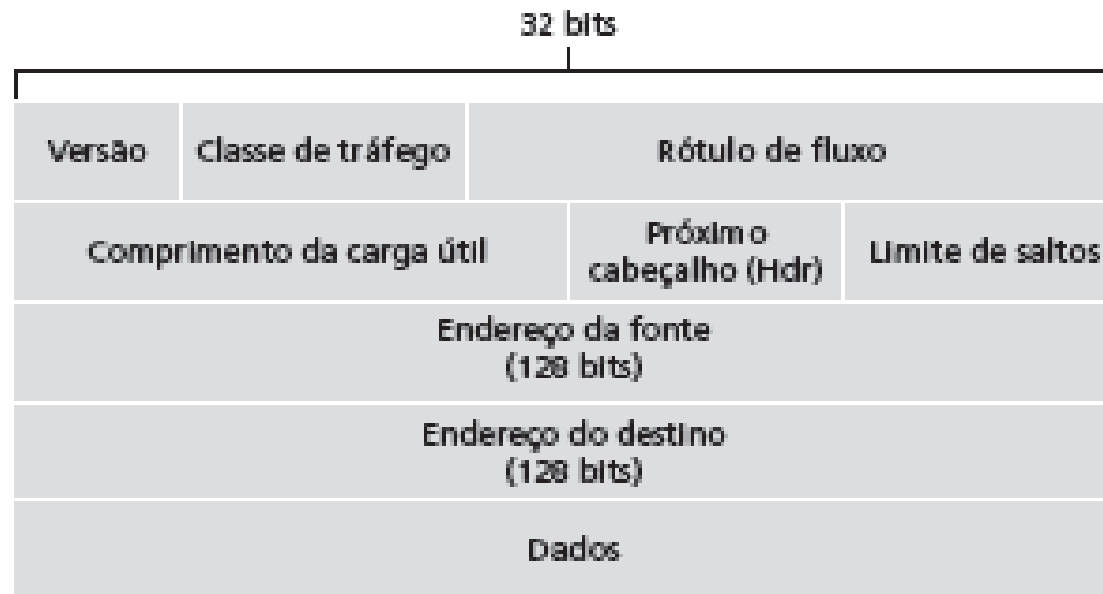
- “**Hop Limit**” .. decrementado de “1” em cada roteador.
- “**endereços**” .. expansão de 4 bytes para 16 bytes, contempla endereçamento “multicast”; “unicast” e “anycast”, ou seja, datagrama é entregue a qualquer “host” de um grupo.

Octet	0								1								2								3											
Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31				
0	Version				Traffic Class								Flow Label																							
32	Payload Length																Next Header								Hop Limit											
64	Source Address																																			
96																																				
128																																				
160																																				
192	Destination Address																																			
224																																				
256																																				
288																																				

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.4 – Protocolo IPv6

- “**endereços**” .. expansão de 4 bytes para 16 bytes, contempla endereçamento “multicast”; “unicast” e “anycast”, ou seja, datagrama é entregue a qualquer “host” de um grupo;
- “**dados**” .. carga útil do datagrama IPv6 não é manipulada pelo IPv6, mas simplesmente extraída do datagrama e entregue ao protocolo especificado no campo de próximo cabeçalho – “**next header**”.



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.4 – Protocolo IPv6

- Campos NÃO presentes no IPv6:
- “**fragmentação / remontagem**” .. estas operações podem ser realizadas somente pelo remetente e destinatário.
- ... se não for possível processar o datagrama no roteador, o roteador descarta e na sequência envia uma msg. ICMP de erro ao remetente.
- “**soma verificação**” .. funcionalidade considerada redundante na camada de rede, uma vez que tanto a camada de transporte quanto a camada de enlace contemplam tal funcionalidade.
- “**opções**” .. não faz parte do cabeçalho padrão, mas também não foi excluído, esta presente no cabeçalho como “próximo cabeçalho.
- ... sugere que o IPv6 poderá comportar adição no cabeçalho, assim como nos cabeçalhos dos protocolos TCP e UDP.

## 4 - Camada de Rede / 4.4 - Protocolo IP

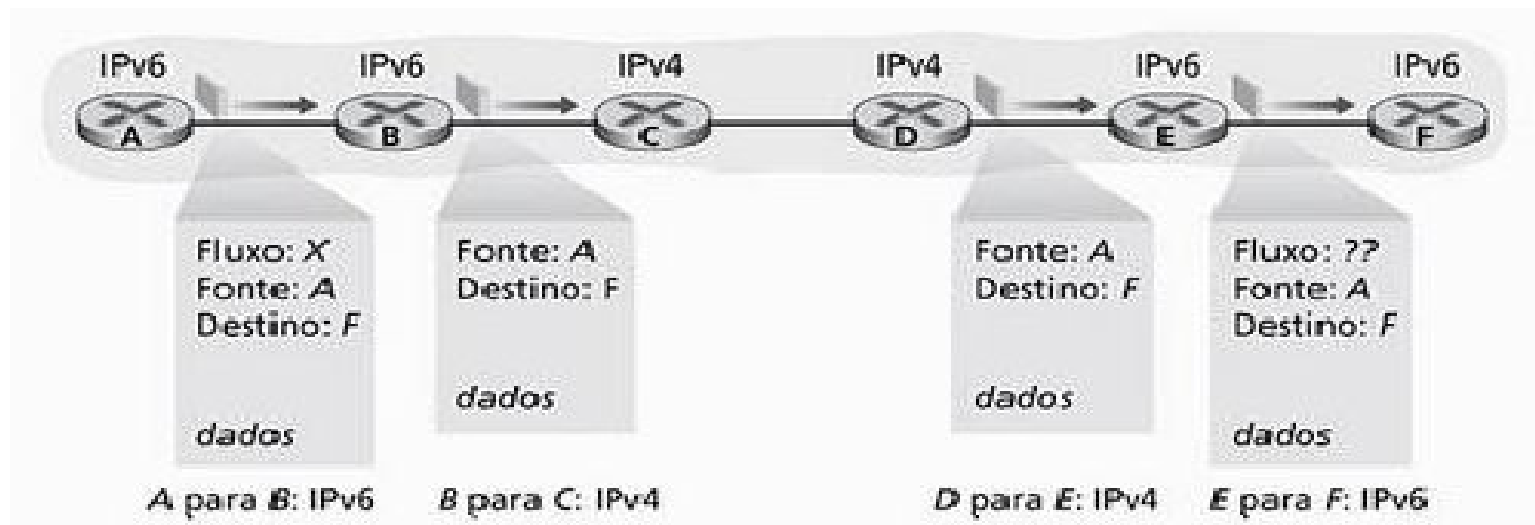
### ... 4.4.4 – Protocolo IPv6

- “**problema**” .. IPv6 pode receber e rotear datagramas IPv4, mas o IPv4 não pode manusear datagramas IPv6.
- “**dia da conversão**” .. modificar em uma data específica todas as máquinas da Internet através da atualização do IPv4 para IPv6.
- “**pilha dupla**” .. permitir que nós IPv6 contemplem uma implementação completa do IPv4 para que possam, em determinados momentos, enviar e receber datagramas IPv4 além de IPv6.
- ... datagramas IPv6 serão transportados como carga útil em datagramas IPv4 quando for necessário, não obstante, depende da habilidade dos roteadores e “hosts” no que se refere a conversão.

## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.4 – Protocolo IPv6

- e.g., “A” usa IPv6 e deseja enviar datagramas ao nó “F” que também utiliza IPv6, contudo ao longo do caminho há roteadores que não conseguem rotear datagramas IPv6 !!??
- “**solução**” - conversão de datagramas IPv6 para IPv4, no entanto, contempla alguns problemas, pois nem todos os campos no IPv6 ou IPv4 estão presentes em ambos os formatos.



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.4 – Protocolo IPv6

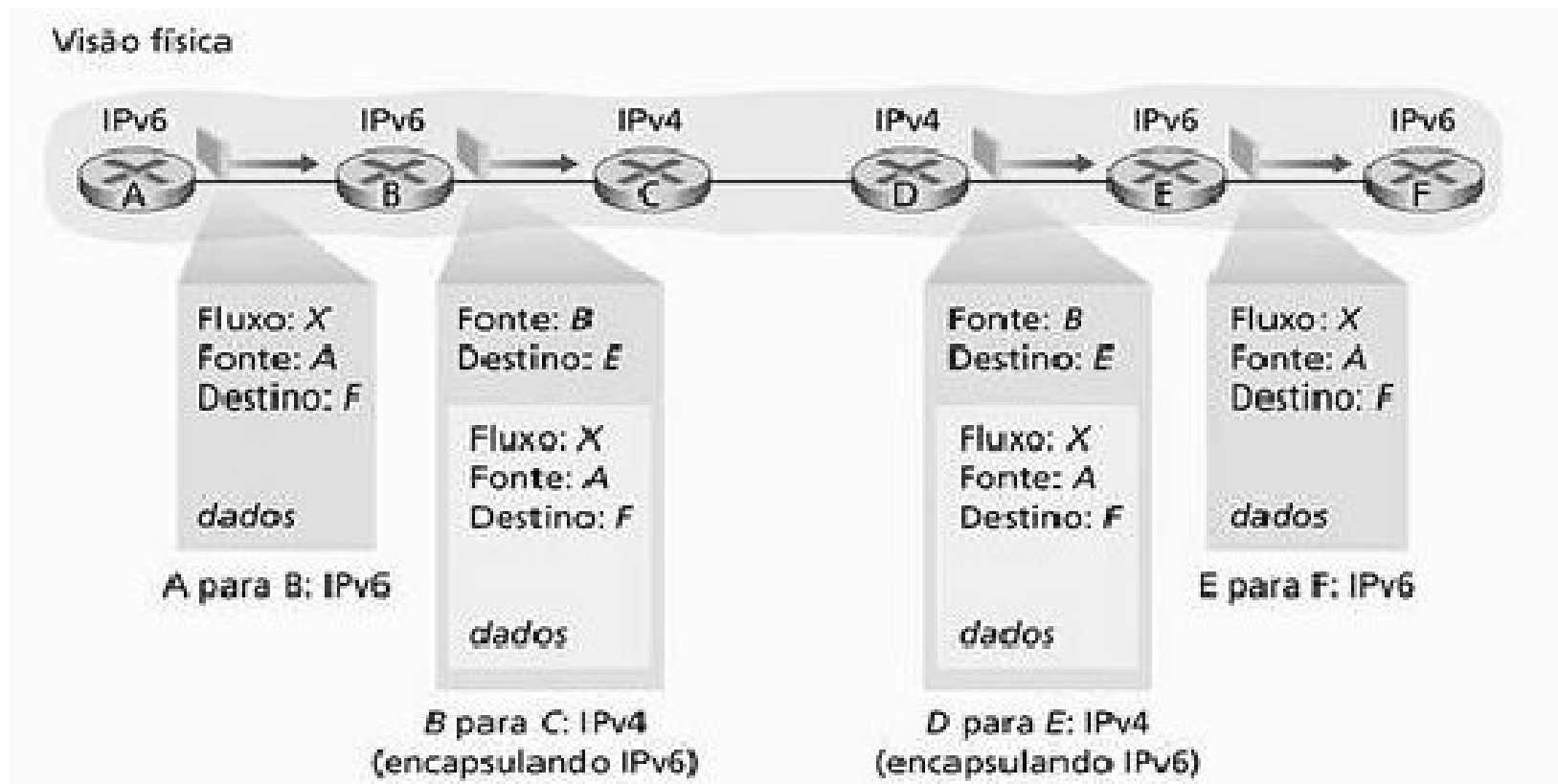
- e.g., “A” usa IPv6 e deseja enviar datagramas ao nó “F” que também utiliza IPv6, contudo ao longo do caminho há roteadores que não conseguem rotear datagramas IPv6 !!??
- “**alternativa**” .. permitir que datagramas IPv6 sejam encapsulados em datagramas IPv4, ou seja, datagrama IPv6 = carga útil do IPv4.
- .. visão lógica do encapsulamento de datagramas IPv6 em IPv4.



## 4 - Camada de Rede / 4.4 - Protocolo IP

### ... 4.4.4 – Protocolo IPv6

- “**IPv6 em IPv4**” .. permitir que datagramas IPv6 sejam encapsulados em datagramas IPv4, ou seja, carga útil do IPv4.
- .. visão física do encapsulamento de datagramas IPv6 em IPv4.





## 4.5 – Algoritmos de Roteamento

- **“roteador”** .. indexa tabela de repasse e determina a interface de saída para a qual o pacote deve ser encaminhado.
- **“algoritmos de roteamento”** - trocam e calculam informações que são utilizadas para configurar as tabelas de repasse.
- **“camada de rede”** .. determina o caminho que os pacotes percorrem entre remetentes e destinatários.
- **“serviços de datagrama”** .. pacotes diferentes podem percorrer rotas diferentes para o mesmo par remetente e destinatário.
- **“serviço de circuitos virtuais”** .. pacotes diferentes percorrem a mesma rota para o mesmo par remetente e destinatário.

## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

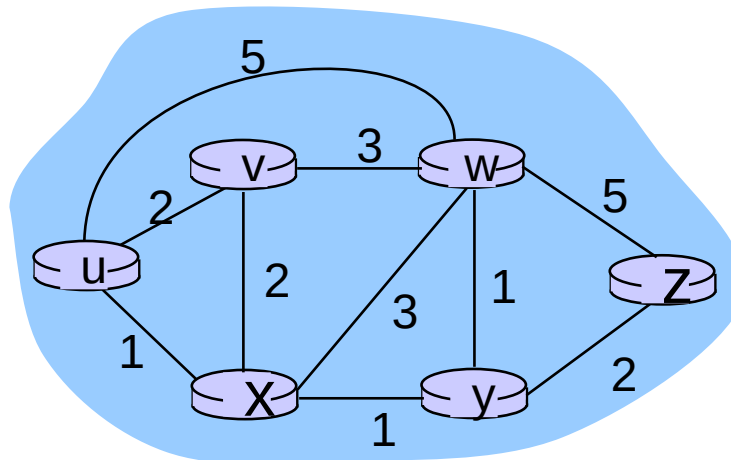
### ... 4.5 – Algoritmos de Roteamento

- “**roteador default**” ou “**roteador de 1º salto**” .. roteador na sub-rede e no qual está ligado o “host” remetente ou destinatário.
- “roteador da fonte” .. roteador “default” para o “host” remetente.
- “roteador do destino” .. roteador “default” para o “host” destino.
- “**problema**” .. rotear pacotes do “host” remetente para o “host” destinatário se reduz ao problema de rotear pacotes do “roteador da fonte” para o “roteador de destino”.
- “**objetivo**” .. dado um conjunto de roteadores conectados por enlaces, deseja-se descobrir um caminho do roteador fonte para o roteador de destino segundo uma função de custo.

## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5 – Algoritmos de Roteamento

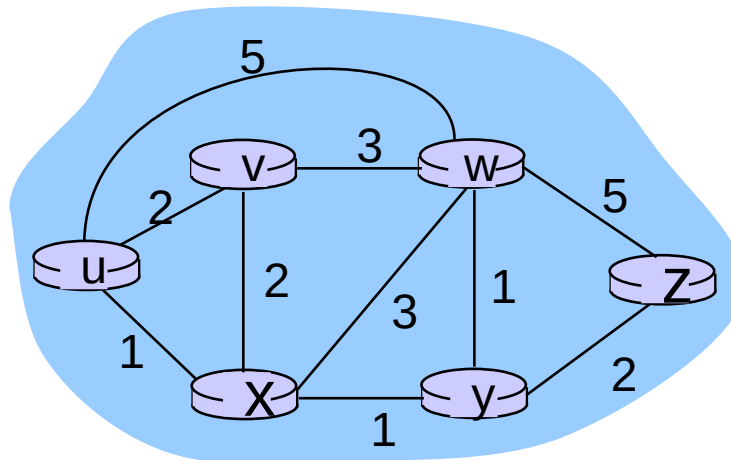
- “Grafo  $G = (N,E)$ ” .. conjunto de “N” nós e “E” arestas no qual cada aresta é um par de nós do conjunto “N” de todos os nós.
- ... no contexto do roteamento da camada de rede, os nós do grafo representam roteadores e as arestas representam os enlaces físicos.
- e.g., Seja o Grafo  $G = (N,E)$ , onde  $N = \{ u, v, w, x, y, z \}$  e
- $E = \{ (u,v), (u,x), (u,w), (v,x), (v,w), (x,w), (x,y), (w,y), (w,z), (y,z) \}$



## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5 – Algoritmos de Roteamento

- “**caminho**” - sequência de nós “ $(x_1, x_2, \dots, x_p)$ ” tal que cada um dos pares “ $(x_1, x_2)$ ”, “ $(x_2, x_3)$ ”, ..., “ $(x_{p-1}, x_p)$ ” são arestas em “E”.
- “**considerações**” - para qualquer aresta “ $(x,y)$ ” em “E”, “ $c(x,y)$ ” representa o custo da aresta entre os nós “x” e “y”;
- .. grafos são direcionais, ou seja, uma aresta “ $(x,y)$ ” é igual a aresta “ $(y,x)$ ”, assim como “ $c(x,y)$ ” é igual a “ $c(y,x)$ ”.
- .. custo do caminho “ $(x_1, x_2, \dots, x_p)$ ” é a soma de todos os custos dos pares “ $(x_k, x_{k+1})$ ”, ou seja, “ $c(x_1, x_2) + \dots + c(x_{p-1}, x_p)$ ”.



## ... 4.5 – Algoritmos de Roteamento

- “**hipótese**” .. se todas as arestas do grafo tiverem o mesmo custo, encontrar o caminho de menor custo significa encontrar o caminho no grafo com o menor nro. de enlaces entre fonte e destino.
- “**função custo**” ... pode ser inversamente proporcional à largura de banda do enlace ou proporcional ao congestionamento do mesmo.
- Qual o caminho de MENOR CUSTO ??
- ... diferentes algoritmos fornecem diferentes respostas em razão da função custo utilizada para encontrar o caminho de menor custo.
- “**algoritmos de roteamento**” .. trocam e calculam informações que são utilizadas para configurar as tabelas de repasse.

## ... 4.5 – Algoritmos de Roteamento

- **“algoritmo de roteamento global”** - calcula o caminho de menor custo entre fonte e destino com informações de toda a rede, ou seja, considera-se todos os nós e custos de todos os enlaces.
- ... são também denominados de “algoritmos de estado de enlace” ou “algoritmos com informação global de estado de enlace”.
- **“algoritmo de roteamento descentralizado”** - calcula-se o caminho de menor custo usando o conhecimento parcial da rede, ou seja, não há informação dos custos de todos os enlaces.
- ... são também denominados de “algoritmo de vetor de distâncias” pois cada nó mantém um vetor de estimativas de custos (distâncias) de um nó até todos os outros nós da rede.

## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5 – Algoritmos de Roteamento

- “**algoritmos de roteamento estático**” .. rotas mudam muito lentamente ao longo do tempo, muitas vezes por intervenção humana.
- “**algoritmos de roteamento dinâmico**” .. rotas mudam à medida que mudam as cargas de tráfego ou mesmo a topologia da rede.
- “**algoritmo sensível à carga**” .. custos dos enlaces variam dinamicamente para refletir o nível corrente de congestionamento, com tendência a escolha de rotas que evitem tais enlaces.
- “**algoritmo insensível à carga**” .. custos dos enlaces não refletem explicitamente seu nível de congestionamento, assim, a escolha de rota não considera o congestionamento dos enlaces.

## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### 4.5.1 – Algoritmo de Roteamento Link-State

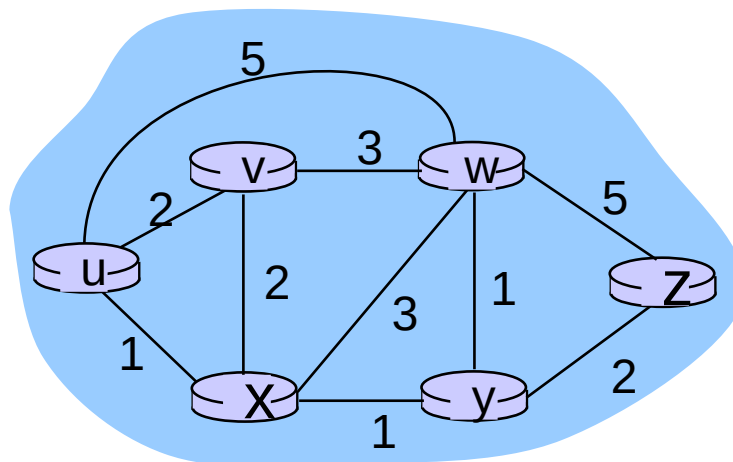
- **“Algoritmo de Dijkstra”** .. tanto a topologia da rede quanto os custos dos enlaces são conhecidos por todos os nós.
- ... normalmente, estas informações são encaminhadas a cada um dos nós através de “broadcast de estado do enlace”.
- ... adicionalmente, todos os nós compartilham, o mesmo conjunto de informações da rede, permitindo que tenham um visão idêntica.
- ... cada nó pode com base nas informações, rodar o algoritmo e calcular o conjunto de caminhos de menor custo para todos os nós.
- ... algoritmo iterativo que tem a propriedade de, após a k-ésima iteração, conhecer os caminhos de menor custo para “k” nós de destino, dentre os caminhos de menor custo até todos os nós de destino.



## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.1 – Algoritmo de Roteamento Link-State

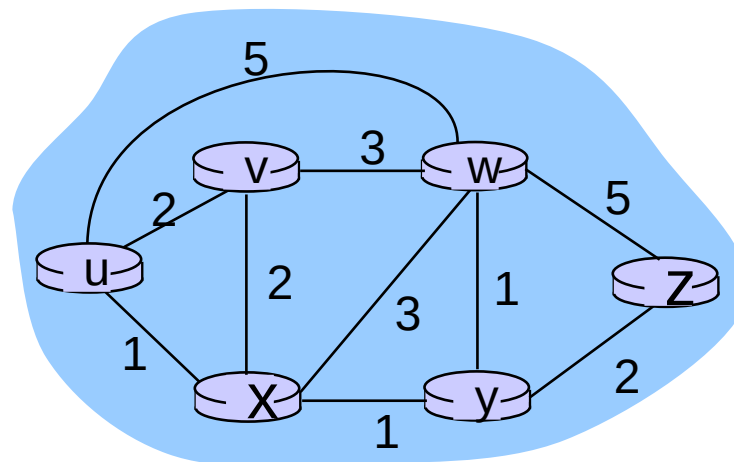
- “**notação de termos**” .. Algoritmo de Dijkstra
- $c(x,y)$  .. custo do enlace do nó “x” a “y”, normalmente representado por um inteiro quando vizinhos e “infinito” se não vizinhos.
- $D_x(v)$  .. custo do caminho de menor custo entre o nó fonte “x” e o nó destino “v” até a iteração corrente do algoritmo.



## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.1 – Algoritmo de Roteamento Link-State

- “**notação de termos**” .. Algoritmo de Dijkstra
- $p(v)$  .. nó anterior ou vizinho de “v” ao longo do caminho corrente de menor custo desde o nó fonte “x” até o nó “v”.
- $N'$  .. subconjunto de nós do conjunto de TODOS os NÓS “N” cujo caminho de menor custo é conhecido (pode ser calculado).



## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.1 – Algoritmo de Roteamento Link-State

- Algoritmo de Estado de Enlace .. nó fonte “u”

01 .. inicialização ..

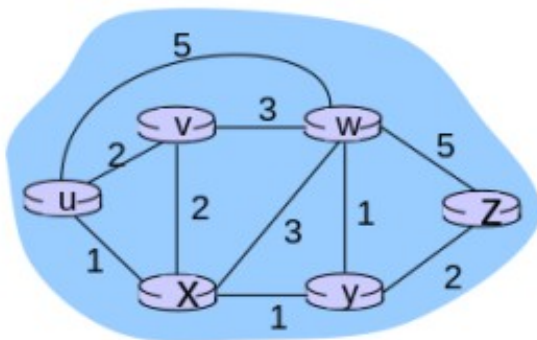
02  $N' = \{u\}$

03 para todos os nós “k” {

04     então  $D(k) = c(u, k)$  /\* se “k” for um vizinho de “u” \*/

05     senão  $D(k) = \text{“infinito”}$  /\* se “k” não é vizinho de “u” \*/

06 }

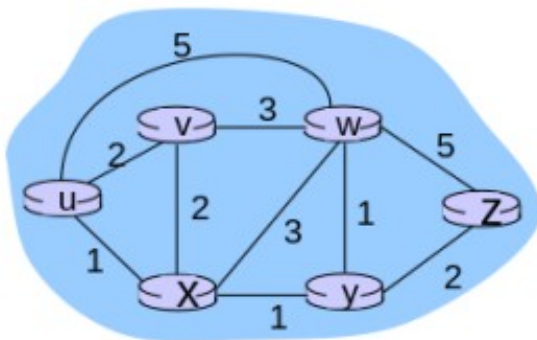


Etapa	$N'$	$D(v), p(v)$	$D(w), p(w)$	$D(x), p(x)$	$D(y), p(y)$	$D(z), p(z)$
0	u	2,u	5,u	1,u	$\infty$	$\infty$
1						
2						
3						
4						
5						

## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.1 – Algoritmo de Roteamento Link-State

- Seja alguns dos estágios de execução do Alg. Dijkstra
- ... na inicialização, os caminhos de menor custo correntemente conhecidos a partir de “u” até os vizinhos diretamente ligados “v”, “x” e “w” são inicializados para 2, 1 e 5 respectivamente.
- ... custos até “y” e “z” são estabelecidos como “infinito”, pois estes nós não estão diretamente conectados a “u”.



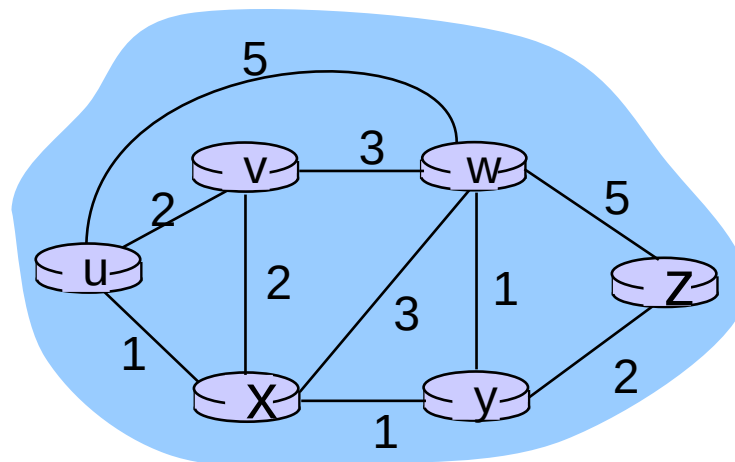
Etapas	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)
0	u	2,u	5,u	1,u	$\infty$	$\infty$
1						
2						
3						
4						
5						

## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.1 – Algoritmo de Roteamento Link-State

- Algoritmo de Estado de Enlace .. nó fonte “u”

```
07 LOOP {  
08   encontre “g” não em N', tal que D(g) é mínimo  
09   adicione “g” a N'  
10   atualize D(k) para cada vizinho “k” de “g” e não em N'  
11    $D(k) = \min\{ D(k), D(g) + c(g, k) \}$   
12   /* novo custo para “k” é o velho custo para “k” ou o custo do */  
13   /* menor caminho conhecido para “w” + custo de “w” para “k” */  
14 } ATÉ N' - N
```



## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

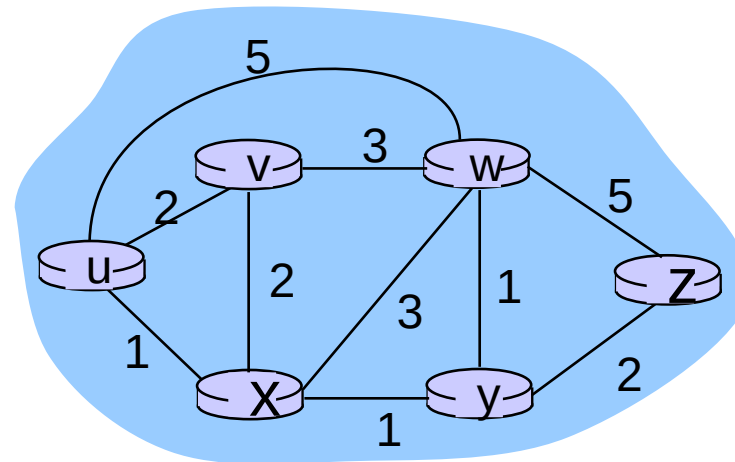
### ... 4.5.1 – Algoritmo de Roteamento Link-State

- e.g., “**alguns dos estágios de execução**” do Alg. Dijkstra ..
- “1ª Iteração” .. ao examinar os nós que ainda não foram adicionados a  $N'$ , descobre-se que o nó de menor custo ao final da Iteração #0 é “x”;
- ... adiciona-se “x” a  $N'$  e na sequência atualiza-se “ $D(k)$ ” para cada vizinho “k” de “x” e que não pertença ao conjunto  $N'$ .

```
07 LOOP {
08   encontre “g” não em  $N'$ , tal que  $D(g)$  é mínimo
09   adicione “g” a  $N'$ 
10   atualize  $D(k)$  para cada vizinho “k” de “g” e não em  $N'$ 
11    $D(k) = \min\{ D(k), D(g) + c(g, k) \}$ 
12   /* novo custo para “k” é o velho custo para “k” ou o custo do */
13   /* menor caminho conhecido para “w” + custo de “w” para “k” */
14 } ATÉ  $N' - N$ 
```

## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.1 – Algoritmo de Roteamento Link-State



Etapa	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)
0	u	2,u	5,u	1,u	$\infty$	$\infty$
1	ux	2,u	4,x		2,x	$\infty$
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					

## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.1 – Algoritmo de Roteamento Link-State

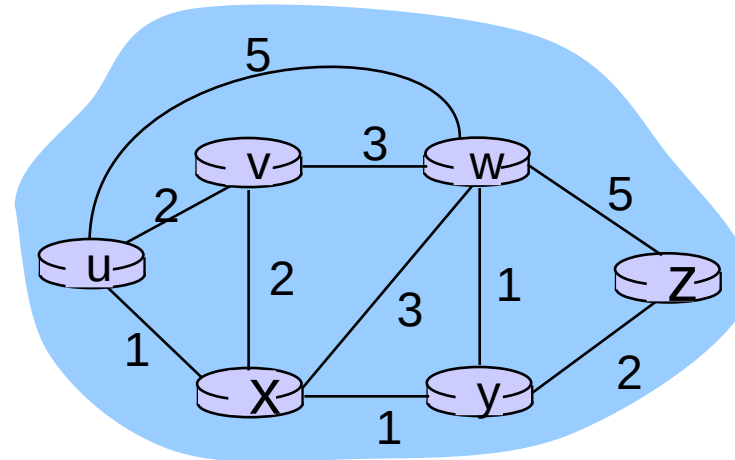
- e.g., “**alguns dos estágios de execução**” do Alg. Dijkstra ..
- “2ª Iteração” - nós “v” e “y” são os que tem os caminhos de menor custo e em razão do empate escolhe-se arbitrariamente um dos dois.

```
07 LOOP {
08   encontre “g” não em N', tal que D(g) é mínimo
09   adicione “g” a N'
10   atualize D(k) para cada vizinho “k” de “g” e não em N'
11    $D(k) = \min\{ D(k), D(g) + c(g, k) \}$ 
12   /* novo custo para “k” é o velho custo para “k” ou o custo do */
13   /* menor caminho conhecido para “w” + custo de “w” para “k” */
14 } ATÉ N' - N
```



## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.1 – Algoritmo de Roteamento Link-State



Etapa	N'	D(v), p(v)	D(w), p(w)	D(x), p(x)	D(y), p(y)	D(z), p(z)
0	u	2,u	5,u	1,u	$\infty$	$\infty$
1	ux	2,u	4,x		2,x	$\infty$
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					

## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.1 – Algoritmo de Roteamento Link-State

- e.g., “**alguns dos estágios de execução**” do Alg. Dijkstra ..
- “1ª Iteração” .. ao examinar os nós que ainda não foram adicionados a  $N'$ , descobre-se que o nó de menor custo ao final da Iteração #0 é “x”;
- ... adiciona-se “x” a  $N'$  e na sequência atualiza-se “ $D(k)$ ” para cada vizinho “k” de “x” e que não pertença ao conjunto  $N'$ .
- “2ª Iteração” - nós “v” e “y” são os que tem os caminhos de menor custo e em razão do empate escolhe-se arbitrariamente um dos dois.

Etapa	$N'$	$D(v), p(v)$	$D(w), p(w)$	$D(x), p(x)$	$D(y), p(y)$	$D(z), p(z)$
0	u	2,u	5,u	1,u	$\infty$	$\infty$
1	ux	2,u	4,x		2,x	$\infty$
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					

## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.1 – Algoritmo de Roteamento Link-State

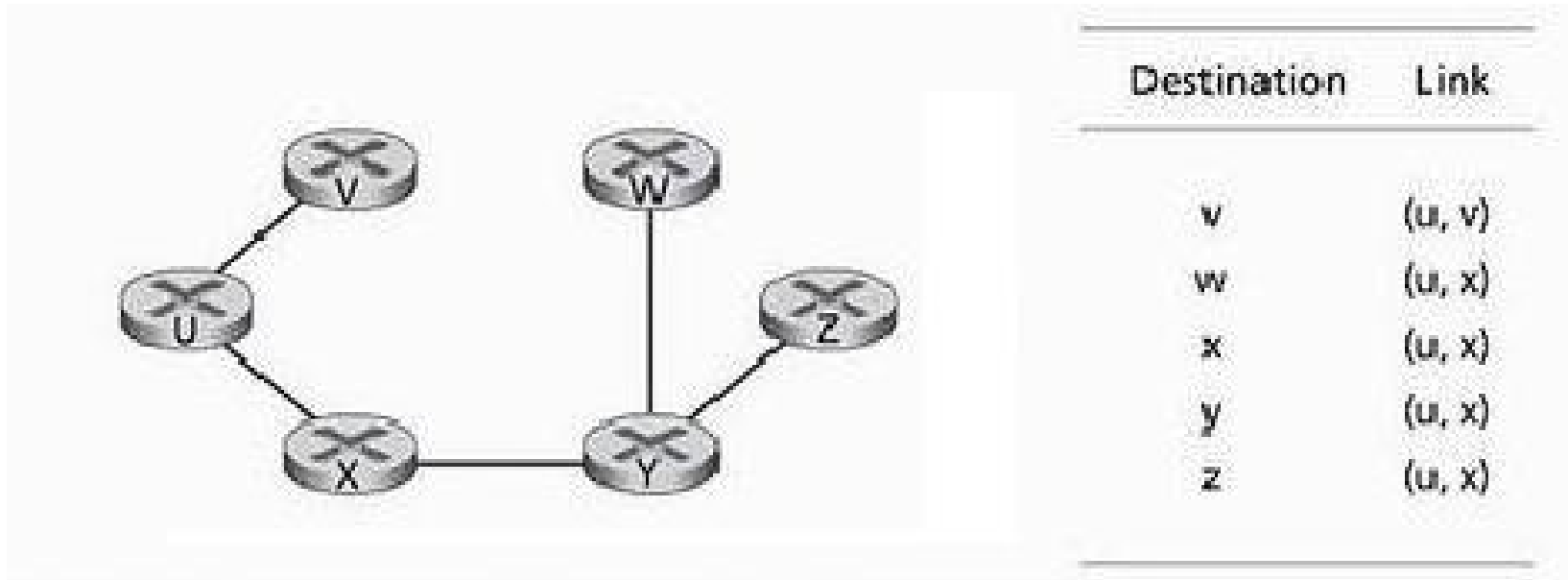
- e.g., “**alguns dos estágios de execução**” do Alg. Dijkstra ..
- ... quando o algoritmo termina, temos para cada nó, seu predecessor ao longo do caminho de menor custo a partir do nó da fonte.
- ... tem-se também o predecessor para cada um desses predecessores, assim, podemos contruir o caminho inteiro desde a fonte até os demais.

<b>Etapas</b>	<b>N'</b>	<b>D(v), p(v)</b>	<b>D(w), p(w)</b>	<b>D(x), p(x)</b>	<b>D(y), p(y)</b>	<b>D(z), p(z)</b>
0	u	2,u	5,u	1,u	$\infty$	$\infty$
1	ux	2,u	4,x		2,x	$\infty$
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					

## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.1 – Algoritmo de Roteamento Link-State

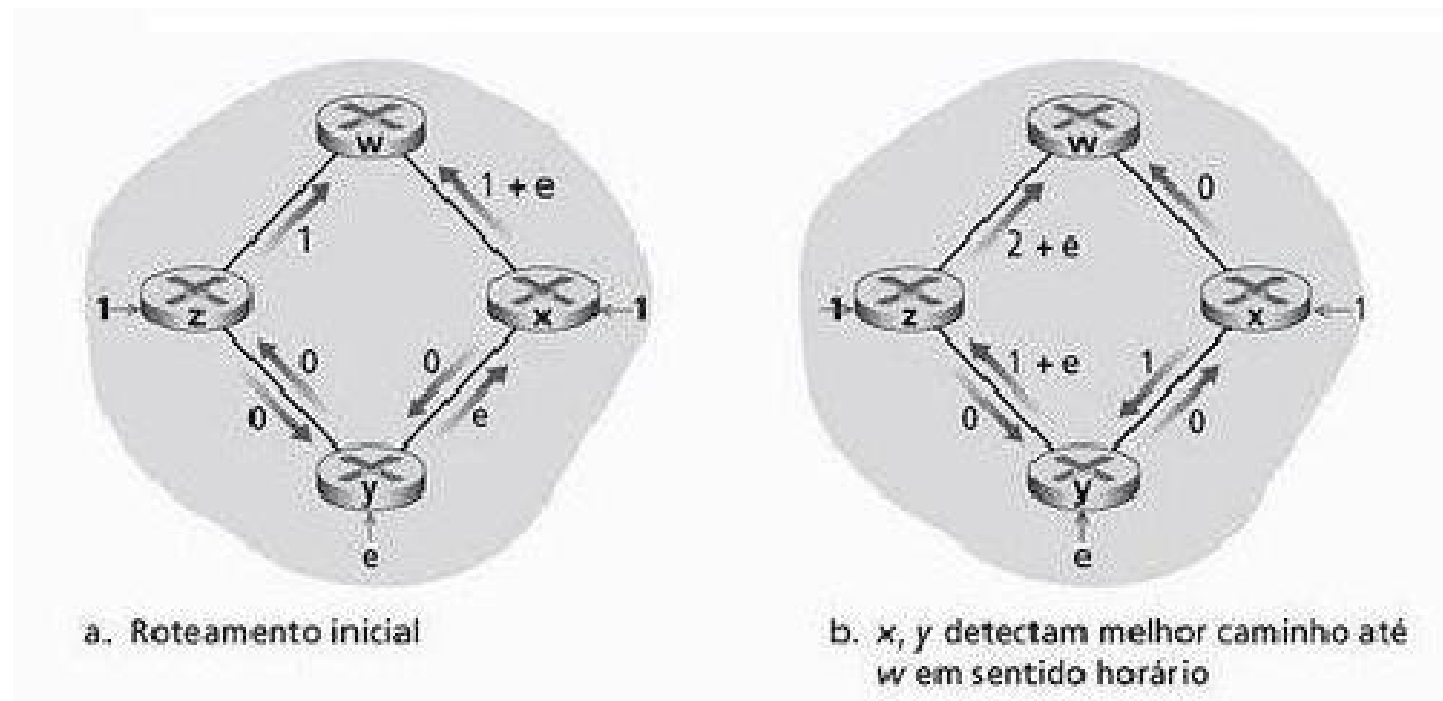
- Qual é a complexidade do cálculo do Alg. de Dijkstra ?
- ... em termos gerais, o nro. total de nós que precisamos pesquisar em todas as iterações é " $n*(n+1)/2$ " e, assim, a complexidade do algoritmo de estado de enlace para o pior caso é de ordem " $n^2$ " ou  $O(n^2)$ .
- Caminhos de MENOR CUSTO resultantes da execução do algoritmo bem como a tabela de repasse do nó "u" para a rede.



## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.1 – Algoritmo de Roteamento Link-State

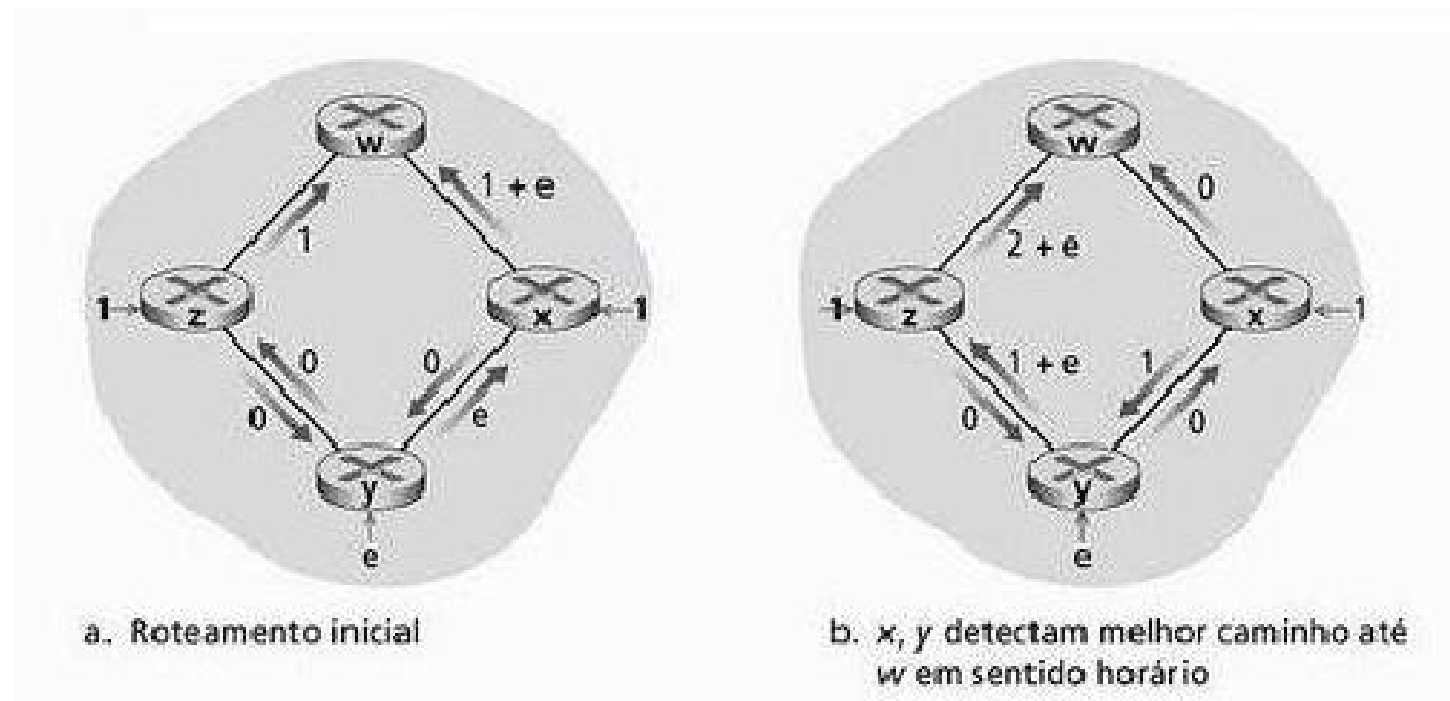
- e.g., considere uma topologia onde os custos dos enlaces são iguais à carga transportada pelo enlace (diretamente proporcional).
- ... além disso, neste contexto, os custos dos enlaces não são simétricos, isto é, “ $c(u,v)$ ” é igual a “ $c(v,u)$ ” somente se a carga transportada em ambos os sentidos for a mesma.



## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.1 – Algoritmo de Roteamento Link-State

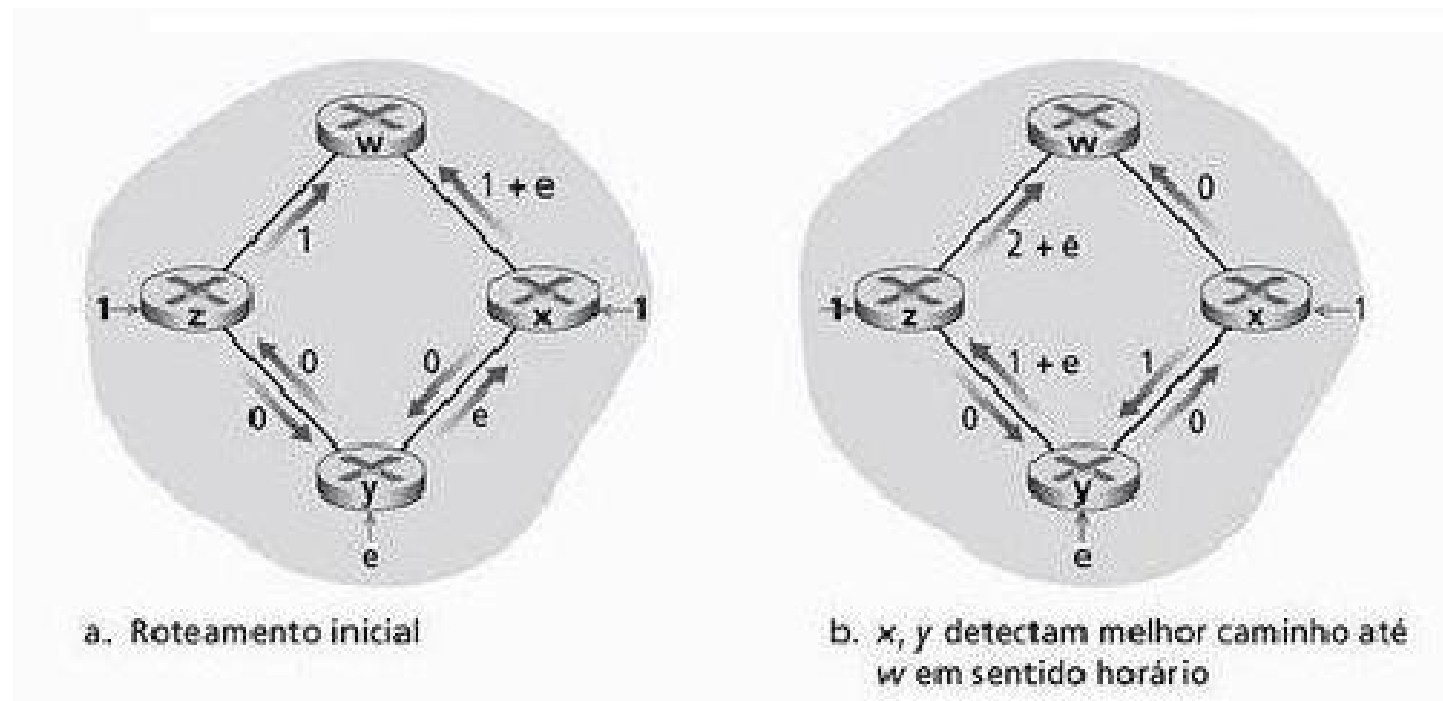
- ... nó “z” origina uma unidade de tráfego destinada a “w”; nó “x” origina uma unidade destinada a “w”.
- ... após o algoritmo ser executado, os nós “x”, “y” e “z” detectam um caminho de custo zero até “w” no sentido anti-horário e todos direcionam o seu tráfego para as rotas anti-horárias.



## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.1 – Algoritmo de Roteamento Link-State

- ... na próxima vez em que o algoritmo for executado, os nós “x”, “y” e “z” detectam um caminho de custo zero até “w” no sentido horário e todos direcionam o seu tráfego para as rotas horárias.
- **“problema”** ... como evitar tais oscilações ?!?!



## ... 4.5.1 – Algoritmo de Roteamento Link-State

- **“Solução #1”** - custo dos enlaces não dependa da intensidade de tráfego, por outro lado, um dos objetivos do roteamento é evitar enlaces muito congestionados, logo, necessário repensar !!
- **“Solução #2”** - assegurar que os roteadores (todos) não executem o algoritmo sincronamente ou de forma coordenada, embora possam executar com a mesma periodicidade.
- **“resultado de pesquisas”** .. mostram que roteadores na Internet podem se auto sincronizar, mesmo que inicialmente executem o algoritmo com o mesmo período, mas em diferentes momentos.
- ... para evitar essa auto-sincronização, cada roteador pode variar aleatoriamente o momento em que envia um anúncio de enlace.



## 4.5.2 – Algoritmo de Roteamento Distance Vector

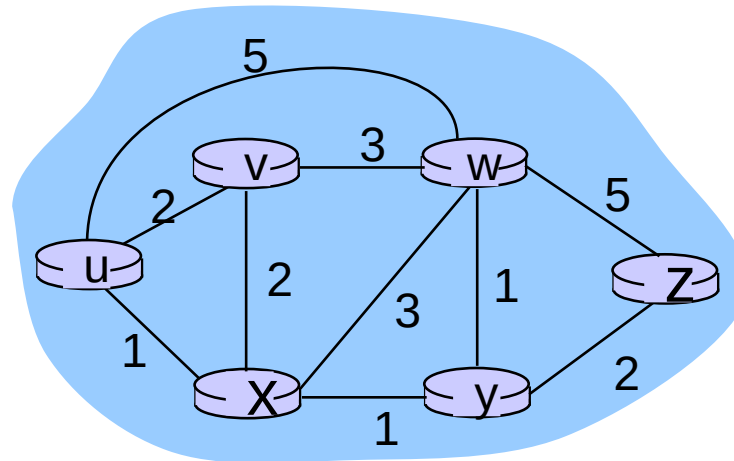
- “**Alg. Vetor de Distâncias**” - algoritmo distribuído onde cada nó tem informação parcial da topologia da rede, normalmente de nós vizinhos e sobre os quais calcula as rotas.
- “**iterativo**” .. troca de informações acerca da topologia é contínua até que os nós vizinhos não tenham mais informações para trocar.
- “**assíncrono**” .. nós executam o algoritmo quando detectam alteração nos enlaces ou quando recebem novas métricas dos vizinhos, ou seja, nós executam o algoritmo de modo assíncrono.

## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- **“notação de termos”** .. Algoritmo Vetor de Distâncias.
- $d_x(y)$  .. custo do caminho de menor custo do nó “x” ao nó “y” cujo valor é dado pela Equação de Bellman-Ford ...  $\min_v \{ c(x,v) + d_v(y) \}$  onde “ $\min_v$ ” é calculado para todos os vizinhos de “x”.
- ... ao transitarmos do nó “x” para nó “y” pelo nó “v”, o caminho de menor custo de “x” a “y” será o caminho com o menor valor da expressão  $c(x,v) + d_v(y)$  para todos os vizinhos “v”.

## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- e.g., Seja o Grafo  $G = (N, E)$ , onde  $N = \{ u, v, w, x, y, z \}$  e  $E = \{ (u,v), (u,x), (u,w), (v,x), (v,w), (x,w), (x,y), (w,y), (w,z), (y,z) \}$
- ... nó “u” tem como vizinhos os nós “v”, “x” e “w” e para vários caminhos do grafo é fácil perceber que  $d_v(z) = 5$ ;  $d_x(z) = 3$ ;  $d_w(z) = 3$ .
- ... ao considerar estes valores na Eq. Bellman-Ford juntamente com  $c(u,v) = 2$ ,  $c(u,x) = 1$ , e  $c(u,w) = 5$  teremos o caminho de menor custo.
- **“caminho de menor custo”** ..  $\min \{ 2+5; 1+3; 5+3 \} = 4$



## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- “Eq. Bellman-Ford” - cada nó começa com  $D_x(y)$ , ou seja, uma estimativa do custo do caminho de menor custo do próprio nó “x” e o nó “y” para todos os nós em “N”.
- $D_x = [ D_x(y) : y \text{ em } N ]$  .. vetor de distâncias do nó “x”, ou seja, vetor de estimativas de custo de “x” até todos os outros nós, “y” em “N”;
- ... cada nó mantém o custo para cada um dos seus vizinhos, ou seja,  $c(x,v)$  com o qual está diretamente ligado.
- ... cada nó mantém  $D_x = [ D_x(y) : y \text{ em } N ]$  contendo a estimativa de “x” para seus custos até todos os destinos “y” em “N”.
- ... cada nó mantém os vetores de distâncias de seus vizinhos, isto é,  $D_v = [ D_v(y) : y \text{ em } N ]$  para cada vizinho “v” em “N”

## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- Alg. Vetor de Distância (DV):

```
01 .. inicialização ..
02  para todos os destinos “y” em N
03     $D_x(y) = c(x, y)$  /* se “y” não é um vizinho,  $c(x, y) = \text{infinito}$  */
04  para cada vizinho “w”
05     $D_w(y) = ??$  para todos os destinos “y” em N
06  para cada vizinho em “w”
07    envia um vetor de distâncias  $D_x = [ D_x(y): y ]$  para “w”
08  ...
09 LOOP
...
...
20 FOREVER
```

## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- Alg. Vetor de Distância (DV):

```
08 ...
09 LOOP
10 wait( até que ocorra mudança no custo do enlace ao vizinho “w”
11       ou até a recepção de um vetor de distâncias do vizinho “w” )
12
13 para cada “y” em N
14    $D_x(y) = \min_v \{ c(x, v) + D_v(y) \}$ 
15
16 if  $D_x(y)$  mudou para algum destino “y”
17   envia vetor de distâncias  $D_x = [D_x(y): y \text{ em } N]$ 
18   (cont.) .. para todos os vizinhos !!
19
20 FOREVER
```

## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- “**estado de enlace**” .. algoritmo global no sentido de exige que cada nó obtenha em primeiro lugar, um mapa completo da rede antes de executar o Algoritmo de Dijkstra.
- ... OSPF
- “**vetor de distâncias**” .. algoritmo descentralizado onde cada nó utiliza informações que recebe de seus vizinhos, ou seja, não há um mapa completo da rede antes de rodar o Alg. DV.
- ... RIP; BGP; ISO IDRP, IPX da Novell; ARPAnet da Arq. TCP/IP

## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- e.g., considere 03 nós interconectados por 03 enlaces cujos custos são apresentados abaixo - Alg. de Vetor de Distância – “x” e “y”.

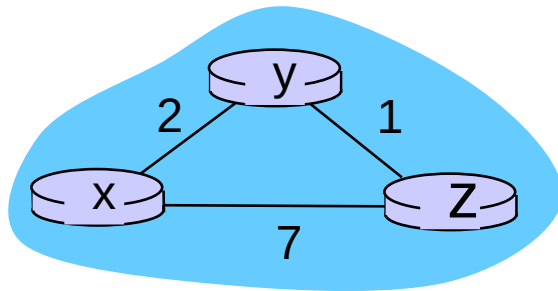


Tabela do nó x

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

		Custo até		
		x	y	z
Destino	x	0	2	3
	y	2	0	1
	z	7	1	0

		Custo até		
		x	y	z
Destino	x	0	2	3
	y	2	0	1
	z	3	1	0

Tabela do nó y

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

		Custo até		
		x	y	z
Destino	x	0	2	7
	y	2	0	1
	z	7	1	0

		Custo até		
		x	y	z
Destino	x	0	2	3
	y	2	0	1
	z	3	1	0

Tabela do nó z

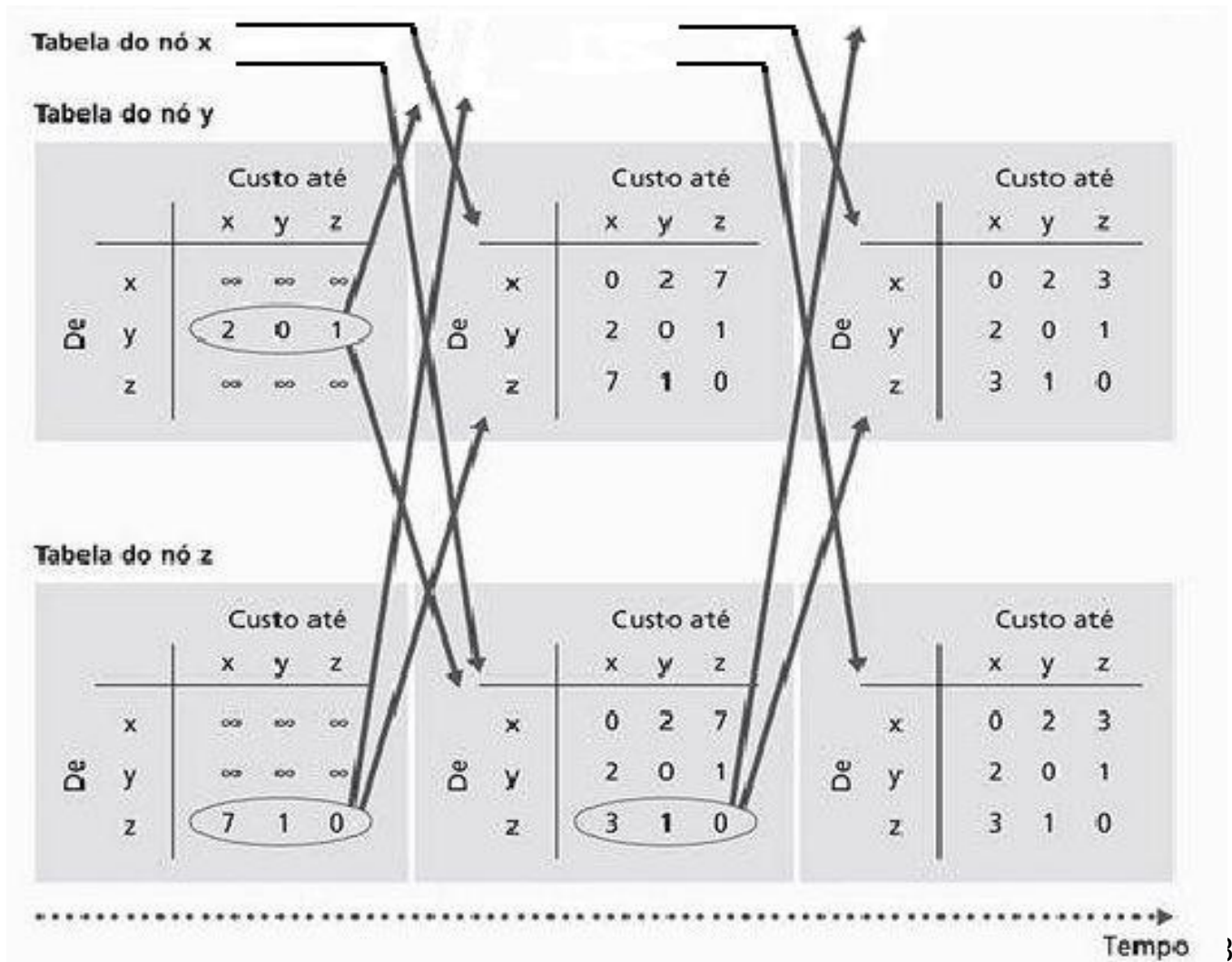
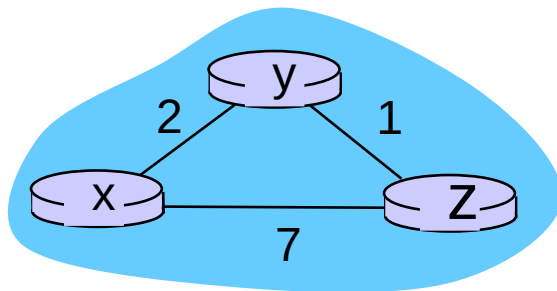




## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- e.g., considere 03 nós interconectados por 03 enlaces cujos custos são apresentados abaixo - Alg. de Vetor de Distância – “x” e “y”.



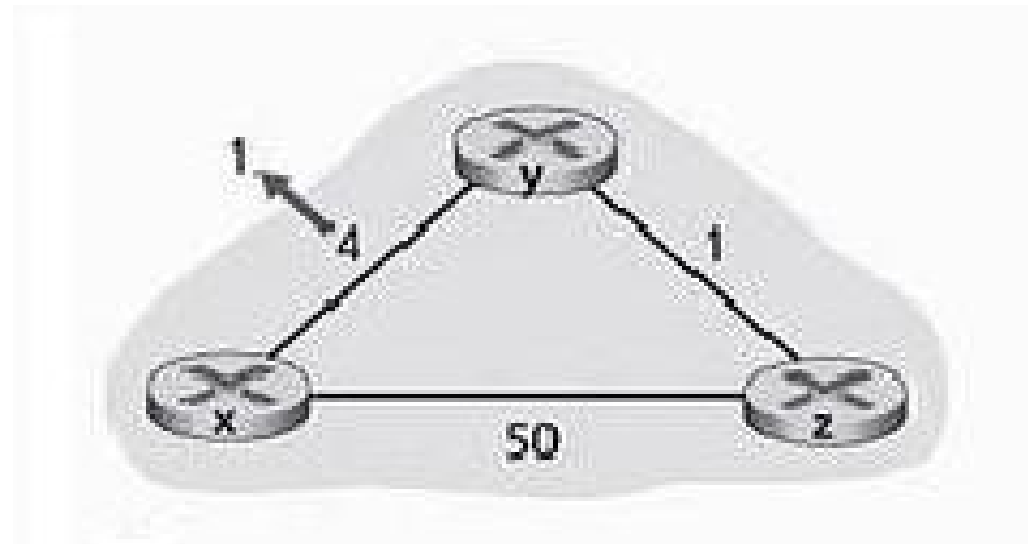
## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- Mudanças no Custo do Enlace e Falha no Enlace:
- ... quando um nó detecta uma mudança no custo de enlace até um vizinho (10-11), ele atualiza seu vetor de distâncias (13-14).
- ... se houver modificação no custo do caminho de menor custo, também informa a seus vizinhos (16-17) seu novo vetor de distâncias.

```
10  wait( até que ocorra mudança no custo do enlace ao vizinho “w”
11      ou até a recepção de um vetor de distâncias do vizinho “w” )
12
13  para cada “y” em N
14       $D_x(y) = \min_v \{ c(x, v) + D_v(y) \}$ 
15
16  if  $D_x(y)$  mudou para algum destino “y”
17      envia vetor de distâncias  $D_x = [D_x(y): y \text{ em } N]$ 
```

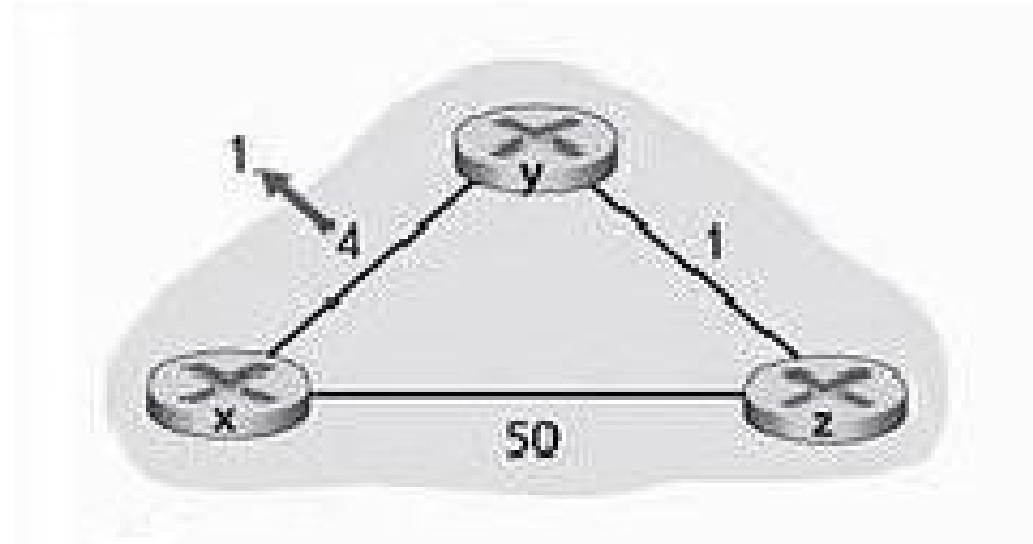
## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- e.g., considere 03 nós “x”, “y” e “z” interconectados por 03 enlaces cujos custos são:  $c(x,y) = 4 \gg 1$ ;  $c(y,z)=1$  e  $c(x,z)=50$
- “x” detecta mudança no custo em  $t_0$  e, assim, atualiza seu vetor de distâncias e informa essa mudança a seus vizinhos;
- no tempo  $t_1$ , “z” recebe a atualização de “x”, recalcula um novo custo para “x”, ou seja, “5” para “2” e envia um novo vetor aos vizinhos;



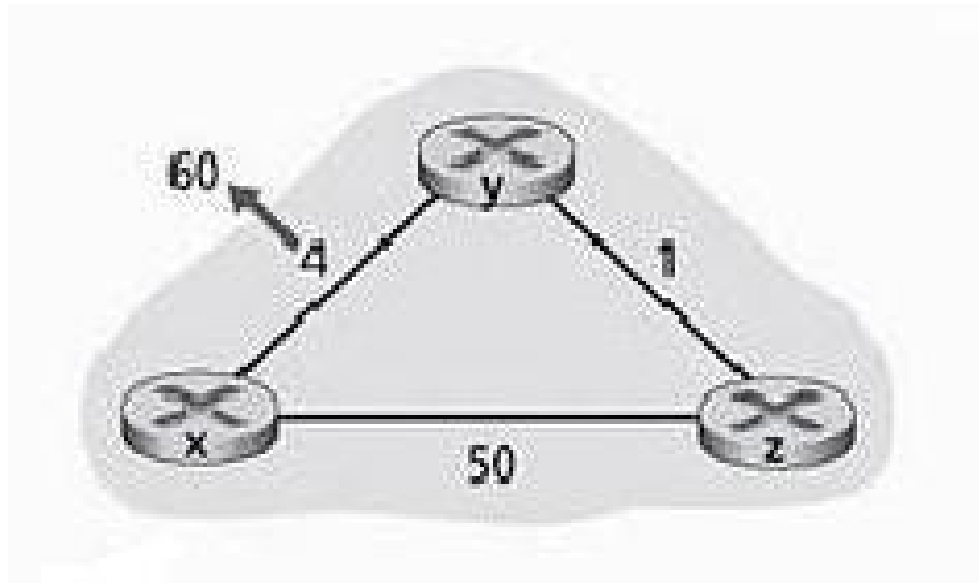
## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- e.g., considere 03 nós “x”, “y” e “z” interconectados por 03 enlaces cujos custos são:  $c(x,y) = 4 \gg 1$ ;  $c(y,z)=1$  e  $c(x,z)=50$
- no tempo  $t_2$ , “y” recebe a vetor de distâncias de “z” e ao recalcular os custos, verifica que não houve mudança ( $C(y,x)=4 \gg C(y,x) = 1$ ).
- **“conclusão”** .. “y” não propaga msg. aos vizinhos !!



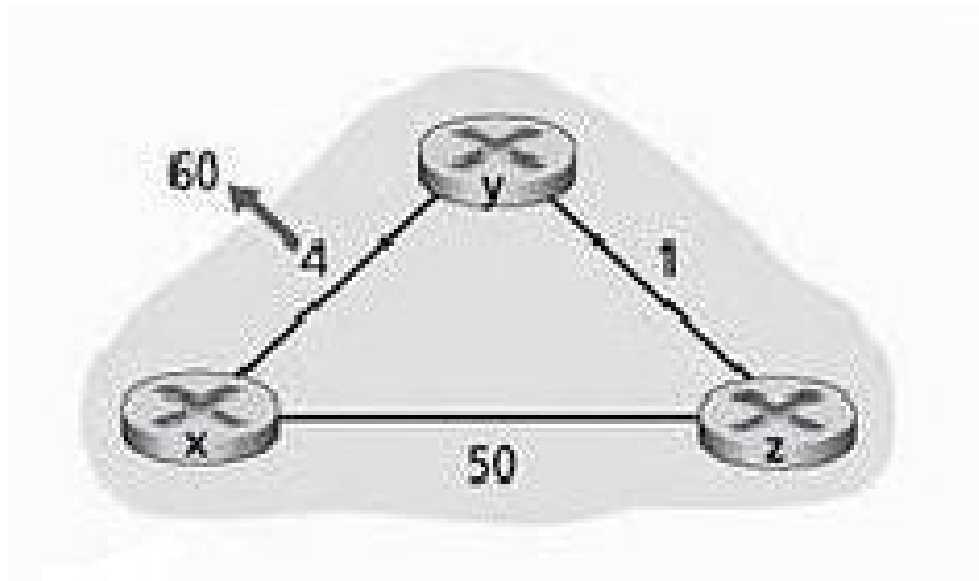
## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- e.g., considere 03 nós “x”, “y” e “z” interconectados por 03 enlaces cujos custos são:  $c(x,y) = 4 \gg 60$ ;  $c(y,z)=1$  e  $c(x,z)=50$
- $D_y(x) = \min\{ c(y,x) + D_x(x); c(y,z) + D_z(x) \} = \min\{ 60+0 ; 1+5 \} = 6$
- **“dúvida”** .. mas como  $D_z(x)=5$  se o  $c(z,x)$  é 50 ?? .. o que acontece é que “z” pode chegar a “x” por “y”, ou seja,  $1+4 = 5$ .



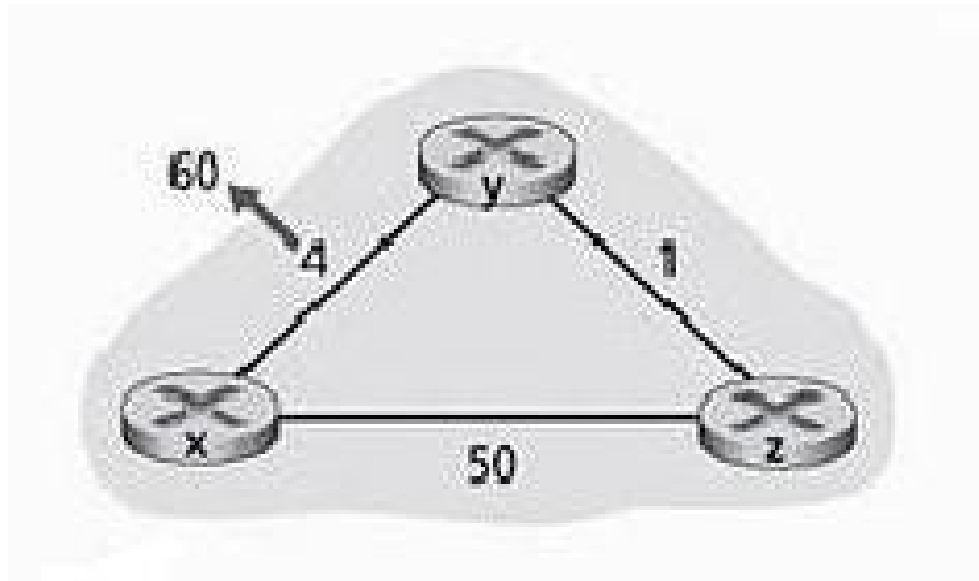
## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- e.g., considere 03 nós “x”, “y” e “z” interconectados por 03 enlaces cujos custos são:  $c(x,y) = 4 \gg 60$ ;  $c(y,z)=1$  e  $c(x,z)=50$
- ... com a visão global da rede, pode-se ver que o novo custo via “z” está errado, mas as informações que “y” tem é que seu custo direto até “x” é 60 e que “z” disse a “y” que “z” pode chegar a “x” ao custo de “5”.
- ... tem-se na sequência um “loop” na rede para chegar a “x”, pois “y” faz a rota através de “z”, que por sua vez faz a rota através de “y”.



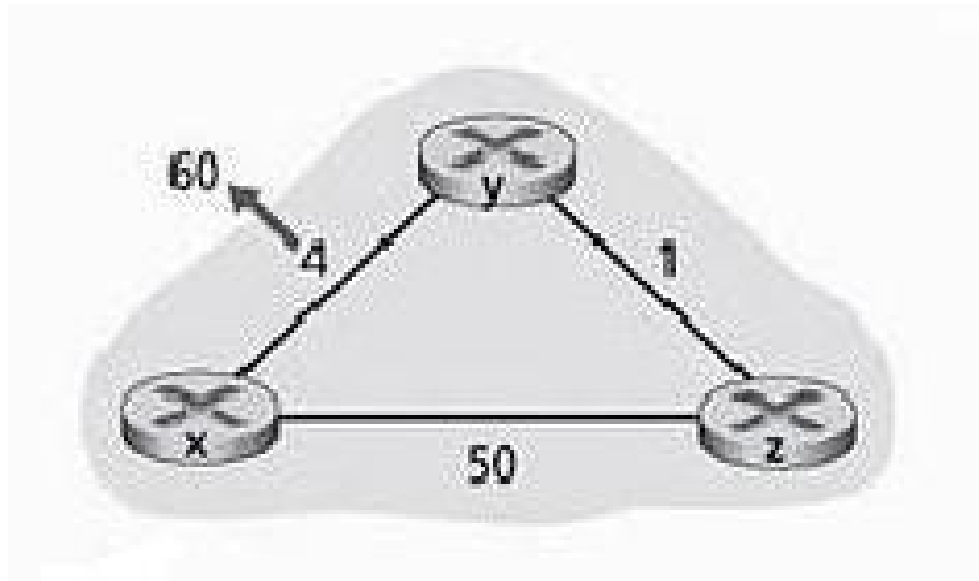
## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- e.g., considere 03 nós “x”, “y” e “z” interconectados por 03 enlaces cujos custos são:  $c(x,y) = 4 \gg 60$ ;  $c(y,z)=1$  e  $c(x,z)=50$
- “x” detecta mudança no custo em  $t_0$  ( $4 \gg 60$ ), atualiza seu vetor de distâncias e informa essa mudança a seus vizinhos “y” e “z”.
- “y” sabe que o custo direto a “x” é 60 e que “z” disse a “y” que “z” pode chegar a “x” com um custo de “5” ( custo via “z” está errado !! ).



## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

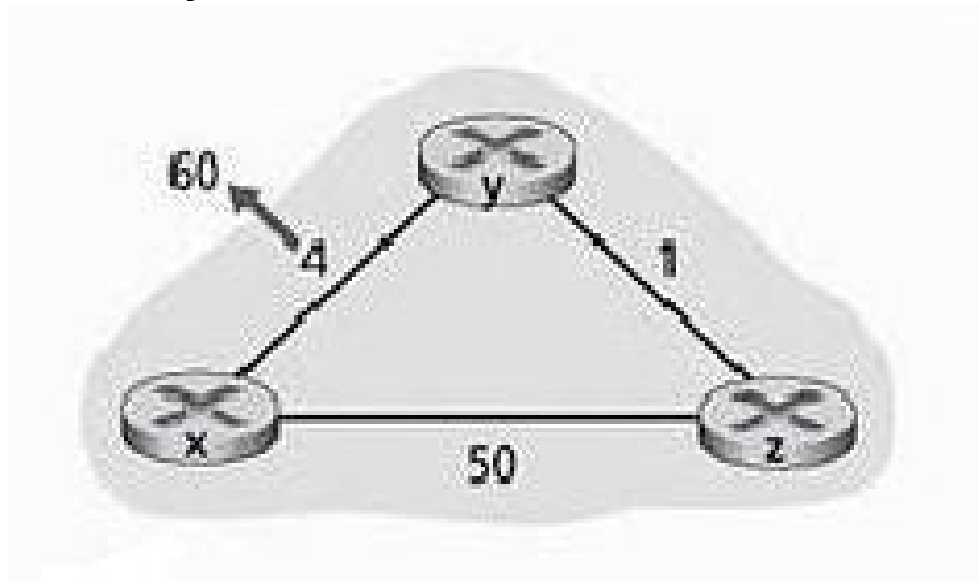
- e.g., considere 03 nós “x”, “y” e “z” interconectados por 03 enlaces cujos custos são:  $c(x,y) = 4 \gg 60$ ;  $c(y,z)=1$  e  $c(x,z)=50$
- ... em  $t_1$  teremos um “loop” de roteamento, pois para chegar a “x”, “y” faz a rota através de “z”, que por sua vez faz a rota através de “y”.
- ... tão logo “y” tenha calculado um novo custo mínimo de “x”, ele informará a “z” esse novo vetor de distâncias no tempo  $t_1$ .





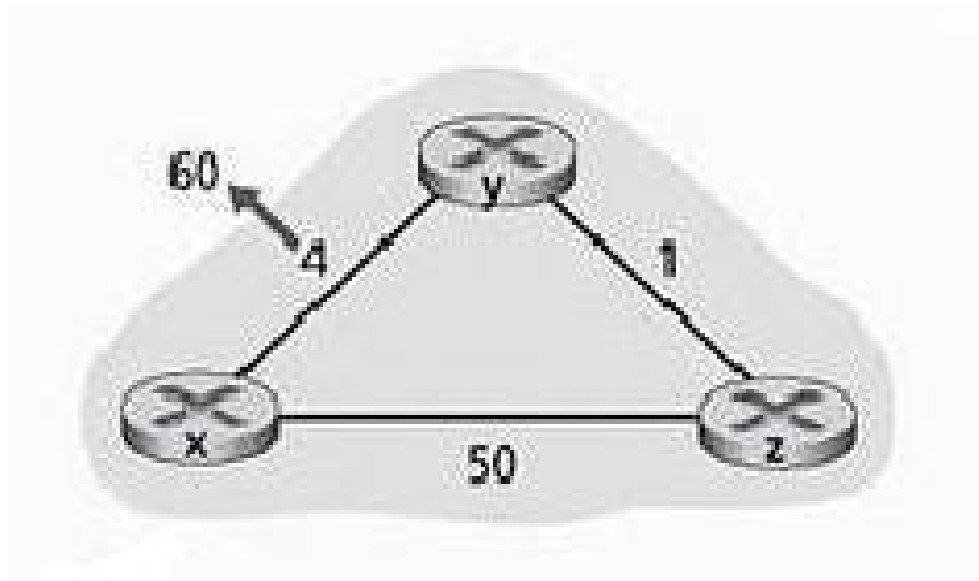
## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- e.g., considere 03 nós “x”, “y” e “z” interconectados por 03 enlaces cujos custos são:  $c(x,y) = 4 \rightarrow 60$ ;  $c(y,z)=1$  e  $c(x,z)=50$
- ... “z” recebe em  $t_2$  o vetor de distâncias de “y” que indica que o custo mínimo de “y” até “x” é 6, mas “z” sabe que pode chegar a “y” com um custo de “1” e, assim, calcula um novo custo.
- ... como  $D_z(x) = \min\{ 50 + 0 ; 1 + 6 \} = 7$ , ou seja, custo de “z” a “x” aumentou, “z” informa a “y” o seu novo vetor.



## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- “**conclusão**” - aumento do custo de enlace propaga-se muito devagar fazendo-se com que a convergência seja lenta.
- ... problema é denominado “**problema da contagem ao infinito**”.



## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- “**solução**” - cenário descrito anteriormente pode ser evitado usando-se a técnica “**poisoned reverse**”.
- ... se Z passa por Y para chegar a X, “z” informa a “y” que sua distância a “x” é infinita, assim, “y” não mais irá rotear para “x” passando por “z”.
- Obs.: ... avaliação mais abrangente mostra que a “**reversão envenenada**” não resolve o problema da contagem ao infinito para “loops” que envolvam 03 ou mais nós, pois não são detectados pela técnica.

## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- **“complexidade da mensagem”**
- LS - com “N” nós e “E” enlaces  $\gg O(N * E)$  mensagens enviadas.
- DV - troca msgs. apenas entre vizinhos  $\gg$  tempo convergência varia.
- **“velocidade de convergência”**
- LS - algoritmo  $O(N^2)$  requer  $O(N * E)$  mensagens (pode ter oscilações).
- DV - tempo de convergência varia; pode ter “loops” de roteamento bem como o problema da contagem ao infinito.

## ... 4.5.2 – Algoritmo de Roteamento Distance Vector

- “**robustez**” - o que acontece se roteador der defeito?
- LS .. nó pode anunciar custo incorreto do enlace; cada nó calcula apenas sua própria tabela de repasse;
- DV .. nó pode anunciar custo incorreto do enlace e como a tabela de cada nó é utilizada por outros nós >> erro se propaga pela rede.

## 4.5.3 – Roteamento Hierárquico

- **“premissa para roteamento ideal”** .. todos os roteadores são idênticos bem como o algoritmo de roteamento.
- **“possíveis problemas”**
- **“escalabilidade”** - quanto maior o nro. de roteadores, maior é a sobrecarga de cálculo, armazenamento bem como a comunicação de informações para a tabela de roteamento.
- **“autonomia administrativa”** - autonomia na gestão de roteadores pelas empresas sob as quais são mantidos.
- **“autonomia administrativa”** - transparência ao público quanto a organização interna da rede são fatores que devem ser considerados.

## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

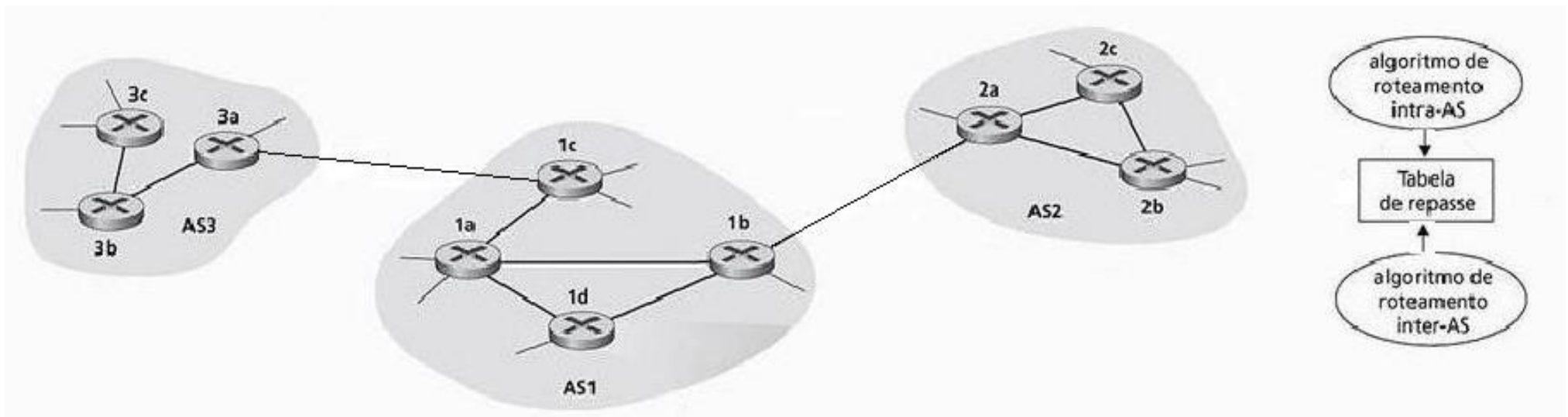
### ... 4.5.3 – Roteamento Hierárquico

- **“possíveis problemas”**
- **“escalabilidade”** - quanto maior o nro. de roteadores, maior é a sobrecarga de cálculo, armazenamento bem como a comunicação de informações para a tabela de roteamento.
- **“autonomia administrativa”** - autonomia na gestão de roteadores pelas empresas sob as quais são mantidos.
- **“autonomia administrativa”** - transparência ao público quanto a organização interna da redes são fatores que devem ser considerados.
- **“solução”** - agrupar os roteadores em ASs, ou seja, criar grupos de roteadores sob o mesmo domínio administrativo (p.ex., mesmo ISP ou mesma Rede Corporativa).

## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.3 – Roteamento Hierárquico

- “**solução**” - agrupar os roteadores em ASs, ou seja, criar grupos de roteadores sob o mesmo domínio administrativo (p.ex., mesmo ISP ou mesma Rede Corporativa)
- “**Protocolo Intra-AS**” .. denominação do algoritmo de roteamento dentro de um Sistema Autônomo (AS) da Internet.
- ... para conectar ASs entre si, um ou mais “roteadores de borda” em um AS tem a tarefa de transmitir pacotes a destinos externos ao AS.



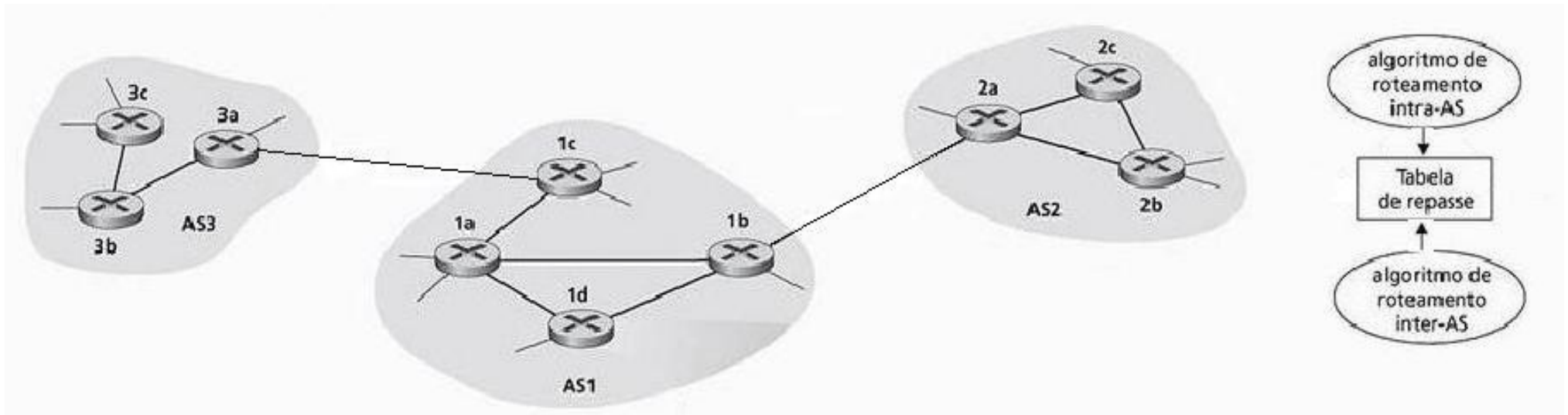


### ... 4.5.3 – Roteamento Hierárquico

- “**pergunta**” .. como um roteador que está dentro de algum AS sabe como rotear um pacote até um destino que está fora do AS ?
- “**roteador de borda único**” .. próprio algoritmo de roteamento interno (Intra-AS) já determinou o caminho de menor custo entre cada roteador interno e o roteador de borda do AS.
- ... uma vez que o pacote é roteado para fora do AS através do roteador de borda, o AS que está na outra extremidade do enlace assume a responsabilidade de rotear o pacote até o destino final.

## ... 4.5.3 – Roteamento Hierárquico

- e.g., considere o roteamento de um pacote pelo roteador “2b” para um destino fora do Sistema Autônomo “AS2”.
- ... “2b” transmite o pacote para “2a”, que por sua vez retransmite a “1b”, portanto, trabalho de “AS2” para este pacote está finalizado.
- ... uma vez que o pacote é roteado para fora do AS pelo roteador de borda, o AS que está na outra extremidade do enlace assume a responsabilidade de rotear o pacote até o destino final.



### ... 4.5.3 – Roteamento Hierárquico

- “**pergunta**” .. como um roteador que está dentro de algum AS sabe como rotear um pacote até um destino que está fora do AS ?
- “**dois ou mais roteadores de borda**” .. nestes casos, o repasse é significativamente mais desafiador !!
- ... embora a tarefa seja decidir para qual enlace de saída o pacote será repassado, tal decisão exige que cada AS saiba ou não quais destinos são alcançáveis a partir de cada outro AS vizinho.

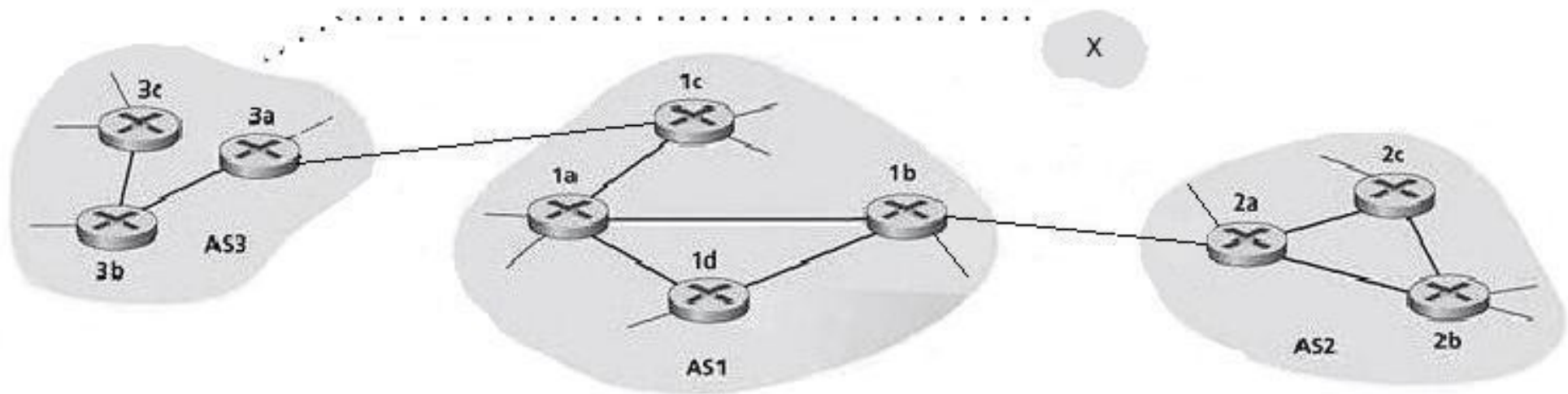
### ... 4.5.3 – Roteamento Hierárquico

- “**dois ou mais roteadores de borda**” .. nestes casos, o repasse é significativamente mais desafiador !!
- ... informações sobre condições de alcance de AS vizinhos bem como a propagação dessas informações para todos os roteadores internos são gerenciadas pelo Protocolo de Roteamento Inter-ASs.
- Obs.: ASs da Internet rodam o mesmo Protocolo de Roteamento Inter-ASs, ou seja, o BGP4 permitindo que os roteadores recebam informações de protocolos de roteamento Intra-ASs e Inter-ASs.

## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.3 – Roteamento Hierárquico

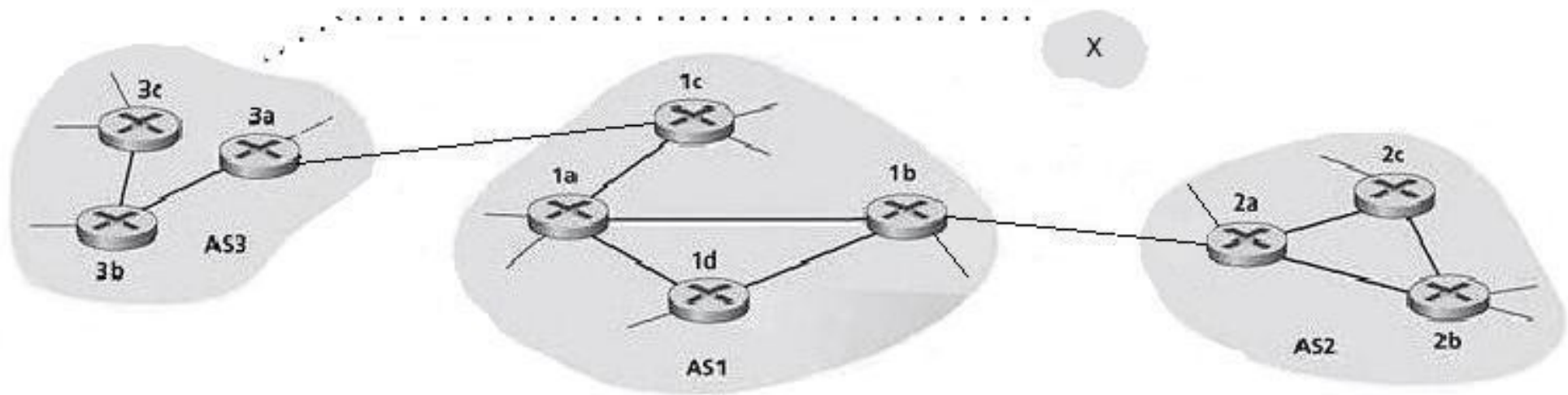
- e.g., considere uma sub-rede “x” e suponha que “AS1” saiba que “x” é alcançável através de “AS3”, mas não alcançável através de AS2.
- “AS1” propaga esta informação para todos os seus roteadores, permitindo que através do PR Intra-AS os caminhos de menor custo sejam calculados para cada um dos roteadores, destinos e saídas do AS.



## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.3 – Roteamento Hierárquico

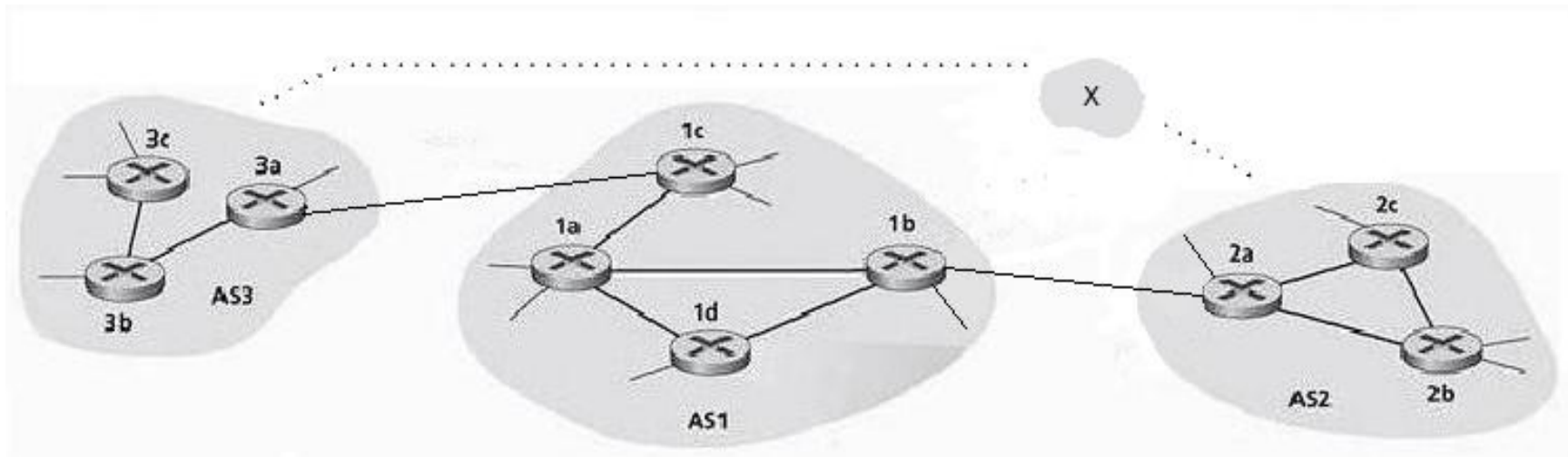
- “AS1” propaga esta informação para todos os seus roteadores, permitindo que através do PR Intra-AS os caminhos de menor custo sejam calculados para cada um dos roteadores, destinos e saídas do AS.
- ... seja “l” a interface do roteador “1d” que é o caminho de menor custo entre “1d” e “1c”, assim “1d” insere o registro “x,l” na sua tabela.



## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.3 – Roteamento Hierárquico

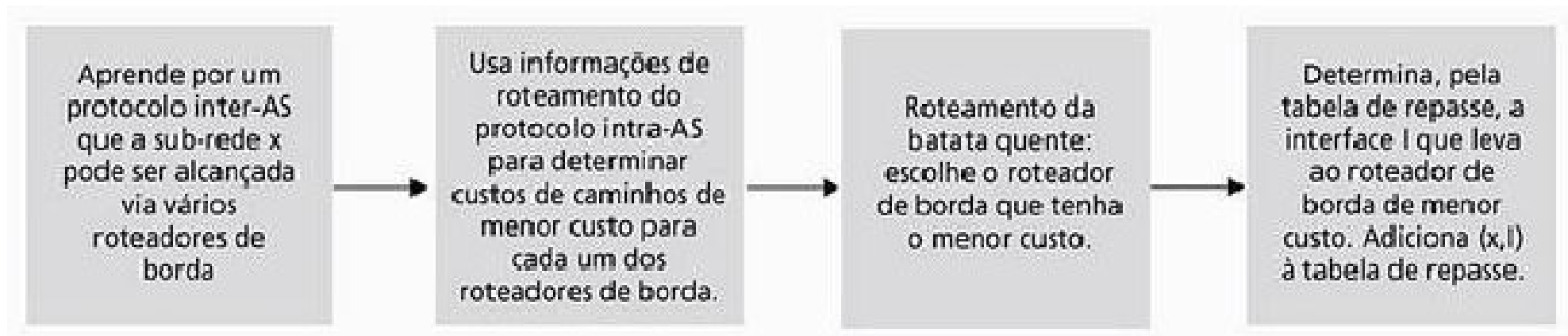
- e.g., considere uma sub-rede “x” e suponha que “AS1” saiba que a sub-rede “x” possa ser alcançada através de “AS3”, mas também pode ser alcançada através de AS2.



- “roteamento da batata quente”** .. escolhe o roteador de borda que tenha o menor custo para encaminhar datagramas à sub-rede “x”.

## ... 4.5.3 – Roteamento Hierárquico

- e.g., considere uma sub-rede “x” e suponha que “AS1” saiba que a sub-rede “x” possa ser alcançada através de “AS3”, mas também pode ser alcançada através de AS2.
- **“roteamento da batata quente”** .. escolhe o roteador de borda que tenha o menor custo para encaminhar datagramas à sub-rede “x”.

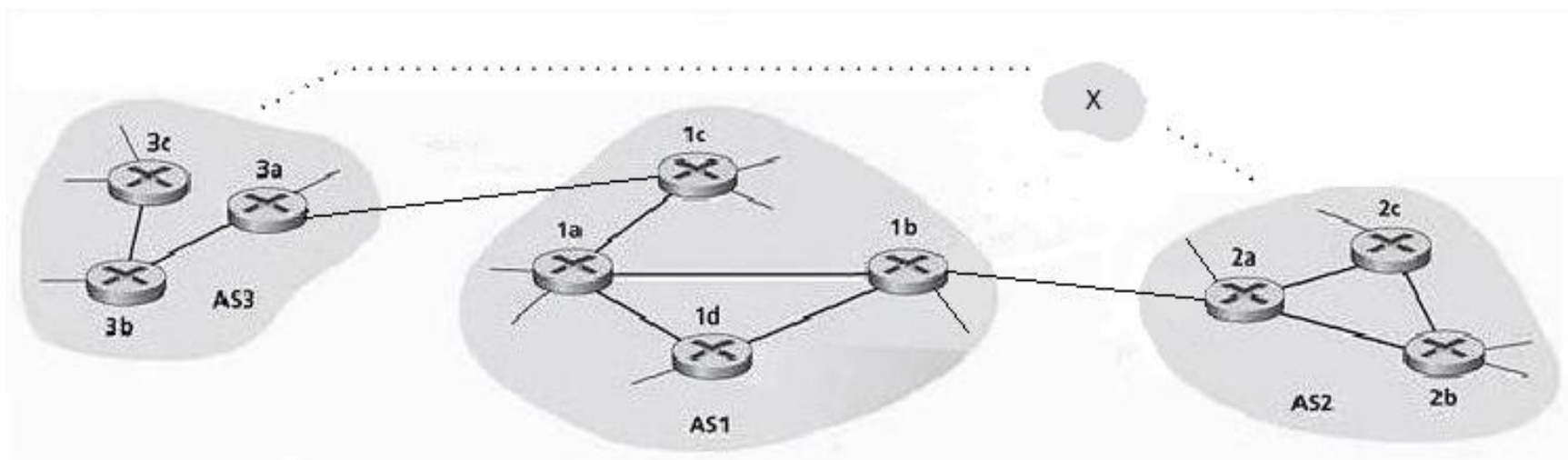




## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.3 – Roteamento Hierárquico

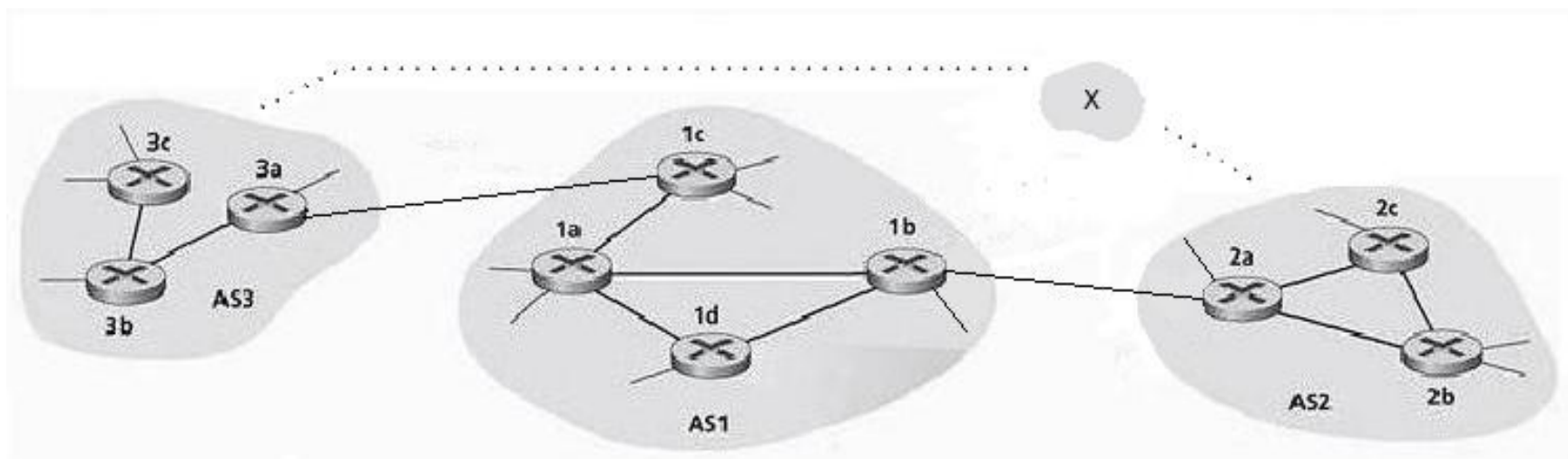
- e.g., considere que o “AS2” e “AS3” estejam conectados com outros ASs que não aparecem no diagrama.
- “AS1” toma conhecimento através do PR Inter-AS que a sub-rede “x” pode ser alcançada a partir de “AS2” via roteador de borda “1b”.
- “AS1” toma conhecimento através do PR Inter-AS que a sub-rede “x” pode ser alcançada a partir de “AS3” via roteador de borda “1c”.
- **“problema”** .. o que fazer ???!



## 4 - Camada de Rede / 4.5 - Algoritmos de Roteamento

### ... 4.5.3 – Roteamento Hierárquico

- e.g., considere que o “AS2” e “AS3” estejam conectados com outros ASs que não aparecem no diagrama.
- Qual rota inserir na Tabela de Repasse de “1d” para alcançar a sub-rede “x”, ou seja, há 02 caminhos, então qual escolher ??

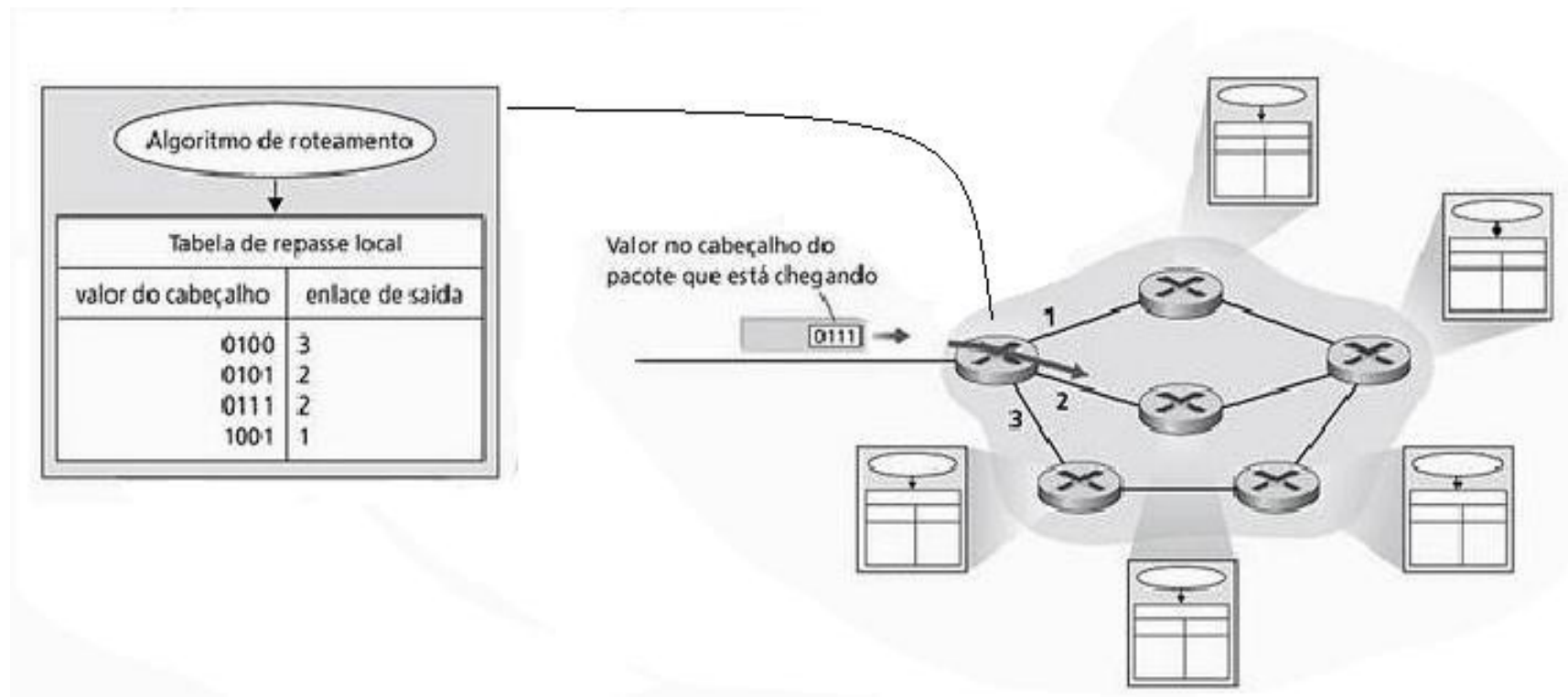


### ... 4.5.3 – Roteamento Hierárquico

- “**solução**” - utilizar o roteamento da “**batata quente**”, ou seja, AS repassa o pacote através do caminho de menor custo o mais rápido possível com o menor custo possível.
- ... roteador envia o pacote ao roteador de borda que tiver o menor custo roteador-roteador de borda entre todos os roteadores de borda que tem um caminho para o destino ora estabelecido.
- Obs.: ASs tem bastante flexibilidade para decidir quais destinos irá anunciar a seus ASs vizinhos, ou seja, decisão “política” que depende mais de questões econômicas do que técnicas.

## 4.6 – Roteamento na Arq. TCP/IP

- Roteamento na Internet .. determina o caminho percorrido por um datagrama entre o remetente e destinatário.
- .. roteamento na Arq. TCP/IP incorpora muitos dos princípios discutidos nos Algoritmos de Estado de Enlace (LS) e Vetor de Distâncias (DV).



## ... 4.6 – Roteamento na Arq. TCP/IP

- “**Autonomous Systems (ASs)**” .. conjunto de roteadores sob o mesmo controle e/ou domínio administrativo e técnico e que executam o mesmo protocolo de roteamento.
- Roteamento Intra-AS ou “Interior Gateway Protocol” .. usado para determinar o encaminhamento de pacotes dentro de um AS.
- RIP - Routing Information Protocol.
- OSPF - Open Shortest Path First.
- IGRP - Interior Gateway Routing Protocol (Cisco).

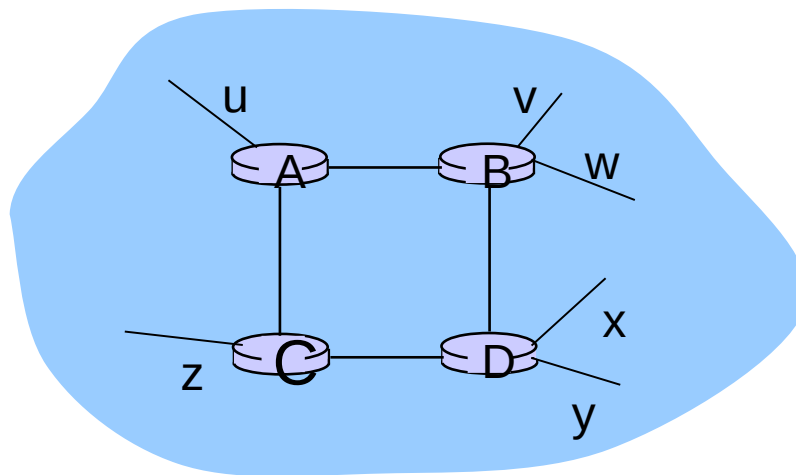
## 4.6.1 – Roteamento RIP

- Roteamento Intra-AS .. determina a execução do algoritmo de roteamento de um Sistema Autônomo (AS).
- RIP ... foi incluído na distribuição BSD-UNIX em 1982, mas a origem vem da Arquitetura XNS (Xerox Network Systems).
- ... normalmente disponibilizado entre ISPs (Internet Service Providers) de nível mais baixo ou em redes corporativas.

## 4 - Camada de Rede / 4.6 - Roteamento na Arq. TCP/IP

### ... 4.6.1 – Roteamento RIP

- RIP [RFC 1058] .. utiliza a contagem de saltos como métrica de custo e onde cada enlace tem custo unitário.
- “**custos de enlaces**” .. são definidos desde o roteador de origem ao roteador de destino através do número de saltos.
- “**salto**” .. número de sub-redes percorridas ao longo do caminho mais curto, ou seja, menor número de saltos entre o roteador de origem e o roteador de destino, incluindo o destino.

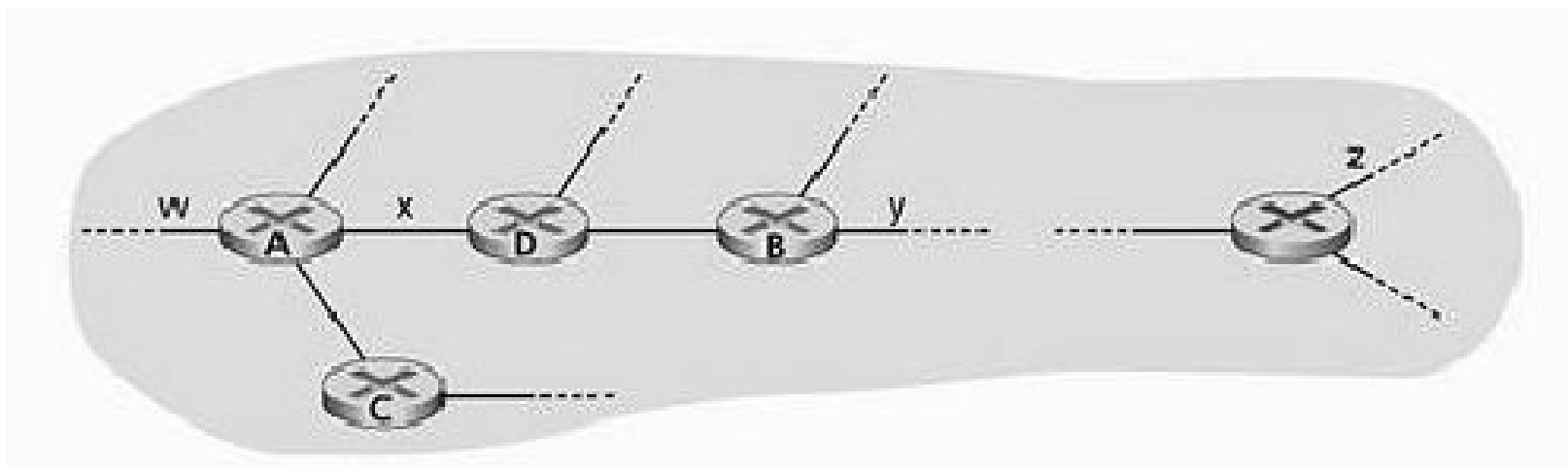


destino	saltos
u	1
v	2
w	2
x	3
y	3
z	2

## 4 - Camada de Rede / 4.6 - Roteamento na Arq. TCP/IP

### ... 4.6.1 – Roteamento RIP

- “**anúncios**” .. mensagens do RIP (Routing Information Protocol) ..
- “**vetores de distância**” .. trocados entre vizinhos a cada 30 seg. por meio de msgs. de resposta (também conhecida como “anúncios”).
- “**anúncio**” - contém até 25 sub-redes de destino dentro do AS, bem como as distâncias entre o remetente e cada uma dessas sub-redes.





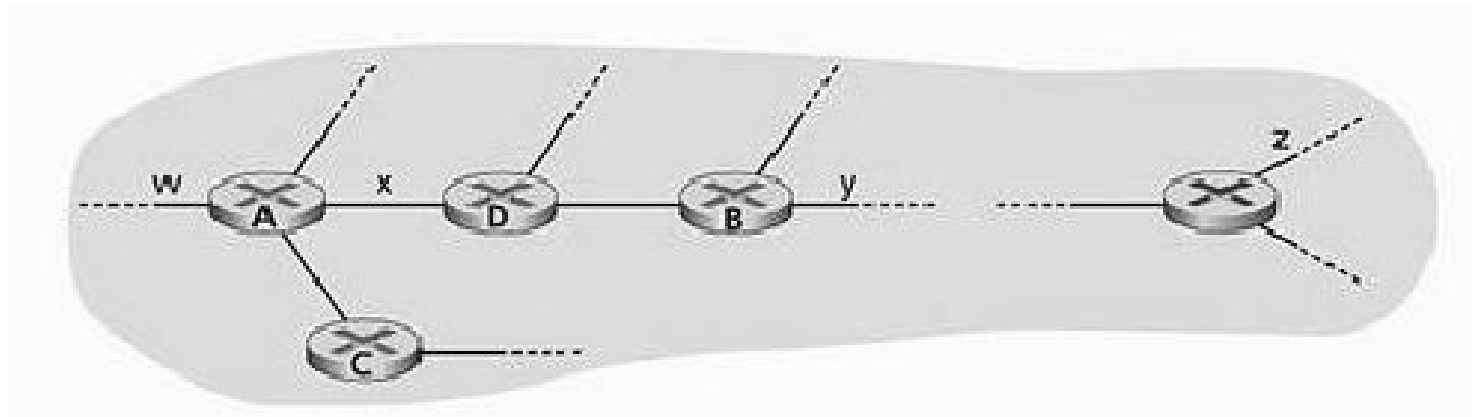
## ... 4.6.1 – Roteamento RIP

- RIP [RFC 1058] .. utiliza a contagem de saltos como métrica de custo e onde cada enlace tem custo unitário.
- “**restrição**” .. custo máximo de um caminho é de até 15 saltos, ou seja, diâmetro do sistema autônomo não pode ser superior a 15 enlaces.
- “**tabela de roteamento**” .. tabela de roteamento inclui o vetor de distâncias e a tabela de repasse do próprio roteador.
- ... tabela de roteamento é mantida por cada roteador dentro do AS.

## 4 - Camada de Rede / 4.6 - Roteamento na Arq. TCP/IP

### ... 4.6.1 – Roteamento RIP

- e.g., considere as sub-redes “w”, “x”, “y” e “z” como mostrado na figura bem como a tabela de roteamento de “D” antes de receber uma msg. ou “anúncio” do roteador “A”.



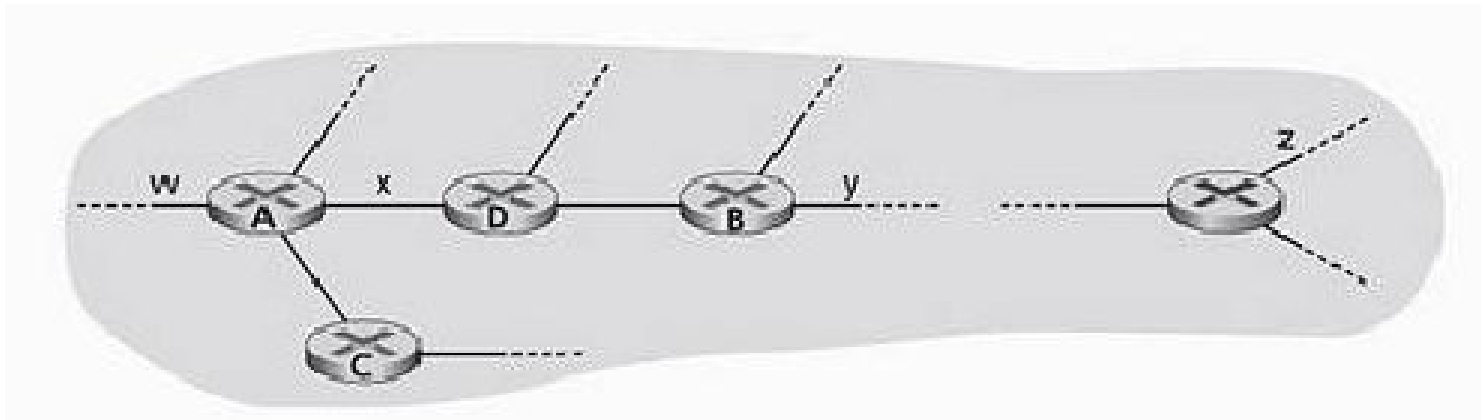
Sub-rede de destino	Roteador seguinte	Número de saltos até o destino
w	A	2
y	B	2
z	B	7
x	—	1
...	...	...

Tabela de Roteamento  
do Roteador "D"  
antes do Anúncio de "A"

## 4 - Camada de Rede / 4.6 - Roteamento na Arq. TCP/IP

### ... 4.6.1 – Roteamento RIP

- ... roteador “A” informa que o custo para atingir “z” é 4, enquanto “D” tem custo “7” para atingir “z”, logo poderá se beneficiar do “anúncio” quanto “A” encaminhá-lo a “D”.



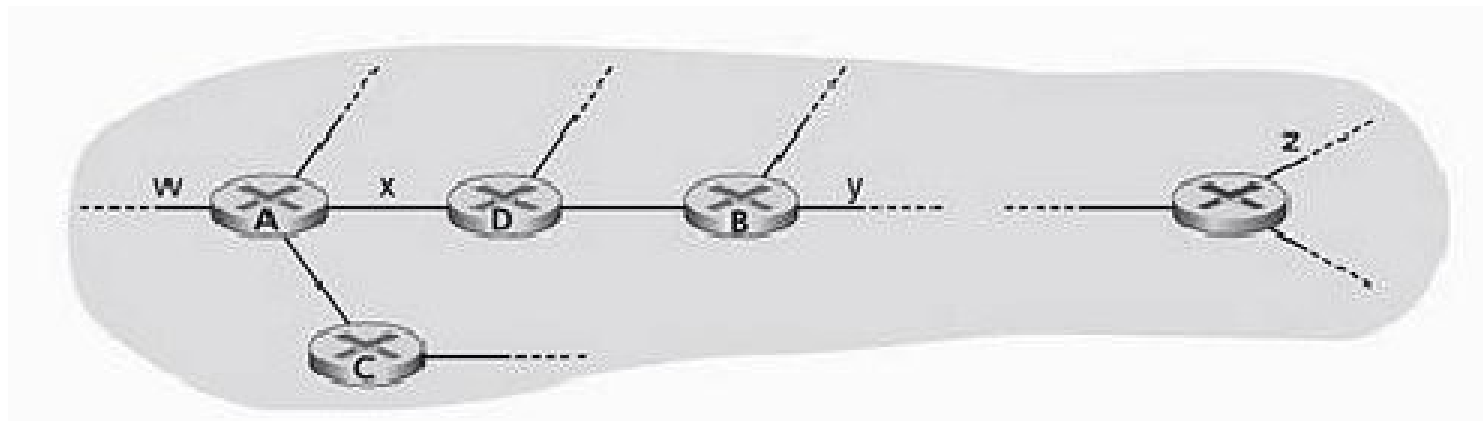
Sub-rede de destino	Roteador seguinte	Número de saltos até o destino
z	C	4
w	—	1
x	—	1
...	...	...

Anúncio do Roteador "A"

## 4 - Camada de Rede / 4.6 - Roteamento na Arq. TCP/IP

### ... 4.6.1 – Roteamento RIP

- ... quando o roteador “A” anunciar sua tabela de rotas, o roteador “D” irá atualizar sua tabela de rotas para considerar os caminhos de menor custo, p.ex., sub-rede “z” ao custo “4”.

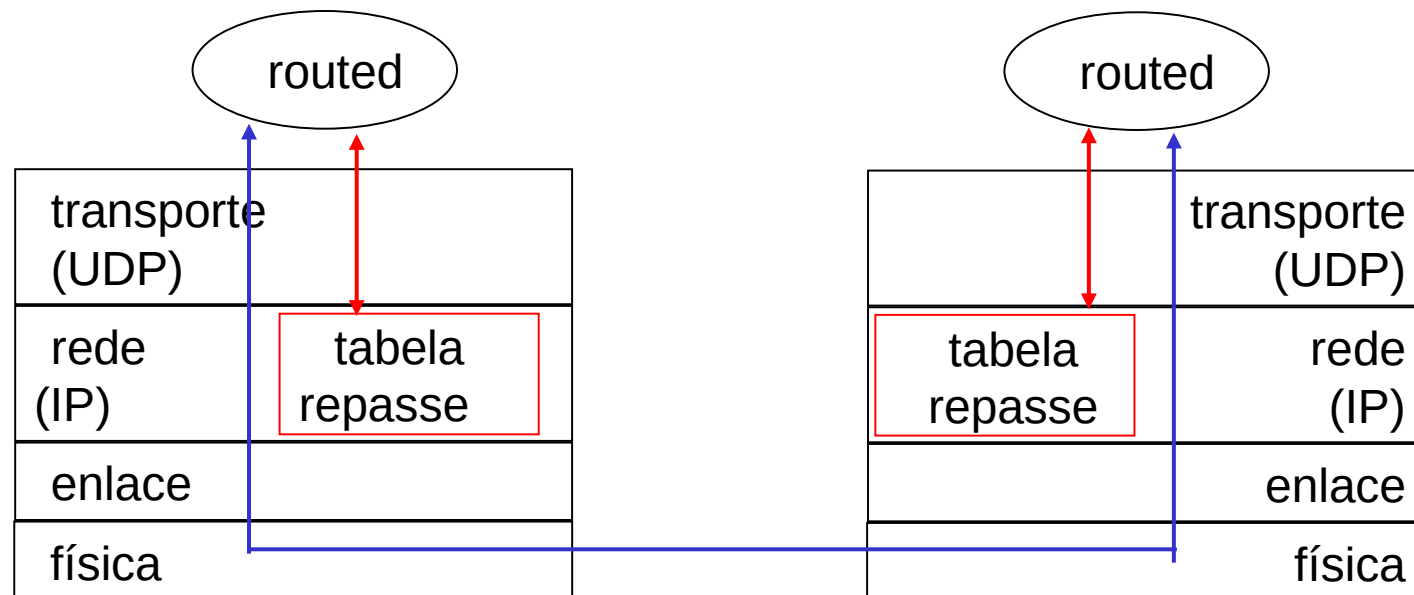


Sub-rede de destino	Roteador seguinte	Número de saltos até o destino
w	A	2
y	B	2
z	A	5
...	...	...

Tabela de Roteamento  
Roteador "D" após  
receber Anúncio de "A"

## ... 4.6.1 – Roteamento RIP

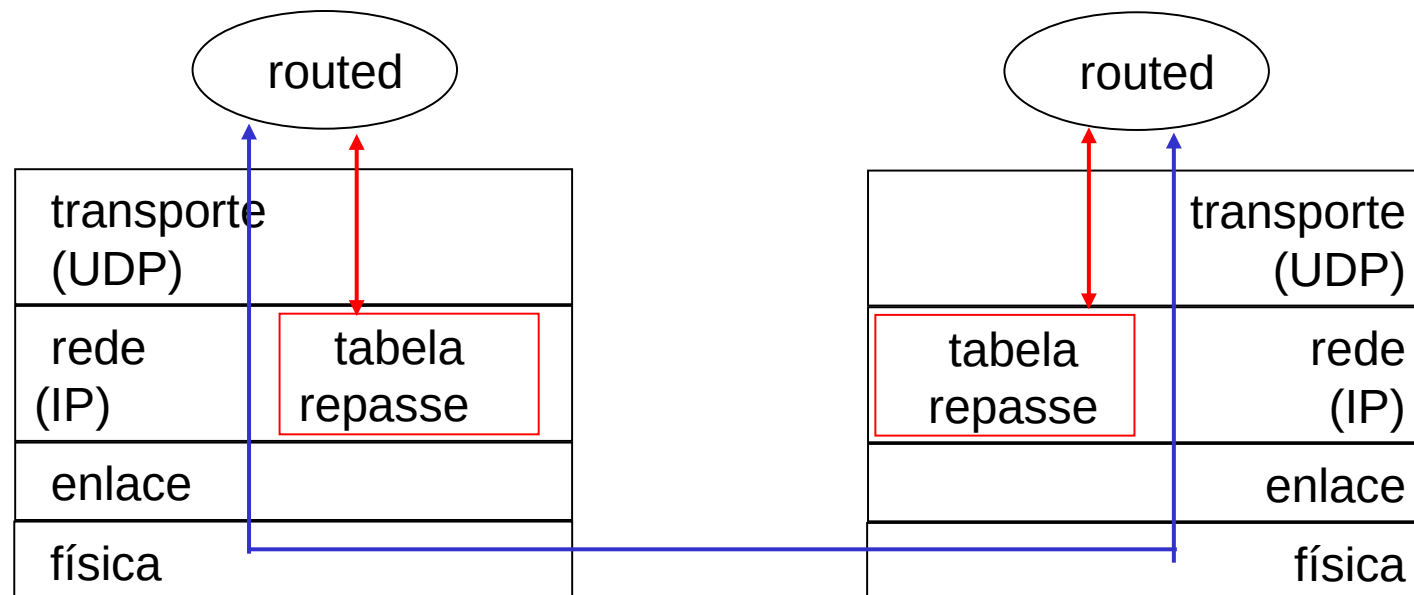
- “**falha de recuperação**” .. se um roteador não receber ao menos um anúncio de seus vizinhos a intervalos de 180 segundos » vizinho ou enlace é declarado morto.
- ... roteador modifica sua tabela de rotas e, em seguida, propaga essa informação enviando “anúncios” a seus roteadores vizinhos.
- ... vizinhos por sua vez, recalculam suas tabelas e enviam novos anúncios aos seus vizinhos caso haja alterações em suas tabelas.



## 4 - Camada de Rede / 4.6 - Roteamento na Arq. TCP/IP

### ... 4.6.1 – Roteamento RIP

- RIP é implementado como processo da Camada de Aplicação.
- ... msgs. de requisição e resposta são trocadas pelos roteadores na porta “520” através de segmentos UDP carregados dentro do IP.
- ... reversão envenenada usada para impedir “loops” de pingue-pongue (distância infinita = 16 saltos).



## 4.6.2 – Roteamento OSPF

- **OSPF [RFC 2178]** .. publicamente disponível em ISPs de níveis mais alto, ao passo que o RIP em ISPs de nível mais baixo.
- RFC 2178 .. especifica a versão mais recente do OSPF (Open Shortest Path First) e utiliza o Algoritmo Link State (LS).
- ... estado de enlace necessita do “broadcasting” de informações de estado de enlace e um algoritmo do caminho de menor custo.
- ... constrói um mapa topológico completo ou grafo de todo o sistema autônomo e na sequência executa o Alg. de Dijkstra para determinar a árvore de caminho mais curto para todas as sub-redes.

## ... 4.6.2 – Roteamento OSPF

- OSPF .. provê mecanismos para determinar o caminho de menor custo para um dado conjunto de pesos de enlaces (Alg. de Dijkstra).
- ... não impõe política para o modo como são determinados os pesos dos enlaces, em vez disso, provê mecanismos para determinar os caminhos de menor custo para um dado conjunto de pesos.
- ... roteador transmite informações de roteamento a todos os outros roteadores no AS e não apenas a seus roteadores vizinhos.
- Anúncios OSPF são carregados diretamente no IP com código de protocolo da camada superior igual a “89”.
- ... isto exige que o próprio Protocolo OSPF implemente funcionalidades de transferência confiável de msgs. e transmissão “broadcast”.



## ... 4.6.2 – Roteamento OSPF

- **“recursos avançados”** do OSPF:
- **“segurança”** .. msgs. autenticadas para impedir intrusão maliciosa, ou seja, apenas roteadores de confiança podem participar.
- **“múltiplos caminhos”** .. múltiplos caminhos de menor custo são permitidos, ou seja, não é preciso escolher ou calcular um caminho único para carregar todo o tráfego.
- **“múltiplas métricas de custo”** .. p.ex., custo “baixo” para enlace de satélite quando TOS for “melhor esforço” ou custo “alto” para enlace de satélite quando TOS for para “tempo real”.
- **“suporte integrado para unicast e multicast”** - Multicast OSPF (MOSPF) usa mesma base de dados de topologia que o OSPF.

## ... 4.6.2 – Roteamento OSPF

- “**recursos avançados**” do OSPF:
- “**roteamento hierárquico**” .. capacidade de estruturar hierarquicamente em áreas um sistema autônomo, onde cada área dispõe do seu próprio algoritmo de roteamento.
- ... dentro da área um ou mais roteadores de borda de áreas são responsáveis pelo roteamento de pacotes para fora da área.
- ... uma área é configurada para a área de “backbone”, cujo papel é rotear tráfego entre as outras áreas do sistema autônomo.
- “**backbone**” .. contém todos os roteadores de borda de área que estão dentro do sistema autônomo bem como pode conter outros roteadores.

## ... 4.6.2 – Roteamento OSPF

- **“organização hierarquia”** .. em 02 níveis, área local e backbone.
- ... anúncios de estado do enlace somente na área.
- ... cada nó tem topologia de área detalhada e somente direção conhecida (caminho mais curto) para redes em outras áreas.
  
- **“roteadores de borda”** .. “resumem” distâncias às redes na própria área, anunciam para outros roteadores de borda.
- **“roteadores de backbone”** .. roteamento limitado ao “backbone”, cujo papel principal é rotear tráfego entre as outras áreas do AS.

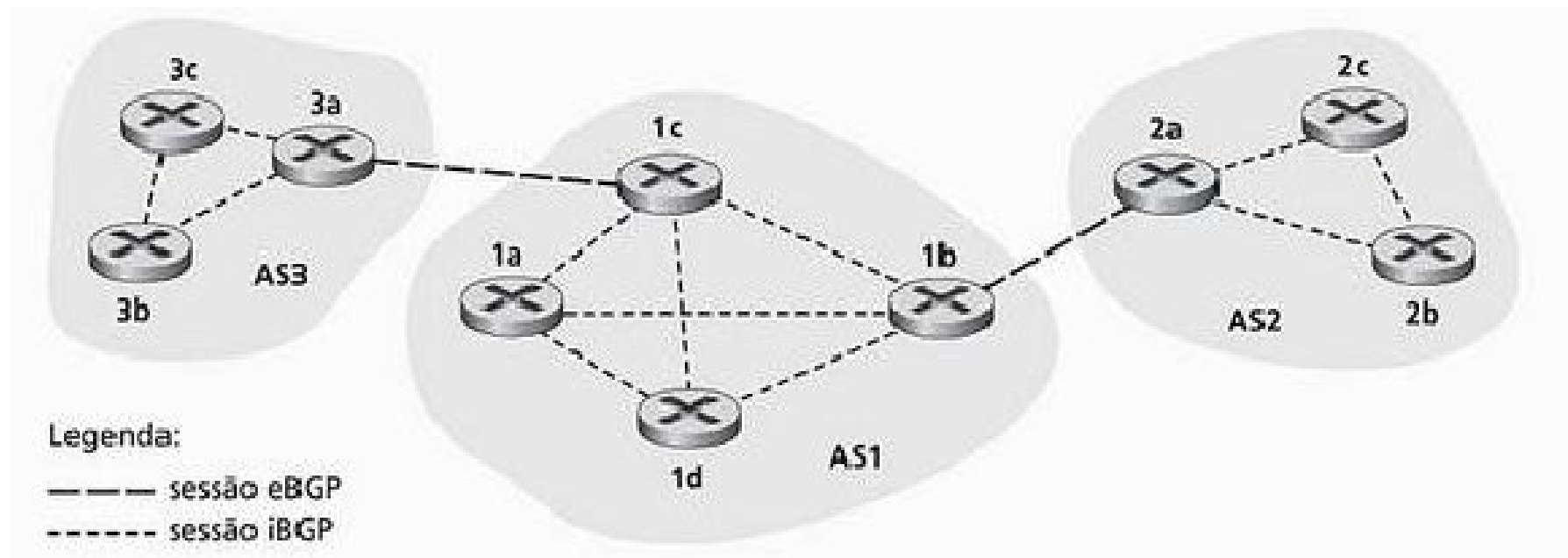
### 4.6.3 – Roteamento BGP

- **“Border Gateway Protocol”** [RFC 4271] - especifica como são determinados os caminhos entre fonte e destino que abrangem vários ASs.
- ... obter dos ASs vizinhos informações de atingibilidade de sub-redes que compõem os próprios sistemas autônomos.
- ... propagar a informação de atingibilidade a todos os roteadores internos ao sistema autônomo.
- ... determinar rotas “boas” para sub-redes com base na informação de atingibilidade e na política do sistema autônomo.
- ... garantir que todos os sistemas autônomos da Arq. TCP/IP saibam da existência dessa sub-rede e como chegar até a mesma.

## 4 - Camada de Rede / 4.6 - Roteamento na Arq. TCP/IP

### ... 4.6.3 – Roteamento BGP

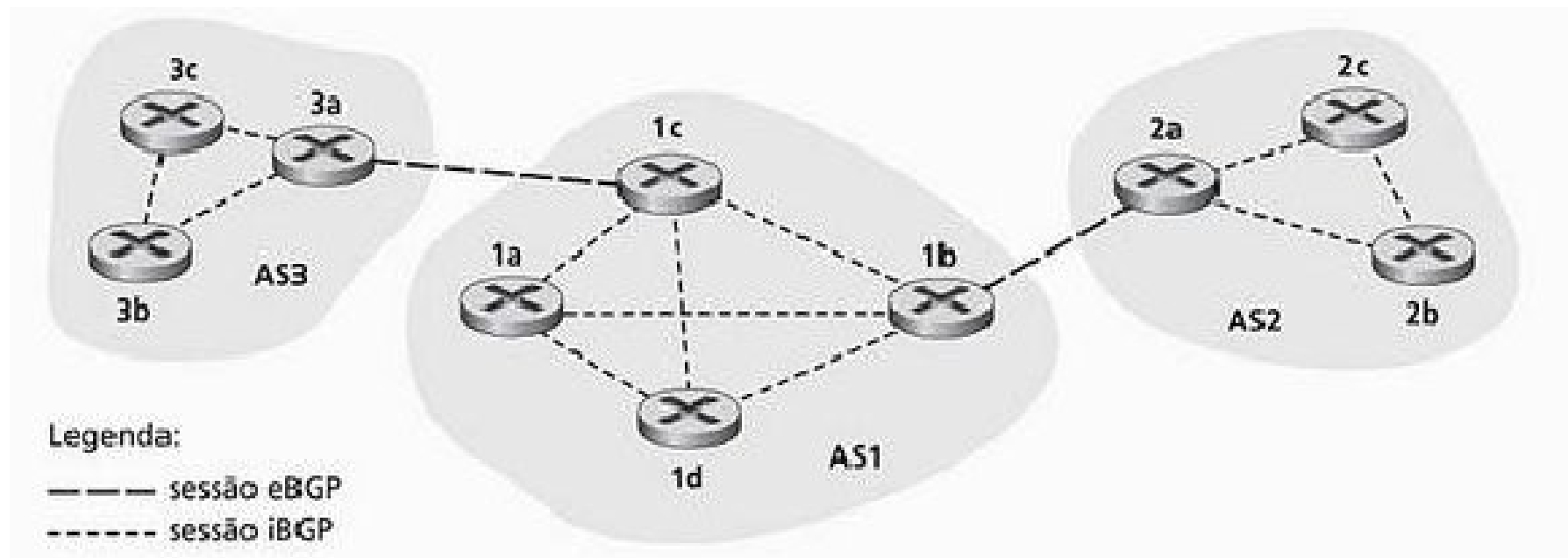
- “**mensagens no BGP**” .. msgs. são trocadas através de conexões TCP semipermanentes usando a porta “179”.
- ... normalmente tem-se uma conexão semipermanente entre roteadores internos ao sistema autônomo, bem como, entre os roteadores de bordas de sistemas autônomos diferentes.



## 4 - Camada de Rede / 4.6 - Roteamento na Arq. TCP/IP

### ... 4.6.3 – Roteamento BGP

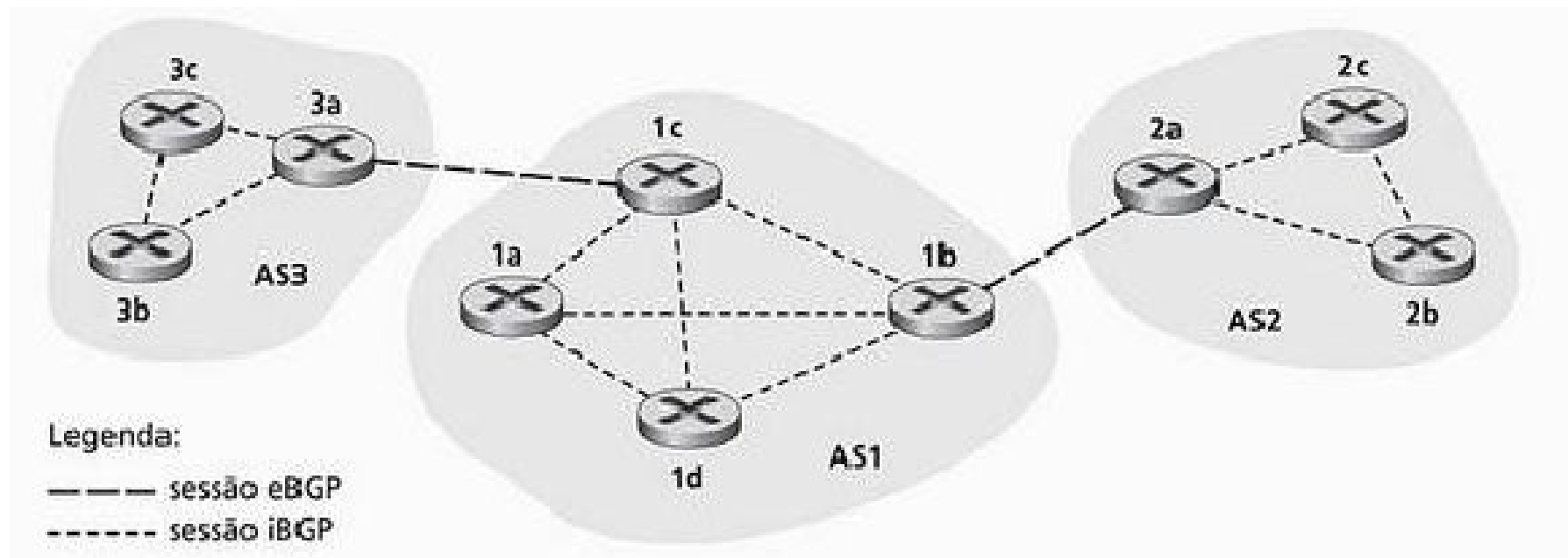
- **“Roteadores Pares BGP”** .. são os roteadores nas extremidades de cada conexão TCP, ou seja, 02 roteadores interconectados.
- **“Sessão BGP”** .. conexão TCP juntamente com as mensagens enviadas através da conexão entre roteadores pares.



## 4 - Camada de Rede / 4.6 - Roteamento na Arq. TCP/IP

### ... 4.6.3 – Roteamento BGP

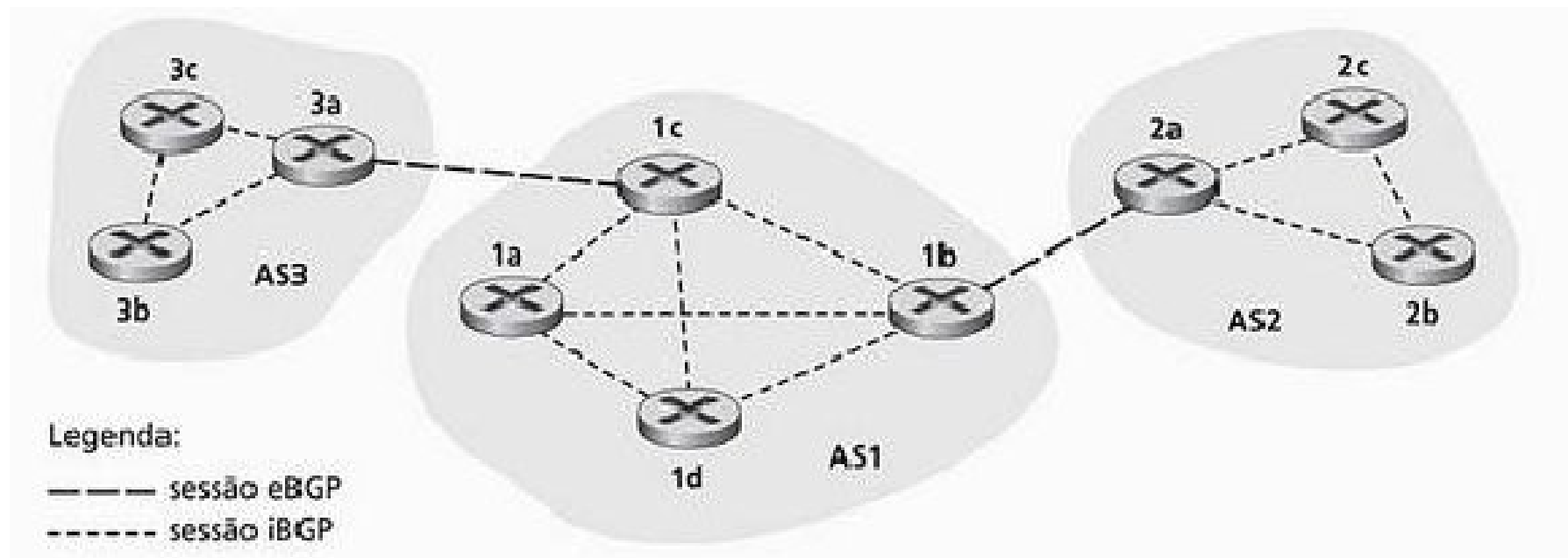
- **“Sessão BGP Externa”** (eBGP) .. sessão BGP que abrange 02 sistemas autônomos diferentes através dos roteadores de borda.
- **“Sessão BGP Interna”** (iBGP) .. sessão BGP que abrange 02 roteadores diferentes no mesmo sistema autônomo.



## 4 - Camada de Rede / 4.6 - Roteamento na Arq. TCP/IP

### ... 4.6.3 – Roteamento BGP

- Anúncio de Prefixo em uma Sessão BGP .. além do prefixo de sub-rede, outros atributos são também informados.
- “**AS-PATH**” .. contém os ASs pelos quais passou o anúncio para um dado prefixo de sub-rede, ou seja, quando um anúncio passa em um AS, o AS adiciona o seu ASN (AS Number) ao atributo AS-PATH.

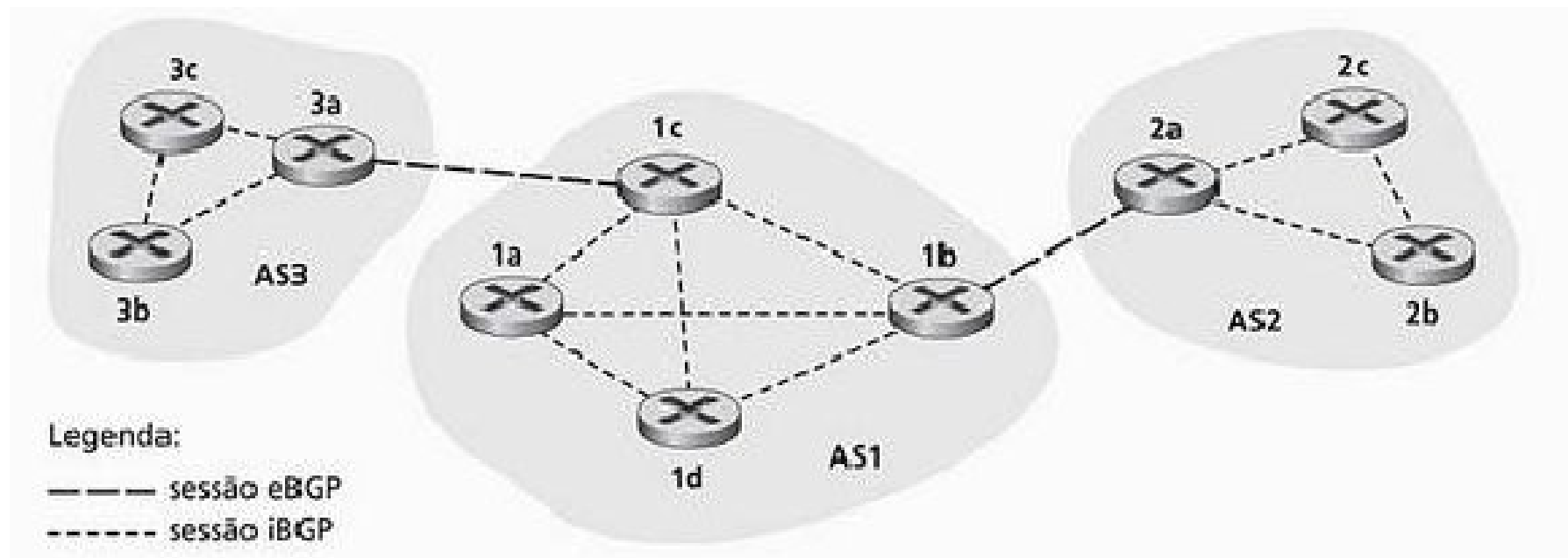




## 4 - Camada de Rede / 4.6 - Roteamento na Arq. TCP/IP

### ... 4.6.3 – Roteamento BGP

- Anúncio de Prefixo em uma Sessão BGP .. além do prefixo de sub-rede, outros atributos são também informados.
- ... pode ser usado para detectar “**looping**” de anúncios, caso um roteador perceba que seu ASN está contido na lista do anúncio.
- “**NEXT-HOP**” .. interface do roteador que inicia o AS-PATH.

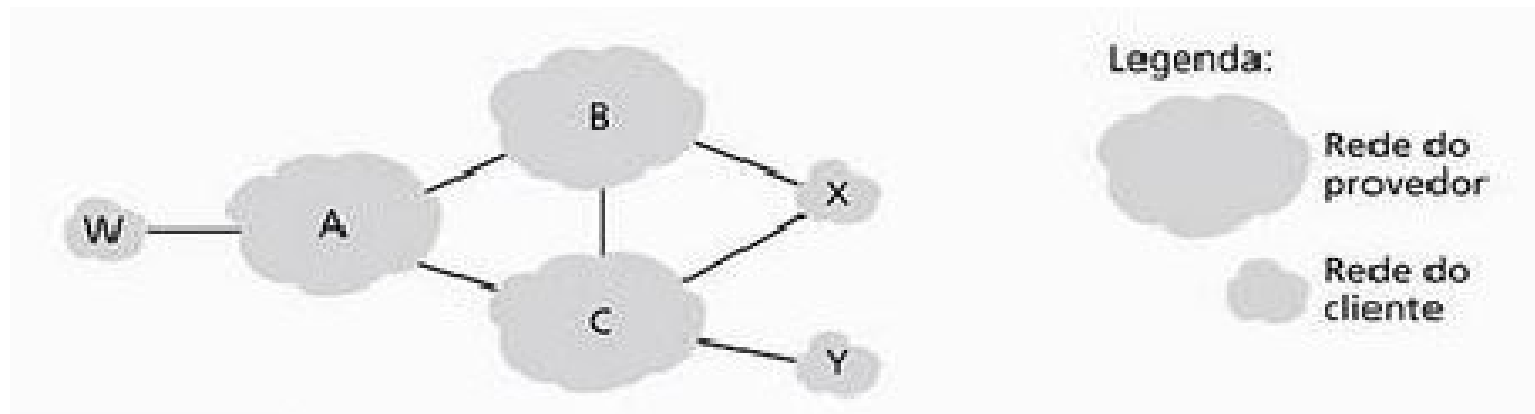


### ... 4.6.3 – Roteamento BGP

- “**exterior**” BGP (eBGP) e “**interior**” BGP (iBGP) ... podem conhecer mais do que uma rota para qualquer prefixo determinado e, neste caso cabe ao BGP selecionar uma das rotas possíveis.
- ... rotas recebem como um de seus atributos um valor de preferência local, permitindo-se que seja selecionado rotas com valores mais altos.
- e.g., considere 06 sistemas autônomos “A”, “B”, “C”, “W”, “X” e “Y” interconectados, sendo que “W”, “X” e “Y” são redes “stub” e “A”, “B” e “C” são redes provedoras de “backbone”.

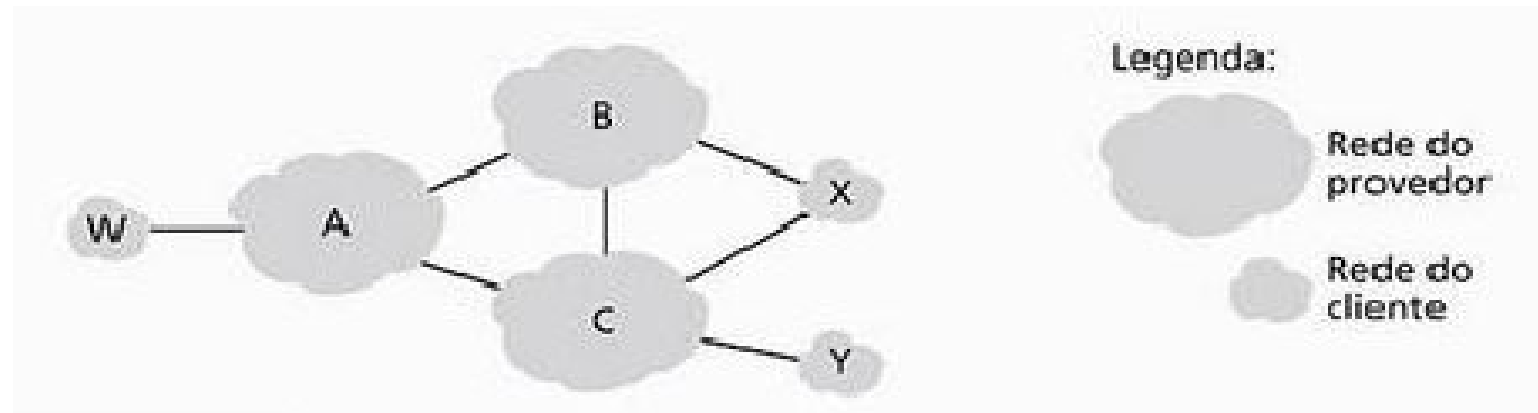
### ... 4.6.3 – Roteamento BGP

- e.g., considere 06 sistemas autônomos “A”, “B”, “C”, “W”, “X” e “Y” interconectados, sendo que “W”, “X” e “Y” são redes “stub” e “A”, “B” e “C” são redes provedoras de “backbone”.
- ... como “A”, “B” e “C” estão conectados uns aos outros, os mesmos fornecem informação completa sobre o BGP e suas redes clientes.
- “**regra**” .. tráfego que entrar na rede “stub” deve ser destinado a essa rede, e tráfego que sair da rede “stub” deve ter sido originado nela.



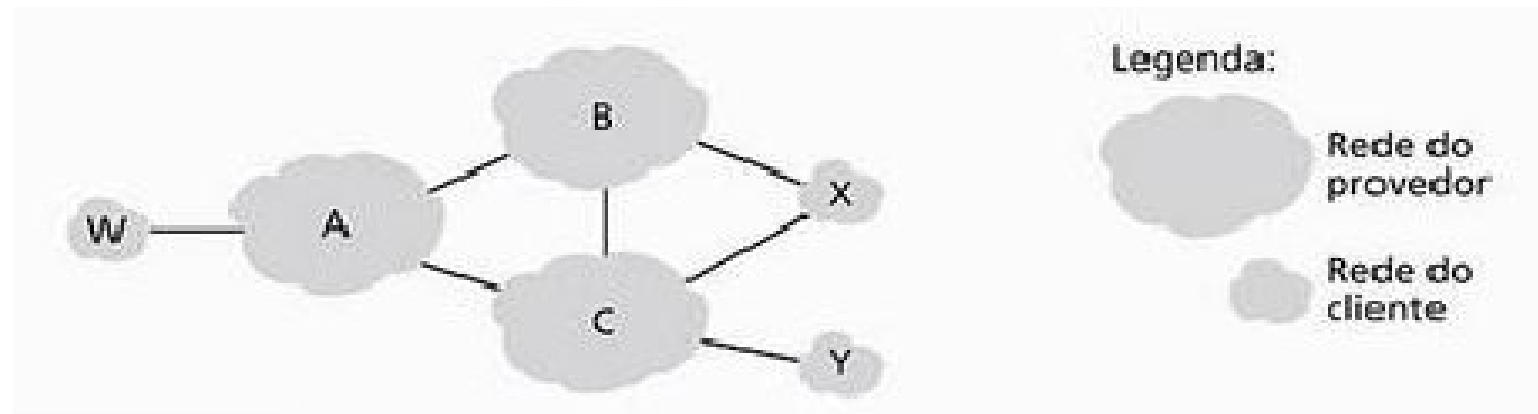
## ... 4.6.3 – Roteamento BGP

- e.g., considere 06 sistemas autônomos “A”, “B”, “C”, “W”, “X” e “Y” interconectados, sendo que “W”, “X” e “Y” são redes “stub” e “A”, “B” e “C” são redes provedoras de “backbone”.
- ... “W” e “Y” são redes “stubs”, enquanto “X” é uma rede “stub” com múltiplas interconexões pois se liga ao resto da rede por 02 provedores.
- ... considerando as restrições impostas, como “X” pode ser impedido de repassar tráfego entre “B” e “C” ??



## ... 4.6.3 – Roteamento BGP

- ... considerando as restrições impostas, como “X” pode ser impedido de repassar tráfego entre “B” e “C” ??
- **“solução”** .. tal restrição pode ser alcançada controlando o modo como as rotas BGP são anunciadas, ou seja, “X” não repassará datagramas pois é uma rede “stub” e não um rede provedora de “backbone”.
- ... “X” anuncia aos seus vizinhos que não há nenhum caminho para quaisquer outros destinos a não ser ele mesmo, ainda que conheça um caminho, digamos, “XCY” que chegue até a rede “Y”.

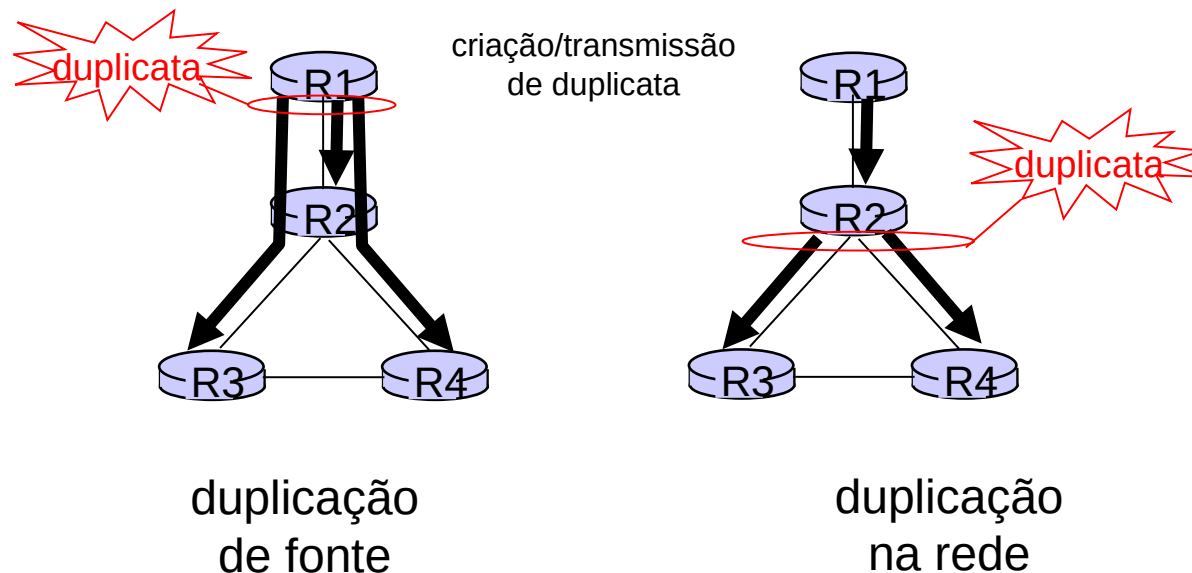


## 4.7 – Roteamento Broadcast e Multicast

- “**resumo**” .. nas seções anteriores, descreveu-se os protocolos de roteamento que suportam comunicação individual em que um único nó de origem envia um pacote a um único nó de destino.
- .. nesta seção, volta-se a atenção para protocolos de roteamento por difusão (“**broadcast**”) e para um grupo (“**multicast**”).
- “**roteamento broadcast**” .. serviço de entrega de pacotes enviado de um nó fonte a todos os outros nós da rede.
- “**roteamento multicast**” .. serviço de entrega de pacotes enviado de um nó fonte para um subconjunto de nós da rede.

## 4.7.1 – Roteamento Broadcast

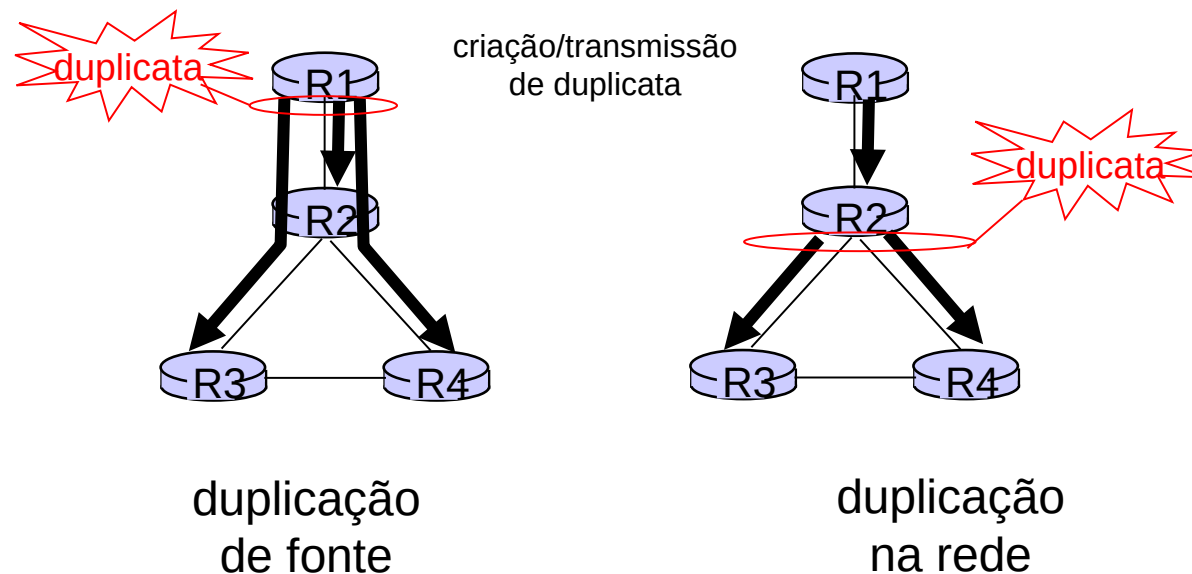
- “**roteamento broadcast**” .. serviço de entrega de pacotes enviado de um nó fonte a todos os outros nós da rede.
- “**mecanismo**” .. nó fonte replica “N” cópias do pacote para na sequência endereçar cada uma das cópias para cada um dos destinatários, ou seja, replicação do roteamento “unicast” de “N” caminhos.
- “**desvantagem**” .. ineficiência quanto ao uso de banda em razão da replicação de pacotes tantas vezes quanto o nro. de destinatários.



## 4 - Camada de Rede / 4.7 - Roteamento Broadcast e Multicast

### ... 4.7.1 – Roteamento Broadcast

- “**desvantagens adicionais**” ..
- “**conhecimento dos destinatários**” .. “broadcast” por replicação do “unicast” exige o conhecimento dos endereços de todos os destinos.
- “**atualização de rotas**” .. replicação do “unicast” é inapropriado ao ser utilizado na atualização de rotas “unicast” como no Alg. LS.





## ... 4.7.1 – Roteamento Broadcast

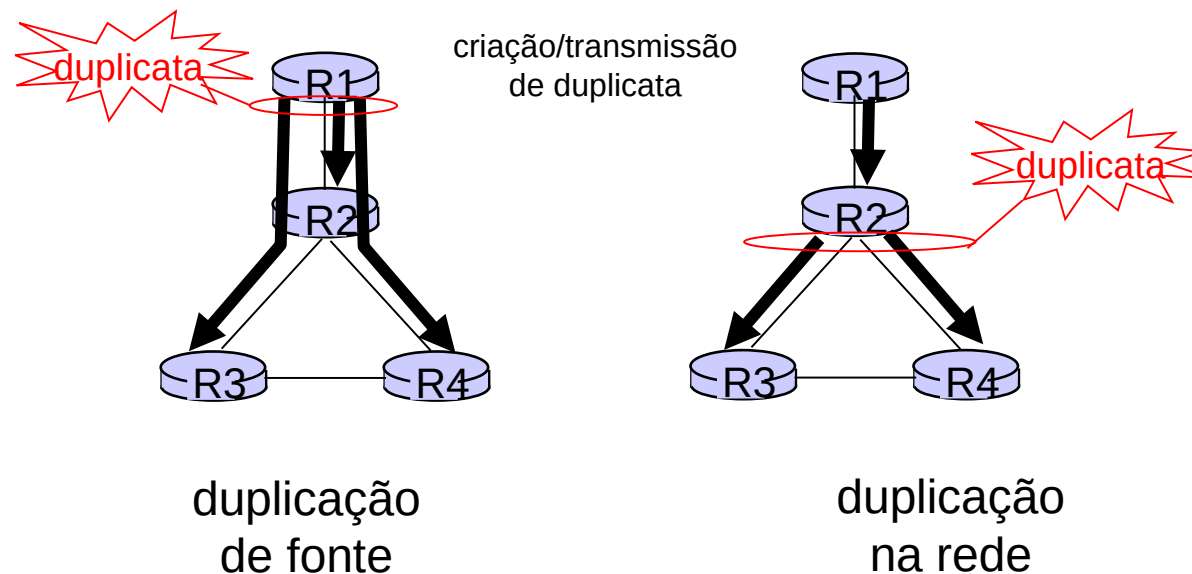
- “**objetivo**” .. discutir “soluções” de implementação do “broadcast” nas quais os nós da rede desempenham papel ativo na replicação, repasse e cálculo de rotas “broadcast”.
- “**premissa**” .. rede é modelada como um Grafo  $G = (N, E)$  onde “N” é o conjunto de nós e a “E” um conjunto de enlaces compostos por duplas, ou seja, 02 nós pertencentes ao conjunto “N”.
- “**algumas abordagens**” .. a) inundação não controlada; b) inundação controlada; c) variação da inundação controlada; d) “broadcast” por “spanning tree” ou árvore de escoamento.

## ... 4.7.1 – Roteamento Broadcast

- **“Inundação NÃO Controlada”** .. nó fonte envia uma cópia do pacote a todos os seus vizinhos, que repetem o processo de cópia e reenvio aos seus vizinhos.
- ... considerando que o grafo seja conectado, todos os nós do grafo vão receber uma cópia do pacote “broadcast”.
- **“problema”** ... se o grafo contiver ciclos, uma ou mais cópias de cada pacote do “broadcast” permanecerão no ciclo por um longo tempo.
- e.g., considere 04 nós “ $R_1$ ”, “ $R_2$ ”, “ $R_3$ ” e “ $R_4$ ” de modo que “ $R_1$ ” está conectado a “ $R_2$ ”, “ $R_2$ ” está conectado a “ $R_3$ ” e a “ $R_4$ ” e finalmente “ $R_3$ ” está conectado a “ $R_4$ ” >> grafo contém 01 ciclo.

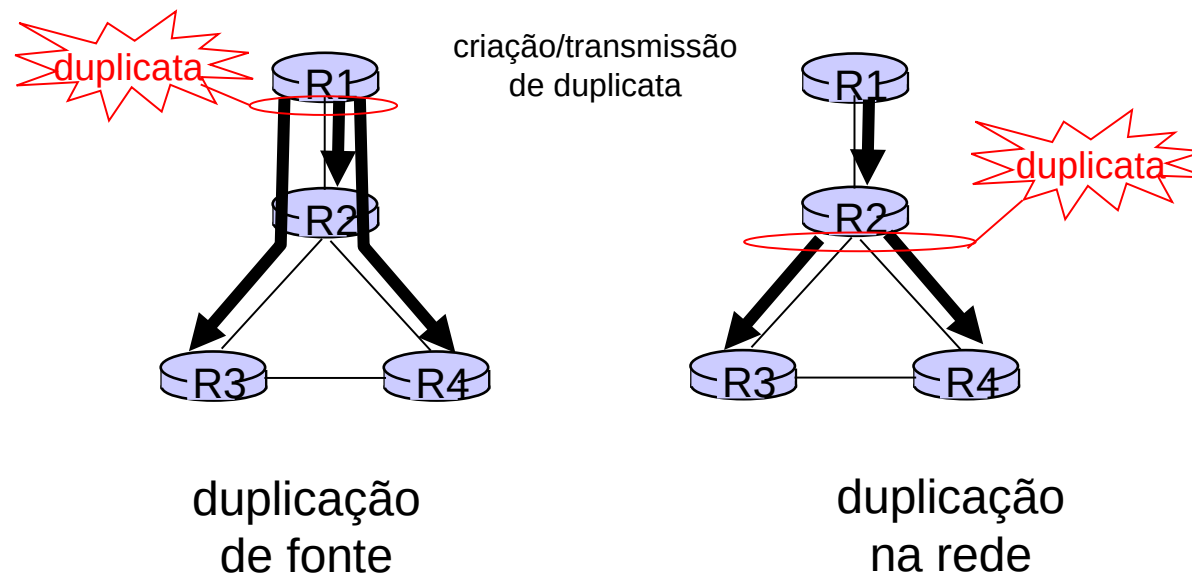
## ... 4.7.1 – Roteamento Broadcast

- e.g., considere 04 nós “ $R_1$ ”, “ $R_2$ ”, “ $R_3$ ” e “ $R_4$ ” de modo que “ $R_1$ ” está conectado a “ $R_2$ ”, “ $R_2$ ” está conectado a “ $R_3$ ” e a “ $R_4$ ” e finalmente “ $R_3$ ” está conectado a “ $R_4$ ” >> grafo contém 01 ciclo.
- ... “ $R_2$ ” inunda “ $R_3$ ”; “ $R_3$ ” inunda “ $R_4$ ”; “ $R_4$ ” inunda “ $R_2$ ” e “ $R_2$ ” inunda “ $R_3$ ” >> ciclo sem fim de pacotes “broadcast” em ambos os sentidos.



## ... 4.7.1 – Roteamento Broadcast

- e.g., considere 04 nós “ $R_1$ ”, “ $R_2$ ”, “ $R_3$ ” e “ $R_4$ ” de modo que “ $R_1$ ” está conectado a “ $R_2$ ”, “ $R_2$ ” está conectado a “ $R_3$ ” e a “ $R_4$ ” e finalmente “ $R_3$ ” está conectado a “ $R_4$ ” >> grafo contém 01 ciclo.
- **“tempestade de broadcast”** - multiplicação infindável de pacotes broadcast pode comprometer a utilização da rede por outras aplicações.



## ... 4.7.1 – Roteamento Broadcast

- **“Inundação Controlada”** - nó fonte envia uma cópia do pacote a todos os seus vizinhos, que por sua vez repetem o processo de cópia e reenvio aos seus vizinhos de forma controlada.
- ... cada nó decide se deve ou não repassar um pacote, em razão, p.ex. de já ter recebido e repassado uma cópia anterior do mesmo pacote.
- **“mecanismo”** .. contemplar no pacote “broadcast” informações como nro. de sequência e endereço do nó fonte ou identificador.
- ... nó mantém tais informações para cada pacote recebido, duplicado e repassado, de modo que possa consultá-lo e decidir o que fazer.

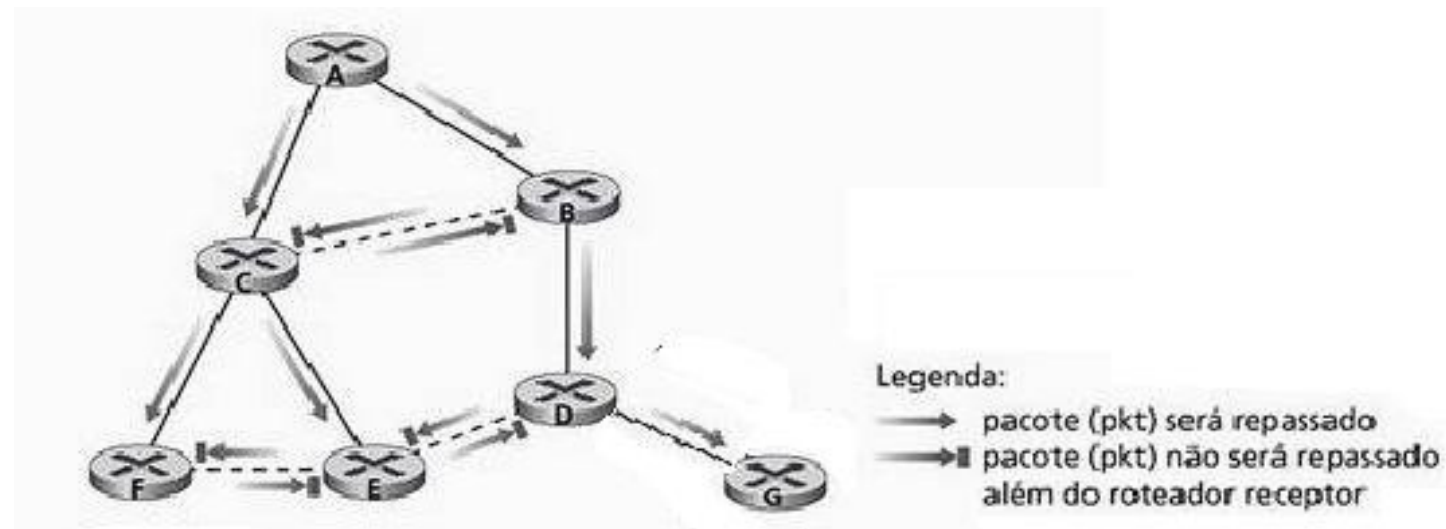
## ... 4.7.1 – Roteamento Broadcast

- **“Variação da Inundação Controlada”** - roteador transmite o pacote recebido para todos os seus enlaces exceto pelo enlace que representa o caminho de menor custo até o remetente.
- ... ou seja, enlace através do qual o pacote foi recebido deve ser o enlace de caminho mais curto até o remetente, caso contrário o roteador simplesmente descarta o pacote.
- ... por esta razão, esta abordagem é também chamada de **“repasse pelo caminho inverso”** (Reverse Path Forwarding – RPF).

## 4 - Camada de Rede / 4.7 - Roteamento Broadcast e Multicast

### ... 4.7.1 – Roteamento Broadcast

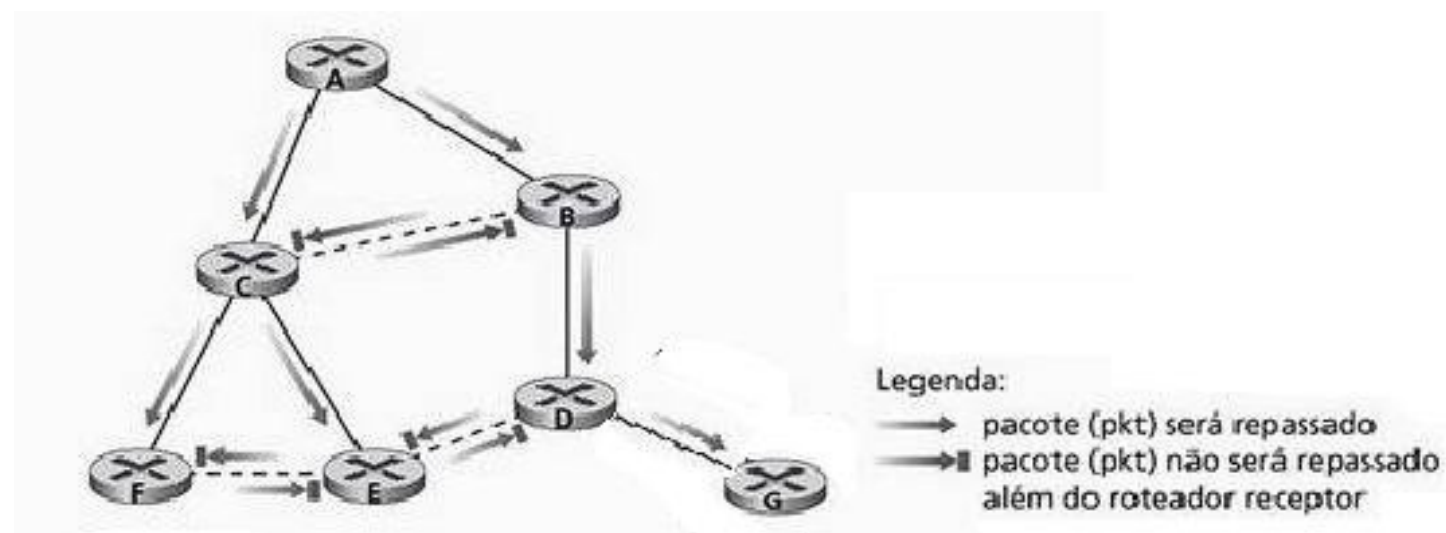
- e.g., seja uma rede com 07 nós “A”, “B”, “C”, “D”, “E”, “F” e “G” interconectados conforme o diagrama abaixo.
- (1) “A” inicia o “broadcast” para os nós “B” e “C”.
- (2) “B” repassa o pacote da fonte “A” que recebeu de “A” para “C” e “D” pois o recebeu pelo caminho de menor custo até “A”.



## 4 - Camada de Rede / 4.7 - Roteamento Broadcast e Multicast

### ... 4.7.1 – Roteamento Broadcast

- e.g., seja uma rede com 07 nós “A”, “B”, “C”, “D”, “E”, “F” e “G” interconectados conforme o diagrama abaixo.
- (3) “C” repassa o pacote da fonte “A” para “B”, “E” e “F”, (enlaces de saída) pois o recebeu pelo caminho de menor custo até “A”.
- (4) “B” descarta quaisquer pacotes da fonte “A” que receber de quaisquer outros nós, p.ex., “C” e “D”, pois estes nós não estão no caminho de menor custo até o nó “A”.

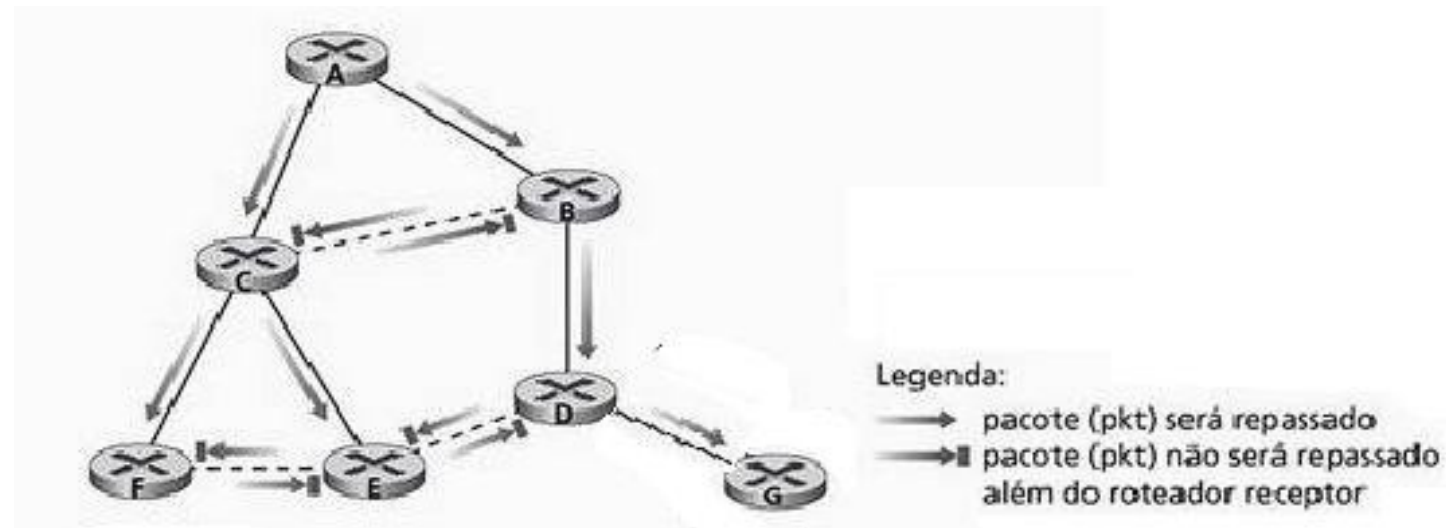




## 4 - Camada de Rede / 4.7 - Roteamento Broadcast e Multicast

### ... 4.7.1 – Roteamento Broadcast

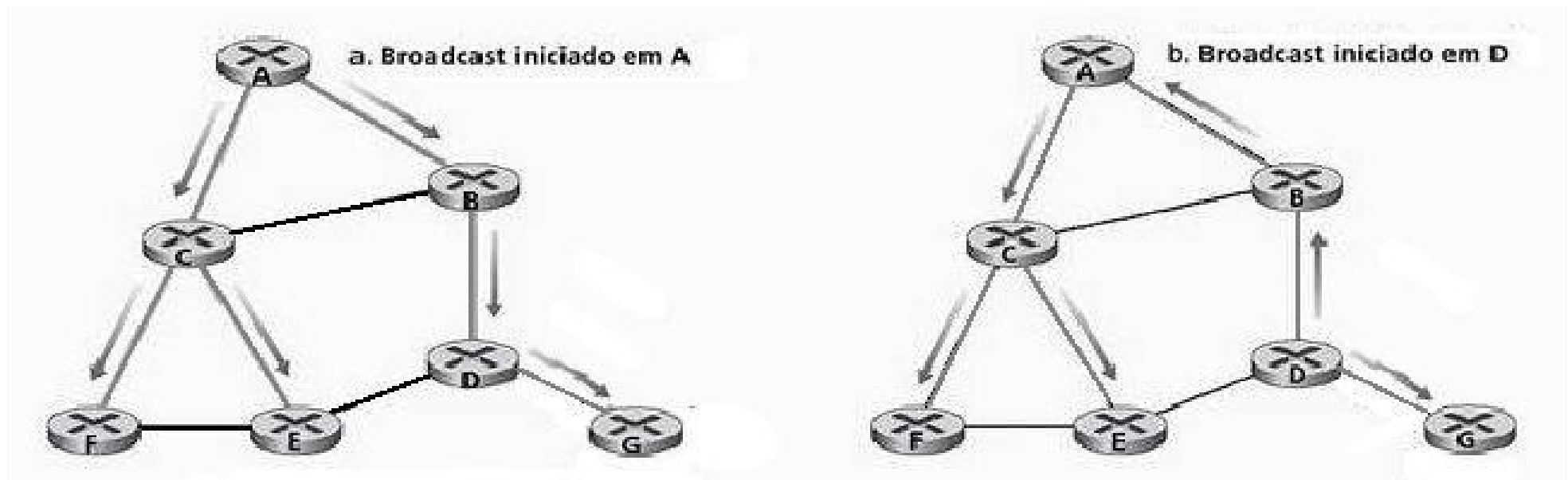
- e.g., seja uma rede com 07 nós “A”, “B”, “C”, “D”, “E”, “F” e “G” interconectados conforme o diagrama abaixo.
- (5) “C” descarta quaisquer pacotes da fonte “A” que receber de quaisquer outros nós, p.ex., “B”, “E” e “F”, pois estes nós não estão no caminho de menor custo até o nó “A”.
- **“reverse path forwarding”** .. precisa conhecer apenas o próximo vizinho em seu caminho “unicast” mais curto até o remetente e usa a identidade do vizinho para determinar se repassa ou não um pacote.



## 4 - Camada de Rede / 4.7 - Roteamento Broadcast e Multicast

### ... 4.7.1 – Roteamento Broadcast

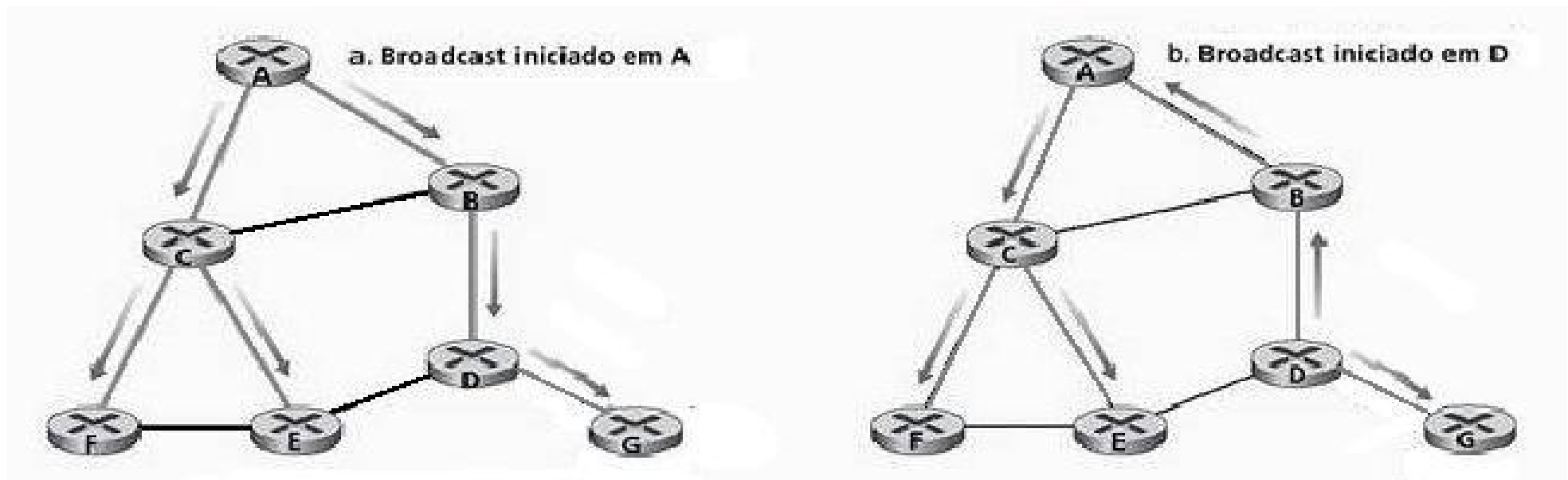
- Broadcast por “**Spanning Tree**” .. roteador transmite o pacote recebido para aos enlaces pertencentes a “spanning tree”.
- ... mesmo a inundação controlada com nros de sequência ou o repasse pelo caminho reverso não evitam o envio de pacotes redundantes.



## 4 - Camada de Rede / 4.7 - Roteamento Broadcast e Multicast

### ... 4.7.1 – Roteamento Broadcast

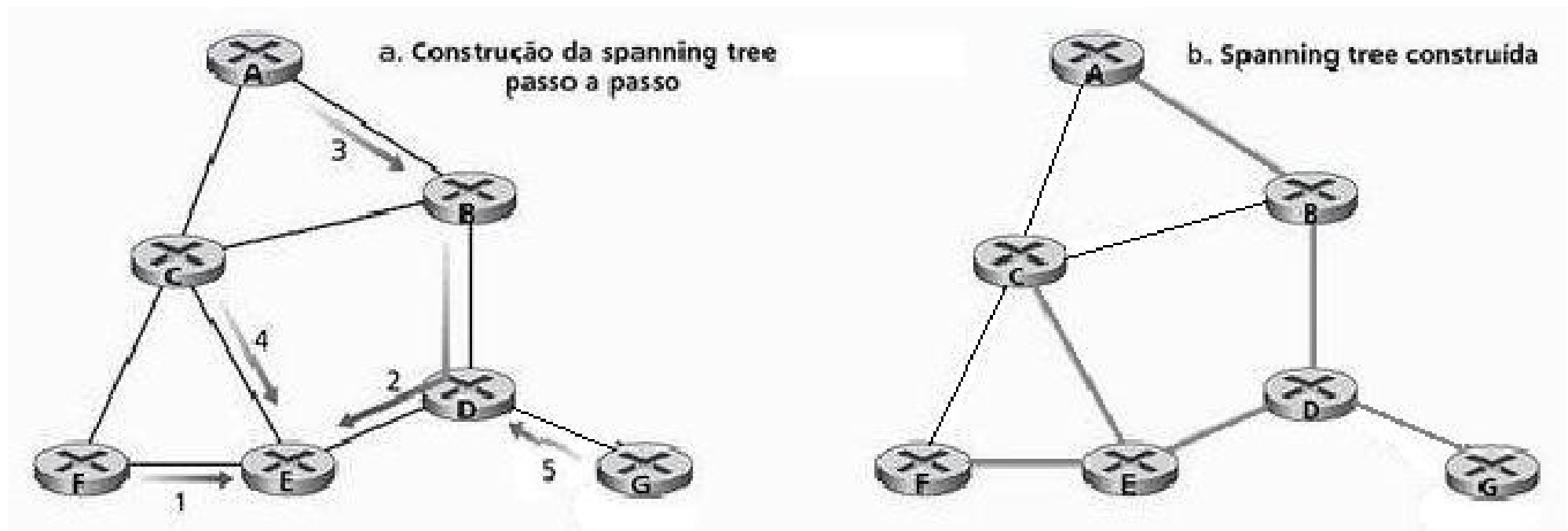
- “**Spanning Tree**” de um Grafo  $G = (N, E)$  é um Grafo  $G' = (N, E')$  onde “ $E'$ ” é um subconjunto de enlaces em “ $E$ ” e no qual o grafo “ $G'$ ” é conectado e não contém nenhum ciclo.
- (1) primeiro passo consiste na criação da “spanning tree”.
- (2) se um nó deseja enviar um pacote, o nó simplesmente o repassa para todos os seus vizinhos na “spanning tree”.



## 4 - Camada de Rede / 4.7 - Roteamento Broadcast e Multicast

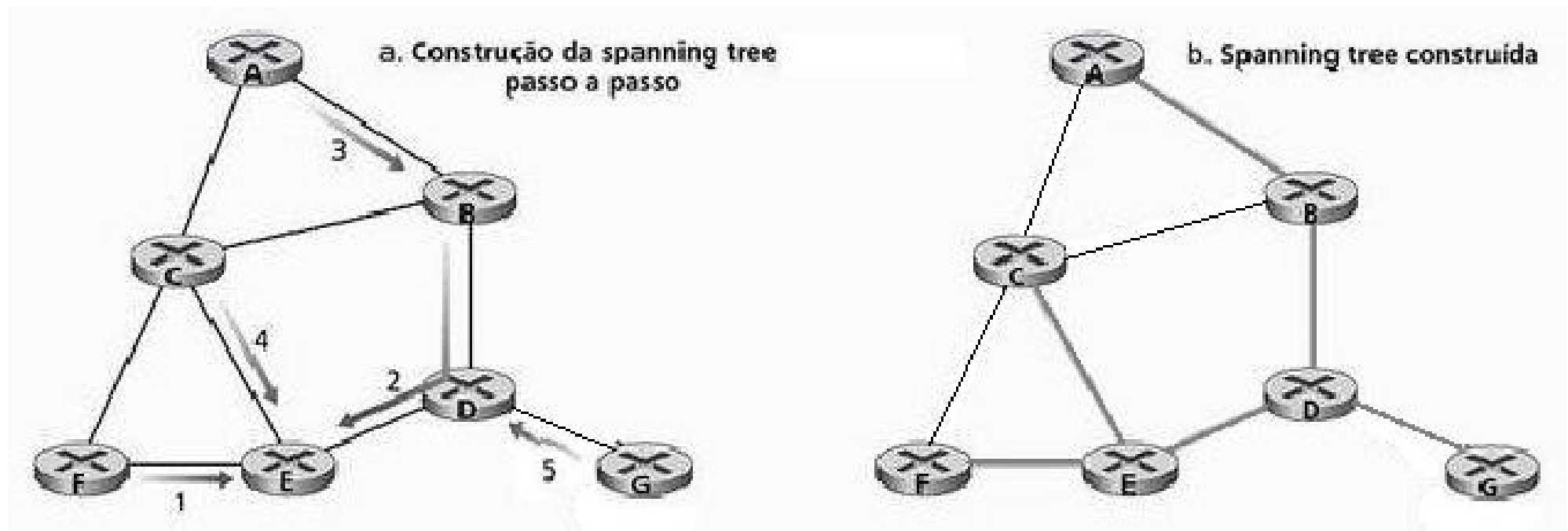
### ... 4.7.1 – Roteamento Broadcast

- “**complexidade**” - criação a manutenção da “spanning tree”, assim, surge a necessidade de algoritmos distribuídos de árvore.
- “spanning tree” baseada em um centro – suponha que o nó “E” seja selecionado como o centro e que o nó “F” se junte a árvore e repasse a “E” uma mensagem de adesão.



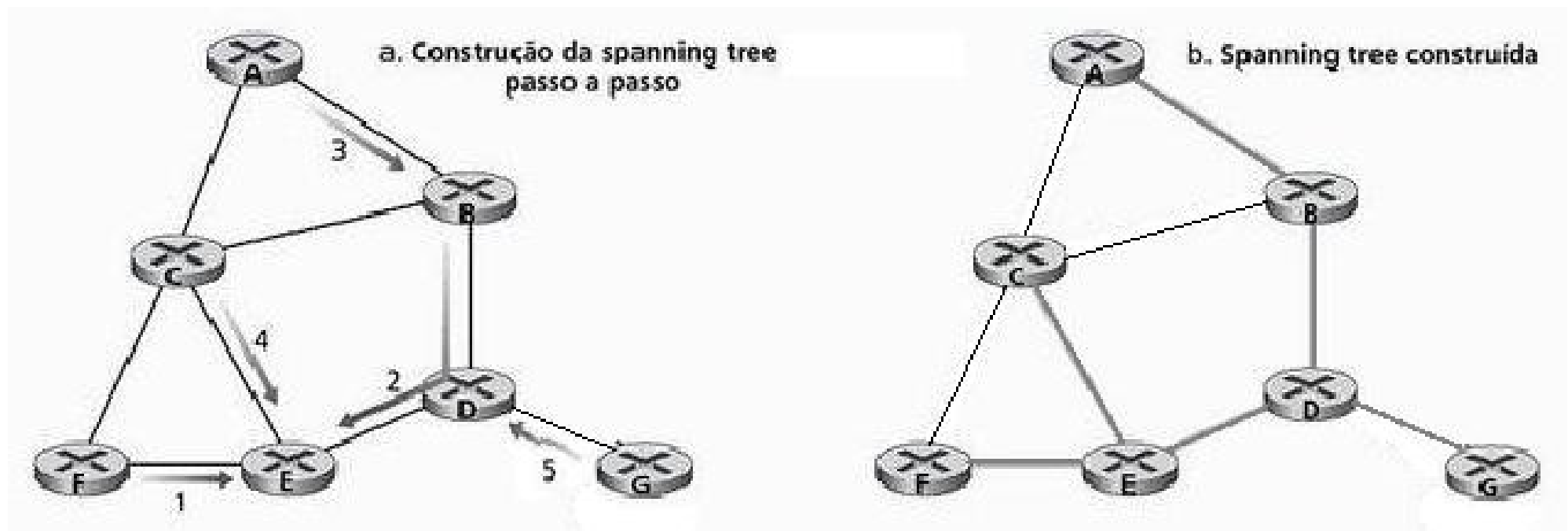
## ... 4.7.1 – Roteamento Broadcast

- Enlace “EF” se torna a “spanning tree” inicial e o nó “B” se junta a “spanning tree” enviando a “E” sua mensagem de adesão à árvore.
- .. assuma que a rota do caminho individual de “E” para “B” seja por “D”, então, a mensagem de adesão à árvore resulta no enxerto do caminho “BDE” à spanning tree.



### ... 4.7.1 – Roteamento Broadcast

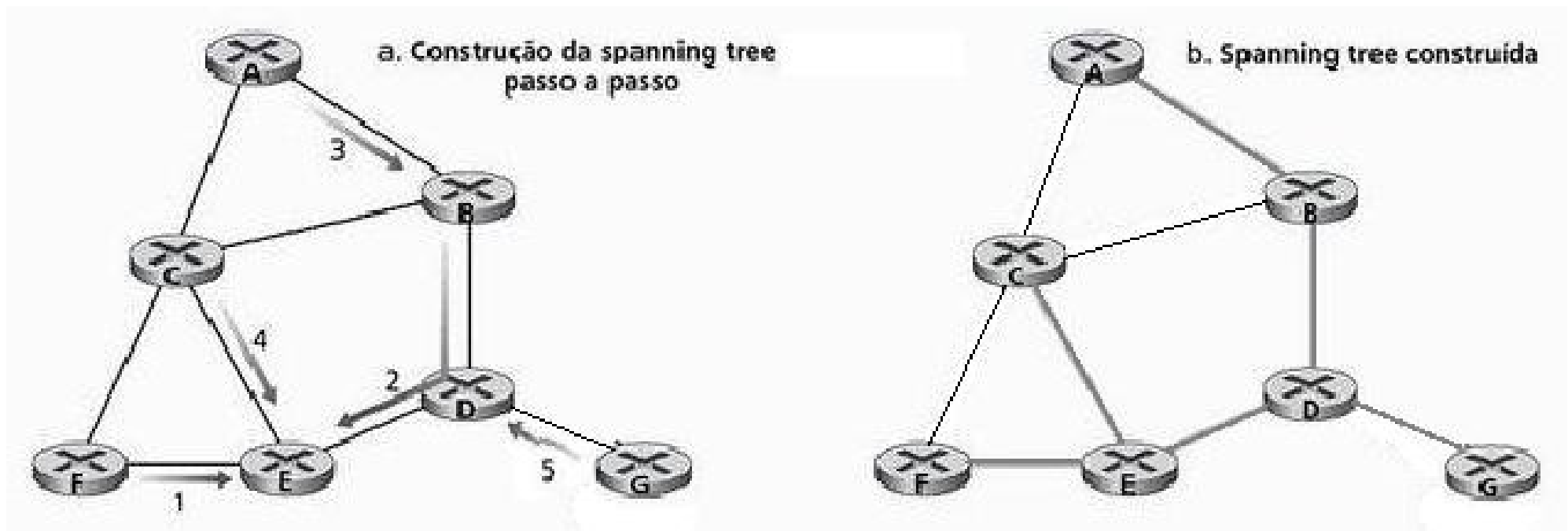
- .. na sequência, nó “A” se junta ao grupo repassando sua mensagem de adesão à árvore em direção a “E”.
- Se o caminho individual de “A” a “E” passar por “B” e uma vez que “B” já se juntou à “spanning tree”, a chegada da mensagem de adesão em “B” resultará no enxerto imediato do enlace “AB” à spanning tree.



## 4 - Camada de Rede / 4.7 - Roteamento Broadcast e Multicast

### ... 4.7.1 – Roteamento Broadcast

- Em seguida, o nó “C” se junta à “spanning tree” repassando sua mensagem de adesão à árvore diretamente a “E”.
- Por fim, como o roteamento individual de “G” para “E” tem de passar pelo nó “D”, quando “G” enviar sua mensagem de adesão à árvore a “E”, o enlace “GD” será enxertado à spanning tree no nó D.



## 4.7.2 – Roteamento Multicast

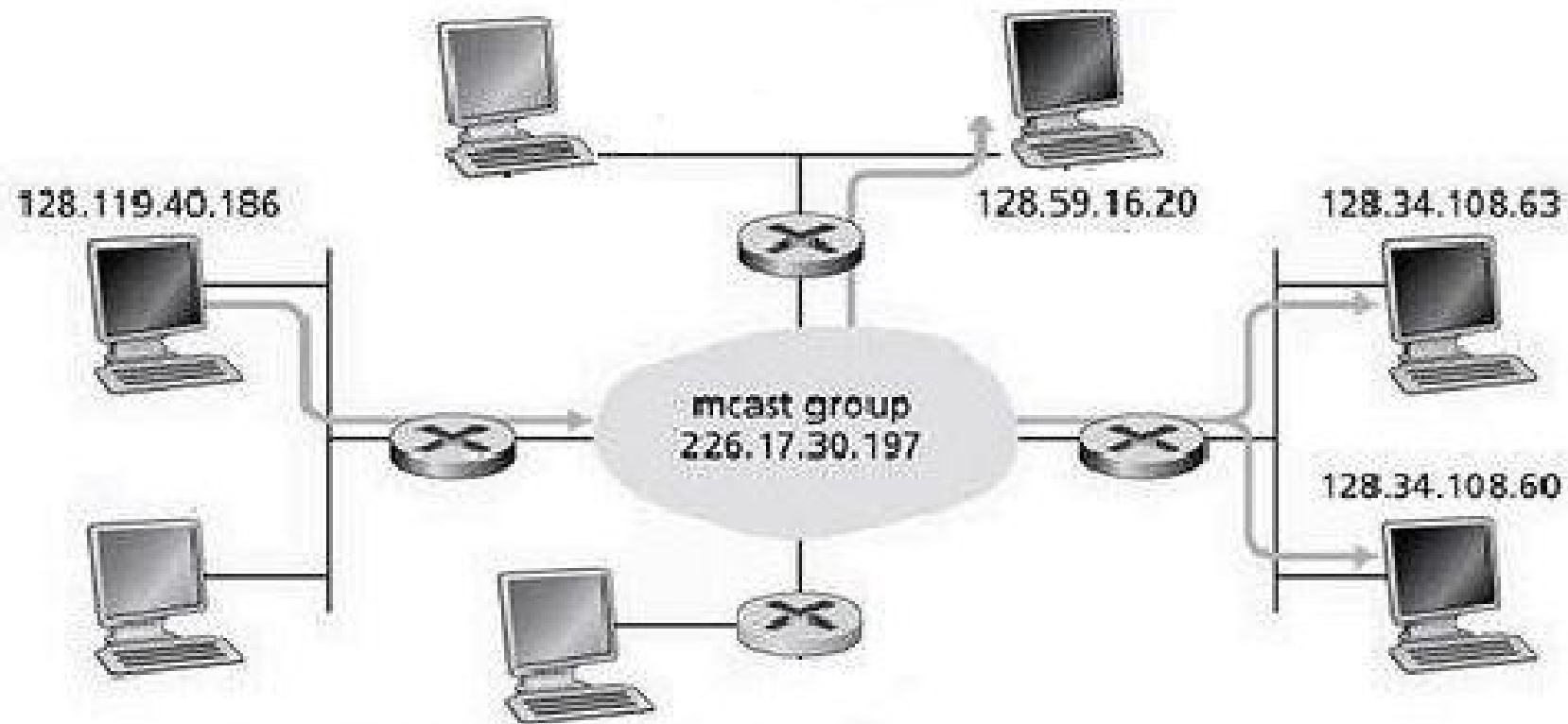
- “**roteamento multicast**” - serviço de entrega de pacotes enviado de um nó fonte para um subconjunto de nós da rede.
- Como identificar os destinatários de um pacote “multicast” ?
- Como endereçar um pacote a um dos destinatários ?
  
- “**solução**” - utilizar endereçamento indireto, ou seja, um identificador é utilizado para representar um grupo de nós.
- ... cópia do pacote endereçada ao grupo através do identificador do grupo é entregue a todos os destinatários associados ao grupo.
- ... Arq. TCP/IP um endereço IP Classe D é o identificador do grupo e o conjunto de nós associados ao endereço é o “grupo multicast”.



## 4 - Camada de Rede / 4.7 - Roteamento Broadcast e Multicast

### ... 4.7.2 – Roteamento Multicast

- Protocolo IGMP (Internet Group Management Protocol) .. opera sobre o roteador de 1º salto ligado diretamente ao “host”.



Legenda:



Roteador ao qual está ligado um membro do grupo

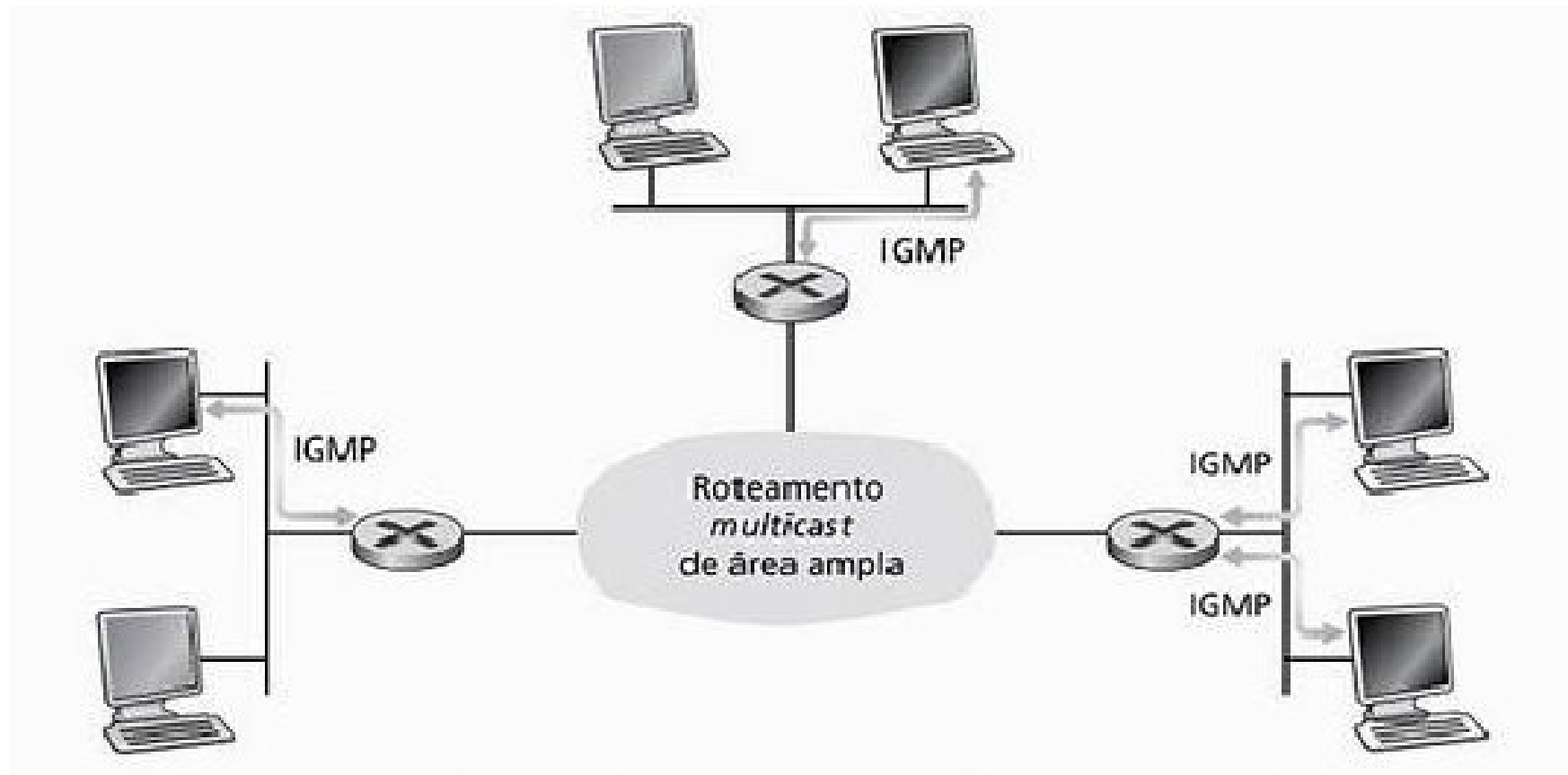


Roteador ao qual nenhum membro do grupo está ligado

## 4 - Camada de Rede / 4.7 - Roteamento Broadcast e Multicast

### ... 4.7.2 – Roteamento Multicast

- IGMP provê os meios para que um “host” informe ao roteador de 1º salto (roteador de borda) que uma aplicação do “host” quer se juntar a um grupo “multicast” específico.



## ... 4.7.2 – Roteamento Multicast

- IGMP não provê meios para coordenar os roteadores “multicast”, ou seja, está limitado a “host” e ao roteador de 1º salto.
- ... surge a necessidade de coordenar os roteadores participantes de grupos “multicast” para garantir que os datagramas sejam entregues aos seus destinos finais .. “hosts” que participam do grupo “multicast”.
- Protocolos “Multicast” da Camada de Rede:
- PIM (Protocol Independent Multicast)
- DVMRP (Distance Vector Multicast Routing Protocol)
- MOSPF (OSPF Multicast)

## ... 4.7.2 – Roteamento Multicast

- IGMP contempla msgs. de 03 tipos encapsuladas em datagramas IP com o nro. de protocolo igual a “2”.
- “**membership\_query**” .. msg. enviada pelo roteador a todos os “hosts” em uma interface para determinar o conjunto de todos os grupos “multicast” aos quais se ligaram os “hosts” naquela interface.
- “**membership\_report**” .. resposta dos “hosts” à msg. anterior ou solicitação do “host” para se juntar a um grupo “multicast”.
- “**leave\_group**” .. msg. opcional, pois o roteador pode detectar que um “host” não participa de um grupo “multicast” observando se o mesmo respondeu ou não à uma msg. “membership\_query”.