

PTC 3360

3. A Camada de Rede – Parte IV

(Kurose, Seções 4.3 e 4.4)

Setembro 2023

Capítulo 2 - Conteúdo

2.1 A camada de aplicação

2.2 Princípios da transferência confiável de dados (recorte da camada de transporte)

2.3 Camada de rede

A Introdução à camada de rede

B O que tem dentro de um roteador?

C Atrasos e vazão

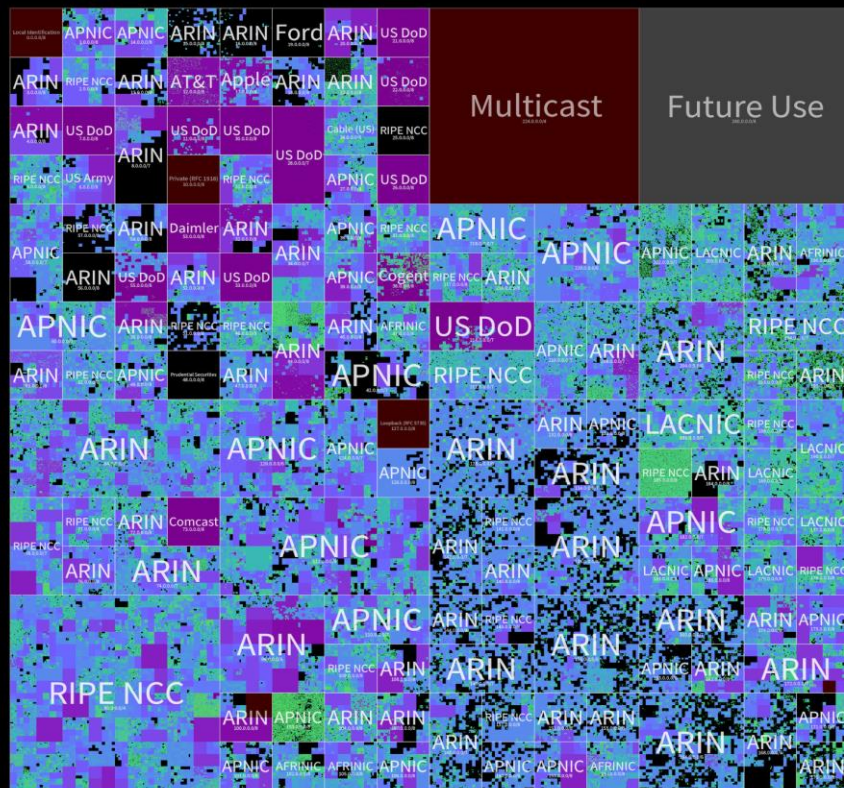
D **Endereçamento IP:** Sub-redes e endereços;
obtendo um endereço IP; **esgotamento do IPv4**

E Repasse generalizado e SDN

Poucos endereços IPv4

The IPv4 Internet

January 2023



This is a visual representation of the Internet's IPv4 address space, as seen by the University of Oregon Route Views project on January 1st, 2023. The map is divided into 16,777,216 pixels, with each pixel representing a single /24, or 256 IPv4 addresses. Each contiguous block represents an announcement in the global BGP address table. The address space is presented as a Hilbert curve, a fractal representation ensuring all blocks appear as squares or rectangles. The colors represent the smallest announcement within that pixel, from /8 to /32. Black areas are unannounced, red areas are special blocks defined in RFC 5735 which are not globally routable, and grey is reserved for future use (though is not expected to ever be globally routable, due to historical limitations).

Announcement Size



Copyright © 2015-2023 Ryan Finnin. This work is licensed under a Creative Commons Attribution 4.0 International License. Based on works and software © 2009 The Measurement Factory. Based on data collected by the University of Oregon.

- O gráfico tem 16.777.216 pixels
- Cada pixel representa um endereço /24 (256 endereços IPv4)
- As cores indicam alocações de /8 a /32
- Cor preta indica não anunciada
- Cor vermelha não é globalmente roteável
- Cor cinza é para uso futuro, mas dificilmente servirá para roteamento global

Asia-Pacific
Network
Information
Center



RIPE Network
Coordination
Centre



Fonte

Camada de rede 4-3

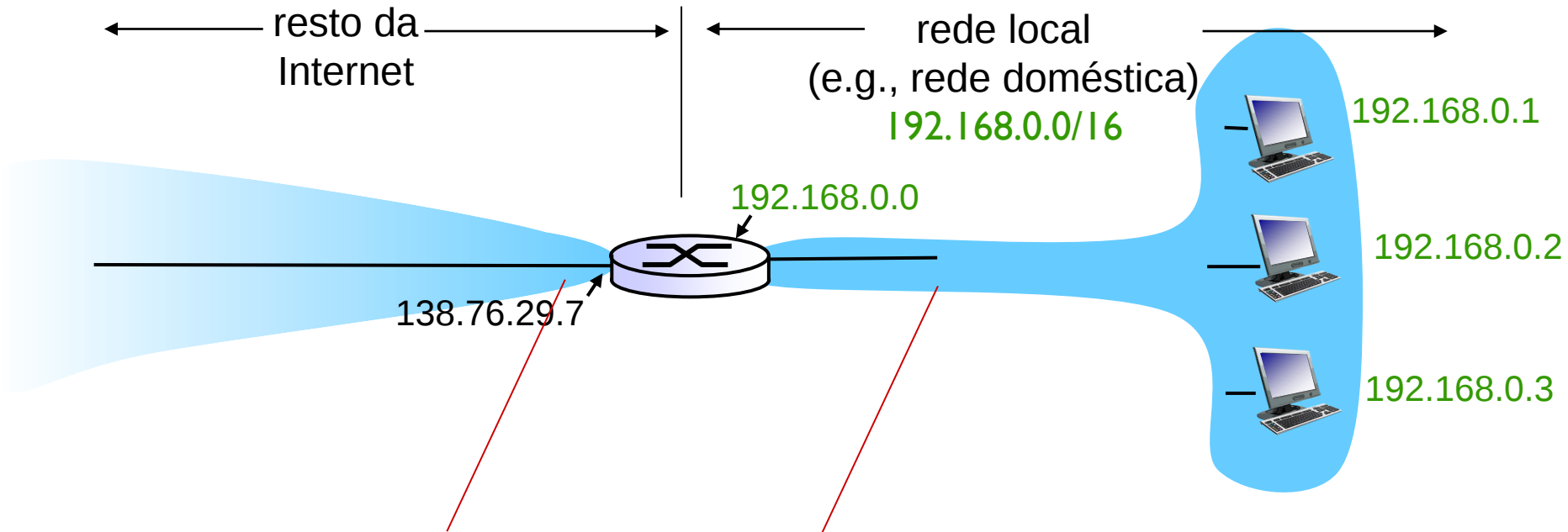
Poucos endereços IPv4

Q: Como resolver problema de insuficiência de endereços IPv4?

Resposta 1 (provisória): NAT

Resposta 2 (definitiva): IPv6

NAT: Network Address Translation



todos os datagramas *saem*
da rede local com
mesmo único endereço
fonte IP NAT: 138.76.29.7,
diferentes números de
portas fontes

datagramas com fonte ou destino
nessa rede tem endereço 192.168.0.0/16
para fonte ou destino (como usual)

NAT: Network Address Translation

Motivação:

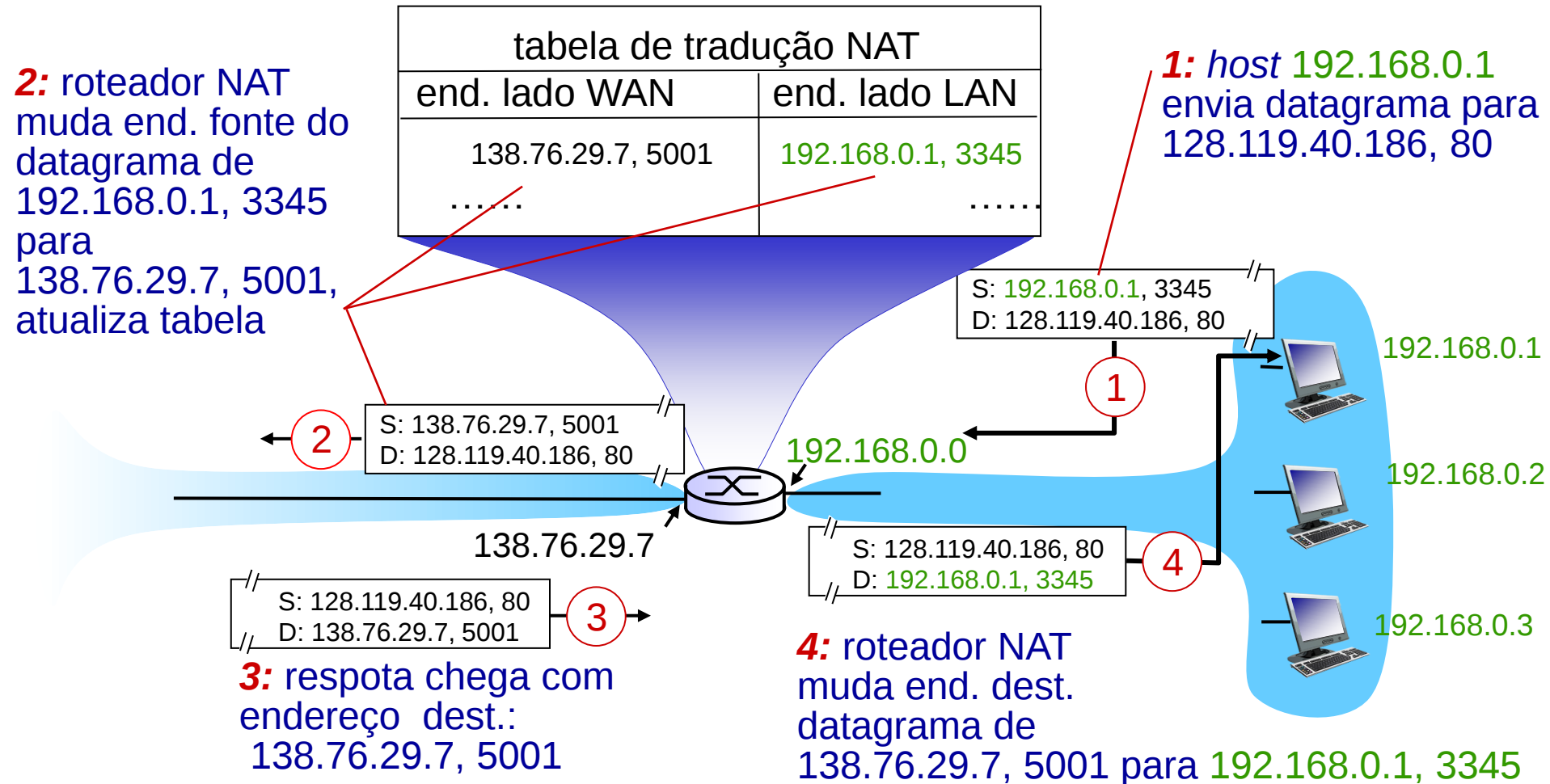
- ❖ Rede local inteira usa apenas um único IP do ponto de vista do mundo exterior:
 - Não é necessário intervalo de endereços do ISP: apenas um endereço IP para todos os dispositivos
 - Pode-se mudar endereços dos dispositivos na rede local sem notificar mundo exterior
 - Pode-se mudar ISP sem mudar endereços dos dispositivos na rede local
 - Dispositivos na rede local não são explicitamente endereçáveis – invisíveis do ponto de vista do mundo exterior (um aumento na segurança)

NAT: Network Address Translation

Implementação: roteador NAT precisa:

- *Datagramas saindo:* trocar (endereço IP fonte, # porta) de cada datagrama que sai por (endereço NAT IP, novo # porta)
... clientes/servidores remotos vão responder usando (endereço NAT IP, novo # porta) como endereço destino
- *Lembrar (em uma tabela de tradução NAT)* cada par de tradução (endereço IP fonte, # porta) para (endereço NAT IP, novo # porta)
- *Datagramas chegando :* trocar (endereço NAT IP, novo # porta) nos campos destino de cada datagrama que chega pelo correspondente (endereço IP fonte, # porta) armazenado na tabela NAT
- RFC1918 – endereços para rede privada: 10/8, 172.16/12 e 192.168/16
- Obs: Algumas operadoras usam o carrier-grade NAT para os seus clientes na faixa de endereços 100.64.0.0/10

NAT: Network Address Translation



NAT: Network Address Translation

- ❖ Campo número de porta tem 16 bits:
 - Mais de 60 000 conexões simultâneas com um único endereço do lado WAN!
- ❖ NAT é controverso!
 - Roteadores deveriam processar apenas até a camada de rede! Não deveriam acessar # porta!
 - Servidores em redes NAT têm problemas!
 - Viola argumento fim-a-fim:
 - A possibilidade NAT precisa ser levada em conta por projetistas de *apps*, e.g. aplicações P2P
 - Falta de endereços deveria ser resolvida usando IPv6

IPv6: Motivação

- ❖ *Motivação inicial:* espaço de endereços de 32 bits em breve estará completamente alocada (percebeu-se no começo da década de 90!)
- ❖ IPv6 – [RFC 1883](#) (dez/1995)... [RFC 7112](#) (jan/2014)
- ❖ Feb. 2011 – IANA alocou último conjunto de endereços IPv4 a registrador regional
- ❖ Agosto 2020 – [Fim do estoque de endereços IPv4 para a região da América Latina e o Caribe](#)
- ❖ *Motivação adicional:*
 - Cabeçalho fixo e ausência de verificação de erros ajuda a aumentar velocidade de processamento e repasse

Endereços IPv6

- ❖ Endereços com 128 bits ou 16 bytes
- ❖ Permite $3,4 \times 10^{38}$ endereços! (Muito ou pouco?)
- ❖ Existem da ordem de 10^{22} estrelas no universo
- ❖ Existem da ordem de 10^{22} grãos de areia na Terra

Endereços IPv6

The IPv6 Internet

January 2023



This is a visual representation of the Internet's in-use global unicast IPv6 address space, as seen by the University of Oregon Route Views project on January 1st, 2023. The map is divided into 16,777,216 pixels, with each pixel representing 16 ISP-level /32 allocations. A /32 contains approximately 4.3 billion standard /64 networks, and a /64 contains approximately 18 quintillion IP addresses. Each contiguous block represents an announcement in the global BGP address table. The address space is presented as a Hilbert curve, a fractal representation ensuring all blocks appear as squares or rectangles. Colors represent the smallest announcement within that pixel, from /12 to /64. The space depicted is 2000::/4; 3000::/4 is currently allocated but not in use, and 11 more /4s may be allocated to meet future demand.

Announcement Size



Velociraptor Aerospace Dynamics
<https://vad.solutions/ipmap>



Copyright © 2015-2023 Ryan Frittle. This work is licensed under a Creative Commons Attribution 4.0 International License. Based on works and software © 2009 The Measurement Factory. Based on data collected by the University of Oregon.

- O gráfico tem 16.777.216 pixels
- Cada pixel representa um endereço /32 (i.e., 4,3 bilhões de endereços /64 e um /64 contém ~18 quintilhões (18×10^{18}) de endereços!!)
- As cores indicam alocações de /12 a /64

Fonte

Camada de rede 4-12

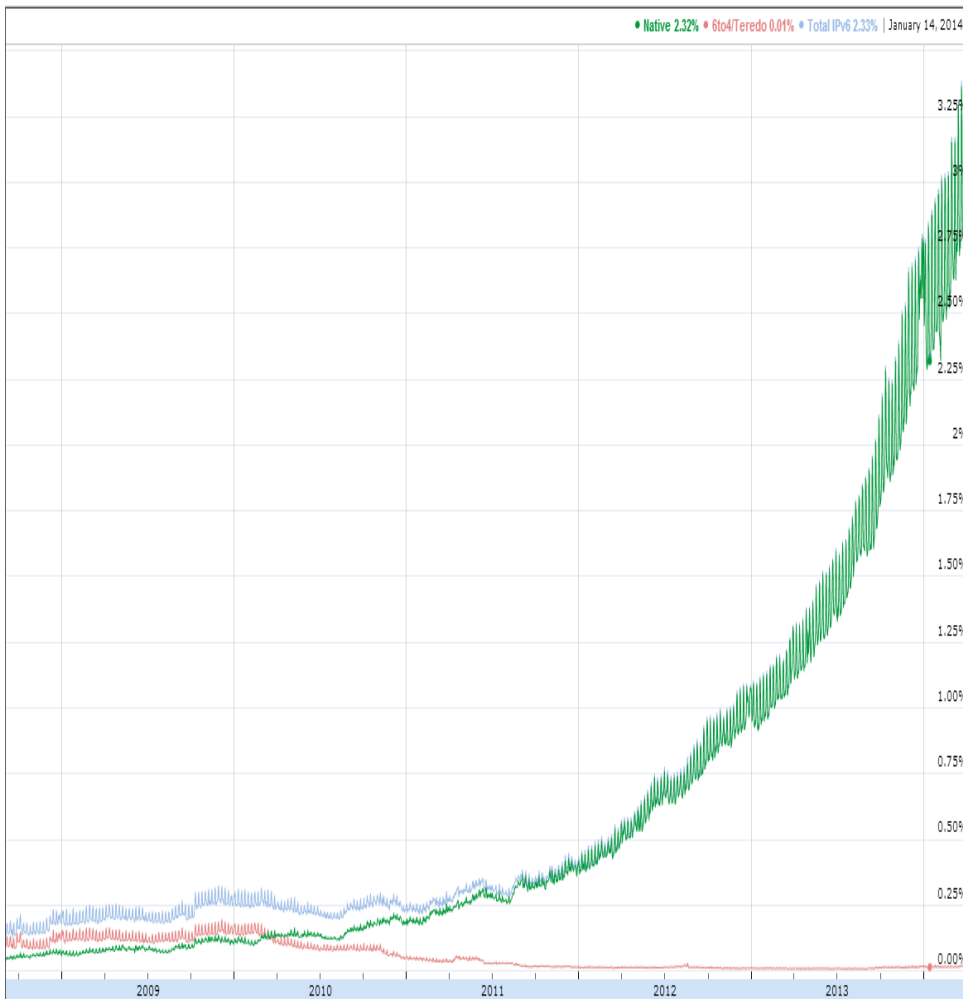
Endereços IPv6

- ❖ Os endereços IPv6 são escritos como oito grupos de 4 dígitos hexadecimais.
 - ❖ Por exemplo,
 - `2001:0db8:85a3:08d3:1319:8a2e:0370:7344`
 - ❖ Se um grupo de vários dígitos seguidos for 0000, pode ser omitido.
 - ❖ Por exemplo,
 - `2001:0db8:85a3:0000:0000:0000:0000:7344`
- é o mesmo endereço IPv6 que:
- `2001:0db8:85a3::7344`

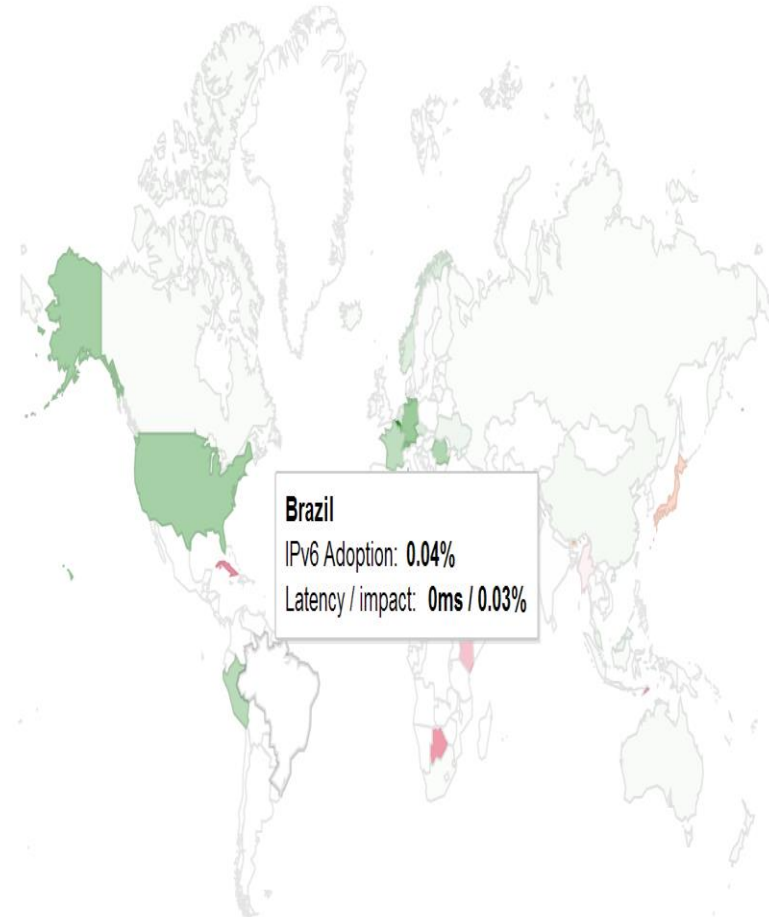
Estado em janeiro de 2014

Veja

- <http://ipv6.br/> (Exercício: Teste sua conectividade IPv6!)
- <http://www.google.com/intl/en/ipv6/statistics.html#tab=ipv6-adoption>

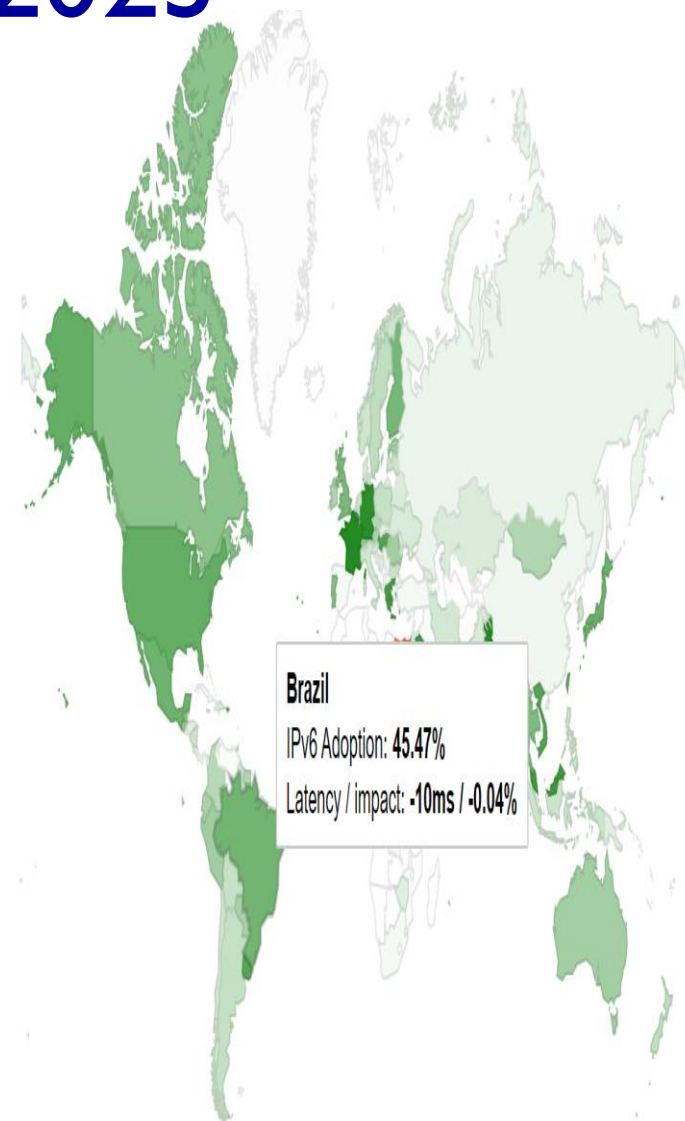
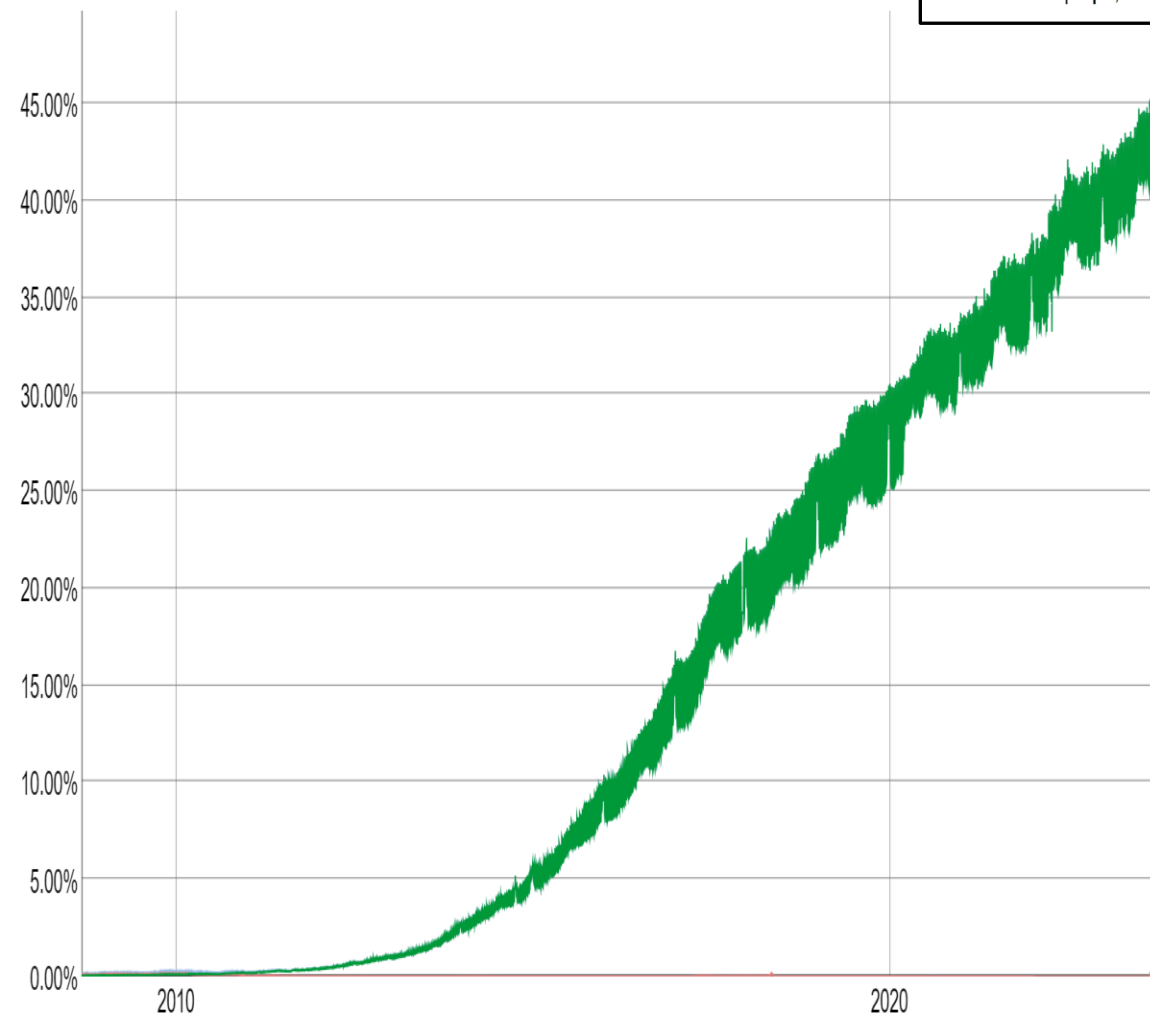


Per-Country IPv6 adoption

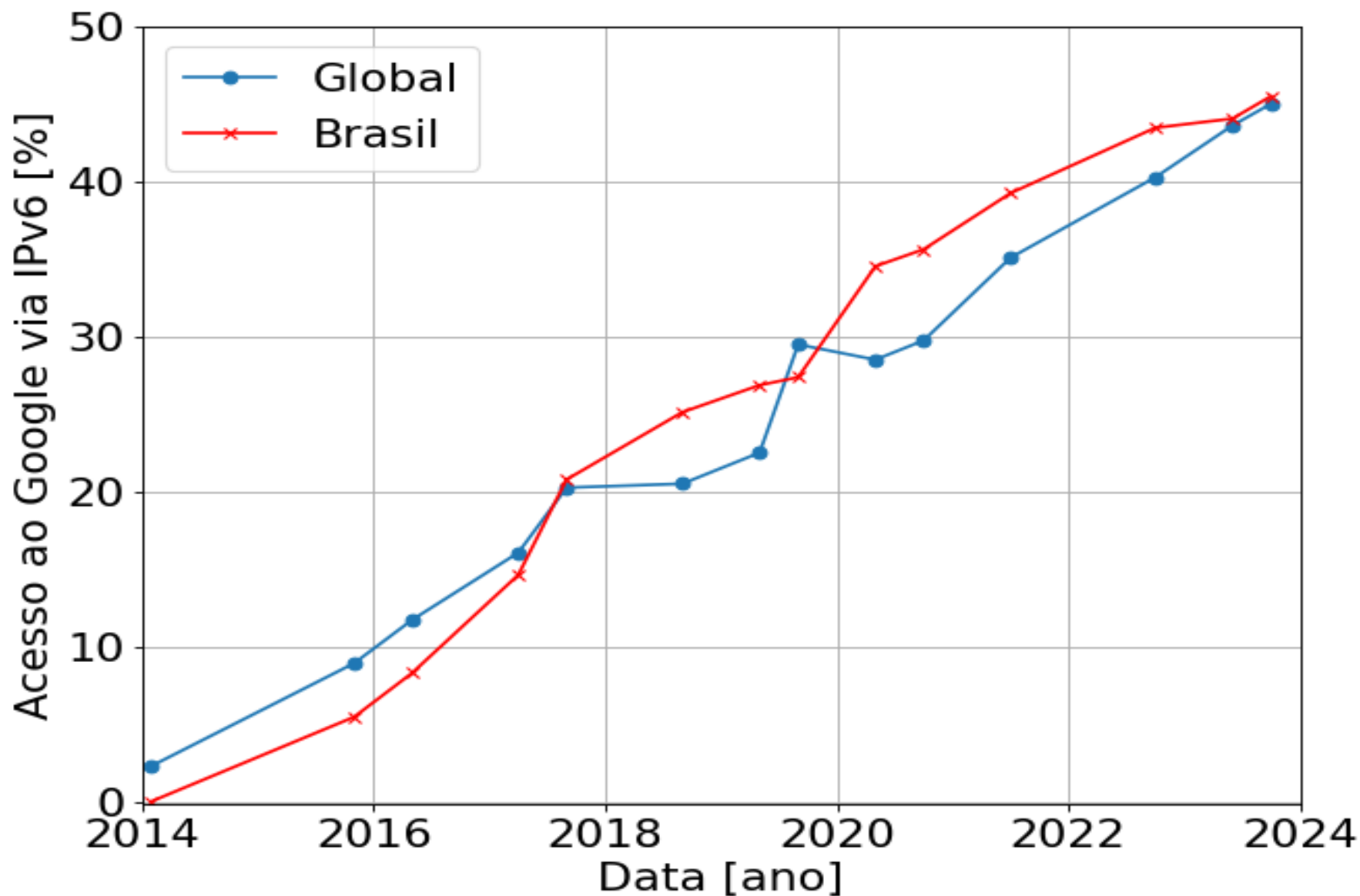


Estado atual: setembro 2023

Native: 44.98% 6to4/Teredo: 0.00% Total IPv6: 44.98% | Sep 9, 2023



Evolução (colab)



Capítulo 2 - Conteúdo

2.1 A camada de aplicação

2.2 Princípios da transferência confiável de dados (recorte da camada de transporte)

2.3 Camada de rede

A Introdução à camada de rede

B O que tem dentro de um roteador?

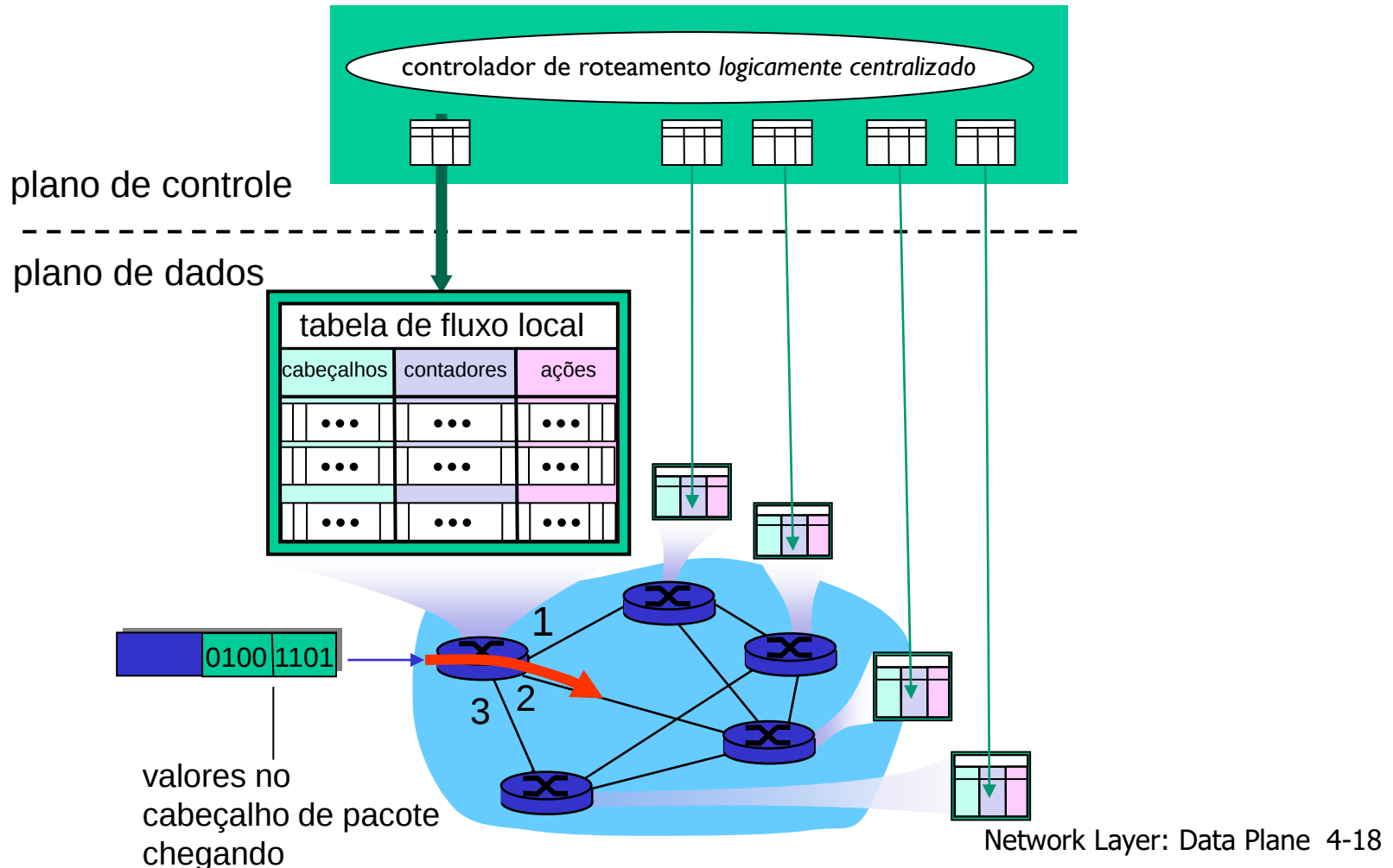
C Atrasos e vazão

D Endereçamento IP: Sub-redes e endereços;
obtendo um endereço IP; esgotamento do IPv4

E Repasse generalizado e SDN

Repasse generalizado e SDN

Cada roteador contém uma *tabela de fluxo (flow table)* que é computada e distribuída por um controlador de roteamento *logicamente centralizado*



Abstração do plano de dados Open Networking Foundation (ONF)

Members

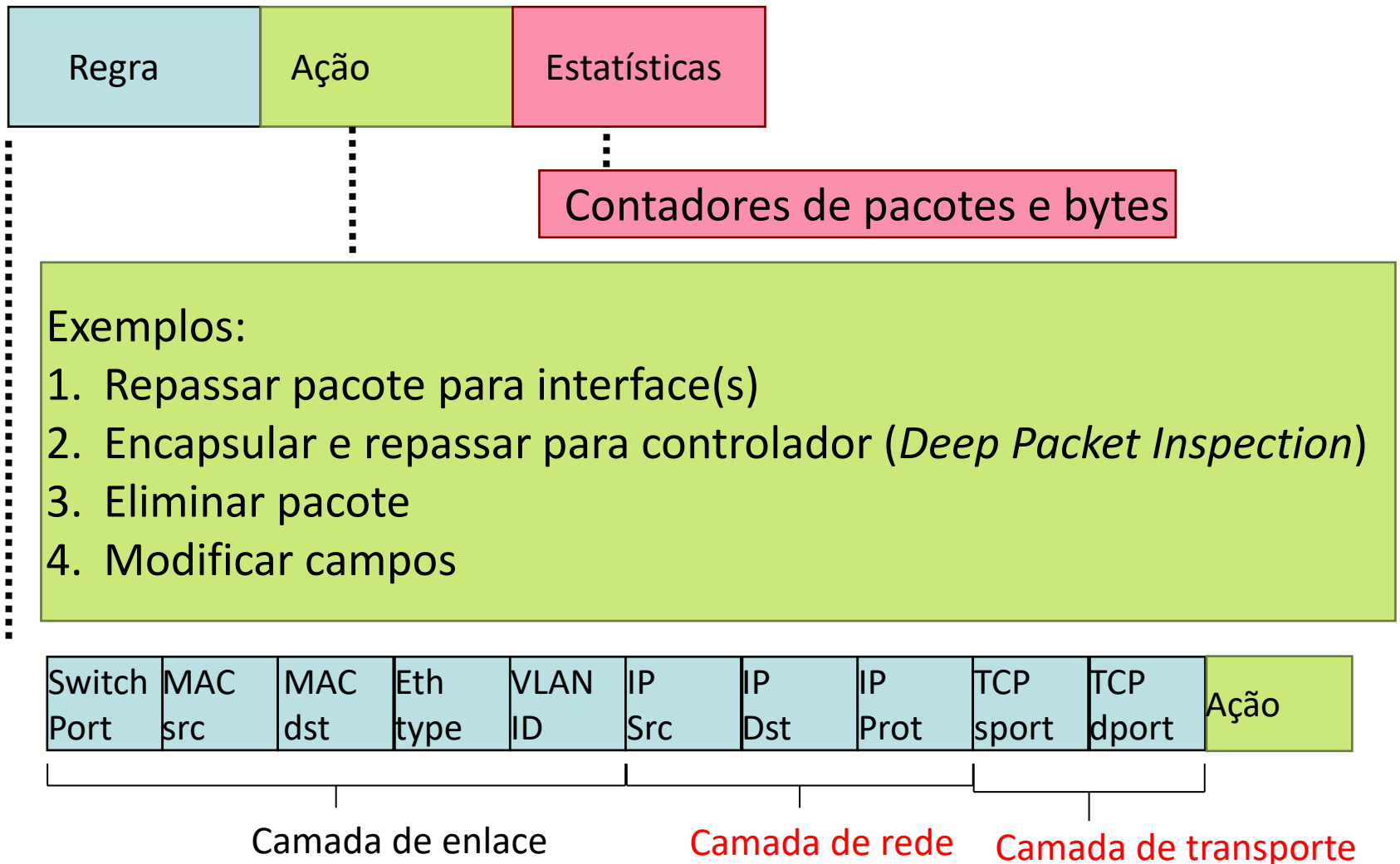


Abstração do plano de dados - ONF

- **Ação:** definida por diversos parâmetros e não apenas pelo IP destino; cabeçalhos de todas as camadas podem definir ação sobre dado pacote
- **Repasse generalizado:** regras simples de manipulação de pacotes – como comutador atua é *definido por software* (SDN)
- Tabela de fluxo em um roteador (**comutador de pacotes**) é computada e distribuída pelo controlador e define regras para casamento+ação do roteador
- Campos da tabela:
 - **Regra:** **casar** valores nos campos de cabeçalho do pacote
 - **Ações:** **para pacote casado:** eliminar, repassar, modificar, enviar pacote para controlador, etc.
 - **Contadores:** #bytes e # pacotes

OpenFlow 1.0: Entradas da Tabela de fluxo

■ Linha *simplificada* da tabela



Abstração ONF

- *Casamento+ação*: unifica diferentes tipos de dispositivos
- Roteador
 - *Casamento*: prefixo IP de destino mais longo
 - *Ação*: repassa para um enlace de saída
- Switch
 - *Casamento*: endereço MAC destino
 - *Ação*: repassa ou inunda (*broadcast*)
- Firewall
 - *Casamento*: endereços IP ou números de porta TCP/UDP
 - *Ação*: repassa ou elimina
- NAT
 - *Casamento*: número IP e porta
 - *Ação*: reescreve endereço e porta

Software livre (ONOS): [aqui](#)

Exemplo CISCO: [aqui](#)

Exemplos

Repasse baseado no destino (roteador tradicional) :

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Ação
*	*	*	*	*	*	51.6.0.8	*	*	*	porta 6

datagramas IP destinados ao endereço IP 51.6.0.8 devem ser repassados para a interface de saída 6 do comutador

Firewall:

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Ação
*	*	*	*	*	*	*	*	*	81	drop

não repassar pacotes (bloquear) todos datagramas destinados à porta TCP 81

Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Ação
*	*	*	*	*	128.119.1.1	*	*	*	*	drop

não repassar (bloquear) todos os datagramas enviados pelo host 128.119.1.1

Exemplos

Repassa baseado na camada 2 (switch):

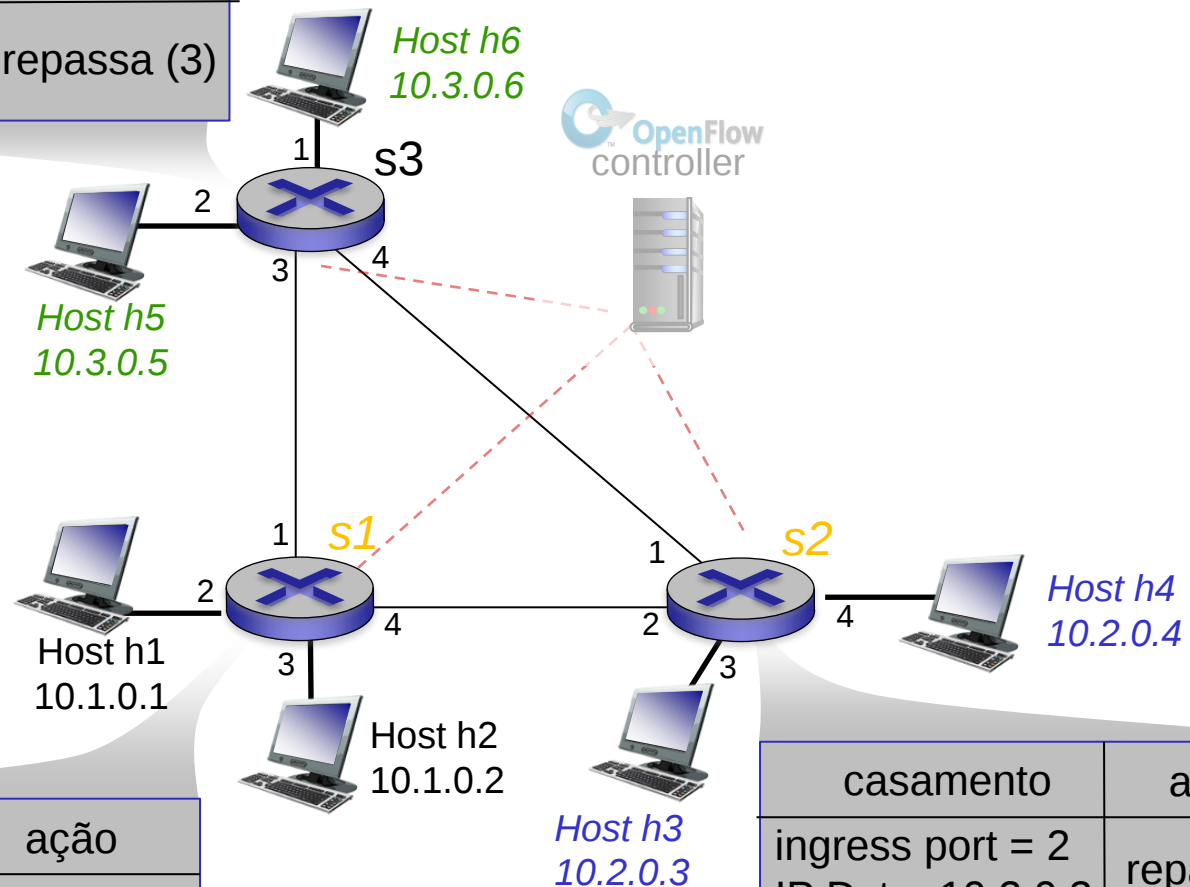
Switch Port	MAC src	MAC dst	Eth type	VLAN ID	IP Src	IP Dst	IP Prot	TCP sport	TCP dport	Ação
*	22:A7:23: 11:E1:02	*	*	*	*	*	*	*	*	porta 6

quadros de camada 2 do endereço MAC 22:A7:23:11:E1:02 devem ser repassados para interface de saída 6

Exemplo ONF

Exemplo: datagramas dos hosts *h5* e *h6* devem ser enviados para *h3* ou *h4*, via *s1* e de lá para *s2*

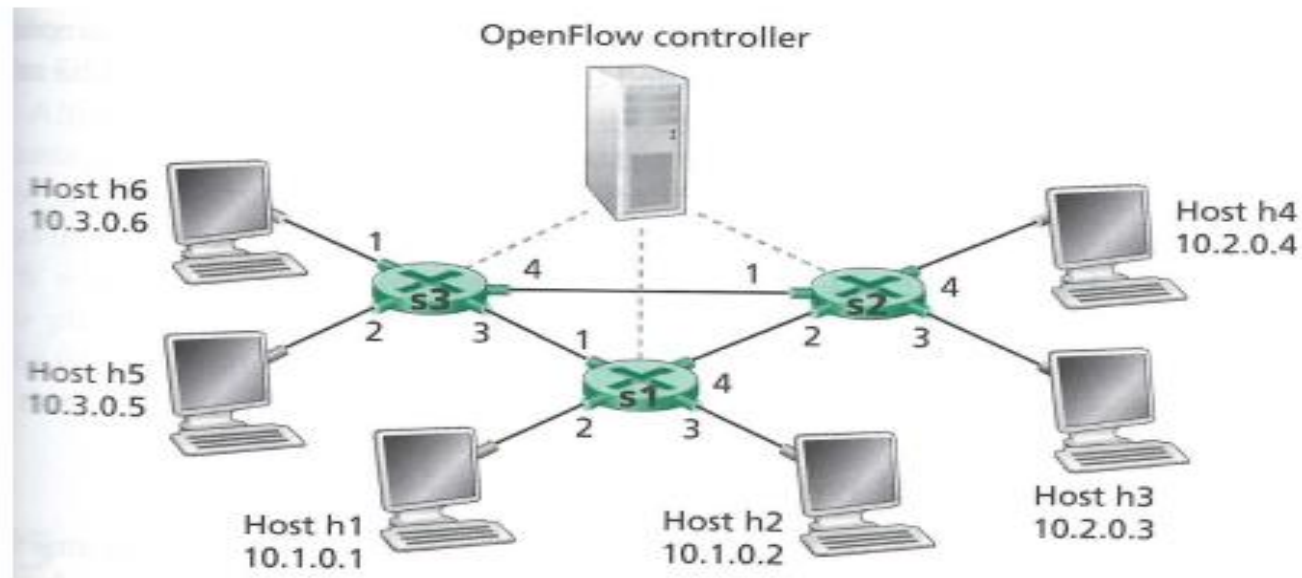
casamento	ação
IP Src = 10.3.*.* IP Dst = 10.2.*.*	repassa (3)



casamento	ação
ingress port = 1 IP Src = 10.3.*.* IP Dst = 10.2.*.*	repassa (4)

casamento	ação
ingress port = 2 IP Dst = 10.2.0.3	repassa (3)
ingress port = 2 IP Dst = 10.2.0.4	repassa (4)

- 5) (2,0 pontos) [Kurose and Ross, 2017, p. 397] Considere a rede SDN *OpenFlow* mostrada na figura a seguir.



Suponha que o comportamento de repasse para datagramas chegando dos *hosts* h_3 ou h_4 em s_2 sejam os seguintes:

- qualquer datagrama chegando do *hosts* h_3 e destinado a h_1 , h_2 , h_5 ou h_6 deve ser repassado em sentido horário na rede;
- qualquer datagrama chegando do *hosts* h_4 e destinado a h_1 , h_2 , h_5 ou h_6 deve ser repassado em sentido anti-horário na rede.

Especifique as entradas da tabela de fluxo de s_2 que implementam esse comportamento de repasse.

Tabela de Fluxo de s_2	
Casamento	Ação