

Curso de Redes de Computadores

Adriano Mauro Cansian
adriano@acmesecurity.org

Capítulo 5

Camada de Enlace de Dados

1

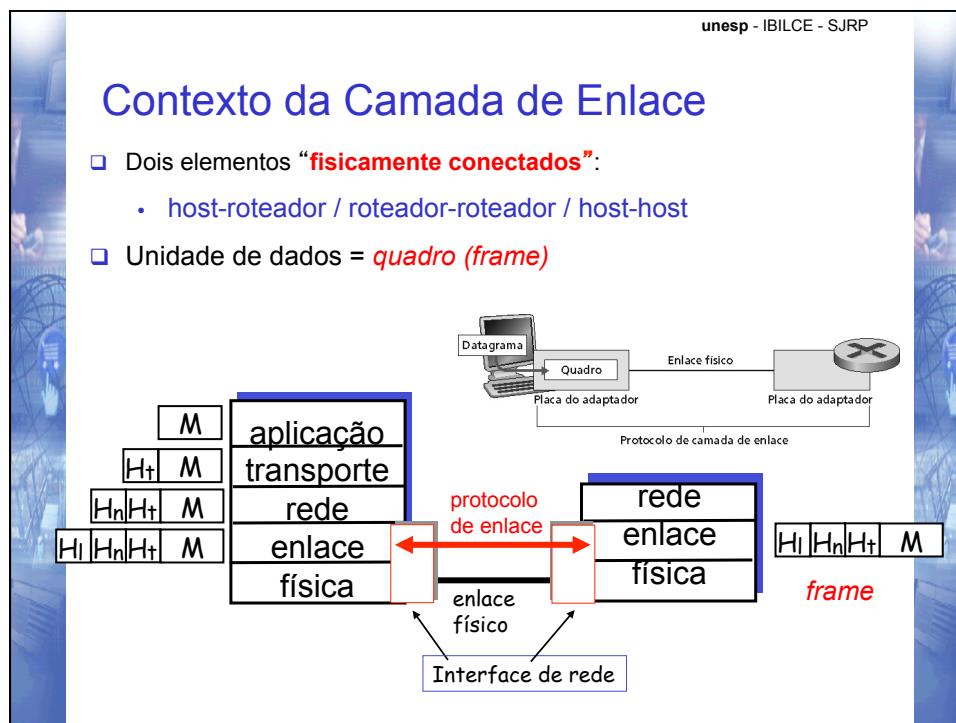
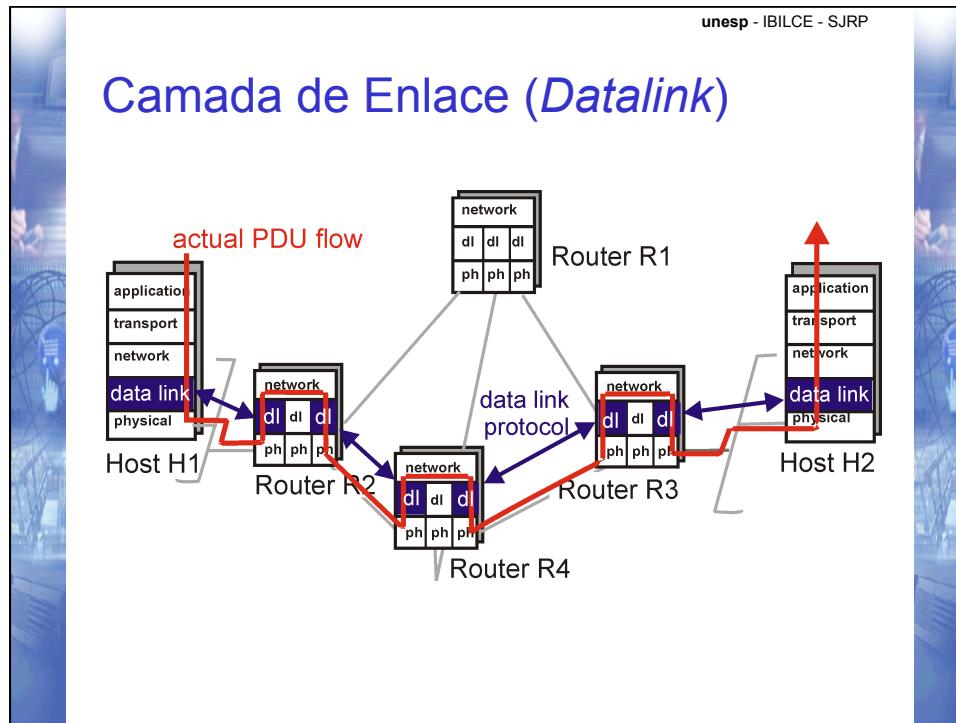
A Camada de Enlace

Objetivos:

- ❑ Entender os serviços da camada de enlace:
 - Compartilhando um canal broadcast: acesso múltiplo;
 - Disputa para acesso ao meio.
 - Endereçamento da camada de enlace;
- ❑ Implementação de tecnologias da camada de enlace.

Visão Geral:

- ❑ Funcionamento da camada de enlace.
- ❑ Protocolos de acesso múltiplo e LANs
- ❑ Endereçamento da camada de enlace e ARP.
- ❑ Tecnologias específicas da camada de enlace:
 - Ethernet
 - Switches



Serviços da Camada de Enlace (1)

- 1. Enquadramento e acesso ao enlace:**
 - Encapsula datagrama num *frame*.
 - Inclui cabeçalho e cauda (“*header*” e “*trailer*”).
 - Implementa **acesso ao canal** se este for compartilhado.
 - “Endereços físicos” ou “endereços MAC” → nos cabeçalhos para identificar **origem e destino**.
- 2. Entrega confiável:**
 - **Pouco usada nos meios de redes locais atuais** em fibra óptica, cabo coaxial e alguns tipos de pares trançados devido a taxas de erro de bit muito baixas.
 - **Usada em enlaces de rádio**, onde a meta é reduzir erros assim evitando a retransmissão fim a fim.

Serviços da Camada de Enlace (2)

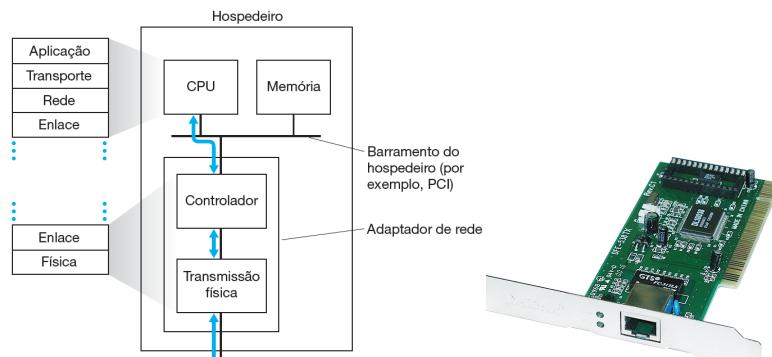
- 3. Controle de Fluxo:**
 - Compatibilizar taxas de produção e consumo de quadros entre remetentes e receptores.
- 4. Detecção e Correção de Erros:**
 - Erros são causados por atenuação do sinal e por ruído.
 - Receptor detecta presença de erros.
 - Receptor sinaliza ao remetente para retransmissão, ou simplesmente descarta o quadro em erro.
 - Há mecanismos que permitem que o receptor localize e corrija o erro sem precisar da retransmissão.
 - Em alguns casos.

Principalmente em *Wireless*

unesp - IBILCE - SJRP

Implementação do Protocolo de Enlace (1)

- ❑ Protocolo da camada de enlace é implementado **totalmente na interface** (na placa de rede).
- ❑ “*Software in hardware*”.



unesp - IBILCE - SJRP

Implementação do Protocolo de Enlace (2)

- ❑ Operações de “**envio**” no adaptador:
 - **Encapsula.**
 - Insere **header** com endereços, número de sequência, informações de realimentação e outros controles.
 - Inclui bits de detecção de erros.
 - Implementa **controle de acesso ao canal**.
 - Realiza a geração e **transmissão** do sinal.

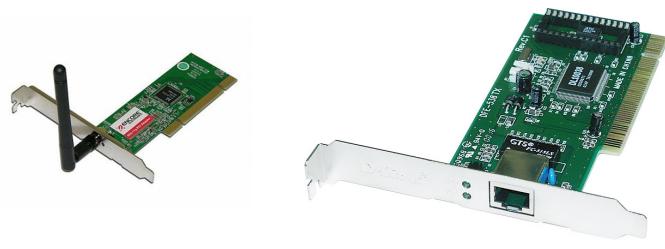


unesp - IBILCE - SJRP

Implementação do Protocolo de Enlace (3)

❑ Operações “**recebe**” do adaptador:

- Verificação e correção de **erros**.
- Interrupção do host.
 - Para enviar o *frame* para a camada superior.
- Atualiza informações de estado.



unesp - IBILCE - SJRP

Detecção e correção de erros

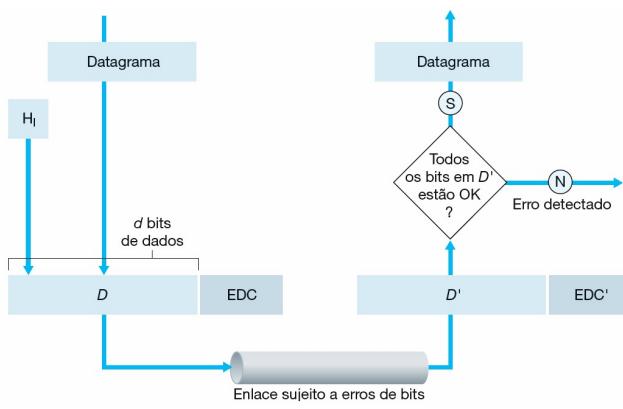
Detecção de Erros - métodos

- ❑ Bits de paridade.
 - Técnica Unidimensional.
 - Detecta erros em um único bit.
 - Técnica Bidimensional.
 - Detecta e **corrigir** em um único bit.
- ❑ Métodos de “Checksum”.
 - Códigos de Redundância Cílica
 - CRC - *Cyclic Redundancy Codes*.



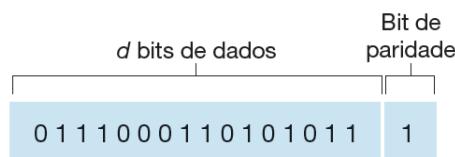
Detecção de Erros

- ❑ D = Dados protegidos por verificação de erros.
- ❑ EDC = bits de Detecção e Correção de Erros (redundância)
- ❑ Um campo maior de EDC permite melhorar detecção e correção.



Detecção de Erros

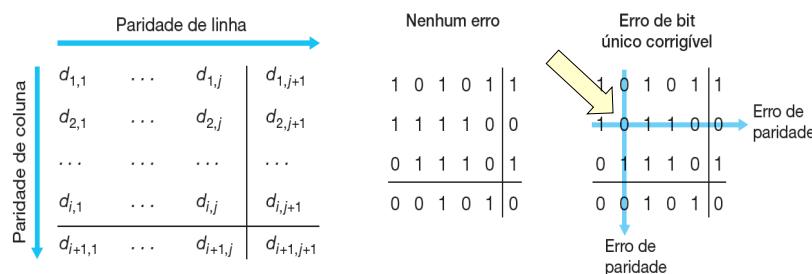
- O desafio do receptor é determinar se D' é ou não igual ao D original, uma vez que recebeu apenas D' e EDC' .



Paridade de 1 Bit:
Deteta erros em um único bit

Verificação de paridade

- Uma das maneiras mais simples de detectar erros é utilizar um único bit de paridade.
- A figura abaixo mostra uma generalização bidimensional do esquema de paridade de bit único.

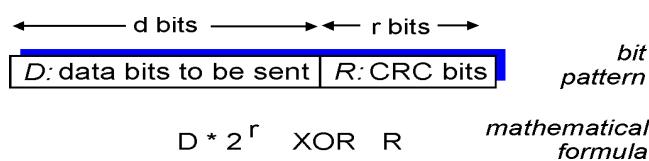


Cálculo do da soma de verificação

- ❑ Um método simples de soma de verificação é **somar os inteiros de k bits e usar o total resultante como bits de detecção de erros.**
- ❑ O complemento de 1 dessa soma é a soma de verificação que é carregada no cabeçalho do segmento.
 - Já visto anteriormente no TCP, UDP e IP.
- ❑ Lembrando: no IP, a **soma de verificação é calculada sobre o cabeçalho IP.**
- ❑ Métodos de soma de verificação exigem relativamente pouca sobrecarga no pacote.

Métodos de “Checksum”

- ❑ **Checksum “da Internet”:**
 - Emissor considera dados como compostos de inteiros de 16 bits;
 - Soma todos os campos de 16 bits (usando aritmética de complemento de um) e acrescenta a soma ao pacote;
 - O receptor repete a mesma operação e compara o resultado com o *checksum* enviado com o quadro.

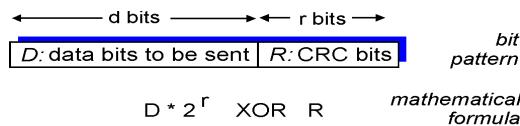


Métodos de “Checksum”

Códigos de Redundância Cíclica

(*Cyclic Redundancy Codes*):

- Dados considerados como a sequência de coeficientes de um polinômio D .
- É escolhido um polinômio Gerador G ($\Rightarrow r+1$ bits)
- Divide-se (módulo 2) o polinômio $D * 2^r$ por G .
- Acrescenta-se o resto R a D .
- Observa-se que, por construção, a nova sequência $\langle D, R \rangle$ agora é exatamente divisível por G .



Implementação de CRC

- EMISSOR** realiza em tempo real, por hardware, a divisão da sequência D pelo polinômio G , e acrescenta o resto R a D .

- O **RECEPTOR** divide $\langle D, R \rangle$ por G :
- Se o resto for diferente de zero, a transmissão teve erro.
- Padrões internacionais de polinômios G de graus 8, 12, 15 e 32 já foram definidos.

$$CRC_{32}(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x^1 + x^0$$

XOR

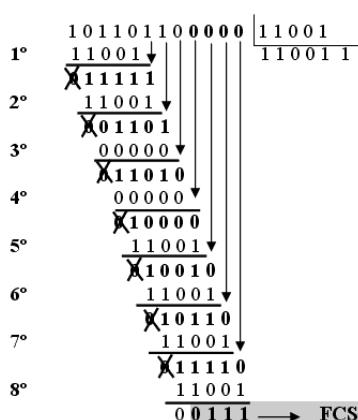
Modulo-2 Calculation

$$\begin{array}{r}
 1001100101 \\
 \text{XOR } 0100110111 \\
 = 1101010010
 \end{array}$$

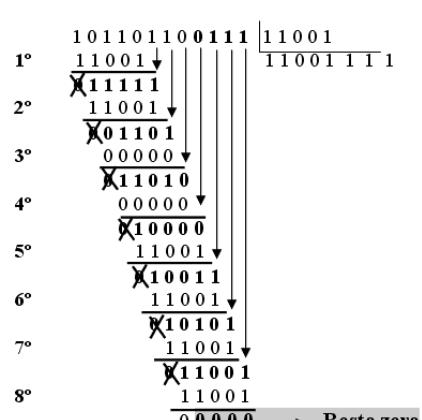
XOR-Function		X1	X2	Y
0	0	0		0
0	1	1		1
1	0	1		1
1	1	0		0

Exemplo de CRC

Na transmissão



Na Recepção



Dados a enviar 101101100111

Métodos de *CRC* em Ethernet (1)

- Detecção de Erros
 - Acrescenta ao quadro um CRC 32 bits
 - Valor escolhido tal que os bits do quadro (exceto o preâmbulo) sejam divisíveis (bit a bit, usando álgebra XOR) por um código gerador
 - Código gerador de 32 bits (padrão IEEE):
100000100110000010001110110110111
 - Detecta qualquer número ímpar de erros de bits, erros de rajada de até 33 bits, e 99,999999767% das rajadas acima disto

Métodos de *CRC* em Ethernet (2)

- Detecção de Erros – Exemplo:
 - Cálculo de CRC de 4 bits:

Dados:	1101011011
Dividendo:	1101011011 0000
Código Gerador:	10011
Quociente:	1100001010
Resto (CRC):	1110
Dados + CRC:	1101011011 1110
	[D]
	[G]
	[D/G]
	[R]
	[D+R]
 - Um quadro recebido será descartado caso não seja divisível pelo código gerador

CRC – Exemplos para estudo e exercícios

❑ Como exemplo, assista aos vídeos:

- Cálculo de CRC – parte 1 e 2 – Prof. Othon Batista
 - <http://youtu.be/XWcJcybL3JQ> (Verif. 09/6/2015)
 - <http://youtu.be/wyUNSzDbFjg> (Verif. 09/6/2015)

❑ Cálculo detalhado:

- <http://www.emcu.it/CRC/CRCuk.html> (Verif. 09/6/2015)

❑ Exercícios: como funcionam os códigos de verificação de erros

- (Kurose & Ross 5^a. Ed. Tópico 5.2).

Enlaces e Protocolos de Múltiplo Acesso

Enlaces e Protocolos de Múltiplo Acesso

❑ Três tipos de enlace:

- Ponto-a-ponto.
 - Cabo único: ADSL, USB, etc...
- **Difusão ou Broadcast.**
 - Cabo ou meio compartilhado: Ethernet, rádio, etc...
- Comutado.
 - Switchs, ATM, etc..

Começaremos a estudar
enlaces com **difusão (broadcast)**

Assim, veremos os protocolo de Múltiplo Acesso

Protocolos de Múltiplo Acesso (1)

❑ Canal de comunicação:

- Único e **compartilhado**.

❑ Se duas ou mais transmissões acontecem ao mesmo tempo:

- Resulta em interferência do sinal transmitido.
- **Chama-se “colisão”.**

Apenas um nó pode transmitir **com sucesso**
num dado instante.

Protocolos de Múltiplo Acesso (2)

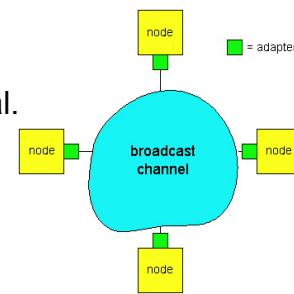
❑ Protocolo de múltiplo acesso:

- Possuem um algoritmo distribuído.
 - Define como as estações compartilham o canal.
 - Define quando cada estação pode transmitir.
- A comunicação sobre o compartilhamento do canal deve seguir pelo próprio canal.

Reflexão: humanos usam protocolos de acesso múltiplo o tempo todo!

Protocolos de Controle de Acesso ao Meio (MAC – Media Access Control)

- ❑ Protocolo MAC: deve coordenar transmissões de hosts diferentes para **minimizar ou evitar colisões**.
- ❑ Diferentes tipos de protocolos de múltiplo acesso.
- ❑ São divididos em 3 classes:
 - Como protocolos de:
 - Particionamento do Canal.
 - Acesso Randômico.
 - Revezamento.



Protocolos MAC: taxonomia

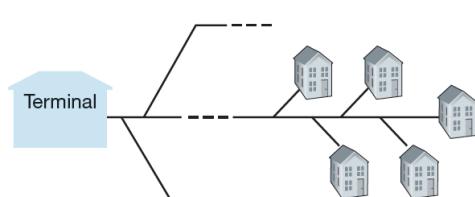
Três grandes classes:

- ❑ **Particionamento de canal** → divisão do canal igual entre os nós.
 - Dividem o canal em pedaços menores
 - Compartimentos de tempo ou de frequência.
 - Aloca um pedaço de uso exclusivo para cada nó (*host*).
 - Já bastante discutido no curso de redes: não será revisto.
- ❑ **Acesso Aleatório** → cada nó transmite quando tiver dados.
 - Pode causar **colisões**: dois ou mais enviem ao mesmo tempo.
 - Deve permitir “recuperação” das colisões.
- ❑ **Revezamento** → cada nó transmite na sua vez.
 - Compartilhamento estritamente coordenado.
 - Passagem de permissão evita colisões.

Objetivo a ser alcançado:
eficiente, **justo**, **simples** e **descentralizado**.

Exemplos de acesso múltiplo

Compartilhado com fio
(por exemplo, rede de acesso a cabo)



Compartilhado sem fio
(por exemplo, Wi-Fi)



Particionamento do canal

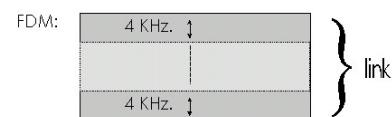
FDMA

TDMA

CDMA

Protocolos de Particionamento do Canal

- ❑ **TDMA** (Multiplexação por Divisão de Tempo): canal dividido em N intervalos de tempo (“slots”), um para cada usuário.
 - Ineficiente com usuários de pouco demanda ou quando carga for baixa.
- ❑ **FDMA** (Multiplexação por Divisão de Frequência): frequência subdividida em pedaços menores.



TDM:



All slots labelled **2** are dedicated to a specific sender-receiver pair.

Já discutidos nos capítulos anteriores. Os alunos devem aproveitar para fazer uma boa revisão deste tópico

Protocolos de Acesso Randômico

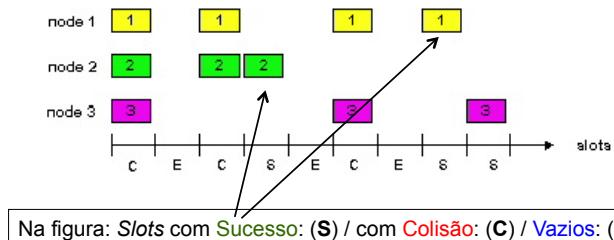
1. **SLOTTED ALOHA**
2. **ALOHA**
3. **CSMA**
4. **CSMA/CD**

Protocolos de Acesso Randômico

- ❑ Ideia básica → a estação transmite **aleatoriamente**.
 - **Se houver dados a transmitir, simplesmente transmite.**
 - Sem coordenação entre estações.
 - Ocupam **toda** a banda R do canal.
 - Se houver “**colisão**” entre as transmissões de duas ou mais estações, elas retransmitem depois de esperar randômica.
- ❑ O protocolo **MAC de acesso randômico** especifica como detectar colisões, e como se recuperar delas.
- ❑ Protocolos MAC de acesso randômico:
 - (a) **SLOTTED ALOHA**
 - (b) **ALOHA**
 - (c) **CSMA e CSMA/CD**

Slotted Aloha

- ❑ O tempo é dividido em *slots* de tamanho igual.
 - Exemplo: o *slot* é igual ao tempo para enviar o tamanho de um pacote cheio.
- ❑ Estação que tem dados a enviar transmite no início do próximo *slot*.
- ❑ Se houver uma **colisão**, a origem retransmite o pacote a cada *slot* com probabilidade P , até conseguir sucesso.
- ❑ S-ALOHA é eficiente na utilização do canal.
 - É completamente descentralizado.



Eficiência do Slotted Aloha

- ❑ Se N estações tem pacotes para enviar,
 - e cada uma transmite em cada *slot* com probabilidade p , então
 - A probabilidade S de uma transmissão com sucesso é:
Para uma estação específica: $S = p (1-p)^{N-1}$
- ❑ Para que qualquer uma das N estações consiga transmitir com sucesso num *slot*:

$$S = N p (1-p)^{N-1}$$

Otimizando, obtemos que o valor ótimo de P é $P = 1/N$
Para N muito grande temos $S = 1/e$ (aproximadamente 0,37).

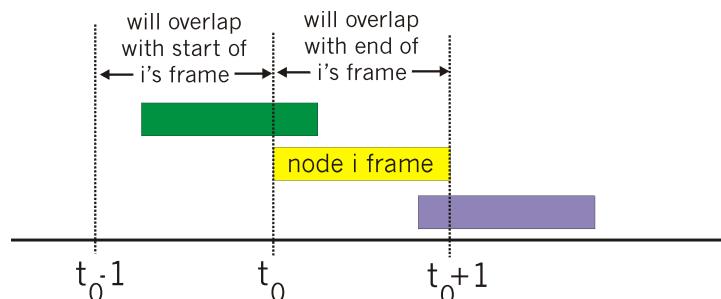
Ponto de máximo: uso do canal para envio de dados úteis: **37% do tempo!**

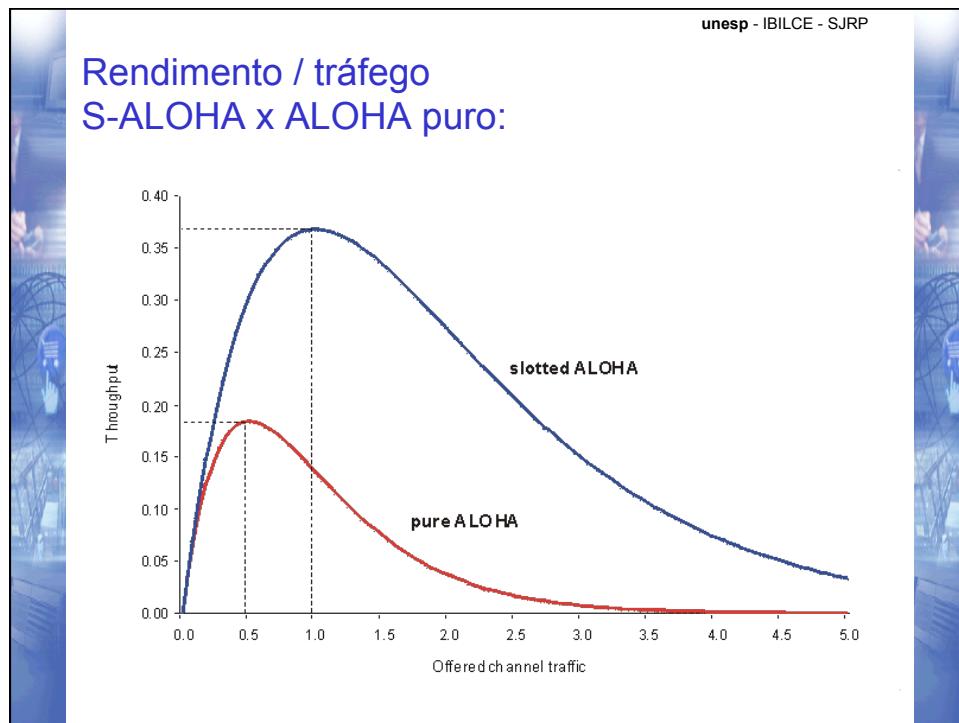
ALOHA puro: sem slots (1)

- ❑ *Slotted ALOHA* → requer sincronização dos *slots*.
- ❑ Um versão mais simples: ALOHA puro
 - Não exige *slots*.
 - Uma estação transmite sem aguardar o início de um *slot*.
 - A probabilidade de **colisão aumenta**.
 - Fica duas vezes o tamanho de *Slotted-Aloha*.
 - Pacote pode colidir com outros pacotes transmitidos dentro de uma janela
- ❑ Repetindo os cálculos e considerações, obtemos que a vazão é reduzida pela metade: **$S = 1/(2e) \approx 0,18$** .

Ponto de máximo, para envio de dados úteis: 18% do tempo!

ALOHA puro: sem slots (1)





unesp - IBILCE - SJRP

CSMA (*Carrier Sense Multiple Access*)

- ❑ **CSMA: escuta antes de transmitir.**
 - Se detecta que o canal está sendo usado, adia transmissão.
 - **CSMA persistente:** tenta novamente assim que o canal se tornar ocioso.
 - **CSMA não-persistente:** tenta novamente, depois de intervalo randômico.
- ❑ **Colisões ainda podem ocorrer:** duas (ou mais) estações podem detectar o canal ocioso ao mesmo tempo.
- ❑ **Janela de vulnerabilidade** → devido ao retardo ida e volta entre as estações envolvidas.
 - No caso de colisão, é desperdiçado todo o tempo de transmissão do pacote.

unesp - IBILCE - SJRP

Colisões



unesp - IBILCE - SJRP

Colisões em CSMA

Colisões podem ocorrer:
o atraso de propagação
implica que dois nós podem
não ouvir as transmissões do
outro.

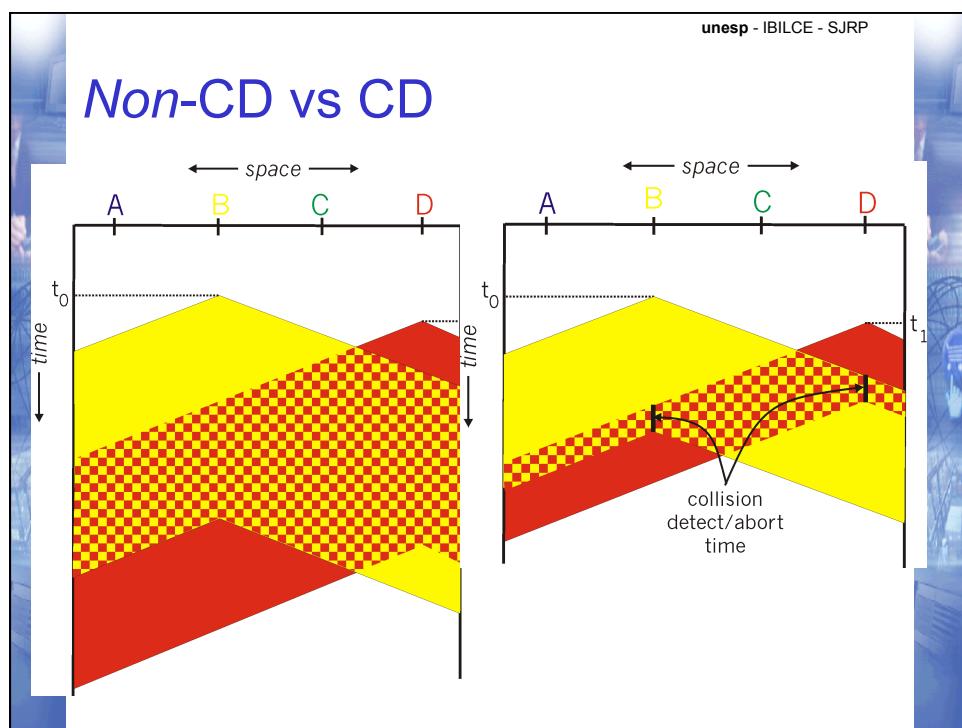
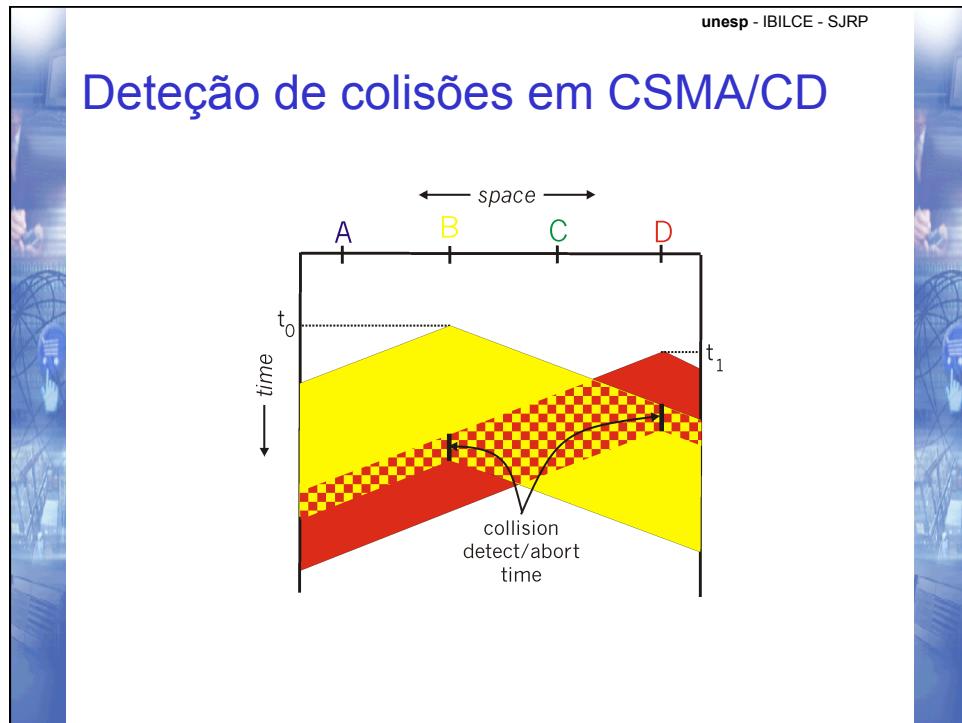
Colisão:
todo o tempo de transmissão
do pacote é desperdiçado.

Lembre:
O papel da **distância** e do
atraso de propagação na
determinação da possibilidade
de colisão.

unesp - IBILCE - SJRP

CSMA/CD (Detecção de Colisões)

- ❑ **CSMA/CD** → escuta do meio e adiamento.
 - Entretanto, colisões detectadas rapidamente, em poucos “intervalos de bit”.
- ❑ Transmissão é então abortada, reduzindo consideravelmente o desperdício do canal.
- ❑ Detecção de colisões é **fácil em rede locais usando cabo**
 - Medir a intensidade do sinal na linha, ou comparar sinais Tx e Rx.
- ❑ CSMA/CD pode conseguir utilização do canal perto de 100% em redes locais.



unesp - IBILCE - SJRP

Algoritmo CSMA/CD (1)

```
A: escuta canal; SE ocioso
ENTÃO {
    transmite e monitora o canal;
    se detectou outra transmissão
        então {
            aborta e envia sinal de "jam";
            atualiza número de colisões;
            retarda de acordo com o algoritmo de retardo
                exponencial (ou backoff exponencial);
            vai para A
        }
    senão {terminado este quadro; zera número de
        colisões}
}
senão {espera o final da transmissão atual e vai
    para A}
```

unesp - IBILCE - SJRP

Algoritmo CSMA/CD (2)

□ Sinal “Jam” :

- para garantir que todos os outros transmissores tomem conhecimento da colisão.
- Sinal possui 48 bits;

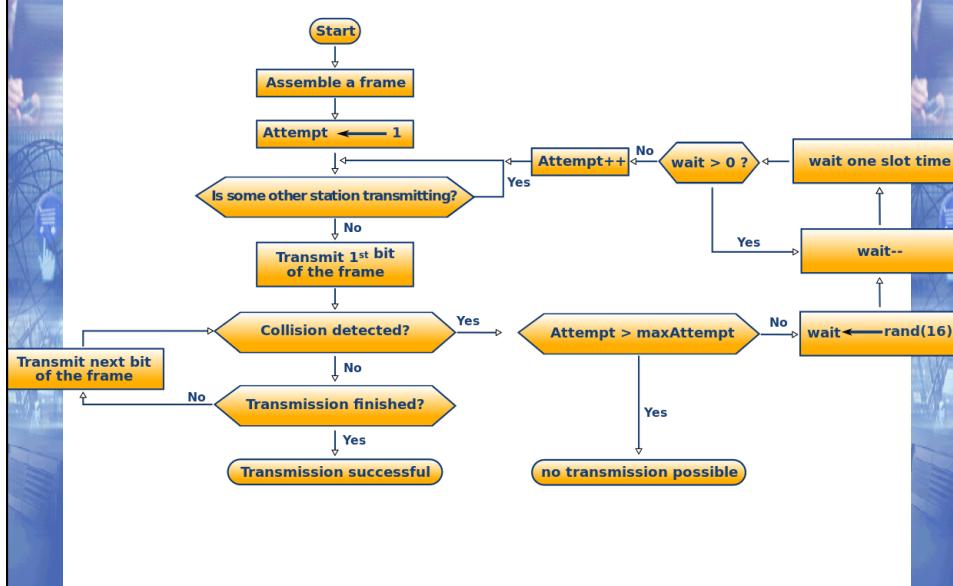
□ Backoff Exponencial:

- Meta é adaptar a taxa oferecida por transmissores à estimativa da carga atual
 - Isto é, retardar quando carga da rede estiver elevada.

Backoff exponencial

- ❑ Após sofrer a *n*-ésima colisão em seguida para um quadro:
- ❑ Escolhe um valor de K aleatoriamente de $\{0, 1, 2, \dots, 2^m-1\}$ onde $m = \min(n, 10)$
 - Depois da **primeira** colisão, escolhe K entre $\{0, 1\}$.
 - **O retardo é ($K \times 512$ BTT)**
 - [1 BTT = tempo para transmitir 1 bit]
 - Depois da segunda colisão escolhe K de $\{0, 1, 2, 3\} \dots$
 - Depois de dez ou mais colisões, escolhe K de $\{0, 1, 2, 3, 4, \dots, 1023\}$
- ❑ **Não mantém estado.**
- ❑ Cada pacote que colide é tratado de forma independente.

CSMA/CD simplificado



Eficiência do CSMA/CD

- ❑ Nota-se que neste esquema um novo quadro tem **uma chance de sucesso na primeira tentativa, mesmo com tráfego pesado.**
- ❑ **Eficiência Ethernet** com tráfego pesado e número grande de nós:

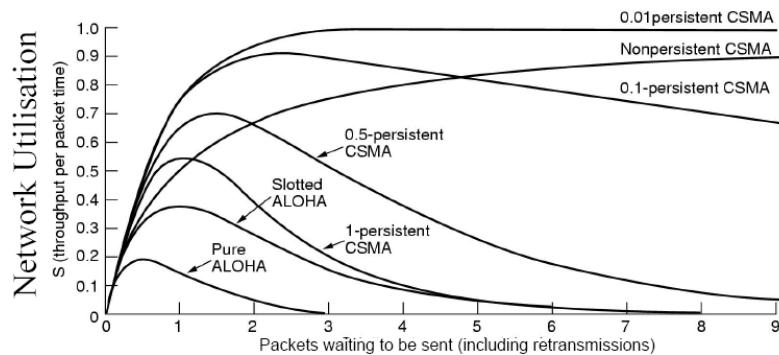
- Eficiência tende a 1 quando t_{prop} tende a 0.
- Tende a 1 quando t_{trans} tende ao infinito.
- Muito melhor do que o ALOHA, e ainda é descentralizado, simples e barato.

$$\text{Efficiency} = \frac{1}{1 + (5 * \frac{t_{prop}}{t_{trans}})}$$

Exercício: determinar o rendimento % do CSMA/CD em uma situação de alta carga de tráfego na rede.

Ver Livro Tannenbaum – Redes de Computadores.

Eficiência dos protocolos de acesso rândomico



unesp - IBILCE - SJRP

Pergunta / exercício: **Quanto tempo é necessário para se perceber uma colisão?** (Isto é, qual o tempo do slot de contenção?)

Determinação do slot de contenção

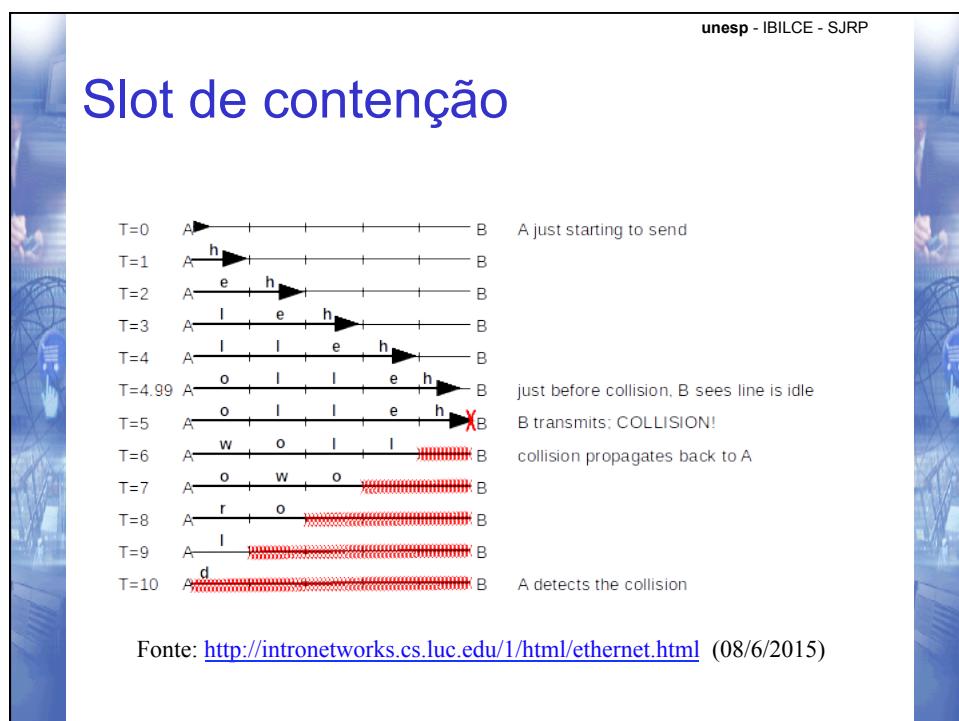
Tempo do percurso $A \rightarrow B = T$

Pior caso:

- (a): Estação **A** começa a transmitir no instante $t = 0$.
- (b) e (c): Num pequeno instante antes de o sinal chegar à estação mais distante, no tempo $t = (T - \varepsilon)$ temos que **B** começa a transmitir e percebe a colisão, quase instantaneamente. Então, interrompe a transmissão.
- (d) Mas o pequeno efeito da colisão não chega até a estação original **A** num tempo melhor que $2T - \varepsilon$ (tempo de ida e volta de A até B).

Em outras palavras: na pior das hipóteses, a estação **A** só poderá ter certeza de haver se apoderado do canal após transmitir durante $2T$ sem escutar uma colisão.

Portanto **$2T$ é o tempo necessário para que a estação **A** esteja segura que assumiu o controle**. Num cabo de 1 Km, $T = 5$ microsegundos.



Protocolos MAC de “revezamento”

❑ Até aqui já vimos:

- Protocolos MAC de **particionamento ESTÁTICO de canal (TDM, FDM)**.
 - Podem **compartilhar equitativamente o canal**;
 - Porém, uma única estação não consegue usar toda a capacidade do canal.
- Protocolos MAC de **acesso randômico** permitem que um único usuário utilize toda a capacidade do canal;
 - Entretanto, eles não conseguem compartilhar o canal de **maneira equalitária**.

❑ Mas existem também os protocolos de **revezamento**.

Protocolos MAC de “revezamento”

❑ Protocolos MAC de **revezamento** conseguem tanto justeza como acesso individual a toda a capacidade do enlace.

- Custo: maior complexidade de controle.

❑ Dois grandes métodos:

- **Polling**: uma estação Mestre numa rede local “convida” em ordem as estações escravas a transmitir seus pacotes.
- **Token (ficha) de permissão (ou “bastão”)** é passada seqüencialmente de estação a estação.
 - É possível aliviar a latência e melhorar tolerância a falhas
 - *Token Bus* e *Token Ring*.

Exercício: Protocolo DOCSIS “3 em 1”

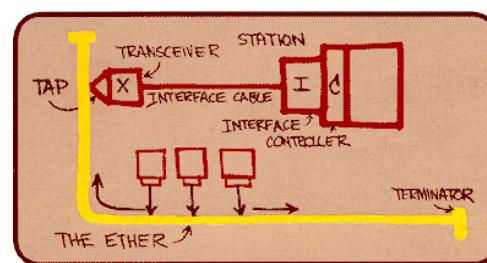
(Livro Texto K&R – 6ª edição - Tópico 5.3.4)

Ethernet e suas evoluções

Ethernet

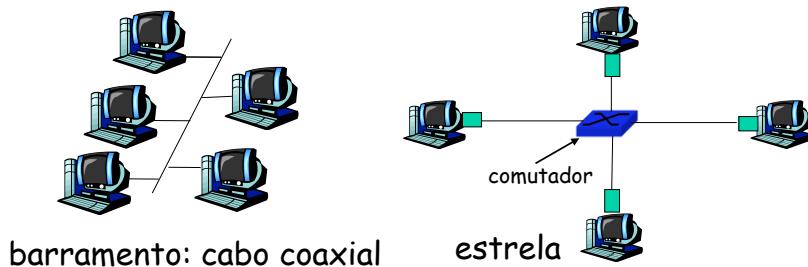
- ❑ Muitíssimo difundida porque:
 - Muito barata! < R\$ 20 para 100 Mbps
 - Uma das mais antigas tecnologias de rede local.
 - Mais simples e menos cara do que redes usando Token Ring ou ATM.
 - Acompanhou aumento de velocidade: 10, 100, 1000 Mbps.
 - Muitas tecnologias (cobre, fibra, etc), mas todas compartilham características comuns.

Bob Metcalfe &
David Boggs (1970)



Topologia de estrela

- ❑ Topologia de barramento popular até meados dos anos 90:
 - todos os nós no mesmo domínio de colisão (podem colidir uns com os outros).
- ❑ Atualmente: topologia de estrela prevalece:
 - **Comutador (switch)** ativo no centro.
 - Cada “ponta” roda um protocolo Ethernet (separado):
 - Nós não colidem uns com os outros (transmitem mutuamente).

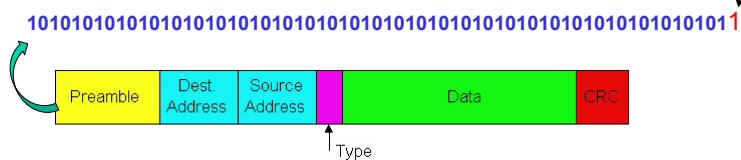


Ethernet: não confiável e sem conexão

- ❑ **Sem conexão:** sem apresentação entre origem e destino.
- ❑ **Não confiável:** nó de destino não envia confirmações ou não confirmações ao nó origem
 - Fluxo de datagramas passados à camada de rede pode ter lacunas (datagramas faltando)
 - Lacunas preenchidas se aplicação estiver usando TCP.
 - Caso contrário, aplicação verá lacunas.
- ❑ Protocolo MAC da Ethernet: **CSMA/CD**.

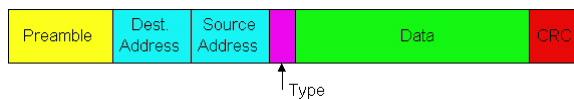
Estrutura de frame Ethernet (1)

- ❑ Interface do emissor encapsula datagrama IP (ou outro pacote da camada de rede) em um **frame Ethernet**.
 - Contém: **Preâmbulo**, **Cabeçalho**, **Dados** e **CRC**.
- ❑ **Preâmbulo de 8 bytes**:
 - 7 bytes com o padrão **10101010** seguidos por **um byte** com o padrão **10101011**.
 - Usado para sincronizar **receptor** ao *clock* do remetente.



Estrutura de frame Ethernet (2)

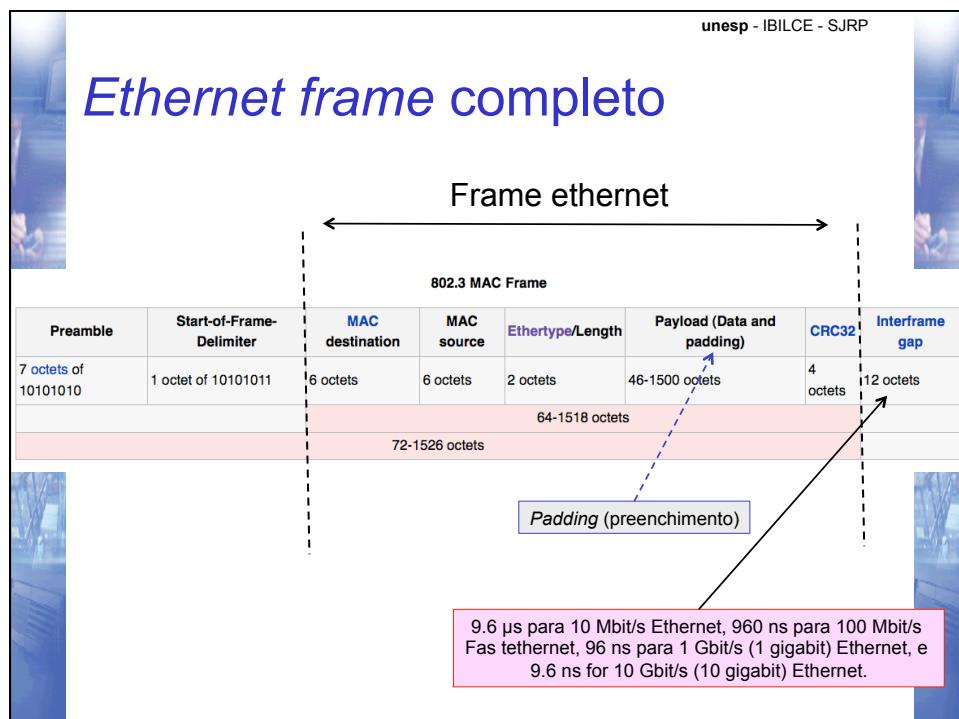
- ❑ O **Cabeçalho** contém Endereços de Destino e Origem, e um campo “Tipo” → indica a carga transportada.
- ❑ **Endereços: 6 bytes (48 bits)**
 - O frame é recebido por todas as interfaces numa LAN, e descartado se não casar o endereço de destino com o de quem recebe → lembre que é uma rede de difusão (*broadcast*).
- ❑ **Tipo**: indica o protocolo da camada superior,
 - Usualmente IP, mas existe suporte para outros
 - IPX da Novell, NETBIOS, AppleTalk, e outros.
- ❑ **CRC**: verificado pelo receptor: se for detectado um erro, o quadro será descartado



unesp - IBILCE - SJRP

Ethernet frame types

Value	Meaning
0000-05DC	Reserved for use with IEEE LLC/SNAP
0800	Internet IP Version 4
0805	CCITT X.25
0900	Ungermann-Bass Corporation network debugger
0BAD	Banyan Systems Corporation VINES
1000-100F	Berkeley UNIX Trailer encapsulation
6004	Digital Equipment Corporation LAT
6559	Frame Relay
8005	Hewlett Packard Corporation network probe
8008	AT&T Corporation
8014	Silicon Graphics Corporation network games
8035	Internet Reverse ARP
8038	Digital Equipment Corporation LANBridge
805C	Stanford University V Kernel
809B	Apple Computer Corporation AppleTalk
80C4-80C5	Banyan Systems Corporation
80D5	IBM Corporation SNA
80FF-8103	Wellfleet Communications
8137-8138	Novell Corporation IPX
818D	Motorola Corporation
FFFF	Reserved



Padding

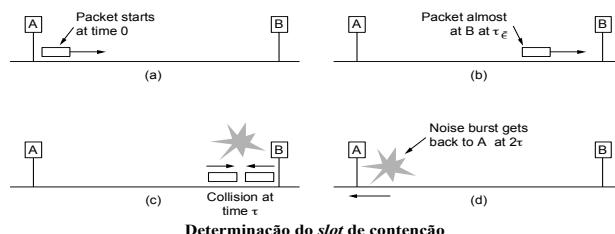
- **Padding** : preenchimento

Para evitar problemas e permitir a distinção de quadros válidos, o tamanho do *frame* que vai do **destination address** até o **checksum** deve ser maior ou igual a 64 bytes

→ se a **parte de dados (payload)** de um quadro válido for menor que **46 bytes**, o quadro será preenchido com 0's até o tamanho mínimo.

Razão mais importante → **tempo de propagação**: evitar que uma estação conclua a transmissão de um quadro curto antes de o primeiro bit ter atingido a extremidade do cabo.

Influencia diretamente no **tamanho máximo de cabo** que o protocolo pode usar.



A estação A só poderá ter certeza de haver se apoderado do canal após transmitir durante $2T$ sem escutar uma colisão.

Portanto $2T$ é o tempo necessário para que a estação A esteja segura que assumiu o controle. Num cabo de 1 Km, $T = 5$ microsegundos.

O *frame* deve ter um tamanho mínimo para garantir a detecção da colisão.

Assim a estação A (ou B) não pode interromper a transmissão antes do *frame* ter ido e voltado até a distância mais longa do cabo, caso contrário não há como detectar que o frame foi alterado.

LANs, endereçamento e ARP

LAN – Local Area Network

❑ **Ethernet:**

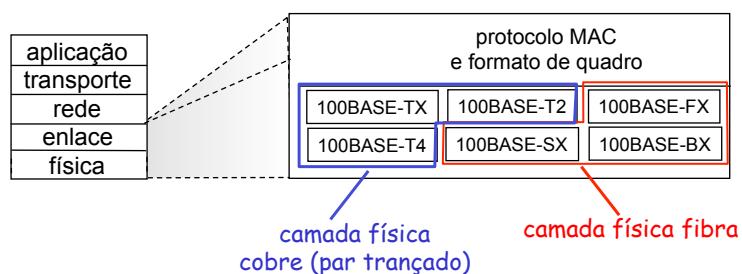
- É a tecnologia mais popular e barata de rede local.
 - (Apesar que o wifi é um grande concorrente)
- **Usa o protocolo CSMA/CD;**
- 10 Mbps (IEEE 802.3), Fast Ethernet (100 Mbps), Gigabit Ethernet (1000 Mbps);
- **Meio mais usado: Par trançado (TP – Twisted pair).**



Padrões Ethernet 802.3: camadas de enlace e física

muitos padrões Ethernet diferentes

- Protocolo MAC e formato de quadro comuns
- Diferentes velocidades: 2 Mbps, 10 Mbps, 100 Mbps, 1Gbps, 10G bps.
- Diferentes meios da camada física:
 - Fibra, cabo, compatibilidade com wi-fi



Endereços físicos e ARP.

Endereço IP:

- Usado para levar o pacote à **rede de destino**.

Endereço **físico** (ou endereço MAC).

- MAC Address = *Media Access Control* Address

Endereço MAC tem **48** bits:

- Formato: **00:26:08:01:82:9A**
- Representado em hexadecimal (6 octetos).
- Gravado na ROM da placa de rede.
 - Mas pode ser alterado por software.



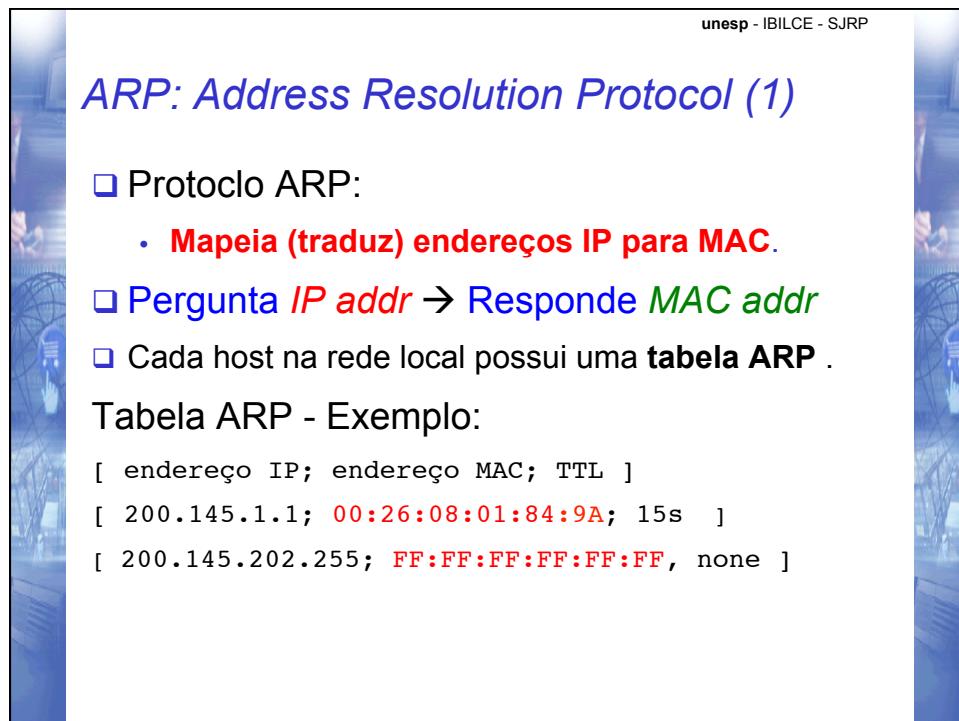
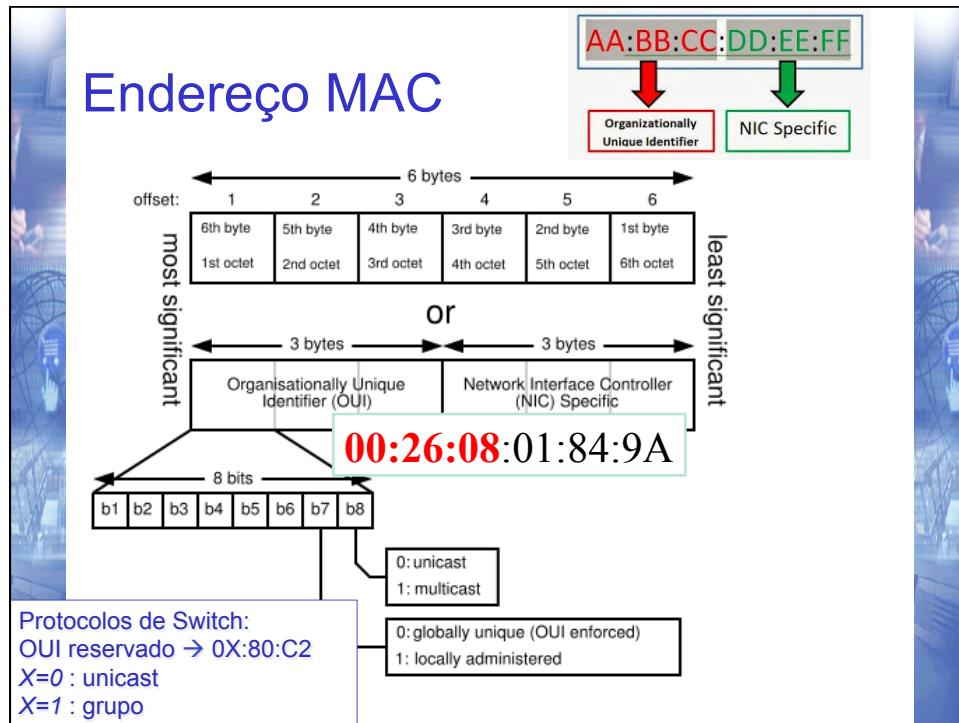
unesp - IBILCE - SJRP

Endereço MAC (endereço físico)

- Intervalo de endereços MAC é administrada pelo IEEE (www.ieee.org).

 - Um fabricante licencia (“compra”) uma parte do espaço de endereços
 - Para garantir unicidade.

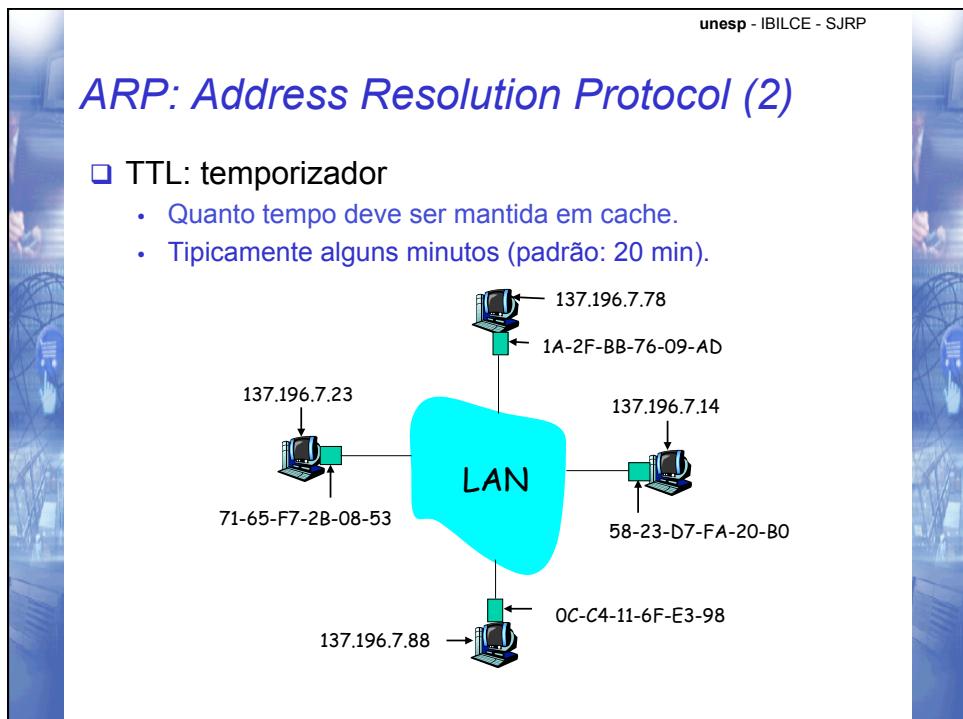
- Endereço MAC não tem estrutura hierárquica → para garantir portabilidade.**
 - Não depende da rede.
- Endereço IP é hierárquico → NÃO é portátil.
 - Depende da rede.
- Endereço MAC de difusão (*broadcast*):
 - All 48 bits one: FF:FF:FF:FF:FF:FF



unesp - IBILCE - SJRP

```
cmd Command Prompt
C:\Users\lhughes.HUGHESNET>arp -a
Interface: 172.20.2.1 --- 0xb
Internet Address      Physical Address      Type
172.20.0.1              00-1b-21-1d-c1-59  dynamic
172.20.0.9              00-1f-33-3e-95-2c  dynamic
172.20.0.11             00-17-a4-ec-11-9c  dynamic
172.20.0.13             00-15-f2-2e-b1-1c  dynamic
172.20.0.33             00-19-0a-ec-ed-b5  dynamic
172.20.1.2              00-19-d5-5c-00-8d  dynamic
172.20.1.3              00-1a-05-97-de-e0  dynamic
172.20.1.5              90-06-ba-82-00-77  dynamic
172.20.1.9              c0-3f-0e-a4-f7-24  dynamic
172.20.1.18             00-19-d2-6b-1d-6c  dynamic
172.20.1.12             00-19-5b-68-5e-2b  dynamic
172.20.1.14             00-15-99-68-3d-4e  dynamic
172.20.1.15             00-13-10-25-44-9e  dynamic
172.20.255.255          ff-ff-f8-ff-ff-ff  static
224.0.0.2                01-00-5e-00-00-02  static
224.0.0.22               01-00-5e-00-00-16  static
224.0.0.252              01-00-5e-00-00-fc  static
225.145.134.86           01-00-5e-11-86-56  static
226.210.124.76           01-00-5e-52-7c-4e  static
226.239.246.216          01-00-5e-6f-f6-d8  static
227.11.77.24             01-00-5e-0b-4d-18  static
227.23.82.81             01-00-5e-17-52-51  static
227.158.132.81           01-00-5e-1e-84-51  static
229.224.223.80           01-00-5e-60-df-50  static
230.36.183.188            01-00-5e-24-b7-be  static
230.73.54.94              01-00-5e-49-36-5e  static
230.242.147.68            01-00-5e-72-93-44  static
231.231.137.118          01-00-5e-67-89-76  static
232.78.190.77             01-00-5e-4e-be-4d  static
233.144.196.87            01-00-5e-10-c4-57  static
234.47.172.93              01-00-5e-2f-ac-5d  static
234.97.55.96              01-00-5e-61-37-60  static
234.139.69.24              01-00-5e-0b-45-18  static
235.161.79.70              01-00-5e-21-4f-46  static
236.21.203.80              01-00-5e-15-cb-50  static
236.127.195.94              01-00-5e-7f-c3-5e  static
237.204.138.71              01-00-5e-4c-8a-5b  static
239.5.34.95                01-00-5e-05-22-5f  static
239.192.152.143             01-00-5e-40-98-8f  static
239.255.255.250             01-00-5e-7f-ff-fa  static
239.255.255.253             01-00-5e-7f-ff-fd  static
255.255.255.255            ff-ff-f8-ff-ff-ff  static
C:\Users\lhughes.HUGHESNET>
```

Tabela ARP

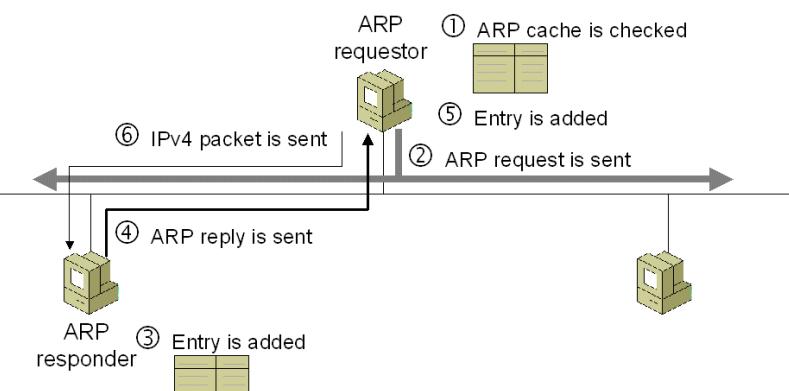


ARP: Address Resolution Protocol (3)

1. Host **A** quer enviar pacote para endereço IP de destino **X na mesma rede local**.
2. Host de origem primeiro verifica se sua tabela ARP contém o endereço IP **X (verifica se há cache)**.
3. Se **X não** há cache, o host **envia um query ARP em broadcast**
 - **MAC destino = FF:FF:FF:FF:FF:FF**
 - **[IP X, MAC (?)] → enviado em broadcast**
4. Todos os hosts na rede local aceitam o pacote ARP.
 1. Destino é broadcast → todos aceitam
 2. Cada um verifica se é ele que tem aquele IP. **Quem tiver, responde.**
5. O host de IP **X** responde direto ao host **A** com pacote ARP informando seu próprio endereço MAC :

[IP X, MAC (X)] → enviado para A.
6. Endereço MAC de X é **armazenado em cache** na tabela ARP de **A**.

ARP request & response



unesp - IBILCE - SJRP

ARP Request

```

Frame 88246: 42 bytes on wire (336 bits), 42 bytes captured (336 bits)
Ethernet II, Src: AsustekC_24:32:9c (00:22:15:24:32:9c), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
  Destination: Broadcast (ff:ff:ff:ff:ff:ff)
  Source: AsustekC_24:32:9c (00:22:15:24:32:9c)
    Type: ARP (0x0806)
  Address Resolution Protocol (request)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (0x0001)
    [Is gratuitous: False]
    Sender MAC address: AsustekC_24:32:9c (00:22:15:24:32:9c)
    Sender IP address: 172.20.2.1 (172.20.2.1)
    Target MAC address: 00:00:00:00:00:00 (00:00:00:00:00:00)
    Target IP address: 172.20.0.11 (172.20.0.11)

0000 ff ff ff ff ff ff 00 22 15 24 32 9c 08 06 00 01  ....:$.....
0010 08 00 06 04 00 01 00 22 15 24 32 9c ac 14 02 01  ....:$.....
0020 00 00 00 00 00 00 ac 14 00 0b  .....

Fonte:
http://www.v6edu.com/joomla/sixscape/index.php/technical-backgrounder/tcp-ip/ip-the-internet-protocol/ipv4-internet-protocol-version-4/ipv4-address-resolution-with-arp

```

unesp - IBILCE - SJRP

ARP Reply

```

Frame 88247: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
Ethernet II, Src: HewlettP_ec:11:9c (00:17:a4:ec:11:9c), Dst: AsustekC_24:32:9c (00:22:15:24:32:9c)
  Destination: AsustekC_24:32:9c (00:22:15:24:32:9c)
  Source: HewlettP_ec:11:9c (00:17:a4:ec:11:9c)
    Type: ARP (0x0806)
    Trailer: 0000000000000000000000000000000000000000000000000000000000000000
  Address Resolution Protocol (reply)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (0x0002)
    [Is gratuitous: False]
    Sender MAC address: HewlettP_ec:11:9c (00:17:a4:ec:11:9c)
    Sender IP address: 172.20.0.11 (172.20.0.11)
    Target MAC address: AsustekC_24:32:9c (00:22:15:24:32:9c)
    Target IP address: 172.20.2.1 (172.20.2.1)

0000 00 22 15 24 32 9c 00 17 a4 ec 11 9c 08 06 00 01 .:$....
0010 08 00 06 04 00 02 00 17 a4 ec 11 9c ac 14 00 0b ::$....
0020 00 22 15 24 32 9c ac 14 02 01 00 00 00 00 00 00 ::$....
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 :$.....

Fonte:
http://www.v6edu.com/joomla/sixscape/index.php/technical-backgrounder/tcp-ip/ip-the-internet-protocol/ipv4-internet-protocol-version-4/ipv4-address-resolution-with-arp

```

unesp - IBILCE - SJRP

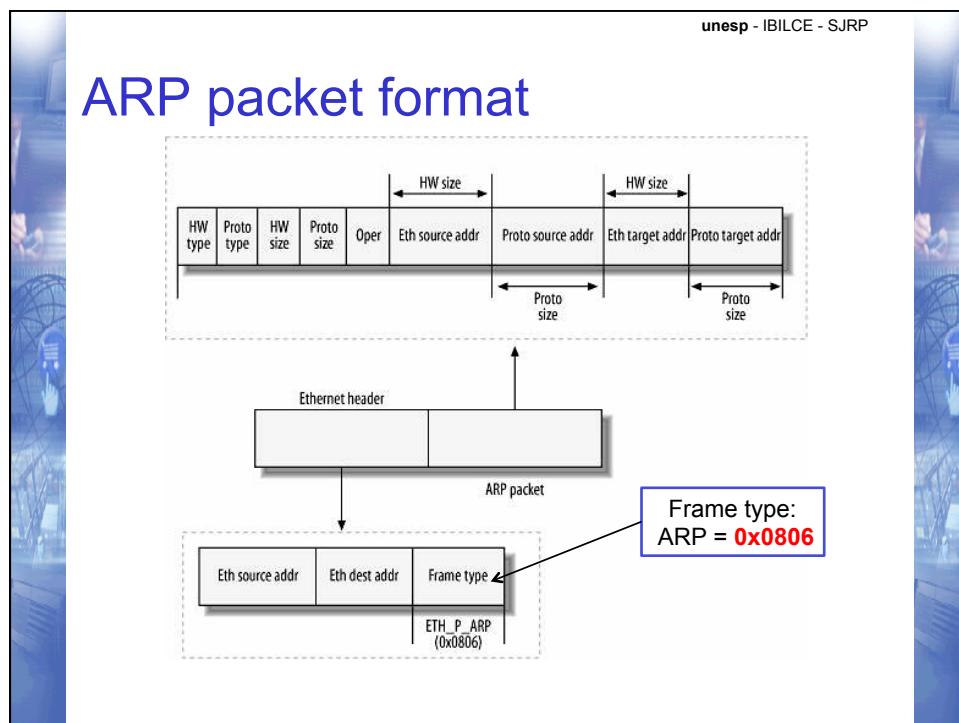
Gratuitous ARP

```

Frame 1527: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
Ethernet II, Src: D-Link_2f:14:6b (00:19:5b:2f:14:6b), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
  Destination: Broadcast (ff:ff:ff:ff:ff:ff)
  Source: D-Link_2f:14:6b (00:19:5b:2f:14:6b)
  Type: ARP (0x0806)
    Trailer: aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
  Address Resolution Protocol (request/gratuitous ARP)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (0x0001)
      [is gratuitous: True]
    Sender MAC address: D-Link_2f:14:6b (00:19:5b:2f:14:6b)
    Sender IP address: 10.3.2.56 (10.3.2.56)
    Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Target IP address: 10.3.2.56 (10.3.2.56)
  
```

"Hey everyone! I own IP address 10.3.2.56 and MAC address 00:19:5b:2f:14:6b. Who owns IP address 10.3.2.56?".

Para quê?



unesp - IBILCE - SJRP

ARP packet format

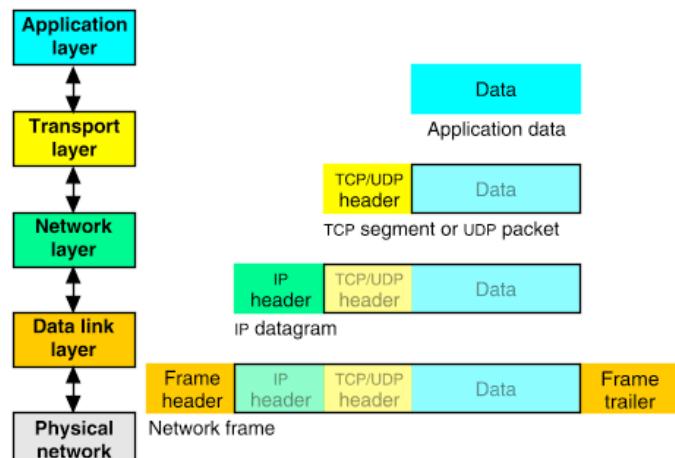
Level 2-ARP

Word Offset	Byte 0	Byte 1	Byte 2	Byte 3
0x0000	Hardware Type (0x01)		Protocol Type (0x80)	
0x0010	HLEN (0x06)	PLEN (0x04)	Operation	
0x0020	Sender Hardware Address			
0x0030	Sender Protocol Address			
0x0040				
0x0050	Target Hardware Address			
0x0060			Target Protocol Address	
0x0070				

unesp - IBILCE - SJRP

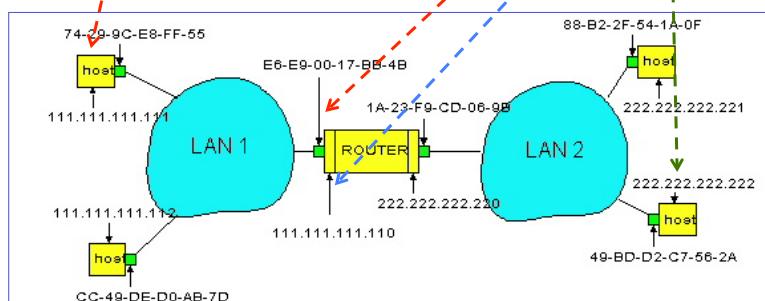
Encapsulamento, roteamento e encaminhamento

Encapsulamento



Roteando um pacote para outra rede.

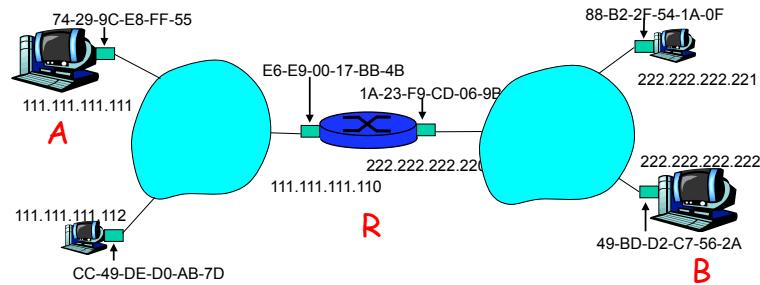
- ❑ Exemplo: rotear pacote do endereço IP de origem <111.111.111.111> ao endereço de destino <222.222.222.222>.
- ❑ Na tabela de rotas na origem, encontra roteador 111.111.111.110
- ❑ Na tabela ARP na origem, tira endereço MAC E6-E9-00-17-BB-4B, e etc...



unesp - IBILCE - SJRP

Endereçamento: roteando para outra LAN

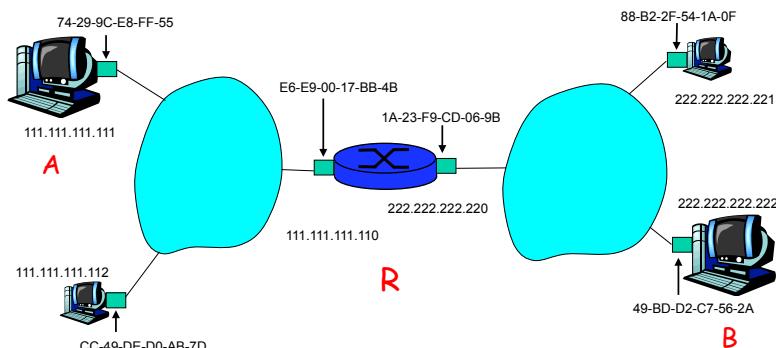
acompanhamento: enviar datagrama de **A** para **B** via **R**.
 suponha que A saiba o endereço IP de B



- ❑ Duas tabelas ARP no roteador **R**, uma para cada rede IP (LAN)

unesp - IBILCE - SJRP

Como acontece o roteamento e o encaminhamento?



unesp - IBILCE - SJRP

Como acontece o roteamento e o encaminhamento?

- A cria datagrama IP com origem **A**, destino **B**
- A usa ARP para obter endereço MAC de **R** para **111.111.111.110**
- A cria quadro da camada de enlace com endereço MAC de **R** como destino, quadro contém datagrama IP **A-para-B**
- NIC de **A** envia quadro
- NIC de **R** recebe quadro
- R** remove datagrama IP do quadro Ethernet, vê o seu destinado a **B**
- R** usa ARP para obter endereço MAC de **B**
- R** cria quadro contendo datagrama IP **A-para-B** e envia para **B**

The diagram shows three nodes: host A, router R, and host B. Host A has two interfaces with MAC addresses 74-29-9C-E8-FF-55 and 111.111.111.111. Router R has two interfaces with MAC addresses E6-E9-00-17-BB-4B and 111.111.111.110. Host B has two interfaces with MAC addresses 88-B2-2F-54-1A-0F and 222.222.222.221. An intermediate host with MAC address CC-49-DE-D0-AB-7D and IP 111.111.111.112 is also shown. Arrows indicate the flow of frames from A to R and from R to B. Labels like '222.222.222.220' and '1A-23-F9-CD-06-9B' are present on the links between R and the intermediate host.

unesp - IBILCE - SJRP

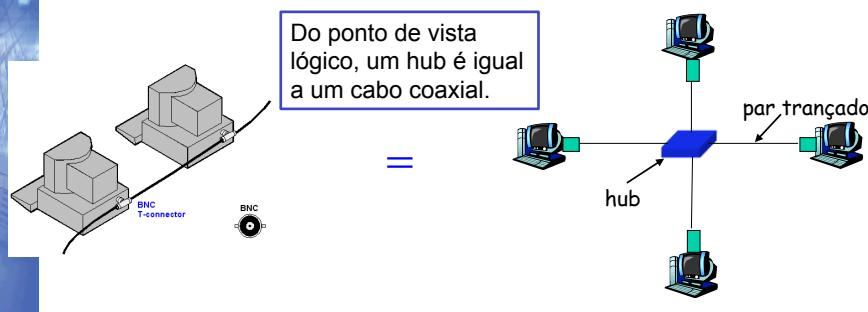
Estruturas de redes locais: hubs e switches

A photograph of a black metal rack-mountable network switch. It has a front panel with several ports, a small display screen, and various status LEDs. The model name 'DriSwitch 2000' is printed on the front panel. The brand 'DATACOM' is also visible.

Hubs

Repetidores da camada física (repetidores “burros”):

- Todos os nós conectados ao hub podem colidir uns com os outros.
- Sem *buffering (cache)* de quadros.
- Sem CSMA/CD no hub: placas do host devem detectar colisões.
- Bits que chegam num enlace são replicados em *todos* os outros enlaces, na mesma velocidade.



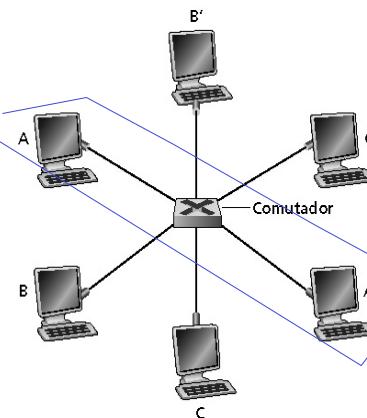
Comutador (switch) Ethernet (1)

- Um *switch* Ethernet é um dispositivo para “**conexões dedicadas**” ponto-a-ponto.
- Um host ligado a um *switch* através de uma conexão dedicada ponto-a-ponto **sempre detecta que o meio está ocioso**:
 - **Não haverá colisões entre duas portas.**
- *Switch* Ethernet provê combinações de conexões compartilhadas/dedicadas, a 10/100/1000 Mbps.

Comutador (switch) Ethernet (2)

- Acesso dedicado ponto a ponto.
- Switch possui muitas interfaces (**portas**).
- Hospedeiros possuem conexão direta ao switch.
- Sem colisões** e **full duplex**.

Switching:
A-para-A' e B-para-B',
simultaneamente, sem
colisões



Comutador (switch) Ethernet (3)

- É um **dispositivo de camada de enlace**.
- Armazena** e encaminha quadros Ethernet (*store-and-forward*).
- Examina o cabeçalho do quadro e, **seletivamente**, encaminha o quadro **baseado no endereço MAC de destino**.
- Quando um quadro está para ser encaminhado no segmento, usa CSMA/CD para acessar o segmento.
- Transparente.
 - Hospedeiros são percebem a presença dos switches.
 - Mantém a compatibilidade reversa com ethernet e CSMA/CD
- Plug-and-play & self-learning* (auto-aprendizado).
 - Switches não precisam ser configurados.

Comutador (*switch*) Ethernet (4)

- ❑ **Switches suportam comutação “*cut-through*”:**
 - O quadro é re-encaminhado imediatamente ao destino, **sem esperar a montagem do quadro inteiro no buffer do comutador.**
 - Há uma pequena redução na latência.
- ❑ **Switches Ethernet variam em tamanho.**
 - Os mais rápidos incorporam uma rede de interconexão.
 - Chamada de *backplane* ou *switch fabric*, de alta capacidade.

Self learning (auto-aprendizado)

- ❑ Um switch possui uma **tabela de comutação**.
- ❑ Entradas na tabela do *switch* posuem o formato:

[endereço MAC host, **interface**, marca de tempo]
- ❑ Entradas expiradas na tabela são descartadas.
- ❑ **Switch aprende** quais hosts estão em quais portas.
 - Quando recebe um quadro, o switch “aprende” em qual porta está o emissor.
 - Associa o MAC Address à porta.
 - Registra o par [emissor / porta] na tabela.
 - Uma porta pode ter mais de um MAC Address associado a ela (quando isso acontece e por quê?)

Decisão, encaminhamento e filtragem

Quando um switch recebe um quadro:

```

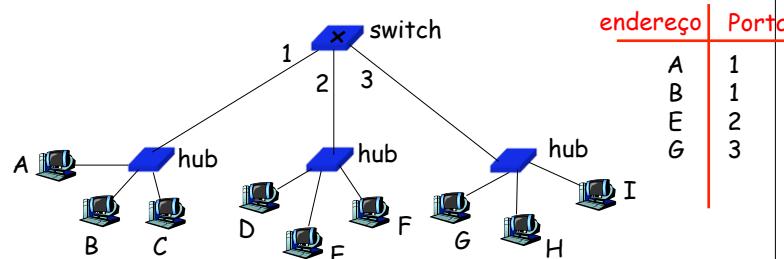
1. indexa a tabela do switch usando endereço MAC de destino
2. if entrada for encontrada para o destino
    then{
        if destino está na porta deste quadro que chegou
            then descarta o quadro.
        else encaminha o quadro na porta indicada.
    }
else flood;

```

Caso não encontre a entrada do destino na tabela, encaminha para todas as portas, exceto para aquela de o quadro chegou.

Exemplo de filtragem e encaminhamento (1)

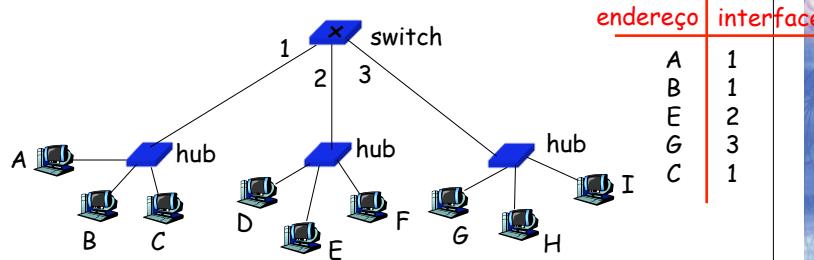
Suponha que C envia um quadro para D



- Switch recebe o quadro de C, com destino a D.
 - Registra na tabela que C está na porta 1
 - Como D não está na tabela, o switch encaminha o quadro para as portas 2 e 3 (não envia para 1 porque C está na 1).
 - Quadro é recebido por D.

Exemplo de filtragem e encaminhamento (2)

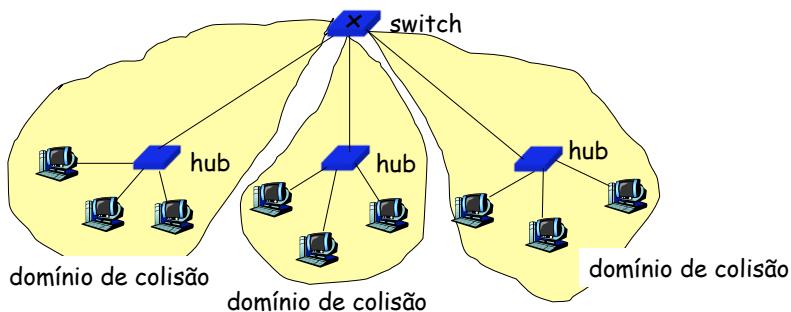
Suponha que D responde com um quadro para C.

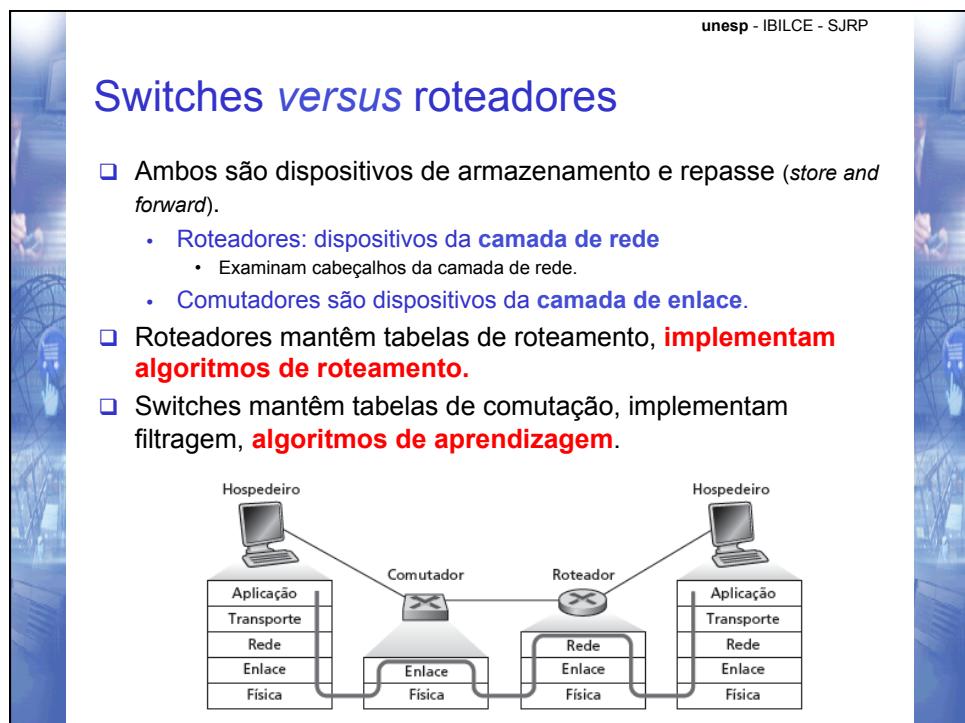
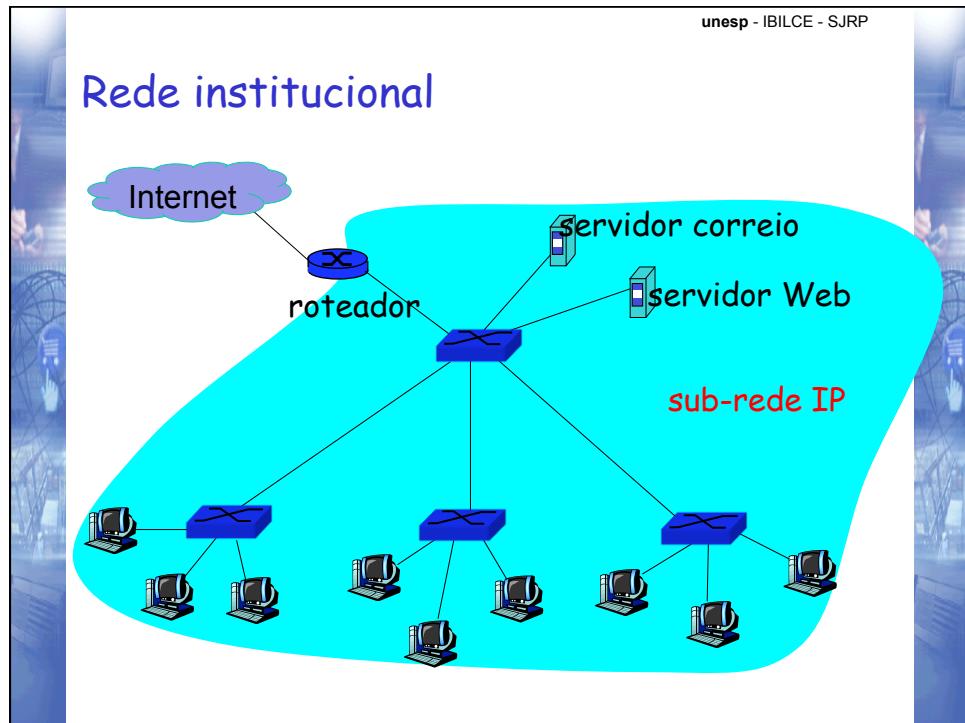


- Switch recebe quadro proveniente de D com destino a C.
 - Registra na tabela para saber que D está na porta 2
 - Como C está na tabela, o switch encaminha o quadro apenas para a porta 1.
 - Quadro é recebido por C.

Switch: isolamento de tráfego

- A instalação do switch quebra as sub-redes em segmentos de LAN
- Switch **filtre** pacotes:
 - Alguns quadros do mesmo segmento de LAN não são usualmente encaminhados para outros segmento de LAN.
 - Segmentos se tornam separados em **domínios de colisão**.



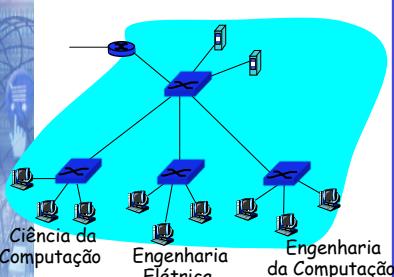


VLANs

VLANs: motivação

O que acontece se:

- ❑ Usuário da CC muda para EE, mas quer se conectar ao comutador CC?
- ❑ Único domínio de *broadcast*:
 - Todo tráfego de broadcast da camada 2 (ARP, DHCP) cruza a LAN inteira
 - Questões de eficiência, segurança/privacidade.



unesp - IBILCE - SJRP

VLANs

VLAN baseada em porta: portas de comutador agrupadas (por software) para que **único** switch físico...

Virtual Local Area Network:

switch(es) admitindo capacidades de VLAN podem ser configurados para definir múltiplas LANs **virtuais** por única infraestrutura de LAN física.

...opere como **múltiplos** switches virtuais

unesp - IBILCE - SJRP

LANS Virtuais

- ❑ SEGMENTO = Domínio de **Colisão**.
 - Os computadores de um Hub estão no mesmo segmento físico.
- ❑ VLAN = Domínio de **Broadcast**.
 - O tráfego de broadcast só pode passar de uma VLAN para outra **apenas através de um roteador**.

unesp - IBILCE - SJRP

VLAN baseada em porta

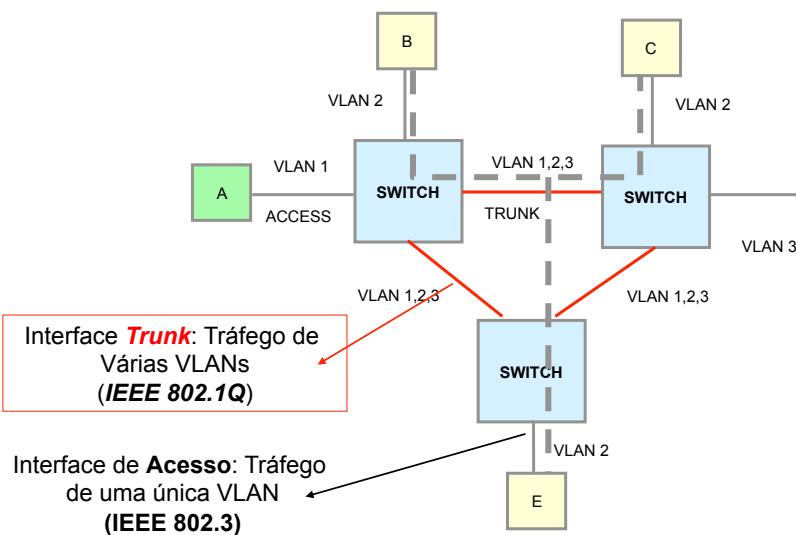
- isolamento de tráfego:** quadros de / para portas 1-8 só podem alcançar portas 1-8
 - também podem definir VLAN com base em endereços MAC das extremidades, em vez de porta do comutador.
- Inclusão dinâmica:** portas podem ser atribuídas dinamicamente entre VLANs.
- Repasso entre VLANs:** é feito por roteamento (assim como em switches separados).
 - Na prática, fornecedores vendem uma combinação de comutador e roteador: *router-switch*.

unesp - IBILCE - SJRP

VLANS em múltiplos switches

- porta de tronco:** carrega quadros entre VLANs definidas sobre vários switches físicos
 - Quadros repassados dentro da VLAN entre switches não podem ser quadros 802.1 comuns:
 - Devem ter informação de **VLAN ID**.
 - Protocolo **802.1q** inclui campos de cabeçalho adicionais para quadros repassados entre portas de tronco.

Interligação de Switches

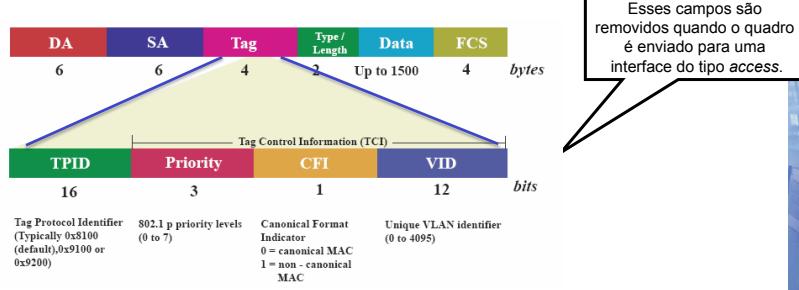


Modos das Portas de Switch

- ❑ Portas de um switch podem operar em 2 modos:
 - Modo **Access**.
 - Cada porta do switch pertence a uma única VLAN.
 - Quadros Ethernet: formato normal padrão.
 - Modo **Trunk**.
 - O tráfego de múltiplas VLANs é multiplexado em um único link físico.
 - Usualmente interconectam switches.
 - Quadros Ethernet: formato especial de VLAN (visto em seguida).
 - Caso necessário, apenas computadores com placas de rede especiais podem se conectar a essas portas trunk.
 - Entretanto, não é usual computadores se conectarem em portas trunk.

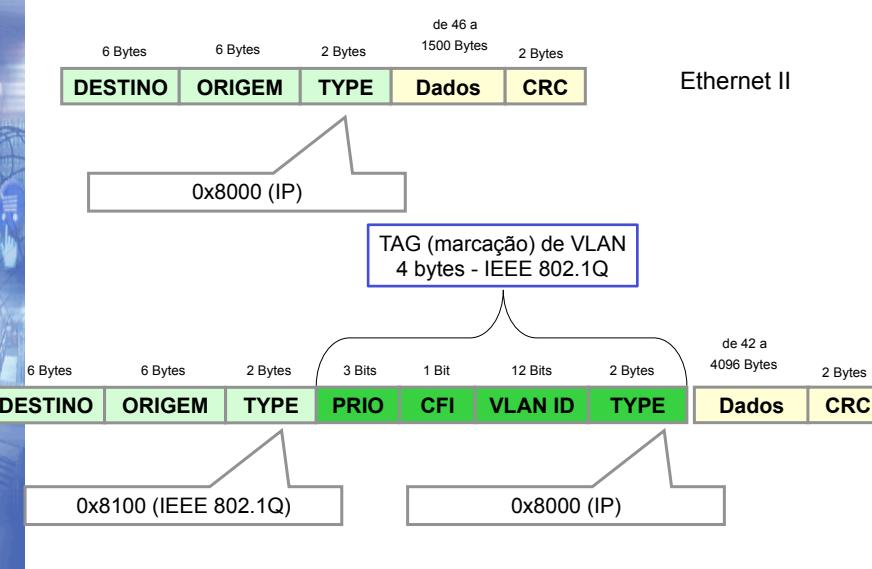
Protocolos Trunk

- ❑ Os quadros nas interfaces **Trunk** são formatados em quadros especiais para identificar a quais LANs eles pertencem.
- ❑ O **IEEE 802.1Q** é um protocolo para interface **Trunk**.

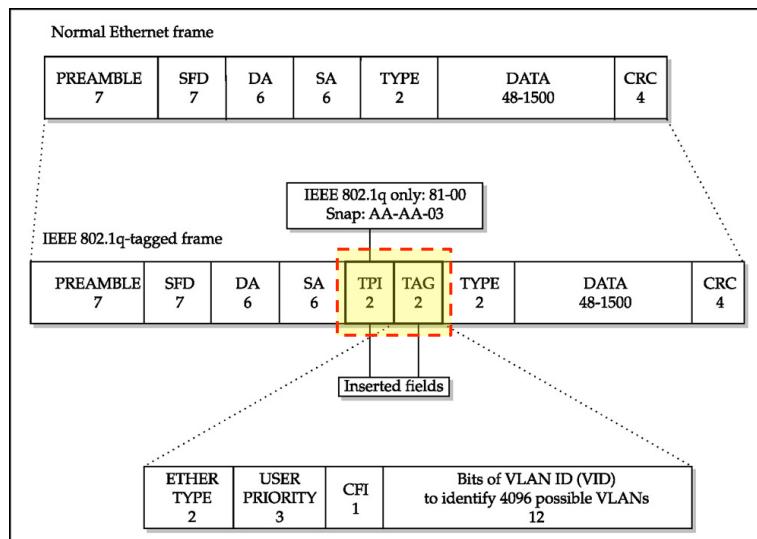


Esses campos são removidos quando o quadro é enviado para uma interface do tipo access.

Formato IEEE 802.1Q



Formato IEEE 802.1Q – visão completa



Tipos de Tráfego: Exemplos

- Switches Ethernet precisam diferenciar o tráfego, pois cada tipo de aplicação pode ter requisitos de QoS distintos (**campo PRIO**):
 1. Gerenciamento da Rede: alta disponibilidade
 2. Voz: Atraso < 10 ms
 3. Video: Atraso < 100 ms
 4. Carga Controlada
 5. *Excellent Effort: Best Effort* para usuários vip.
 6. *Best Effort: Best Effort* para os demais usuários.
 7. *Background: Transferências em batch, jogos, etc.*

unesp - IBILCE - SJRP

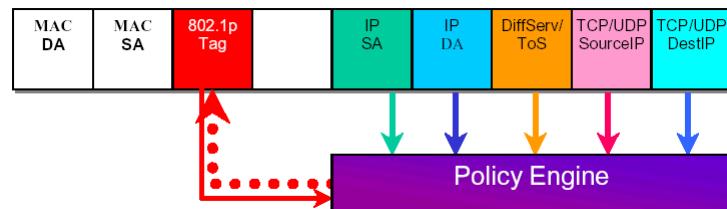
Uso de Prioridade: Exemplo

- ❑ De acordo com a abordagem do padrão 802.1p, os diferentes tipos de tráfego podem ser tratados utilizando 8 níveis de prioridade:
 - 000 = 0 : *Best Effort*
 - 001 = 1 : *Background*
 - 010 = 2 : Não Utilizado
 - 011 = 3 : *Excellent Effort*
 - 100 = 4 : Carga Controlada
 - 101 = 5 : Vídeo
 - 110 = 6 : Voz
 - 111 = 7 : Controle de Rede

unesp - IBILCE - SJRP

Remarcação de Prioridade

- ❑ switches permitem criar regras de (re)marcação em função dos campos dos cabeçalhos de transporte e rede.



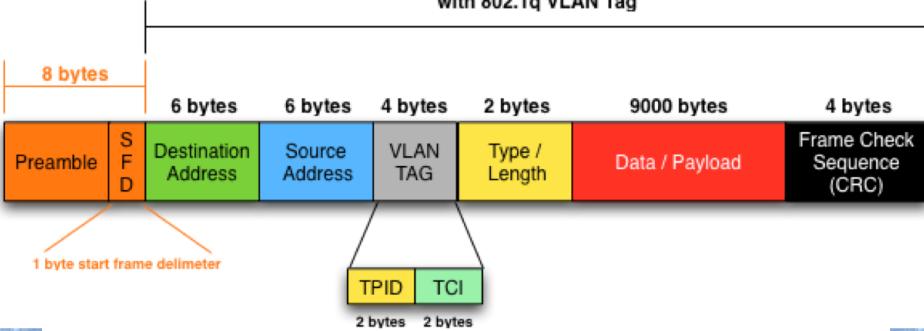
Mapeamento DSCP - QOS

- Para integração com a marcação **diffserv**, o switch permite mapear códigos de DSCP em níveis de prioridade.
 - Diffserv* são vistos em protocolos multimídia usando rede TCP/IP.

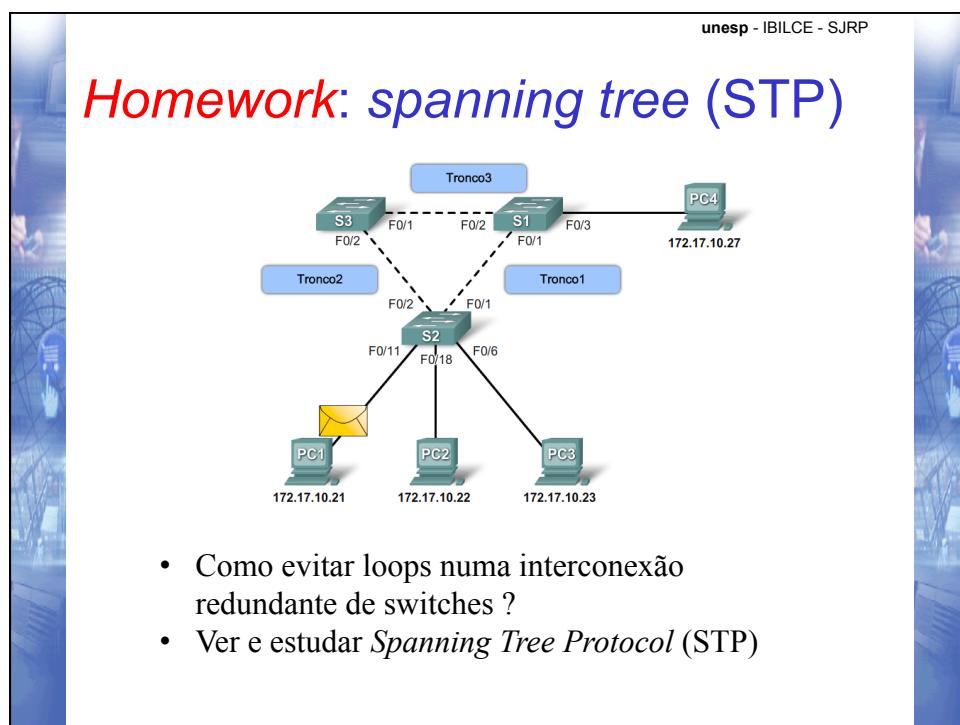
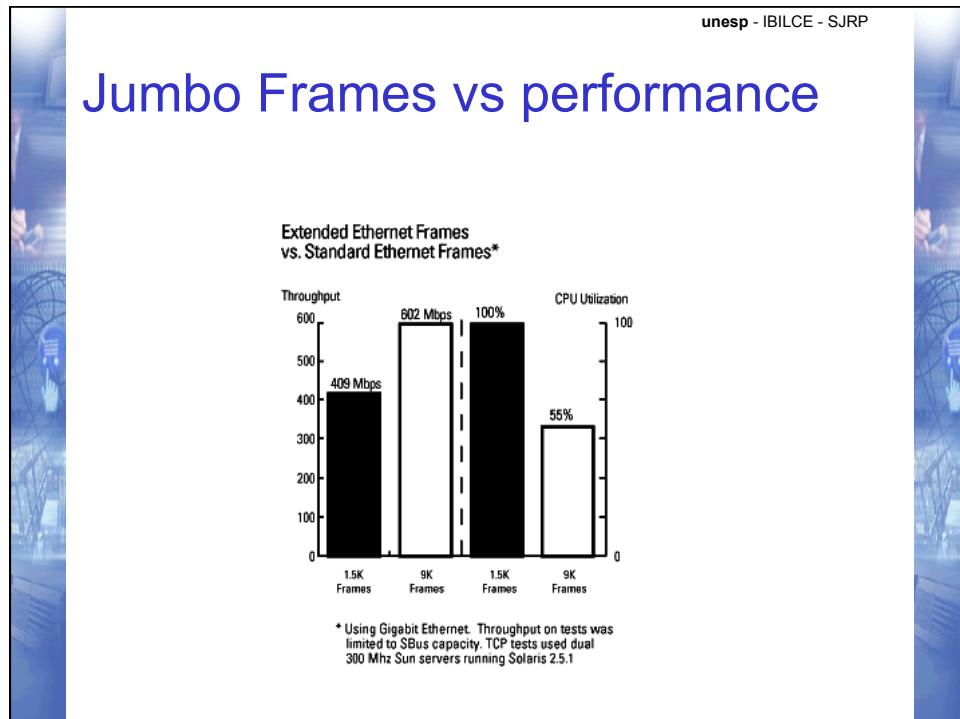
Nós retornaremos a este assunto quando for discutido o tópico de “**Procolos Multimedia na Internet**” com *diffserv* e DSCP.

Jumbo Frames

9022 byte maximum Jumbo frame size
with 802.1q VLAN Tag



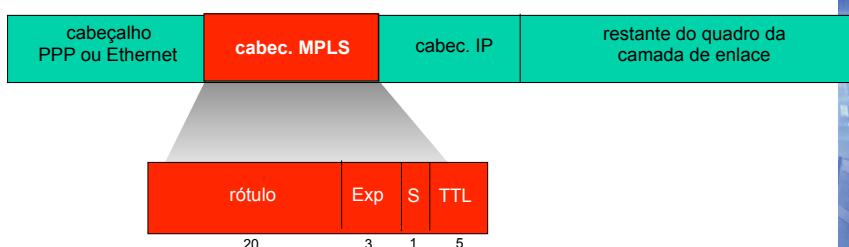
<http://sd.wareonearth.com/~phil/jumbo.html>



unesp - IBILCE - SJRP

Homework: Multiprotocol Label Switching (MPLS)

- ❑ Objetivo inicial: agilizar o repasse do IP usando rótulo de tamanho fixo (em vez de endereço IP) para fazer o repasse.
 - Ideias apanhadas da técnicas de *Virtual Circuit* (VC)
 - Mas, datagrama IP ainda mantém endereço IP.



cabeçalho PPP ou Ethernet	cabec. MPLS	cabec. IP	restante do quadro da camada de enlace
------------------------------	-------------	-----------	---

rótulo	Exp	S	TTL
20	3	1	5

unesp - IBILCE - SJRP

Homework: Multiprotocol Label Switching (MPLS)

- ❑ Também chamado roteador comutado por rótulo
- ❑ Encaminha pacotes à interface de saída com base apenas no valor do rótulo (não inspeciona endereço IP).
 - Tabela de repasse MPLS distintas das tabelas de repasse do IP
- ❑ Protocolo de sinalização necessário para configurar repasse:
 - RSVP-TE
 - repasse possível ao longo de caminhos que o IP sozinho não permitiria (Exemplo: roteamento específico da origem).
 - Utiliza-se MPLS para engenharia de tráfego.

Camada de enlace - Resumo

- ❑ Princípios dos serviços da camada de enlace:
 - Compartilhando um canal broadcast: acesso múltiplo.
 - Endereçamento da camada de enlace.
 - Protocolos de acesso ao meio
- ❑ Instanciação e implementação de várias tecnologias da camada de enlace.
- ❑ Ethernet.
- ❑ LANS comutadas / switches.
- ❑ VLANs.

Referência:

James F. Kurose & Keith W. Ross.
Redes de Computadores e a Internet

Capítulo 5 – Camada de Enlace de Dados

5a. Edição (2010)
Editora: Pearson Education
ISBN: 8588639971