



Pontifícia Universidade Católica de Minas Gerais
Bacharelado em Ciência da Computação
Redes de Computadores I

Redes de Computadores I

Prof^a: Raquel Mini
raquelmini@pucminas.br
2º semestre de 2019

CAMADA FÍSICA

Camada Física

Camada de Aplicação
Camada de Transporte
Camada de Rede
Camada de Enlace
Camada Física

Camada Física

- Meios de transmissão
- Largura de banda e taxa de dados
- Unidades métricas
- Atrasos em redes de comutação de pacotes

Meios de Transmissão

- Objetivo da camada física:
 - ◆ Transportar uma sequência de bits de uma máquina para outra
- Problema a ser resolvido:
 - ◆ Codificação de bits

Meios de Transmissão

- O tipo de meio físico a ser usado depende, dentre outros fatores de:
 - ◆ Largura de banda (*bandwidth*)
 - ◆ Atraso (*delay*) ou latência (*latency*) ou retardo
 - ◆ Custo
 - ◆ Facilidade de instalação e manutenção

Meios de Transmissão

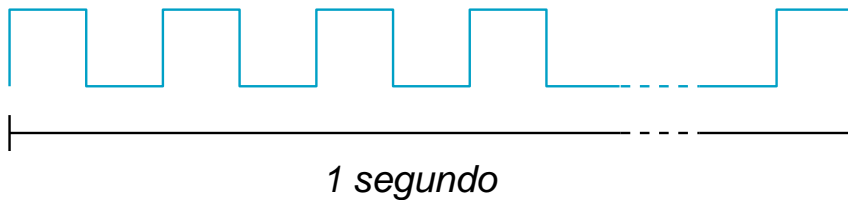
- Os meios podem ser agrupados em:
 - ◆ Guiados: as ondas são guiadas através de um caminho físico (par trançado, cabo coaxial ou fibra óptica)
 - ◆ Não-guiados: as ondas se propagam sem haver um caminho físico (ondas de rádio, microondas ou infravermelho)

Largura de Banda

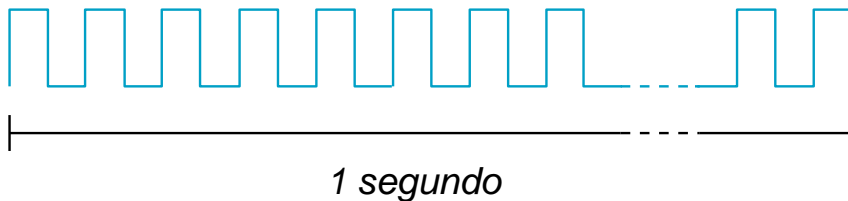
- A faixa de frequências transmitidas sem serem fortemente atenuadas denomina-se largura de banda
- A largura de banda é uma propriedade física do meio de transmissão e, em geral, depende da construção, da espessura e do comprimento do meio
- Limitando-se a largura de banda, limita-se a taxa de dados

Taxa de Dados

- Número de bits que podem ser transmitidos por uma rede em um período de tempo



1 Mbps: 1 milhão de bits por segundo ($1\mu\text{s}$ para transmitir cada bit)



2 Mbps: 2 milhões de bits por segundo ($0.5\mu\text{s}$ para transmitir cada bit)

Exercícios

1. Queremos enviar uma sequência de imagens de tela de computador por fibra óptica. A tela tem 480x640 pixels e cada pixel tem 24 bits. Há 60 imagens de tela por segundo. Qual é a taxa de dados necessária?
2. Quanto tempo leva para transmitir uma mensagem de 32 KB por um canal de 10 Mbps?

MB, Mbps, KB, Kbps

- **b** significa bits e **B** bytes
- Mega significa 2^{20} ou 10^6 ?
- Kilo significa 2^{10} ou 10^3 ?
- Largura de banda
 - ◆ Está relacionada com velocidade de clock (Hz)
 - ◆ Mbps significa 10^6 bits por segundo
- Mensagem a ser transmitida
 - ◆ Mensagens são armazenadas na memória e estas são medidas em potências de 2
 - ◆ MB significa 2^{20} bytes

Unidades Métricas

Exp.	Explicit	Prefix	Exp.	Explicit	Prefix
10^{-3}	0.001	milli	10^3	1,000	Kilo
10^{-6}	0.000001	micro	10^6	1,000,000	Mega
10^{-9}	0.000000001	nano	10^9	1,000,000,000	Giga
10^{-12}	0.000000000001	pico	10^{12}	1,000,000,000,000	Tera
10^{-15}	0.000000000000001	femto	10^{15}	1,000,000,000,000,000	Peta
10^{-18}	0.000000000000000001	atto	10^{18}	1,000,000,000,000,000,000	Exa
10^{-21}	0.000000000000000000001	zepto	10^{21}	1,000,000,000,000,000,000,000	Zetta
10^{-24}	0.000000000000000000000001	yocto	10^{24}	1,000,000,000,000,000,000,000,000	Yotta

Exercício

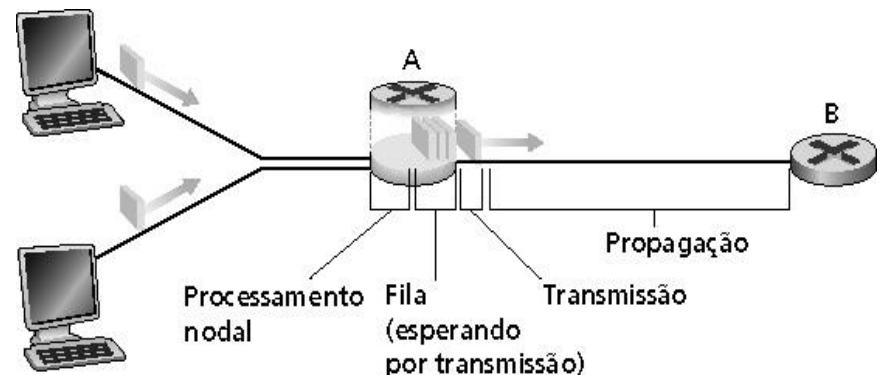
3. Imagine que você tenha treinado Bernie, seu cachorro São Bernardo, para carregar uma caixa de três fitas de 8 mm, em vez de um cantil de conhaque. Cada uma dessas fitas contém 7 GB. O cachorro pode viajar a seu lado, onde quer que você esteja, a 18 km/h. Para que intervalo de distância Bernie terá uma taxa de dados mais alta que uma linha de transmissão cuja taxa de dados é de 150 Mbps?

Atrasos em Redes de Comutação de Pacotes

- Um pacote começa no sistema final (origem), passa por uma série de roteadores e termina sua jornada em outro sistema final (destino)
- Quando o pacote chega a um roteador, vindo do nó anterior, o roteador examina o cabeçalho do pacote para determinar o enlace de saída apropriado e, em seguida, o direciona ao enlace

Atrasos em Redes de Comutação de Pacotes

- Quando um pacote viaja de um nó ao nó subsequente (sistema final ou roteador), ele sofre ao longo do caminho diferentes tipos de atrasos:
 - ◆ Atraso de processamento
 - ◆ Atraso de fila
 - ◆ Atraso de transmissão
 - ◆ Atraso de propagação



Atrasos em Redes de Comutação de Pacotes

- **Atraso de processamento:** tempo requerido para examinar o cabeçalho do pacote e determinar para qual fila direcioná-lo
- **Atraso de fila:** o pacote sofre um atraso de fila enquanto espera para ser transferido no enlace

Atrasos em Redes de Comutação de Pacotes

- **Atraso de transmissão:** um pacote é transmitido assim que todos os pacotes que chegaram antes tenham sido transmitidos (depende da velocidade de transmissão do enlace e do tamanho do pacote)
- **Atraso de propagação:** assim que um bit é lançado no enlace, ele precisa se propagar até o próximo roteador, o bit se propaga à velocidade de propagação do enlace (depende da velocidade de propagação e do tamanho do enlace)

Exercício

4. Considere dois computadores, A e B, conectados por um único enlace de taxa R bps. Suponha que esses computadores estejam separados por m metros e que a velocidade de propagação ao longo do enlace seja de s metros/segundo. O Computador A tem de enviar um pacote de L bits ao computador B.
- a) Expresse o atraso de propagação, d_{prop} .
 - b) Determine o tempo de transmissão do pacote, d_{trans} .
 - c) Ignorando os atrasos de processamento e de fila, obtenha uma expressão para o atraso fim-a-fim.

Exercício (cont.)

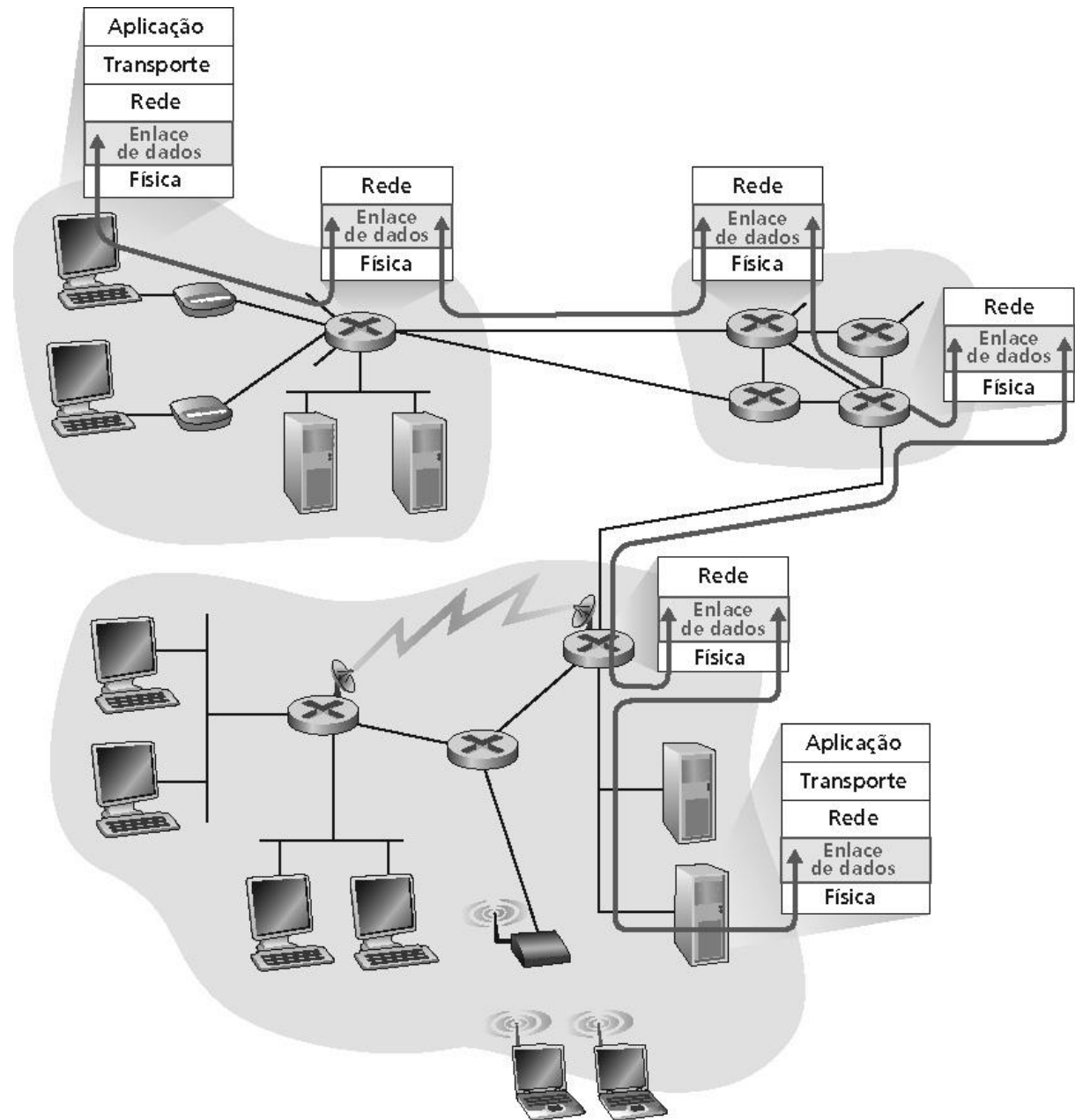
- d) Suponha que o computador A comece a transmitir o pacote no instante $t=0$. No instante $t = d_{trans}$, onde estará o último bit do pacote?
- e) Suponha que d_{prop} seja maior do que d_{trans} . Onde estará o primeiro bit do pacote no instante $t = d_{trans}$?
- f) Suponha que d_{prop} seja menor do que d_{trans} . Onde estará o primeiro bit do pacote no instante $t = d_{trans}$?
- g) Suponha que $s=2,5 \times 10^8 \text{m/s}$, $L = 100 \text{ bits}$ e $R = 28 \text{ Kbps}$. Para qual distância d_{prop} é igual a d_{trans} ?

CAMADA DE ENLACE

Camada de Enlace

Camada de Aplicação
Camada de Transporte
Camada de Rede
Camada de Enlace
Camada Física

Camada de Enlace



Camada de Enlace

- Funções da camada de enlace:
 - ◆ Enquadramento: agrupar sequência de bits em quadros
 - ◆ Detecção e correção de erros

- Subcamada de controle de acesso ao meio (MAC)
 - ◆ Protocolos da camada de enlace de redes difusão:
 - ▶ Aloha
 - ▶ CSMA
 - ▶ Ethernet

Funções da Camada de Enlace

- Responsável pela comunicação entre dois computadores adjacentes
 - ◆ Adjacente significa que dois computadores estão fisicamente ligados por um canal de comunicação FIFO (*first-in-first-out*), ou seja, que preserva a ordem que os bits foram enviados

Funções da Camada de Enlace

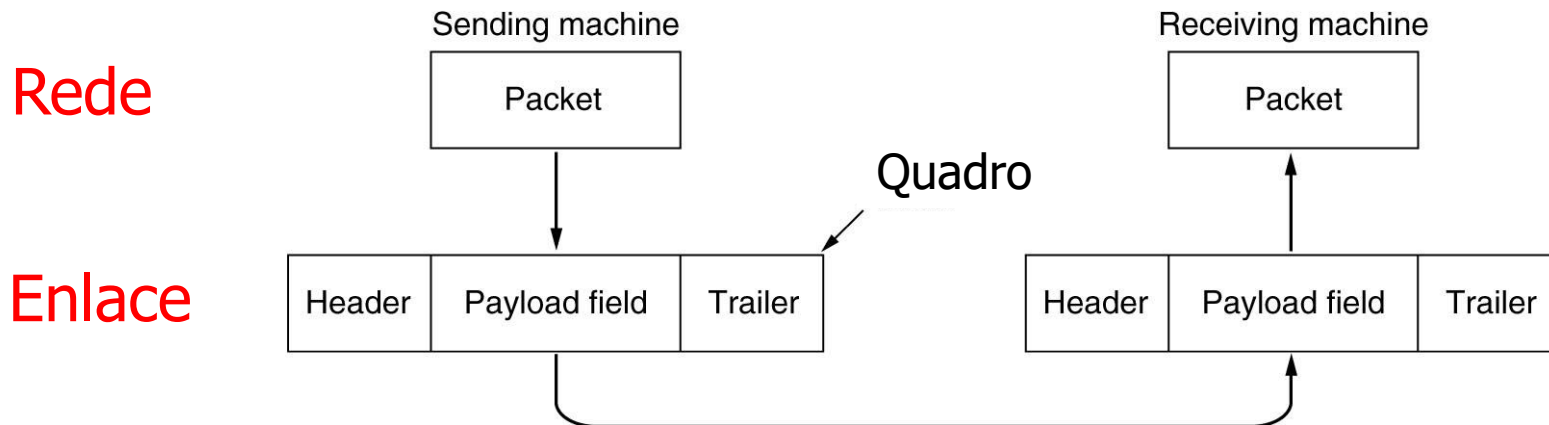
- A camada física aceita um fluxo de bits brutos e tenta entregá-los ao destino:
 - ◆ Não há garantia de que esse fluxo de bits seja livre de erros
 - ◆ O número de bits recebidos pode ser menor, igual ou maior que o número de bits transmitidos
 - ◆ Os bits podem ter valores diferentes dos bits originalmente transmitidos
 - ◆ Permitem uma taxa máxima de transferência
 - ◆ Possuem um tempo de propagação diferente de zero

Funções da Camada de Enlace

- A camada de enlace de dados é responsável por transformar um canal de transmissão bruto em uma linha que pareça livre de erros para a camada de rede (detectar e, se necessário, corrigir erros)
- O transmissor divide os dados de entrada em quadros com algumas centenas ou alguns milhares de bytes
- Redes tipo difusão devem implementar um mecanismo de controle de acesso ao meio (subcamada de controle de acesso ao meio – MAC)

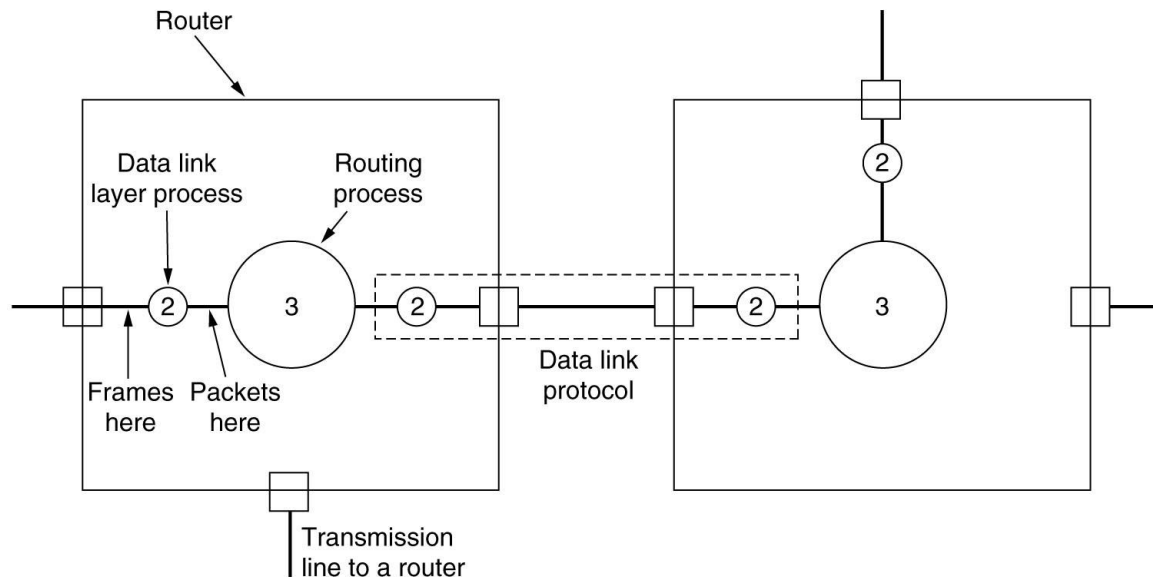
Funções da Camada de Enlace

- A camada de enlace recebe os pacotes da camada de rede e os encapsula em quadros para transmissão
 - ◆ Quadro = cabeçalho (header) + carga útil (pacote recebido da camada de rede) + final de quadro (trailer)



Protocolo da Camada de Enlace

- O protocolo da camada de enlace define o formato dos quadros trocados entre os nós nas extremidades do enlace, bem como as ações realizadas por esses nós ao enviar e receber os quadros



Enquadramento

- Problema a ser resolvido:
 - ◆ Como agrupar sequências de bits em quadros para que possam ser processados como unidades de informação?
 - ou, de outra forma,
 - ◆ Como fazer delimitação de quadros?

Enquadramento

- Soluções:
 1. Intervalos de tempo
 2. Contagem de caracteres
 3. Bytes de *flags*, com inserção de bytes
 4. *Flags* iniciais e finais, com inserção de bits
 5. Violação de codificação da camada física

Soluções para o Enquadramento

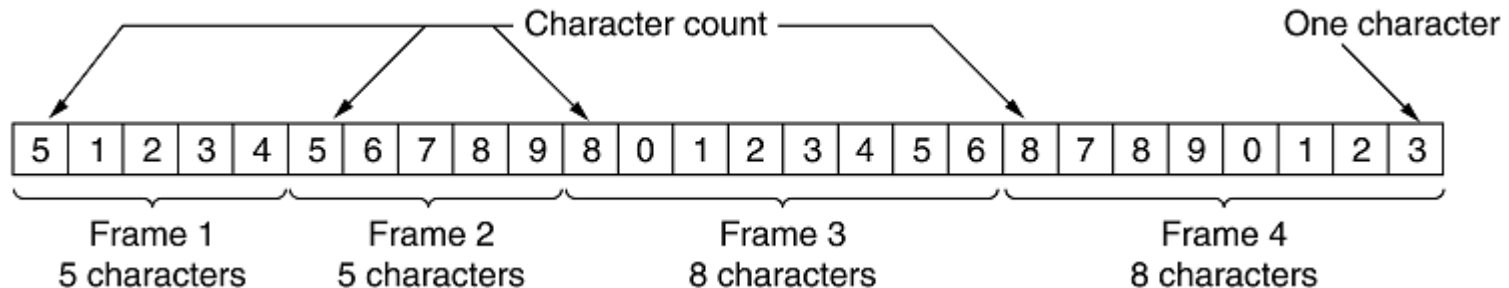
1. Intervalos de tempo

- ◆ Inserir intervalos de tempo entre transmissões de quadro
- ◆ As redes raramente oferecem qualquer garantia em relação à temporização
- ◆ É possível que esses intervalos sejam condensados ou que outros intervalos sejam inseridos durante a transmissão

Soluções para o Enquadramento

2. Contagem de caracteres

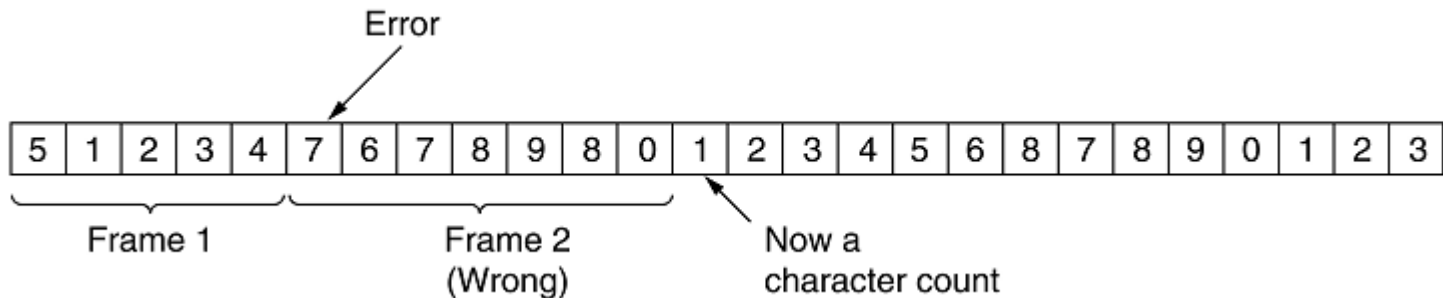
- ◆ Usa um campo no cabeçalho para especificar o número de caracteres no quadro



Soluções para o Enquadramento

2. Contagem de caracteres

- ◆ Problema: erro nesse campo faz com que o receptor perca a sincronização

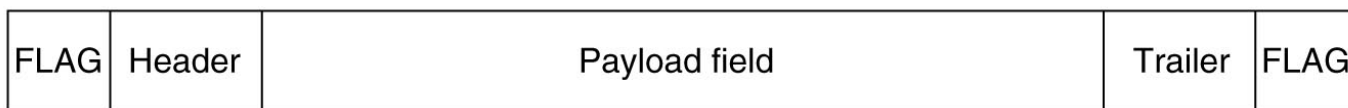


- Não é usado na prática para protocolos da camada de enlace
- Pode ser utilizado pelos protocolos acima da camada de enlace?

Soluções para o Enquadramento

3. Bytes de *flags*, com inserção de bytes

- ◆ Cada quadro começa e termina com bytes especiais
- ◆ Se o receptor perder a sincronização, ele poderá simplesmente procurar pelo byte de *flag* para descobrir o fim do quadro atual
- ◆ Dois bytes de *flag* consecutivos indicam o fim de um quadro e o início do próximo



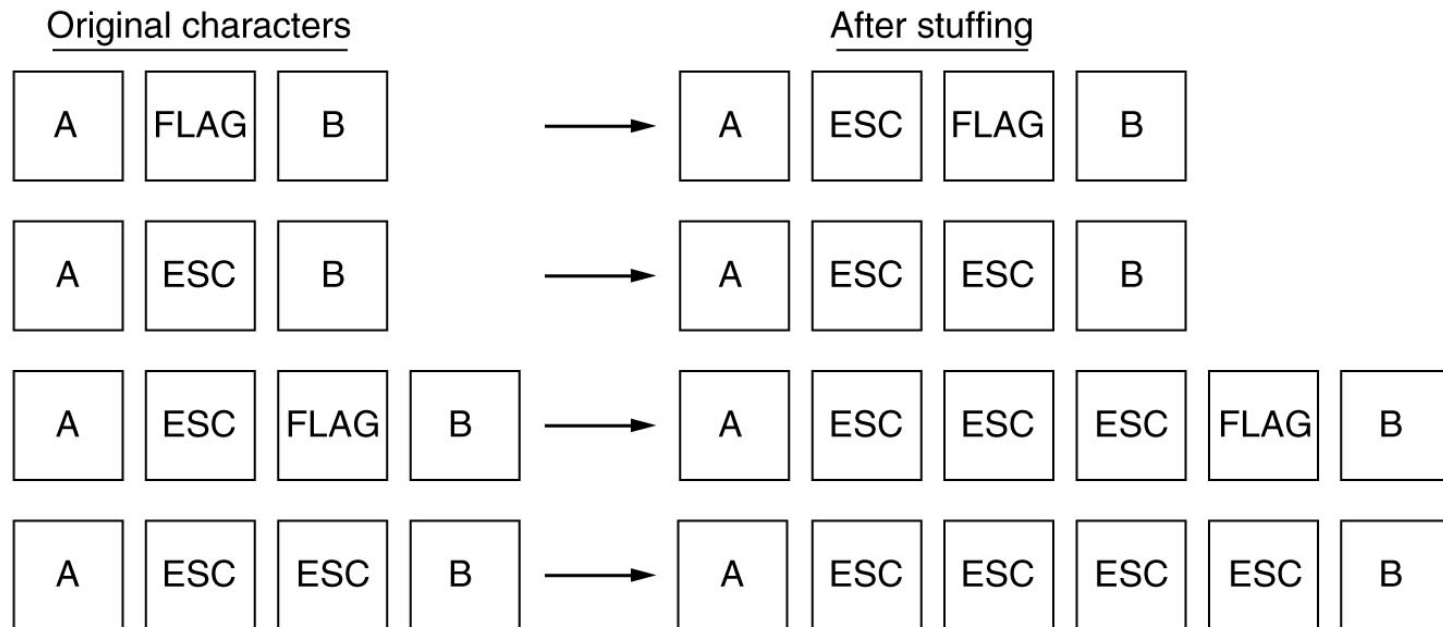
Soluções para o Enquadramento

3. Bytes de *flags*, com inserção de bytes

- ◆ E se o padrão de bits do byte do *flag* ocorrer nos dados?
- ◆ A camada de enlace do transmissor deve incluir um caractere de escape especial (ESC) imediatamente antes de cada byte de *flag* “acidental” nos dados
- ◆ A camada de enlace da extremidade receptora remove o byte de escape antes de entregar os dados à camada de rede

Soluções para o Enquadramento

3. Bytes de *flags*, com inserção de bytes



Soluções para o Enquadramento

4. *Flags* iniciais e finais, com inserção de bits
- ◆ Permite enviar um número arbitrário de bits
 - ◆ Quadros são delimitados por uma sequência especial de bits (*flag*) que possui o seguinte padrão: 01111110
 - ◆ Bits são transmitidos de forma transparente:
 - ▶ Transmissor: ao encontrar cinco bits 1 consecutivos insere um bit 0
 - ▶ Receptor: ao receber cinco bits 1 seguido de um bit 0 remove o bit 0 (*bit stuffing*)

Soluções para o Enquadramento

4. *Flags* iniciais e finais, com inserção de bits

Dados originais

0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0

Dados transmitidos

0 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 0 0 1 0

Bits inseridos



Dados armazenados na memória do receptor

0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0

Soluções para o Enquadramento

4. *Flags* iniciais e finais, com inserção de bits
- ◆ O limite entre dois quadros pode ser reconhecido sem qualquer tipo de ambiguidade pelo padrão de *flags*
 - ◆ Se o receptor perder o controle de onde estão os dados, bastará varrer a entrada em busca de sequências de *flags*, pois elas nunca ocorrem dentro dos dados, apenas nos limites dos quadros

Soluções para o Enquadramento

- 5. Violação de codificação da camada física
 - ◆ Método é baseado numa característica da camada física
 - ◆ O início e fim de quadro são determinados por um código de transmissão inválido
 - ◆ Usado no padrão IEEE 802

Soluções para o Enquadramento

- 5. Violação de codificação da camada física
 - ◆ LANs codificam bit 1 por um par alto-baixo e o bit 0 pelo par baixo-alto
 - ◆ Todo bit de dados tem uma transição intermediária
 - ◆ A combinação alto-alto e baixo-baixo não são usadas para dados e, por isso, podem ser empregadas na delimitação de quadros

Exercícios

5. A codificação de caracteres a seguir é usada em um protocolo de enlace de dados: A: 01000111, B: 11100011, FLAG: 01111110, ESC: 11100000. Mostre a sequência de bits transmitida (em binário) para o quadro de quatro caracteres: A B ESC FLAG quando é utilizado cada um dos métodos de enquadramento a seguir:
- a) Contagem de caracteres:
 - b) Bytes de *flag* com inserção de bytes:
 - c) Bytes de *flag* no início e no fim, com inserção de bits:

Exercícios

6. Um string de bits 0111101111101111110 precisa ser transmitido na camada de enlace de dados. Qual é o string realmente transmitido após a inserção de bits?

Detecção e Correção de Erros

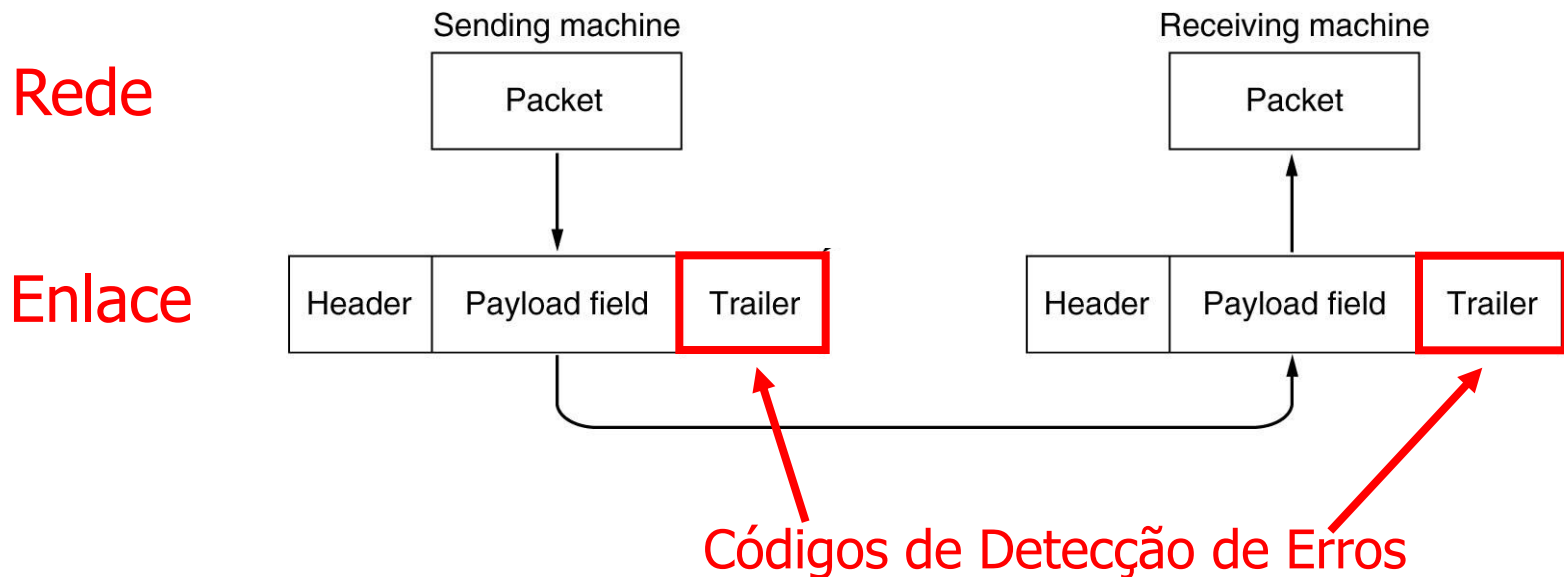
- Duas abordagens principais:
 - ◆ Incluir informações redundantes suficientes para que o receptor seja capaz de deduzir quais devem ter sido os dados transmitidos (**Códigos de Correção de Erros**)
 - ◆ Incluir informações redundantes suficientes apenas para permitir que o receptor deduza que houve um erro, mas sem identificar qual, para que o pacote seja descartado (**Códigos de Detecção de Erros**)

Detecção e Correção de Erros

- Códigos de detecção de erros:
 - ◆ Usados em canais altamente confiáveis, como as fibras
 - ◆ O bloco defeituoso é retransmitido
- Códigos de correção de erros:
 - ◆ Usados em canais como enlaces sem fio que geram muitos erros

Detecção de Erros

- São acrescentados na parte final do quadro



Códigos de Detecção de Erros

- Código de verificação de redundância cíclica (*cyclic redundancy check* – CRC) ou código polinomial
 - ◆ As *strings* de bits são representações de polinômios com coeficientes 0 e 1 apenas
 - ◆ Um quadro de m bits é considerado um polinômio com m termos, variando desde x^{m-1} até x^0 (grau $m-1$)
 - ▶ Exemplo: 110001 representa o polinômio $x^5 + x^4 + x^0$

Códigos de Detecção de Erros

■ CRC

- ◆ O transmissor e o receptor devem concordar em relação ao polinômio gerador, $G(x)$, antecipadamente
- ◆ Para calcular o total de verificação de um quadro com m bits, que corresponde ao polinômio $M(x)$, o quadro deve ter mais bits do que o polinômio gerador

Códigos de Detecção de Erros

■ CRC

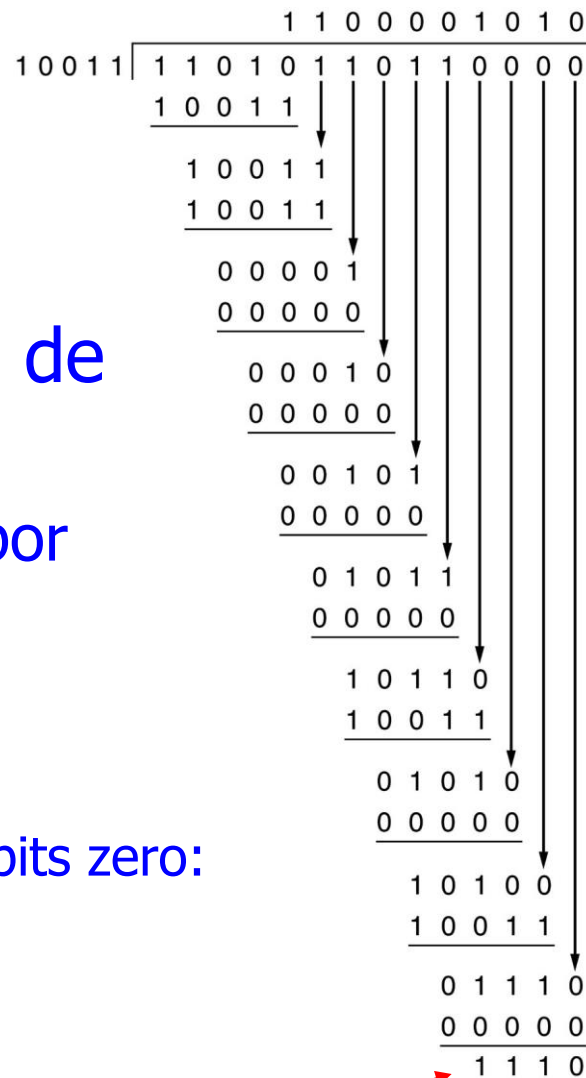
- ◆ Quadro verificado = quadro + total de verificação
- ◆ Acrescentar um total de verificação no final do quadro, de forma que o polinômio representado pelo quadro verificado seja divisível por $G(x)$
- ◆ Quando obtiver o quadro verificado, o receptor tentará dividi-lo por $G(x)$, a existência de um resto indica que houve um erro de transmissão

Códigos de Detecção de Erros

- Algoritmo para calcular o total de verificação
 - ◆ Seja r o grau de $G(x)$. Acrescente r bits zero à extremidade de baixa ordem do quadro, de modo que ele passe a conter $m+r$ bits e corresponda ao polinômio $x^r M(x)$
 - ▶ Quadro: 1101011011
 - ▶ Gerador: 10011
 - ▶ Mensagem após o acréscimo de 4 bits zero:
11010110110000

Códigos de Detecção de Erros

- Algoritmo para calcular o total de verificação
 - ◆ Divida a string de bits $x^r M(x)$ por $G(x)$
 - ▶ Quadro: 1101011011
 - ▶ Gerador: 10011
 - ▶ Mensagem após o acréscimo de 4 bits zero: 11010110110000



Resto

Códigos de Detecção de Erros

- Algoritmo para calcular o total de verificação
 - ◆ Some o resto (que tem sempre r ou menos bits) ao string de bits correspondente a $x^r M(x)$
 - ◆ O resultado é o quadro verificado que deverá ser transmitido
 - ▶ Quadro: 1101011011
 - ▶ Gerador: 10011
 - ▶ Mensagem após o acréscimo de 4 bits zero:
11010110110000
 - ▶ Quadro verificado: 11010110111110

Códigos de Detecção de Erros

- Algoritmo para calcular o total de verificação:
 - ◆ Quando obtiver o quadro verificado, o receptor tentará dividi-lo pelo polinômio gerador
 - ◆ A existência de um resto indica que houve um erro de transmissão
 - ◆ Se houve erro de transmissão, o pacote é descartado

Exercícios

7. Qual é o resto obtido pela divisão módulo 2 de x^7+x^5+1 pelo polinômio gerador x^3+1 ?
8. Um fluxo de bits 10011101 é transmitido com a utilização do método de CRC padrão descrito no texto. O polinômio gerador é x^3+1 . Mostre o string de bit real transmitido. Suponha que o terceiro bit a partir da esquerda seja invertido durante a transmissão. Mostre que esse erro é detectado na extremidade receptora.

Exercícios

9. Os protocolos de enlace de dados quase sempre colocam o CRC em um final, em vez de inseri-lo no cabeçalho. Por quê?

Subcamada de Controle de Acesso ao Meio (MAC)

Problema de Alocação de Canais

- Protocolos para
 - ◆ Canais difusão, ou
 - ◆ Canais de acesso múltiplo (multiacesso), ou
 - ◆ Canais de acesso aleatório
- Problema básico a ser resolvido:
 - ◆ Como "gerenciar" o acesso a canais difusão
- Protocolos responsáveis por fazer esse gerenciamento:
 - ◆ Protocolos de acesso ao meio, ou *Medium Access Control* (MAC)
- Sub-camada MAC está presente em quase todas as LANs

Problema de Alocação de Canais

- Problema:
 - ◆ Como alocar um único canal difusão entre vários usuários?
- Duas classes de algoritmos:
 - ◆ Alocação estática
 - ◆ Alocação dinâmica

Alocação Estática de Canais

- FDM (*Frequency Division Multiplexing*)
 - ◆ Se existem N usuários, a largura de banda é dividida em N partes do mesmo tamanho e a cada usuário será atribuída uma parte
- FDM é a forma tradicional quando:
 - ◆ Existe um número pequeno e fixo de usuários
 - ◆ Cada um possui um tráfego pesado

Alocação Estática de Canais

- Outro cenário:
 - ◆ Grande número de estações
 - ◆ Esse número varia ao longo do tempo
 - ◆ Tráfego é em rajadas
- Normalmente, FDM não é a solução:
 - ◆ Sub-canais ficam ociosos quando não há nada a transmitir
 - ◆ Em sistemas de computação, o tráfego é tipicamente em rajadas

Alocação Dinâmica de Canais

- Estações:
 - ◆ Existem N estações independentes que geram quadros a serem transmitidos
 - ◆ A estação fica bloqueada até o quadro ser totalmente transmitido

Alocação Dinâmica de Canais

- Único canal de comunicação:
 - ◆ Todas estações compartilham um único canal de comunicação para transmissão e recepção
 - ◆ Do ponto de vista de hardware, as estações são equivalentes
 - ◆ Do ponto de vista de software, as estações podem ter prioridades

Alocação Dinâmica de Canais

- Colisões:
 - ◆ A transmissão “simultânea” de dois ou mais quadros por estações diferentes causa uma colisão
 - ◆ Estações são capazes de detectar colisões
 - ◆ Quadros envolvidos em colisões devem ser transmitidos mais tarde

Alocação Dinâmica de Canais

- Política de transmissão de quadros ao longo do tempo:
 - ◆ Tempo contínuo (*continuous time*): qualquer instante
 - ◆ Tempo segmentado (*slotted time*): o tempo é dividido em intervalos discretos (*slots*), as transmissões de quadros sempre começam no início de um *slot*

Alocação Dinâmica de Canais

- Detecção de portadora para transmissão de quadro:
 - ◆ Com detecção (*carrier sense*): as estações conseguem detectar se o canal está sendo usado antes de tentarem utilizá-lo
 - ◆ Sem detecção (*no carrier sense*): as estações não conseguem detectar se o canal está sendo usado antes de tentarem utilizá-lo

Protocolos da Camada de Enlace de Redes Difusão

Protocolos de Acesso Múltiplo

- Aloha:
 - ◆ Puro, *Slotted*
- CSMA:
 - ◆ Persistente, não-persistente
 - ◆ Com detecção de colisões

Aloha

- Princípio:
 - ◆ Usuários transmitem quando têm dados a serem enviados
- Haverá colisões:
 - ◆ Serão detectadas
 - ◆ Deve-se esperar um tempo aleatório antes de tentar transmitir novamente

Aloha

User

A



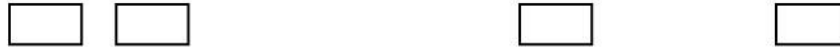
B



C



D



E

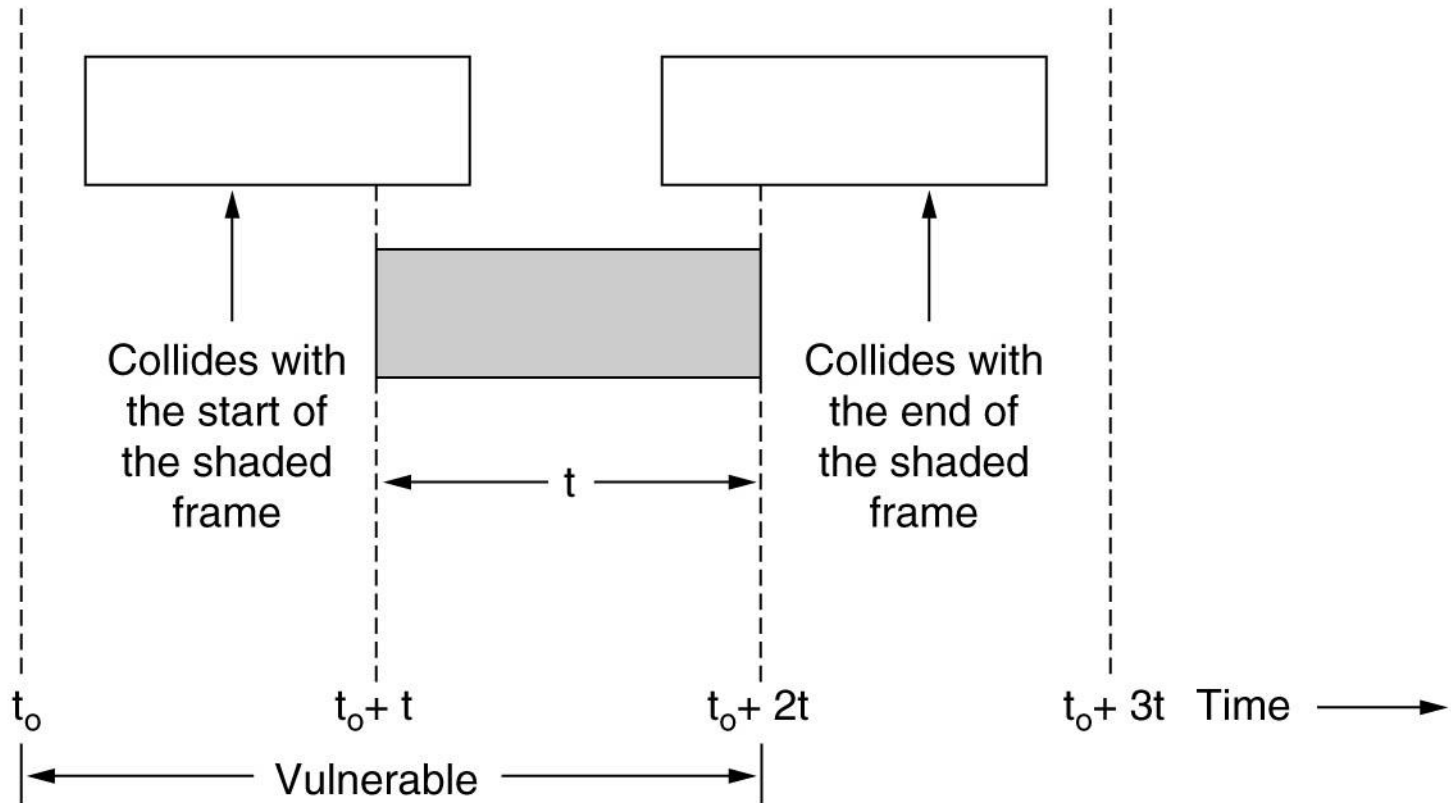


Time →

Aloha

- Se o primeiro bit de um quadro se sobrepor apenas ao último bit de um quadro quase terminando, os dois quadros serão totalmente destruídos e terão de ser retransmitidos posteriormente
 - ◆ O total de verificação não consegue fazer distinção entre uma perda total e uma perda parcial

Aloha



- Final dos quadros gerados entre t_0 e $t_0 + t$ irá colidir com o início do quadro sombreado
- Início dos quadros gerados entre $t_0 + t$ e $t_0 + 2t$ irá colidir com o final do quadro sombreado

Colisões no Aloha

- Tempo de quadro representa o período de tempo necessário para transmitir o quadro padrão de comprimento fixo
- Um quadro irá colidir com quadros gerados em um intervalo de tempo de 2 tempos de quadro
- Intervalo de vulnerabilidade = 2 tempos de quadro

Slotted Aloha

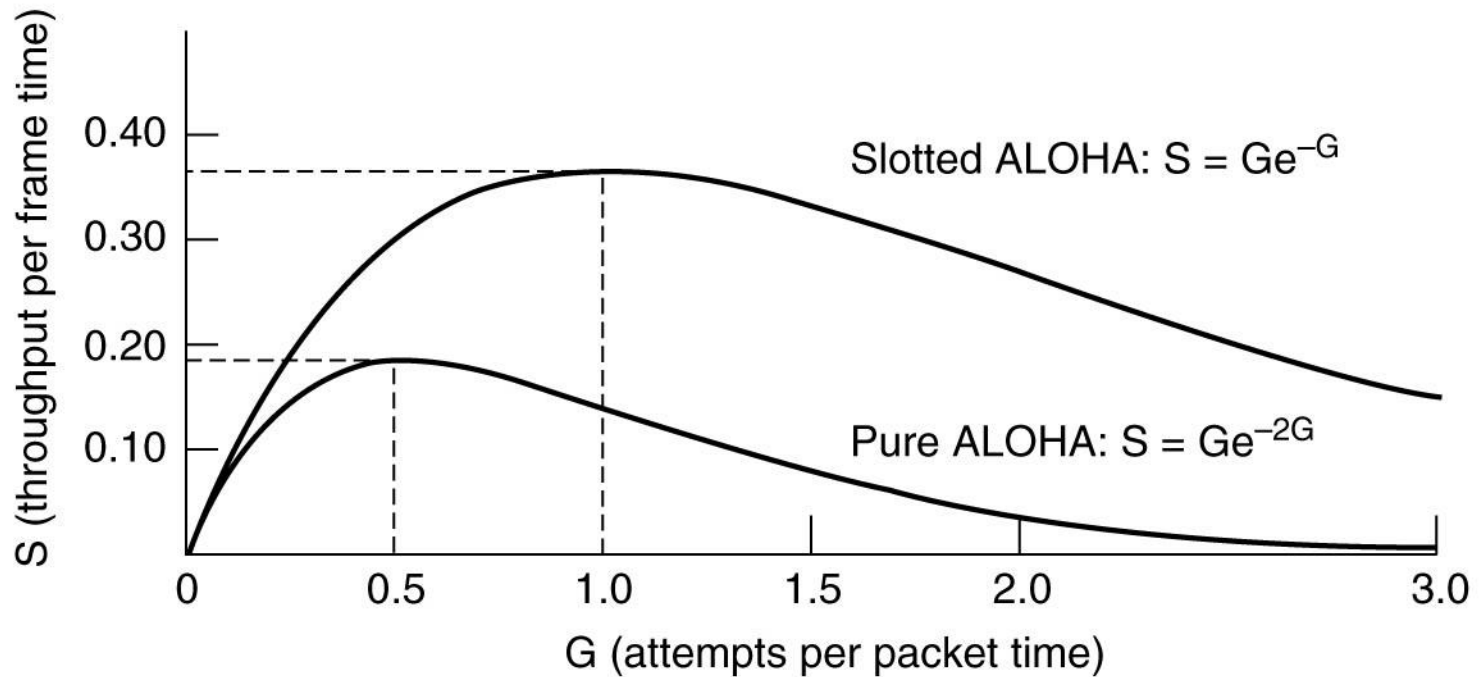
■ Princípio:

- ◆ Dividir o tempo em intervalos discretos, onde cada intervalo corresponde a um quadro
- ◆ Usuários devem ser capazes de identificar os limites desses intervalos:
 - ▶ Uma estação especial poderia emitir um sinal no início de cada intervalo

Colisões no Slotted Aloha

- Um quadro irá colidir com quadros gerados em um intervalo de tempo de 1 tempo de quadro
- Intervalo de vulnerabilidade = 1 tempo de quadro

Slotted Aloha



Exercício

10. Compare o retardo do ALOHA puro com o do *slotted* ALOHA com uma carga mínima (por exemplo, apenas uma estação usando o canal). Qual deles é menor? Explique sua resposta.

Exercício

11. Suponha um enlace de comunicação que utiliza CRC com o polinômio gerador $x^8+x^5+x^4+x^2+1$ que recebe duas mensagens:

10011001111010101111000

10011001111010101110000

1. Qual é o tamanho do total de verificação deste enlace? Justifique.
2. Estas mensagens contêm erros? Justifique.
3. Encontre a mensagem original (sem o total de verificação) das mensagens sem erro.
4. Para as mensagens que não contêm erros, apresente um erro que não seria detectado pelo CRC.

Protocolos CSMA

- Protocolos CSMA (*Carrier Sense Multiple Access*): protocolos de acesso múltiplo com detecção de portadora
- Três tipos básicos:
 - ◆ 1-persistente
 - ◆ não-persistente
 - ◆ p-persistente

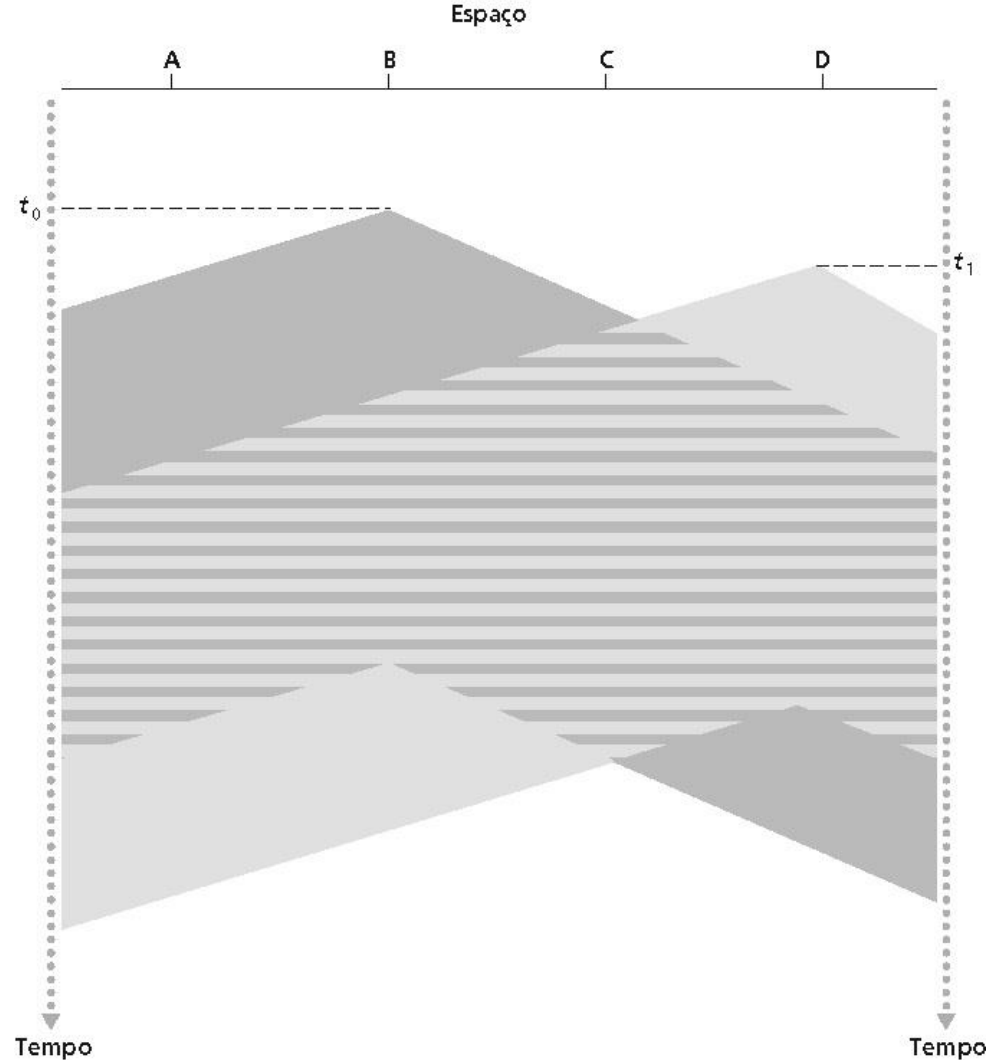
Protocolos CSMA 1-persistente

- Princípio do 1-persistente:
 - ◆ Uma estação ao desejar transmitir escuta o canal
 - ◆ Se estiver ocupado espera até ficar livre
 - ◆ Transmite o quadro quando o canal fica livre
 - ◆ Se ocorre uma colisão, a estação espera um tempo aleatório e começa o processo todo novamente

Protocolos CSMA 1-persistente

- É chamado 1-persistente porque sempre transmite ao verificar que o canal está desocupado, ou seja,
 - ◆ A probabilidade de transmitir ao encontrar o canal livre é 1
- Quando as colisões irão ocorrer?

Colisões no CSMA



- Quanto mais longo for o atraso de propagação, maior será a chance de um nó que detecta portadora ainda não perceber uma transmissão que já começou em outro nó da rede

Protocolos CSMA Não-persistentes

- Similar ao 1-persistente
- Diferença:
 - ◆ Ao verificar que o canal está ocupado espera um período de tempo aleatório e começa o processo novamente
 - ◆ Método menos guloso que tem um desempenho melhor que o 1-persistente

Protocolos CSMA

p-persistente

- É usado em canais com *slots*
- Princípio do p-persistent:
 - ◆ Estação escuta o canal
 - ◆ Se livre, transmite com probabilidade p
 - ◆ Senão, espera até o próximo slot ($P=1-p$)
 - ◆ Repete o processo novamente
 - ◆ Se ocorre colisão, a estação espera um tempo aleatório e repete o processo

Comparação entre os CSMA

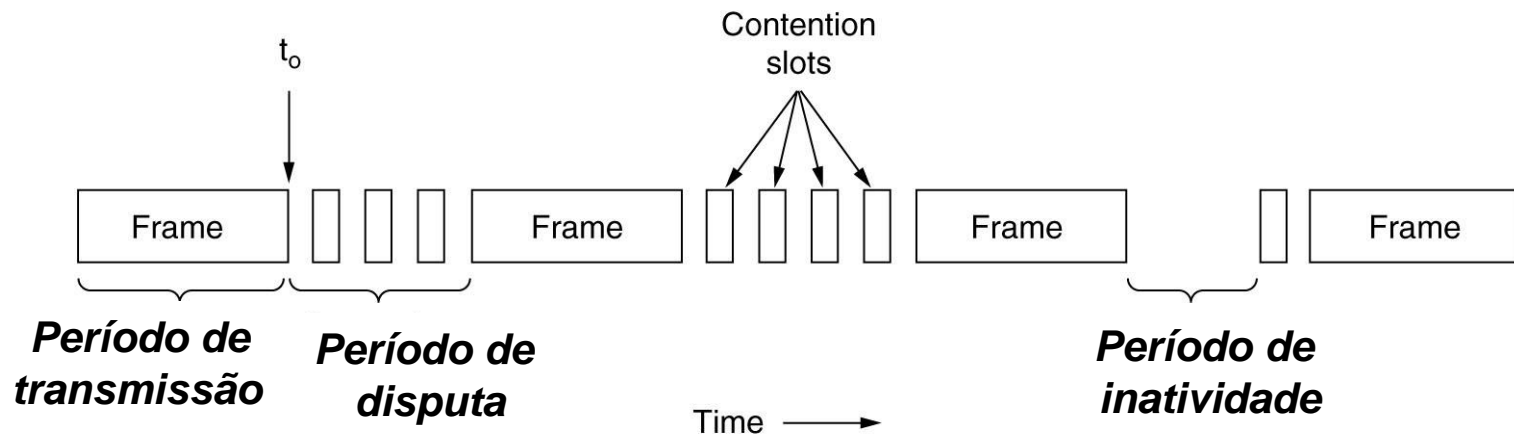
	1-persistente	não-persistente	p-persistente
Canal ocupado	Espera até que ele fique desocupado	Espera um tempo aleatório e começa o processo novamente	Espera até o próximo <i>slot</i>
Canal desocupado	Transmite um quadro	Transmite um quadro	Transmite com probabilidade p ; com $(1-p)$ espera até o próximo <i>slot</i>
Colisão	Espera tempo aleatório e começa o processo novamente	Espera tempo aleatório e começa o processo novamente	Espera tempo aleatório e começa o processo novamente

CSMA: sempre escuta o canal antes de transmitir

Protocolos CSMA/CD

- CD: *Collision Detection*
- Melhoria introduzida:
 - ◆ Uma estação ao detectar colisão para de transmitir imediatamente o quadro
 - ◆ Economiza tempo e BW
- CSMA/CD consiste em alternar períodos de contenção e transmissão

Protocolos CSMA/CD



Protocolos CSMA/CD

- Questão importante:
 - ◆ Quanto tempo uma estação deve esperar para saber se houve uma colisão ou não?
 - ▶ Duas vezes o tempo de propagação no cabo de ponta-a-ponta
- Conclusão importante:
 - ◆ Uma colisão não ocorre após esse período de tempo

Protocolos CSMA/CD

- Colisões afetam o desempenho do sistema principalmente em cabos longos e quadros curtos
- Foi padronizado como IEEE 802.3 (Ethernet)

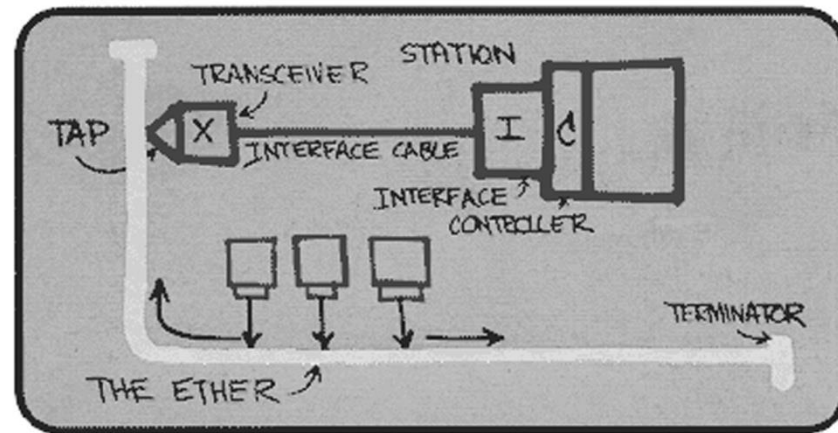
Exercício

12. Uma LAN CSMA/CD de 10 Mbps (não 802.3) com a extensão de 1 km tem uma velocidade de propagação de $200\text{m}/\mu\text{s}$. Não são permitidos repetidores nesse sistema. Os quadros de dados têm 256 bits, incluindo 32 bits de cabeçalho, totais de verificação e outras formas de overhead. O primeiro slot de bits depois de uma transmissão bem-sucedida é reservado para o receptor capturar o canal com o objetivo de enviar um quadro de confirmação de 32 bits. Qual será a taxa de dados efetiva, excluindo o overhead, se partirmos do princípio de que não há colisões?

Ethernet

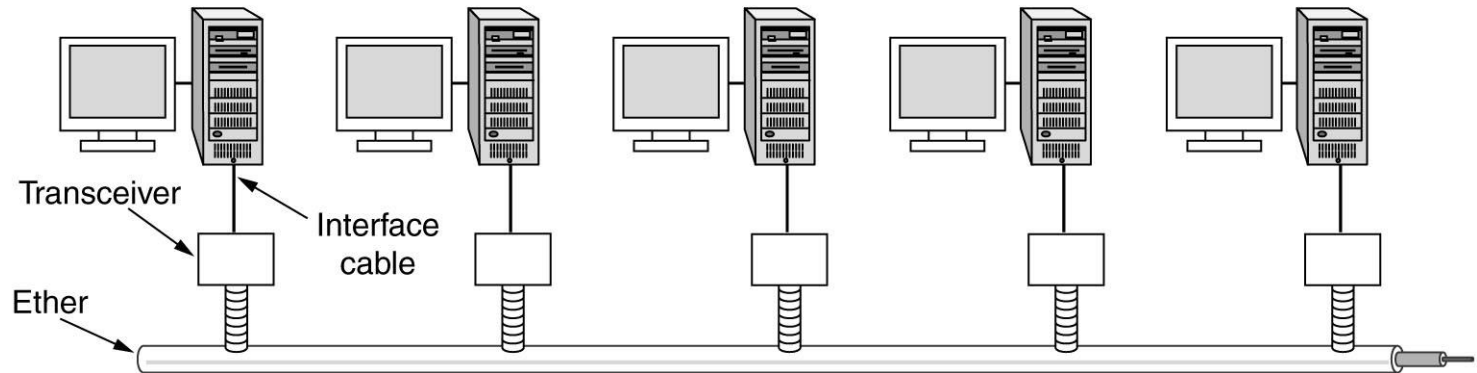
Ethernet

- A Ethernet foi implementada em 1976 por Metcalfe e Boggs no PARC (Palo Alto Research Center) da Xerox



Esboço da Ethernet por Bob Metcalfe

Ethernet



Topologia de barramento da Ethernet



Topologia em estrela da Ethernet

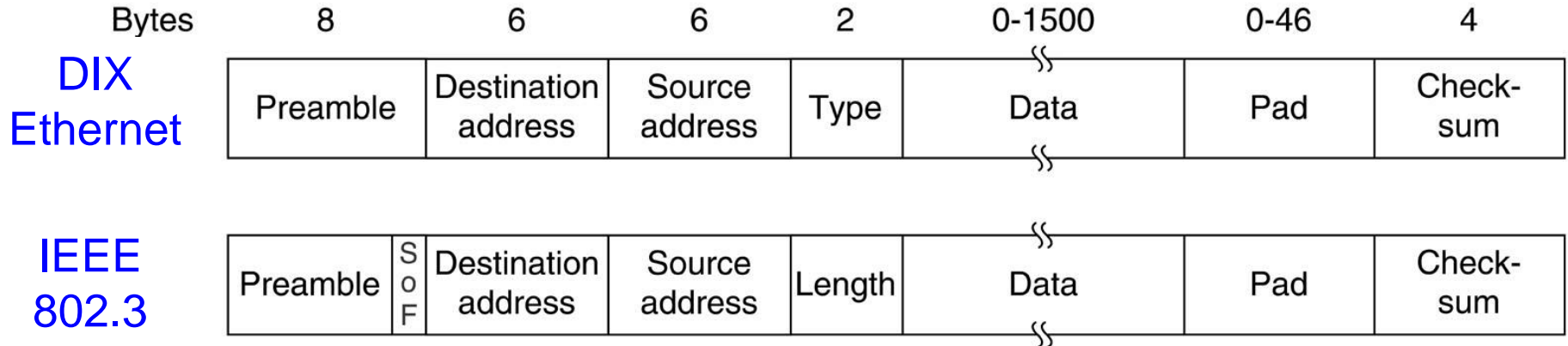
Ethernet

- Em 1978, a DEC, a Intel e a Xerox criaram um padrão para uma Ethernet de 10Mbps, chamado de **padrão DIX**
- Com pequenas alterações, o padrão DIX se tornou o padrão IEEE 802.3 em 1983
 - ◆ Padrão define uma família de redes CSMA/CD com velocidades de 1, 10, 100, 1000, 1000 Mbps em diferentes meios

Ethernet

- Funcionamento:
 - ◆ Estação escuta o canal antes de transmitir
 - ◆ Se estiver ocupado espera até ficar livre
 - ◆ Transmite o quadro se o canal estiver livre
 - ◆ Se ocorre uma colisão, a estação espera um tempo aleatório e começa o processo todo novamente

Ethernet: Quadro



■ Preâmbulo:

- ◆ usado para sincronização entre transmissor e receptor
- ◆ 7 bytes 10101010 e 1 byte 10101011

Protocolo Ethernet

- Endereço:
 - ◆ Endereço LAN do adaptador da origem e do destino
- Tipo:
 - ◆ Identifica o protocolo da camada de rede que deve receber o pacote
 - ◆ Permite que a Ethernet “multiplexe” os protocolos da camada de rede
- Comprimento:
 - ◆ Número de bytes do campo de dados

Protocolo Ethernet

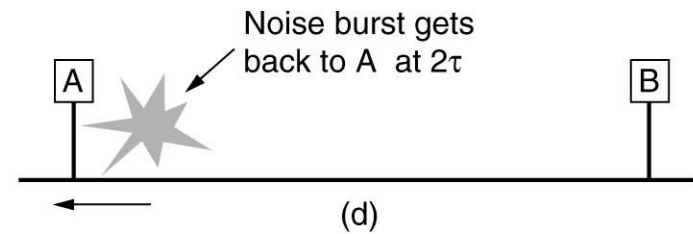
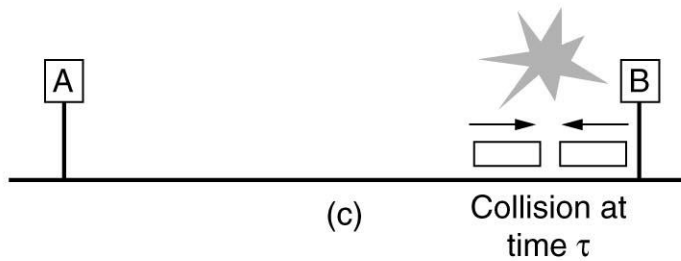
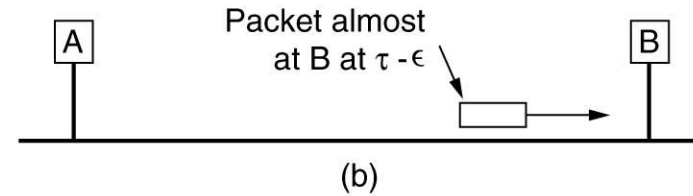
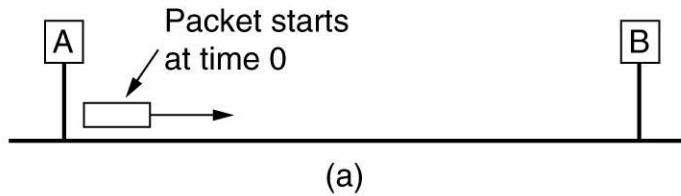
- Dados:
 - ◆ Carrega o datagrama IP
 - ◆ $46 \text{ bytes} \leq \text{Dados} \leq 1500 \text{ bytes}$

$64 \text{ bytes} \leq \text{tamanho total quadro} \leq 1518 \text{ bytes}$

Protocolo Ethernet

- Preenchimento (Pad):
 - ◆ Campo de dados deve ser ≥ 46
 - ◆ Caso contrário, *pad* = 46 – esse valor
 - ◆ Prevenir que uma estação termine de transmitir um quadro antes do primeiro bit chegar no extremo do cabo e ocorra uma colisão

Protocollo Ethernet



Protocolo Ethernet

- Por que 64 bytes?
- Para uma rede a
 - ◆ 10 Mbps,
 - ◆ comprimento máximo de 2500 metros, e
 - ◆ quatro repetidores
 - ◆ Tempo mínimo de transmissão = $50 \mu\text{s}$
 - ◆ Tamanho mínimo do quadro = 64 bytes

Protocolo Ethernet

- Total de verificação (*checksum*)
 - ◆ Utiliza o código CRC (*cyclic redundancy check*) para detecção de erros

Ethernet: Algoritmo de Espera

- CSMA/CD com recuo binário exponencial
 - ◆ Ao ocorrer uma colisão, as estações devem esperar (sortear) um intervalo de tempo de espera
 - ◆ Tempo é dividido em intervalos (slots) = $51,2 \mu\text{s}$

Ethernet: Algoritmo de Espera

- *Slots* de espera:

- ◆ Número inteiro no intervalo $[0 .. 2^c - 1]$, onde c é o número de colisões consecutivas
- ◆ Para c de 10 a 16 o nº max de *slots* é 1023
- ◆ Valor max de c é 16, quando a tentativa de transmitir é encerrada

Ethernet: Algoritmo de Espera

- Ausência de colisão não garante recepção correta
 - ◆ Pode ocorrer erro de *checksum*
- CSMA/CD não provê confirmação
- Forma simples e rápida de permitir confirmação:
 - ◆ Reservar o primeiro slot, após uma transmissão com sucesso, para o destinatário

Exercício

13. Considere a construção de uma rede CSMA/CD que funciona a 1 Gbps sobre um cabo de 1 km, sem repetidores. A velocidade do sinal no cabo é 200.000 km/s. Qual é o tamanho mínimo do quadro?