

COMUNICAÇÃO DE DADOS

UnB

Parte 4

Prof. Fabio Montoro
março 1995
r1 (1996)

4 CAMADA DE ENLACE

4.1 INTRODUÇÃO

A camada de enlace se encarrega da transferência da informação binária (dados) entre dois pontos da rede, passando pelos equipamentos de transmissão, meio de comunicação e recepção. A camada de enlace se preocupa com:

- Agrupar e transmitir os bits de dados em quadros;
- Possuir um esquema de detecção de erros na recepção;
- Controlar o fluxo dos dados;
- Fazer a arbitragem do meio de comunicação;
- Endereçar os quadros de forma apropriada.

4.2 FORMAÇÃO DOS QUADROS

A primeira função da camada de enlace é agrupar os dados em conjuntos de bits (quadros, blocos ou quadros) e transmiti-los. Em geral os autores chamam esse conjunto de “quadro”, quando se trata da camada de enlace. No entanto há literaturas que chamam de quadro, independentemente de que camada se trata, pois em geral essa nomenclatura diferenciada gera algumas dificuldades de interpretação

Ao longo deste trabalho chamarei de quadro, quando se tratar da camada 2.

O quadro, além dos dados propriamente ditos, possui campos com informações de controle, como por exemplo o campo de verificação de erros, normalmente chamado de FCS (*Frame Check Sequence*). O campo FCS é montado na origem e verificado no destino. Dessa forma o receptor pode saber se há erro no quadro recebido.

Normalmente o campo de dados é variável. Além dos dados, todos os demais campos são de controle.

A próxima tabela mostra os tamanhos dos quadros de alguns protocolos. Nos bytes de controle estou incluindo o preâmbulo inicial (um byte de flag), no caso dos protocolos delimitados por flags.

Tabela 4.1: Comprimento dos quadros de diversos protocolos

Protocolo	Dados [bytes]	Controle [bytes]
BSC (IBM)	0 a 254	6 a 15
SDLC (IBM)	16 a 4096	5
HDLC (X.25)	16 a 4096	5 ou 6
PPP	0 a 1500	4 a 7
Ethernet	46 a 1500	26
Frame Relay	1 a 8192	5 a 7

Na prática às vezes um protocolo executa funções de mais de uma camada, de forma integral ou mesmo parcial. Por exemplo: o protocolo BSC somente executa funções da camada de enlace, mas o protocolo Ethernet também executa funções da camada física. O protocolo X.25 executa funções das camadas física, de enlace e de rede.

4.3 BSC

Foi criado pela IBM por volta de 1960 para ser uma solução de comunicação de dados (textos) em redes de longa distância (WAN), semi-duplex, ponto-a-ponto ou multiponto, orientadas a caractere, entre uma controladora de terminais e uma controladora de comunicações (TCU = "Transmission Control Unit").

Conhecido como BSC, "Binary Synchronous Control", ou Bisync, esse protocolo é um clássico e vale a pena dedicar algumas linhas a ele. Continuou sendo utilizado nas redes IBM, mesmo após o lançamento posterior do SDLC.

Utiliza caracteres com significados especiais, ou seja, caracteres de controle.

Como o protocolo é orientado a caractere, para facilitar, a tabela 4.1 supõe que cada caractere tem 8 bits. Se, por acaso, for utilizado um código com 7 bits por caractere, por exemplo, a contagem por bytes fica prejudicada. A tabela 4.1 também considera um byte por caractere. A figura 4.1 mostra o quadro BSC e indica o tamanho de alguns campos em bytes.

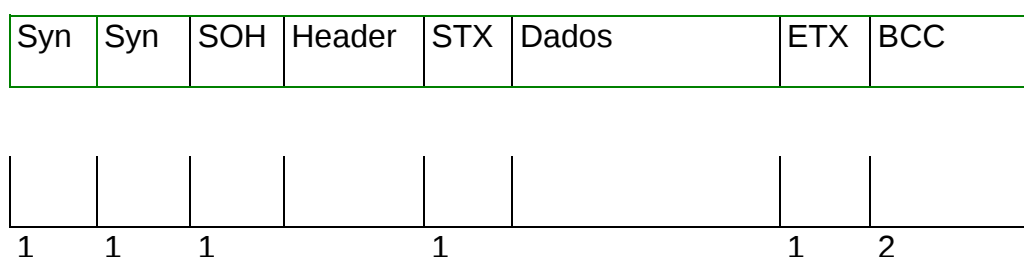


Fig. 4.1: Quadro BSC

O quadro começa com dois caracteres de sincronismo (Syn) para garantir que o receptor vai identificar corretamente o início do quadro. Em seguida vem um caractere de início de cabeçalho (SOH = Start Of Header), então vem o cabeçalho, depois um caractere de início do texto (STX = Start Of Text) e então os dados. Após os dados vem um caractere de fim de texto (ETX = End Of Text) e dois caracteres de verificação de integridade do quadro (BCC).

Esse é o típico quando há dados a transmitir. Alguns caracteres de controle utilizados em quadros especiais, são:

ACK = "Acknowledgement" = caractere enviado pelo receptor para informar que o quadro foi bem recebido.

NACK = "Negative ACK" = caractere enviado pelo receptor para informar que o quadro recebido não foi aceito (continha erros).

ENQ = "Enquiry" = caractere enviado por uma estação para saber se a outra estação deseja receber (ponto-a-ponto).

DLE = “Data Link Escape” = caractere transmitido antes de um caractere de controle para que este não seja interpretado como tal. Com auxílio desse caractere pode-se transmitir informações transparentes.

EOT = “End Of Transmission” = utilizado para indicar o fim de uma transmissão. Esse caractere também é utilizado para responder que não há nada a transmitir.

O procedimento básico desse protocolo, para transferir dados, é o stop-and-wait:

- Lado "A" transmite um quadro com os dados
- Lado "B" transmite um ACK se recebeu sem erros ou um NACK se recebeu com erros.
- Lado "A" retransmite o quadro recusado, se for o caso.

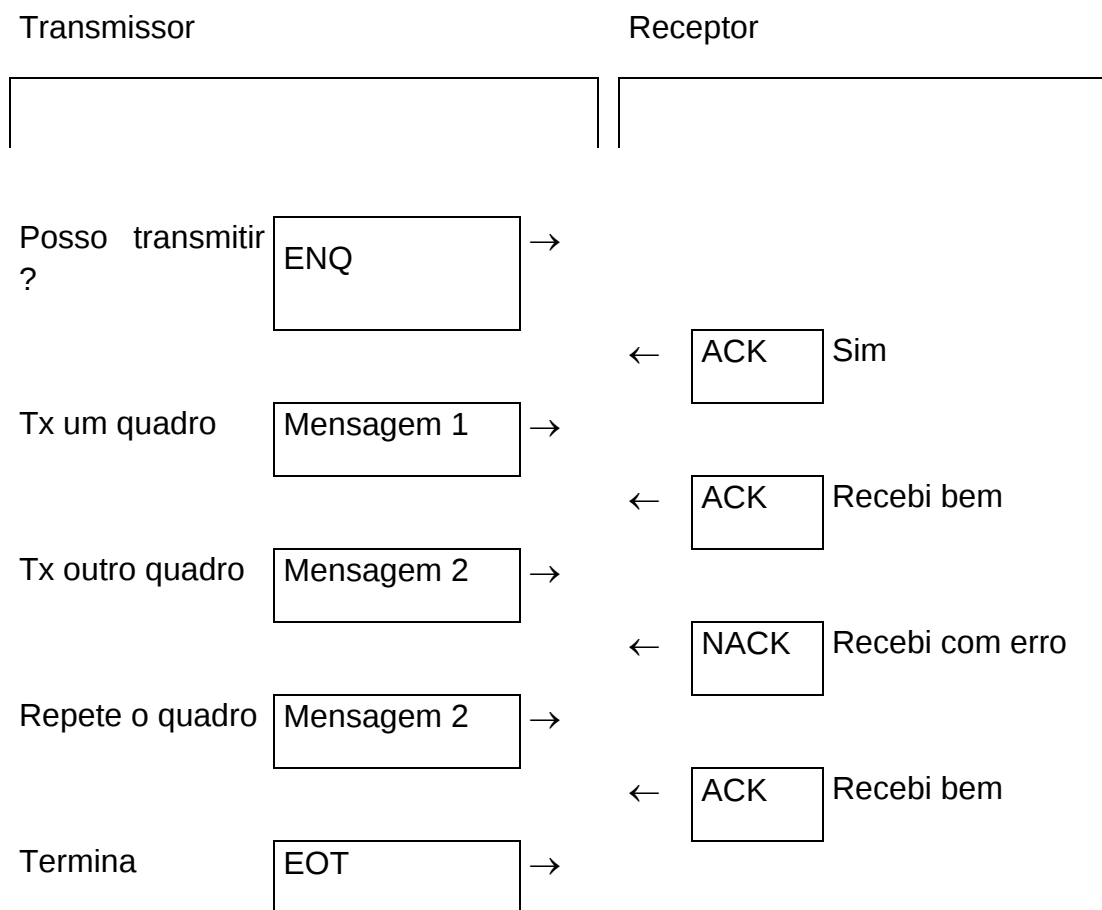


Fig. 4.2: Sequência de mensagens do protocolo BSC

4.4 SDLC

Lançado pela IBM em 1973, o protocolo "Synchronous Data Link Control" é orientado a bit. Seu quadro tem o formato apresentado a seguir. Entrar em mais detalhes sobre esse protocolo fugiria ao escopo deste livro.

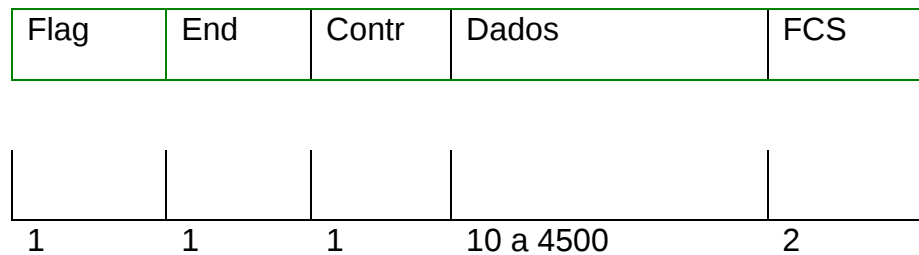


Fig. 4.3: Quadro SDLC

4.5 HDLC

Publicado como norma ISO TC97 em 1979, o protocolo "High-level Data Link Control" foi baseado no SDLC da IBM e possui um quadro similar a este.

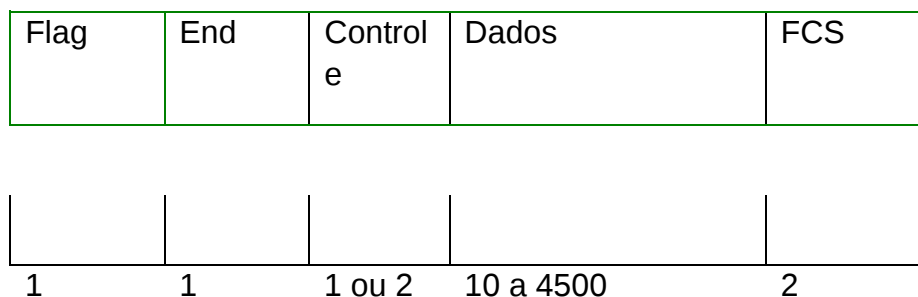


Fig. 4.4: Quadro HDLC

É um dos protocolos de enlace mais populares dentre os utilizados em linhas seriais síncronas de redes WAN. Foi adotado, por exemplo no X.25 (chamado de LAP-B) e na ISDN (chamado de LAP-D).

O fabricante Cisco possui uma versão personalizada do protocolo HDLC1, na qual foi introduzido um campo após o campo de controle, para identificar qual é o protocolo da camada L3 que está encapsulado. A Cisco

¹ CCNA Intro Exam Certification Guide, Wendell Odom, Cisco Press, 2004, pág 91.

utiliza os mesmos códigos de protocolos utilizados no campo “type” do quadro Ethernet.

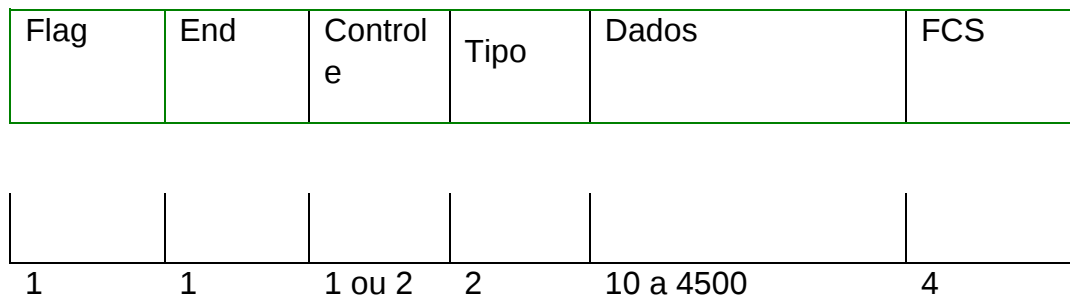


Fig. 4.5: Quadro HDLC proprietário Cisco

4.6 PPP

O protocolo PPP ("Point to Point Protocol") faz parte da comunidade Internet2 e está definido na RFC 1661. Foi idealizado principalmente para encapsular pacotes IP, mas pode encapsular qualquer outro protocolo e transmitir em um enlace serial, que pode ser assíncrono com caracteres de 8 bits sem paridade ou síncrono orientado a bit. O quadro foi baseado no HDLC e também é delimitado por flags.

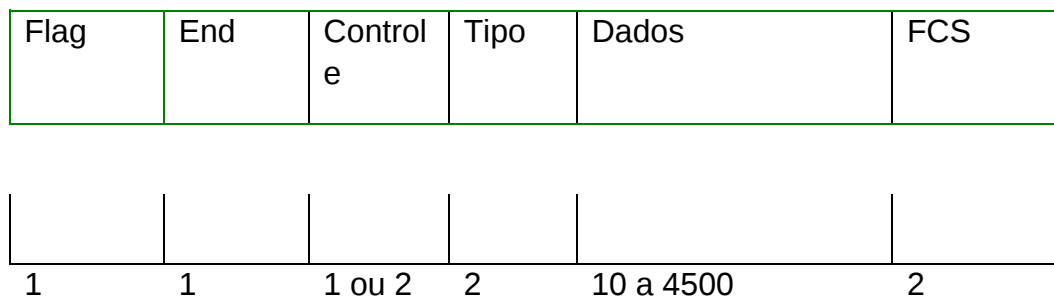


Fig. 4.6: Quadro PPP

Possui um procedimento no estabelecimento da conexão, onde são negociados diversos parâmetros, tal como a supressão ou não dos campos "Controle" e "Endereço" a fim de aumentar a eficiência. O protocolo também permite monitorar a qualidade da conexão.

² IETF – Internet Engineering Task Force, representa a comunidade Internet através de grupos de trabalho responsáveis pela elaboração de documentos chamados RFC (Request For Comments) que são os padrões utilizados na Internet.

Uma característica interessante do PPP é que ele permite várias conexões de camada 3 em uma mesma conexão de camada 2, ou seja, em um mesmo enlace.

Mais detalhes sobre o PPP fugiria do escopo deste curso.

4.7 ETHERNET

Este protocolo será abordado com mais detalhes nos capítulos 5 e 6 adiante. A figura 4.7 mostra o formato do quadro Ethernet.

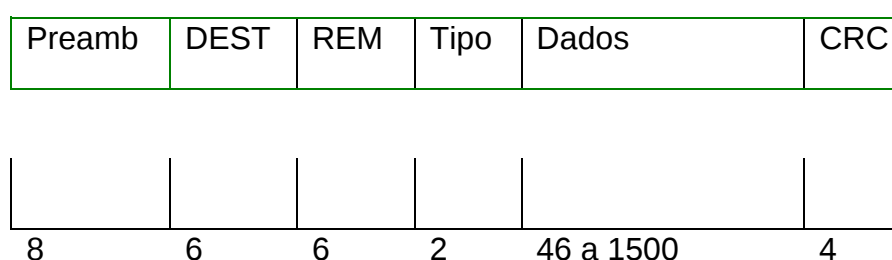


Fig. 4.7: Quadro Ethernet

4.8 FRAME RELAY

Frame Relay é um protocolo cuja aplicação principal é operar em uma rede de comutação de pacotes. Apresenta uma interface que define as regras de acesso à rede. Por isso o Frame Relay é apontado por alguns autores como uma interface lógica ou um protocolo de interface.

O Frame Relay oferece um serviço não confiável de transmissão de quadros porque não possui correção de erros nem controle de congestionamento.

Surgiu da necessidade de um protocolo mais apropriado a canais que apresentassem baixa taxa de erro, como os que estavam surgindo, como os enlaces em fibra ótica ou as redes digitais, por exemplo, onde se pode transmitir em velocidades acima de 10 Mbps com baixa taxa de erro, contra o cenário anterior onde os canais operavam até 64 kbps com taxa de erro elevada. Em ambientes que apresentavam alta taxa de erro, como as linhas telefônicas comuns, um protocolo robusto, com correção de erros, como o X.25, era o mais indicado. No entanto, em uma situação de baixa taxa de erro o

X.25 apenas sobrecarrega a comunicação com controles dispensáveis, introduzindo excesso de overhead.

O Frame Relay é bem mais simples que o X.25.

Os três protocolos mais comuns, para redes de comutação de quadros, são o X.25, o Frame Relay e o ATM.

Tabela 4.2: Protocolos para redes comutadas

Parâmetro	X.25	Frame Relay	ATM
Quadro	variável	variável	fixo
Campo de dados	até 4096 bytes	até 8192 bytes	48 bytes
Velocidade alvo	até 64 kbps	até 45 Mbps	até 10 Gbps
Correção erro	sim	não	não

A figura 4.8 mostra o formato do quadro Frame Relay.

Flag	Endereço	Dados	FCS	Flag
1	2-4	0-8192	2	1

Fig. 4.8: Quadro Frame Relay

- Flag = sequência delimitadora de quadro = 01111110 (7Eh)
- Endereço = contém o identificador do enlace e outras informações
- Dados = pode ter até 8192 bytes
- FCS = verificação de integridade (Frame Check Sequence)

4.8.1 ENDEREÇAMENTO FRAME RELAY

O campo endereço pode ter 2, 3 ou 4 bytes. Seu tamanho é determinado pelos bits EA (últimos à direita em cada byte). O campo endereço, no protocolo Frame Relay, possui funções adicionais de sinalização. Dentro do campo endereço há sub-campos, conforme relacionados a seguir e ilustrados na figura 4.9, que mostra as três alternativas de comprimento do DLCI: cada linha é um byte e cada bit com significado especial está delimitado.

DLCI = campo Data Link Control Identifier (identificador do enlace), composto de duas partes que totalizam 10 bits;

C/R = bit indicador Comando/Resposta;

EA = bit de extensão do endereço (0 --> tem mais um byte);

BECN = bit “Backward Explicit Congestion Notification”;

FECN = bit “Forward Explicit Congestion Notification”;

DE = bit “Discard Elected”;

D/C = bit que indica se os 6 últimos bits são DLCI (D/C= 0) ou Controle (D/C= 1)

DLCI			C/R	0
DLCI	Fecn	Becn	DE	1

DLCI			C/R	0
DLCI	Fecn	Becn	DE	0
DLCI ou controle			D/C	1

DLCI			C/R	0
DLCI	Fecn	Becn	DE	0
DLCI				0
DLCI			D/C	1

Fig. 4.9: Alternativas de endereçamento no Frame Relay

Notas:

- Não há sinalização in-band. Não há campo de controle para isso;
- Só há um tipo de quadro: dados;
- O quadro de gerenciamento LMI utiliza um endereço DLCI especial;
- Não há número sequencial de quadros, logo, não há controle de erros;
- Não há controle de fluxo.

4.8.2 NUVEM FRAME RELAY

É um conjunto de equipamentos comutadores interligados entre si, onde qualquer acesso a essa nuvem deve ser feito utilizando o protocolo Frame Relay. Cada comutador é um nó da rede. O protocolo entre os comutadores não precisa necessariamente ser o Frame Relay.

Qualquer equipamento que fale outro protocolo nativo, como por exemplo, SDLC, HDLC, BSC, PPP, Ethernet, assíncrono, etc, para se ligar à nuvem vai precisar de um equipamento chamado **FRAD** (Frame Relay Access Device) que recebe o protocolo nativo, de um lado, e fala Frame Relay do outro.

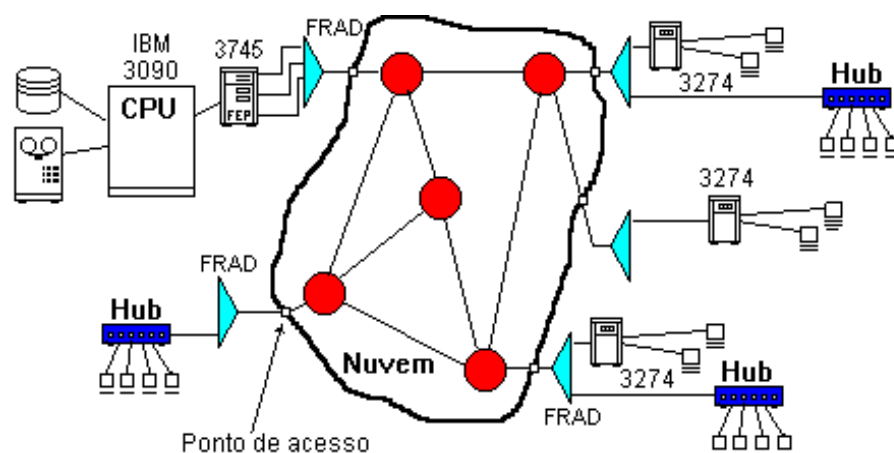


Fig. 4.10: Nuvem Frame Relay

O Frame Relay permite estabelecer circuitos virtuais comutados (SVC) e circuitos virtuais permanentes (PVC), sendo este último o mais comum.

Pode-se estabelecer um ou mais circuitos virtuais entre quaisquer dois pontos de acesso à nuvem e de um mesmo ponto de acesso podem sair vários circuitos virtuais para diversos outros pontos de acesso, ou seja, esse ponto pode estar "ligado" a vários outros.

Uma vez estabelecido o circuito virtual, pode haver tráfego de dados entre os dois pontos. Esses dados serão roteados pelos comutadores intermediários até chegarem ao ponto destino.

A nuvem Frame Relay, portanto, é um meio de transporte de dados, formatados segundo seus protocolos nativos, entre pontos diversos, chamados de pontos de acesso. Em cada ponto de acesso deve haver um FRAD.

A nuvem Frame Relay é montada em cima de canais de dados transparentes, tipo canais digitais em fibra ótica ou enlaces rádio, que podem ser operados por empresas de transporte de dados transparentes (Embratel, Teles, etc).

A nuvem Frame Relay também pode ser de propriedade da mesma empresa que opera o canal ou de uma prestadora de serviço de transporte em Frame Relay, como uma operadora pública de telecomunicações (Embratel, Teles, etc), ou ainda, ser privativa, ou seja, ser toda de propriedade de uma empresa que a utiliza para transporte de sua rede corporativa.

Uma das vantagens do Frame Relay é que uma nuvem privativa pode se ligar a uma nuvem pública.

4.8.3 ROTEAMENTO NO FRAME RELAY (SWITCHING)

Cada enlace lógico entre dois nós comutadores ou entre um nó e um equipamento FRAD, possui um número DLCI (Data Link Connection Identifier). Muitos autores e fabricantes chamam o comutador Frame Relay de switch Frame Relay.

Cada ligação física entre esses dispositivos pode ter vários enlaces lógicos, portanto, vários DLCIs.

Cada nó comutador monta uma tabela para cada porta física de entrada, que especifica em qual porta física e em qual DLCI deve sair o quadro.

A figura 4.11 mostra uma rede com 5 nós e 4 FRADs.

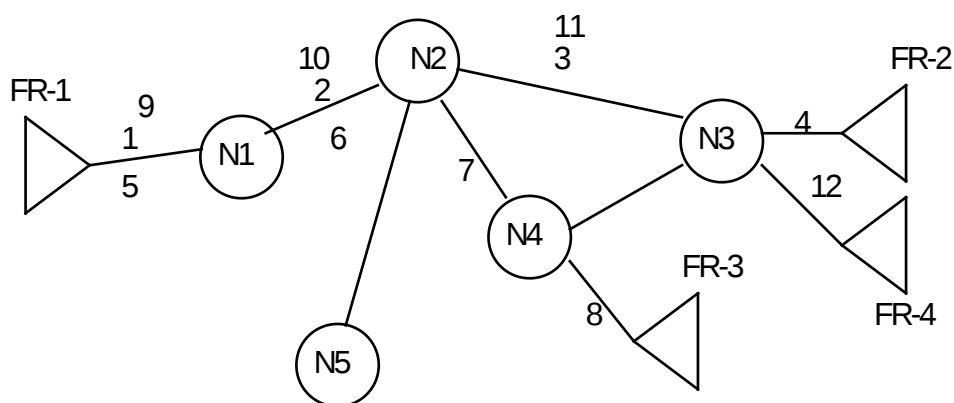


Fig. 4.11: Rede com comutadores Frame Relay

Vejamos as tabelas de roteamento dos nós:

Tabela 3.11: Tabelas de roteamento dos comutadores Frame Relay

PVC	Rota	N1	N2	N3	N4	N5
1	FR1-FR2	1-2	2-3	3-4	---	---
2	FR1-FR4	9-10	10-11	11-12	---	---
3	FR1-FR3	5-6	6-7	---	7-8	---

4.8.4 CAPACIDADE DE DLCIS

A capacidade de enlaces lógicos vai depender do tamanho do campo DLCI no quadro Frame Relay. Como o significado do DLCI é local, ou seja, para cada comutador, os números podem ser repetidos pela rede.

O DLCI pode ter:

10 bits ---> end = 2 bytes ou end = 3 bytes com D/C=1

16 bits ---> end = 3 bytes com D/C=0

17 bits ---> end = 4 bytes com D/C=1

23 bits ---> end = 4 bytes com D/C=0

A tabela a seguir ilustra as capacidades. Note que, para DLCI de 10 bits, podem existir, no máximo, 975 DLCIs diferentes.

Tabela 3.12: Capacidade das diversas alternativas de DLCI

10 bits	16 bits	17 bits	23 bits	Uso
0	0	0	0	Anexo D
1 a 15	1 a 1023	1 a 2047	1 a 131071	reservados
16 a 991	1024 a 63487	2048 a 126975	131072 a 4194303	DLCI
992 a 1022	63488 a 5534	126976 a 131070		reservados
1023	65535	131071		LMI

4.8.5 CONTROLE DE CONGESTIONAMENTO

O objetivo de um controle de congestionamento é evitar que as filas, nos nós de comutação, ultrapassem determinados limites, a fim de evitar perda dos dados e um eventual colapso do nó.

O controle de congestionamento é um desafio nas redes Frame Relay porque esse protocolo é simples e possui poucos recursos para isso. Seu principal objetivo é ser eficiente na transmissão.

Um nó Frame Relay não executa controle de fluxo.

O controle do congestionamento deve também ser assumido pelo usuário. O nó monitora o grau de congestionamento e informa ao usuário. O usuário limita o fluxo de seu tráfego da forma que achar melhor.

Há dois procedimentos básicos de controle de congestionamento:

- Prevenção de congestionamento
- Recuperação de uma situação de congestionamento

A figura 4.12 mostra que o retardo de transmissão dos quadros aumenta com o tráfego. A prevenção deve ser disparada quando o retardo atinge o ponto "A". No ponto "B" já se deve usar um procedimento de recuperação.

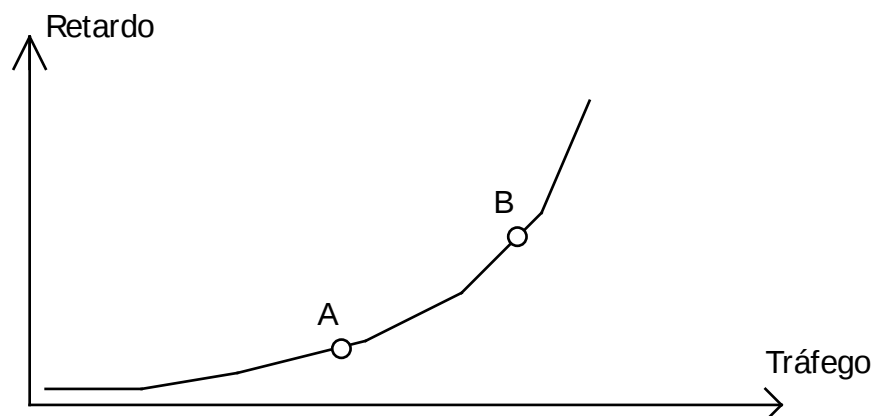


Fig. 4.12: Retardo em um comutador Frame Relay

4.8.6 PREVENÇÃO DE CONGESTIONAMENTO

No protocolo Frame Relay a prevenção do congestionamento ocorre por sinalização explícita, através dos bits BECN e FECN:

BECN = Um nó (sinalizador) envia um quadro com BECN=1 para notificar ao nó receptor do quadro (outro nó ou usuário) que pode iniciar seu procedimento de prevenção de congestionamento no sentido inverso do quadro recebido. Ou seja, os quadros que o nó sinalizado transmitir ao nó sinalizador

podem encontrar congestionamento. Ao receber um quadro com esse bit marcado o nó receptor deve manter o bit marcado ao retransmitir o quadro.

FECN = Similar ao anterior, indica ao nó sinalizado que pode iniciar seu procedimento de prevenção de congestionamento no sentido do quadro recebido, ou seja, o nó sinalizado vai receber quadros que vem de uma área congestionada. Da mesma forma o nó receptor deve manter o bit marcado ao passar o quadro.

Tanto o FECN quanto o BECN são marcados pelo nó de rede, não pelo usuário. Frequentemente esses bits são ignorados pelos FRADs, ou somente contados a fim de fornecer informação sobre o nível de congestionamento da rede.

O nó de comutação Frame Relay fica monitorando suas filas, ou seja, seus buffers de saída de quadros. O processador do nó mede o comprimento de cada fila de saída e inicia o cálculo quando a fila fica vazia:

a - fila está vazia

b - após um tempo T mede a fila = F1

c - após outro tempo T mede novamente = F2

d - calcula:
$$\bar{F} = \frac{\sum_{n=1}^N T \cdot F_n}{NT}$$

e - se o valor encontrado ultrapassar um certo limite,
marca o bit FECN ou BECN.

Um comutador que receber um quadro excedendo Bc mas abaixo de Bc+Be, marca o bit DE, ficando o quadro eleito para descarte, o que pode ocorrer caso ele entre em uma região de congestionamento.

Se o comutador receber um quadro excedendo Bc+Be pode simplesmente descartar o quadro ou marcar seu bit DE (normalmente essa opção é configurável no comutador).

O comutador possui 3 limiares de congestionamento na fila de quadros a transmitir: médio, severo e total.

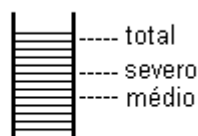


Fig. 4.13: Fila de saída de um comutador Frame Relay

O comportamento do comutador é dado pelo fluxograma apresentado na figura 4.14.

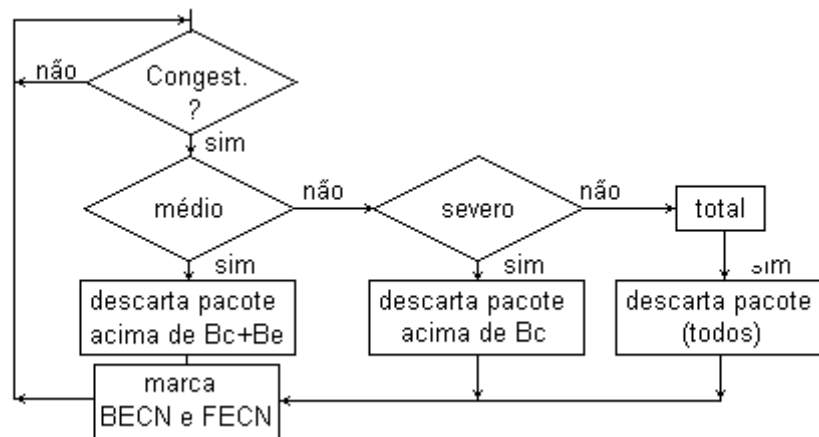


Fig. 4.14: Comportamento de um comutador Frame Relay

4.8.7 FRAME RELAY COMO UM SERVIÇO

As normas sobre Frame Relay introduziram os conceitos³ de CIR⁴, rajada de dados comprometida (Bc) e rajada de dados em excesso (Be).

O CIR é a taxa de transmissão de informação comprometida com o usuário (FRAD) a ser transmitida pelo respectivo nó. A taxa que exceder ao CIR pode ser descartada se houver congestionamento. Apesar de haver o comprometimento, o nó pode não conseguir dar vazão ao CIR. Quando isso ocorrer, o nó vai descartar primeiro os quadros que excederem ao CIR e depois aqueles eleitos pelo usuário.

A rajada de dados comprometida (Bc) é outro parâmetro do serviço Frame Relay, que define a quantidade máxima de dados que a rede suporta transferir, num determinado intervalo de tempo.

A rajada de dados em excesso (Be) é a quantidade de dados que a rede vai tentar transferir, além do Bc, em condições normais, num determinado intervalo de tempo.

As figuras 4.15 a 4.17 ilustram 3 situações distintas. Elas mostram a quantidade de bits transmitidos com o tempo. Indicam, também, que os parâmetros Bc e Be são determinados no tempo "T".

$$\text{Normalmente} \quad T = \frac{B_c}{CIR} = 1 \text{ s}$$

³ Ref. 15 págs. 2.1-2.10.

⁴ *Committed Information Rate*

A primeira situação mostra que o FRAD transmitiu abaixo do CIR, a segunda mostra que o FRAD ultrapassou o CIR e o Bc e na terceira ultrapassou até $(Bc+Be)$ - nesse caso os quadros que ultrapassaram foram descartados. A linha tracejada indica os bits efetivamente transmitidos do FRAD para o nó da rede.

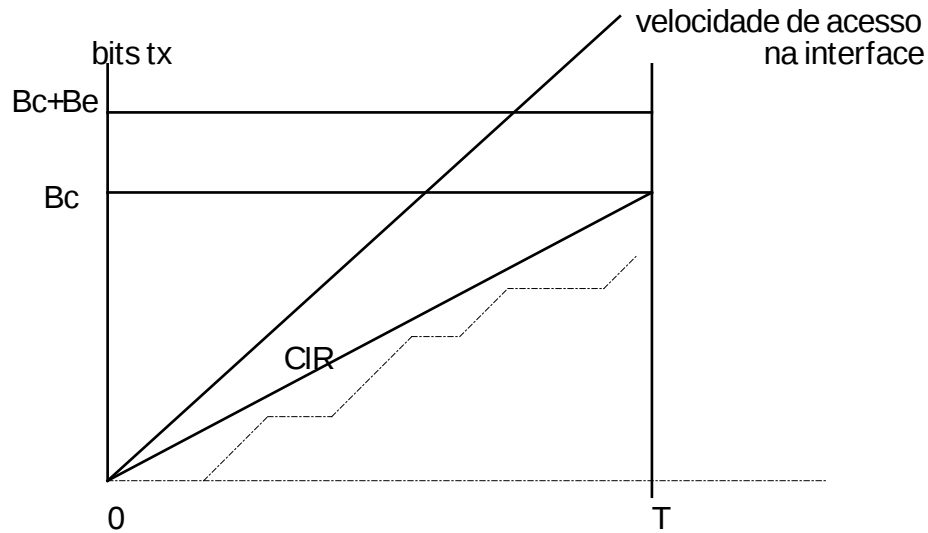


Fig. 4.15: Transmissões Frame Relay dentro do CIR

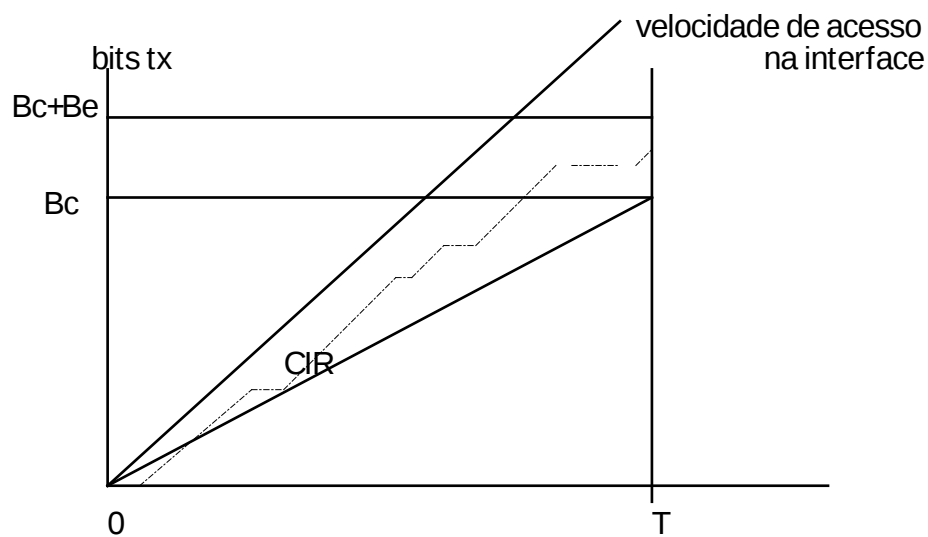


Fig. 4.16: Transmissões Frame Relay ultrapassando o CIR

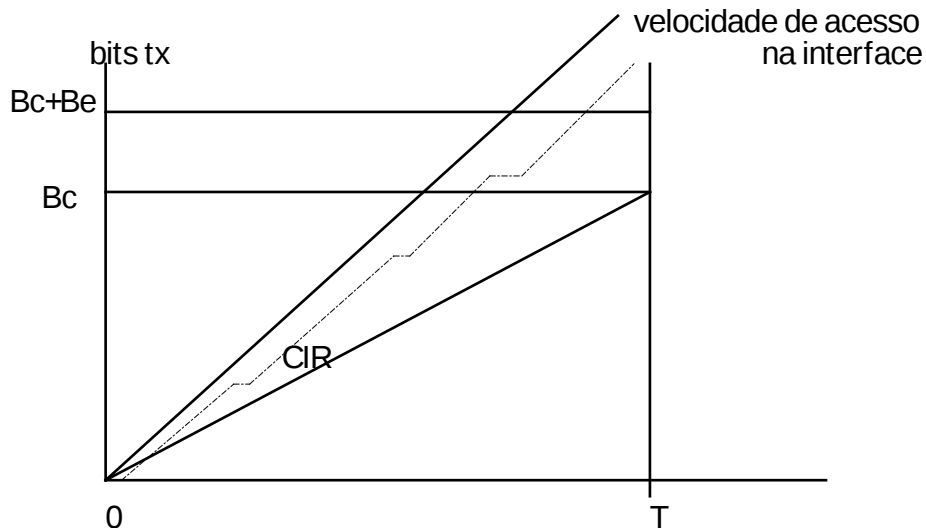


Fig. 4.17: Transmissões Frame Relay e um quadro eleito para descarte

4.8.8 CONFIGURAÇÃO LÓGICA DA INTERFACE FRAME RELAY

Sob o ponto de vista lógico, uma interface Frame Relay (FRI = “Frame Relay Interface”) pode ser configurada como ^{5 6 7}:

- UNI-DCE
- UNI-DTE
- NNI

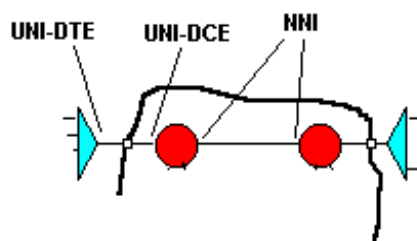


Fig. 4.18: Configurações das interfaces Frame Relay

A interface UNI-DCE se comporta como um provedor de informações de gerenciamento sobre a rede, para a interface UNI-DTE a ela conectada. Sob o

⁵ Ref. 14 págs. 2.24-2.29.

⁶ Ref. 15 págs. 8.6-8.18.

⁷ Ref. 44 págs. 11-20.

ponto de vista físico, a interface UNI-DCE pode ser configurada para fornecer ou receber o sincronismo.

A interface UNI-DTE tem o comportamento complementar da UNI-DCE, ou seja, é a interface do equipamento de acesso, que vai solicitar informações de gerenciamento da rede.

A interface NNI (“Network to Network Interface”) é utilizada entre dois comutadores (comutadores) da rede Frame Relay. Permite que duas redes se liguem por meio de um protocolo padronizado. Essa interface executa as funções tanto de UNI-DCE quanto de UNI-DTE, com relação ao gerenciamento.

4.8.9 GERENCIAMENTO LMI

O procedimento chamado LMI (“Link Management Interface”) permite ao FRAD obter informações de gerenciamento da rede. O FRAD envia um quadro LMI solicitando informações e o comutador responde, basicamente com dois tipos de relatórios: integridade do enlace e relatório do estado de todos os PVCs.

O processo de troca de quadros LMI entre FRAD e comutador é contínuo, semelhante a um polling, chamado de “heart beat” ou “keep alive”, monitorando constantemente a conexão. A ausência de quadros keepalive indica que a conexão caiu.

Há três tipos de procedimentos de gerenciamento, além de procedimentos proprietários:

- LMI original
- ANSI T1.617 Annex D
- ITU-T Q.933 Annex A
- Cisco (proprietário)

A especificação original LMI foi criada por um grupo de empresas americanas⁸. Os procedimentos Annex D e Annex A surgiram posteriormente e se incorporaram às especificações oficiais do protocolo Frame Relay.

Alguns fabricantes de comutadores e FRADs suportam os três procedimentos. Há equipamentos que oferecem a opção de reconhecimento automático do LMI utilizado pelo comutador Frame Relay. O tipo de LMI deve ser configurado no equipamento.

⁸ Digital Equipment, Northern e Stratacom

O LMI original e o proprietário da Cisco utilizam o DLCI “1023” para a comunicação FRAD-comutador. Os procedimentos Annex A9 e ANSI T.1617 Annex D utilizam o DLCI “0” (zero).

Alguns comutadores também podem ser configurados para detectar automaticamente que procedimento o FRAD está utilizando.

A configuração do tipo de LMI a ser utilizado no acesso Frame Relay é fundamental.

4.8.10 CONTROLE DE ERROS

Os mecanismos de detecção e correção de erros que surgem nos quadros recebidos, são funções típicas da camada de enlace.

Há basicamente dois tipos de erros:

- Quadros que não chegam
- Quadros que chegam com bits errados

A detecção de erros é feita pela verificação do campo FCS, normalmente uma sequência resultante de uma operação de redundância cíclica (CRC) em cima da maioria dos bits do quadro. Uma vez detectado que há erro no quadro, o receptor deve, de alguma forma, receber novamente esse quadro.

Os três métodos mais comuns de correção dos erros são:

- Stop-and-wait: o transmissor transmite um quadro com dados e espera o aceite (ACK) do receptor. Somente transmite novo quadro se receber um ACK do receptor (sem erro). O receptor verifica a FCS; se não houver erro no quadro de dados, envia um ACK ao transmissor. Se o receptor receber um quadro com erro, pede uma retransmissão, enviando uma rejeição (NACK).
- Go-back-N: o transmissor pode enviar vários quadros de dados e o receptor vai aceitando cada um deles. Quando o transmissor recebe um NACK ele volta e retransmite todos os quadros a partir daquele que foi recebido com erro. Nesse processo deve haver uma numeração sequencial dos quadros.
- Selective-reject: similar ao anterior, porém somente os quadros rejeitados serão retransmitidos.

O transmissor normalmente tem um tempo máximo de espera (timeout) pelo ACK ou NACK, que é um parâmetro do protocolo a ser definido na instalação. Após o tempo de timeout o transmissor supõe que o quadro não foi recebido e o retransmite.

⁹ É dessa forma que a norma é conhecida e citada nas literaturas.

Nem todos os protocolos da camada de enlace implementam a correção de erros por retransmissão, mas todos implementam a verificação de erros.

Se o protocolo não implementa a correção de erros, o quadro será recebido mesmo com erro ? Há duas alternativas: sim, e os dados serão passados para o protocolo da camada superior que, supostamente, deve cuidar desse problema; e não, o protocolo de enlace simplesmente descarta o quadro com erros, sem avisar o transmissor, e, novamente, a camada superior deve cuidar do problema.

4.8.11 CONTROLE DE FLUXO

É um procedimento, normalmente implementado nos protocolos das camadas de enlace e rede, que tenta ajustar o fluxo de dados à capacidade do receptor.

Normalmente o receptor possui uma memória para armazenar os quadros recebidos até conseguir processá-los (buffer de entrada).

Nem sempre o receptor tem capacidade para processar os quadros no ritmo em que estão chegando. De alguma forma o fluxo deve ser contido para não haver perda de quadro por falta de espaço no buffer de entrada.

O método mais simples, implementado na camada de enlace é o stop-and-wait. Claro, nesse caso o transmissor só envia o próximo quadro quando receber o ACK do receptor, então, o próprio procedimento para correção de erros pode ser utilizado para controlar o fluxo. A deficiência desse método é, por ser essencialmente semi-duplex, ter uma eficiência limitada.

Outro método é o da janela deslizante, que supera a limitação do método anterior pois permite a transmissão de vários quadros antes de receber alguma confirmação.

4.8.12 JANEJALA DESLIZANTE

Nesse caso, o receptor tem um buffer para "n" quadros, e o transmissor pode, inicialmente, transmitir até "n" quadros sem receber confirmação. O transmissor mantém o controle de quantos quadros foram transmitidos e ainda não foram aceitos - deve ser no máximo, igual a "n". O receptor pode aceitar mais de um quadro de uma vez. Para isso, é preciso existir um campo no quadro que especifique seu número sequencial.

A quantidade de quadros que pode ser transmitida, em um determinado instante, é a janela. No início a janela é igual a "n" quadros. Logo que o transmissor transmite um quadro a janela diminui de um (cai para "n-1"). Quando o receptor recebe um "ACK" a janela aumenta de um.

A figura 3.32 ilustra o funcionamento de uma janela deslizante de 4 quadros: no início a janela é igual a 4 quadros ($j=4$), ou seja o transmissor pode transmitir até 4 quadros sem receber confirmação do receptor. O transmissor vai transmitindo os quadros, sem receber confirmações, até $j=0$ (transmissões 1 a 4) quando fecha a janela. Logo que o transmissor recebe a confirmação de um quadro, a janela se abre de uma unidade. Então o transmissor transmite mais um quadro (transmissão 5). Pelo exemplo, o receptor é mais lento e está “segurando” o transmissor. Note que o receptor somente vai confirmar o recebimento do primeiro quadro após o tempo “ t_1 ”. Note também que o canal, deve ser duplex, pois o receptor está transmitindo ao mesmo tempo que o transmissor.

Dependendo do protocolo, o receptor pode confirmar mais de um quadro de uma só vez, como exemplificado na mesma figura, quando o receptor confirma a recepção dos quadros 2 e 3 ao mesmo tempo (recepção 2,3), abrindo a janela para $J=2$.

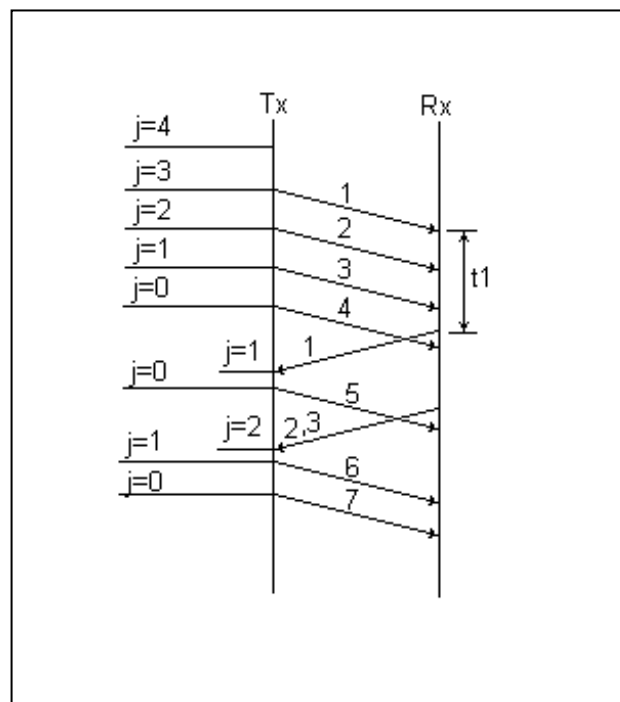


Fig. 4.19: Funcionamento da janela deslizante

4.8.13 TAXA REAL DE TRANSMISSÃO DE ENLACE

Um enlace possui dois equipamentos de transmissão, um em cada extremidade (modem, por exemplo), que operam na velocidade de transmissão v_t [bps].

Essa não é a velocidade média real de transmissão dos dados do usuário (bits de informação), devido aos seguintes fatos:

- a) O protocolo de enlace adiciona bits de controle aos pacotes com diversas finalidades, como por exemplo, fazer detecção e correção de erros, controle de fluxo, etc.
- b) Há um tempo gasto entre a transmissão de dois pacotes, que depende também do protocolo. Esse tempo inclui, entre outros aspectos, o tempo gasto para informar se o pacote recebido deve ser retransmitido, os tempos de propagação, etc.
- c) Pode haver retransmissão do pacote se ele chegar ao seu destino com erro.

Define-se TRT, Taxa Real de Transmissão, como:

$$TRT = \frac{\text{nº de bits de informação aceitos pelo ETD}}{\text{tempo total gasto para isso}} = \frac{I}{T}$$

A eficiência do enlace (ou utilização do enlace) pode ser definida como:

$$U = \frac{\text{tempo transmitindo os dados}}{\text{tempo total gasto para isso}} = \frac{I / v_t}{T} = \frac{TRT}{v_t}$$

4.8.14 EFICIÊNCIA DO STOP-AND-WAIT

Esse é o método clássico para controle de erros, utilizado pelos protocolos da camada de enlace. Vamos calcular sua TRT, considerando que o protocolo é orientado a caractere.

A quantidade de bits de informação por pacote, I , é dada por:

$$I = u \cdot (B - C)$$

u = bits de informação (úteis) por caractere

B = total de caracteres por pacote

C = caracteres de controle por pacote

O tempo gasto para transmitir esses bits, considerando uma transmissão semi-duplex e um protocolo do tipo stop-and-wait, é dado por:

$$T = N \left(\frac{B \cdot b}{v_t} + \Delta_t \right)$$

N = transmissões necessárias para o pacote ser aceito

B = total de caracteres por pacote

b = total de bits por caractere

v_t = taxa de transmissão do equipamento de transmissão

Δ_t = tempo gasto entre a transmissão de dois pacotes

Então, substituindo na equação de definição da TRT, obtemos:

$$TRT = \frac{u \cdot (B - C)}{N \left(\frac{B \cdot b}{v_t} + \Delta_t \right)}$$

O parâmetro “N” pode ser colocado em termos da taxa de retransmissão do canal, que é, inclusive, uma informação de fácil medida.

$$N = \frac{\text{pacotes transmtidos}}{\text{pacotes certos}} = \frac{\text{pacotes certos} + \text{pacotes erados}}{\text{pacotes certos}} = \frac{c + e}{c}$$

A taxa de retransmissão de pacote é definida como:

$$R = \frac{e}{c + e}$$

Então, substituindo, obtemos:

$$N = \frac{1}{1 - R}$$

Finalmente, utilizando esse resultado, podemos expressar a TRT como:

$$TRT = \frac{(1-R) \cdot u \cdot (B-C)}{\frac{B \cdot b}{v_t} + \Delta_t}$$

A taxa de retransmissão de pacote, “R”, é função da taxa de erro de bit, imposta pelo meio de comunicação (equipamento de transmissão e canal). Essa relação, supondo que não haja correção de erro no modem, é dada aproximadamente por¹⁰:

$$R \cong 1 - (1 - E \cdot b \cdot e_m)^{B \cdot b}$$

E = taxa de erro de bit do meio de comunicação

e_m = fator que desconta os erros múltiplos causados pelos randomizadores e equalizadores digitais dos equipamentos de transmissão. Podemos utilizar o valor $e_m = 0,3$; para modems com equalizador adaptativo.

B = comprimento do pacote em caracteres

b = bits por caractere

Convém salientar que a fórmula para “R” é aproximada e que pode-se lançar mão de uma estatística de retransmissões, às vezes obtida através do sistema de gerenciamento do computador hospedeiro do aplicativo, a fim de se evitar um erro de projeto.

O tempo de transmissão de um pacote é dado por:

$$t_{\text{pacote}} = \frac{B \cdot b}{v_t}$$

Considerando que não há erros na transmissão ($R=0$), que os bits dos caracteres são todos úteis (normalmente o que ocorre em uma transmissão síncrona) e desprezando os caracteres de controle do pacote, chegamos a uma fórmula mais simplificada para a TRT, que destaca sua dependência em relação ao intervalo de tempo entre dois pacotes e o tempo de transmissão do pacote. Utilizando as mesmas considerações, chegamos a uma fórmula simplificada para a eficiência:

$$TRT = \frac{v_t}{1 + \frac{\Delta_t}{t_{\text{pacote}}}}$$

$$U = \frac{1}{1 + \frac{\Delta_t}{t_{\text{pacote}}}}$$

¹⁰ Ref. 7.

O intervalo de tempo entre dois pacotes depende dos seguintes fatores:

- a) Tempo de propagação na transmissão entre os dois pontos.
- b) Tempo de processamento do receptor para analisar o pacote recebido e iniciar sua resposta.
- c) Tempo do "ACK". O protocolo stop-and-wait possui um procedimento de correção de erros que espera a confirmação do lado remoto de que o pacote foi bem recebido.
- d) Tempo do retardo RTS-CTS. Os equipamentos de transmissão consomem um tempo para se sincronizarem a cada pacote, principalmente quando operam no modo semi-duplex.
- e) Tempo de retardo nos equipamentos de transmissão.

A figura 3.33 ilustra o retardo entre pacotes, detalhado em 9 segmentos.

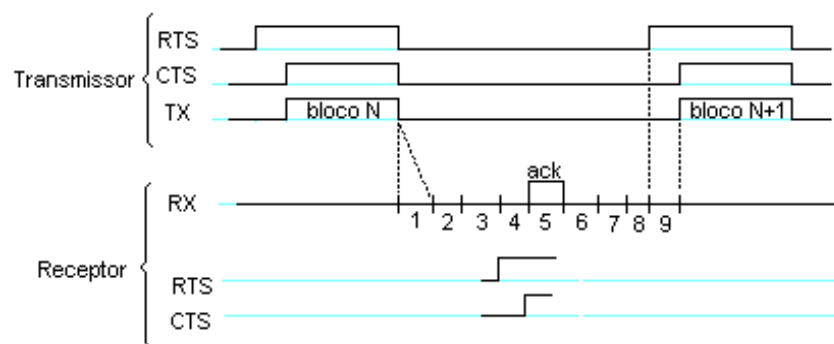


Fig. 3.33: Retardos entre dois pacotes

Os segmentos são

1. propagação do pacote (t_{prop})
2. retardo no modem
3. processamento do receptor
4. RTS-CTS do receptor
5. Tempo do ACK
6. Propagação do ACK (t_{prop})
7. Retardo no modem
8. Processamento do transmissor
9. RTS-CTS do transmissor

O tempo de propagação do pacote pode ser pequeno como 2 ms para curtas distâncias ou chegar a 300 ms quando o canal inclui uma subida de satélite.

O tempo de retardo no modem varia de 3 a 60 ms, dependendo de sua arquitetura interna.

Os tempos de retardo RTS-CTS podem ser nulos, como nos casos de circuitos que operam com modems duplex, por exemplo. Os modems que operam semi-duplex a dois fios (como os antigos V.27 e V.29), precisam desses retardos para sincronizar seus circuitos antes de iniciar a transmissão do pacote.

O retardo Δ_t inclui duas vezes o tempo de propagação pelo canal t_{prop} .

A partir da equação da TRT, é possível determinar o comprimento ótimo do pacote ($M_{máx}$), derivando-se em relação a “Bb”, igualando a zero para determinar o máximo e resolvendo para “M”.

$$M_{máx} = \frac{C - v_t \cdot \Delta_t}{2} + \sqrt{\frac{v_t^2 \cdot \Delta_t^2}{4} + \frac{C \cdot v_t \cdot \Delta_t}{2} + \frac{C^2}{4} - \frac{v_t \cdot \Delta_t + C}{\log_e(1 - E \cdot b \cdot e_m)}}$$

4.8.15 Eficiência da janela deslizante

Os métodos de correção de erros que utilizam a técnica da janela deslizante (selective-reject e go-back-N), vão transmitindo os pacotes em janelas e são semelhantes, quanto à eficiência, se não houver erro (se não houver retransmissão).

No caso anterior, stop-and-wait, somente um pacote poderia estar circulando no enlace a cada instante, independentemente da relação Δ_t/t_{pacote} . Agora, no caso da janela deslizante, pode haver mais de um pacote circulando no enlace se $\Delta_t > t_{pacote}$.

A eficiência do enlace depende do tamanho da janela e da relação Δ_t/t_{pacote} .

Considerando somente o retardo do enlace, (desprezando os caracteres de controle do pacote e considerando a taxa de erro nula), vamos analisar a eficiência. Temos duas situações:

$t_{janela} > \Delta_t + t_{pacote}$ o pacote chega ao transmissor antes de terminar o tempo da janela. Então a eficiência não será impactada pelo retardo do enlace, pois o transmissor pode transmitir continuamente sem

esperar por aceites. Nesse caso a eficiência será $U=1$, se considerarmos uma transmissão contínua.

$$t_{\text{janela}} < \Delta_t + t_{\text{pacote}}$$

o aceite do pacote chega ao transmissor depois de terminar o tempo da janela. Nesse caso a eficiência será impactada pelo retardo do enlace. O transmissor tem que parar sua transmissão quando termina o tempo da janela e esperar um aceite. Veja a próxima figura. Vamos supor que desejamos fazer uma transmissão cuja janela possui “J” pacotes. O transmissor transmite “J” pacotes e pára. O receptor manda o aceite após terminar de receber o primeiro pacote. Após receber esse primeiro aceite, o transmissor pode mandar outro pacote. Então, por inspeção da figura 3.34, podemos dizer que a eficiência será dada por:

$$U = \frac{J \cdot t_{\text{pacote}}}{\Delta_t + t_{\text{pacote}}} = \frac{J}{1 + \frac{\Delta_t}{t_{\text{pacote}}}}$$

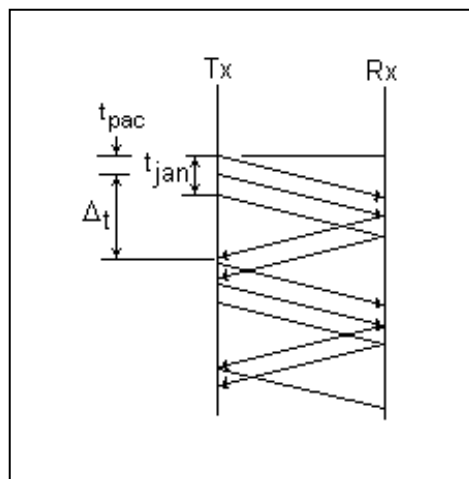


Fig. 3.34: Tempo de aceite maior que a janela

Portanto, conclui-se que a eficiência da janela deslizante é “J” vezes maior que a do método stop-and-wait.

- o - o - o -