

Capítulo 1 - INTRODUÇÃO

- *Comunicação* uma das maiores necessidades da sociedade humana. Desenvolvida desde os ancestrais (fumaça, etc.)

Telégrafo (1838 por Samuel Morse) inaugura a era de comunicação por sinais elétricos (telefone, rádio, tv).

Outras áreas tiveram grande evolução, p. ex. processamento e armazenagem de informações. Computadores foram, provavelmente o maior avanço do século.

Conjunção destas tecnologias (comunicação e processamento de informações) revolucionou o mundo moderno.

EVOLUÇÃO DOS SISTEMAS DE COMPUTAÇÃO

Década de 60 -

- computadores enormes e pesados.
- operação apenas por técnicos altamente especializados.
- processamento de *jobs* em *batch*.
- avanços no sentido de *terminais interativos* através de linhas de comunicação e de processamento em *time-sharing* com novas técnicas de processamento (S.O.).

Década de 70

- avanço no sentido de distribuição da capacidade computacional.
- requisitos operacionais de micros e minis computadores se tornam menos rígidos.
- desenvolvimento tecnológico levou a contínua redução do custo das máquinas, facilitando sua difusão e dispersão.

Década de 80

- alto custo de alguns periféricos motivou a evolução de técnicas de interconexão para seu uso compartilhado.
- interconexão motivou o estudo de ambientes computacionais cooperativos.
- requisitos de desempenho, confiabilidade e modularidade levaram ao desenvolvimento tanto de processamento paralelo como de processamento distribuído.

EVOLUÇÃO DAS ARQUITETURAS

Computadores projetados até década de 80 sempre foram baseados no modelo de *Von Neumann* - mecanismo simples para processamento puramente sequencial.

Novas arquiteturas foram propostas, como: sistemas de CPU única e múltiplas unidades funcionais, máquinas *pipeline*, processadores de matriz (array processors).

Sistemas de Multiprocessadores Fortemente Acoplados (vários elementos processadores com sequências múltiplas e independentes compartilhando espaço de memória) contornam o controle centralizado de Von Neumann, tendo como características:

- dois ou mais processadores com capacidades aproximadas,
- todos os processadores dividem o mesmo acesso à memória,
- canais de entrada/saída, unidades de controle e periféricos são compartilhados,
- um único sistema operacional controla todo o sistema.

Sistemas de Processamento Distribuído (Fracamente Acoplados): coleção de processadores interconectados lógica e fisicamente, colaboram na execução cooperativa de uma aplicação, mas com controle geral descentralizado. Têm como características:

- A interação entre processadores se dá por troca de mensagens.
- mínima interferências nas tarefas paralelas favorece desempenho.
- não se obtém simultaneidade mas sim sincronismo.
- modularidade permite maior confiabilidade.
- alta expansibilidade.

Vantagens de Sistemas multiprocessados (fortemente/fracamente acoplados)

- relação custo/benefício dos microprocessadores.
- responsividade.
- modularidade.
- confiabilidade.
- concorrência.

Desvantagens de Sistemas multiprocessados

- software aplicativo mais complexo.
- decomposição de tarefas mais complexa.
- software de diagnóstico mais difícil.
- fortemente dependente da comunicação.

- difícil o estabelecimento de controle do sistema.

Redes de Computadores - conjunto de módulos processadores capazes de trocar informações e compartilhar recursos quando interligados por um sistema de comunicação. Sistema de comunicação corresponde a um arranjo topológico interligando processadores através de enlace físico e de suas regras (protocolos).

Redes Locais - LANs (Local Area Network)

- surgiram em laboratórios de pesquisa e universidades.
- buscam compartilhar informação e recursos, preservando a independência de processamento.
- cobrem distâncias entre 100 m e 25 Km.
- apresentam altas taxas de transmissão (0,1 a 100 Mbps).
- apresentam baixas taxas de erros (10^{-8} a 10^{-11}).
- geralmente pertencem à propriedade privada

Redes Metropolitanas - MANs (Metropolitan Area Networks)

- cobrem áreas metropolitanas.
- apresentam características semelhantes às LANs.
- apresentam maiores taxas de transmissão que as LANs.

Redes Geograficamente Distribuídas - WANs (Wide Area Networks)

- compartilhamento de recursos especializados por usuários dispersos geograficamente.
- por questões de custo são gerenciadas por organismos públicos.
- também devido a custos, apresentam taxas de transmissão mais baixas (normalmente da ordem de centenas de Kbps).

Parâmetros para comparação de redes

Cada arquitetura possui características que afetam sua adequação a certa aplicação em particular. Os atributos mais considerados são:

- custo - referente a estações de processamento, de interfaces com o meio de comunicação e do próprio meio de comunicação.
- retardo de transferência - soma do retardo de acesso (intervalo de tempo entre a geração da mensagem e sua efetiva captura do meio para transmissão) com o retardo de transmissão (intervalo de tempo entre o início da transmissão pela estação origem e sua efetiva recepção pela estação destino).
- desempenho (intimamente relacionado com velocidade, retardo de transferência) - capacidade efetiva de transmissão da rede, afetado pela escolha estrutura de conexão, protocolos de comunicação, meios de transmissão e utilização da rede.

- confiabilidade - avaliada em termos de diversos fatores:
- do tempo médio entre falhas - MTBF (Medium time between failures) - geralmente medido em horas e relacionado com componentes,
- tolerância a falhas - capacidade de recuperação automática após uma falha,
- degradação amena (gracefull degradation) - relacionada com a própria aplicação,
- tempo de reconfiguração entre falhas - relacionado com ativação de caminhos alternativos após falhas,
- tempo médio de reparo - MTTR (medium time to repair) - relacionado com redundância, autoteste e manutenção.
- modularidade - refere-se a possíveis alterações de desempenho e funcionalidade (modificação, crescimento e uso de componentes básicos) da rede sem alteração do projeto.
- compatibilidade (interoperabilidade) - capacidade da rede em ligar-se a dispositivos produzidos por vários fabricantes.
- sensibilidade tecnológica - capacidade de suportar inovações tecnológicas.

Capítulo 2 - TOPOLOGIAS

TOPOLOGIA de uma rede de comunicação - forma com que os enlaces físicos e os nós de comutação se organizam de forma a criar caminhos físicos entre estações conectadas pela rede.

2.1 - LINHAS DE COMUNICAÇÃO

Ligações físicas podem ser:

- ligações ponto-a-ponto - presença de apenas dois pontos de comunicação, um em cada extremidade do enlace ou ligação,
- ligações multiponto - presença de três ou mais dispositivos de comunicação podendo utilizar o mesmo enlace.

Figura 2.1 - Tipos de ligação

A forma de comunicação através do meio físico pode ser tipo:

- *simplex* - o enlace é utilizado apenas em um dos dois possíveis sentidos de transmissão.
- *half-duplex* - o enlace é utilizado nos dois possíveis sentidos de transmissão, porém apenas um por vez.
- *full-duplex* - o enlace é utilizado simultaneamente nos dois possíveis sentidos de transmissão.

Figura 2.2 - Comunicação simplex, half-duplex e full-duplex

2.2 - REDES GEOGRAFICAMENTE DISTRIBUÍDAS

Uma alternativa intuitiva para ligação em rede seria *topologia totalmente ligada*, com ligações ponto-a-ponto mostrado na figura 2.3. Troca de mensagens se dá sem enlaces. Uma rede com N pontos teria $N(N-1)/2$ ligações. Inviabilidade quanto a custo.

Fig. 2.3 - Topologia totalmente ligada

Outro caso extremo é a *topologia em anel* (figura 2.4). Normalmente tem-se ligação em um único sentido, com mensagem circulando no anel até atingir o destino.

Fatores limitantes (velocidade e confiabilidade):

- aumento de pontos intermediários entre pontos inicial e final de uma ligação com aumento do retardo de transmissão;
- inexistência de caminhos alternativos para tráfego de mensagens.

Topologia intermediária é *topologia parcialmente ligada* ou *topologia em grafo* (figura 2.5). Nem todas estações se encontram ligadas mas existem caminhos alternativos para caso de falhas ou congestionamentos. Estações que não tenham ligação com a que desejam alcançar encaminham mensagens a outra que o tenha.

Fig. 2.4 - Topologia em anel

Fig. 2.5 - Topologia em Grafo

Comunicação entre módulos processadores (DTE-Data Terminal Equipment) pode ser por:

- chaveamento de circuitos: canal entre DTEs fonte e destino é estabelecido exclusivamente até que a conexão seja desfeita, como em ligação telefônica.
- chaveamento de mensagens: mensagem é transmitida a um DTE que a armazena até obter canal para transmiti-la ao destino ou outro ponto intermediário.
- chaveamento de pacotes: mensagem é cortada em pacotes que podem ser transmitidos por diferentes canais, sendo reagrupada no destino.

Roteamento é a escolha de rota entre nó origem e nó destino. Pode ser realizado:

- antes do envio da mensagem - estabelece-se uma *conexão* entre origem e destino, definindo a rota enquanto perdurar a conexão.
- passo a passo - cada nó intermediário do caminho fim a fim escolhe o próximo nó quando recebe a mensagem.

Diversos algoritmos de roteamento foram propostos. É desejável que o roteamento seja adaptativo.

Qualquer topologia que não a totalmente ligada deve definir *mecanismos de endereçamento*. Quando há conexão, endereços de origem e destino só são necessários no estabelecimento da conexão. Quando não há conexão, cada mensagem ou pacote deve indicar origem e destino.

Redes com chaveamento por pacotes sobrecarregam estação com:

- escolha do caminho para cada pacote - roteamento,
- armazenagem dos pacotes recebidos - a serem roteados para outras estações,
- detecção de erros de transmissão e recepção,
- reagrupamento de pacotes recebidos - quando estação destino,
- gerenciamento do hardware de comunicação.

Sistemas externos de controle auxiliam DTEs nestas tarefas:

- Equipamentos de comunicação de dados (DCE - Data Communicating Equipments) realizam roteamento e gerenciamento da rede,
- equipamentos de chaveamento de dados (DSE - Data Switching Equipment) ou nós de comutação, colaboram na concentração do tráfego interno e restauração de sinais.

Em redes distribuídas, comutadas por pacotes, um DCE normalmente é compartilhado por vários DTEs.

Arranjo topológico formado por DCEs em conjunto com nós de comutação (DSEs) e regras de comunicação é chamado de *sub-rede de comunicação*. Sub-redes são, normalmente, operadas por empresas especializadas de serviços de comunicação.

Fig. 2.6 - Rede geograficamente distribuída

2.3 - REDES LOCAIS E METROPOLITANAS

Topologias inviáveis em redes geograficamente distribuídas podem ser usadas em redes locais e metropolitanas, devido aumento de confiabilidade e desempenho disponível com novos meios de transmissão.

Topologia em estrela

Cada nó é interligado a um nó central (mestre), por onde passam todas as mensagens. Nó central é o centro de controle da rede, interligando os outros nós (escravos). Pode haver comunicações simultâneas, se estações envolvidas forem diferentes.

Fig. 2.7 - Topologia em estrela

Arranjo em estrela é melhor escolha se a comunicação necessária na rede puder ser suportada por esta topologia.

Nó central com função única de chaveamento entre estações é denominado *comutador* ou *switch*.

O nó central pode ter funções de:

- gerenciamento da comunicação,
- processamento de informações supridas pelos nós "escravos",
- compatibilizar velocidade de comunicação entre transmissor e receptor,
- converter protocolos,
- fornecer proteção à rede,
- realizar operações de diagnóstico da rede.

Redes em estrela podem também atuar por difusão (broadcasting), onde só o nó destino copia a mensagem. Caso contrário, um nó apenas se comunica com um outro através do nó central.

Gerenciamento pode ser do tipo: chaveamento de pacotes, de mensagens ou mesmo ATM (Asynchronous Transfer Mode)

Problemas principais de redes em estrela são:

- confiabilidade - falhas no nó central podem parar o sistema,
- modularidade - configuração só se expande até os limites do nó central.
- desempenho - limitado pela capacidade de processamento do nó central.

Topologia em anel

Arranjo de estações conectadas através de um caminho fechado.

Por questões de confiabilidade o anel não interliga diretamente as estações. Estas se ligam a repetidores que estão conectados ao meio físico.

São capazes de transmitir bidirecionalmente, mas no geral são unidirecionais, simplificando protocolos e repetidores e evitando o problema de roteamento.

Fig. 2.8 - Topologia em anel

Repetidores são projetados para recepção e transmissão simultânea, diminuindo o retardo na transmissão.

Dependendo do protocolo utilizado, uma mensagem enviada por um nó entra na rede e nela circula até ser retirada pelo nó destino (retardo equivalente ao endereçamento) ou voltar ao nó de origem (retardo de um bit).

Redes com retirada de mensagem pelo nó origem permitem *multicast* e *broadcast*. Pode-se, ainda, ter estação no *modo espião* para gerenciamento de tráfego.

A necessidade do nó remover seletivamente mensagens requer um repetidor ativo em cada nó. Uma falha no repetidor compromete toda o enlace da rede. O repetidor pode estar interno ou externo à máquina. Caso menos vulnerável é o externo, com alimentação separada da estação.

Questões de confiabilidade exigem a colocação de relés no repetidor, para realizar remoção física do repetidor defeituoso (Fig. 2.9).

Fig. 2.9 - Acionamento do relé em caso de falha

Interface com relé apresenta três diferentes estados de operação (Fig. 2.10):

- *escuta* - cada bit que chega ao repetidor é retransmitido com o menor retardo possível, apenas o suficiente para realizar:
- análise de fluxo de dados para procura de padrões de bits como endereço da estação local ou permissão de controle.
- em caso de mensagem destinada à estação local, realizar a cópia de cada bit do fluxo de entrada, ao mesmo tempo em que os retransmite.
- modificação de bits do fluxo de entrada para a retransmissão quando necessário o controle de erros.
- *transmissão* - os bits recebidos da estação local são transmitidos pela interface, embora possa receber fluxo de bits do anel. Nesta caso podem ocorrer duas situações:
- interface recebe bits transmitidos por ela mesma (baixo retardo da rede): retorna os bits à estação para verificação ou apenas descarta.
- interface recebe bits transmitidos por outra estação: bits são armazenados para posterior retransmissão.
- *Bypass* - usado para aumentar confiabilidade da rede em caso de repetidor/estação em falha e otimiza rede para estação não ativa.

Fig. 2.10 - Modos de operação do repetidor em um anel

Melhoras na confiabilidade e modularidade levam a propostas complementares da topologia em anel.

- a. introdução de *HUBS* (concentrador) - elemento concentrador de repetidores do anel (Fig. 2.11). Apresenta como vantagens:
 - isolamento de falhas se torna mais simples já que existe ponto de acesso central para o sinal, evitando busca por todo o anel.
 - modularidade é grande já que se pode adicionar novos nós sem a parada da rede.

Fig. 2.11 - Concentrador

Redes em anel teoricamente podem chegar a distancia infinita. São limitadas por manutenção, confiabilidade e retardo cumulativo dos repetidores.

- a. introdução de *anel de interconexão de concentradores* - criado pela limitação prática de número de estações em um só anel. Alternativa de maior confiabilidade que esta opção é

anel duplo (anel de backup) que recupera falhas nos segmentos entre concentradores (Fig. 2.12).

Fig. 2.12 - Anel de concentradores e duplo anel

- b. introdução de *bridges* (pontes) - interligação de vários aneis com uma bridge, formando sub-redes (Fig. 2.13). A bridge encaminha os pacotes de dados de uma sub-rede a outra com base nas informações de endereçamento.

Cada sub-rede pode operar de forma independente. Uma falha no anel só afeta a sub-rede. Uma falha na bridge não afeta a operação intra rede.

Fig. 2.13 - Múltiplos aneis ligados por bridge

Redes em anel têm vulnerabilidade a erros e pouca tolerância a falhas. Caso de perda do controle de acesso leva a difíceis controle e decisão de qual nó deva recuperá-lo.

Topologia em barra

Nas redes em topologia de barra comum, todas as estações se ligam ao mesmo meio de transmissão. Diferente das outras topologias, tem configuração multiponto. Cada nó conectado à barra pode ouvir todas as informações transmitidas.

Fig. 2.14 - Topologia em barra

Os mecanismos de acesso ao meio podem ser :

- centralizados - o direito de acesso é determinado por uma estação especial da rede. Oferece os mesmos tipos de problemas de confiabilidade da rede em estrela.
- descentralizados - a responsabilidade se distribui entre os nós. Apresenta o mesmo problema de confiabilidade da rede em anel.

Topologias em barra podem usar interfaces passivas, cujas falhas não param todo o sistema, nem aumentam o retardo.

Estações se ligam ao meio através de *transceptores* (transmissor/receptor) que transmitem e recebem sinais e reconhecem a presença de sinais no meio. O transceptor deve apresentar alta impedância ao cabo, já que a ligação ao meio deve interferir o mínimo nas características elétricas do meio. Este deve terminar, em ambos os extremos, com cargas iguais à sua impedância característica, para evitar reflexões.

Fig. 2.15 - Ligação ao meio em redes em barra

Distância máxima e número máximo de nós dependem do meio de transmissão usado, da taxa de transmissão e quantidade de ligações ao meio.

Pode-se usar repetidores, que se tornam ponto de diminuição da confiabilidade.

O uso de Hubs facilita a localização e isolamento de falhas, além de facilitar a inserção de novas estações sem parar o sistema (Fig. 2.16).

Fig. 2.16 - Rede em barra com hub

Hubs podem ser interconectados como forma de expansão da rede (Fig. 2.17).

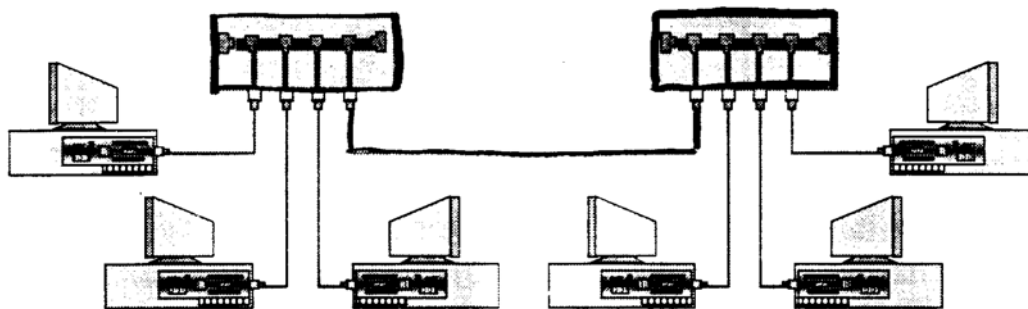


Fig. 2.17 - Rede em barra interconectada por hubs

Notar que hubs podem ser usados tanto em anel como em barra.

Isto distingue *topologia lógica* (organização das interfaces com seus métodos de acesso) da *topologia física* (distribuição física utilizada na instalação da rede).

Switches são elementos concentradores, cuja implementação interna embora desconhecida garante interface coerente às estações mesmo com arquiteturas que não utilizam apenas um meio compartilhado mas possibilitam a troca de mensagens entre estações simultaneamente.

Capítulo 3 -ARQUITETURAS DE REDES DE COMPUTADORES

Experiência mostrou como boa opção estruturar redes como um conjunto de *camadas hierárquicas*.

Cada camada é um programa ou processo, implementado em hardware ou software, se comunica com processos correspondentes em outra máquina.

Uma camada não se comunica diretamente com sua correspondente em outra máquina mas se utiliza dos serviços oferecidos pelas camadas inferiores.

Os limites entre níveis adjacentes são chamados *interfaces*.

As regras que organizam a comunicação entre camadas de nível N são chamados *protocolos de nível N*.

Fig. 3.1 - Camadas, Protocolos e Interfaces em Sistema Aberto

Arquitetura da rede é formada por níveis, interfaces e protocolos. Cada nível oferece serviços ao nível superior a partir de seus serviços e dos oferecidos pelas camadas inferiores

3.1 - O MODELO OSI DA ISO

O Modelo de Referência para Conexão de Sistemas Abertos (Reference Model for Open Systems Interconnection - RM-OSI) com arquitetura de sete níveis foi elaborado pela ISO - International Organization for Standardization.

Propósito do modelo de referência é fornecer base comum para o desenvolvimento coordenado de padrões para interconexão de sistemas. O modelo não trata questões de implementação.

O RM-OSI por si só não define uma arquitetura, já que não especifica os serviços e protocolos de cada camada. apenas indica o que cada uma deve fazer.

Dois sistemas obedecendo o RM-OSI podem não interagir, por oferecer protocolos e serviços incompatíveis nas camadas. Para que se garanta a interconexão há necessidade que sistemas escolham *perfis funcionais* compatíveis (grupos de serviços/protocolos padronizados).

Exemplo de perfis funcionais são MAP (Manufacturing Automation Protocol) e o TOP (Technical and Office Protocol), inicialmente elaborados pela GM e Boeing, respectivamente.

Fig. 3.2 - Perfil funcional MAP/TOP

a) Terminologia do RM-OSI

Serviço representa um conjunto de funções oferecidas a um *usuário* por um *fornecedor*.

Serviço é oferecido em um *ponto de acesso* (Service Access Point - SAP).

Elementos ativos nas camadas são *entidades*, representada por um software (processo) ou hardware (placa de interface de rede).

Entidades de mesma camada, em máquinas diferentes são *entidades pares* ou *parceiras*.

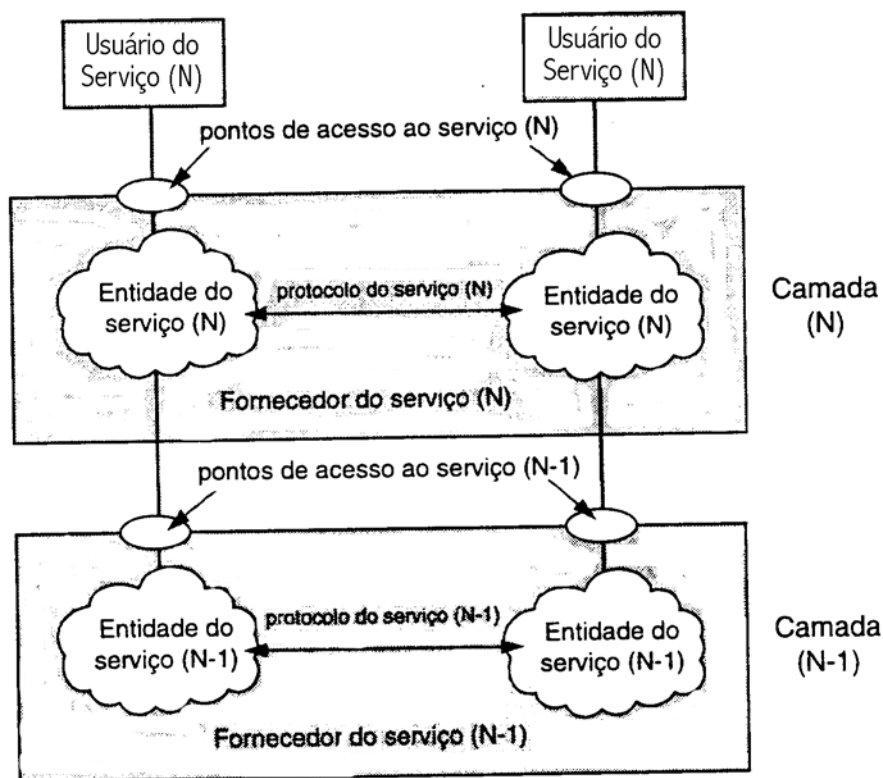


Fig. 3.3 - Fornecedores e usuários de serviços

Serviço oferecido por uma camada é especificado pelas *primitivas de serviço* e pela ordem com que são apresentadas. Um serviço pode ser:

- confirmado (com confirmação) - acordo entre solicitante e acolhedor. Primitivas envolvidas: service.REQUEST (invocada pelo usuário), service.INDICATION (entregue ao acolhedor), service.RESPONSE (invocada pelo acolhedor) e service.CONFIRMATION (entregue ao solicitante).
- não-confirmado (sem confirmação) - sem acordo entre solicitante e acolhedor. Primitivas envolvidas: service.REQUEST (invocada pelo usuário), service.INDICATION (entregue ao acolhedor).
- iniciado pelo fornecedor - serviço gerado pelo fornecedor em resposta a alguma condição interna. Primitiva envolvida: service.INDICATION (entregue ao acolhedor e solicitante).

Fig. 3.4 - Tipos de serviço OSI

O serviço oferecido por uma camada pode ser *orientado a conexão* ou *não orientado a conexão*.

Transmissão orientada a conexão o serviço tem 3 fases:

1. estabelecimento de conexão: usuário e fornecedor negociam parâmetros e opções de serviço. Em caso de negociação favorável, a conexão se estabelece e as primitivas são executadas no contexto da conexão. Ex. estabelecimento de conexão de serviço de transporte, com confirmação, as seguintes primitivas são trocadas: T-CONNECT.request (solicitante ao

fornecedor de serviço), T-CONNECT.indication, T-CONNECT.response e T-CONNECT.confirmation, nesta ordem.

1. transferência de dados: usuários de serviço trocam dados. Ex. continuação do anterior: solicitante envia ao fornecedor do serviço a primitiva T-DATA.request acompanhada dos dados. O fornecedor providencia o transporte dos dados e os entrega ao destinatário pela primitiva T-DATA.indication.
1. liberação de conexão: ligação entre usuário é desfeita. Pode ocorrer de três formas: desconexão ordenada - usuários concordam com o encerramento em serviço confirmado; desconexão abrupta como resposta a solicitação de um dos envolvidos, em serviço não confirmado; e ; desconexão abrupta pelo fornecedor, no serviço iniciado pelo fornecedor.

Na transmissão não orientada a conexão:

- uma única unidade de dados é transmitida do SAP origem para um ou mais SAPs destino, sem que haja conexão entre eles.
- toda a informação necessária para transmitir os dados (endereços, parâmetros de qualidade de serviços, etc.) é passada junto com os dados para a camada que oferece o serviço, através da primitiva service.REQUEST.
- o fornecedor do serviço entrega a unidade de dados transmitida aos destinatários através da primitiva service.INDICATION.

b) O Nível Físico

Permite o envio de uma cadeia de bits pela rede sem se preocupar com o significado dos bits ou como são agrupados.

O protocolo de nível físico se dedica à transmissão de uma cadeia de bits.

Não é função deste nível o tratamento de problemas de erros de transmissão.

c) O Nível de Enlace de Dados

Detecta e opcionalmente corrige erros ocorridos no nível físico. Para isto executa a partição da cadeia de bits em *quadros*, cada um contendo alguma forma de redundância para detecção de erros.

Também cuida de evitar que o transmissor envie mais dados ao receptor do que este consegue receber. Usa métodos de controle de fluxo, informando ao emissor o tamanho disponível do buffer do receptor.

d) O Nível de Rede

Fornece ao nível de transporte uma independência quanto a questões de chaveamento e roteamento associados com a conexão da rede.

Em redes ponto a ponto, está ligado ao roteamento e seus efeitos (controle de tráfego, etc.).

Em redes do tipo difusão, com uma única rota, torna-se irrelevante.

Duas filosofias de serviço são oferecidas:

- serviço de *datagrama* (não orientado a conexão) cada *pacote* (unidade de dados do nível 3) carrega por completo o endereçamento de destino.
- serviço de *circuito virtual* (orientado a conexão): o transmissor inicialmente envia pacote para estabelecer a conexão. Ao circuito é dado um número que representa a conexão, que é usado pelos outros pacotes.

e) O Nível de Transporte

Tem como funções:

- fornecer comunicação fim a fim (origem-destino) confiável, uma vez que o nível de rede não garante a entrega dos pacotes, nem sua ordem.
- otimizar as conexões de rede. Faz: *multiplexação* (várias conexões de transporte partilhando a mesma conexão de rede) quando uma conexão de transporte não gera tráfego suficiente para ocupar toda a capacidade de conexão da rede; *splitting* (uma conexão de transporte ligada a várias conexões de rede) para aumentar a vazão de uma conexão de transporte.
- controle de fluxo, evitando que transmissor envie mensagens em taxa maior que a capacidade do receptor.
- controle de sequência fim a fim.
- detecção e recuperação de erros fim a fim (entidade de transporte para entidade de transporte).
- segmentação e blocagem de mensagens.

f) O Nível de Sessão

Fornecer serviços que permitem estruturar os circuitos oferecidos pelo nível de transporte. Os principais são:

- gerenciamento de token: controlam a posse e passagem do token entre entidades de aplicação que estão utilizando o serviço. Usado principalmente quando aplicação exige comunicação half-duplex em circuito full-duplex.
- controle de diálogo: permite retomar um diálogo interrompido por falha na rede, comunicada pelo nível de transporte. Caso a rede retorne a estar operacional, retoma-se a comunicação a partir do último *ponto de sincronização*. Toda vez que um usuário do serviço de sessão recebe um ponto de sincronização, este deve responder com um aviso de recebimento.
- gerenciamento de atividades: *atividade* corresponde a partes do intercâmbio de dados. Em uma conexão de sessão só pode ocorrer uma atividade por vez, mas podem existir atividades consecutivas durante uma conexão. Uma atividade pode ser interrompida e depois recomeçada na mesma sessão ou em conexões de sessões subsequentes. Ex.: co-edição interrompida devido ao encerramento de uma sessão ou mensagem de mais alta prioridade é retomada em nova sessão ou na mesma sessão após a execução da atividade de mais alta prioridade.

g) O Nível de Apresentação

Realiza adequação de dados antes de seu envio para o nível de sessão. Os principais serviços são de:

- transformação de dados: criptografia de dados, compressão de textos.

- formatação de dados: conversão de padrões de terminais e arquivos para padrões de rede e vice-versa.
- seleção de sintaxes: adequa a sintaxe entre entidades fim.
- conexão de apresentação: manutenção da conexão de apresentação.

h) O Nível de Aplicação

Oferece funções de gerenciamento e mecanismos genéricos de suporte à construção de aplicações distribuídas.

Os elementos de serviço genérico são:

- ACSE (Association Control Service Element): estabelecimento de uma associação com um ou mais usuários.
- ROSE (Remote Operations Service Element): suporte a chamadas de procedimentos remotos.
- RTSE (Reliable Transfer Service Element): serviço confiável de transferência de dados, tornando transparente ao usuário os mecanismos de recuperação de erros.

Além dos serviços genéricos, existem os elementos específicos de cada protocolo de aplicação como FTAM (File Transfer, Access and Management), DS (Directory Service) e MHS (Message Handling Service).

i) Transmissão de Dados no Modelo OSI

A comunicação entre dois sistemas no ambiente OSI, representada pela figura 3.5, desenrola-se por:

- aplicativo entrega ao nível de aplicação do sistema A os dados a serem comunicados ao outro sistema. Estes dados recebem o nome de *Unidade de Dados de Serviço* (Service Data Unit - SDU) do nível de aplicação.
- camada de aplicação agrega um cabeçalho denominado *Informação de Controle de Protocolo* (Protocol Control Information - PCI). O objeto resultante é a *Unidade de Dados de Protocolo* (Protocol Data Unit - PDU), que é a unidade de informação trocada entre as entidades pares nos dois sistemas. O PDU (dados mais cabeçalho) é passado à camada de apresentação.
- A PDU da camada de aplicação se transforma em SDU da camada de apresentação. Esta camada agrega seu cabeçalho, de forma a compor o PDU da camada de apresentação.

Fig. 3.5 - Transmissão de dados no ambiente OSI

- o processo repete-se pelas camadas, até o nível de enlace, que além do cabeçalho geralmente agrega um fecho contendo a *Sequência de Verificação de Quadro* (Frame Check Sequence - FCS), para detecção de erro.
- a PDU da camada de enlace, chamada de quadro (frame), é transmitida pelo nível físico.
- o frame é recebido pelo nível físico do sistema B.
- ocorre o processo inverso, onde cada entidade par retira o cabeçalho de seu nível, até que os dados sejam entregues ao usuário do sistema B.

j) Aplicabilidade do RM-OSI em Redes Locais

Embora o RM-OSI possa ser usado em redes locais, foi desenvolvido com foco nas redes geograficamente distribuídas. Suas características, eventualmente, tornam-se desnecessárias em redes locais.

Em redes locais, o nível físico se utiliza de um meio de alta velocidade e baixa taxa de erros, o que influencia as necessidades de outros níveis.

A nível de enlace, a delimitação de quadros pode se tornar desnecessária, podendo reconhecer a delimitação apenas pela existência de sinal no meio de comunicação. Pelo alto desempenho do meio físico, não se aplica a possibilidade de recuperação de erros, principalmente porque para algumas aplicações os requisitos de tempo real podem ser mais importantes que confiabilidade da transmissão.

A camada de rede geralmente não tem relevância em redes locais, uma vez que geralmente as transmissões são por difusão. Em muitas redes a camada de transporte é construída acima do nível de enlace. Opcionalmente, implementa-se um nível *inter-redes*, cumprindo o roteamento de pacotes através de várias redes. Também não se considera recuperação de erros a este nível, devido ao bom desempenho do meio físico.

A implementação das demais camadas de protocolos fica dependente das aplicações a que se destinam.

3.2 - O PADRÃO IEEE 802

Em 1980 a IEEE Computer Society criou um comitê para elaboração de padrões para redes locais de computadores. O Projeto 802 elaborou padrões que foram posteriormente aceitos pela ANSI (American National Standards Institute) e pela ISO como padrão internacional denominado ISO8802.

O modelo elaborado considera uma arquitetura de 3 camadas.

- *Nível Físico*: encarrega-se das funções de codificação/decodificação de sinais, geração e remoção de preâmbulos para sincronização e transmissão/recepção de bits.
- *Nível de Controle de Acesso ao Meio* (MAC-Medium Access Control): dividida em partes especiais, otimizadas para diferentes topologias. Encarrega-se de:
 - a. na transmissão, montar os dados a serem transmitidos em quadros com campos de endereços e detecção de erros, e na recepção, desmontar os quadros, reconhecendo endereços e detectando erros.
 - b. gerenciar a comunicação do enlace.
- *Nível de Conexão Lógica* (LLC-Logical Link Control): fornece um ou mais SAPs para usuários de redes, com suas sub-funções.

Estas camadas correspondem aos níveis 1 e 2 do RM-OSI.

A figura 3.6 apresenta a relação entre os padrões IEEE802 e o RM-OSI.

Fig. 3.6 - Relação entre os padrões IEEE802 e RM-OSI

O padrão 802.1 descreve o relacionamento entre os diversos padrões IEEE802 e destes com o RM-OSI. Também contém padrões para gerenciamento da rede e informações para a ligação inter-redes.

O padrão ANSI/IEEE802.2 (ISO 8802/2) descreve a camada superior do nível de enlace, a que utiliza o Protocolo de Controle de Ligação Lógica (Logical Link Control Protocol).

Os demais protocolos da especificação IEEE802 correspondem às diferentes opções de nível físico e sub-camada MAC para diferentes tipos de redes locais:

- padrão IEEE802.3 (ISO 8802/3): rede em barramento com método de acesso CSMA-CD.
- padrão IEEE802.4 (ISO 8802/4): rede em barramento com permissão de acesso (token).
- padrão IEEE802.5 (ISO 8802/5): rede em anel com permissão de acesso (token).
- padrão IEEE802.6 (ISO 8802/6): rede em barramento com método de acesso DQDB (Distributed Queue Dual Bus).

3.3 - ARQUITETURA DA INTERNET - TCP/IP

O desenvolvimento desta arquitetura foi patrocinado pela Defense Advanced Research Projects Agency (DARPA). Esta arquitetura baseia-se em um serviço de transporte orientado a conexão determinado pelo *Transmission Control Protocol* (TCP) e em um serviço de rede não orientado a conexão (datagrama não confiável) determinado pelo *Internet Protocol* (IP). Os padrões da Internet não são elaborados por organismo oficial, mas pelo IAB - Internet Activity Board.

Arquitetura Internet dá ênfase à interligação de redes com diferentes tecnologias.

A interligação de redes é feita por uma máquina conectada a ambas as redes, responsável por transferir mensagens de uma rede para outra. Esta máquina é chamada de *internet gateway* ou *internet router*.

Os gateways precisam conhecer a forma de ligação (topologia) da inter-rede (fig. 3.7), para rotear corretamente as mensagens de forma transparente ao usuário.

Fig. 3.7 - Ilustração da inter-rede

A arquitetura TCP/IP é formada por quatro camadas, que se posicionam sobre a camada intra-rede que não faz parte do modelo (fig. 3.8).

Fig. 3.8 - Arquitetura Internet TCP/IP

- *nível de aplicação*: usuários usam programas de aplicação para acessar serviços disponíveis na inter-rede. Aplicações interagem com nível de transporte para enviar e receber dados. As aplicações podem utilizar-se do serviço de transporte orientado a conexão - TCP (serviço de circuito virtual) ou do User Datagram Protocol - UDP, não orientado a conexão.
- *nível de transporte*: permite a comunicação fim-a-fim entre aplicações. Com o protocolo TCP os serviços são: controle de erro, controle de fluxo, sequenciação e multiplexação do acesso ao nível inter-rede. Com o protocolo UDP tem-se apenas multiplexação/ de multiplexação do acesso ao nível inter-rede.
- *nível inter-rede*: responsável pela transferência de dados pela inter-rede, da máquina origem à destino. Recebe do nível de transporte os pedidos de transmissão de pacotes já com o endereço de onde devem ser entregues. O pacote é encapsulado em um datagrama IP e o algoritmo de roteamento é executado. Este nível também processa pacotes recebidos das interfaces de rede, realizando o algoritmo de roteamento para recebê-lo ou transmiti-lo.
- *nível de interface de rede*: compatibiliza a tecnologia despecífica de cada rede ao protocolo IP. Recebe os datagramas IP do nível de inter-rede e os transmite através da rede específica. Neste caso os endereços IP que são lógicos são transformados em endereços físicos dos hosts ou dos gateways a retransmiti-los.

Algumas das principais aplicações disponível na Internet TCP/IP são:

- Simple Mail Transfer Protocol - SMTP: serviço store-and-forward de mensagens.
- File Transfer Protocol - FTP: serviço de transferência de arquivos.
- TELNET: serviço de terminal virtual.
- Domain Name System - DNS: serviço de mapeamento de nomes em endereços de rede.

COMPARAÇÃO ENTRE OSI E INTERNET

As diferenças entre os modelos RM-OSI e INTERNET advém de suas filosofias de construção. O RM-OSI foi desenvolvido formalmente de forma a tratar todos os aspectos do problema de interconexão aberta de sistemas. O modelo Internet foi desenvolvido com o objetivo de resolver um problema prático - interligar redes com tecnologias distintas, buscando uma solução simples e funcional para o problema de interconexão de sistemas abertos.

Uma comparação das duas arquiteturas se vê na figura 3.9.

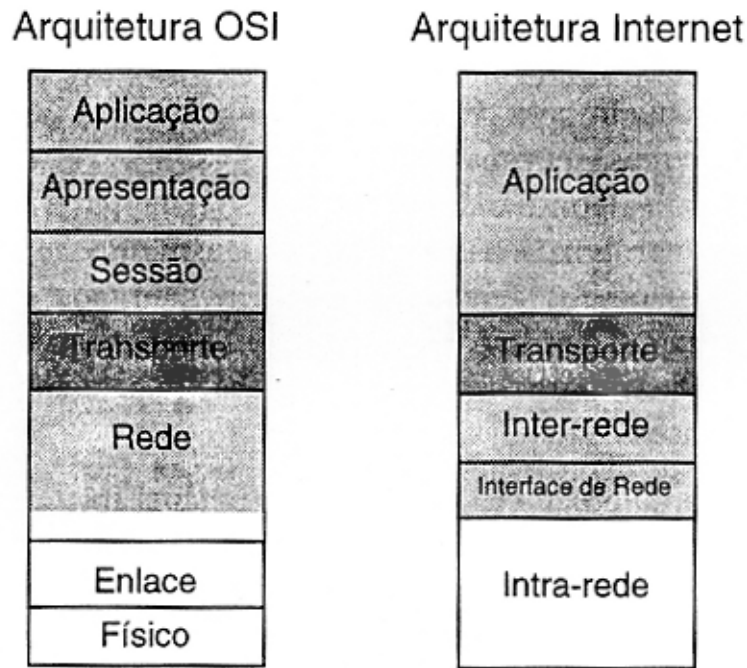


Fig. 3.9 - Arquiteturas OSI e Internet TCP/IP

O NÍVEL DE ENLACE

O Nível de Enlace tem por principais funções detectar e, opcionalmente, corrigir erros que possam ter ocorrido ao nível físico. Também pode executar um controle de fluxo de dados, oferecer os serviços de conexão, controlar o acesso ao meio de transmissão e realizar multiplexação de vários usuários de rede.

Para tal, a cadeia de bits (quadros) trocada entre o nível físico e o de enlace contém uma sequência de bits adicionais denominados *Frame Check Sequence - FCS*, que permite a detecção de erros.

Os bits adicionais do FCS são calculados por um algoritmo (redundância cíclica, paridade, etc.) e agregados ao quadro. Quando o destinatário recebe o quadro, calcula o FCS e o compara ao agregado no quadro. Caso haja erro o quadro é descartado e opcionalmente enviado um aviso ao sistema transmissor.

Há a possibilidade do nível de enlace usar bits de redundância que permitam também a correção do erro sem necessidade de retransmissão, caso bastante útil com enlaces de alto custo e de alto retardo de transferência (enlaces de satélite).

DELIMITAÇÃO DE QUADROS

A criação e reconhecimento dos quadros são feitas essencialmente de quatro maneiras:

- contagem de caracteres*: um campo no cabeçalho do quadro indica o número de caracteres do quadro.
- caracteres delimitadores e transparência de caracteres*: utiliza caracteres especiais para indicação do início e término do quadro.

No caso do carácter delimitador aparecer no meio do quadro é inserido (stuffed) antes dele um outro carácter especial que o torna transparente como delimitador.

- c. *flags e transparência de bits*: utiliza sequências especiais de bits, denominadas flags, para delimitar quadros e transparência de bits (bit stuffing) para evitar que estas sequências sejam reconhecidas como delimitadores quando localizadas no conjunto dos dados. O protocolo HDLC usa 01111110 como delimitador. Quando 5 '1s' consecutivos são encontrados nos dados, é inserido (stuffed) um '0' que é retirado quando do recebimento.
- d. *violação de códigos do nível físico*: método usado quando a codificação dos bits a nível físico tem alguma redundância. Os símbolos não utilizados para representação de dados são usados para delimitação de quadros.

CONTROLE DE ERROS NO ENLACE

O controle de erros no meio físico pode ser feito pelo reconhecimento de quadros com erro. Para isto o nível de enlace pode enviar ao transmissor quadros especiais indicando reconhecimento positivo de quadro (recebido correto) ou negativo (recebido com erro), caso em que o quadro deve ser retransmitido.

O reconhecimento negativo, embora desnecessário, acelera a retransmissão, diminuindo o retardo. A transmissão do reconhecimento tanto pode ser um quadro de controle de nível 2 como ser agregado a um campo de controle de um quadro de informação (piggybacking).

Para o caso do quadro de reconhecimento ser degradado, o transmissor usa um temporizador para medir o intervalo de tempo (timeout) suficiente para receber o reconhecimento. Caso o reconhecimento não chegue, o quadro é retransmitido. Como os quadros podem ser retransmitidos, estes devem ser numerados para permitir reconhecimento entre original e cópia.

Os procedimentos para controle de erros são:

- algoritmo de bit alternado (stop-and-wait): só é enviado um novo quadro pelo transmissor após receber o reconhecimento do quadro anterior. Neste caso apenas um bit é necessário para identificar um quadro de seu antecessor. O 1º quadro recebe bit '0', o 2º '1' e o 3º novamente bit '0'. A figura 4.1 representa o diagrama de estados tanto do transmissor como do receptor.

Fig. 4.1 - Diagrama de estados do algoritmo de bit alternado

A técnica de bit alternado é simples porém ineficiente, uma vez que o canal fica inoperante enquanto o transmissor espera por reconhecimento. A figura 4.2 mostra um exemplo de troca de quadros neste protocolo.

- janela n com retransmissão integral (go-back-n): desenvolvido para aumentar a eficiência do protocolo de bit alternado. Permite que o transmissor envie vários quadros sem ter recebido reconhecimento de quadros anteriores. A largura da janela de transmissão corresponde ao número máximo de quadros, devidamente numerados, que podem ser enviados antes da chegada do reconhecimento.

Fig. 4.2-Controle de erro por timeout no protocolo de bit alternado

Quando ocorre um erro em algum quadro não há envio do reconhecimento. Se o transmissor não recebe o reconhecimento, todos os quadros da janela são retransmitidos. A parte a da figura 4.3 representa este protocolo.

- janela n com retransmissão seletiva (selective repeat): neste caso o transmissor apenas retransmite o quadro onde houve erro. A parte b da figura 4.3 representa este protocolo.

Nos dois últimos protocolos a eficiência pode ser aumentada ao se considerar que o receptor pode enviar o reconhecimento do último quadro recebido corretamente. Neste caso considera-se que os anteriores a ele também foram recebidos corretamente.

CONTROLE DE FLUXO DE ENLACE

O nível de enlace também trata a situação onde o transmissor, sistematicamente, envia quadros mais depressa que o receptor pode receber. Mecanismos de controle de fluxo compatibilizam as velocidades de transmissão e recepção.

Fig. 4.3 - Protocolos de janela n

No protocolo de bit alternado o próprio mecanismo de retransmissão de incumbe de controlar o fluxo, pois o novo quadro só é enviado após o transmissor receber o reconhecimento.

Nos protocolos com janela n ($n > 1$) o controle se faz por janelas de transmissão e de recepção, num mecanismo chamado de janela deslizante (Sliding Windows).

O número máximo de quadros que o transmissor pode enviar sem receber reconhecimento é determinado pela largura (T) de janela de transmissão. Após enviar T quadros sem receber reconhecimento, o transmissor aguarda até receber o reconhecimento, que além de indicar a correção dos quadros recebidos indica a disponibilidade em receber novos quadros.

A figura 4.4 ilustra este mecanismo de controle.

Fig. 4.4 - Controle de fluxo em protocolo de janela n ($n=4$)

Notar que, para não confundir o transmissor, se a numeração dos quadros vai de 0 a n o tamanho da janela deve ser igual a n.

A janela de recepção pode ser menor que a de transmissão ($T \geq R$). Neste caso, o receptor, ao completar o tamanho de sua janela de recepção de tamanho R, envia um quadro especial (*receive-not-ready*) ao transmissor indicando que não está em condições de receber novos quadros. O transmissor suspende a transmissão de quadros. Após isto, o receptor envia os quadros para o nível de rede, e reabilita a transmissão enviando outro quadro especial ao transmissor (*receive-ready*).

TIPOS DE SERVIÇOS

O nível de enlace fornece três tipos de serviço ao nível de rede:

- *serviço sem conexão e sem reconhecimento* (datagrama não confiável): apropriado a redes com taxa de erros a nível físico muito baixas. A correção fica para camadas superiores. Adequado para aplicações onde a demora na transmissão pode ser pior que a perda de parte dos dados (ex. transmissão de voz em tempo real).
- *serviço sem conexão com reconhecimento*: adequado quando um pequeno volume de dados deve ser transferido de forma confiável. O receptor envia ao transmissor quadros com reconhecimento (positivo ou negativo) das mensagens recebidas. A possibilidade de perda de quadros com o reconhecimento exige um mecanismo de eliminação de quadros duplicados.
- *serviço orientado a conexão*: garante a entrega de quadros ao receptor sem erros e na ordem em que foram enviados. Também garante que não serão entregues quadros duplicados à camada de rede. Este serviço converte o canal não confiável, oferecido pelo nível físico, em um canal confiável para uso do nível de rede.

CONTROLE DE ACESSO

O nível de enlace se responsabiliza pelo controle de acesso ao meio de transmissão. Para isto se utiliza de mecanismos que disciplinam o acesso ao meio físico.

Diferentes métodos de controle de acesso ao meio de transmissão, bem como os diferentes padrões para isto, serão estudados detalhadamente.

MULTIPLEXAÇÃO

Como a estação tem o controle o acesso ao meio pode multiplexar o acesso à conexão física para vários usuários do nível de rede.

PROTOSCOLOS DE ACESSO AO MEIO

Os protocolos de acesso ao meio, regras para acesso ao meio físico, são funções da camada de enlace do modelo OSI. Na maioria das vezes foram desenvolvidos para uma topologia particular de rede, embora possam ser usados em qualquer topologia.

A avaliação de protocolos de controle de acesso considera atributos específicos como:

- *capacidade*: vazão máxima, em porcentagem da banda passante disponível, que o método de acesso pode tirar do meio. Variáveis que podem afetar a capacidade são: taxa de transmissão, comprimento da rede, número de nós, tamanho do quadro, tamanho do cabeçalho, retardo em cada estação.
- *justiça ou equidade (fairness)*: desejável na maioria das redes. Permite que as estações venham a ter acesso aos recursos compartilhados. Justiça não implica em inexistência de prioridade. Cada estação deve ser tratada com igualdade dentro de sua classe de prioridade.
- *prioridade*: desejável principalmente nas aplicações de tempo real.
- *estabilidade em sobrecarga*: característica desejável com carregamento pesado da rede. Protocolos com acesso separado por nó são estáveis e não apresentam variações no retardo. Aqueles baseados em contenção são dependentes da implementação, às vezes necessitando tratamento de conflitos para os tornarem estáveis.
- *retardo de transferência*: soma dos retardos de acesso e transmissão. Embora, no geral, se apresente como variável aleatória, em alguns protocolos tem valor máximo limitado.

Os métodos de acesso podem ser divididos em duas classes: a dos métodos baseados em contenção e a daqueles com acesso ordenado sem contenção.

1- MÉTODOS DE ACESSO BASEADO EM CONTENÇÃO

Neste caso não existe uma ordem de acesso e nada impede que dois nós provoquem colisão, acarretando perda de mensagem. A estratégia de controle depende de como a estação detecta colisão e retransmite a mensagem.

1.1 - ALOHA

Desenvolvido para a rede Aloha, interligando o Centro de Computação da Universidade do Havaí a terminais espalhados pelas ilhas. Não é uma rede local, mas inspirou a maioria dos protocolos de contenção.

A rede Aloha tem dois canais de rádio, um transmitindo do computador para um terminal e outro dos terminais para computador, onde pode ocorrer conflitos.

Cada terminal só pode ouvir o canal do computador. Quando quer transmitir, simplesmente transmite, independentemente da ocupação do canal.

A detecção de colisão é feita pelo disparo de um temporizador aleatório durante a transmissão. Caso um quadro de reconhecimento de transmissão não chegue do computador, o quadro original é retransmitido. A aleatoriedade do temporizador é para reduzir probabilidade de novas colisões.

Um aperfeiçoamento consiste em fazer com que quadros colidam por inteiro, minimizando o tempo total de colisão. Nesta técnica, denominada *slotted-aloha*, o computador divide o tempo em intervalos (*slots*) iguais. Cada terminal apenas pode começar a transmitir no início de um slot. Esta restrição impõe um certo retardo no início das transmissões de quadros.

O método Aloha de detecção de colisão limita a capacidade de utilização da rede a 18% enquanto o Slotted-Aloha a 37%.

A figura 5.1 representa estas duas técnicas.

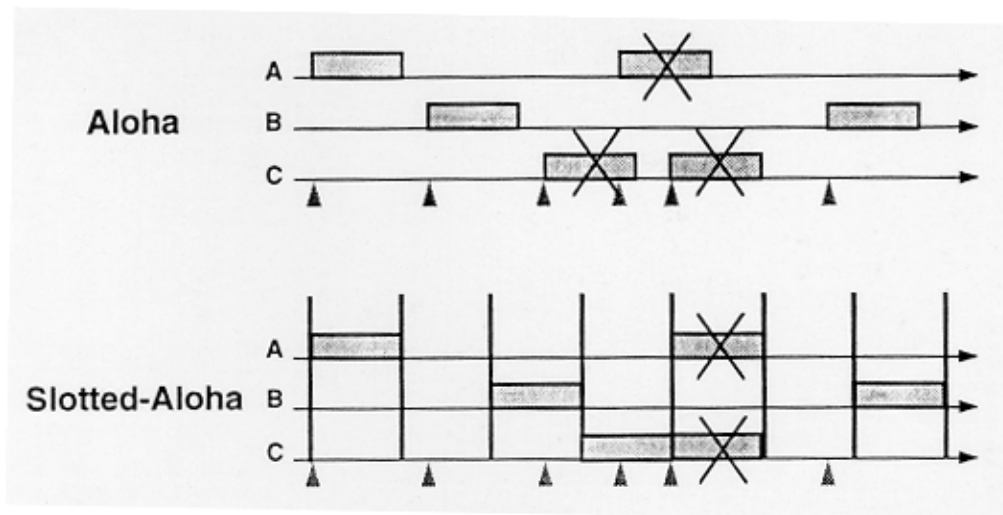


Fig. 5.1 - Técnicas Aloha e Slotted-Aloha de controle de acesso

Nestas técnicas não existe prioridade de acesso, embora isto possa ser conseguido com o controle do tempo do temporizador. Também não garantem um limite para o máximo de retardo.

A grande vantagem destas técnicas é sua simplicidade. Elas são adequadas em aplicações com tráfego pequeno e onde prioridade e tempo de resposta limitado não são importantes.

1.2 - CSMA

O método Acesso Múltiplo com Sensoriamento de Portadora - CSMA (*Carrier-Sense Multiple Access*) busca ao máximo evitar a colisão de quadros e, caso ocorram, que colidam por inteiro. Em algumas variantes, ao detectar a colisão aborta a transmissão.

Quando uma estação quer transmitir, ela primeiro ouve o meio para saber se existe transmissão. Se não existir transmissão, então tem permissão de transmitir, caso contrário espera um tempo. A colisão só ocorre quando há simultaneidade no sensoriamento do meio.

Várias estratégias são usadas para aumentar a eficiência do método, como: np-CSMA, p-CSMA e CSMA-CD.

1.2.a - np-CSMA e p-CSMA

Nestas estratégias, as estações, após transmitirem, esperam o reconhecimento da mensagem por um tempo referente ao retardo de propagação de ida-e-volta e o fato de que a estação que deve enviar o reconhecimento também disputa o direito de acesso ao meio. A não chegada de um reconhecimento implica em uma colisão. A distinção entre os dois métodos é o algoritmo que especifica a ação de cada uma ao encontrar o meio ocupado.

Na np-CSMA (non-persistent Carrier-Sense Multiple Access), quando a máquina percebe o meio ocupado com uma transmissão, fica esperando por um intervalo de tempo aleatório, antes de tentar novamente a transmissão.

A figura 5.2 representa a estratégia de controle não persistente.

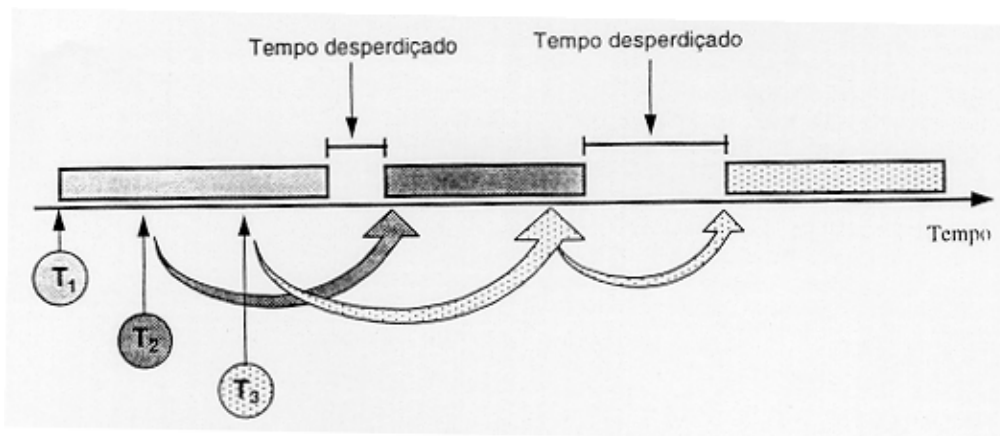


Fig. 5.2 - Método de acesso np-CSMA

Na p-CSMA (*p-persistent Carrier-Sense Multiple Access*) a estação continua a escutar o meio até que este fique livre. Então transmite com uma probabilidade p ou espera por um intervalo fixo de tempo e então transmite, com probabilidade p , ou continua a esperar por outro intervalo, com probabilidade $(1 - p)$, seguindo assim até transmitir ou até que uma outra estação ganhe o acesso ao canal. Um caso especial é o 1-persistente, quando a máquina escuta o meio até ficar livre e então certamente transmite.

A figura 5.3 representa a estratégia CSMA persistente.

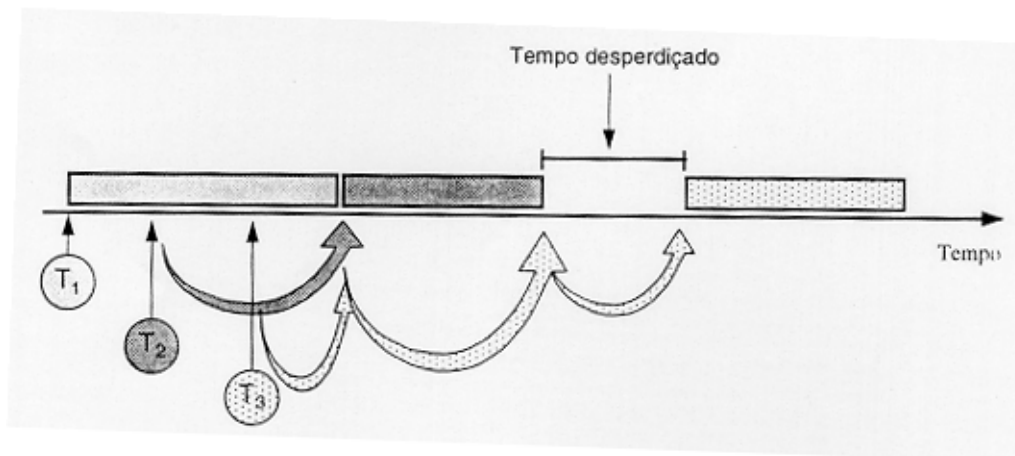


Fig. 5.3 - Método de acesso CSMA persistente

O algoritmo não persistente é eficiente para evitar colisões, pois estações esperando pelo meio, provavelmente vão esperar por intervalos de tempo diferentes. No entanto, pode haver um tempo perdido por o canal ficar livre. No algoritmo 1-persistente esta perda de tempo é eliminada, mas pode haver colisão se mais de uma estação está no aguardo. O algoritmo p-persistente apresenta um ponto intermediário destes casos.

Estas estratégias permitem, em tráfego baixo, uma capacidade em torno de 85%.

Como no Aloha, não existe prioridade de acesso, que, no entanto, pode ser controlada pelo temporizador.

Em tráfego pesado, ambas as técnicas exibem instabilidade, ao apresentarem grande taxa de colisão e grande retardo.

1.2.b - CSMA/CD

No método CSMA/CD (*Carrier-Sense Multiple Acces with Collision Detection*) a detecção de colisão é feita durante a transmissão. O nó escuta a rede durante toda a transmissão e a aborta assim que ocorra colisão, então espera por um tempo para tentar a retransmissão.

Um fator limitante do método é a necessidade que o quadro tenha um tamanho mínimo para que uma colisão possa ser detectada por todas as estações. Se t_p é o tempo de propagação, C a taxa de transmissão e M o tamanho mínimo do quadro tem-se:

$$M \geq 2 C t_p \text{ para redes em banda básica e,}$$

$$M \geq 4 C t_p \text{ para redes em banda larga.}$$

Detectada a colisão, duas técnicas de retransmissão podem ser utilizadas:

- *espera aleatória exponencial truncada (truncated exponential back off)*: ao detectar a colisão, a estação espera por um tempo aleatório de 0 a um limite superior. A cada nova colisão o limite é dobrado, de forma a controlar o canal e mantê-lo estável mesmo com tráfego alto. A duplicação é detida em um certo ponto.
- *retransmissão ordenada (orderly back off)*: após a detecção da colisão, as estações só podem começar a transmitir em intervalos de tempo a eles determinado previamente. Terminada a transmissão das mensagens colididas, a estação com o menor intervalo tem direito de transmitir sem colisão. Se não o fizer passa seu direito à segunda, e sucessivamente até que ocorra uma transmissão, quando o algoritmo CSMA/CD se reinicia.

Neste esquemas o retardo limitado também não pode ser garantido.

O CSMA/CD com espera aleatória exponencial truncada tornou-se padrão internacional (ISO 8802-3/IEEE 802.3), sendo atualmente o mais popular, com larga escala de produção em chipset para comunicação.

1.2.c - CSMA/CA

O algoritmo CSMA/CA (*Carrier-Sense Multiple Access with Collision Avoidance*) considera que a cada transmissão, com ou sem colisão, a rede entra em modo onde as estações só podem começar a transmitir em intervalos de tempo a elas determinados. Ao acabar uma transmissão, a estação com o primeiro intervalo de tempo tem direito de transmitir sem colisão. Caso não o faça, o direito passa para a estação com o segundo intervalo de tempo, e assim sucessivamente até que ocorra uma transmissão, quando o processo se re-inicia. Se nenhuma estação transmite, a rede entra em CSMA comum, podendo ocorrer colisões. Uma transmissão ou colisão que ocorra retorna ao algoritmo.

Alguns aperfeiçoamentos podem ser considerados:

- conforme ANSI, se nenhuma estação transmitir nos intervalos estabelecidos, ao invés de ir para CSMA a estação com mais alta prioridade transmite um quadro sem informação, reiniciando o processo.
- acrescenta justiça ao algoritmo, obrigando que uma estação, após transmitir, só possa tentar nova transmissão depois que todas as outras tenha tido esta oportunidade.
- permite diálogo entre estações. Depois de cada transmissão, a estação destino tem o direito de transmitir no primeiro intervalo de alocação. Caso ela não responda, segue-se o algoritmo normal. Corre-se, com isto, o risco que duas estações monopolizem o meio.

1.2.d - M-CSMA

O algoritmo M-CSMA (*Multichannel Carrier-Sense Multiple Access*) utiliza de múltiplos canais em paralelo. Permite alta capacidade de transmissão agregada, embora a taxa de transmissão individual de cada canal tenha baixos valores.

Os múltiplos canais podem ser implementados com mais de um fio ou fibra, e mesmo com técnicas de multiplexação como TDM (*Time Division Multiplexing*), FDM (*Frequency Division Multiplexing*) e WDM (*Wavelength Division Multiplexing*).

Tem-se duas alternativas para a seleção do canal de transmissão. O canal pode ser selecionado aleatoriamente, e depois inspecionado para verificação de ocupação, ou seleciona-se um canal que esteja livre no momento.

O ganho de desempenho com o paralelismo é grande, uma vez que a probabilidade de que duas estações transmitam ao mesmo tempo no mesmo canal é baixa, diminuindo com o aumento do número de canais.

1.3 ReC-RING

No algoritmo ReC-Ring (*Resolvable Contention Ring*) uma estação só começa a transmissão quando percebe o canal desocupado. O quadro transmitido desloca-se em uma só direção, percorre todo o anel sendo retirado pela estação origem. Quando uma estação verifica que o anel transporta um ou mais quadros, aguarda o final para poder transmitir.

As estações conhecem sua posição relativa no anel. Uma colisão ocorre quando uma estação estiver transmitindo e chegar um quadro de estação anterior a ele no anel. Ao detectar a colisão, o nó suspende a transmissão, compara seu índice com o da outra estação transmissora. Caso seu índice seja maior, espera o fim do trem de quadros, caso contrário descarta o quadro recebido e retransmite o seu.

2 - ACESSO ORDENADO SEM CONTENÇÃO

Os métodos de acesso ordenado ao meio de comunicação evitam o problema de colisão, além de terem a característica de oferecer um retardo de transferência limitado. Cada método é mais adequado a uma topologia, mas podem ser utilizados em outras.

2.1 - POLLING

Geralmente usado em topologia de barramento comum. As estações só transmitem quando interrogadas pelo nó central, o controlador da rede. Caso não tenha quadros a transmitir envia um quadro de status, avisando estar em operação.

Outra forma de polling é eficiente para barramentos muito grandes e com baixa frequência de transmissão. O nó controlador consulta a estação mais distante se tem quadros a transmitir. A

estação pode responder enviando o quadro, sendo que o controlador reassume o controle após o término da transmissão, consultando a estação seguinte. Caso a estação não precise transmitir, o controlador consulta a estação seguinte.

Diversas variantes ao algoritmo de polling podem aumentar sua eficiência: criação de barramentos de controle, para eliminar os quadros de controle da rede; multiplexar em frequência as informações de controle com as de dados; estações fazerem requisições de transmissão ao invés de serem consultadas (para poucas estações com este propósito); estações com pouco tráfego serem consultadas com maior espaçamento, etc.

Nesta técnica as prioridades podem ser estabelecidas e o retardo de transferência é limitado. No entanto, apresenta todos os inconvenientes da centralização, principalmente a questão de confiabilidade.

Polling é adequado quando as estações têm comportamento muito bem conhecido, a ponto de serem usados no estabelecimento da metodologia de passagem do controle.

2.2 - SLOT

Desenvolvido para topologia em anel, recebe nome de Anel de Pierce (Pierce 1972) ou Anel Segmentado.

O método divide o espaço de comunicação em pequenos segmentos (*slots*) dentro dos quais as mensagens podem ser armazenadas. Cada slot contém um bit indicando se está cheio ou vazio, como representado pela figura 5.4. Ao querer transmitir, a estação espera por um slot vazio e o preenche com sua mensagem.

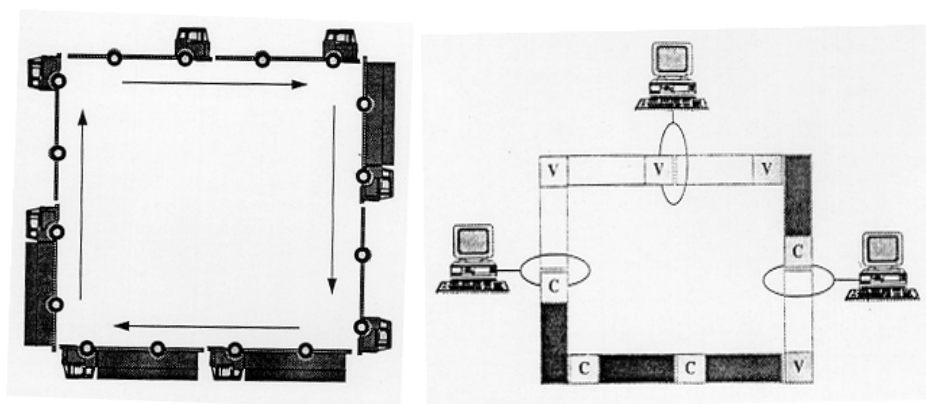


Fig. 5.4 - Ilustrações de um anel de slots

O número de slots que circulam no anel é invariante. A soma dos retardos nos repetidores mais o tempo de propagação no anel forma a *latência do anel*. Podem existir tantos bits circulando no anel quanto a latência do anel permitir. A latência sempre pode ser ampliada pela adição de um buffer de retardo em cada estação. Na inicialização dos slots, um dispositivo gera, inicializa como vazios e inicializa bits de controle

Toda estação sabe o número de slots no anel. Assim a estação pode esperar sua mensagem retornar, e retirá-la ao esvaziar o mesmo slot que havia preenchido. Outra alternativa seria a própria estação destino esvaziar o slot, o que aumentaria a eficiência da utilização do anel mas reduziria a confiabilidade e aumentaria a latência do anel. Neste caso, para garantir justiça, a estação que esvaziou o slot não o poderia preencher, podendo aumentar a ineficiência do anel.

O dimensionamento de slots requer um compromisso de equilíbrio. Gasta-se tempo para dividir os quadros variáveis nos tamanhos fixos dos slots, o que sugeriria a construção de slots grandes, aumentando-se a latência através de buffers de retardo. No entanto, com slots grandes, a eficiência pode diminuir por ter-se que esperar muito tempo para um slot passar. Também, com slots grandes, alguns seriam apenas parcialmente preenchidos, causando ineficiência.

Neste sentido, uma condição de equilíbrio tem levado a slots geralmente pequenos, como exemplo o anel de Cambridge com 38 bits sendo apenas 16 bits de dados.

Outra questão importante corresponde à necessidade da centralização de alguns controles. Se por algum ruído o bit de preenchimento é trocado, um slot vazio pode ficar circulando indefinidamente como se estivesse cheio. Se por algum motivo uma estação falhar em esvaziar o slot, este acabará também ficando indefinidamente na rede. Devido a isto alguma estação deve ser estabelecida como monitora para corrigir tais problemas.

2.2.a - Fasnet

A rede Fasnet é uma variante do método de slots, usando duas barras de transmissão unidirecionais. As estações utilizam as duas barras para receber e para transmitir, como mostra a figura 5.5.

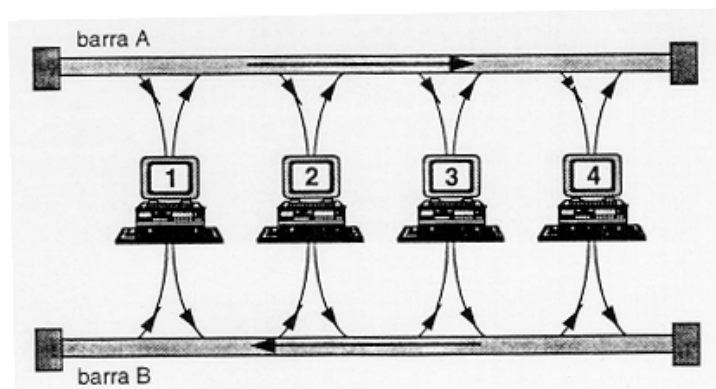


Fig. 5.5 - Topologia da Fasnet

Uma estação usa a barra A para transmitir dados para uma estação com índice maior que o seu, e usa a barra B quando o índice do destino for menor que o seu. A estação de menor índice inicia os slots no sentido crescente, enquanto a estação de maior índice os inicia no outro sentido.

Como exemplo da sequência de controle de uma das barras tem-se: a primeira estação em um sentido gera um trem de slots. As estações buscam realizar suas transmissões. Caso os dados não caibam em um só slot, podem usar slots consecutivos até sua cota de transmissão. Depois que todas as estações tenham esgotado sua cota de transmissão em um sentido, a última estação recebe um slot vazio. Isto a faz enviar, à primeira estação, um slot reconhecendo o fim dos dados (slot com end-bit igual a 1). Ao receber este slot, a primeira estação gera novo trem de slots.

2.2.b - ATM Ring

O método ATMR (*Asynchronous Transfer Mode Ring*), é usado em redes de duplo anel, com transmissão simultânea e em sentido inverso nos dois aneis, que são acessados por slots. O nó destino libera o slot recebido, podendo utilizá-lo. O ATMR foi implementado em um chip, operando em taxa de 622Mbps.

Os usuários são ligados ao anel através de unidades de controle de acesso. Quando um usuário quer transmitir, faz uma solicitação onde define a banda passante de que necessita. Com base neste pedido, é estabelecida uma janela de acesso de certo tamanho. A soma do tamanho das janelas de acesso dos usuários em uma unidade de acesso define o tamanho da janela de acesso da unidade.

Quando todas as unidades do anel param de transmitir é necessário restaurar o valor das janelas de transmissão. Para detectar a inatividade, usa-se o *identificador de atividade* (*Access Control Field - ACF*), colocado em um dos campos do cabeçalho das células.

Uma unidade que ainda não esgotou sua transmissão escreve seu identificador no ACF de todas as células que passam, enquanto as inativas não alteram o campo ACF. Quando a unidade recebe uma célula com o seu identificador, reconhece que as outras estão inativas, termina sua transmissão e inicia o ciclo de restauração das janelas (reset).

Restabelece-se a transmissão restaurando-se o valor das janelas de transmissão das unidades.

2.3 - INSERÇÃO DE RETARDO

A figura 5.6 esquematiza o método de inserção de retardo.

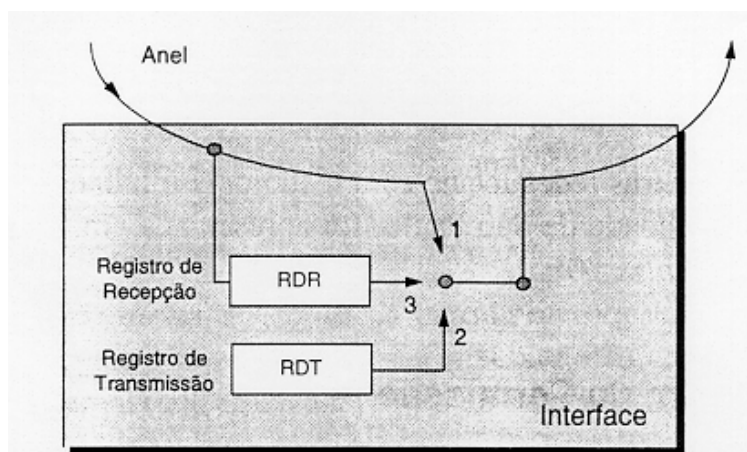


Fig. 5.6 - Anel com inserção de retardo

Quando um quadro deve ser transmitido, é posto no registro de transmissão. A chave é passada da posição 1 para 2 ao final do quadro que esteja sendo recebido ou quando não houver transmissão. Terminada a inserção do quadro, passa-se para a posição 3, dando vazão aos dados recebidos, agora com retardo de um quadro. A chave só é passada para posição 1 quando um quadro seu retorna completo, sendo então retirado do anel.

Como outras redes em anel, algumas funções de controle devem ser centralizadas, devido a problemas como quadros que ficam indefinidamente no anel por ter endereço de origem adulterado ou por estação que saiu do anel.

Embora o retardo de transmissão seja variável, o retardo de transferência é limitado, sendo adequado a aplicações com esta exigência.

2.3 - PASSAGEM DE PERMISSÃO

Esquema de controle onde um padrão especial (*token*) é passado sequencialmente como permissão de transmissão de uma estação a outra. Somente a interface com a permissão pode transmitir em um dado instante.

Embora se adapte muito bem a anel e barramento este esquema de controle pode ser usado em outras topologias.

2.3.a - Passagem de Permissão em Barramento (*Token Bus*)

Em redes com topologia em barra, assim que acaba a transmissão de uma estação esta passa a permissão para a próxima a transmitir. A permissão é um padrão variável, identificando a próxima estação. A permissão é passada até que se feche um ciclo, quando recomeça, simulando um anel virtual.

A ordem física não guarda relação com a ordem lógica do anel virtual, sendo que estações não pertencentes ao anel virtual, embora não possam transmitir, recebem quadros.

As principais funções são:

- adição e retirada de estações do anel virtual - estações devem poder ser inseridas e retiradas do anel virtual, evitando passagem desnecessária de token.
- gerenciamento de falhas - correção de perda de token (ruídos, falhas de transmissão ou recepção, passagem para estação que deixou o anel), estações com endereços duplicados.
- Inicialização do anel virtual - na partida da rede ou na recuperação de token perdido.

Por apresentar um retardo de transferência limitado, este método é desejável para uso em aplicações de controle de processos e outras aplicações de tempo real. Este método foi tornado padrão internacional (ISO 8802-4/IEEE 802/4), sendo adotado na arquitetura MAP para ambientes industriais.

Uma desvantagem do método é o overhead quando o tráfego é baixo.

Uma variante do método de passagem de permissão em barramento é a rede ***Expressnet***.

Cada estação se liga duas vezes ao mesmo barramento com transmissão unidirecional (transmissão e recepção) de forma que a ordenação das estações na transmissão se repita na recepção, como mostra a figura 5.7. A sequência do token é simulado pela ordem física das estações.

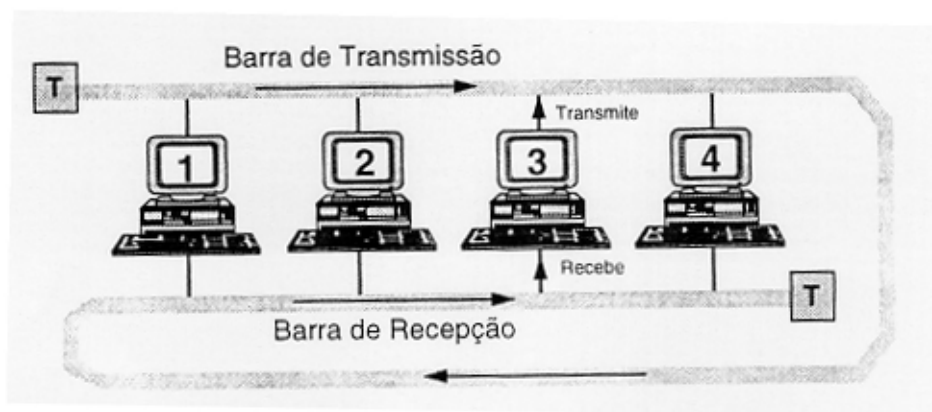


Fig. 5.7 - Topologia da Expressnet

Quando uma estação deseja transmitir, ela escuta a barra com um sensor colocado anteriormente a seu ponto de conexão. Acontecendo da barra estar livre, transmite. Quando uma estação termina a transmissão, todas as estações posteriores a ela na barra de transmissão, e que tiverem quadros a transmitir, começam a transmitir. Quando da ocorrência de colisão, todas as estações interrompem sua transmissão, exceto a primeira (posição superior) esperando por nova oportunidade. Ao terminar a transmissão uma estação não mais escuta a barra, aguardando por um novo ciclo.

Um ciclo é formado por um trem de quadros separados entre si por intervalos de tempo menores ou iguais ao tempo de propagação do sinal entre as estações mais distantes na barra (t_p). O fim de trem ocorre quando a estação detecta ociosidade na barra por um tempo maior que t_p , situação que a estação mais próxima do início da barra detecta primeiramente.

Notar que neste método não há passagem explícita da permissão de transmissão, pois o nó ativo mais próximo do início da barra inicia o trem de quadros e passa implicitamente a permissão relativamente à posição de cada estação na barra.

Este método resolve o problema da colisão, mas não evita a perda de alguns bits no início de uma transmissão. 2.3.b - Passagem de Permissão em Anel (*Token Ring*)

Técnica de controle mais antiga (1969) para anel, baseia-se em um pequeno quadro contendo a permissão (padrão fixo) circulando pelo anel, denominado *permissão livre*.

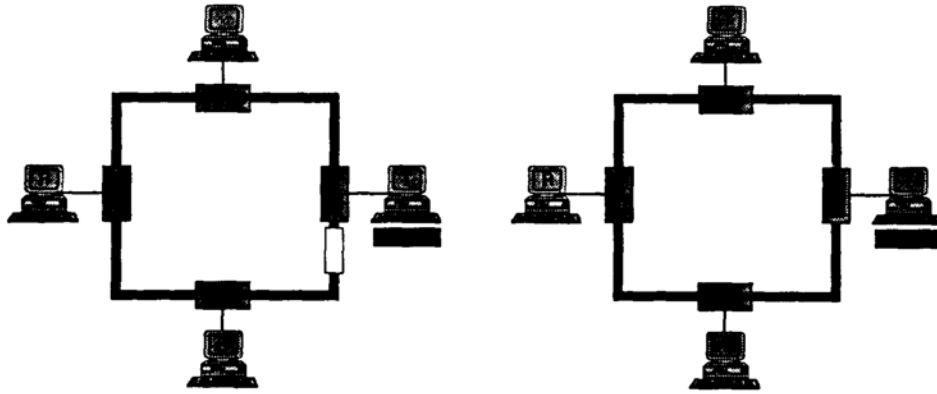
Uma estação aguarda a permissão e quando a recebe altera o padrão para *permissão ocupada*, passando a transmitir seus quadros. Ao receber seus quadros a estação os retira do anel e insere nova permissão livre.

O tipo de inserção da permissão livre no anel pode ser: *single packet*, *single token* e *multiple token*.

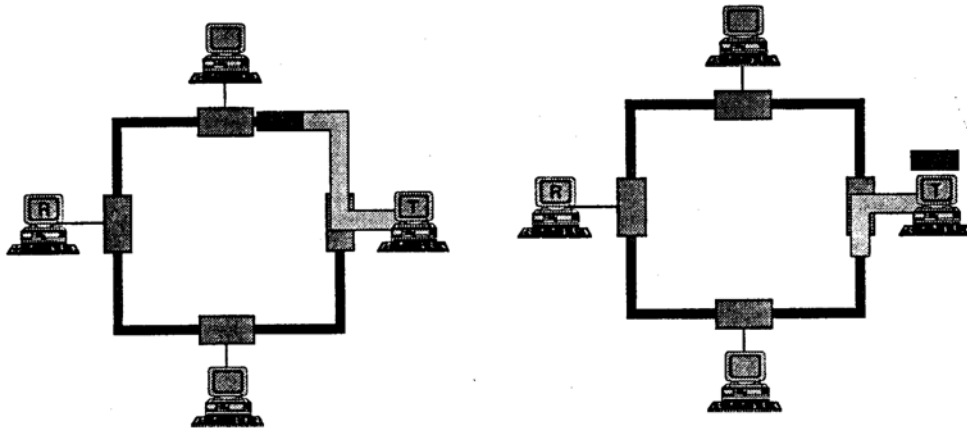
No modo *single packet*, o transmissor só insere a permissão livre no anel depois de receber de volta a permissão ocupada e retirar seu quadro do anel, havendo apenas um quadro e uma permissão circulando no anel. A figura 5.8 representa este modo de inserção.

No modo *single token* a permissão livre é inserida no anel pela estação transmissora assim que recebe de volta a permissão ocupada, adiantando o tempo de recepção e retirada de seu quadro. Neste caso, embora mais de um quadro possa estar circulando no anel, existe uma única permissão circulando. Este método é representado pela figura 5.9.

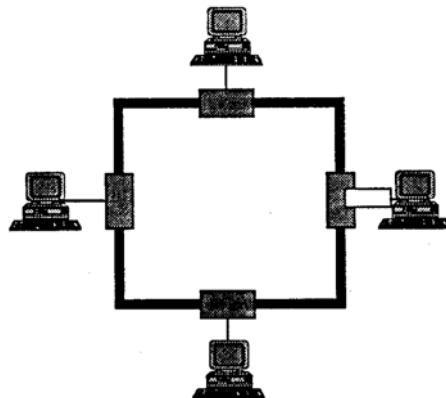
No modo *multiple token* o transmissor insere uma permissão livre no anel imediatamente após terminar de transmitir o último bit de seu quadro. Neste caso pode haver vários quadros e várias permissões simultaneamente no anel, mas apenas uma permissão livre. A figura 5.10 representa este método.



a) estação recebe permissão livre b) estação muda permissão p/ ocupada

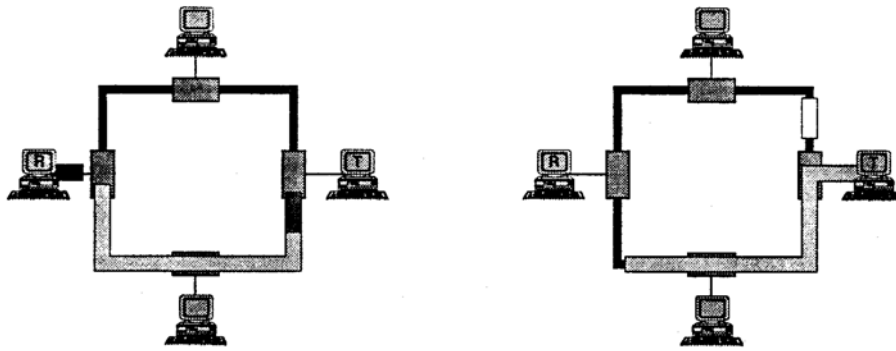


c) estação transmite mensagem d) estação retira sua mensagem do anel



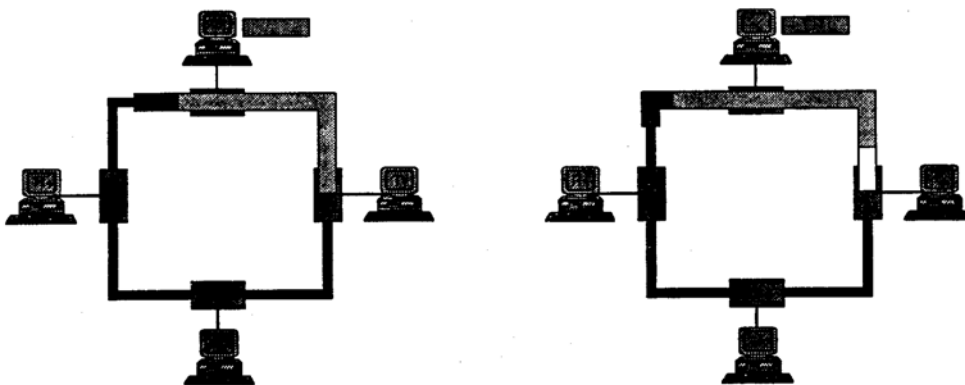
e) estação insere permissão livre no anel

Fig. 5.8 - Método de acesso Token Ring single packet

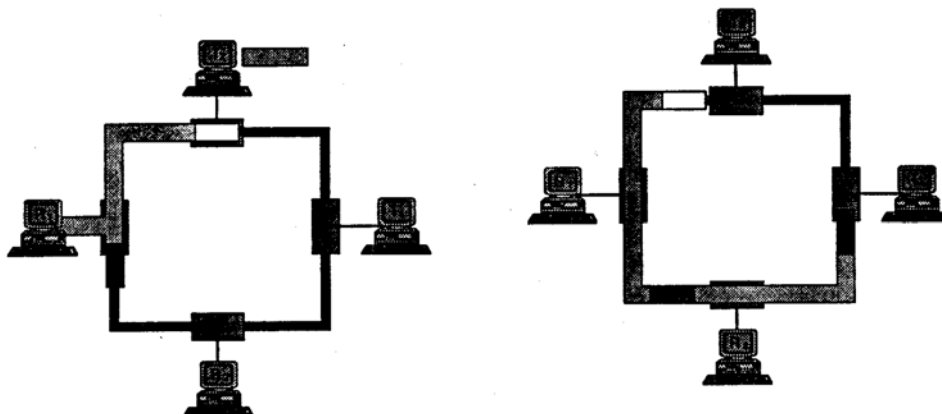


d) estação recebe sua mensagem e) estação libera permissão e de volta retira sua mensagem

Fig. 5.9 - Token Ring single token (a - c iguais a single packet)



c) estação 1 transmite d) estação 1 insere permissão sua mensagem livre no anel



e) estação 2 recebe f) estação 2 transmite sua mensagem permissão livre e insere permissão livre no anel

Fig. 5.10 - Token Ring multiple token (a - b iguais a single packet)

A eficiência de cada método depende da relação (a) entre a latência do anel e o tempo de transmissão de um quadro. Para $a \leq 1$ single token e multiple token são equivalentes mas superiores

a single packet. Quando a latência aumenta ($a > 1$), multiple token é mais eficiente que single token, que, por sua vez, supera o single packet.

Quando a estação origem retira o quadro, pode-se fazer com que a estação destino comande certos bits do quadro, indicando o resultado da transmissão, como por exemplo: quadro recebido, quadro não copiado por falta de espaço, quadro com erro, etc.

Da mesma forma que em outros protocolos de anel, este método precisa de estação(ões) monitora(s) para a correção de erros como quadro circulando indefinidamente ou perda do token.

Este método é similar ao token bus, mas não apresenta overhead na passagem do token para tráfego baixo e nenhum preâmbulo é necessário para sincronismo após a iniciação do anel, fazendo com que sua eficiência de uso do meio seja maior, principalmente com tráfego baixo.

A ISO padronizou este método (ISO 8802-5) seguindo a padronização proposta pelo IEEE 802/5, tendo como modo básico o single token, podendo usar o multiple token quando a taxa de transmissão adotada for de 16 Mbps.

Um método alternativo é o *FAST RING*, onde a permissão circula no anel, mas as estações que possuem quadros a transmitir podem fazê-lo sempre que sentirem o anel livre. Quando uma estação está transmitindo e recebe a permissão livre, continua transmitindo seu quadro e passa a permissão assim que acabar. Se ela recebe outro quadro enquanto transmite, encerra sua transmissão e deixa passar adiante o quadro que recebeu. Caso a estação que tem a permissão estiver transmitindo e receber quadros, remove-os do anel e continua sua transmissão.

Este método foi elaborado como forma de aumentar a eficiência da utilização do meio físico, sendo principalmente adequada para redes de alta velocidade onde o tempo de transmissão é curto, podendo tornar o tempo de espera pela permissão proporcionalmente muito grande.

2.4 - PROTOCOLOS COM RESERVA

Protocolos baseados em reserva foram desenvolvidos para redes de satélites, caso em que existe grande atraso de propagação quando comparado com redes locais ou metropolitanas, o que quase inviabiliza protocolos baseados em contenção e em passagem explícita de permissão.

Muitos dos protocolos desenvolvidos para redes de satélites foram adaptados para redes metropolitanas quando estas tiveram suas velocidades aumentadas e seus enlaces cada vez mais distantes.

2.4.1 - IFFO - Interleaved Frame Flush-Out

O método IFFO (1980) baseia-se em ciclos de tempo formados slot de status, slots reservados e slots alocados por contenção, conforme mostra a figura 5.11.

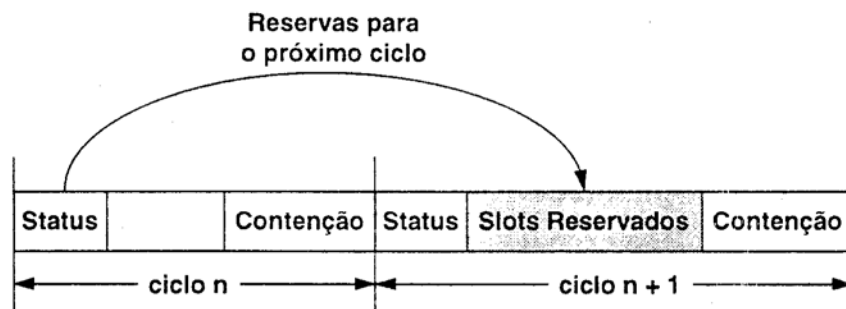


Fig. 5.11 - Acesso a canais de satélite baseado em reservas

O slot de status é sub-dividido em mini-slots, um para cada estação da rede, sendo usado para efetuar as reservas.

Quando uma estação recebe um quadro a ser transmitido, durante a parte de reservas do ciclo $n-1$, o transmissor efetua uma reserva durante o ciclo n para poder transmitir o quadro no ciclo $n+1$. O mesmo ocorre para os quadros que chegam durante o último slot dentre os alocados por contenção.

Quando um slot chega durante o intervalo dos slots alocados por contenção (exceto o último) ele pode ser transmitido imediatamente, em um desses slots. Se houver colisão, os quadros colididos são transmitidos nos slots reservados do próximo ciclo.

Neste método, os quadros são descarregados (flushed-out) no máximo dois ciclos após sua chegada para transmissão.

2.4.2 - CRMA - Cyclic-Reservation Multiple Access

O método CRMA (1990) supõe interconexão das estações por uma topologia de barra dobrada (*folded bus*), como mostra a figura 5.12. Uma barra A é para transmissão e outra (B) para recepção.

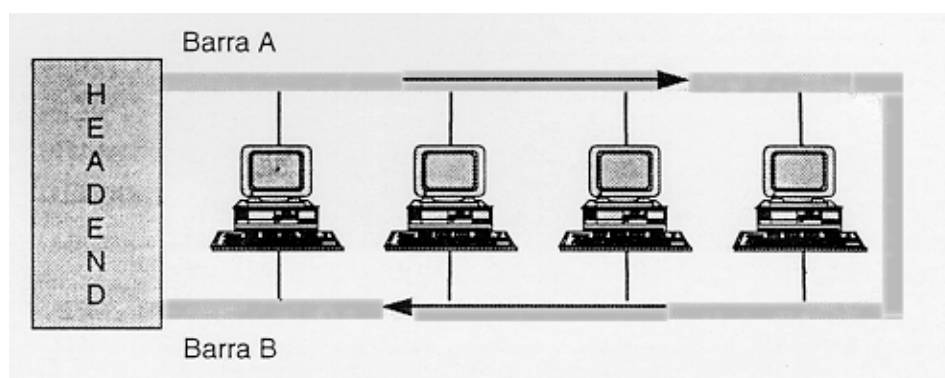


Figura 5.12 - Topologia em barra dobrada

Neste método slots livres são gerados pelos *headend*, com um bit no cabeçalho do slot indicando o estado de livre, ou ocupado. O headend define o tamanho do ciclo, de tamanho variável, já que dependente do volume de reservas feito em cada ciclo. Cada ciclo é iniciado por um comando *start cycle*, sendo terminado ao ser enviado novo *start cycle*. Desta forma, vários ciclos podem co-existir na barra.

O mecanismo do método baseia-se nos comandos *reservation*, periodicamente emitidos pelo headend para coletar requisições. Cada comando tem um campo que identifica o número do ciclo e outro com o totalizador de reservas. Ao retornar ao headend, as reservas são colocadas em uma fila global de reservas.

Uma estação, ao identificar o ciclo para o qual fez reservas, transmite o número de slots reservados.

Caso a fila global ficar vazia, o headend gera slots vazios que podem ser usados mesmo que as estações não tenham reservas.

O método também prevê mecanismos de cancelamento de reservas para limitar o número de reservas que um mesmo nó pode efetuar.

Um aperfeiçoamento é o método *CRMA-II* (1994), destinado a LANs e MANs operando na faixa de Gbps, originalmente projetado para duplo-anel mas que também pode ser usado em barra.

Os dados são transmitidos em slots compatíveis com padrão ATM. Cada slot pode estar livre ou ocupado por uma célula.

A estação destino pode ocupar o slot onde chega a mensagem, otimizando a utilização do meio.

Há dois tipos de acesso a slots: acesso por reserva e acesso imediato. O acesso imediato é feito nos slots que não foram marcados como reservados (slots grátis), provenientes de slots esvaziados pela estação destino.

O mecanismo de reserva garante a justiça no acesso. O processo de reserva é iniciado pelo *escalonador* que envia um comando de reserva. Cada estação coloca no campo de carga (*payload*) o número de slots já transmitidos (T) e o número de solicitações de reserva com base no número de slots aguardando transmissão. Com base nestes valores o escalonador executa um algoritmo de justiça que concede reserva de slots a cada estação.

Por questões de confiabilidade e reconfiguração, a função de escalonador é replicada em todas as estações da rede. No caso de anel único, um só escalonador é ativado por vez, e no caso de anel duplo tem-se um para cada anel. Na topologia em barra, o escalonador é incorporado ao headend.

2.4.3 - DQDB - Distributed Queue Dual Bus

O método *DQDB* considera uma rede composta de duas barras unidirecionais, interconectando ponto a ponto várias estações. As barras suportam comunicação em direções opostas, oferecendo um caminho full-duplex entre quaisquer duas estações, como mostra a figura 5.13.

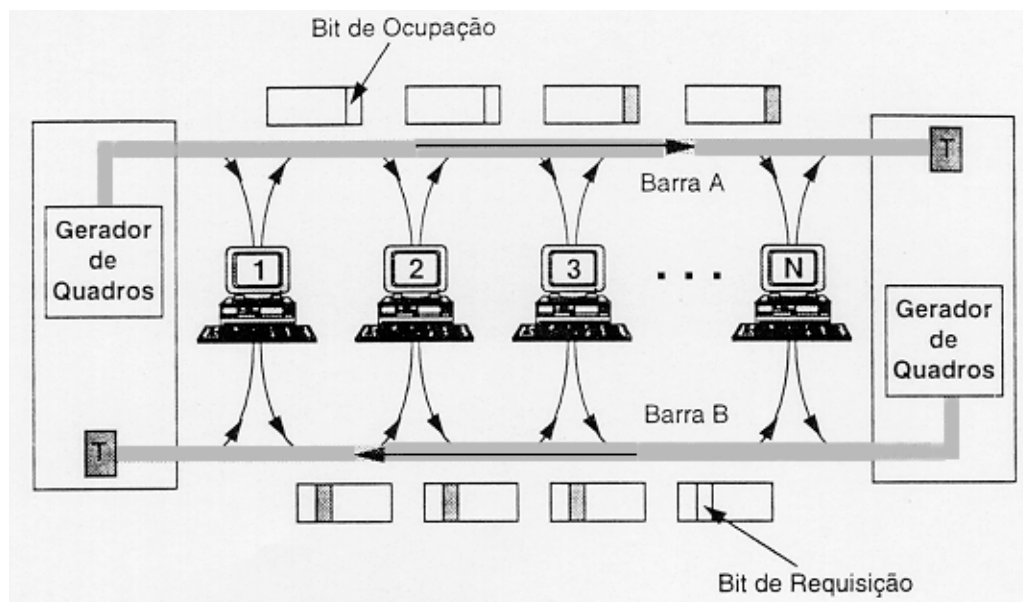


Fig. 5.13 - Sub-rede DQDB

Dois métodos de controle são definidos para as barras DQDB, a Fila Arbitrada (Queued Arbitrated - QA) e Pré-Arbitrada (Pre-Arbitrated - PA).

A barra DQDB é segmentada em slots de tamanhos fixos, com tempo de duração de transmissão de 53 octetos, identificados como slots PA ou QA.

Em cada barra, a primeira estação na direção do fluxo é a responsável pela geração dos slots a uma taxa constante. Cada transmissão deve ser feita dentro de um slot, que contém as células DQDB, unidades de transmissão de dados.

Os slots PA são usados para transferência de dados que necessitam de banda passante garantida e retardo constante. O acesso é determinado pelo gerador de slots, único capaz de escrever nos *Identificador de Canal Virtual* (VCI - Virtual Circuit Identifier). O gerador de slots identifica a unidade de acesso a quem o slot se destina, escrevendo seu endereço no VCI. Estes slots são criados periodicamente e quando a unidade de acesso o encontra, usa-o.

Os slots QA somente são gerados quando não houverem slots PA para serem gerados. O acesso a slots QA é controlado pelo protocolo de fila distribuída, tanto na barra A (barra de transmissão) como na barra B (barra reversa), conforme esquematizado na figura 5.14.

Cada slot tem um Campo de Controle de Acesso - ACF (Access Control Field) que contém um bit de ocupação (busy bit) e um campo de requisição. O bit de ocupação indica se o slot está ocupado, e os bits de requisição que um segmento QA foi colocado na fila de espera para transmissão na outra barra.

Quando uma unidade tem segmento QA a transmitir, emite uma requisição na barra reversa, usando o primeiro slot livre. Esta requisição avisa as unidades anteriores (sentido da barra de transmissão) de sua necessidade.

Pelo protocolo, cada unidade só pode enfileirar um segmento QA a cada nível de prioridade. Também mantém um contador de requisições com os pedidos das unidades à sua frente, contando as requisições na barra reversa. Uma unidade que não tenha nada a transmitir deixa um slot passar vazio na barra de transmissão, então decrementa o contador de requisições (RQ) já que outra unidade já poderá usar o slot vazio.

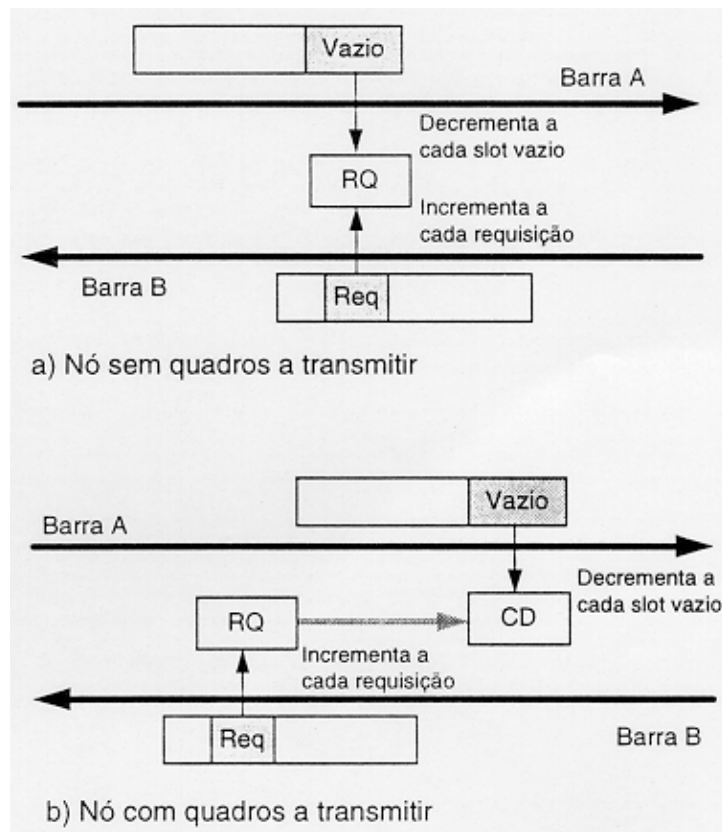


Fig. 5.14 - Algoritmo de controle de acesso QA

Quando uma unidade de acesso quer transmitir um segmento QA, além de enviar sua requisição na barra reversa, transfere o valor de RQ para outro contador (CD), que conterá o número de estações à sua frente que deverão transmitir primeiro, formando a fila distribuída. A cada segmento que passa vazio este contador é decrementado. Quando este contador atingir o valor zero, a unidade pode transmitir.. Cada nova requisição na barra reversa volta a incrementar o contador RQ.

Nada impede que haja transmissão antes do envio da requisição, se houver slots vazios, ou seja, o envio de um segmento QA e da requisição são operações independentes.

NÍVEL DE REDE

O Nível de Rede fornece os meios funcionais e procedimentais para transmissão de dados orientada e não-orientada a conexão entre entidades do nível de transporte.

O transporte de pacotes, da origem ao destino, mesmo que através de várias sub-redes, é oferecida de forma transparente ao nível de transporte, aí incluídas questões de chaveamento e roteamento.

Ao contrário do nível de enlace que oferece transporte de dados DTE (data terminal equipment) - DCE (data communicating equipment), DCE-DCE, DCE-DSE (data switching equipment) e DSE-DSE, o serviço do nível de rede é de DTEorigem/DTEdestino.

1 - ORGANIZAÇÃO INTERNA DO NÍVEL DE REDE OSI

as funções do nível de rede são:

- multiplexação;
- endereçamento;
- mapeamento entre endereços de rede e endereços de enlace;
- roteamento;
- estabelecimento e liberação de conexões de rede;
- transmissão de unidades de dados do serviço de rede;
- segmentação (mapeamento de SDU de rede-PDU de transporte- em várias PDUs de rede - SDU de enlace) e blocagem (mapeamento de várias SDUs de rede em uma PDU de rede);
- detecção e recuperação de erros;
- sequenciação;
- controle de congestionamento.

Quando as funções de rede são realizadas através de combinações de várias sub-redes, as funções de roteamento são especificadas através de sub-camadas, definidas pela ISO como mostra a figura 6.1. Assim separam-se as funções de roteamento das sub-redes individuais das funções de roteamento entre sub-redes.

Fig. 6.1 - Organização interna do nível de rede OSI

A *sub-camada de acesso à rede* depende da tecnologia de sub-rede específica. Suas funções são

- transferência de dados dentro da sub-rede;
- estabelecimento, manutenção e liberação de conexões em sub-redes orientadas a conexão;
- roteamento e retransmissão de informações dentro da sub-rede.

Os protocolos de acesso à sub-rede (*Subnetwork Access Protocol* - SNACP) podem variar de uma rede para outra, por exemplo, o SNACP para Ethernet é diferente e muito mais simples que os protocolos de X.25 para rede pública.

A *sub-camada dependente da sub-rede* tem por função harmonizar sub-redes com serviços diferentes. Por exemplo, a sub-camada independente de rede, no modelo OSI, usa pontos de acesso à rede (SAPs de rede) para endereçamento, enquanto outras sub-redes, como X-25, usam endereçamento por estação, requerendo uma multiplexação dos pontos de acesso a serviços de redes utilizados para endereçar processos na estação. Este caso seria um exemplo típico de função do protocolo de convergência dependente de sub-rede (*Subnetwork Dependent Convergence Protocol* - SNDCP).

Existem, potencialmente, dois SNDCPs para cada tipo de sub-rede, um que adapta o serviço de sub-rede ao serviço de rede orientado a conexão e outro a serviço de rede sem conexão,

Quando o serviço oferecido pela sub-camada de acesso à sub-rede é semelhante ao serviço desejado, o SNDCP não é necessário. Neste caso a função da sub-camada dependente da sub-rede resume-se à manipulação de parâmetros do serviço da sub-rede e o SNDCP é também referenciado como função de convergência dependente da sub-rede (*Subnetwork Convergence Function* -

SNDCF). Caso os sistemas finais de origem e destino fizerem parte da mesma sub-rede, o SNDCP é o responsável pelo funcionamento do serviço de rede.

A *sub-camada independente da sub-rede* é responsável pelo fornecimento de serviço de rede OSI entre dois sistemas finais. O protocolo de convergência independente de sub-rede (*Subnetwork Independent Convergence Protocol* - SNICP) só é usado quando os sistemas finais estão localizados em sub-redes distintas. A principal função da sub-camada independente da sub-rede é o roteamento entre redes.

Resumindo, tem-se:

- SNAcP torna possível a transferência de dados dentro de uma sub-rede;
- SNDCP fornece um serviço de rede OSI na sub-rede (orientado e não-orientado a conexão);
- SNICP realiza adaptações finais para a conexão de sistemas finais localizados em sub-redes distintas.

Note-se que cada sub-camada utiliza-se dos serviços fornecidos pela sub-camada inferior para executar suas funções.

2- SERVIÇOS DO NÍVEL DE REDE

2.1 - Endereçamento

Existem dois tipos básicos de endereçamento:

- *endereçamento hierárquico*: endereço é composto pelos vários níveis hierárquicos de que a rede se compõe (endereço do SAP formado pelo nº da rede, nº da estação e nº da porta associada) ex. IP onde o SAP de rede (único) é formado por endereço da (sub) rede e endereço da estação. Também o X.121 da ITU-T usa este método para endereçamento de redes públicas- um código de 3 dígitos para país, um dígito para rede e 10 dígitos para endereçamento dentro da rede.
- *endereçamento horizontal*: endereços não tem relação com o lugar onde estão as entidades dentro da rede. ex. endereços globalmente administrados adotado pelo IEEE802.

Endereços hierárquicos aparentam certa vantagem , uma vez que contém apresentam informação explícita sobre o local onde se localiza a entidade. O endereço horizontal favorece esquemas de reconfiguração, já que permite certa mobilidade das entidades sem necessidade de uma renumeração.

2.2 - Roteamento

A implementação de roteamento exige uma estrutura de dados (tabela) que informe os possíveis caminhos e seus custos, de forma a permitir uma certa escolha. Essencialmente existem dois métodos de roteamento:

- roteamento por rota fixa - uma vez criada a estrutura de dados, esta não é alterada. Os caminhos alternativos só são considerados em caso de falha. Embora simples, em geral provoca má utilização da rede, só sendo conveniente em tráfego regular e bem equilibrado.

- roteamento adaptativo - a rota é escolhida de acordo com a carga da rede. Nas tabelas de rotas mantém-se informação sobre o tráfego, devendo ser periodicamente atualizadas. A atualização pode ser feita de diferentes modos:
- isolado - atualização é realizada baseada nas filas para os diversos caminhos e outras informações locais. Algoritmos podem buscar tanto o menor caminho quanto o menos carregado. Menor caminho tanto pode ser o caminho com o menor número de nós intermediários (saltos) como a menor distância geográfica ou ainda o caminho com menor retardo (mais rápido).
- distribuído - cada nó envia, periodicamente, aos outros, informações locais sobre carga na rede. Quando um nó quer descobrir o caminho mais curto até um nó destino, envia um pacote por difusão na rede, com um campo com o endereço destino e outro o caminho, inicialmente apenas com o endereço de destino. Cada nó intermediário acrescenta seu endereço (caso ainda não esteja no endereço, caso esteja simplesmente descarta o pacote) e difunde o pacote para todos seus enlaces na rede. Este procedimento segue até que o pacote chegue ao nó destino, que envia o pacote de volta com o endereço mais curto obtido dentre todos os pacotes que recebeu.
- centralizado - um ponto central (Centro de Controle de Roteamento) da rede recebe dos nós as informações locais sobre carga. O ponto central atualiza as tabelas, que são enviadas aos nós e gateways, que realizam a ligação entre redes. Método conveniente quando as mudanças nas tabelas se referem a mudanças na topologia da rede, caso de pouca frequência. O método apresenta certos problemas como o volume de processamento para redes grandes e tráfego muito variável, alto tráfego entre os nós e o ponto central, confiabilidade reduzida devido à centralização e atrasos no recebimento das atualizações pelos diferentes nós devido a retardos variados.
- hierárquico - os nós são divididos em regiões, sendo que cada nó mantém as informações de rotas das regiões a que pertence
- transmissão por difusão - corresponde ao envio do pacote a todos os usuários da inter-rede. Vários métodos podem ser usados. Cada nó difunde o pacote a todos seus enlaces, ou a todos menos aquele por onde chegou, ou cada pacote deve conter uma lista de destinos desejados, etc. Este tipo gera um tráfego enorme na rede.

2.3 - Tipos de serviço

Os tipos de serviço oferecidos pela camada de rede são

- *serviços com conexão*
- *serviços sem conexão*

Os serviços com conexão, chamados de serviços de circuito virtual, criam um caminho entre a origem e o destino que permanece até o fim da comunicação, quando é desfeita.

Nestes serviços tem-se

- mecanismos implícitos de controle de erro e fluxo, garantindo a sequência na entrega dos pacotes.
- possibilidade de pré-alocação de recursos (armazenagem nos gateways e nós), garantindo melhor controle de congestionamento a nível de sub-rede.

No entanto estes serviços são altamente complexos, exigindo desempenho de processamento e espaço de armazenagem que muitas vezes não estão disponíveis em nós e gateways.

Os serviços sem conexão, também chamados de serviços de datagrama, consideram cada mensagem de forma individual, buscando entregá-la ao destino através do caminho mais conveniente, conforme definido pelos algoritmos de roteamento.

Nestes serviços a inter-rede não garante a entrega dos pacotes, nem garante a chegada do pacote ao destino final (datagrama não confiável). Sua principal vantagem é a simplicidade.

Independentemente dos serviços oferecidos pela inter-rede, o serviço oferecido pela rede (sub-rede) também pode ser com conexão ou sem. Note-se que os serviços oferecidos pela rede são independentes daqueles oferecidos pela inter-rede. Pode-se implementar qualquer combinação entre eles.

As primitivas oferecidas pelos serviços de rede no padrão OSI são muito parecidas com as definidas pelo IEEE 802.2.

Nos serviços com conexão tem-se um grupo de primitivas responsáveis pela conexão, outro pela desconexão, outro pela troca de dados e outro pelo relato de falhas irreversíveis na rede. Nos serviços sem conexão tem-se um grupo de primitivas para troca de dados e outro para troca de informações sobre desempenho médio da rede entre usuários.

2.4 - Controle de congestionamento

Existe *congestionamento* em uma rede quando houver pacotes em excesso em uma sub-rede ou parte da sub-rede, degradando o desempenho da rede.

Controle de congestionamento garante que a rede seja capaz de transmitir o tráfego oferecido. Notar que isto é diferente de controle de fluxo que se relaciona com o tráfego entre um transmissor e um determinado receptor. Algumas redes tentam realizar o controle de congestionamento apenas com o controle de fluxo.

O controle de congestionamento pode ser feito por:

- descarte de pacotes - um pacote que chegar a um nó da rede onde não houver de espaço de armazenagem, este é simplesmente descartado. Mecanismos para otimização do descarte e avaliação se o pacote não é uma confirmação que libera buffer podem ser aplicadas.
- pré-alocação de buffers - em serviços com conexão pode-se pré-alocar buffers de forma permanente no momento do estabelecimento da conexão, não havendo risco de congestionamento. No entanto há uso ineficiente dos recursos, uma vez que o buffer pré-alocado não pode ser compartilhado mesmo que ocioso.
- controle isorrítmico - permissões de transmissão circulam pela sub-rede. Sempre que um nó deseja introduzir um pacote na sub-rede ele deve primeiro capturar uma permissão e destruí-la. Esta é restaurada pelo nó receptor do pacote. Este controle oferece os mesmos inconvenientes do acesso ao meio por permissão, agravando ainda o próprio tráfego da rede.
- controle de tráfego no enlace - cada nó monitora a utilização percentual de cada uma de suas linhas de saída. Caso a utilização ultrapasse certo limiar a linha entre em estado de alerta. Um nó que receba um pacote a ser despachado por linha de alerta deve enviar ao nó origem um pacote de alerta com seu identificador e o nó destino original. Neste caso o nó origem deve reduzir o tráfego para o destino ou alterar a rota.

3 - O PROTOCOLO X-25

Este protocolo foi padronizado pela ITU-T como o padrão X-25 PLP (Packet Layer Protocol) como forma de implementação dos níveis físico, de enlace e de rede para ligação DTE-DCE. Este padrão não especifica os protocolos internos à sub-rede, ou seja, a comunicação DCE-DCE e DCE-DSE. Uma rede X-25 é representada pela figura 6.2.

Fig. 6.2 - O protocolo X-25 e o modelo OSI

No X-25 a referência para o nível físico é explicitada pelo padrão X-21 e X-21 bis, que definem as interfaces digital e analógica. No nível de enlace os protocolos utilizados são o *Link Access Procedure* (LAP) e *Link Access Procedure Balanced* (LAPB), semelhantes ao IEEE 802.2.

No nível de rede o X-25 utiliza o serviço com conexão. Duas formas de conexão são possíveis: circuitos virtuais permanentes (conexões pré-estabelecidas) e chamadas virtuais (conexões alocadas em tempo de comunicação).

O sistema de endereçamento do X-25 é definido pela recomendação X.121 da ITU-T. O controle de fluxo e de erro é realizado através de janela deslizante. A confirmação de recebimento de pacotes pode ser de dois tipos, selecionadas por um bit (bit D) do pacote de dados. Se D=0 indica que o DCE recebeu o pacote, se D=1 a confirmação indica que o DTE remoto recebeu o pacote, ou seja, houve comunicação fim a fim.

O padrão original não incluía a facilidade de datagrama, que em 1984 foi anexada como recurso *fast select*. Com este recurso o pacote de estabelecimento de conexão (call request) é expandido para incluir 128 bytes de usuário. O DTE destino rejeita a conexão mas recebe a mensagem agregada. O pacote de rejeição (clear request) também é expandido de 128 bytes para incluir dados de resposta.

4 - O PROTOCOLO IP (Internet Protocol)

Projetado para interconexão de redes que utilizam tecnologia de comutação de pacotes.

Ambiente consiste de *hosts* ligados a redes. Estas redes são interligadas através de *gateways*, também chamados *roteadores* (*routers*), conforme mostra a figura 6.3.

As redes que compõem a *inter-rede* tanto podem ser LANs como WANs.

Fig. 6.3 - Conceito de inter-rede

O protocolo IP não tem conexões, sendo responsável por transferir *datagramas* da estação origem ao destino, cada um identificado por seu endereço IP. Devido à existência de redes onde o tamanho máximo de pacotes é pequeno, este protocolo também fornece serviços de fragmentação e remontagem de datagramas longos.

Por apresentar um serviço sem conexão, o IP trata cada datagrama como unidade independente, não fazendo reconhecimento fim-a-fim nem entre nós intermediários. Também não realiza qualquer

controle de erros, exceto um checksum do cabeçalho onde se encontra as informações de endereçamento.

As principais características deste protocolo são:

- serviço de datagrama não confiável;
- endereçamento hierárquico;
- facilidade de fragmentação e remontagem de pacotes;
- identificação da importância do datagrama e do nível de confiabilidade exigido;
- identificação da urgência de entrega e ocorrência futura, ou não, de pacotes na mesma direção (pré-alocação e controle de congestionamento);
- campo especial indicando qual o protocolo de transporte a ser utilizado no nível superior;
- roteamento adaptativo distribuído nos gateways;
- descarte e controle de tempo de vida dos pacotes inter-redes no gateway.

4.1 - Endereços IP

Endereços IP são números de 32 bits, normalmente separados em quatro octetos (em decimal). Uma primeira parte identifica uma rede específica na inter-rede e outra identifica o host na rede.

Um endereço IP não identifica especificamente uma máquina individual, mas uma conexão à inter-rede. Por isso, um gateway conectando n redes tem um endereço para cada conexão.

O endereço IP tanto pode identificar a rede como um todo quanto um certo host. Por convenção, um endereço de rede tem o campo identificador do host com todos os bits iguais a 0. Pode-se referir a todos os hosts de uma rede, colocando-se todos os bits identificadores do host como 1. Um endereço com todos os 32 bits iguais a 1 indica uma difusão de mensagem para a rede do host origem. O endereço 127.0.0.0 é reservado para teste e comunicação entre processos da mesma máquina.

O IP usa três classes diferentes de endereços, em função do tamanho da rede, já que estas podem variar desde redes locais até grandes redes públicas.

A *classe A*, usada para redes de grande porte, os endereços de rede variam de 1 a 126 (primeiro octeto: primeiro bit 0 e os 7 restantes para identificação da rede. Os 24 bits restantes definem o endereço local). Neste caso, cada rede pode endereçar por volta de 16 milhões de hosts.

A *Classe B* usa dois octetos para identificar a rede e dois para endereço de hosts. Estes endereços variam de 128.1 até 191.255 (127 no primeiro octeto e 255 do segundo são usados para funções especiais). Neste caso, cada rede pode endereçar por volta de 65 mil hosts.

A *Classe C* usa três octetos para identificação da rede e um para identificação de hosts. Estes endereços ficam na faixa de 192.1.1 até 223.254.254, sendo que cada rede pode endereçar até 254 hosts. Os endereços acima de 223 no primeiro octeto ficaram reservados para aplicações futuras.

4.2 - Roteamento

O roteamento inter-redes é a principal função do protocolo IP. O protocolo assume que um host sabe enviar datagramas a qualquer host de sua rede. Para o caso de outras redes, o roteamento é

feito pelos gateways ou roteadores, com o módulo IP do host apenas enviando o datagrama ao gateway conectado à sua rede, considerando um roteamento hierárquico.

O roteamento tanto pode ser feito por software em uma máquina normal, ou por equipamento específico quando o tráfego é alto.

Quando o datagrama deve ser enviado a host da mesma rede da origem, o módulo IP simplesmente o entrega à interface de rede que mapeia o endereço IP em endereço físico do host. Após isto, encapsula o datagrama IP em um quadro de rede e o transmite ao destino.

Para o caso em que o destino não se encontre na mesma rede realiza-se o roteamento. Cada computador possui uma *Tabela de Roteamento IP* que relacionam os pares endereço de rede / endereço de roteador. Nesta tabela o módulo IP identifica o roteador a ser usado para alcançar a rede onde está o destinatário. Este roteador pode não estar ligado à rede do destinatário, mas deve fazer parte do caminho para alcançá-la. Por exemplo, na figura 6.4, caso o host 200.1.4.20 queira enviar datagramas à rede 200.1.2, recupera de sua tabela o endereço do gateway 200.1.4.30.

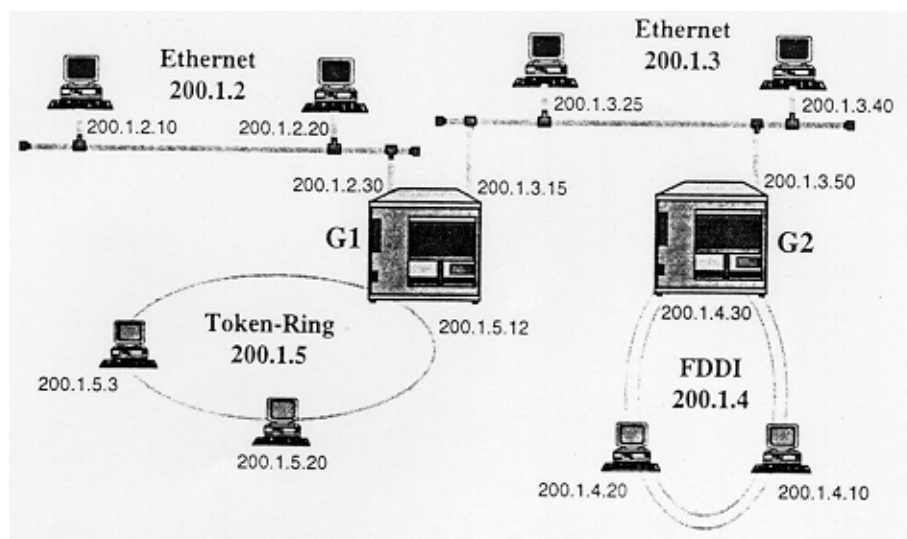


Fig. 6.4 - Exemplo de inter-rede

O uso de tabelas de roteamento pode se tornar improdutivo quando a inter-rede interligar muitas redes individuais (caso da Internet). Neste caso, outras estratégias são adotadas. Por exemplo, quando um único roteador apresenta conexão à inter-rede, usa-se um *roteador default*. Não há necessidade de haver uma entrada para cada uma das redes distintas da inter-rede, já que todas as rotas passam pelo mesmo roteador. O roteador default também pode ser usado caso haja mais de um roteador na rede. Quando não for encontrada entrada específica para o datagrama, este é enviado ao roteador default. Um protocolo específico (ICMP - Internet Control Message Protocol), no roteador default, analisa o endereço e, caso não seja a melhor entrada, retorna ao módulo IP uma mensagem *redirect*, que carrega o endereço do roteador correto. O módulo IP, então, acrescenta em sua tabela a nova entrada e envia o datagrama.

LIGAÇÃO INTER-REDES

Entre estações origem e destino de uma comunicação podem existir redes diferentes. A inter-rede deve resolver a questão da independência dessas redes.

Estações especiais são usadas para conectar física e logicamente estas redes, denominadas de *gateways*. Os gateways encarregam-se da conversão de protocolos necessárias ao se levar uma mensagem de uma rede para outra, conforme representado na figura 7.1.

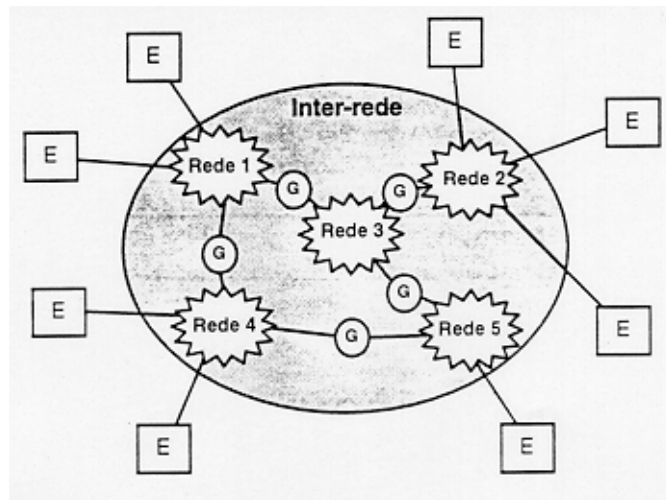


Fig. 7.1 - A inter-rede

Gateways são classificados de acordo com o mais alto protocolo em que realizam conversão. Tem-se então para conversão:

- ao nível físico - *repeater* (repetidor)
- ao nível de enlace - *bridge* (ponte)
- ao nível de rede - *router* (roteador)
- ao nível de transporte - gateway de transporte
- ao nível de sessão - gateway de sessão
- ao nível de apresentação - gateway de apresentação
- ao nível de aplicação - gateway de aplicação

1. ROTEADORES

Repetidores interligam duas ou mais redes idênticas, como mostrado na figura 7.2. Repetem os pacotes de uma rede para a outra, sem realizar qualquer tratamento sobre eles, atuando apenas ao nível físico, como mostra a figura 7.3.

Fig. 7.2 - Redes ligadas por repetidor

Fig. 7.3 - Nível de protocolo no RM-OSI onde atuam repetidores

Além de simplesmente repetir os sinais de uma rede para outra o repetidor tem funções especiais:

- em redes em anel onde a estação destino se responsabiliza pela retirada dos quadros, o repetidor os retira da rede onde atua como retransmissor, quando passa o quadro para a outra rede.
- em redes com protocolos baseados em contenção, o repetidor tem a responsabilidade de realizar a detecção de colisão e realizar a retransmissão.

Note-se que situações especiais podem ocorrer com repetidores. Não se pode ter um caminho fechado entre dois repetidores da rede, o que implicaria em duplicações infinitas dos quadros. Também, o repetidor pode gerar tráfego inútil na rede ao simplesmente repetir os quadros recebidos, se estes quadros não se destinarem à rede que interliga. Neste caso os

2. PONTES

Pontes repetem apenas os pacotes destinados às redes que interliga, ou a redes intermediárias na rota dos pacotes, como mostrado na figura 7.4 . Sua atuação se dá a nível de protocolos de enlace, como mostra a figura 7.5.

Fig. 7.4 - Redes interligadas por pontes

Fig. 7.5 - Nível de protocolo no RM-OSI onde atuam pontes

As pontes apresentam os mesmos problemas dos repetidores quanto à retirada de quadros em redes em anéis e a seu uso em redes baseadas em contenção. No entanto, a questão de duplicação de quadros pode ser tratada, uma vez que a duplicação de quadros é finita e igual ao número de caminhos fechados entre a origem e o destino.

As principais funções realizadas pelas pontes são:

- filtro de entrada - recebe apenas os pacotes endereçados às redes por ela interligadas, direta ou indiretamente.
- armazenagem - de quadros no transporte de uma rede para outra.
- transmissão - atuando como um repetidor comum.

As grandes vantagens das redes são:

- isolar falhas de uma rede, não permitindo influência da falhas nas outras.
- permitir aumento de desempenho de redes, ao subdividi-las em sub-redes.

A interligação de redes IEEE802 através de pontes é feita na camada MAC, podendo ser dos tipos:

- *ponte transparente (Transparent Bridge)*; e
- *ponte com roteamento na origem (Source Routing Bridge)*.

2.1 Pontes Transparentes

As pontes transparentes operam abaixo da interface definida pelo serviço MAC. Recebem este nome por não interferirem nas LANs às quais forem conectadas.

Sua ligação às LANs é feita através de *portas*. Uma ponte com X portas possui X entidades MAC, possuindo assim X endereços MAC, um para cada porta.

As pontes transparentes operam em modo promíscuo, escutando todos os quadros das redes onde estão conectadas. Para cada quadro consultam uma *tabela de rotas* formada por endereço de destino / porta de saída. Verificam, inicialmente, se o endereço destino é diferente da porta de entrada. Caso seja, e esteja na tabela, o quadro é transmitido na porta correta. Caso o endereço não se encontre na tabela, é transmitido por todas as portas, exceto aquela por onde chegou.

Questões mais específicas e detalhadas da interligação de redes IEEE802 através de pontes são especificadas no padrão IEEE802.1D.

2.2 Pontes com Roteamento na Origem

Neste esquema, desenvolvido pelo comitê IEEE802.5, a estação origem escolhe o caminho que o quadro deve seguir e inclui esta informação de roteamento no cabeçalho do quadro. A rota é definida como uma sequência de pares identificador da rede / identificador da ponte, identificadores únicos para cada rede e cada ponte.

Uma ponte, ao analisar um quadro, verifica na informação de roteamento se o identificador da LAN através da qual chegou é seguido de seu identificador. Caso positivo, retransmite o quadro para a rede cujo endereço esteja após o seu na informação de roteamento. Caso contrário, descarta o quadro. Este esquema pressupõe que cada estação conheça o caminho para todas as outras.

3. ROTEADORES E GATEWAYS

Gateways de forma geral podem interligar várias redes distintas, como mostra a figura 7.6, e são classificados em dois tipos: *gateways conversores de meio* (*media conversion gateway*) e *gateways tradutores de protocolos* (*protocol-translation gateway*).

Gateways conversores de meio são mais simples, sendo bastante usados em inter-redes com serviço de datagrama. Sua função é:

- receber um pacote do nível inferior;
- tratar o cabeçalho inter-redes do pacote para descobrir o roteamento necessário;
- construir novo pacote com novo cabeçalho inter-redes, se necessário;
- enviar o novo pacote ao próximo destino, segundo o protocolo da rede local em que se encontra.

Figura 7.6 - Interligação de redes usando gateways

Um gateway conversor de meio, na camada de rede é chamado de *roteador* (*router*). A figura 7.7 representa o nível RM-OSI onde os roteadores atuam.

Fig. 7.7 - Nível de protocolo no RM-OSI onde atuam roteadores

Os gateways tradutores de protocolo atuam traduzindo mensagens de uma rede em mensagens da outra rede, com a mesma semântica de protocolo, para os diferentes níveis (ex. traduz um open de

uma rede em call request de outra). Com frequência não se pode mapear funções um - a - um, requerendo que a conversão seja feita em termos do serviço oferecido. Isto torna os gateways complexos e caros. Este tipo de gateway pode atuar em qualquer nível acima do de enlace RM-OSI. A figura 7.8 representa um gateway de aplicação.

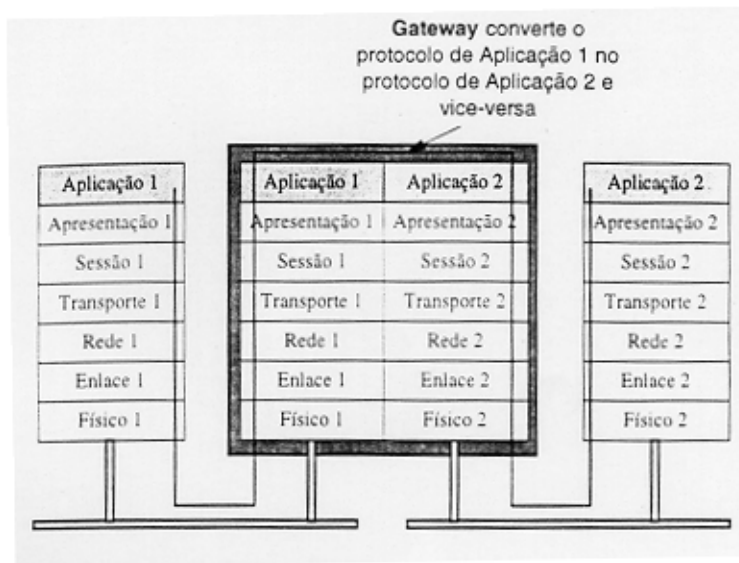


Fig. 7.8 - Gateway de aplicação no RM-OSI

Redes pertencentes a organizações diferentes podem ser interligadas através de gateways separados fisicamente, como mostra a figura 7.9. Esta separação facilita a implementação e a operação do conjunto.

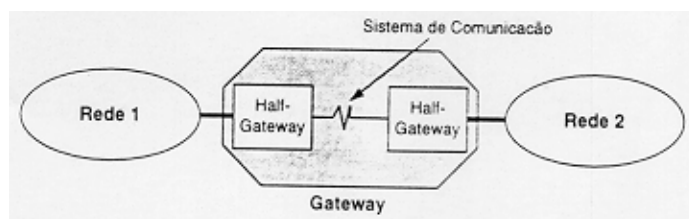


Fig. 7.9 - Half-gateway

Cada interface é denominada de *half-gateway*. As metades se comunicam através de um sistema de comunicação, cabendo a cada half-gateway, a realização do protocolo de comunicação entre eles. Cabe, também, a cada half-gateway, a realização do protocolo de comunicação entre eles.

Nível de Transporte

A camada de transporte é responsável pela movimentação de dados, de maneira eficiente e confiável, entre processos (usuários) em execução nos equipamentos conectados a uma rede de computadores, independentemente da rede (ou redes) física.

O objetivo principal da camada de transporte é tornar transparentes, para seus usuários, possíveis variações da confiabilidade do serviço fornecido pela camada de rede. O serviço de rede é responsável pela transferência de dados da origem para o destino, ao passo que o nível de transporte é responsável por sua segurança.

Serviços oferecidos ao nível de sessão:

- *Serviços não-orientados à conexão*: a camada de transporte mapeia o pedido de transmissão de uma SDU (*Unidades de Dados de Serviço*) de transporte em um pedido de transmissão feito ao serviço não-orientado à conexão fornecido pela camada de rede.
- *Serviços orientados à conexão*: fornece meios para estabelecer, manter e liberar conexões de transporte entre um par de usuários, possivelmente entidades da camada de sessão, através de pontos de acesso ao serviço de transporte (T-SAPs).

Uma vez estabelecida a conexão, começa a fase de transferência de dados, onde são executadas algumas das seguintes funções:

- *Fornecer comunicação fim a fim* (origem-destino) *confiável*, uma vez que o nível de rede não garante a entrega dos pacotes, nem sua ordem.
- *Otimizar as conexões de rede*. Faz: *multiplexação* (várias conexões de transporte partilhando a mesma conexão de rede) quando uma conexão de transporte não gera tráfego suficiente para ocupar a capacidade de conexão da rede; *splitting* (uma conexão de transporte ligada a várias conexões de rede) para aumentar a vazão de uma conexão de transporte.
- *Controle de fluxo*, evitando que o transmissor envie mensagens em taxa maior que a capacidade do receptor.
- *Controle de sequência fim a fim*.
- *Detecção e recuperação de erros fim a fim* (entidade de transporte para entidade de transporte).
- *Segmentação e blocagem das mensagens*.
- *Transporte de dados expresso* (dados de sinalização e interrupções).

Endereçamento

Os pontos de acesso a serviços de transporte (T-SAPs) são análogos aos SAPs de enlace e de rede. A forma de associação de um processo a um T-SAP está fora do modelo OSI e depende exclusivamente do sistema operacional local.

Uma questão importante a ser resolvida dentro da camada de transporte é a associação de T-SAPs a SAPs de rede.

Na forma mais simples, uma estrutura de endereços hierárquicos identifica o SAP de rede dentro de um campo do endereço do T-SAP.

Se o endereçamento não é hierárquico, é necessário a realização de algum mapeamento para localizar o SAP de rede. Deve haver um servidor de nomes que use os T-SAPs como entrada e retorne SAPs de rede como saída.

Estabelecimento de Conexões

O que complica o estabelecimento de uma conexão de transporte é a perda, armazenagem ou duplicação de pacotes. Devido a um congestionamento na rede, um pedido de transferência, por exemplo, não receberia a resposta de recebimento deste pedido a tempo e, então, enviaria um outro pedido. Caso não haja nenhum controle, dois pedidos idênticos de transferência chegariam a máquina destino.

Uma solução para este problema pode ser conseguida estabelecendo um tempo de vida máximo na rede para um pacote. Com isso, pode-se calcular um tempo T a partir do qual podemos ter certeza de que o pacote e suas confirmações não causarão problemas.

Como melhoria, pode-se acrescentar uma numeração a cada T-PDUs, de forma que esta numeração nunca se repita dentro do período T definido no parágrafo anterior (método de *Tomlinsom*).

Encerramento de Conexões

Apesar de ser muito mais fácil encerrar do que estabelecer uma comunicação, há o problema da perda de dados depois que um dos lados encerra a conexão. Uma forma de resolver este problema é terminar uma conexão apenas após decorrido um certo tempo sem que chegue nenhuma T-PDU. Além disso, uma combinação de temporizadores com confirmação de conexão pode ser tentada, a fim de minimizar o problema.

Qualidade de Serviço

A camada de transporte tem como uma das principais funções a ampliação da qualidade de serviço (*Quality of Service - QoS*) fornecida pela camada de rede.

A qualidade de serviço pode ser caracterizada por uma série de parâmetros específicos (parâmetros QoS). A seguir, têm-se alguns destes parâmetros:

- retardo no estabelecimento da conexão.
- retardo no encerramento da conexão.
- taxa de bits transferidos por segundo (vazão).
- retardo de transferência médio.
- retardo de transferência máximo.
- taxa de erro.
- prioridade.

É função da camada de transporte examinar os parâmetros requeridos e determinar se pode ou não fornecer o serviço.

A definição da camada de transporte RM-OSI não determina a codificação ou os valores para os parâmetros QoS, estes são fornecidos pelo usuário e pela própria camada de transporte.

Controle de Fluxo e de Erros

Devido ao fato de o número de conexões que a camada de transporte tem de gerenciar ser, em geral, muito grande, o protocolo de transporte deve permitir que uma estação transmissora solicite (negocie) espaço de armazenamento na estação receptora, que pode ser feita para cada conexão ou para todas as conexões correntes entre as duas estações.

Protocolos de Transporte da ISO

Como camada de transporte está intimamente ligada à camada de rede, as funções que devem ser executadas pelo protocolo de transporte dependem do serviço de rede sobre o qual ele opera.

Com base nos serviços de rede existentes, a ISO definiu 5 tipos de protocolos de transporte: TP0, TP1, TP2, TP3 e TP4. Os quatro primeiros operam com base em serviços de rede orientados à conexão, já o TP4 opera com serviços de rede orientados ou não-orientados à conexão.

Para diferenciar estes 5 protocolos, é necessário classificar os serviços de redes existentes em 3 tipos:

- *Tipo A*: redes que consideram erro qualquer perda de dados. A quantidade de pacotes duplicados, perdidos, fora de sequência ou danificados é desprezível. O serviço de rede é orientado à conexão.
- *Tipo B*: redes que consideram erro qualquer perda de dados. A ocorrência de erros é mais freqüente, sendo menos confiáveis que as do tipo A. Enquadram-se nesse tipo as redes públicas que fornecem serviço de circuito virtual. O serviço de rede é orientado à conexão.
- *Tipo C*: redes que não detectam erros quando pacotes são perdidos, duplicados, entregues fora de ordem ou danificados. Essas redes fornecem o serviço de datagrama não-confiável (serviço sem conexão e sem confirmação).

Com base nestes tipos de rede, pode-se agora estender as características dos 5 protocolos ISO:

- *TP0*: é o mais simples dos cinco. Fornece mecanismos para estabelecer e encerrar conexões de transporte, utilizando uma conexão de rede para cada conexão de transporte aberta. As funções executadas na fase de transferência de dados são: transferência de PDUs (*Unidade de Dados de Protocolo*), segmentação e remontagem de SDUs (*Unidades de Dados de Serviço*) quando seu tamanho excede o comprimento máximo imposto pela camada de rede. Este protocolo é utilizado em rede *tipo A*.
- *TP1*: semelhante ao TP0, porém foi projetado para recuperar erros causados pela re-inicialização (reset) de conexões de rede. Utilizado em redes onde ocorrem re-inicializações com certa freqüência. No TP1, as PDUs são numeradas e armazenadas enquanto a entidade de transporte de origem não receber o reconhecimento enviado pela entidade de transporte de destino. Quando a uma re-inicialização ocorre, o fornecedor do serviço de transporte TP1 sabe ressincronizar e continuar do ponto em que tinha parado. Nenhum controle de erro ou fluxo é realizado, além do fornecido pela camada de rede. O TP1 é utilizado em rede tipo B, podendo ser usado também pela rede tipo A.
- *TP2*: este protocolo concentra o tráfego das conexões de transporte em uma única conexão de rede. Executa também as funções de controle de fluxo e segmentação/remontagem de SDUs, não realizando qualquer controle de erro ou recuperação em caso de re-inicialização de uma conexão. Esse protocolo é apropriado para redes *tipo A*.
- *TP3*: combina as funções dos protocolos TP1 e TP2, implementando as funções: multiplexação/demultiplexação de conexões de redes, controle de fluxo, segmentação/remontagem e restabelecimento de conexões no caso de re-inicialização de conexões de redes. Pode ser usado em redes *tipo A* ou *B*.
- *TP4*: projetado para redes *tipo C* (embora possa ser usado em redes *tipo A* ou *B*), que fornecem serviço sem conexão e sem confirmação. Como neste tipo de rede muitos dos problemas que ocorrem durante a transmissão dos dados não são sequer detectados, cabe ao TP4 detectar e recuperar pacotes perdidos, duplicados, danificados e entregues fora de sequência. Este protocolo fornece os serviços de: multiplexação de conexões de rede e permite também que a carga de uma conexão de transporte seja distribuída (*splitting*) em várias conexões de rede; segmentação e blocagem; controle de fluxo e serviço expresso.

Protocolos de Transporte da Arquitetura Internet TCP/IP

A arquitetura Internet possui dois tipos de protocolo em sua camada de transporte:

TCP (Transmission Control Protocol):

- opera no modo orientado à conexão fornecendo um serviço confiável de transferência de dados fim a fim;
- foi projetado para funcionar com base em um serviço de rede sem conexão e sem confirmação;
- a figura a seguir mostra o posicionamento do TCP na arquitetura Internet TCP/IP:

Fig. 8.1 - Arquitetura Internet

- *interface entre os processos de aplicação e o TCP*: conjunto de chamadas semelhantes às que os sistemas operacionais fornecem aos processos de aplicação para manipulação de arquivos;
- *interface entre o TCP e a camada inferior*: mecanismo de troca de informações assíncrona.
- permite que vários processos, em um único host, possam simultaneamente usar seus serviços, utilizando o conceito de *porta*. Cada um dos usuários (processos de aplicações) que o TCP está atendendo em um dado momento é identificado por uma porta diferente. Para evitar conflito entre as portas de vários TCPs, o identificador de porta é concatenado ao endereço IP, definindo um *socket*. Um *socket*, que é equivalente a um T-SAP, identifica univocamente um usuário TCP em toda a inter-rede.

UDP (User Datagram Protocol):

opera no modo sem conexão e fornece um serviço datagrama não-confiável, sendo uma simples extensão do protocolo IP (*Internet Protocol*). O UDP recebe os pedidos de transmissão de mensagens entregues pelos processos de aplicação da estação de origem e os encaminha ao IP, que é o responsável pela transmissão. Na estação destino, o processo inverso ocorre: o IP entrega as mensagens (datagramas) recebidas ao UDP que as entrega aos processos de aplicação. A principal função do UDP é multiplexar na origem e demultiplexar no destino, o acesso ao nível inter-rede.

Interfaces para Programas de Aplicações

Estas interfaces (*API - Application Program Interface*) são interfaces disponíveis para programadores. A seguir estão listadas algumas das interfaces construídas tendo como base a camada de transporte, e tem como finalidade básica descobrir qual o T-SAP do processo que deseja comunicar-se:

- *Servidor de Processos*: ao ser solicitado, por um cliente (que estabeleceu uma conexão com este servidor), por algum serviço, o servidor de processos escolhe um T-SAP livre na estação e gera o processo do serviço requisitado, dizendo a este para escutar o T-SAP escolhido. O servidor de

processos envia, então, uma mensagem ao cliente remoto contendo o T-SAP escolhido e encerra a conexão.

- *Servidor de Nomes*: é utilizado em casos em que um serviço já esteja em execução. Para encontrar o T-SAP de um determinado serviço, o usuário cliente estabelece a conexão com este servidor. O cliente, então, envia uma mensagem solicitando o endereço do serviço, recebendo como resposta o T-SAP requisitado. Ao receber a resposta, o cliente encerra a conexão com este servidor e abre uma conexão com o T-SAP do serviço desejado. (Neste modelo, é exigido que cada serviço na rede se cadastre no servidor de nomes, indicando seu T-SAP.)

- *Berkley Sockets*: são APIs, geralmente para a linguagem C, que funcionam como uma chamada do sistema operacional (UNIX). A seguir, tem-se algumas destas chamadas para o sistema 4.3BSD para VAX, para comunicação em ambiente TCP/IP:

- *sockfd* = association(família, tipo);

uma associação (*association*) cria um *socket* e é definida como uma quintupla {protocolo, endereço local, processo local, endereço remoto, processo remoto}. O argumento *família* define o uso dos protocolo TCP/IP. *Tipo* indica se é um *socket* para TCP, UDP ou IP. A chamada retorna um inteiro, o descritor do socket: *sockfd*.

- bind(*sockfd*, *endereço);

a chamada *bind* (juntar), que tem como argumentos o *sockfd* e um ponteiro para endereço, que no ambiente TCP/IP se constitui de: <endereço de porta, endereço de rede, endereço de estação>. Quando este parâmetro é especificado como nulo (NULL), o sistema escolhe um T-SAP ocioso. Esta chamada completa o *processo local* e o *endereço local* de uma associação.

- end = getsockname(*sockfd*);

retorna o *endereço local* e o *processo local* associado ao *sockfd*.

- listen(*sockfd*, num_pedidos);

aloca espaço para o armazenamento de pedidos de conexão.

- connect(*sockfd*, *processo_remoto, *end_remoto);

estabelece a conexão entre dois *sockets*, completando a quintupla da associação. A chamada só retorna depois que a comunicação esteja estabelecida ou devido a algum erro que seja reportado no processo.

- end = getpeername(*sockfd*);

retorna o *endereço remoto* e o *processo remoto* associado ao *sockfd*.

- send(*sockfd*);

- sendto(*sockfd*, endereço_dest, processo_dest);

- recv(*sockfd*);

- recvfrom(*sockfd*, endereço_orig, processo_orig);

estas chamadas são usadas para o envio e recebimento de dados. Todas elas podem ser usadas em protocolos com e sem conexão.

- close(*sockfd*);

é usada para eliminar um socket. Se existir uma conexão associada ao socket, ela é encerrada.

No sistema 4.3BSD, *sockets* são implementados dentro do *kernel* do Unix. Todas as chamadas do sistema são pontos de entrada no *kernel*. A seguir, encontra-se a implementação de redes deste sistema:

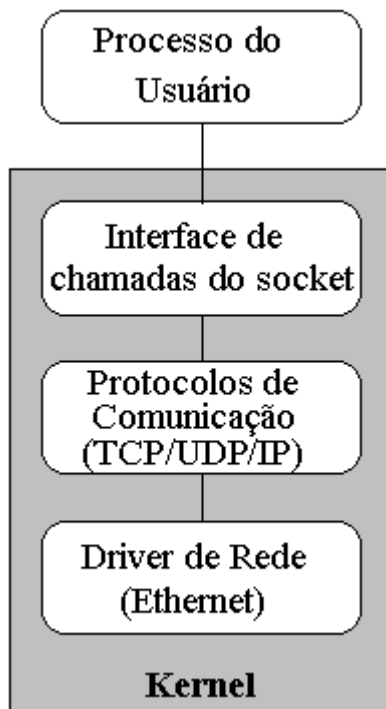


Fig. 8.2 - Implementação de redes no sistema 4.3BSD

NÍVEL DE SESSÃO

O objetivo do nível de sessão é fornecer os meios necessários para que seus usuários possam organizar e sincronizar seus diálogos e gerenciar suas trocas de dados.

Para dar suporte ao diálogo, o nível de sessão cria uma conexão lógica, chamada conexão de sessão, responsável por fornecer vários serviços que permitem estruturar o diálogo entre aplicações. Operando no modo orientado à conexão, o nível de sessão provê os serviços necessários para que seus usuários possam estabelecer conexões, transmitir e receber dados, e encerrar conexões de forma ordenada.

Uma conexão de sessão é mapeada em uma única conexão de transporte em um dado instante. Entretanto, a duração das conexões de sessão e transporte podem ser diferentes. Nesse caso, uma conexão de transporte pode suportar várias conexões de sessão consecutivas, ou uma conexão de sessão pode estender-se sobre várias conexões de transporte. A possibilidade de uma conexão de sessão estender-se sobre mais de uma conexão de transporte permite que o serviço de sessão recupere erros na rede que provoquem o encerramento de conexões de transporte.

Não existe controle de fluxo no nível de sessão. Para prevenir sobrecargas, as entidades de sessão utilizam o serviço de controle de fluxo fornecido pelo nível de transporte.

No modo não-orientado à conexão o nível de sessão mapeia endereços de transporte em endereços de sessão, repassando os dados que recebe dos seus usuários para o nível de transporte, e vice-versa. Este serviço só foi definido por imposição da estrutura hierárquica do modelo OSI, pois sua funcionalidade é nula, servindo apenas de intermediário entre seus usuários e o nível de transporte.

1. Conceitos Básicos

Os conceitos básicos são utilizados pelo nível de sessão para prestar seu serviço de estruturação e sincronização do intercâmbio de dados entre seus usuários.

1.1- Token

Um token é um recurso de uma conexão de sessão, concedendo a um usuário o direito exclusivo de executar determinados serviços:

Dados: usado para controlar o direito de transmitir dados. Quando o token de dados é utilizado em uma conexão de sessão, ela se torna half-duplex. Só o usuário que possui o token pode transmitir, conseqüentemente, só ocorrem transmissões em um único sentido da conexão por vez.

Sincronização secundária: define o usuário que tem direito de emitir pontos de sincronização secundários.

Sincronização principal/Atividade: é usado para gerenciar a emissão de pontos de sincronização principal e a estrutura das atividades.

Encerramento de conexão: define quem tem o direito de encerrar a conexão de sessão.

1.2 - Pontos de Sincronização

São definidos dois tipos de pontos de sincronização no nível de sessão: ponto de sincronização principal e ponto de sincronização secundário.

Fig. 1 - Relação entre os pontos de sincronização

Ponto de Sincronização Principal e Unidade de Diálogo

O ponto de sincronização principal é usado para estruturar a troca de dados em unidades de diálogo. Dois pontos de sincronização principais adjacentes definem uma unidade de diálogo.

Ao definir um ponto de sincronização principal, o usuário suspende a transmissão de dados até que o ponto de sincronização seja reconhecido pelo seu parceiro no diálogo. O ponto de sincronização principal marca o limite para recuperação de dados, não sendo possível recuperar dados enviados antes do último ponto de sincronização principal transmitido e reconhecido.

Uma vez que os pontos de sincronização principais são necessariamente reconhecidos, um diálogo pode ser mapeado em uma função específica de uma aplicação.

Ponto de Sincronização Secundário

Os pontos de sincronização secundários são usados para estruturar a troca de dados dentro de uma unidade de diálogo. Eles aumentam a flexibilidade em termos de recuperação de falhas.. Após transmitir um ponto de sincronização secundário, o transmissor não precisa suspender a transmissão de novos dados para esperar pelo reconhecimento do ponto de sincronização secundário. A qualquer momento, é possível ressincronizar o diálogo em qualquer um dos pontos de sincronização secundários dentro da unidade de diálogo corrente.

1.3 - Atividade

Fig. 2 - Relação entre pontos de sincronização e unidade de diálogo

O conceito de atividade permite que usuários do serviço de sessão possam distinguir unidades lógicas de um trabalho que são denominadas atividades.

Atividades podem ser interrompidas e posteriormente retomadas na mesma conexão ou em conexões de sessão subseqüentes. Uma atividade pode ser usada, por exemplo, para suspender uma transferência em curso, de um arquivo muito longo, para que seja transmitido um arquivo de maior prioridade. O serviço de sessão simplesmente guarda o último ponto de sincronização (principal ou secundário) confirmado, permitindo que a atividade seja retomada a partir desse ponto. Cabe ao usuário do serviço de sessão armazenar qualquer outro tipo de informação necessária ao reinício da atividade suspensa.

2 - Considerações sobre o Nível de Sessão

O nível de sessão estrutura a conexão de transporte, organizando a cadeia de octetos nela transmitida em unidades de diálogo e atividades. O serviço de sessão orientado à conexão é fornecido em três fases. inicialmente é estabelecida uma conexão de sessão, em seguida ocorre a fase de transferência de dados e por fim a conexão é encerrada.

2.1 - Estabelecimento de Conexão de Sessão

O serviço de estabelecimento de conexão de sessão permite que dois usuários do nível de sessão estabeleçam uma conexão entre si. No momento do estabelecimento da conexão, os dois usuários e o fornecedor do serviço de sessão negociam cooperativamente os valores dos parâmetros que definem os serviços que serão usados na conexão de sessão. Esses parâmetros são o identificador da conexão, a qualidade do serviço, as unidades funcionais e a distribuição inicial dos tokens que estarão disponíveis na conexão.

2.2 - Transferência de Dados

O nível de sessão fornece três tipos de serviço de transferência de dados: o serviço de transferência normal, o serviço de transferência expresso e o serviço de transferência de dados tipados.

O *serviço de transferência de dados normal* permite que o usuário do nível de sessão transfira unidades de dados do serviço de sessão (SSDUs) para outro usuário. Só o usuário que tiver a posse do token de dados poderá transmitir SSDUs.

O *serviço de transferência de dados expresso* torna possível o envio mais rápido de SSDUs expressas. Esse serviço é normalmente utilizado para envio de informações de controle quando o transporte de dados está bloqueado, por exemplo, pelo mecanismo de controle de fluxo do nível de transporte.

O *serviço de transferência de dados tipados* permite que um usuário do nível de sessão envie uma SSDU para outro, mesmo que ele não possua o token que controla o acesso ao serviço de transferência de dados.

Gerenciamento de Token

O gerenciamento dos tokens é feito com base nas primitivas de serviço *give-token*, *please-token* e *give-control*. A primitiva *give-token* é usada por um usuário para passar o controle de um token ao seu parceiro na conexão de sessão. A primitiva *please-token* é usada pelo usuário que não possui um token para solicitar que seu parceiro lhe entregue o token. A primitiva *give-control* é usada para passar todos os tokens de um usuário para outro.

Sincronização de Conexão de Sessão

O serviço de sincronização de conexão de sessão permite que usuários do nível de sessão possam definir e identificar pontos de sincronização ao longo de seu diálogo. Ele permite também que a conexão de sessão possa ser ressincronizada para um estado definido por um ponto de sincronização, o que implica no descarte dos dados transmitidos após o referido ponto.

O serviço de sessão, como definido no padrão ISO 8326, não se responsabiliza pelo armazenamento de dados que já tenham sido transmitidos. O serviço de sessão simplesmente marca a cadeia de dados transmitida com números atribuídos seqüencialmente, pontos de sincronização. Para que outros dados previamente enviados sejam retransmitidos pelo nível de sessão, o usuário do nível de sessão deve ter ele próprio armazenado os dados, devendo então entregá-los novamente ao nível de sessão para retransmissão.

Gerenciamento de Atividade

Só é permitida uma atividade em uma conexão de sessão em um determinado instante do tempo, porém, podem existir várias atividades consecutivas durante uma conexão de sessão. Uma pode adicionalmente estender-se sobre várias conexões de sessão.

O usuário do nível de sessão pode transmitir dados fora das atividades, através do mecanismo denominado *capability data*. Como, usualmente, cada atividade é normalmente associada a uma tarefa específica de uma aplicação, o mecanismo *capability data* pode ser usado para transmitir informações de controle que não estão relacionadas a nenhuma tarefa particular da aplicação.

- Uma atividade em uma conexão de sessão:

- Várias atividades em uma conexão de sessão:

- Uma atividade em várias conexões de rede:

Notificação de Exceções

O serviço de notificação de exceções torna possível a divulgação de situações excepcionais que ocorrem ao longo de uma conexão de sessão. A notificação pode ser enviada por um usuário ou pela entidade fornecedora do serviço de sessão.

2.3 - Encerramento de Conexão de Sessão

O serviço de encerramento de conexão de sessão permite que usuários do nível de sessão negociem o encerramento de uma conexão de forma ordenada. Nesse tipo de desconexão, os dados que estiverem em trânsito na conexão serão entregues antes de seu encerramento, não sendo assim extraviados.

3 -Unidades Funcionais do Serviço de Sessão

Uma unidade funcional é um agrupamento lógico de serviços. As unidades funcionais ativas em uma conexão de sessão determinam os serviços e os tokens que estarão disponíveis na conexão.

UNIDADE FUNCIONAL	SERVIÇO	TOKEN
Kernel	• estabelecimento da conexão • transferência de dados normal • encerramento ordenado • aborto iniciado pelo usuário • aborto iniciado pelo fornecedor	
Encerramento negociado	• encerramento ordenado • give-token • please-token	encerramento encerramento
Transferência half-duplex	• transferência de dados normal • give-token • please-token	dados dados
Transferência duplex	• transferência de dados normal	
Dados expressos	• transferência de dados expressa	

Dados tipados	• transferência de dados tipados	
Capability data	• transferência de capability data	
Sincronização secundária	• definição pto de sincron.secund. • give-token • please-token	sinc. secund. sinc. secund.
Sincronização simétrica	• sincronização simétrica	
Sincronização principal	• definição de pto sincron. princip. • give-token • please-token	sincr. princip sincr. princip
Ressincronização	• ressincronização	
Exceção	• comun. de exceção do usuário • comun. exceção do fornecedor	
Gerenciamento de atividade	• iniciar atividade • interromper atividade • retomar atividade • descartar atividade • encerrar atividade • give-control • give-token • please-token	atividade atividade atividade atividade atividade atividade atividade atividade

NÍVEL DE APRESENTAÇÃO

O nível de Apresentação tem como função realizar o mapeamento e a conversão entre uma representação utilizada para intercâmbio e a representação interna de aplicações. Compatibiliza a forma como são apresentadas as informações intercambiadas pelos usuários. Provê um formato comum para a representação dos dados transferidos, liberando a entidade de aplicação origem de qualquer consideração sobre como representar suas informações para processamento na entidade de aplicação destino.

Devido a complexidade das estruturas intercambiáveis por diversas aplicações, surgem problemas relacionados a definição das estruturas, e como a instância de uma estrutura pode ser intercambiada e interpretada por diferentes sistemas, independentemente da plataforma de hardware ou software utilizadas.

Para solucionar tais questões a ISO introduziu noções de sintaxe abstrata, sintaxe concreta e sintaxe de transferência, aliados à definição da camada de apresentação do RM-OSI.

Sintaxe Abstrata

É uma especificação da organização de uma estrutura de forma independente da codificação interna utilizada por máquinas para representá-la.

Torna possível a definição de tipos de dados e permite também que sejam atribuídos valores aos tipos de dados.

Como exemplo, de tipos de dados podemos citar inteiros, caracteres e structs, encontrados na linguagem C.

A ISO define um conjunto de tipos primitivos através de uma linguagem chamada ASN.1 (*Abstract Syntax Notation One*).

Sintaxe Concreta

É a especificação de um formato para codificação de instâncias de estruturas de dados a partir de uma sintaxe abstrata.

Por exemplo, um número inteiro pode ser codificado em sinal magnitude ou complemento de dois. A figura 10.1 dá uma idéia de como variam as representações concretas do número 534 em diferentes ambientes.

Fig. 10.1 - Exemplo de especificação ASN.1

Para definição de uma sintaxe concreta para o padrão ASN.1, a ISO definiu o padrão ASN.1 BER (*Basic Encoding Rules*). Ele contém o conjunto de regras que permite a construção de uma sequência de octetos que representam instâncias de estruturas definidas em ASN.1.

O padrão BER utiliza a abordagem "TLV" (*tag, length e value*) para realizar o mapeamento entre sintaxe abstrata e concreta.

No campo tag é colocado um rótulo. O campo length indica quantos octetos foram usados para codificar o valor atribuído à instância. O campo value carrega uma codificação de um valor específico do tipo de dado.

Cada valor é codificado com o menor número de octetos necessário para preservar a precisão de seu conteúdo.

Exemplo de sequência de octetos, na representação ASN.1 BER:

Fig. 10.2 - Exemplo de codificação ASN.1 Basic Encoding Rules (BER)

Sintaxe de Transferência

Utilizada pela sintaxe concreta para o intercâmbio e representação interna de aplicações. Realiza o mapeamento e conversão entre representações de intercâmbio e internas.

Contexto de Apresentação

É definido pelas combinações de sintaxe abstrata e sintaxe de transferência. Por exemplo, pode-se representar uma sintaxe abstrata específica através de uma ou mais sintaxes de transferências.

Fig. 10.3 - Conceitos básicos no nível de apresentação

A figura 10.3 ilustra a relação entre os conceitos de sintaxes abstrata, sintaxe de transferência e contexto de apresentação.

Dois usuários do nível de apresentação utilizam três sintaxes abstratas (A, B e C) e uma sintaxe de transferência (T) em sua comunicação. Dessa forma, três contextos são usados para definir que tipo de conversão as entidades de apresentação precisam realizar nos dados.

Considerações sobre o Nível de Apresentação

No RM-OSI, do nível físico ao de sessão, os dados do usuário são vistos como uma seqüência de octetos. O nível de apresentação deve ser capaz de converter a representação abstrata dos dados utilizados pelas aplicações (nível de aplicação) na referida seqüência.

Na prática, o que o nível de apresentação faz é transformar os tipos de dados da aplicação a serem intercambiados, em uma sintaxe concreta, e apresentar a seqüência de octetos resultante na forma de SSDUs (*Session Service Data Units*) ao serviço de sessão. No receptor são executadas as operações inversas.

Existem algumas funções no nível de apresentação criadas para ampliar os serviços do nível de sessão. São elas:

- **Gerenciamento de Contextos em uma Conexão de Apresentação**
- Mantém um registro sobre a combinação de sintaxes abstratas e de transferências utilizadas em conexão de apresentação. Quando é estabelecida uma conexão de apresentação, são definidos identificadores de contexto de apresentação (PCI - Presentation Context Identifier). Cada PCI relaciona uma sintaxe abstrata a uma sintaxe de transferência. É realizada uma negociação entre as entidades de aplicação. A coleção de contextos negociados define o conjunto de contextos da conexão (DCS - Defined Context Set). É possível adicionar-se ou remover-se contextos.

- **Transformação de Representações**

- Esta função é utilizada para mapear as instâncias de tipos de dados especificadas em sintaxes abstratas para sintaxes de transferência apropriadas. Em um sistema real, os dados definidos em termos de uma sintaxe abstrata são representados no ambiente local por uma sintaxe concreta. Necessita-se de uma transformação entre essas sintaxes para a transferências dos dados.
- Ela se dá em duas fases: inicialmente a representação concreta local é mapeada na representação abstrata, e depois esta é transformada na representação de transferência.
- Após a transformação podem ser usadas técnicas de compactação e criptografia .
- Em uma comunicação são utilizadas três versões de sintaxes concretas dos dados: sintaxe concreta usada pela entidade de aplicação origem, sintaxe concreta usada pela entidade de aplicação destino e a sintaxe concreta usada entre as entidades de apresentação. Elas podem ser idênticas ou não.

Unidades Funcionais

Assim como o nível de sessão, os serviços do nível de apresentação são agrupados em unidades funcionais:

A unidade funcional *presentation-kernel* está sempre disponível. Se duas aplicações utilizam uma única representação para os dados que trocam, apenas esta unidade funcional é necessária. A camada de apresentação funciona como serviço nulo. Caso contrário a camada de apresentação deve executar transformações.

A unidade funcional *gerenciamento de contexto* permite modificar o DCS ao longo da conexão de apresentação.

A unidade funcional *restauração de contexto* é usada junto com o serviço de sincronização da conexão do nível de sessão.

NÍVEL DE APLICAÇÃO

11.1 - O Nível de Aplicação no RM - OSI

O padrão ISO 9545 (Application Layer Structure) e a emenda (Extended Application Layer Structure) definem os conceitos, a terminologia e os modelos utilizados por todos os padrões do nível de aplicação OSI. A recomendação ITU-T X.207 é equivalente a esse padrão e foi usada como referência básica nesta seção.

Como único nível do RM-OSI que fornece serviços diretamente aos processos de aplicação, o nível de aplicação necessariamente provê todos os serviços OSI que podem ser usados por esses processos para trocar informações entre si.

O serviço do nível de aplicação, além da transferência de dados inclui, entre outros, os seguintes serviços:

- Identificação dos futuros parceiros na comunicação (por exemplo pelo nome ou endereço).

- Determinação da qualidade de serviço aceitável na conexão (por exemplo, retardo de transferência, taxa de erro tolerável, custo do serviço etc.).
- Sincronização das aplicações participantes.
- Definição da responsabilidade na recuperação de erros.
- Especificação de aspectos relativos à segurança (por exemplo, autenticação, controle de acesso, integridade dos dados etc.).
- Seleção do modo de diálogo (full duplex ou half duplex).
- Identificação de sintaxes abstratas.

O nível de aplicação contém todas as funções necessárias à comunicação entre sistemas abertos que não são executadas pelos outros níveis, incluindo, tanto funções executadas por programas, quanto funções executadas por seres humanos.

11.1.1- Processo de Aplicação

Dentro de um sistema aberto OSI, um *processo de aplicação* é uma representação abstrata de um conjunto de recursos que podem ser usados para executar uma atividade de processamento de dados específica. Só as partes do conjunto de recursos relativas ao aspecto de comunicação são assuntos dos padrões elaborados no âmbito do RM-OSI.

Os processos de aplicação são identificados global e univocamente por seu *título de processo de aplicação*.

Como nível superior do RM-OSI, o nível de aplicação provê aos processos de aplicação o acesso ao ambiente desse modelo. O ambiente OSI é definido pela ISO [ISO 92] como sendo uma representação abstrata do conjunto de conceitos, elementos, funções, serviços, protocolos, etc. definidos pelo RM-OSI e os padrões específicos dele derivados que, quando aplicados, tornam possível a comunicação entre sistemas abertos.

Os módulos que compõem um processo de aplicação são divididos em duas partes. Uma delas implementa as funções de processamento de dados particulares a uma dada aplicação; a outra, a *entidade de aplicação*, trata dos aspectos relativos ao intercâmbio de informações, ou seja, dos serviços de comunicação OSI. A figura 11.1 mostra a estrutura do nível de aplicação do RM-OSI.

11.1.2- Entidade de Aplicação

Uma entidade de aplicação OSI (Application Entity -AE), representa um conjunto de facilidades de comunicação de um processo de aplicação que são responsáveis pelo intercâmbio de arquivos e mensagens. Um processo de aplicação contém uma ou mais entidades de aplicação.

As entidades de aplicação em um processo de aplicação, se distinguem através do uso de um qualificador. O qualificador concatenado ao título do processo de aplicação forma o *título da entidade de aplicação*.

Uma AEI (Application Entity Invocation) é uma utilização específica de parte, ou de todos os recursos de comunicação de uma dada AE. A AEI é usada para atender os requisitos de

comunicação de um processo de aplicação. Processos de aplicação trocam informações por meio de AEs que utilizam os protocolos do nível de aplicação e os serviços do nível de apresentação.

Fig. 11.1 - Estrutura do Nível de Aplicação do RM-OSI

11.1.3- Elemento do Serviço de Aplicação

Um *elemento do serviço de aplicação* (Application Service Element - ASE) define um componente básico de uma entidade de aplicação. Os ASEs implementam facilidades específicas no ambiente de comunicação OSI, por exemplo, o ACSE (Association Control Service Element) estabelece associações entre entidades de aplicação, e o ROSE (Remote Operation Service Element) permite que uma aplicação execute uma operação em um processo de aplicação remoto.

Cada ASE é definido por dois padrões: um define o serviço por ela oferecido e o outro o protocolo que ela utiliza para implementar o serviço. As ASEs são combinadas de diferentes formas para formar diferentes tipos de entidades de aplicação.

11.1.4- Objeto do Serviço de Aplicação

Um *objeto do serviço de aplicação* (Application Service Object - ASO) é uma composição usada para estruturar a camada de aplicação. Um ASO encapsula diversos agrupamentos de funções utilizadas na comunicação entre aplicações. Cada agrupamento específico presta um serviço e utiliza um protocolo de aplicação específico. Os agrupamentos são representados por ASEs ou outros ASOs, assim, um ASO pode conter:

- Um ou mais ASEs e uma função de controle.
- Um ou mais ASOs e uma função de controle.
- Um ou mais ASEs, um ou mais ASOs, e uma função de controle.

Uma ASOI (Application Service Object Invocation) representa uma utilização específica de parte, ou de todos os recursos de um lado ASO. Uma ASOI inclui a informação de estado que espelha seu comportamento em uma comunicação.

Uma AE é um ASO, em uma configuração particular onde um ASO não é contido por nenhum outro ASO. Uma AEI é uma ASOI do ASO equivalente à AE. A figura 11.1 ilustra bem a situação.

Dentro de uma ASOI existem outras ASOIs ou ASEIs que correspondem aos ASOs ou ASEs que compõem o ASO em questão.

A função de controle de uma ASOI determina como seus componentes ASEIs e ASOIs operam cooperativamente de modo a habilitar a ASOI a fornecer um serviço OSI específico, como resultado de sua operação conjunta com as ASOIs parceiras.

11.1.5- Função de Controle

Uma função de controle (Control Function - CF) coordena a atuação conjunta dos componentes ASEs e ASOs onde ela reside. A função de controle de uma ASOI é responsável pelo fornecimento do serviço OSI que cabe à ASOI. Para tal, a CF utiliza de forma coordenada os serviços OSI fornecidos pelos ASOIs e ASEIs contidos na ASOI, e os serviços OSI externos à ASOI, fornecidos por outras ASOIs ou pelo nível de apresentação.

Para executar suas funções, a CF de uma ASOI mantém informações de estado relativas à coordenação local dos componentes da ASOI e relativas à coordenação da interação das ASOIs e ASEIs contidas na ASOI em questão, com suas parceiras localizadas em outras ASOIs.

11.1.6- Associação no Nível de Aplicação

Uma conexão é definida no RM-OSI como um serviço fornecido a usuários que estão no nível superior. Como não existe nível superior ao de aplicação, não podem existir conexões nesse nível. Em vez disso, as entidades de aplicação estabelecem associações, para viabilizar o intercâmbio de informações entre processos de aplicação. Uma *associação* é uma relação cooperativa entre entidades de aplicação formada para que possam ser intercambiadas unidades de dados do nível de aplicação (APDUS), através do serviço fornecido pelo nível de apresentação.

A recomendação X.207 define uma *associação* entre ASOs (ASO-association) como um relacionamento cooperativo entre duas ou mais ASOIs. As associações ASO são abstrações que relacionam ASOIs que estão intercambiando APDUs. As características entre um contexto ASOs são descritas em um contexto ASO.

Note que uma associação no nível de aplicação é um caso particular de uma associação entre ASOIs, que utiliza o serviço de apresentação para comunicação e coordenação de sua operação conjunta.

11.1.7- Contexto de Aplicação

Um *contexto ASO* define o conjunto de regras que governa a comunicação entre ASOIs, e o conhecimento que elas devem compartilhar para que possam operar cooperativamente em uma associação ASO. Um contexto ASO inclui:

- Um conjunto de regras e informações de estado.
- O número de ASOIs que são parceiras em uma associação ASO.
- A duração da associação ASO, e a especificação da forma como ela pode ser estabelecida e encerrada.

Um *contexto de aplicação* é um tipo de contexto ASO. O contexto de aplicação descreve o comportamento de uma associação ASO específica.

11.2 - Padrões ISO para Serviços de Aplicação

Os serviços de aplicação são organizados em elementos de serviço de aplicação de uso geral e elementos de serviço de aplicação específicos.

Os elementos de serviço de uso geral implementam uma série de funções básicas que são utilizadas pelas entidades de aplicação e por outros elementos de serviço. A ISO padronizou os seguintes elementos de serviço de uso geral: ACSE (Association Control Service Element) e CCR (Commitment, Concurrency and Recovery elementos de serviço de uso geral: ACSE (Association Control Service Element) e CCR (Commitment, Concurrency and Recovery).

Vários padrões têm sido definidos para fornecer serviços específicos de aplicação. Alguns exemplos são o X.400 que define um serviço de correio eletrônico, o FTMA (File Transfer, Access and Management) para manipulação de arquivos, e o X.500 que define um serviço de diretório

A tabela 11.1 lista os principais padrões utilizados no nível de aplicação:

Tabela 11.1 - padrões utilizados no nível de aplicação

11.2.1 - ACSE (Association Control Service Element)

Cada tipo de aplicação demanda uma combinação diferente de ASEs para formar o protocolo de aplicação específico. O ACSE é o elemento de serviço que gerencia as associações, estando presente na grande maioria das aplicações OSI.

O ACSE acrescenta à semântica da conexão de apresentação, que dá suporte à associação, a possibilidade de identificação de objetos através de nomes. Cada entidade de aplicação possui um título que identifica o serviço por ela oferecido. Cabe salientar que o título é independente de qualquer consideração sobre endereçamento e roteamento. Ele é simplesmente um nome.

O ACSE fornece um serviço de encerramento da associação ordenado, com os dados em trânsito sendo entregues antes da desconexão.

11.2.2 - RTSE (Reliable Transfer Service Element)

O RTSE fornece mecanismos genéricos que, em casos onde uma falha do sistema de comunicação interrompe a conexão inicialmente utilizada na transferência de dados, permitem o restabelecimento da transferência de dados do ponto onde foi suspensa, em uma nova conexão.

O RTSE utiliza o ACSE, para estabelecer e encerrar associações, e alguns dos mecanismos de sincronização e gerenciamento de tokens e de atividades fornecidas pelo serviço de sessão (através do serviço de apresentação), para garantir a confiabilidade na transferência de dados.

A figura 11.2 mostra as primitivas de serviço trocadas pelo RTSE com seus usuários, com o ACSE e com o serviço de apresentação.

Fig. 11.2 - Primitivas de serviço trocadas pelo RTSE com seus usuários, com o ACSE e com o serviço de apresentação

O RTSE utiliza o ACSE para encerrar uma associação de forma ordenada, sem perda de dados em trânsito. Existe a alternativa de abortar a associação, neste caso, com possível perda de dados.

11.2.3 - ROSE (Remote Operations Service Elements)

O principal serviço fornecido pela ROSE é a solicitação de execução de uma operação em um sistema aberto remoto.

A sua função é coordenar as interações solicitação/resposta entre entidades de aplicação, que acontecem no contexto de uma associação de aplicação.

O ROSE implementa também a noção de Linked-operations ou Callback.

A utilização das linked-operations do ROSE depende da associação acordada entre as entidades iniciadora entidade da classe da associação acordada entre entidades iniciadora (entidade que emitiu a primitiva A-ASSOCIATE.request do ACSE) e respondedora (entidade que emitiu a primitiva A-ASSOCIATE.response do ACSE). Nas associações de classe 1, o ROSE só permite que a entidade iniciadora solicite execução de operações remotas. Nas associações de Classe 2 só o acolhedor pode emitir solicitações. E nas associações de classe 3 ambos podem emitir solicitações.

A figura 11.3 ilustra a utilização do ROSE em uma associação de classe 3.

Fig. 11.3 - Mecanismo linked operations no ROSE

A interação entre duas entidades envolvidas na execução de uma operação é caracterizada por uma classe de operação ROSE. A classe de operação define se o modo de operação é síncrono (O solicitante após invocar uma operação aguarda a resposta da tentativa de execução antes de invocar outra operação), ou assíncrono (o solicitante pode invocar várias operações sem ter recebido os resultados de solicitações anteriores feitas).

11.2.4 - CCR (Commitment, Concurrency and Recovery)

O CCR fornece mecanismos que permitem que duas entidades de aplicação distintas coordenem seu processamento, no sentido de garantir que a execução de determinadas operações só ocorra, quando ambas as entidades estiverem prontas para executá-las. O CCR provê também mecanismos que permitem a recuperação dos dados processados para um estado consistente, quando ocorre alguma falha durante a execução do procedimento.

No CCR uma ação atômica é definida como sendo uma operação de processamento indivisível. Se ocorrer algum erro durante o processamento da ação atômica, o valor que os dados delimitados possuíam antes do início do processamento da ação é restaurado.

O CCR adota o modelo mestre/escravo para o relacionamento das duas entidades de aplicação envolvidas no processamento de uma ação atômica. O intercâmbio entre o mestre e o escravo baseia-se no procedimento mostrado na figura 11.4:

Fig. 11.4 - Execução de uma ação atômica sob a coordenação do CCR

A primeira fase do procedimento começa quando o mestre informa ao escravo que será iniciado o processamento de uma ação atômica. Durante o processamento, o mestre e o escravo podem trocar dados entre si. Quando termina o processamento, o mestre e o escravo podem trocar dados entre si. Quando termina o processamento da ação atômica, o mestre pergunta ao escravo se ele está preparado para atualizar os dados delimitados. Se o escravo responder que sim, a primeira fase é encerrada e o procedimento entra na segunda fase. Se ele responder que não, a ação retorna (roll-back) para seu início.

Na segunda fase do procedimento two-fase-commitment, o mestre atualiza os dados em seu sistema local e solicita ao escravo que faça o mesmo. Cabe salientar que o mestre atualiza os dados, mas mantém uma cópia dos valores dos dados delimitados antes da atualização. Ao terminar a atualização dos dados, o escravo envia uma mensagem para o mestre avisando se a atualização foi bem sucedida ou não. Em caso positivo, o mestre desfaz-se da cópia dos valores iniciais dos dados delimitados. Em caso de negativo, a ação retorna para o seu início, fazendo com que o mestre e o escravo restaurem os valores iniciais dos dados delimitados.

11.2.5 - SERVIÇO DE CORREIO ELETRÔNICO X.400

No X.400, como em outros sistemas de correio eletrônico baseados em computadores, os usuários trocam informações na forma de mensagens, podendo o sistema, opcionalmente, avisar ao remetente da entrega ou do recebimento da mensagem. O MHS (Message Handling Systems) pode ser usado para a troca de correspondência interpessoal ou para troca de documentos comerciais.

Uma mensagem no MHS consiste em um envelope e um conteúdo.

No conteúdo são armazenadas as informações que o remetente deseja enviar ao destinatário. Se a mensagem for impessoal (IPM) o conteúdo consiste em um cabeçalho e um corpo. O corpo contém os dados propriamente ditos, e o assunto, etc.

O MHS é definido em termos dos componentes funcionais, mostrados na figura 11.5:

Fig. 11.5 - Modelo funcional do MHS

O sistema consiste em um MTS (Message Transfer System) e de vários UAs (User Agents). Um UA é como uma caixa postal do usuário, auxiliando-o na criação, envio e recepção de mensagens. O usuário pode ser uma pessoa ou um processo de aplicação. Existe um UA por usuário do sistema de correio eletrônico.

Cada UA é ligado a um único MTA. O MTA recebe e entrega mensagens entre os UAs a ele conectados diretamente, e transfere para outros MTAs as mensagens endereçadas aos UAs que não estão diretamente ligados a ele.

Quando o caminho a ser seguido por uma mensagem está interrompido, ou quando a mensagem é corrompida por erros de transmissão, o MTA faz várias tentativas de transmissão, dentro de um intervalo de tempo limitado. Se dentro desse intervalo ele não conseguir entregar a mensagem, ela é destacada e uma notificação é enviada para o remetente.

O X.400 define um esquema global para definição de nomes e de endereços, onde cada usuário é identificado de forma ambígua por um endereço. Os endereços no MHS consistem nos seguintes campos:

- Nome do país: código com dois caracteres (ISO 3166) ou com três caracteres (CCITT X.121) que identifica o país onde está o usuário.
- Nome ADMD: nome do domínio administrativo ao qual está conectado o usuário
- Nome PRDM: nome do domínio privado ao qual está conectado o usuário.
- Nome da organização: nome da organização onde o usuário é encontrado.
- Nome da unidade na Organização: nome da unidade dentro da organização onde o usuário é encontrado.
- Nome pessoal: nome do usuário

11.2.6 - SERVIÇO DE DIRETÓRIO X.500

Um diretório é um banco de dados, mantido conjuntamente por uma coleção de sistemas abertos distribuídos, com informações sobre um conjunto de objetos reais (instância de identidades de aplicação, usuários, máquinas, etc.) desse sistemas. Os usuários do diretório, que podem ser pessoas ou processos de aplicação, podem ler ou modificar a informação nele armazenada, caso tenham permissão para tal.

Os comitês de padronização reconhecendo a importância do serviço de diretório eletrônico em qualquer sistema de comunicação de grande porte, elaboram a série de padrões X.500, publicada em 1988, e os padrões ISO 9594, partes 1 a 8, que atingiram o status de padrões internacionais (IS) em 1990.

A função mais utilizada do serviço de diretório é a recuperação do endereço de entidades com base em seu nome lógico. Um nome lógico deve ser facilmente manipulado e memorizado por seres humanos, e independentes da localização da entidade. É muito comum encontrar pessoas e outros objetos que são reconhecidos por vários nomes. O serviço de diretório associa um único nome a cada um dos seus objetos, permitindo que eles seja, localizados por nomes alternativos e abreviações.

O serviço de diretório pode ser utilizado em procedimentos de controle de acesso, centralizando a administração das informações. O diretório pode ser usado para armazenar a senha de um usuário específico, ou chaves usadas em sistemas de criptografia, devidamente protegidas contra violações. A figura 11.6 mostra um exemplo onde o usuário B utiliza o serviço de diretório para autenticar a senha do usuário A.

Fig. 11.6 - Autenticação de um usuário através da verificação de senha

Uma vantagem do esquema acima é que a identidade de um usuário pode ser autenticada por diferentes aplicações, que não precisam armazenar as senhas de seu cliente. Outra vantagem é que quando o usuário deseja modificar sua senha, só precisa atualizar um único ponto, o diretório. A DIB (Directory Information Base) contém todas as informações que podem ser obtidas através do serviço de diretório. A DIB é composta por instâncias de objetos chamadas entradas. Cada entrada consiste em um ou mais atributos.

A presença e o tipo de atributos de uma entrada dependem da classe do objeto que a instancia. Por exemplo, as entradas com instâncias de objetos de classe "entidade de aplicação", incluem os seguintes atributos: nome, endereço de apresentação, descrição, contexto de aplicação usados, etc.

O serviço de diretório é distribuído, sendo composto por vários DSAs (Directory Service Agents). Cada DSA gerencia uma parte do diretório. Um usuário do serviço de diretório é definido como sendo uma pessoa ou um processo de aplicação. Um usuário tem acesso ao diretório através de um DUA (Directory User Agents). Existe um DUA para cada usuário do serviço de diretório. O DUA comunica-se com um DSA, atendendo a solicitação de usuários, para ter acesso a informações no diretório. Se a informação estiver disponível no DSA, ele responde devolvendo a informação solicitada ao DUA, que então a entrega ao usuário. Se o DSA não possuir a informação solicitada, ele pode:

- Devolver uma indicação (referral), com o endereço de apresentação de outro DSA que contém, ou pelo menos está próximo da informação desejada.
- Nesse caso o DUA emite uma nova solicitação para o novo DSA, como pode ser observado na figura a).
- Contatar com outros DSAs, repassando duas variações: na primeira os vários DSAs são consultados simultaneamente c). E na segunda vários DSAs são consultados, porém, um de cada vez d).

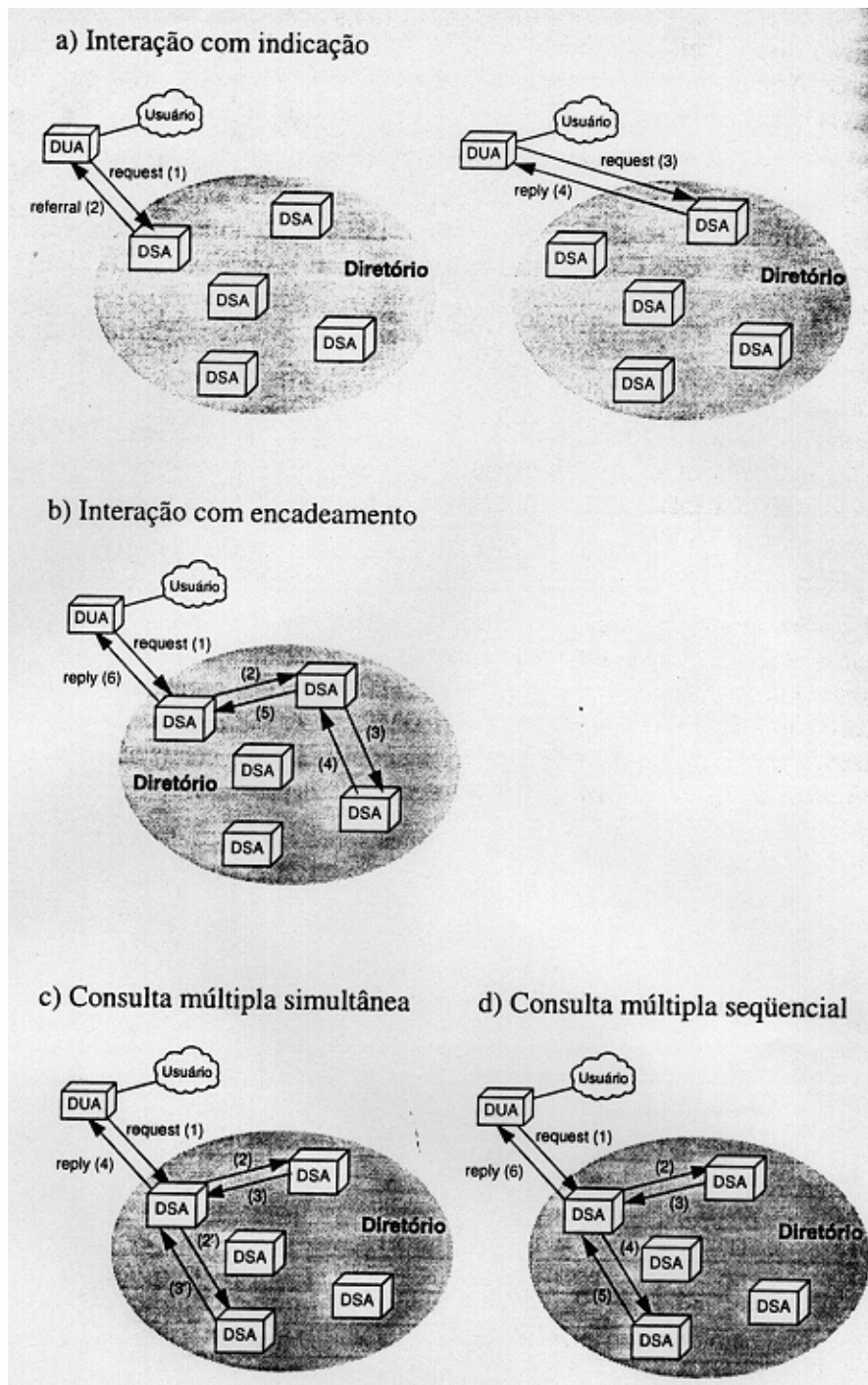


Fig. 11.7 - Interação entre DUA e DSA

11.2.7 - FTAM (File Transfer, Access and Management)

O FTMA permite que seus usuários tenham acesso a arquivos em sistemas remotos. O FTAM permite que os arquivos sejam transferidos como uma unidade, ou que tenham acesso por partes em operações de leitura e escrita, e que operações de criação, remoção e manipulação dos atributos de arquivos possam ser realizadas.

As propriedades de um arquivo FTAM são descritas pelos valores de seus atributos. O atributo nome do arquivo identifica o arquivo de forma não ambígua. Esse atributo foi substituído pelo atributo pathname no adendo Filestore Management. Outros exemplos de atributos são: lista de ações permitidas (leitura, escrita, inserção, etc.), lista de controle de acesso, tamanho, data e hora de criação, identificação do criador, etc.

O FTAM adotou uma estrutura em árvore para organizar o acesso a seus arquivos. Cada nó da estrutura é associado a, no máximo, uma unidade de dados. O FTAM permite que um arquivo seja decomposto em sub-árvores denominadas FADU (File Access Data Units), que definem a unidade de acesso ao arquivo. Se a estrutura em árvore do arquivo tiver apenas um nível, o arquivo é dito ser não estruturado. Se a árvore tiver dois níveis, a estrutura assemelha-se a de um arquivo sequencial (flat). A estrutura de operação do FTAM é mostrada na figura 11.8.

Fig. 11.8 - Modelo de operação FTAM

11.2.8 - Gerenciamento de Redes OSI

O gerenciamento de redes fornece mecanismos para monitoração, controle e coordenação da utilização de recursos em um ambiente OSI. O gerenciamento de redes OSI é o tema dos padrões ISO, e das recomendações do CCITT correspondentes.

Os documentos que definem o gerenciamento de redes OSI dividem essa tarefa em cinco partes:

- Gerenciamento de falhas: define recursos que permitem a detecção, o isolamento, e a correção da operação irregular dos componentes do ambiente OSI.
- Gerenciamento de contabilização: provê os meios para que possa ser medida a utilização dos recursos gerenciados e para que possam ser calculados os custos dessa utilização.
- Gerenciamento de configuração : define os meios utilizados para identificar, coletar dados, supervisionar e controlar a configuração dos componentes do ambiente OSI, no sentido de garantir a operação contínua do serviço de comunicação.
- Gerenciamento de desempenho: fornece os mecanismos utilizados para avaliar o comportamento dos recursos gerenciados e calcular a eficiência das atividades de comunicação.
- Gerenciamento de segurança: Endereça os aspectos relativos à segurança no acesso aos recursos utilizados no ambiente OSI.

O modelo de gerenciamento OSI define um sistema integrado de gerenciamento, que é composto de agentes, gerentes e objetos gerenciados junto aos agentes, processa as informações, se for necessário, solicita aos sistemas agentes que executem operações de gerenciamento a fim de controlar o funcionamento dos objetos gerenciados.

O agente coleta informações relativas ao funcionamento dos objetos que gerencia, armazena essas informações na MIB (Management Information Base) e realiza operações de gerenciamento sobre esses objetos atendendo a solicitações enviadas pelo gerente.

Um objeto gerenciado é uma representação de um recurso real do ambiente OSI sujeito ao gerenciamento.

A arquitetura do modelo OSI é representada na figura 11.9.

Fig. 11.9 - Arquitetura do modelo de gerenciamento OSI

11.3 - O Nível de Aplicação Internet TCP/IP

Na arquitetura Internet TCP/IP as aplicações, especificadas em documentos denominados RFCs (Request For Comments), são implementadas de forma isolada. Não existe um padrão que defina como deve ser a estrutura de uma aplicação, como no RM-OSI. As aplicações trocam dados utilizando diretamente a camada de transporte (TCP ou UDP), através de chamadas padronizadas, como APIs.

Além de compartilhar o mesmo conjunto de primitivas de transporte, muitas das aplicações Internet TCP/IP adotam o modelo cliente/servidor, onde denomina-se servidor a qualquer aplicação que ofereça um serviço a outra aplicação, podendo o serviço ser requisitado através da inter-rede. Os servidores aceitam solicitações, que chegam pela inter-rede, executam o serviço solicitado e enviam de volta a aplicação solicitante. Uma aplicação torna-se cliente quando envia uma solicitação a um servidor e espera por uma resposta.

Na figura 11.10 são apresentadas as principais aplicações disponíveis na arquitetura Internet TCP/IP.

Fig. 11.10 - Aplicação Internet TCP/IP

11.3.1 - DNS (Domain Name System)

O DNS é um esquema de gerenciamento de nomes, hierárquico e distribuído. Ele define a sintaxe dos nomes usados na Internet, regras para delegação de autoridade na definição de nomes, um banco de dados distribuído que associa nomes a atributos (ex. Endereço IP) e um algoritmo distribuído para mapear nomes em endereços. O DNS é especificado nas RFCs 882, 883 e 973.

Normalmente as aplicações utilizam um endereço de 32 bits para abrir uma conexão ou enviar um datagrama IP, porém usuários preferem identificar máquinas através de nomes, sendo necessário um banco de dados para permitir a uma aplicação encontrar um endereço, quando ela conhece o nome da máquina a ser comunicada.

Um conjunto de servidores de nomes mantém o banco de dados com nomes e endereços das máquinas conectadas a Internet. Sendo esta um tipo de informação armazenada no domain system (sistema de domínios). É usado um conjunto de servidores interconectados e não um único servidor centralizado. Assim os servidores de nome formam uma árvore, correspondendo à estrutura institucional. Os nomes adotam também uma estrutura similar.

Exemplo típico é o nome Copacabana. Inf. Puc-Rio.Br. Para encontrar o endereço IP dessa máquina pode ser necessário consultar até quatro servidores de nomes. Inicialmente consulta-se um servidor raiz para descobrir onde está o servidor Br, responsável pela gerência de nomes das instituições brasileiras ligadas a Internet. O raiz informa, após consulta, o endereço IP de vários servidores par o

nível Br (fim de continuidade). Um servidor Br, consultado, devolve o endereço do servidor Puc-Rio. Com esse endereço pode-se solicitar o endereço do servidor Inf e então consultá-lo sobre o endereço da máquina Copacabana. O resultado da busca é o endereço Internet do nome Copacabana.Inf.Puc-Rio.Br.

Cada nível percorrido é um domínio. O nome completo do exemplo é um nome de domínio, como são também os nomes dos níveis superiores: Inf.Puc-Rio.Br, Puc-Rio.Br e Br. Nem sempre todos os domínios de um nome são acessados (um servidor de nomes pode conter informações sobre mais de um domínio) e além disso as aplicações podem acessar o DNS por um processo local (servidor para as aplicações e um cliente DNS), pode ser implementado para guardar os últimos acessos feitos e resolver a consulta a nível local, eliminando o algoritmo de percorrer a árvore de domínios.

O DNS não se limita a manter e gerenciar endereços Internet, mas também armazenar informações sobre usuários, listas de distribuição ou outros objetos e é importante para o sistema de correio eletrônico, onde são definidos registros no DNS que identificam a máquina que manipula as correspondências relativas a um dado nome.

11.3.2 - FTP (File Transfer Protocol)

O FTP permite que um usuário em um computador transfira, renomeie ou remova arquivos remotos; ou crie, remova e modifique diretórios remotos. O FTP só permite a transferência de arquivos completos. É necessário que o usuário solicitante (cliente) envie seu nome (login name) e sua senha para o servidor; caso o usuário não tenha sido registrado ele será impedido de executar qualquer operação.

A operação do FTP baseia-se no estabelecimento de duas conexões entre o cliente (módulo FTP solicitando acesso a arquivos remotos) e o servidor (módulo FTP que fornece o acesso a seus arquivos locais), como na figura 11.11.

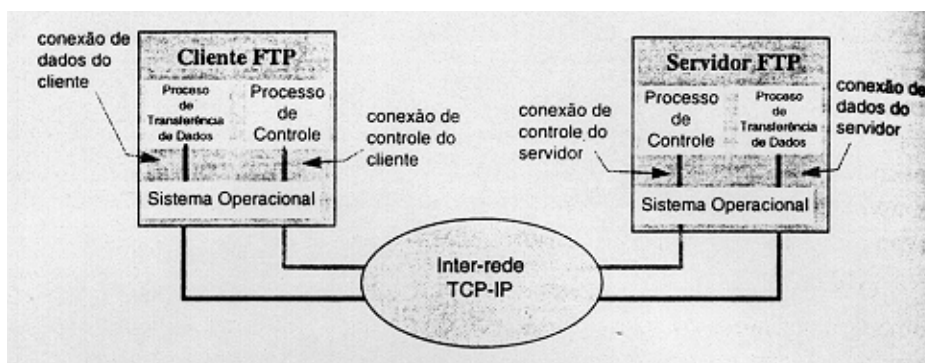


Fig. 11.11 - Conexões FTP de controle e de transferência de dados

O FTP permite que arquivos do tipo texto ou binário sejam transferidos, os de texto são manipulados como compostos por uma cadeia de caracteres ASCII ou EBCDIC, os do tipo binário são tratados como sequências de octetos e transferidos sem nenhuma conversão. O FTP providência a compatibilização do código dos caracteres, delimitadores de linha etc., onde os dados têm representações diferentes.

A arquitetura TCP/IP define, adicionalmente, um protocolo que fornece um serviço simplificado de transferência de arquivos, o TFTP (Trivial File Transfer Protocol). O TFTP restringe sua operação

apenas a transferência de arquivos, não implementando mecanismos de autenticação e opera em uma única conexão. Utiliza o UDP para o transporte de blocos de dados de tamanho fixo. Como o serviço fornecido pelo UDP não garante a entrega dos blocos ao destinatário o TFTP utiliza o protocolo de bit alternado para transmitir seus blocos.

11.3.3 - NFS (Network File System)

O NFS, desenvolvido pela Sun Microsystems, permite que um sistema tenha acesso a arquivos localizados remotamente de um modo integrado e transparente. O NFS fornece a ilusão que os discos, impressoras, ou outros dispositivos, fisicamente localizados em um sistema remoto, estão diretamente conectados ao sistema local.

Os dispositivos virtuais adicionais são associados a dispositivos fisicamente localizados em outras máquinas. O NFS permite, por exemplo, que estações que sem dispositivos de armazenamento, ou que possuam dispositivos com pouca capacidade, armazenem informações de modo transparente em equipamentos que possuam espaço disponível. Permite também a usuários em máquinas distintas compartilhar o acesso a arquivos comuns.

O NFS cria um sistema de arquivos virtual que estende o sistema de arquivo local. O sistema de arquivos virtual possui entradas que identificam os arquivos remotos de forma não ambígua no servidor NFS onde eles estão fisicamente localizados. A figura 11.12 mostra como o NFS é incorporado a um sistema.

Fig. 11.12 - Arquitetura de um cliente NFS

O NFS, ao contrário das outras aplicações Internet, não é implementado por um único utilitário, sendo dividido em três componentes: o protocolo NFS propriamente dito, uma ferramenta genérica que controla a execução de chamadas a procedimentos remotos (Remote Procedure Call - RPC) e uma terceira ferramenta que compatibiliza formatos diferentes de representação de dados denominada de XDR (External Data Representation). A divisão foi realizada para permitir que o RPC e o XDR pudessem ser utilizados em outras aplicações.

RPC (Remote Procedure Call)

O mecanismo RPC foi projetado para facilitar o desenvolvimento de aplicações distribuídas baseadas no modelo de interação cliente-servidor. No módulo cliente, o programador declara alguns procedimentos como sendo remotos, forçando o compilador a incorporar à chamada de procedimento, o código que repassa os argumentos da chamada ao cliente RPC. No módulo servidor, são implementados os procedimentos a serem compartilhados, que são declarados como procedimentos servidores.

Quando o programa cliente faz uma chamada a um procedimento remoto, os argumentos de chamada são entregues ao cliente RPC, que compõe uma mensagem e a envia ao servidor, passando então a aguardar o retorno do resultado. O servidor RPC, ao receber uma mensagem invocando a execução de um procedimento, dispara sua execução e, ao seu término, envia os resultados obtidos para o cliente RPC. O cliente RPC, ao receber de volta a mensagem com o resultado da execução do procedimento remoto, entrega esses resultados ao processo de aplicação, colocando-os nos respectivos argumentos de chamada de procedimento. Do ponto de vista do processo de aplicação, o

procedimento é executado como se fosse local. O RPC esconde todos os aspectos de intercâmbio de mensagens relativos à execução remota do procedimento.

XDR (External Data Representation)

A ferramenta XDR permite que os programadores escrevam aplicações distribuídas onde são intercambiados dados entre máquinas que os representam de formas distintas, sem que seja necessário escrever procedimentos para compatibilizar as representações.

A solução adotada na ferramenta XDR foi a definição de uma representação independente de qualquer máquina que é usada para codificar os dados intercambiados.

Quando se deseja transmitir dados, um programa de aplicação chama os procedimentos XDR que convertem a representação local para a representação de transferência e, em seguida, transmite os dados convertidos. O processo que recebe os dados chama os procedimentos XDR, que convertem à representação local específica.

11.3.4 - TELNET

O protocolo TELNET permite que um usuário utilizando uma máquina A estabeleça uma sessão interativa com uma máquina B na rede. A partir daí, todas as teclas pressionadas na máquina A são repassadas para a máquina B como se o usuário estivesse utilizando um terminal ligado diretamente a ela. Os comandos digitados na máquina A são processados na B e o resultado de sua execução é enviado de volta para ser exibido no monitor da máquina A.

O módulo cliente do TELNET permite que o usuário identifique a máquina a qual deseja se conectar pelo nome ou por seu endereço IP. A segunda opção torna possível o estabelecimento de conexões remotas em ambientes, onde não seja possível fazer a associação entre nomes e endereços IP.

O protocolo TELNET é construído com base em três idéias principais: o conceito de terminal virtual de rede; o princípio de negociação de opções; e o tratamento equivalente de terminais e processos.

Quando uma conexão TELNET é estabelecida, assume-se que ambas extremidades estão ligadas a um terminal virtual de rede (NVT - Network Virtual Terminal). Um NVT é um dispositivo lógico que fornece uma representação padronizada de um terminal, isto elimina a necessidade das máquinas participantes da conexão conhecerem as características do terminal usado por seus parceiros. Tanto o servidor como o cliente, mapeiam as características de seus terminais locais em características do NVT, assumindo que o parceiro faz o mesmo.

O princípio da negociação de parâmetros permite que os usuários envolvidos em uma conexão negociem opções que definem o comportamento do terminal virtual. Exemplo: o formato de representação dos caracteres utilizados e o modo de operação (half ou full duplex).

Uma vez que este protocolo trata os terminais e o processo de forma equivalente e simétrica, o cliente não precisa ser de fato um terminal, podendo ser um processo de aplicação qualquer. A simetria da conexão implica que ambas as extremidades podem tomar a iniciativa das negociações de opções, numa arquitetura cliente/servidor básica, o servidor é passivo (só responde a solicitações do cliente).

A figura 11.13 mostra a conexão TELNET.

Fig. 11.13 - Conexão TELNET

Funcionamento:

Programa TELNET inicia execução numa máquina (cliente). Junto com comando de disparo da execução do TELNET, o usuário informa o nome ou o endereço IP da máquina a ser conectada. Uma conexão TCP é estabelecida, e a máquina destino assume o papel de servidor TELNET. Estabelecida a conexão, todas as teclas pressionadas pelo usuário são capturadas pelo cliente, enviadas via TCP ao servidor remoto, que processa as teclas e envia de volta ao cliente os caracteres que devem ser mostrados no vídeo do terminal.

11.3.5 - SMTP (Simple Mail Transfer Protocol)

O SMTP é o protocolo usado no sistema de correio eletrônico na arquitetura Internet TCP/IP. A figura 11.14 mostra os componentes conceituais utilizados no sistema.

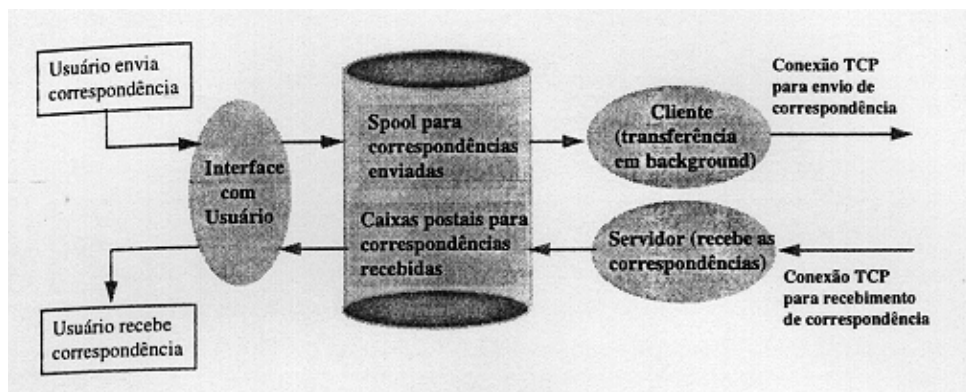


Fig. 11.14 - Arquitetura do correio eletrônico SMTP

Um usuário, ao desejar enviar uma mensagem, utiliza o módulo interface com o usuário para compor a mensagem e solicita ao sistema de correio eletrônico que a entregue ao destinatário. Ao recebê-la, o sistema armazena uma cópia da mensagem em seu spool, junto com o horário de armazenamento e a identificação do remetente e do destinatário. A transferência da mensagem é executada por um processo background, permitindo ao usuário remetente, após entregá-la ao sistema, possa executar outras aplicações. O processo de transferência, em background, mapeia o nome da máquina de destino e seu endereço IP, e tenta estabelecer uma conexão TCP com o servidor do correio eletrônico da máquina destino. Se a conexão for estabelecida, o cliente envia uma cópia da mensagem ao servidor, e este a armazena em seu spool. Havendo sucesso, o servidor avisa ao cliente que recebeu e armazenou uma cópia da mensagem. O cliente após receber a confirmação, retira sua cópia do spool local. Se não houve sucesso, o cliente anota o horário da tentativa e suspende sua execução. Periodicamente acorda, verifica se há mensagens no spool e tenta transmiti-las, se não for enviada num dado período, o serviço de correio devolve-a ao remetente, informando a não transmissão.

Ao ser conectado um usuário ao sistema, este verifica a existência de mensagens na caixa postal do usuário, se sim, envia um aviso a ele e este quando conveniente ativa o módulo de interface com usuário para as receber.

A mensagem SMTP divide-se em duas partes: cabeçalho e corpo, separados por uma linha em branco. No cabeçalho são especificadas os dados necessários a transferência da mensagem. No corpo são transportadas as informações da mensagem. O formato do corpo é livre e as mensagens são transferidas no formato texto.

Os usuários do sistema de correio eletrônico são localizados por um par de Identificadores, um para o nome da máquina de destino e o outro a caixa postal do usuário.

O SMTP especifica como o sistema de correio eletrônico transfere mensagens de uma máquina para outra. O módulo interface com o usuário e a forma como as mensagens são armazenadas não são definidos pelos SMTP.

11.3.6 - WWW (World-Wide Web)

O WWW foi desenvolvido para permitir o acesso a informações organizadas na forma de um hipertexto, que engloba documentos armazenados em servidores espalhados por toda a inter-rede.

Um programa cliente WWW executa no computador do usuário. Quando entra em execução, ele mostra um objeto WWW, normalmente um documento com texto e até imagens, onde algumas frases e imagens são destacadas. A seleção de uma área destacada (ex. por mouse), denominada âncora, faz o programa cliente buscar outro objeto WWW, possivelmente em outra máquina, o servidor WWW.

Os objetos WWW que podem ser menus, documentos (hipertextos), imagens etc são endereçados através de strings denominadas URIs (Universal Resource Identifiers). Exemplo:

`http:// info.cern.ch/hipertext/WWW/TheProject.html`

O prefixo http indica o espaço de endereçamento e define a interpretação do restante da string. No exemplo, deve ser usado o protocolo HTTP, contendo o restante da string, o endereço do servidor que deve ser contatado e um substring que deve ser entregue ao servidor para ele possa localizar um determinado objeto.

O HTTP (Hypertext Transfer Protocol) é um protocolo usado para transferência de informações no WWW, os dados transferidos podem ser do tipo texto não estruturado, hipertextos, imagens ou qualquer outro tipo de dados. É um protocolo Internet semelhante ao FTP. Utiliza uma conexão TCP para cada transferência de objeto. A conexão só é mantida enquanto dura a transferência do objeto.

A figura 11.15 mostra a Home Page WWW do Departamento de Engenharia Elétrica da UNESP em Bauru.



Fig. 11.15 - Exemplo de documento WWW

11.3.7 - SNMP (Simple Network Management Protocol)

O sistema de gerenciamento de redes da arquitetura Internet TCP/IP opera na camada de aplicação e baseia-se no protocolo SNMP (Single Network Management Protocol), sendo os padrões de estrutura definidos nos documentos RFCs 1155, 1156 e 1157.

Como na esquema de gerenciamento da OSI, os processos que implementam as funções de gerenciamento Internet atuam como agentes ou gerentes. Os agentes coletam junto aos objetos gerenciados as informações relevantes para o gerenciamento da rede. O gerente processa as informações recolhidas pelos clientes, objetivando detectar a presença de falhas no funcionamento dos componentes da rede (hosts, gateways, processos executando os protocolos de comunicação etc.) para tomar providências para contornar os problemas oriundos de falhas. O gerente envia comandos aos agentes, solicitando uma leitura no valor das variáveis dos objetos gerenciados (get e response), ou modificando seu valor (put). Na troca de informações entre o gerente e o agente, são aplicados mecanismos de autenticação para evitar que usuários não autorizados interfiram no funcionamento da rede. A troca de mensagens entre o gerente e o agente é definida pelo protocolo SNMP, que define o formato e a ordem que deve ser seguida no intercâmbio de informações de gerenciamento.

As informações sobre os objetos gerenciados são armazenadas na MIB (Management Information Base), que contém informações sobre hosts, gateways e processos que executam os protocolos de comunicação. A MIB é especificada em ASN.1.

A figura 11.16 representa o modelo de gerenciamento SNMP.

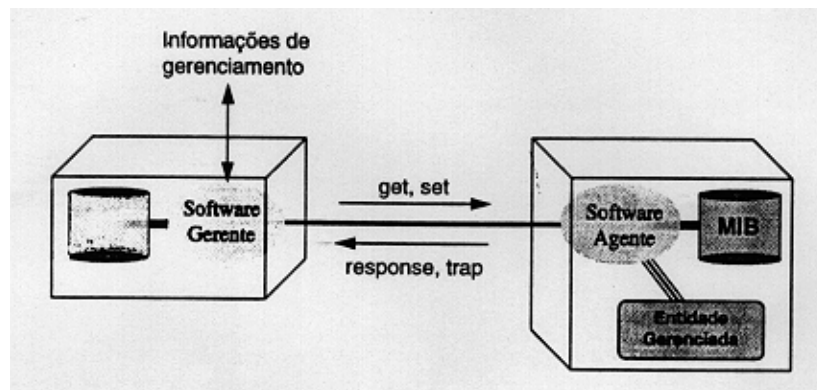


Fig. 11.16 - Modelo de gerenciamento SNMP

<http://www.dee.feb.unesp.br/redes/96/96cap1.html>

<http://www.dee.feb.unesp.br/redes/96/96cap2.html>

<http://www.dee.feb.unesp.br/redes/96/96cap3.html>

<http://www.dee.feb.unesp.br/redes/96/96cap4.html>

<http://www.dee.feb.unesp.br/redes/96/96cap5.html>

<http://www.dee.feb.unesp.br/redes/96/96cap6.html>

<http://www.dee.feb.unesp.br/redes/96/96cap7.html>

<http://www.dee.feb.unesp.br/redes/96/96cap8.html>

<http://www.dee.feb.unesp.br/redes/96/96cap9.html>

<http://www.dee.feb.unesp.br/redes/96/96cap10.html>

<http://www.dee.feb.unesp.br/redes/96/96cap11.html>