

***Engenharia de Controle e Automação***  
***Disciplina: Redes Industriais - 7º Período***  
***Professor: José Maurício S. Pinheiro***

***AULA 2: Modelo de Referência - OSI***

O final da década de 1970 apresentava uma perspectiva de crescimento em termos de comunicação de dados devido ao investimento e desenvolvimento de novos sistemas. Por outro lado, havia uma tendência que poderia acarretar uma crise no setor, a heterogeneidade de padrões entre os fabricantes, praticamente impossibilitando a interconexão entre sistemas de fabricantes distintos. Os fabricantes então começaram a perseguir alguns objetivos necessários para a construção de um sistema aberto. Esses objetivos foram:

- **Interoperabilidade:** capacidade que sistemas abertos possuem de troca de informações entre eles, mesmo que sejam fornecidos por fabricantes diversos;
- **Interconectividade:** é a maneira pela qual os computadores de fabricantes distintos podem ser conectados;
- **Portabilidade da aplicação:** é a capacidade de um software de rodar em várias plataformas diferentes;
- **Escalabilidade:** capacidade de um software rodar com desempenho aceitável em computadores de capacidades diversas, desde computadores pessoais até supercomputadores.

**Padronização**

Um padrão é um conjunto de normas e procedimentos. O cumprimento destas normas e procedimentos pode ser obrigatório (normalmente quando relacionados à segurança do homem) ou recomendável (quando relacionados à qualidade de produtos e serviços). Padrões visam homogeneizar produtos e serviços com níveis aceitáveis de qualidade e segurança, minimizar investimentos em estoques, compatibilizar equipamentos de diferentes procedências, etc.

Um padrão é dito “de facto” quando foi adotado sem nenhuma ação de entidade reguladora, por exemplo, IBM-PC. Por outro lado, padrões “de jure” são produzidos por entidades reguladoras, nacionais ou internacionais, governamentais ou não, por exemplo, ISO-9000. Assim, a ISO (*International Organization for Standardization*) passou a se ocupar em criar um padrão de arquitetura aberta baseada em camadas.

Foi então definido o *Modelo de Referência para Interconexão de Sistemas Abertos* (*Reference Model for Open Systems Interconnection - RM OSI*). Entretanto, somente em 1983 a ISO internacionalizou a padronização do RM-OSI (documento ISO 7498), juntamente com o CCITT (“Comité Consultatif International Télégraphique et Téléphonique”) através de sua recomendação X.200.

## 1. Serviços e Protocolos

### 1.1. Serviço

Um serviço é um conjunto de primitivas que uma camada oferece à camada superior seguinte, ou seja, um serviço funciona como uma interface entre duas camadas, onde a inferior age como provedora do serviço e a superior como a usuária do serviço. O serviço determina as operações que a camada está apta a realizar, porém, não tem vínculo nenhum com o modo como isso deve ser feito.

Os serviços oferecidos por uma determinada camada podem ser utilizados por uma camada seguinte acima através de interfaces lógicas, conhecidas como SAP's (*Service Access Point*). A troca de informações entre camadas é feita por uma associação chamada conexão.

Os SAP's são interfaces lógicas entre as entidades (N) e (N+1). Portanto, quando a entidade (N+1) precisa utilizar o serviço provido pela camada (N), ela busca este no SAP(N). As informações entre entidades (N+1) são trocadas através de uma *conexão* (N), estabelecida na camada (N) usando o protocolo (N).

### 1.2. Protocolo

Para que os pacotes de dados trafeguem de uma origem até um destino, através de uma rede, é importante que todos os dispositivos da rede usem a mesma linguagem, ou protocolo. Um protocolo é uma coleção de regras que rege o formato e significado dos dados, pacotes ou mensagens trocadas entre componentes de uma mesma camada. Os protocolos não são visíveis ao usuário, sendo que os mesmos podem ser alterados, porém, para isso, os serviços devem continuar sendo os mesmos.

## 2. Arquitetura Aberta e o Modelo em Camadas

A utilização de um ambiente de arquitetura aberta nos oferece algumas vantagens:

- Liberdade de escolha entre soluções de diversos fabricantes;
- Acesso mais rápido a novas tecnologias e a preços mais acessíveis;
- Redução de investimentos em novas máquinas, já que os sistemas e os softwares de aplicação são portáteis para os vários tipos de máquinas existentes.

A adoção de um modelo baseado em camadas também não é arbitrária. Considerando que uma rede de computadores tem como objetivo o processamento de tarefas distribuídas pela rede de forma harmônica e cooperativa entre os vários processos de aplicação, o projeto desta deve levar em conta vários fatores como:

- Considerar todos os eventos possíveis de acontecer durante a comunicação;

- Conhecer todos os efeitos e causas destes eventos;
- Especificar em detalhes todos os aspectos técnico-operacionais dos meios físicos a serem utilizados como suporte à comunicação;
- Detalhes das próprias aplicações a serem executadas.

O conjunto de camadas é hierárquico, ou seja, cada camada baseia-se na camada inferior. O projeto de uma camada é restrito ao contexto dessa camada e supõe que problemas fora deste contexto já estejam devidamente resolvidos. Na realidade existem duas vantagens práticas na utilização de uma arquitetura em camadas. Em primeiro lugar, a complexidade do esforço global de desenvolvimento é reduzida através de *abstrações* (não interessa para uma determinada camada como as demais possibilitam o fornecimento de seus serviços, só o que elas oferecem). Na arquitetura hierárquica, a camada (N) sabe apenas que existe a camada (N-1), prestadora de determinados serviços e a camada (N+1), que lhe requisita os serviços. A camada (N) não toma conhecimento da existência das camadas (N± 2), (N± 3), etc.

O segundo aspecto é relacionado com a *independência* entre as camadas. A camada (N) preocupa-se apenas em utilizar os serviços da camada (N-1), independentemente do seu protocolo. É assim que uma camada pode ser alterada sem mudar as demais (facilidade de manutenção) - desde que os serviços que ela presta não sejam modificados. É assim também que novas aplicações podem ser executadas, na camada apropriada, aproveitando os mesmos serviços já fornecidos pelas outras camadas (redução dos esforços para evoluções).

O modelo OSI da ISO define apenas a arquitetura do sistema. O padrão criado para o modelo OSI, então, define exatamente o que cada camada deve fazer, mas não define como isto será feito, ou seja, define os serviços que cada camada deve prestar, mas não o protocolo que o realizará. O modelo OSI se encaixa como um conjunto de funções que possibilitam que máquinas distintas possam se comunicar e trocar informações. Ele possui sete camadas onde cada camada é responsável por uma determinada função específica.

Os princípios utilizados para se chegar a estas camadas são:

- Uma camada deve ser criada onde é necessário um nível de abstração diferente;
- Cada camada deve desempenhar uma função bem definida;
- A função de cada camada deve ser definida tendo em vista protocolos e padrões internacionais;
- As fronteiras entre as camadas devem ser escolhidas de forma a minimizar o fluxo de informações através das interfaces;
- O número de camadas deve ser suficiente para que não seja preciso agrupar funções em uma mesma camada por necessidade e ser suficiente para que a arquitetura fique manejável.
- Cada camada é usuária dos serviços prestados pela camada imediatamente inferior e presta serviços para a camada imediatamente superior. Esta troca de informações entre as camadas adjacentes ocorre por meio da troca de primitivas de serviços nas interfaces entre as camadas.

A conversação entre duas pessoas é um bom exemplo de abordagem em camadas para analisar o fluxo de informações. Em uma conversação, cada pessoa que deseja comunicar-se começa por criar uma ideia. Em seguida deve-se tomar uma decisão de como comunicar a ideia de maneira correta. Por exemplo, uma pessoa poderia decidir falar, cantar ou gritar, e qual idioma usar. Finalmente a ideia seria entregue. Por exemplo, a pessoa cria o som que transporta a mensagem.

Este processo pode ser dividido em camadas separadas que podem ser aplicadas a todas as conversações. A camada superior é a ideia que será comunicada. A camada do meio é a decisão de como será comunicada a ideia. A camada inferior é a criação do som para transportar a comunicação.

O mesmo método de dividir uma tarefa em camadas explica como uma rede de computador distribui informações a partir de uma fonte até o seu destino. Quando os computadores enviam informações através de redes, todas as comunicações têm origem na fonte e depois trafegam até um destino.

A informação que navega pela rede é conhecida como dado ou pacote. Um pacote é uma unidade de informações logicamente agrupadas que se desloca entre sistemas de computadores (Figura 1). Conforme os dados são passados entre as camadas, cada camada acrescenta informações adicionais que possibilitam uma comunicação efetiva com a camada correspondente no outro computador.

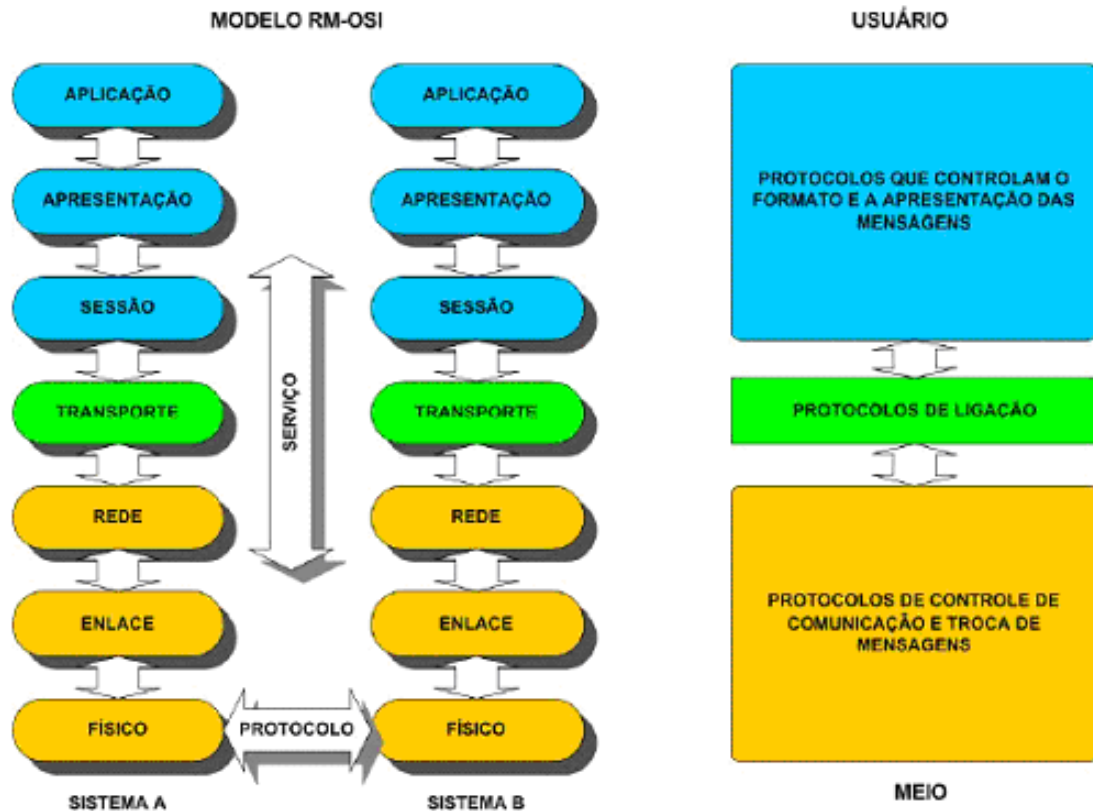


**Figura 1 - Transporte de pacote de dados desde a origem até o destino**

### **3. As Camadas do Modelo OSI**

Apesar de o modelo OSI ser dividido em sete camadas (Figura 2), pode-se considerar genericamente que as três camadas mais baixas cuidam dos aspectos relacionados à transmissão propriamente dita e a camada de transporte lida com a comunicação fim-a-fim, enquanto que as três camadas superiores se preocupam com os aspectos relacionados à aplicação, já ao nível de usuário.

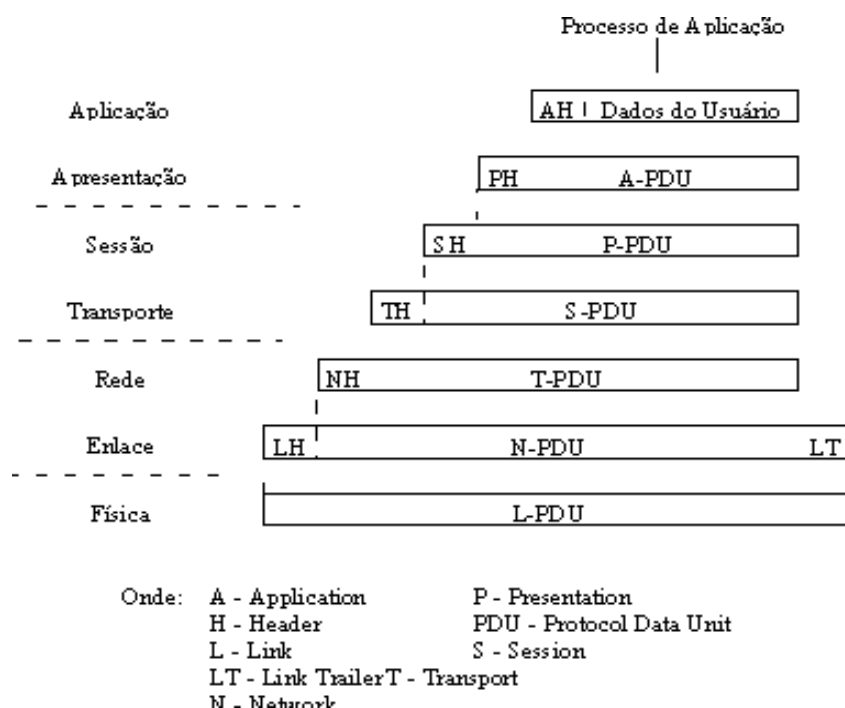
Devemos lembrar que ao se falar em serviços, estamos falando em camadas adjacentes (níveis diferentes, no mesmo sistema), e protocolo falamos de entidades pares (no mesmo nível, em sistemas diferentes). A comunicação entre sistemas ocorre ao nível de camadas, ou seja, a camada de aplicação do sistema A se comunica com a camada de aplicação do sistema B e assim por diante até o nível físico, onde ocorre a comunicação física entre os sistemas.



**Figura 2 - Modelo RM - OSI**

Uma maneira fácil e simplista de se enxergar a funcionalidade de um modelo em camadas, como o modelo OSI, é imaginar que cada camada tem como função adicionar um cabeçalho aos dados do usuário a serem transmitidos para outro sistema. Deste modo a função de cada camada do outro sistema é exatamente a inversa, ou seja, retirar os cabeçalhos dos dados que chegam e entregá-los ao usuário em sua forma original.

Conforme mostra a Figura 3, os dados entregues pelo usuário à camada de aplicação do sistema recebem a denominação de SDU (Service Data Unit). A camada de aplicação, então, junta à SDU (no caso, os dados do usuário) um cabeçalho chamado PCI (Protocol Control Information). O objeto resultante desta junção é chamado de PDU (Protocol Data Unit), que corresponde à unidade de dados especificada de certo protocolo da camada em questão.



**Figura 3 – Exemplo de funcionalidade do Modelo OSI**

### 3.1. Camada Física ou de Acesso ao Meio

A Camada Física é responsável pela interface física entre os equipamentos e os protocolos a serem seguidos para a transmissão das informações entre os diversos sistemas de informação e gerencia a transferência física da informação sobre os meios de transmissão possíveis. Com esta camada, o modelo OSI permite a flexibilidade do uso de vários meios físicos para interconexão, com procedimentos de controle diferentes.

Ela envolve a definição de todos os aspectos relacionados com as características mecânicas (propriedades físicas da interface com o meio físico de transmissão, incluindo, por exemplo, o tipo de conector utilizado), elétricas (representação de um bit em termos de, por exemplo, nível de tensão utilizado e taxa de transmissão de bits), funcionais (definem as funções a serem executadas por esta interface) e procedurais (especificam a sequência de eventos trocados durante a transmissão de uma série de bits através do meio de transmissão).

### 3.2. Camada de Enlace de Dados ou de Ligação Lógica

A Camada de Enlace gerencia a transferência da informação através do canal de transmissão. É responsável pelo controle do fluxo da informação na rede, bem como sua sincronização e garantia de entrega. Proporciona ainda, a detecção e correção de erros de transmissão.

Sua principal função é tornar o meio físico mais confiável e isento de erros para as camadas superiores, fornecendo mecanismos para ativar, manter e desativar a conexão. Para cumprir sua função, nesta camada são implementados instrumentos de controle e detecção de erros. Nesta camada,



os bits de informação são agrupados em unidades chamadas "frames". O *Institute of Electrical and Electronic Engineers* (IEEE) divide a camada de Enlace de Dados em duas subcamadas: LLC e MAC.

- A subcamada LLC é responsável por fornecer às camadas superiores um meio de transmissão que pareça livre de erros. Sua implementação independe da subcamada MAC.
- Na subcamada MAC estão os métodos de acesso com os endereços de hardware de cada sistema, também conhecidos como MAC Address e o CRC. Um dos seus objetivos é ocultar das camadas superiores o tipo de meio físico que está sendo usado, assim como o método de acesso.

No processo de transmissão, esta camada acrescenta suas próprias informações de controle como o endereço de origem e de destino, o comprimento do frame, os protocolos das camadas superiores envolvidos na comunicação e um protocolo responsável pela verificação de erros conhecido como "Cyclic Redundancy Check" (CRC).

### **3.3. Camada de Rede**

A camada de rede é responsável pelo gerenciamento das informações entre uma rede composta de múltiplos segmentos. Proporciona o encaminhamento e endereçamento da informação, quer na origem, quer no destinatário da transmissão. Na arquitetura de sistemas abertos, alguns sistemas são destinatários terminais de dados, enquanto outros funcionam apenas como nós intermediários que repassam as informações para outros sistemas.

A camada de rede tem como função favorecer uma trajetória de conexão de rede entre um par de entidades da camada de transporte, inclusive passando por nós intermediários. É o protocolo voltado para a operação da rede propriamente dita. Algoritmos de roteamento e de controle de congestionamento são agrupados nessa camada.

### **3.4. Camada de Transporte**

Proporciona a interface entre as três camadas superiores e as três camadas inferiores, isolando o utilizador dos aspectos funcionais e físicos da rede. Garante ainda, a comunicação ponto-a-ponto, define e controla a qualidade da transmissão. Há uma camada de transporte sobre a camada de rede para aliviar as entidades de camadas superiores das tarefas do transporte de dados entre elas.

O propósito principal da camada de transporte é oferecer serviço de transferência de dados de forma transparente entre as entidades da camada de sessão. O termo "transparente" refere-se ao fato de que as entidades de sessão não têm a necessidade de conhecer os detalhes da transferência dos dados. Os usuários da camada de transporte são identificados pelos seus endereços.

A Camada de Transporte fornece uma comunicação ponto a ponto confiável e transparente, através de mecanismos de seqüenciamento, controle de fluxo e

confirmação / negação do recebimento de pacotes. Normalmente é usada para compensar a falta de confiabilidade das camadas inferiores.

### **3.5. Camada de Sessão**

A Camada de Sessão fornece uma estrutura de controle para a comunicação entre aplicações. Cuida do mecanismo conhecido como "Name-to-station Address Translation" (NAT), ou seja, a tradução de endereços para o nome de uma estação de rede específica. Gerencia a transferência organizada da informação, desde o modo como se processa o diálogo até o gerenciamento da troca de dados entre as entidades de apresentação. Para isso, a camada de sessão fornece serviços para o estabelecimento de uma conexão de sessão entre duas entidades de apresentação através do uso de uma conexão de transporte.

A camada de sessão tem como serviços a administração da sessão (login / autenticação e logoff) e o diálogo da sessão, controlando a troca de dados, delimitando e sincronizando operações entre duas entidades.

### **3.6. Camada de Apresentação**

A Camada de Apresentação realiza a conversão do formato de dados de forma que eles sejam entendidos por todos os sistemas envolvidos na comunicação. Esta camada também faz a compressão / descompressão e criptografia / descryptografia. Esta camada também resolve problemas de diferenças de sintaxe entre sistemas abertos comunicantes.

Através dos serviços da camada de apresentação, as aplicações no ambiente OSI podem estabelecer a comunicação sem custos excessivos oriundos de variações de interfaces, transformações ou modificações das próprias aplicações.

### **3.7. Camada de Aplicação**

Na Camada de Aplicação está o suporte das aplicações do usuário do sistema. Sua função é definir a semântica da informação a transmitir/receber. Os serviços desta camada são usados pelos próprios usuários do ambiente OSI. Essa camada serve de "janela" entre usuários comunicantes, através da qual ocorre a troca das informações entre esses usuários. Cada usuário é representado para os demais por sua entidade de aplicação devida. É importante salientar que a totalidade de uma aplicação não se encontra nesta camada, apenas uma parte, que precisa se comunicar com entidades remotas. É a responsável ainda pela negociação do modo como a informação a transmitir é representada (sintaxe). A camada de apresentação fornece os serviços que podem ser selecionados pela camada de aplicação para a interpretação da sintaxe dos dados trocados. Gerencia a entrada, troca, amostra e controle de dados estruturados. Sob a responsabilidade da Camada de Aplicação estão os serviços como transferência de arquivos, gerenciamento de redes e emulação de terminais.



Na Figura 4 temos exemplos de dispositivos de redes e sua “posição” dentro do modelo OSI.

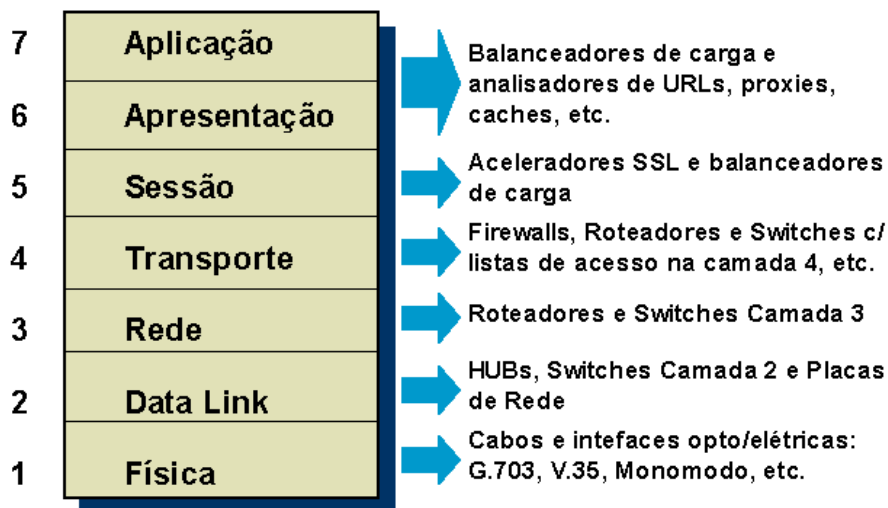


Figura 4 - O modelo OSI e a relação com os dispositivos de rede

#### 4. Processo de encapsulamento

Todas as comunicações numa rede começam em uma origem e são enviadas a um destino. As informações enviadas através da rede são conhecidas como dados ou pacotes de dados. Se um computador (host A) desejar enviar dados para outro computador (host B), os dados devem primeiro ser empacotados através de um processo chamado encapsulamento (Figura 5).

O encapsulamento empacota as informações de protocolo necessárias antes que trafeguem pela rede. Assim, à medida que o pacote de dados desce pelas camadas do modelo OSI, ele recebe cabeçalhos, trailers e outras informações.

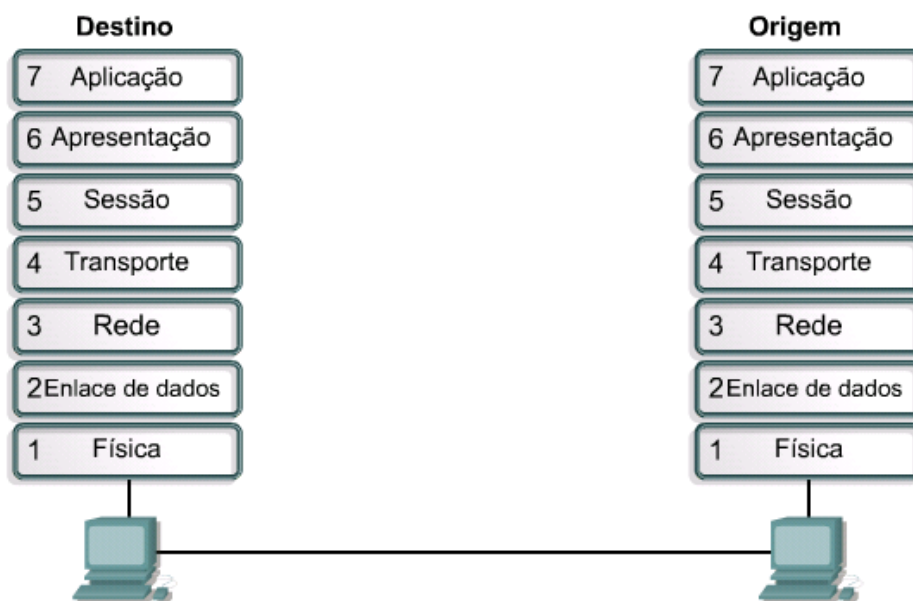
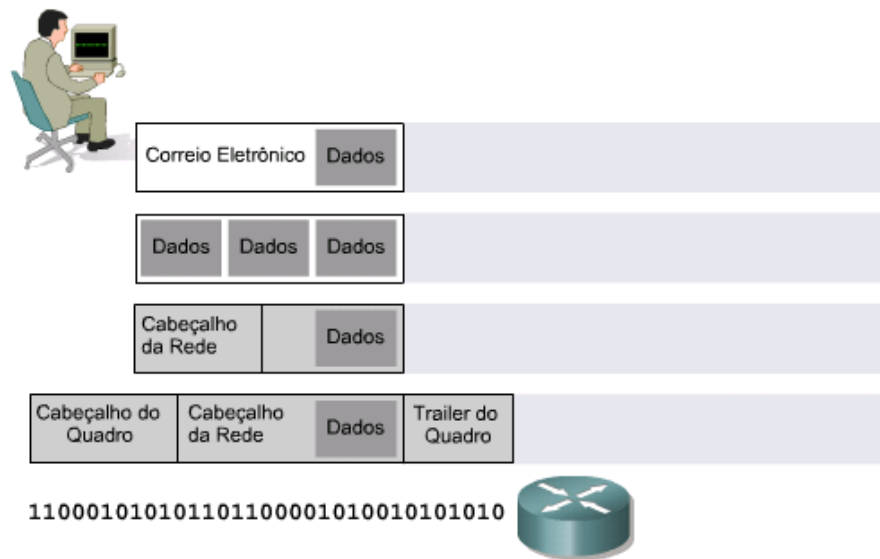


Figura 5 – Processo de Encapsulamento

Uma vez que os dados são enviados pela origem, eles viajam através da camada de aplicação em direção às outras camadas. O empacotamento e o fluxo dos dados que são trocados passam por alterações à medida que as camadas executam seus serviços para os usuários finais. Na Figura 6 é mostrado que as redes devem efetuar as cinco etapas de conversão a seguir para encapsular os dados:



**Figura 6 - Etapas de Conversão dos dados**

- Gerar os dados - Quando um usuário envia uma mensagem de correio eletrônico, os seus caracteres alfanuméricos são convertidos em dados que podem trafegar na rede;
- Empacotar os dados para transporte fim-a-fim - Os dados são empacotados para transporte na rede. Usando segmentos, a função de transporte assegura que os hosts da mensagem em ambas as extremidades do sistema de correio eletrônico possam comunicar-se com confiabilidade.
- Adicionar o endereço IP da rede ao cabeçalho - Os dados são colocados em um pacote ou datagrama que contém um cabeçalho de pacote contendo endereços lógicos de origem e destino. Esses endereços ajudam os dispositivos da rede a enviar os pacotes através da rede por um caminho escolhido.
- Adicionar o cabeçalho e o trailer da camada de enlace de dados - Cada dispositivo da rede deve colocar o pacote dentro de um quadro. O quadro permite a conexão com o próximo dispositivo da rede diretamente conectado no link. Cada dispositivo no caminho de rede escolhido requer enquadramento de forma que possa conectar-se com o próximo dispositivo.
- Converter em bits para transmissão - O quadro deve ser convertido em um padrão de 1s e 0s (bits) para transmissão no meio físico. Uma

função de sincronização de relógio (clock) permite que os dispositivos diferenciem esses bits à medida que trafegam no meio físico. O meio físico das redes interconectadas pode variar ao longo do caminho usado. Por exemplo, a mensagem de correio eletrônico pode ser originada em uma rede local, atravessar um backbone do campus e sair por um link da WAN até alcançar seu destino em outra rede local remota.

## 5. Primitivas de Serviços

Além de um meio de comunicação de dados entre os computadores, uma série de outros requisitos é necessária: 1) o sistema origem da informação deve indicar à rede a identificação do destino e certificar-se que o destino está preparado, ou tem interesse, em receber os dados; 2) o formato da informação enviada pode ser incompatível com o formato da informação esperada pelo destino, levando à necessidade de uma translação de formato; 3) em geral, o envio da informação por parte da origem provoca uma reação do lado do destino, implicando em um relacionamento, ou diálogo, entre a origem e o destino.

As primitivas de serviços são informações trocadas entre duas camadas adjacentes de forma a realizar um serviço. No modelo OSI são definidos quatro tipos de primitivas:

- **Requisição (Request):** utilizada para solicitar ou ativar um determinado serviço;
- **Indicação (Indication):** informa a ocorrência de um determinado evento;
- **Resposta (Response):** utilizada para responder a um determinado evento;
- **Confirmação (Confirmation):** utilizada para confirmar a execução de um serviço solicitado.

As primitivas possuem parâmetros de entrada e saída. Por exemplo, em um pedido de conexão, os parâmetros podem especificar a máquina à qual se conectar, o tipo de serviço desejado e o tamanho máximo de mensagem a ser utilizada e os parâmetros em uma indicação de conexão podem conter a identidade do solicitante, o tipo de serviço e o tamanho máximo de mensagem proposto. Quem cuida dos detalhes desta negociação é o protocolo.

Os serviços prestados podem ser basicamente de dois tipos: confirmado e não-confirmado. No serviço confirmado, há um *pedido*, uma *indicação*, uma *resposta* e uma *confirmação*. Já no serviço não-confirmado, há apenas um *pedido* e uma *indicação*. Um exemplo de um serviço confirmado é o estabelecimento de uma conexão, enquanto que a desconexão é um serviço não-confirmado. Vejamos o exemplo de um serviço de conexão na Figura 7:

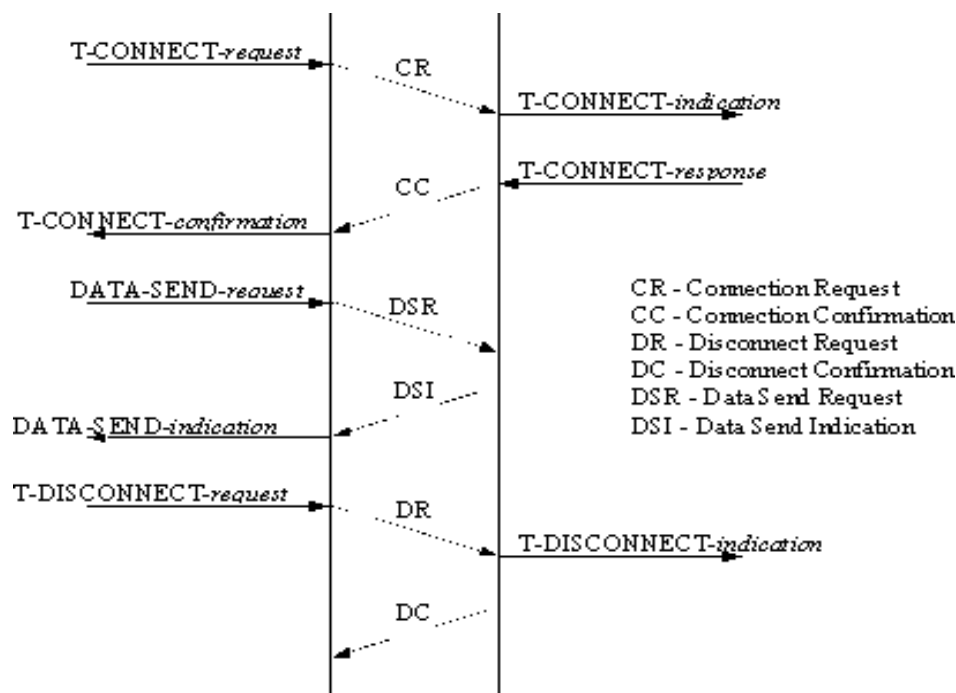


Figura 7 - Diagrama de Tempo de Estabelecimento de Conexão

Este serviço pode ser descrito da seguinte forma:

*Requisição - request.- CONEXÃO* - solicita o estabelecimento de uma conexão;  
*Indicação - indication.- CONEXÃO* - informa à parte chamada;  
*Resposta - response.- CONEXÃO* - entidade chamada aceita ou rejeita chamadas;  
*Confirmação - confirmation.- CONEXÃO* - indica ao solicitante se a chamada foi aceita;  
*Requisição - request.- DADOS* - solicita a transmissão de dados;  
*Indicação - indication.- DADOS* - avisa sobre a chegada de dados;  
*Requisição - request.- DESCONEXÃO* - solicita que a conexão seja liberada;  
*Indicação - indication - DESCONEXÃO* - informa ao parceiro sobre o pedido.

Um exemplo didático do processo é a analogia com uma ligação telefônica:

*request.CONEXÃO* – você disca o telefone da pessoa;  
*indication.CONEXÃO* - o telefone dela toca;  
*response.CONEXÃO* - ela atende o telefone;  
*confirmation.CONEXÃO* - você ouve o sinal de chamada parar de tocar;  
*request.DADOS* - você convida a pessoa para sair;  
*indication.DADOS* - ela ouve seu convite;  
*request.DADOS* - ela responde que sim;  
*indication.DADOS* - você ouve a aceitação dela;  
*request.DESCONEXÃO* - você desliga o telefone;  
*indication.DESCONEXÃO* - ela ouve e desliga também.

## **Exercícios**

1. O que é o modelo de referência OSI?
2. O que é um sistema aberto?
3. Como pode ser definido um protocolo de comunicação de dados?
4. Quais são as camadas do modelo de referência OSI?
5. Qual é a função da camada de rede?
6. Cite três protocolos da camada de aplicação.
7. A que camadas do modelo de referência OSI estão associados os conceitos de frame e pacote?