

# **Protocolos Modelo OSI**

**AULA 08**

**Prof. Carlos Louzada**

# Protocolo

- Protocolo é a “linguagem” usada para computadores “conversarem” em uma rede.
- Por exemplo: através de um protocolo o transmissor pode informar ao receptor que tipo de dados ele está enviando.
- Na realidade em uma comunicação dentre dois computadores vários protocolos são usados, cada um com uma finalidade diferente.

# Modelo OSI

- O **Modelo OSI** (*Open System Interconnection*) é um modelo de rede de computador referência da ISO dividido em camadas de funções, criado em 1971 e formalizado em 1983, com objetivo de ser um padrão, para protocolos de comunicação entre os mais diversos sistemas em uma rede local (Ethernet), garantindo a comunicação entre dois sistemas computacionais (*end-to-end*).
- Este modelo divide as redes de computadores em **7 camadas**, de forma a se obter camadas de abstração. Cada protocolo implementa uma funcionalidade assinalada a uma determinada camada.

- Segundo Tanenbaum o Modelo OSI não é uma arquitetura de redes, pois não especifica os serviços e protocolos exatos que devem ser usados em cada camada. Ele apenas informa o que cada camada deve fazer.
- O Modelo OSI permite comunicação entre máquinas heterogêneas e define diretivas genéricas para a construção de redes de computadores (seja de curta, média ou longa distância) independente da tecnologia utilizada.

# História

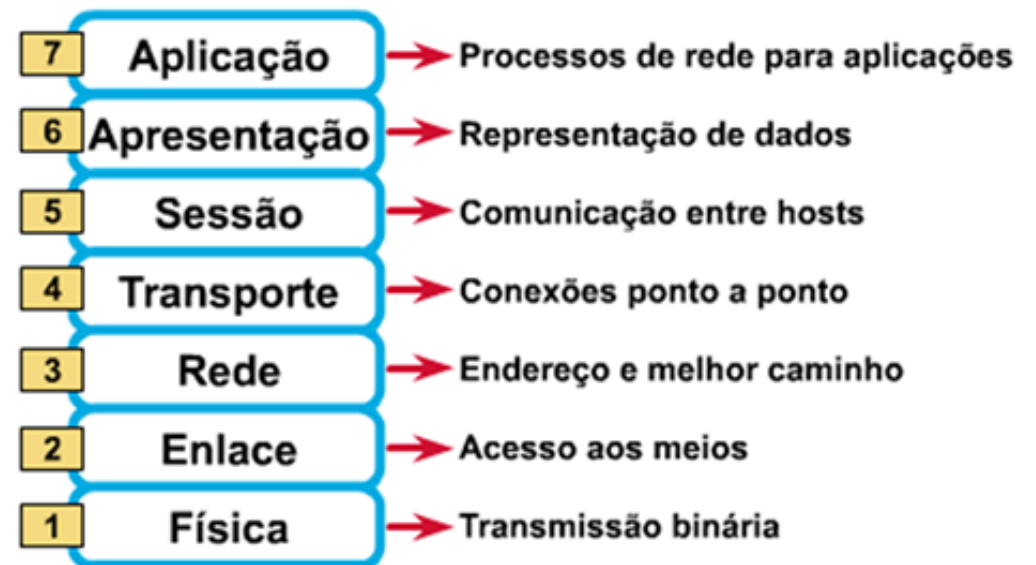
- A Organização Internacional para a Normalização (*International Organization for Standardization* - ISO), foi uma das primeiras organizações a definir formalmente uma arquitetura padrão com objetivo de facilitar o processo de interconectividade entre máquinas de diferentes fabricantes;
- Em 1984 a ISO lançou o padrão chamado Interconexão de Sistemas Abertos (*Open Systems Interconnection* - OSI) ou Modelo OSI.

- A Organização Internacional para a Normalização (ISO) começou a desenvolver a sua estrutura de arquitetura OSI, com quatro componentes principais: um modelo abstrato de rede, o chamado Modelo de Referência Básico ou sete camadas do modelo, e um conjunto de protocolos específicos e outros dois de menor relevância.
- O conceito de um modelo de sete camadas foi fornecida pelo trabalho de Charles Bachman, Serviços de Informação da Honeywell. Vários aspectos do projeto OSI evoluíram a partir de experiências com a ARPANET, a Internet incipiente, NPLNET, EIN, CYCLADES rede e o trabalho em IFIP WG6.1.

- Quando as redes de computadores surgiram, as soluções eram, na maioria das vezes, proprietárias, isto é, uma determinada tecnologia só era suportada por seu fabricante.
- Dessa forma, um mesmo fabricante era responsável por tudo na rede: cabos, placas, equipamentos, protocolos, software, etc.
- O modelo OSI é um modelo de 7 camadas. Importante notar que este modelo foi criado especificamente para redes com comutação de pacotes e não para redes com comutação de circuito.
- O estudo desse protocolo é extremamente didático, pois através dele há como entender como deveria ser um “protocolo ideal”, bem como facilita enormemente a comparação do funcionamento de protocolos criados por diferentes fabricantes.

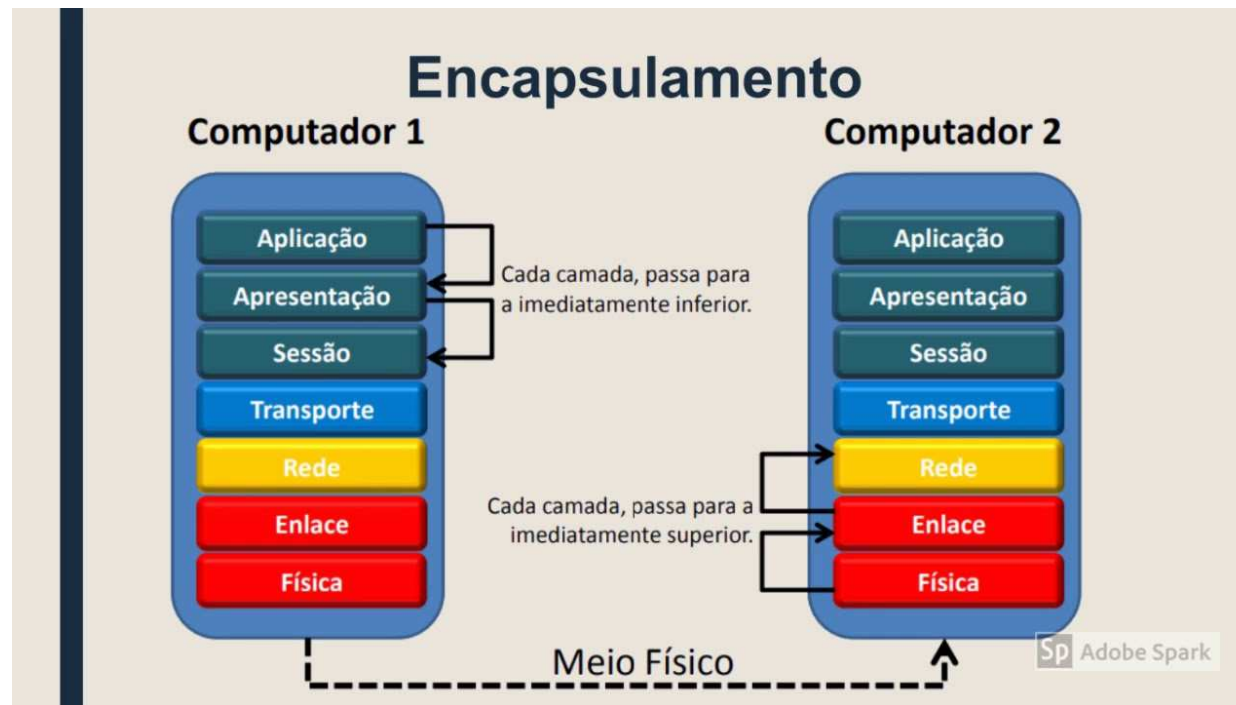


## Funções das Camadas





- Na transmissão de um dado, cada camada pega as informações passadas pela camada superior, acrescenta informações pelas quais ela seja responsável e passa os dados para a camada imediatamente inferior.
- Esse processo é conhecido como *encapsulamento*.



# Camada 7 - Aplicação

- A camada de aplicação corresponde às aplicações (programas) no topo da camada OSI que serão utilizadas para promover uma interação entre a máquina-usuário (máquina destinatária e o usuário da aplicação).
- Esta camada também disponibiliza os recursos (protocolo) para que tal comunicação aconteça, por exemplo, ao solicitar a recepção de *e-mail* através do aplicativo de *e-mail*, este entrará em contato com a camada de Aplicação do protocolo de rede efetuando tal solicitação (POP3 ou IMAP).
- Tudo nesta camada é relacionado ao software. Alguns protocolos utilizados nesta camada são: HTTP, SMTP, FTP, NFS, Telnet, SIP, RDP, IRC, SNMP, NNTP, POP3, IMAP, BitTorrent, DNS, ICMP, SMTP etc.

# Camada 6 - Apresentação

- A camada de Apresentação, também chamada camada de **Tradução**, converte o formato do dado recebido pela camada de Aplicação em um formato comum a ser usado na transmissão desse dado, ou seja, um formato entendido pelo protocolo usado.
- Um exemplo comum é a conversão do padrão de caracteres (código de página) quando o dispositivo transmissor usa um padrão diferente do ASCII. Pode ter outros usos, como **compressão de dados** e **criptografia**.

- Os dados recebidos da camada 7 estão descomprimidos, e a camada 6 do dispositivo transmissor fica responsável por comprimir esses dados.
- A transmissão dos dados torna-se mais rápida, já que haverá menos dados a serem transmitidos: os dados recebidos da camada 4 foram "encolhidos" e enviados à camada 1.
- Para aumentar a segurança, pode-se usar algum esquema de criptografia neste nível, sendo que os dados só serão decodificados na camada 6 do dispositivo receptor.
- Ela trabalha transformando os dados em um formato no qual a camada de aplicação possa aceitar, minimizando todo tipo de interferência.
- Um exemplo de protocolo que opera nesta camada é o SSL (*Secure Socket Layer*), responsável por criptografar dados gerados por protocolos operando na camada de aplicação.

## Camada 5 - Sessão

- Responsável pela troca de dados e a comunicação entre hosts, a camada de Sessão permite que duas aplicações em computadores diferentes estabeleçam uma comunicação, definindo como será feita a transmissão de dados, pondo marcações nos dados que serão transmitidos.
- Se porventura a rede falhar, os computadores reiniciam a transmissão dos dados a partir da última marcação recebida pelo computador receptor.

# Camada 4 - Transporte

- A camada de transporte é responsável por receber os dados enviados pela camada de sessão e segmentá-los para que sejam enviados a camada de rede, que por sua vez, transforma esses **segmentos** em **pacotes**.
- No receptor, a camada de Transporte realiza o processo inverso, ou seja, recebe os pacotes da camada de rede e junta os segmentos para enviar à camada de sessão.
- Isso inclui controle de fluxo, ordenação dos pacotes e a correção de erros, tipicamente enviando para o transmissor uma informação de recebimento, garantindo que as mensagens sejam entregues sem erros na sequência, sem perdas e duplicações.

- A camada de transporte separa as camadas de nível de aplicação (camadas 5 a 7) das camadas de nível físico (camadas de 1 a 3).
- A camada 4, Transporte, faz a ligação entre esses dois grupos e determina a classe de serviço necessária como orientada à conexão, com controle de erro e serviço de confirmação ou sem conexões e nem confiabilidade.
- O objetivo final da camada de transporte é proporcionar serviço eficiente, confiável e de baixo custo.
- O hardware e/ou software dentro da camada de transporte e que faz o serviço é denominado entidade de transporte.

- A entidade de transporte comunica-se com seus usuários através de primitivas de serviço trocadas em um ou mais TSAP (Transport Service Access Point), que são definidas de acordo com o tipo de serviço prestado: **orientado** ou **não à conexão**.
- Estas primitivas são transportadas pelas TPDU (*Transport Protocol Data Unit*).
- Na realidade, uma entidade de transporte poderia estar simultaneamente associada a vários TSA e NSAP (*Network Service Access Point black*).
- No caso de multiplexação, associada a vários TSAP e a um NSAP e no caso de *splitting*, associada a um TSAP e a vários NSAP.



## A ISO define o protocolo de transporte para operar em dois modos:

1. Orientado à conexão;
  2. Não-Orientado à conexão.
- Como exemplo de protocolo orientado à conexão, temos o TCP, e de protocolo não orientado à conexão, temos o UDP.
  - É óbvio que o protocolo de transporte não orientado à conexão é menos confiável.
  - Ele não garante - entre outras coisas - a entrega das TPDU, nem tão pouco a ordenação delas.
  - Entretanto, onde o serviço da camada de rede e das outras camadas inferiores é bastante confiável - como em redes locais - o protocolo de transporte não orientado à conexão pode ser utilizado, sem o *overhead* inerente a uma operação orientada à conexão.

- O serviço de transporte baseado em conexões é semelhante ao serviço de rede baseado em conexões.
- O endereçamento e controle de fluxo também são semelhantes em ambas as camadas.
- Para completar, o serviço de transporte sem conexões também é muito semelhante ao serviço de rede sem conexões.
- Constatado os fatos acima, surge a seguinte questão: "Por que termos duas camadas e não uma apenas?".
- A camada de rede é parte da sub-rede de comunicações e é executada pela concessionária que fornece o serviço (pelo menos para as WAN).
- Quando a camada de rede não fornece um serviço confiável, a camada de transporte assume as responsabilidades, melhorando em suma importância a qualidade do serviço.

## Camada 3 - Rede

- A camada de rede fornece os meios funcionais e de procedimento de transferência de comprimento variável de dados de sequências de uma fonte de acolhimento de uma rede para um host de destino numa rede diferente (em contraste com a camada de ligação de dados que liga os hosts dentro da mesma rede), enquanto se mantém a qualidade de serviço requerido pela camada de transporte.
- A camada de rede realiza roteamento de funções, e também pode realizar a fragmentação e remontagem e os erros de entrega de relatório.

- Roteadores operam nesta camada, enviando dados em toda a rede estendida e tornando a Internet possível.
- Este é um esquema de endereçamento lógico - os valores são escolhidos pelo engenheiro de rede.
- O esquema de endereçamento não é hierárquico.

## A camada de rede pode ser dividida em três subcamadas:

- **Sub-rede de acesso** - considera protocolos que lidam com a interface para redes, tais como X.25;
- **Sub-rede dependente de convergência** - necessária para elevar o nível de uma rede de trânsito, até ao nível de redes em cada lado;
- **Sub-rede independente de convergência** - lida com a transferência através de múltiplas redes. Controla a operação da sub rede roteamento de pacotes, controle de congestionamento, tarifação e permite que redes heterogêneas sejam interconectadas.

## Camada 2 – Link de Dados

- A camada de ligação de dados também é conhecida como de enlace ou *link* de dados.
- Esta camada detecta e, opcionalmente, corrige erros que possam acontecer no nível físico.
- É responsável por controlar o fluxo (recepção, delimitação e transmissão de quadros) e também estabelece um protocolo de comunicação entre sistemas diretamente conectados.

- A camada de link de dados pega os pacotes de dados recebidos da camada de rede e os transforma em quadros ou células que serão trafegados pela rede, adicionando informações como o endereço da placa de rede de origem, o endereço da placa de rede de destino, dados de controle, os dados em si e dados de correção de erro (*checksum* ou CRC).
- A diferença entre quadro e célula é que o quadro possui um tamanho configurável, enquanto que a célula possui sempre o mesmo tamanho.
- Se o pacote de dados recebido da camada trê's for maior do que o tamanho do quadro de dados sendo usado pela rede, então o pacote de dados será dividido em tantos quadros quantos forem necessários para a transmissão completa do pacote.

- O quadro criado pela camada link de dados é enviado para a camada física, que converte esse quadro em sinais elétricos para serem enviados através do meio (cabo ou ar, no caso de redes sem fio).
- Esta camada é ainda responsável por verificar se o meio onde os quadros serão transmitidos está disponível e pode ser usado.

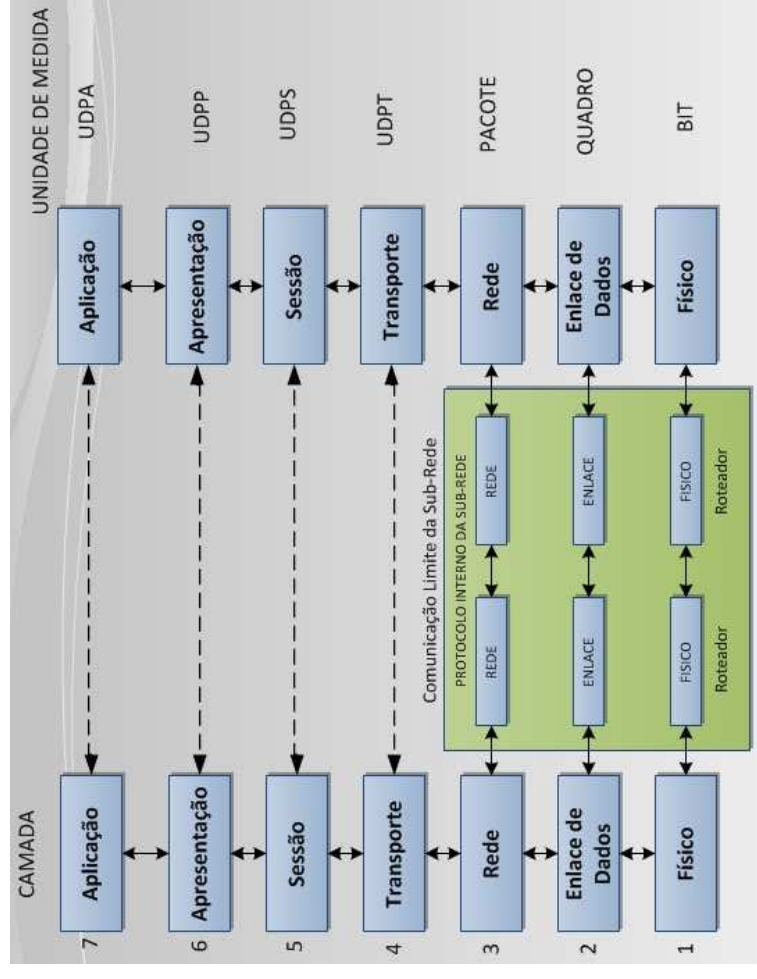
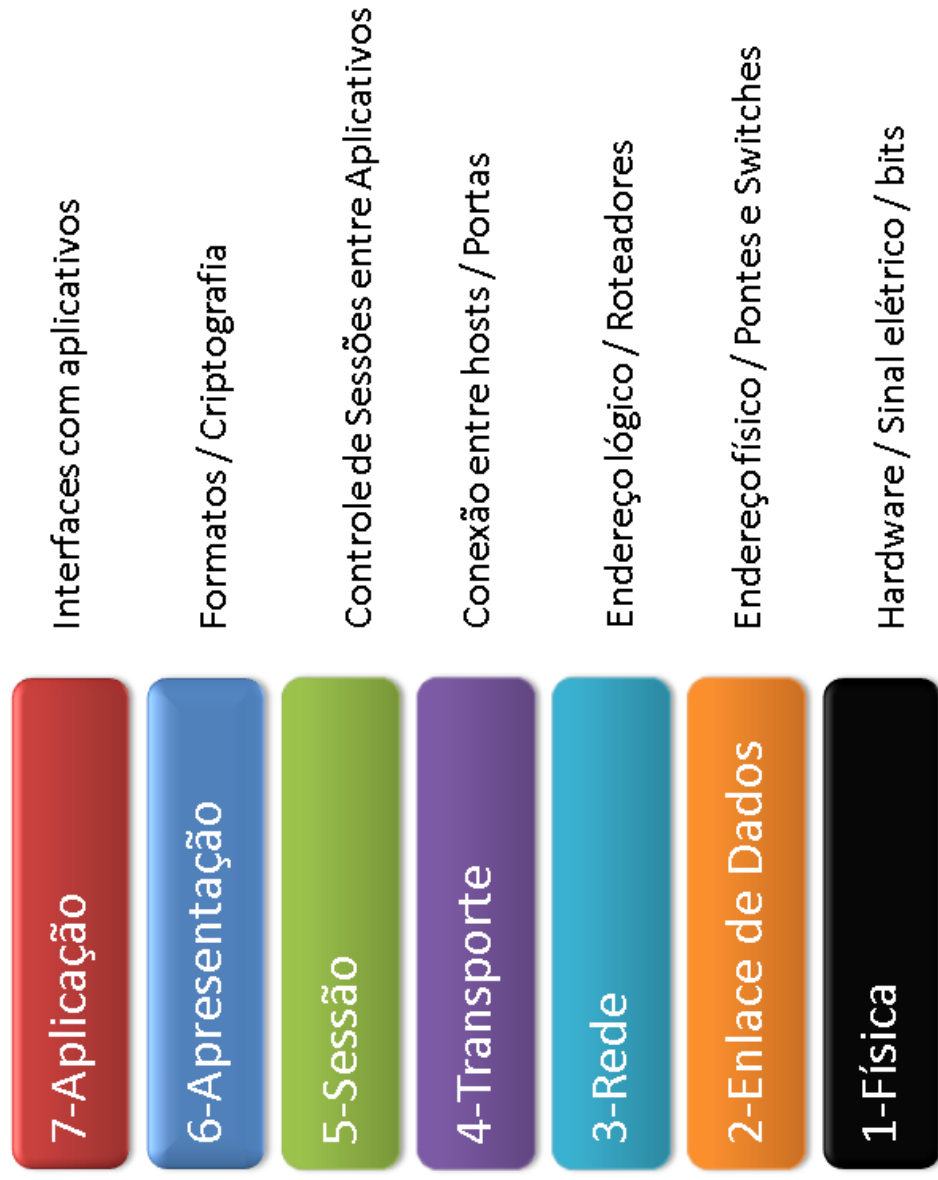


# Camada 1 – Física

- A camada física define especificações elétricas e físicas dos dispositivos.
- Em especial, define a relação entre um dispositivo e um meio de transmissão, tal como um cabo de cobre ou um cabo de fibra óptica.
- Isso inclui o layout de pinos, tensões, impedância da linha, especificações do cabo, temporização, hubs, repetidores, adaptadores de rede, adaptadores de barramento de host (HBA usado em redes de área de armazenamento) e muito mais.

- A camada física é responsável por definir se a transmissão pode ser ou não realizada nos dois sentidos simultaneamente.
- Sendo a camada mais baixa do modelo OSI, diz respeito a transmissão e recepção do fluxo de bits brutos não-estruturados em um meio físico.
- Ela descreve as interfaces elétricas, ópticas, mecânicas e funcionais para o meio físico e transporta sinais para todas as camadas superiores.

- A camada física pega dos quadros enviados pela camada de link de dados e efetua a codificação de modulação de quadro, transmitindo o quadro através de sinais elétricos, luminosos ou de radiofrequência, dependendo se a rede usa cabos metálicos, fibra óptica ou é do tipo sem fio, respectivamente.
- O papel dessa camada é efetuado pela placa de rede dos dispositivos conectados em rede. Note que a camada física não inclui o meio onde dados circulam.
- O máximo com que essa camada se preocupa é com o tipo de cabo usado para a transmissão e recepção dos dados, mas o cabo em si não é responsabilidade dessa camada.
- Assim como a camada dois, a camada um é controlada por hardware e definida pela arquitetura de rede sendo usada.



Camadas	Protocolos	Função
7 Aplicação	HTTP, RTP, SMTP, FTP, SSH, Telnet, SIP, RDP, IRC, SNMP, NNTP, POP3, IMAP, BitTorrent, DNS...	Prover serviços de rede às aplicações
6 Apresentação	XDR, TLS...	Criptografia, codificação, compressão e formatos de dados
5 Sessão	NetBIOS...	Iniciar, manter e finalizar sessões de comunicação
4 Transporte	NetBEUI, TCP, UDP, SCTP, DCCP, RIP...	Transmissão confiável de dados, segmentação
3 Rede	IP, (IPv4, IPv6), Ipsec, ICMP, ARP, RARP, NAT	Endereçamento lógico e roteamento; Controle de tráfego
2 Enlace	Ethernet, IEEE 802.1Q, HDLC, Token ring, FDDI, PPP, Switch, Frame relay, ATM...	Endereçamento físico; Transmissão confiável de quadros
1 Física	Modem, 802.11 WIFI, RDIS, RS-232, EIA-422, RS-449, Bluetooth, USB, 10BASE-T, 100BASE-TX, ISDN, SONET, DSL...	Interface com meios de transmissão e sinalização

**FIM!**