=======================================================

***O Modelo de Referência ISO/OSI***

=======================================================

O modelo de referência ISO/OSI não determina uma arquitetura de rede específica, apenas define um modelo ou padrão que pode ser seguido para a construção de uma arquitetura de rede.

A importância da discussão do modelo de referência OSI está, principalmente, na forma como os conceitos estão organizados em camadas com funções bem definidas.

O modelo OSI está organizado em sete camadas bem definidas: física, enlace, rede, transporte, sessão, apresentação e aplicação.

Cada camada tem como objetivo abstrair a complexidade das camadas inferiores, com funções definidas e formas de usar os recursos da camada imediatamente inferior.

Uma camada fornece à camada superior um serviço através de uma interface simplificada.

A Figura a seguir representa o modelo ISO/OSI na forma de uma pilha de camadas, cada qual com uma função distinta e ligadas a uma camada inferior e a uma camada superior.



# ***Física***

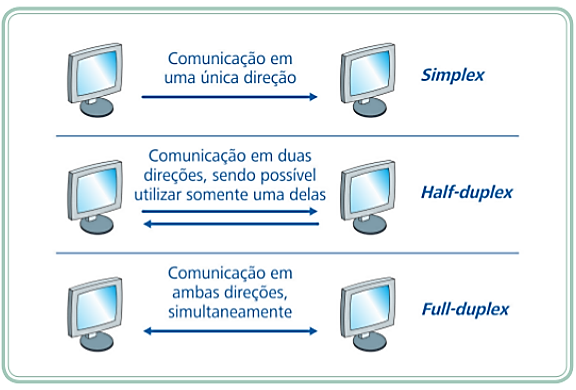
A camada física fornece as características Mecânicas, Elétricas, funcionais e de procedimentos para manter conexões físicas para a transmissão de bits entre os sistemas ou equipamentos (SOARES, et al., 1995).

A camada física fornece as características Mecânicas, Elétricas, funcionais e de procedimentos para manter conexões físicas para a transmissão de bits de dados, na forma de sinais elétricos, ópticos ou outra forma de onda eletromagnética, não existindo qualquer controle de erros de transmissão.

Estão incluídos na camada física os meios de Transmissão: **Cabos Metálicos** (Transmissão de Sinais Elétricos), **Cabos Ópticos** (Transmissão de Ondas Luminosas), entre outros e os componentes de hardware envolvidos na transmissão: Interfaces, Hub, Hardware para transmissão de ondas no espectro eletromagnético (rede sem-fio), etc.

Na camada física são tratadas questões como taxa de transferência de bits, modo de conexão (Simplex, Half-Duplex, Full-Duplex), Topologia de Rede, etc.

Na Figura a seguir são apresentados os modos de comunicação simplex, half-duplex e full duplex, como forma de exemplificar o funcionamento de cada um.



# ***Enlace***

O objetivo da camada de enlace é detectar e opcionalmente corrigir erros de transmissão da camada física, assim convertendo um canal de transmissão não confiável em um canal confiável, para uso pela camada de rede, logo acima.

Para se conseguir um canal de transmissão confiável na camada de enlace, geralmente são usadas algumas técnicas de identificação ou correção nos quadros de bits transmitidos, por meio de inclusão de bits redundantes.

A correção ou retransmissão de um quadro, quando detectado um erro, é opcional e geralmente é deixada para as camadas superiores do modelo.

A camada de enlace também tem a função de prover um mecanismo de controle de fluxo.

Essa função controla o envio de dados pelo transmissor de modo que o receptor não seja inundado com uma quantidade de dados que não consiga processar (SOARES, et al., 1995).

# ***Rede***

A camada de rede deve fornecer à camada de transporte um meio para transferir datagramas (também chamados de **Pacotes** dependendo do contexto) pelos pontos da rede até o seu destino.

Os datagramas (ou pacotes) são unidades básicas de dados, fragmentos de dados das camadas superiores ou aplicações, com os cabeçalhos necessários para a transmissão.

Nessa camada temos o conceito de encaminhamento (ou roteamento) de datagramas, que trata da forma como os datagramas devem ser encaminhados (roteados) pelos nós (roteadores) da rede, de um computador de origem a um computador de destino.

A camada de rede oferece duas classes de serviços: **Orientados à Conexão** e não **Orientados à Conexão**.

No serviço orientado à conexão primeiramente, um transmissor e um receptor estabelecem uma conexão.

Todos os pacotes transmitidos posteriormente entre eles são pertencentes àquela **Conexão** (Circuito) e normalmente, seguem o mesmo caminho.

No serviço de datagrama não orientado à conexão, cada datagrama enviado é independente dos enviados anteriormente, sem estabelecimento de conexão.

Cada datagrama contém em seu cabeçalho a informação do endereço do **Transmissor** (Origem, Remetente do Pacote) e do **Receptor** (Destinatário).

Os nós **Intermediários** (Roteadores) se encarregam de selecionar o melhor caminho e **Encaminhar** (Rotear) os **Datagramas** (Pacotes) do **Transmissor** (Remetente) até o **Receptor** (Destinatário) (SOARES, et al., 1995).

# ***Transporte***

Até agora, na camada de rede e inferiores, a transferência ocorre, de fato, apenas entre os nós (máquinas) próximos na rede.

A camada de transporte, por outro lado, permite que os dados trafeguem em um circuito virtual direto da origem ao destino, sem preocupar-se com a forma que os pacotes de dados viajam na camada de rede e inferiores.

A camada de transporte, dessa forma, é responsável pela transferência fim a fim de dados entre processos de uma máquina de origem e processos de uma máquina de destino.

A transferência de dados, na camada de transporte, ocorre de modo transparente, independente da tecnologia, topologia ou configuração das redes nas camadas inferiores.

É tarefa da camada de transporte cuidar para que os dados sigam ao seu destino sem erros e na sequência correta, condições para que se crie a ideia de um caminho fim a fim.

Além da detecção e recuperação de erros e controle da sequência dos dados, outras funções desta camada são: multiplexação de conexões e controle de fluxo.

A multiplexação permite que vários processos diferentes nas máquinas de origem e destino troquem dados ao mesmo tempo.

Os pacotes de dados de vários processos de uma máquina de origem são enviados para vários processos em uma máquina de destino.

Como o meio, usado nas camadas inferiores é compartilhado, os pacotes de dados precisam ser multiplexados (escalonados, embaralhados, misturados), de modo que se tem a impressão de que as transferências ocorrem simultaneamente, em paralelo.

O aluno, neste ponto, pode estar se perguntando como os pacotes multiplexados dos vários processos encontram os processos de destino corretos?

Para que isso ocorra, a camada de transporte possui mecanismos para identificar cada pacote ao seu devido fluxo de dados entre os processos.

Uma forma de identificação ou endereçamento de pacotes, com relação ao processo de origem e destino, será vista quando tratarmos dos protocolos de transporte da internet.

O controle de erros possui mecanismo para identificar erros de transmissão (pacotes com dados corrompidos, por exemplo) e prover a recuperação desse erro, seja por meio da retransmissão do pacote ou outra forma de reconstrução da informação do pacote.

O controle de sequência visa garantir a ordem correta da informação, independentemente da ordem em que os pacotes de dados chegaram ao destino.

Outra função importante da camada de transporte é o controle de fluxo.

O destinatário e o emissor dos pacotes podem ter limites diferentes quanto a quantidade de dados que podem receber ou enviar.

Um mecanismo de controle de fluxo evita que o destino receba mais dados do que tem condições de receber e processar.

Basicamente, o controle de fluxo permite que a máquina de origem ajuste o seu volume de pacotes enviados de acordo com a capacidade do destino em receber pacotes naquele momento, seja aumentando ou diminuindo a vazão do fluxo de pacotes, conforme a reação observada do destino.

# ***Sessão***

A camada de sessão possui mecanismos que permitem estruturar os circuitos oferecidos pela camada de transporte.

As principais funções da camada de sessão são: **Gerenciamento de Token**, **Controle de Diálogo** e **Gerenciamento de Atividades**.

O **Gerenciamento de Token** é necessário em algumas aplicações, quando a troca de informações é Half-Duplex, ao invés de Full-Duplex, permite que apenas o proprietário do token possa transmitir dados naquele momento.

O **Controle de Diálogo** usa o conceito de ponto de sincronização, quando a conexão para a transferência de dados de uma aplicação é interrompida, por erro, a transferência pode ser reestabelecida do ponto onde havia parado.

O **Conceito de Atividade** permite que as aplicações ou serviços oferecidos aos usuários coordenem as partes constituintes da transferência de dados.

Cada atividade possui um conjunto de dados que devem ser trocados entre o serviço na origem e na aplicação de destino.

Apenas uma atividade é executada (dados transmitidos) por vez, porém, uma atividade por ser suspensa, é reordenada e retomada.

=======================================================

***Simplex, Half-Duplex e Full-Duplex***

=======================================================

Esses canais de comunicação fornecem caminhos para transmitir informações, podem ser tanto por meio de transmissão física quando por uma conexão lógica em um meio multiplexado.

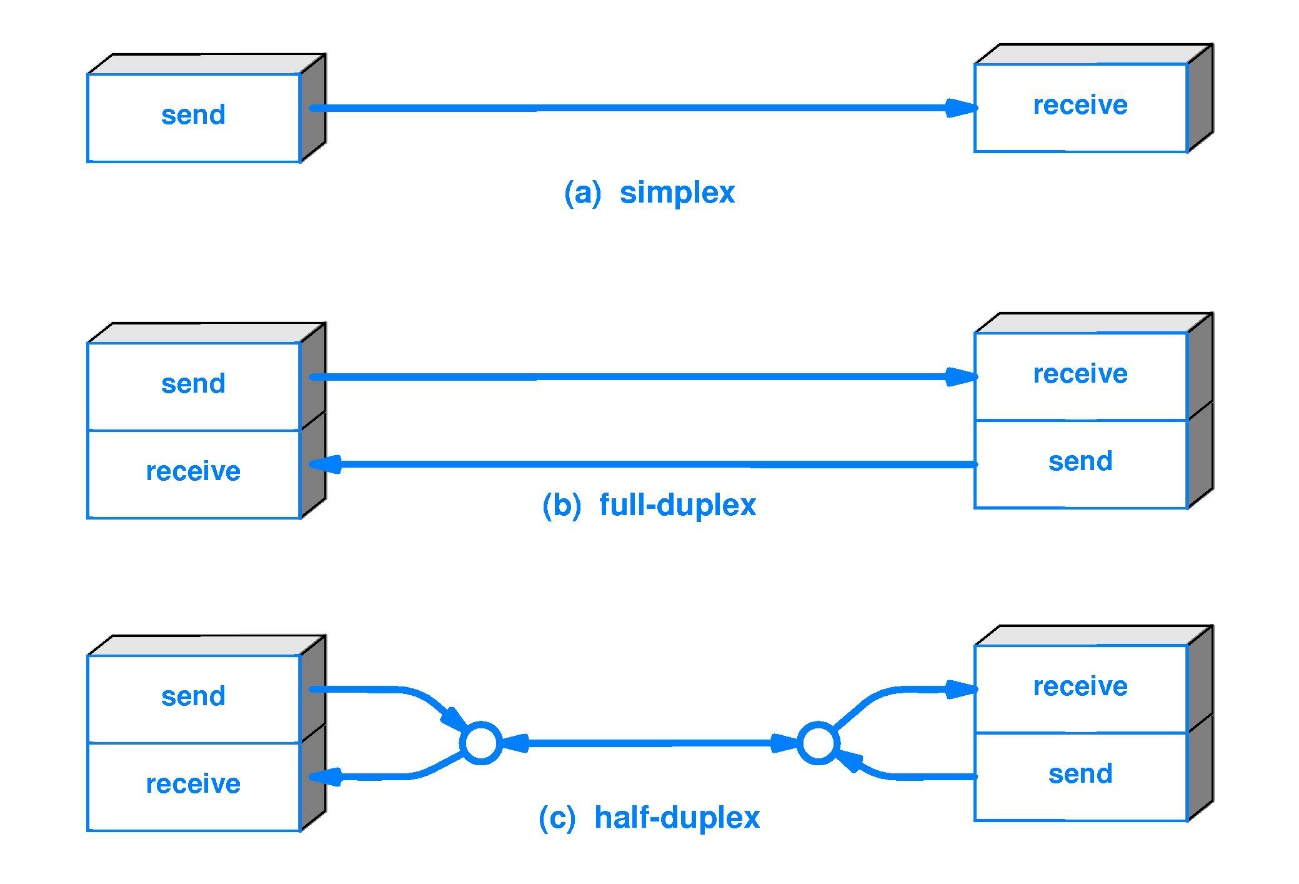
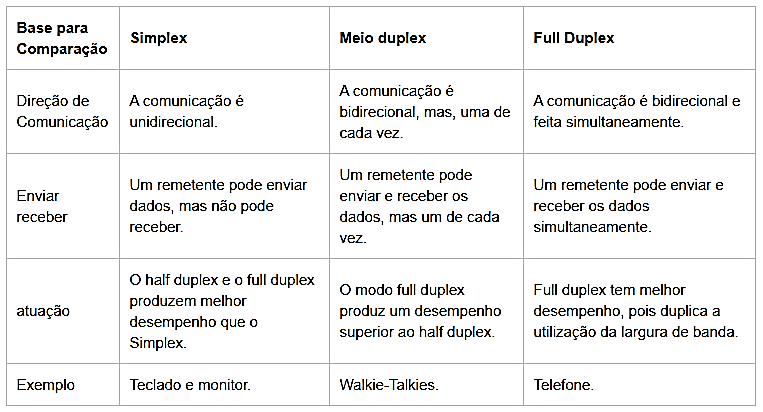
***Simplex***

**Caracteriza uma ligação onde os dados circulam em Apenas um Sentido**, de um transmissor para um receptor, **sendo que este papel Não se Inverte nunca no período de transmissão**, a transmissão tem sentido unidirecional.

***Half Duplex***

Em uma rede, a informação deve ser transmitida em Ambas as Direções. A palavra "Half" em "Half Duplex" **Significa que em um intervalo de tempo definido, a informação só pode ir em uma direção.**

**Full Duplex**

Permite que o trafego Bidirecional Simultâneo, **Possibilitando que a informação flua continuamente em ambas as direções** -- o que significa que pacotes de dados nunca precisam esperar para serem enviados.

# ***Apresentação***

A camada de apresentação cuida da formatação dos dados, transformação, compressão e criptografia, não havendo multiplexação (combinação de dois ou mais canais de informação por apenas um meio de transmissão) de dados na camada de apresentação.

O propósito desta camada é converter as informações que são recebidas da camada de aplicação para um formato “entendível” na transmissão desses dados.

Como exemplo de conversão, estão os caracteres diferentes do padrão usual ASCII que precisam ser “Tratados” ou quando os dados recebidos são criptografados sobre diferentes formas de criptografia, desta forma também sendo necessário uma conversão destes dados (SILVA, 2010).

# ***Aplicação***

Na camada de aplicação estão os aplicativos, propriamente ditos, dos usuários ou os serviços dos sistemas.

Esta camada cuida da comunicação entre as aplicações, sendo que cada aplicação possui protocolos específicos de comunicação, também, aos protocolos usados na comunicação de dados entre essas aplicações.

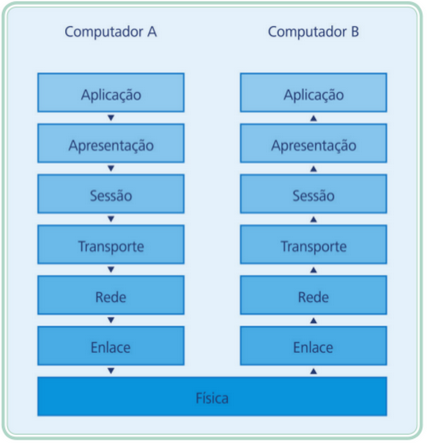
As aplicações que oferecem recursos aos usuários ou aos sistemas mais conhecidos atualmente são aquelas que oferecem serviços no padrão da internet:

* Aplicação para Navegação;
* Transferência de Arquivos;
* Transferência de E-Mail;
* Terminal Remoto e Outros.



***Principais funções das Camadas***

Na exemplificação, como forma de entendimento dos conceitos das camadas de rede pertencentes ao modelo OSI, é caracterizada a trajetória de um pacote de rede, realizada na troca de dados entre dois dispositivos.

Isto ajuda a explicar o caminho percorrido por um pacote, passando pelas camadas.

Na Figura anterior, traçamos o caminho seguindo pelo tráfego de informações de um computador hipotético “A” até um computador “B”.

A informação a ser comunicada encontra-se na aplicação do computador “A” que deve enviá-la à aplicação no computador “B”.

Nesse ponto, repare que as únicas funções de interesse da camada de aplicação é a “conversa” (comunicação) entre as aplicações nos computadores distintos, como o que enviar e o que responder.

A aplicação em “A” “conversa” apenas com a outra aplicação em “B”.

Para a transferência de fato dos dados, a camada de aplicação usará os serviços da camada imediatamente inferior, a camada de apresentação.

A camada de apresentação, por sua vez, fará o uso da camada de sessão e assim sucessivamente até a camada física.

A transferência de dados (neste caso, codificados em bits brutos) ocorre de fato e unicamente na camada física.

Ao longo da descida até a camada física, cada camada encapsulou os pacotes de dados e adicionou os seus cabeçalhos de controle e endereçamento, relativo a cada camada.

Quando os pacotes chegam à outra ponta do meio de transmissão da camada física e assim alcançam o computador “B”, o processo inverso acontece.

Cada pacote é processado conforme as informações de endereçamento e controle e tem os cabeçalhos da camada removidos e colocados na camada superior.

O processo acontece até que o pacote atinge o processo de destino devido (aplicação), na camada de aplicação de “B”.

Historicamente, o modelo OSI não se tornou padrão para a internet, embora muitos protocolos e tecnologias de rede tenham sido desenvolvidos baseados nele.

Embora o RM-OSI não tenha se tornado o padrão dominante para a ligação de redes e muitos dos protocolos baseados nele tenham sido suplantados pelo TCP/IP, o estudo do modelo justifica-se pela generalidade dos conceitos adotados na sua construção (TANENBAUM, 2003).

=======================================================

***O Modelo de Referencia TCP/IP***

=======================================================

O modelo de referência **TCP/IP** é mais simplificado que o modelo de referência OSI, possuindo quatro camadas principais: aplicação, transporte, internet e interface de rede.

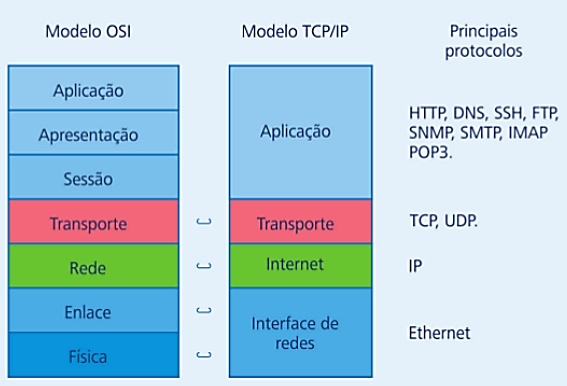
A semelhança entre o modelo de referência **OSI** e o modelo **TCP/IP** está no fato dos dois estarem baseados no conceito de pilha (contendo protocolos independentes). Como características o modelo **TCP/IP** possui:

Quatro camadas – sendo as camadas de rede, transporte e aplicação, comum tanto ao modelo de referência OSI, como ao modelo **TCP/IP**.

Adaptativo – sua criação baseou-se na adaptação para protocolos existentes, enquanto que o modelo de referência **OSI** (criado antes dos protocolos) apresenta-se como mais genérico e flexível.

Na Figura a seguir, é possível visualizar a diferença de camadas entre o modelo **OSI** tradicional e o modelo **TCP/IP**, que na verdade abstrai algumas camadas existentes no modelo **OSI**.

Ao lado das camadas é possível observar também os principais protocolos que trabalham em camadas específicas. Esta associação **Modelo/Camadas/Protocolos**, ajuda no entendimento de como funciona uma rede de computadores no todo.



# ***Camada de Interface de Rede***

Esta camada tem como objetivo principal conectar um **Dispositivo de Rede** (Computador, Notebook, Etc.) a uma rede, utilizando para isso um protocolo.

Nesta camada, a exemplo de como ocorre na camada física do modelo **OSI**, é tratada a informação em mais baixo nível (**Bits que Trafegam pela Rede**) entre as diferentes tecnologias para este fim: **Cabo de par Trançado, Fibra Óptica, etc.** (SCRIMGER, 2001).

# ***Camada de Internet***

Esta camada tem o objetivo de permitir aos dispositivos de rede enviar pacotes e garantir que estes pacotes cheguem até seu destino. Cabe a camada de internet especificar o **Formato do Pacote**, bem como, o **Protocolo Utilizado**, neste caso o protocolo **IP (*Internet Protocol*).**

Semelhante a camada de rede do modelo de referência **OSI**, cabe a camada de internet realizar a entrega dos pacotes **IP** no destino e realizar o roteamento dos pacotes.

# ***Camada de Transporte***

A camada de transporte do modelo **TCP/IP** possui a mesma função da camada de transporte do modelo de referência OSI, ou seja, garantir a comunicação entre os dispositivos de origem e destino do pacote.

Fazem parte desta camada dois protocolos bastante populares nas redes de computadores: o protocolo **TCP (*Transmission Control Protocol*)** e o **UDP (*User Datagram Protocol*).**

***Protocolo TCP***

Considerado um protocolo confiável (devido a quantidade de verificações, confirmações e demais procedimentos realizados), o protocolo TCP garante a entrega dos pacotes aos computadores presentes na rede.

O fluxo dos pacotes de rede passa desta camada (depois de fragmentados) para a camada de internet (para onde são encaminhados). No computador destino é feita a verificação e montagem de cada um dos pacotes, para então ser efetivado o recebimento dos mesmos.

***Protocolo UDP***

**Protocolo Sem Confirmação** **(UDP)** é comumente utilizado na transferência de dados, porém, não realiza nenhuma operação de confirmação e verificação de pacotes na estação destino (procedimento realizado pela própria aplicação).

Apesar de ser classificado como um protocolo não- confiável, o **UDP** é mais rápido que o **TCP** (justamente por ter um mecanismo de funcionamento mais simplificado), sendo utilizado em requisições que não necessitam de confirmação, como é o caso de consultas **DNS**.

# ***Camada de Aplicação***

Esta camada tem por objetivo **Realizar a Comunicação** entre os **Aplicativos** e os **Protocolos de Transporte**, responsáveis por dar encaminhamento a estes pacotes.

Os Protocolos da Camada de Transporte são usualmente conhecidos e desempenham diferentes funções, conforme exemplos a seguir:

***Protocolo SMTP***

Responsável pela comunicação junto ao servidor de e-mails, para entrega destes, ao programa cliente que recebe as mensagens.

***Protocolo HTTP***

Acionado cada vez que um usuário abre um **Browser** (Navegador) e digita um endereço de um site da internet.

***Protocolo FTP***

Utilizado cada vez que um usuário acessa um endereço de **FTP**, para fazer **Download** ou **Upload de Arquivos** (KUROSE, 2010).

Além dos exemplos de protocolos de aplicação citados acima, existem diversos outros que realizam procedimentos importantes para nossas principais atividades do dia-a-dia, como é o caso dos protocolos de aplicação: **DNS**, **SSH**, **POP3**.

=======================================================

***2 – Quais as diferenças entre o modelo ISO/OSI e TCP/IP***

R:

A diferença entre ISO/OSI e TCP/IP é que o modelo OSI é uma padronização, um modelo conceitual que serve de base para criar outros modelos, e o modelo TPC/IP temos implementado na prática, combinando algumas camadas do OSI em uma só.

***3 – As camadas 2 e 4 do modelo ISO/OSI têm funções razoavelmente semelhantes. No entanto, existe uma grande diferença entre elas. Qual é essa diferença? Por que as duas camadas são necessárias?***

R:

A diferença entre as duas camadas é que a camada de enlace a segunda camada ela faz correção ou retransmissão de um quadro, quando detectado um erro, é opcional e geralmente é deixada para as camadas superiores do modelo. a sua necessidade e pois ela tem a função de prover um mecanismo de controle de fluxo.

Essa função controla o envio de dados pelo transmissor de modo que o receptor não seja inundado com uma quantidade de dados que não consiga processar.

A camada de Transporte ela é parecida com a camada de enlace pois ela também faz a verificação dos pacotes uma forma de identificação ou endereçamento de pacotes com relação ao processo de origem e destino será vista quando tratarmos dos protocolos de transporte da internet como ela faz conexão entre as máquinas de origem e destino ela é necessária pois ela faz o controle de sequência visa garantir a ordem em que os pacotes de dados chegaram ao destino além disso sua importância se dá também pelo controle de fluxo.

***4 – Na camada de transporte do modelo TCP/IP pode-se utilizar os protocolos TCP e UDP. Qual a diferença entre eles? Quais as vantagens de um em relação ao outro?***

R:

O TCP é um protocolo orientado à conexão e fornece a comunicação confiável com vários recursos. Por outro, o UDP é um protocolo de transporte em tempo real sem conexão, que oferece um processo de comunicação mais rápido, mas com pouca confiabilidade.

O TCP  é mais confiável, pois possui a garantia de que os dados enviados serão entregues ao receptor, já o UDP não oferece garantia e os dados podem ser corrompidos ou perdidos durante a entrega.

O TCP possui controle de fluxos que garante que um remetente não sobrecarregue um receptor enviando muitos pacotes de uma vez. Já o UDP não fornece controle de fluxo. Com ele, os pacotes chegam em um fluxo contínuo ou são descartados.

O UDP possui uma velocidade maior que o TCP, (justamente por ter um mecanismo de funcionamento mais simplificado), sendo utilizado em requisições que não necessitam de confirmação, como é o caso de consultas DNS.