

# **Servidores e seus sistemas operacionais**

**Prof. Orlando Saraiva Júnior**  
**[orlando.saraiva@unesp.br](mailto:orlando.saraiva@unesp.br)**

# Objetivo da aula

---

Apresentação do plano de ensino, metodologia de avaliação e metas.

Revisão de redes de computadores.

## Ementa

Identificação dos componentes para montar um servidor, compatibilidade, detalhes sobre as especificações de cada componente, identificação e correção de problemas. Instalação e configuração de softwares (aplicativos gerais, gerenciadores de bancos de dados e sistemas operacionais). Compreender os conceitos, mecanismos e funcionamento dos Sistemas operacionais modernos. Criação de partições, formatação de discos, instalação de *drivers*. Virtualização de servidores para a implementação de serviços de gerenciamento de usuários, serviços de comunicação e de armazenamento de dados através da utilização de uma intranet composta por aplicativos da internet. Configuração de aplicações e serviços Web. Migração de Web sites e aplicações. Configuração de sites seguros (Controle de acesso e Autenticação)

---

## **Objetivos Gerais:**

Apresentar os recursos tecnológicos empregados em servidores de redes de computadores.

## **Objetivos Específicos:**

Instalar sistema Operacional e aplicativos, instalar drivers, executar softwares de teste. Conhecer as tecnologias de diagnóstico partindo dos conceitos básicos como análise e identificação do problema (diagnóstico) e definição do plano de ação. Conhecer e compreender o funcionamento dos principais serviços de dados através da implementação e utilização de sistemas Operacionais de gerenciamento de redes e dados em plataformas de software livre e proprietário. Configurar, gerenciar e manter os principais serviços e protocolos de servidores Web. Conhecer tecnologias de Web Proxy. Instalar, configurar e implementar os aspectos de segurança dos serviços Web. Praticar em laboratório de arquitetura e redes de computadores: instalação e configuração de sistemas operacionais, utilitários e hardware.

---

---

Nota Final = Pesquisa ( 10%) + Projeto ( 40% ) + Prova ( 50 % )

Na Pesquisa (entrega no dia 15/06), o aluno deve fazer uma pesquisa individual sobre os tipos de sistemas operacionais.

No Projeto (entrega no dia 05/12), individualmente o aluno deve instalar um sistema operacional Unix-like, instalar um servidor web com PHP e um servidor SGDB ( PostgreSQL ou MySQL ou Oracle ) .

Na prova escrita ( 28/11), o aluno deve fazer uma prova escrita. Aos que perderem, prova sub será no dia 12/12

# Revisão

Conjunto de regras sobre o modo como se dará a comunicação entre as partes envolvidas.

Um protocolo define o formato e a ordem das mensagens trocadas entre duas ou mais entidades comunicantes, bem como as ações realizadas na transmissão e/ou recebimento de uma mensagem ou outro evento.

Para facilitar a interconexão de sistemas a ISO (*International Standards Organization*) criou o modelo OSI (*Open System Interconnection*), para que os fabricantes criassem seus dispositivos a partir desse modelo.

A divisão por camadas de protocolos simplifica o projeto do software de rede.



# Modelo OSI

---

Aplicação

Apresentação

Sessão

Transporte

Rede

Link de dados ( enlace )

Física

Para reduzir a complexidade do projeto lógico em redes, a maioria delas é organizada como uma pilha de **camadas** ou **níveis**, colocada uma sobre as outras.

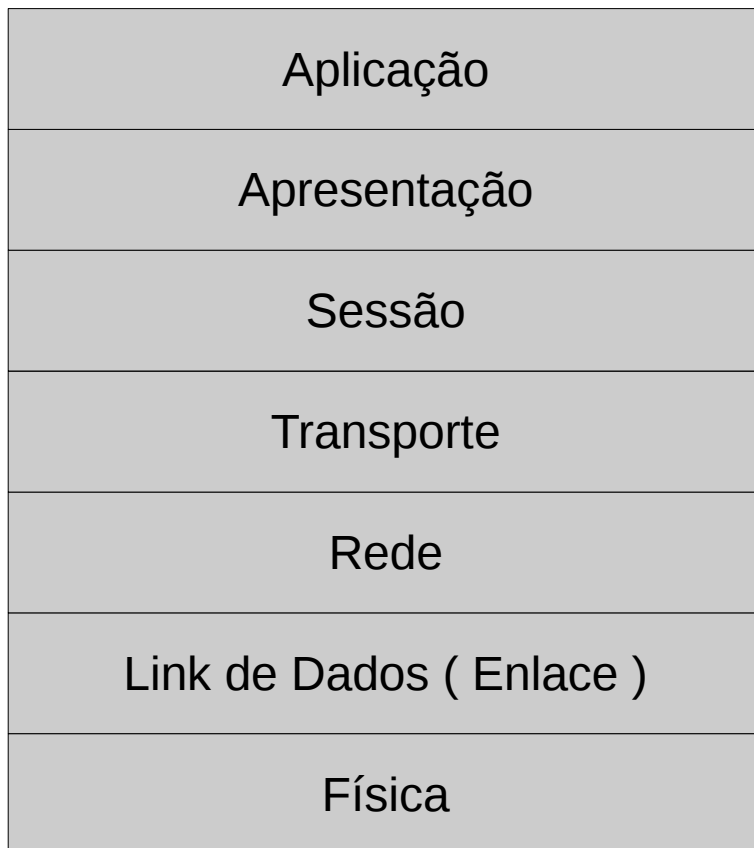
O número de camadas, o nome, o conteúdo e função de cada camada é o que difere uma rede de outra.

Em cada camada há protocolos específicos, que precisam estar presentes no sistemas finais cliente e servidor.

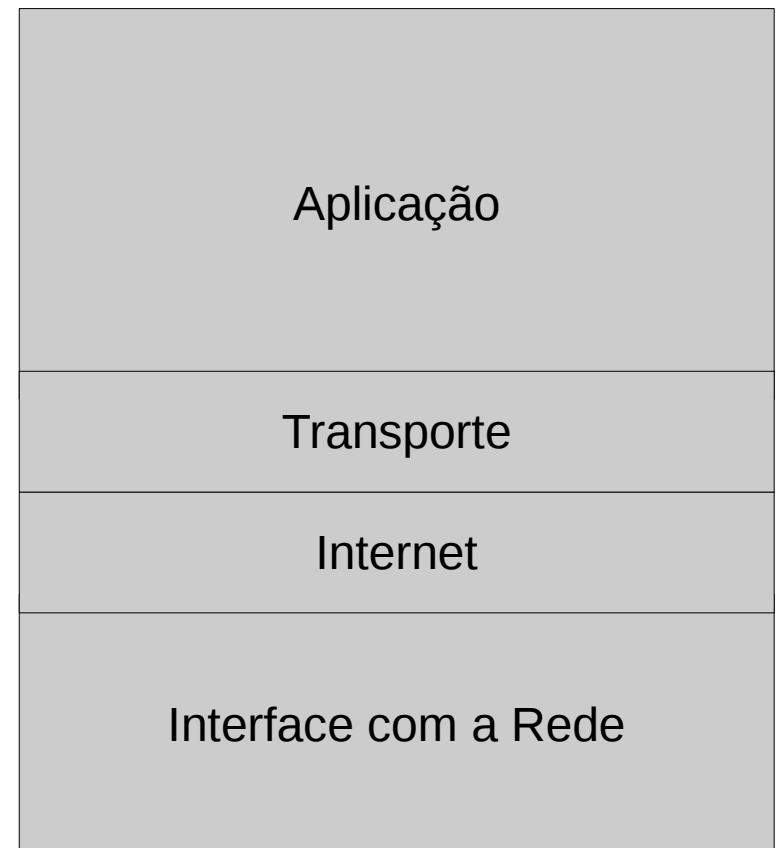
# Camadas de protocolos

---

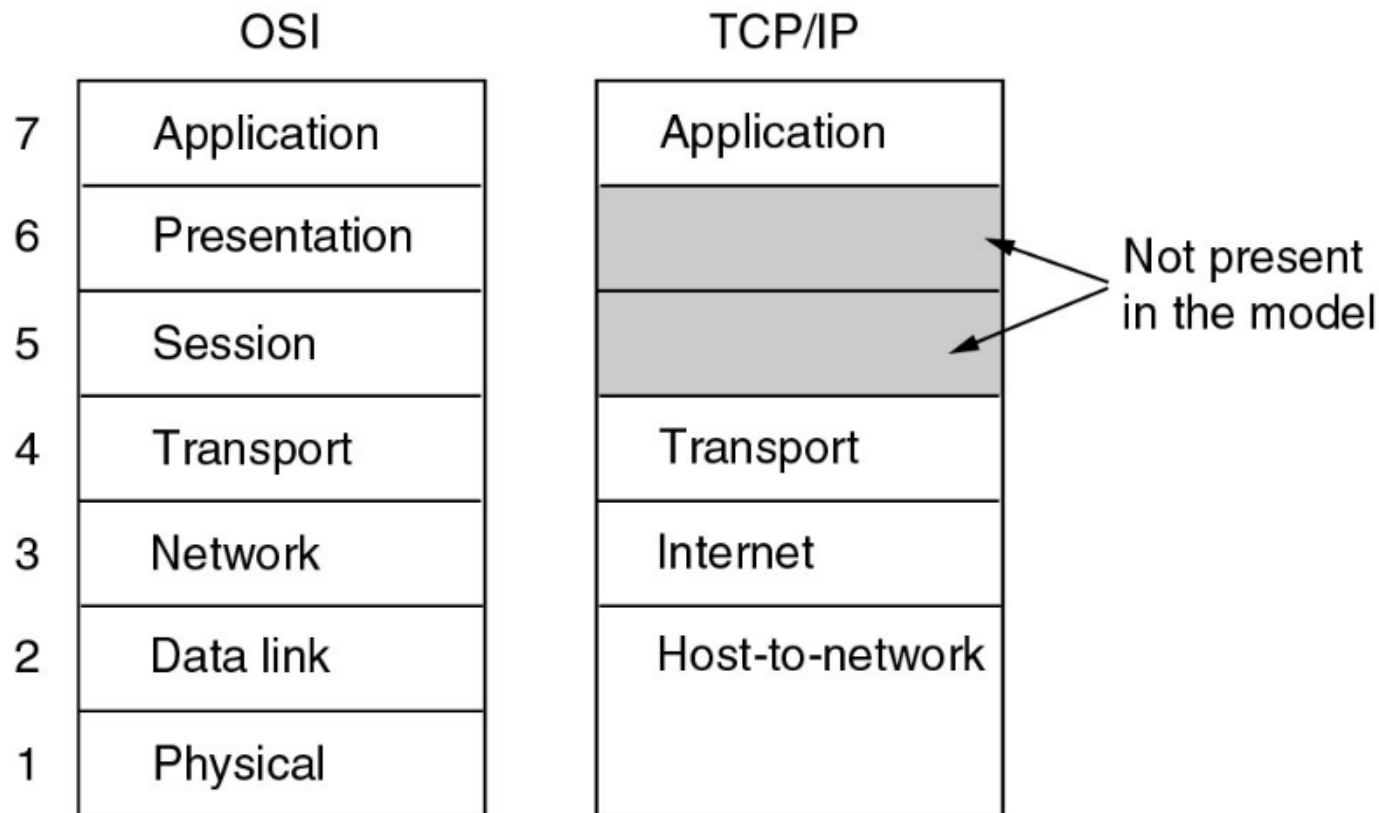
## Modelo OSI



## Modelo TCP/IP



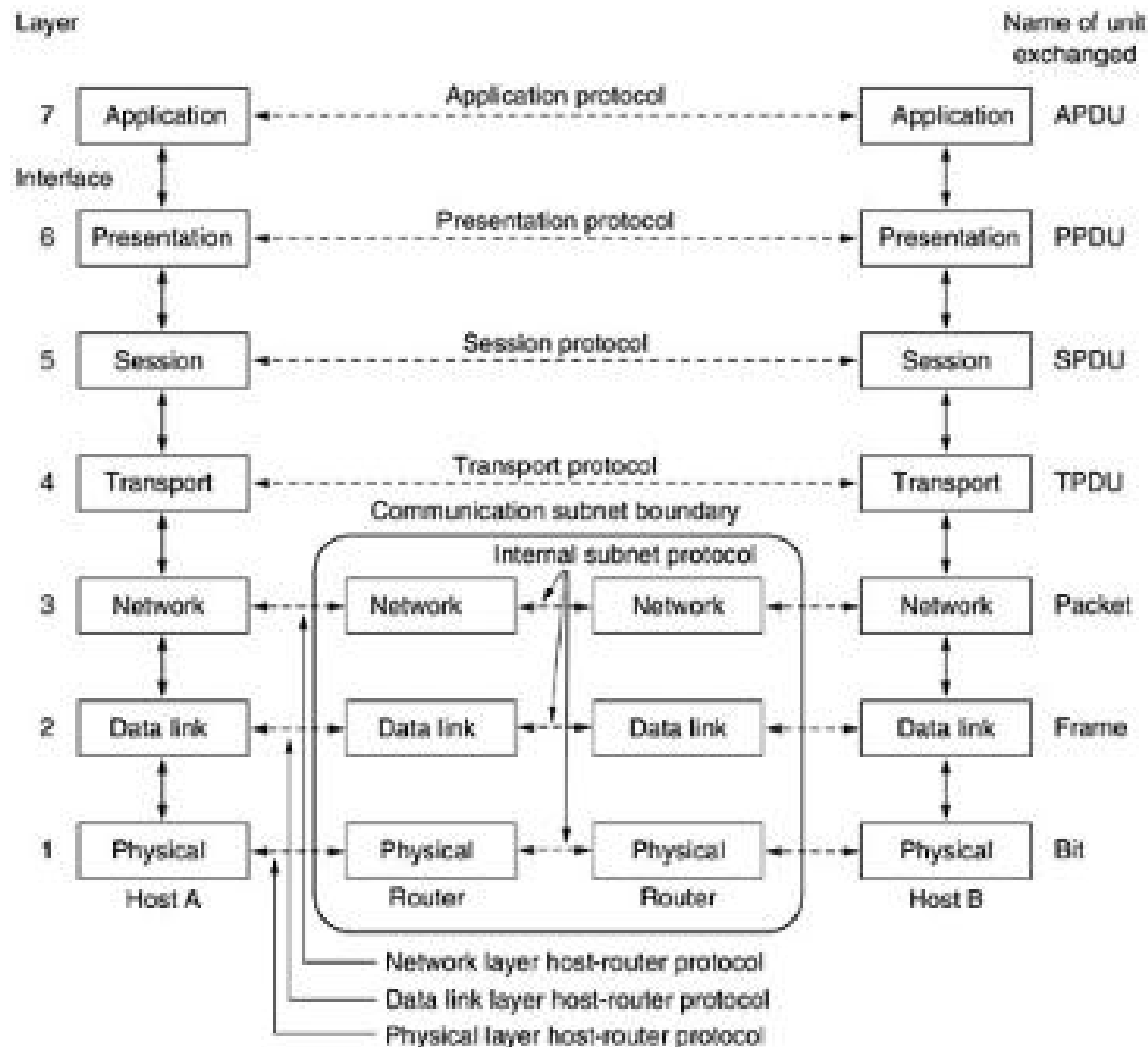
# Camadas de protocolos



O modelo de referência TCP/IP.

A. S. Tanenbaum, Computer Networks, 4<sup>th</sup> ed., Prentice Hall PTR, 2003.

# Camadas de protocolos



# Camadas de protocolos

## Questões relacionadas

---

Todas as camadas precisam de um mecanismo para identificar os transmissores e os receptores.

Como em geral uma rede possui muitos computadores, e alguns deles possui vários processos, é necessário um meio para que um processo de uma máquina especifique com quem ela deseja se comunicar.

Surge a necessidade de se criar uma forma de **endereçamento** para definir um destino específico.

# Camadas de protocolos

## Questões relacionadas

---

Outra preocupação se deve à **transferência de dados**. Em alguns sistemas, os dados são transferidos em apenas um sentido; em outros, os dados trafegam em ambos os sentidos.

O protocolo também deve definir a quantos canais lógicos corresponde a conexão e quais as suas prioridades.

O **controle de erro** é uma questão importante, pois os circuitos de comunicação física não são perfeitos. Muitos códigos de detecção e correção de erros são conhecidos, mas as partes de conexão devem chegar a um consenso quanto ao que está sendo usado.

Além disso, o receptor deve ter algum meio para informar ao transmissor quais mensagens foram recebidas corretamente e quais não foram.

# Serviços

## Primitivas de Serviço

---

Um **serviço** é especificado formalmente por um conjunto de primitivas ( operações ) disponíveis para que um processo do usuário acesse o serviço.

Essas primitivas informam ao serviço que ele deve executar alguma ação ou relatar uma ação executada por uma entidade par. Se a pilha de protocolos estiver localizada no sistema operacional, essas primitivas são **chamadas ao sistema**.

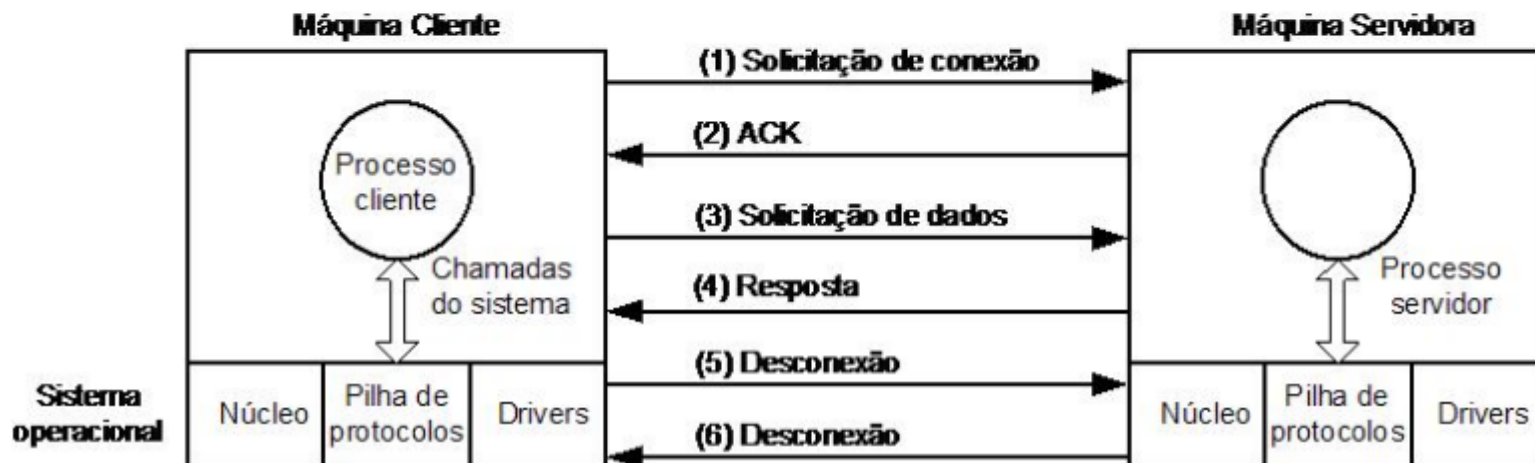
O conjunto de primitivas disponíveis depende da natureza do serviço que está sendo fornecido. As primitivas para um serviço orientado a conexões são diferentes das que são oferecidas em um serviço sem conexões.



# Serviços

## Primitivas de serviço de uma conexão simples

Primitiva	Significado
LISTEN	Espera bloqueada por uma conexão de entrada
CONNECT	Estabelece uma conexão com um par que está à espera
RECEIVE	Espera bloqueada por uma mensagem de entrada
SEND	Envia uma mensagem ao par
DISCONNECT	Encerra uma conexão



## Primitivas de serviço de uma conexão simples

O servidor executa **LISTEN** para indicar que está preparado para aceitar conexões de entrada. Um caminho comum para implementar **LISTEN** é torná-la uma chamada de sistema bloqueada.

O processo cliente executa **CONNECT** para estabelecer uma conexão com o servidor. A chamada de **CONNECT** precisa especificar a quem se conectar.

A próxima etapa é a execução de **RECEIVE** pelo servidor, afim de se preparar para aceitar a primeira conexão.

Depois, o cliente executa o **SEND** para transmitir a solicitação.

Ao terminar, a execução do **DISCONNECT** é solicitada, encerrando a conexão.

# Camadas

## Camada 1

---



Pega os quadros enviados pela camada enlace e os transforma em sinais compatíveis com o meio onde os dados deverão ser transmitidos (elétrico, óptico, etc...)

A camada Física:

- Não sabe o significado dos dados que está transmitindo
- Não inclui o meio onde os dados circulam (cabo de rede), apenas precisa saber qual o meio e tipo de conector para fazer a conversão correta.

# Camadas

## Camada 1

---

Aplicação
Apresentação
Sessão
Transporte
Rede
Link de Dados ( Enlace )
Física

Problemas relacionados:

**Largura de banda:** É a quantidade em bits/s que a rede suporta, uma propriedade física do meio de transmissão.

**Vazão** é a taxa ( em bits/s) em que os bits são transmitidos entre um emissor e um receptor.

**Latência** é o atraso de tempo entre o momento que um evento iniciou e o momento que os efeitos iniciam.

Pesquisar: Ferramenta iperf

# Camadas

## Camada 1

---

O objetivo da camada física é **transmitir um fluxo bruto de bits** de uma máquina para outra. Vários meios físicos podem ser usados para realizar a transmissão. Cada um tem seu próprio nicho em termos de **largura de banda, retardo, custo e facilidade de instalação e manutenção.**

# Camadas

## Camada 2

---



Funções:

Encapsulamento (ou enquadramento):  
identificação dos quadros dentro de  
sequências de bits enviadas e recebidas  
da camada física.

Controle e detecção de erros

controle de Fluxo

Endereçamento

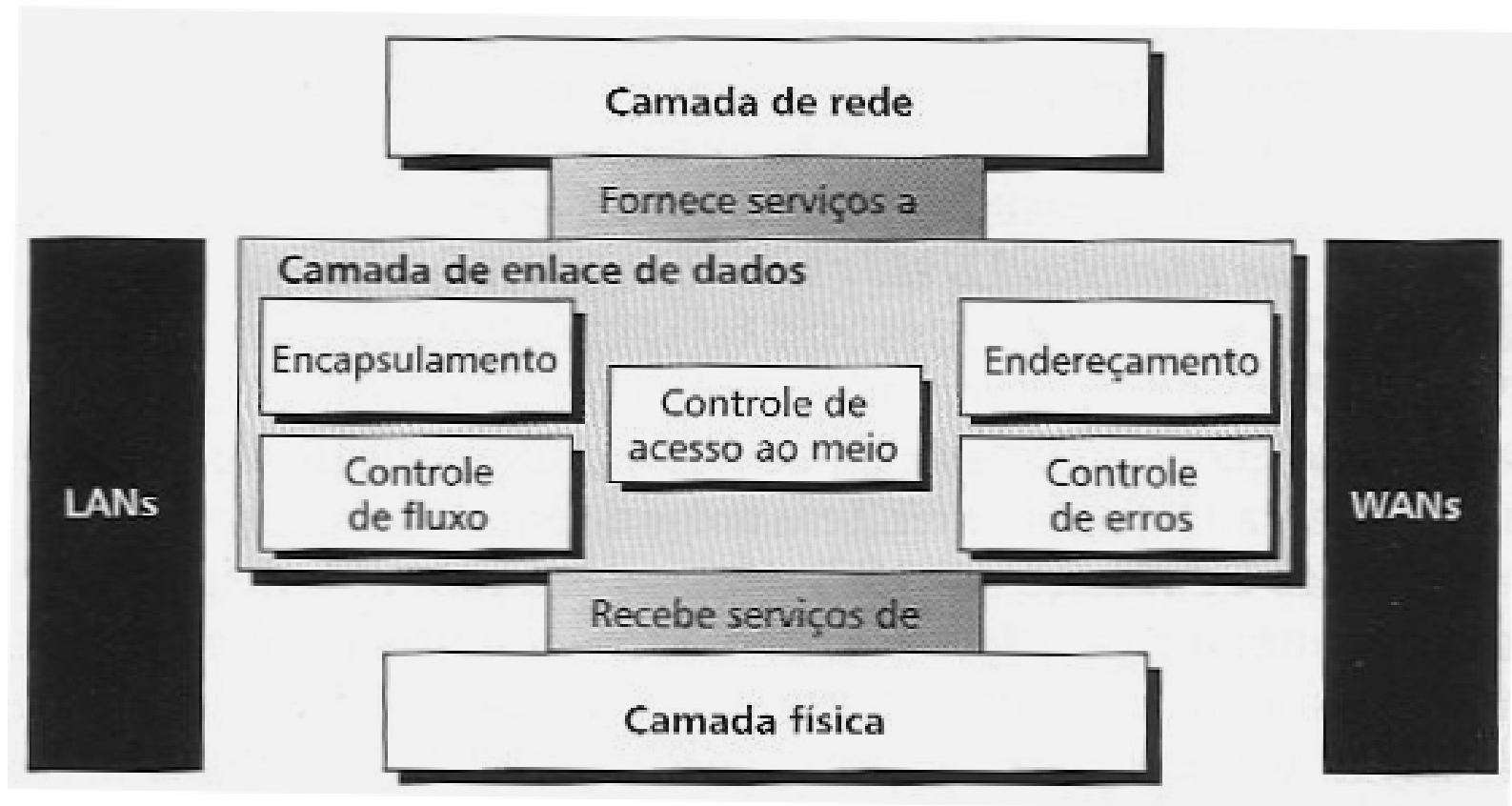
Controle de acesso ao meio ( MAC )

Gerenciamento de enlace

# Camadas

## Camada 2

---

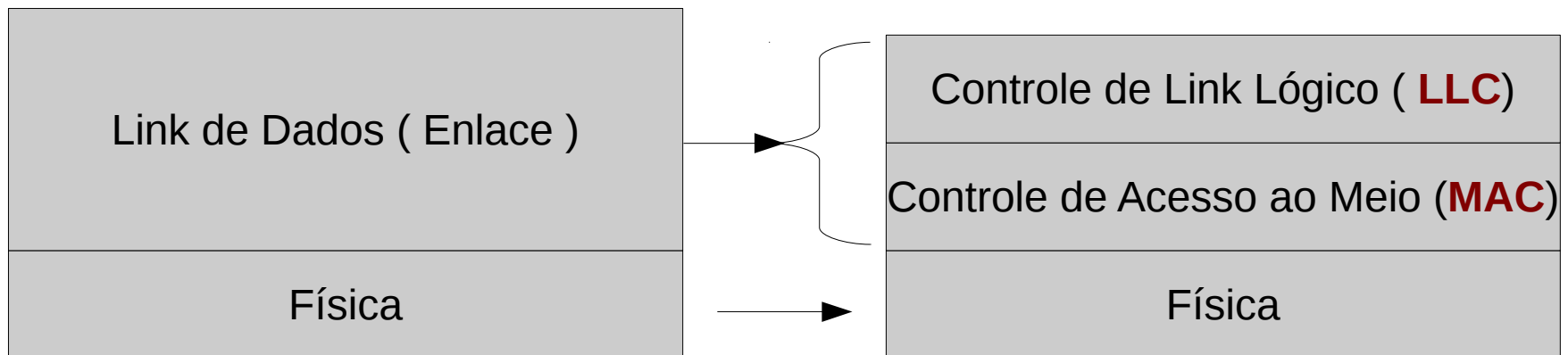


# Camadas

## Camada 2

---

Esta camada é quebrada em duas sub-camadas (IEEE 802) : a de Controle de Enlace Lógico - LLC (Logical Link Control), que efetua a verificação de erro e a de Controle de Acesso ao Meio - MAC (Medium Access Control), que trata de pegar e entregar dados ao meio.



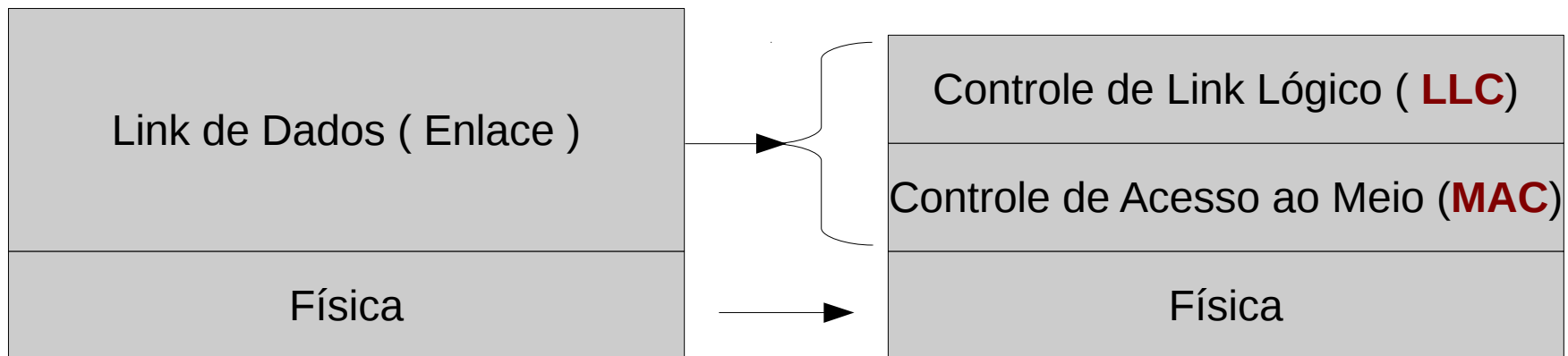


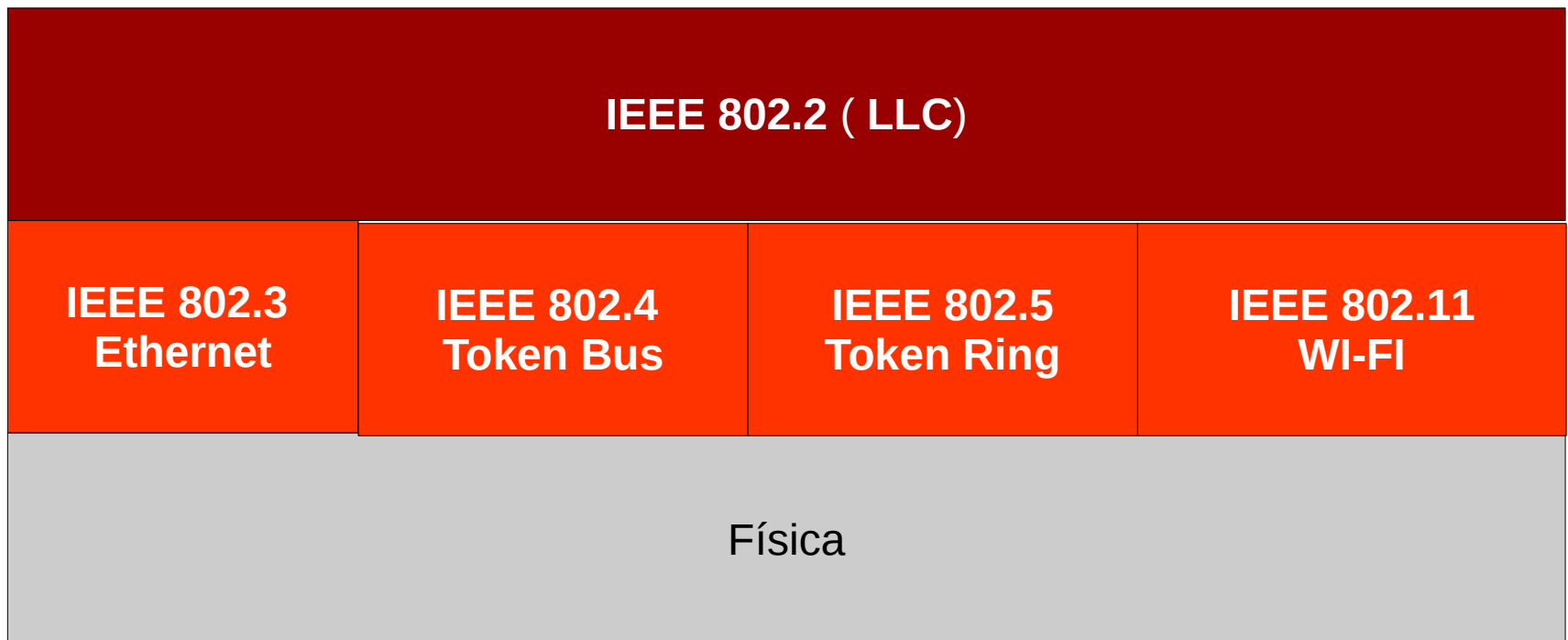
# Camadas

## Camada 2

---

O protocolo LLC ( IEEE 802.2) pode ser usado sobre todos os protocolos IEEE do subnível MAC, como por exemplo, o IEEE 802.3 (Ethernet), IEEE 802.4 (Token Bus) e IEEE 802.5 (Token Ring). Ele oculta as diferenças entre os protocolos do subnível MAC. Usa-se o LLC quando é necessário controle de fluxo ou comunicação confiável.





# Camadas

## Camada 3

---

Aplicação
Apresentação
Sessão
Transporte
<b>Rede</b>
Link de Dados ( Enlace )
Física

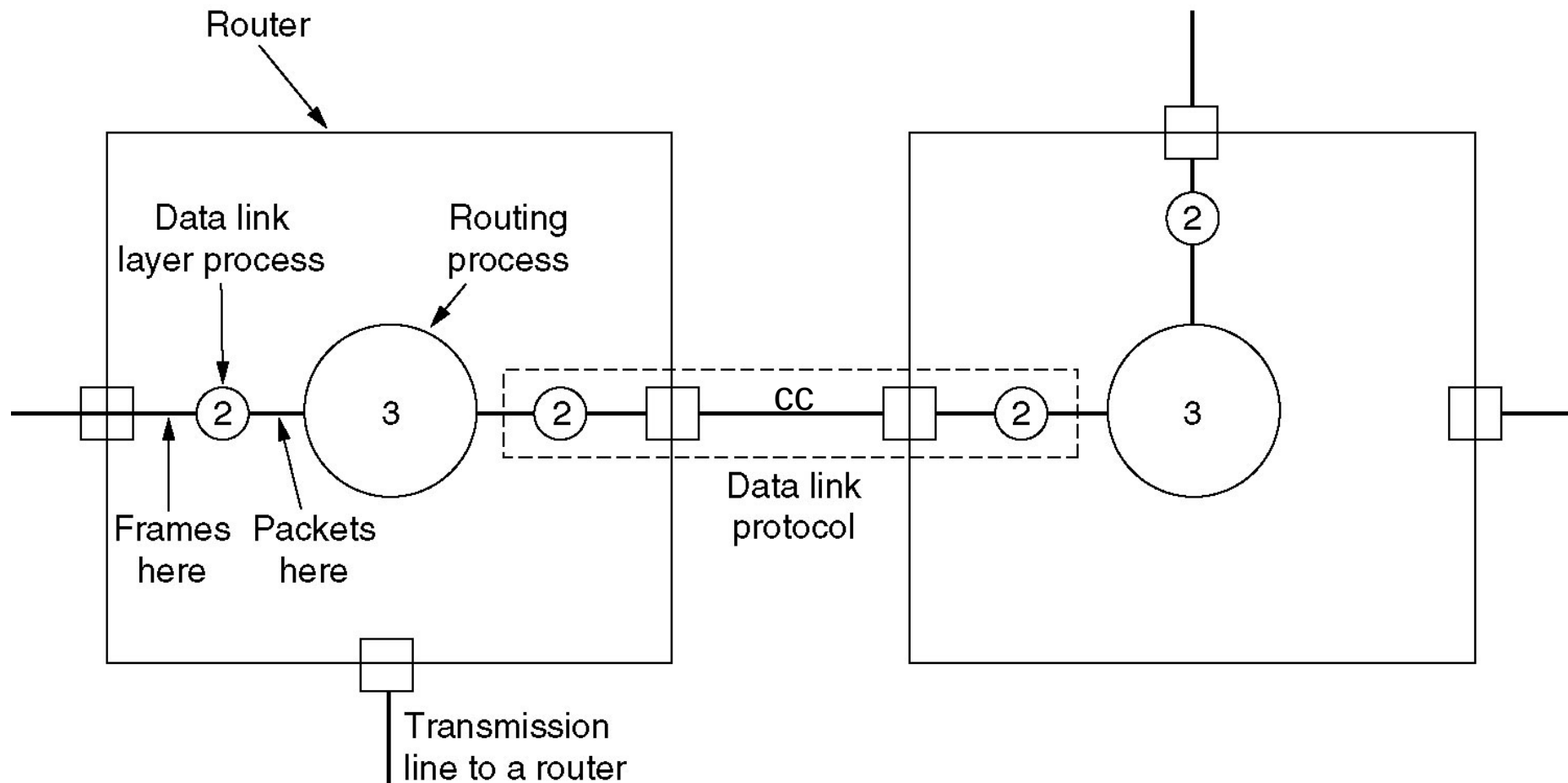
Funções:

Transportar pacotes da estação de origem até a estação de destino, onde quer que a origem e o destino estejam.

Lida com a transmissão fim a fim. Possivelmente atravessando múltiplas LANs e através de múltiplos roteadores.

Contraste com a camada de enlace de dados, onde ocorre a comutação dentro da mesma LAN ou LANs vizinhas.

Controlar o congestionamento



# Camadas

## Camada 3 - IP

---

0	4	8	16	19	24	31
Version	HLEN	Service Type	Total Length			
Identification			Flags	Fragment Offset		
Time to Live (TTL)		Protocol	Header Checksum			
Source IP Address						
Destination IP Address						
IP Options (if any)					Padding	

Data

# Camadas

## Camada 4

---



Funções:

Transferência de dados entre duas máquinas, independente da aplicação em uso.

Socket !

Principais protocolos: TCP e UDP

# Dúvidas

**Prof. Orlando Saraiva Júnior**  
**[orlando.saraiva@unesp.br](mailto:orlando.saraiva@unesp.br)**

O que são RFCs ?

Navegar no wikipedia e pesquisar:

- Camadas OSI
- Camadas TCP/IP
- Camada de Transporte

Quais as funções das camadas 5, 6 e 7 do modelo OSI ?