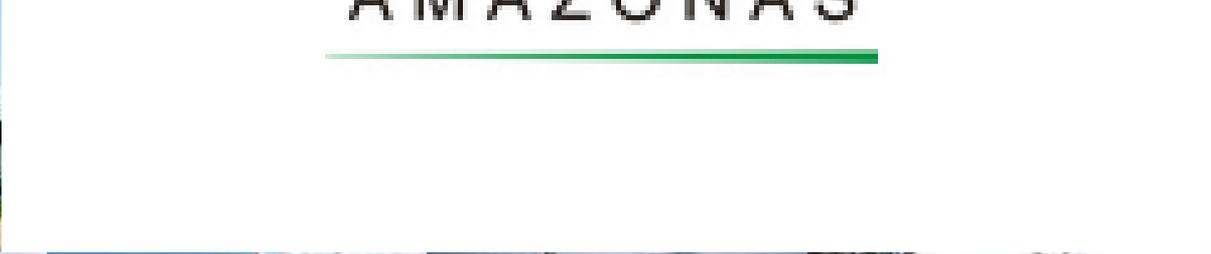




UEA

UNIVERSIDADE
DO ESTADO DO
AMAZONAS



**Escola Superior de Tecnologia – EST
Curso de Engenharia Elétrica
Redes de Comunicações de Dados I**

M.Sc. Bruno da Gama Monteiro

Manaus - AM
2021

Sumário

- 1 Hierarquias de Protocolos
- 2 Questões de Projeto Relacionadas às Camadas
- 3 Serviços Orientados e Não Orientados a Conexões
- 4 Primitivas de Serviço

- ❖ No projeto das primeiras redes de computadores, o *hardware* foi a principal preocupação e o *software* ficou em segundo plano.
- ❖ Atualmente, o *software* de rede é altamente estruturado
- ❖ Examinaremos com alguns detalhes a técnica de estruturação do *software*

Sumário

1 Hierarquias de Protocolos

2 Questões de Projeto Relacionadas às Camadas

3 Serviços Orientados e Não Orientados a Conexões

4 Primitivas de Serviço

1 Hierarquias de Protocolos

Para reduzir a complexidade do projeto, a maioria das redes é organizada como uma pilha de **camadas** ou **níveis**, colocadas umas sobre as outras. A quantidade de camadas, o nome, o conteúdo e a função difere de uma rede para outra.

O **OBJETIVO** de cada camada é oferecer determinados serviços as camadas superiores, isolando essas camadas dos detalhes de implementação desses recursos.

Cada camada é uma espécie de máquina virtual, oferecendo determinados serviços à camada acima dela.

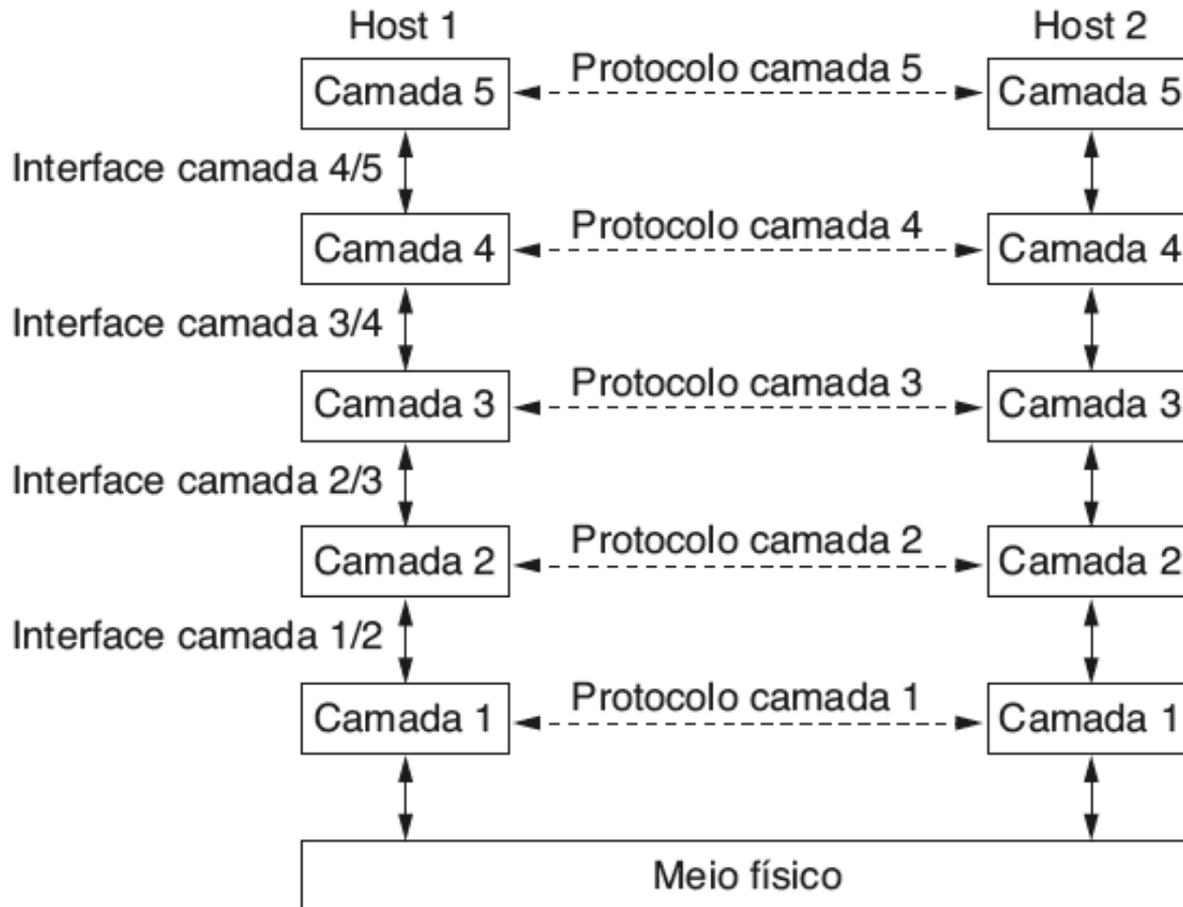
1 Hierarquias de Protocolos

A camada **n** de uma máquina se comunica com a camada **n** de outra máquina. Coletivamente, as regras e convenções usadas nesse dialogo são conhecidas como o **protocolo** da camada **n**.

Protocolo é um acordo entre as partes que se comunicam, estabelecendo como se dará a comunicação.

A violação do protocolo dificultará a comunicação, se não torna-la impossível.

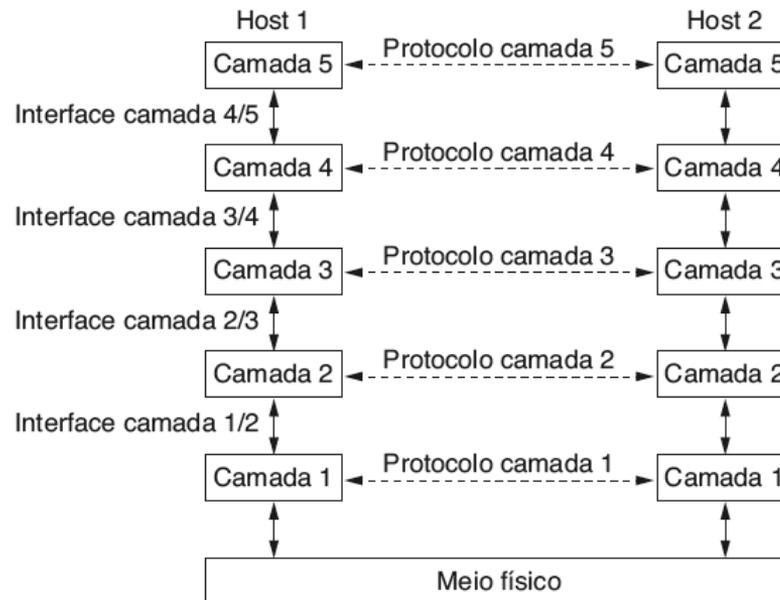
1 Hierarquias de Protocolos



Camadas, Protocolos e Interfaces.

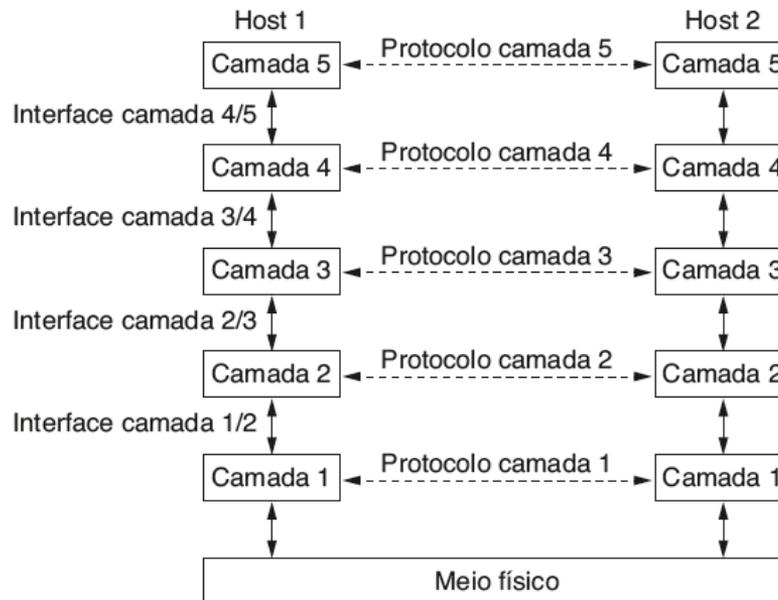
1 Hierarquias de Protocolos

- ❖ Entidades que ocupam camadas correspondentes em diferentes máquinas são pares (*peers*).
- ❖ Pares podem ser processos de *software*, dispositivos de *hardware* ou mesmo seres humanos.
- ❖ São os pares que se comunicam utilizando o protocolo.



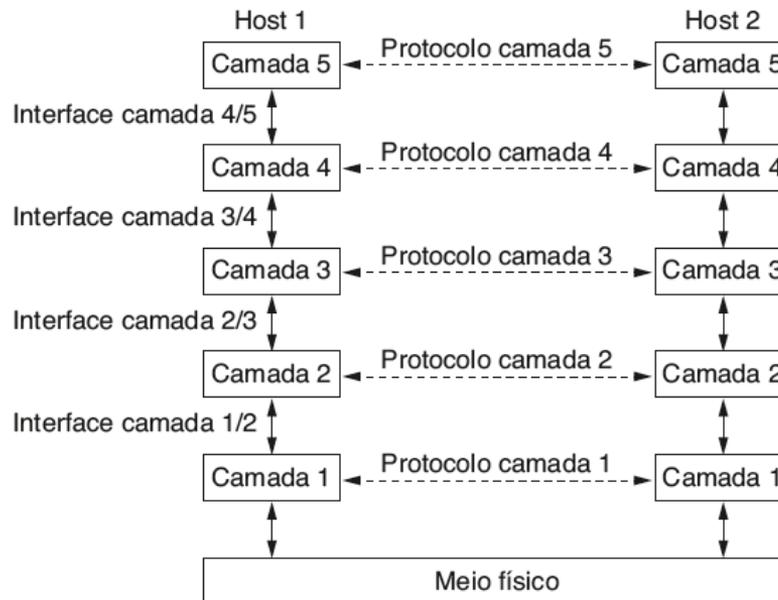
1 Hierarquias de Protocolos

- ❖ Dados não são transferidos diretamente da camada n de uma máquina para a camada n de outra máquina.
- ❖ Cada camada transfere os dados e as informações de controle para a camada imediatamente abaixo dela, até ser alcançada a camada mais baixa.



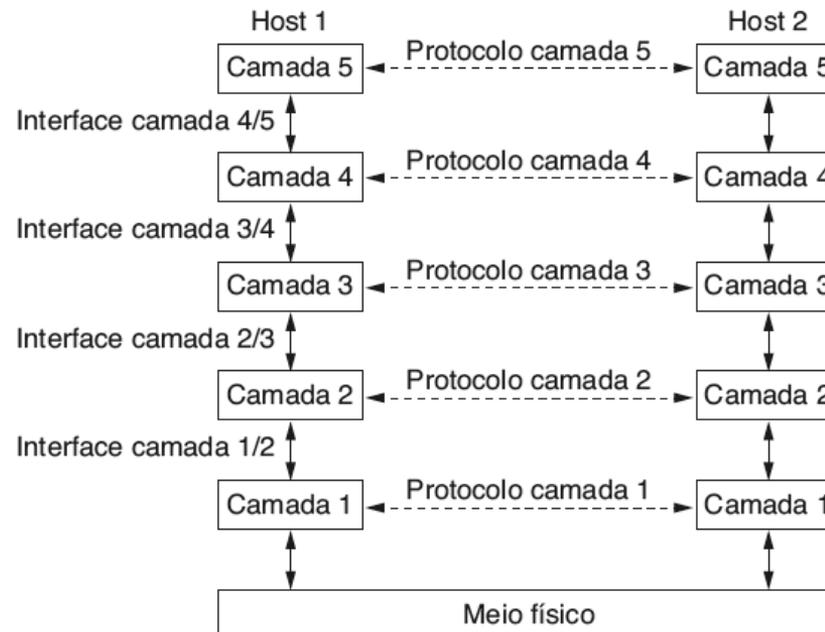
1 Hierarquias de Protocolos

- ❖ Abaixo da camada 1 encontra-se o **meio físico** através do qual se dá a comunicação propriamente dita.
- ❖ A comunicação virtual é mostrada por linhas pontilhadas e a comunicação física por linhas contínuas.



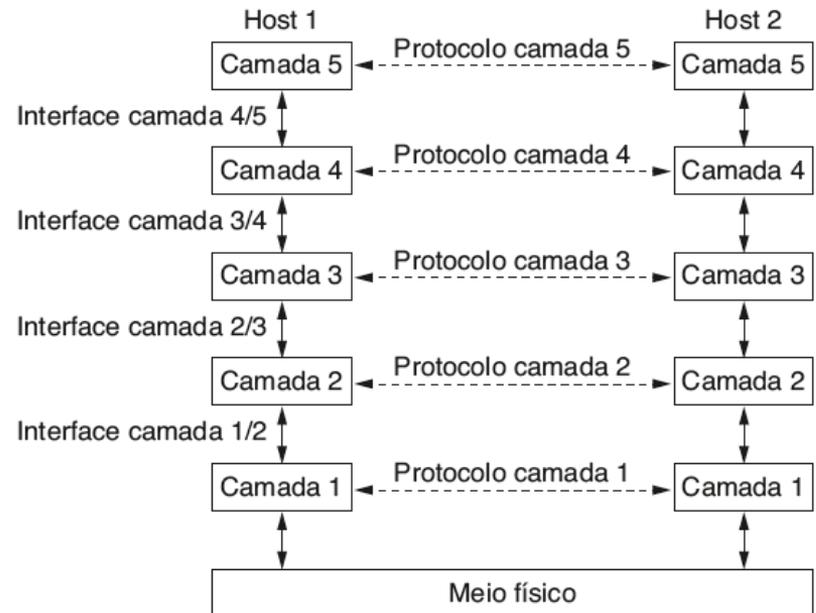
1 Hierarquias de Protocolos

- Entre cada par de camadas adjacentes existe uma interface que define as operações e os serviços que a camada inferior tem a oferecer a camada que se encontra acima dela.
- Não é necessário que as interfaces de todas as máquinas de uma rede sejam iguais, desde que cada uma delas possa usar todos os protocolos de maneira correta.



1 Hierarquias de Protocolos

- Uma lista de protocolos usados por um determinado sistema, um protocolo por camada, é chamada **pilha de protocolos**.
- **Arquitetura de rede** é o conjunto de camadas e protocolos.



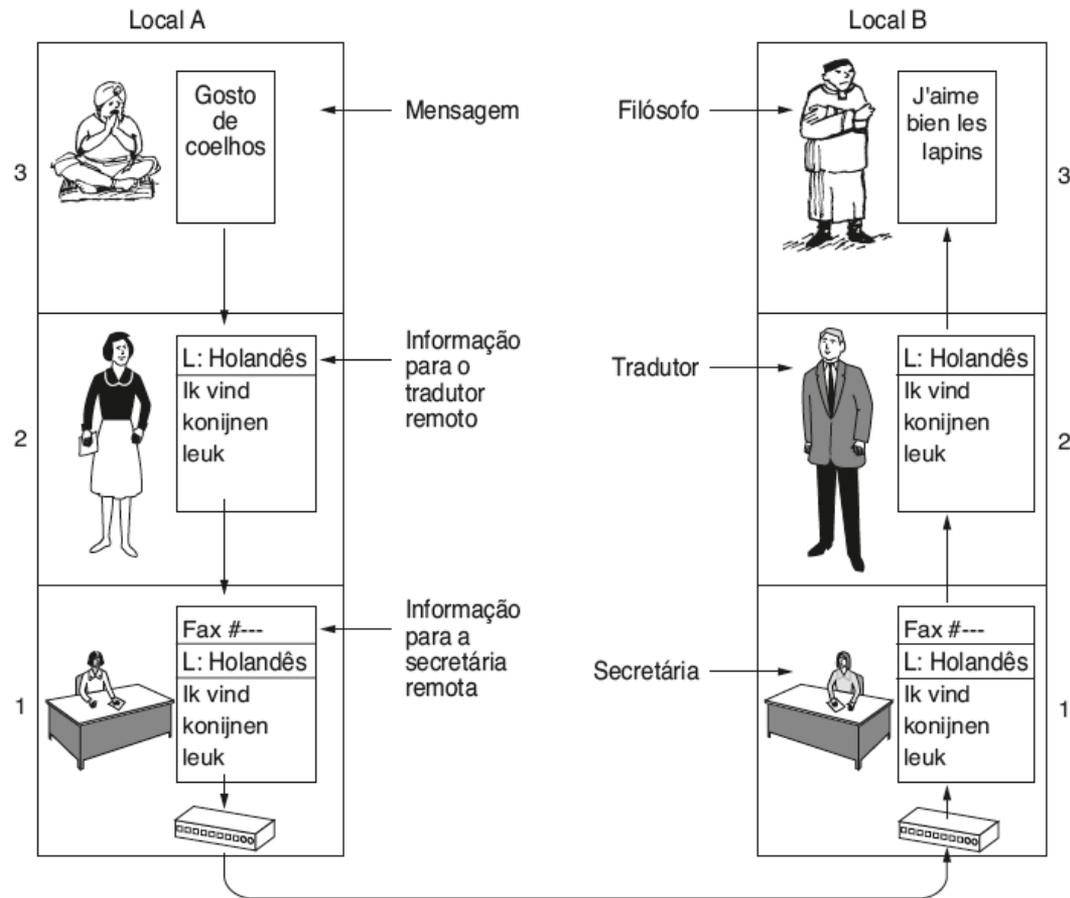
1 Hierarquias de Protocolos

Dois filósofos (processos pares na camada 3), um fala urdu e português e o outro fala chinês e francês.

Como não falam um idioma comum, eles contratam tradutores (processos pares na camada 2), que tem cada qual uma secretária (processos pares na camada 1).

O filósofo 1 deseja transmitir sua predileção por coelhos a seu par. Envia uma mensagem (em português) através da interface 2/3 a seu tradutor “Gosto de coelhos”.

Analogia – Comunicação em camadas



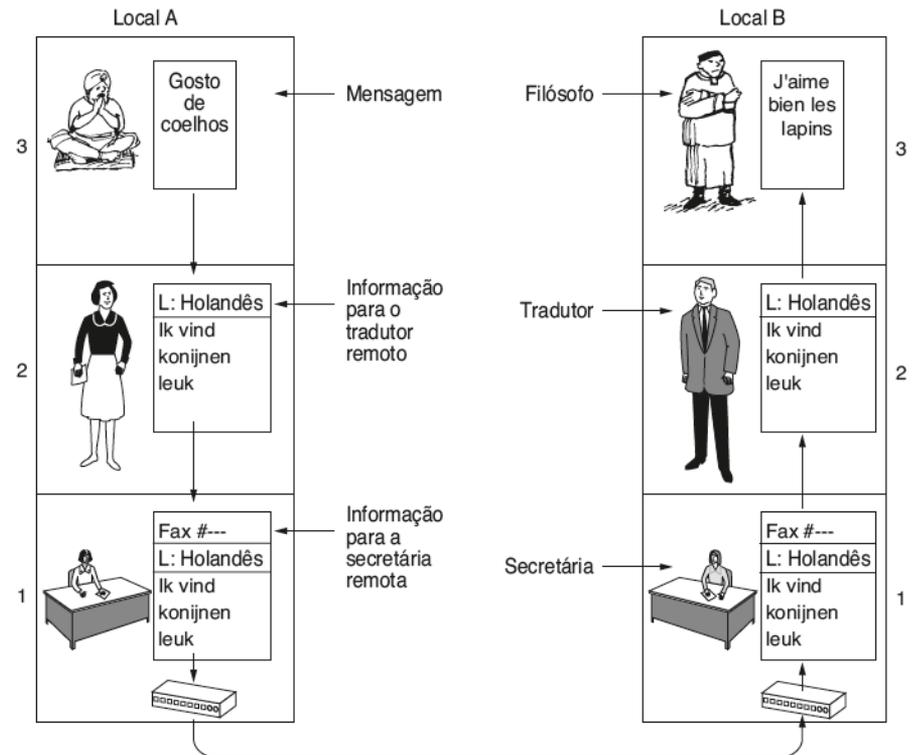
A arquitetura filósofo-tradutor-secretária.

1 Hierarquias de Protocolos

Tradutores resolveram usar um idioma neutro, o holandês, a mensagem foi convertida para 'Ik vind konijnen leuk'. A escolha do idioma é o protocolo da camada 2, que deve ser processada pelos pares dessa camada.

A tradutora entrega a mensagem a uma secretária para ser transmitida, por exemplo por fax (o protocolo da camada 1).

Quando chega, a mensagem é traduzida para o francês e passada através da interface 2/3 para o filósofo.



Sumário

1 Hierarquias de Protocolos

2 Questões de Projeto Relacionadas às Camadas

3 Serviços Orientados e Não Orientados a Conexões

4 Primitivas de Serviço

2 Questões de Projeto Relacionadas às Camadas

- ❖ Primeira **questão de projeto** é a **Confiabilidade**: criar uma rede que opere corretamente, embora sendo composta de uma coleção de componentes que não são confiáveis.

Exemplos: bits de um pacote trafegando pela rede podem ser recebidos com problemas devido ao ruído, sinais sem fio aleatórios, falhas de *hardware*, *bugs* de *software* etc...

- ❖ Mecanismo para localizar erros na informação recebida usa códigos para **detecção de erros**.

2 Questões de Projeto Relacionadas às Camadas

- ❖ Informações recebidas incorretamente só serão retransmitidas quando forem recebidas corretamente.
- ❖ Códigos mais poderosos permitem a correção de erro, a mensagem correta é recuperada a partir de bits possivelmente incorretos que foram originalmente recebidos.
- ❖ Mecanismos da **detecção e correção de erros** acrescentam informações redundantes, são usados nas camadas mais baixas, protegendo pacotes enviados por enlaces individuais e nas camadas altas, para verificar se o conteúdo correto foi recebido.

2 Questões de Projeto Relacionadas às Camadas

- ❖ Outra questão de **confiabilidade**: descobrir um caminho que funcione através de uma rede. Existem vários caminhos entre origem e destino e pode haver enlaces e/ou roteadores com defeito.
- ❖ A rede deve tomar a decisão automaticamente: **roteamento**.
- ❖ **Segunda questão de projeto**: se refere à **evolução da rede**. As redes se tornam maiores e novos projetos aparecem precisando ser conectados à rede existente.

2 Questões de Projeto Relacionadas às Camadas

- ❖ Terceira questão de projeto é a **alocação de recursos**.
- ❖ Redes precisam de mecanismos que dividem seus recursos de modo que um *host* não interfira no outro:
Ex: **Multiplexação estatística** – compartilha a largura de banda da rede dinamicamente com base nas estatísticas de demanda.
- ❖ **Controle de fluxo**: Normalmente usa-se uma espécie de ***feedback*** do receptor para o transmissor para impedir que um transmissor rápido envie uma quantidade excessiva de dados a um receptor mais lento.

2 Questões de Projeto Relacionadas às Camadas

- ❖ Redes tem mais a oferecer do que simplesmente largura de banda.
- ❖ Para transporte de vídeo ao vivo (tempo real) a prontidão na entrega é muito importante, enquanto a rede também precisa oferecer serviços que desejam uma alta vazão.
- ❖ **Qualidade do serviço:** mecanismo que reconcilia essas demandas concorrentes.

2 Questões de Projeto Relacionadas às Camadas

- ❖ Última questão de projeto: **proteção da rede** defendendo-a contra diferentes tipo de ameaças.
- ❖ Mecanismos que oferecem **confidencialidade** defendem contra a bisbilhotagem nas comunicações.
- ❖ Mecanismos de **autenticação** impedem que alguém finja ser outra pessoa.

Sumário

- 1 Hierarquias de Protocolos
- 2 Questões de Projeto Relacionadas às Camadas
- 3 Serviços Orientados e Não Orientados a Conexões**
- 4 Primitivas de Serviço

Camadas podem oferecer dois tipos diferentes de serviços as camadas situadas acima delas: **serviços orientados a conexões** e **serviços sem conexões**.

- ❖ **Serviço orientado a conexões** se baseia no sistema telefônico. Para falar com alguém, você tira o fone do gancho, discar o número, fala e, em seguida, desliga. Da mesma forma, para utilizar um serviço de rede orientado a conexões, primeiro o usuário do serviço estabelece uma conexão, utiliza a conexão, e depois libera a conexão.

- ❖ O aspecto essencial de uma **conexão** é que ela funciona como um tubo: transmissor empurra objetos (bits) em uma extremidade, e esses objetos são recebidos pelo receptor na outra extremidade.
- ❖ Em alguns casos, quando uma conexão é estabelecida, o transmissor, o receptor e a sub-rede conduzem uma **negociação** sobre os parâmetros a serem usados, como o tamanho máximo das mensagens, a qualidade do serviço exigida e outras questões.

- ❖ **Circuito** é outro nome para uma conexão com recursos associados, como uma largura de banda fixa. Isso vem desde a rede telefônica, em que um circuito era caminho pelo fio de cobre que transportava uma conversa telefônica.

❖ **Serviço sem conexão** se baseia no sistema postal.

Cada mensagem (carta) carrega o endereço de destino completo e cada uma delas é roteada (encaminhada) através do sistema, independentemente de todas as outras.

Em geral, quando duas mensagens são enviadas ao mesmo destino, a primeira a ser enviada é a primeira a chegar.

- ❖ Cada serviço pode ser caracterizado por uma qualidade de serviço. Alguns serviços são confiáveis, no sentido de nunca perderem dados.
- ❖ Em geral, um serviço confiável é implementado para que o receptor confirme o recebimento de cada mensagem, de modo que o transmissor se certifique de que ela chegou. Uma situação típica em que um serviço orientado a conexões confiável é apropriado é a transferência de arquivos.

3 Serviços Orientados e Não Orientados a Conexões

	Serviço	Exemplo
Orientados a conexões	Fluxo de mensagens confiável	Sequência de páginas
	Fluxo de bytes confiável	Download de filme
	Conexão não confiável	VoIP
Sem conexões	Datagrama não confiável	Lixo de correio eletrônico
	Datagrama confirmado	Mensagem de texto
	Solicitação/resposta	Consulta a banco de dados

Seis diferentes tipos de serviços.

Obs: Um **datagrama** é uma unidade de transferência básica associada a uma **rede** de comutação de pacotes em que a entrega, hora de chegada, e a ordem não são garantidos.

- ✓ **Serviço orientado a conexões** confiável tem duas variações secundárias: **sequência de mensagens** e **fluxos de bytes**.
- ✓ **Sequência de mensagens** - Quando duas mensagens de 1.024 bytes são enviadas, chegam como duas mensagens distintas de 1.024 bytes.
- ✓ **fluxos de bytes** – sem limite de mensagens. Quando 2.048 bytes chegam ao receptor, não se sabe se foram enviados como uma mensagem de 2.048 bytes, duas mensagens de 1.024 bytes ou 2.048 mensagens de 1 byte.

	Serviço	Exemplo
Orientados a conexões	Fluxo de mensagens confiável	Sequência de páginas
	Fluxo de bytes confiável	Download de filme
	Conexão não confiável	VoIP
Sem conexões	Datagrama não confiável	Lixo de correio eletrônico
	Datagrama confirmado	Mensagem de texto
	Solicitação/resposta	Consulta a banco de dados

- ✓ **Tráfego de voz digital Voice Over IP (VoIP)** – os atrasos são inaceitáveis, usuários preferem ouvir um pouco de ruído na linha ou uma palavra truncada de vez em quando a experimentar um atraso para aguardar confirmações.
- ✓ **Videoconferência** – alguns pixels errados são toleráveis, porém uma imagem parada enquanto o fluxo é interrompido e reiniciado para a correção de erros é irritante.

	Serviço	Exemplo
Orientados a conexões	Fluxo de mensagens confiável	Sequência de páginas
	Fluxo de bytes confiável	Download de filme
	Conexão não confiável	VoIP
Sem conexões	Datagrama não confiável	Lixo de correio eletrônico
	Datagrama confirmado	Mensagem de texto
	Solicitação/resposta	Consulta a banco de dados

➤ **Datagramas confirmados**

exemplo em que não há necessidade de estabelecer uma conexão para enviar uma única mensagem curta com uma confiabilidade.

Semelhante a enviar uma carta registrada e receber um aviso de recebimento.

Ao receber o aviso, o transmissor fica certo de que a mensagem foi entregue ao destinatário.

	Serviço	Exemplo
Orientados a conexões	Fluxo de mensagens confiável	Sequência de páginas
	Fluxo de bytes confiável	Download de filme
	Conexão não confiável	VoIP
Sem conexões	Datagrama não confiável	Lixo de correio eletrônico
	Datagrama confirmado	Mensagem de texto
	Solicitação/resposta	Consulta a banco de dados

➤ Solicitação/Resposta

Transmissor envia um único datagrama contendo uma solicitação; a resposta contém a réplica.

Usada para a comunicação cliente-servidor. Cliente emite uma solicitação servidor responde.

Ex: cliente do smartphone consulta servidor de mapas para receber os dados de mapa do local atual.

	Serviço	Exemplo
Orientados a conexões	Fluxo de mensagens confiável	Sequência de páginas
	Fluxo de bytes confiável	Download de filme
	Conexão não confiável	VoIP
Sem conexões	Datagrama não confiável	Lixo de correio eletrônico
	Datagrama confirmado	Mensagem de texto
	Solicitação/resposta	Consulta a banco de dados

Se existe comunicação confiável, qual o motivo de alguma aplicação preferir utilizar uma comunicação não confiável?

A comunicação confiável (confirmada) pode não estar disponível. Exemplo: a **Ethernet** não fornece comunicação confiável. Ocasionalmente, os pacotes podem ser danificados em transito.

Os retardos provenientes de um serviço confiável podem ser inaceitáveis, em especial nas aplicações de tempo real como as de multimídia.

4 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, como ocorre com frequência, as primitivas serão normalmente chamadas do sistema.

4 Primitivas de Serviço

Primitivas para um **serviço orientado a conexões** são **diferentes** das que são oferecidas em um **serviço sem conexões**.

Exemplo mínimo das **primitivas de serviço (operações)** que poderiam ser fornecidas para implementar um fluxo de bytes confiável em um sistema cliente/servidor

Primitiva	Significado
LISTEN	Bloco que espera por uma conexão de entrada
CONNECT	Estabelecer uma conexão com um par que está à espera
ACCEPT	Aceitar uma conexão de entrada de um par
RECEIVE	Bloco que espera por uma mensagem de entrada
SEND	Enviar uma mensagem ao par
DISCONNECT	Encerrar uma conexão

Seis primitivas de serviços que fornecem um serviço orientado para conexões.

4 Primitivas de Serviço

Exemplo de um protocolo simples que implementa o serviço usando **datagramas confirmados**:

I - Primeiro, o servidor executa **LISTEN** para indicar que esta preparado para aceitar conexões de entrada. Depois de executar a primitiva, o processo servidor fica bloqueado ate surgir uma solicitação de conexão.

II - Em seguida, o processo cliente executa **CONNECT (1)** para estabelecer uma conexão com o servidor. A chamada de **CONNECT** precisa especificar a quem se conectar; assim, ela poderia ter um parâmetro fornecendo o endereço do servidor.

4 Primitivas de Serviço

III – Quando o pacote chega ao servidor, o sistema operacional vê que o pacote está solicitando uma conexão. Verifica se existe um ouvinte e, se houver, desbloqueia o ouvinte. O servidor, então estabelece uma conexão com a chamada **ACCEPT** (2), enviando de volta uma confirmação ao processo cliente para aceitar a conexão.

IV - Em seguida, o processo cliente executa **CONNECT** para estabelecer uma conexão com o servidor. A chamada de **CONNECT** precisa especificar a quem se conectar; assim, ela poderia ter um parâmetro fornecendo o endereço do servidor.

4 Primitivas de Serviço

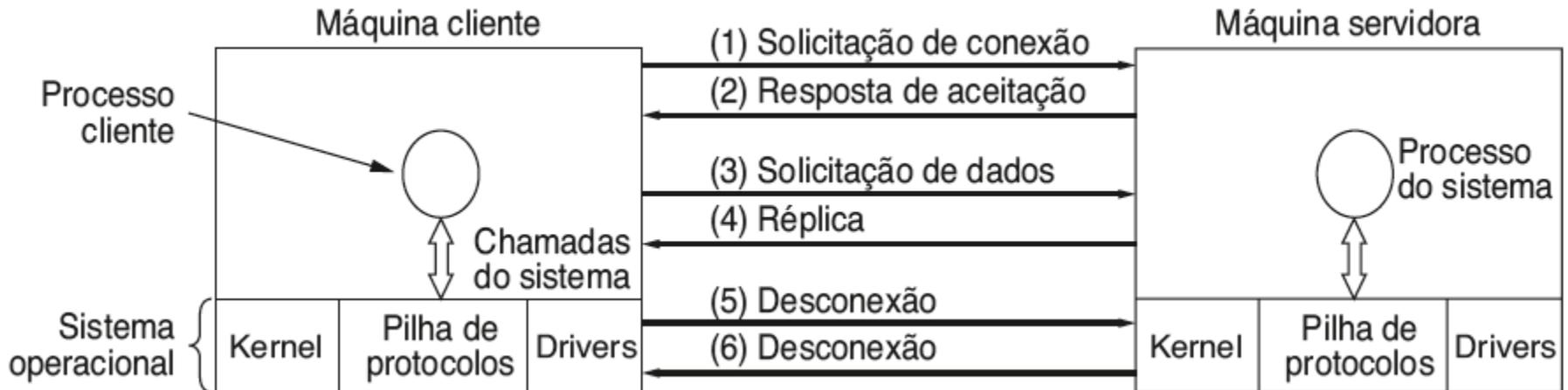
V –O servidor executa **RECEIVE** a fim de se preparar para aceitar a primeira solicitação. Imediatamente após ser liberado de **LISTEN**. A chamada de **RECEIVE** bloqueia o servidor.

VI –Cliente executa **SEND** (3) para transmitir sua solicitação, seguida pela execução de **RECEIVE** (4) para receber a resposta.

VII – Para finalizar, o cliente utiliza **DISCONNECT** (5) para encerrar a conexão. Servidor recebe o pacote e emite uma **DISCONNECT** (6).

4 Primitivas de Serviço

Figura que resume o funcionamento possível de uma comunicação cliente-servidor com datagramas confirmados, de modo que podemos ignorar os pacotes perdidos.



Uma simples interação cliente-servidor usando datagramas de reconhecimento

Referências

TANENBAUM, Andrew. S; Wetherall, David. **Redes de Computadores**. 5.ed. Traduzido por Daniel Vieira. São Paulo: Pearson Prentice Hall, 2011. Tradução de: Computer networks.

FIM

Básica:

- 1 STALLINGS, William, Data and computer communications – Prentice Hall – 1997.
- 2 KUROSE, James F., Redes de computadores e a internet. Ed. São Paulo, 2006.
- 3 TANENBAUM, André S., rede de Computadores, Ed. Campos, 2003.

Complementar:

- 4 SOUZA, Lindeberg Barros, Redes de Computadores: Dados, Voz, Imagem, Ed. Érica, 2005.
- 5 SOARES, Luiz Fernando Gomes, Rede de Computadores: Das redes LANs, MANs e WANs Redes ATM – Ed. Campos, 1995
- 6 SOARES, Neto Vicente. Telecomunicação – Redes de alta velocidade – Ed. Érica, 2005.