

Redes de Computadores

Camada de Aplicação

Aplicação e Protocolo de Aplicação

Aplicação: processos distribuídos em comunicação

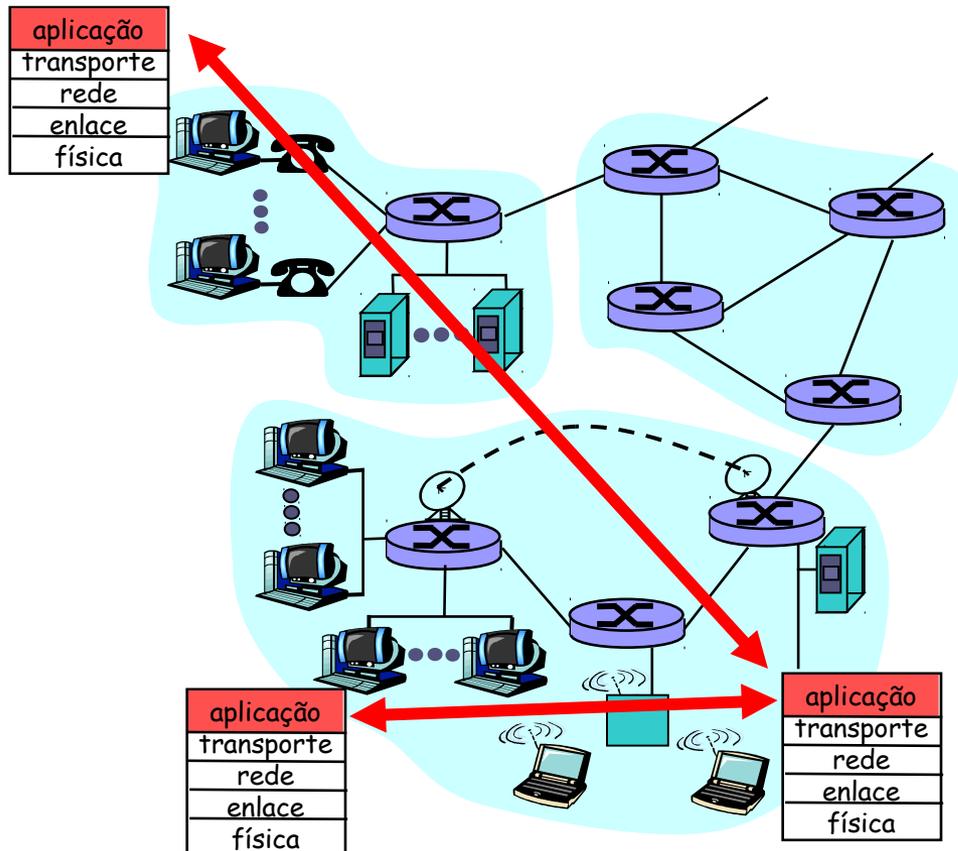
- rodam nos computadores usuários da rede como programas de usuário
- trocam mensagens para realização da aplicação
- e.x., email, ftp, Web

Protocolos de aplicação

- fazem parte das aplicações
- definem mensagens trocadas e as ações tomadas
- usam serviços de comunicação das camadas inferiores

Rede de Computadores

Aplicação e Protocolo de Aplicação



Arquiteturas de Aplicações de Redes

● Arquitetura Cliente-Servidor

- **Servidor:** sempre em funcionamento, atende as requisições dos clientes, endereço fixo;
- **Cliente:** pode estar em funcionamento ou não, requisita algo ao servidor, geralmente endereço variável, geralmente inicia a comunicação;
 - **Server Farm:** um grupo de servidores trabalhando em conjunto, para atender a demanda dos clientes como um único grande servidor.

Arquiteturas de Aplicações de Redes

● Arquitetura P2P

- Não há um servidor sempre funcionando;
- Pares de hospedeiros (*peers*) comunicam-se diretamente entre si;
- Possui alta escalabilidade;
- Pelo fato de ser distribuído e descentralizado, não há garantias de um arquivo (por exemplo) ser encontrado num exato momento;
- Ex.: rede Gnutella.

Arquiteturas de Aplicações de Redes

● Arquitetura Híbrida (P2P e Cliente-Servidor)

- Possui características das duas arquiteturas anteriores;
- Possui um controle feito através de servidor;
- A comunicação é feita entre pares;
- Ex: Napster (falecido), MSN.

Comunicação de Aplicações

- Quem se comunica não é a aplicação é um **processo**.
- Processo: programa executado num host.
 - dentro do mesmo host: comunicação interprocessos;
 - processos sendo executados em diferentes hosts se comunicam através da troca de mensagens pela rede.
- Numa aplicação de rede temos um par de processos, um encarregado de enviar mensagens requisitando algo (cliente) e outro para a recepção e resposta destas requisições (servidor).
- “No contexto de uma sessão de comunicação entre um par de processos, o processo que inicia a comunicação (o primeiro a contatar o outro no início da sessão) é rotulado de **cliente**. O processo que espera ser contactado para iniciar a sessão é o **servidor**.”

SOCKETS

- Uma mensagem enviada de um processo para outro, em hosts distintos, tem de passar pela rede.
- Um processo recebe e envia mensagens para rede através de seu socket.
- **Socket é a interface entre a camada de aplicação e a camada de transporte.**
- Socket também denominada **Interface de Programação da Aplicação (*Application Programming Interface – API*)**, através da qual uma aplicação utiliza a rede para enviar e receber mensagens.

Endereçamento entre Processos

- Uma mensagem que é enviada de um host para outro tem que identificar, além do endereço do sistema final (IP), o endereço do processo destino.
- É utilizada o **número de porta** de destino para identificar o processo.
- Ex: 80 – HTTP e 25 – SMTP.