

# Sistemas Cliente-Servidor

---

Disciplina Bancos de Dados II (INE 5616 – 2006-1)

Curso de Sistemas de Informação

**Prof. Renato Fileto**

**INE/CTC/UFSC**

# Cliente - Servidor

---

## Arquitetura cliente/servidor:

- Os **servidores** oferecem serviços a processos usuários, ou seja, executam a tarefa solicitada e enviam uma resposta ao cliente que se traduz nos dados solicitados.
- Os **clientes** solicitam um determinado serviço, através do envio de uma mensagem ao servidor. Enquanto o processo servidor está trabalhando a solicitação, o cliente está livre para realizar outras tarefas.

# Cliente

---

- O processo cliente realiza o envio das mensagens fazendo algum pedido ao servidor.
- O cliente é a parte que interage com o usuário, possui a interface que o usuário utiliza para requisitar as tarefas ao servidor, sendo chamado de **front-end** da aplicação.
- Os processos clientes gerenciam as atividades dos usuários e realizam as validações dos dados informados por estes.

# Servidor

- É o processo que responde a uma mensagem solicitando a realização de alguma tarefa por parte do cliente. O processo servidor é chamado de **back-end**.
- O processo servidor pode oferecer serviços a muitos clientes, realiza pesquisas, filtragens e atualizações em bancos de dados.
- Os serviços podem ser realizados diretamente pelo processo servidor ou através de processos escravos criados por este para atender cada pedido do cliente, o que libera o processo mestre do servidor para receber outras solicitações.

# Transparência de Localização

---

Um aspecto importante nos sistemas cliente/servidor:

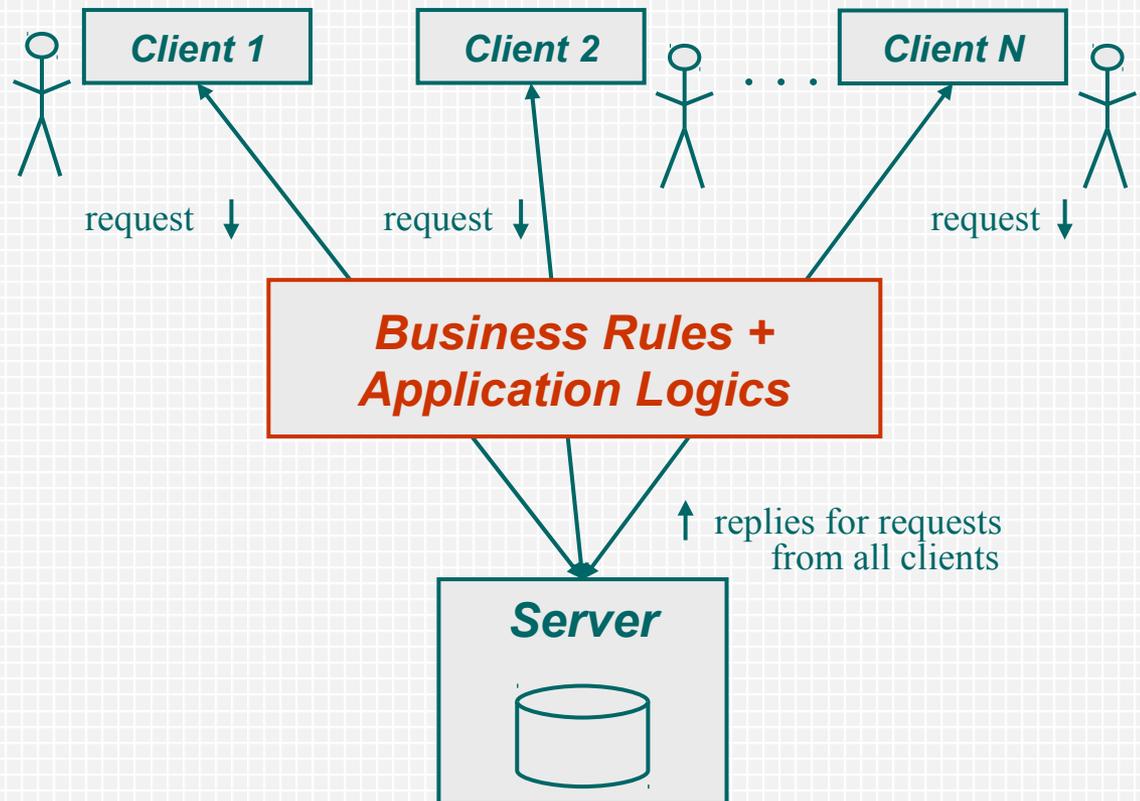
**Transparência** - para o usuário não deve existir diferença entre acessar um recurso local ou remoto, não deve ser motivo de preocupação para o usuário a localização do servidor e a natureza da comunicação.

# Camadas de Cliente-Servidor

- **Arquitetura cliente/servidor** - forma como os aplicativos são estruturados.
- **Duas variantes:**

- **Two-Tier**

- **Three-Tier**



# Camadas de Cliente-Servidor

**Two-Tier:** organização em duas camadas - o cliente comunica-se diretamente com o servidor.

- A base de dados fica no servidor.
- As regras e a lógica da aplicação no cliente.
- Problema de manutenção - toda vez que uma aplicação for alterada, tanto bancos de dados como aplicações clientes precisam ser alteradas.
- A aplicação cliente precisa ser instalada em todos os nós.

# Camadas de Cliente-Servidor

---

**Three-Tier:** uma camada intermediária é criada entre o servidor e o cliente.

- A função da camada intermediária é armazenar as regras do negócio e a lógica da aplicação.
- O cliente fica responsável apenas pela interface com o usuário.
- Qualquer alteração na camada intermediária é imediatamente assumida por todas as aplicações e pelo banco de dados.

# Características do Cliente

---

- Interface gráfica
- Validação da entrada de dados
- Possibilidade de impressão local ou remota
- Capacidade de detectar quando uma solicitação feita não chega ao servidor
- Exigência de investimentos em equipamentos para suportar interfaces gráficas sofisticadas
- Dados não estão disponíveis localmente; se houver replicação local, a consistência deve ser mantida!

# Características do Servidor

- Oferece serviços (e.g., busca e atualizações no banco de dados) para muitos clientes, centralizando a manipulação de dados, manutenção de consistência e integridade
- Deve ter capacidade de processamento, acesso a disco, memória e comunicação para atender os pedidos dos vários clientes
- Controle de concorrência (e.g., utilizando exclusão mútua no acesso a dados)
- Segurança: controle de acesso via senhas, a diversos níveis de informação
- Confiabilidade: a centralização facilita o gerenciamento do backup
- Caso falhe o servidor, todos os serviços são paralizados

# Comunicação

---

- A **comunicação** entre o cliente e o servidor é **transacional e cooperativa**.
- A característica **transacional** indica que **o servidor envia apenas os resultados relevantes do pedido do cliente**. Assim, a **quantidade de dados transferida tende a ser a menor** quantidade necessária para o cliente executar seu trabalho.
- A natureza **cooperativa** do paradigma cliente/servidor significa que ocorre um **processamento significativo e colaborativo nos extremos cliente e servidor**.

# Comunicação (II)

---

- Sistemas Cliente-Servidor **dispensam um mecanismo especial para sincronizar os componentes (clientes e servidor(s))**; a natureza da passagem de mensagens elimina a necessidade de um mecanismo explícito.
- A comunicação cliente/servidor é baseada em **"Chamadas de Procedimento Remoto" (Remote Procedure Calls - RPC)**.

# Comunicação (III)

---

- Minimização do tráfego
- Não dispensa protocolos de mais baixo nível para checagem da transmissão de pacotes
  - Para evitar erros nos dados
  - Para evitar que o cliente ou servidor trabalhem sem necessidade
- Compartilhamento de periféricos

# Balanceamento das Aplicações

- **“Clientes Gordos”**: regras de negócio e lógica da aplicação concentradas no lado cliente

*Exemplo extremo: servidores de arquivos*

→ Altos custos de transmissão de dados

- **“Servidores Gordos”**: regras de negócio e lógica da aplicação concentradas no lado servidor

*Exemplo extremo: mainframes com terminais*

→ Interação homem-máquina, muito pobre e dependente de comunicação constante com o servidor

# Desvantagens de Cliente-Servidor

(em relação a sistemas centralizados)

---

- Aumento do custo do hardware, no servidor e no cliente
- Custo do software, incluindo servidor de banco de dados, aplicativos, ferramentas de desenvolvimento e administração
- Custo administrativo e de pessoal de suporte, incluindo treinamento
- Dificuldades de configuração e identificação/análise de problemas, devido à complexidade

# Cliente - Servidor

---

Está claro pra você?

Veja com clareza como é  
a nova era dos negócios.