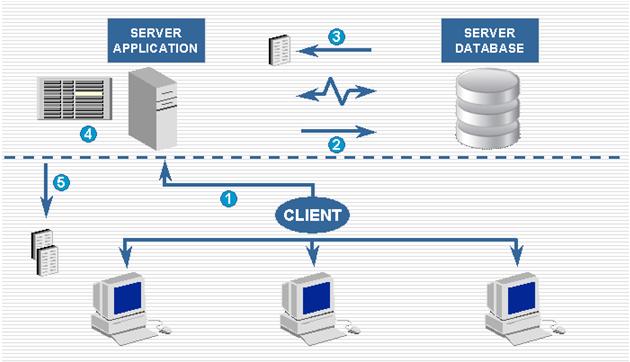
**Arquitetura Client/Server e Three-tier**



Muitos desenvolvedores dizem que seus sistemas são baseados em arquitetura Client/Server ou Three-Tier. Então vamos entender o que são essas arquiteturas e se realmente estão sendo utilizadas no dia-a-dia.

A filosofia Client/Sever vêm da idéia de deixar para o servidor todo o processamento, retirando assim da máquina do cliente essa tarefa que muitas vezes é tão pesada.

A questão é que muitos profissionais acreditam que só pelo fato do Banco de Dados estar centralizado em um servidor dedicado e todo mundo acessando esse servidor ele está utilizando essas arquiteturas.

O fato do banco de dados estar isolado em um servidor e através das máquinas Client você acessar os dados que estão nesse servidor não quer dizer que você esteja **utilizando arquitetura Client/Server ou Three-Tier**.

O Servidor com certeza tem ‘N’ processadores, ‘N’ GB de memória e um poder de processamento muito superior ao Client que tem apenas um processador e talvez 512mb de memória.

Só pode dizer que está sendo utilizado essas arquiteturas quando você começar a retirar da parte do Client todo o processamento, ou seja, você começar a trabalhar com Stored Procedure que ficam no servidor de banco de dados e/ou Componentes que ficam no servidor de aplicativo.

Vejamos agora a figura abaixo como é realizado o caminho em uma Arquitetura Three-Tier desde um simples clique em botão “Gravar” do seu Front-End até o retorno com a mensagem “Dados gravados com sucesso”.

1. Quando o cliente clica no botão “Gravar” o seu aplicativo envia uma solicitação para o sistema operacional da máquina do Client. Com isso o sistema operacional realiza o empacotamento e solicita para a rede levar esse pacote até o servidor de componente.
2. O sistema operacional do Server Application recebe o pacote, abre e identifica que é uma requisição para o Serviço de Componente. O componente (programa executável que foi desenvolvido por você) então é acionado e solicita informações para o [**banco de dados**](https://www.devmedia.com.br/conceitos-fundamentais-de-banco-de-dados/1649). Mais uma vez o sistema operacional é acionado, empacota a requisição e solicita para rede para levar esse pacote até o Server Database.
3. Nesse instante o [**SGBD**](https://www.devmedia.com.br/arquitetura-de-um-sgbd/25007) é acionado, abre uma conexão, realiza todo seu trabalho de verificar o parse do comando, checar se a informação está na memória e então executa a solicitação. Quando tem o retorno ele devolve para quem o solicitou, que nesse caso foi o Server Application.
4. É nesse instante que o componente recebe os dados ele realiza todo processamento “pessado” (regras de negócios, cálculos, enfim).
5. Após o término do processamento o componente devolve para o Client que fez a solicitação.

#### Conclusão

Utilizando essas arquiteturas sua aplicação ficará muito mais rápida, pois você estará retirando da máquina do Client todo processamento e deixando assim ela leve para o usuário final utilizar todo potencial do seu desktop.

Fonte: https://www.devmedia.com.br/arquitetura-client-server-e-three-tier/5865