



Arquitetura de SGBD

Prof. Antonio Almeida de Barros Junior



Agenda

- Caracterização de SGBDs
- SGBDs Centralizados
- SGBDs Cliente-Servidor
- SGBDs Distribuídos Homogêneos
- Multi-SGBDs Heterogêneos
- SGBDs Paralelos
- SGBDs e a Web
- SGBDs para Estações Móveis



Arquitetura de Sistema de Banco de Dados

A arquitetura de um sistema de banco de dados é bastante influenciada pelo **sistema de computador** básico em que ela trabalha, em particular, por aspectos da arquitetura de computador como redes, paralelismo e distribuição.



Caracterização de SGBDs

- Quanto ao Modelo de Dados:
 - relacional
 - orientado-a-objeto
 - semi-estruturado
 - outros



Caracterização de SGBDs

- Quanto à Arquitetura:
 - centralizada
 - cliente-servidor
 - distribuída
 - paralela
 - extensível
 - outras



Caracterização de SGBDs

- Quanto às características dos dados ou da aplicação:
 - dados clássicos
 - texto/documentos
 - multimídia
 - geográfico
 - outros

Caracterização de SGBDs

- Características das Arquiteturas:
 - Distribuição do Controle (Autonomia Local):
 - A0. sistemas totalmente integrados
 - A1. sistemas semi-autônomos
 - A2. sistemas autônomos
 - Distribuição dos Dados:
 - D0. sistemas centralizados
 - D1. sistemas cliente-servidor
 - D2. sistemas distribuídos (peer-to-peer systems)
 - Heterogeneidade:
 - H0. sistemas homogêneos
 - H1. sistemas heterogêneos

Caracterização de SGBDs

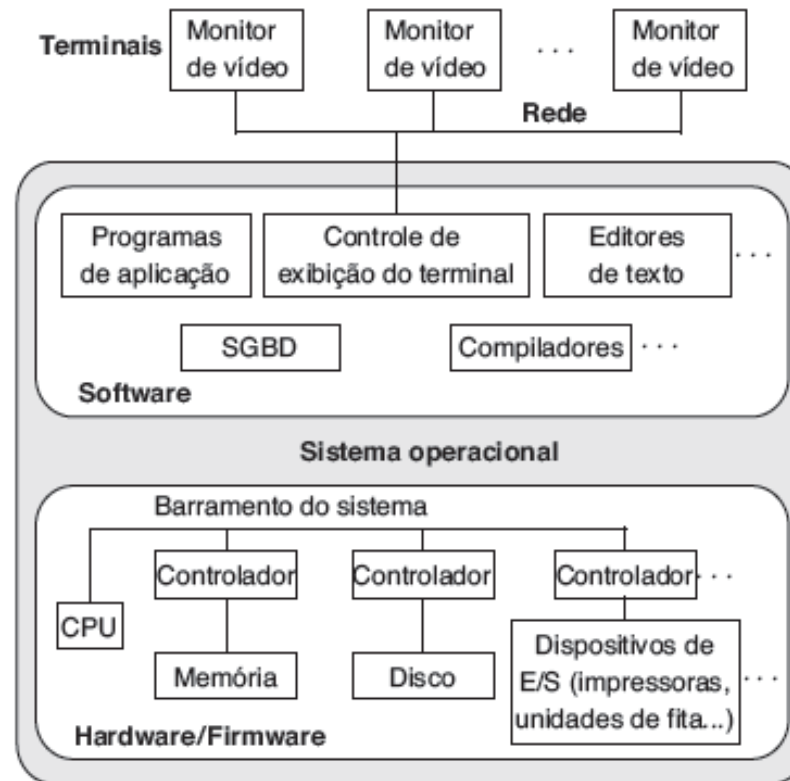
- Exemplos de Arquiteturas:
 - (A0,H0,D1)
 - Sistemas locais integrados, homogêneos, com arquitetura cliente-servidor (SGBD Cliente-Servidor)
 - (A2,H1,D1)
 - Sistemas locais autônomos, heterogêneos, com arquitetura cliente-servidor
 - (A0,H0,D2)
 - Sistemas locais integrados, homogêneos, com arquitetura distribuída (SGBDD Homogêneo)
 - (A2,D2,H2)
 - Sistemas locais autônomos, heterogêneos, com arquitetura distribuída (Multi-SGBD Heterogêneo)

Arquitetura Centralizada

Na arquitetura centralizada, existe um computador com grande capacidade de processamento, o qual é o hospedeiro do SGBD e emuladores para os vários aplicativos. Esta arquitetura tem como principal vantagem a de permitir que muitos usuários manipulem grande volume de dados. Sua principal desvantagem está no seu alto custo, pois exige ambiente especial para mainframes e soluções centralizadas.



Arquitetura Centralizada



Uma arquitetura física centralizada.

SGBDs Centralizados

- Descrição do Banco de Dados:

- **nível externo:**

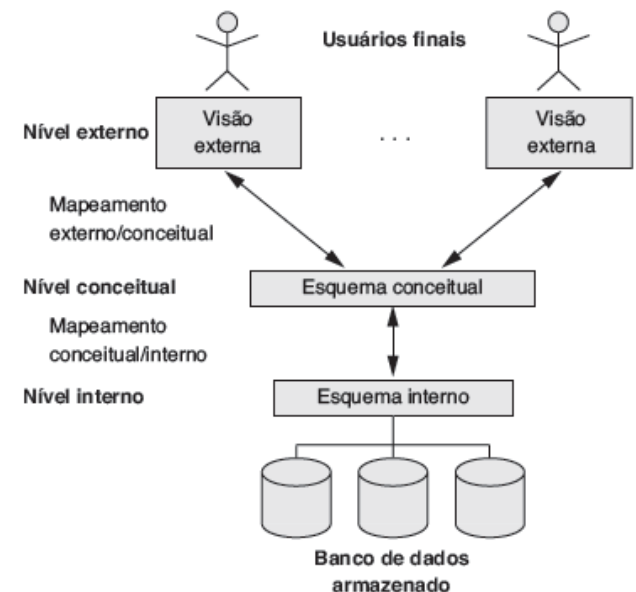
- especificação da organização conceitual do BD, vista por um grupo de usuários;

- **nível conceitual:**

- especificação da organização conceitual do BD, ou seja, o que o BD armazena;

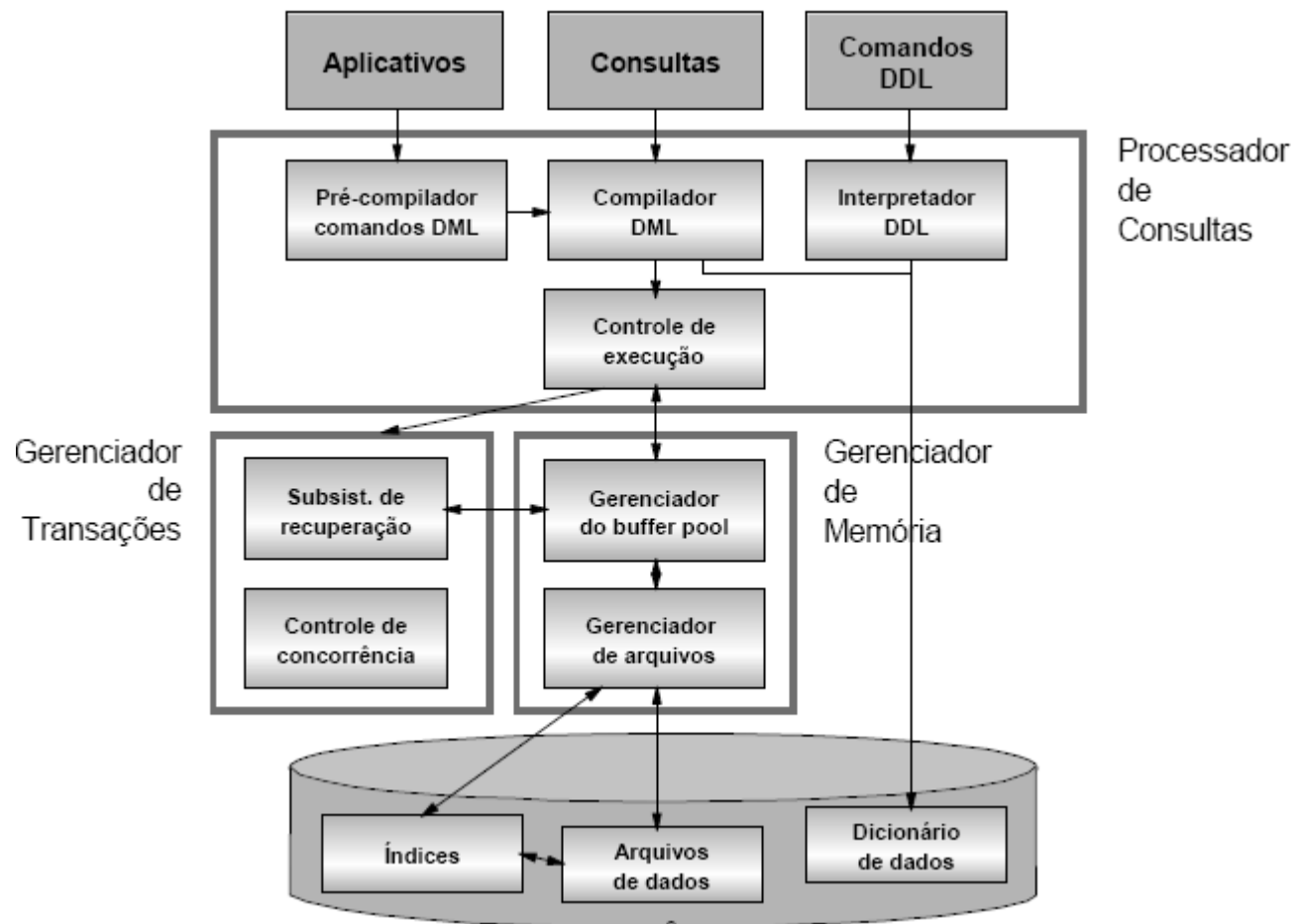
- **nível físico ou interno:**

- especificação das estruturas de armazenamento do BD, ou seja, como o BD está armazenado;



A arquitetura de três esquemas.

SGBDs Centralizados - Arquitetura





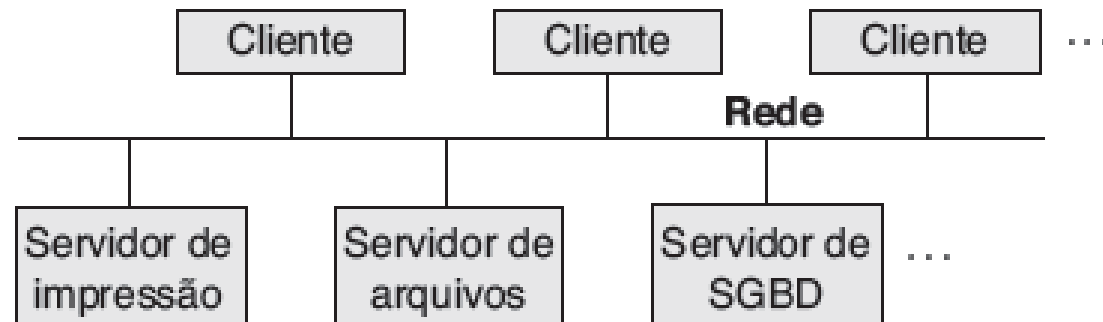
Arquitetura Cliente-Servidor

Na arquitetura Cliente-Servidor, o cliente (*front_end*) executa as tarefas do aplicativo, ou seja, fornece a interface do usuário (tela, e processamento de entrada e saída). O servidor (*back_end*) executa as consultas no DBMS e retorna os resultados ao cliente.

Apesar de ser uma arquitetura bastante popular, são necessárias soluções sofisticadas de software que possibilitem: o tratamento de transações, as confirmações de transações (*commits*), desfazer transações (*rollbacks*), linguagens de consultas (*stored procedures*) e gatilhos (*triggers*).

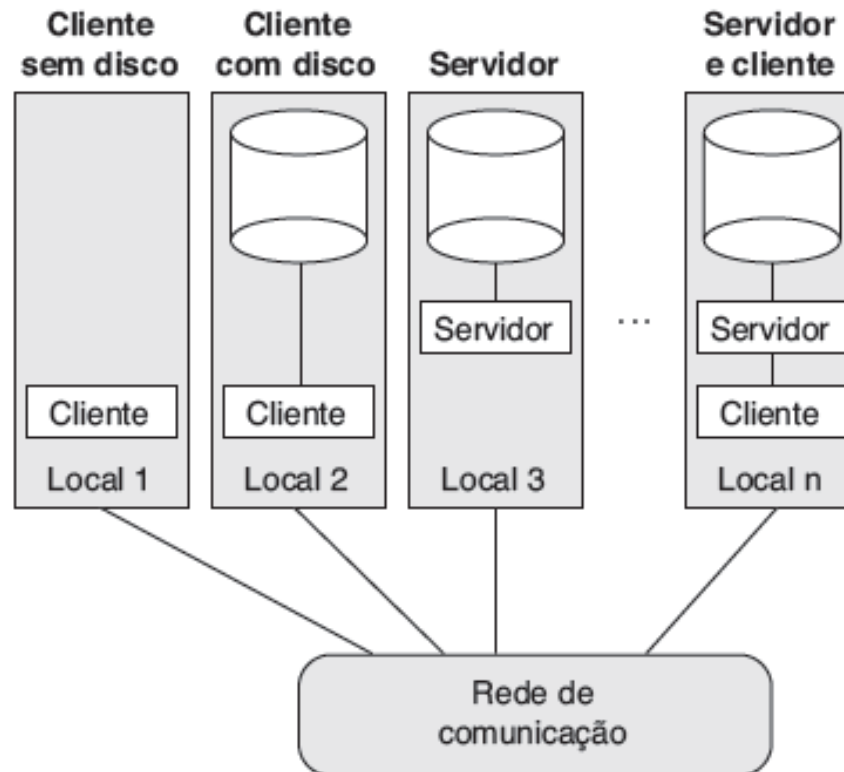
A principal vantagem desta arquitetura é a divisão do processamento entre dois sistemas, o que reduz o tráfego de dados na rede.

Arquitetura Cliente-Servidor



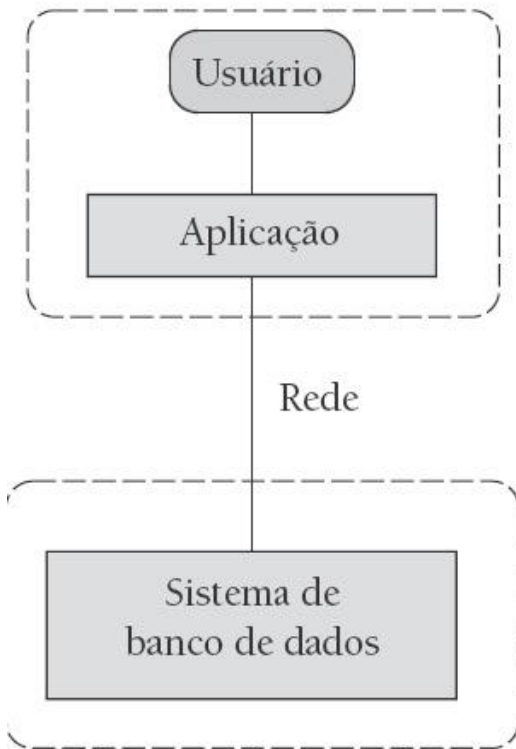
Arquitetura cliente/servidor lógica em duas camadas.

Arquitetura Cliente-Servidor

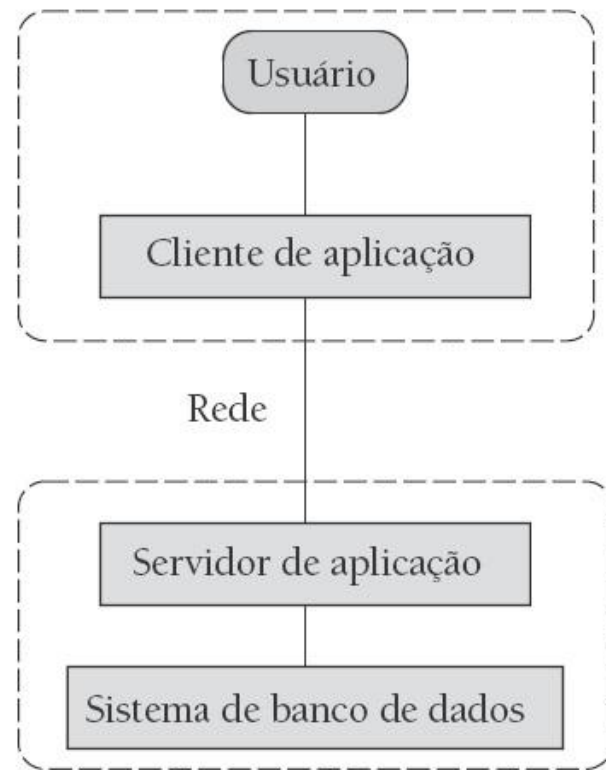


Arquitetura cliente/servidor física em duas camadas.

Arquitetura Cliente-Servidor



(a) Arquitetura de duas camadas



(b) Arquitetura de três camadas

SGBDs Cliente-Servidor

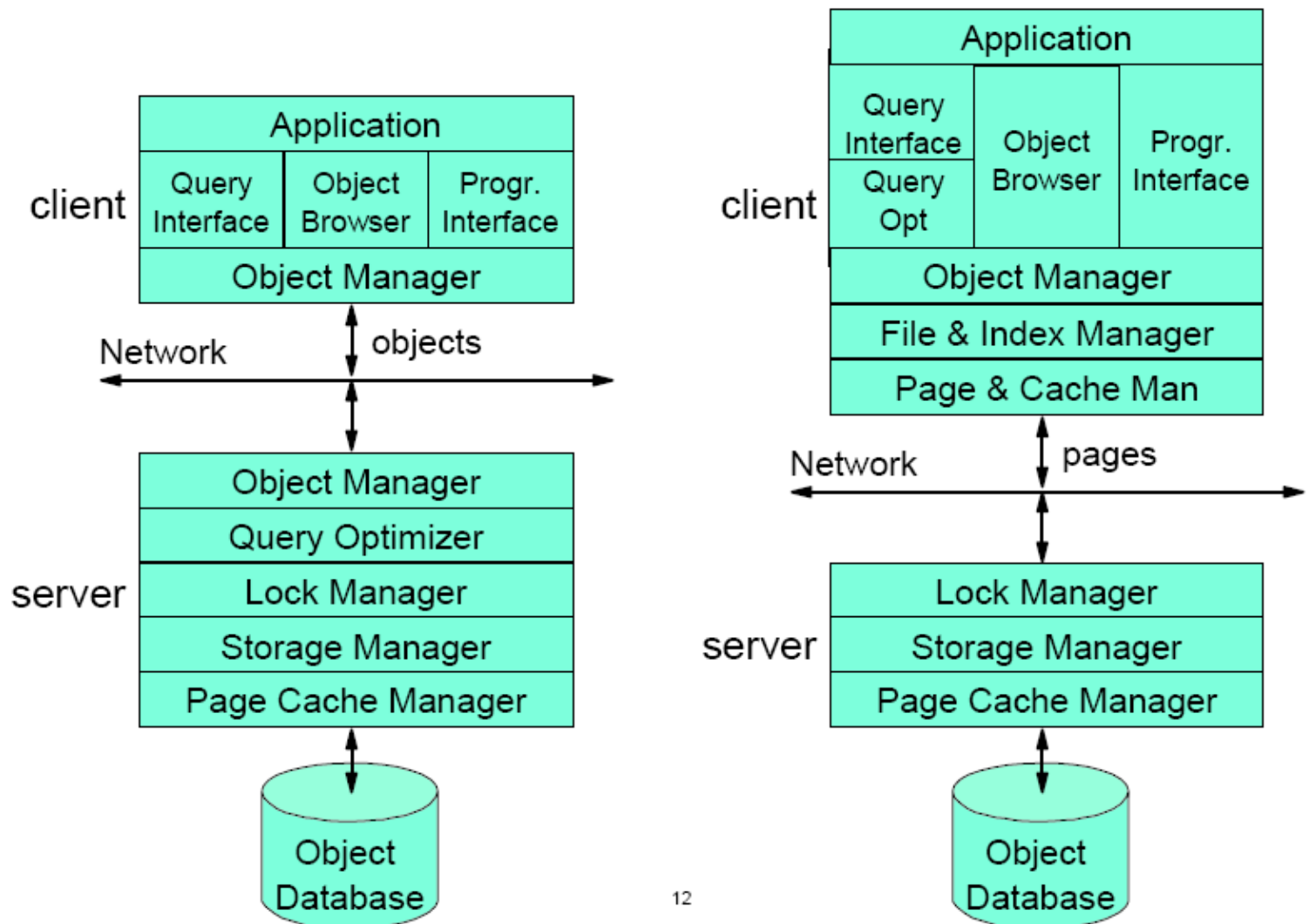
- **Servidor de Objetos:**
 - servidor transfere objetos para a *object cache area* do cliente
 - cliente é responsável por:
 - executar os métodos comandados pela aplicação
 - obter do servidor os objetos necessários
 - gerenciar os objetos armazenados na *object cache area* local, incluindo a criação, atualização e remoção de objetos
 - servidor é responsável pelo:
 - processamento de consultas, incluindo a recolocação dos objetos atualizados no banco
 - gerência de transações, incluindo a sincronização da *object cache area* dos cliente
 - servidor pode fazer *pre-fetch* de objetos e enviá-los ao cliente



SGBDs Cliente-Servidor

- **Servidor de Páginas:**
 - servidor transfere páginas para a *buffer pool area* do cliente
 - cliente é responsável por:
 - executar os métodos comandados pela aplicação
 - implementar toda a semântica associada a objetos
 - servidor é responsável pelo:
 - armazenamento físico dos objetos apenas

SGBDs Cliente-Servidor





Arquitetura Distribuída

Nesta arquitetura, a informação está distribuída em diversos servidores.

Cada servidor atua como no sistema cliente-servidor, porém as consultas oriundas dos aplicativos são feitas para qualquer servidor indistintamente.

Caso a informação solicitada seja mantida por outro servidor ou servidores, o sistema encarrega-se de obter a informação necessária, de maneira transparente para o aplicativo, que passa a atuar consultando a rede, independente de conhecer seus servidores.

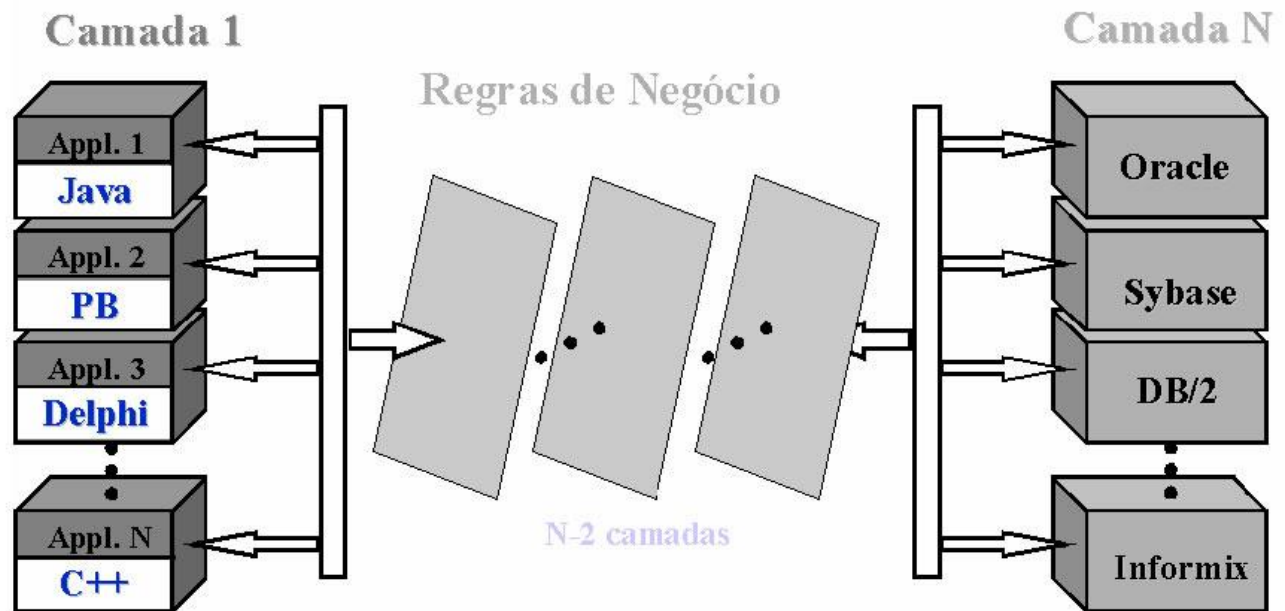


Arquitetura Distribuída

Exemplos típicos são as bases de dados corporativas, em que o volume de informação é muito grande e, por isso, deve ser distribuído em diversos servidores. Porém, não é dependente de aspectos lógicos de carga de acesso aos dados, ou base de dados fracamente acopladas, em que uma informação solicitada vai sendo coletada numa propagação da consulta numa cadeia de servidores.

A característica básica é a existência de diversos programas aplicativos consultando a rede para acessar os dados necessários, porém, sem o conhecimento explícito de quais servidores dispõem desses dados.

Arquitetura Distribuída





SGBDs Distribuídos

- **BD Distribuído:**
 - banco de dados armazenado em vários servidores, conectados por uma rede de comunicação de dados;
- **SGBD Distribuído:**
 - estende as técnicas de armazenamento de dados, processamento de consultas e gerência de transações para permitir a implementação de BDDs;

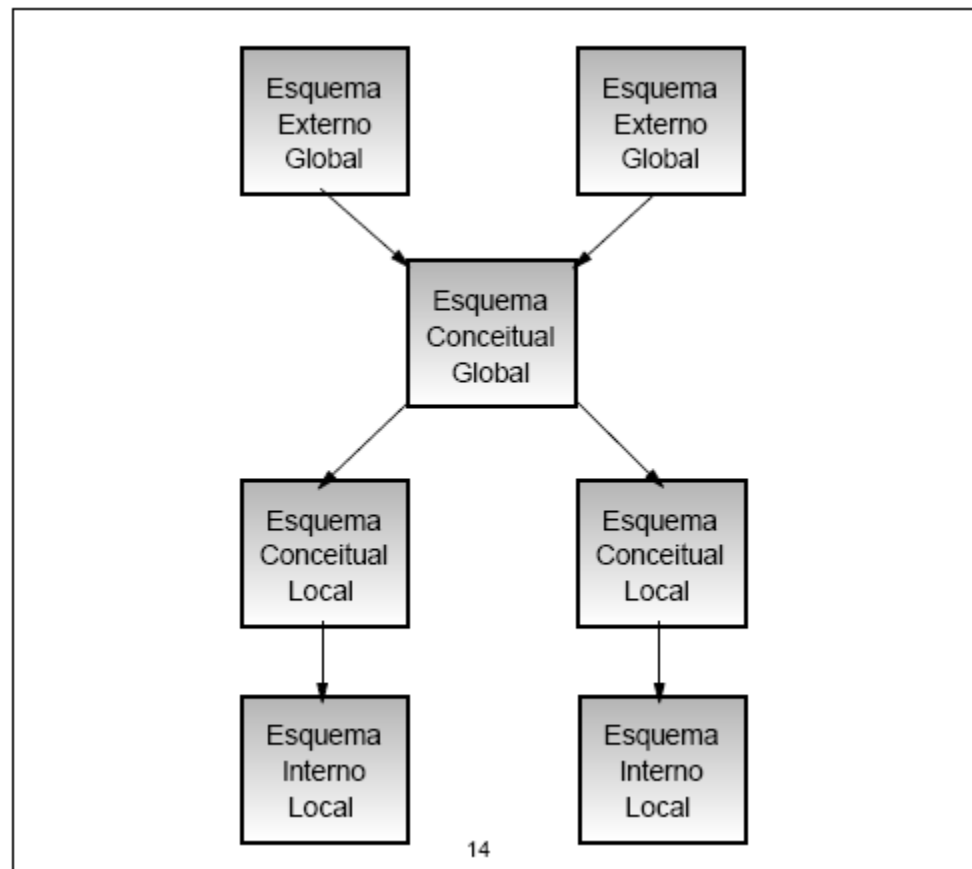


SGBDs Distribuídos

- **Bancos de dados distribuídos homogêneos**
 - Mesmo software/esquema em todos os sites, os dados podem estar particionados entre os sites;
 - *Objetivo*: oferecer uma visão de um único banco de dados, ocultando os detalhes da distribuição;
- **Bancos de dados distribuídos heterogêneos**
 - Software/esquema diferente em sites diferentes;
 - *Objetivo*: integrar bancos de dados existentes para oferecer funcionalidade útil;

SGBDs Distribuídos Homogêneos

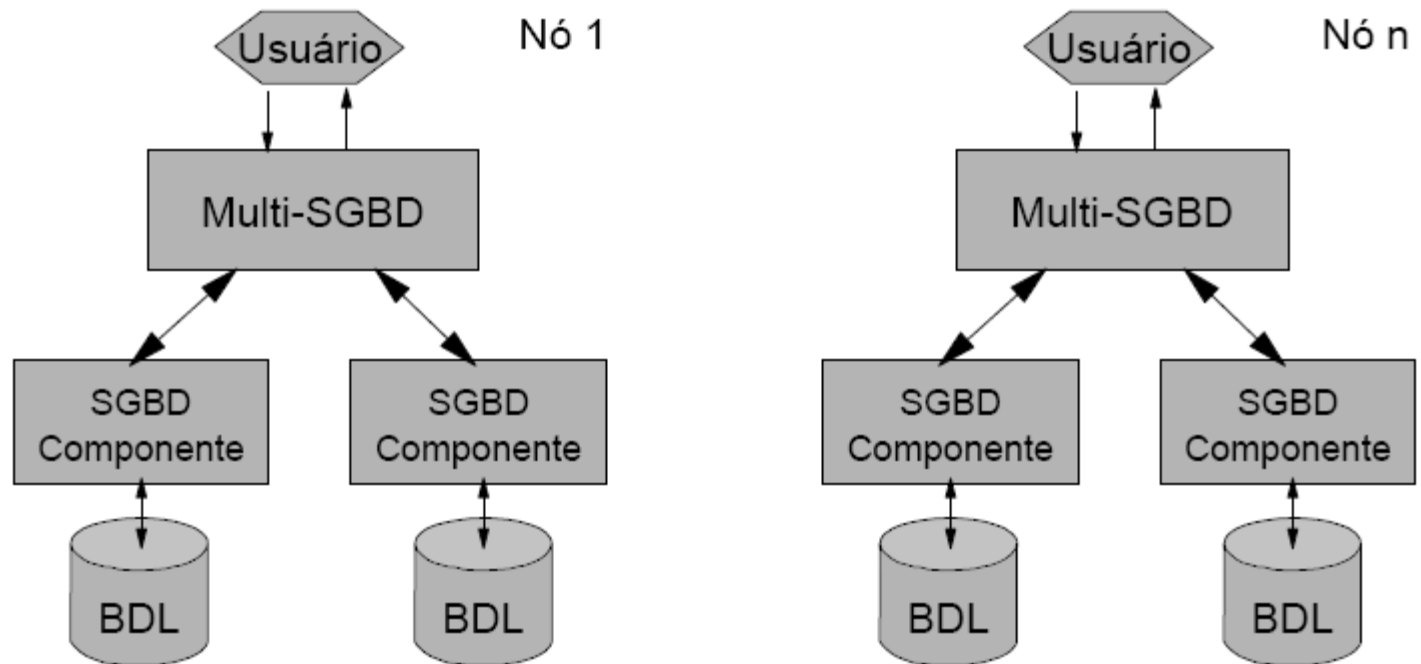
- Um Banco de Dados Distribuído



SGBDs Distribuídos Heterogêneos

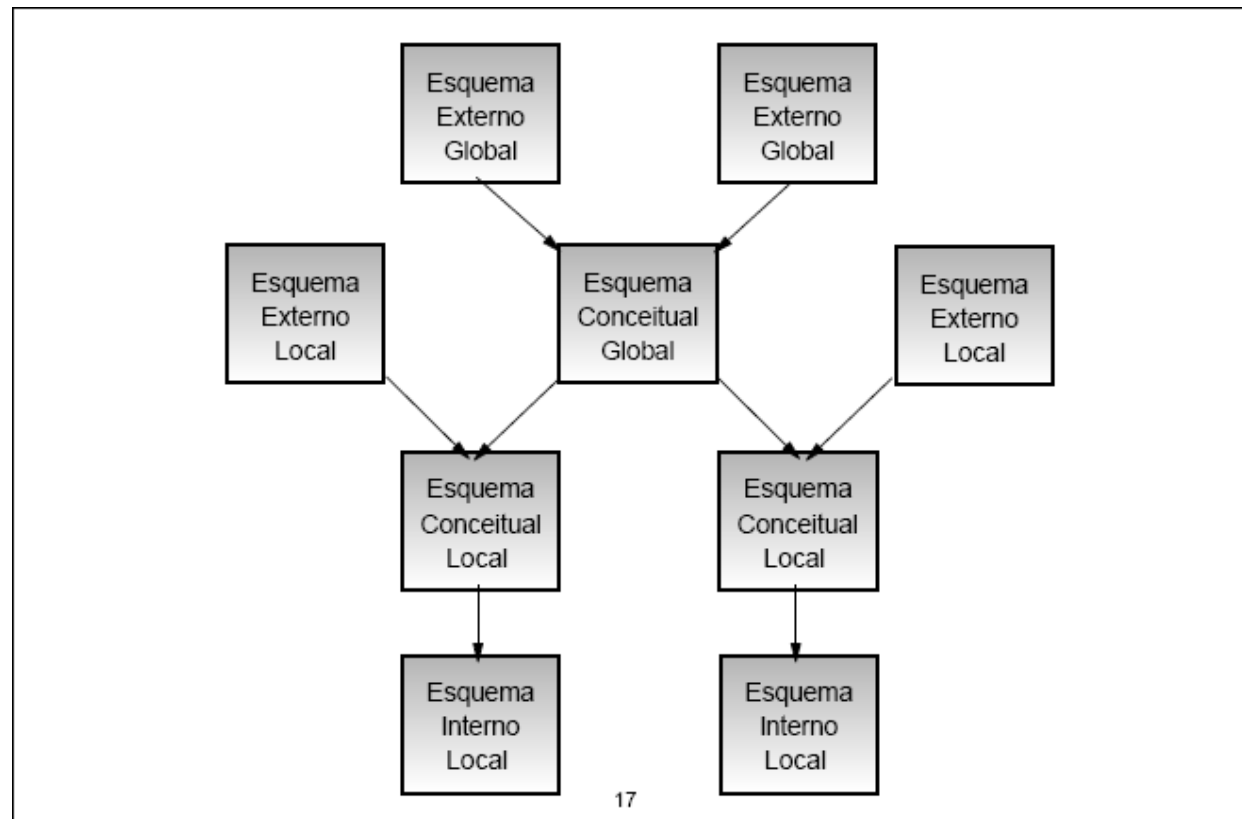
- Arquitetura

- cada nó pode conter um ou mais SGBDs locais



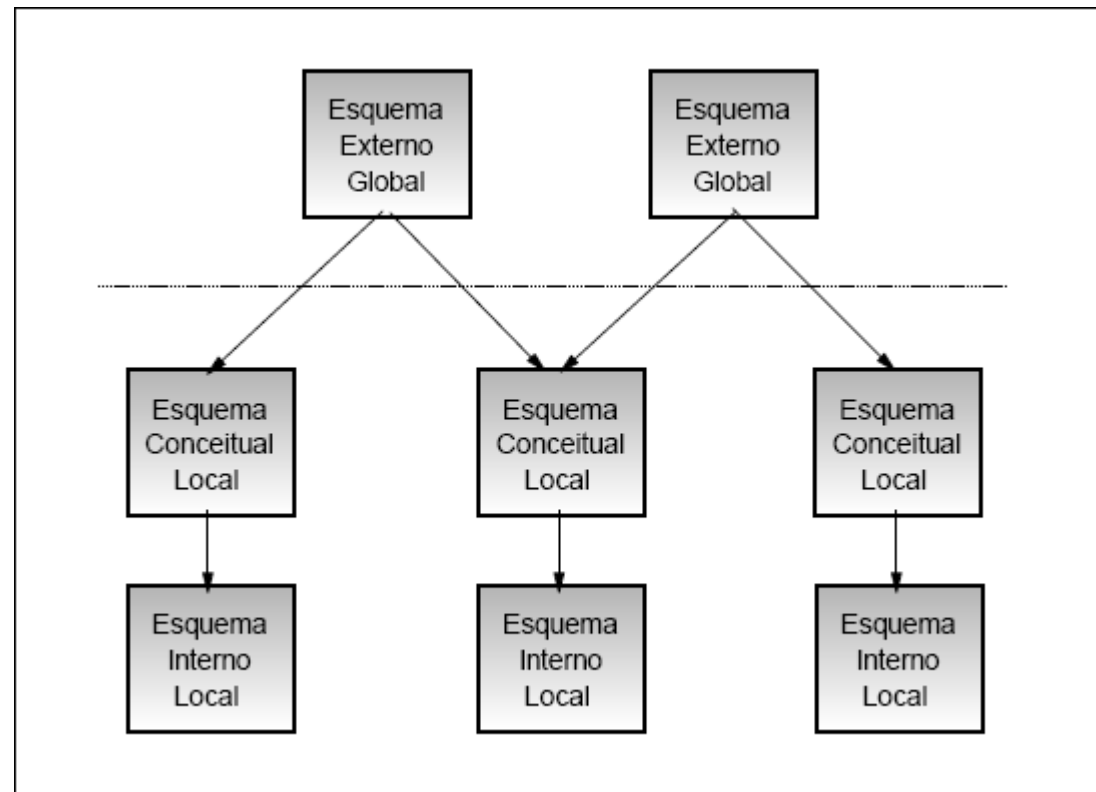
SGBDs Distribuídos Heterogêneos

- Multi-SGBDD Heterogêneo com Esquema Conceitual Global:



SGBDs Distribuídos Heterogêneos

- Multi-SGBDD Heterogêneo sem Esquema Conceitual Global:

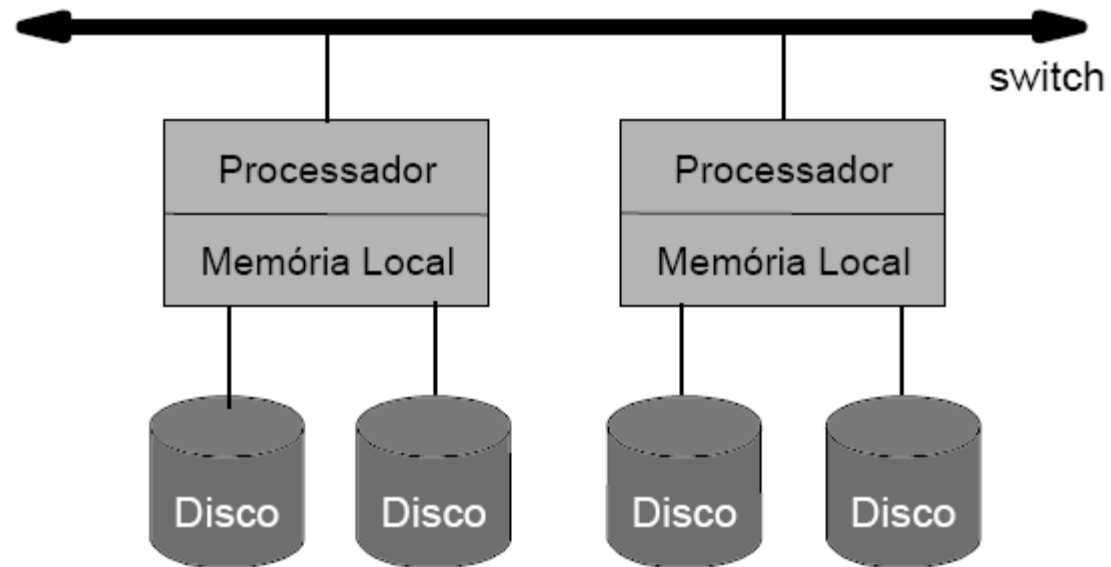


SGBDs Paralelos

- Introdução:

- combinam técnicas de gerência de dados e processamento paralelo para aumentar desempenho e confiabilidade:
 - particionamento do BD em discos controlados por multiprocessadores resulta em aumento da taxa de transferência de dados da memória secundária para memória principal (*I/O bandwidth*);
 - paralelização do processamento interno de consultas resulta em diminuição do tempo de resposta;
 - paralelização do processamento de transações resulta em aumento da capacidade do sistema (*throughput*);

SGBDs Paralelos

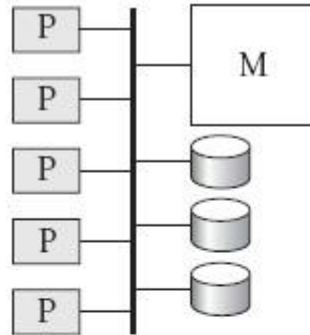




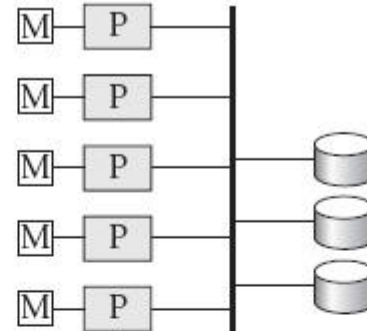
Arquiteturas de SGBDs Paralelos

- **Memória compartilhada**
 - processadores compartilham uma memória comum;
- **Disco compartilhado**
 - processadores compartilham um disco comum;
- **Nada compartilhado**
 - processadores não compartilham memória comum nem disco comum;
- **Hierárquica**
 - híbrido das arquiteturas acima.

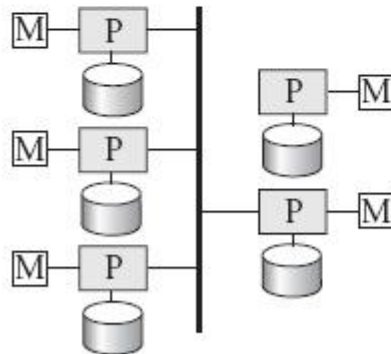
Arquiteturas de SGBDs Paralelos



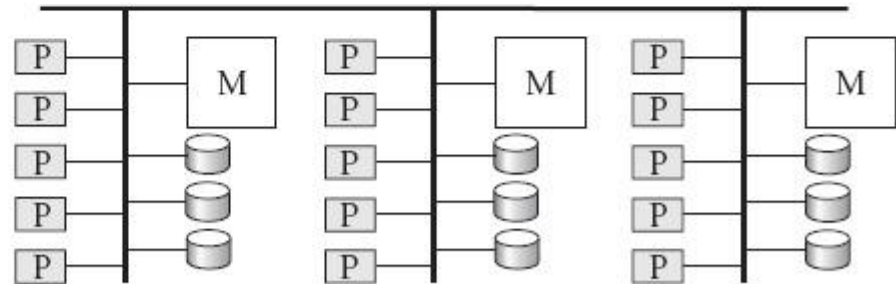
(a) memória compartilhada



(b) disco compartilhado



(c) nada compartilhado



(d) hierárquico



Arquiteturas de SGBDs Paralelos

- **Memória compartilhada**

- Processadores e discos têm acesso a uma memória comum, normalmente por meio de um barramento ou por uma rede de interconexão.
- Comunicação extremamente eficiente entre os processadores - os dados na memória compartilhada podem ser acessados por qualquer processador sem ter que movê-los usando software.
- Desvantagem - arquitetura não é expansível além de 32 ou 64 processadores, pois o barramento ou a rede de interconexão se torna um gargalo.
- Muito usada para graus de paralelismo menores (4 a 8).

Arquiteturas de SGBDs Paralelos

- **Disco compartilhado**

- Todos os processadores podem acessar diretamente todos os discos por meio de uma rede de interconexão, mas os processadores têm memórias privadas.
 - O barramento da memória não é um gargalo;
 - A arquitetura oferece um grau de tolerância a falhas - se um processador falhar, os outros processadores podem assumir suas tarefas, pois o banco de dados é residente nos discos que são acessíveis por todos os processadores.
- Exemplos: clusters IBM Sysplex e DEC (agora parte da Compaq) rodando Rdb (agora Oracle Rdb) foram os primeiros usuários comerciais.
- Desvantagem: o gargalo agora ocorre na interconexão com o subsistema de disco.
- Sistemas de disco compartilhado podem se expandir até um número um pouco maior de processadores, mas a comunicação entre os processadores é mais lenta.

Arquiteturas de SGBDs Paralelos

- **Nada compartilhado**

- O nó consiste em um processador, memória e um ou mais discos. Os processadores em um nó se comunicam com outro processador em outro nó, usando uma rede de interconexão. Um nó funciona como servidor para os dados no disco ou discos que o nó possui.
- Exemplos: Teradata, Tandem, Oracle-n CUBE
- Os dados acessados a partir de discos locais (e acessos a memória locais) não passam pela rede de interconexão, minimizando assim a interferência do compartilhamento de recursos.
- Os multiprocessadores de nada compartilhado podem ser expandidos até milhares de processadores sem interferência.
- Desvantagem principal: custo de comunicação e acesso a disco não local, o envio de dados envolver a interação do software nas duas pontas.

Arquiteturas de SGBDs Paralelos

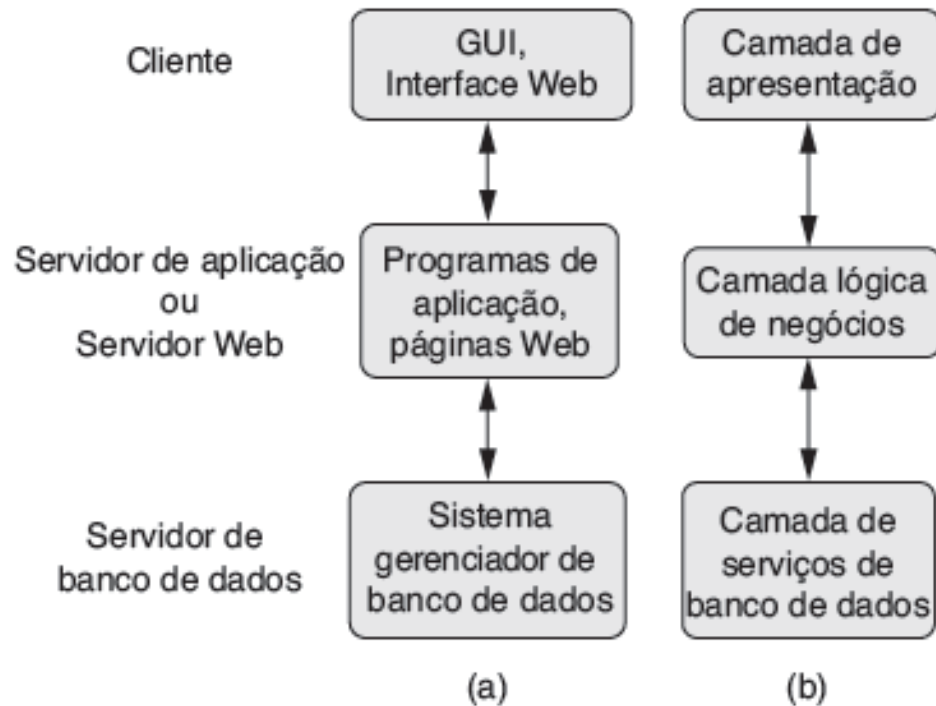
- **Hierárquica**

- Combina características de arquiteturas de memória compartilhada, disco compartilhado e nada compartilhado.
- Nível superior é uma arquitetura nada compartilhado - os nós conectados por uma rede de interconexão e não compartilha discos ou memória entre si.
- Cada nó do sistema poderia ser um sistema de memória compartilhada com alguns processadores.
- Como alternativa, cada nó poderia ser um sistema de disco compartilhado, e cada um dos sistemas compartilhando um conjunto de discos poderia ser um sistema de memória compartilhada.
- Reduz a complexidade de programar tais sistemas por arquiteturas de memória virtual distribuída.
- Também chamada **arquitetura de memória não uniforme**.

SGBDs e a Web

- **Arquiteturas de três camadas e n camadas para aplicações Web**
 - Muitas aplicações Web utilizam uma arquitetura chamada de **arquitetura de três camadas**, que acrescenta uma camada intermediária entre o cliente e o servidor de banco de dados.
 - A camada intermediária é chamada de **servidor de aplicação** ou **servidor Web**, dependendo da aplicação.
 - Os clientes têm interfaces GUI e algumas regras de negócios adicionais, específicas da aplicação.
 - A camada de apresentação exibe informações ao usuário e permite a entrada de dados. A camada de lógica de dados cuida das regras e restrições intermediárias antes dos dados serem passados para o usuário ou devolvidos ao SGBD.

SGBDs e a Web



Arquitetura cliente/servidor lógica de três camadas, com algumas nomenclaturas comumente utilizadas.



SGBDs e a Web

- **WIIS - *Web Information Integration System*:**
 - sistema para processar consultas sobre dados extraídos de vários Web sites.
- **WIIS X Multi SGBDD:**
 - WIIS compartilham com Multi SGBDDs os mesmos problemas
 - WIIS devem ainda lidar com:
 - um grande número de Web sites
 - metadados escassos sobre os Web sites
 - maior autonomia dos componentes



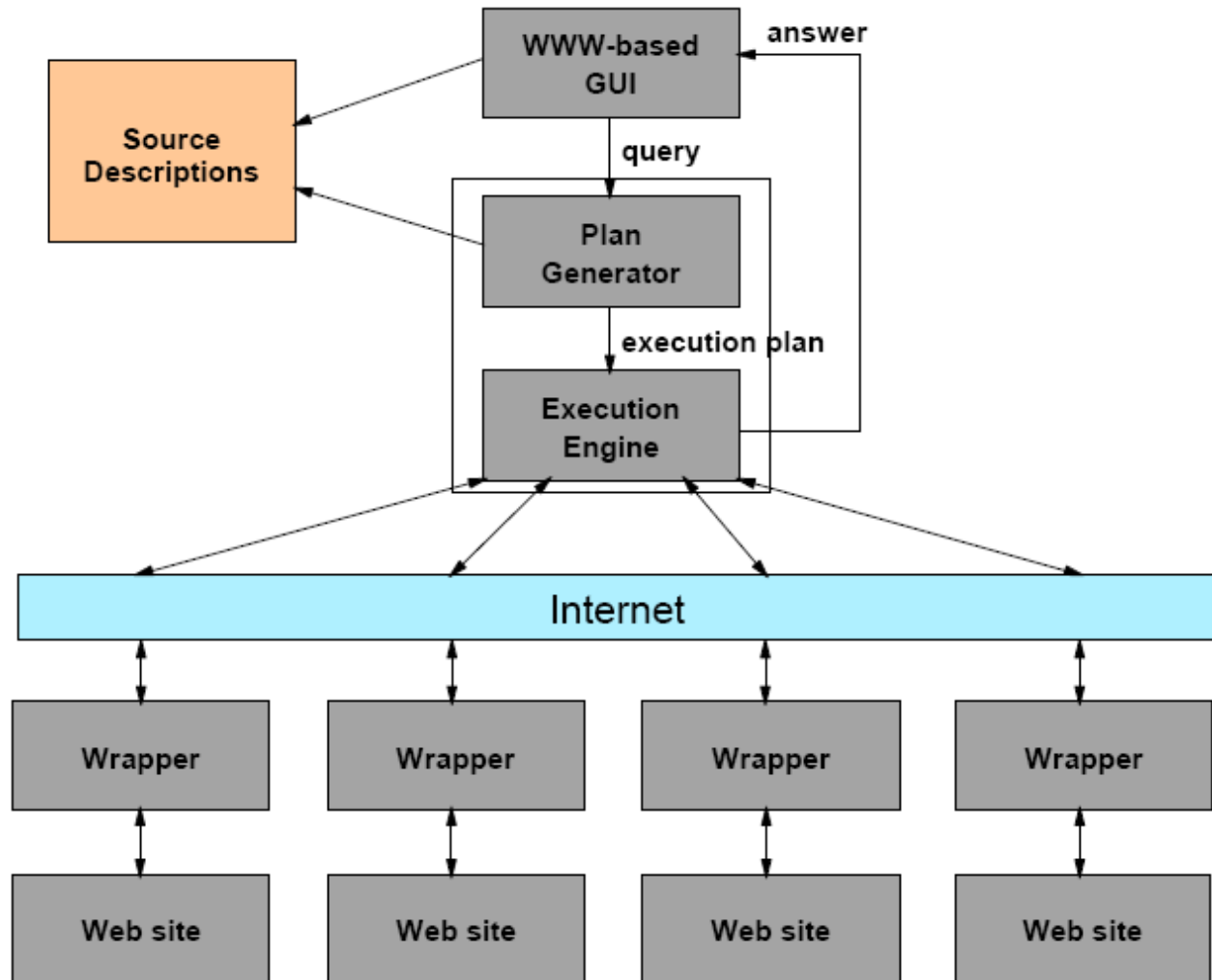
SGBDs e a Web

- **Enfoques para construção de WIIS:**
 - Enfoque de Data Warehouse:
 - dados de várias fontes são extraídos e armazenados em uma warehouse;
 - dados são atualizados periodicamente;
 - interessante quando:
 - os dados são pouco voláteis;
 - processamento de consultas deve ser eficiente.
 - Multi-SGBD:
 - dados são mantidos nos Web sites;
 - consultas são decompostas e enviadas aos várias Web sites envolvidas;
 - processamento de consultas é mais sofisticado;
 - interessante quando:
 - o número de Web sites é muito grande;
 - os dados são voláteis;
 - há pouco controle sobre os Web sites.

SGBDs e a Web

- **Componentes de um WIIS:**
 - *Wrapper*:
 - programa construído para traduzir os dados de um Web site para uma forma que pode ser processada pelo WIIS;
 - *Web source*:
 - combinação de um *Wrapper* com o Web site associado;
 - *Mediated Schema*:
 - conjunto de visões sobre os dados armazenados nos Web sites;
 - *Source Descriptions*:
 - conjunto de metadados e regras de reescrita indicando ao WIIS como reformular as consultas sobre o *mediated schema* para consultas sobre as *Web sources*.

SGBDs e a Web



SGBDs para Estações Móveis

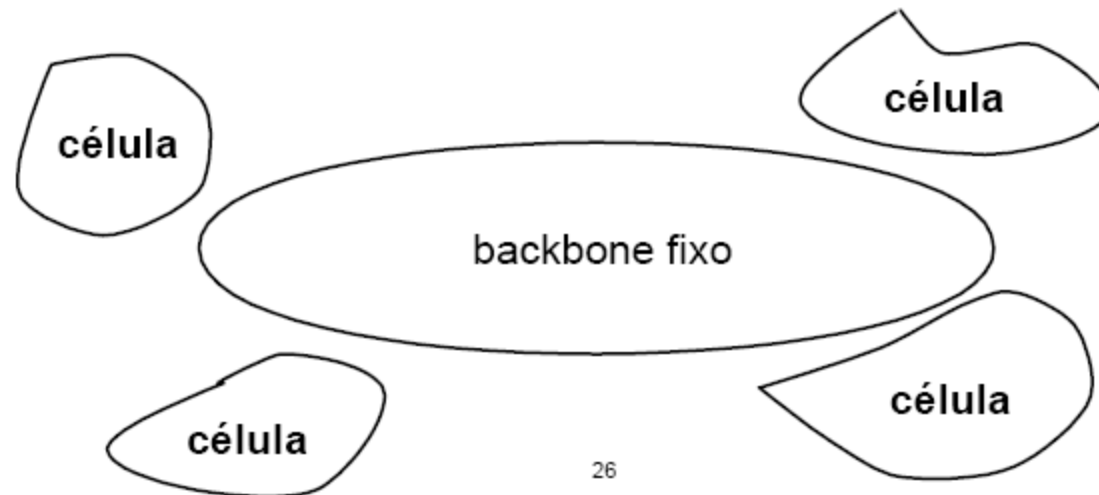
- **Características:**

- topologia:

- um *backbone* fixo com estações de controle também fixas;
 - conjunto de células nas quais transitam as estações móveis;

- localização dos dados:

- dados residem tanto nas estações fixas quanto nas móveis ("*walkstations*");



SGBDs para Estações Móveis

- **Manutenção do Diretório:**

- Problema:

- como localizar as estações móveis que contêm os dados desejados;

- Alternativas:

- armazenar a localização da estação móvel apenas na sua estação de base;
 - distribuir a informação sobre a localização das estações móveis;
 - pré-anunciar o roteiro de cada estação móvel.

- ***Caching:***

- Problema:

- limite da bateria da estação móvel impõe restrições sobre o grau de *caching*;

- Alternativas:

- mover consultas para estações fixas, transferindo apenas resultados;
 - adotar filtros semânticos mais sofisticados para reduzir o tamanho do *cache*.

Referências

- Silberschatz, A.; Korth, H. F.; Sudarshan, S.; Sistema de Banco de Dados. 5ed, Ed. Campus, 2006. ISBN: 8535211071.
- Date, C. J.; Introdução a Sistemas de Banco de Dados. 8ed, Ed. Campus, 2004. ISBN: 8535212736.
- Elmasri, R.; Navathe, S. B.; Fundamentals of Database Systems. 5ed, Ed. Addison Wesley, 2006. ISBN: 9780321369574.
- Andre Rodrigo Sanches – Notas de aula. Disponível em: <<http://www.ime.usp.br/~andrers/aulas/bd2005-1/>>.
- Marco Antonio Casanova – Notas de aula. Departamento de Informática – PUC-RIO.