

## Aula 011

### Professores:

Marta Mattoso

Vanessa Braganholo

### Conteúdo:

Arquiteturas

# Organização do Curso

- ➡ Conceitos Gerais
- ➡ SGBDs e Modelo de Dados
- ➡ Modelo ER
- ➡ Arquiteturas**
- ➡ Modelo Relacional
- ➡ Álgebra Relacional
- ➡ Mapeamento ER-Relacional
- ➡ SQL
- ➡ Normalização
- ➡ Evolução dos Modelos

# Parte 1 (Finalizando): Introdução e Modelagem conceitual;

Capítulos do livro texto Elmasri/Navathe:

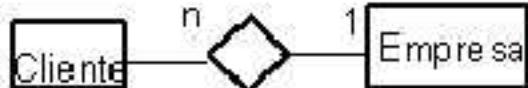
1. Banco de dados e os usuários de banco de dados;
- 2. Sistemas de banco de dados: conceitos e **arquitetura**;
3. Modelagem de dados usando o modelo Entidade-Relacionamento;

## Arquiteturas

Aplicações de BD



### Modelo ER



### Modelo Relacional

	Recomend.	Nome	Cod. Area	Número
1	1 Projeto Passagístico		303	444-3222
2	2 Construção Lagos		303	776-6899
3	3 Nova Direção		303	767-7783
4	4 Projeto Interiores		303	543-8876

### Modelo Físico

registro	A-102	Pensridge	400
registro 1	A-303	Round Hill	350
registro 2	A-215	Maurine	280
registro 3	A-101	Downtown	520
registro 4	A-222	Redwood	700
registro 5	A-201	Pensridge	900
registro 6	A-217	Brighton	750
registro 7	A-110	Downtown	600
registro 8	A-210	Pensridge	700

# O que é a Arquitetura ?

→ Define a estrutura do sistema

- Identificação dos componentes
- Definição das funções de cada componente
- Definição de relacionamentos e interações entre os componentes

# Modelo de Referência

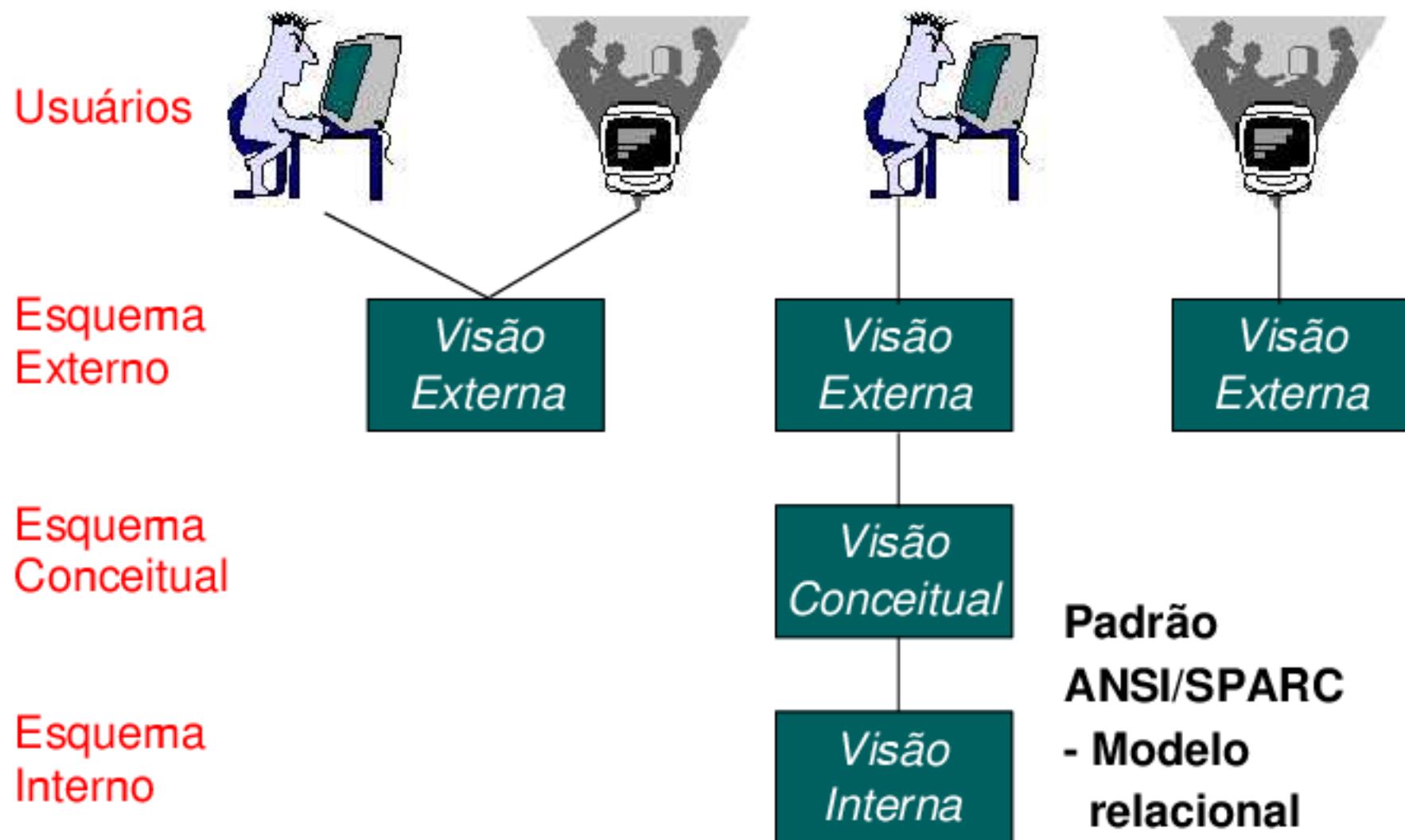
- Uma referência conceitual cujo propósito é dividir o trabalho de padronização em partes gerenciáveis e mostrar num nível mais alto como essas partes se relacionam entre si.
- Comparações

# Arquiteturas de SGBD

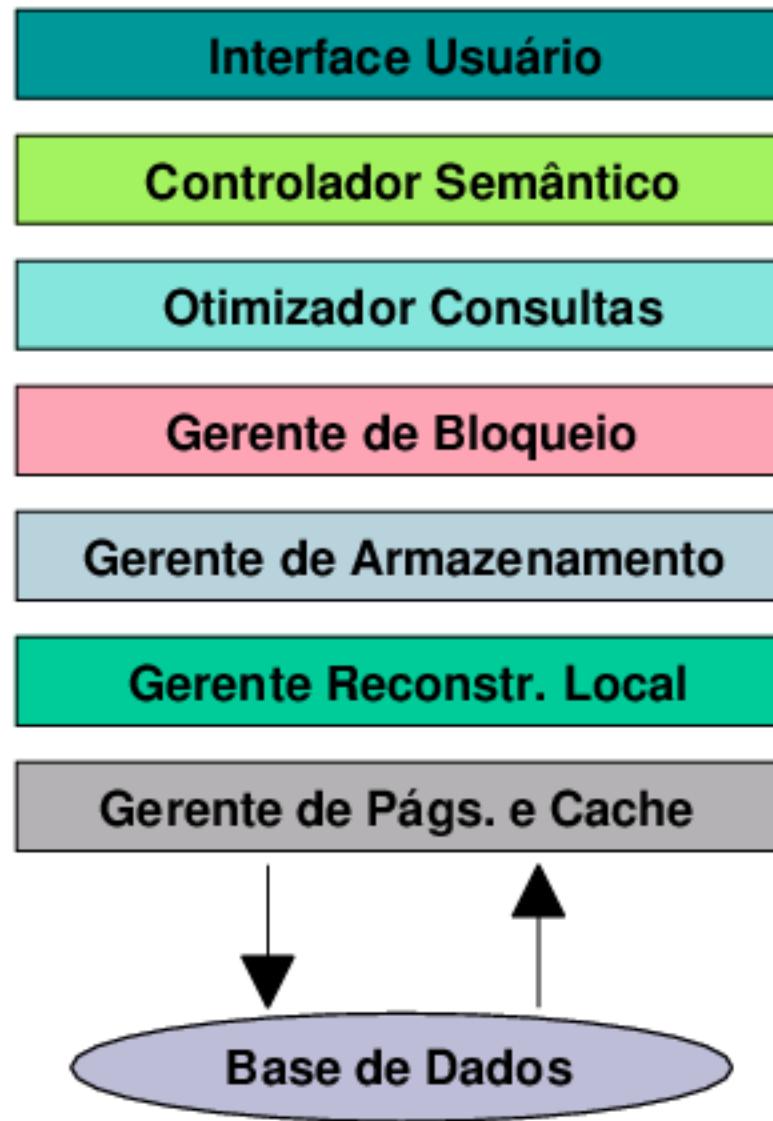
## → Abordagens

- Baseada em dados
  - Identifica as diferentes formas de descrever dados e especifica as unidades funcionais que irão definir e usar os dados de acordo com as formas.
- Baseada em componentes
  - Os componentes do sistema são definidos junto com as relações entre os componentes.
  - Bom para projeto e implementação de sistemas.
- Baseada em funções
  - Classes de usuários são identificados junto com a funcionalidade que o sistema fornecerá a cada classe.
  - Os objetivos são identificados. Mas como alcançar ?

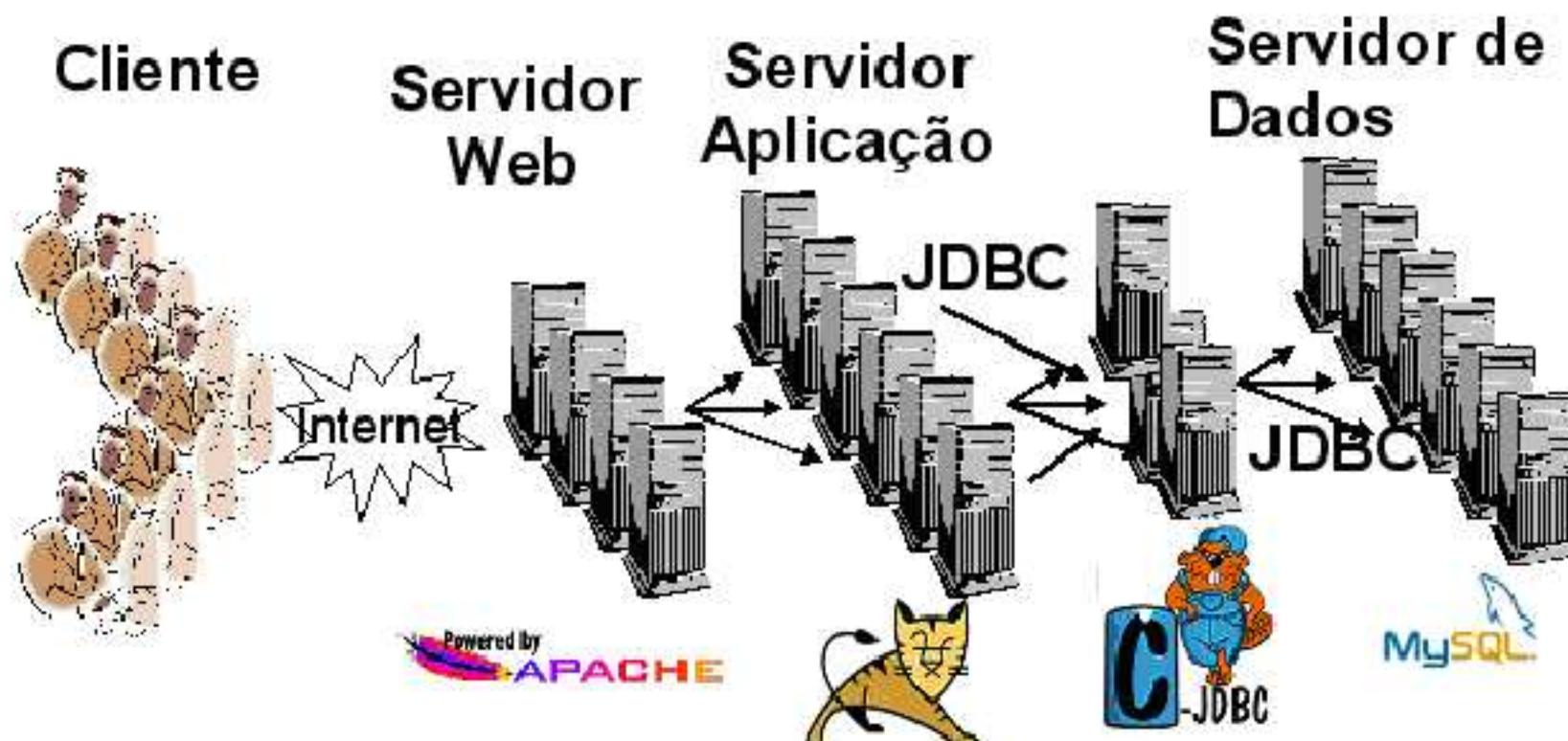
# Arquitetura de dados



# Arquitetura de componentes



# Arquitetura de funções



<http://c-jdbc.objectweb.org/> - c-jdbc@objectweb.org

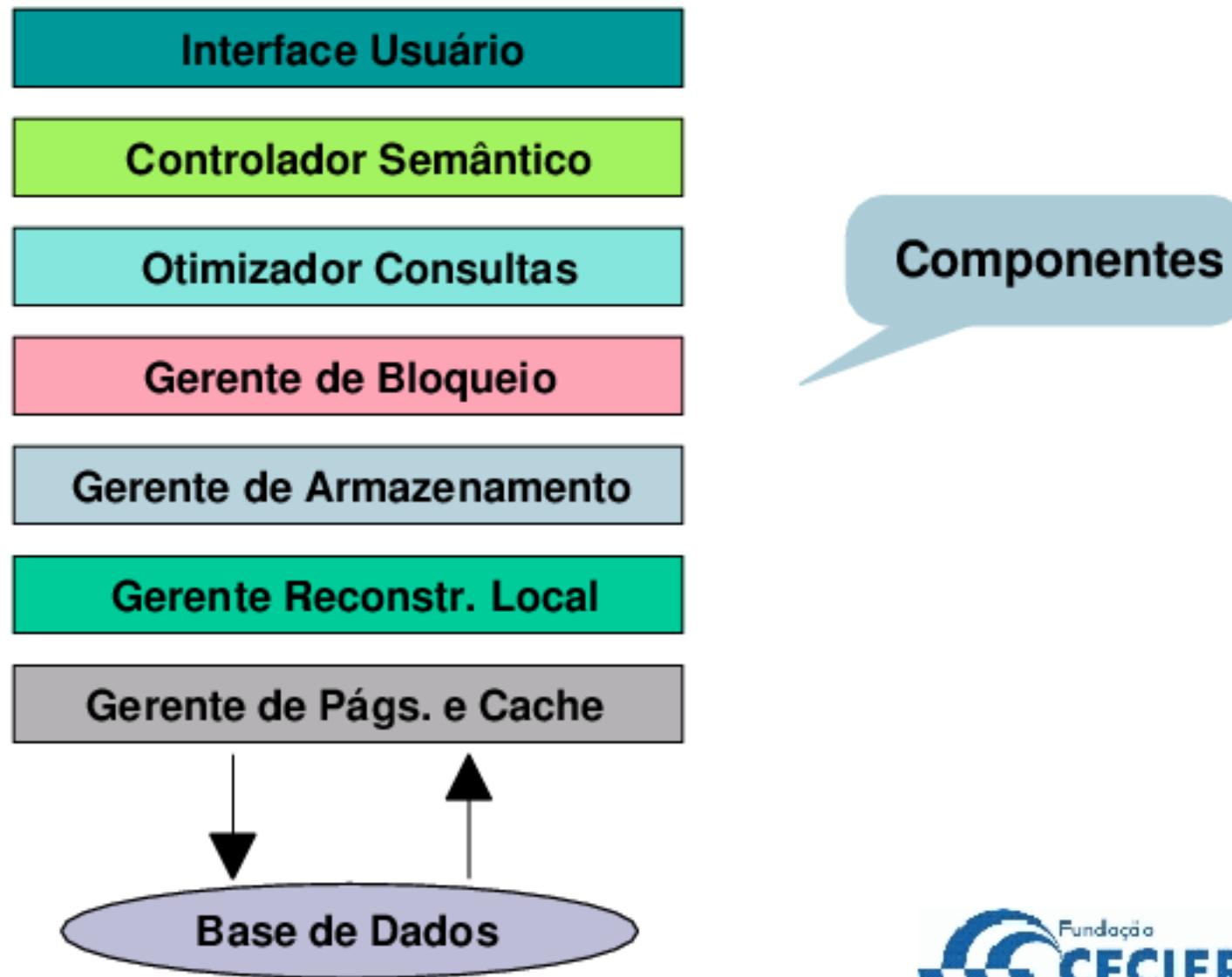
Slides adaptados de Emmanuel Cecchet

# Abordagens são complementares

## → Arquiteturas

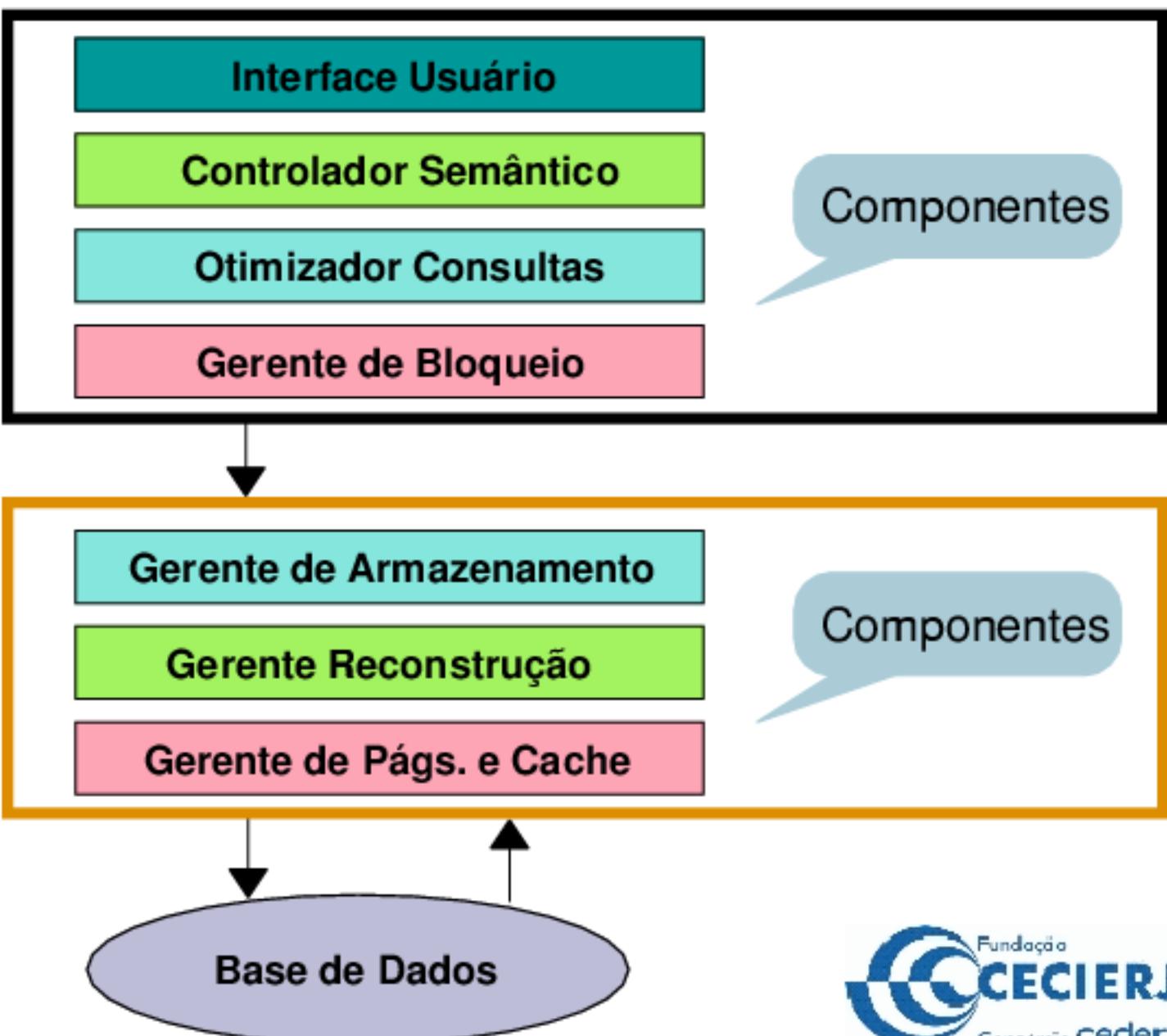
- Baseada em componentes
- Baseada em funções
- Baseada em dados

# Abordagens são complementares

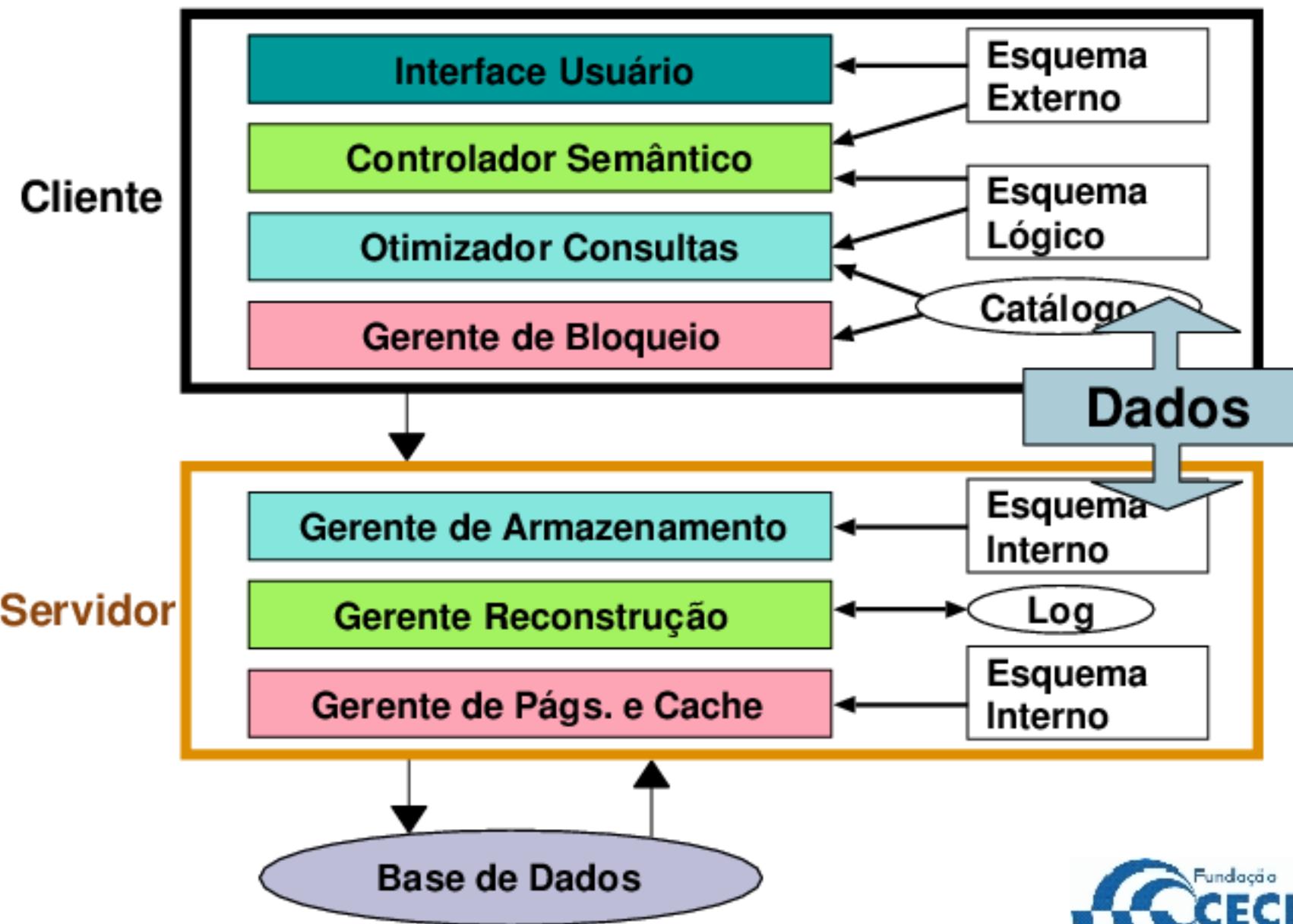


# Abordagens são complementares

**Cliente**  
↑  
**Funções**  
↓  
**Servidor**



# Abordagens são complementares

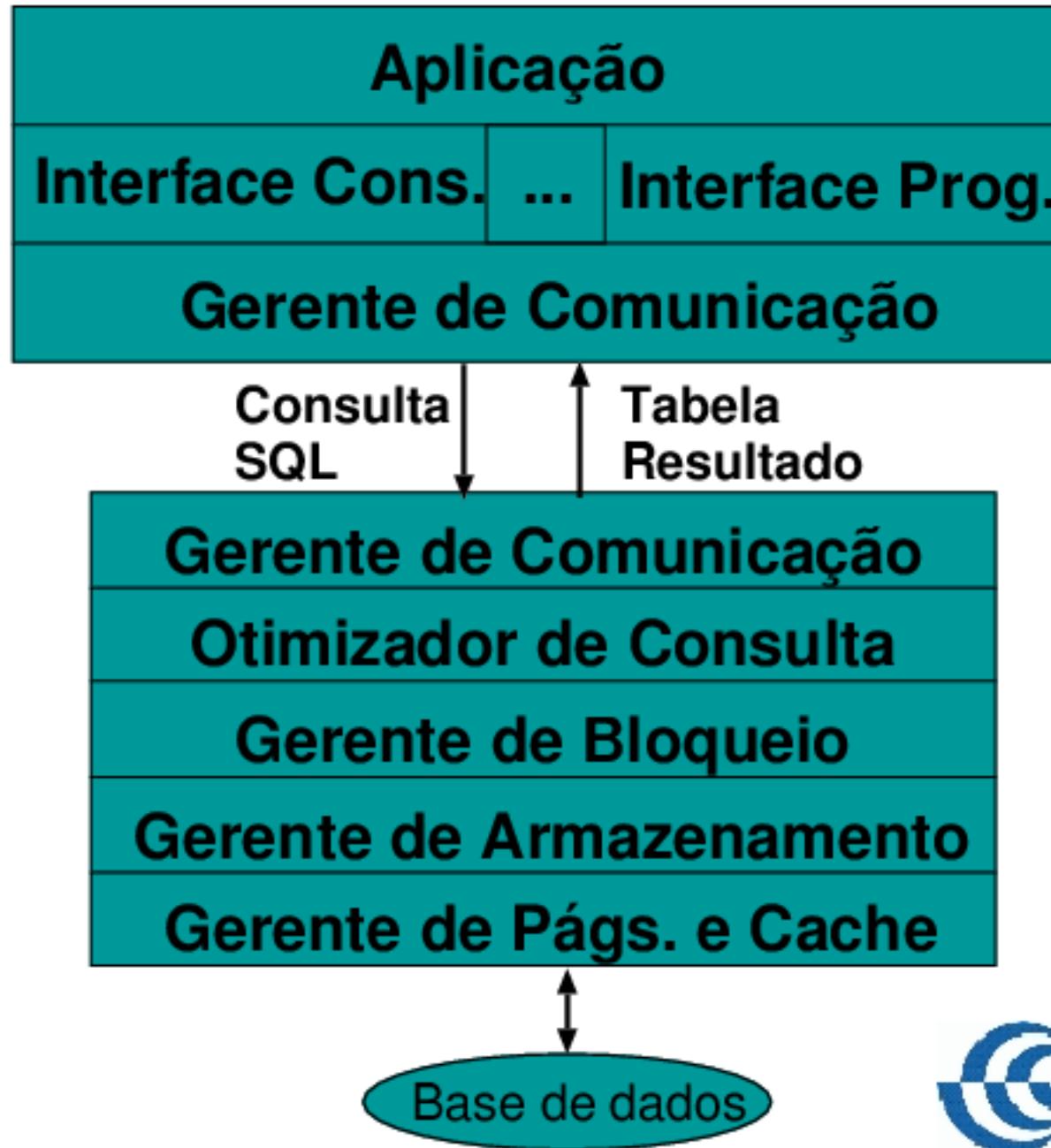


# Interfaces na Arquitetura

LP

API

SGBD



# Interfaces : Aplicação e SGBD

## → Tipos de interfaces

- Linguagem do SGBD, SQL
- Linguagem de Programação
- Gráficas

## → Linguagens do SGBD

- Definição
- Manipulação

# Linguagem de Definição de Dados (LDD)

- Usada pelo ABD e projetistas da base de dados para especificar o *esquema lógico*
- Em vários SGBDs a LDD é também usada para definir os *esquemas internos*

# Linguagem de Manipulação de Dados (LMD)

→ Usada para especificar leituras e atualizações na base de dados

- Comandos da LMD podem ser *embutidos* numa linguagem de programação de propósito geral (**linguagem hospedeira**), como Java, C, Perl
- Ou a LMD pode ser usada de modo isolado como **linguagem de consulta**, aplicada diretamente à base de dados
- LMD no modelo relacional: SQL

# Interfaces do SGBD

- ➔ Interface isolada de Linguagem de Consulta
- ➔ Interface onde a LMD fica embutida numa Linguagem de Programação :
  - Abordagem- pré-compilação
  - Abordagem- chamada de rotina
- ➔ Interfaces gráficas:
  - Menu, para web
  - Formulários, para usuários casuais
  - Linguagem Natural
  - Combinações
- ➔ Interfaces de administração (ABD)

# Interface isolada consultas (MySQL)

```
shell> mysql -h host -u user -p
```

```
Enter password: *****
```

```
Welcome to the MySQL monitor. Commands end with ; or \g.
```

```
Your MySQL connection id is 25338 to server version: 5.0.24-standard
```

```
Type 'help;' or '\h' for help.
```

```
mysql>
```

```
mysql> Select * from CUSTOMER where name = 'Silva  
orderby CUST_NO asc ;
```

# LMD em LP

```
var  
frmClientes: TfrmClientes;  
dbKey: String = 'CUST_NO';  
tbName: String = 'CUSTOMER';
```

...

```
procedure TfrmClientes.btnCompletoClick(Sender: TObject);  
var ordem : String;  
begin  
  SQLQuery1.close;  
  if ( rgrOrdem.ItemIndex = 0 ) then  
    ordem := ' asc'  
  else  
    ordem := ' desc';  
  
  SQLQuery1.SQL.Text = 'Select * from ' + tbName +  
                      ' order by ' + dbKey + ordem;  
  lbISQLEcho.Text := SQLQuery1.SQL.Text;  
  SQLQuery1.Open;  
end;
```

# LMD em LP

```
var  
frmClientes: TfrmClientes;  
dbKey: String = 'CUST_NO';  
tbName: String = 'CUSTOMER';  
...
```

Esse foi um exemplo feito na Aula 17 do curso de Programação I

```
procedure TfrmClientes.btnCompletoClick(Sender: TObject);  
var ordem : String;  
begin  
  SQLQuery1.close;  
  if ( rgrOrdem.ItemIndex = 0 ) then  
    ordem := ' asc'  
  else  
    ordem := ' desc';  
  
  SQLQuery1.SQL.Text = 'Select * from ' + tbName +  
                      ' order by ' + dbKey + ordem;  
  lbISQLEcho.Text := SQLQuery1.SQL.Text;  
  SQLQuery1.Open;  
end;
```

# Embutido em Perl

```
use DBI;
use strict;
$SIG{__DIE__}=\&die_handler;
my $dsn= 'DBI:mysql:test';
my $username= 'mysql';
my $password= 'foo';
my $dbh= DBI->connect($dsn, $username, $password,
    {AutoCommit => 1, RaiseError => 1}) or die_handler();

my $query= "INSERT INTO foo (name) VALUES (?)";

my $sth= $dbh->prepare($query);

for my $val ('jack', 'jill', 'john', 'jason')
{
    $sth->execute($val);
}
```

# Conexão JDBC

Esse exemplo mostra como obter uma instância de Connection a partir do DriverManager.

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

... try {
    Connection conn = DriverManager.getConnection
("jdbc:mysql://localhost/test?user=monty&password=greatsqldb");

    // Trabalha com a conexão

    ...
} catch (SQLException ex) {
    // handle any errors
    System.out.println("SQLException: " + ex.getMessage());
    System.out.println("SQLState: " + ex.getSQLState());
    System.out.println("VendorError: " + ex.getErrorCode());
}
```

## Conexão JDBC (Cont.)

```
... // com a conexão já estabelecida
```

```
Statement stmt = null;
```

```
ResultSet rs = null;
```

```
try {
```

```
    stmt = conn.createStatement();
```

```
    rs = stmt.executeQuery("Select * from CUSTOMER where name = 'Silva'  
                           orderby CUST_NO asc");
```

```
// Agora pode-se trabalhar com o resultado da consulta usando rs  
( ResultSet) ....
```

```
}
```

```
Ao final { // liberar os recursos
```

```
if (rs != null) {
```

```
    try {
```

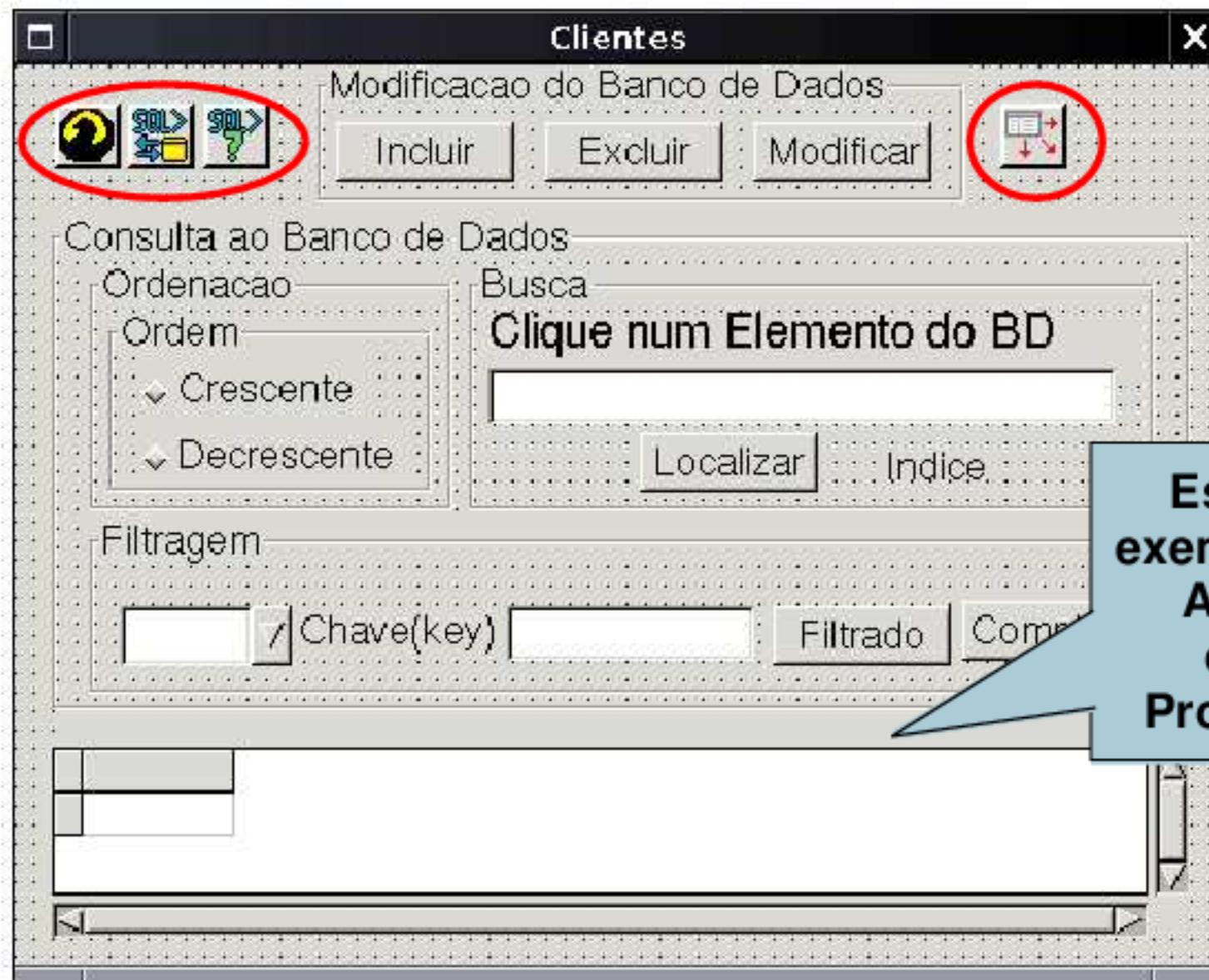
```
        rs.close();
```

```
    } catch (SQLException sqlEx) { // ignore }
```

```
    rs = null;
```

```
}
```

# Menu para a Web



Esse foi um  
exemplo feito na  
Aula 17 do  
curso de  
Programação I

# Formulários - MySQL

**MySQL Table Editor**

Table Name: City Database: world Comment:

Columns and Indices | Table Options | Advanced Options

Column Name	Datatype	NOT NULL	AUTO INC	Flags	Default Value	Comment
Id	INTEGER	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/> UNSIGNED <input type="checkbox"/> ZEROFILL		
Name	CHAR(35)	<input checked="" type="checkbox"/>		<input type="checkbox"/> BINARY <input type="checkbox"/> ASCII <input type="checkbox"/> UNICODE		
Country	CHAR(3)	<input checked="" type="checkbox"/>		<input type="checkbox"/> BINARY <input type="checkbox"/> ASCII <input type="checkbox"/> UNICODE		
District	CHAR(20)	<input checked="" type="checkbox"/>		<input type="checkbox"/> BINARY <input type="checkbox"/> ASCII <input type="checkbox"/> UNICODE		
Population	INTEGER	<input checked="" type="checkbox"/>		<input type="checkbox"/> UNSIGNED <input type="checkbox"/> ZEROFILL	0	

Indices | Foreign Keys | Column Details

**Index Settings:**

- Index Name: PRIMARY
- Index Kind: PRIMARY
- Index Type: BTREE

Index Columns (Use Drag'n'Drop)

- Id
- District

+ -

Apply Changes | Discard Changes | Close

# Formulários - MySQL

The screenshot shows the MySQL Query Browser interface. The query window displays the following SQL code:

```
SELECT C.Country, C.Language, C.IsOfficial FROM CountryLanguage C
WHERE C.IsOfficial = 'T'
ORDER BY C.Country
```

The results are shown in a table titled "Tabsheet 1". The table has three columns: Country, Language, and IsOfficial. The data includes:

Country	Language	IsOfficial
UKR	Russian	T
BLZ	English	T
BMU	English	T
BOL	Spanish	T
BOL	Ket <sup>Q</sup> ua	T
BOL	Aimara	T
BRA	Portuguese	T
BRB	English	T
BRN	Malay	T
BTN	Dzongkha	T
CAN	English	T
CAN	French	T
CCK	English	T
CHE	German	T
CHE	French	T
CHE	Italian	T
CHE	Romanish	T
CHL	Spanish	T

At the bottom left, it says "238 rows in set.". Below the table are buttons for Edit, Apply Changes, Discard Changes, First, Last, and Search.

The right side of the interface includes a "Schemata" panel listing databases: bugs, mysql, test, vbmysql, vbmysqlforum, and webinos. There are also "Params" and "Functions" sections.

# Administração

MySQL Administrator - root@localhost:3306

File Edit View Tools Extras Windows Help

User Information Global Privileges Schema Privileges Table/Column Privileges Resources

**jack, (Jack Foley)**  
Login and additional information on the user

Login Information

MySQL User:  The user has to enter this MySQL User name to connect to the MySQL Server

Password:  Fill out this field if you want to set the user's password

Confirm password:  Again, enter the user's password to confirm

Additional Information

Full name:  The user's full name

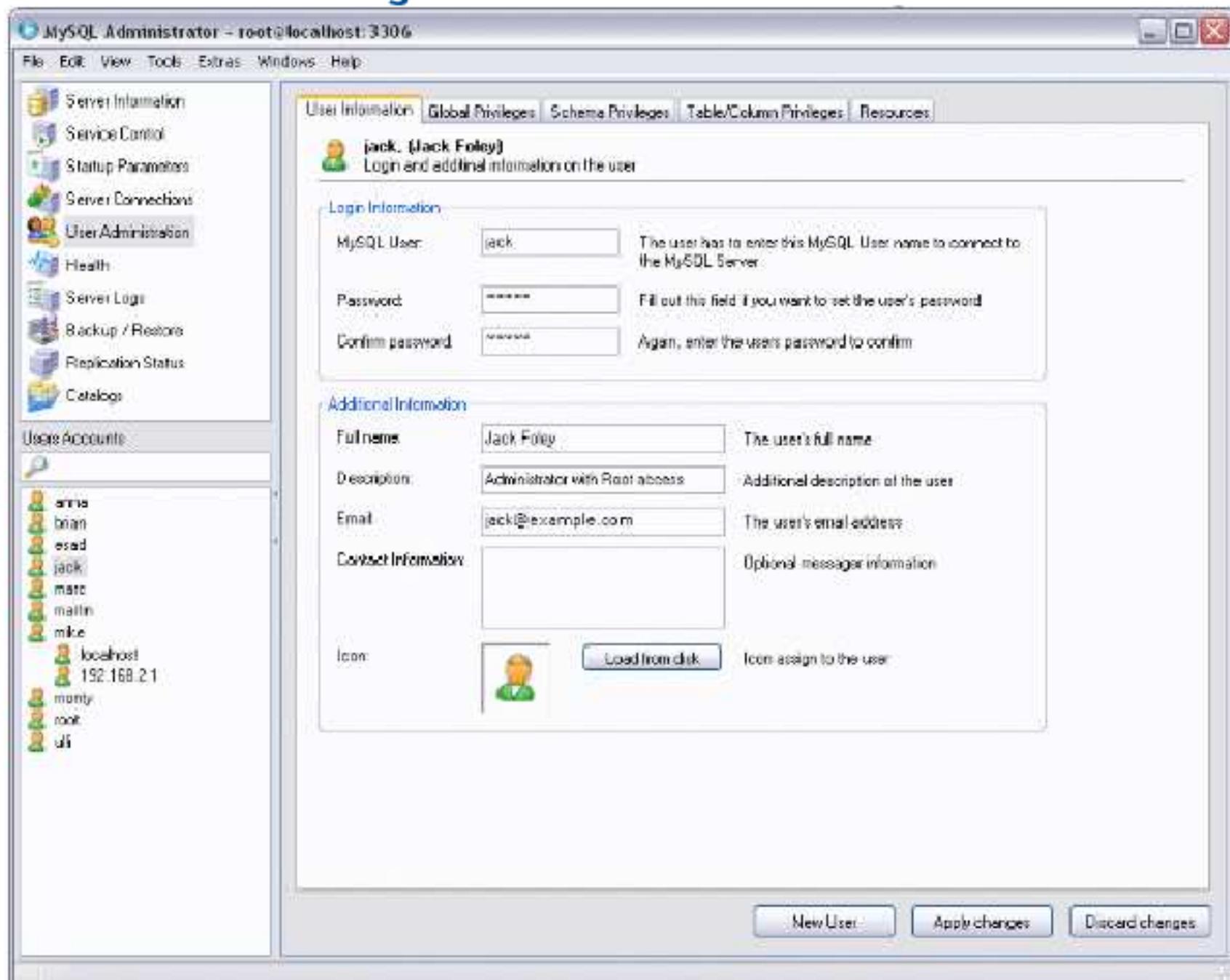
Description:  Additional description of the user

Email:  The user's email address

Contact Information:  Optional messenger information

Icon:  Load from disk Icon assign to the user

New User Apply changes Discard changes



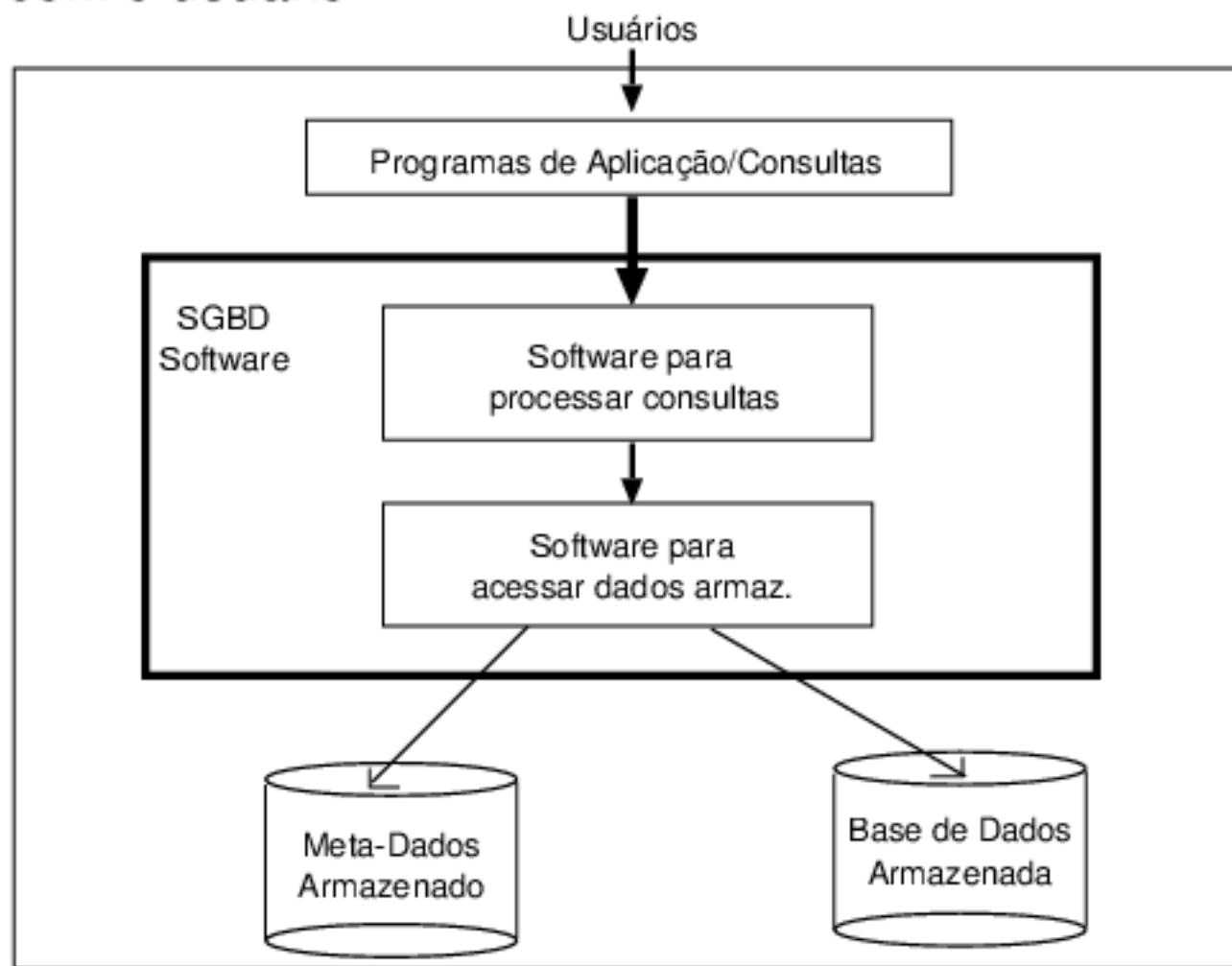
The screenshot shows the MySQL Administrator application window. The left sidebar contains navigation links like Server Information, Service Control, Startup Parameters, Server Connections, User Administration (which is selected), Health, Server Logs, Backup / Restore, Replication Status, and Catalogs. Below this is a list of User Accounts with icons: anna, brian, esed, jack (selected), matt, matin, mike, localhost, 192.168.21, monty, root, and uli. The main panel is titled 'User Information' and shows details for the 'jack, (Jack Foley)' user. It includes tabs for Global Privileges, Schema Privileges, Table/Column Privileges, and Resources. Under 'Login Information', fields are shown for MySQL User (jack), Password, and Confirm password. Under 'Additional Information', fields are shown for Full name (Jack Foley), Description (Administrator with Root access), Email (jack@example.com), Contact Information (empty), and an Icon selection area. At the bottom are buttons for New User, Apply changes, and Discard changes.

# Arquiteturas baseadas em funções

- Centralizada
- Cliente/servidor
- Servidores específicos
- N-camadas (cliente/servidor)
- Distribuída
- Heterogêneas

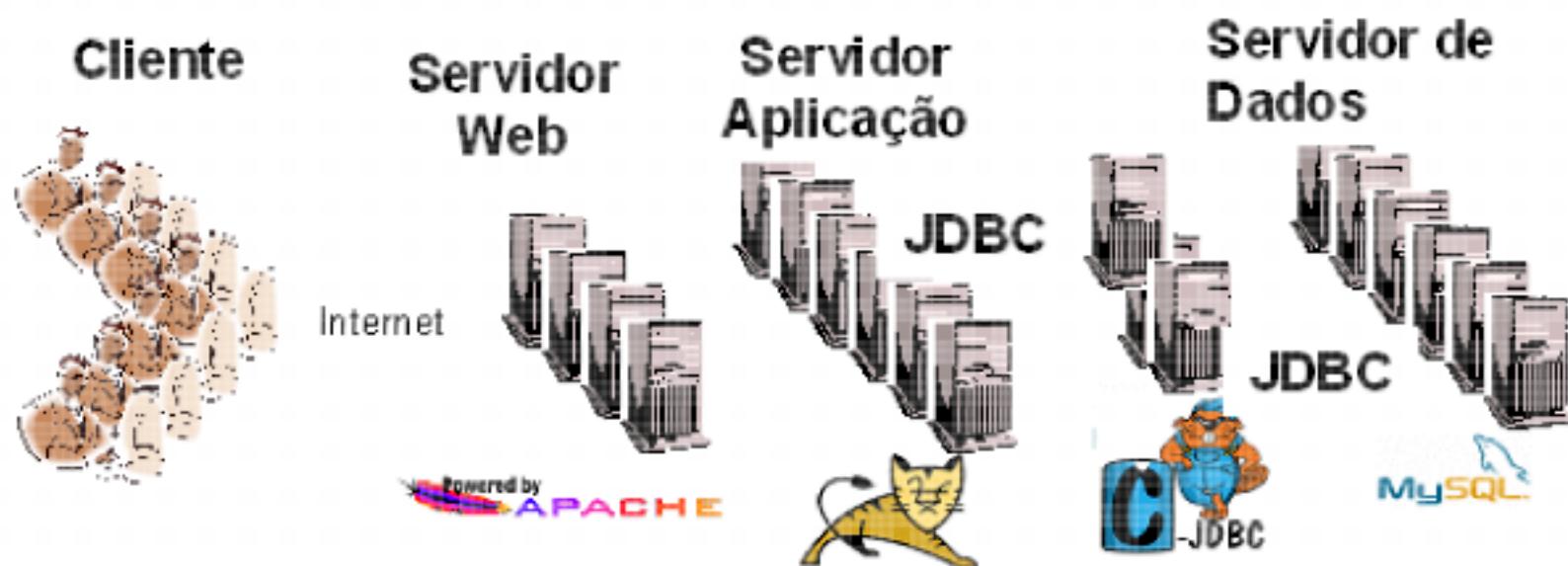
# Centralizada

→ Combina todos os componentes num único sistema incluindo o software do SGBD, Hardware, programas de aplicação e interfaces com o usuário



# Arquiteturas Cliente-Servidor

- Clientes
- Servidores especializados com funções específicas
- Servidor de Dados



<http://o-jdbc.objectweb.org/> - o-jdbc@objectweb.org

Slides adaptados de Emmanuel Cecchet

# Vantagens Arquitetura Cliente/Servidor

- Divisão de trabalho mais adequada
- Escalonamento de recursos horizontal e vertical
- Máquinas cliente com bom preço/desemp.
- Pode-se usar ferramentas conhecidas (cliente)
- Cliente acessa dados remotos (via padrões)
- Todos serviços do SGBD disponíveis p/ cliente
- Melhor preço/desemp. do sistema global

## Cientes:

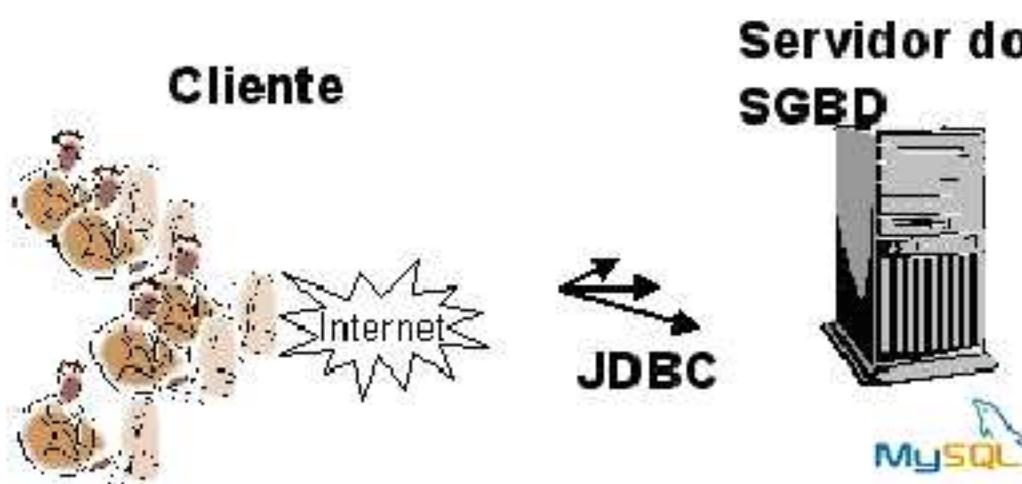
- Prove as interfaces apropriadas e uma versão-cliente do SGBD para ter acesso e utilizar os recursos do servidor
- Podem ser máquinas sem discos, ou PCs ou estações de trabalho com discos contendo apenas software do cliente instalado
- São conectados aos servidores através de alguma forma de (LAN: rede local, rede sem fio, etc.)

# Servidor do SGBD

- Permite consultas e processamento de transações no SGBD aos clientes
- Às vezes chamado de servidor de dados ou servidor de consultas e transações

# Arquitetura Cliente/Servidor 2-camadas

- Interface com usuário e programas da aplicação executam no lado do cliente
- Interface chamada **ODBC (Open Database Connectivity** - ver Capítulo 9) fornece uma Interface ao Programa da Aplicação (API) que permite aos programas no lado do cliente chamadas ao SGBD. Os fabricantes de SGBD em geral fornecem *drivers* ODBC e **JDBC** (Java).



# Arquitetura Cliente/Servidor 4-camadas

- ➡ Muito usada em aplicações Web
- ➡ Servidor Web - conectividade (*applets, browser*)
- ➡ Servidor Aplicação - regras de negócio
- ➡ Servidor de Dados - consultas e transações no SGBD
- ➡ Naturalmente estende para *n*-camadas



# Arquitetura Cliente/Servidor Distribuído

- SGBDs executam em clusters de PCs
- Consultas são distribuídas aos servidores de dados (SGBDS) gerenciadas por um servidor de distribuição

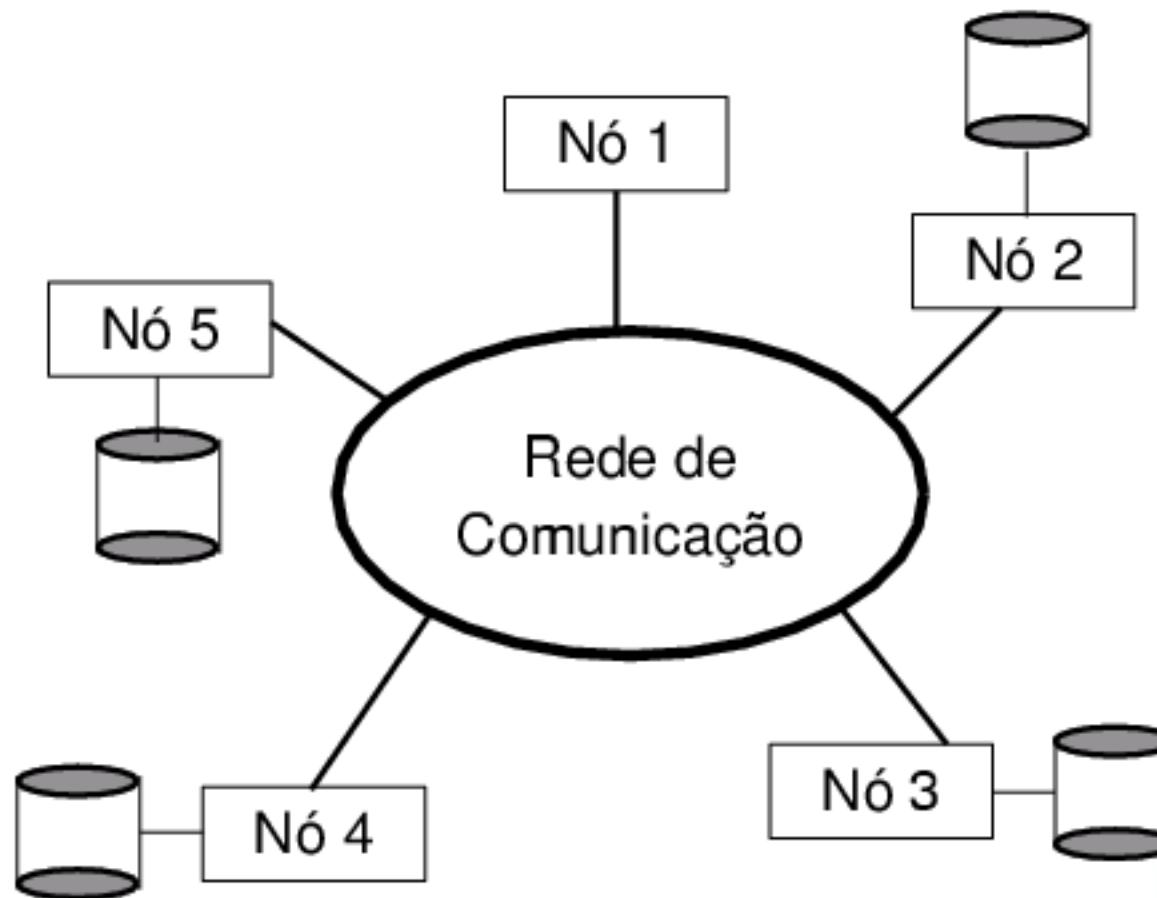


<http://c-jdbc.objectweb.org/> - c-jdbc@objectweb.org

Slides adaptados de Emmanuel Cecchet

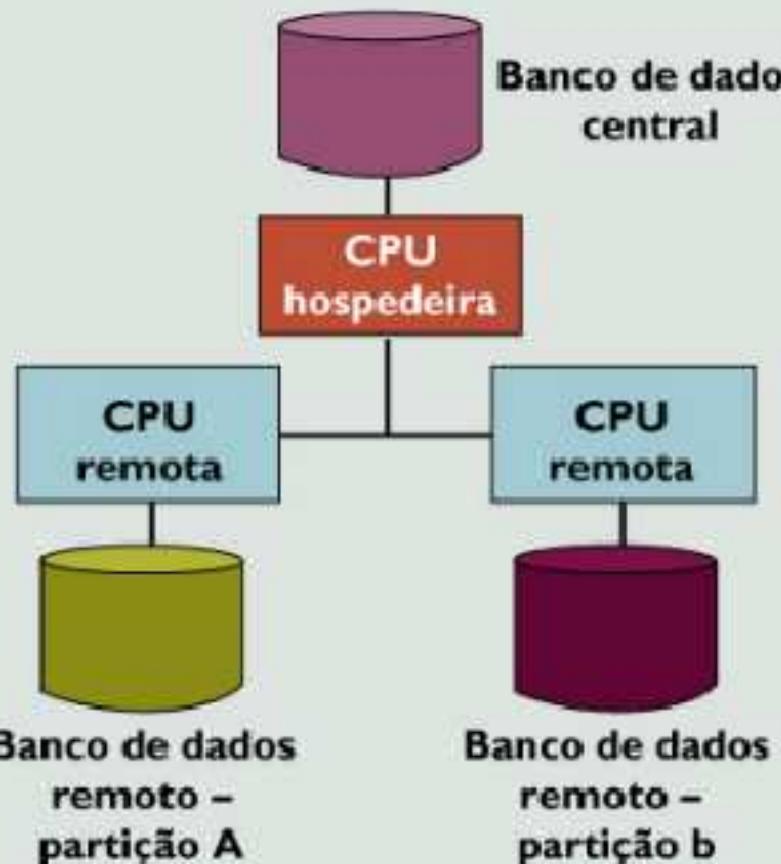
# SGBD Distribuído

→ O servidor do SGBD pode ser um software especial para acessar bases de dados distribuídos

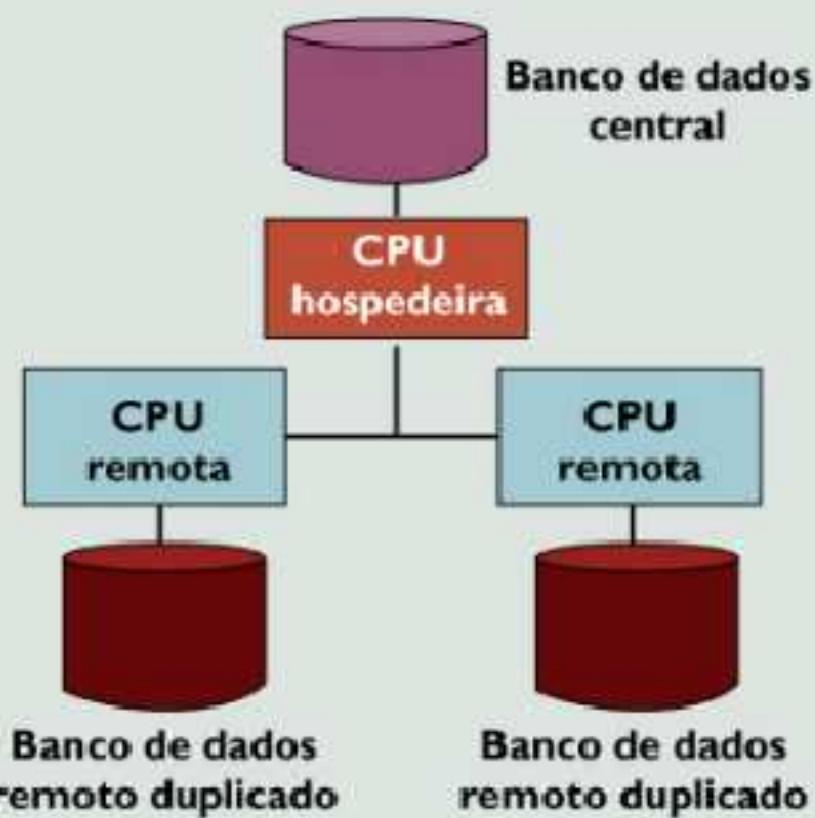


# SGBD Distribuído

(a) Banco de dados particionado



(b) Banco de dados duplicado



Fonte: Laudon e Laudon - Editora Pearson 204

# Variações de ambientes distribuídos

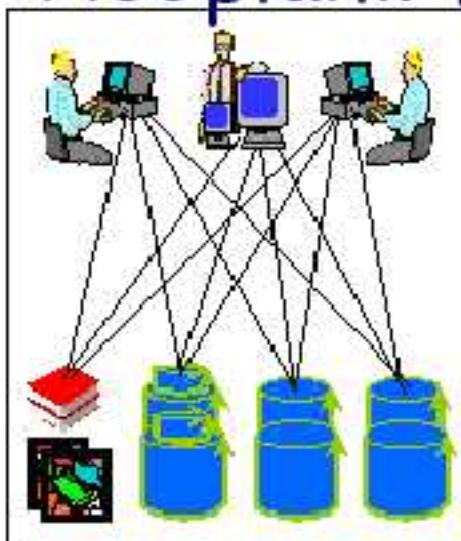
- SGBD-D homogêneos
- SGBD-D heterogêneos
- Sistemas federados ou multi-bases de dados

# Integração de Sistemas Heterogêneos (sistemas legados)

- Ponto único de acesso a fontes de informações distribuídas e heterogêneas
- Visão homogênea sobre dados heterogêneos
- Acesso eficiente a dados relevantes
- Acesso uniforme via Internet - Intranet

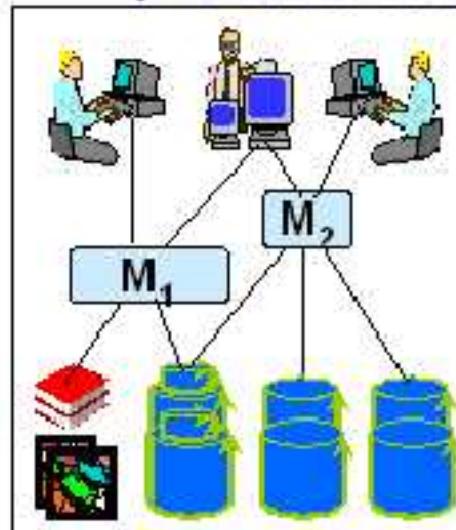
# Sistemas Legados

Acoplam. fraco



Um módulo de  
acesso por  
aplicação

Federado



Um mediador  
por domínio

Integrado



Um módulo  
global de  
integração

## Exercício

→ Quais dos seguintes itens desempenham papéis fundamentais na representação de informações do mundo real numa base de dados? Explique resumidamente.

1. Linguagem de definição de dados
2. Linguagem de manipulação de dados
3. Gerente de páginas e cache
4. O modelo de dados

# Resposta

Vamos ver um de cada vez:

1. A LDD é importante na representação da informação pois é utilizada para definir o esquema lógico
2. A LMD é usada para acesso e modificação de dados, não é importante na representação dos dados
3. O gerente de páginas e cache não é muito importante na representação, pois traz páginas arbitrárias para a memória principal independente de como os dados estejam representados
4. O modelo de dados é fundamental na representação da informação. Ele determina que mecanismos de representação de dados estão presentes no SGBD. A LDD possui o conjunto de construtores da linguagem específicos ao modelo de dados do SGBD.