



UNIVERSIDADE\*FEDERAL  
DE\*MINAS\*GERAIS

Introdução a Banco de Dados

# Modelagem

Rodrygo L. T. Santos  
[rodrygo@dcc.ufmg.br](mailto:rodrygo@dcc.ufmg.br)

# Modelo de dados, esquema, instância

**Modelo de dados:** construções reutilizáveis

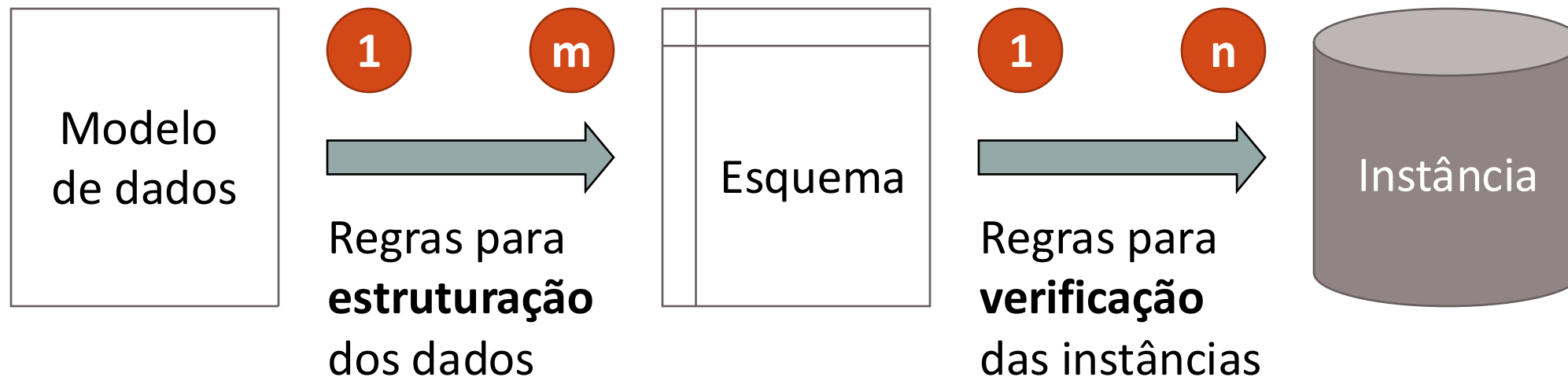
- Entidades + relacionamentos + restrições  
(+operações de leitura e escrita)

**Esquema:** descrição dos dados a partir de um modelo

- Notação textual ou gráfica

**Instância:** estado de um banco de dados

# Modelo de dados, esquema, instância



# Modelo

*Information Retrieval*

P. BAXENDALE, Editor

---

## A Relational Model of Data for Large Shared Data Banks

E. F. CODD

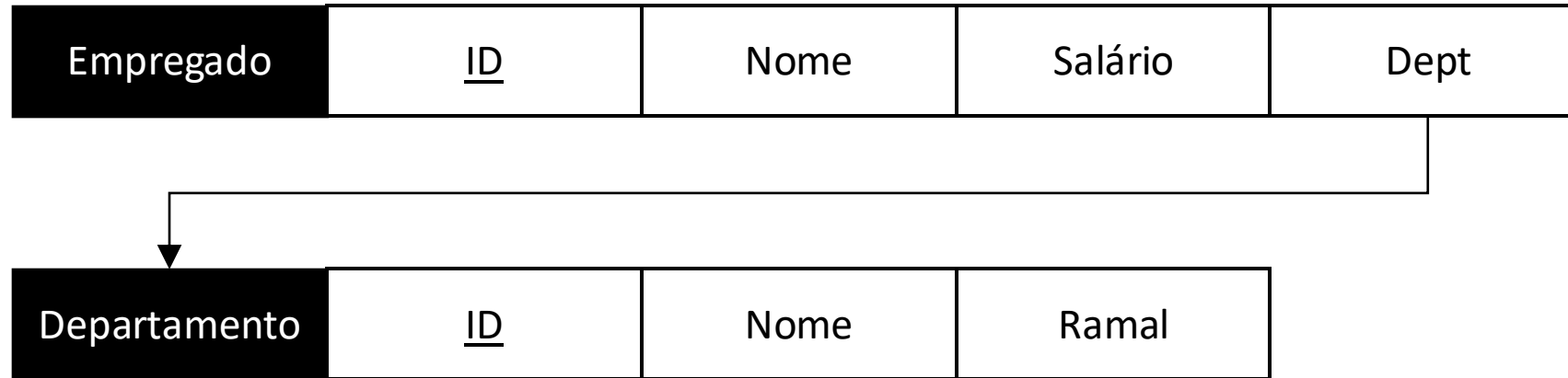
*IBM Research Laboratory, San Jose, California*

Future users of large data banks must be protected from having to know how the data is organized in the machine (the internal representation). A prompting service which supplies such information is not a satisfactory solution. Activities of users at terminals and most application programs should remain

The relational view (or model) of data described in Section 1 appears to be superior in several respects to the graph or network model [3, 4] presently in vogue for non-inferential systems. It provides a means of describing data with its natural structure only—that is, without superimposing any additional structure for machine representation purposes. Accordingly, it provides a basis for a high level data language which will yield maximal independence between programs on the one hand and machine representation and organization of data on the other.

A further advantage of the relational view is that it forms a sound basis for treating derivability, redundancy, and consistency of relations—these are discussed in Section 2. The network model, on the other hand, has spawned a

# Esquema



**Empregado.ID** deve ser **único** e **não-nulo**

**Departamento.ID** deve ser **único** e **não-nulo**

**Empregado.Dept** deve ser **não-nulo**

**Empregado.Dept** deve referenciar um valor válido de **Departamento.ID**

# Instância

Empregado			
ID	Nome	Salário	Dept
032	J Silva	380	21
074	M Reis	400	25
089	C Melo	520	28
092	R Silva	480	25
112	R Pinto	390	21
121	V Simão	905	28

Departamento		
ID	Nome	Ramal
21	Pessoal	142
25	Financeiro	143
28	Técnico	144

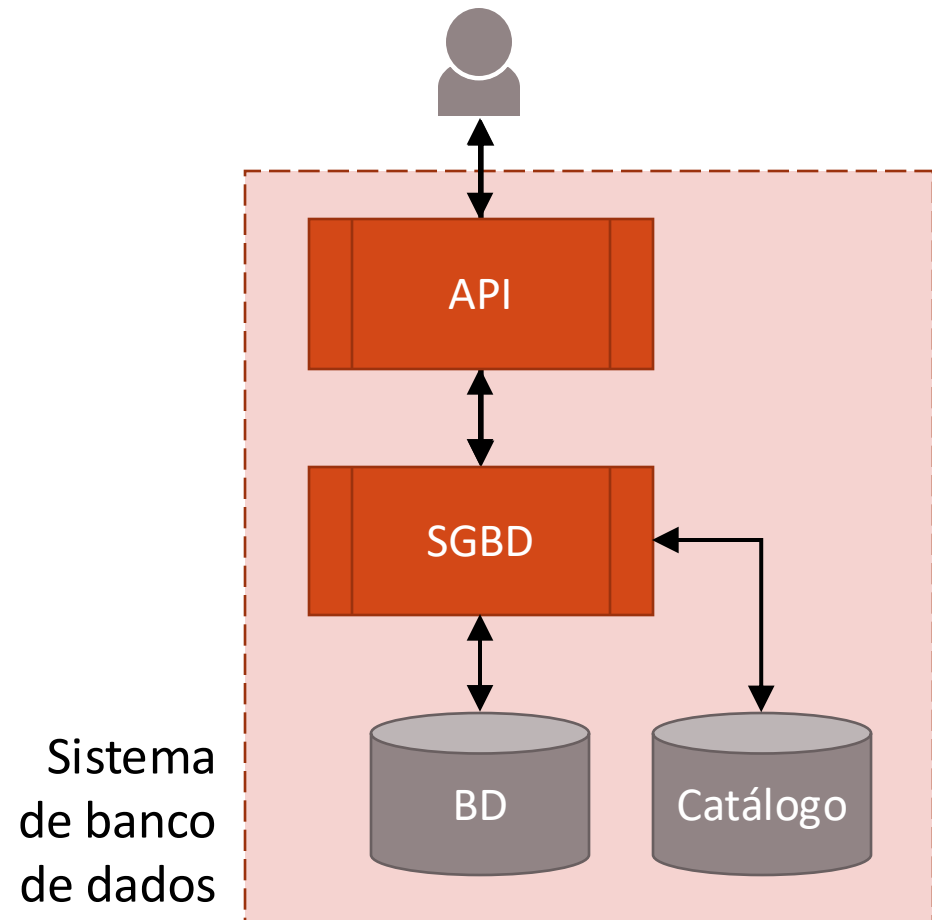
# Esquemas e instâncias no SGBD

## Instâncias → banco de dados

- Estado inicialmente vazio
- Muda frequentemente

## Esquema → catálogo

- Criado antes das instâncias
- Tende a mudar pouco



Introdução a Banco de Dados

# Categorias de Modelos

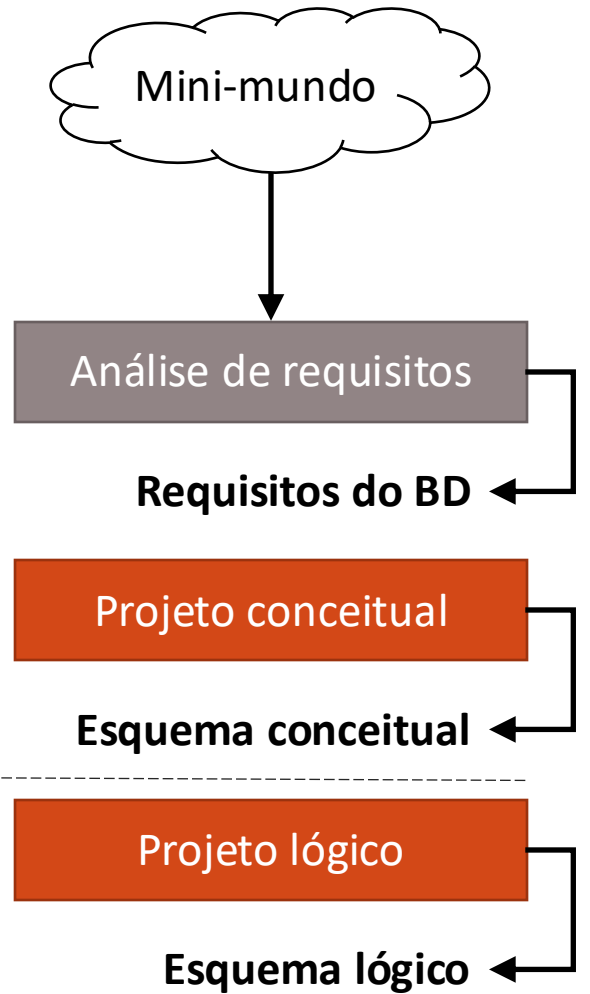
Rodrygo L. T. Santos  
rodrygo@dcc.ufmg.br



# Projeto de banco de dados (em IBD)

Independente de SGBD

Dependente de classe de SGBD



# Modelos conceituais

Descrevem a estrutura de um BD de uma forma mais próxima da percepção dos usuários

- Independentes de aspectos de implementação
- Construções: entidades, relacionamentos, atributos

# Modelo entidade-relacionamento (ER)

## The Entity-Relationship Model—Toward a Unified View of Data

PETER PIN-SHAN CHEN

Massachusetts Institute of Technology

---

A data model, called the entity-relationship model, is proposed. This model incorporates some of the important semantic information about the real world. A special diagrammatic technique is introduced as a tool for database design. An example of database design and description using the model and the diagrammatic technique is given. Some implications for data integrity, infor-

# Modelos lógicos (representacionais)

Descrevem a estrutura de um BD da forma como será manipulado através do SGBD

- Independentes de um SGBD específico
- Construções: estruturas de dados

# Modelo relacional

*Information Retrieval*

P. BAXENDALE, Editor

---

## A Relational Model of Data for Large Shared Data Banks

E. F. CODD

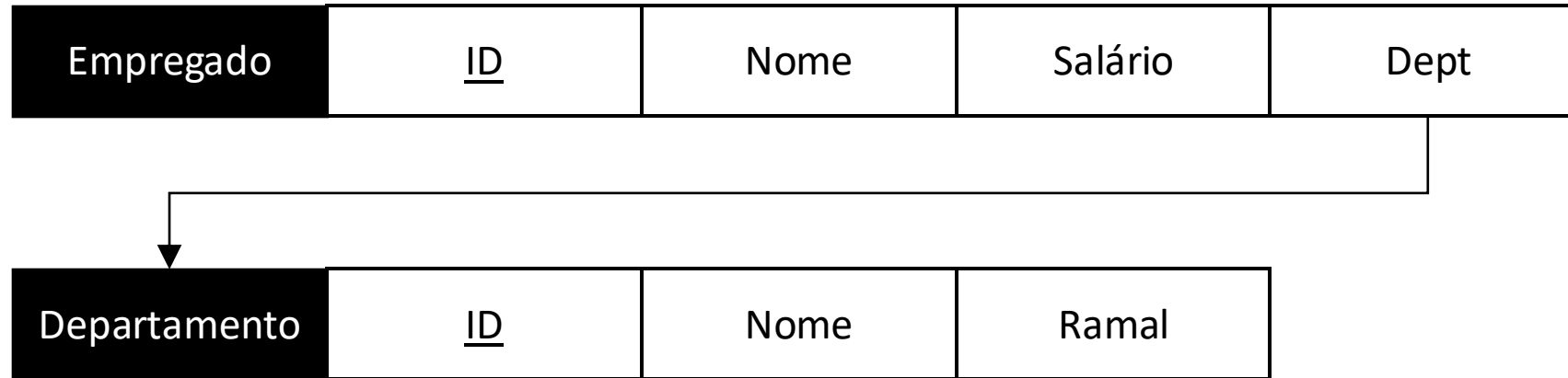
*IBM Research Laboratory, San Jose, California*

Future users of large data banks must be protected from having to know how the data is organized in the machine (the internal representation). A prompting service which supplies such information is not a satisfactory solution. Activities of users at terminals and most application programs should remain

The relational view (or model) of data described in Section 1 appears to be superior in several respects to the graph or network model [3, 4] presently in vogue for non-inferential systems. It provides a means of describing data with its natural structure only—that is, without superimposing any additional structure for machine representation purposes. Accordingly, it provides a basis for a high level data language which will yield maximal independence between programs on the one hand and machine representation and organization of data on the other.

A further advantage of the relational view is that it forms a sound basis for treating derivability, redundancy, and consistency of relations—these are discussed in Section 2. The network model, on the other hand, has spawned a

# Esquema relacional



**Empregado.ID** deve ser **único** e **não-nulo**

**Departamento.ID** deve ser **único** e **não-nulo**

**Empregado.Dept** deve ser **não-nulo**

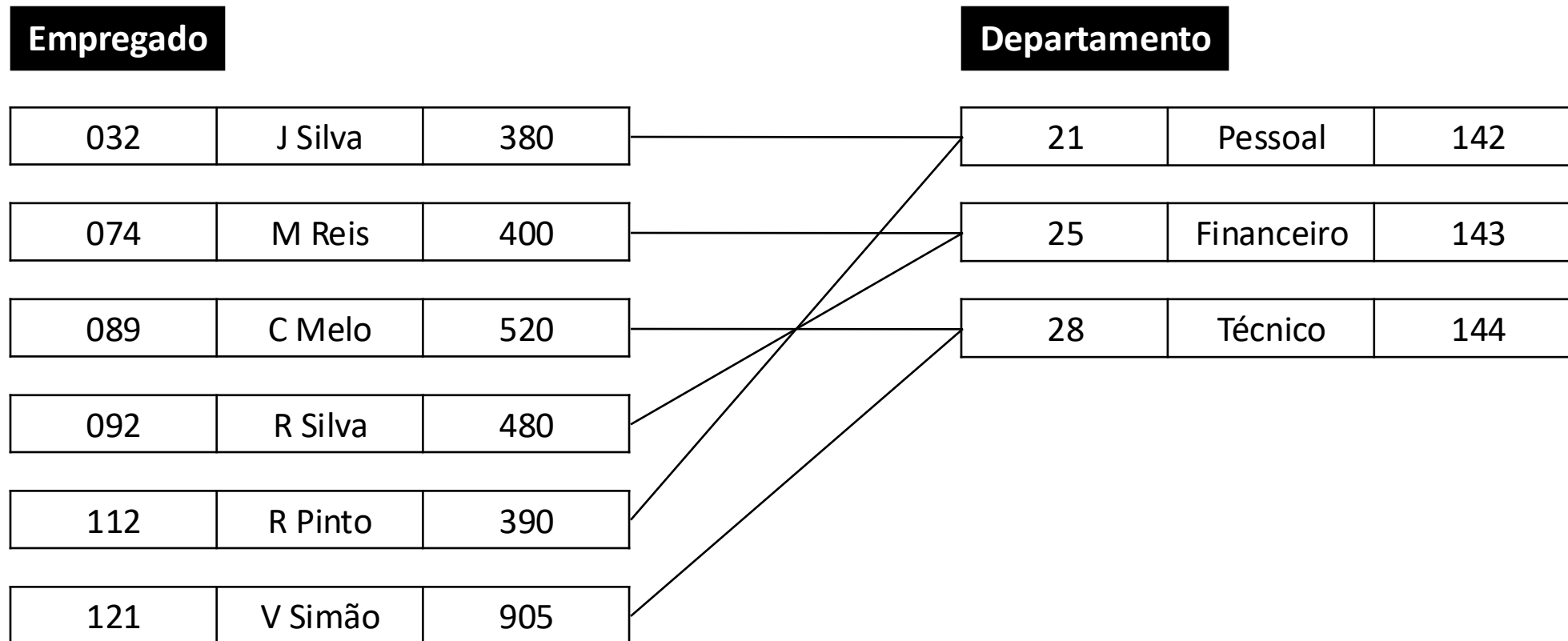
**Empregado.Dept** deve referenciar um valor válido de **Departamento.ID**

# Instâncias de um BD relacional

Empregado			
ID	Nome	Salário	Dept
032	J Silva	380	21
074	M Reis	400	25
089	C Melo	520	28
092	R Silva	480	25
112	R Pinto	390	21
121	V Simão	905	28

Departamento		
ID	Nome	Ramal
21	Pessoal	142
25	Financeiro	143
28	Técnico	144

# Instâncias de um BD em rede





# Instâncias de um BD hierárquico

## Departamento

21	Pessoal	142	25	Financeiro	143	28	Técnico	144
----	---------	-----	----	------------	-----	----	---------	-----

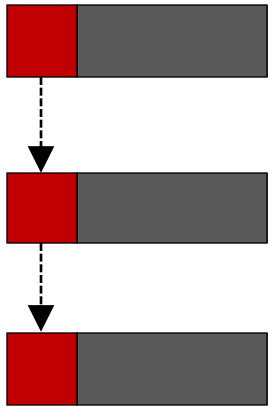
## Empregado

032	J Silva	380	074	M Reis	400	089	C Melo	520
112	R Pinto	390	092	R Silva	480	121	V Simão	905

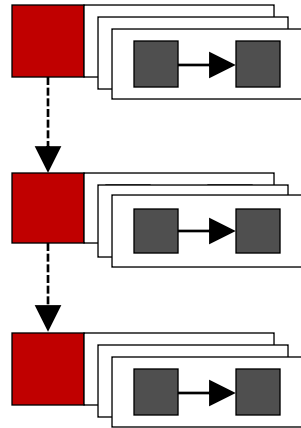
# Modelos não-relacionais modernos



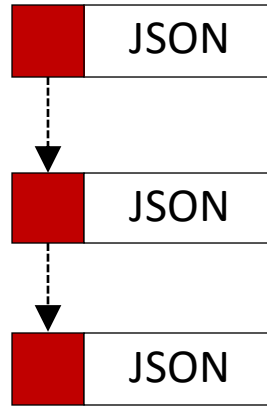
*chave-valor*



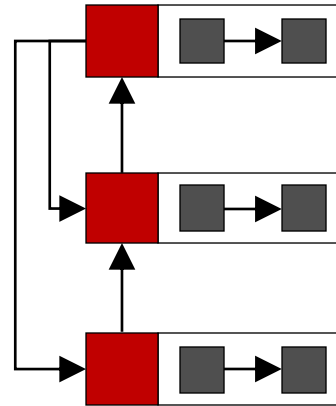
*fam. colunas*



*documentos*



*grafos*



# Modelos físicos

Descrevem como os dados são armazenados

- Dependentes de um SGBD específico
- Construções: definição de tipos, formato dos registros, estruturas de índices, compressão



UNIVERSIDADE FEDERAL  
DE MINAS GERAIS

Introdução a Banco de Dados

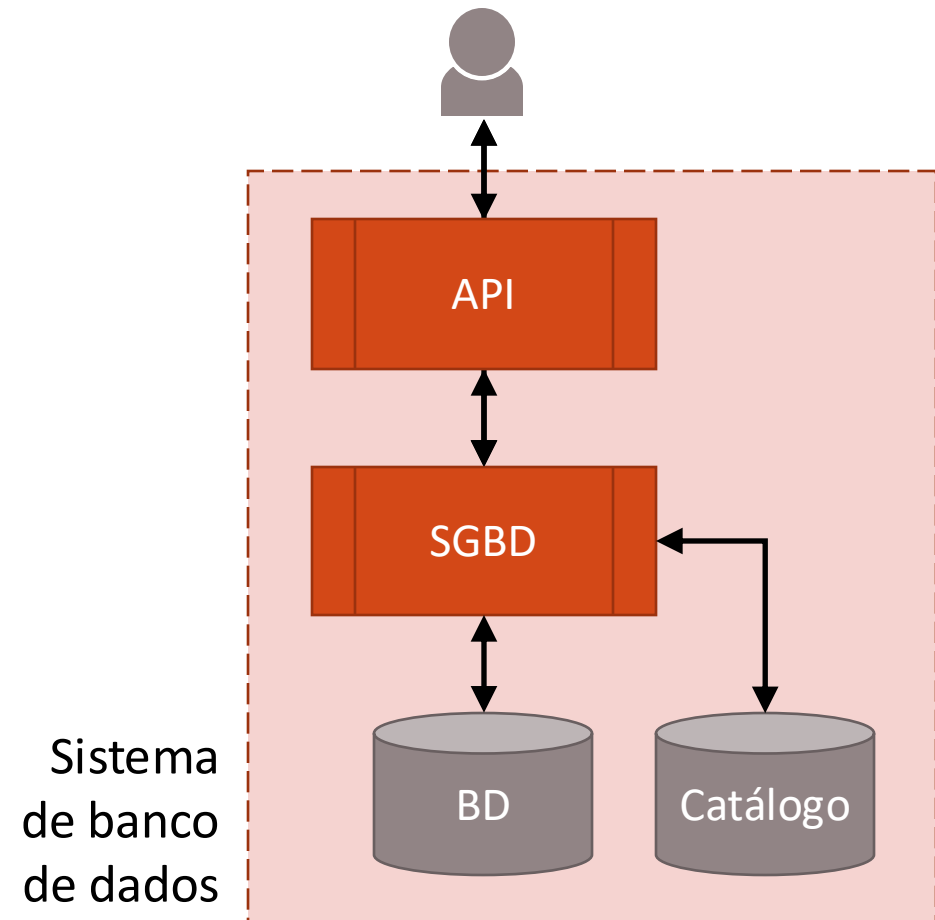
# Arquitetura de SGBDs

Rodrygo L. T. Santos  
[rodrygo@dcc.ufmg.br](mailto:rodrygo@dcc.ufmg.br)

# Sist. de gerência de banco de dados (SGBD)

## Requisitos básicos

- Abstração de acesso aos dados armazenados
- Suporte a múltiplas visões dos dados armazenados



# Arquitetura de três níveis

**Nível externo:** o que será exibido

Descreve visões dos dados

---

**Nível conceitual:** o que será armazenado

Descreve entidades, relacionamentos, operações

---

**Nível interno:** como será armazenado

Descreve o armazenamento físico



# Arquitetura de três níveis

**Nível conceitual:** o que será armazenado

**Empregado**

- **ID** int
- **Nome** varchar(100)
- **Salário** decimal(8,2)
- **Dept** int



# Arquitetura de três níveis

**Nível externo:** o que será exibido

## Empregado

- **ID** int
- **Nome** varchar(100)
- **Salário** decimal(8,2)
- **Dept** int

## Pagamento

- **ID** int
- **Salário** decimal(8,2)



The diagram illustrates a three-level architecture. On the left, under the heading 'Nível externo: o que será exibido', there are two entity sets: 'Empregado' and 'Pagamento'. 'Empregado' has attributes ID (int), Nome (varchar(100)), Salário (decimal(8,2)), and Dept (int). 'Pagamento' has attributes ID (int) and Salário (decimal(8,2)). To the right of these is a large orange rectangle representing the 'SGBD' (Database Management System).

SGBD



# Arquitetura de três níveis

**Nível externo:** o que será exibido

## Empregado

- **ID** int
- **Nome** varchar(100)
- **Salário** decimal(8,2)
- **Dept** int

## Alocação

- **ID** int
- **Dept** int



The diagram illustrates a three-level architecture. On the left, the 'Nível externo' (External Level) is represented by a list of attributes for an 'Empregado' (Employee): ID (int), Nome (varchar(100)), Salário (decimal(8,2)), and Dept (int). In the center, the 'Alocação' (Allocation) level is shown as a light gray box containing a list of attributes: ID (int) and Dept (int). On the right, the 'SGBD' (Database Management System) is represented by a tall orange rectangle. The text 'SGBD' is centered within this rectangle.

SGBD

# Arquitetura de três níveis

**Nível interno:** como será armazenado

**Empregado**

- **ID** int
- **Nome** varchar(100)
- **Salário** decimal(8,2)
- **Dept** int

**Empregado**

- **ID** byte(4)
- **Nome** byte(100)
- **Salário** byte(4)
- **Dept** byte(4)



# Independência de dados

**Nível externo:** o que será exibido

Descreve visões dos dados

---

**Nível conceitual:** o que será armazenado

Descreve entidades, relacionamentos, operações

---

**Nível interno:** como será armazenado

Descreve o armazenamento físico



# Independência de dados

**Nível externo:** o que será exibido

Descreve visões dos dados

— INDEPENDÊNCIA LÓGICA —

**Nível conceitual:** o que será armazenado

Descreve entidades, relacionamentos, operações

— INDEPENDÊNCIA FÍSICA —

**Nível interno:** como será armazenado

Descreve o armazenamento físico



# Independência de dados lógica

Capacidade de alterar o esquema conceitual sem alterar esquema externo ou aplicações

- Exemplo: adição de novas entidades

Pode ser facilmente violada

- Exemplo: remoção de uma tabela ou coluna

# Independência de dados física

Capacidade de mudar o esquema interno sem mudar o esquema conceitual (ou externo)

- Exemplo: alteração do mecanismo de compressão

Confere um bom nível de isolamento

- Usuários poupados de detalhes de implementação
- Desenvolvedores livres para implementar melhorias

Introdução a Banco de Dados

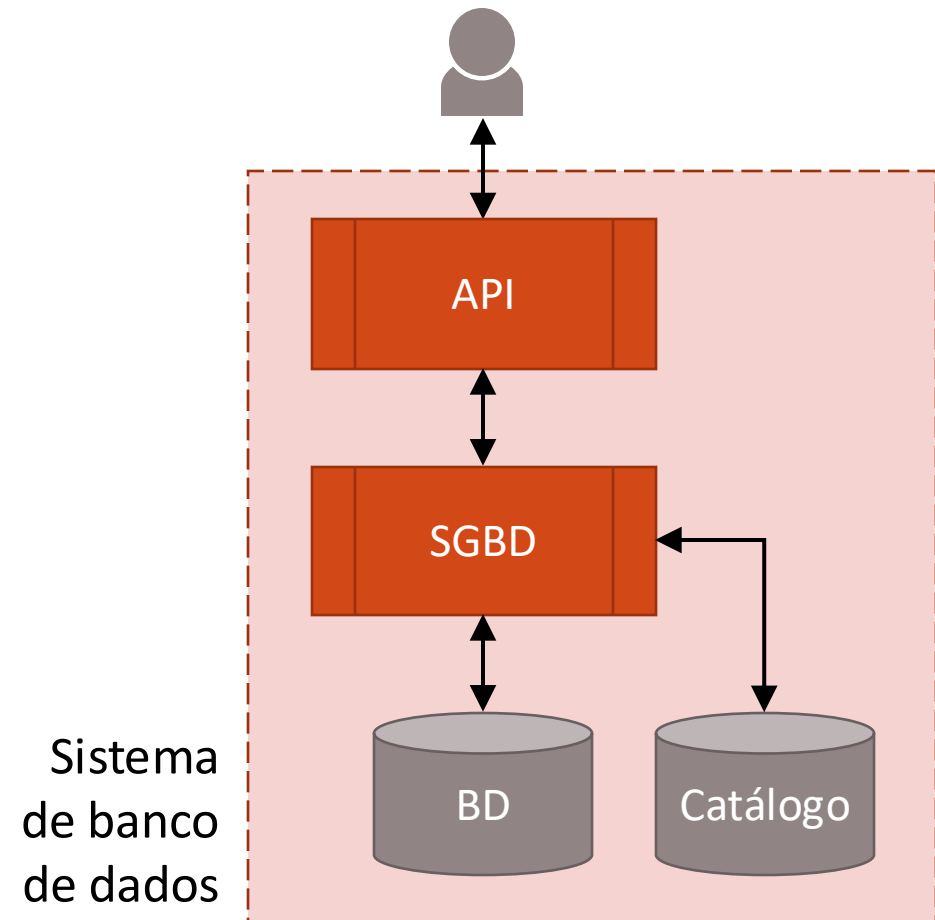
# Componentes de SGBDs

Rodrygo L. T. Santos  
rodrygo@dcc.ufmg.br

# Sist. de gerência de banco de dados (SGBD)

## Requisitos básicos

- Abstração de acesso aos dados armazenados
- Suporte a múltiplas visões dos dados armazenados





# Abstração de acesso

Linguagem de definição de dados (DDL)

- Definição de esquemas

Linguagem de manipulação de dados (DML)

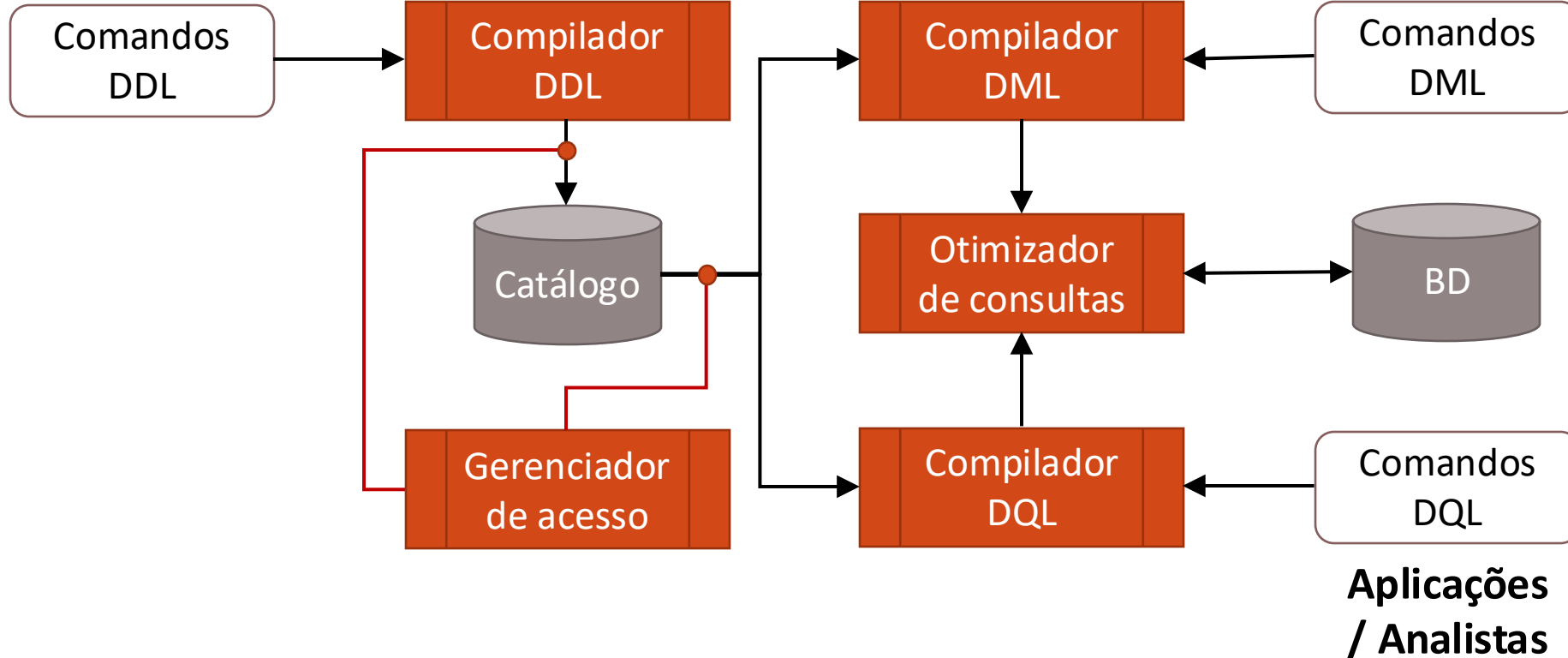
- Inserção, remoção, modificação de dados

Linguagem de consulta de dados (DQL)

- Seleção de dados de interesse

# Componentes básicos

**Administrador  
/ Projetista**



# Componentes adicionais

## Controle de concorrência

- Permite acesso multiusuário consistente

## Distribuição de dados

- Permite escalabilidade e disponibilidade

## Monitoramento de desempenho

- Fornece estatísticas para o DBA

# Componentes utilitários

## Carga (*loading*)

- Importação de dados em arquivo (e.g., CSV)

## Cópia de segurança (*backup*)

- Restauração em caso de falhas

## (Re-)Organização de arquivos

- Melhoria do desempenho