

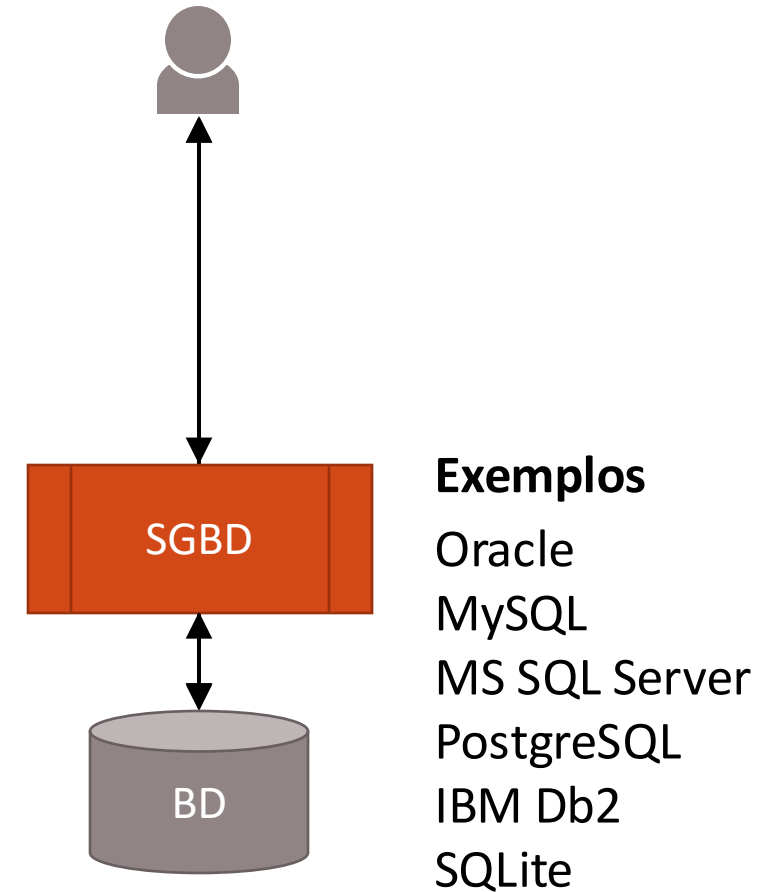
## Introdução a Banco de Dados

# Revisão

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

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

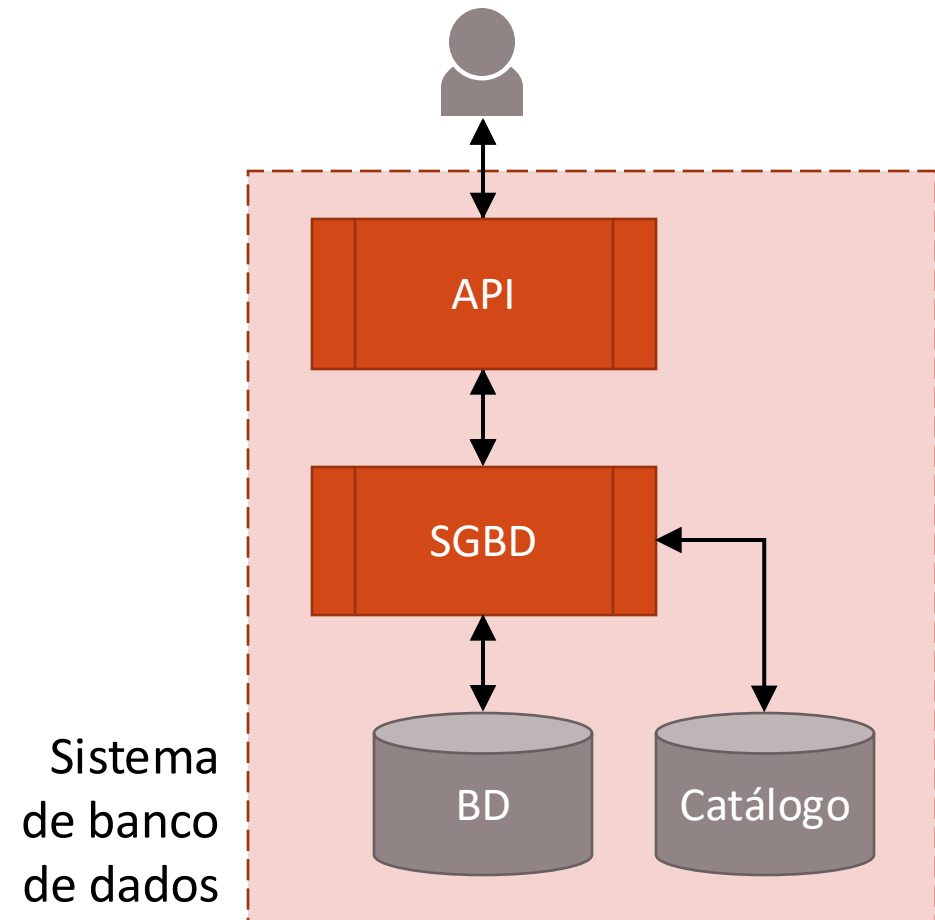
**SGBD:** conjunto de programas que permite criar e manter (manipular) um banco de dados



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

**SGBD:** conjunto de programas que permite criar e manter (manipular) um banco de dados

- **BD + SGBD** = sistema de banco de dados



# 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



# **Vantagens da utilização de um SGBD**

Armazenamento persistente dos dados

Autodescrição dos dados (meta-dados)

Isolamento entre programas e dados

Suporte a múltiplas visões dos dados

Compartilhamento de dados

Garantias de consistência

# Quando não usar um SGBD

Aplicações simples, sem mudanças frequentes

- Investimento inicial é alto
- Modelagem foca em generalidade, não eficiência

Aplicações monousuário, ou de tempo real

- Custo adicional para prover outras facilidades funcionais (segurança, concorrência, etc.)

# Modelo de dados, esquema, instância

**Modelo de dados:** construções reutilizáveis

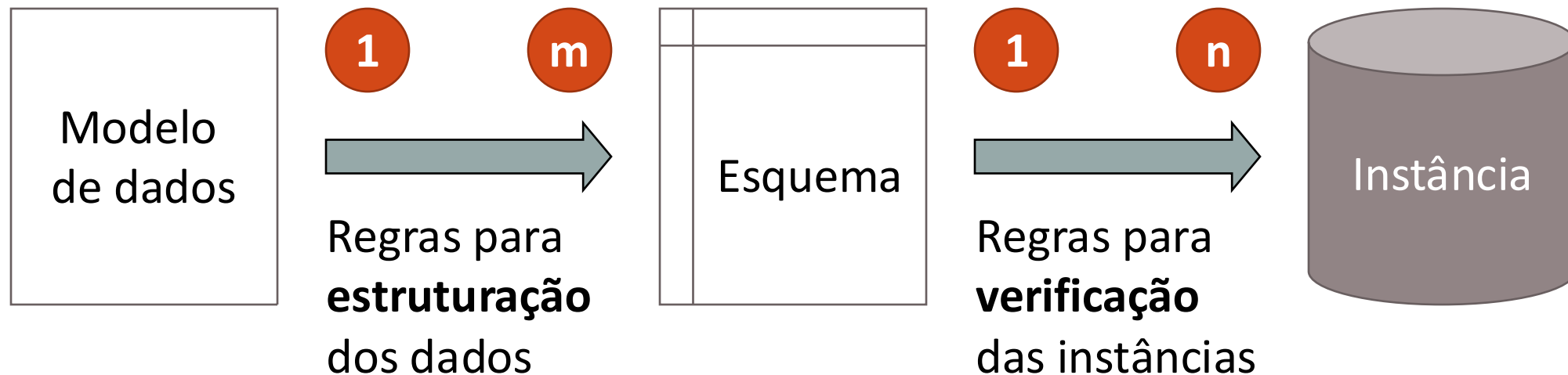
- Entidades + relacionamentos + restrições  
(+operações de leitura e escrita)

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

- Notação textual ou gráfica

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

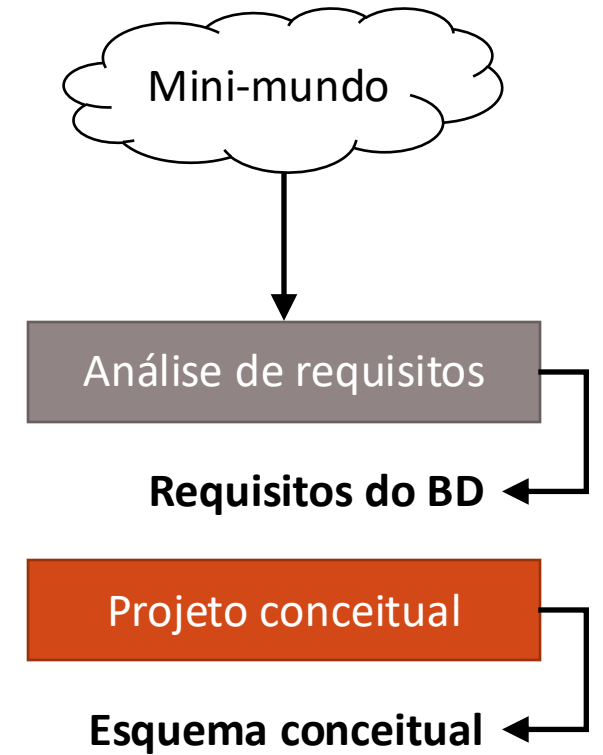
# Modelo de dados, esquema, instância





# Projeto de banco de dados (em IBD)

Independente 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

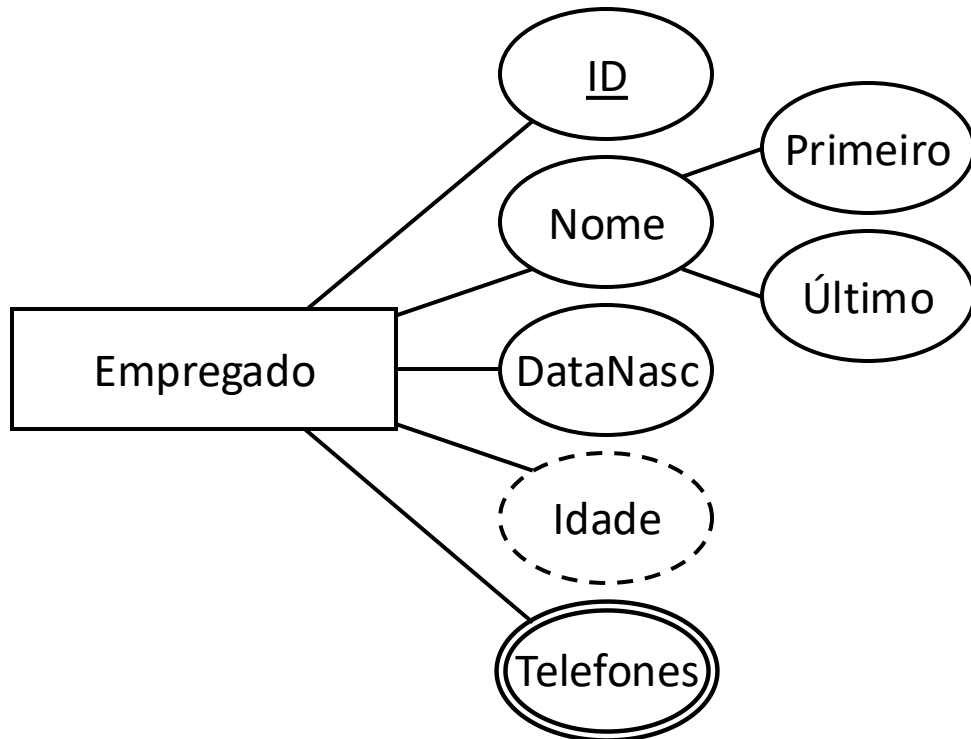
Massachusetts Institute of Technology

---

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-

# Tipo de entidade

Descreve o esquema para um conjunto de entidades que compartilham os mesmos atributos



•  $e_1$   
(001, (J, Silva), 1970, 50, {1234-5678})

•  $e_2$   
(002, (M, Pires), 1990, 30, {3333-4444, 9999-1111})

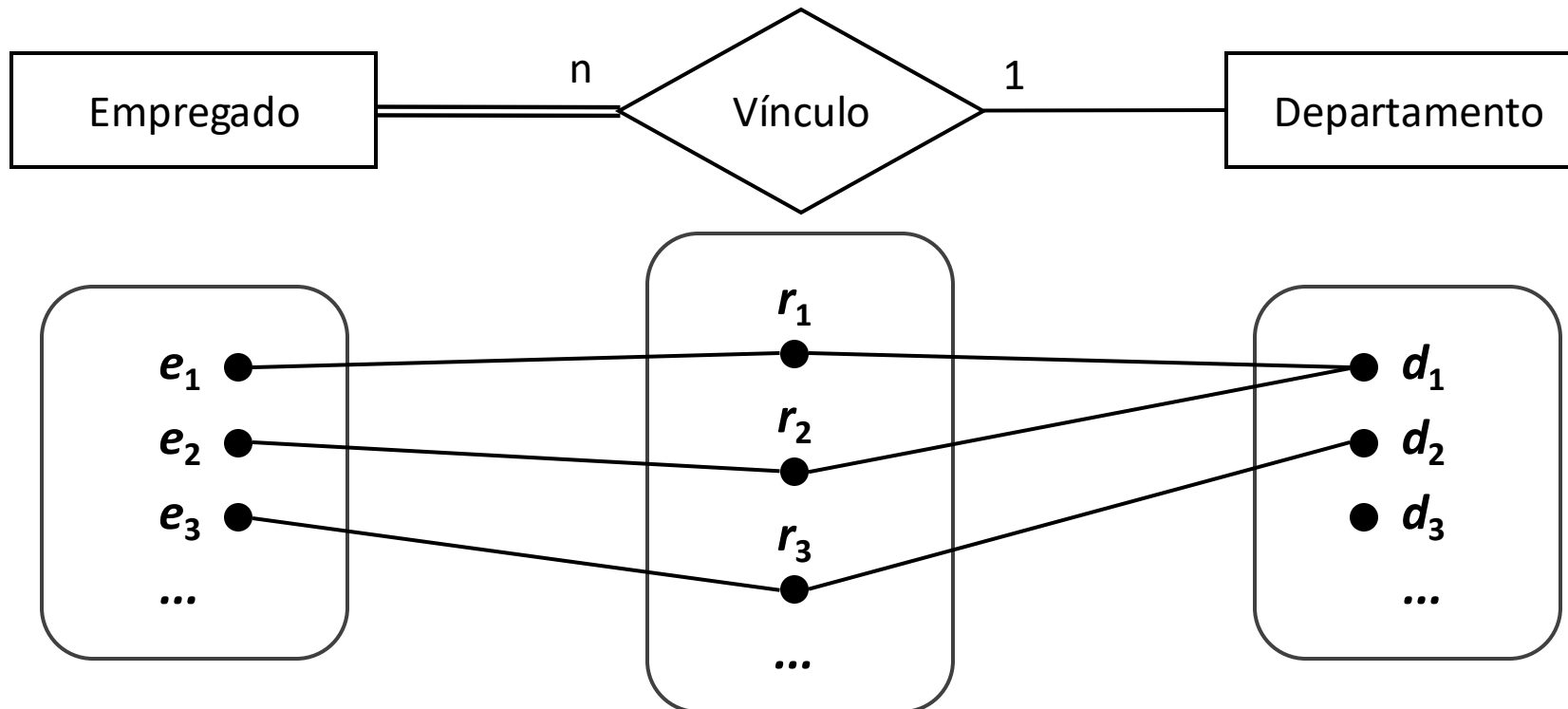
•  $e_3$   
(003, (L, Souza), 1995, 25, {5555-2233, 9888-7777})

•  $e_4$   
(004, (P, Maia), 2000, 20, {5678-1234})

...

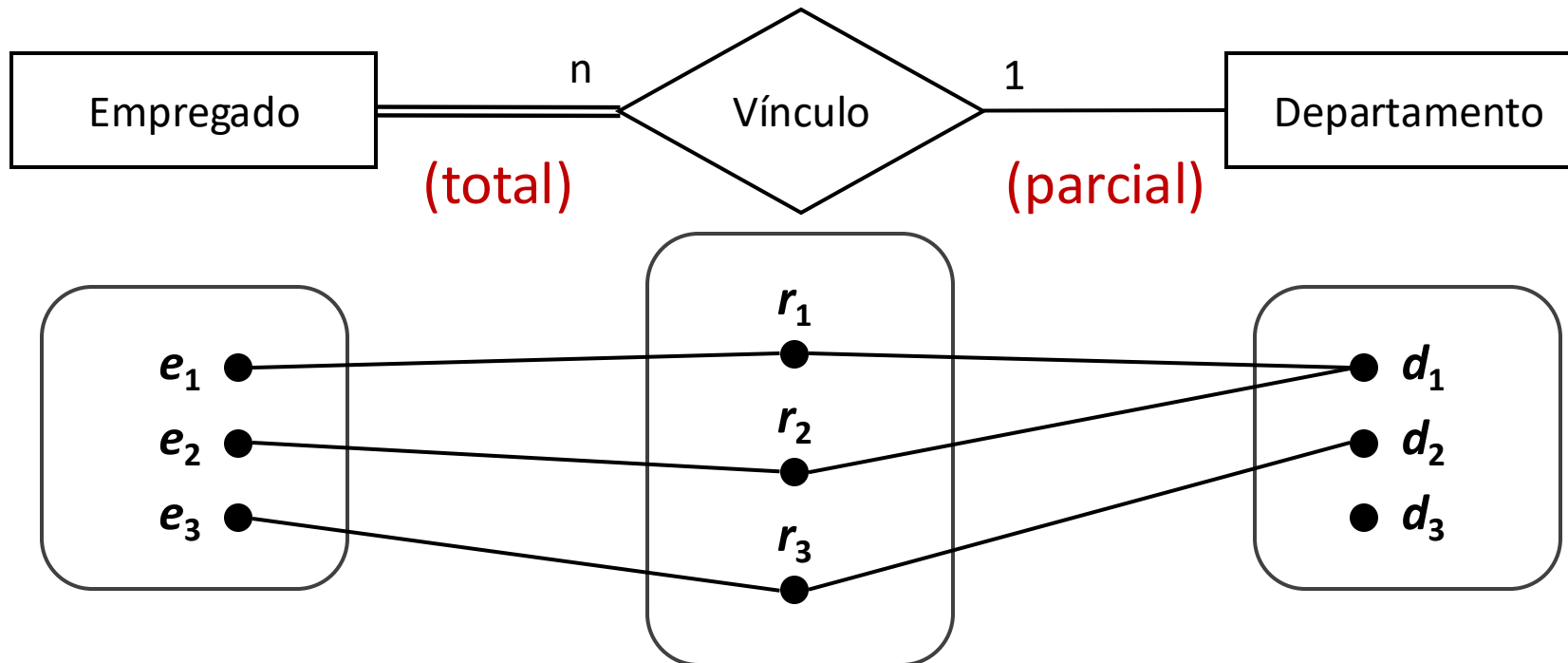
# Tipo de relacionamento

Define um conjunto de relacionamentos entre instâncias de um ou mais tipos de entidade



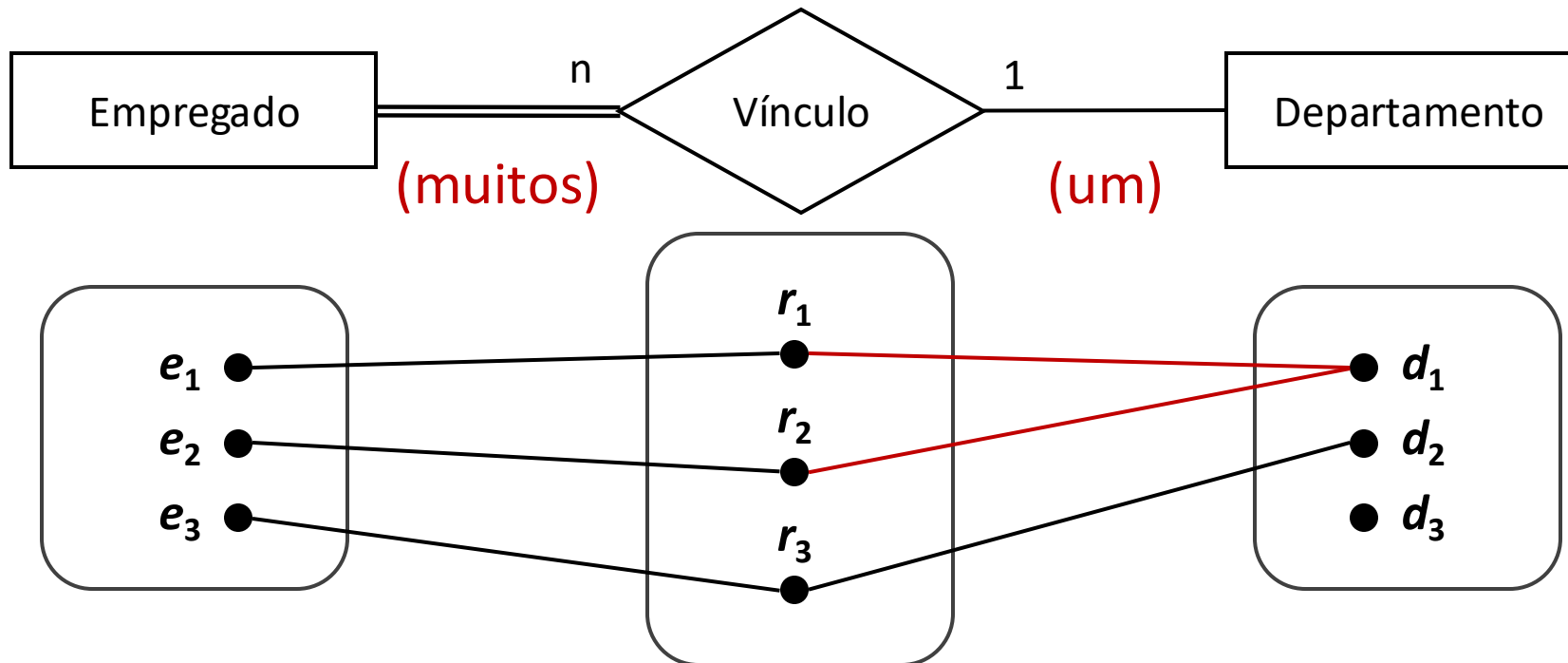
# Restrição de participação

Número **mínimo** de instâncias de um relacionamento a que uma instância de entidade **deve** se associar

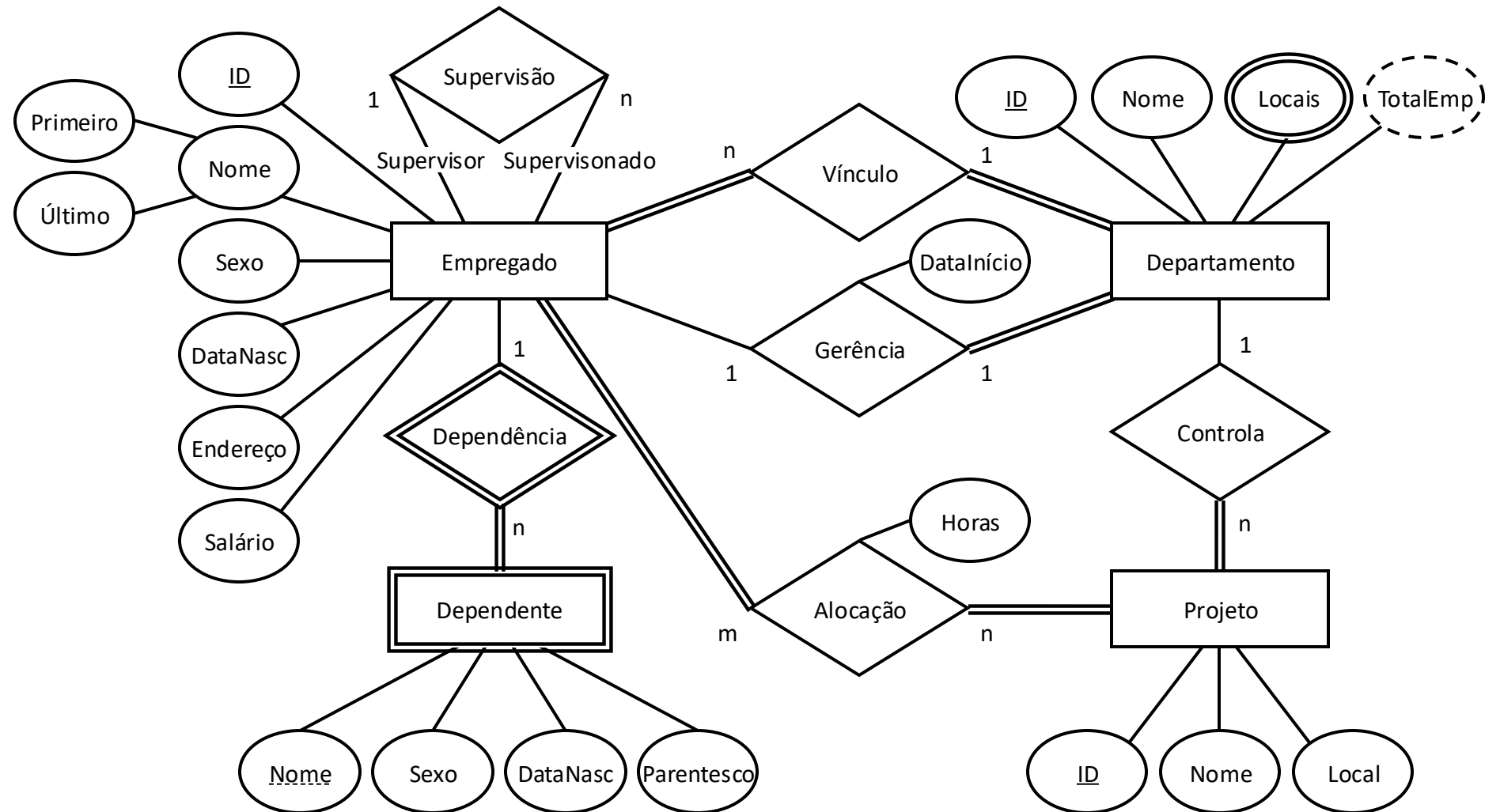


# Restrição de cardinalidade

Número **máximo** de instâncias de um relacionamento a que uma instância de entidade **pode** se associar

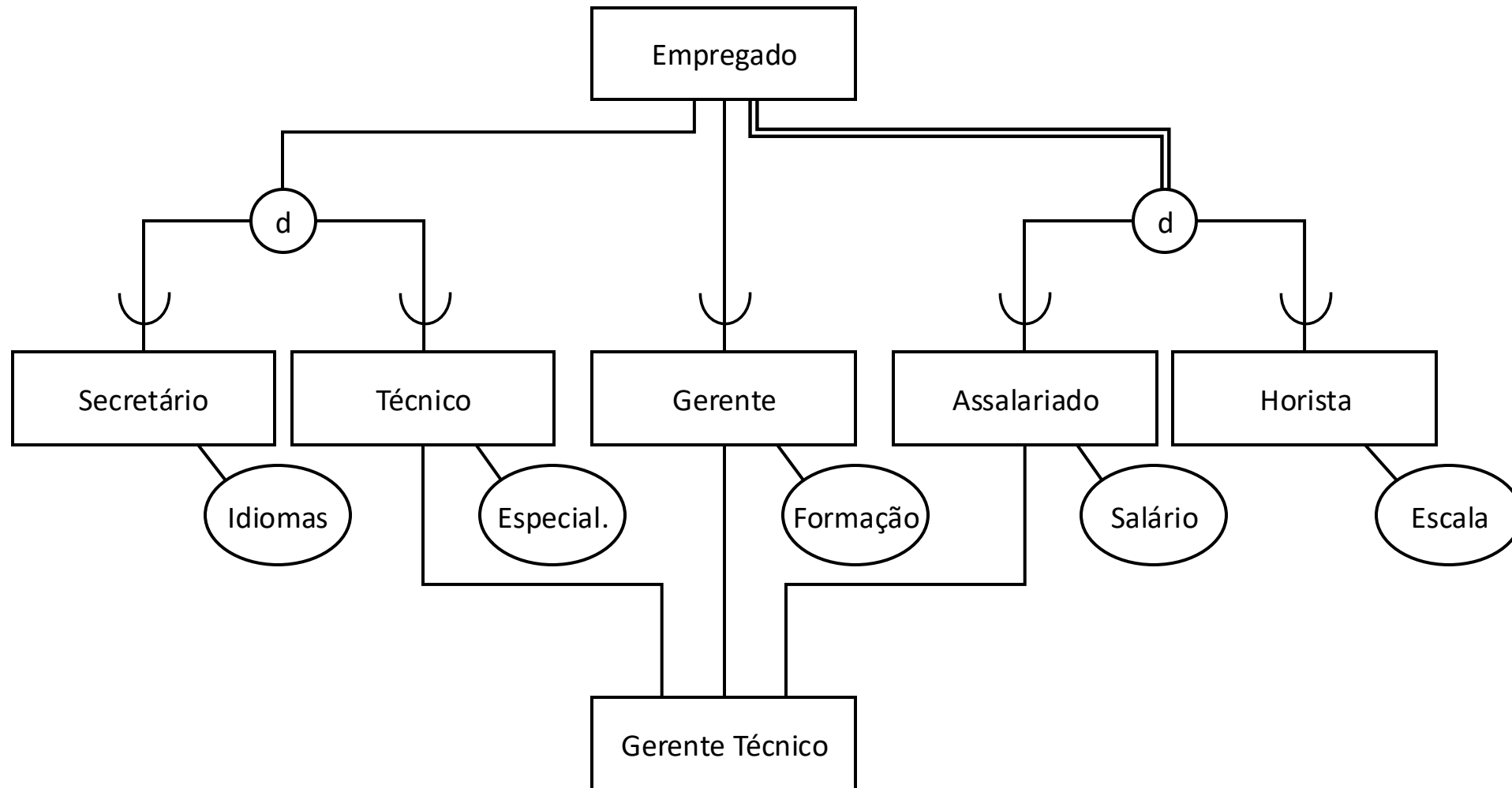


# Esquema ER





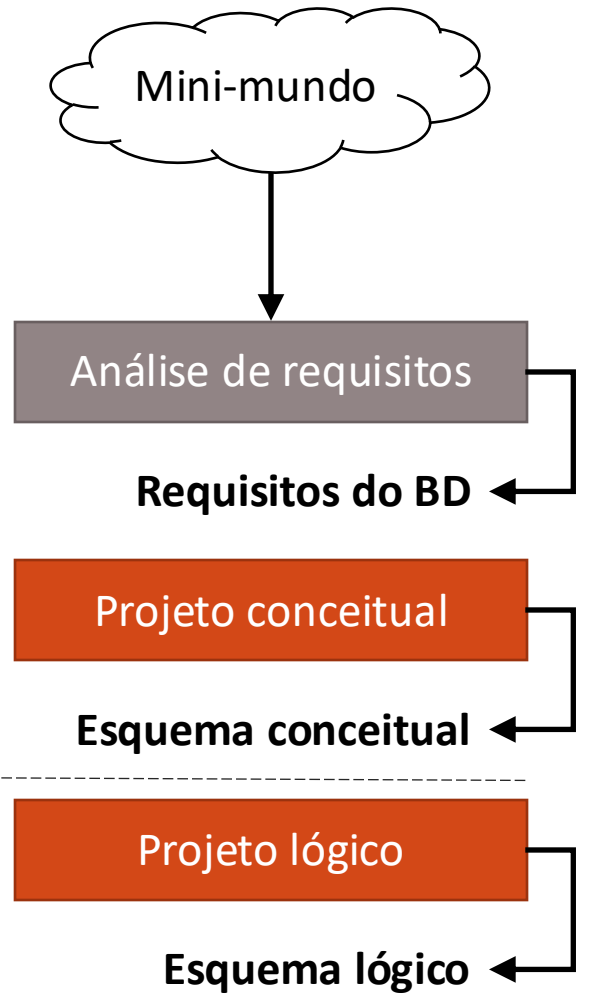
# Esquema EER



# Projeto de banco de dados (em IBD)

Independente de SGBD

Dependente de classe de SGBD



# 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

# Relação

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

## Esquema de relação

Empregado(ID, Nome, Salário, Dept)

## Relação

$$\begin{aligned} e(\text{Empregado}) \subseteq \\ & \text{dom}(\text{ID}) \times \\ & \text{dom}(\text{Nome}) \times \\ & \text{dom}(\text{Salário}) \times \\ & \text{dom}(\text{Dept}) \end{aligned}$$

## Tupla

$e_1 \in e(\text{Empregado})$   
e.g.  $e_1: (032, \text{J Silva}, 380, 21)$

# Restrições de integridade

Restrição de domínio

Restrição de chave

Restrição sobre nulos

Restrição de integridade de entidade

Restrição de integridade referencial

**Esquema  
relacional**  
esquemas de  
relação  
+  
restrições de  
integridade

# Operações sobre relações

## Operações de leitura

- **R**ead (consulta)

## Operações de escrita

- **C**reate (inserção)
- **U**ppdate (atualização)
- **D**elete (remoção)

Risco à integridade  
do banco de dados

# Inserção

**Domínio:** valor fora do domínio

**Nulos:** valor inserido é nulo

**Chave:** valor já existe

**Integridade de entidade:** se chave contém nulos

**Integridade referencial:** se chave estrangeira não-nula referencia tupla inexistente



# Atualização

## Atributo não-chave

- **Domínio:** novo valor fora do domínio
- **Nulos:** novo valor inserido é nulo

## Atributo chave estrangeira

- **Integridade referencial:** novo valor (não-nulo) da chave estrangeira referencia tupla inexistente

# Atualização

Atributo chave primária

- **Chave:** novo valor já existe
- **Integridade de entidade:** novo valor contém nulos
- **Integridade referencial:** valor antigo é referenciado por alguma chave estrangeira no banco de dados

# Remoção

**Integridade referencial:** valor removido é referenciado por alguma chave estrangeira no banco de dados

# Ações corretivas

