

# Banco de Dados

## 03 – Modelagem: Projeto Lógico

Prof<sup>a</sup> Cristina Verçosa Pérez Barrios de Souza

[cristina.souza@pucpr.br](mailto:cristina.souza@pucpr.br)





# Tópicos

- › Projeto Lógico
  - Modelo Relacional



# Conceitos Iniciais



# Modelo Relacional

- › Criado nos anos 70 pelo matemático **Edgar F. Codd**, começou a ser usado comercialmente nos anos 80.
- › Considerado como o **primeiro modelo de dados** para **aplicações comerciais**.
- › Existe uma grande **base teórica e matemática** para este modelo, que apoia eficientemente o **projeto de banco de dados relacionais**
- › Permite um **processamento eficiente** das necessidades de informação de seus usuários.



# Projeto Lógico de Banco de Dados

## › Solução computacional – Tratamento de Dados

### – 1ª Fase)

- › A concepção da **estrutura de persistência** inicia com o **Projeto Conceitual**, modelado com um **Diagrama Entidade-Relacionamento (DER)**
- › O **DER** é independente de **SGDB** e, portanto, deve ser o primeiro modelo gerado após a entrevista para **levantamento de requisitos**.

### – 2ª Fase) ←

- › O **Projeto Lógico** é dependente de SGDB.
- › O modelo de SGDB a ser utilizado é o **modelo relacional**. ←

# PROJETO LÓGICO DE BANCO DE DADOS:

## 1ª Fase - Projeto Conceitual

Gera o **MODELO CONCEITUAL**

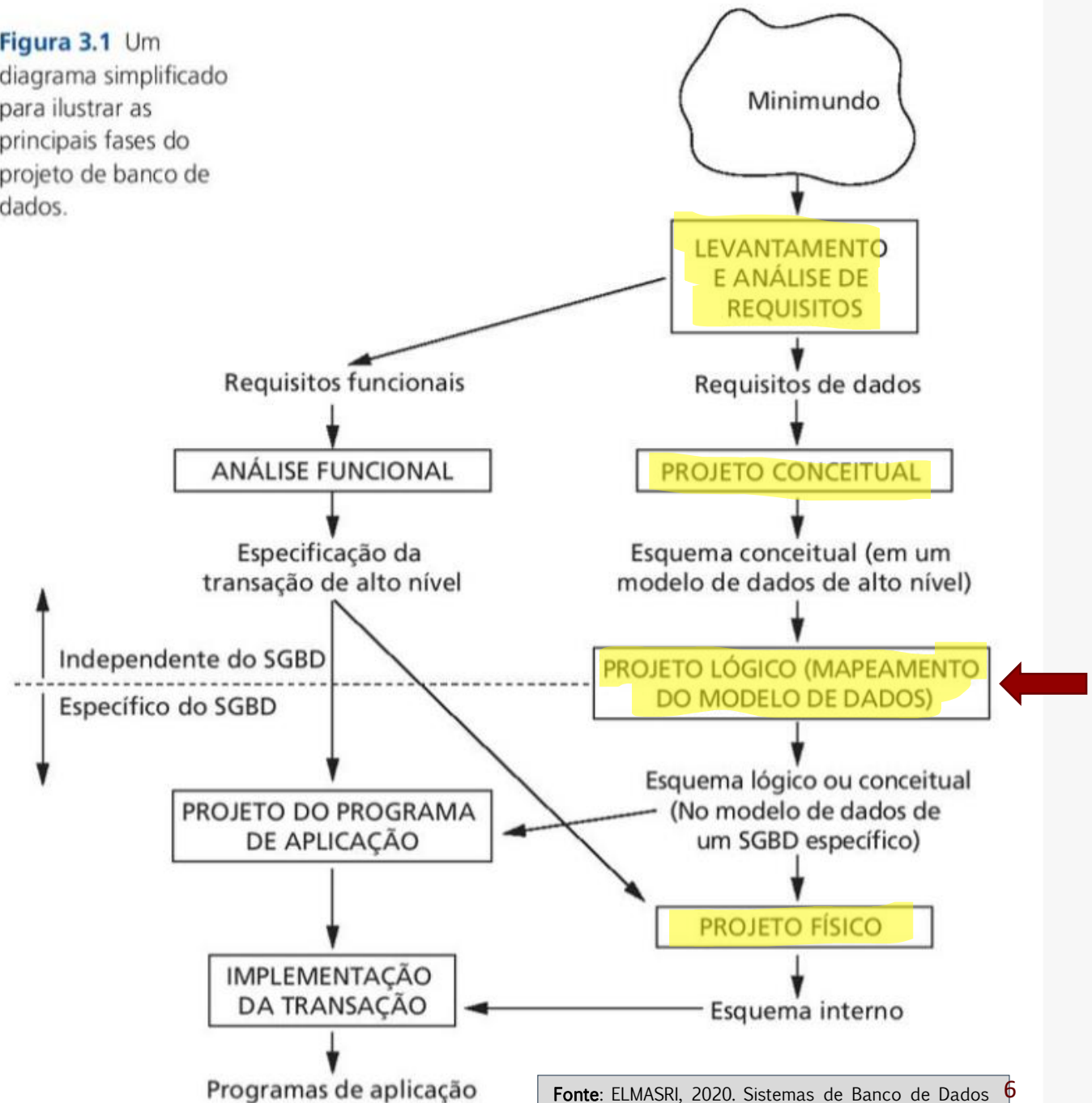
## 2ª Fase - Projeto Lógico

Gera o **MODELO LÓGICO**

## 3ª Fase - Implementação

Gera o **MODELO FÍSICO**

**Figura 3.1** Um diagrama simplificado para ilustrar as principais fases do projeto de banco de dados.





# **MODELO RELACIONAL:**

## **Visão Geral**



# Modelo Relacional

## › Visão Informal

### – Aspecto Estrutural

- › Os **dados** em um banco de dados relacional são percebidos pelo usuário como **tabelas** (\* Teoria dos Conjuntos)

### – Aspecto de Integridade

- › As tabelas satisfazem certas **restrições**

### – Aspecto de Manipulação

- › Os **operadores** que o usuário utiliza **para manipular as tabelas têm como resultado outras tabelas**





# Terminologia

Termo Relacional Formal	Termo Informal Equivalente
<b>Relação</b>	<b>Tabela</b>
<b>Tupla</b>	<b>Linha ou registro</b> (par ordenado, ou trio ordenado, ou etc...)
<b>Atributo</b>	<b>Coluna ou campo</b> (cada coluna faz referência a um <b>Domínio</b> )
<b>Cardinalidade da Relação</b>	<b>Número de linhas ou registros</b> (quantidade de tuplas da relação)
<b>Domínio</b>	<b>Coleção de valores legais</b> (conjuntos de onde provêm os objetos relacionados)
<b>Chave Primária</b>	<b>Identificador único</b>



# Principais Conceitos

## › Relações

- Termo matemático para **tabela**
- Conjunto de **tuplas** (registros) com os mesmos **atributos**

## › Atributos

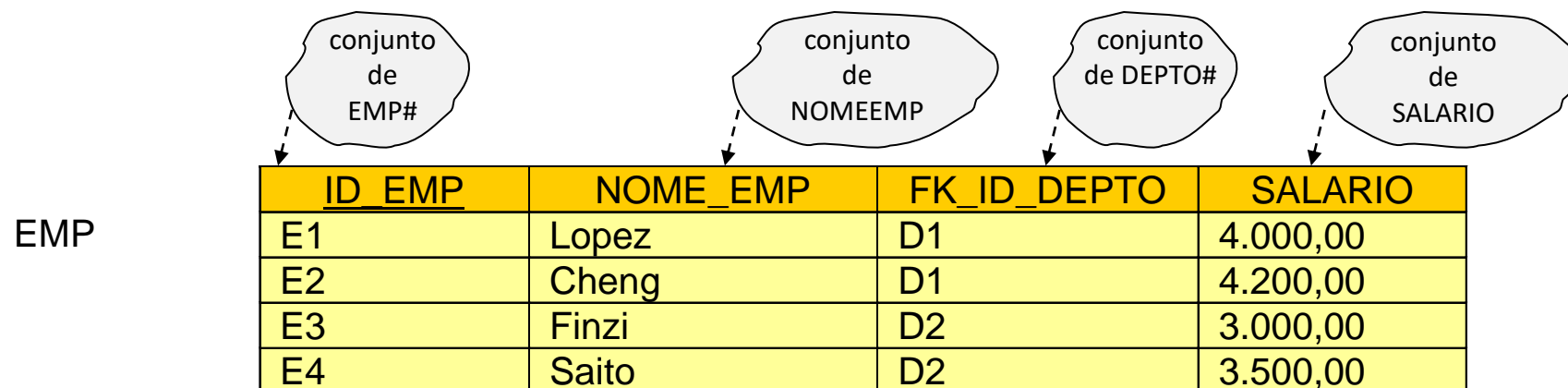
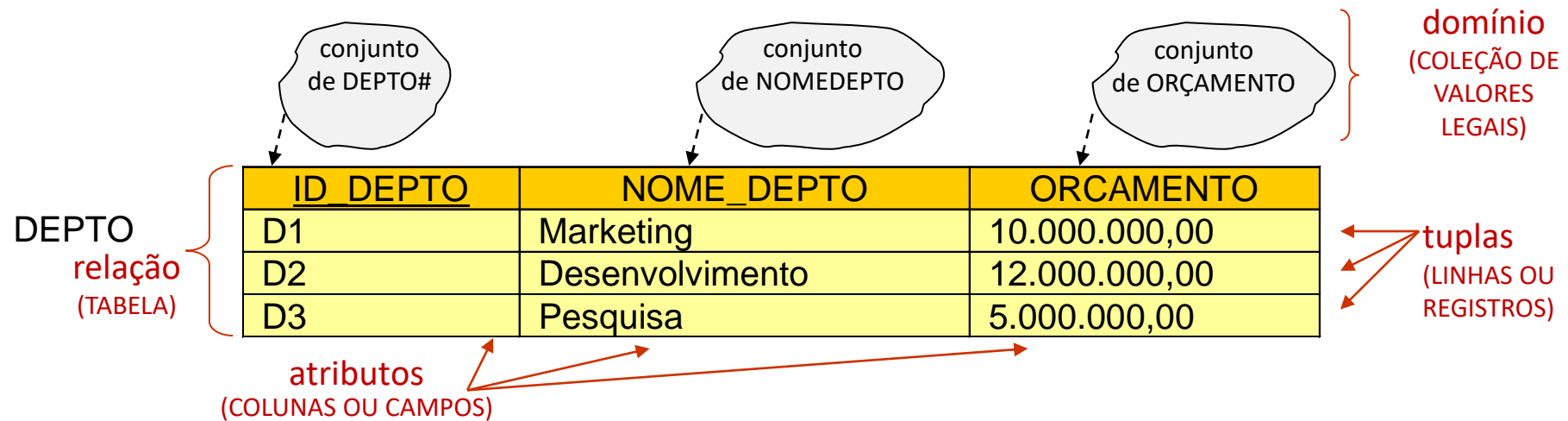
- Propriedades descritivas de cada membro de um conjunto de relações
- Para cada atributo, ou coluna, de uma relação, há um conjunto de valores permitidos, chamado de **domínio** do atributo

## › Domínios

- Conjunto de **valores possíveis** para um dado **atributo**
- Devem representar a menor unidade semântica de dados - **valor atômico**
- Um **atributo** deve ser definido sobre exatamente um domínio, mas um domínio pode ser associado a mais de um **atributo**.



# Exemplos





# Esquema x Instância

## › Esquema de Relação

- Compreende uma lista de **atributos** (colunas) seus **domínios** correspondentes
- Similar à **definição de tipos** em uma linguagem de programação

## › Instância de Relação

- **Variável de relação**
- Valores presentes na relação / tabela, em um certo momento
- “*Snapshot*” da tabela
- Similar ao **valor** de uma variável em uma linguagem de programação, **que muda ao longo do tempo**



# Esquema x Instância

## › Esquema

- $A_1, A_2, \dots, A_n$  são atributos
- $R = (A_1, A_2, \dots, A_n)$  é o esquema da relação
- Exemplo:
  - › *Professor* = (ID, Nome, CPF)
  - › *Disciplina* = (ID\_disciplina, nome, ementa, credits)

## › Instância

- Valores atuais de uma relação são especificados por uma tabela

ID	Nome	CPF
123	Antônio Carlos	123456789-00
234	Renata Flores	234567890-01
345	Bianca Castro	345678901-23

*Snapshot* (instantâneo) da base



# Chaves

## > Super-chave

- Conjunto de um ou mais **atributos** que permite identificar de maneira única uma **tupla** em uma relação

CLIENTE

CPF	NOME	SOBRENOME
12345678911	José	Silva
23456789122	Maria	Pereira
34567891233	José	Souza

Diagram illustrating the concept of a Super-chave (Superkey) using the CLIENTE table:

- The combination of **CPF** and **SOBRENOME** is labeled as a **superchave** (superkey).
- The **CPF** attribute alone is labeled as a **superchave**.
- The combination of **NOME** and **SOBRENOME** is labeled as **não pode ser superchave** (cannot be a superkey).



# Chaves

## › Chave Candidata

- **Superchaves** para as quais nenhum subconjunto de atributos possa ser superchave.

CLIENTE

CPF	NOME	SOBRENOME
12345678911	José	Silva
23456789122	Maria	Pereira
34567891233	José	Souza

Diagram illustrating candidate keys and superkeys for the CLIENTE table:

- CPF** is a candidate key and a superkey (chave candidata e superchave).
- NOME** is a candidate key and a superkey (chave candidata e superchave).
- SOBRENOME** is a candidate key and a superkey (chave candidata e superchave).
- CPF + SOBRENOME** is not a candidate key (não pode ser chave candidata).



# Chaves

## › Chave primária

- *Primary Key (PK)*
- Chave candidata escolhida pelo projetista do banco de dados para a identificação de tuplas, ou registros, em uma tabela

CLIENTE

<u>PK_CPF</u>	NOME	SOBRENOME
12345678911	José	Silva
23456789122	Maria	Pereira
34567891233	José	Souza

chave primária  
e  
chave candidata  
e  
superchave

Chave candidata escolhida pelo projetista





# Chaves

## › Chave estrangeira

- *Foreign Key* (FK)
- Uma tabela pode incluir entre seus atributos a **chave primária** (ou chave candidata) **de outra tabela** (ou relação).

EMP

<u>ID_EMP</u>	NOME_EMP	FK_ID_DEPTO	SALARIO
E1	Lopez	D1	4.000,00
E2	Cheng	D1	4.200,00
E3	Finzi	D2	3.000,00
E4	Saito	D2	3.500,00

Chave primária na relação DEPTO

chave primária



# Restrições de Integridade

## › Integridade em Banco de Dados

- Precisão de dados no banco
- Dados íntegros = dados não adulterados, pois atendem a regras pré-estabelecidas
- Exemplo:
  - › Um campo status pode aceitar apenas valores de 1 a 100
  - › CPF é um campo de valor único, não se repete nas linhas da tabela

## › Restrições

- Constraints
- Regra com a qual os dados devem estar em conformidade
- Meios pelos quais o banco pode ser protegido contra acessos indevidos
- Forma de implementar regras de negócio no banco de dados



# Exemplos de Restrições

## › NOT NULL

- Especifica que a coluna **não** pode conter valores nulos (**valor requerido**)

## › PRIMARY KEY

- Especifica a coluna que identifica unicamente uma linha da tabela

## › UNIQUE

- Especifica que os valores na coluna devem ser únicos
- Não são permitidos valores nulos

## › CHECK

- Especifica regras para os valores da coluna



# Restrições de Integridade

## › Consistência em Banco de Dados

- Garante que todas as **restrições de integridade** de um banco de dados são **satisfeitas**
- As **restrições asseguram** que as **mudanças feitas** no banco de dados, **por usuários autorizados**, mantenham a **consistência** de dados

## › Correção em Banco de Dados

- Quando os **dados refletem** o verdadeiro estado das coisas no **mundo real**

Banco Consistente  $\neq$  Banco Correto



# PROJETO LÓGICO: Modelo Relacional



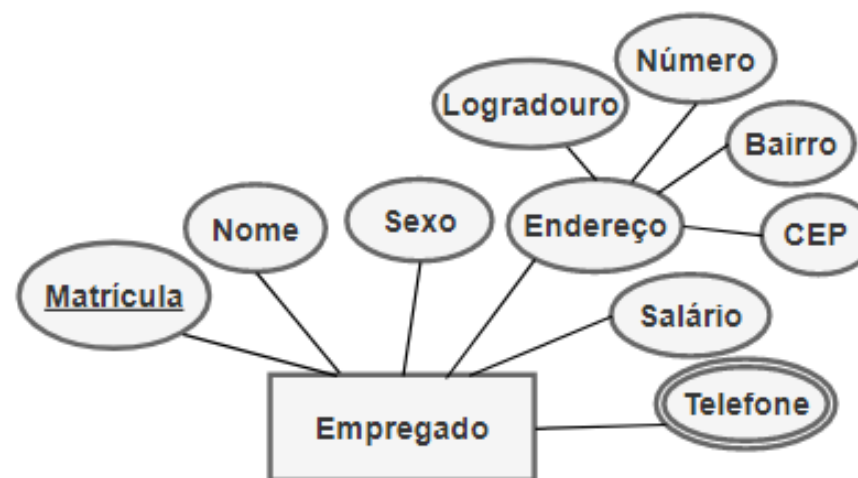
# Mapeamento MER → MR

## › **Modelo Entidade Relacionamento (MER) e Modelo Relacional (MR)**

- O **MER** é utilizado para fazer o **projeto conceitual** do banco de dados
- O **MR** é utilizado para fazer o **projeto lógico**.

## › Exemplo: mapeamento de **Entidades**

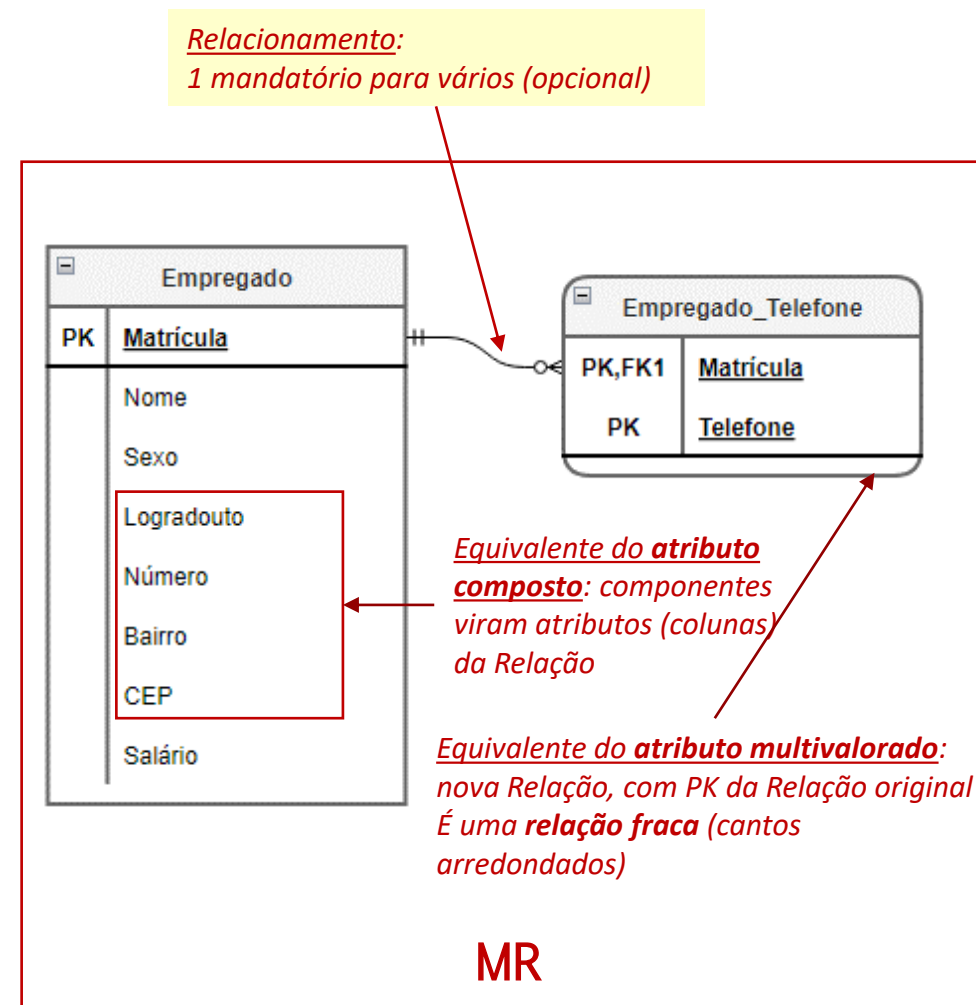
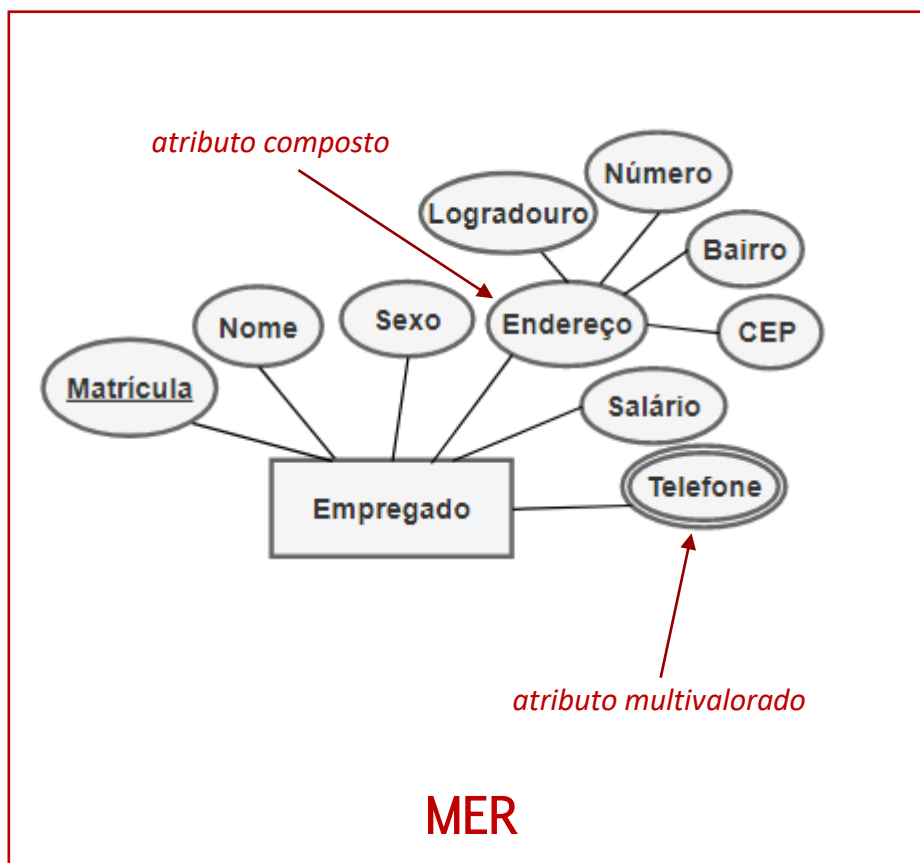
- Considere o MER da figura.





# Mapeamento MER → MR: entidades

## › Transformações

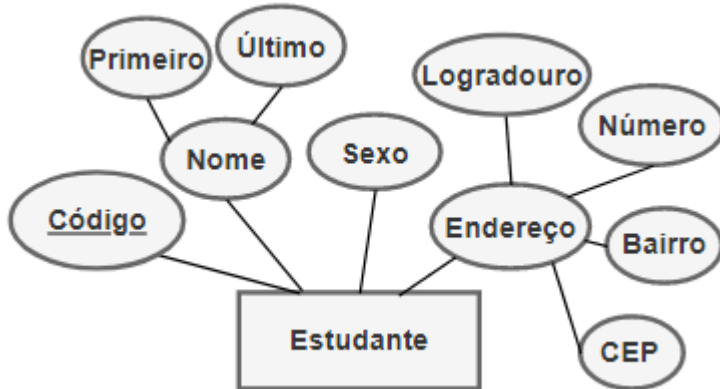




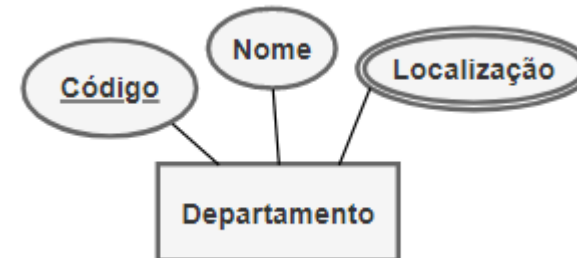
# Mapeamento MER → MR

## › Prática 1:

- faça o mapeamento **MER → MR** das **entidades** e seus **atributos** (simples e compostos), nos exercícios **(a)** e **(b)** a seguir.



a)



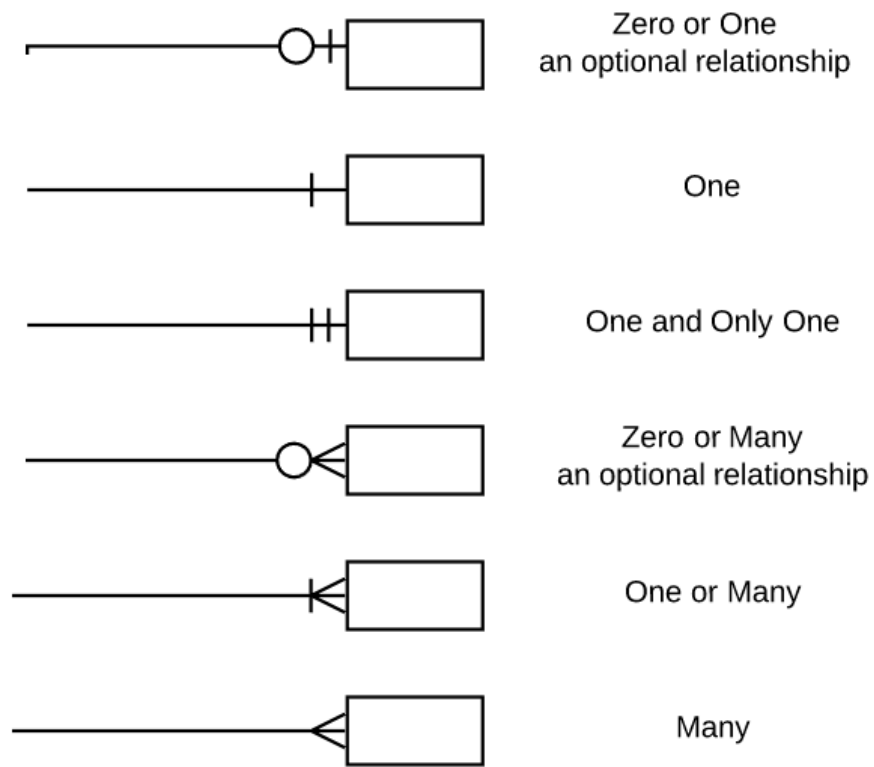
b)





# Mapeamento MER → MR: relacionamentos

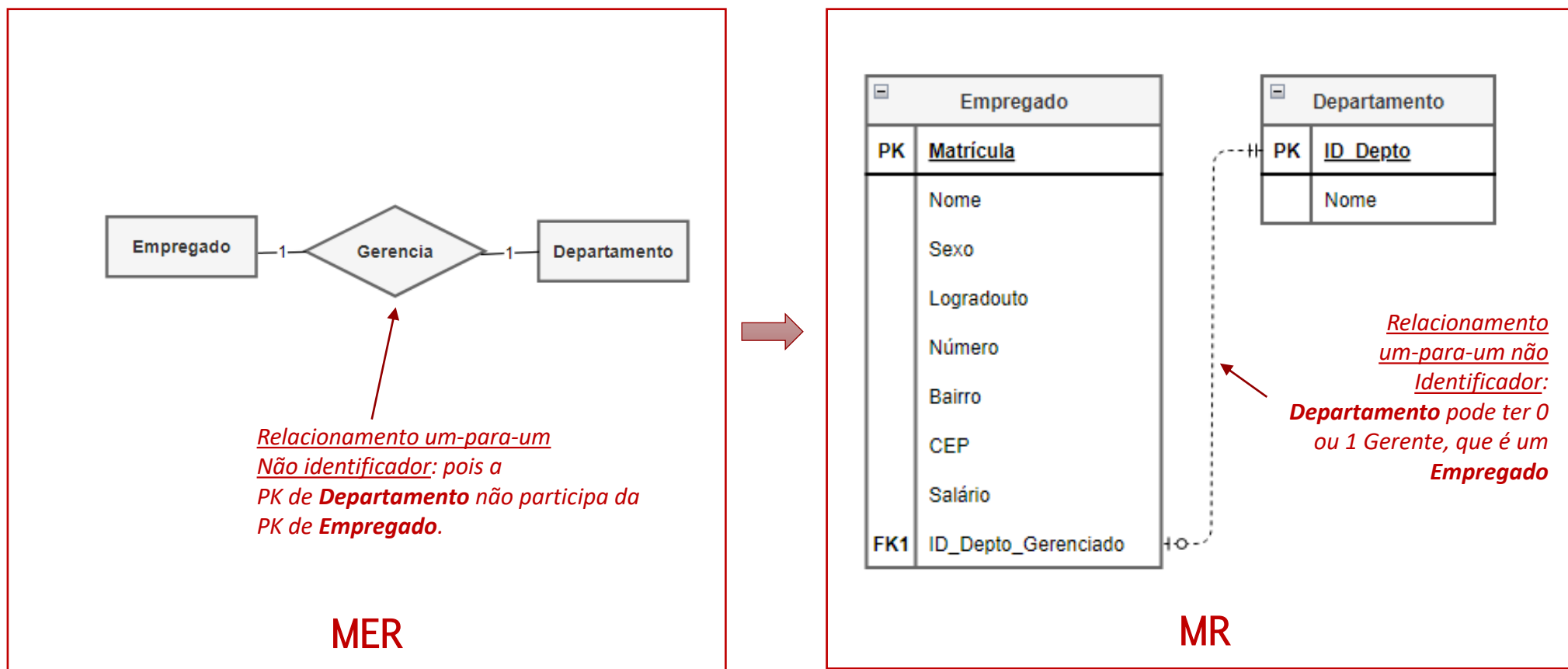
› Alguns exemplos de relacionamentos





# Mapeamento MER → MR: relacionamentos

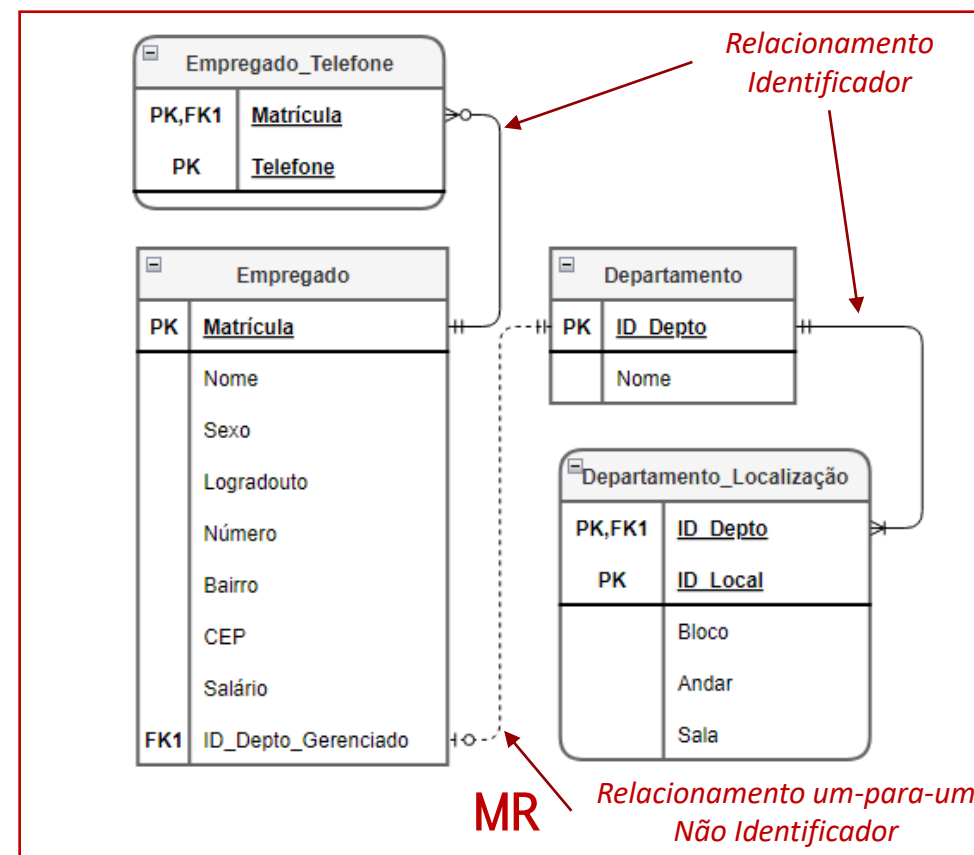
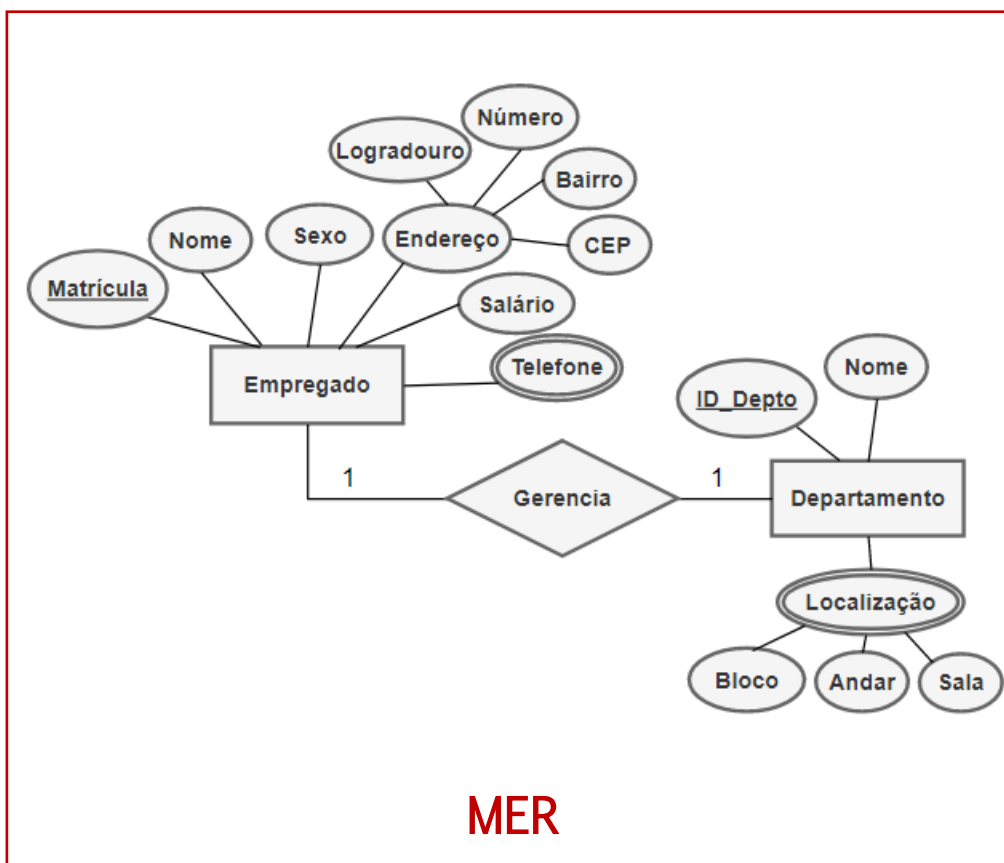
## › Relacionamento um-para-um





# Mapeamento MER → MR: relacionamentos

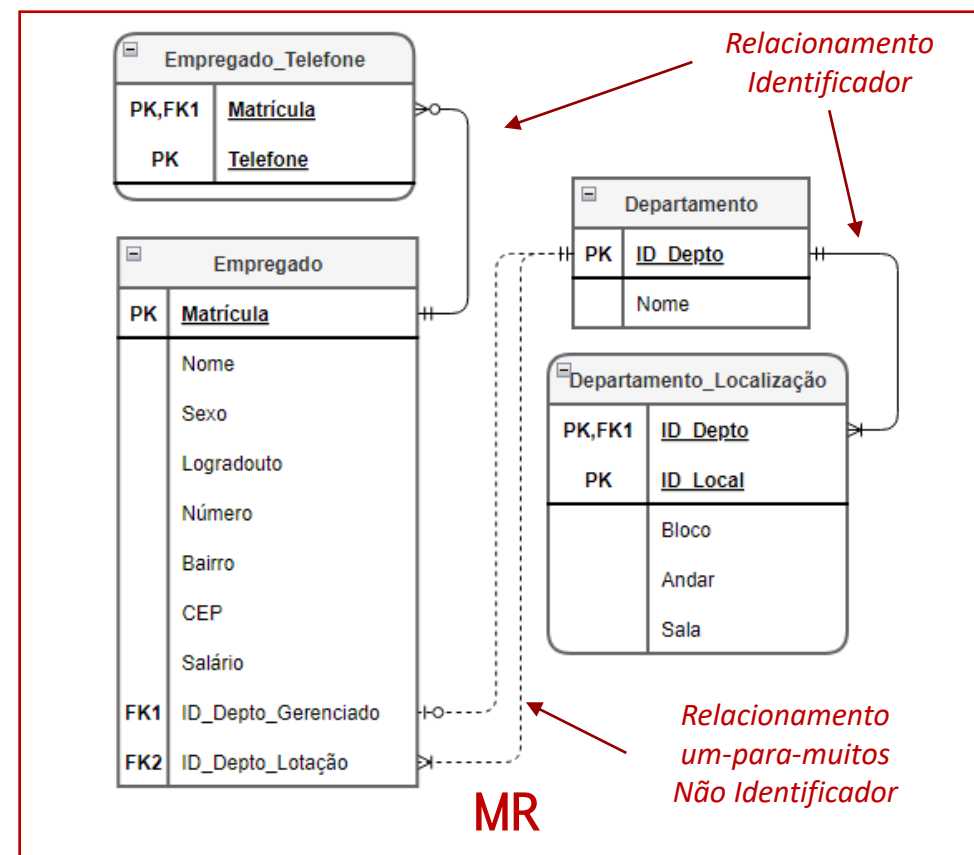
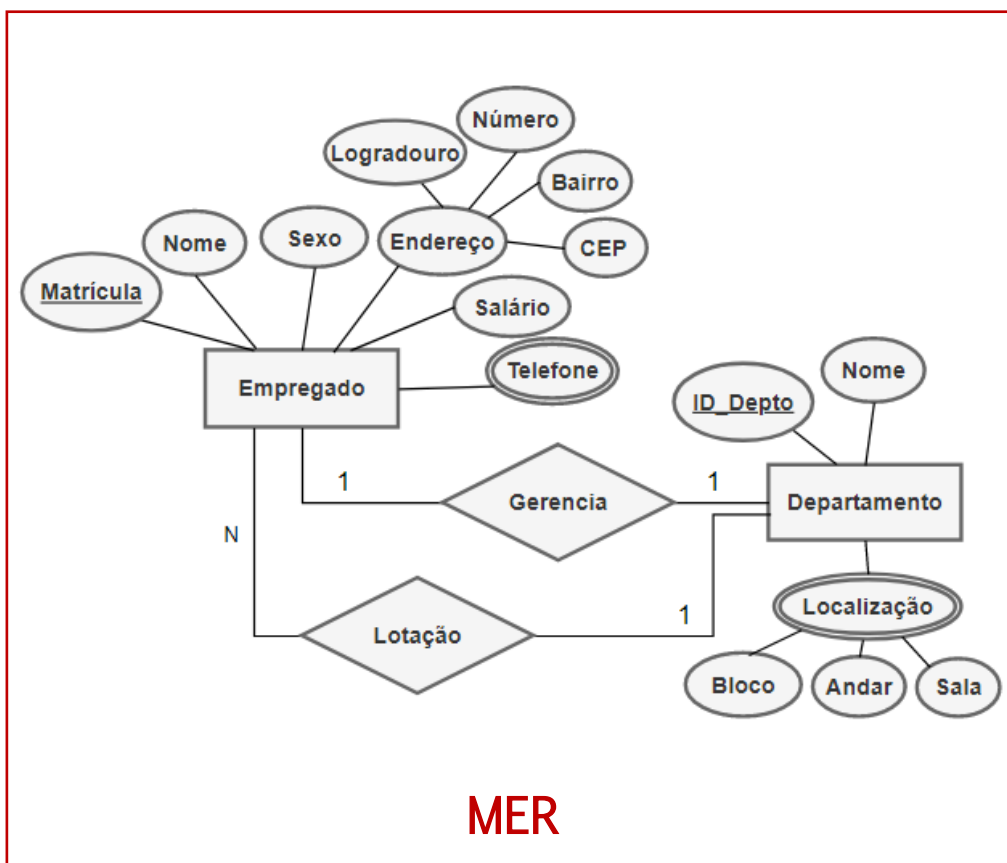
## › Relacionamento um-para-um





# Mapeamento MER → MR: relacionamentos

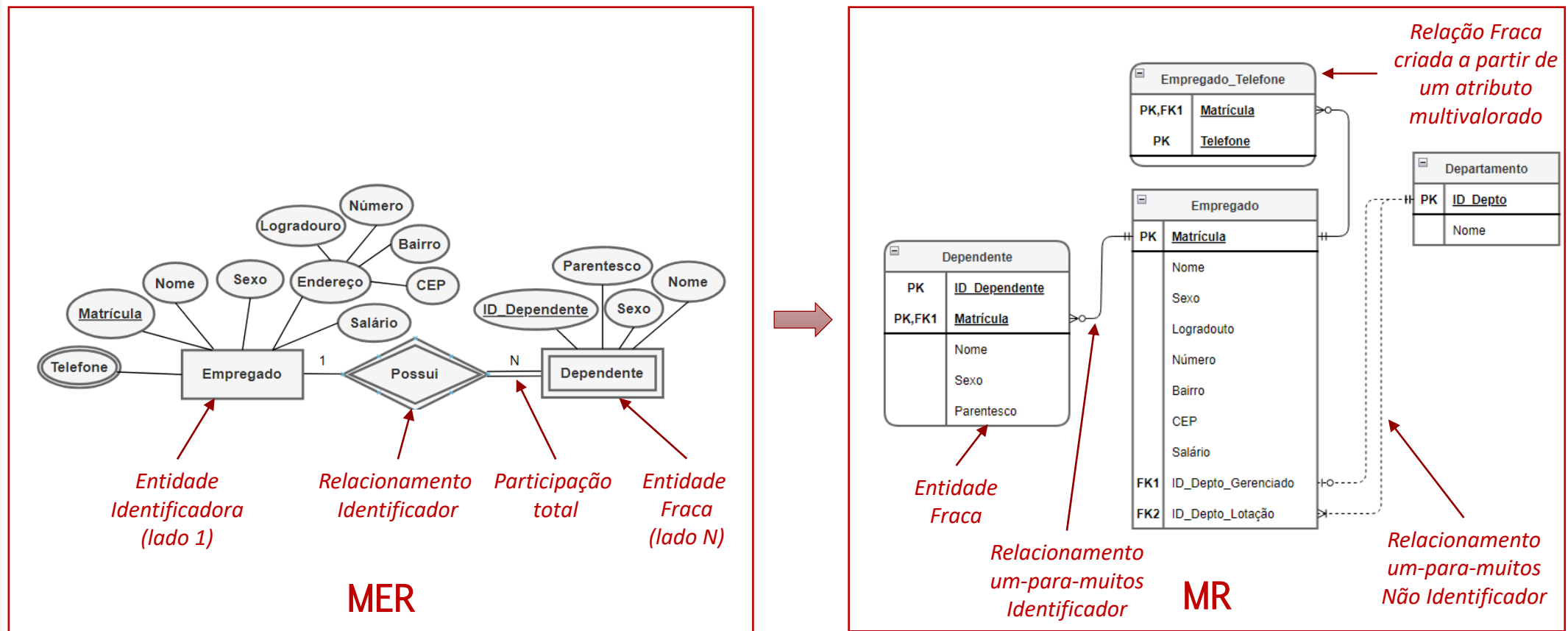
## › Relacionamento um-para-muitos (entidades fortes)





# Mapeamento MER → MR: relacionamentos

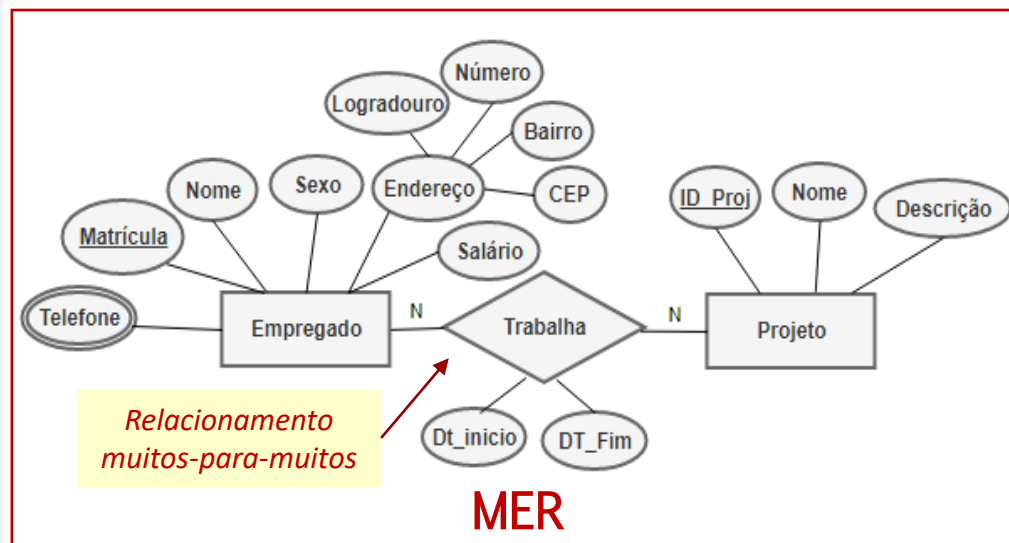
## › Relacionamento um-para-muitos (entidade fraca)





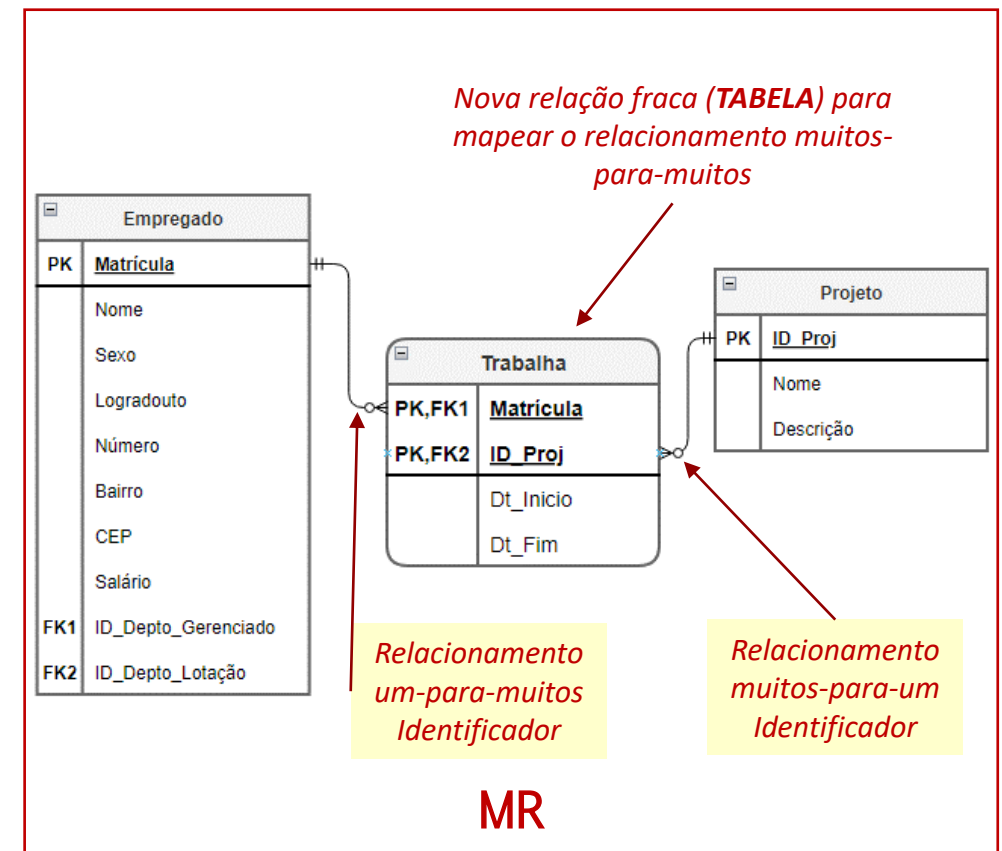
# Mapeamento MER → MR: relacionamentos

## › Relacionamento muitos-para-muitos (entidades fortes)



Nova Tabela Gerada (entidade fraca):

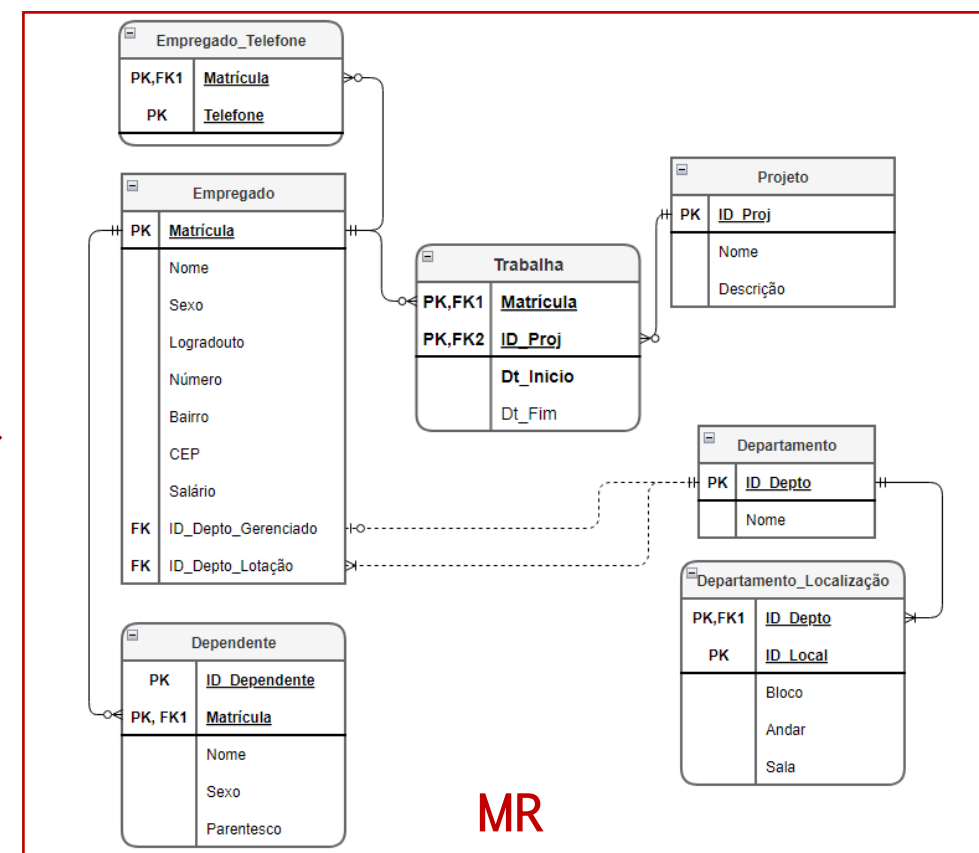
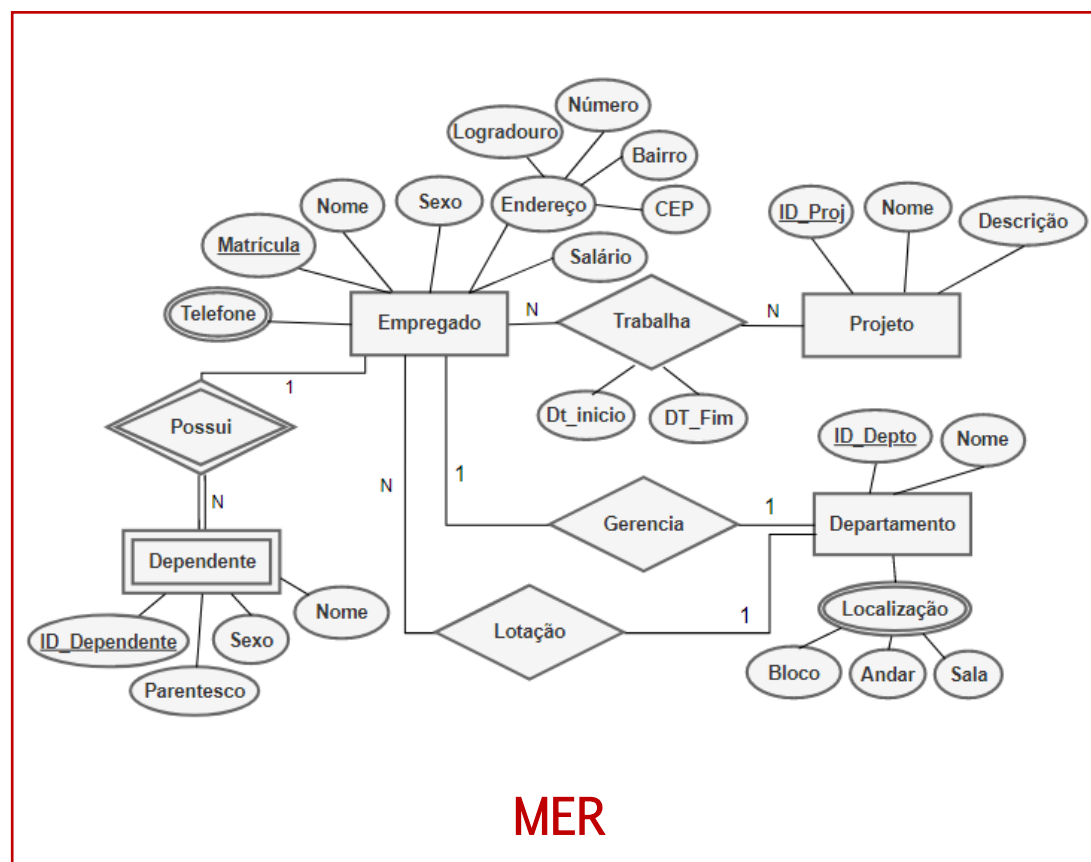
Trabalha			
<u>Matricula</u>	<u>ID_Proj</u>	Dt_Inicio	Dt_Fim





# Mapeamento MER → MR

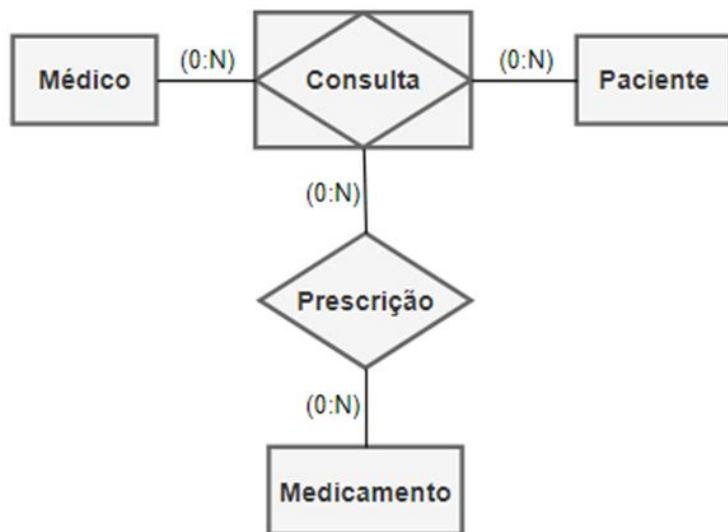
## › Exemplo completo



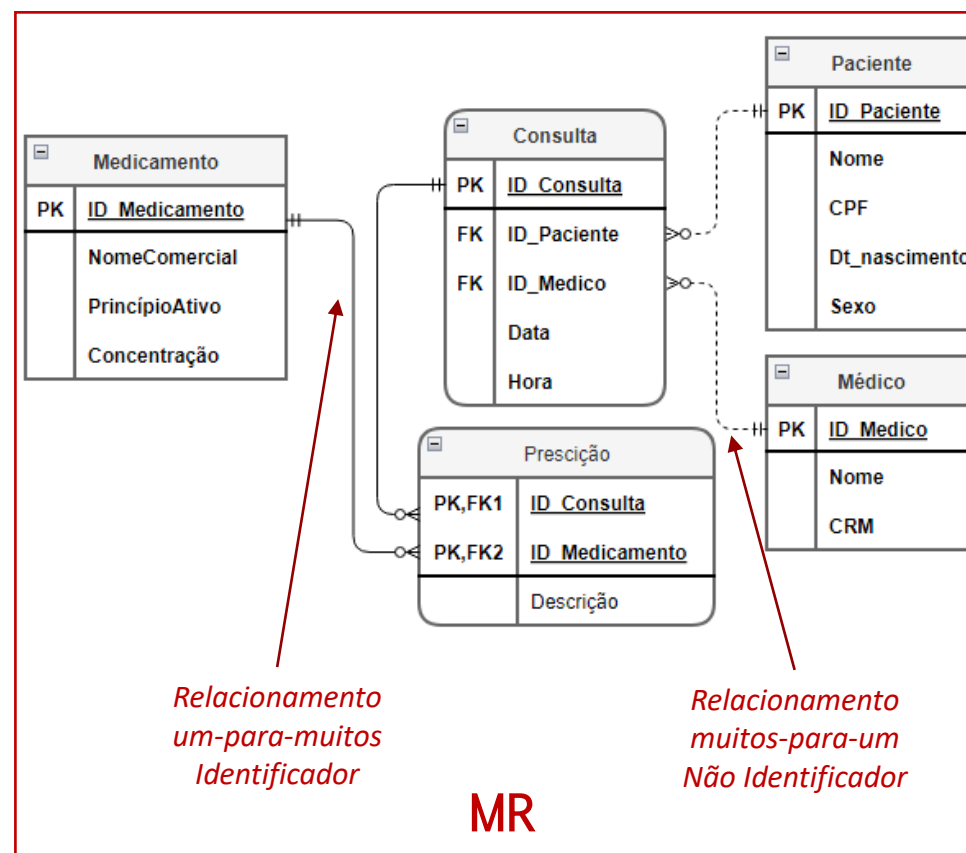


# Mapeamento MER → MR

## › Entidade Associativa



MER

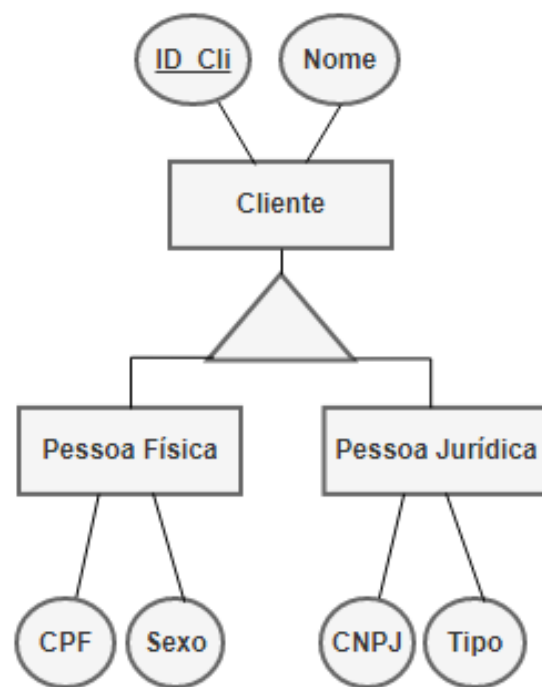






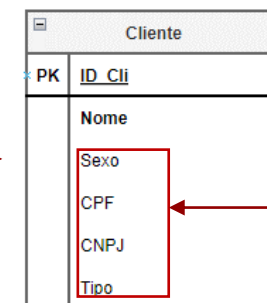
# Mapeamento MER → MR

## › Especialização / Generalização



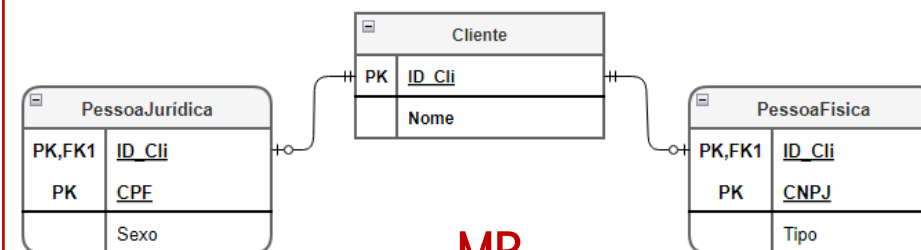
MER

*1ª. Opção: criar relação de supertipo (pai) e copiar os atributos dos subtipos (filhos). Isto ocasiona valores **NULOS**.*



Podem ser **NULOS**.

*2ª. Opção: criar relações para subtipos (filhos), cada uma com seus atributos específicos.*

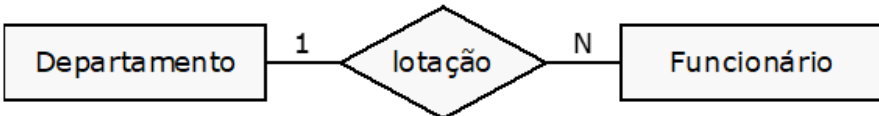
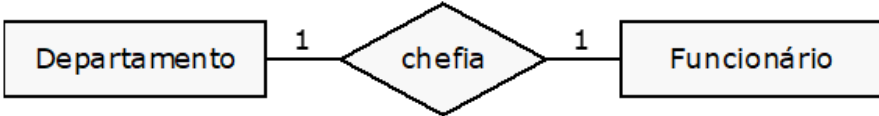
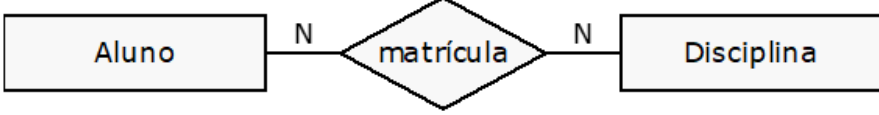
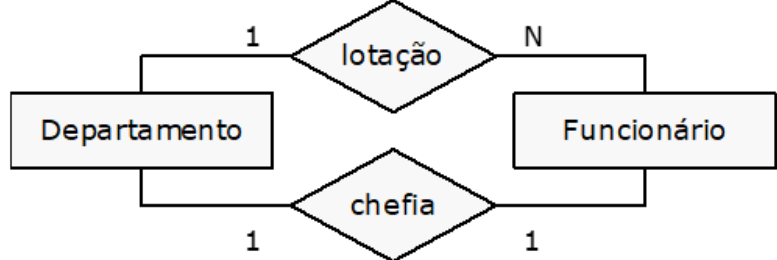


MR



# Mapeamento MER → MR: relacionamentos

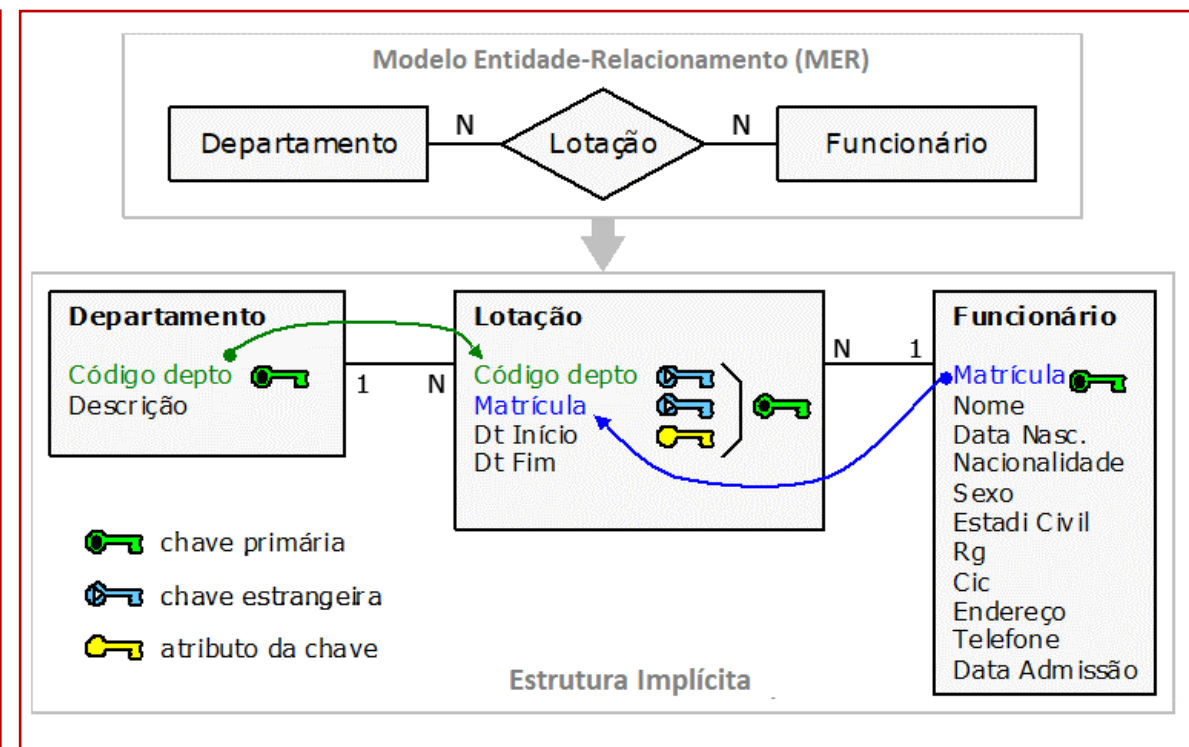
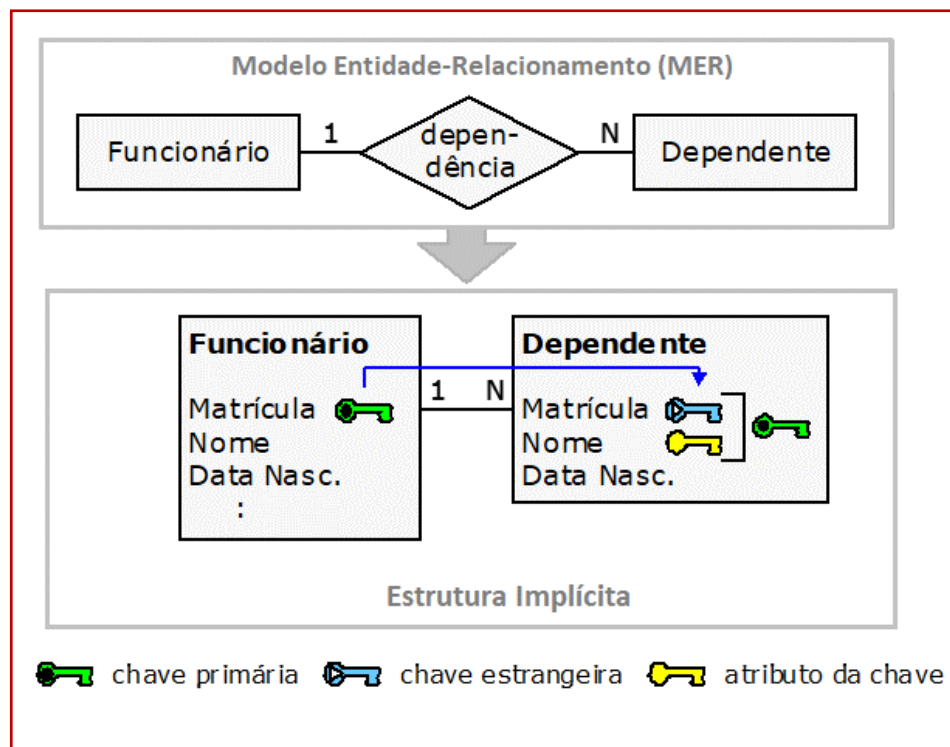
## › Resumo rápido: relacionamentos

Modelo		Descrição Textual
1		Um departamento lota <b>N</b> funcionários; cada funcionário é lotado em <b>1</b> departamento.
2		Um departamento é chefiado por <b>1</b> funcionário; um funcionário é chefe de <b>1</b> departamento.
3		Um aluno pode se matricular em <b>N</b> disciplinas; cada disciplina pode ter <b>N</b> alunos matriculados.
4		Um departamento lota <b>N</b> funcionários e possui <b>1</b> chefe (que também é um funcionário); cada funcionário é lotado em <b>1</b> departamento e um funcionário é chefe de <b>1</b> departamento.  Obs.: Esse modelo é uma junção dos modelos 1 e 2.



# Mapeamento MER → MR: relacionamentos

## › Resumo rápido: relacionamentos



### Chave Estrangeira (FK):

- Sempre é uma **Chave Primária (PK)** em outra **Tabela**
- Sempre fica no lado **N**



# Mapeamento MER → MR: relacionamentos

› Resumo rápido: mapeamento MER → MR

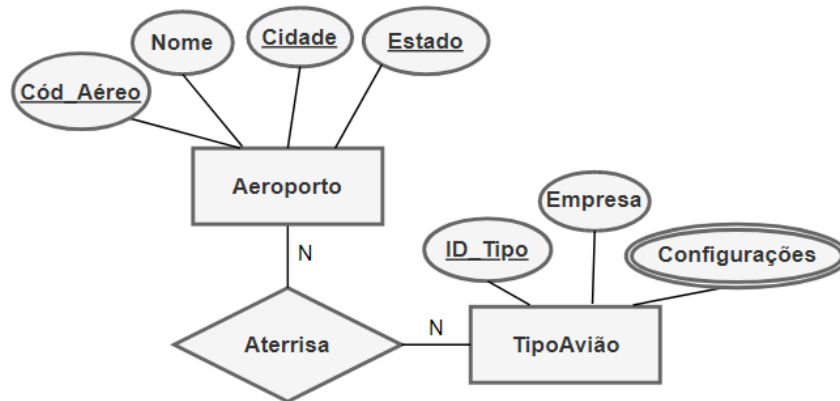
Modelo Entidade-Relacionamento (MER)	Modelo Relacional (MR)
Entidade	Tabela (Relação)
Relacionamento 1 : 1 ou 1 : N	Tabela + UMA chave estrangeira (FK)
Relacionamento M : N	Tabela + DUAS chaves estrangeiras (FKs)
Atributo composto	Conjunto de atributos na Tabela original
Atributo multivalorado	Tabela + UMA chave estrangeira (FK)
Relacionamento n-ário	Tabela + n-chaves estrangeiras (FKs)
Chave	Chave Primária (PK)
Entidade fraca	Tabela + chaves estrangeiras (FKs)



# Mapeamento MER → MR

## > Prática 2:

a)



b)

