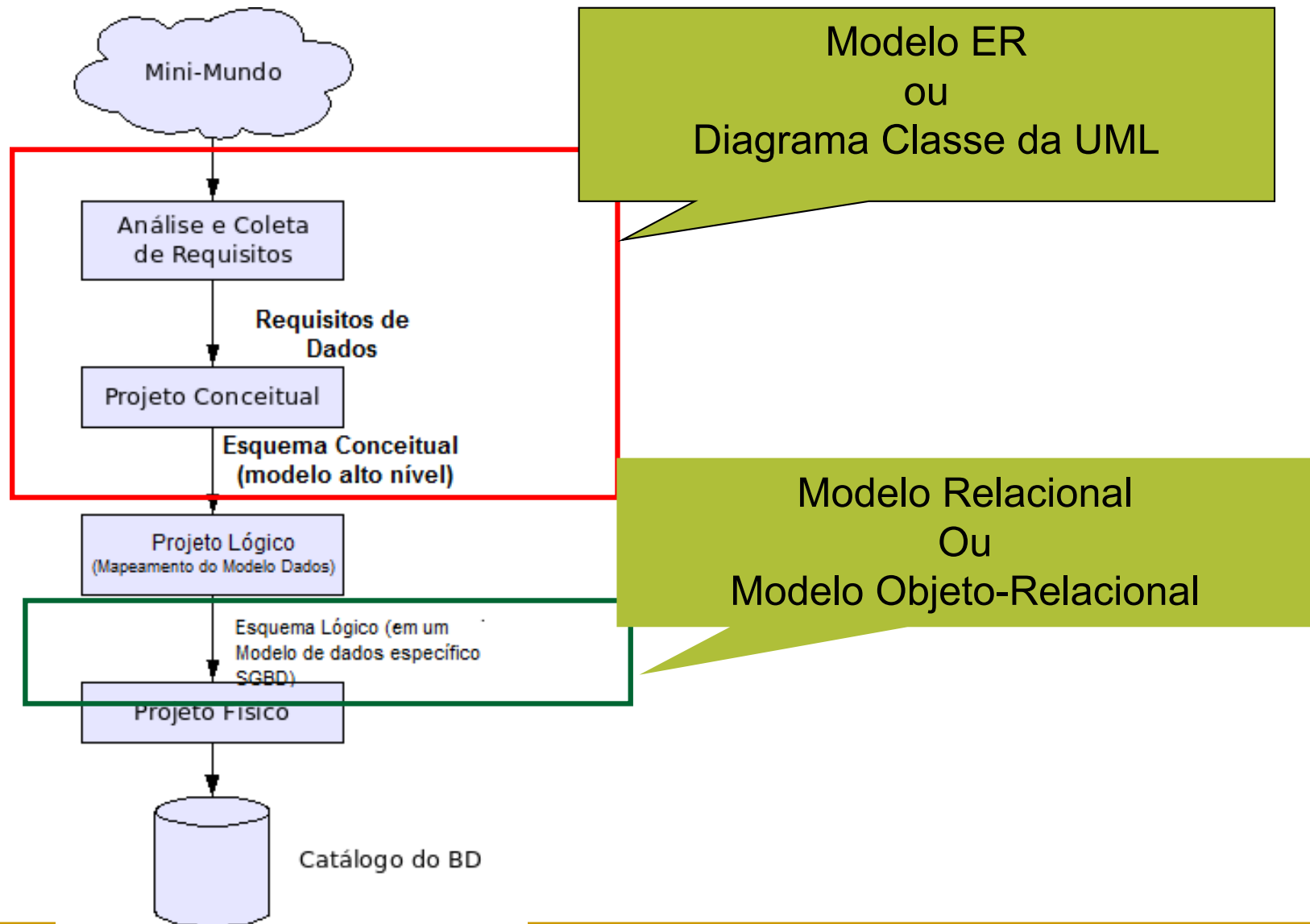

Banco de Dados

Modelo Relacional

Profa.: Márcia Sampaio Lima

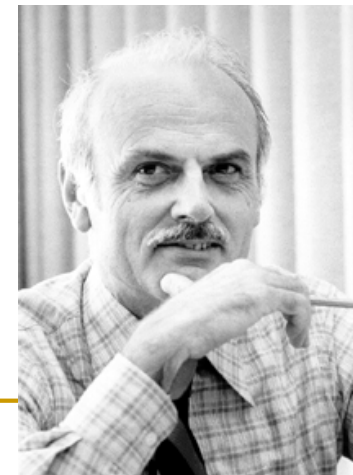
EST - UEA

Projeto de um Banco de Dados



Modelo Relacional

- Modelo de dados mais utilizado em BDs.
- Serve de base para a grande maioria dos SGBDs existentes hoje no mercado.
- Simples, estrutura uniforme.
- Baseado em conceitos matemáticos da teoria de conjuntos.
- Inventado por Edgar F. Codd



Modelo Relacional

- Finalidade representar os dados como uma coleção de **relações**:
 - Cada relação é representada por uma **tabela**, ou um arquivo. Porém, este é mais restrito que uma tabela.
 - Toda tabela pode ser considerada um arquivo, porém, nem todo arquivo pode ser considerado uma tabela.
-

Modelo Relacional

■ Tabelas ou Relações:

Clientes:

Cli_CPF	Cli_Nome	Cli_Sexo	Cli_DataNascimento
098.989.899-90	Maria de Lurdes	F	12/03/1976
124.968.457-07	Pedro Antônio	M	20/09/1981
342.098.452-65	Francisco Silva	M	14/07/1972
626.097.438-56	Sofia Cristina	F	03/10/1972

Modelo Relacional

- Tabelas ou Relações:
 - Quando uma relação é pensada como uma tabela de valores, cada linha nesta tabela representa uma coleção de dados relacionados.

Clientes:

Cli_CPF	Cli_Nome	Cli_Sexo	Cli_DataNascimento
098.989.899-90	Maria de Lurdes	F	12/03/1976
124.968.457-07	Pedro Antônio	M	20/09/1981
342.098.452-65	Francisco Silva	M	14/07/1972
626.097.438-00	Sofia Cristina	F	03/10/1972

Modelo Relacional

- Tabelas ou Relações:
 - Estes valores podem ser interpretados como fatos descrevendo uma instância de uma entidade ou de um relacionamento.

Clientes:

Cli_CPF	Cli_Nome	Cli_Sexo	Cli_DataNascimento
098.989.899-90	Maria de Lurdes	F	12/03/1976
124.968.457-07	Pedro Antônio	M	20/09/1981
342.098.452-65	Francisco Silva	M	14/07/1972
626.097.438-56	Sofia Cristina	F	03/10/1972

Instância

Modelo Relacional

- Tabelas ou Relações:
 - O nome da tabela e das colunas desta tabela são utilizados para facilitar a interpretação dos valores armazenados em cada linha da tabela.

Clientes:

Cli_CPF	Cli_Nome	Cli_Sexo	Cli_DataNascimento
098.989.899-90	Maria de Lurdes	F	12/03/1976
124.968.457-07	Pedro Antônio	M	20/09/1981
342.098.452-65	Francisco Silva	M	14/07/1972
626.097.438-56	Sofia Cristina	F	03/10/1972

Modelo Relacional

- Tabelas ou Relações:
 - Todos os valores em uma coluna são necessariamente do mesmo tipo: domínio.

Clientes:

Cli_CPF	Cli_Nome	Cli_Sexo	Cli_DataNascimento
098.989.899-90	Maria de Lurdes	F	12/03/1976
124.968.457-07	Pedro Antônio	M	20/09/1981
342.098.452-65	Francisco Silva	M	14/07/1972
626.097.438-56	Sofia Cristina	F	03/10/1972

Modelo Relacional

■ DOMÍNIO

- ❑ Representa o conjunto de valores válidos de um atributo de uma relação.
 - ❑ Pode ser representado por
 - Um intervalo de valores
 - ❑ Ex: Idade = de 18 a 50 anos
 - Uma enumeração de valores
 - ❑ Ex: Estado Civil = Solteiro, Casado, Viúvo etc.
 - Um Tipo
 - ❑ Ex: D-FONE: inteiro
-

Modelo Relacional

■ Domínio D:

- ❑ É um conjunto de valores atômicos.
- ❑ Por atômico, podemos compreender que cada valor do domínio é indivisível.
- ❑ Especificação do domínio: destacar o tipo, o tamanho e a faixa do atributo que está sendo especificado. Por exemplo:

Coluna	Tipo	Tamanho	Faixa
Idade	Numérico		0-120
Nome	Caracter	30	
Salário	Numérico		500,00- 129999,99

Modelo Relacional

- Terminologia do modelo relacional:
 - ❑ Cada tabela é chamada de **relação**;
 - ❑ Uma linha de uma tabela é chamada de **tupla ou registro**;
 - ❑ O nome de cada coluna é chamado de **atributo ou campo**;
 - ❑ O tipo de dado que descreve cada coluna é chamado de **domínio**;
 - ❑ **Esquema**: representa a estrutura da tabela ou relação.
-

Modelo Relacional

Cientes:

Esquema	Atributo ou campo			
	Cli_CPF	Cli_Nome	Cli_Sexo	Cli_DataNascimento
	098.989.899-90	Maria de Lurdes	F	12/03/1976
	124.968.457-07	Pedro Antônio	M	20/09/1981
	342.098.452-65	Francisco Silva	M	14/07/1972
	626.097.438-56	Sofia Cristina	F	03/10/1972

Domínio

Tupla ou Registro

Relação ou Tabela

Modelo Relacional

- **Esquema de relação R**, denotado por:

$$R(A_1, A_2, \dots, A_n)$$

- Cada atributo A_i é o nome do papel desempenhado por um domínio **D** no esquema relação **R**.
- **D** é domínio de A_i , denotado por $dom(A_i)$.
- O grau de **R** é o número de atributos presentes em seu esquema de relação.

Grau de R == n

Modelo Relacional

- **Instância r** de um esquema relação denotado por:

$$r(R)$$

- $r(R)$ é um conjunto de n -tuplas $r = [t_1, t_2, \dots, t_n]$ onde os valores de $[t_1, t_2, \dots, t_n]$ devem estar contidos no domínio **D**.
- O valor **nulo (NULL)** faz parte do domínio de um atributo e representa um valor não conhecido para uma determinada tupla.

Modelo Relacional

- Uma n-tupla t :

$$t = \langle v_1, v_2, \dots, v_n \rangle$$

- Onde, v_i é o valor correspondente ao atributo A_i .

Chave...

- **Chave Primária:**

Atributo que identifica unicamente uma tupla.

- Características:

- Imparidade
- Obrigatoriedade

Chaves..

- **Chave Candidata:**

- É qualquer conjunto não nulo de atributos que tem condições de ser chave primária.

- **Chave Alternativa**

- Uma chave candidata que não foi escolhida como chave primária.

- **Super Chave** (Chave composta)

- Chave primária formada por mais de um atributo

- **Chave Estrangeira:**

Um atributo que corresponde a uma chave primária em outra relação.

Modelo Relacional

- Chave Primária (*Primary Key* – PK)
 - É um atributo ou um conjunto de atributos utilizado para identificar uma única tupla em uma relação
 - Toda tabela ou relação deve ter uma chave primária
-

Modelo Relacional

■ Chave Primária

Cli_CPF	Cli_Nome	Cli_Sexo	Cli_DataNascimento
098.989.899-90	Maria de Lurdes	F	12/03/1976
124.968.457-07	Pedro Antônio	M	20/09/1981
342.098.452-65	Francisco Silva	M	14/07/1972
626.097.438-56	Sofia Cristina	F	03/10/1972

Modelo Relacional

■ Chave Primária

<u>Cli_CPF</u>	Cli_Nome	Cli_Sexo	Cli_DataNascimento
098.989.899-90	Maria de Lurdes	F	12/03/1976
124.968.457-07	Pedro Antônio	M	20/09/1981
342.098.452-65	Francisco Silva	M	14/07/1972
626.097.438-56	Sofia Cristina	F	03/10/1972

Modelo Relacional

- Chave Primária

- Características

- Pode ser simples ou composta
 - É única para cada tabela ou relação
 - Não pode ter valores repetidos
 - Não pode ser nulo (NULL)
-

Modelo Relacional

- Chave Primária
 - Pode ser simples ou composta

Vei_placa	Vei_estad o	Vei_modelo	Vei_marca	Vei_ano	Chassi
JWZ-9096	AM	Astra	Chevrolet	2007	0459F75G
JNR-7876	AM	EcoSport	Ford	2008	6783H29K
JWZ-9096	PE	Siena	Fiat	2007	23J96JU8
JWZ-7853	SP	Astra	Chevrolet	2004	67KI23HO

Modelo Relacional

■ Chave Primária

- Composta: formada por dois ou mais atributos

<u>Vei_placa</u>	<u>Vei_estad</u> <u>o</u>	Vei_modelo	Vei_marca	Vei_ano	Chassi
JWZ-9096	AM	Astra	Chevrolet	2007	0459F75G
JNR-7876	AM	EcoSport	Ford	2008	6783H29K
JWZ-9096	PE	Siena	Fiat	2007	23J96JU8
JWZ-7853	SP	Astra	Chevrolet	2004	67KI23HO

Modelo Relacional

■ Chave Primária

- Composta: formada por dois ou mais atributos

Dependente

<u>Dep_RGRe sponsavel</u>	<u>Dep_Nome Dep</u>	Dep_dataNasc	Dep_relacao	Dep_sexo
10101010	Jorge	27/12/86	Filho	M
10101010	Luiz	18/11/79	Filho	M
20202020	Fernanda	14/02/69	Cônjuge	F
20202020	Angelo	10/02/95	Filho	M

Modelo Relacional

■ Chave Primária

- Uma tabela pode ter mais de uma chave primária?

□ Não

<u>Vei_placa</u>	<u>Vei_estad</u> <u>o</u>	Vei_modelo	Vei_marca	Vei_ano	Chassi
JWZ-9096	AM	Astra	Chevrolet	2007	0459F75G
JNR-7876	AM	EcoSport	Ford	2008	6783H29K
JWZ-9096	PE	Siena	Fiat	2007	23J96JU8
JWZ-7853	SP	Astra	Chevrolet	2004	67KI23HO

Modelo Relacional

■ Chave Primária

- Não pode ter valores repetidos

<u>Mec_Matricul</u> <u>a</u>	Mec_Nome	Mec_dataAdmissa o	Mec_CPF	Mec_RG
098	João Paulo	20/03/2007	098.098.987-90	348372
283	Pedro Silva	10/09/1999	567.699.059-65	213255
098	Marcelo Paiva	03/04/2005	434.145.059-90	455763
045	Cleber Silva	10/11/2005	324.059.183-96	342829

Modelo Relacional

■ Chave Primária

□ Não pode ser nulo

■ Nulo = ausência de valor

<u>Mec_Matricul</u> <u>a</u>	Mec_Nome	Mec_dataAdmissa o	Mec_CPF	Mec_RG
098	João Paulo	20/03/2007	098.098.987-90	348372
265	Pedro Silva	10/09/1999	NULL	213255
NULL	Marcelo Paiva	03/04/2005	434.145.059-90	NULL
045	Cleber Silva	10/11/2005	324.059.183-96	342829

Chaves

Chave Primária

Atributos

Tab. PILOTO

IdPiloto#	Nome	CPF	Endereço
0101	João	123456	Recife
0035	José	234567	São Paulo
.	.	.	.
0987	Pedro	567890	Recife

Tuplas

**T
A
B
E
L
A**

Chave Alternativa

Modelo Relacional

- Uma relação pode ser definida como um conjunto de tuplas distintas.
- A combinação dos valores dos atributos em uma tupla não pode se repetir na mesma tabela.
- Existirá sempre um subconjunto de atributos em uma tabela que garantem que não haverá valores repetidos para as diversas tuplas da mesma.

$$t1[PK] \neq t2[PK]$$

Modelo Relacional

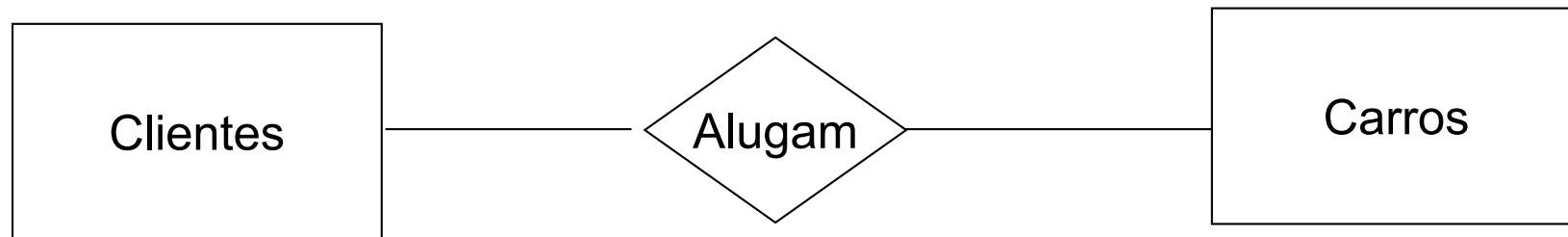
- Chave Estrangeira (*Foreing Key* – FK)
 - É a forma como se representa os relacionamentos entre duas relações R1 e R2 Modelo Relacional
 - Um conjunto de atributos é uma FK na relação R1 se satisfizer as seguintes regras:
 - Devem ter o mesmo domínio que os atributos da PK da relação R2
 - Deve ter os mesmos valores da PK de uma tupla de R2 ou deve ser null

$$t1[FK] = t2[PK]$$

Modelo Relacional

- Chave Estrangeira (*Foreing Key* – FK)
 - R1 é chamada de **relação referência** e R2 é a **relação referida**

Modelo Relacional



Modelo Relacional

- Chave Estrangeira (*Foreing Key* – FK)
 - Qual o cliente que fez o aluguel?

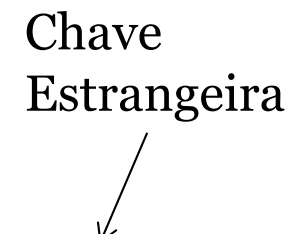
Aluguel

Alu_data	Alu_devPrevista	Alu_devolucao	Alu_Valor
20/03/2009	25/03/2009	25/03/2009	400,00
20/03/2009	23/03/2009	24/03/2009	320,00
25/03/2009	27/03/2009	27/03/2009	160,00
28/03/2009	03/04/2009	02/04/2009	400,00

Modelo Relacional

■ Chave Estrangeira (*Foreing Key* – FK)

Chave
Estrangeira



Alu_data	Alu_devPrevista	Alu_devolucao	Alu_Valor	Cli_CPF
20/03/2009	25/03/2009	25/03/2009	400,00	098.989.899-90
20/03/2009	23/03/2009	24/03/2009	320,00	124.968.457-07
25/03/2009	27/03/2009	27/03/2009	160,00	NULL
28/03/2009	03/04/2009	02/04/2009	400,00	626.097.438-56

Modelo Relacional

- Chave Estrangeira (*Foreing Key* – FK)
 - Cli_CPF é uma FK
 - Mesmo domínio da PK de Cliente
 - Possui valores que estão na PK de Cliente ou é NULL

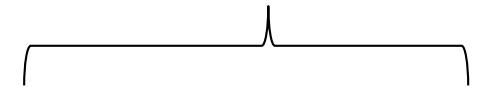
Alu_data	Alu_devPrevista	Alu_devolucao	Alu_Valor	Cli_CPF
20/03/2009	25/03/2009	25/03/2009	400,00	098.989.899-90
20/03/2009	23/03/2009	24/03/2009	320,00	124.968.457-07
25/03/2009	27/03/2009	27/03/2009	160,00	NULL
28/03/2009	03/04/2009	02/04/2009	400,00	626.097.438-56

Modelo Relacional

- Chave Estrangeira (*Foreing Key* – FK)

- Qual carro foi alugado?

Chave
Estrangeira



Alu_data	Alu_dev Prevista	Alu_ devolucao	Alu_Valor	Cli_CPF	Vei_Placa	Vei_Estado
20/03/2009	25/03/2009	25/03/2009	400,00	098.989.899-90	JWZ-9096	AM
20/03/2009	23/03/2009	24/03/2009	320,00	124.968.457-07	JNR-7876	AM
25/03/2009	27/03/2009	27/03/2009	160,00	NULL	JWZ-9096	PE
28/03/2009	03/04/2009	02/04/2009	400,00	626.097.438-56	JWZ-7853	SP

Modelo Relacional

■ Restrições

- São limitações impostas aos dados de um banco de dados para que sejam válidos
 - Tipos de restrições
 - Restrições de Chaves
 - Restrições de Nulos
 - Restrições de Domínio
-

Modelo Relacional

- **Restrições de integridade:**

- São usados para garantir a exatidão e a consistência dos dados em uma BD relacional.
- Garantem que dados representem assertivamente a realidade modelada.
- Tipos de restrições de integridade:

- .

Modelo Relacional

■ Restrições de integridade:

- ❑ São usados para garantir a exatidão e a consistência dos dados em uma BD relacional.
 - ❑ Garantem que dados representem assertivamente a realidade modelada.
 - ❑ Tipos de restrições de integridade:
-

Modelo Relacional

■ Restrições de Integridade:

- ❑ **Integridade de Domínio:** diz respeito ao controle sintático e semântico de um dado fazendo referência ao tipo de definição do domínio.
 - ❑ **Integridade de Entidade:** diz respeito aos valores de chave primária que devem ser únicos e não nulos.
 - ❑ **Integridade Referencial:** diz respeito aos valores de um atributo chave estrangeira e os valores do atributo chave primária correspondente.
-

Modelo Relacional

■ Restrições de Chaves

□ Chave primária

- Não pode repetir, não pode ser nulo.

- A PK é usada para individualizar as tuplas.

- Se ocorre NULL na PK, poderemos não identificar alguma tupla.

- Impede que uma chave primária se repita.

- Um campo chave primária diferencia de forma única os registros (linhas) de uma relação (tabela).

Modelo Relacional

<u>Cli_CPF</u>	Cli_Nome	Cli_Sexo	Cli_DataNascimento
098.989.899-90	Maria de Lurdes	F	12/03/1976
NULL 	Pedro Antônio	M	20/09/1981
342.098.452-65	Francisco Silva	M	14/07/1972
NULL 	Sofia Cristina	F	03/10/1972

Modelo Relacional

- Restrições de Chaves

- Chave estrangeira

- Restrição de Integridade Referencial

- É usada para manter a consistência entre as tuplas em duas relações.

- Não pode remover valores da PK da **relação referida** se houver registros na **relação referência** com tais valores

- Não pode inserir valores na FK da **relação referência** se os mesmos não constarem na PK da **relação referida**
-

Modelo Relacional

Notação para o Relacional:

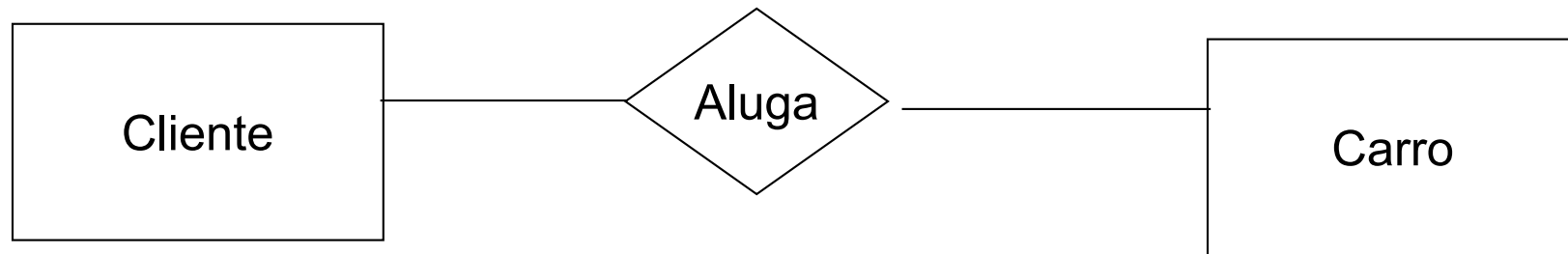
TabNome (PK, Col₁, Col₂, FK*, ..., Col_n)

Exemplo:

Clientes (IdCliente, Nome, Endereço, Fone)

Funcionarios (Matric, Nome, Fone, IdDepto*)

Modelo Relacional



Cliente (cli_cpf, cli_nome, cli_sexo, cli_dataNascimento)

Aluguel (alu_codigo, alu_data, alu_dataDevolucao, Alu_valor,
alu_devPrevista, cli_cpf*, car_codigo*)

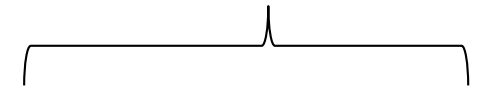
Carro (car_codigo, car_modelo, car_ano, car_cor)

Modelo Relacional

- Chave Estrangeira (*Foreing Key* – FK)

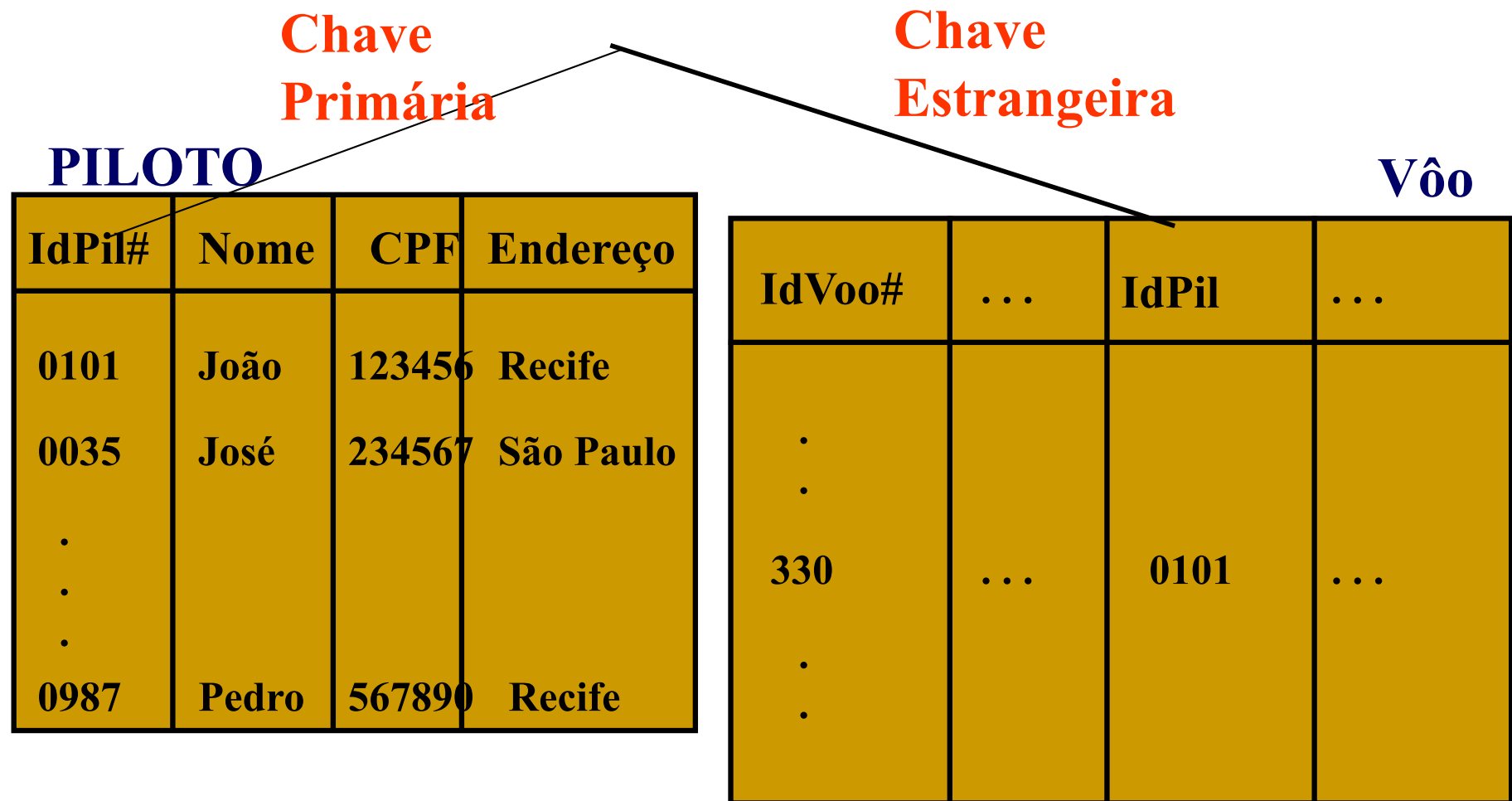
- Qual carro foi alugado?

Chave
Estrangeira



Alu_data	Alu_dev Prevista	Alu_ devolucao	Alu_Valor	Cli_CPF	Vei_Placa	Vei_Estado
20/03/2009	25/03/2009	25/03/2009	400,00	098.989.899-90	JWZ-9096	AM
20/03/2009	23/03/2009	24/03/2009	320,00	124.968.457-07	JNR-7876	AM
25/03/2009	27/03/2009	27/03/2009	160,00	NULL	JWZ-9096	PE
28/03/2009	03/04/2009	02/04/2009	400,00	626.097.438-56	JWZ-7853	SP

Chave Estrangeira



Modelo Relacional

- Restrição de Nulos
 - Indica que um atributo não pode ser nulo
 - Deve ser **obrigatoriamente** preenchido
 - Exemplo: toda tupla de `ALUNO` possui um valor válido, não nulo para o atributo:
 - Nome de `Aluno` é forçado a ser **NOT NULL**
-

Modelo Relacional

■ Restrição de Nulos

- ❑ Cliente (cli_codigo, cli_nome, cli_cpf, cli_sexo, cli_cnpj, cli_telefone)
- ❑ cli_nome NOT NULL
- ❑ cli_telefone NULL
- ❑ Se cli_cpf estiver preenchido, cli_sexo deve estar preenchido e cli_cnpj deve ser nulo
- ❑ Se cli_cnpj estiver preenchido, cli_cpf e cli_sexo devem estar nulos.

Modelo Relacional

- Restrição de Domínio

- Todo atributo possui um conjunto de valores que é capaz de armazenar
 - Esse conjunto de valores é chamado de domínio. Sendo A um atributo temos:
 - $\text{dom}(A)$
 - Exemplo: inteiro curto, inteiro longo, números reais, caracteres, booleanos, data, hora, moeda, etc.
-

Modelo Relacional

- Restrição de Domínio

- O domínio indica os possíveis valores de um atributo.
- A integridade de domínio verifica se os dados são do tipo permitido (alfanumerico, numerico,etc), tamanho do campo, se ele pode ser nulo ou não.

- Por exemplo, é possível definir que um atributo "idade" de um funcionário é sempre um valor inteiro positivo.
-

Modelo Relacional

■ Restrição de Domínio

- Existem domínios Ilimitados e Limitados.

- Domínio Ilimitado

 - cli_codigo Inteiro

 - cli_nome varchar(100)

Restrição de
tamanho



- Domínio Limitado

 - cli_sexo char(1)

M – Masculino
F - Feminino



Modelo Relacional

- Operações de Atualização e Tratamento de violação de Restrições
 - Operações que modificam ou atualizam o BD:
 - **Insert**
 - **Delete**
 - **Update**
 - Estas operações podem violar certos tipos de restrições.
 - Sempre que forem aplicadas, o estado do BD deve ser verificado.
-

Modelo Relacional

■ Operação de Insert

- ❑ Insere novas tuplas na relação.
 - ❑ Pode violar as seguintes restrições:
 - Restrição de domínio:
 - ❑ Se um dos valores do atributo dado não pertence ao domínio especificado.
 - Restrição de chave:
 - ❑ Se o valor de, em uma nova tupla t , já existir em outra tupla da relação.
 - Restrição de Entidade:
 - ❑ Se a PK da nova tupla for NULL.
-

Modelo Relacional

- Operação de Insert
 - Pode violar as seguintes restrições:
 - Integridade Referencial:
 - Se o valor de qq FK em t referir-se a uma tupla inexistente.

Modelo Relacional

- Operação de Delete
 - Exclui tuplas na relação.
 - Pode violar as seguintes restrições:
 - Integridade Referencial:
 - Se a tupla removida for FK de outras tuplas do BD.
-

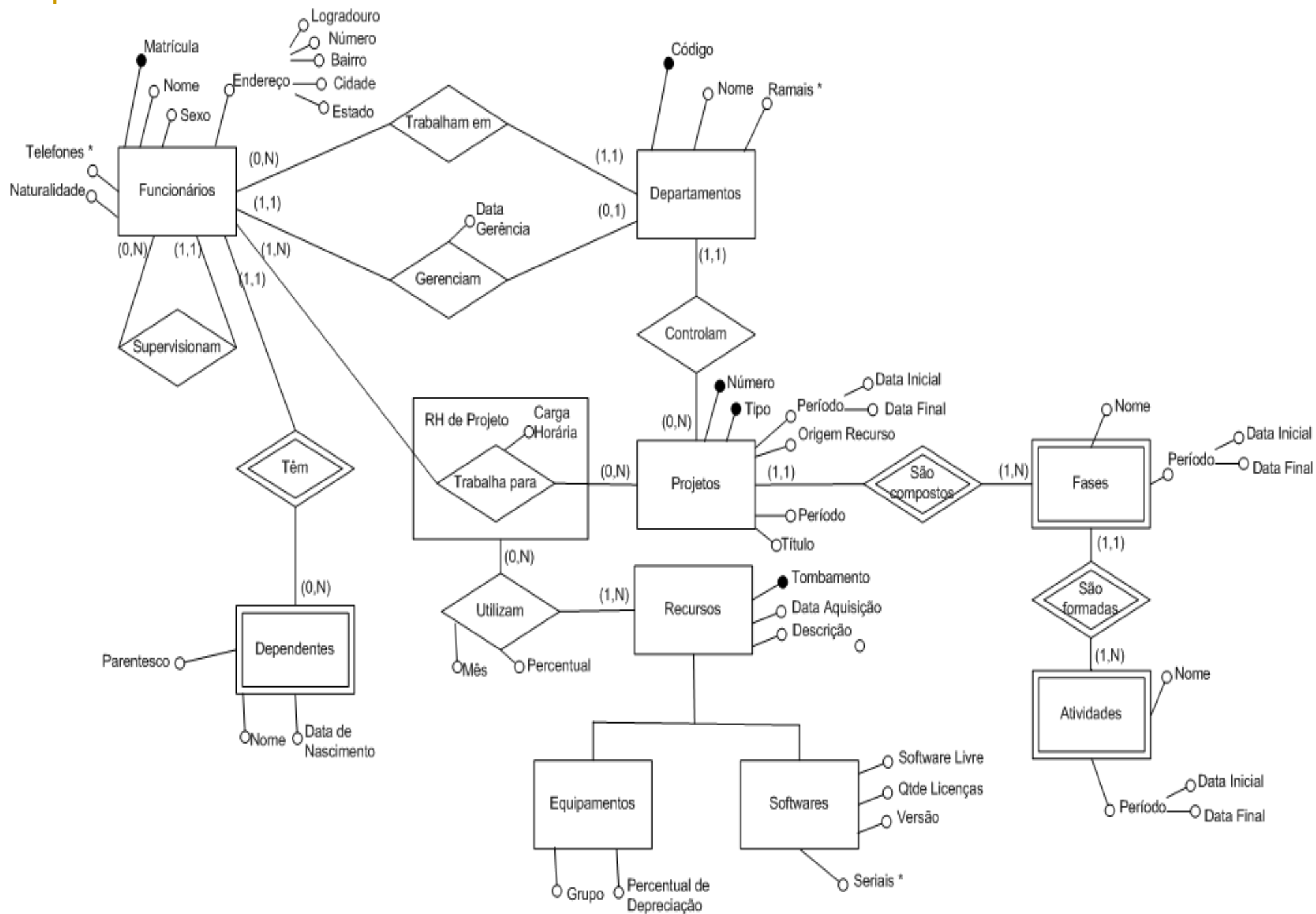
Modelo Relacional

■ Operação de Update

- ❑ Muda os valores de um ou mais atributo de uma tupla.
 - ❑ Tipo de violação:
 - Se não for PK e nem FK:
 - ❑ Verificar a restrição de domínio.
 - Se for PK:
 - ❑ Similar a remover a tupla e inserir uma nova tupla.
 - Se for FK:
 - ❑ Integridade Referencial.
-

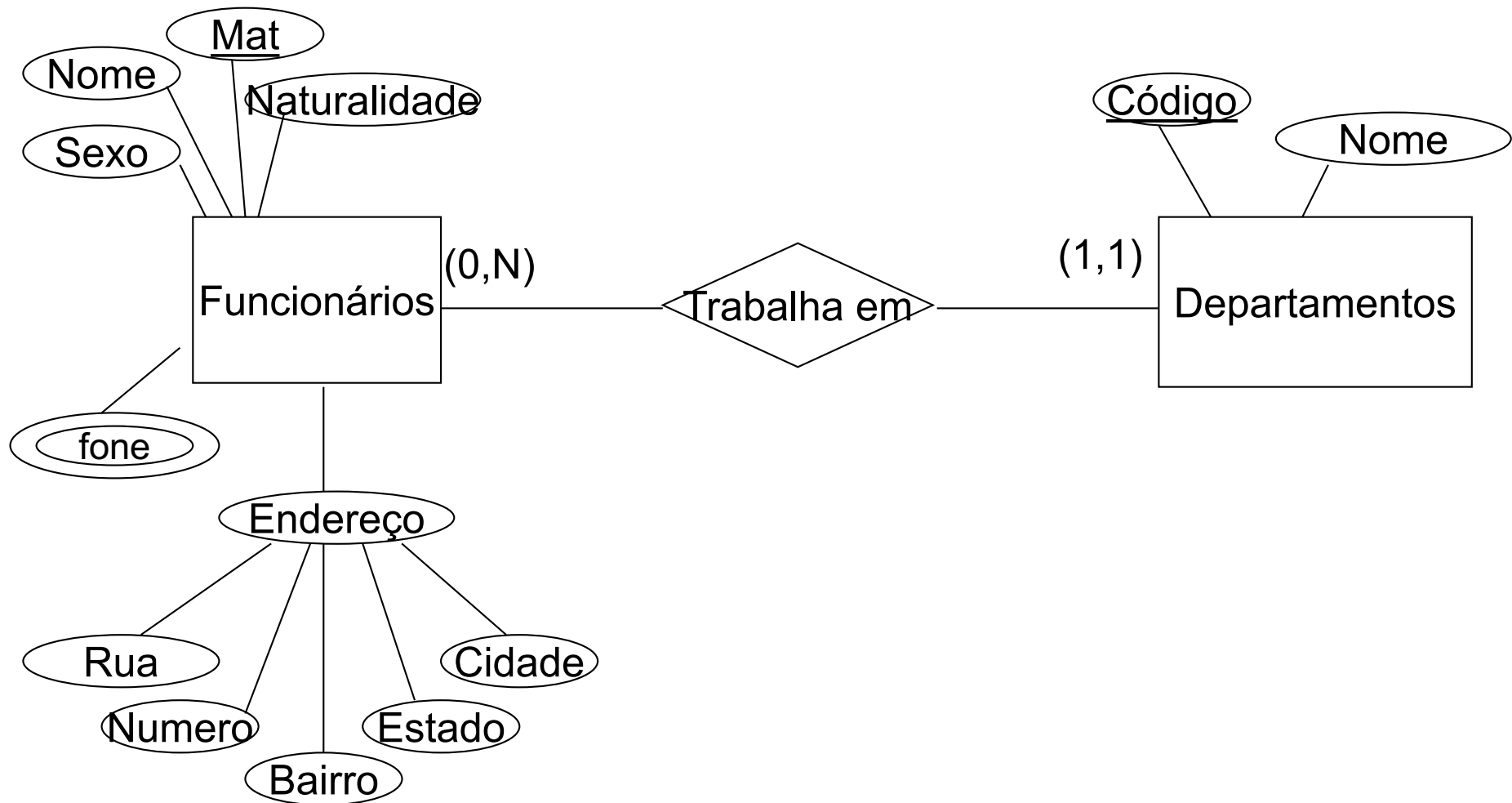
Modelo Relacional

MAPEAMENTO DO MER PARA O MODELO RELACIONAL



Mapeamento do MER para Relacional

- PASSO 1 – Conjunto de Entidades Regular
 - ❑ Todo conjunto de entidades regular E_1 deve ser mapeado para uma relação R_1
 - ❑ Deve ser definida a PK de R_1
 - **Normalmente**, é a mesma chave primária de E_1
 - ❑ Todos os atributos simples de E_1 devem ser mapeados como atributos de R_1
 - ❑ Cada componente de um atributo composto de E_1 deve ser mapeada como um atributo de R_1



Mapeamento do MER para Relacional

■ PASSO 1 - Exemplo

- ❑ Funcionario (fun_mat , fun_nome, fun_sexo, fun_naturalidade, fun_rua, fun_numero, fun_bairro, fun_cidade, fun_estado)
 - ❑ Departamento (dep_codigo, dep_nome)
-

Mapeamento do MER para Relacional

■ PASSO 1 - Outro Exemplo de notação

- Funcionario = {fun_matricula# , fun_nome, fun_sexo, fun_naturalidade, fun_rua, fun_numero, fun_bairro, fun_cidade, fun_estado}

Funcionario

<u>fun_mat</u>	fun_nome	fun_sexo	fun_naturalidade	fun_rua	fun_numero	fun_bairro	fun_cidade	fun_estado
----------------	----------	----------	------------------	---------	------------	------------	------------	------------

- Departamento = {dep_codigo#, dep_nome}

Departamento

<u>dep_codigo</u>	dep_nome
-------------------	----------

Mapeamento do MER para Relacional

■ PASSO 1 - Outro Exemplo de notação

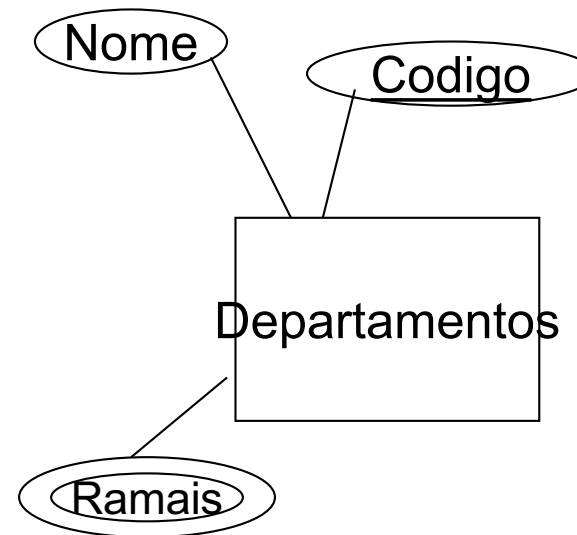
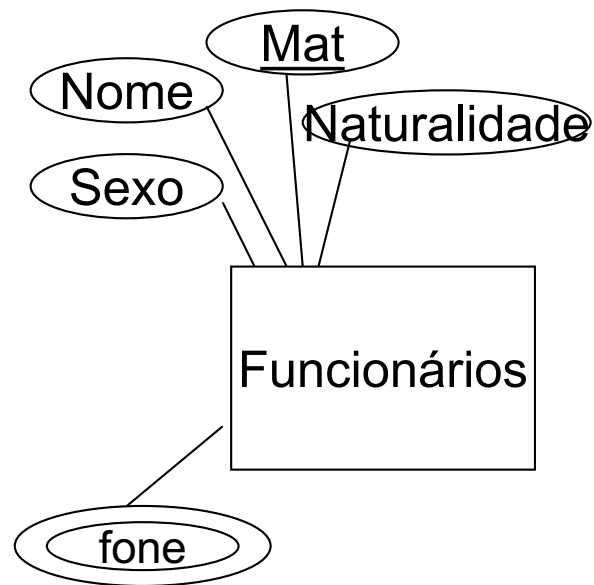
- Funcionario = {fun_matricula , fun_nome, fun_sexo, fun_naturalidade, fun_rua, fun_numero, fun_bairro, fun_cidade, fun_estado}
- `Funcionario(fun_matricula , fun_nome, fun_sexo, fun_naturalidade, fun_rua, fun_numero, fun_bairro, fun_cidade, fun_estado)`
- Departamento = {dep_codigo, dep_nome}
- `Departamento(dep_codigo , dep_nome)`

Mapeamento do MER para Relacional

■ PASSO 2 – Atributo Multivalorado

- Todo atributo multivalorado A_1 de um conjunto de entidades E_1 deve ser mapeado para uma nova relação R_2
- Em R_2 deve haver uma FK que referencia R_1 (mapeamento de E_1)
- A chave primária de R_2 deve ser composta pelo conjunto de atributos da FK associado ao atributo correspondente a A_1

Mapeamento do MER para Relacional



Mapeamento do MER para Relacional

■ PASSO 2 – Exemplo

■ Fun_Telefone (fun_matricula*, fun_telefone)

Funcionario

<u>fun_mat</u>	fun_nome	fun_sexo	fun_naturalidade	fun_rua	fun_numero	fun_bairro	fun_cidade	fun_estado
----------------	----------	----------	------------------	---------	------------	------------	------------	------------

Fun_Telefone

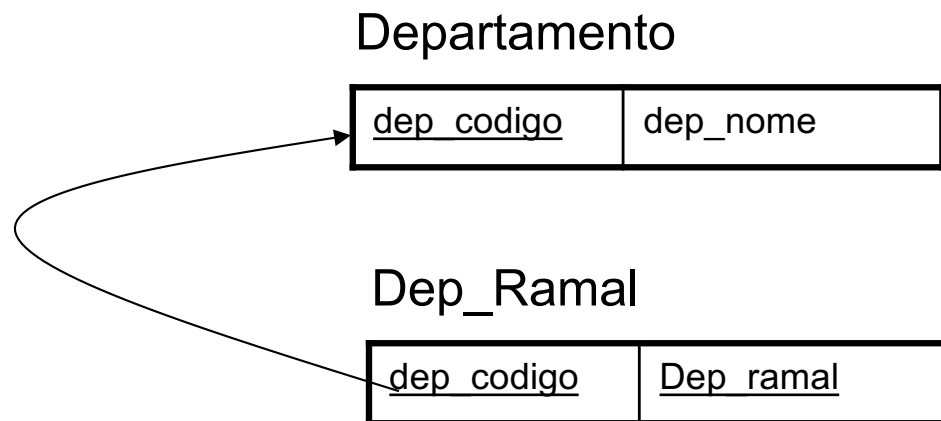
<u>fun_matricula</u>	<u>fun_telefone</u>
----------------------	---------------------



Mapeamento do MER para Relacional

■ PASSO 2 – Exemplo

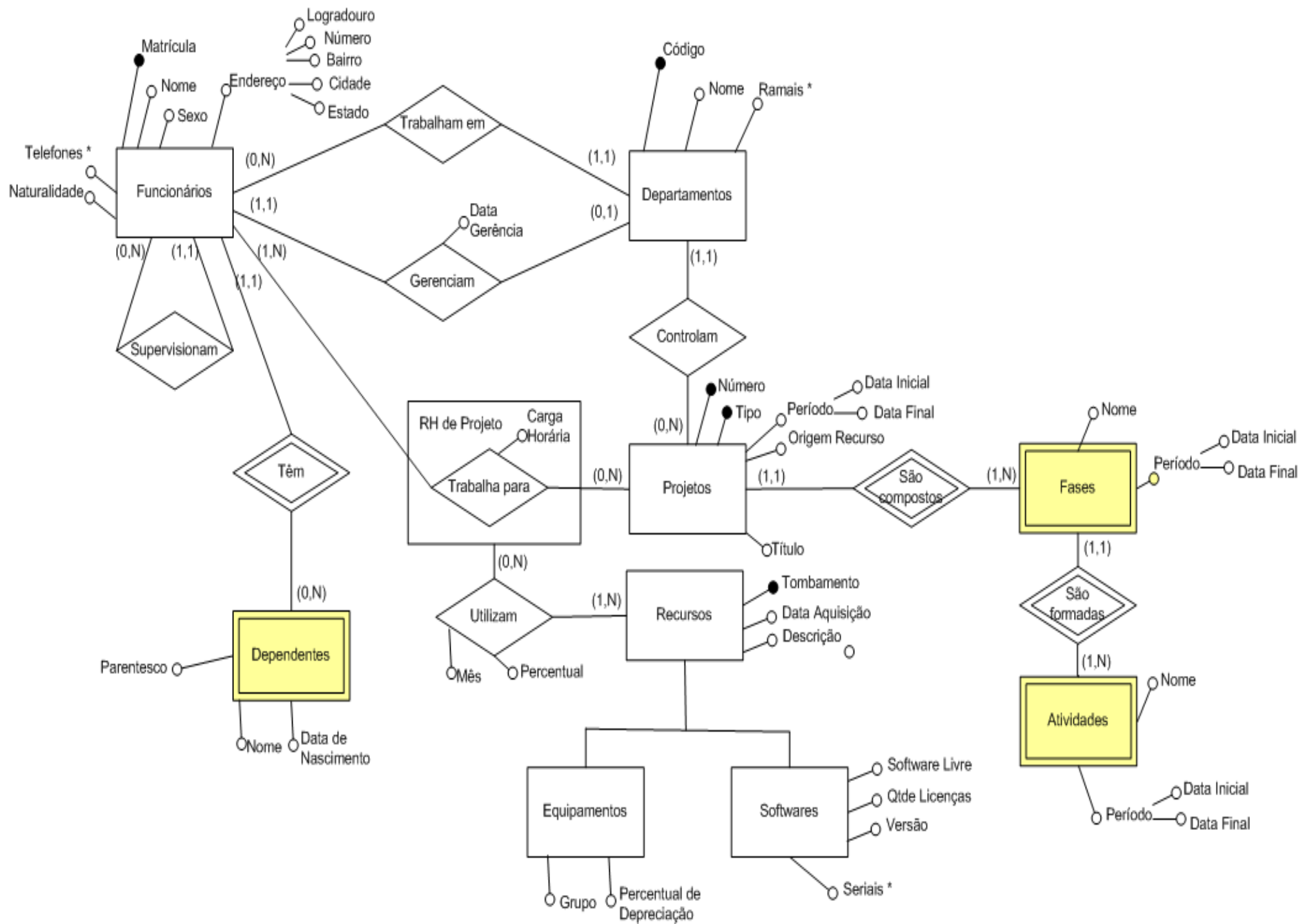
□ Dep_Ramal (dep_codigo^{*}, dep_ramal)



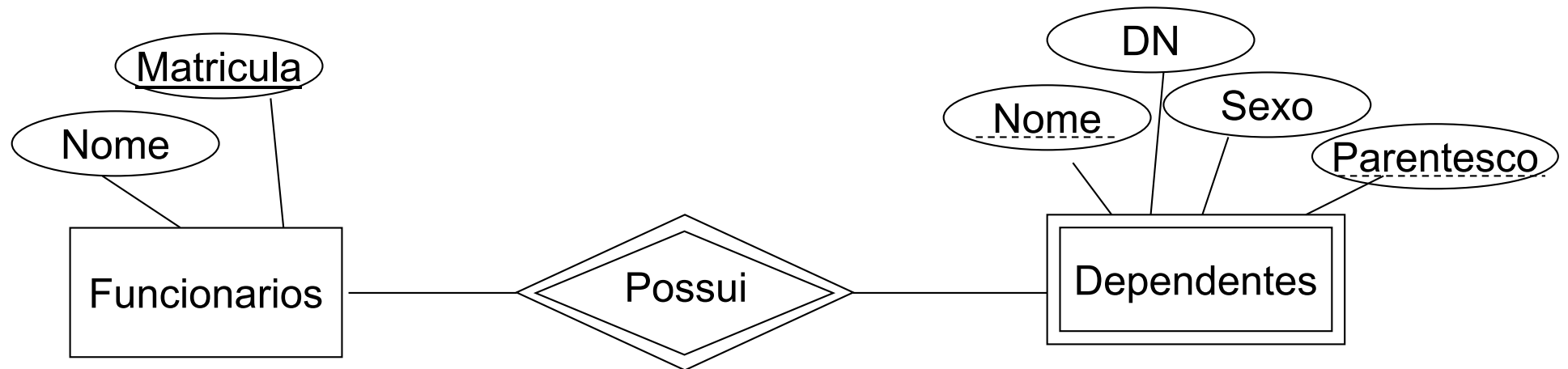
Mapeamento do MER para Relacional

■ PASSO 3 – Conjunto de Entidades Fracas

- Todo conjunto de entidades fraca E_1 deve ser mapeado para um relação R_1
- Em R_1 devem ser incluídos todos os atributos simples de E_1
- Considerando E_2 o conjunto de entidades forte de E_1 , incluir em R_1 a FK da relação correspondente a E_2
- A PK de R_1 deve ser a combinação da chave primária da relação correspondente a E_2 e da chave parcial de E_1



Mapeamento do MER para Relacional



Mapeamento do MER para Relacional

■ PASSO 3 - Exemplo

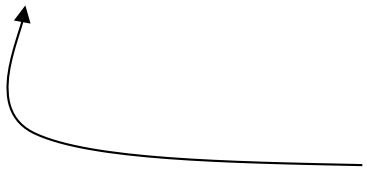
- Dependente = {fun_matricula*, dep_nome, dep_parentesco, dep_dataNascimento}
- Dependente (fun_matricula*, dep_nome, dep_parentesco, dep_dataNascimento)

Funcionario

<u>fun_mat</u>	fun_nome	fun_sexo	fun_naturalidade	fun_rua	fun_numero	fun_bairro	fun_cidade	fun_estado
----------------	----------	----------	------------------	---------	------------	------------	------------	------------

Dependente

<u>fun_matricula</u>	<u>dep_nome</u>	<u>dep_parentesco</u>	dep_dataNascimento
----------------------	-----------------	-----------------------	--------------------



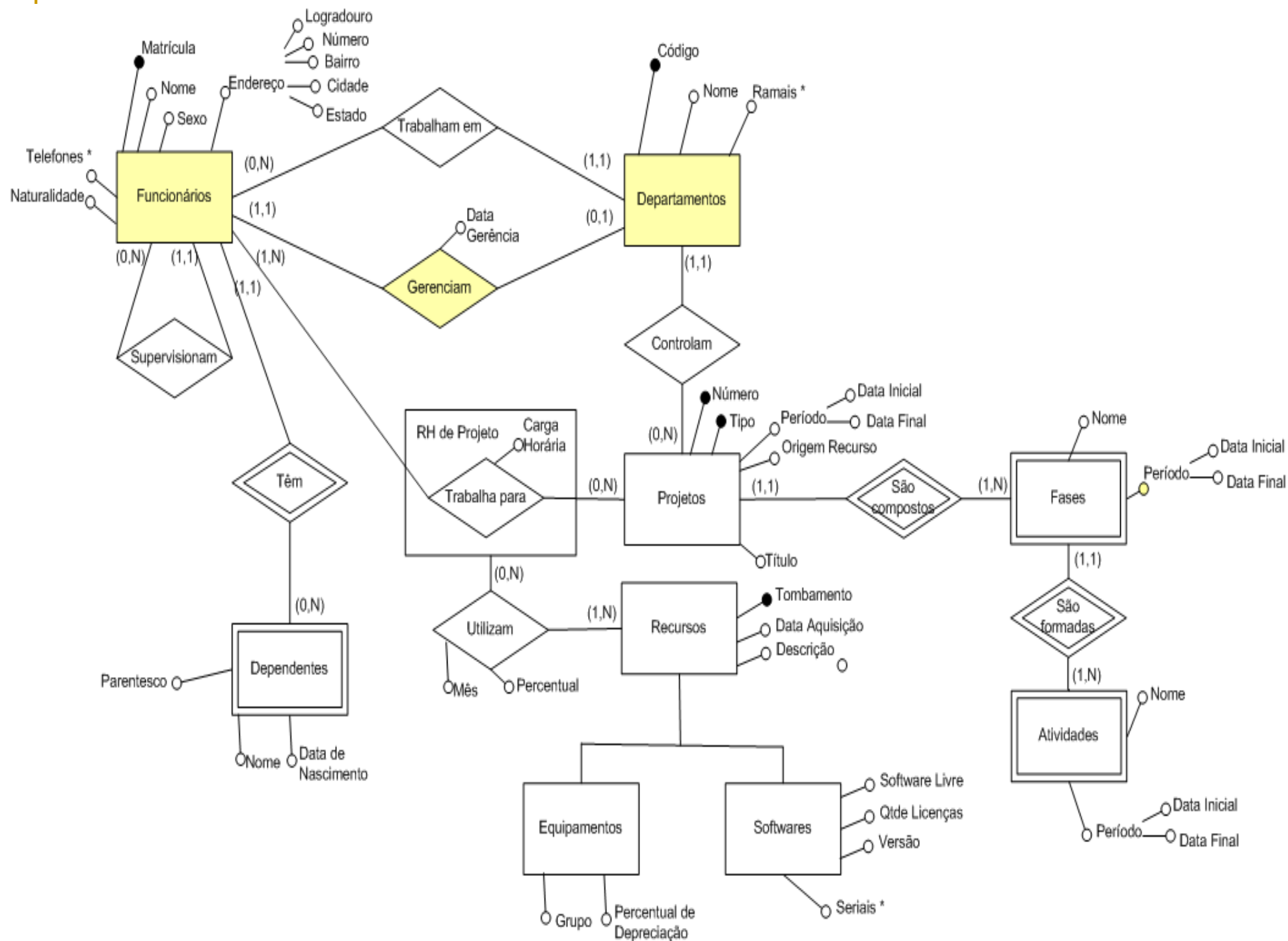
Mapeamento do MER para Relacional

- PASSO 4
 - Relacionamentos
 - Um-para-um
 - Um-para-muitos
 - Muitos-para-muitos
-

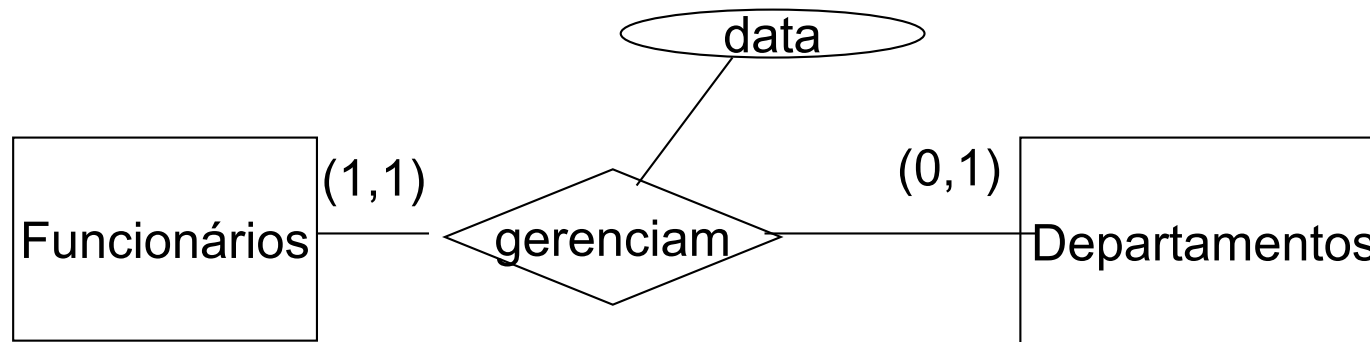
Mapeamento do MER para Relacional

■ PASSO 4.1 - Um-para-um

- Sendo um relacionamento um-para-um entre dois conjuntos de entidades E_1 e E_2 , no qual E_2 tem participação total.
 - Inserir na relação correspondente a E_2 , como FK, a chave primária da relação correspondente a E_1 .
 - Se o relacionamento possuir atributos, os mesmos devem também ser inseridos na relação correspondente a E_2
-



Mapeamento do MER para Relacional



Mapeamento do MER para Relacional

■ PASSO 4.1 – Exemplo

- ❑ Funcionario (fun_matricula, fun_nome, fun_sexo, fun_naturalidade, fun_rua, fun_numero, fun_bairro, fun_cidade, fun_estado)
- ❑ Departamento (dep_codigo, dep_nome, **fun_matricula*, ger_dataGerencia**)

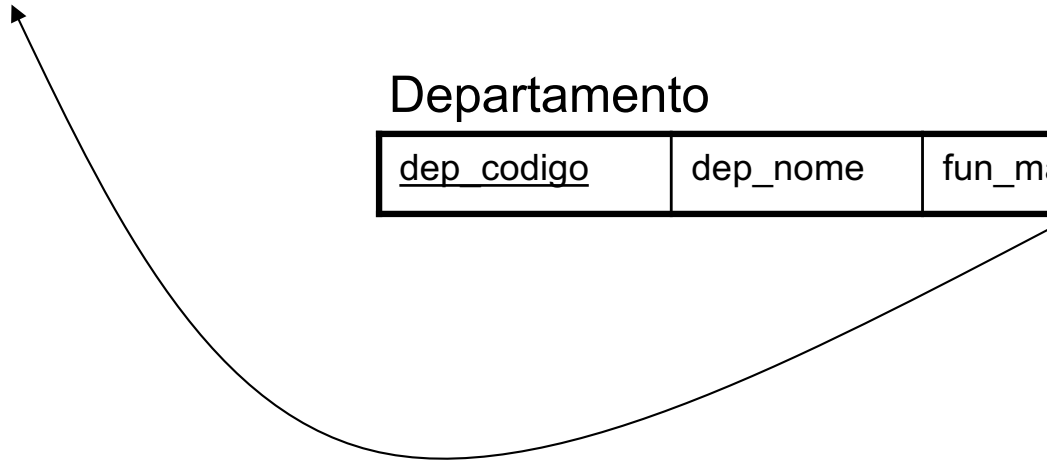
Mapeamento do MER para Relacional

Funcionario

<u>fun_mat</u>	fun_nome	fun_sexo	fun_naturalidade	fun_rua	fun_numero	fun_bairro	fun_cidade	fun_estado
----------------	----------	----------	------------------	---------	------------	------------	------------	------------

Departamento

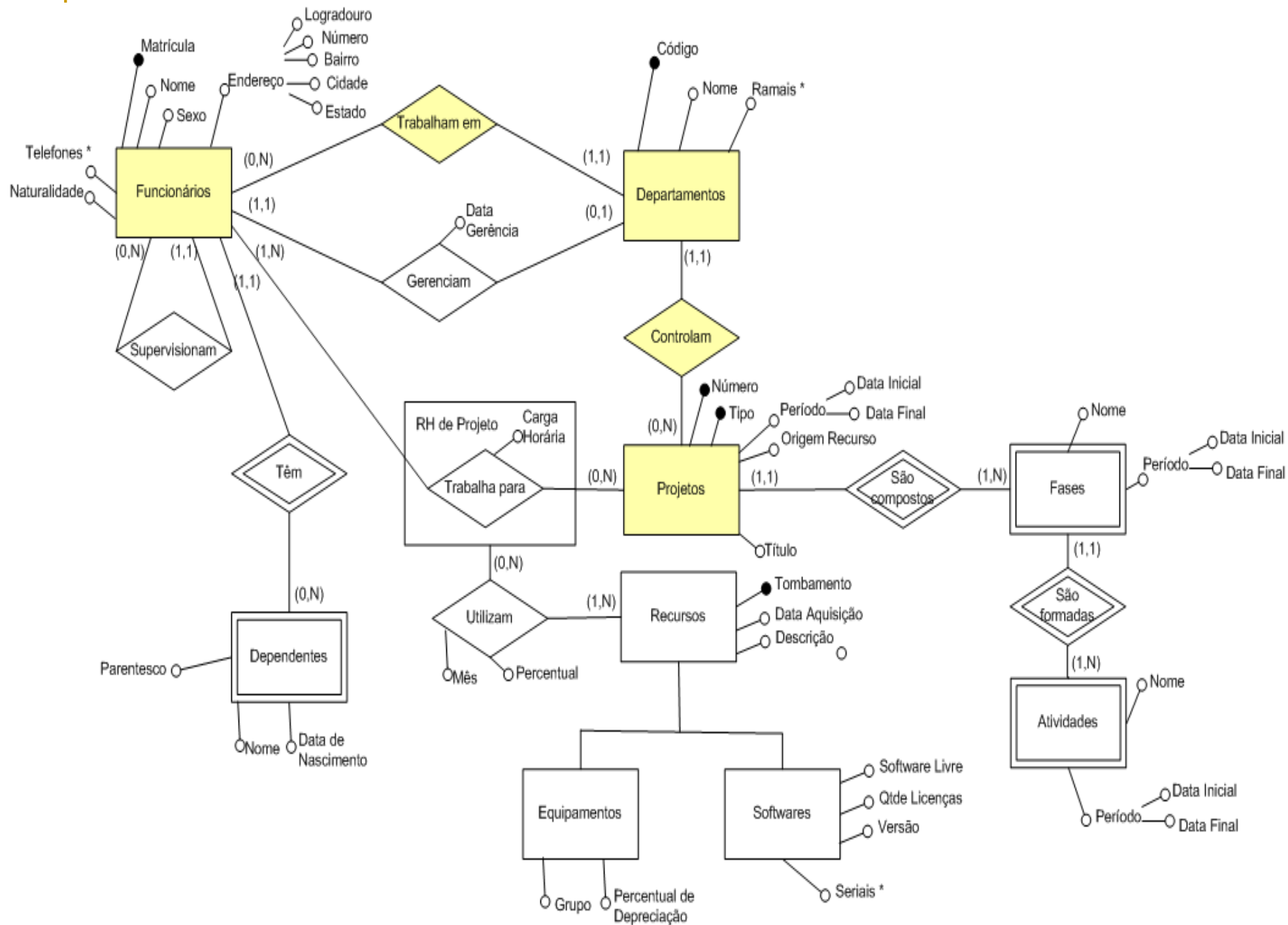
<u>dep_codigo</u>	dep_nome	fun_matricula	ger_dataGerencia
-------------------	----------	---------------	------------------



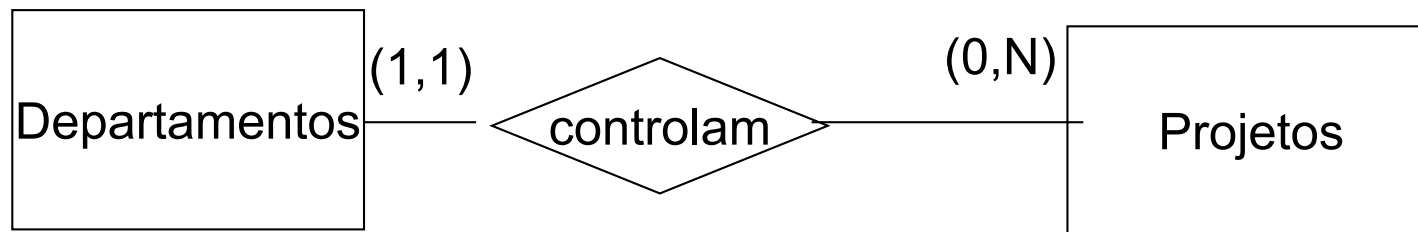
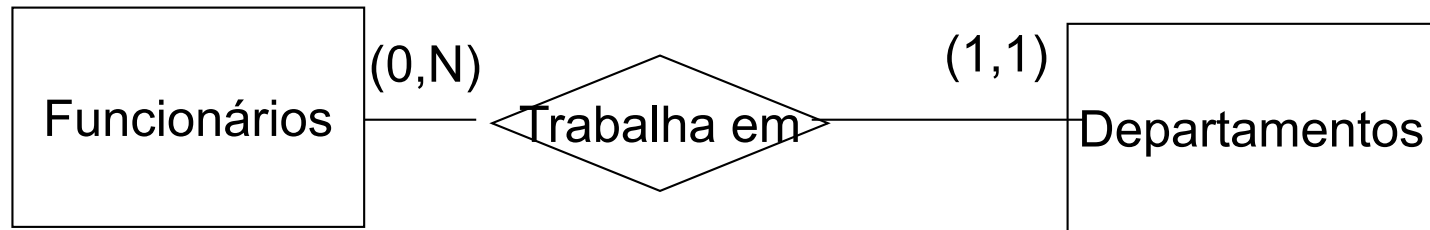
Mapeamento do MER para Relacional

■ PASSO 4.2 - Um-para-muitos

- Sendo um relacionamento um-para-muitos (1:N) entre dois conjuntos de entidades E_1 e E_2 , no qual E_2 está do “lado” N
- Inserir na relação correspondente a E_2 , como FK, a chave primária da relação correspondente a E_1
- Se o relacionamento possuir atributos, os mesmos devem também ser inseridos na relação correspondente a E_2



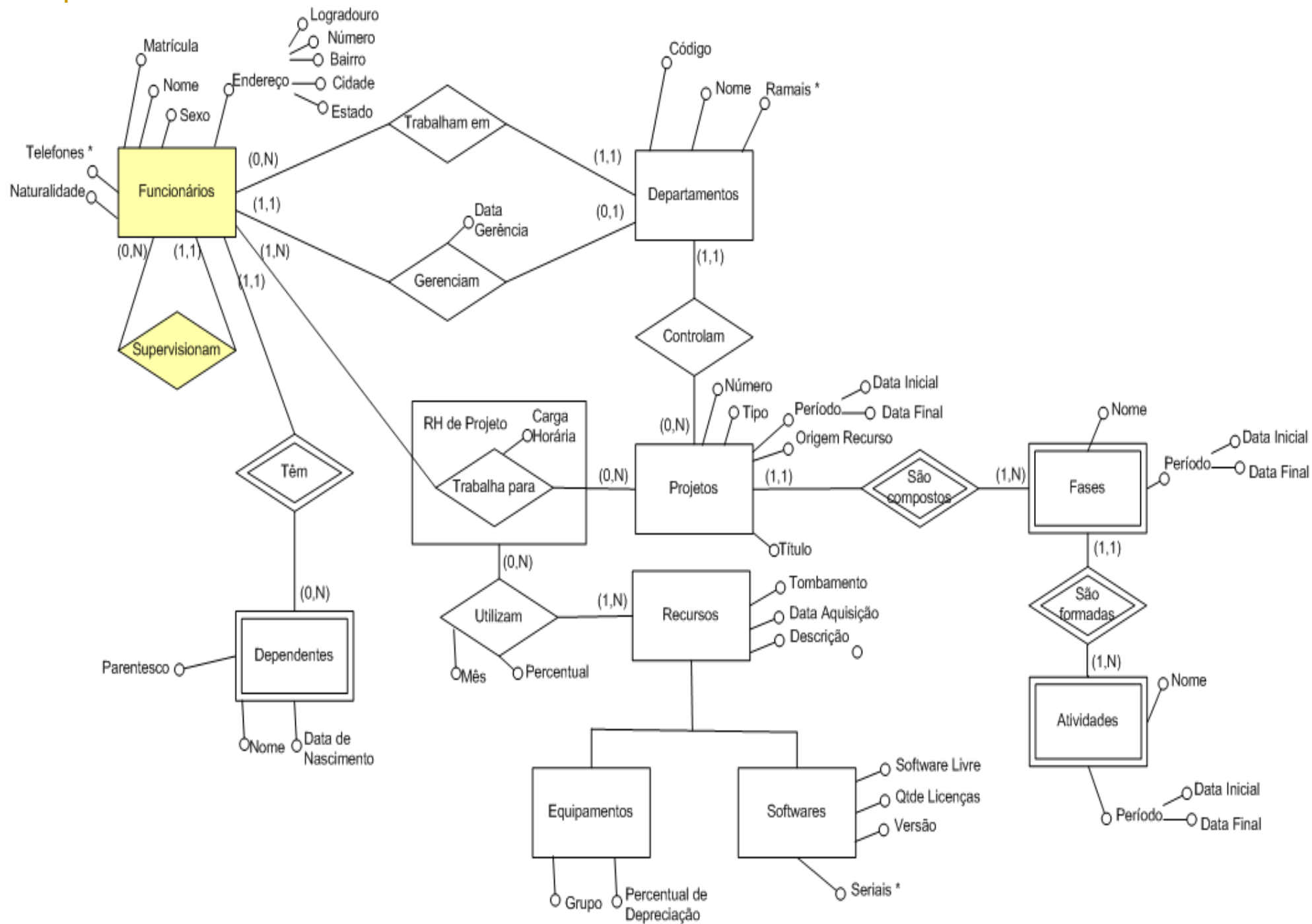
Mapeamento do MER para Relacional



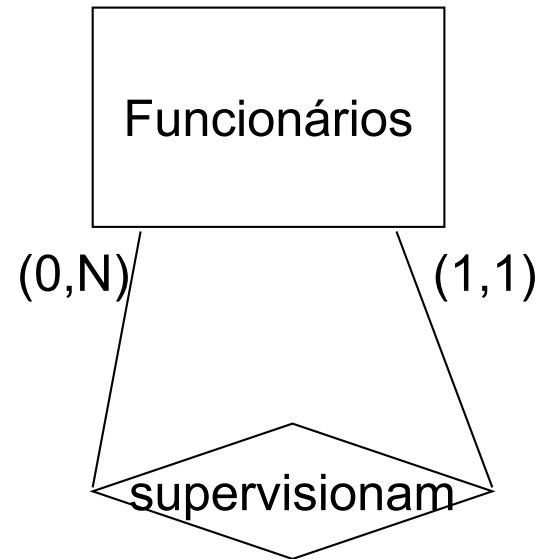
Mapeamento do MER para Relacional

■ PASSO 4.2 - Exemplo

- ❑ Funcionario (fun_matricula, fun_nome, fun_sexo, fun_naturalidade, fun_logradouro, fun_numero, fun_bairro, fun_cidade, fun_estado, **dep_codigo***)
- ❑ Departamento (dep_codigo, dep_nome, fun_matricula*)
- ❑ Projeto (pro_numero, pro_tipo, pro_titulo, pro_inicio, pro_final, pro_origemRecurso, **dep_codigo***)



Mapeamento do MER para Relacional

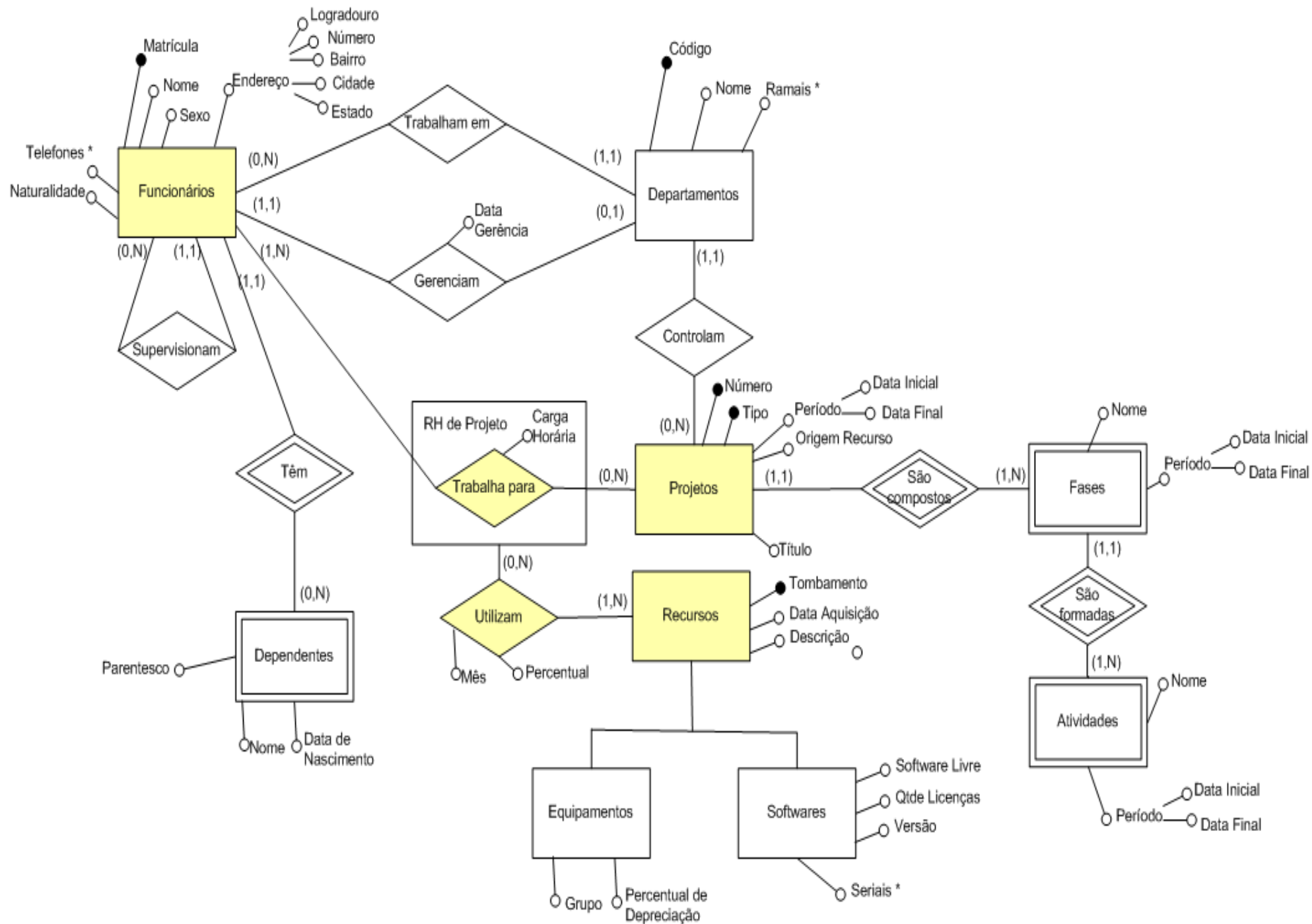


Mapeamento do MER para Relacional

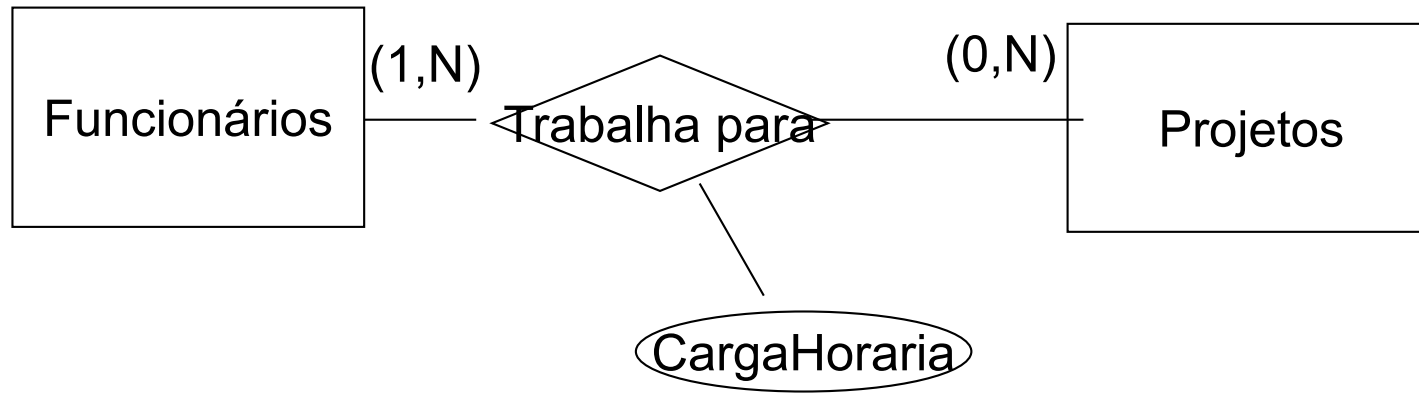
- PASSO 4.2 - Exemplo
 - Funcionario (fun_matricula, fun_nome, fun_sexo, fun_naturalidade, fun_logradouro, fun_numero, fun_bairro, fun_cidade, fun_estado, **dep_codigo***, **fun_supervisor***)
-

Mapeamento do MER para Relacional

- PASSO 4.3 - Muitos-para-muitos
 - ❑ Sendo um relacionamento muitos-para-muitos (N:M) entre dois conjuntos de entidades E1 e E2
 - ❑ Criar uma nova relação R1 correspondente ao relacionamento
 - ❑ A PK da relação R1 deve ser **composta** pela chave primária das relações de E1 e E2
 - ❑ Se o relacionamento possuir atributos, os mesmos devem ser inseridos na relação R1



Mapeamento do MER para Relacional



Mapeamento do MER para Relacional

- ❑ Projeto (pro_numero, pro_tipo, pro_titulo, pro_inicio, pro_final, pro_origemRecurso, **dep_codigo***)
- ❑ RH_projeto (fun_matricula*, pro_numero*, pro_tipo*, rhp_cargaHoraria)

Mapeamento do MER para Relacional

- PASSO 5 – Herança

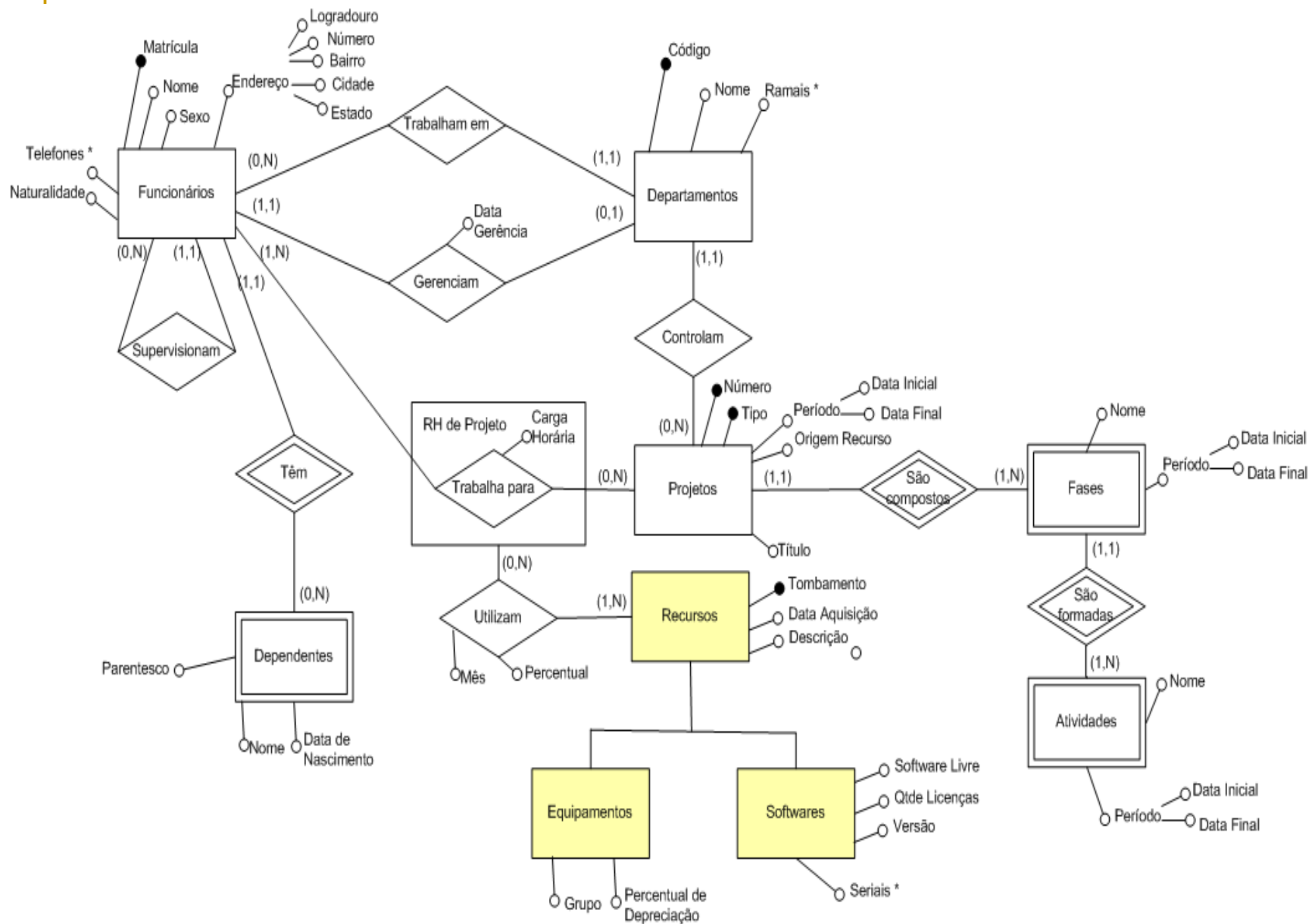
- Existem duas formas de se mapear um relacionamento de herança

- Forma 1: Relação única

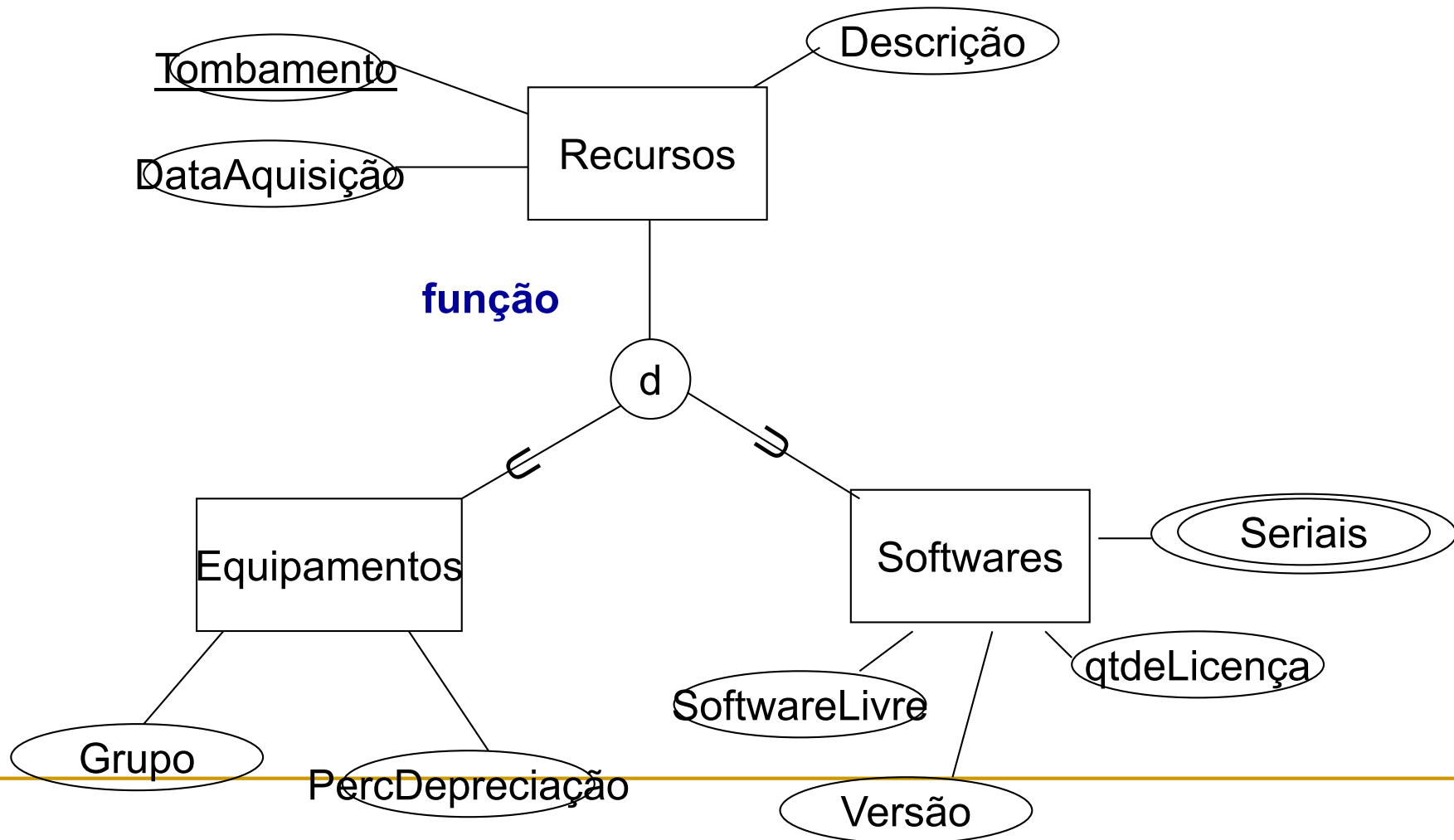
- Forma 2: Uma relação para a superclasse e uma para cada subclasse

Mapeamento do MER para Relacional

- PASSO 5 – Herança – Forma 1
 - Deve ser criada uma relação única R1 para a superclasse e para as subclasses
 - R1 deve conter os atributos simples da superclasse e os atributos de todas as subclasses
-



Mapeamento do MER para Relacional



Mapeamento do MER para Relacional

- PASSO 5 – Exemplo – Forma 1
 - Recurso (rec_tomabamento, rec_dataAquisicao, rec_descricao, rec_grupo, rec_percDepreciacao, rec_softwareLivre, rec_qtdeLicenca, rec-versao)
 - Software_Serial(rec_tombamento*, sof_serial)
-

Mapeamento do MER para Relacional

- PASSO 5 – Herança – Forma 2
 - Deve ser criada uma relação R1 para a superclasse e uma relação para cada subclasse
 - R1 deve conter todos os atributos simples da superclasse. Também deve ser incluído um atributo para indicar os possíveis tipos
 - As relações correspondentes às subclasses deve conter, como FK e PK, a chave primária de R1

Mapeamento do MER para Relacional

■ PASSO 5 – Exemplo – Forma 2

- ❑ Recurso (rec_tombamento#, rec_dataAquisicao, rec_descricao, rec_tipo)
- ❑ Equipamento (rec_tombamento#*, equ_grupo, que_percDepreciacao)
- ❑ Software (rec_tombamento#*, sof_livre, sof_qtdeLicenca, sof-versao)
- ❑ Software_Serial (rec_tombamento#*, sof_serial#)



PROJETO DE DADOS



Projeto de Dados

- Documento com a especificação dos detalhes de implementação dos dados
 - Descrição dos atributos
 - Tipo de dados e tamanho
 - Nulo?
 - Domínio
 - Chave primária, chave estrangeira e chave alternativa
-

Projeto de Dados

<Nome da Tabela>: <descrição da tabela>

Atributo	Descrição	Tipo	Tamanho	Domínio

PK = {atr1, atr2}

FK = {atr3} -> Relação Referida.atr3

AK = {atr4}

Projeto de Dados

Departamento: dados a respeito dos departamentos da empresa

Atributo	Descrição	Tipo	Tamanho	Nulo	Domínio
dep_codigo	Código do departamento incrementado automaticamente	Integer (AutoInc)	-	Não	-
dep_nome	Nome do departamento	Varchar	100	Não	-
ger_matricula	Matrícula do gerente do departamento	Integer	-	Não	-
ger_dataGerencia	Data em que o gerente assumiu o departamento	Date	-	Não	-

PK = {dep_codigo}

FK = {ger_matricula} -> Funcionario.fun_matricula

AK = {dep_nome}

Funcionario: dados a respeito dos funcionários da empresa

Atributo	Descrição	Tipo	Tamanho	Nulo	Domínio
fun_matricula	Matrícula do funcionário	Integer	-	Não	-
fun_nome	Nome do funcionário	Varchar	100	Não	-
fun_sexo	Sexo do funcionário	Smallint	-	Não	0 – Masculino 1 - Feminino
fun_naturalidade	Cidade em que o funcionário nasceu	Varchar	70	Não	-
fun_logradouro	Logradouro em que o funcionário mora	Varchar	100	Sim	-
fun_numero	Número da residência em que o funcionário mora	Varchar	10	Sim	-
fun_bairro	Bairro em que o funcionário mora	Varchar	50	Sim	-
fun_cidade	Cidade em que o funcionário mora	Varchar	100	Sim	-
fun_estado	Estado em que o funcionário mora	Varchar	70	Sim	-
dep_codigo	Código do departamento em que o funcionário trabalha	Integer	-	Não	-
fun_supervisor	Supervisor do funcionário	Integer	-	Sim	-

Continuação...

PK = {fun_matricula}

FK = {dep_codigo} - > Departamento.dep_codigo ;
 {fun_supervisor} - > Funcionario.fun_matricula

Projeto de Dados

■ Dicas de Projeto

- ❑ Diferença entre Char e Varchar
 - ❑ Char e Varchar representam conjunto de caracteres
 - ❑ Char é um conjunto de caracteres fixo
 - ❑ Varchar é um conjunto de caracteres variável
-

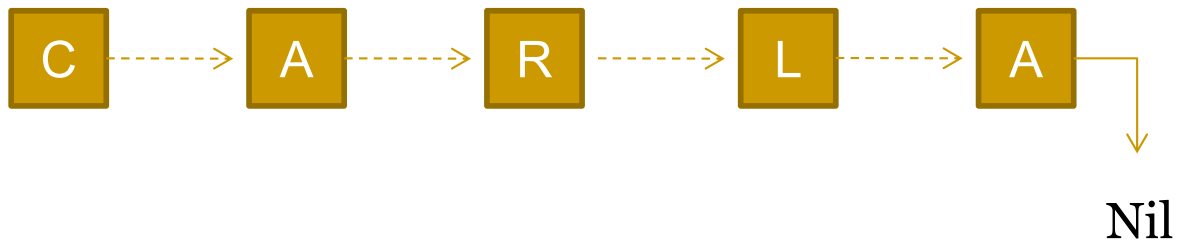
Projeto de Dados

- Dicas de Projeto

- Char(10)



- Varchar(10)



Projeto de Dados

- Dicas de Projeto

- Não definir um tamanho para o campo maior que o necessário

- Outros tipos de dados

- Integer, Smallint, Decimal, Numeric, Date, Time

Projeto de Dados

■ Dicas de Projeto

- ❑ Atributos com domínios limitados podem ser convertidos em tipo inteiro
- ❑ Projeto (pro_numero, **pro_tipo**, pro_titulo, pro_inicio, pro_final, pro_origemRecurso, dep_codigo*)

❑ Domínio

Valor	Descrição
0	Desenvolvimento de software
1	Desenvolvimento de hardware
2	Desenvolvimento de biotecnologia

Exercícios

- Fazer o mapeamento do MER para o Modelo Relacional das lista de exercício 1, 2 e 3.
 - Criar o projeto de dados de todos os exercícios anteriores
 - Entregar
-