

# Laboratório de Banco de Dados

Revisão

Modelo Relacional



Junho de 2025

# Agenda

- Origem do modelo, conceitos de relação, tuplas e atributos.
- Esquema do Banco de Dados.
- Restrições de integridade: chaves e integridade referencial.
- Diagramas de esquema de banco de dados relacional.
- Mapeamento de Diagrama Entidade-Relacionamento para Diagrama Relacional.

# Origens do Modelo Relacional

- O Modelo de Dados Relacional é fundamentado matematicamente pela teoria dos conjuntos e lógica de predicados de primeira ordem.
- Este modelo foi primeiramente proposto pelo Dr. Edgar F. Codd quando trabalhava na IBM research em 1970: "A Relational Model for Large Shared Data Banks", Communications of the ACM, June 1970.
- Este trabalho causou uma grande revolução na área de Banco de Dados e o Dr. Codd foi agraciado com o ACM Turing Award em 1981, pouco mais de uma década após a publicação do artigo.

# Conceitos de relação, tuplas e atributos

- O termo relação se refere a um conjunto de n-tuplas  $(A_1, A_2, \dots, A_n)$  advindas de um subconjunto do produto cartesiano dos domínios dos atributos  $\text{dom}(A_1) \times \text{dom}(A_2) \times \dots \times \text{dom}(A_n)$ . O domínio é conjunto de valores permitidos para cada atributo.
- Informalmente, uma relação pode ser entendida como uma tabela de valores, onde cada linha corresponde a uma tupla e cada coluna corresponde a um atributo.
- Fazendo um paralelo com o modelo Entidade-Relacionamento, cada tupla (ou linha da tabela) relaciona valores de atributos a fim de descrever uma entidade ou um relacionamento entre entidades do minimundo sendo modelado.

# Conceitos de relação, tuplas e atributos (Cont.)

- Cada coluna da tabela tem um cabeçalho que descreve parcialmente a semântica (o significado) do atributo.
- Formalmente, o cabeçalho da coluna é um metadado que descreve o nome do atributo.
- Os valores dos atributos são dados extraídos do minimundo para caracterizar uma entidade ou relacionamento entre entidades.
- Por exemplo, no contexto de uma Universidade, são atributos o RGA de um aluno, o nome de um professor, o nome de uma disciplina, etc.

# Conceitos de relação, tuplas e atributos (cont.)

- Cada relação (tabela) possui um nome único no banco de dados.
- Cada tupla (linha da tabela) relaciona atributos que representa uma entidade ou relacionamento do minimundo sendo modelado.
- Cada coluna especifica um atributo cujo domínio é composto por valores atômicos (indivisíveis).

nome da relação

atributos (ou colunas)

ID	nome	nome_dept	salário
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

tuplas (ou linhas)

# Conceitos de relação, tuplas e atributos (cont.)

- O valor especial null é membro de todo domínio e indica que o atributo não é conhecido ou não é aplicável a entidade em questão.
- O valor null deve ser evitado, pois causa complicações na definição de muitas operações com o banco de dados.

# Conceitos de relação, tuplas e atributos (cont.)

- No modelo relacional, a ordem das tuplas de uma relação é irrelevante. Por exemplo, as relações abaixo são consideradas iguais.

<i>ID</i>	<i>nome</i>	<i>nome_dept</i>	<i>salário</i>
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

<i>ID</i>	<i>nome</i>	<i>nome_dept</i>	<i>salário</i>
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000



# Esquema e Instância do Banco de Dados

# Instância e Esquema de Relação

- O esquema da relação consiste em sua descrição lógica:  $R = (A_1, A_2, \dots, A_n)$ .
- Por exemplo, a relação professor tem o seguinte esquema:  
professor = (ID, nome, nome\_dept, salário)
- Uma instância da relação  $R$  consiste no conjunto de tuplas  $r(R)$  correntemente armazenadas na tabela. Uma tupla específica de  $r(R)$  pode ser referida por  $t_n$ . Por exemplo, na tabela professor, a tupla  $t_8$  corresponde ao professor Califieri.

*professor*

ID	nome	nome_dept	salário
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

# Instância e esquema do Banco de Dados

- O esquema do banco de dados consiste em seu design lógico, ao passo que uma instância do banco de dados consiste no conteúdo do banco de dados em um determinado instante no tempo.
- Mais precisamente, um Banco de Dados relacional consiste em um conjunto de tabelas (relações) relacionadas. O esquema do Banco de Dados define sua estrutura lógica, o que inclui o esquema das relações e demais restrições do modelo (veremos em breve). Uma instância do Banco de Dados corresponde aos dados armazenados em todas as tabelas em um dado instante no tempo.
- O esquema do BD muda pouco, ao passo que a instância muda sempre que ocorre alguma inserção, atualização ou deleção de dados.

# Restrições do Modelo Relacional

Chave e Integridade Referencial

# Principais de Restrições do Modelo Relacional

- Existem basicamente três tipos principais de restrições no Modelo Relacional:
  - Restrição de domínio
  - Restrição de Chave primária
  - Restrição de Integridade Referencial
- Essas restrições são condições que devem ser mantidas em todas as instâncias válidas do Banco de Dados, e são derivadas da semântica do mundo sendo modelado.
- A seguir detalharemos essas restrições...

# Restrição de Domínio

- Impõe que cada valor de atributo deve pertencer a um domínio (ou pode ser null, se permitido).
- No exemplo ao lado, o domínio do atributo ID poderia ser os inteiros positivos, o domínio dos atributos nome e nome\_dept poderia ser o das string de até 50 caracteres e o domínio do atributo salário poderia ser os números inteiros entre 30000 e 200000.
- Observe que o domínio é extraído de restrições do mundo real.

*professor*

<i>ID</i>	<i>nome</i>	<i>nome_dept</i>	<i>salário</i>
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

# Restrição de Chave

- Dado  $R = (A_1, A_2, \dots, A_n)$  e  $K \subseteq R$ , dizemos que  $K$  é uma superchave de  $R$  se cada combinação de valores  $K$  identifica uma única tupla em qualquer instância  $r(R)$ .
- Em outras palavras, se  $K$  é uma superchave, nunca haverá um estado consistente do Bando de Dados com duas ou mais tuplas tendo exatamente o mesmo valor para todos os atributos de  $K$ .
- Por exemplo,  $\{ID\}$  e  $\{ID, nome\}$  são ambas superchaves da relação professor.

*professor*

<i>ID</i>	<i>nome</i>	<i>nome_dept</i>	<i>salário</i>
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

## Restrição de Chave (cont.)

- Uma superchave  $K$  é uma chave candidata se o conjunto  $K$  é mínimo, isto é, se a remoção de qualquer atributo de  $K$  implicar na perda da propriedade de identificação unívoca de tuplas.
  - Exemplo:  $\{ID\}$  é uma chave candidata para a relação professor, pois a chave não pode ser vazia.
- É possível que haja mais de uma combinação de atributos que sirva como chave candidata. Neste caso, uma delas deve ser selecionada para ser a chave primária da relação.
- Na especificação do esquema de uma relação, os atributos da chave primária normalmente aparecem primeiro e devem ser sublinhados.
  - professor = (ID, nome, nome\_dept, salário)



# Restrição de Integridade Referencial

- As restrições de integridade referencial, envolvem duas relações.
- A mais importante é a chamada restrição de chave estrangeira.
- **Definição:** Um subconjunto de atributos  $A$  da relação  $R_1$  formam uma chave estrangeira para a relação  $R_2$  se, em qualquer estado consistente do banco de dados, o valor de  $A$  em cada tupla de  $r(R_1)$  for igual ao valor da chave primária  $B$  de alguma tupla de  $r(R_2)$ . Nesse caso, a relação  $R_1$  é chamada de referenciadora e  $R_2$  é chamada de relação referenciada.
- Observe que em uma restrição de chave estrangeira, o(s) atributo(s) referenciado(s) deve(m) ser a chave primária da relação referenciada.

# Restrição de Integridade Referencial

*professor*

<u>ID</u>	nome	nome_dept	salário
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

Relação referenciadora  $R_1$

*departamento*

<u>nome_dept</u>	prédio	despesa
Biology	Watson	90000
Comp. Sci.	Taylor	100000
Elec. Eng.	Taylor	85000
Finance	Painter	120000
History	Painter	50000
Music	Packard	80000
Physics	Watson	70000

Relação referenciada  $R_2$

# Diagramas Relacionais

Notação

# Diagramas de Banco de Dados Relacional

- O esquema de um banco de dados relacional pode ser representado graficamente por meio de um Diagramas de Esquema Relacional.
  - Cada relação é descrita por seus atributos listados em uma linha, lado a lado. O nome da relação fica no topo.
  - Os atributos que formam a chave primária devem ser sublinhados e normalmente aparecem primeiro.
  - As chaves estrangeiras aparecem como setas partindo da relação de referenciadora para os atributos da chave primária da relação referenciada.

# Exemplo de notação

## Departamento

<u>Nome_dept</u>	Prédio	Orçamento
------------------	--------	-----------



Indicação de chave  
estrangeira

## Professor

<u>ID</u>	Nome	Nome_dept	Salário
-----------	------	-----------	---------

# Design Relacional Usando Diagramas ER

# Reduzindo Diagramas ER a Esquemas Relacionais

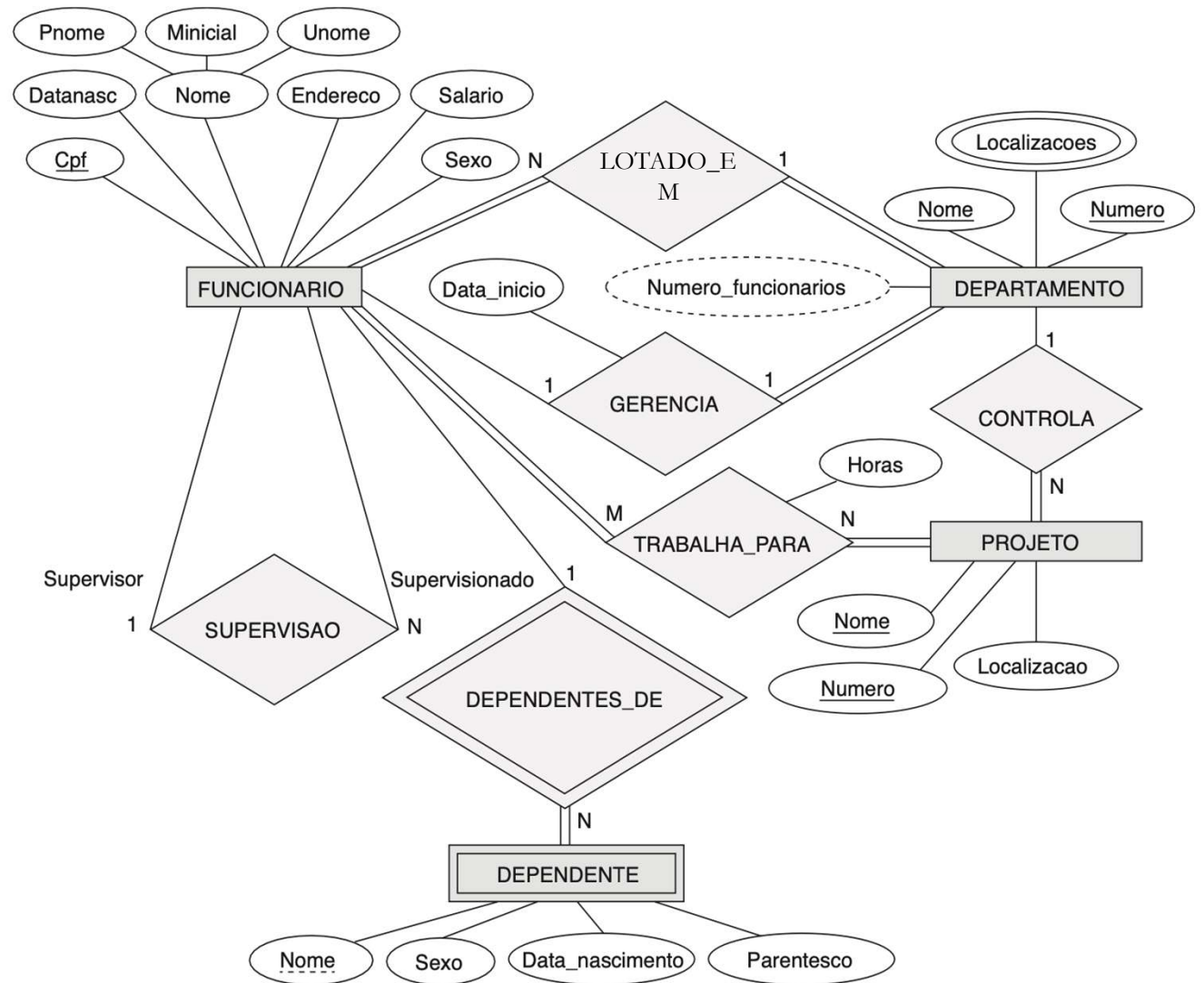
- Tanto o modelo ER quanto o modelo Relacional são representações lógicas de um minimundo.
- Como os dois modelos empregam princípios de design semelhantes, podemos converter um projeto ER em um projeto relacional.
- Para cada tipo de entidade e para cada tipo de relacionamento definido no projeto ER há uma relação única à qual atribuímos o nome do tipo de entidades ou tipo de relacionamentos correspondente.
- As regras de conversão serão discutidas à seguir.

# Reduzindo Diagramas ER a Esquemas Relacionais

- Para realizar o mapeamento completo realizamos os seguintes passos:
  - Passo 1: Mapeamento de Tipos de Entidades Fortes.
  - Passo 2: Mapeamento de Tipos de Entidades Fracas.
  - Passo 3: Mapeamento de Atributos Multivalorados.
  - Passo 4: Mapeamento de Tipos de Relacionamentos Binários 1:1.
  - Passo 5: Mapeamento de Tipos de Relacionamentos Binários 1:N ou N:1.
  - Passo 6: Mapeamento de Tipos de Relacionamentos Binários M:N.
  - Passo 7: Mapeamento de Tipos de Relacionamentos n-ários.
  - Passo 8: Mapeamento de Especializações.
  - Passo 9: Mapeamento de Agregações.



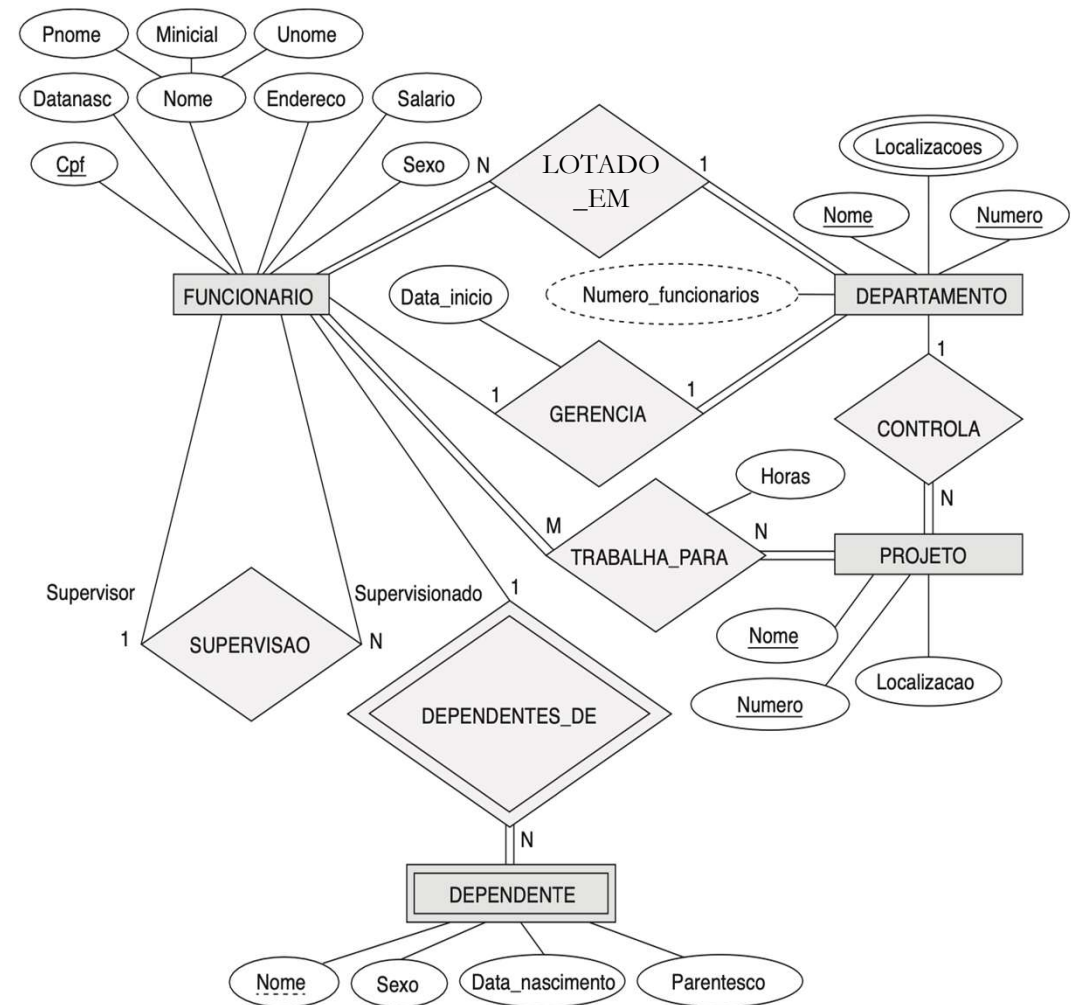
# Banco de Dados EMPRESA



# Passo 1: Mapeamento de Tipos de Entidades Fortes

1. Para cada tipo de entidades forte  $E$ , crie uma tabela  $R$  que inclua todos os atributos simples ou componentes simples de atributos compostos de  $E$ .
  2. A chave primária da tabela  $R$  é a mesma chave do tipo de entidades  $E$ . Se a chave de  $E$  for composta, os atributos simples que a compõe formarão juntos a chave primária da tabela  $R$ .
- A seguir veremos um exemplo.

# Mapeamento de Tipos de Entidades Fortes



# Mapeamento de Tipos de Entidades Fortes

## FUNCIONARIO

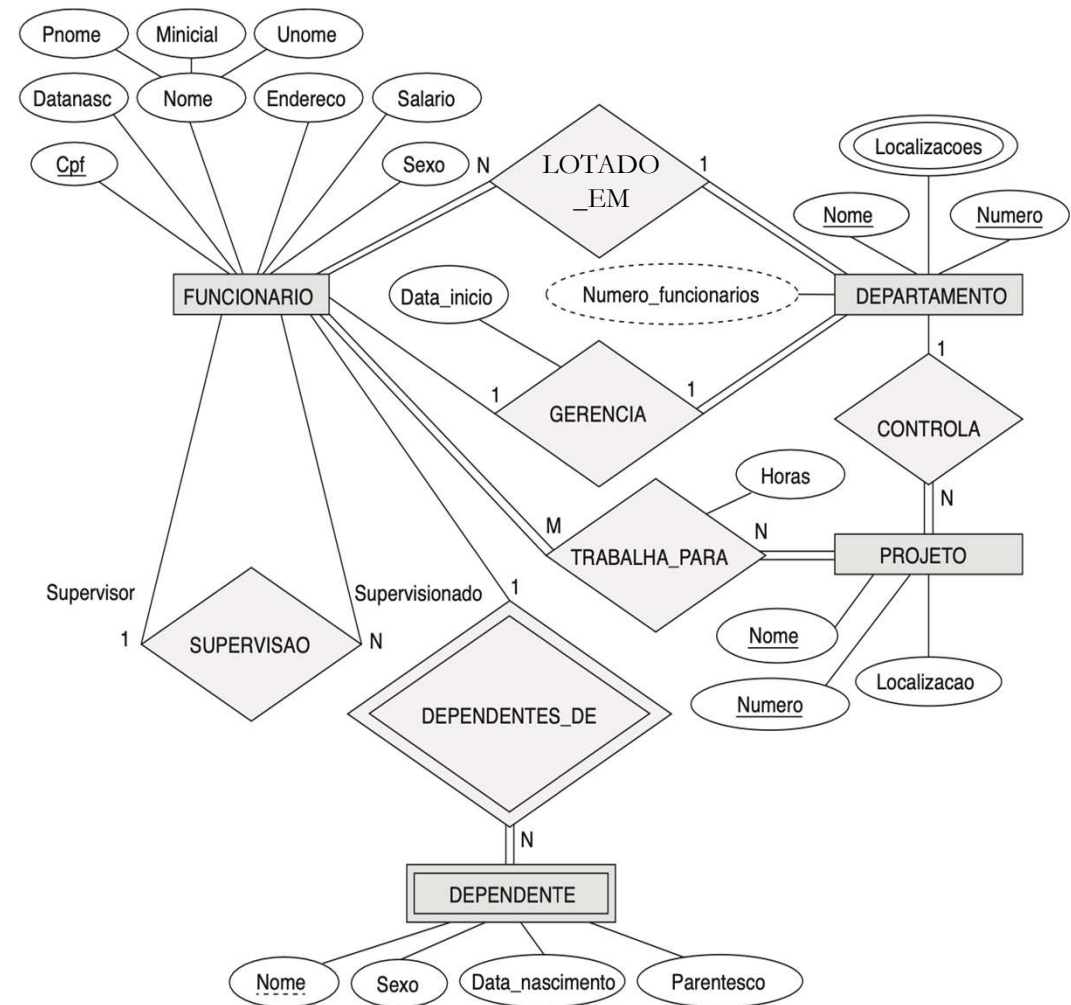
Pnome	Minicial	Unome	<u>Cpf</u>	Datanasc	Endereco	Sexo	Salario
-------	----------	-------	------------	----------	----------	------	---------

## DEPARTAMENTO

Dnome	<u>Dnumero</u>
-------	----------------

## PROJETO

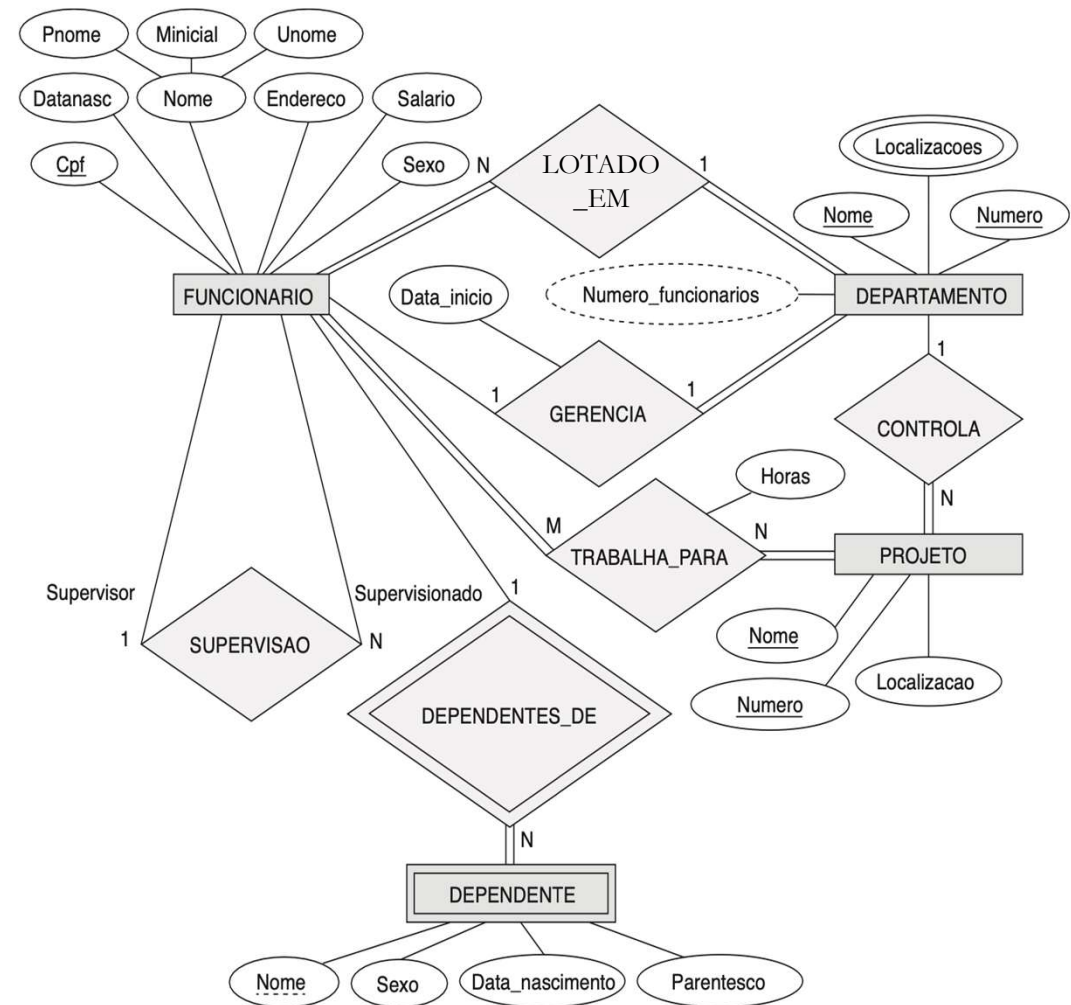
Projnome	<u>Projnumero</u>	Projlocal
----------	-------------------	-----------



## Passo 2: Mapeamento de Tipo de Entidade Fraca

1. Para cada tipo de entidades fraca  $F$ , crie uma tabela  $R$  e inclua todos os atributos simples (e componentes simples de atributos compostos) de  $F$  como atributos de  $R$ .
  2. Inclua como chave estrangeira de  $R$  a chave do(s) tipo(s) de entidade(s) identificadora(s) de  $F$ .
  3. A chave primária de  $R$  é a combinação das chaves do(s) tipo(s) de entidade(s) identificadora(s) com a chave parcial de  $F$  (se existir).
- A seguir um exemplo.

# Mapeamento de Tipos de Entidades Fracas



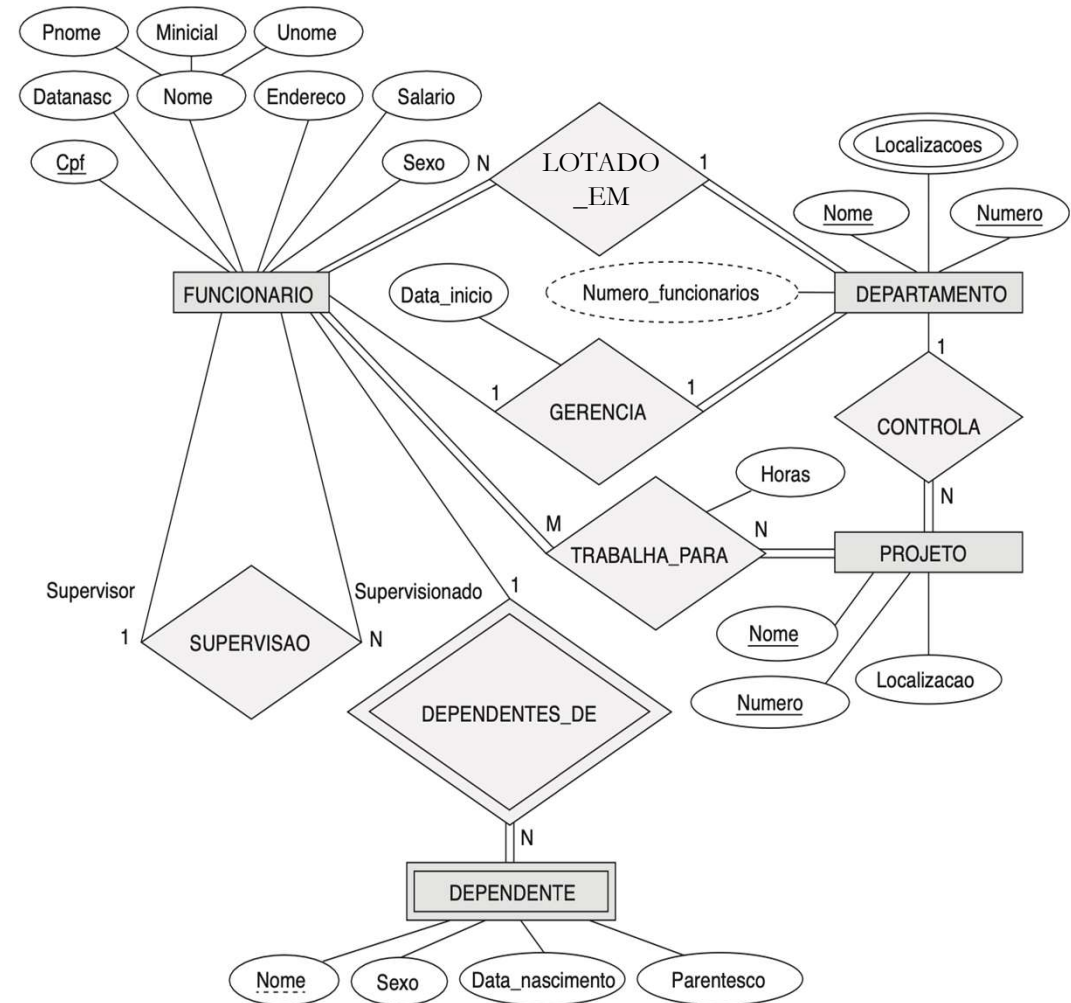
# Mapeamento de Tipos de Entidades Fracas

## DEPENDENTE

<u>Fcpf</u>	<u>Nome_dependente</u>	Sexo	Datanasc	Parentesco
-------------	------------------------	------	----------	------------

## FUNCIONARIO

Pnome	Minicial	Unome	<u>Cpf</u>	Datanasc	Endereco	Sexo	Salario
-------	----------	-------	------------	----------	----------	------	---------



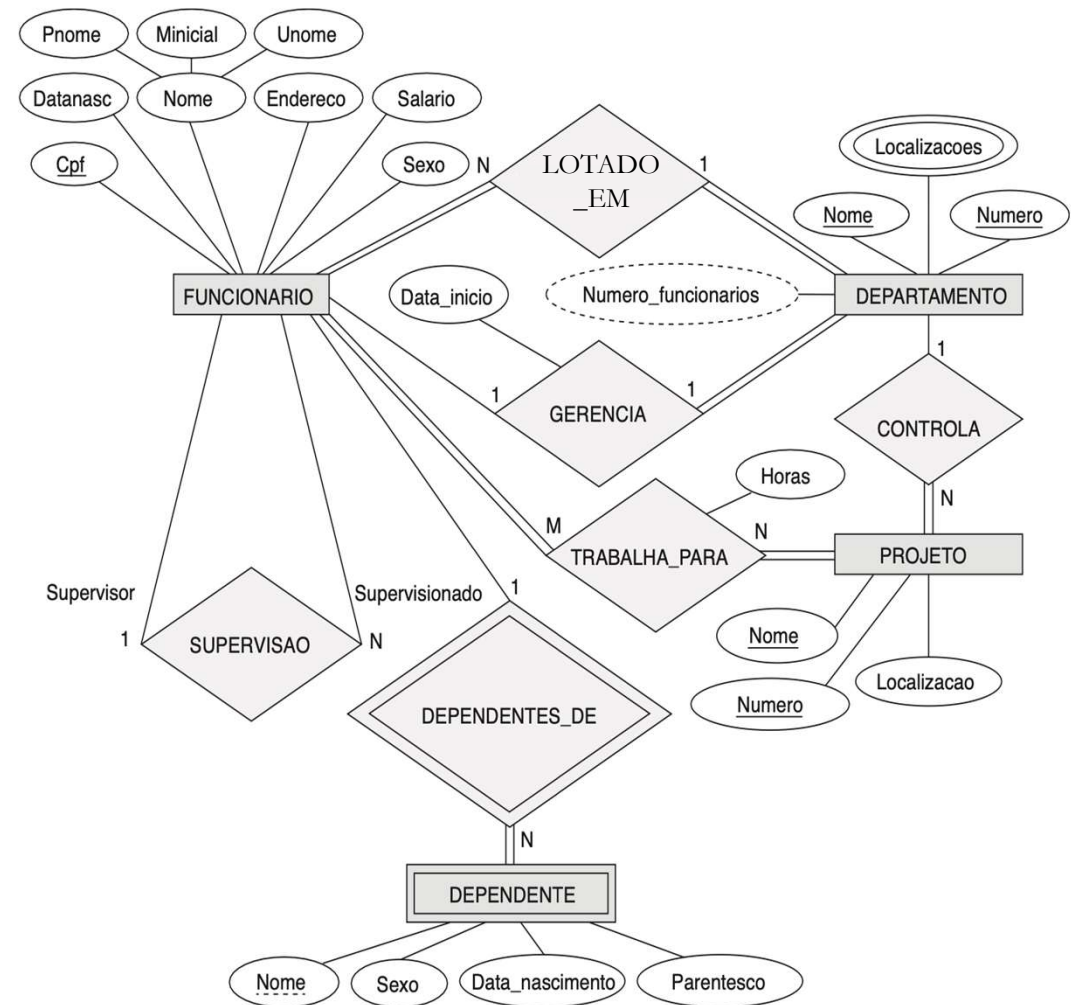


## Passo 3: Mapeamento de Atributos Multivalorados

1. Um atributo multivalorado  $M$  de um tipo de entidades  $E$  é representado por uma nova tabela  $R$ .
  2. A tabela  $R$  deve conter os atributos correspondente a chave de  $E$  e um atributo para representar o atributo multivalorado  $M$ .
  3. A chave primária de  $R$  é a combinação de seus atributos.
  4. Os atributos que foram incluídos por serem chave de  $E$  passam a ser chave estrangeira de  $R$  para a relação associada a  $E$ .
- A seguir veremos um exemplo.



# Mapeamento de Atributos Multivalorados



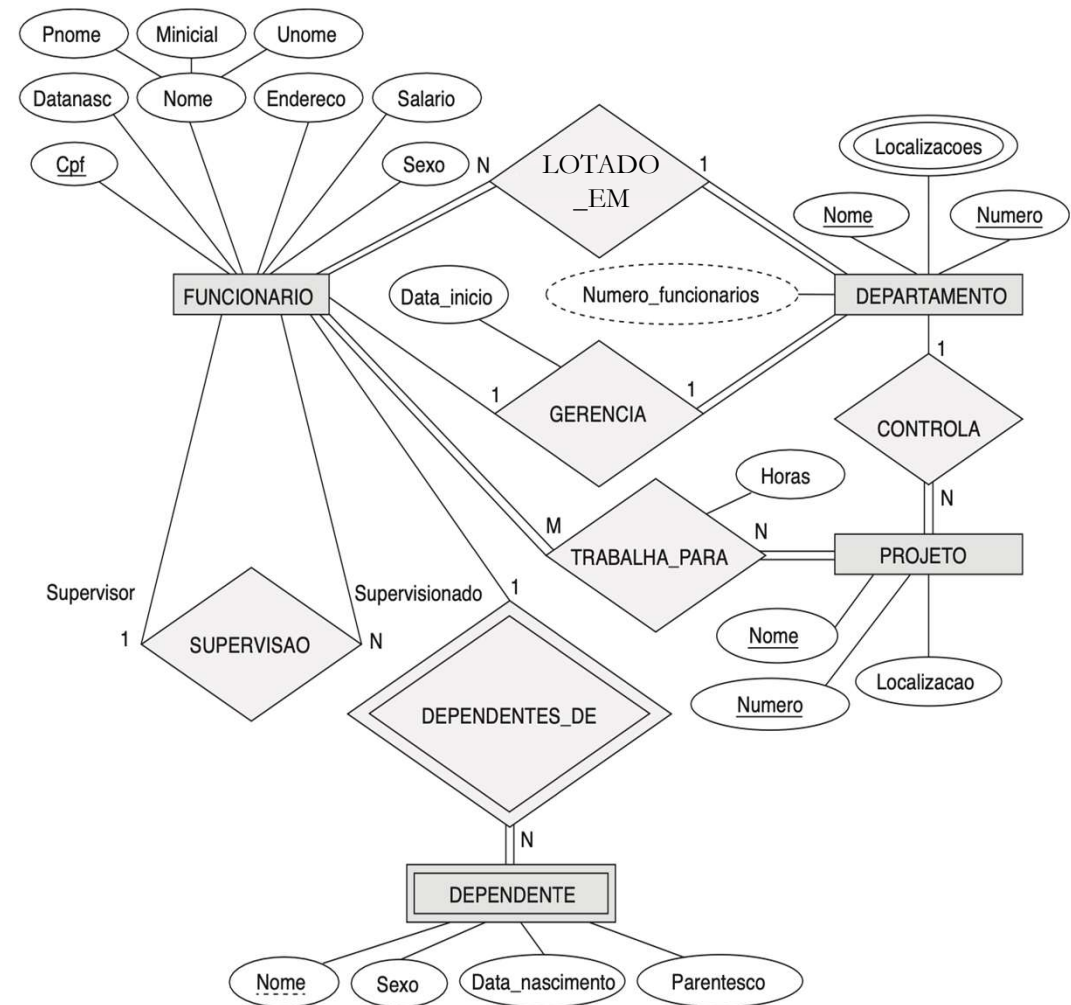
# Mapeamento de Atributos Multivalorados

## LOCALIZACAO\_DEP

<u>Dnumero</u>	<u>Dlocal</u>
----------------	---------------

## DEPARTAMENTO

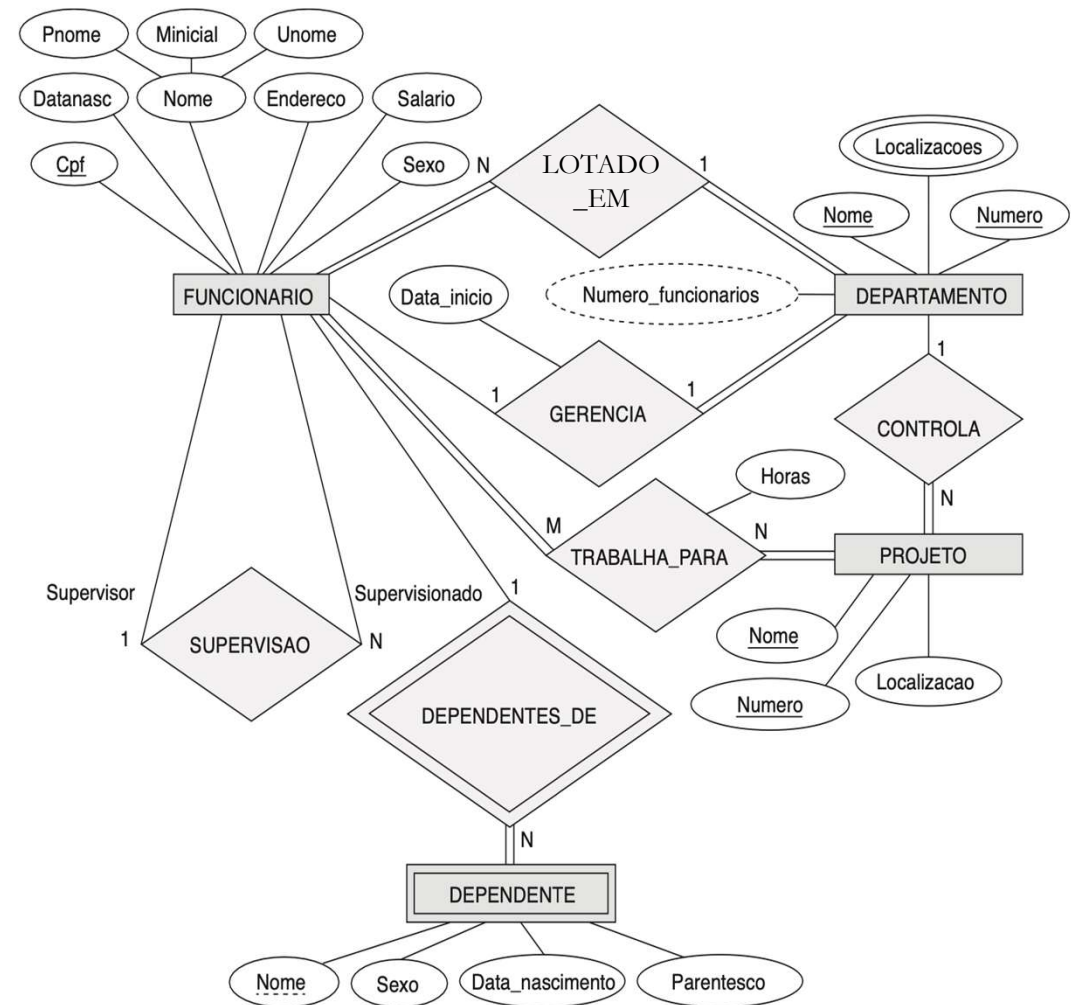
Dnome	<u>Dnumero</u>
-------	----------------



## Passo 4: Mapeamento de Tipos de Relacionamento 1:1

1. Para cada tipo de relacionamentos binário  $R$  1:1, identifique os conjuntos de entidades participantes  $S$  e  $T$ .
2. Primeiramente, crie uma tabela correspondente a  $R$  formada com os atributos das chaves primárias de  $S$  e  $T$  e demais atributos de  $R$  (caso exista). Como a cardinalidade é 1:1, a chave primária do relacionamento pode ser escolhida para ser a chave primária de qualquer uma das entidades participantes ( $S$  ou  $T$ ). Além disso, as chaves primárias de  $S$  e  $T$  são chaves estrangeiras de  $R$  para as relações correspondentes a  $S$  e  $T$ .
3. Em seguida, devemos analisar a possibilidade de eliminar redundância:
  - ❑ Se houver cobertura total de um dos conjuntos de entidades participantes, por exemplo  $S$ , podemos juntar a tabela  $R$  com a tabela oriunda de  $S$  para evitar redundância. Nesse caso, a chave primária de  $T$  entraria como chave estrangeira na tabela oriunda de  $S$ .
  - ❑ Se ambas as participações forem totais podemos juntar as tabelas  $S$  e  $T$  e os atributos do relacionamento (se houver) em uma única tabela. Nesse caso, como a cardinalidade é 1:1, a chave primária da nova tabela pode ser a chave de  $S$  ou a chave de  $T$ .

# Mapeamento de Tipo de Relacionamento 1:1



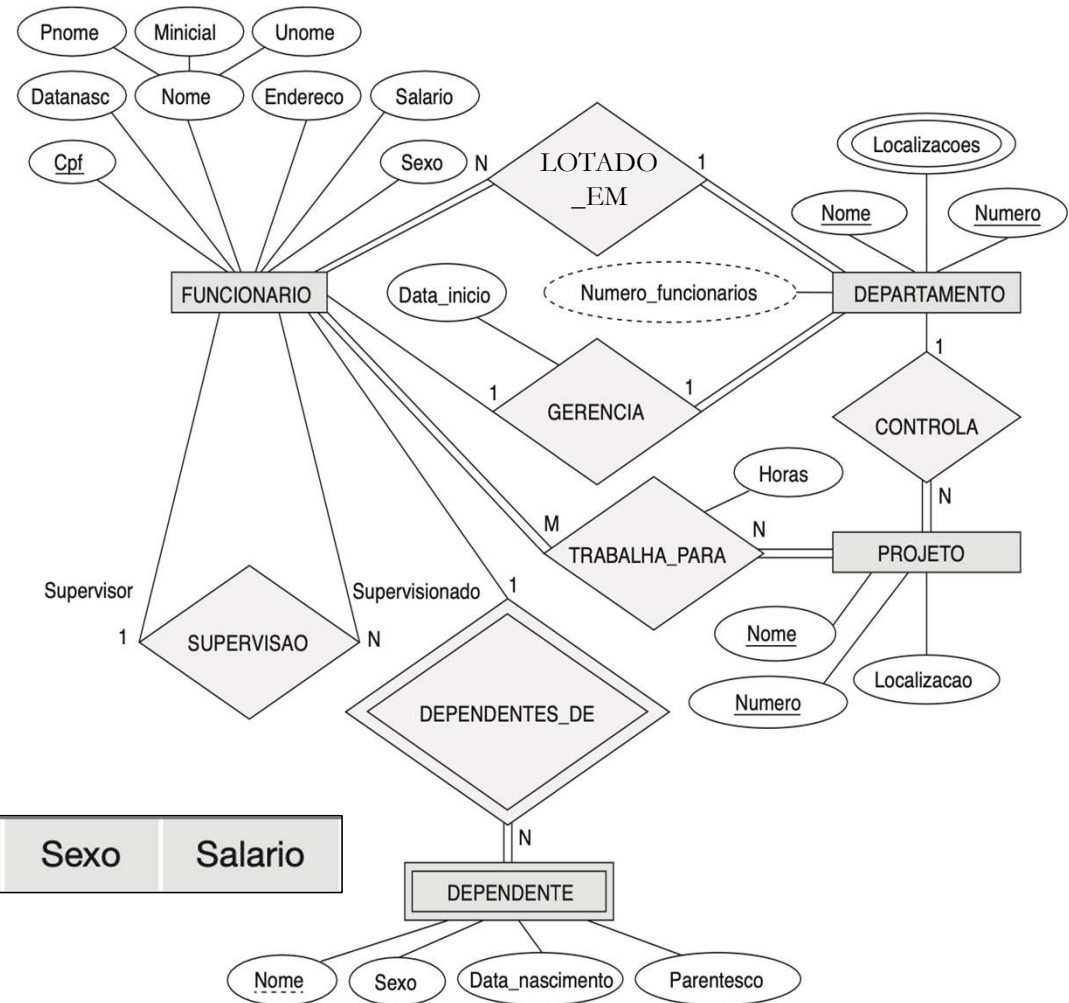
# Mapeamento de Tipo de Relacionamento 1:1

## DEPARTAMENTO

Dnome	<u>Dnumero</u>	Cpf_gerente	Data_inicio_gerente
-------	----------------	-------------	---------------------

## FUNCIONARIO

Pnome	Minicial	Unome	<u>Cpf</u>	Datanasc	Endereco	Sexo	Salario
-------	----------	-------	------------	----------	----------	------	---------

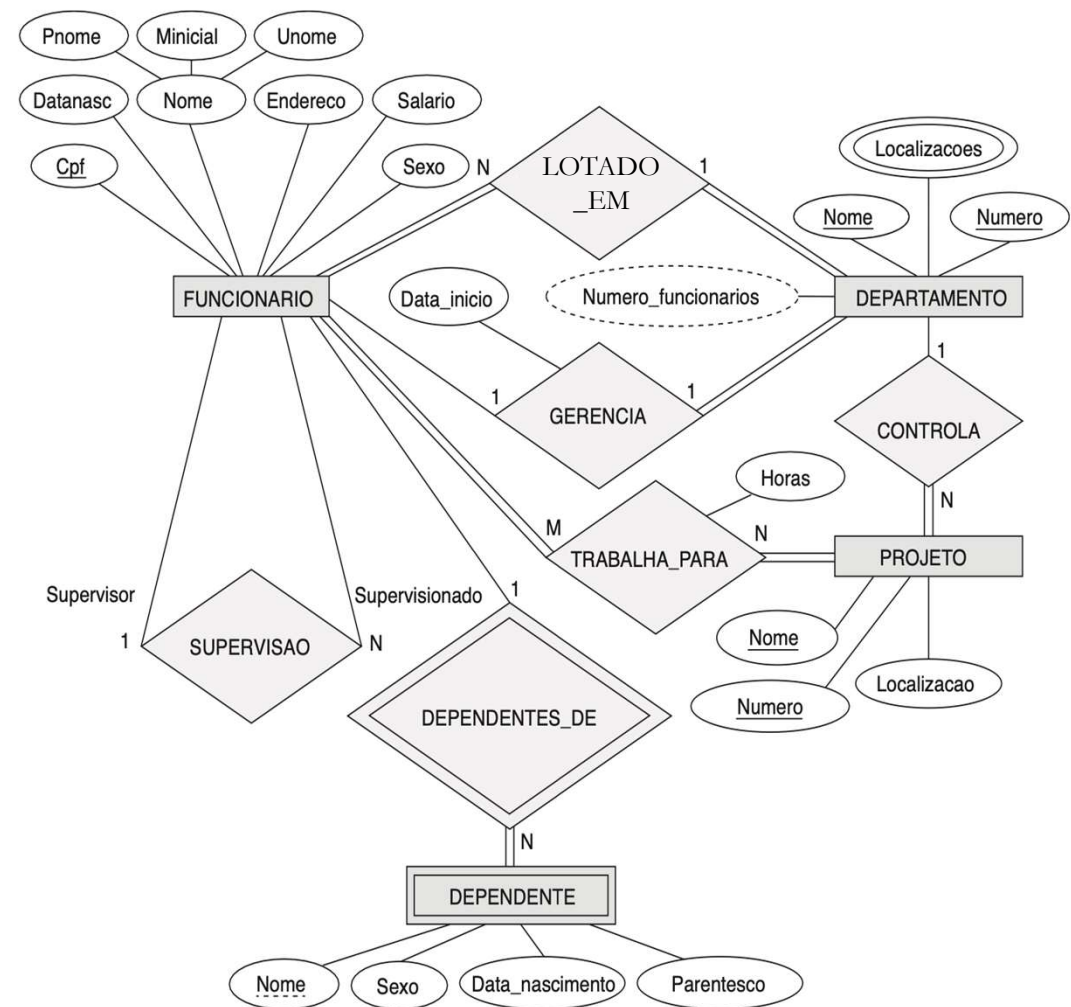


## Passo 5: Mapeamento de Tipos de Relacionamentos binários 1:N (ou N:1)

1. Para cada tipo de relacionamento binário  $R$  com cardinalidade 1:N, identifique os conjuntos de entidades participantes  $S$  e  $T$ .
2. Primeiramente, crie uma tabela correspondente a  $R$  formada com os atributos das chaves primárias de  $S$  e  $T$  e demais atributos de  $R$  (caso exista). Como a cardinalidade é 1:N, a chave primária do relacionamento deve ser escolhida para ser a chave do tipo de entidades que participa uma única vez do relacionamento (o lado N na notação padrão que estamos usando).
3. As chaves de  $S$  e  $T$  são chaves estrangeiras de  $R$  para as tabelas oriundas de  $S$  e  $T$ .
4. Em seguida, devemos analisar a possibilidade de eliminar redundância:
  - ❑ Se o conjunto de entidades que participa uma única vez do relacionamento, por exemplo  $S$ , tiver participação total, então devemos considerar a possibilidade de juntar  $R$  e  $S$ .
  - ❑ Se a cobertura do conjunto de entidades  $S$  for parcial, então juntar as relações não é um erro, mas exige a possibilidade de valores null na chave estrangeira.



# Mapeamento de Tipo de Relacionamento 1:N (ou N:1)



# Mapeamento de Tipo de Relacionamento 1:N (ou N:1)

## PROJETO

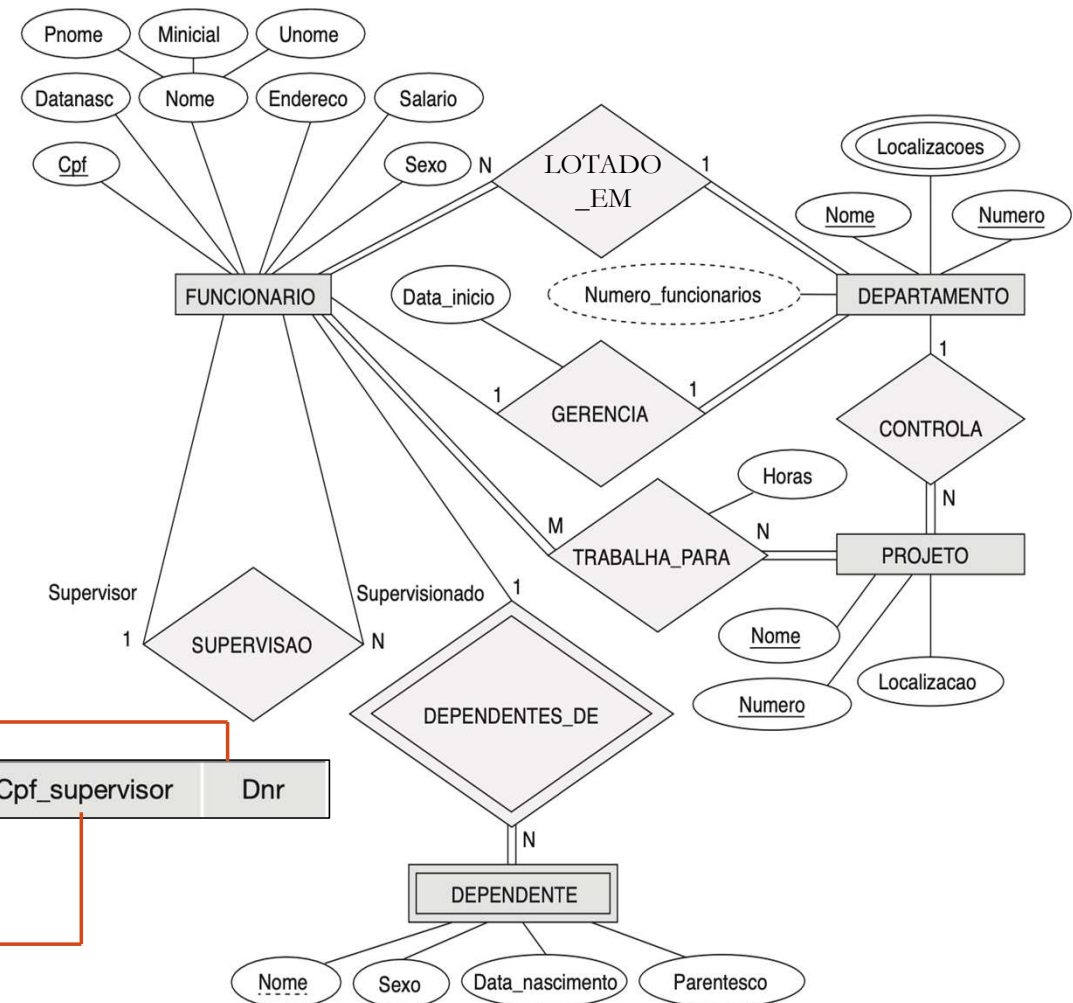
Projnome	<u>Projnumero</u>	<u>Projlocal</u>	Dnum
----------	-------------------	------------------	------

## DEPARTAMENTO

Dnome	<u>Dnumero</u>	Cpf_gerente	Data_inicio_gerente
-------	----------------	-------------	---------------------

## FUNCIONARIO

Pnome	Minicial	Unome	<u>Cpf</u>	Datanasc	Endereco	Sexo	Salario	Cpf_supervisor	Dnr
-------	----------	-------	------------	----------	----------	------	---------	----------------	-----

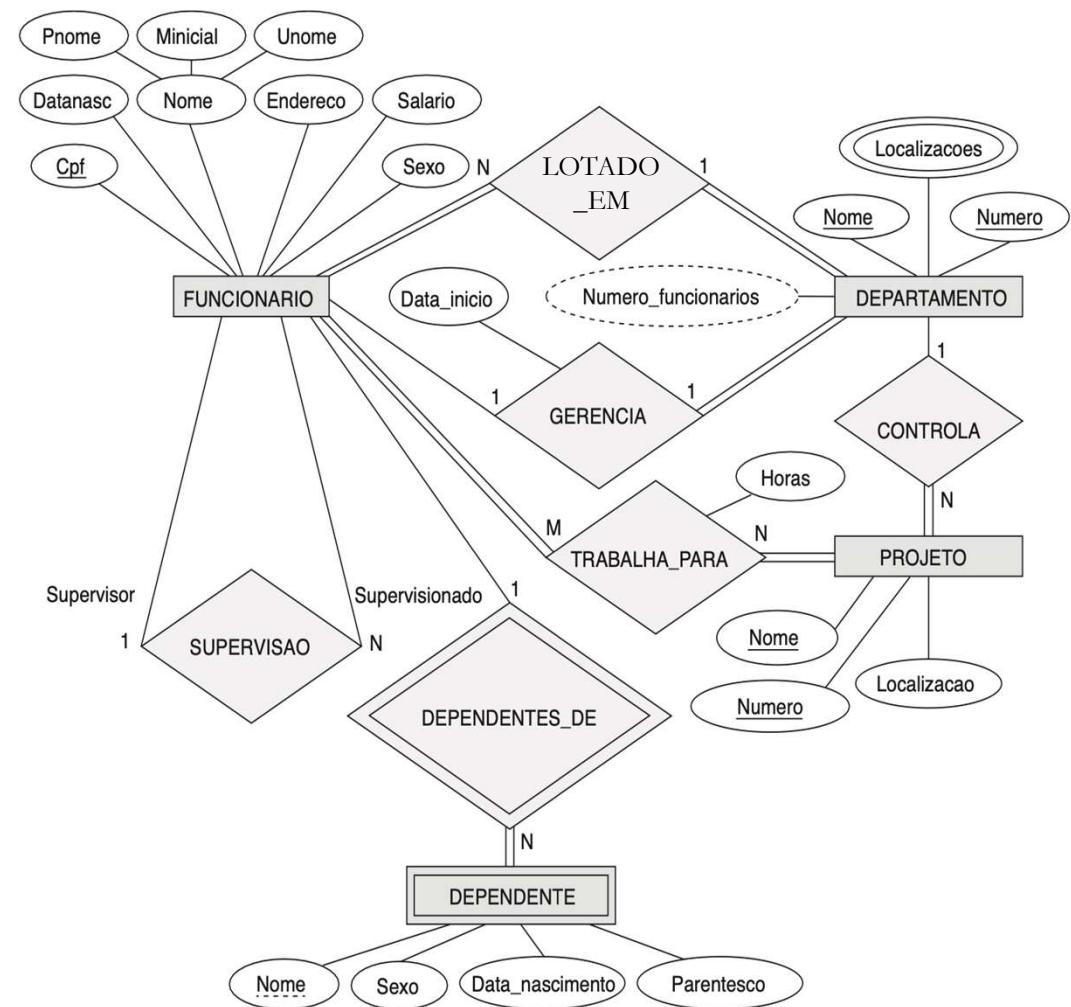




## Passo 6: Mapeamento de Tipos de Relacionamentos binários N:M

- Para cada conjunto de relacionamentos binário  $R$  com cardinalidade N:M, identifique os conjuntos de entidades participantes  $S$  e  $T$ .
- Crie uma tabela correspondente a  $R$  formada com os atributos das chaves primárias de  $S$  e  $T$  e demais atributos de  $R$  (caso exista).
- A chave primária da nova tabela é a chave primária do relacionamento. Como a cardinalidade é N:M, a chave primária da tabela  $R$  deve ser escolhida para ser a combinação das chaves primárias dos conjuntos de entidades participantes do relacionamento  $R$ . Além disso, as chaves primárias de  $S$  e  $T$  são chaves estrangeiras de  $R$  para as tabelas oriundas de  $S$  e  $T$ .

# Mapeamento de Tipo de Relacionamento 1:N (ou N:1)



# Mapeamento de Tipo de Relacionamento 1:N (ou N:1)

## PROJETO

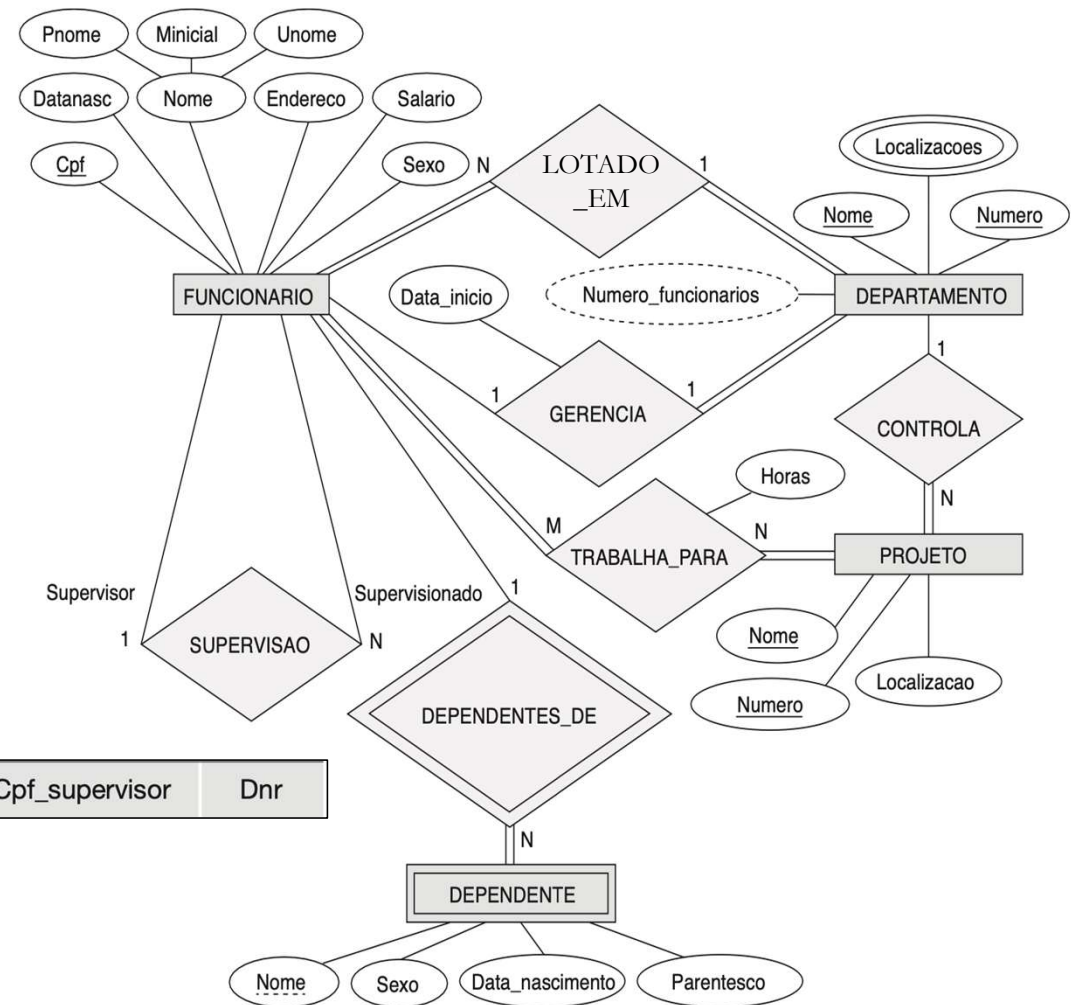
Projnome	<u>Projnumero</u>	<u>Projlocal</u>	Dnum
----------	-------------------	------------------	------

## TRABALHA\_EM

<u>Fcpf</u>	<u>Pnr</u>	Horas
-------------	------------	-------

## FUNCIONARIO

Pnome	Minicial	Unome	<u>Cpf</u>	Datanasc	Endereco	Sexo	Salario	Cpf_supervisor	Dnr
-------	----------	-------	------------	----------	----------	------	---------	----------------	-----



## Passo 7: Mapeamento de Tipos de Relacionamentos n-ários

1. O mapeamento de tipos de relacionamentos n-ários segue a mesma lógica do mapeamento de tipos de relacionamentos N:M.
2. Criamos uma nova tabela na qual incluímos as chaves dos tipos de entidades participantes. A chave primária da tabela é a chave tipo de relacionamento.
3. Além disso, as chaves primárias incluídas na tabela que representa o relacionamento n-ário são também chaves estrangeiras para os tipos de entidades participantes.

## Passo 7: Mapeamento de Tipos de Relacionamentos n-ários

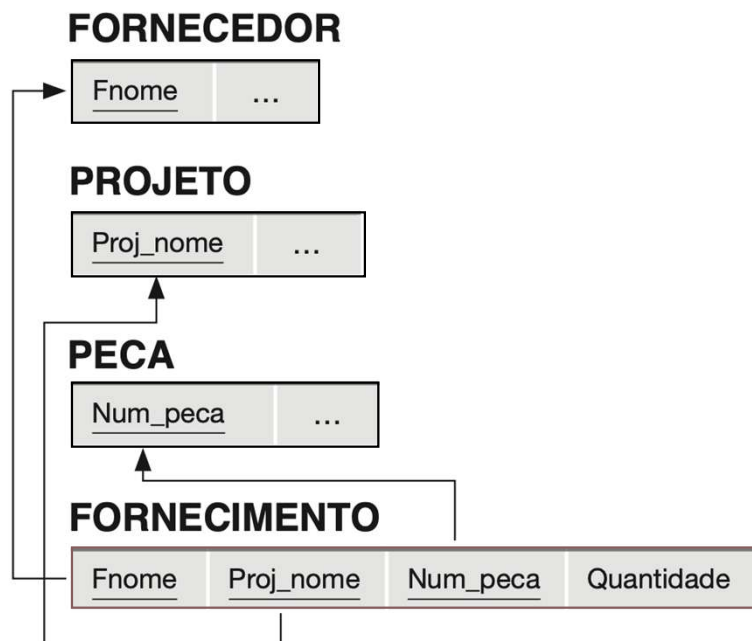


Diagrama Relacional

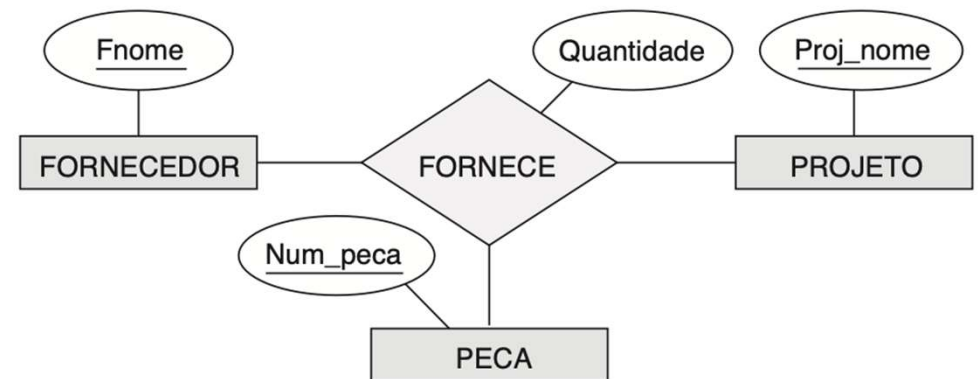
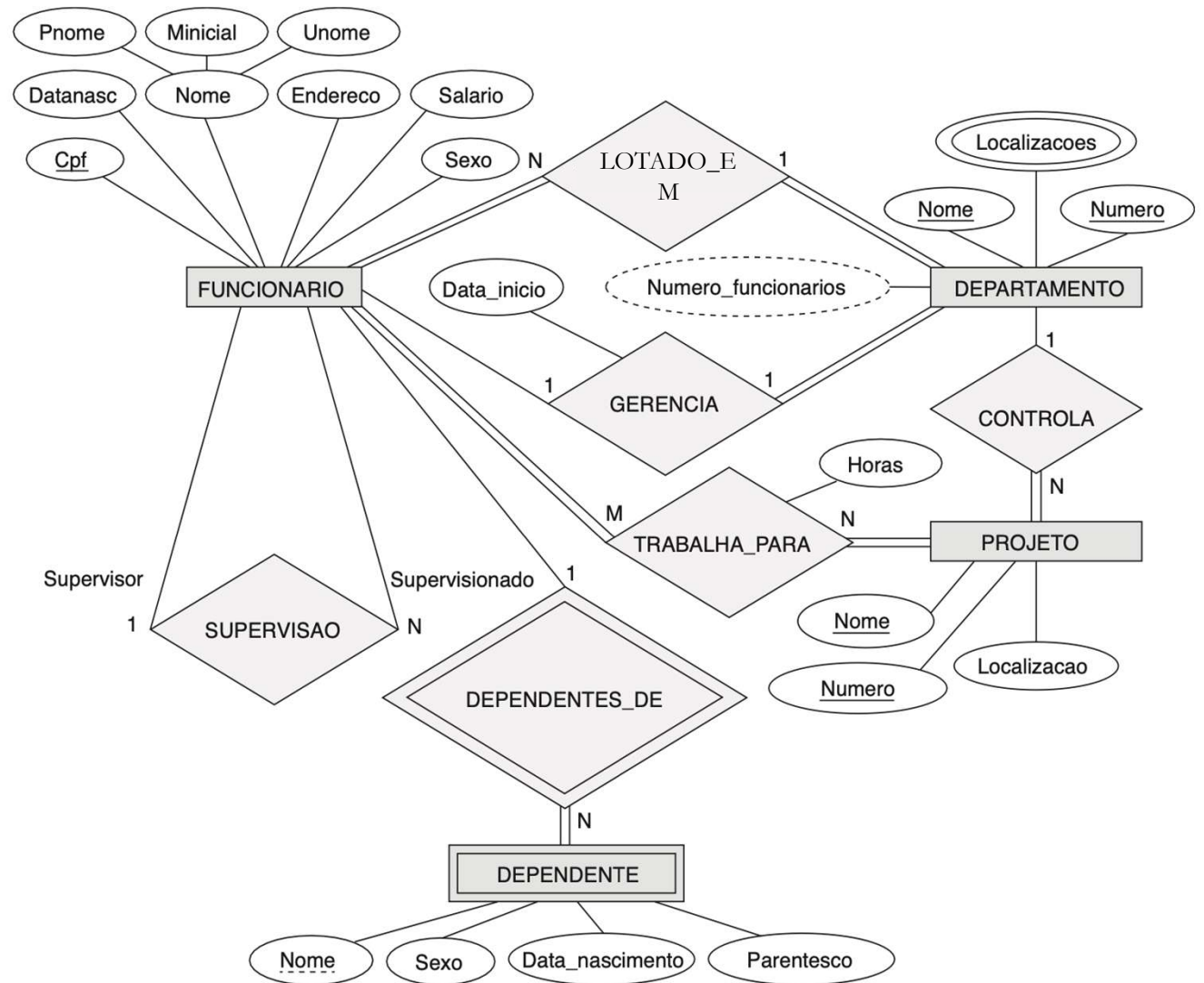


Diagrama ER

# Retomando o Estudo de Caso EMPRESA

# Banco de Dados EMPRESA



# Diagrama Relacional do BD EMPRESA

## **FUNCIONARIO**

Pnome	Minicial	Unome	<u>Cpf</u>	Datanasc	Endereco	Sexo	Salario	Cpf_supervisor	Dnr
-------	----------	-------	------------	----------	----------	------	---------	----------------	-----

## **DEPARTAMENTO**

Dnome	<u>Dnumero</u>	Cpf_gerente	Data_inicio_gerente
-------	----------------	-------------	---------------------

## **LOCALIZACAO\_DEP**

<u>Dnumero</u>	<u>Dlocal</u>
----------------	---------------

## **PROJETO**

Projnome	<u>Projnumero</u>	<u>Projlocal</u>	Dnum
----------	-------------------	------------------	------

## **TRABALHA\_EM**

<u>Fcpf</u>	<u>Pnr</u>	Horas
-------------	------------	-------

## **DEPENDENTE**

<u>Fcpf</u>	<u>Nome_dependente</u>	Sexo	Datanasc	Parentesco
-------------	------------------------	------	----------	------------



# Mapeamento de especializações/generalizações

- Há dois métodos gerais para mapeamentos de especializações/generalizações em tabelas. O primeiro, que chamamos aqui de método 1, funciona em todas as situações e consiste no seguinte:
  1. Crie uma tabela  $S$  para cada tipo de entidade de alto nível (superclasse).
  2. Crie uma tabela para cada tipo de entidades de baixo nível (subclasse) incluindo os atributos locais e a chave primária da superclasse.
  3. A chave primária das tabelas oriundas das subclasses é a mesma chave primária da tabela oriunda da superclasse.
  4. As chaves primárias das tabelas oriundas das subclasses são chaves estrangeiras para a tabela oriunda da superclasse.

# Mapeamento de Especializações/Generalizações

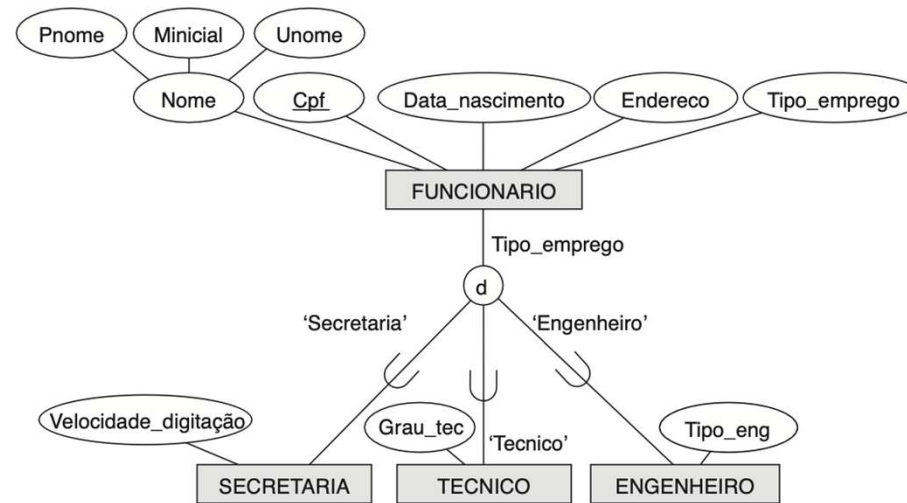


Diagrama  
ER

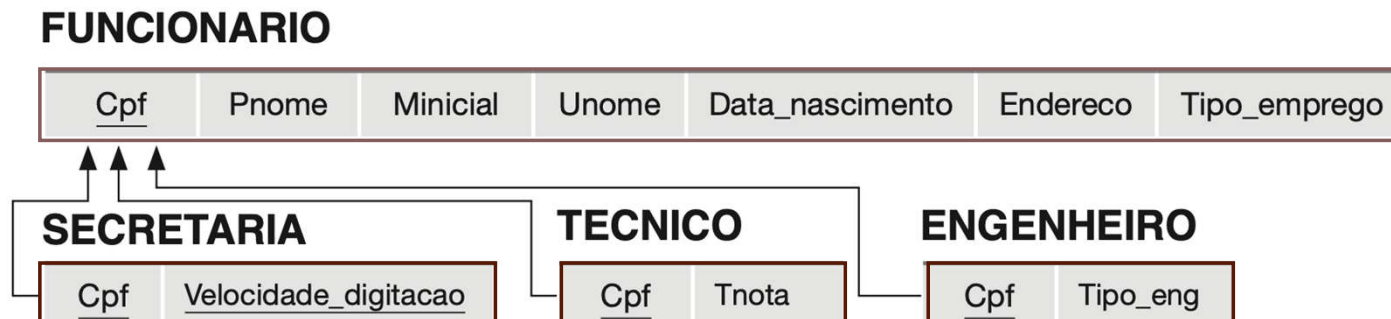


Diagrama  
Relacional

# Mapeamento de especializações/generalizações (cont.)

- Existem outros métodos de mapeamento de especializações/generalizações que levam em conta particularidades da hierarquia.
  - Se a especialização é parcial ou total.
  - Se a especialização é disjunta ou com sobreposição.
- Para mais detalhes consultar os livros de referência, ou material disponibilizado em seu curso anterior de Banco de Dados.

# Mapeamento de agregações

- Para mapear a agregação, trate-a como um conjunto de entidade de alto nível, contendo os atributos do relacionamento agregado (relacionamento *orienta*, no exemplo ao lado). A chave da agregação é a chave do relacionamento agregado.
- O conjunto de entidades de alto nível que representa a agregação se relaciona normalmente, como já explicado, com outros conjuntos de entidades.
- No exemplo ao lado, o conjunto de relacionamento *avaliação\_de* se tornaria uma tabela, pois o relacionamento é N:M. Os atributos dessa tabela os seguintes:

*avaliação\_de* (ID\_prof, ID\_aluno, ID\_proj, ID\_avaliação)

- A chave primária da tabela é a chave do relacionamento e segue as regras já apresentadas anteriormente.

