

# Projeto lógico de BD

## Mapeamento do Modelo ER para Modelo Relacional

---

Banco de Dados I

# Projeto do BD

---

- Projeto conceitual
    - Modelos Conceituais
      - ER
      - Diagrama de Classe
  - Projeto Lógico
    - Modelos lógicos
      - Modelo relacional (modelo de tabelas)
  - Projeto físico
-

# Mapeamento

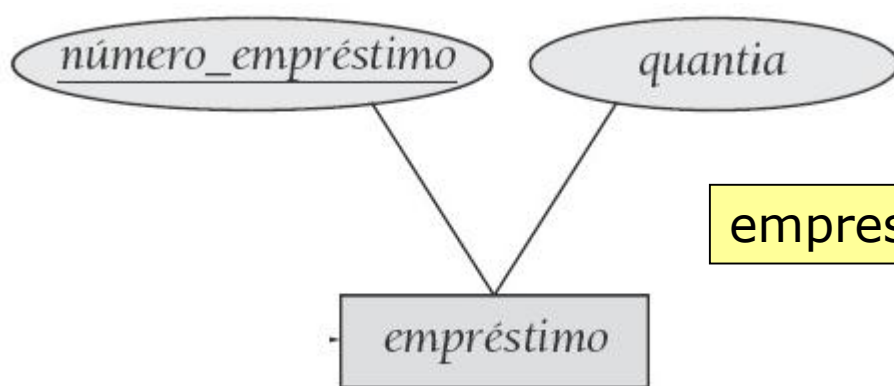
---

- ❑ Um banco de dados representado por um modelo ER pode ser representado por uma coleção de relações e relacionamentos
  - ❑ Não há uma regra universal, o mapeamento é muito dependente de contexto
    - Uma entidade no MER pode ser uma relação ou um atributo de outra relação no MR
    - Um relacionamento no MER pode ser um relacionamento, uma relação ou um atributo no MR
-

# Mapeamento de Entidades Fortes

---

- ❑ Cada conjunto de entidade no modelo ER pode ser representado por uma relação
- ❑ Atributos simples do conjunto de entidade podem ser representados como colunas
- ❑ Chave da entidade pode ser representada como chave primária da relação



`emprestimo(numero_emprestimo,quantia)`

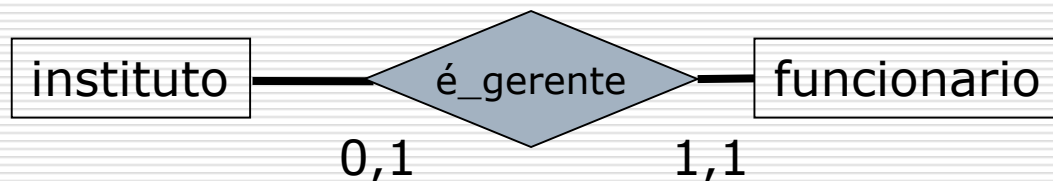
# Mapeamento de Relacionamento

---

- ❑ Relacionamento é implementado com chave estrangeira

instituto(codigo, nome)

funcionario(numero, nome)



Onde incluir a chave estrangeira?

---

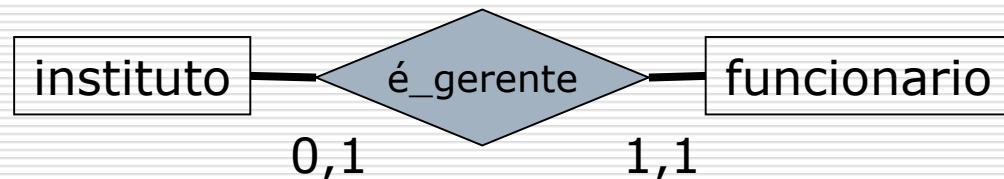
# Mapeamento de Relacionamento

---

- 1:1 – Primeira opção
  - Incluir Chave Estrangeira em uma das tabelas

instituto(codigo, nome)

funcionario(numero, nome)



Em qual tabela incluir a  
chave estrangeira?

---

# Mapeamento de Relacionamento

---

- 1:1 – Primeira opção
  - Chave estrangeira
    - A melhor escolha é a relação com participação total

instituto(codigo, nome, num\_gerente)  
num\_gerente referencia funcionario

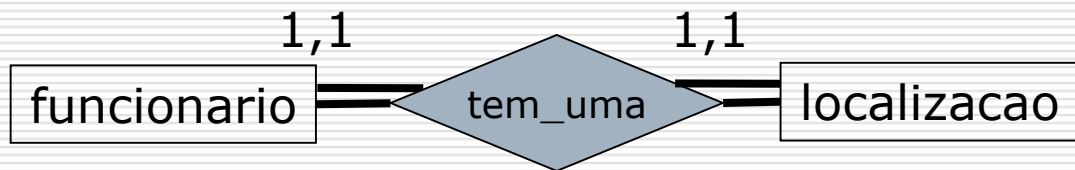
funcionario(numero, nome)

---

# Mapeamento de Relacionamento

- 1:1 – Segunda opção
  - Incorporar os atributos de uma entidade na outra
  - Útil quando o rel é total para as duas entidades envolvidas e uma das entidades não possui outros relacionamentos

localizacao(codigo,descricao,bloco,sala,tipo\_restricao)



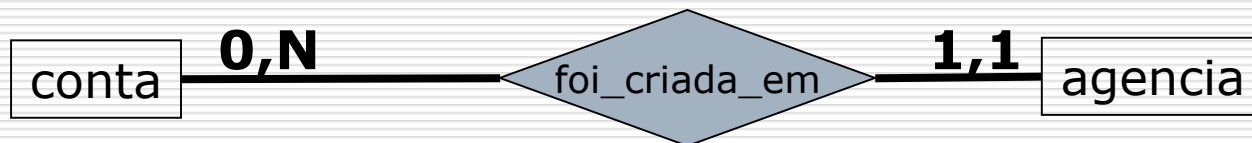
funcionario(numero,nome,codigo,descricao,bloco,sala,tipo\_restricao)



# Mapeamento de Relacionamento

---

- 1:N
  - Inserir a chave estrangeira em uma das tabelas de forma que a coluna correspondente não seja multivalorada



Onde inserir a chave estrangeira?

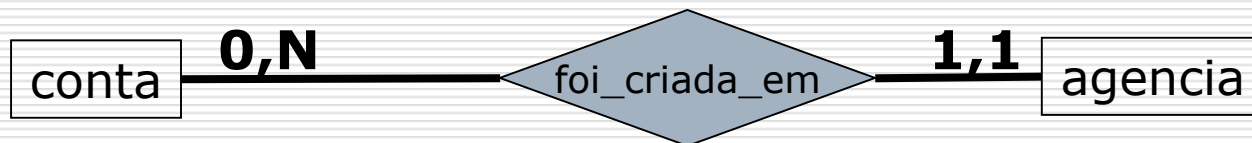
Se a chave estrangeira fosse inserida em agencia, a tabela teria uma coluna contendo vários valores

---

# Mapeamento de Relacionamento

---

- 1:N entre r e s
  - Inserir a chave estrangeira em uma das relações de forma que a coluna correspondente não seja multivalorada



conta(codigo, saldo, numAgencia, etc ....)  
numAgencia referencia agencia

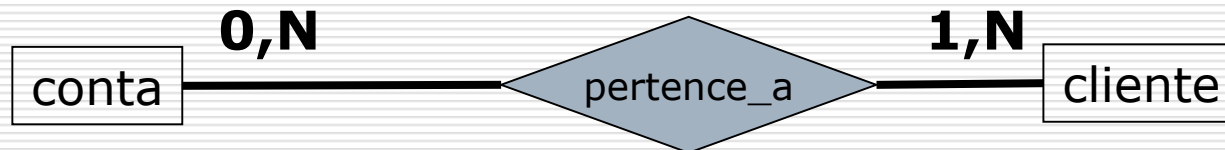
agencia(numero, nome, ....)

---

# Mapeamento de Relacionamento

---

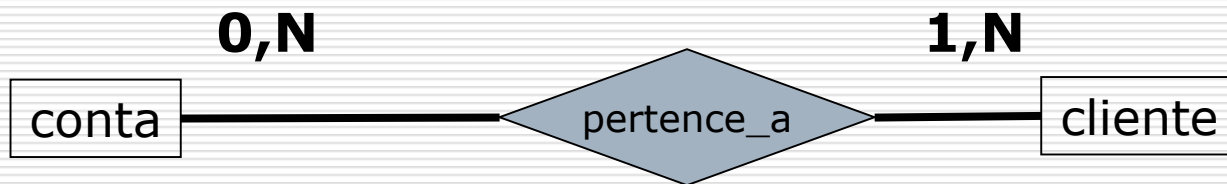
- N:N entre r e s
  - Criar tabela t para representar o relacionamento
  - t é chamada de tabela de relacionamento
  - Chave primária de t
    - Chave primaria de r e s
- Atributos de relacionamento
  - São definidos como colunas da tabela t



# Mapeamento de Relacionamento

---

- N:N entre r e s



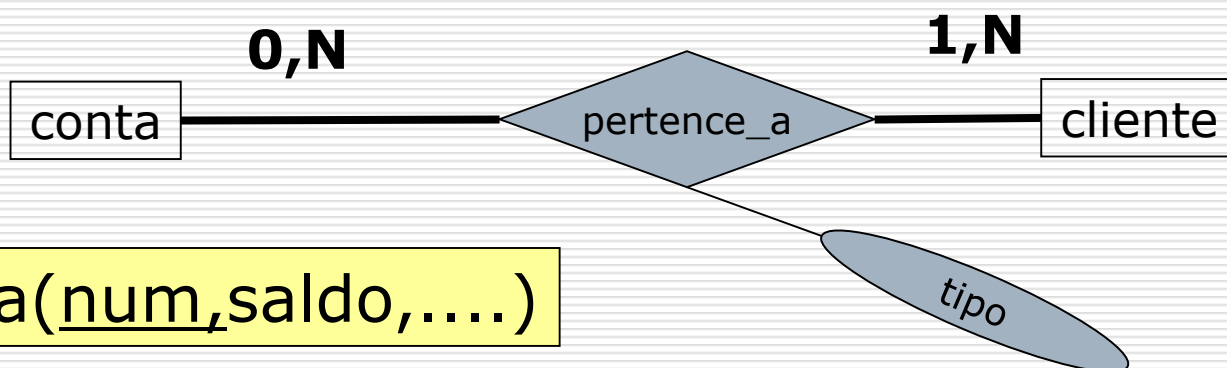
conta(num,saldo,....)

cliente(codigo, nome, ....)

conta\_cliente(codigo,num)  
codigo referencia cliente  
num referencia conta

# Mapeamento de Relacionamento

- N:N entre r e s com atributo



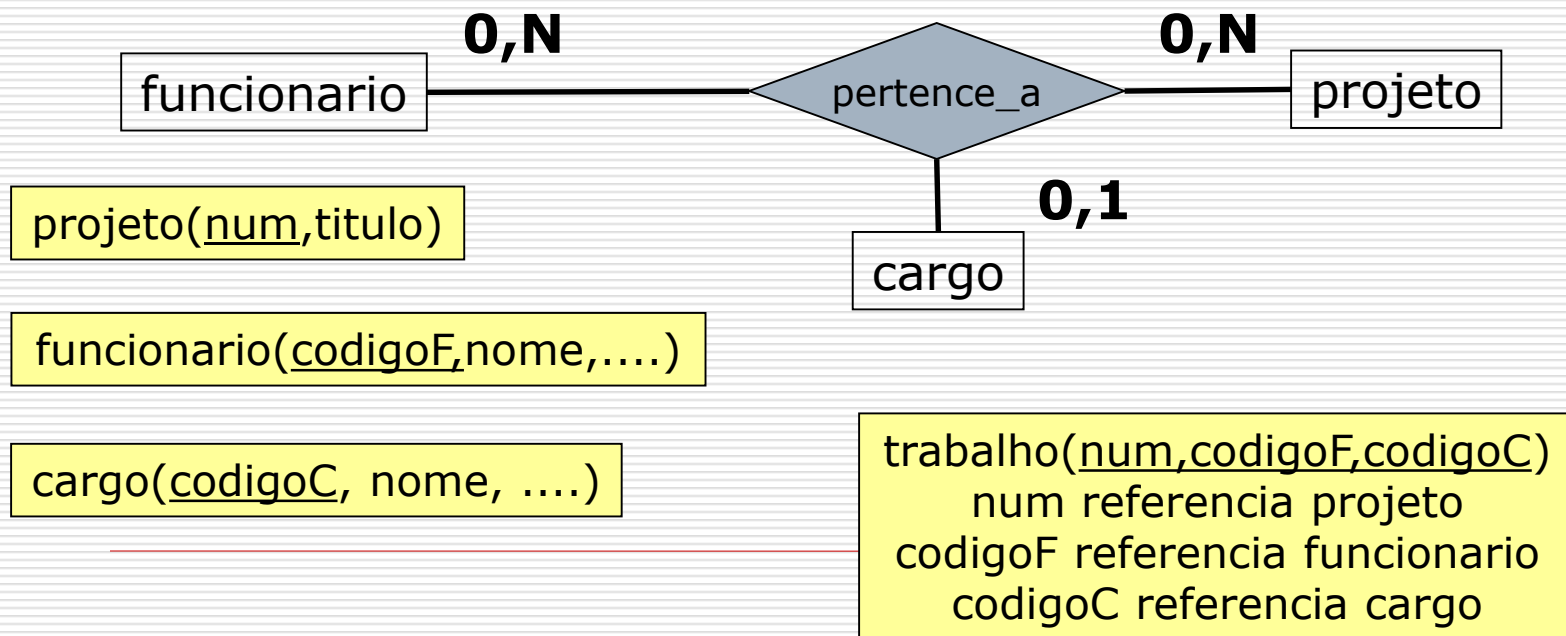
conta(num,saldo,....)

cliente(codigo, nome, ....)

conta\_cliente(codigo,num, **tipo**)  
codigo referencia cliente  
num referencia conta

# Mapeamento de Relacionamento

- ❑ Relacionamento N-ário (r, s e t)
  - Criar uma relação U para representar o relacionamento
    - ❑ Chave primária de U é composta pela chaves de r, s e t

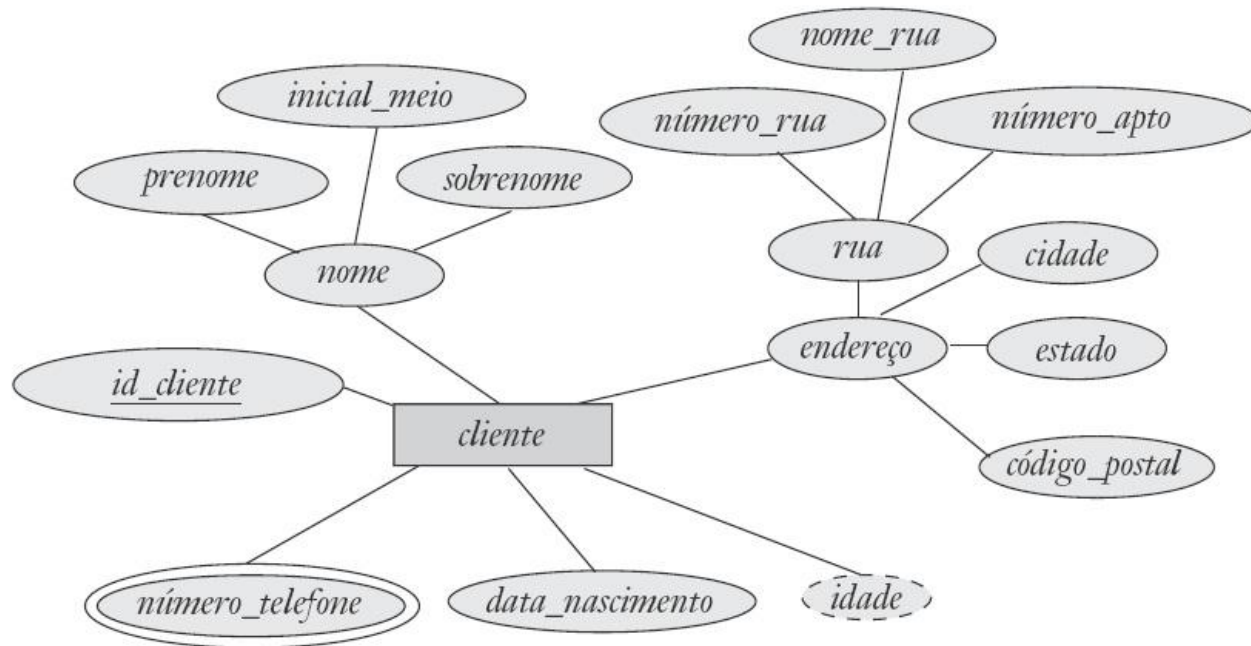


# Mapeamento de atributos compostos e multivalorados

---

- Atributos compostos
    - Cada atributo componente é representado como atributo simples na relação
  - Atributo multivalorado
    - Cria uma nova relação para representar o atributo
-

# Atributo composto

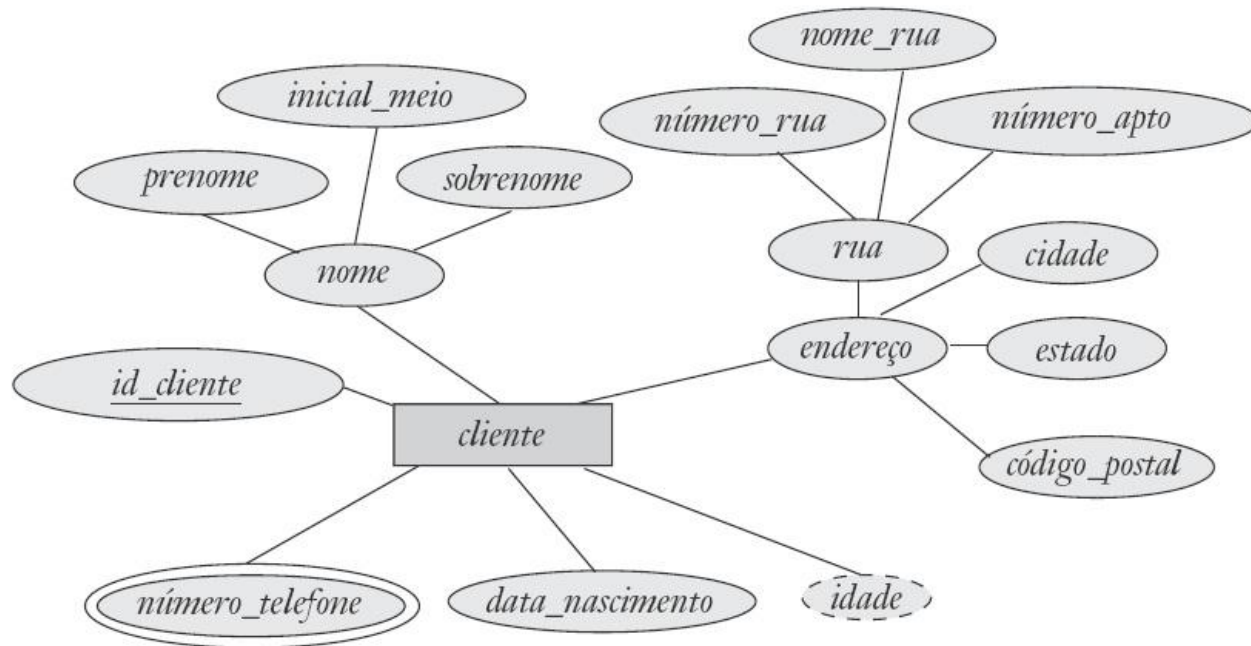


cliente(id\_cliente, prenome, inicial\_meio, sobrenome, ....)

E o endereço?

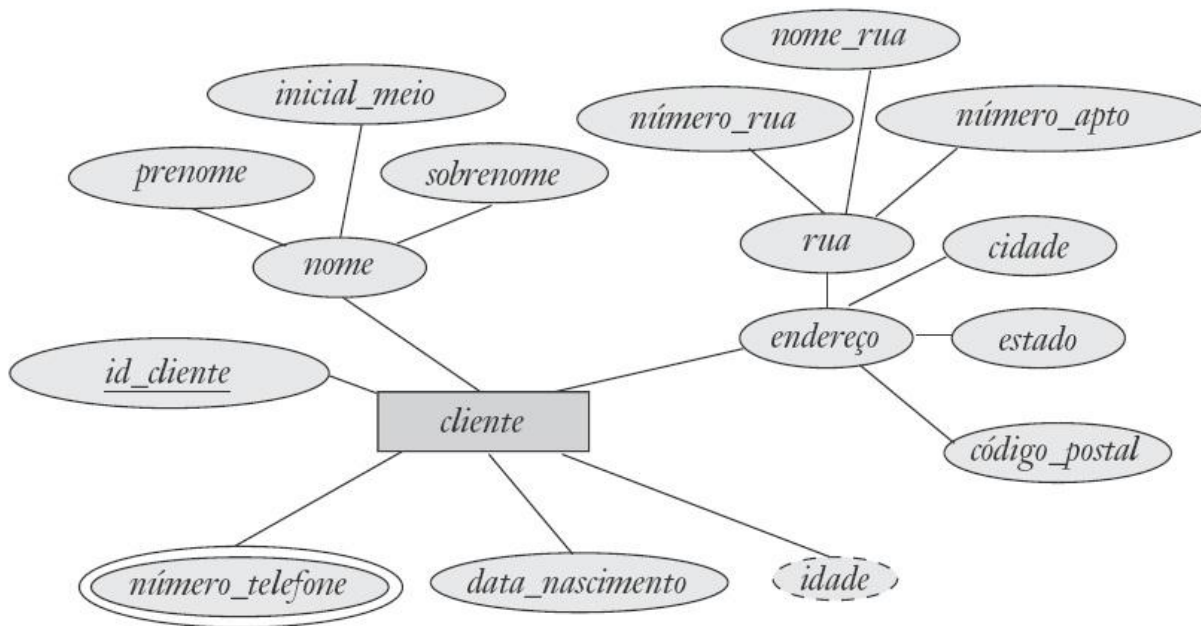


# Atributo composto



cliente(id\_cliente, prenome, inicial\_meio, sobrenome, número\_rua, nome\_rua, número\_apto, cidade, estado, código\_postal, número\_telefone, data\_nascimento, idade)

# Atributo multivalorado



cliente(id\_cliente, prenome, etc ....)

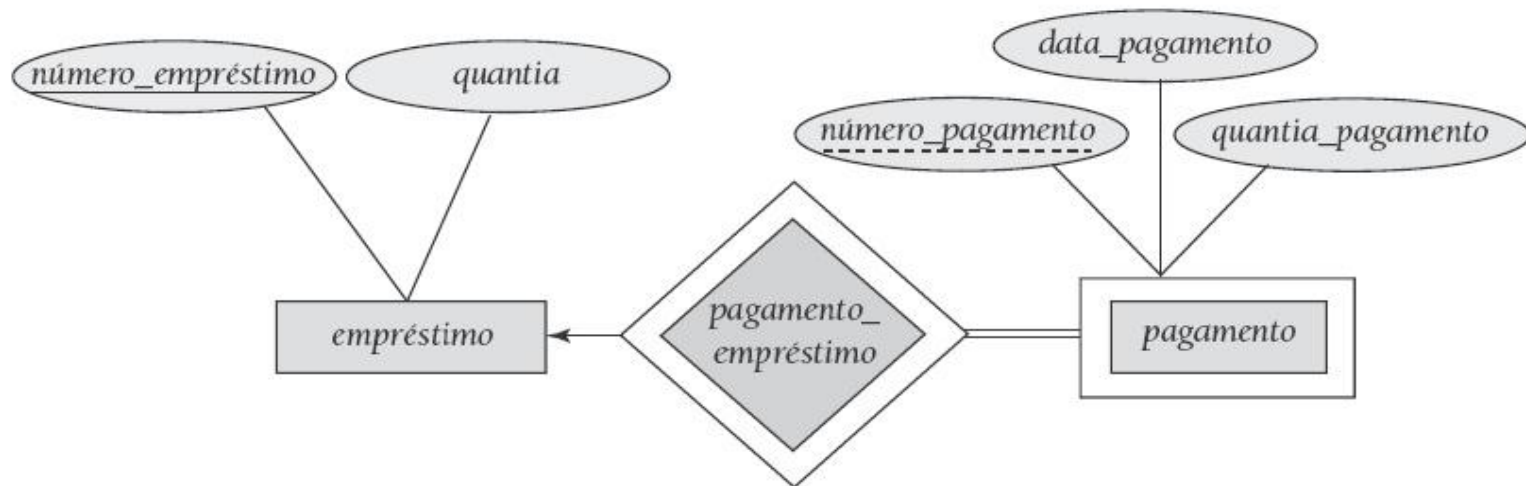
telefone(id\_cliente, número telefone)  
id\_cliente referencia cliente

# Mapeamento de Entidades Fracas

---

- Conjunto de entidade fraca é representado como uma relação
    - Atributos simples são representados como colunas
    - Inclui uma chave estrangeira que referencia a chave primária do conjunto de entidades forte identificadora
    - Chave primária da entidade fraca é composta
      - Chave primária da entidade forte
      - Chave parcial (atributo discriminador) da entidade fraca
-

# Mapeamento de Entidades Fracas

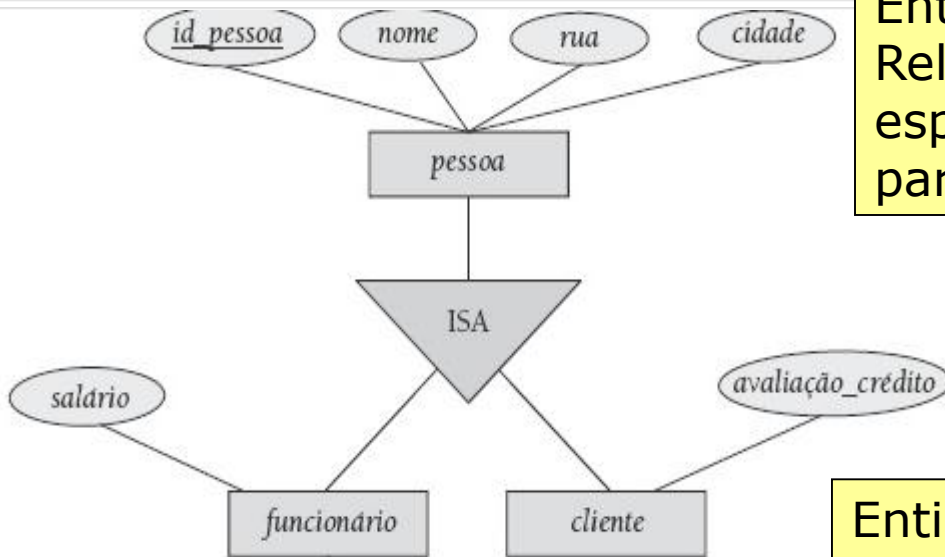


**pagamento**(numero\_emprestimo, numero\_pagamento, data\_pagamento, quantia\_pagamento)

numero\_emprestimo referencia emprestimo (numero\_emprestimo)

# Mapeamento de Especialização e Generalização

- Primeira opção
  - Criar uma relação para cada entidade



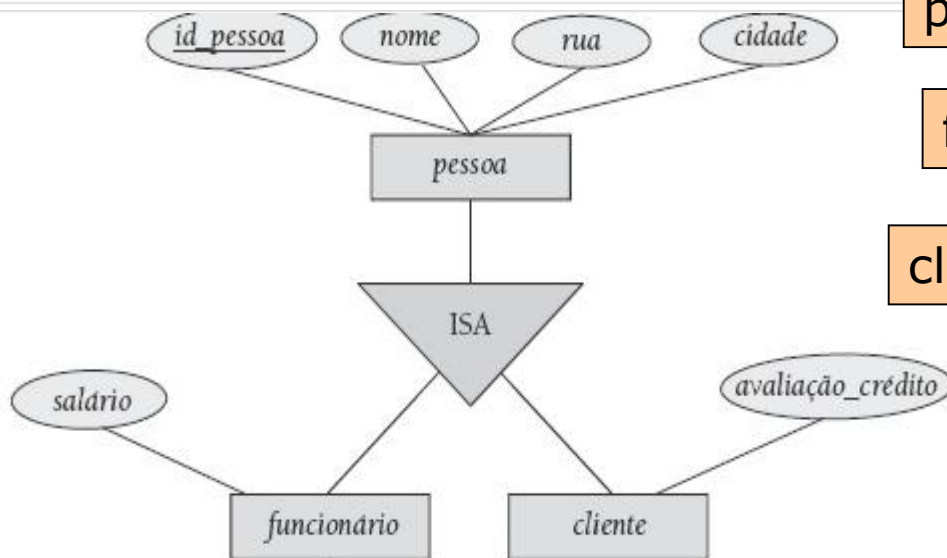
Entidades de nível inferior:  
Relações possuem atributos específicos + chave estrangeira para relação de nível superior

Entidades de nível inferior:  
Chave primária é igual a chave primária da relação de nível superior

# Mapeamento de Especialização e Generalização

## □ Primeira opção

- Criar uma tabela para cada entidade



pessoa(id\_pessoa, nome, etc ...)

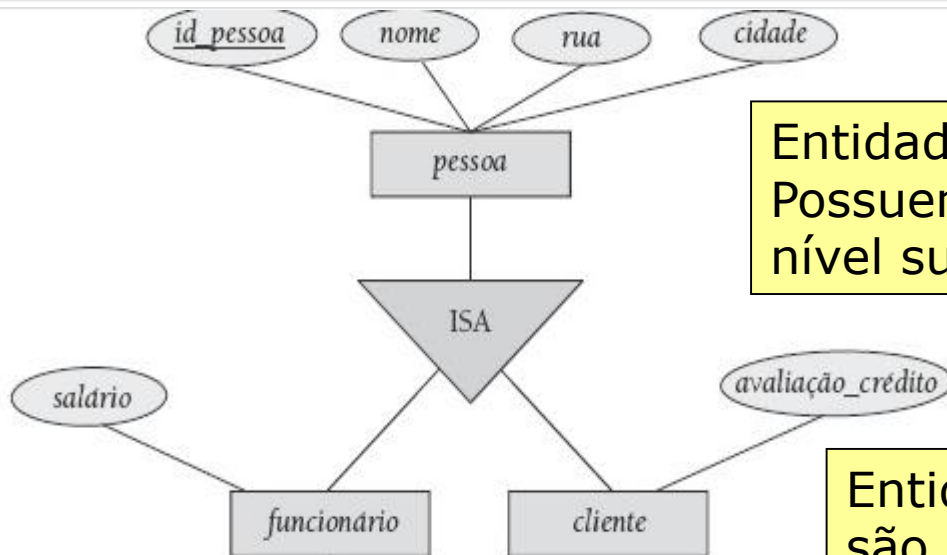
funcionario(id\_pessoa, salario)

cliente(id\_pessoa, avaliacao\_credito)

Vale tanto para generalizações não disjuntas quanto parciais

# Mapeamento de Especialização e Generalização

- Segunda opção
  - Criar tabelas somente para entidade de nível inferior



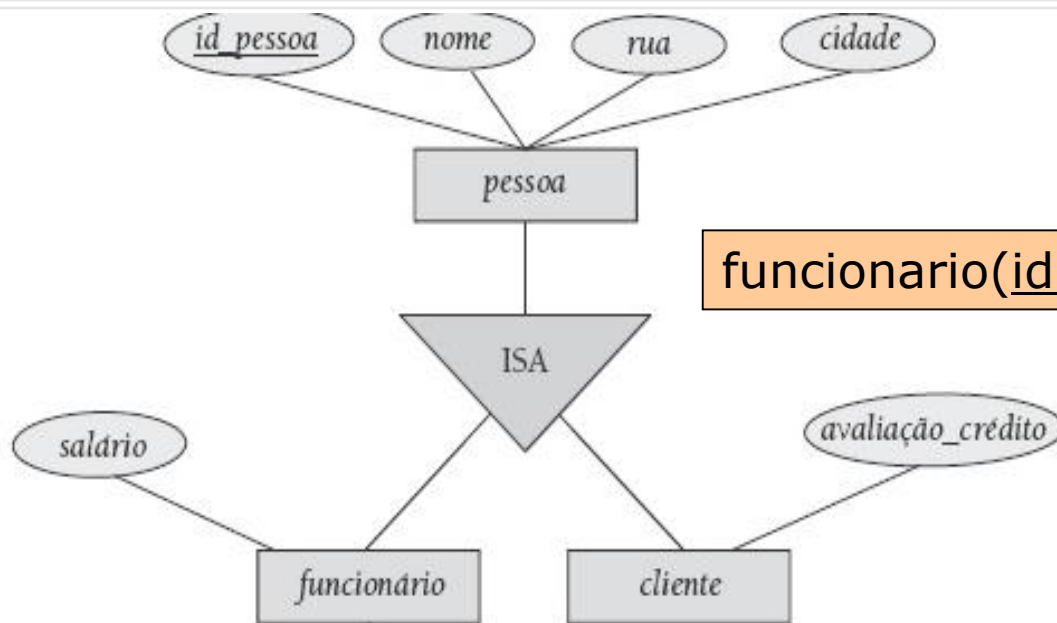
Entidades de nível inferior:  
Possuem os atributos da entidade de nível superior + atributos específicos

Entidades de nível superior não são representadas

# Mapeamento de Especialização e Generalização

## □ Segunda opção

cliente(id\_pessoa, nome, rua, ..., avaliacao\_credito)



funcionario(id\_pessoa, nome, rua, ....., salario)

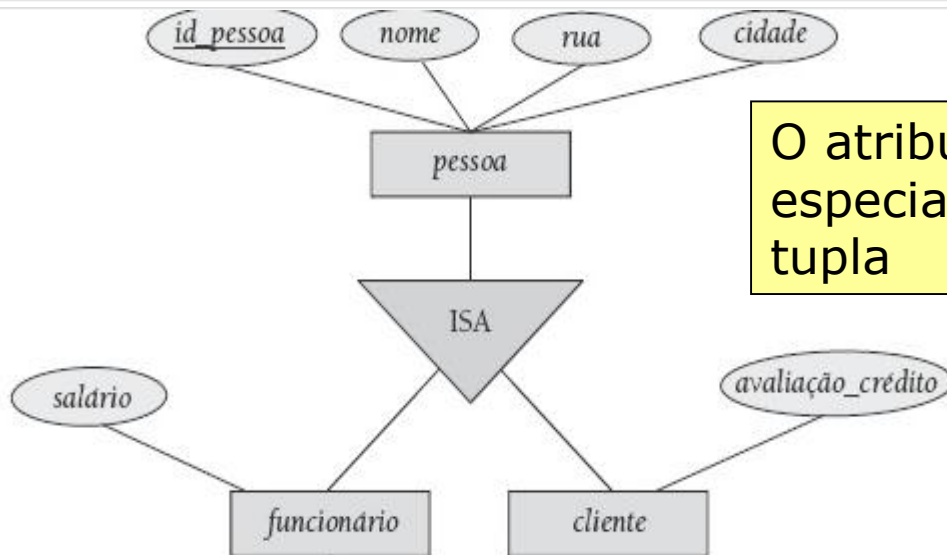
Não é recomendada para generalização parcial ou que não sejam disjuntas



# Mapeamento de Especialização e Generalização

## □ Terceira opção

- Criar uma única tabela com todos os atributos das entidades envolvidas + atributo tipo

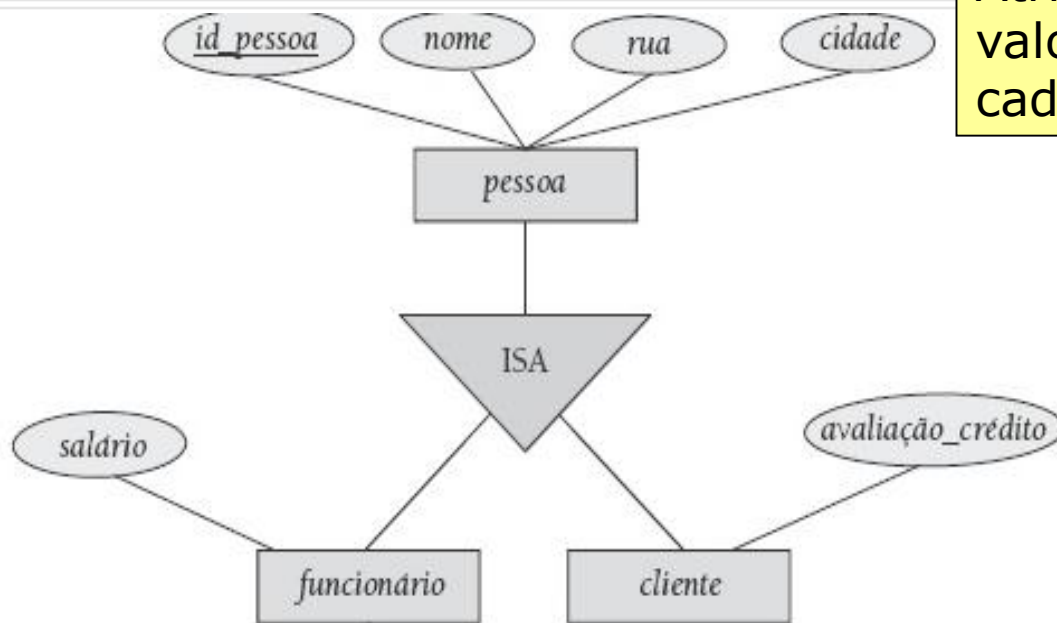


O atributo tipo indica qual é a especialização de uma determinada tupla

# Mapeamento de Especialização e Generalização

## □ Terceira opção

pessoa(id\_pessoa, nome, rua, ..., avaliacao\_credito, salario, **tipo**)



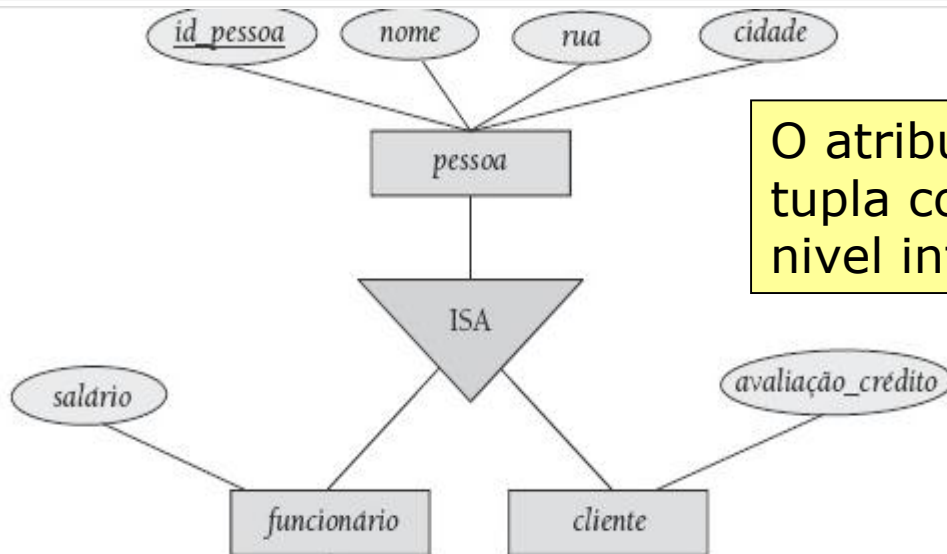
Atributo tipo poderia assumir os valores funcionario ou cliente para cada tupla na relação pessoa

Não é indicada quando entidades de nível inferior possuem muitos atributos específicos

Pode gerar muitas tuplas com valores nulos nos atributos específicos

# Mapeamento de Especialização e Generalização

- ❑ Variação da terceira opção
  - Criar uma única relação com todos os atributos das entidades envolvidas e um atributo booleano para indicar a entidade específica

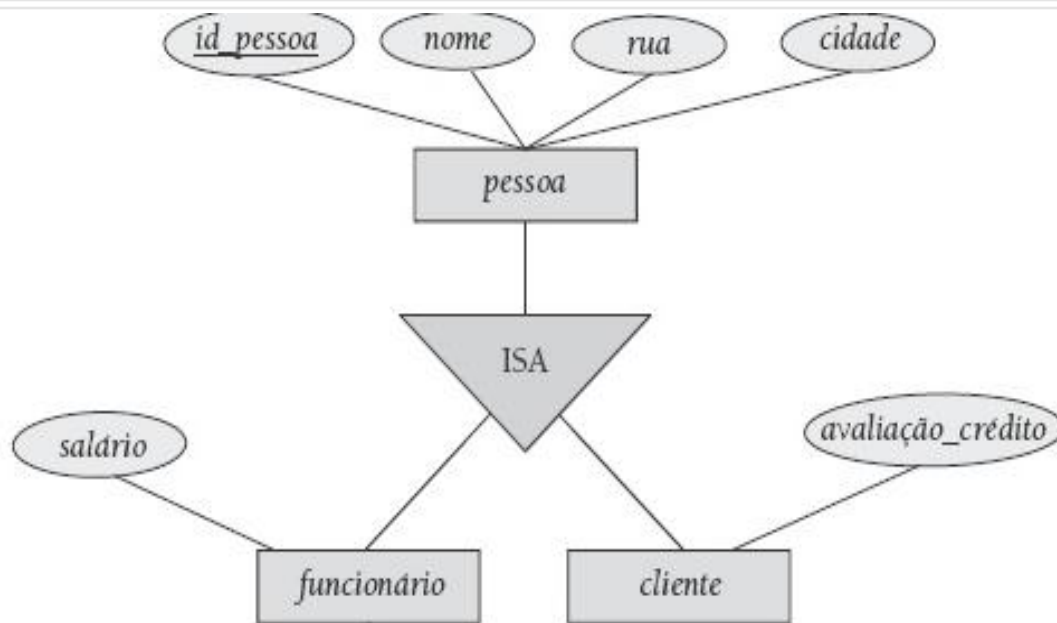


O atributo booleano indica se uma tupla corresponde a uma entidade de nível inferior ou não

# Mapeamento de Especialização e Generalização

## □ Variação da terceira opção

pessoa(id\_pessoa, nome, rua, ..., **f\_tipo**, avaliacao\_credito, **c\_tipo**, salario)



Caso uma tupla for um cliente,  $c\_tipo := true$ ;  
Se a tupla for funcionario  $f\_tipo := true$ ;  
Caso uma pessoa não seja nem funcionario, nem cliente:  
 $c\_tipo := false$   
 $f\_tipo := false$

Útil para generalização que não são disjuntas e parciais