

Banco de Dados

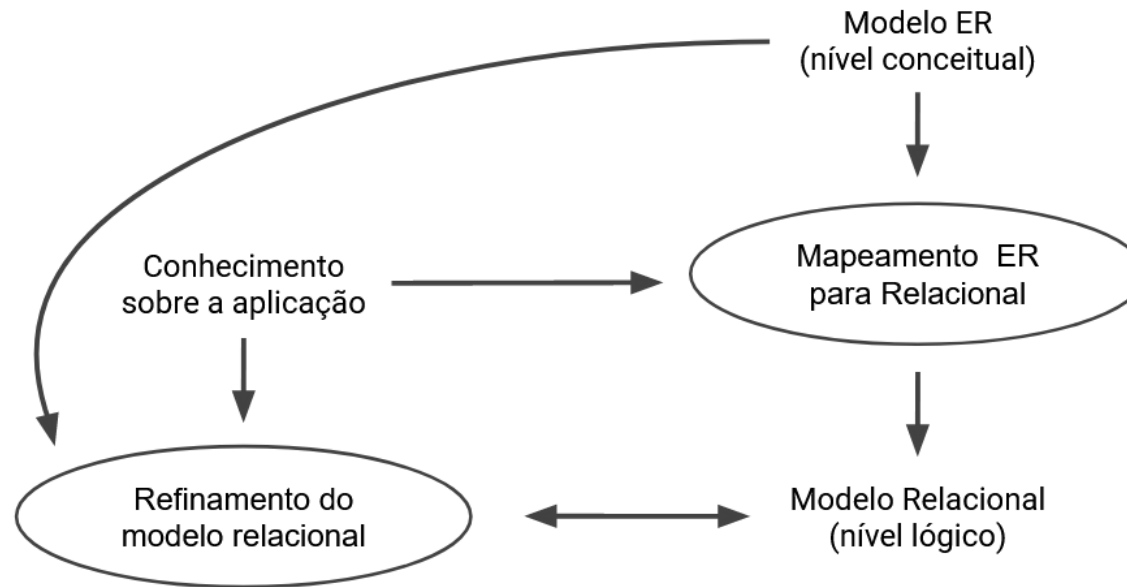
MAPEAMENTO ENTIDADE-RELACIONAMENTO PARA RELACIONAL

Introdução

- Um mesmo **Modelo Entidade-Relacionamento** (projeto conceitual) pode ter diferentes implementações em um **Modelo Relacional** (modelo lógico).
 - Essas diferentes implementações podem resultar em bancos de dados corretos, no entanto, com diferenças no desempenho.
- A partir da experiência acumulada de muitos pesquisadores e profissionais da área de bancos de dados, foram definidas regras para construção do **Modelo Relacional** a partir de um **Modelo Entidade-Relacionamento**.

Introdução

- As regras que serão apresentadas refletem um consenso de como deve ser projetado um banco de dados eficiente.
- Entretanto, o modelo relacional resultante da aplicação dessas regras pode ser considerado um modelo inicial.
 - O resultado pode não atender aos requisitos de performance do projetado, sendo necessário um processo de melhoria e refinamento desse modelo.



Introdução

- Alguns princípios por trás das regras de mapeamento ER para o modelo relacional:
 - **Evitar junções:**
 - Junções são operações para buscar dados de diferentes tabelas, associando as linhas dessas tabelas de acordo com um ou mais campos.
 - Realizar as operações de junções podem envolver diversos acessos ao disco, o que resulta em consultas mais lentas ao banco de dados.
 - **Reduzir o número de chaves:**
 - Os SGBDs relacionais usam as chaves para gerar índices que determinam como os registros de um banco de dados devem estar organizados.
 - Esses índices, assim como os registros, são armazenados em disco, podendo ocupar um espaço considerável.
 - Além disso, os arquivos de índice devem ser atualizados sempre que operações de inserção e remoção de registros são realizadas no banco de dados, o que pode tornar lento a realização dessas operações.
 - **Evitar campos opcionais:**
 - Campos opcionais, mesmo quando não utilizados (valor nulo) tem o seu espaço reservado no disco, desperdiçando esse espaço de armazenamento. SGBDs atuais usualmente utilizam estratégias de armazenamento para evitar esse desperdício).
 - Campos opcionais podem complicar a programação de aplicações que usam o banco de dados, pois implica em verificar se o valor de um campo é vazio ou não antes de utilizá-lo.

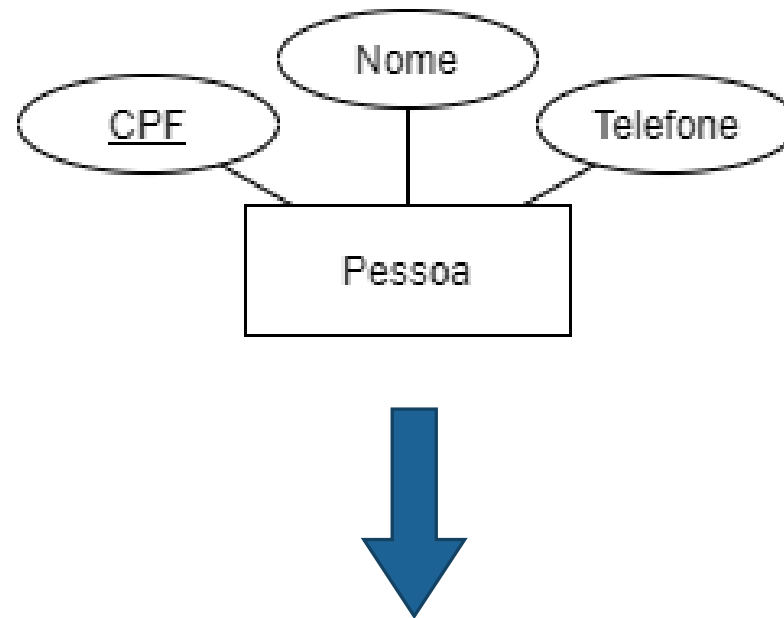
Introdução

- Visão geral do processo de mapeamento ER para o Modelo Relacional:
 - Mapeamento preliminar de entidades e seus respectivos atributos.
 - Mapeamento de generalizações/especializações.
 - Mapeamento de relacionamentos e seus respectivos atributos.

Mapeamento ER para Modelo Relacional

Entidades e atributos

- Entidade forte e seus atributos simples:

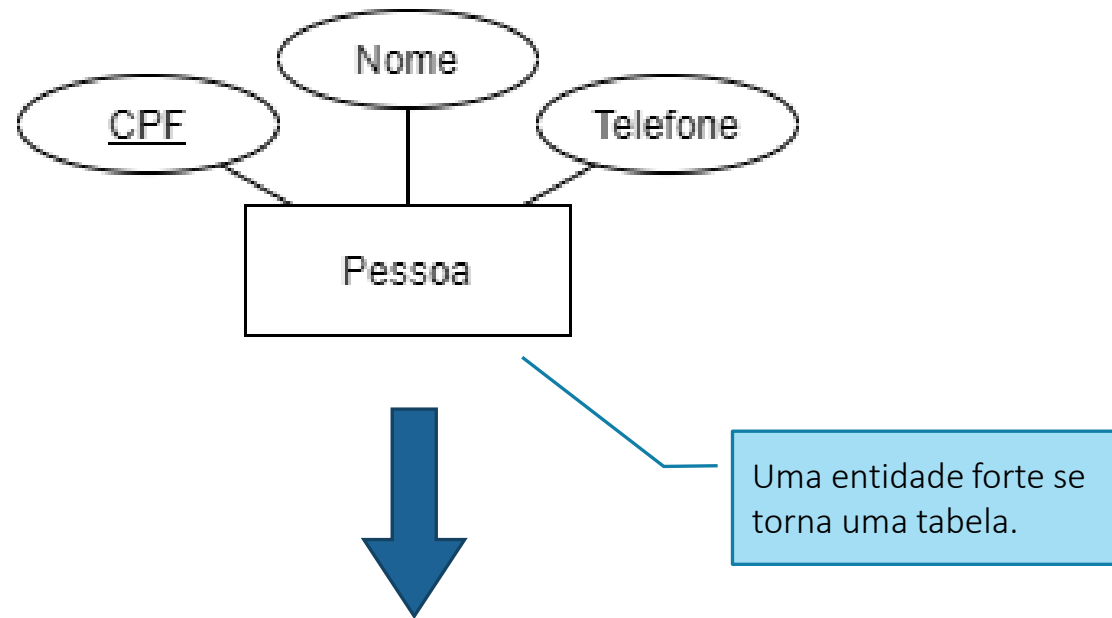


`PESSOA(cpf, nome, telefone)`

Mapeamento ER para Modelo Relacional

Entidades e atributos

- Entidade forte e seus atributos simples:

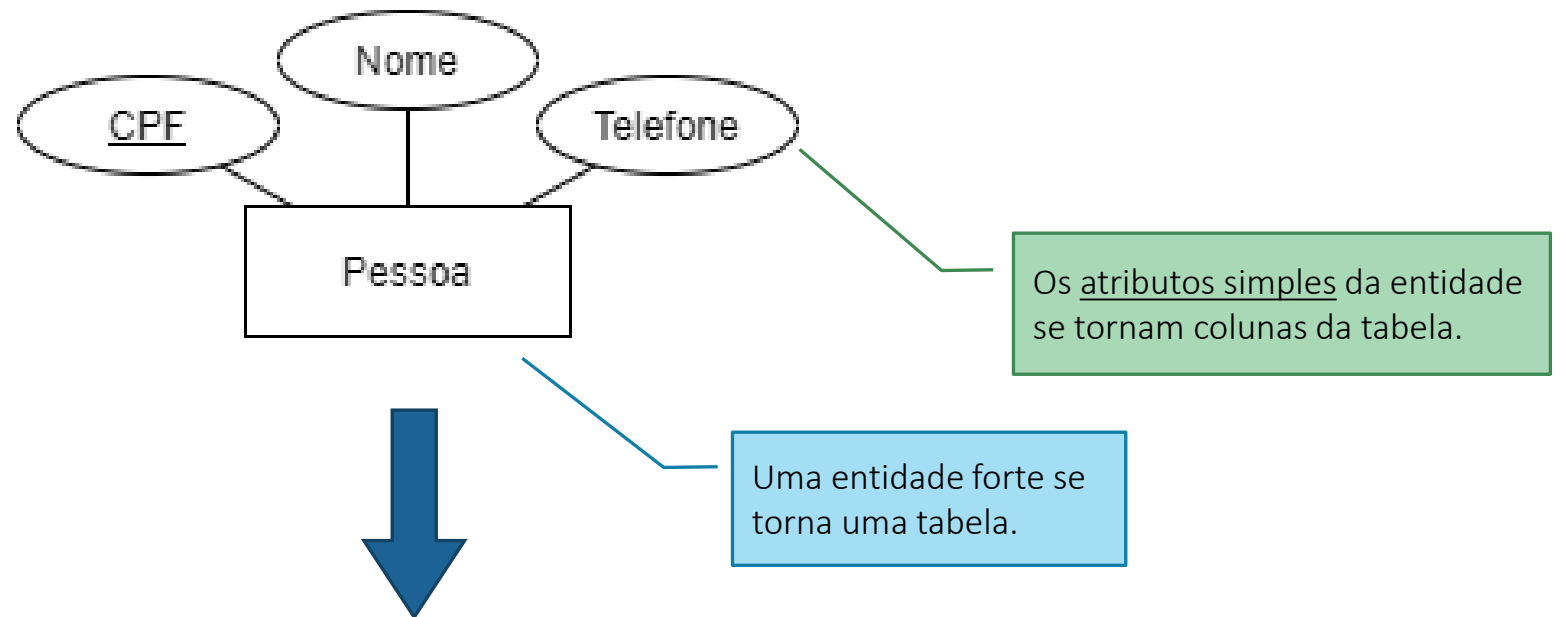


```
PESSOA( cpf, nome, telefone )
```

Mapeamento ER para Modelo Relacional

Entidades e atributos

- Entidade forte e seus atributos simples:

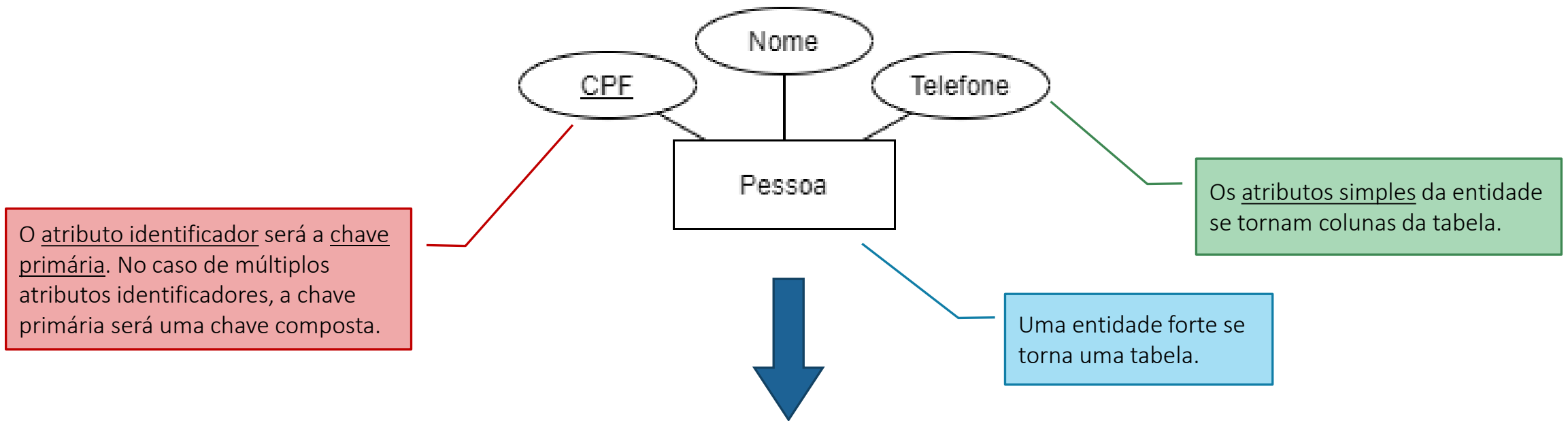


```
PESSOA( cpf, nome, telefone )
```


Mapeamento ER para Modelo Relacional

Entidades e atributos

- Entidade forte e seus atributos simples:

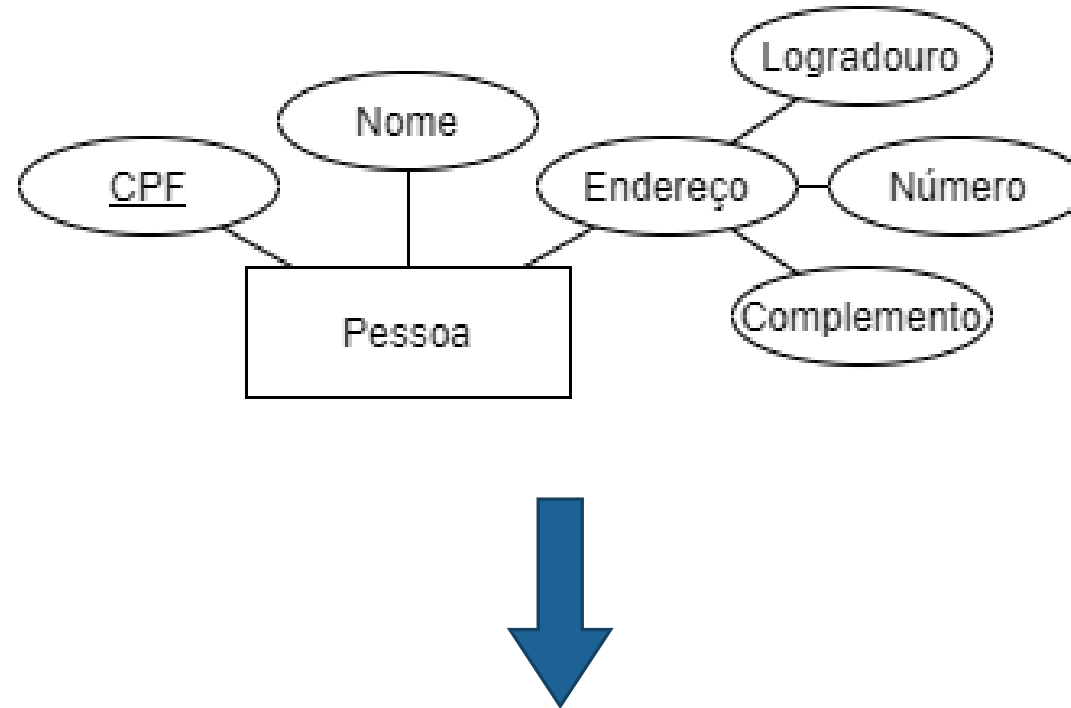


```
PESSOA( cpf, nome, telefone )
```

Mapeamento ER para Modelo Relacional

Entidades e atributos

- Atributos compostos:

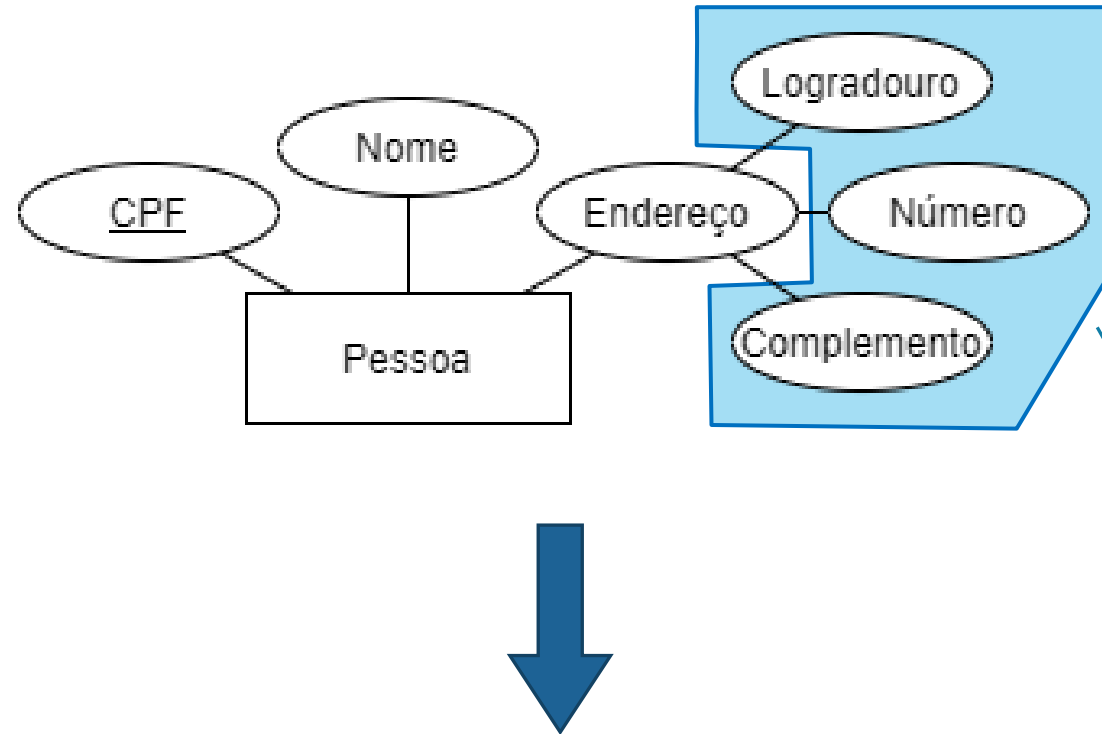


```
PESSOA( cpf, nome, endereco_logradouro, endereco_numero, endereco_complemento )
```

Mapeamento ER para Modelo Relacional

Entidades e atributos

- Atributos compostos:



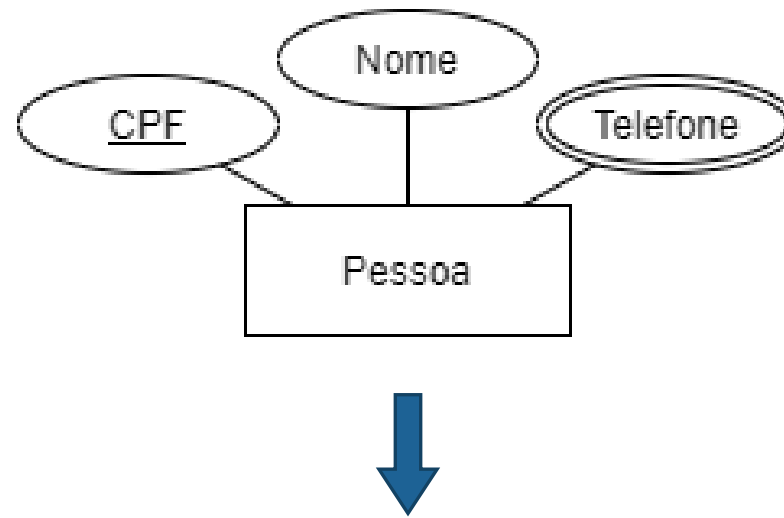
Cada atributo menor que forma o atributo composto se torna uma coluna na tabela da entidade que possui o atributo composto. Geralmente, os nomes das colunas desses atributos são definidos como o nome do atributo composto seguido pela respectiva parte.

```
PESSOA( cpf, nome, endereco_logradouro, endereco_numero, endereco_complemento )
```

Mapeamento ER para Modelo Relacional

Entidades e atributos

- Atributos multivalorados:

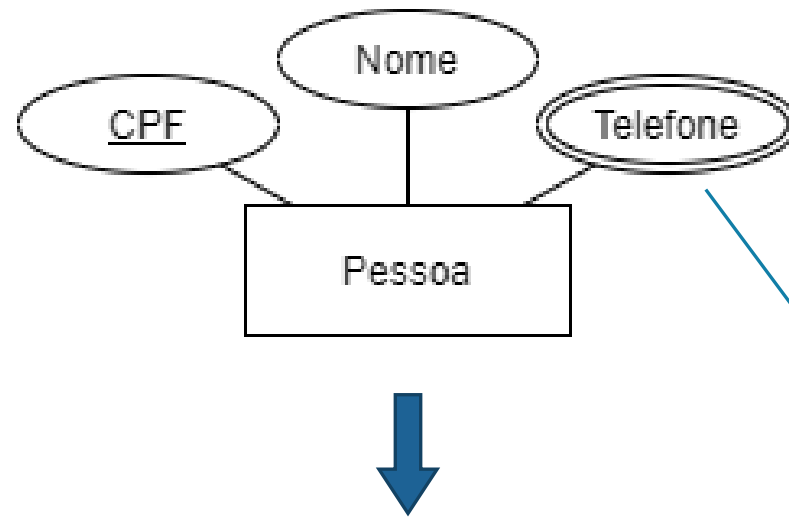


```
PESSOA( cpf, nome )  
PESSOA_TELEFONE( cpf, telefone )  
    cpf referencia PESSOA
```

Mapeamento ER para Modelo Relacional

Entidades e atributos

- Atributos multivalorados:



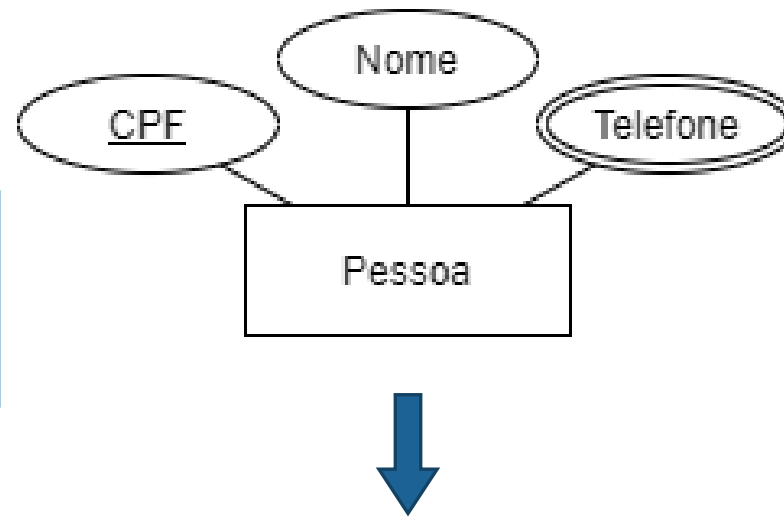
Cada atributo multivalorado é mapeado para sua própria tabela. Geralmente definimos o nome dessa tabela como o nome da tabela da respectiva entidade seguido pelo nome do atributo.

```
PESSOA( cpf, nome )
PESSOA_TELEFONE( cpf, telefone )
               cpf referencia PESSOA
```

Mapeamento ER para Modelo Relacional

Entidades e atributos

- Atributos multivalorados:



Além da coluna que armazena o valor do atributo multivalorado, essa tabela tem coluna(s) que referenciam a tabela da respectiva entidade que possui o atributo.

```
PESSOA ( cpf, nome )
PESSOA_TELEFONE ( cpf, telefone )
               cpf referencia PESSOA
```

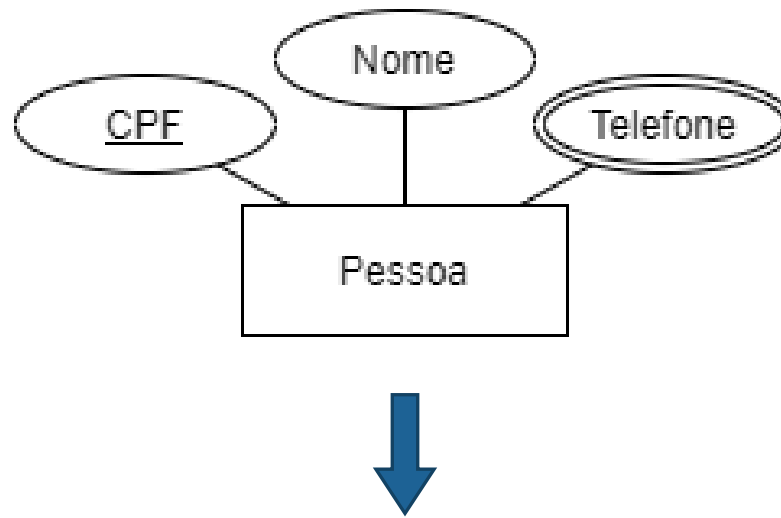
Mapeamento ER para Modelo Relacional

Entidades e atributos

- Atributos multivalorados:

As colunas que referenciam a tabela da entidade que possui o atributo multivalorado e o valor desse atributo devem compor a chave primária.

No exemplo, a tabela **PESSOA_TELEFONE** tem a chave primária composta pelas colunas **cpf** e **telefone**. Assim, cada valor de **telefone** deverá estar associado a um **cpf** (pessoa) e um mesmo **cpf** (pessoa) poderá ter associado múltiplos valores de **telefone**. Observe que é possível um mesmo número de telefone aparece mais de uma vez, porém para diferentes CPFs.

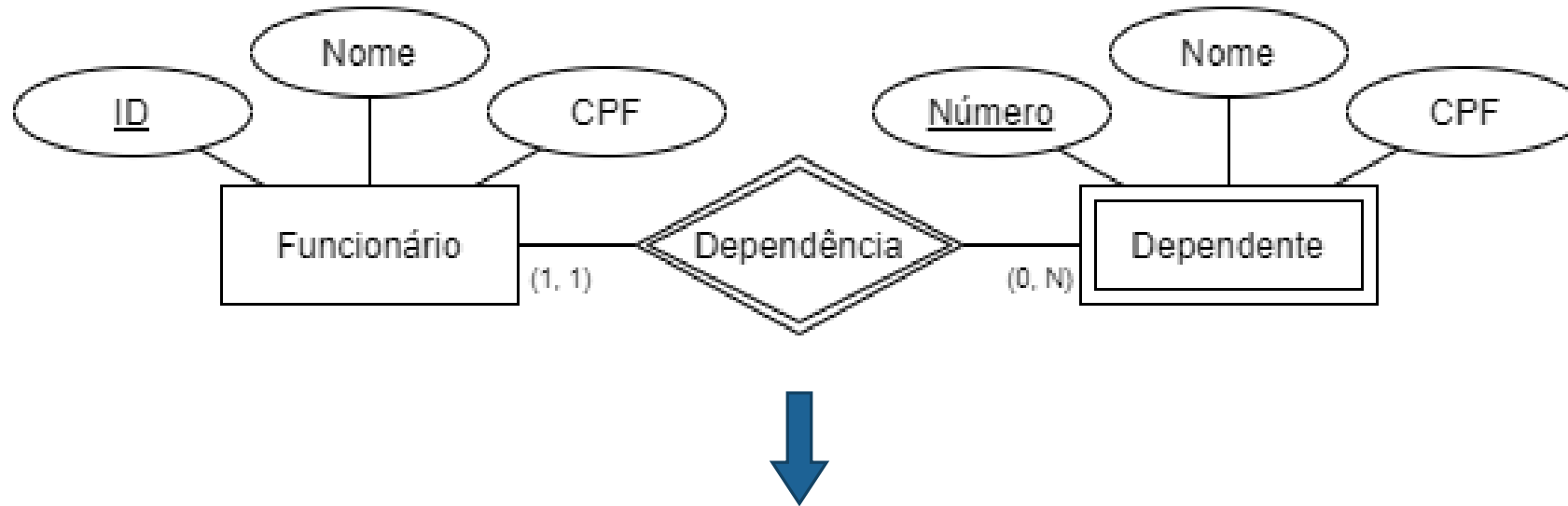


```
PESSOA( cpf, nome )  
PESSOA_TELEFONE( cpf, telefone )  
               cpf referencia PESSOA
```

Mapeamento ER para Modelo Relacional

Entidades e atributos

- Entidade fraca e relacionamento identificador:

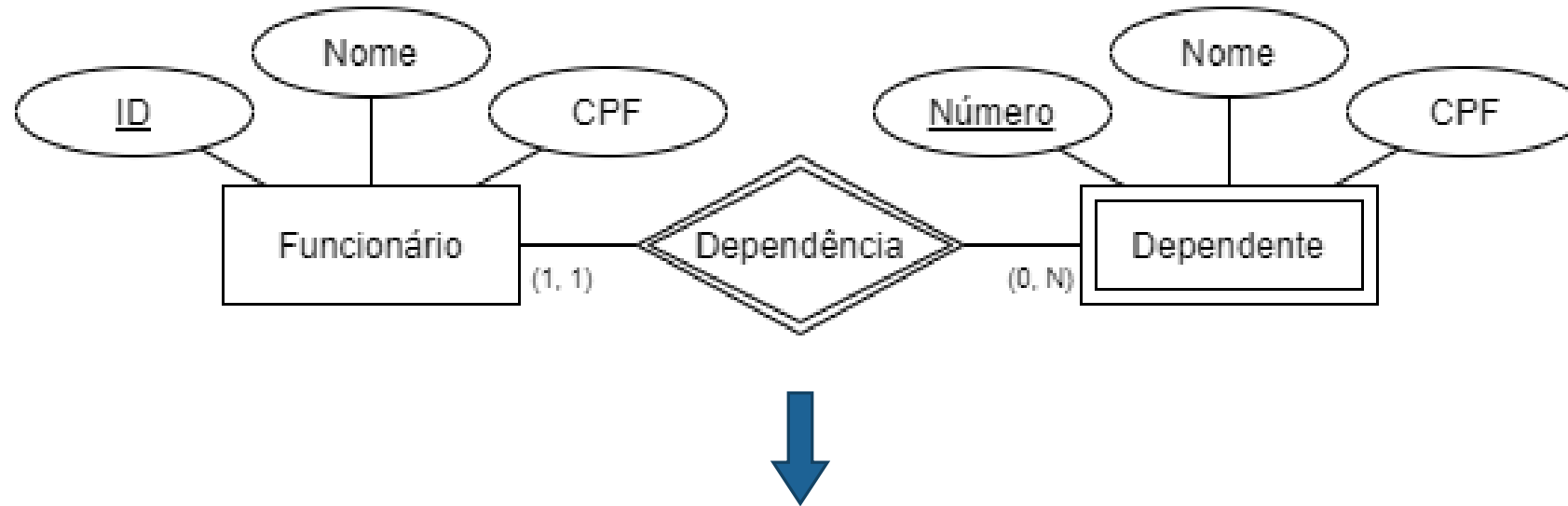


```
FUNCIONARIO( id, cpf, nome )  
DEPENDENTE( funcionario_id, numero, cpf, nome )  
            funcionario_id referencia FUNCIONARIO
```


Mapeamento ER para Modelo Relacional

Entidades e atributos

- Entidade fraca e relacionamento identificador:



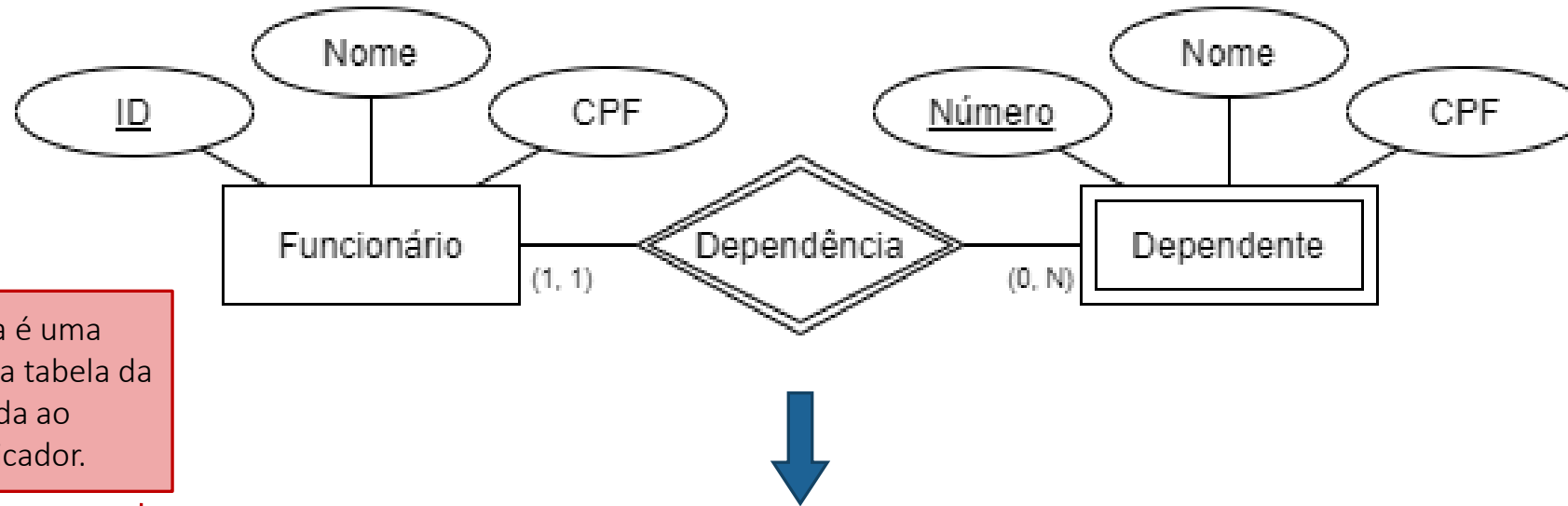
A chave primária da tabela gerada a partir de uma entidade fraca é uma chave composta.

```
FUNCIONARIO( id, cpf, nome )  
DEPENDENTE( funcionario_id, numero, cpf, nome )  
            funcionario_id referencia FUNCIONARIO
```

Mapeamento ER para Modelo Relacional

Entidades e atributos

- Entidade fraca e relacionamento identificador:



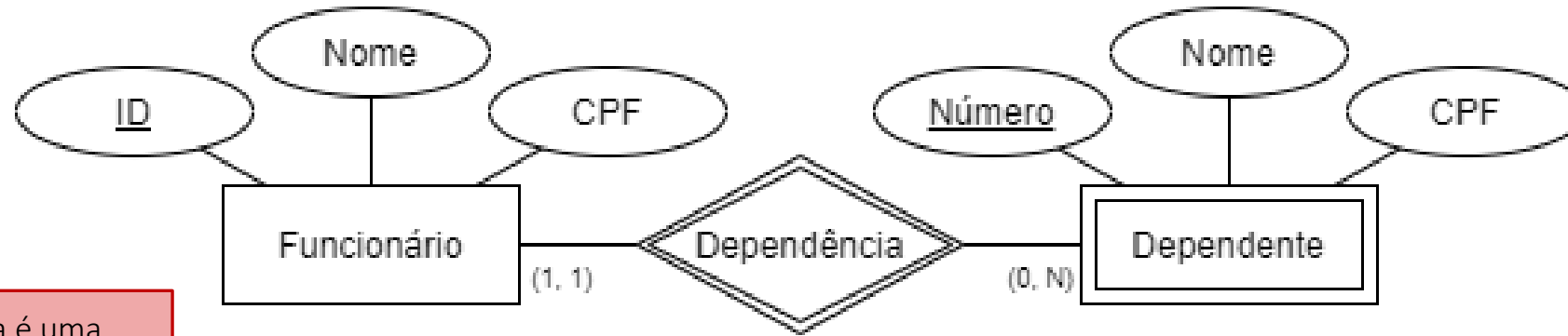
Parte da chave primária é uma chave estrangeira para a tabela da entidade forte conectada ao relacionamento identificador.

```
FUNCIONARIO( id, cpf, nome )
DEPENDENTE( funcionario_id, numero, cpf, nome )
            funcionario_id referencia FUNCIONARIO
```

Mapeamento ER para Modelo Relacional

Entidades e atributos

- Entidade fraca e relacionamento identificador:



Parte da chave primária é uma chave estrangeira para a tabela da entidade forte conectada ao relacionamento identificador.

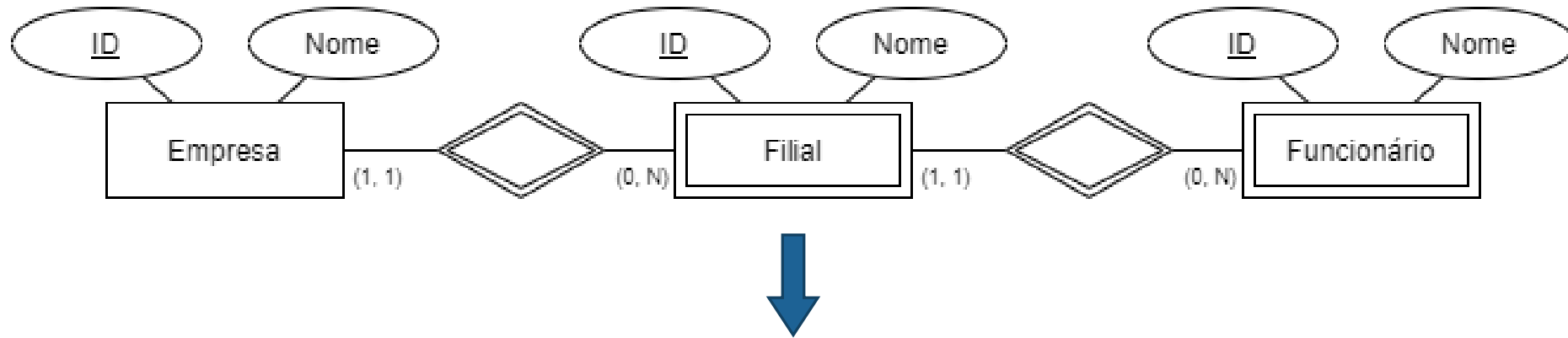
E a outra parte é a coluna derivada do atributo identificador da própria entidade fraca.

```
FUNCIONARIO( id, cpf, nome )  
DEPENDENTE( funcionario_id, numero, cpf, nome )  
funcionario_id referencia FUNCIONARIO
```

Mapeamento ER para Modelo Relacional

Entidades e atributos

- Entidade fraca e relacionamento identificador (recursão):

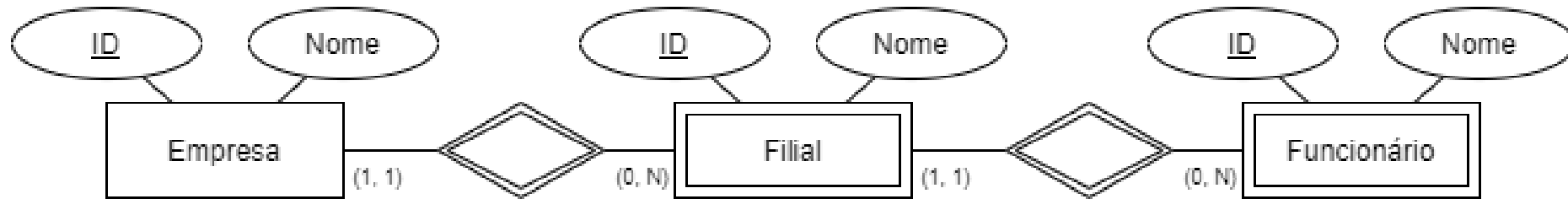


```
EMPRESA( id, nome )  
FILIAL( empresa_id, id, nome )  
        empresa_id referencia EMPRESA  
FUNCIONARIO( empresa_id, filial_id, id, nome )  
        (empresa_id, filial_id) referencia FILIAL
```

Mapeamento ER para Modelo Relacional

Entidades e atributos

- Entidade fraca e relacionamento identificador (recursão):



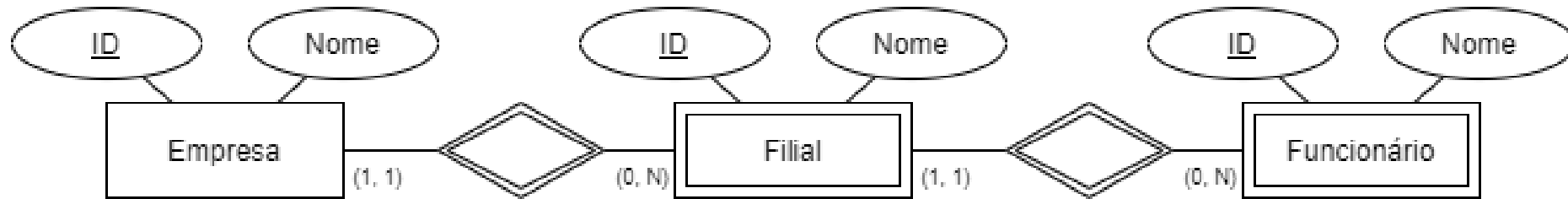
Parte da chave primária de **FILIAL** é uma chave estrangeira para a tabela da entidade forte que possui a filial, nesse caso, a tabela **EMPRESA**.

```
EMPRESA( id, nome )
FILIAL( empresa id, id, nome )
        empresa_id referencia EMPRESA
FUNCIONARIO( empresa id, filial id, id, nome )
            (empresa_id, filial_id) referencia FILIAL
```

Mapeamento ER para Modelo Relacional

Entidades e atributos

- Entidade fraca e relacionamento identificador (recursão):



```
EMPRESA( id, nome )
FILIAL( empresa_id, id, nome )
        empresa_id referencia EMPRESA
FUNCIONARIO( empresa_id, filial_id, id, nome )
        (empresa_id, filial_id) referencia FILIAL
```

Parte da chave primária de **FUNCIONARIO** é uma chave estrangeira para a tabela da entidade da qual a entidade depende, nesse caso, a tabela **FILIAL**.

Observe que o funcionário depende diretamente da filial e não da empresa. Por isso, a chave estrangeira é uma chave composta para a tabela **FILIAL** e não duas chaves estrangeiras (uma para **FILIAL** e outra para **EMPRESA**).

Mapeamento ER para Modelo Relacional

Entidades e atributos

- **Dicas para definição dos nomes de tabelas e colunas:**
 - **Tabelas:**
 - Nomes de uma tabela não podem conter caracteres em branco, caracteres acentuados e/ou caracteres especiais (com exceção do caractere de sublinhado “_”).
 - São frequentemente utilizados nos programas que utilizam um banco de dados. Portanto, nomes curtos diminuem o trabalho dos programadores.
 - Geralmente os nomes das tabelas são definidos em letras maiúsculas.
 - Às vezes, também é utilizado o prefixo “TB_” nos nomes das tabelas.
 - **Colunas:**
 - Também não podem possuir caracteres em branco, caracteres acentuados e/ou caracteres especiais (com exceção do caractere de sublinhado “_”).
 - Os nomes de colunas também são frequentemente utilizados nos programas. Portanto, nomes curtos diminuem o trabalho dos programadores.
 - Os nomes das colunas no modelo relacional não precisam ter os mesmos nomes dos atributos das entidades do DER.
 - Nomes grandes ou compostos podem ser abreviados, porém deve-se manter nomes que sejam significativos.
 - Para colunas que compõem chaves estrangeiras, é recomendável que seus nomes contenham o nome da tabela referenciada como prefixo (ou sufixo) seguido (ou precedido) do mesmo nome da coluna na tabela referenciada.
 - Geralmente, os nomes das colunas são definidos em letras minúsculas.

Mapeamento ER para Modelo Relacional

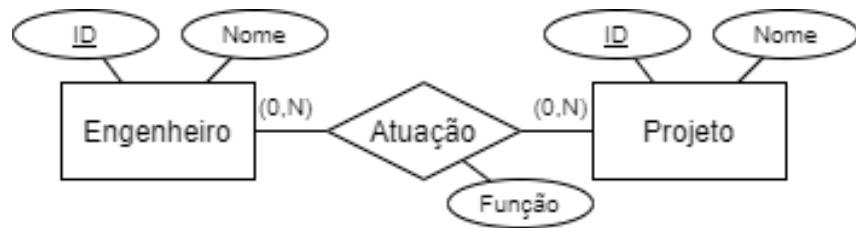
Relacionamentos

- Existem três alternativas para o mapeamento de relacionamentos do Modelo ER para o Modelo Relacional:
 - Tabela própria
 - Adição de colunas a uma das tabelas
 - Fusão de tabelas

Mapeamento ER para Modelo Relacional

Relacionamentos

- Existem três alternativas para o mapeamento de relacionamentos do Modelo ER para o Modelo Relacional:
 - Tabela própria**
 - Adição de colunas a uma das tabelas
 - Fusão de tabelas
- Exemplo de mapeamento por tabela própria:



```
engenheiro( id, nome )
projeto( id, nome )
```

Mapeamento das entidades.

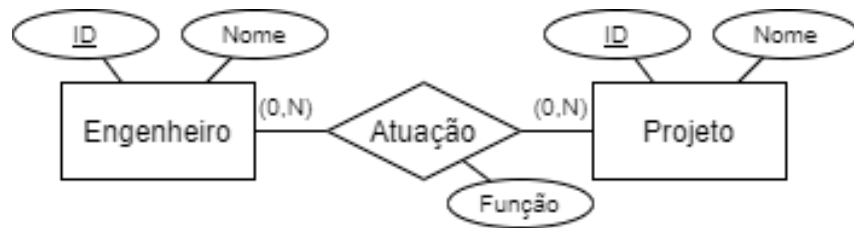
Mapeamento ER para Modelo Relacional

Relacionamentos

- Existem três alternativas para o mapeamento de relacionamentos do Modelo ER para o Modelo Relacional:

- Tabela própria**
- Adição de colunas a uma das tabelas
- Fusão de tabelas

- Exemplo de mapeamento por tabela própria:



```
engenheiro( id, nome )
projeto( id, nome )
```

```
atuacao( id_engenheiro, id_projeto, funcao )
         id_engenheiro referencia engenheiro
         id_projeto referencia projeto
```

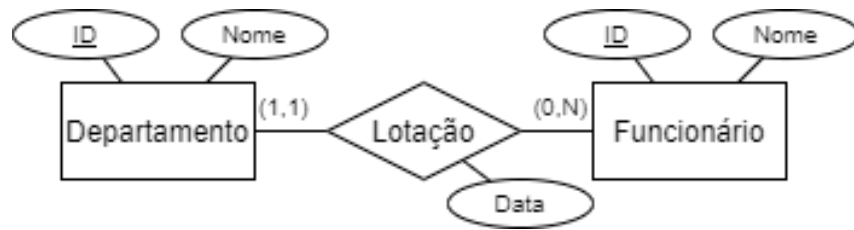
Mapeamento por tabela própria:

- É criada uma tabela.
- São definidas colunas para referenciar cada uma das tabelas (chave estrangeira).
- As colunas chave estrangeira são definidas como a chave primária.
- Atributos do relacionamento são transformados em colunas.

Mapeamento ER para Modelo Relacional

Relacionamentos

- Existem três alternativas para o mapeamento de relacionamentos do Modelo ER para o Modelo Relacional:
 - Tabela própria
 - Adição de colunas a uma das tabelas**
 - Fusão de tabelas
- Exemplo de mapeamento por adição de colunas:



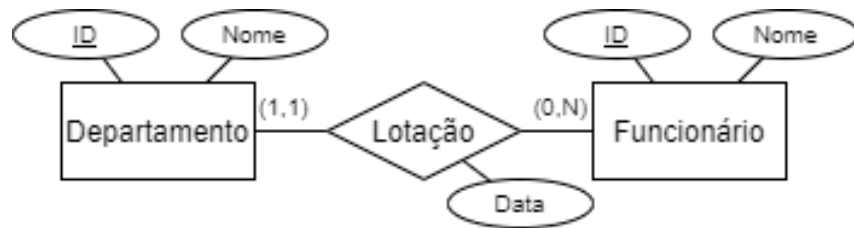
```
DEPARTAMENTO( id, nome )  
FUNCIONARIO( id, nome )
```

Mapeamento
das entidades.

Mapeamento ER para Modelo Relacional

Relacionamentos

- Existem três alternativas para o mapeamento de relacionamentos do Modelo ER para o Modelo Relacional:
 - Tabela própria
 - Adição de colunas a uma das tabelas**
 - Fusão de tabelas
- Exemplo de mapeamento por adição de colunas:



```
DEPARTAMENTO( id, nome )
```

```
FUNCIONARIO( id, nome, id_departamento, data )  
id_departamento referencia DEPARTAMENTO
```

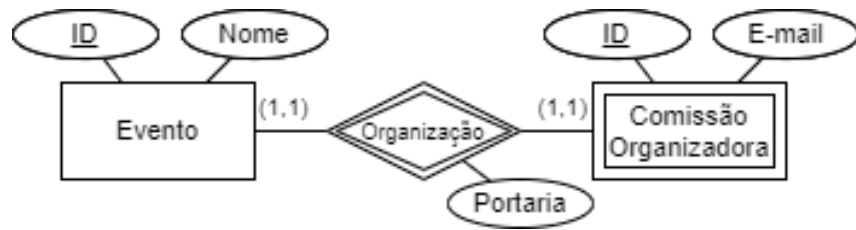
Mapeamento por adição de colunas:

- Na tabela da entidade passiva (o funcionário *é lotado* no departamento), é adicionada uma coluna como chave estrangeira para a outra tabela.
- São adicionadas as colunas dos atributos do relacionamento.

Mapeamento ER para Modelo Relacional

Relacionamentos

- Existem três alternativas para o mapeamento de relacionamentos do Modelo ER para o Modelo Relacional:
 - Tabela própria
 - Adição de colunas a uma das tabelas
 - Fusão de tabelas**
- Exemplo de mapeamento por fusão de tabelas:



```
EVENTO( id, nome )  
COMISSAO( id, email )
```

Mapeamento
das entidades.

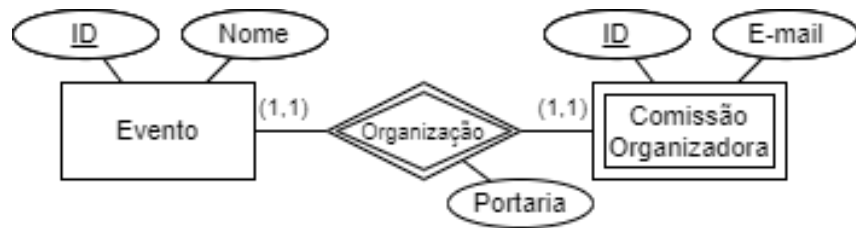
Mapeamento ER para Modelo Relacional

Relacionamentos

- Existem três alternativas para o mapeamento de relacionamentos do Modelo ER para o Modelo Relacional:

- Tabela própria
- Adição de colunas a uma das tabelas
- Fusão de tabelas**

- Exemplo de mapeamento por fusão de tabelas:



~~EVENTO(id, nome)~~
~~COMISSAO(id, email)~~

EVENTO(id, nome, email, portaria)

Mapeamento por fusão de tabelas:

- As tabelas das entidades envolvidas são eliminadas.
- Uma nova tabela é definida para conter os dados das duas entidades que tiveram as tabelas eliminadas.
- As colunas com os dados das tabelas eliminadas são inseridas na nova tabela.
- Os atributos do relacionamento são transformados em colunas na tabela.

Mapeamento ER para Modelo Relacional

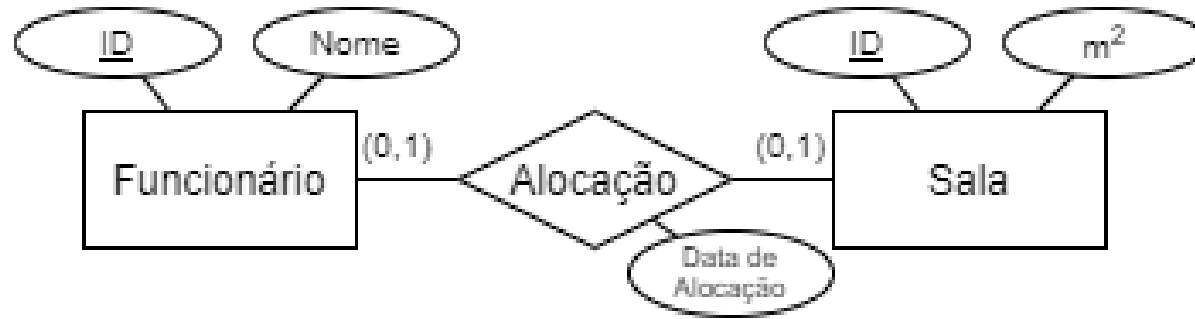
Relacionamentos

- Como vimos, existem três alternativas para o mapeamento de relacionamentos do Modelo ER para o Modelo Relacional:
 - Tabela própria
 - Adição de colunas a uma das tabelas
 - Fusão de tabelas
- A melhor alternativa para realizar o mapeamento de um relacionamento de um Modelo ER para o Modelo Relacional depende das cardinalidades (máxima e mínima) desse relacionamento.
 - Relacionamentos 1:1
 - Ambos opcionais
 - Um opcional, outro obrigatório
 - Ambos obrigatórios
 - Relacionamentos 1:N
 - Entidade de cardinalidade máxima 1 com cardinalidade mínima 0 (opcional)
 - Entidade de cardinalidade máxima 1 com cardinalidade mínima 1 (obrigatória)
 - Relacionamento N:N

Mapeamento ER para Modelo Relacional

Relacionamentos

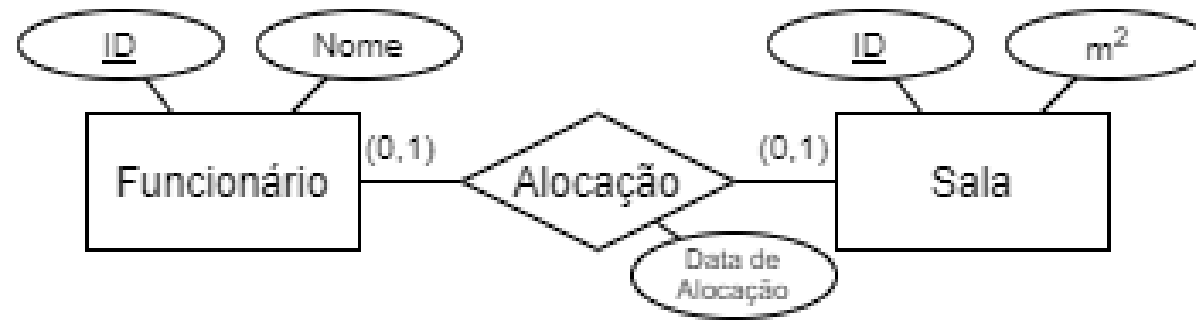
- Relacionamentos 1:1
 - Ambas entidades opcionais



Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:1
 - Ambas entidades opcionais



```
FUNCIONARIO( id, nome )
```

```
SALA( id, metros_quadrados, id_funcionario, data_alocacao )  
    id_funcionario referencia FUNCIONARIO
```

Mapeamento por adição de colunas

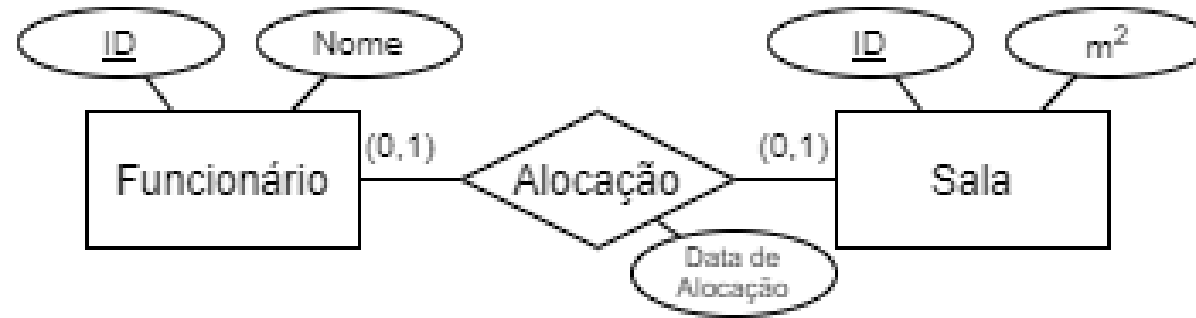
Qualquer uma das entidades poderiam ser escolhidas para a adição de colunas.



Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:1
 - Ambas entidades opcionais



```
FUNCIONARIO( id, nome )
```

```
SALA( id, metros_quadrados, id_funcionario, data_alocacao )  
      id_funcionario referencia FUNCIONARIO
```

Uma das entidades irá manter os atributos do relacionamento e a chave estrangeira para a outra entidade participante. Portanto, deve-se garantir que a chave estrangeira da tabela não se repita para garantir que a cardinalidade máxima seja 1. Dessa forma, ela deve ser definida, também, como uma chave alternativa.

Qualquer uma das entidades poderiam ser escolhidas para a adição de colunas.

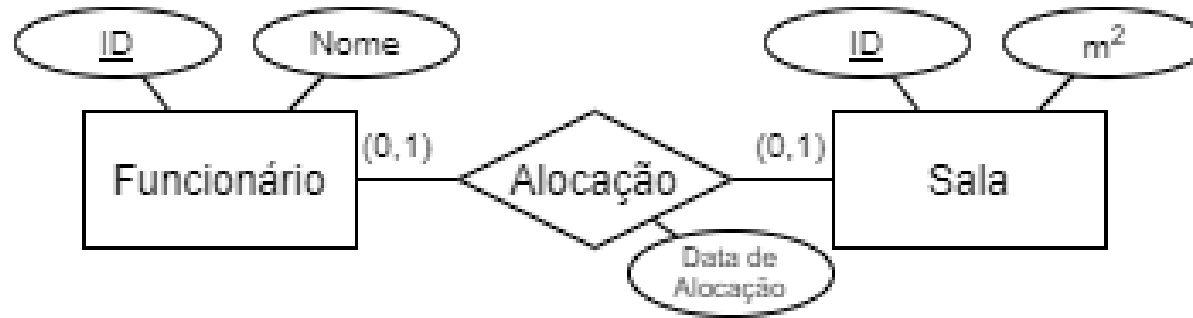
Mapeamento por adição de colunas



Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:1
 - Ambas entidades opcionais



```
FUNCIONARIO( id, nome )  
SALA( id, metros_quadrados )
```

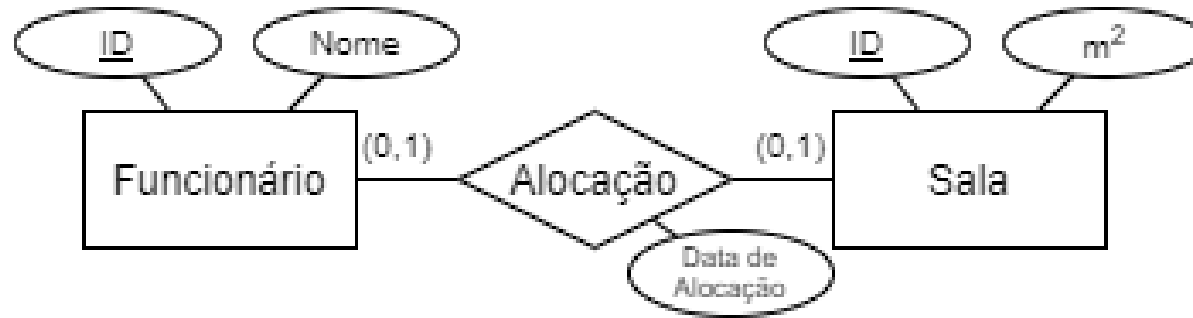
```
ALOCACAO( id_sala, id_funcionario, data_alocacao )  
    id_sala referencia SALA  
    id_funcionario referencia FUNCIONARIO
```

Mapeamento por tabela própria

Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:1
 - Ambas entidades opcionais



```
FUNCIONARIO( id, nome )  
SALA( id, metros_quadrados )
```

```
ALOCACAO( id_sala, id_funcionario, data_alocacao )  
id_sala referencia SALA  
id_funcionario referencia FUNCIONARIO
```

Deve-se garantir que, na tabela gerada pelo relacionamento, as colunas que referenciam as chaves primárias das entidades não tenham valores repetidos para garantir a cardinalidade máxima de 1. Portanto, cada uma dessas colunas devem ser definidas como chaves alternativas, separadamente.

Mapeamento por tabela própria

Mapeamento ER para Modelo Relacional

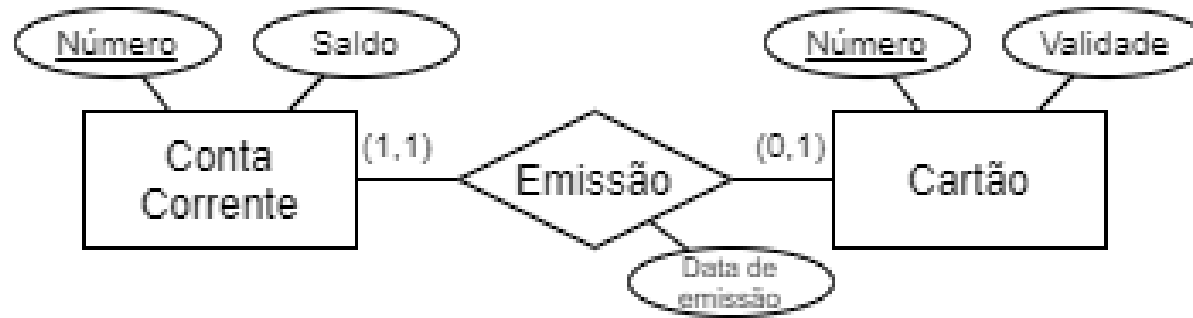
Relacionamentos

- **Relacionamentos 1:1**
 - **Ambas entidades opcionais**
 - Mapeamento por adição de colunas é preferido para essa situação.
 - Resulta em uma estrutura de tabelas com menor número de chaves primárias, necessitando menos junções de tabelas em consultas.
 - Mapeamento por tabela própria é aceitável.
 - No entanto, como o mapeamento é baseado em colunas opcionais, a verificação da opcionalidade de campos é do programador.
 - O mapeamento por fusão de tabelas é inviável.
 - Como a participação das duas entidades é opcional, tanto a coluna que armazena o identificador do funcionário quanto a coluna que armazena o identificador da sala devem ser opcionais no relacionamento. Assim, nenhuma dessas duas colunas poderiam ser utilizadas como chave primária.

Mapeamento ER para Modelo Relacional

Relacionamentos

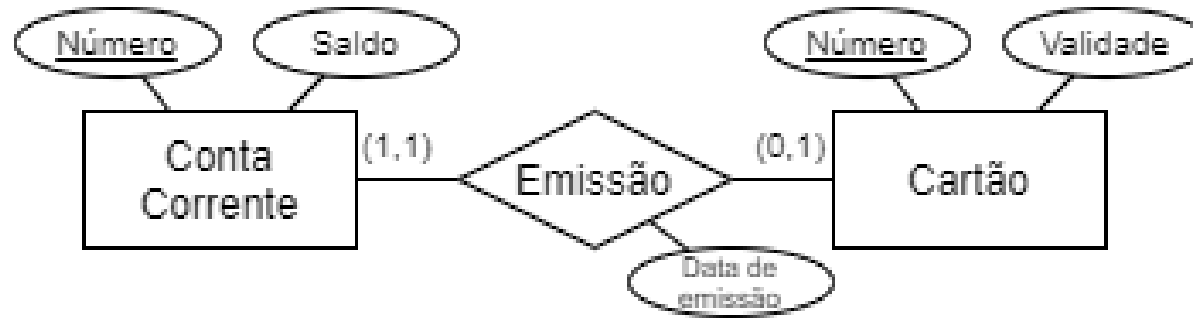
- Relacionamentos 1:1
 - Uma entidade opcional, outra obrigatória



Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:1
 - Uma entidade opcional, outra obrigatória



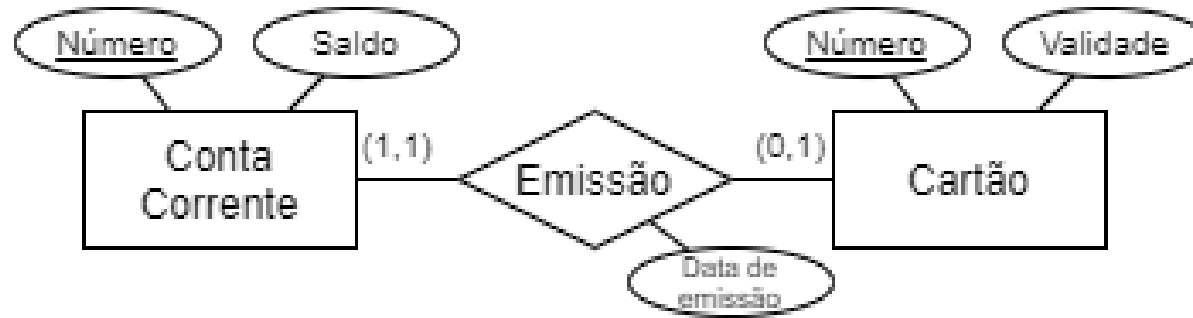
```
CONTA_CORRENTE( numero, saldo, cartao_numero, cartao_validade, data_emissao )
```

Mapeamento por fusão de tabelas

Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:1
 - Uma entidade opcional, outra obrigatória



CONTA_CORRENTE (numero, saldo, cartao_numero, cartao_validade, data_emissão)

Atributos da entidade
Conta Corrente.

Atributos da entidade
Cartão.

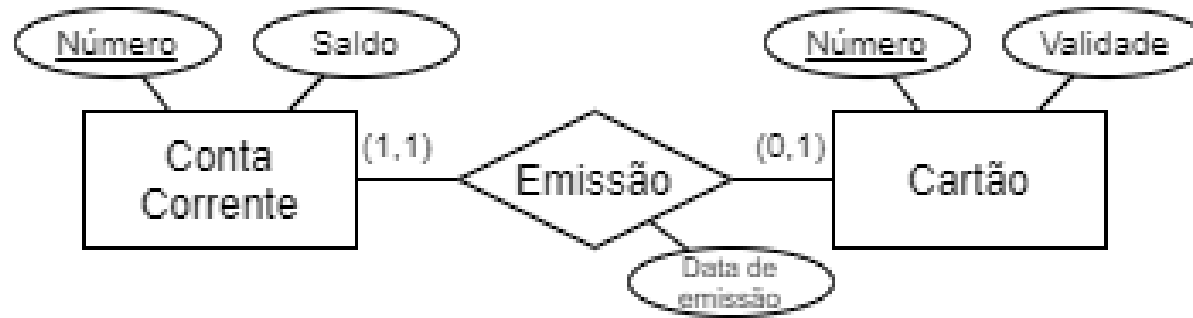
Atributos do
relacionamento.

Mapeamento por fusão de tabelas

Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:1
 - Uma entidade opcional, outra obrigatória



```
CONTA_CORRENTE( numero, saldo, cartao_numero, cartao_validade, data_emissao )
```

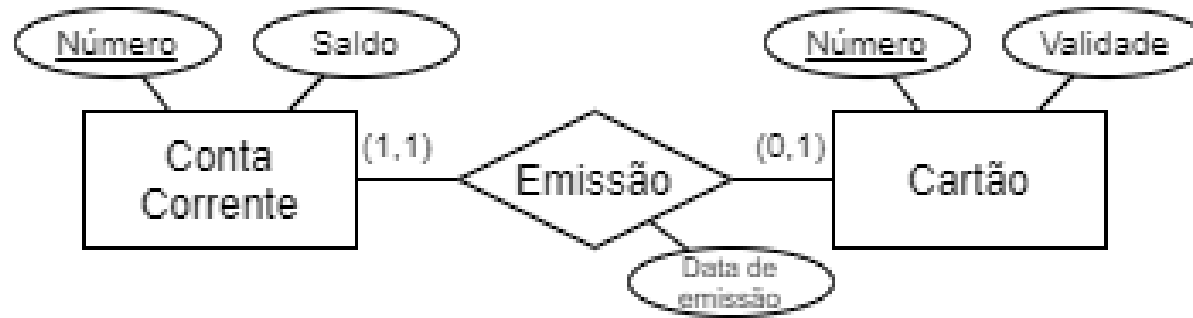
Deve-se garantir que as colunas referentes a atributos identificadores das entidades que foram fundidas na entidade que se tornou tabela não se repita para garantir a cardinalidade máxima de 1. Portanto, ela deve ser definida como chave alternativa.

Mapeamento por fusão de tabelas

Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:1
 - Uma entidade opcional, outra obrigatória



```
CONTA_CORRENTE( numero, saldo )
```

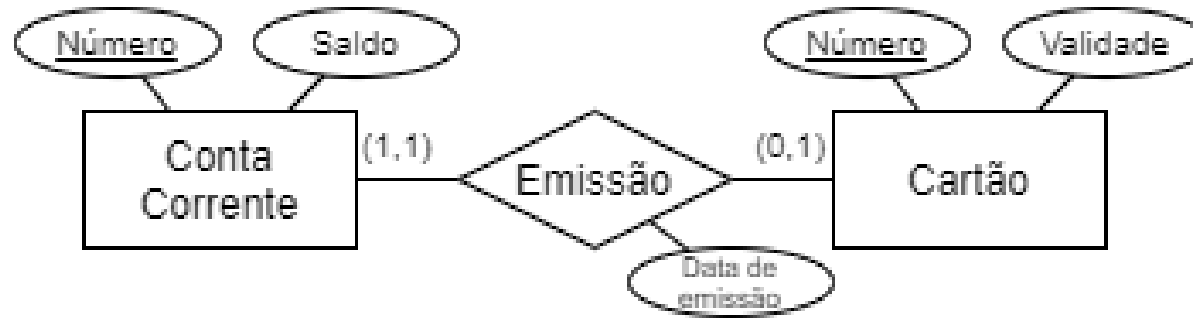
```
CARTAO(numero, validade, conta_corrente_numero, data_emissao )  
      conta_corrente_numero referencia CONTA_CORRENTE
```

Mapeamento por adição de colunas

Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:1
 - Uma entidade opcional, outra obrigatória



```
CONTA_CORRENTE( numero, saldo )
```

```
CARTAO(numero, validade, conta_corrente_numero, data_emissao )  
      conta_corrente_numero referencia CONTA_CORRENTE
```

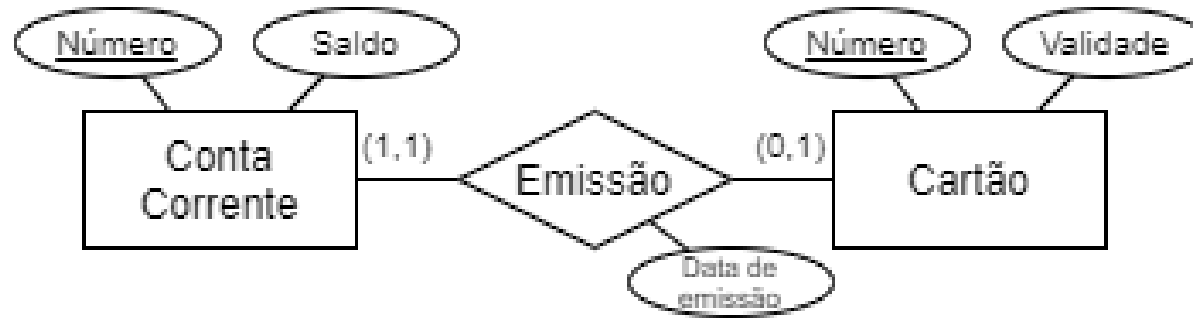
Uma das entidades irá manter os atributos do relacionamento e a chave estrangeira para a outra entidade participante. Portanto, deve-se garantir que a chave estrangeira da tabela não se repita para garantir que a cardinalidade máxima seja 1. Dessa forma, ela deve ser definida, também, como uma chave alternativa.

Mapeamento por adição de colunas

Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:1
 - Uma entidade opcional, outra obrigatória



```
CONTA_CORRENTE( numero, saldo )
```

```
CARTAO( numero, validade )
```

```
CARTAO_CONTA( cartao numero, conta corrente numero, data_emissao )
```

```
cartao_numero referencia CARTAO
```

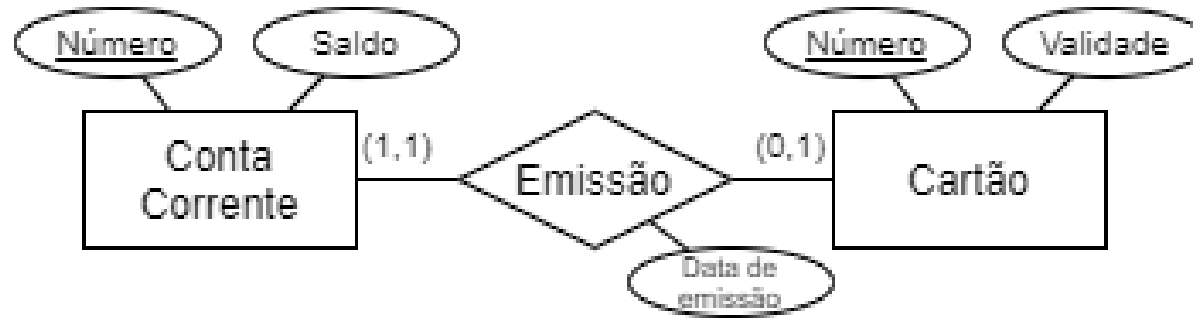
```
conta_corrente_numero referencia CONTA
```

Mapeamento por tabela própria

Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:1
 - Uma entidade opcional, outra obrigatória



```
CONTA_CORRENTE( numero, saldo )
```

```
CARTAO( numero, validade )
```

```
CARTAO_CONTA( cartao numero, conta corrente numero, data_emissao )  
cartao_numero referencia CARTAO  
conta_corrente_numero referencia CONTA
```

Deve-se garantir que, na tabela gerada pelo relacionamento, as colunas que referenciam as chaves primárias das entidades não tenham valores repetidos para garantir a cardinalidade máxima de 1. Portanto, cada uma dessas colunas devem ser definidas como chaves alternativas, separadamente.

Mapeamento por tabela própria

Mapeamento ER para Modelo Relacional

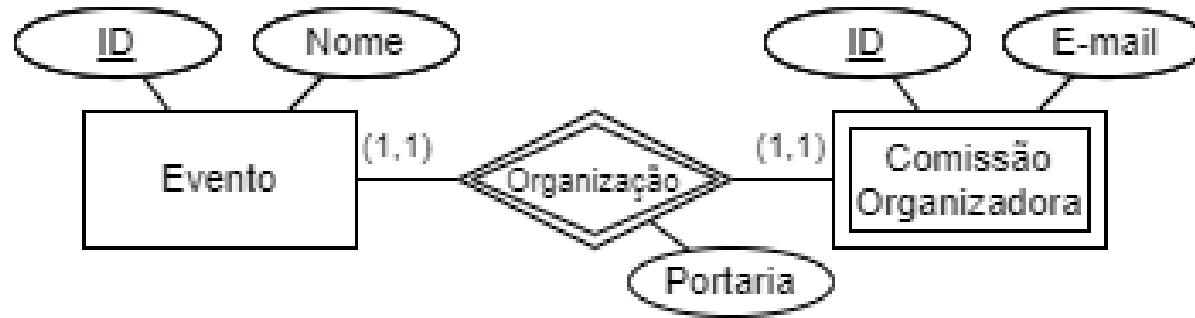
Relacionamentos

- **Relacionamentos 1:1**
 - Uma entidade opcional, outra obrigatória
 - Mapeamento por fusão de tabelas é a alternativa preferida.
 - Ela garante a integridade dos dados em relação à cardinalidade e requer um menor número de junções em consultas.
 - Mapeamento por adição de colunas é aceitável.
 - Essa alternativa é melhor em termos de campos opcionais. Porém, na implementação do banco de dados, deve-se especificar que a chave estrangeira não se repita para garantir a cardinalidade máxima de 1.
 - Mapeamento por tabela própria é a pior alternativa.
 - Necessita que um número maior de índices seja criado e, também, um maior número de junções em consultas.

Mapeamento ER para Modelo Relacional

Relacionamentos

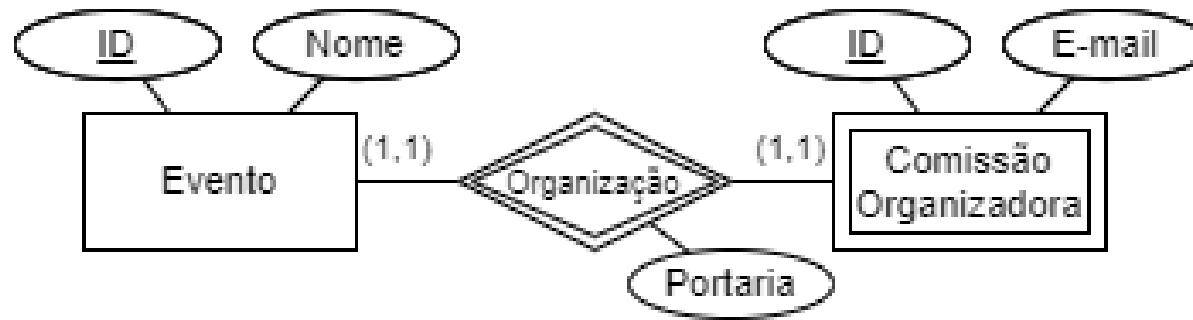
- Relacionamentos 1:1
 - Ambas entidades obrigatórias



Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:1
 - Ambas entidades obrigatórias



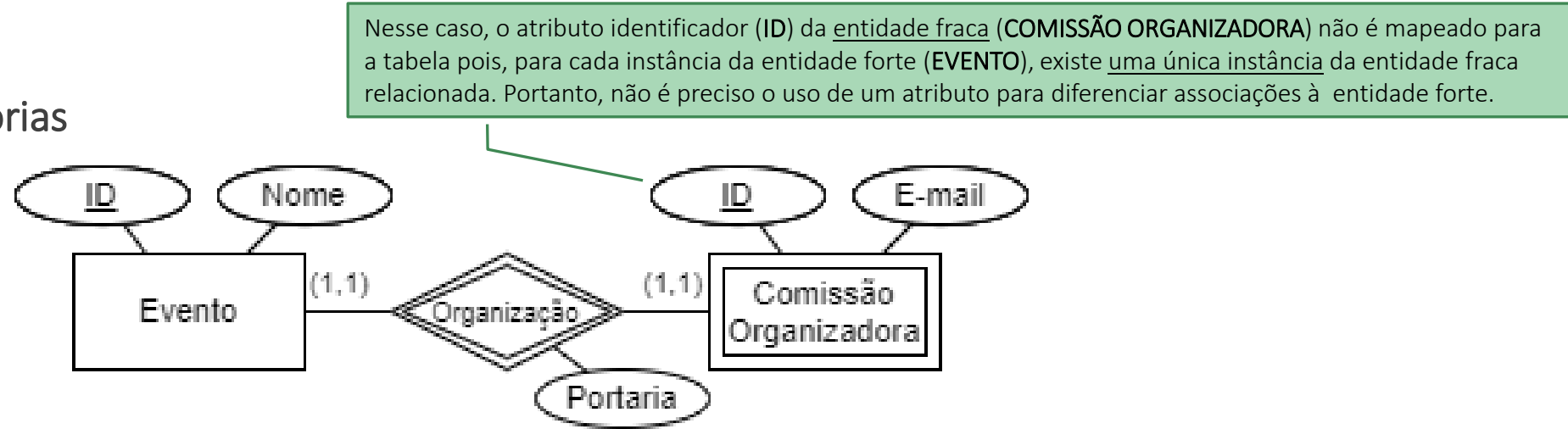
```
EVENTO( id, nome, email, portaria )
```

Mapeamento por fusão
de tabelas

Mapeamento ER para Modelo Relacional

Relacionamentos

- **Relacionamentos 1:1**
 - Ambas entidades obrigatórias



```
EVENTO( id, nome, email, portaria )
```

Mapeamento por fusão
de tabelas

Mapeamento ER para Modelo Relacional

Relacionamentos

- **Relacionamentos 1:1**
 - **Ambas entidades obrigatórias**
- As demais alternativas (mapeamentos por adição de colunas ou tabela própria) também atenderiam plenamente as necessidades de um relacionamento 1:1 com participação obrigatórias de ambas entidades.
- Entidades que participam do relacionamento seriam representadas por duas tabelas distintas, com ambas tabelas tendo o mesmo número de chave primária, e portanto, uma relação um-para-um entre suas linhas.
 - Essa estratégia resulta em uma estrutura com maior número de chaves primárias e, também, maior número de junções em consultas.

Mapeamento ER para Modelo Relacional

Relacionamentos

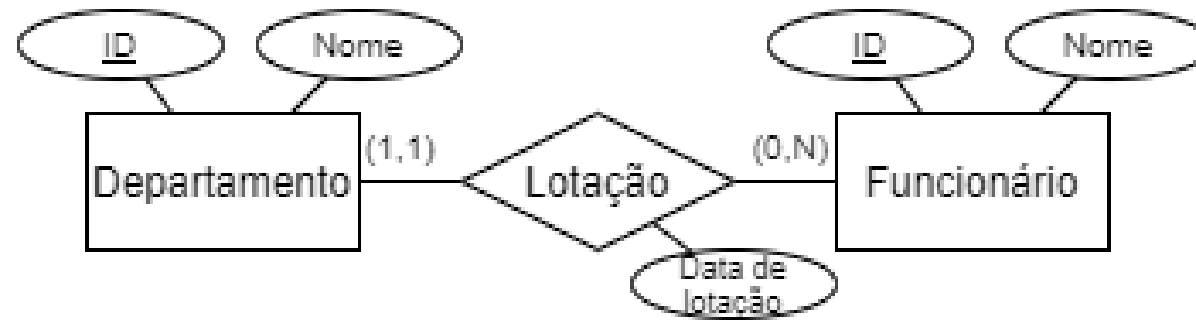
- Relacionamentos 1:1

Relacionamento	Alternativas para o mapeamento		
	Tabela Própria	Adição de Colunas	Fusão de Tabelas
Ambas opcionais	2ª opção	<i>Preferível</i>	<i>Não aplicável</i>
Opcional/Obrigatória	3ª opção	2ª opção	<i>Preferível</i>
Ambas obrigatórias	2ª opção	2ª opção	<i>Preferível</i>

Mapeamento ER para Modelo Relacional

Relacionamentos

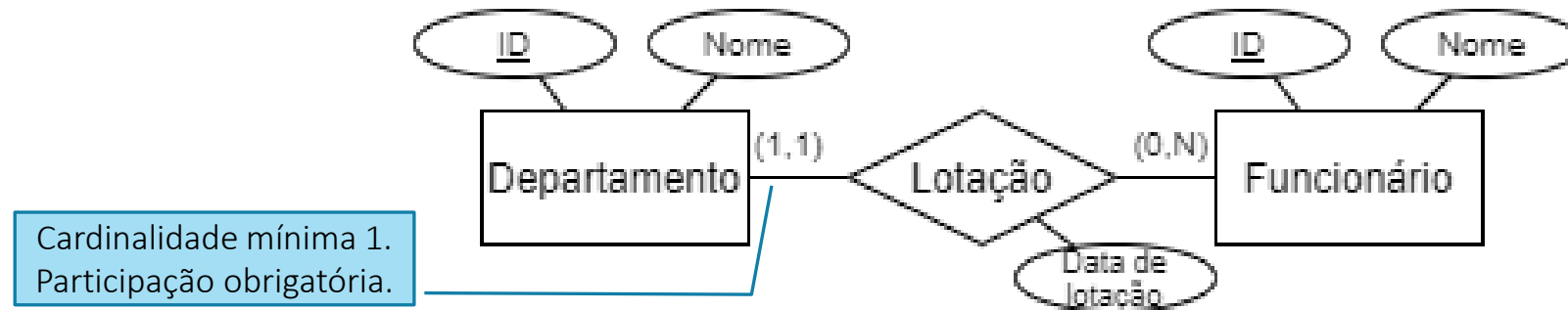
- Relacionamentos 1:N
 - Caso 1: Entidade de cardinalidade máxima 1 tem a cardinalidade mínima 1 (participação obrigatória).
 - Exemplo 1 (sem entidades fracas envolvidas no relacionamento)



Mapeamento ER para Modelo Relacional

Relacionamentos

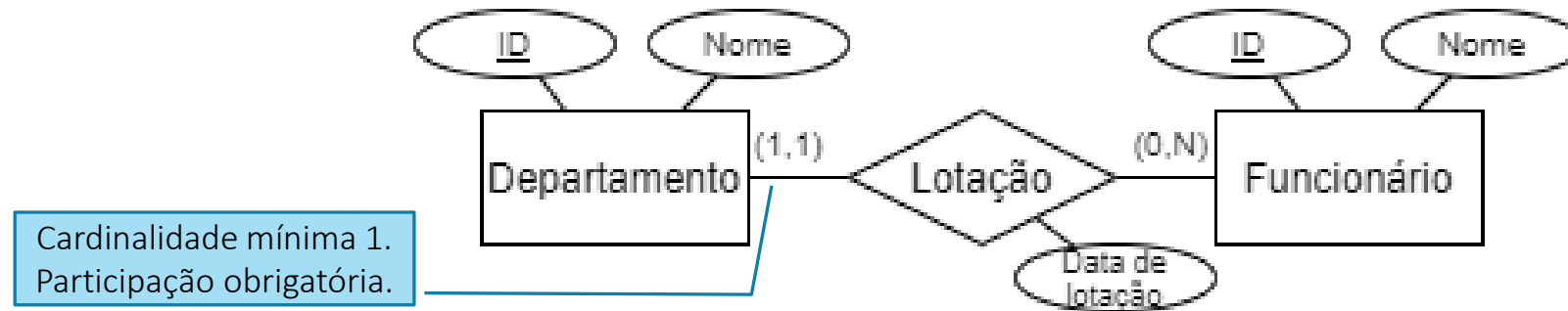
- Relacionamentos 1:N
 - Caso 1: Entidade de cardinalidade máxima 1 tem a cardinalidade mínima 1 (participação obrigatória).
 - Exemplo 1 (sem entidades fracas envolvidas no relacionamento)



Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:N
 - Caso 1: Entidade de cardinalidade máxima 1 tem a cardinalidade mínima 1 (participação obrigatória).
 - Exemplo 1 (sem entidades fracas envolvidas no relacionamento)



```
DEPARTAMENTO( id, nome )
```

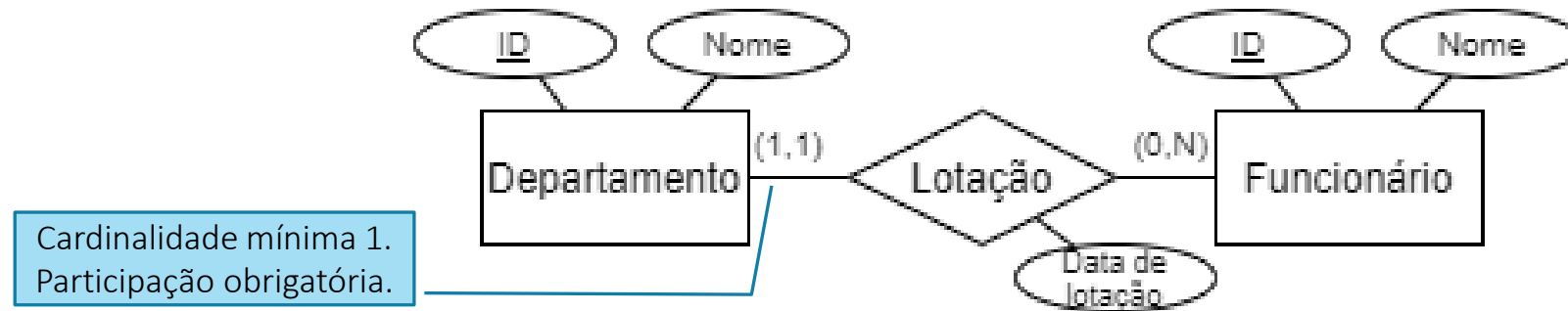
```
FUNCIONARIO( id, nome, departamento_id, data_lotação )  
    departamento_id referencia DEPARTAMENTO
```

Mapeamento por adição
de colunas

Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:N
 - Caso 1: Entidade de cardinalidade máxima 1 tem a cardinalidade mínima 1 (participação obrigatória).
 - Exemplo 1 (sem entidades fracas envolvidas no relacionamento)



```
DEPARTAMENTO( id, nome )
```

```
FUNCIONARIO( id, nome )
```

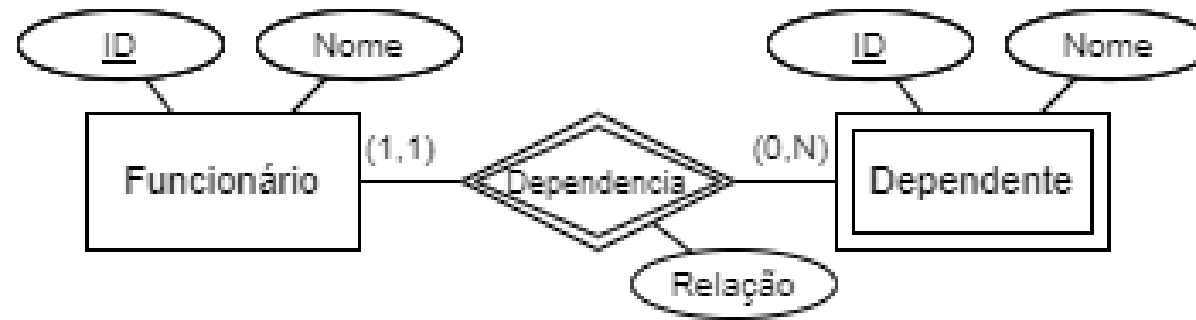
```
LOTACAO( funcionario_id, departamento_id, data_lotacao )  
funcionario_id referencia FUNCIONARIO  
departamento_id referencia DEPARTAMENTO
```

Mapeamento por
tabela própria

Mapeamento ER para Modelo Relacional

Relacionamentos

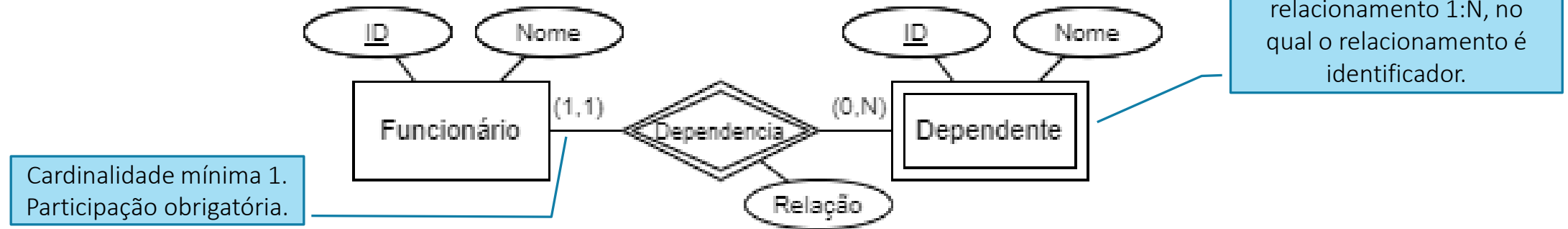
- Relacionamentos 1:N
 - Caso 1: Entidade de cardinalidade máxima 1 tem a cardinalidade mínima 1 (participação obrigatória).
 - Exemplo 2 (com entidades fracas envolvidas no relacionamento)



Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:N
 - Caso 1: Entidade de cardinalidade máxima 1 tem a cardinalidade mínima 1 (participação obrigatória).
 - Exemplo 2 (com entidades fracas envolvidas no relacionamento)

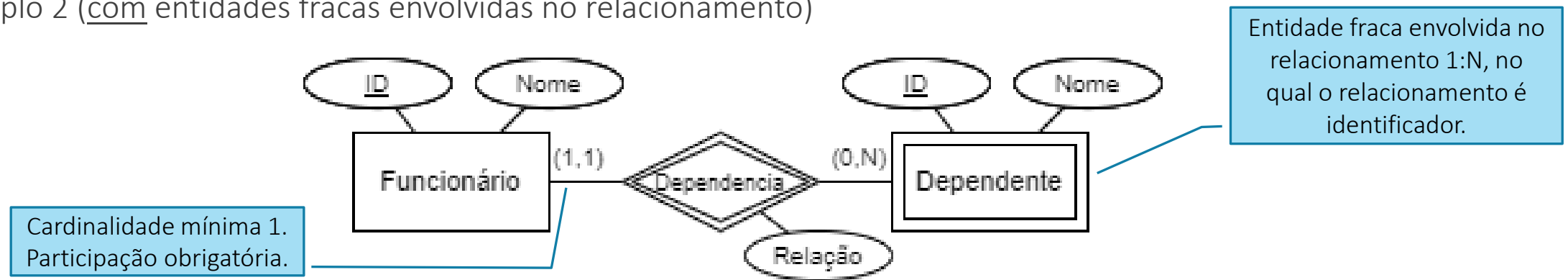


Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:N

- Caso 1: Entidade de cardinalidade máxima 1 tem a cardinalidade mínima 1 (participação obrigatória).
 - Exemplo 2 (com entidades fracas envolvidas no relacionamento)



```
FUNCIONARIO( id, nome )  
DEPENDENTE( funcionario_id, id, nome, relacao )  
    funcionario_id referencia FUNCIONARIO
```

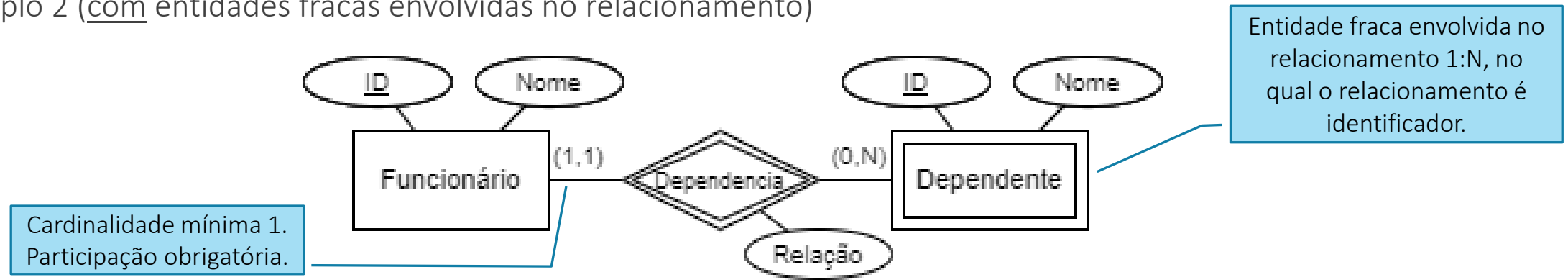
Mapeamento por adição de colunas

Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:N

- Caso 1: Entidade de cardinalidade máxima 1 tem a cardinalidade mínima 1 (participação obrigatória).
 - Exemplo 2 (com entidades fracas envolvidas no relacionamento)



```
FUNCIONARIO( id, nome )  
DEPENDENTE( funcionario_id, id, nome, relacao )  
funcionario_id referencia FUNCIONARIO
```

A coluna **funcionario_id**, que é chave estrangeira para a tabela da entidade forte do relacionamento, deve fazer parte da chave primária, pois a entidade **DEPENDENTE** é uma entidade fraca.

Mapeamento por adição de colunas

Mapeamento ER para Modelo Relacional

Relacionamentos

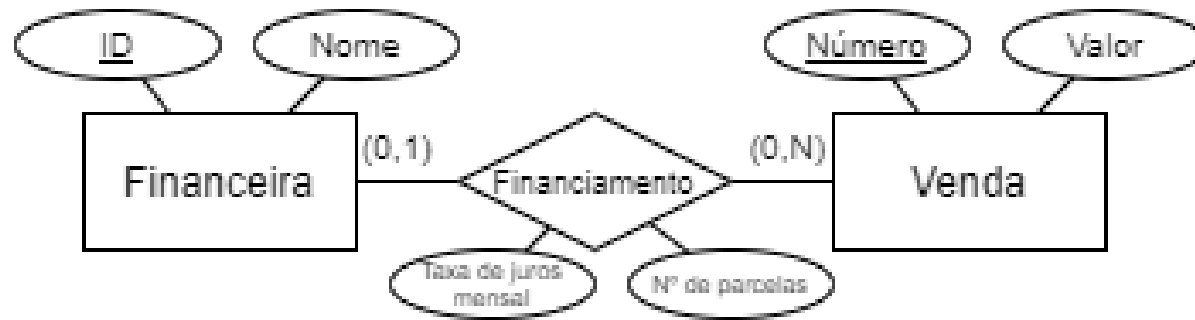
- **Relacionamentos 1:N**

- **Caso 1: Entidade de cardinalidade máxima 1 tem a cardinalidade mínima 1 (participação obrigatória).**
 - O mapeamento por adição de colunas é a estratégia preferível, pois requer a criação de um menor número de chaves e, também, um menor número de junções quando consultas forem realizadas no banco de dados.
 - O mapeamento por tabela própria pode ser utilizado, mas somente nos casos em que não se tem uma entidade fraca envolvida no relacionamento.
 - No entanto, essa estratégia, ao contrário da adição de colunas, requer a criação de mais chaves e um maior número de junções.
 - O mapeamento por fusão de tabelas não é aplicável, pois implicaria em redundância de dados ou tabelas aninhadas.

Mapeamento ER para Modelo Relacional

Relacionamentos

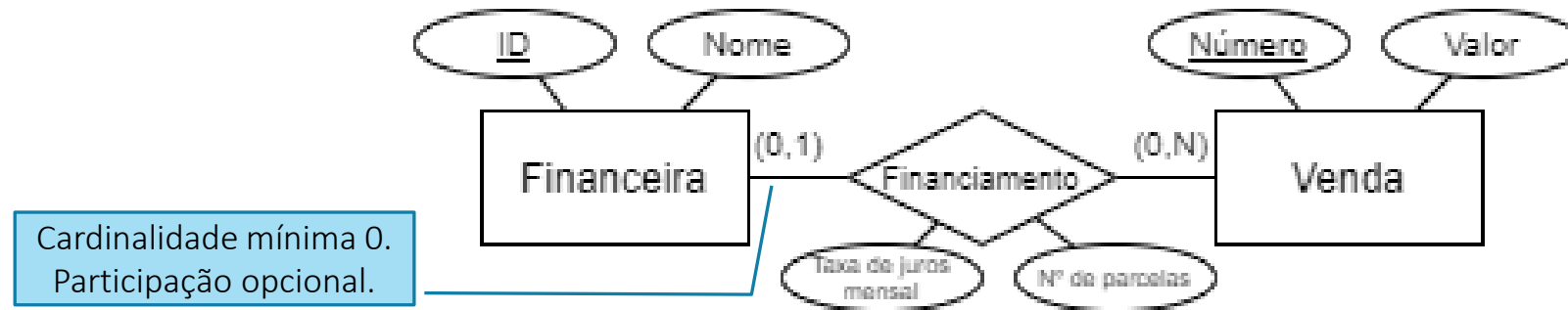
- Relacionamentos 1:N
 - Caso 2: Entidade de cardinalidade máxima 1 tem a cardinalidade mínima 0 (participação opcional).



Mapeamento ER para Modelo Relacional

Relacionamentos

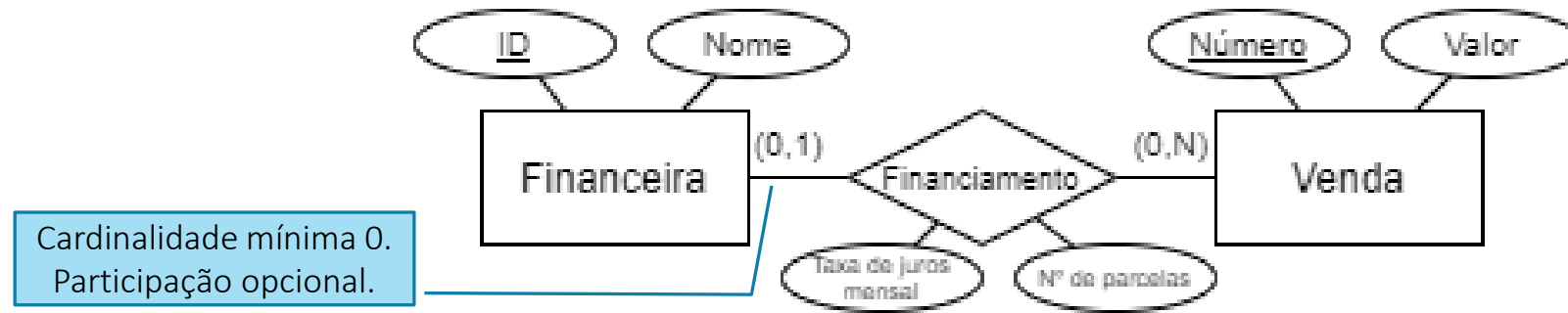
- Relacionamentos 1:N
 - Caso 2: Entidade de cardinalidade máxima 1 tem a cardinalidade mínima 0 (participação opcional).



Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:N
 - Caso 2: Entidade de cardinalidade máxima 1 tem a cardinalidade mínima 0 (participação opcional).



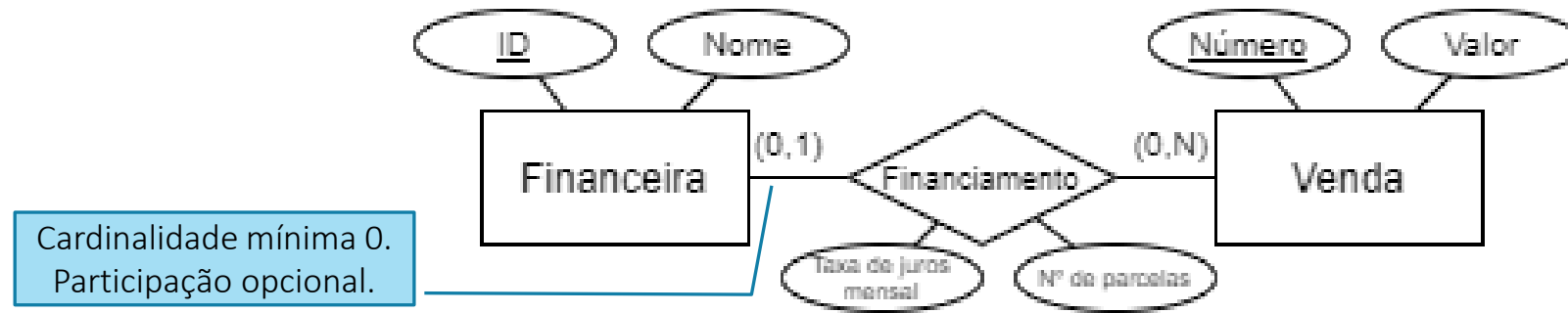
```
FINANCEIRA( id, nome )  
VENDA( numero, valor, financeira_id, numero_parcelas, taxa_juros_mes )  
    financeira_id referencia FINANCEIRA
```

Mapeamento por adição
de colunas

Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos 1:N
 - Caso 2: Entidade de cardinalidade máxima 1 tem a cardinalidade mínima 0 (participação opcional).



```
FINANCEIRA( id, nome )
```

```
VENDA( numero, valor )
```

```
FINANCIAMENTO( venda numero, financeira_id, numero_parcelas, taxa_juros_mes )
```

```
    venda_numero referencia VENDA
```

```
    financeira_id referencia FINANCEIRA
```

Mapeamento por
tabela própria

Mapeamento ER para Modelo Relacional

Relacionamentos

- **Relacionamentos 1:N**

- Caso 2: Entidade de cardinalidade máxima 1 tem a cardinalidade mínima 0 (participação opcional).
 - O mapeamento por adição de colunas é o preferível, já que requer um menor número de chaves e, além disso, necessita de um menor número de junções em possíveis consultas realizadas no banco de dados.
 - O mapeamento por tabela própria é aceitável, sendo a melhor alternativa em relação a campos opcionais.
 - Entretanto, requer a criação de um maior número de chaves e de junções.
 - O mapeamento por fusão de tabelas não é aplicável neste caso.

Mapeamento ER para Modelo Relacional

Relacionamentos

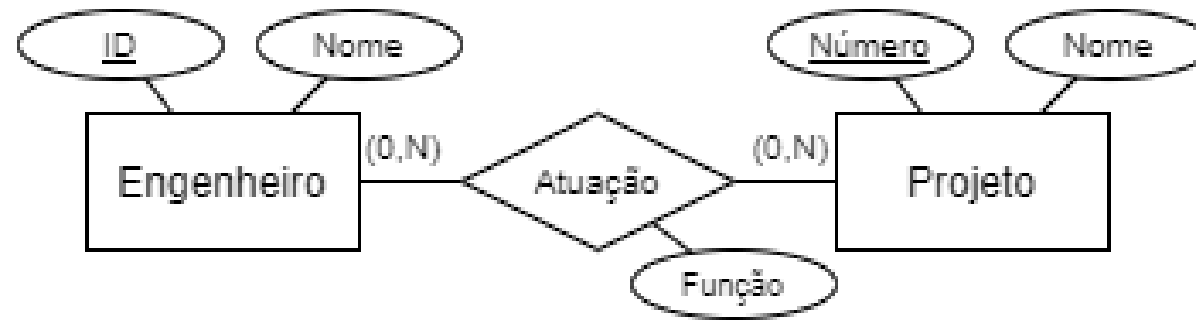
- Relacionamentos 1:N

Relacionamento	Alternativas para o mapeamento		
	Tabela Própria	Adição de Colunas	Fusão de Tabelas
Caso 1 <u>sem</u> entidade fraca	2ª opção	Preferível	Não aplicável
Caso 1 <u>com</u> entidade fraca	Não aplicável	Preferível	Não aplicável
Caso 2	2ª opção	Preferível	Não aplicável

Mapeamento ER para Modelo Relacional

Relacionamentos

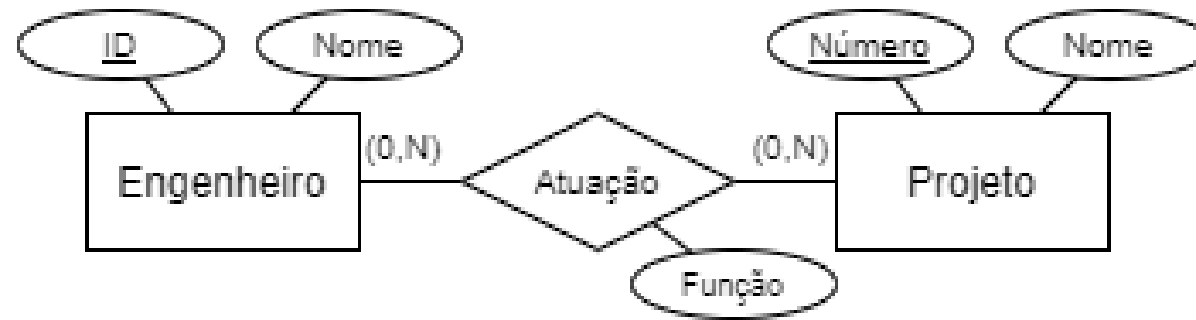
- Relacionamentos N:N



Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos N:N



```
engenheiro( id, nome )
```

```
projeto( numero, nome )
```

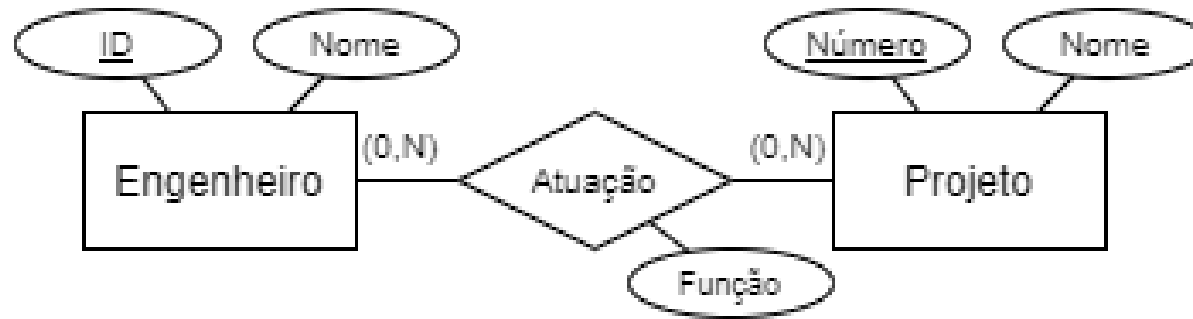
```
atuacao ( engenheiro id, projeto id, funcao )  
    engenheiro_id referencia engenheiro  
    projeto_numero referencia projeto
```

Mapeamento por
tabela própria

Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos N:N



```
engenheiro( id, nome )
```

```
projeto( numero, nome )
```

```
atuacao ( engenheiro id, projeto id, funcao )
```

```
engenheiro_id referencia engenheiro
```

```
projeto_numero referencia projeto
```

As duas colunas chaves estrangeiras devem compor a chave primária da tabela.

Mapeamento por
tabela própria

Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos N:N
 - Os mapeamentos por adição de colunas e fusão de tabelas não são aplicáveis a relacionamentos N para N, independente da cardinalidade mínima.

Relacionamento	Alternativas para o mapeamento		
	Tabela Própria	Adição de Colunas	Fusão de Tabelas
(_, N) : (_, N)	<i>Preferível</i>	<i>Não aplicável</i>	<i>Não aplicável</i>

Mapeamento ER para Modelo Relacional

Relacionamentos

Resumo das alternativas de mapeamento de acordo com as cardinalidades das entidades envolvidas.

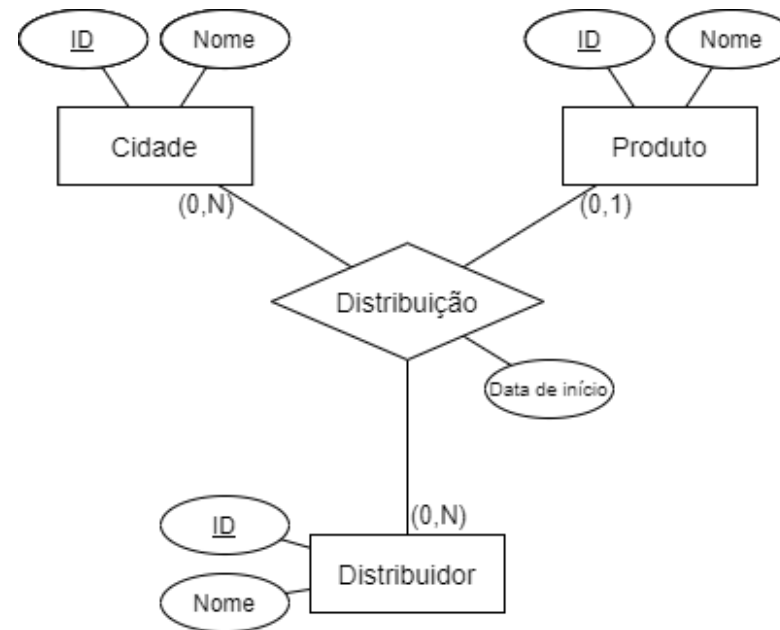
Relacionamento		Alternativas para o mapeamento		
Cardinalidades Máximas	Cardinalidades Mínimas	Tabela Própria	Adição de Colunas	Fusão de Tabelas
1:1	$(0, 1) : (0, 1)$ <i>ambas opcionais</i>	2ª opção	Preferível	Não aplicável
	$(0, 1) : (1, 1)$ <i>opcional/obrigatória</i>	3ª opção	2ª opção	Preferível
	$(1, 1) : (1, 1)$ <i>ambas obrigatórias</i>	2ª opção	2ª opção	Preferível
1:N	$(1, 1) : (_, N)$ <i>obrigatória, <u>sem</u> entidade fraca</i>	2ª opção	Preferível	Não aplicável
	$(1, 1) : (_, N)$ <i>obrigatória, <u>com</u> entidade fraca</i>	Não aplicável	Preferível	Não aplicável
	$(0, 1) : (_, N)$ <i>opcional</i>	2ª opção	Preferível	Não aplicável
N:N	$(_, N) : (_, N)$ <i>qualquer caso</i>	Preferível	Não aplicável	Não aplicável

Mapeamento ER para Modelo Relacional

Relacionamentos

- **Relacionamentos com grau maior que 2**

- Para relacionamentos de grau maior que dois, primeiro é feita uma transformação no DER para uma forma equivalente na qual o relacionamento é transformado em uma entidade.
- Em seguida, as regras de mapeamento de relacionamentos binários (vistas anteriormente) são aplicadas ao DER resultante da transformação.

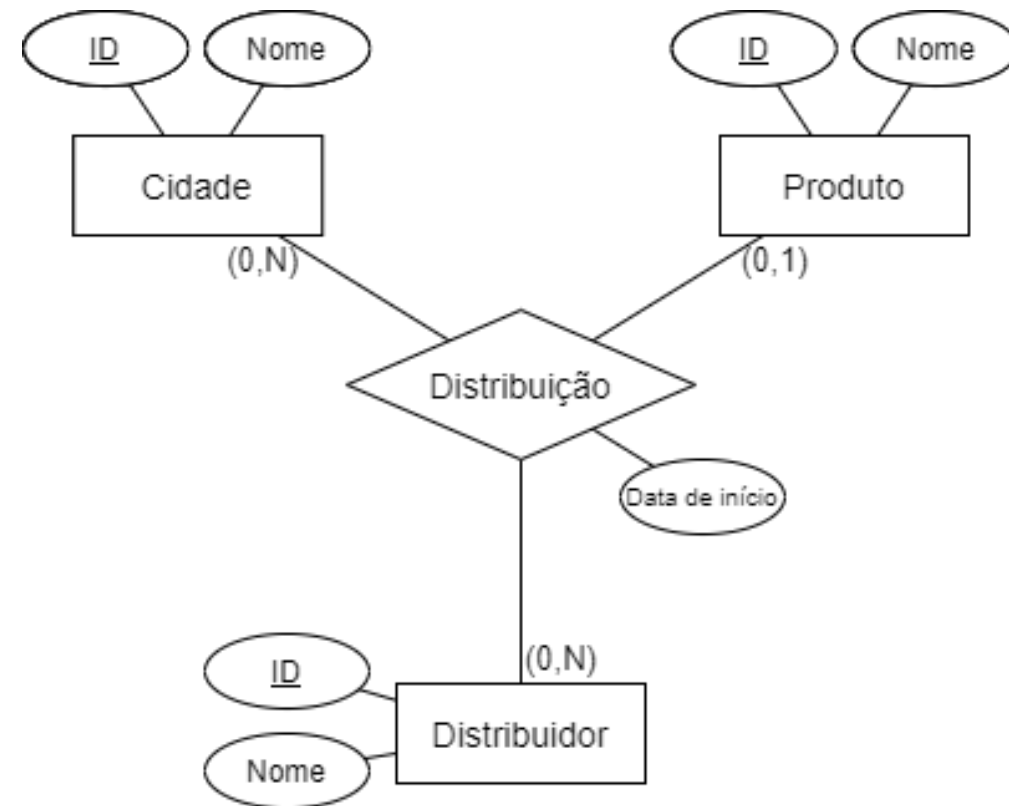
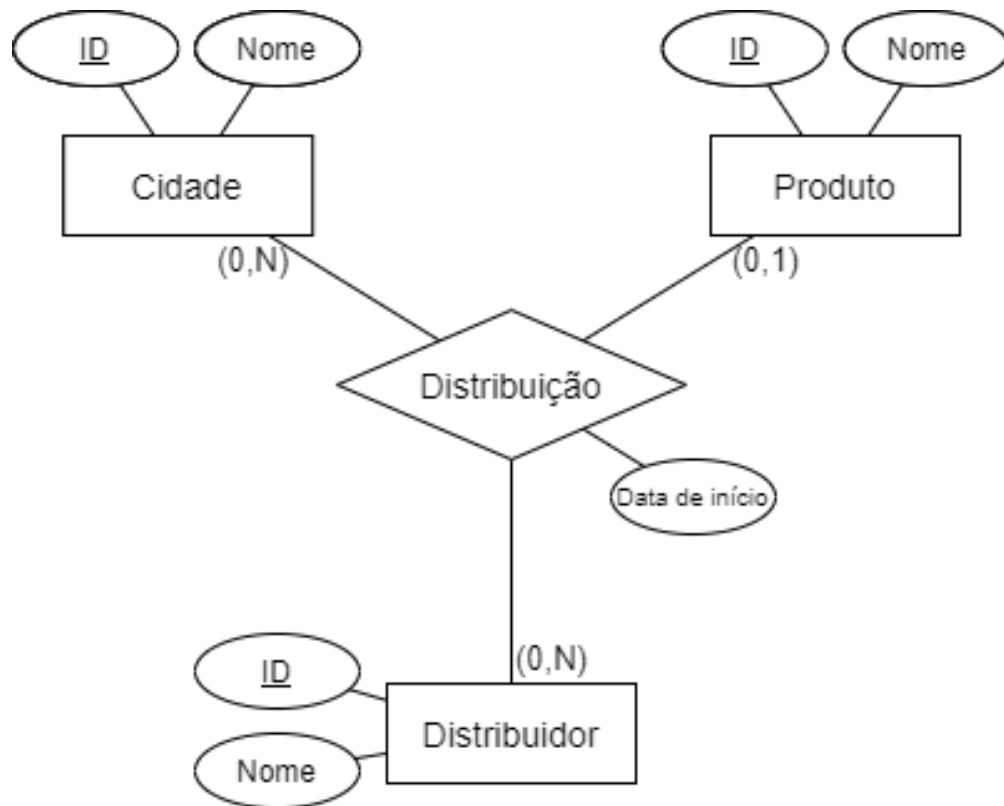


Mapeamento ER para Modelo Relacional

Relacionamentos

- **Relacionamentos com grau maior que 2**

- Transformação do relacionamento ternário para um equivalente com apenas relacionamentos binários.

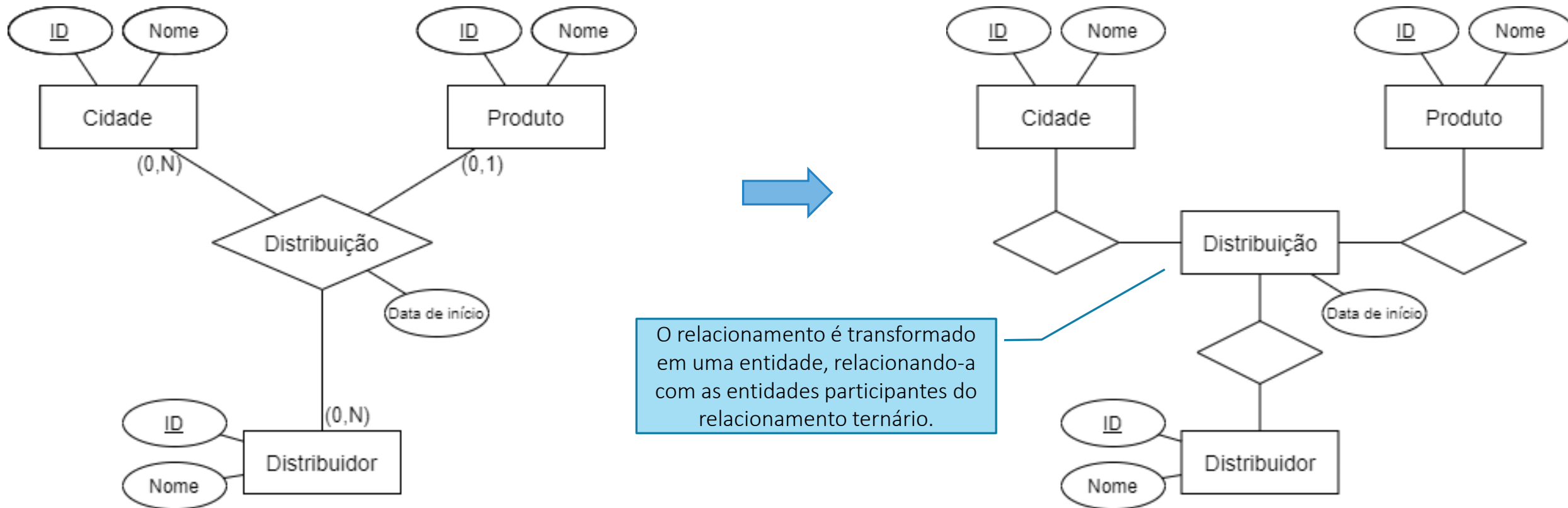


Mapeamento ER para Modelo Relacional

Relacionamentos

- **Relacionamentos com grau maior que 2**

- Transformação do relacionamento ternário para um equivalente com apenas relacionamentos binários.

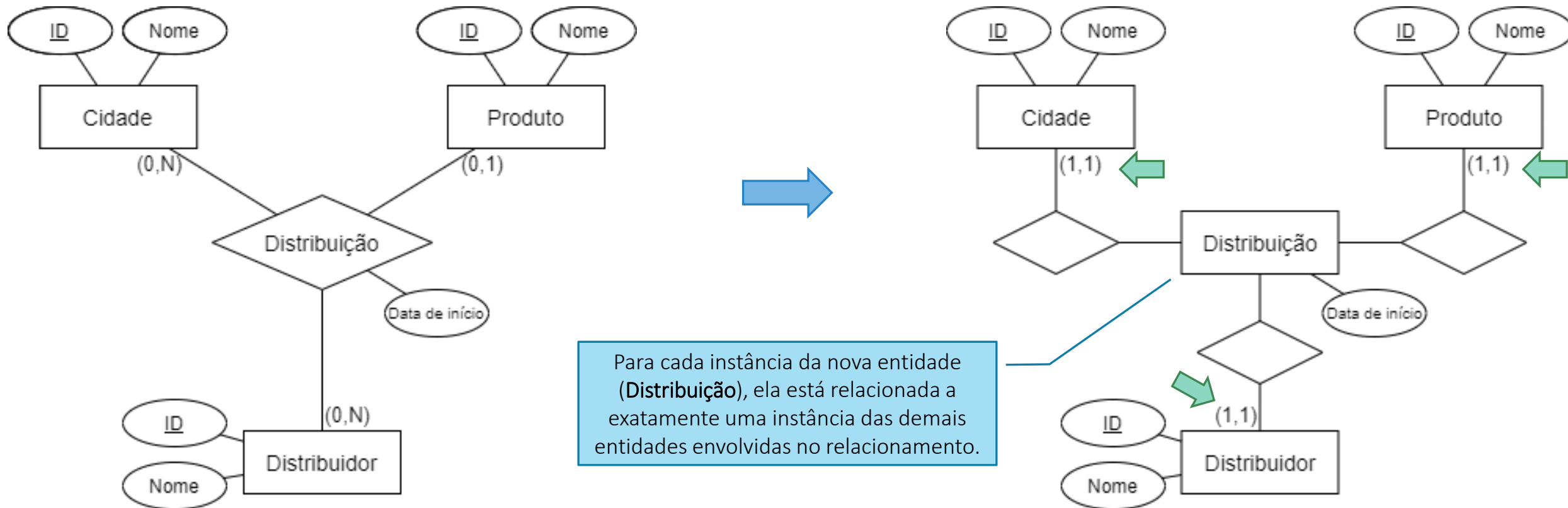


Mapeamento ER para Modelo Relacional

Relacionamentos

- **Relacionamentos com grau maior que 2**

- Transformação do relacionamento ternário para um equivalente com apenas relacionamentos binários.

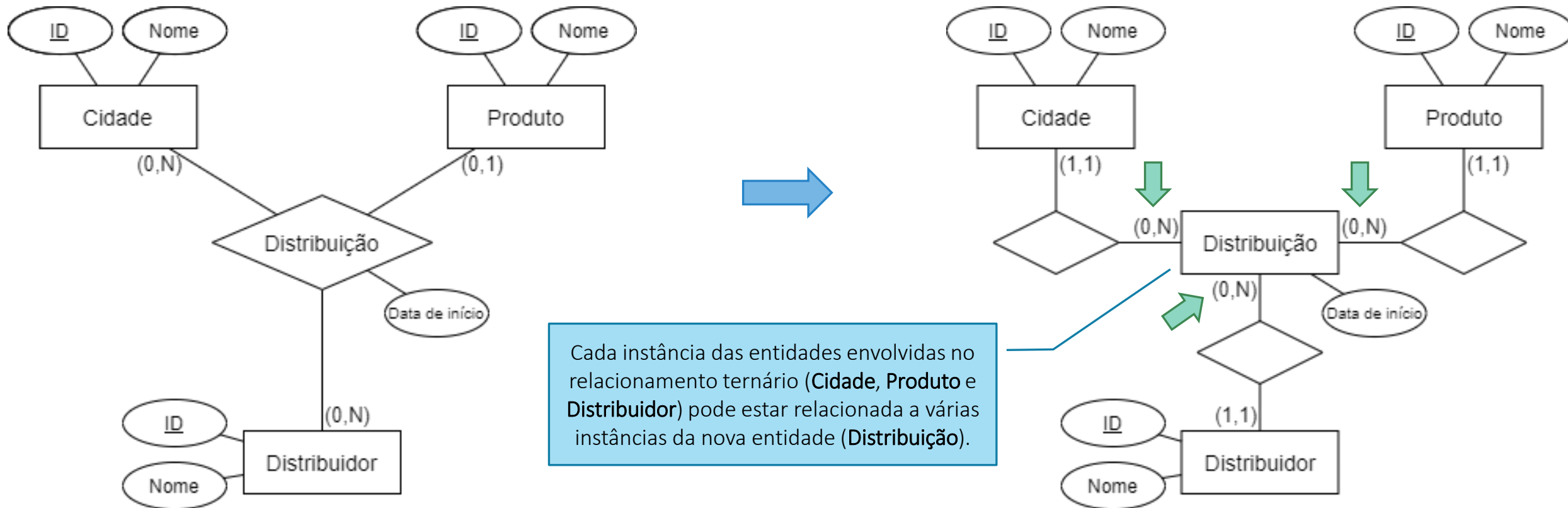


Mapeamento ER para Modelo Relacional

Relacionamentos

- **Relacionamentos com grau maior que 2**

- Transformação do relacionamento ternário para um equivalente com apenas relacionamentos binários.

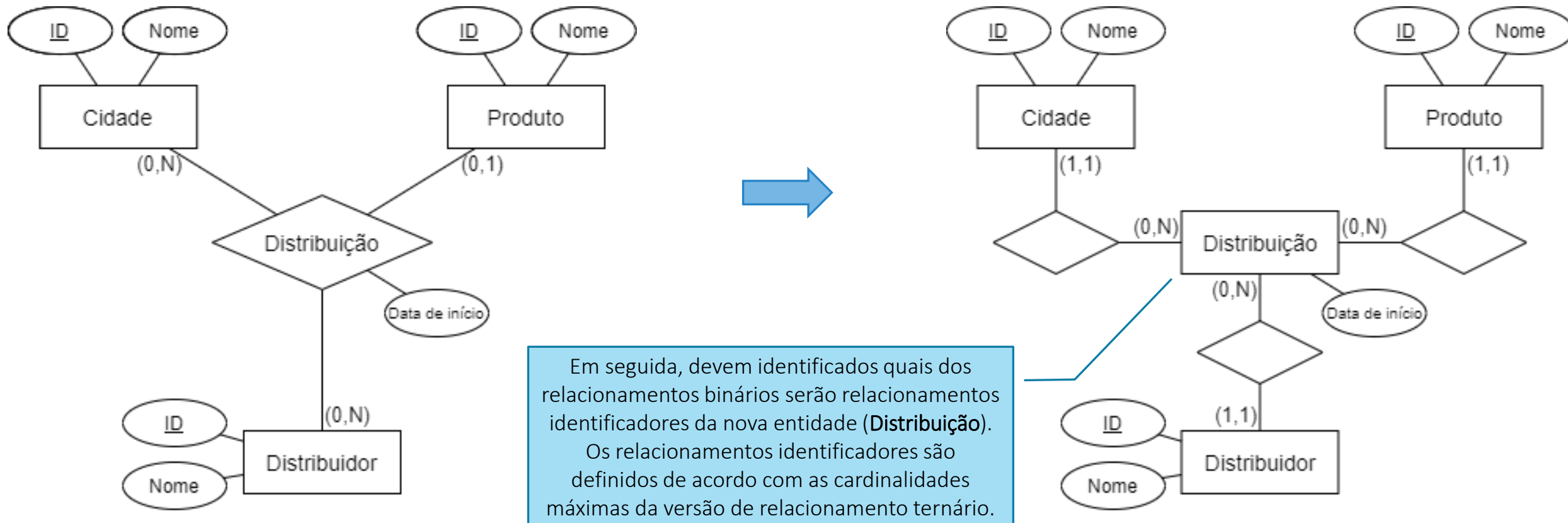


Mapeamento ER para Modelo Relacional

Relacionamentos

- Relacionamentos com grau maior que 2

- Transformação do relacionamento ternário para um equivalente com apenas relacionamentos binários.

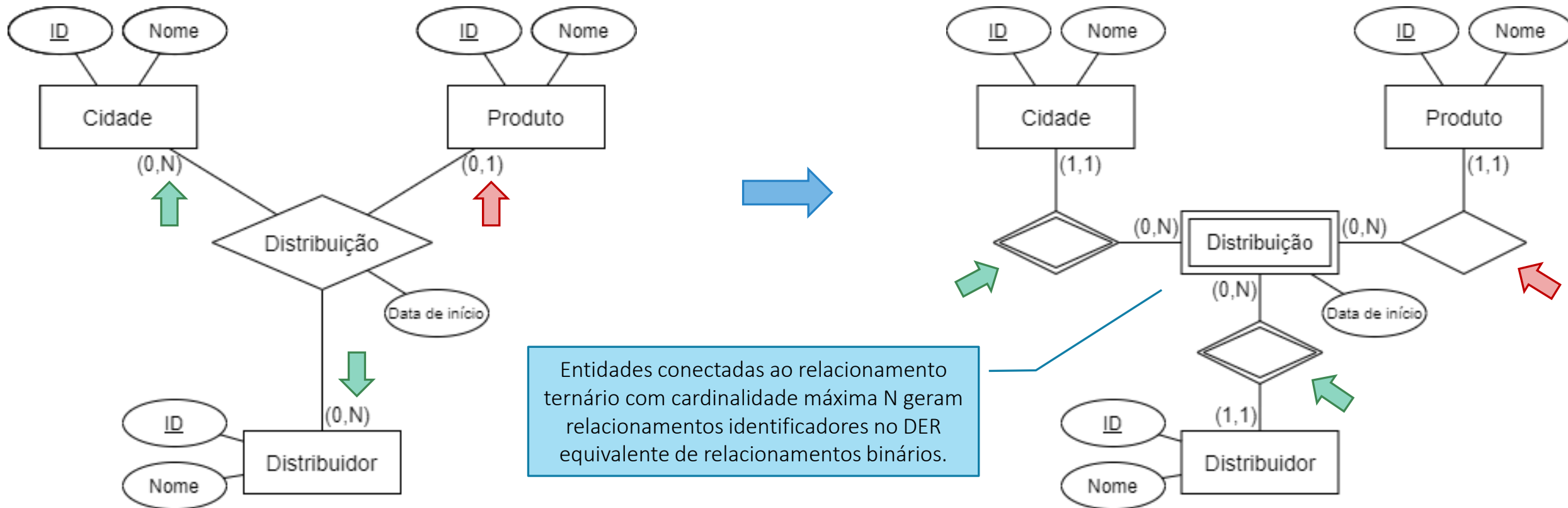


Mapeamento ER para Modelo Relacional

Relacionamentos

- **Relacionamentos com grau maior que 2**

- Transformação do relacionamento ternário para um equivalente com apenas relacionamentos binários.



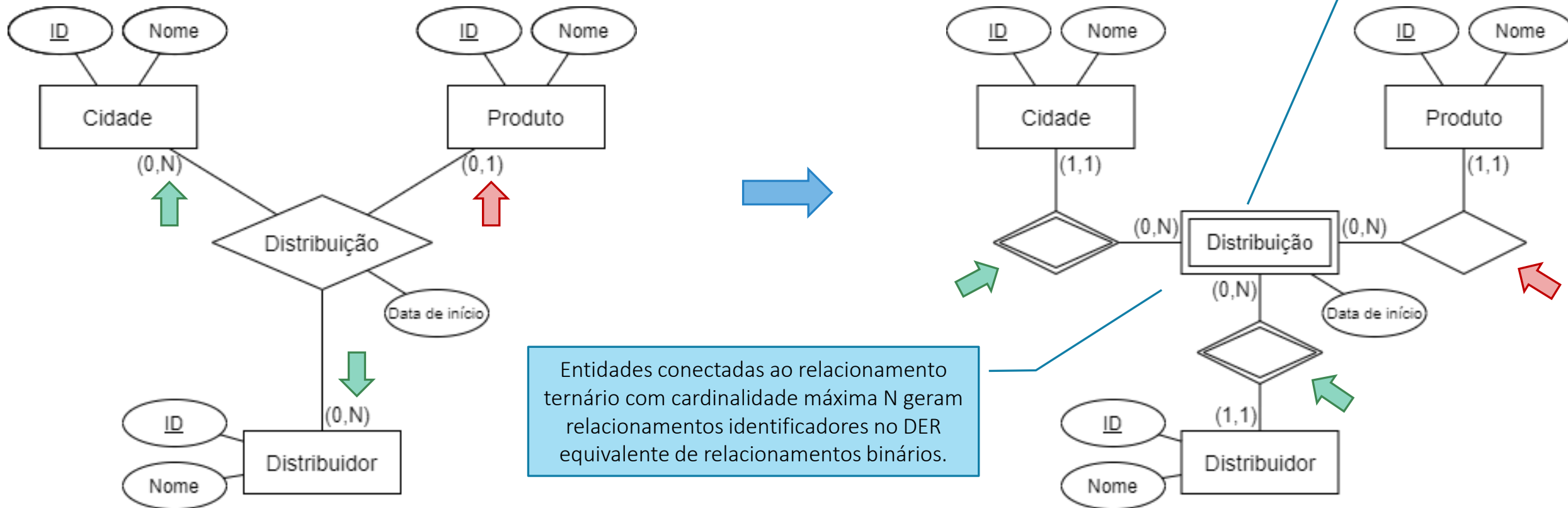
Mapeamento ER para Modelo Relacional

Relacionamentos

- **Relacionamentos com grau maior que 2**

- Transformação do relacionamento ternário para um equivalente com apenas relacionamentos binários.

Por ser identificada por relacionamentos, a entidade criada para substituir o relacionamento ternário é, portanto, uma entidade fraca.



Mapeamento ER para Modelo Relacional

Relacionamentos

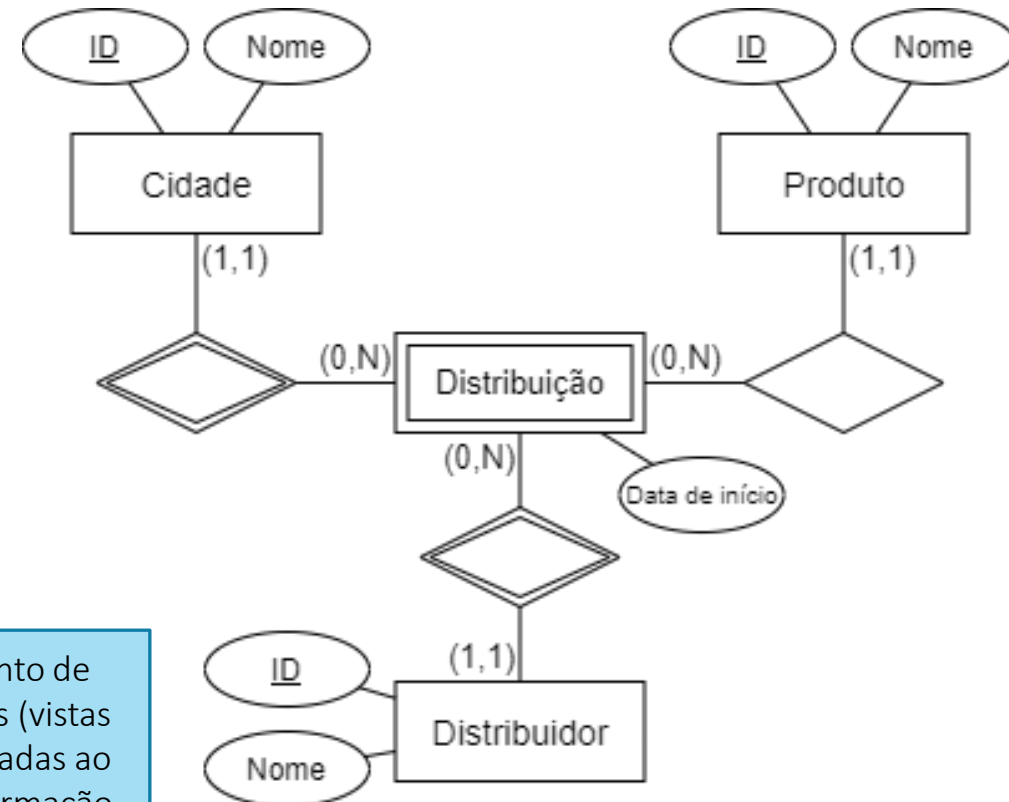
- **Relacionamentos com grau maior que 2**

- Transformação do relacionamento ternário para um equivalente com apenas relacionamentos binários.

```
CIDADE( id, nome )  
DISTRIBUIDOR( id, nome )  
PRODUTO( id, nome )
```

```
DISTRIBUICAO( cidade_id, distribuidor_id,  
              produto_id, data_inicio )  
cidade_id referencia CIDADE  
distribuidor_id referencia DISTRIBUIDOR  
produto_id referencia PRODUTO
```

As regras de mapeamento de relacionamentos binários (vistas anteriormente) são aplicadas ao DER resultante da transformação.

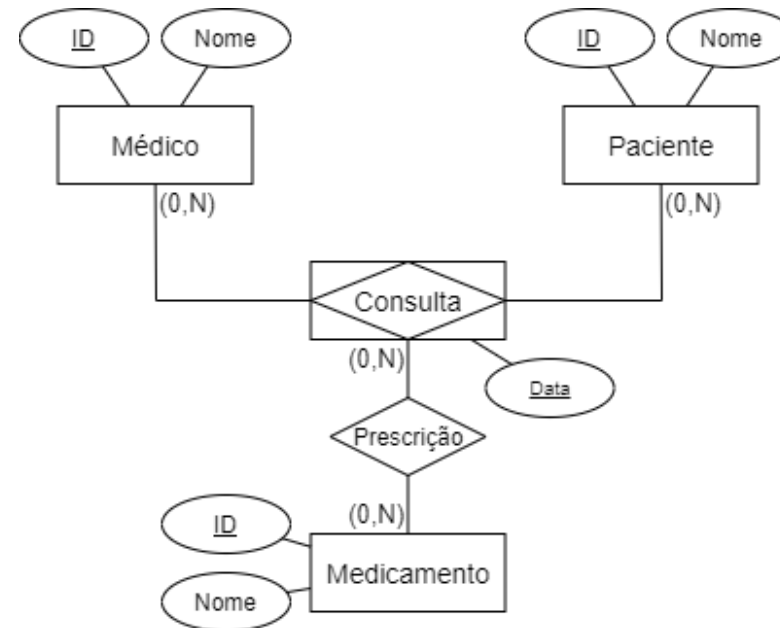


Mapeamento ER para Modelo Relacional

Relacionamentos

- **Entidades associativas (agregação)**

- Assim como fizemos para relacionamentos de grau maior que dois, faremos para realizar o mapeamento de uma entidade associativa do Modelo ER para um Modelo Relacional.
 - Primeiro, transformamos a entidade associativa para uma representação equivalente sem entidades associativas.
 - Em seguida, realizamos o mapeamento dos do diagrama resultante com relacionamentos tradicionais.

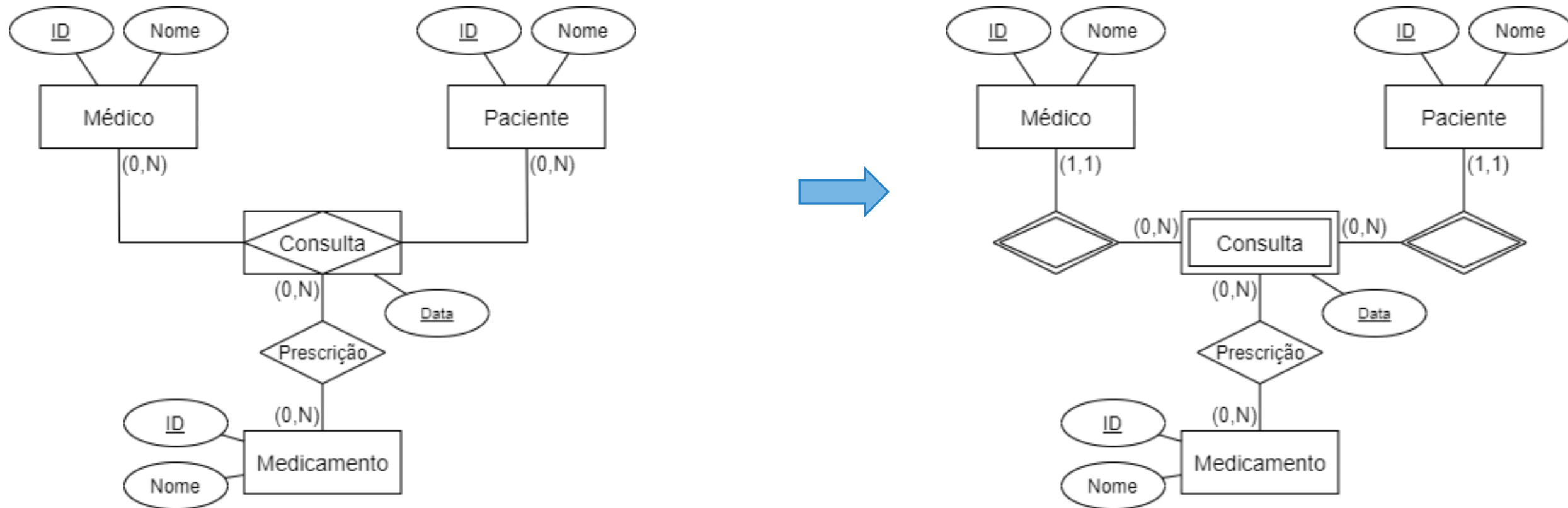


Mapeamento ER para Modelo Relacional

Relacionamentos

- **Entidades associativas (agregação)**

- Transformação de uma entidade associativa para um forma equivalente sem entidades associativas.

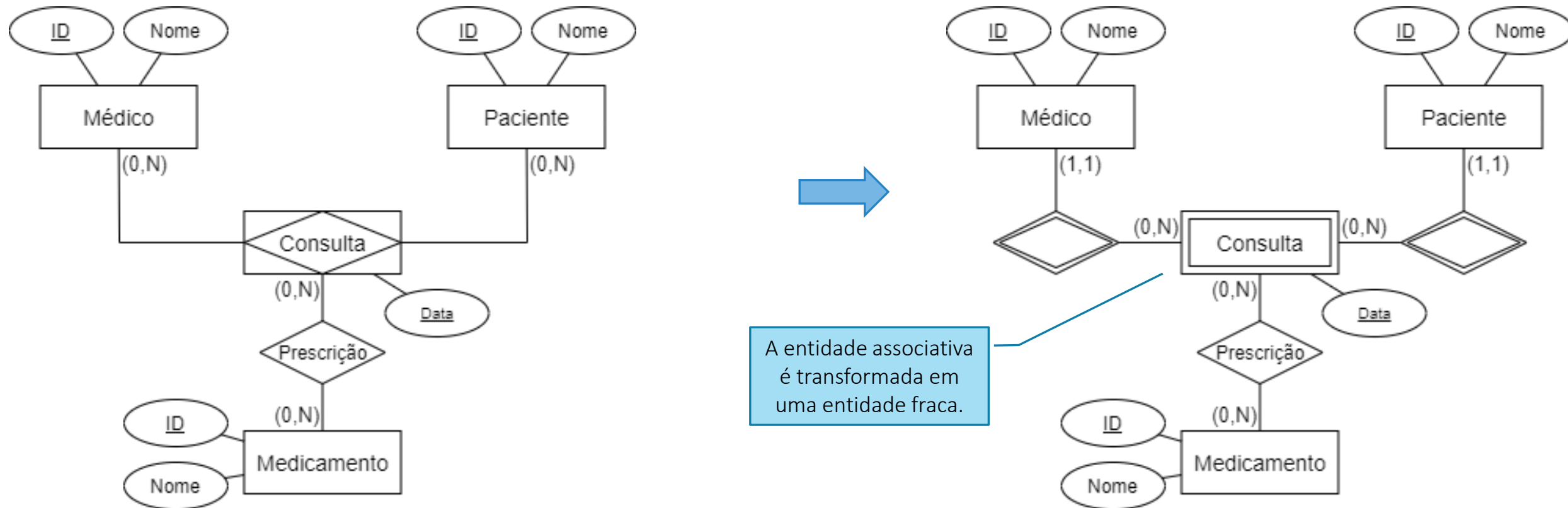


Mapeamento ER para Modelo Relacional

Relacionamentos

- **Entidades associativas (agregação)**

- Transformação de uma entidade associativa para um forma equivalente sem entidades associativas.

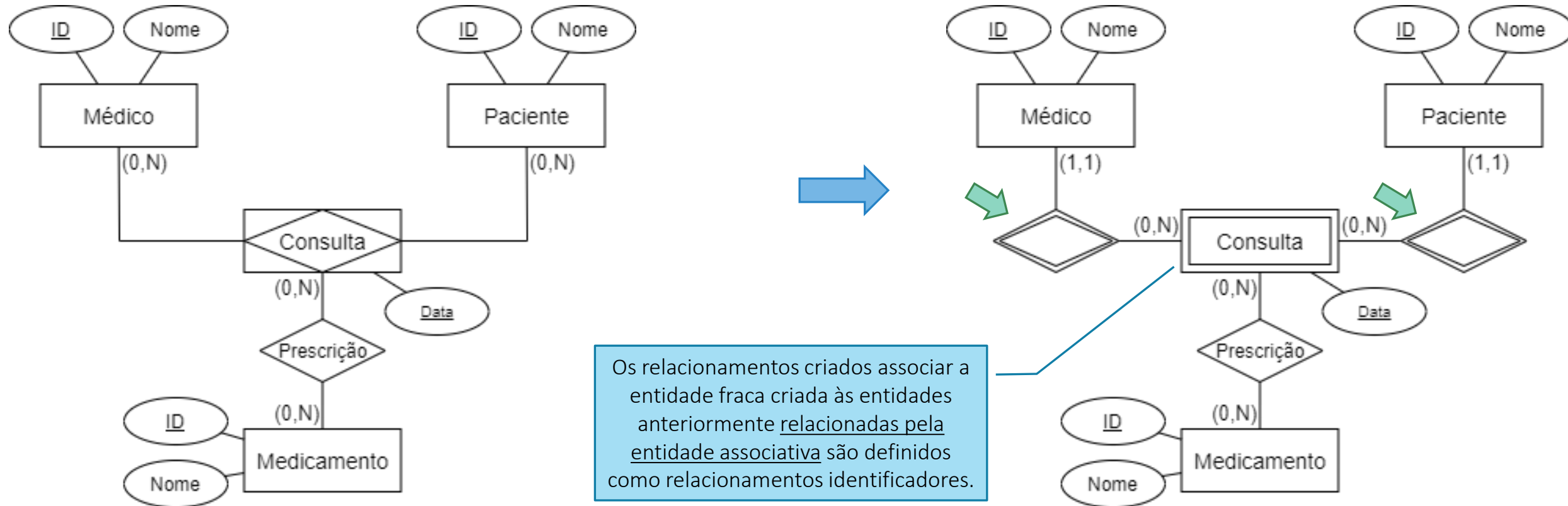


Mapeamento ER para Modelo Relacional

Relacionamentos

- **Entidades associativas (agregação)**

- Transformação de uma entidade associativa para um forma equivalente sem entidades associativas.

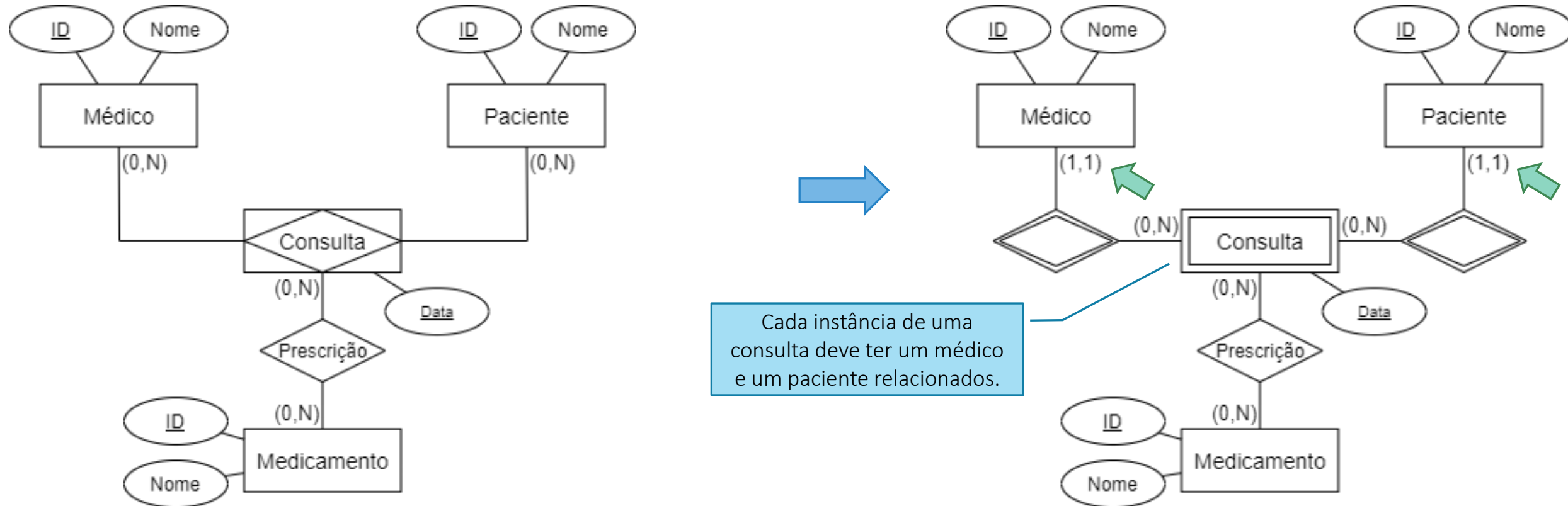


Mapeamento ER para Modelo Relacional

Relacionamentos

- **Entidades associativas (agregação)**

- Transformação de uma entidade associativa para um forma equivalente sem entidades associativas.

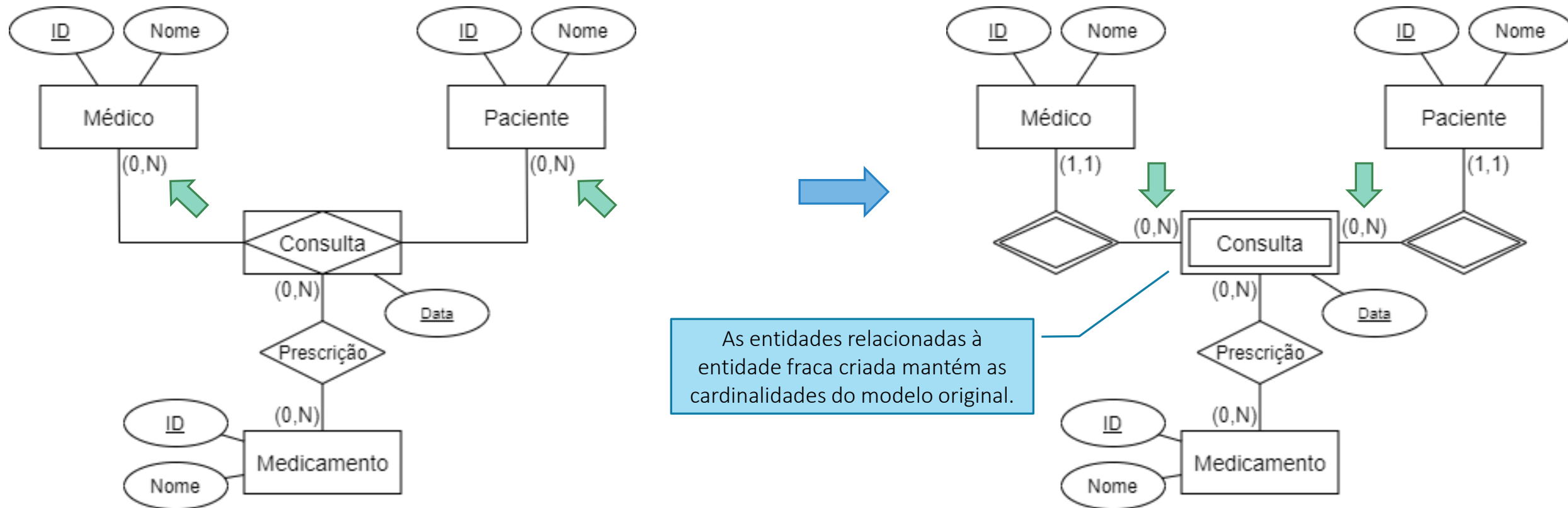


Mapeamento ER para Modelo Relacional

Relacionamentos

- **Entidades associativas (agregação)**

- Transformação de uma entidade associativa para um forma equivalente sem entidades associativas.



Mapeamento ER para Modelo Relacional

Relacionamentos

- Entidades associativas (agregação)

- Transformação de uma entidade associativa para um forma equivalente sem entidades associativas.

```
MEDICO( id, nome )
```

```
PACIENTE( id, nome )
```

```
MEDICAMENTO( id, nome )
```

```
CONSULTA( medico id, paciente id, data )
```

```
medico_id referencia MEDICO
```

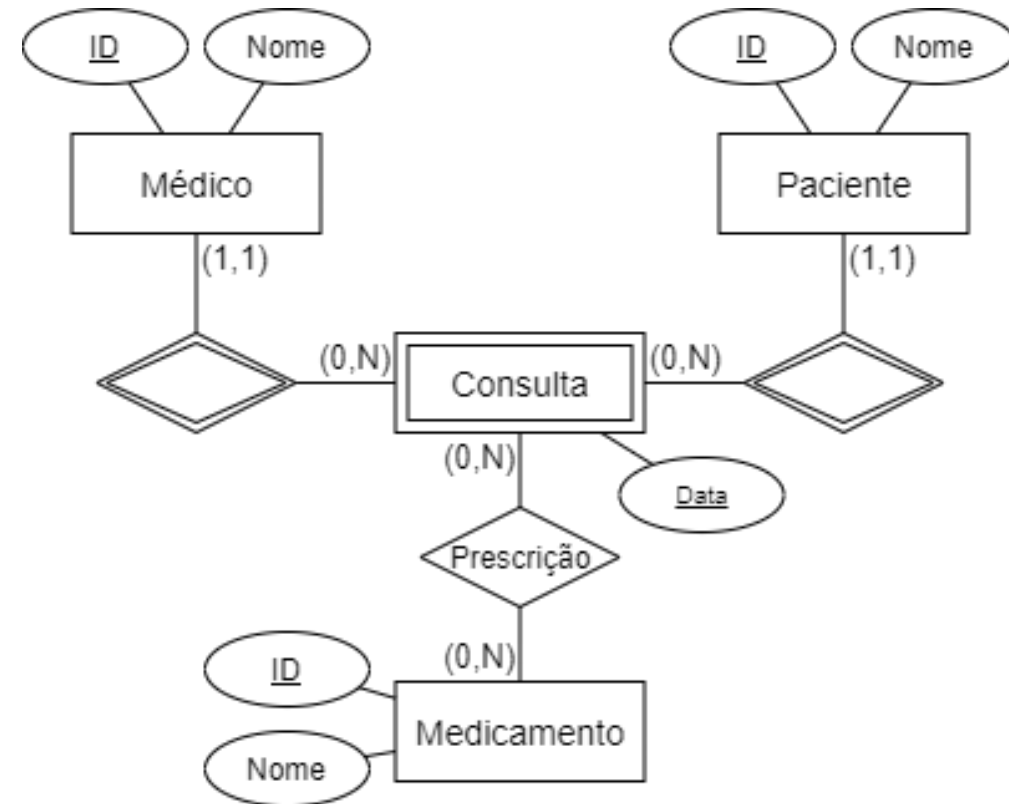
```
paciente_id referencia PACIENTE
```

```
PRESCRICAO( medico id, paciente id, data, medicamento id )
```

```
medico_id, paciente_id, data referencia CONSULTA
```

```
medicamento_id referencia MEDICAMENTO
```

As regras de mapeamento de relacionamentos binários (vistas anteriormente) são aplicadas ao DER resultante da transformação.

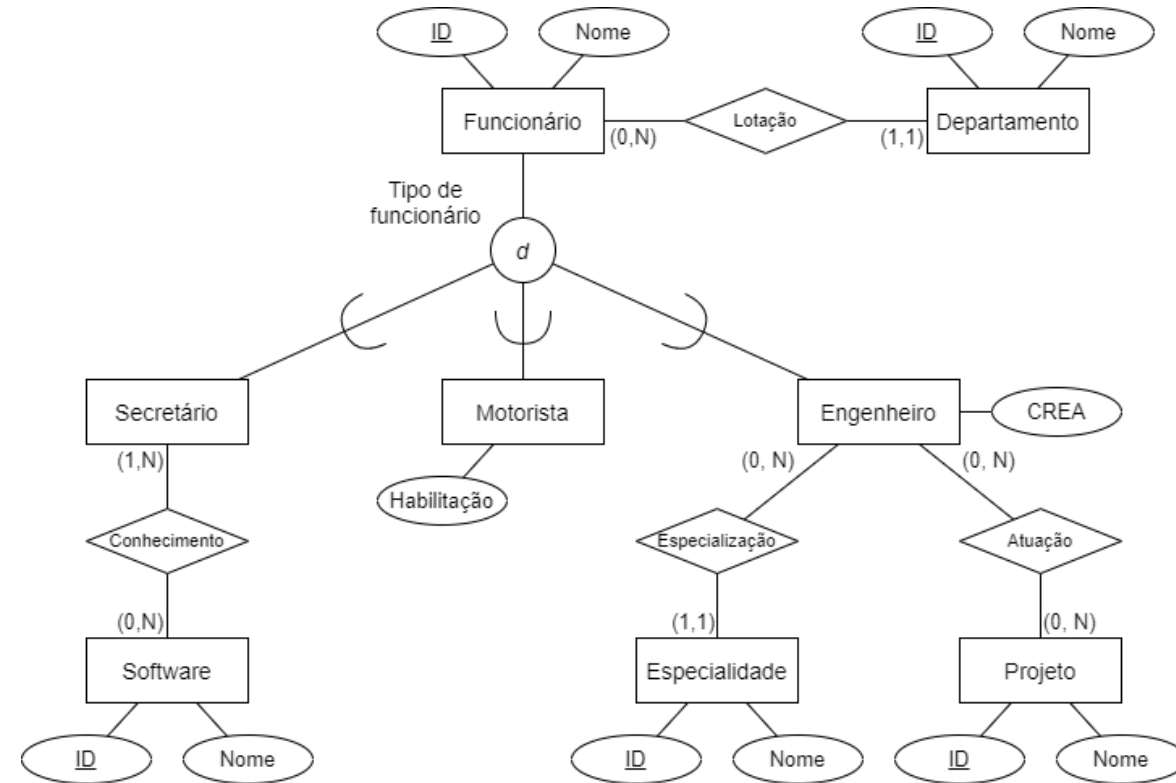


Mapeamento ER para Modelo Relacional

Generalização & Especialização

- Existem três alternativas para realizar o mapeamento hierarquias de Generalização/Especialização do Modelo ER para o Modelo Relacional:

- Tabela única por hierarquia
- Tabelas distintas com tabela para a entidade genérica
- Tabelas distintas sem tabela para a entidade genérica



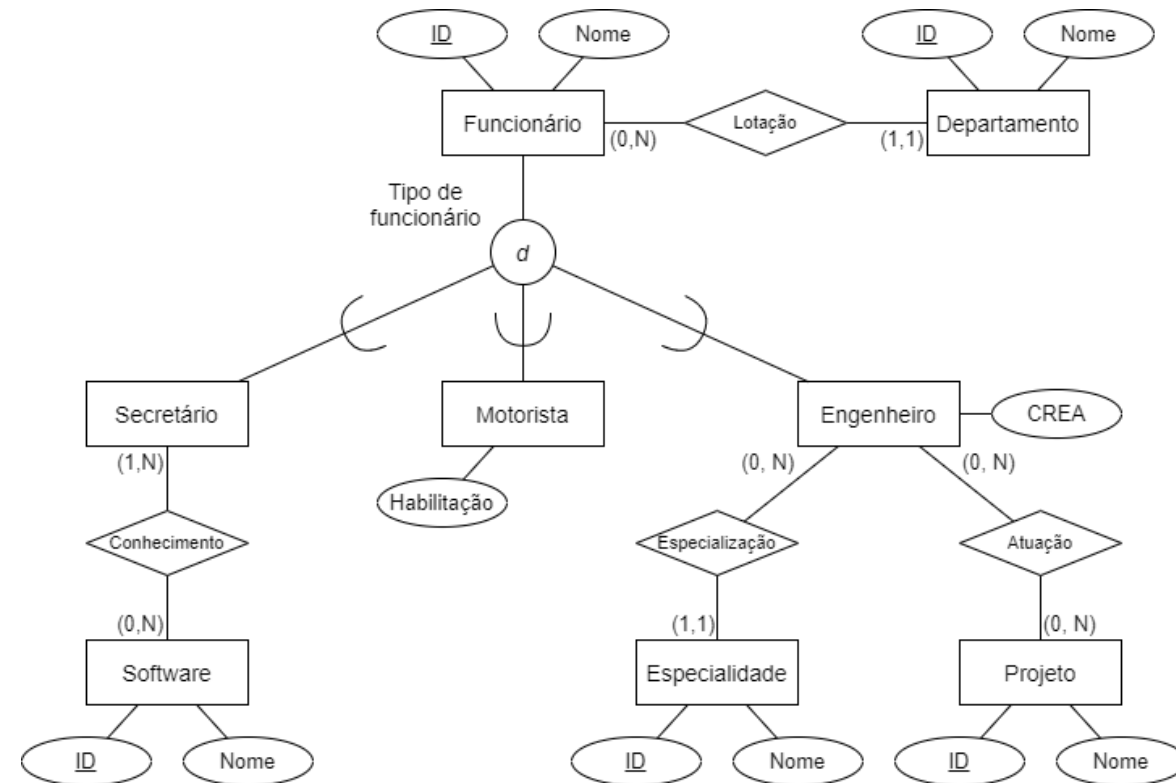
Mapeamento ER para Modelo Relacional

Generalização & Especialização

- Tabela única por hierarquia

- Essa estratégia contém:

- Chave primária da entidade mais genérica
- Uma coluna para identificar a especialização (tipo).
- Uma coluna para cada atributo da entidade genérica.
- Colunas referentes aos relacionamentos que a entidade genérica participa.
 - Para os casos em que o mapeamento for feito por adição de colunas.
- Uma coluna para cada atributo de cada entidade especializada.
- Colunas referentes aos relacionamentos que as entidades especializadas participam.
 - Para os casos em que o mapeamento for feito por adição de colunas.

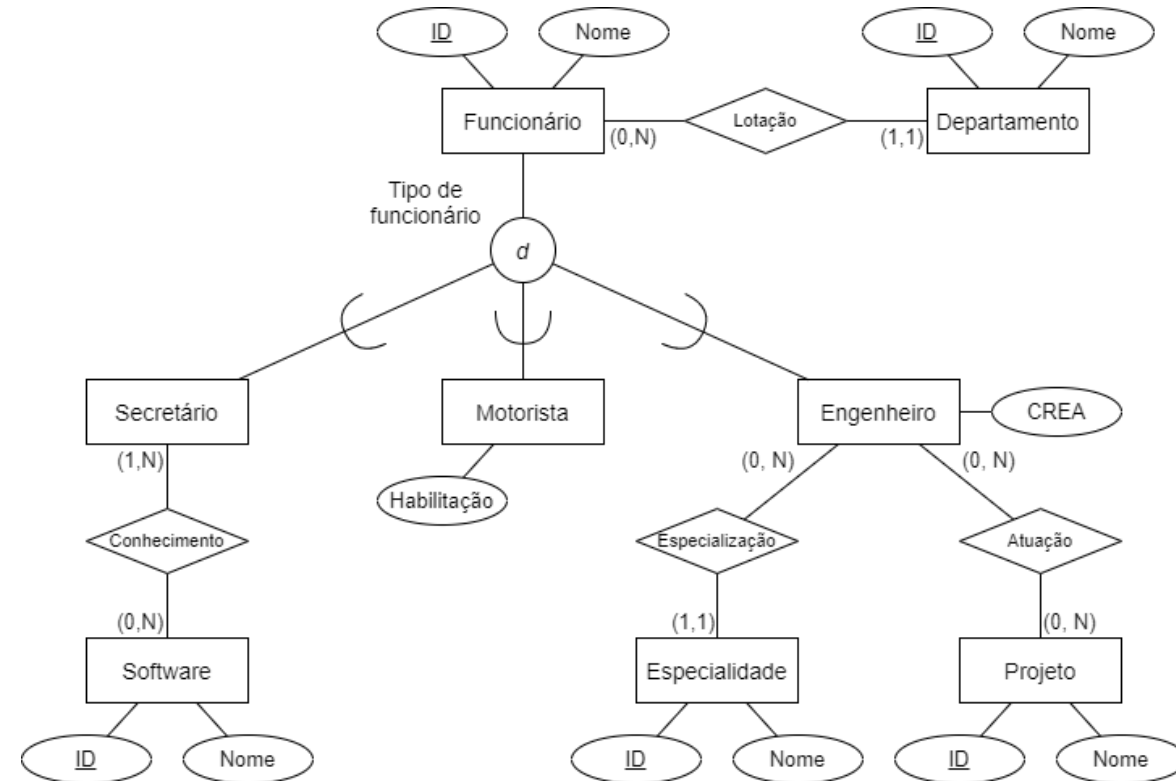


Mapeamento ER para Modelo Relacional

Generalização & Especialização

- Tabela única por hierarquia

```
DEPARTAMENTO( id, nome )  
SOFTWARE( id, nome )  
ESPECIALIDADE( id, nome )  
PROJETO( id, nome )  
  
FUNCIONARIO( id, tipo, nome, id_dep, habilitacao, crea, id_esp )  
    id_dep referencia DEPARTAMENTO  
    id_esp referencia ESPECIALIZACAO  
  
CONHECIMENTO( id func, id soft )  
    id_func referencia FUNCIONARIO  
    id_soft referencia SOFTWARE  
  
ATUACAO( id func, id proj )  
    id_func referencia FUNCIONARIO  
    id_proj referencia PROJETO
```



Mapeamento ER para Modelo Relacional

Generalização & Especialização

- Tabela única por hierarquia

```
DEPARTAMENTO( id, nome )
SOFTWARE( id, nome )
ESPECIALIDADE( id, nome )
PROJETO( id, nome )

FUNCIONARIO( id, tipo, nome, id_dep, habilitacao, crea, id_esp )
    id_dep referencia DEPARTAMENTO
    id_esp referencia ESPECIALIZACAO

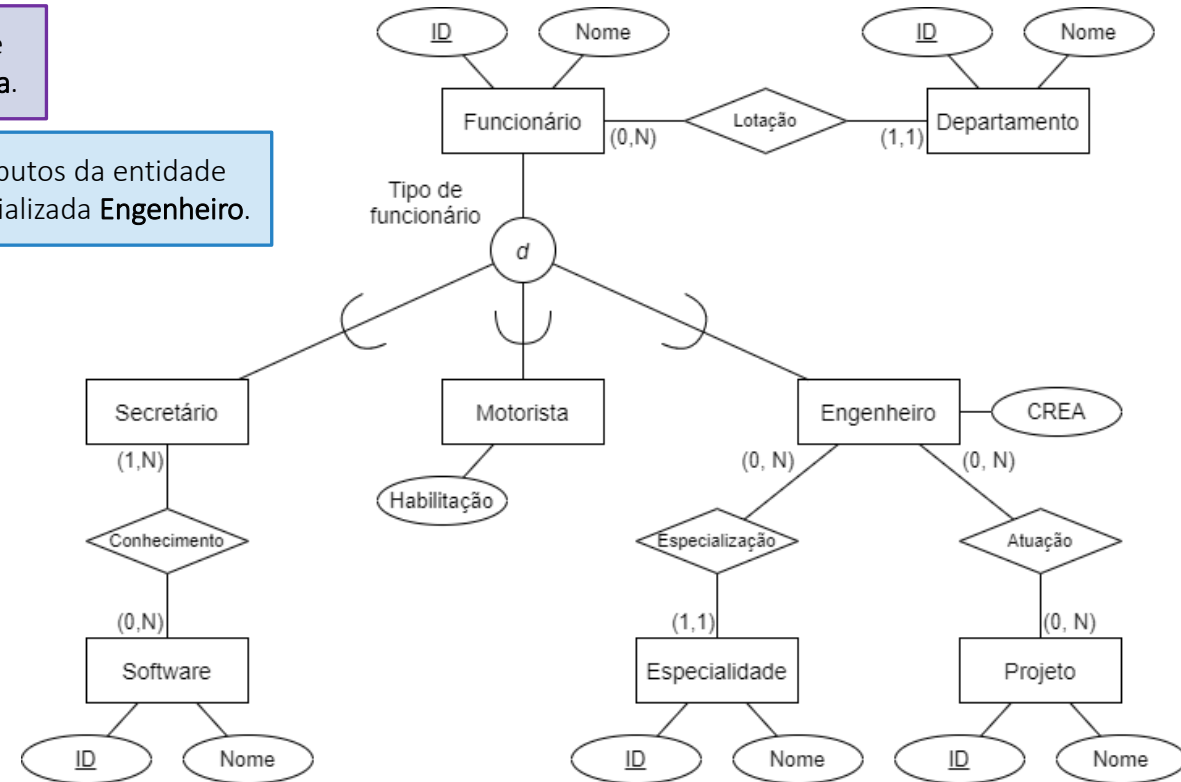
CONHECIMENTO( id func, id soft )
    id_func referencia FUNCIONARIO
    id_soft referencia SOFTWARE

ATUACAO( id func, id proj )
    id_func referencia FUNCIONARIO
    id_proj referencia PROJETO
```

Atributos da entidade genérica **Funcionário**.

Atributos da entidade especializada **Motorista**.

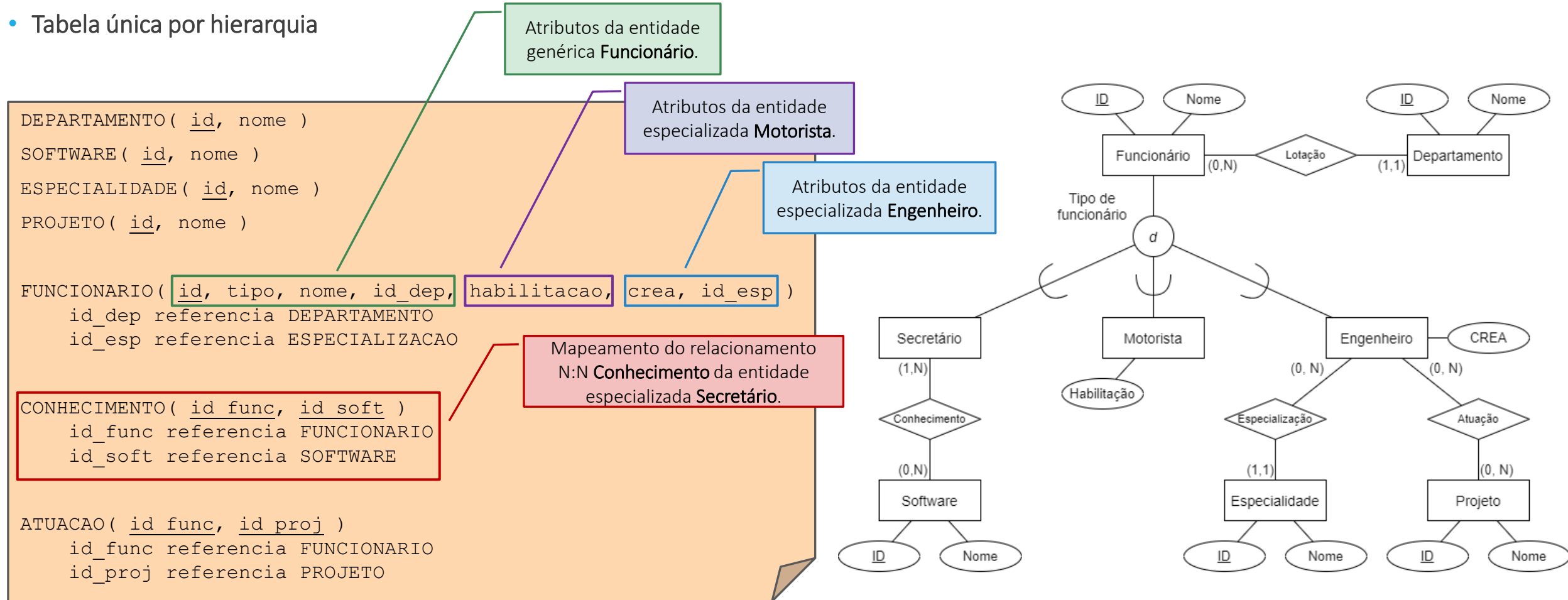
Atributos da entidade especializada **Engenheiro**.



Mapeamento ER para Modelo Relacional

Generalização & Especialização

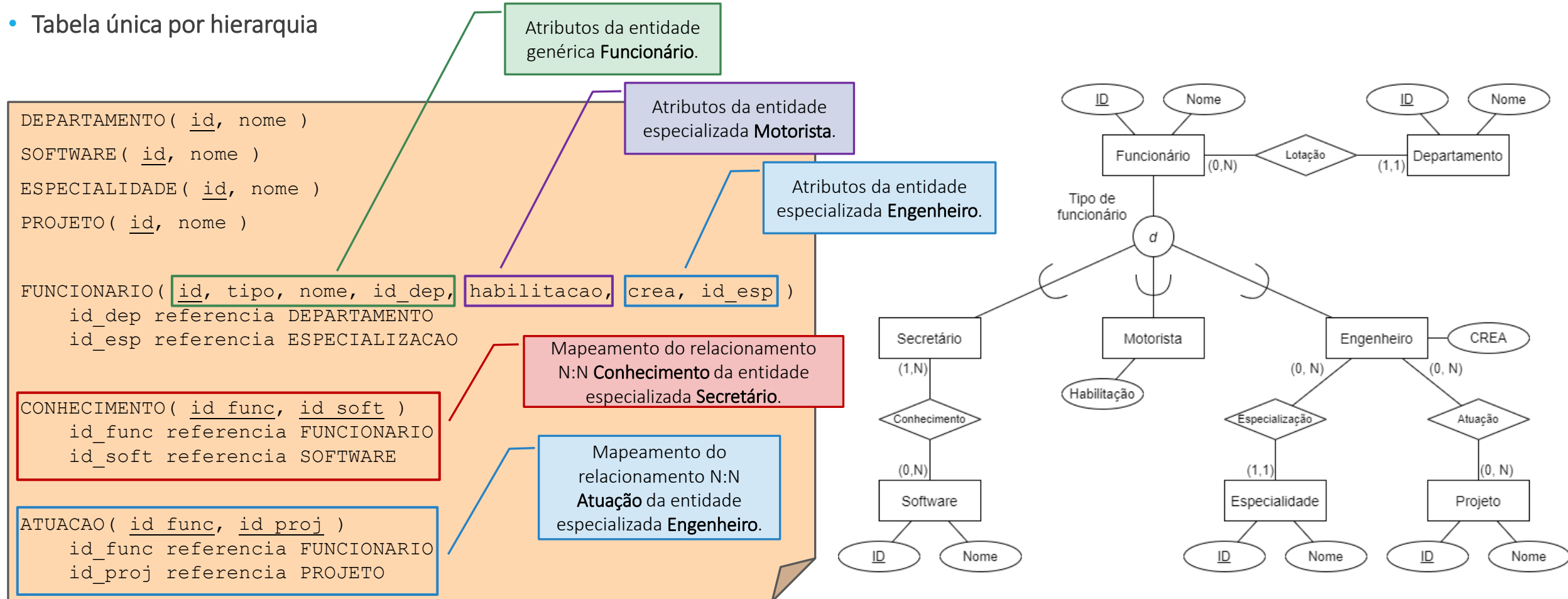
- Tabela única por hierarquia



Mapeamento ER para Modelo Relacional

Generalização & Especialização

- Tabela única por hierarquia



Mapeamento ER para Modelo Relacional

Generalização & Especialização

- Tabela única por hierarquia

```
DEPARTAMENTO( id, nome )
```

```
SOFTWARE( id, nome )
```

```
ESPECIALIDADE( id, nome )
```

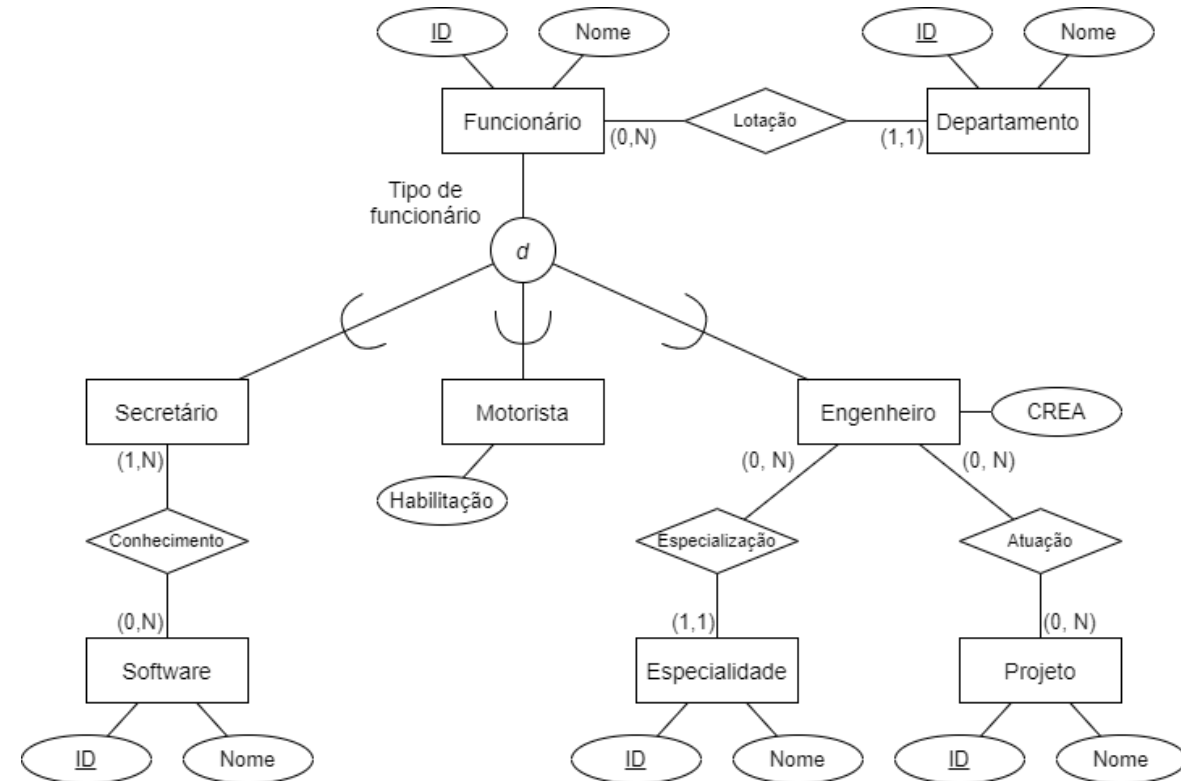
```
PROJETO( id, nome )
```

```
FUNCIONARIO( id, tipo, nome, id_dep, habilitacao, crea, id_esp )  
  id_dep referencia DEPARTAMENTO  
  id_esp referencia ESPECIALIZACAO
```

```
CONHECIMENTO( id func, id soft )  
  id_func referencia FUNCIONARIO  
  id_soft referencia SOFTWARE
```

```
ATUACAO( id func, id proj )  
  id_func referencia FUNCIONARIO  
  id_proj referencia PROJETO
```

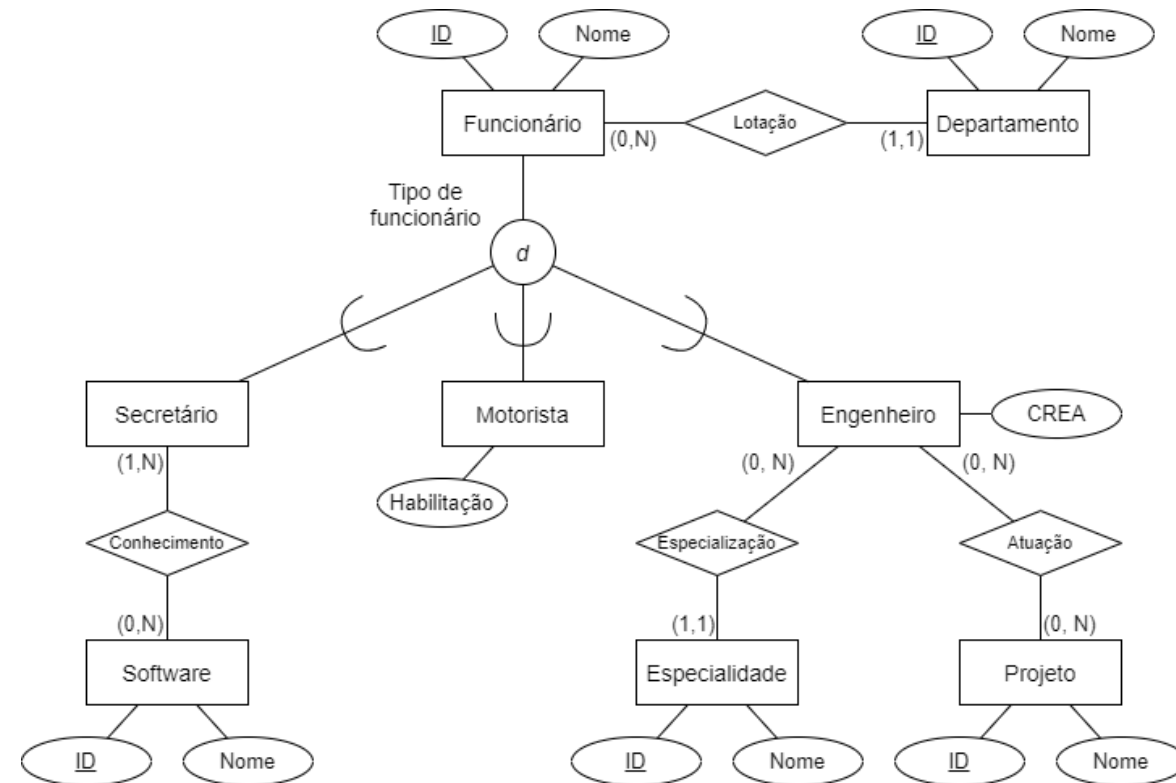
A coluna **tipo** é utilizada para identificar qual o tipo de funcionário uma determinada instância de **Funcionário** é. Como a especialização é parcial, os valores de tipo não se limitam a valores das entidades especializadas.



Mapeamento ER para Modelo Relacional

Generalização & Especialização

- Tabelas distintas com tabela para a entidade genérica
 - Essa estratégia contém:
 - Uma tabela para a entidade genérica com seus atributos.
 - Uma coluna “tipo” na tabela da entidade genérica para identificar a entidade especializada ao qual a instância pertence.
 - Colunas referentes aos relacionamentos que a entidade genérica participa permanecem na tabela da entidade genérica.
 - Para os relacionamentos mapeados através de adição de colunas.
 - Uma tabela para cada entidade especializada.
 - Entidades especializadas que não possuem atributos nem relacionamentos a mapear através da adição por colunas não precisam de uma tabela própria (a tabela da entidade genérica é suficiente para esses casos).
 - A chave primária é uma chave estrangeira para a entidade genérica correspondente, dessa forma, cada instância da entidade especializada tem uma instância na entidade genérica.
 - Colunas referentes aos relacionamentos que uma entidade especializada participa ficam na tabela da entidade especializada.
 - Para os relacionamentos mapeados através de adição de colunas.



Mapeamento ER para Modelo Relacional

Generalização & Especialização

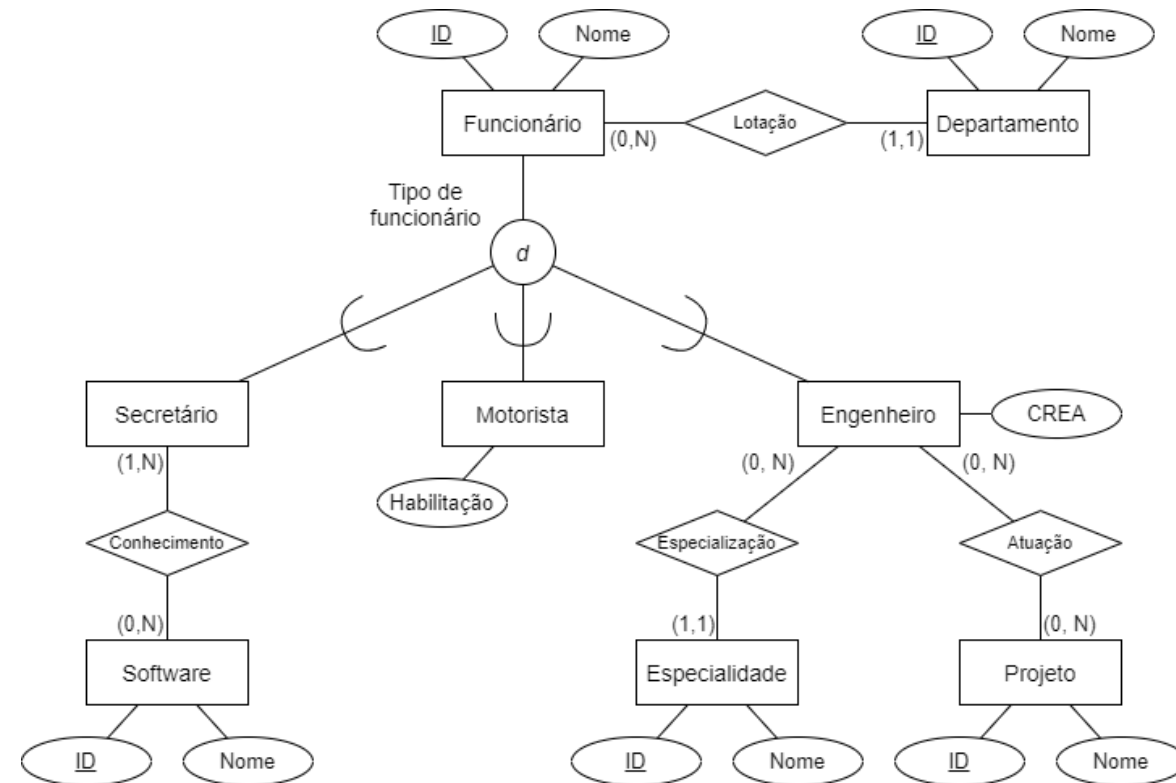
- Tabelas distintas com tabela para a entidade genérica

```
DEPARTAMENTO( id, nome )
SOFTWARE( id, nome )
ESPECIALIDADE( id, nome )
PROJETO( id, nome )

FUNCIONARIO( id, tipo, nome, id_dep )
    id_dep referencia DEPARTAMENTO
MOTORISTA( id, habilitacao )
    id referencia FUNCIONARIO
ENGENHEIRO( id, crea, id_esp )
    id referencia FUNCIONARIO
    id_esp referencia ESPECIALIDADE

CONHECIMENTO( id_func, id_soft )
    id_func referencia FUNCIONARIO
    id_soft referencia SOFTWARE

ATUACAO( id_func, id_proj )
    id_func referencia ENGENHEIRO
    id_proj referencia PROJETO
```



Mapeamento ER para Modelo Relacional

Generalização & Especialização

- Tabelas distintas com tabela para a entidade genérica

```
DEPARTAMENTO( id, nome )
```

```
SOFTWARE( id, nome )
```

```
ESPECIALIDADE( id, nome )
```

```
PROJETO( id, nome )
```

```
FUNCIONARIO( id, tipo, nome, id_dep )  
id_dep referencia DEPARTAMENTO
```

```
MOTORISTA( id, habilitacao )  
id referencia FUNCIONARIO
```

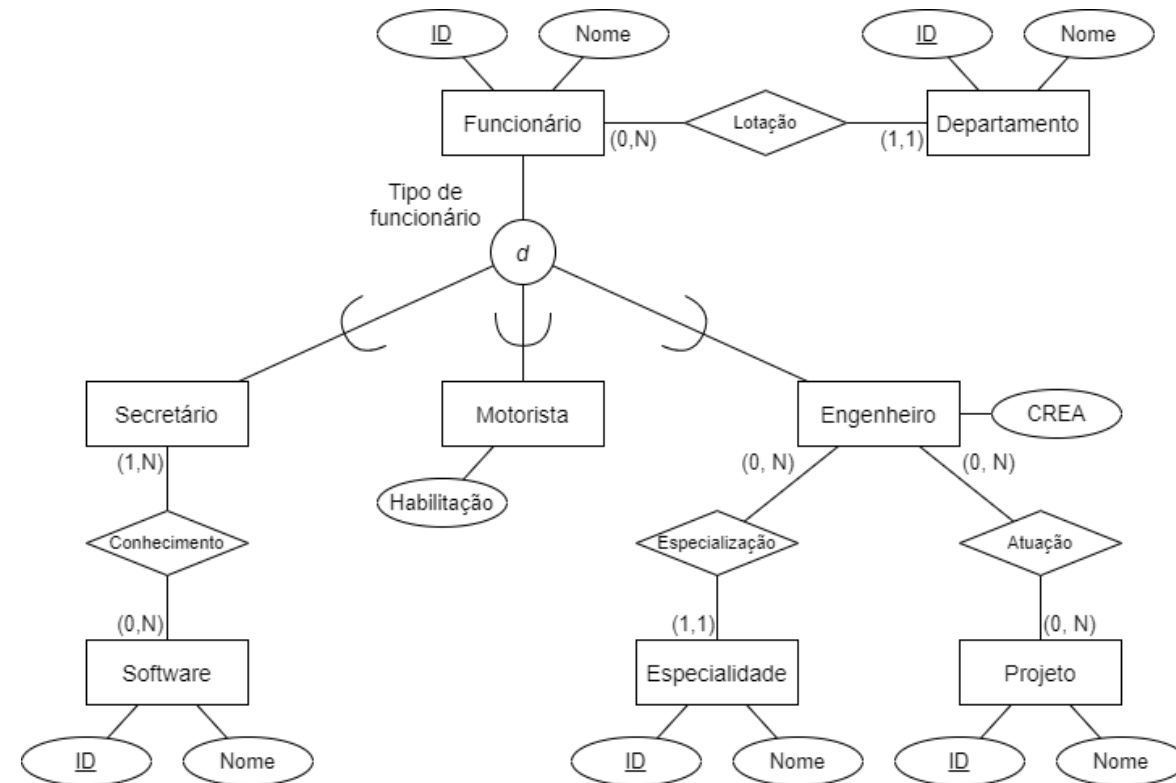
```
ENGENHEIRO( id, crea, id_esp )  
id referencia FUNCIONARIO  
id_esp referencia ESPECIALIDADE
```

```
CONHECIMENTO( id_func, id_soft )  
id_func referencia FUNCIONARIO  
id_soft referencia SOFTWARE
```

```
ATUACAO( id_func, id_proj )  
id_func referencia ENGENHEIRO  
id_proj referencia PROJETO
```

Mapeamento do relacionamento N:N **Atuação** da entidade especializada **Engenheiro**.

Observe que a coluna **id_func** referencia a tabela da entidade especializada **Engenheiro**.



Mapeamento ER para Modelo Relacional

Generalização & Especialização

- Tabelas distintas com tabela para a entidade genérica

```
DEPARTAMENTO( id, nome )
```

```
SOFTWARE( id, nome )
```

```
ESPECIALIDADE( id, nome )
```

```
PROJETO( id, nome )
```

```
FUNCIONARIO( id, tipo, nome, id_dep )  
    id_dep referencia DEPARTAMENTO
```

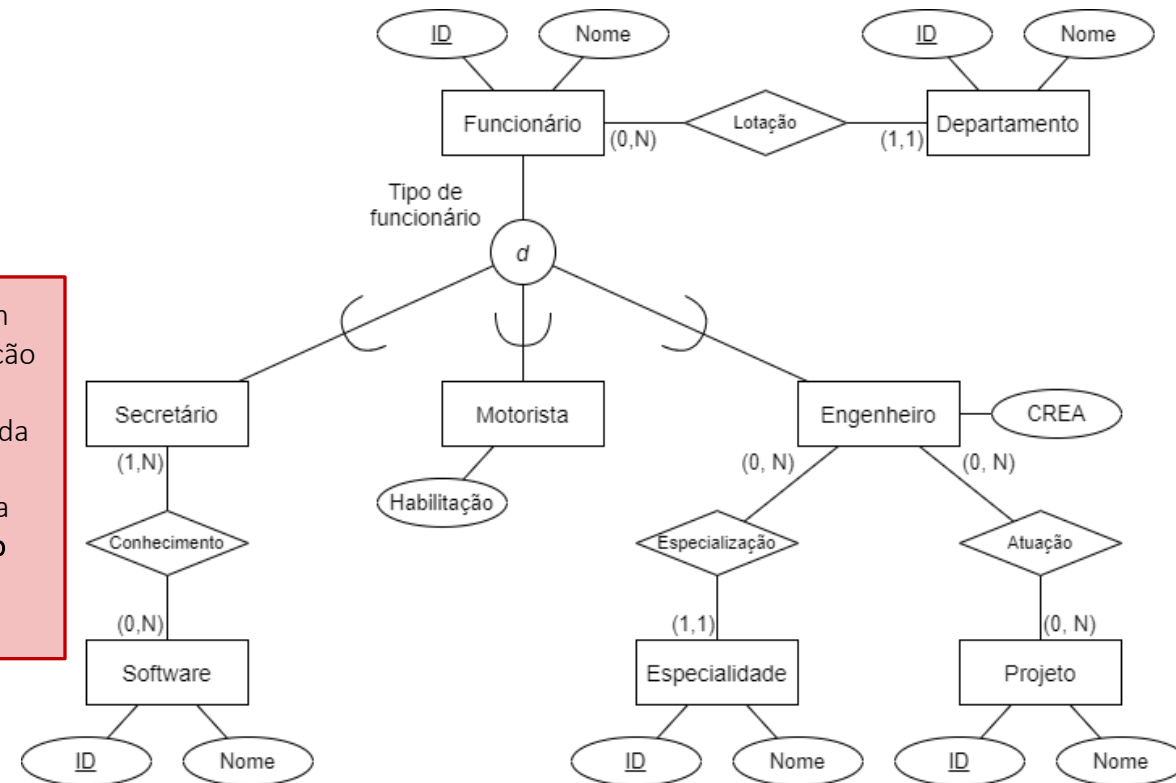
```
MOTORISTA( id, habilitacao )  
    id referencia FUNCIONARIO
```

```
ENGENHEIRO( id, crea, id_esp )  
    id referencia FUNCIONARIO  
    id_esp referencia ESPECIALIDADE
```

```
CONHECIMENTO( id_func, id_soft )  
    id_func referencia FUNCIONARIO  
    id_soft referencia SOFTWARE
```

```
ATUACAO( id_func, id_proj )  
    id_func referencia ENGENHEIRO  
    id_proj referencia PROJETO
```

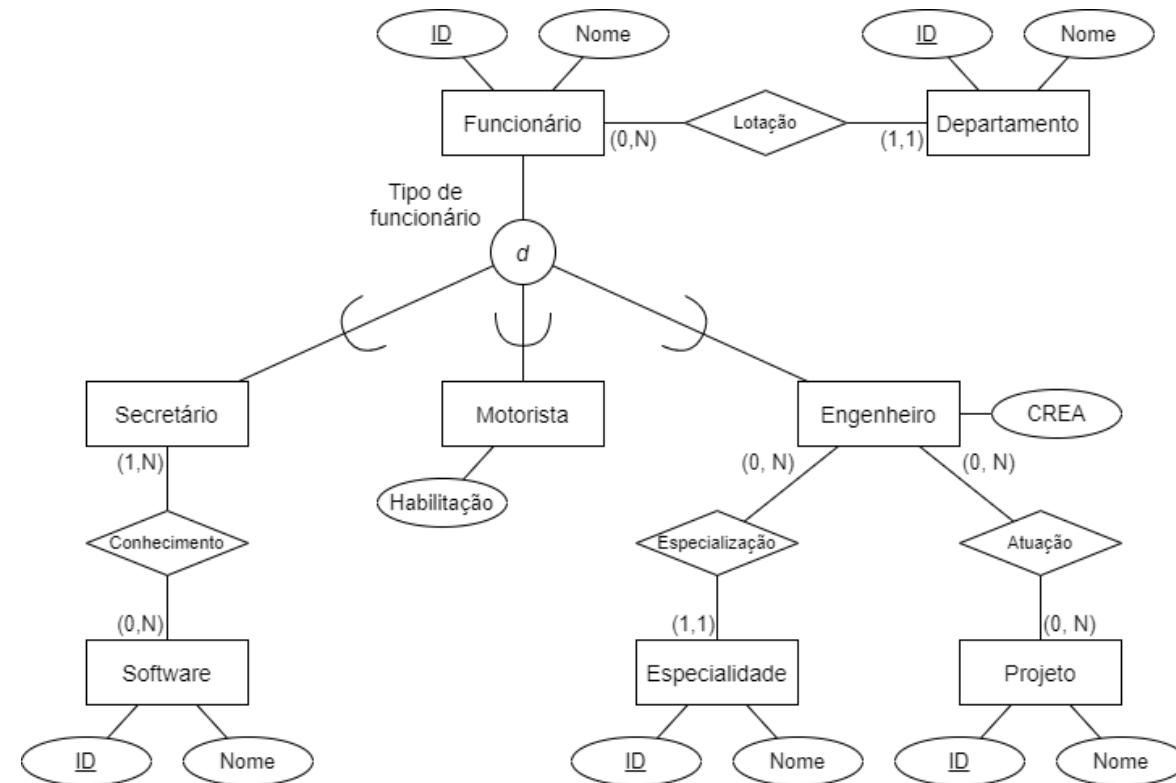
Secretário não tem atributos nem relacionamentos a mapear por adição de colunas. Portanto, não há necessidade de uma tabela separada para ela. Dessa forma, o relacionamento **Conhecimento** da entidade especializada **Secretário** referencia a tabela da entidade genérica **Funcionário**.



Mapeamento ER para Modelo Relacional

Generalização & Especialização

- Tabelas distintas sem tabela para a entidade genérica
- Essa estratégia contém:
 - Uma tabela para cada entidade especializada.
 - Cada tabela deverá conter, além das colunas para os atributos e relacionamentos mapeados por adição de colunas, colunas para os atributos e relacionamentos mapeados por adição de colunas da entidade genérica.
 - Para o caso da generalização/especialização ser parcial, poderão existir instâncias da entidade genérica que não seja instância de nenhuma das entidades especializadas.
 - Nesses casos, cria-se uma tabela contendo apenas os atributos da entidade genérica.
 - As tabelas das entidades especializadas não devem referenciar esta tabela.

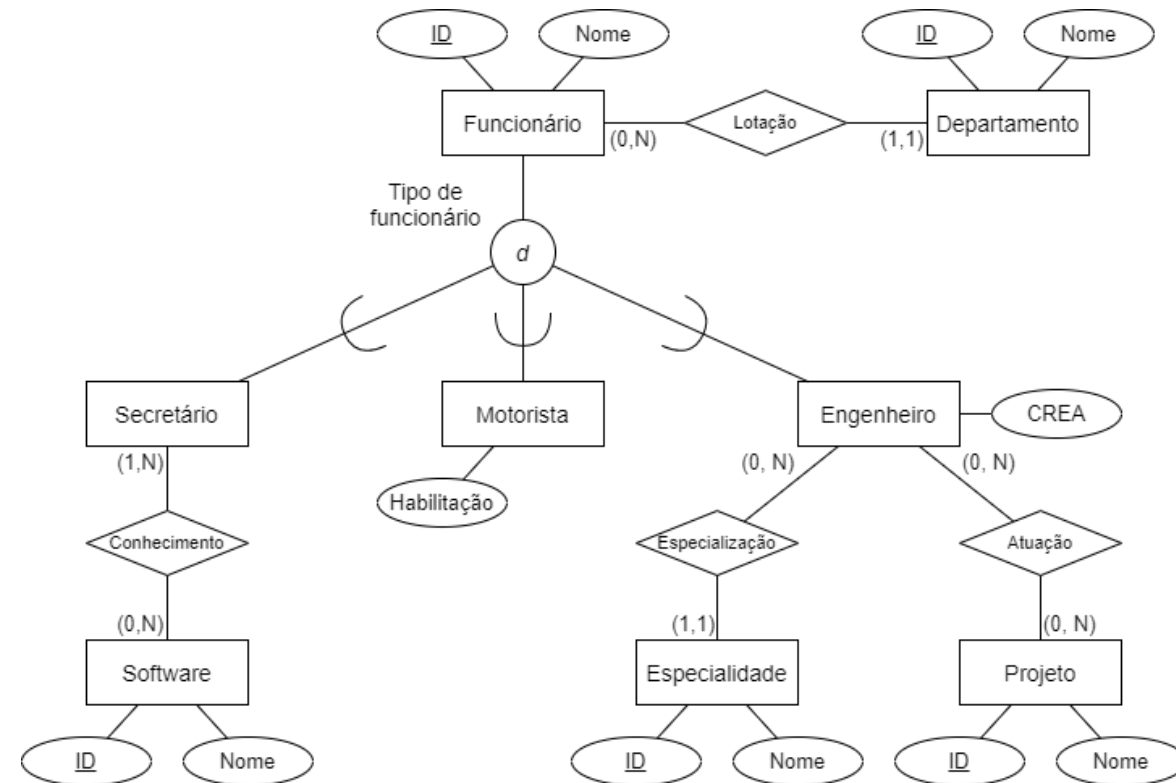


Mapeamento ER para Modelo Relacional

Generalização & Especialização

- Tabelas distintas sem tabela para a entidade genérica

```
DEPARTAMENTO( id, nome )  
SOFTWARE( id, nome )  
ESPECIALIDADE( id, nome )  
PROJETO( id, nome )  
  
FUNCIONARIO( id, tipo, nome, id_dep )  
    id_dep referencia DEPARTAMENTO  
MOTORISTA( id, nome, id_dep, habilitacao )  
    id_dep referencia DEPARTAMENTO  
ENGENHEIRO( id, nome, id_dep, crea, id_esp )  
    id_dep referencia DEPARTAMENTO  
    id_esp referencia ESPECIALIDADE  
  
CONHECIMENTO( id_func, id_soft )  
    id_func referencia FUNCIONARIO  
    id_soft referencia SOFTWARE  
ATUACAO( id_func, id_proj )  
    id_func referencia ENGENHEIRO  
    id_proj referencia PROJETO
```



Mapeamento ER para Modelo Relacional

Generalização & Especialização

- Tabelas distintas sem tabela para a entidade genérica

```
DEPARTAMENTO( id, nome )
```

```
SOFTWARE( id, nome )
```

```
ESPECIALIDADE( id, nome )
```

```
PROJETO( id, nome )
```

```
FUNCIONARIO( id, tipo, nome, id_dep )  
    id_dep referencia DEPARTAMENTO
```

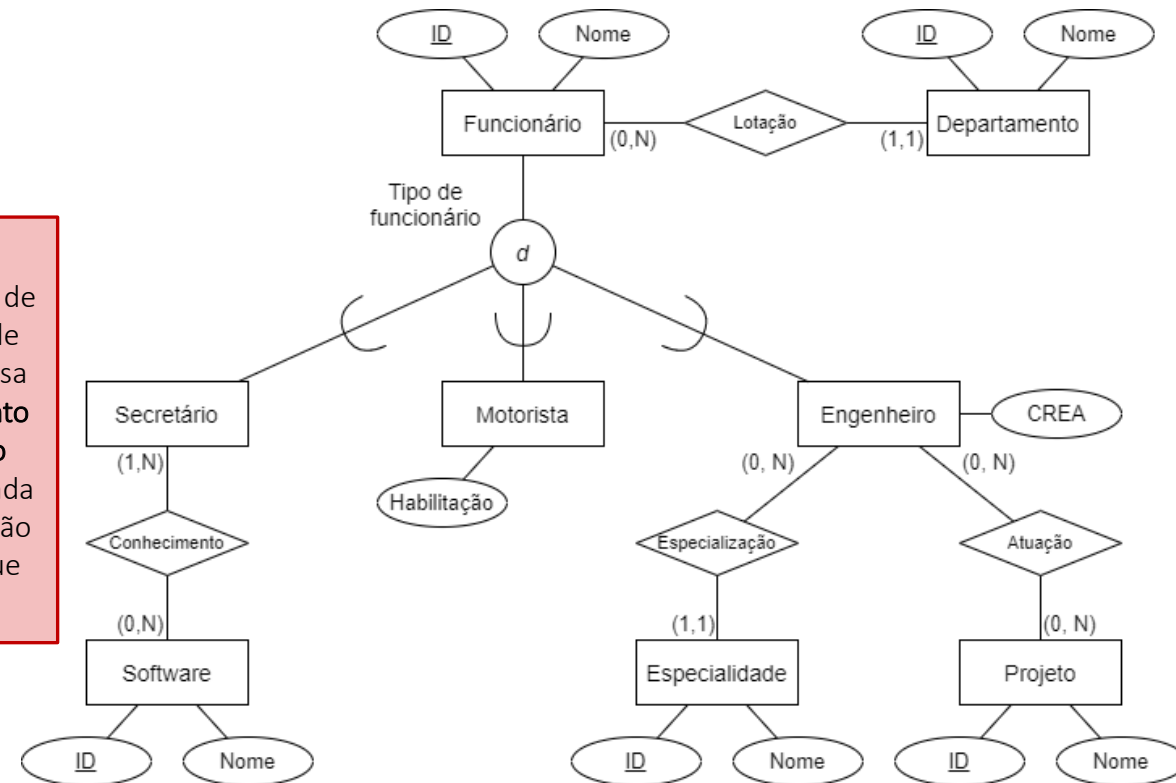
```
MOTORISTA( id, id_dep, habilitacao )  
    id_dep referencia DEPARTAMENTO
```

```
ENGENHEIRO( id, id_dep, crea, id_esp )  
    id_dep referencia DEPARTAMENTO  
    id_esp referencia ESPECIALIDADE
```

```
CONHECIMENTO( id_func, id_soft )  
    id_func referencia FUNCIONARIO  
    id_soft referencia SOFTWARE
```

```
ATUACAO( id_func, id_proj )  
    id_func referencia ENGENHEIRO  
    id_proj referencia PROJETO
```

Secretário não tem atributos nem relacionamentos a mapear por adição de colunas. Portanto, não há necessidade de uma tabela separada para ela. Dessa forma, o relacionamento **Conhecimento** da entidade especializada **Secretário** referencia a tabela **FUNCIONARIO**, criada para armazenar os funcionários que não se enquadram nas especializações que possuem suas próprias tabelas.



Mapeamento ER para Modelo Relacional

Generalização & Especialização

Alternativa para o mapeamento	Vantagens	Desvantagens
Tabela própria	<ul style="list-style-type: none">- Menor número de chaves.- Menor número de junções.	<ul style="list-style-type: none">- Excesso de colunas opcionais.
Tabelas distintas (<u>com</u> tabela para entidade genérica)	<ul style="list-style-type: none">- Melhor para manipulação de colunas opcionais, já que cada tabela terá colunas apenas de seus próprios atributos.	<ul style="list-style-type: none">- Requer um maior número de junções.
Tabelas distintas (<u>sem</u> tabela para entidade genérica)	?	<ul style="list-style-type: none">- A unicidade do identificador entre as instâncias das entidades na hierarquia de generalização/especialização não pode ser garantida pelo SGBD. Essa garantia deverá ser feita na aplicação que utiliza o banco de dados.- Não há como especificar ao SGBD restrições de integridade referenciais que façam referencia ao conjunto de todas as instâncias de todas entidades da hierarquia.- Essa alternativa é raramente utilizada.

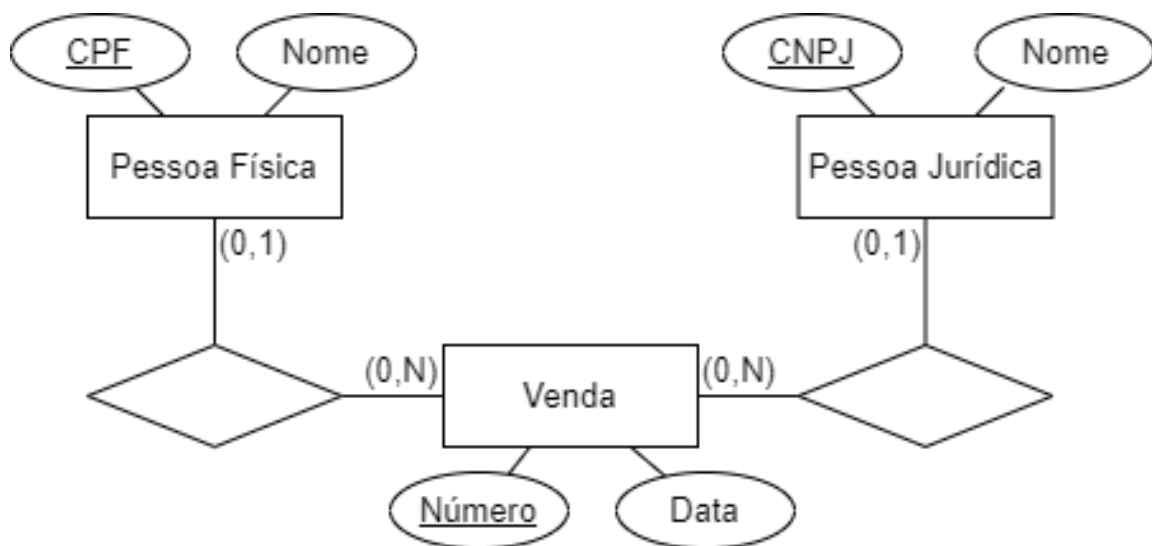
Refinamento do modelo relacional

- Algumas vezes, o esquema criado para um banco de dados a partir das regras de mapeamento descritas anteriormente podem não atender aos requisitos de desempenho impostos a aplicação.
- Nesses casos, é necessário buscar alternativas que resultem em um melhor desempenho da aplicação.
 - Essas alternativas somente devem ser realizadas em casos de real necessidade de melhoria de desempenho, pois do ponto de vista de um projeto de banco de dados e da aplicação, elas são geralmente são piores.
- Alguns exemplos de alternativas aos resultados obtidos a partir das regras de mapeamento:
 - Relacionamentos mutuamente exclusivos
 - Simulação de atributos multivalorados
 - Inserção de informações redundantes.

Refinamento do modelo relacional

Relacionamentos mutuamente exclusivos

- Considerando o diagrama abaixo, temos o resultado do modelo lógico relacional obtido a partir das regras de mapeamento vistas anteriormente:
 - As colunas **cpf** e **cnpj** são opcionais, já que para cada ocorrência de uma venda, apenas uma delas deve ser utilizada.
 - Ou a venda é feita para uma pessoa física (CPF) ou para uma pessoa jurídica (CNPJ).



```
PESSOA_FISICA( cpf, nome )
```

```
PESSOA_JURIDICA( cnpj, nome )
```

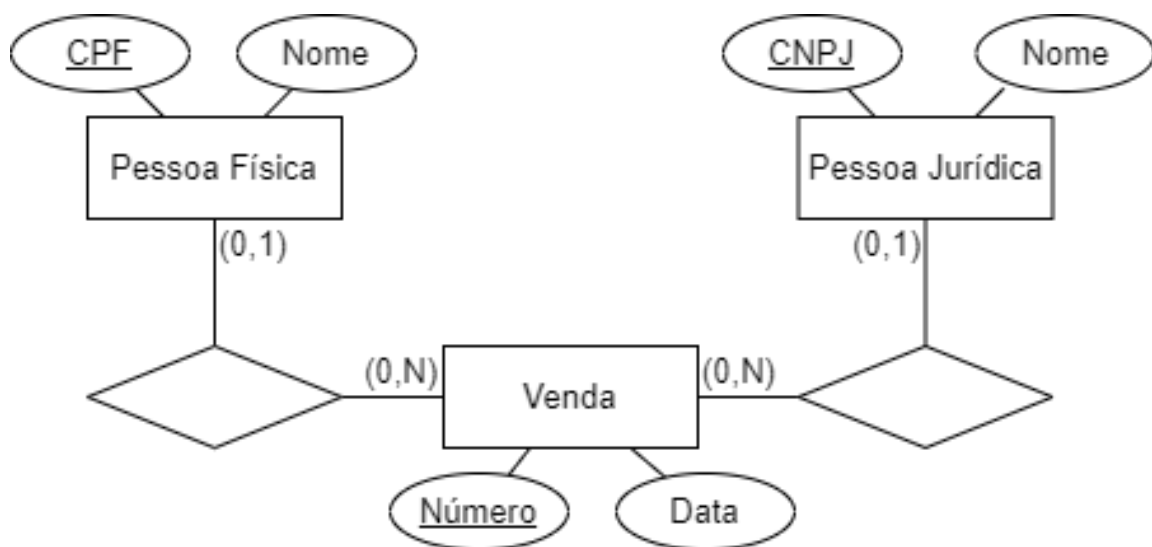
```
VENDA( numero, data, cpf, cnpj )  
  cpf referencia PESSOA_FISICA  
  cnpj referencia PESSOA_JURIDICA
```

Mapeamento obtido através da aplicação das regras de mapeamento ER para Relacional.

Refinamento do modelo relacional

Relacionamentos mutuamente exclusivos

- Considerando o diagrama abaixo, temos o resultado do modelo lógico relacional obtido a partir das regras de mapeamento vistas anteriormente:
 - As colunas **cpf** e **cnpj** são opcionais, já que para cada ocorrência de uma venda, apenas uma delas deve ser utilizada.
 - Ou a venda é feita para uma pessoa física (CPF) ou para uma pessoa jurídica (CNPJ).



```
PESSOA_FISICA( cpf, nome )
```

```
PESSOA_JURIDICA( cnpj, nome )
```

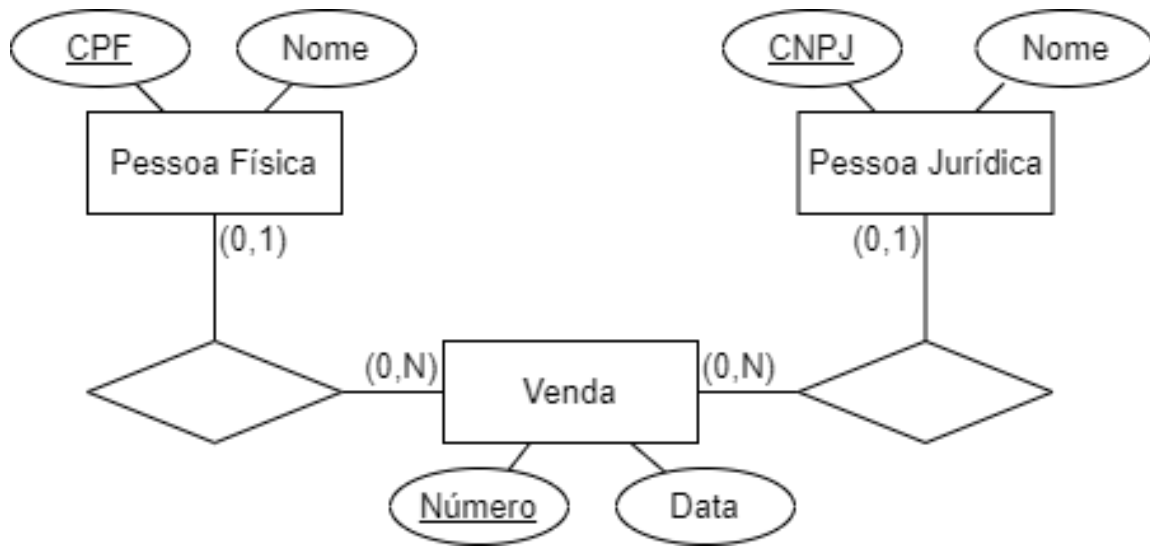
```
VENDA( numero, data, cpf_cnpj, tipo )
```

Em um mapeamento alternativo, pode-se criar uma única coluna na qual pode ser armazenado tanto o CPF quanto o CNPJ. Para isso, deve-se adicionar uma coluna para indicar qual das duas informações é utilizada.

Refinamento do modelo relacional

Relacionamentos mutuamente exclusivos

- Considerando o diagrama abaixo, temos o resultado do modelo lógico relacional obtido a partir das regras de mapeamento vistas anteriormente:
 - As colunas **cpf** e **cnpj** são opcionais, já que para cada ocorrência de uma venda, apenas uma delas deve ser utilizada.
 - Ou a venda é feita para uma pessoa física (CPF) ou para uma pessoa jurídica (CNPJ).



```
PESSOA_FISICA( cpf, nome )
```

```
PESSOA_JURIDICA( cnpj, nome )
```

```
VENDA( numero, data, cpf_cnpj, tipo )
```

Em um mapeamento alternativo, pode-se criar uma única coluna na qual pode ser armazenado tanto o CPF quanto o CNPJ. Para isso, deve-se adicionar uma coluna para indicar qual das duas informações é utilizada.

A desvantagem dessa alternativa é que não é possível especificar ao SGBD que o campo **cpf_cnpj** seja chave estrangeira a duas tabelas simultaneamente.

Refinamento do modelo relacional

Simulação de atributos multivalorados

- Considere o diagrama abaixo:



- Condições de contorno:

- Raros clientes possuem mais que dois telefones de contato.
 - Mesmo que essa situação ocorra, é suficiente armazenar apenas dois telefones.
- No exemplo acima, dificilmente será realizadas buscas no banco de dados usando o número de telefone como critério de seleção.
- Números de telefone são apenas exibidos ou impressos juntos às demais informações do cliente.

Refinamento do modelo relacional

Simulação de atributos multivalorados

- Considere o diagrama abaixo:



- Implementação não normalizada:

```
CLIENTE( cpf, nome, telefone_1, telefone_2 )
```

- Simula um atributo multivalorado através da criação de diversas colunas “telefone” sufixadas por um número (1, 2, ...)
- Vantagens:
 - Telefones dos clientes são obtidos rapidamente, junto com os demais dados do cliente.
 - Requer menos espaço, já que não é necessária a criação da chave primária da tabela **TELEFONE**.
- Desvantagens:
 - Consultas usando o número de telefone como critério de busca torna-se mais complicada e mais lenta.
 - Manter telefones “alinhados à esquerda” exige a criação de rotinas complexas.

Refinamento do modelo relacional

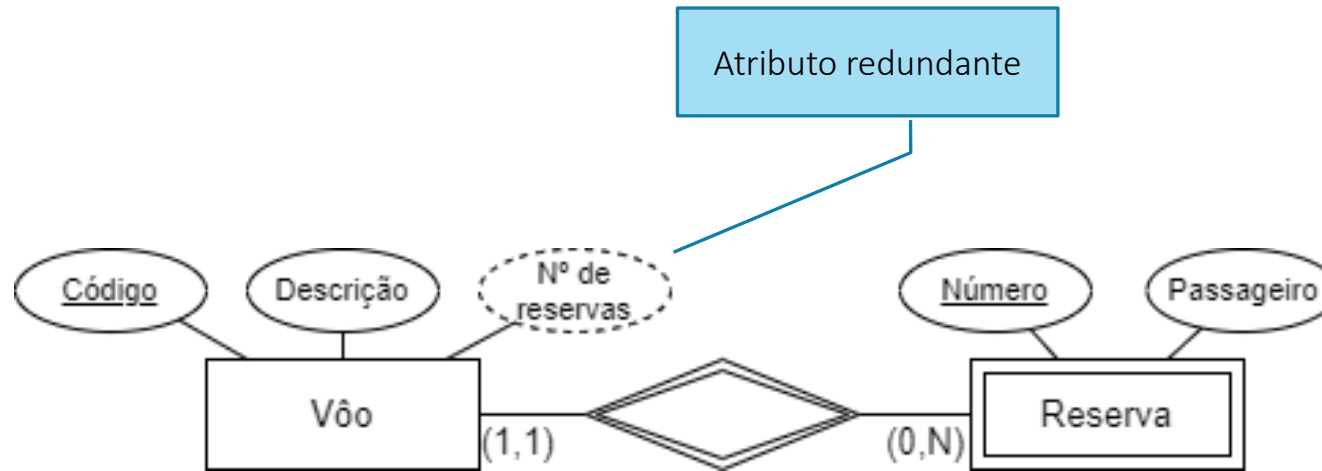
Informações redundantes

- Existem situações nas quais a presença de informações redundantes são capazes de melhorar o desempenho de um banco de dados.
- Atributos que resultam de operações complexas que envolvam diversas entidades do banco de dados.
 - O valor desses atributos é consultado com frequência?
 - O valor desses atributos serve como critério de busca no banco de dados?
- Nessas situações pode ser mais eficiente, considerando o desempenho global do sistema, armazenar redundantemente o atributo derivado.

Refinamento do modelo relacional

Informações redundantes

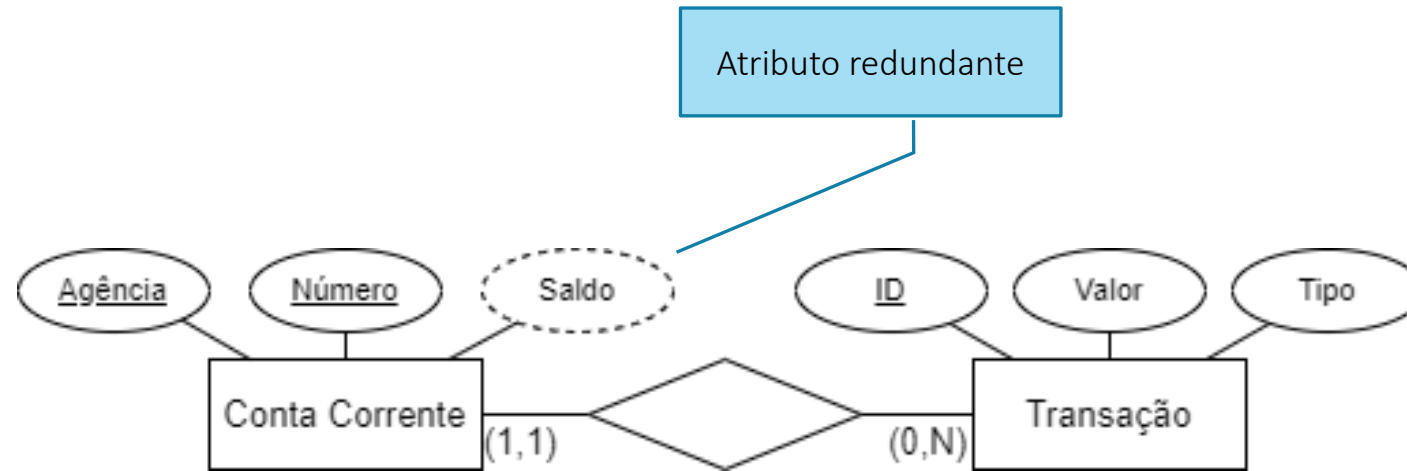
- Exemplos:



Refinamento do modelo relacional

Informações redundantes

- Exemplos:



Saiba mais!

Softwares e ferramentas online

Ferramentas online para modelagem de bancos de dados:

- Draw.io (online e desktop)
Disponível em: <https://app.diagrams.net/>
- Vertabelo (online)
Disponível em: <https://vertabelo.com/>
- DBDesigner (online)
Disponível em: <https://dbdesigner.id/>
- GenMyModel (online)
Disponível em: <https://www.genmymodel.com/>
- Lucidchart (online)
Disponível em: <https://www.lucidchart.com/>

Saiba mais!

Livros, artigos, cursos, páginas, documentos...

- CODD, Edgar F. A relational model of data for large shared data banks. Communications of the ACM, v.13 (6), 377–387. 1970.
Disponível em: <https://doi.org/10.1145/362384.362685>
- ORACLE BRASIL. O que é um banco de dados relacional?
Disponível em: <https://www.oracle.com/br/database/what-is-a-relational-database/>
Acesso em: 21 de out. de 2020.
- ESPOSITO, Dino. Documentos, bancos de dados e consistência eventual, 2015
Disponível em: <https://docs.microsoft.com/pt-br/archive/msdn-magazine/2014/august/cutting-edge-documents-databases-and-eventual-consistency>
Acesso em: 21 de out. de 2020.

Dúvidas?

André L. Maravilha

andre.maravilha@cefetmg.br
<https://andremaravilha.github.io/>

