

5.2.1 Implementação inicial de entidades

Esse passo é razoavelmente óbvio: cada *entidade* é traduzida para uma *tabela*. Neste processo, cada *atributo* da entidade define uma *coluna* desta tabela. Os atributos identificadores da entidade correspondem às colunas que compõem a chave primária da tabela.

Trata-se aqui de uma tradução *inicial*. Pelas regras que seguem nas próximas seções, as tabelas definidas nesta etapa ainda poderão ser fundidas, no caso de algumas alternativas de implementação de relacionamentos 1:1 e hierarquias de generalização/especialização.

A Figura 5.3 apresenta um exemplo da transformação de uma entidade em uma tabela. A figura mostra o DER e o correspondente esquema relacional. A entidade **PESSOA** com seus atributos **código**, **nome** e **endereço** é transformada na tabela denominada **Pessoa** com colunas denominadas **CodigoPess**, **Nome** e **Endereço**. Como o atributo **código** é identificador da entidade, a coluna correspondente a este atributo é a chave primária da tabela.



Esquema relacional correspondente:

Pessoa (CodigoPess, Nome, Endereço, DataNasc, DataAdm)

Figura 5.3: Transformação de entidade em tabela

5.2.1 Implementação inicial de entidades

Esse passo é razoavelmente óbvio: cada *entidade* é traduzida para uma *tabela*. Neste processo, cada *atributo* da entidade define uma *coluna* desta tabela. Os atributos identificadores da entidade correspondem às colunas que compõem a chave primária da tabela.

Trata-se aqui de uma tradução *inicial*. Pelas regras que seguem nas próximas seções, as tabelas definidas nesta etapa ainda poderão ser fundidas, no caso de algumas alternativas de implementação de relacionamentos 1:1 e hierarquias de generalização/especialização.

A Figura 5.3 apresenta um exemplo da transformação de uma entidade em uma tabela. A figura mostra o DER e o correspondente esquema relacional. A entidade **PESSOA** com seus atributos **código**, **nome** e **endereço** é transformada na tabela denominada **Pessoa** com colunas denominadas **CodigoPess**, **Nome** e **Endereço**. Como o atributo **código** é identificador da entidade, a coluna correspondente a este atributo é a chave primária da tabela.



Esquema relacional correspondente:

Pessoa (CodigoPess, Nome, Endereço, DataNasc, DataAdm)

Figura 5.3: Transformação de entidade em tabela

5.2.1.1 Nomes de atributos e nomes de colunas

Não é aconselhável simplesmente transcrever os nomes de atributos para nomes de colunas. Nomes de colunas serão referenciados freqüentemente em programas e outras formas de texto em computador. Assim, para diminuir o trabalho de programadores é conveniente manter os nomes de colunas curtos. Além disso, em um SGBD relacional, o nome de uma coluna não pode conter brancos. Assim, nomes de atributos compostos de diversas palavras devem ser abreviados. Com base nestas considerações, os nomes de atributos [data de nascimento](#) e [data de admissão](#) foram traduzidos para os nomes de colunas [DataNasc](#) e [DataAdm](#) respectivamente.

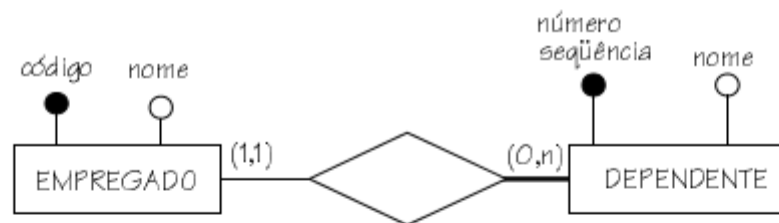
Nas linguagens de banco de dados, o nome da tabela é muitas vezes usado como qualificador do nome da coluna. Exemplificando, para referenciar a coluna [Nome](#) da tabela [Pessoa](#) muitas vezes é usado um termo na forma [Pessoa.Nome](#). Por isso, não é recomendado incluir no nome de uma coluna o nome da tabela em que ela aparece. Assim, é preferível usar o nome de coluna [Nome](#) a usar os nomes de coluna [NomePess](#) ou [NomePessoa](#). A exceção a esta regra é a coluna chave primária da entidade. Como esta coluna

pode aparecer em outras tabelas na forma de chave estrangeira, é recomendável que os nomes das colunas que compõem a chave primária sejam sufixadas ou prefixadas com o nome ou sigla da tabela na qual aparecem como chave primária. Por este motivo, a coluna chave primária da tabela do exemplo recebeu a denominação de [CodigoPess](#).

Outra recomendação quanto a nomeação de colunas é relativa ao uso de abreviaturas. Muitas vezes usa-se determinadas abreviaturas para tipos de campos que se repetem, como [Cod](#) para um código e [No](#) ou [Num](#) para um número. A recomendação é que se use sempre a mesma abreviatura em toda o banco de dados.

5.2.1.2 Relacionamento identificador

Há uma situação na qual a tradução de uma entidade para uma tabela não é trivial. Trata-se da situação na qual uma entidade possui um *relacionamento identificador*. Um exemplo de entidade deste tipo é a entidade **DEPENDENTE** mostrada no diagrama da Figura 5.4. Um dependente é identificado pelo código do empregado ao qual ele está vinculado e por um número de sequência que distingue os diversos dependentes de um mesmo empregado. A regra de tradução de identificadores externos é que, para cada identificador externo seja criada uma coluna (ou várias no caso de o identificador externo ser composto de vários atributos) na tabela em questão, coluna esta que fará parte da chave primária da tabela. A Figura 5.4 mostra o esquema relacional para esta tradução da entidade **DEPENDENTE**.

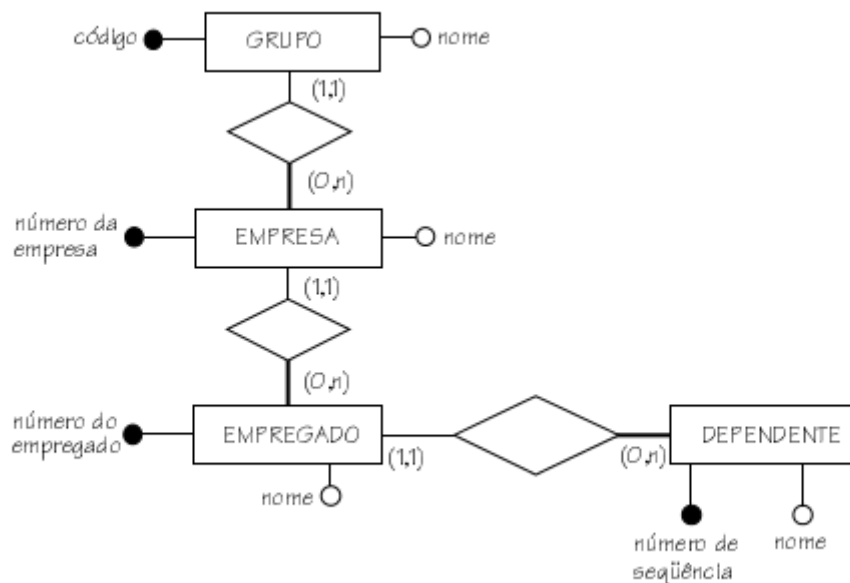


Esquema relacional correspondente:

Dependente (CodigoEmp,NoSeq,Nome)

Figura 5.4: Tradução de entidade com identificador externo

Cabe observar que, na composição da chave primária de uma tabela que possui identificador externo, pode ser necessário colecionar atributos de diversas entidades, conforme mostrado na Figura 5.5.



Esquema relacional correspondente:

[Grupo \(CodGrup, Nome\)](#)

[Empresa \(CodGrup, NoEmpresa, Nome\)](#)

[Empregado \(CodGrup, NoEmpresa, NoEmpreg, Nome\)](#)

[Dependente \(CodGrup, NoEmpresa, NoEmpreg, NoSeq, Nome\)](#)

Figura 5.5: Chave primária composta por diversos identificadores externos

No caso do exemplo, para compor a chave primária da tabela [Dependente](#), é necessário, usar além do número de sequência deste dependente, também o identificador do empregado. Entretanto, um empregado é identificado por seu número e pelo identificador da empresa a qual ele está vinculado. Por sua vez, a empresa é identificada por um número e pelo identificador do grupo ao qual ela pertence. Em outros termos, um dependente é identificado pela combinação das seguintes informações:

- ☐ código do grupo da empresa à qual seu empregado está vinculado
- ☐ número da empresa à qual seu empregado está vinculado
- ☐ número de seu empregado
- ☐ seu número de sequência.

Essa linha de raciocínio nos leva à chave primária da tabela [Dependente](#), que é mostrada na Figura 5.5.

5.2.2 Implementação de relacionamentos

O fator determinante para a tradução a adotar no caso de relacionamentos é a cardinalidade mínima e máxima das entidades que participam do relacionamento. Antes de detalhar a alternativa proposta para cada tipo de relaciona-

mento, apresentamos três formas básicas de tradução de relacionamentos para o modelo relacional.

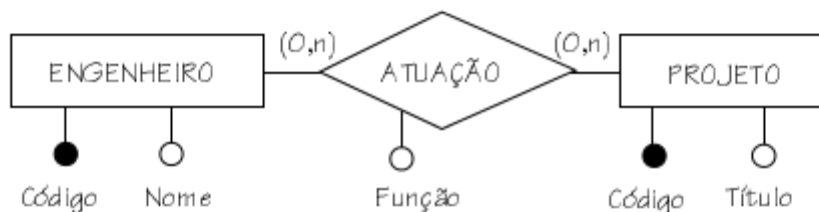
5.2.2.1 Tabela própria

Nesta tradução, o relacionamento é implementado através de uma *tabela própria*. Esta tabela contém as seguintes colunas:

- ❑ colunas correspondentes aos identificadores das entidades relacionadas
- ❑ colunas correspondentes aos atributos do relacionamento.

A chave primária desta tabela é o conjunto das colunas correspondentes aos identificadores das entidades relacionadas. Cada conjunto de colunas que corresponde ao identificador de uma entidade é chave estrangeira em relação a tabela que implementa a entidade referenciada.

Um exemplo deste tipo de tradução é apresentado na Figura 5.6. A parte do esquema do banco de dados que se refere à regra em questão está apresentada em negrito. Essa convenção será usada no restante da apresentação das regras de tradução de relacionamentos.



Esquema relacional correspondente:

Engenheiro (CodEng,Nome)

Projeto (CodProj,Título)

Atuação (CodEng,CodProj,Função)

CodEng referencia Engenheiro

CodProj referencia Projeto

Figura 5.6: Tradução de um relacionamento para uma tabela

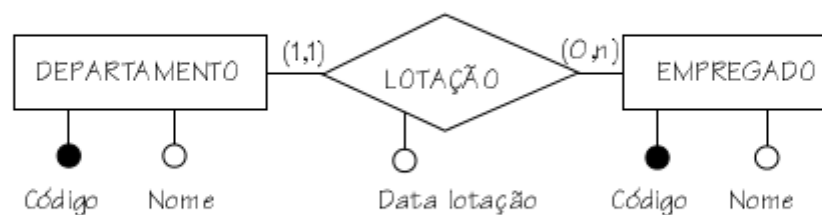
A tabela **Atuação** implementa o relacionamento **ATUAÇÃO**. A chave primária da tabela é formada pelas colunas **CodEng** e **CodProj**, que correspondem aos identificadores das entidades relacionadas (**ENGENHEIRO** e **PROJETO**). Cada uma destas colunas é chave estrangeira das tabelas que implementam as entidades relacionadas. A coluna **Função** corresponde ao atributo do relacionamento.

5.2.2.2 Colunas adicionais dentro de tabela de entidade

A outra alternativa de implementação de um relacionamento é a inserção de colunas em uma tabela correspondente a uma das entidades que participam do relacionamento. Um exemplo deste tipo de tradução é apresentado na Figura 5.7. Esse tipo de tradução somente é possível quando uma das entidades que participa do relacionamento tem cardinalidade máxima um (no caso do exemplo, trata-se da entidade EMPREGADO). A tradução consta em inse-

rir na tabela correspondente à entidade com cardinalidade máxima 1 as seguintes colunas:

- ❑ colunas correspondentes ao identificador da entidade relacionada (no caso do exemplo, o identificador da entidade DEPARTAMENTO); estas colunas formam uma chave estrangeira em relação à tabela que implementa a entidade relacionada - no caso do exemplo, a coluna **CodDept**
- ❑ colunas correspondentes aos atributos do relacionamento - no caso do exemplo, a coluna **DataLota**.



Esquema relacional correspondente:

Departamento (CodDept,Nome)

Empregado (CodEmp,Nome,**CodDept**,DataLota)

CodDept referencia Departamento

5.2.2.3 Fusão de tabelas de entidades

A terceira forma de implementar um relacionamento é através da fusão das tabelas referentes às entidades envolvidas no relacionamento. Um exemplo deste tipo de tradução é apresentado na Figura 5.8. Esta tradução somente pode ser aplicada quando o relacionamento é de tipo 1:1. A tradução consta de implementar todos atributos de ambas entidades, bem como os atributos do relacionamento em uma única entidade.



Esquema relacional correspondente:


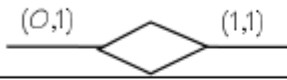
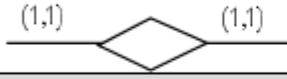
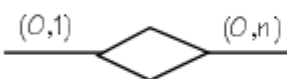

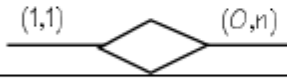

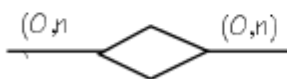
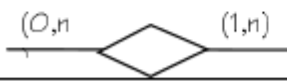
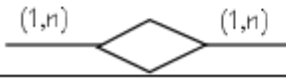
[Conferência \(CodConf, Nome, DataInstComOrg, EnderComOrg\)](#)

Figura 5.8: Tradução de relacionamento através de fusão de tabelas

5.2.3 Detalhes da implementação de relacionamentos

A regra específica que deve ser usada na tradução de um relacionamento é determinada pelas cardinalidades mínima e máxima das entidades envolvi-

das nos relacionamentos. A Tabela 5.1 dá uma visão geral das regras usadas. Para cada tipo de relacionamento a tabela mostra qual a alternativa que deve ser usada (alternativa preferida, indicada pelo símbolo ✓). Para alguns tipos de relacionamentos há ainda segunda alternativa a ser usada (indicada pelo símbolo ±), bem como uma alternativa que não deve ser usada (indicada pelo símbolo ×). Nas seções que seguem, a regra de tradução correspondente a cada um dos tipos de relacionamentos é justificada e exemplificada.

Tipo de relacionamento	Regra de implementação		
	Tabela própria	Adição coluna	Fusão tabelas
Relacionamentos 1:1			
	±	✓	×
	×	±	✓
	×	±	✓
Relacionamentos 1:n			
	±	✓	×
	±	✓	×
	×	✓	×
	×	✓	×
Relacionamentos n:n			
	✓	×	×
	✓	×	×
	✓	×	×

✓ Alternativa preferida ± Pode ser usada × Não usar

5.2.3.1.1 Ambas entidades têm participação opcional

A Figura 5.9 apresenta um exemplo de relacionamento 1:1 no qual a participação de ambas entidades é opcional (a cardinalidade mínima de ambas entidades no relacionamento é zero).

De acordo com a Tabela 5.1, a alternativa preferida de tradução de relacionamentos é a inserção de colunas na tabela referente a uma das entidades que participam do relacionamento. Como é um relacionamento 1:1, qualquer das entidades que participam do relacionamento pode ser a escolhida.

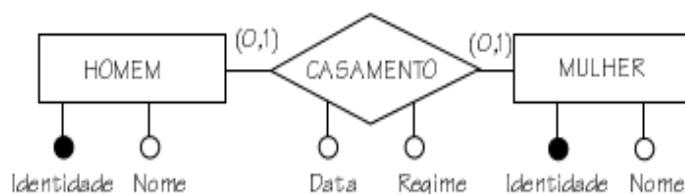


Figura 5.9: Implementação de relacionamento 1:1 com participação obrigatória de ambas entidades

Uma solução poderia ser a apresentada abaixo:

Mulher (IdentM, Nome, **IdentH**, Data, Regime)

IdentH referencia Homem

Homem (IdentH, Nome)

Neste esquema, as colunas referentes ao relacionamento estão marcadas em negrito. Trata-se de colunas referentes aos atributos de casamento, bem como a coluna **IdentH**, chave estrangeira que implementa o relacionamento. Neste caso, optou-se, arbitrariamente, por adicionar colunas à tabela **Mulher**. Da mesma forma, poderiam ter sido adicionadas colunas (identificador da mulher e atributos de casamento) à tabela **Homem**.

A outra alternativa seria a de gerar uma tabela própria para o relacionamento, conforme o esquema abaixo:

Mulher (IdentM, Nome)

Homem (IdentH, Nome)

Casamento (IdentM, **IdentH**, Data, Regime)

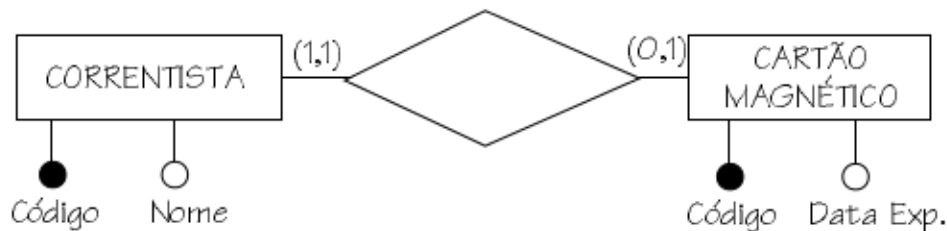
IdentM referencia Mulher

IdentH referencia Homem

A tabela que implementa o relacionamento é a tabela **Casamento**. Nesta tabela, as colunas **IdentH** e **IdentM** são ambas chaves estrangeiras, implementando desta forma a vinculação da linha de casamento às linhas de homem e mulher correspondentes. Como se trata de um relacionamento 1:1, tanto a coluna **IdentH** quanto a coluna **IdentM** podem ser consideradas para a chave primária. No exemplo, a coluna **IdentM** foi escolhida arbitrariamente como chave primária, sendo **IdentH** uma chave alternativa.

5.2.3.1.2 Uma entidade tem participação opcional e a outra tem participação obrigatória

Outra tipo de relacionamento 1:1; é aquele no qual uma das entidades tem participação *obrigatória*, enquanto que a outra entidade tem participação *opcional* (a cardinalidade mínima de uma das entidades é um, a cardinalidade mínima da entidade é zero). Um exemplo desta situação é apresentada na Figura 5.10.



Esquema relacional correspondente:

Correntista ([CodCorrent](#),[Nome](#),[CodCartão](#),[DataExp](#))

Figura 5.10: Implementação de relacionamento com participação obrigatória de uma entidade e participação opcional da outra

Neste caso, a tradução preferida é através da fusão das tabelas correspondentes às duas entidades.

Alternativamente, poderia ser considerada a tradução através da adição de colunas à tabela correspondente à entidade com cardinalidade mínima 0. No caso do exemplo, a tradução resultaria no esquema abaixo:

Correntista ([CodCorrent](#),[Nome](#))
Cartão ([CodCartão](#),[DataExp](#),[CodCorrent](#))
CodCorrent referencia Correntista

5.2.3.1.3 Ambas entidades tem participação obrigatória

O último tipo de relacionamentos 1:1 é aquele na qual ambas entidades tem participação *obrigatória* no relacionamento (a cardinalidade mínima de ambas entidades é um). Um exemplo desta situação é apresentada na Figura 5.11.



Esquema relacional correspondente:

[Conferência](#) ([CodConf](#),[Nome](#),[DataInstComOrg](#),[EnderComOrg](#))

Figura 5.11: Implementação de relacionamento 1:1 com participação obrigatória de ambas entidades

Neste caso, a tradução preferida é através da fusão das tabelas correspondentes às duas entidades.

Nenhuma das demais alternativas atende plenamente. Em ambas alternativas, as entidades que participam do relacionamento seriam representadas através de duas tabelas distintas. Estas tabelas teriam a mesma chave primária e relação um-para-um entre suas linhas. Essa implementação viola os princípios de evitar junções e diminuir o número de chaves primárias estabelecidos no início do capítulo.

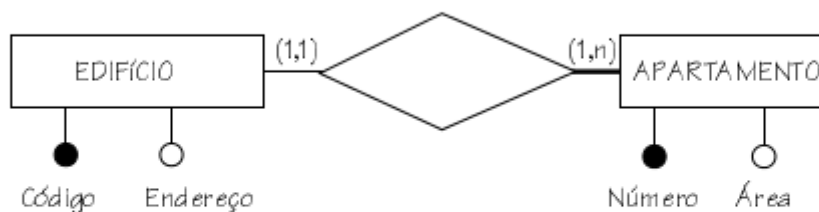
5.2.3.2 Relacionamentos 1:n

No caso de relacionamentos 1:n, a alternativa preferida de implementação é a de *adição de colunas* (Seção 5.2.2.2).

Um exemplo desta tradução é apresentado na Figura 5.12.

Cabe observar que no caso particular deste exemplo, a coluna **CódigoEd** da tabela **Apartamento** (que implementa o relacionamento do apartamento com seu edifício), além de ser chave estrangeira, é também parte da chave primária. Esta situação é típica de uma entidade com *relacionamento identificador*, cuja tradução foi discutida na Seção 5.2.1.2.

No caso de a entidade com cardinalidade máxima 1 ser opcional, isto é, possuir cardinalidade mínima 0, poderia ser considerada uma implementação alternativa. Um exemplo deste tipo de relacionamento é mostrado na Figura 5.13. A entidade **VENDA** está *opcionalmente* ligada a entidade **FINANCEIRA**.



Esquema relacional correspondente:

Edifício (CódigoEd, Endereço)

Apartamento (CódigoEd, NúmeroAp, ÁreaAp)
CódigoEd referencia Edifício

Figura 5.12: Tradução de relacionamentos 1:n através de adição de colunas

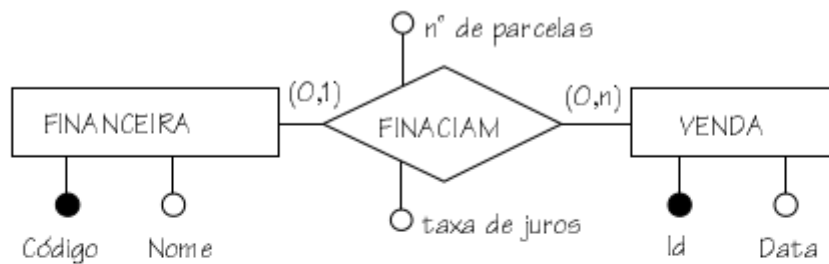


Figura 5.13: Tradução de relacionamentos 1:n no qual a entidade com cardinalidade máxima um é opcional

Para este tipo de relacionamento, pode ser usada alternativamente implementação através de tabela própria.

A implementação através de *adição de colunas* à tabela de entidade **Venda** (implementação preferida) é a seguinte:

Financeira (CodFin, Nome)

Venda (IdVend, Data, **CodFin**, **NoParc**, **TxJuros**)

CodFin referencia Financeira

As colunas em negrito na tabela Venda implementam o relacionamento.

A implementação através de *tabela própria* (implementação alternativa) é a seguinte:

Financeira (CodFin, Nome)

Venda (IdVend, Data)

Fianciam (IdVend, **CodFin**, **NoParc**, **TxJuros**)

IdVend referencia Venda

CodFin referencia Financeira

A implementação por tabela própria tem duas desvantagens em relação à implementação por adição de colunas:

- ❑ Operações que envolvem acesso a dados de uma venda e do respectivo financiamento exigem junções. Na primeira alternativa, isto não ocorre, já que os dados da venda e de seu financiamento estão na mesma linha.

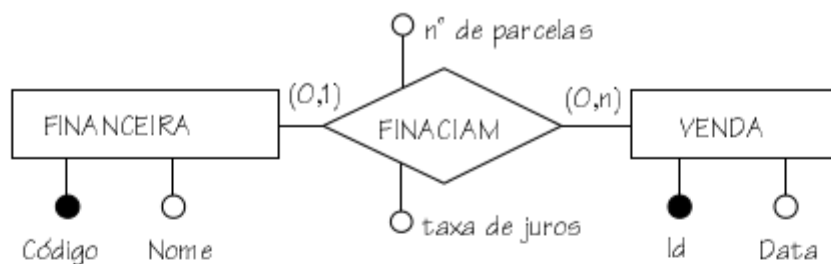


Figura 5.13: Tradução de relacionamentos 1:n no qual a entidade com cardinalidade máxima um é opcional

Para este tipo de relacionamento, pode ser usada alternativamente implementação através de tabela própria.

A implementação através de *adição de colunas* à tabela de entidade **Venda** (implementação preferida) é a seguinte:

Financeira (CodFin, Nome)

Venda (IdVend, Data, **CodFin**, **NoParc**, **TxJuros**)

CodFin referencia Financeira

As colunas em negrito na tabela Venda implementam o relacionamento.

A implementação através de *tabela própria* (implementação alternativa) é a seguinte:

Financeira (CodFin, Nome)

Venda (IdVend, Data)

Fianciam (IdVend, **CodFin**, **NoParc**, **TxJuros**)

IdVend referencia Venda

CodFin referencia Financeira

A implementação por tabela própria tem duas desvantagens em relação à implementação por adição de colunas:

- ❑ Operações que envolvem acesso a dados de uma venda e do respectivo financiamento exigem junções. Na primeira alternativa, isto não ocorre, já que os dados da venda e de seu financiamento estão na mesma linha.

- As tabelas [Venda](#) e [Fianciam](#) possuem a mesma chave primária, sendo o conjunto de valores de [Fianciam](#) um subconjunto de [Venda](#). Tem-se o problema acima mencionado de armazenamento e processamento duplicados de chave primária.

A única vantagem que a implementação por tabela própria apresenta é o fato de nela haver campos que são opcionais em certas linhas e obrigatórios em outras. Este é o caso dos campos [CodFin](#), [NoParc](#) e [TxJuros](#) da tabela [Venda](#) na alternativa de adição de colunas. Estes campos estão obrigatoriamente preenchidos em caso de venda à prazo e vazios em caso contrário.

5.2.3.3 Relacionamentos n:n

Independentemente da cardinalidade mínima, relacionamentos n:n, são sempre implementados através de uma tabela própria. Um exemplo de implementação de relacionamentos n:n é apresentado na Figura 5.6.

A alternativa de adicionar colunas a uma das tabelas correspondentes às entidades que participam do relacionamento não é aplicável. Cada entidade está associada a um número variável de entidades. Para implementar o relacionamento através da adição de colunas, seria necessária uma coluna *multi-valorada*, que comportasse um conjunto de valores de chaves primárias, referente à entidade associada. Entretanto, como vimos no capítulo anterior, as colunas na abordagem relacional são sempre *mono-valoradas*. Assim, esta alternativa não é viável pelas próprias características da abordagem relacional.

5.2.3.4 Relacionamentos de grau maior que dois

As regras apresentadas até este ponto, aplicam-se somente à implementação de relacionamentos binários, isto é, que envolvem apenas duas entidades. Para relacionamentos de grau maior que dois, não são definidas regras específicas. A implementação de um relacionamento de grau maior que dois dá-se na seguinte sequência de passos:

- 1 O relacionamento é transformado em uma entidade. Esta nova entidade é ligado através de um relacionamento binário a cada uma das entidades que participavam do relacionamentos original.
- 2 As regras de implementação de entidades e relacionamentos binários apresentadas acima são aplicadas às entidades e aos relacionamentos binários assim criados.

A Figura 5.14 mostra um relacionamento ternário junto com sua transformação em uma entidade e três relacionamentos binários.

A implementação deste modelo seguindo as regras apresentadas acima resulta no seguinte esquema relacional:

Produto (CodProd, Nome)

Cidade (CodCid, Nome)

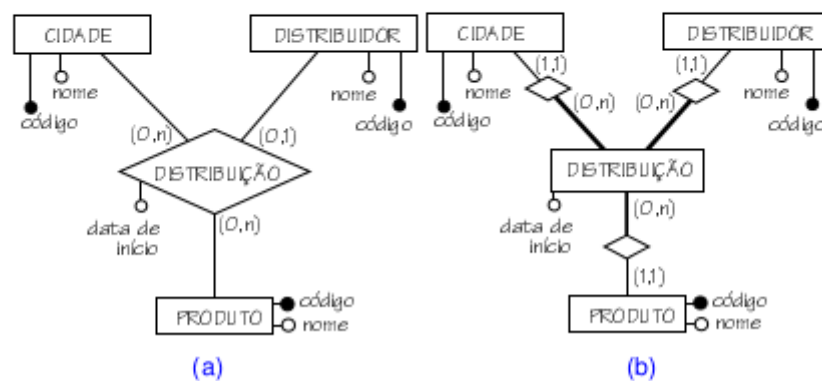
Distribuidor (CodDistr, Nome)

Distribuição (CodProd, CodDistr, CodCid, Nome)

CodProd referencia Produto

CodDistr referencia Distribuidor

CodCid referencia Cidade



5.2.4 Implementação de generalização/especialização

Para a implementação de hierarquias de generalização/especialização na abordagem relacional, há duas alternativas a considerar: (1) uso de uma tabela para cada entidade e (2) uso de uma única tabela para toda hierarquia de generalização/especialização. A seguir apresentamos as duas alternativas, para, depois, discutir quando usar uma ou outra.

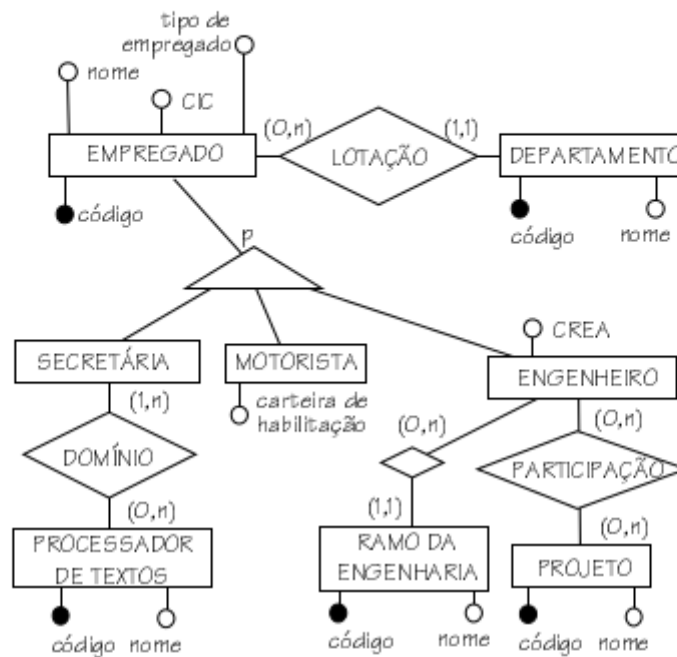
5.2.4.1 Uma tabela por hierarquia

Nesta alternativa, todas tabelas referentes às especializações de uma entidade genérica são fundidas em uma única tabela. Esta tabela terá:

- ☐ Chave primária correspondente ao identificador da entidade mais genérico
- ☐ Caso não exista, uma coluna **Tipo**, que identifica que tipo de entidade especializada está sendo representada por cada linha da tabela
- ☐ Uma coluna para cada atributo da entidade genérica
- ☐ Colunas referentes aos relacionamentos dos quais participa a entidade genérica e que sejam implementados através da alternativa de adicionar colunas à tabela da entidade genérica
- ☐ Uma coluna para cada atributo de cada entidade especializada (estas colunas devem ser definidas como opcionais, já que somente terão valores quando a linha for referente à entidade especializada em questão)
- ☐ Colunas referentes aos relacionamentos dos quais participa cada entidade especializada e que sejam implementados através da alternativa de adicionar colunas à tabela da entidade (estas colunas devem ser definidas como opcionais, já que somente terão valores quando a linha for referente à entidade especializada em questão)

Observe-se que, pela definição acima, uma entidade especializada pode não gerar nenhuma coluna na implementação. Isto ocorrerá caso a entidade especializada não tenha atributos e caso todos relacionamentos dos quais ele participe sejam implementados através de tabelas próprias.

Um exemplo de implementação usando uma única tabela para toda hierarquia de especialização da entidade **EMPREGADO** aparece na Figura 5.15.



Esquema relacional correspondente:

Emp (CódigoEmp,Tipo,Nome,CIC,CódigoDept,
CartHabil,CREA,CódigoRamo)

CódigoDept referencia Depto

CódigoRamo referencia Ramo

Depto (CódigoDept, Nome)

Ramo (CódigoRamo,Nome)

ProcessTexto (CódigoProc,Nome)

Domínio (CódigoEmp,CódigoProc)

CódigoEmp referencia Emp

CódigoProc referencia ProcessTexto

Projeto (CódigoProj,Nome)

Participação (CódigoEmp,CódigoProj)

CódigoEmp referencia Emp

CódigoProj referencia Projeto

Figura 5.15: Hierarquia de generalização/especialização e sua implementação através de tabela única

A tabela **Emp**, que implementa a entidade **EMPREGADO** e suas especializações, contém as seguintes colunas:

- ❑ **CódigoEmp**, chave primária da tabela, correspondente ao identificador da entidade
- ❑ As colunas **Tipo**, **Nome** e **CIC** referentes aos atributos da entidade genérica
- ❑ A coluna **CódigoDept**, que implementa o relacionamento **LOTAÇÃO**
- ❑ A coluna **CartHabil** que implementa os atributos da entidades especializada **MOTORISTA**
- ❑ A coluna **CREA** que implementa os atributos da entidade especializada **ENGENHEIRO**
- ❑ A coluna **CódigoRamo**, que implementa o relacionamento entre **ENGENHEIRO** e **RAMO DA ENGENHARIA**

Pelas regras apresentadas na seção anterior, os relacionamentos **PARTICIPAÇÃO** e **DOMÍNIO** são implementados por tabela própria, não gerando colunas na tabela correspondente à hierarquia de generalização/especialização. Além disso, a entidade **SECRETÁRIA** não gera nenhuma coluna já que não possui atributos nem participa de relacionamentos que gerem colunas.

As colunas que correspondem às entidades especializadas (**CartHabil**)

5.2.4.2 Uma tabela por entidade especializada

A outra alternativa de implementação de uma hierarquia de generalização/especialização é criar uma tabela para cada entidade que compõe a hierarquia, aplicando as regras correspondentes à implementação de entidades e relacionamentos já apresentadas nas seções anteriores.

O único acréscimo que deve ser feito àquelas regras é a inclusão da chave primária da tabela correspondente à entidade genérica., em cada tabela correspondente a uma entidade especializada. Exemplificando, a implementação do modelo ER da Figura 5.15 resultaria no seguinte esquema relacional (parte referente à hierarquia de generalização/especialização em negrito):

- ❑
- ❑
- ❑
- ❑
- ❑
- ❑ **Emp** (**CódigoEmp**, **Tipo**, **Nome**, **CIC**, **CódigoDept**)
- ❑ **CódigoDept** referência Depto
- ❑ **Motorista**(**CódigoEmp**, **CartHabil**)
- ❑ **CódigoEmp** referência Emp
- ❑ **Engenheiro**(**CódigoEmp**, **CREA**, **CódigoRamo**)
- ❑ **CódigoEmp** referência Emp
- ❑ **CódigoRamo** referência Ramo
- ❑ Depto (**CódigoDept**, **Nome**)
- ❑ Ramo (**CódigoRamo**, **Nome**)
- ❑ ProcessTexto (**CódigoProc**, **Nome**)
- ❑ Domínio (**CódigoEmp**, **CódigoProc**)
- ❑ **CódigoEmp** referência Emp
- ❑ **CódigoProc** referência ProcessTexto
- ❑ Projeto (**CódigoProj**, **Nome**)
- ❑ Participação (**CódigoEmp**, **CódigoProj**)
- ❑ **CódigoEmp** referência Emp
- ❑ **CódigoProj** referência Projeto

Para a entidade **EMPREGADO** e cada uma de suas especializações, foi criada uma tabela. Estas tabelas tem todas a mesma chave primária. A tabela **Emp** contém uma linha para cada empregado, independentemente de seu tipo. Nela aparecem as informações comuns a todos os empregados. As informações referentes a cada tipo particular de empregado estão nas tabelas **Motorista** e **Engenheiro**. Em cada uma destas tabelas aparecem linhas somente para empregados pertencentes ao tipo representado pela tabela.

Nas tabelas referentes às entidades especializadas, a chave primária é também chave estrangeira em relação à tabela de empregados. Isso ocorre porque a toda ocorrência de uma entidade especializada corresponde uma ocorrência de entidade genérica, ou seja, a toda linha de uma tabela de entidade especializada corresponde uma linha da tabela da entidade genérica.