

Projeto de Bases de Dados

Parte 2

Número	Nome	Contribuição	Esforço
76221	Emanuel Pereira	30%	3 horas
80832	Margarida Ferreira	35%	11 horas
83532	Miguel Marques	35%	6 horas

Grupo 42,
Turno BD2251795L07 (4ª feira, 11h),
Professor Miguel Amaral.

1. Modelo Relacional

Fornecedor(nif, nome)

Categoria(nome)

CategoriaSimples(nome)

- nome: FK(Categoria)

SuperCategoria(nome)

- nome: FK(Categoria)

RI-1: Categoria.nome não pode existir em CategoriaSimples.nome e SuperCategoria.nome simultaneamente

RI-2: Categoria.nome tem de existir em CategoriaSimples.nome ou em SuperCategoria.nome

constituída(super_nome, sub_nome)

- super_nome: FK(SuperCategoria.nome)
- sub_nome: FK(Categoria.nome)
- RI-3: constituída.super_nome tem de ser diferente de constituída.sub_nome.
- RI-4: Não podem existir ciclos na atribuição de sub_nome e super_nome nesta relação.

RI-5: SuperCategoria.nome tem de existir em constituída.super_nome.

Produto(ean, design, nif, data, nome)

- nif: FK(Fornecedor)
- nome: FK(Categoria)

fornece_sec(nif, ean)

- nif: FK(Fornecedor)
- ean: FK(Produto)

RI-6: Produto.ean tem de existir em fornece_sec.ean.

RI-7: Se Produto.ean for igual a fornece_sec.ean, Produto.nif tem de ser diferente de fornece_sec.nif.

Corredor(nro, largura)

Prateleira(nro, lado, altura)

- nro: FK(Corredor)

planograma(ean, nro, lado, altura, faces, unidades, loc)

- ean: FK(Produto)
- nro, lado, altura: FK(Prateleira.nro, Prateleira.lado, Prateleira.altura)

EventoReposição(operador, instante)

- RI-8: EventoReposição.instante tem de ser anterior ou igual ao momento atual

reposição(operador, instante, ean, nro, lado, altura, unidades)

- operador, instante: FK(EventoReposição.operador, EventoReposição.instante)
- ean, nro, lado, altura: FK(planograma.ean, planograma.nro, planograma.lado, planograma.altura)

RI-9: EventoReposição.operador, EventoReposição.instante tem de existir em reposição.operador, reposição.instante.

RI-10: Se reposição.ean é igual a planograma.ean, então reposição.unidades não pode ser maior que planograma.unidades.

2. Modelo Entidade-Associação vs. Modelo Relacional

2.1. Associações obrigatórias um-para-muitos e associações de entidade fraca

No modelo EA, as associações obrigatórias de um-para-muitos são representadas com uma seta e traço duplo na ligação de uma das entidades à associação. No modelo relacional, estas associações não justificam a criação de uma relação própria. De facto, para modelar o exemplo representado na Figura 1, podem usar-se apenas as seguintes relações, sem haver necessidade de qualquer restrição de integridade adicional:

Fornecedor(nif, nome)

Produto(ean, design, nif, data, nome)

- nif: FK(Fornecedor)
- nome: FK(Categoria)

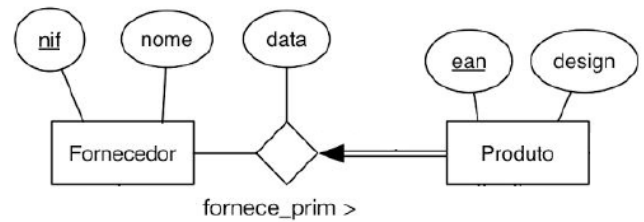


Fig. 1: Associação obrigatória um-para-muitos

Toda a informação relativa à associação fica assim guardada em Produto, nos atributos nif, nome e data.

A unicidade e obrigatoriedade da associação são garantidas pelas regras do modelo relacional. Por um lado, é impossível haver mais de uma atribuição para (nif, nome) para um dado Produto, o que garante que um Produto se associa apenas a um Fornecedor. Por outro lado, a obrigatoriedade advém da impossibilidade de existir uma instância de Produto sem que os atributos nif e nome tenham um valor atribuído.

No caso das associações de entidades fracas a situação é semelhante. Não há necessidade de criar uma nova relação apenas para a associação em si, bastando apenas adicionar a(s) chave(s) da entidade forte à(s) da entidade fraca, devidamente identificada(s) como *foreign key*.

2.2. Associações obrigatórias muitos-para-muitos

No modelo EA, as associações obrigatórias de muitos-para-muitos são representadas com um traço duplo num dos lados da associação. No modelo relacional, verifica-se a necessidade de representar a obrigatoriedade da associação na forma de restrição de integridade. O exemplo da figura 2 pode portanto ser modelado em modelo relacional criando, além das duas relações Fornecedor e Produto explicitadas em 2.2, a seguinte relação:

fornece_sec(nif, ean)

- nif: FK(Fornecedor)
- ean: FK(Produto)

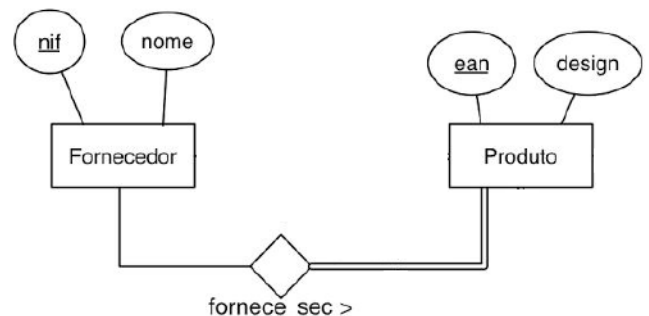


Fig. 2: Associação obrigatória muitos-para-muitos

No entanto, apenas estas relações não garantem a obrigatoriedade da associação, pois nada garante que um Produto está presente na relação fornece_sec. Uma vez que esta condição é necessária, acrescenta-se a seguinte restrição de integridade:

RI: Produto.ean tem de existir em fornece_sec.ean.

2.3. Especializações obrigatórias e disjuntas

As especializações obrigatórias e disjuntas são representadas no modelo EA através de um “X” e um traço duplo na ligação à super-entidade. A especialização apresentada na figura 3 é modelada em modelo relacional a partir das seguintes relações:

Categoria(nome)

CategoriaSimples(nome)

- nome: FK(Categoria)

SuperCategoria(nome)

- nome: FK(Categoria)

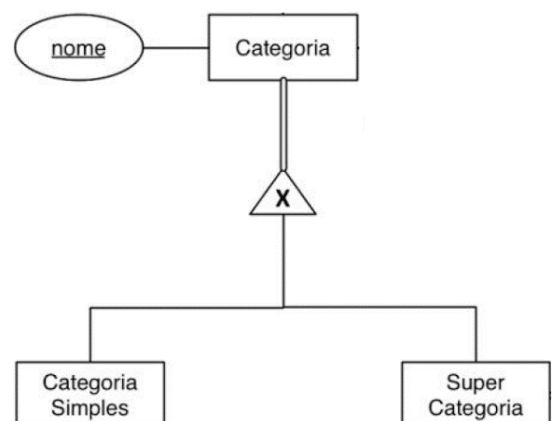


Fig. 3: Especialização obrigatória e disjunta

No entanto, apenas a definição destas relações não garante nem a obrigatoriedade nem a disjunção da especialização. Para tal, há que garantir que:

1. Uma Categoria não pertence simultaneamente a CategoriaSimples e SuperCategoria. Para isso, acrescentamos a seguinte restrição de integridade:

RI: Categoria.nome não pode existir em CategoriaSimples.nome e SuperCategoria.nome simultaneamente

2. Uma Categoria tem de pertencer obrigatoriamente a CategoriaSimples ou a SuperCategoria:

RI: Categoria.nome tem de existir em CategoriaSimples.nome ou em SuperCategoria.nome

3. Álgebra Relacional

1. $\pi_{\text{ean}, \text{design}} (\sigma_{\text{sum} > 10} (\text{ean}, \text{design} \bowtie \text{Gsum}(\text{unidades}) \text{ as } \text{sum}(\sigma_{\text{instante} > 10-01-2017, \text{nome} = \text{"Fruta"} (\text{Produto} \bowtie \text{reposição}))))$
2. Para um dado ean EAN:
 $\pi_{\text{nome}, \text{nif}} (\sigma_{\text{ean} = \text{EAN}} (\pi_{\text{ean}, \text{nif}} (\text{Produto}) \bowtie \text{Fornecedor})) \cup$
 $\pi_{\text{nome}, \text{nif}} (\sigma_{\text{ean} = \text{EAN}} (\text{fornece_sec} \bowtie \text{Fornecedor}))$
3. $\text{Gcount}() (\sigma_{\text{super_nome} = \text{"Congelados"} (\text{constituída}))$
4. $\text{fornecedor_count} \leftarrow \text{nome_cat} \text{Gcount}() \text{ as } \text{count}(\pi_{\text{nome_cat}, \text{nif}, \text{nome}} (\rho_{\text{nome} \mapsto \text{nome_cat}} (\text{Produto}) \bowtie \text{Fornecedor}) \cup \pi_{\text{nome_cat}, \text{nif}, \text{nome}} (\rho_{\text{nome} \mapsto \text{nome_cat}, \text{nif} \mapsto \text{nif_prim}} (\text{Produto}) \bowtie \text{fornece_sec} \bowtie \text{Fornecedor}))$
 $\sigma_{\text{cout} = \text{max}} (\text{fornecedor_count} \times \text{Gmax}(\text{count}) \text{ as } \text{max}(\text{fornecedor_count}))$
5. $(\pi_{\text{nif}, \text{nome}} (\text{Produto}) \div \text{CategoriaSimples}) \bowtie \text{Fornecedor}$
6. $\pi_{\text{nro}, \text{nif}} (\text{planograma} \bowtie \text{Produto}) \div (\pi_{\text{nif}} (\text{Produto}) - \pi_{\text{nif}} (\text{fornece_sec}))$

4. SQL

1.

```
SELECT ean, design
FROM (
    SELECT ean, design, SUM(unidades)
    FROM produto NATURAL JOIN reposição
    WHERE instante > "10/01/2017" AND nome = "Fruta"
    GROUP BY ean, design
    HAVING SUM(unidades) > 10
);
```
2. Para um dado ean EAN:

```
(SELECT nome, nif
FROM (SELECT nif FROM produto) NATURAL JOIN fornecedor
WHERE ean = EAN)
UNION
(SELECT nome, nif
FROM fornece_sec NATURAL JOIN fornecedor
WHERE ean = EAN);
```