

Aula 2 – Modelos e modelagem de dados

Objetivos

Identificar os modelos de dados existentes.

Identificar o processo de modelagem de dados utilizando o Modelo Entidade Relacionamento.

2.1 Modelos de dados

Os SGBDs evoluíram de sistemas de arquivos de armazenamento em disco, criando novas estruturas de dados com o objetivo de armazenar dados de forma mais eficiente e segura. Com o passar do tempo, os SGBDs passaram a utilizar diferentes formas de representação, ou modelos de dados, para descrever a estrutura das informações contidas em seus bancos de dados. A escolha do modelo de dados é a **principal ferramenta** no fornecimento de informações sobre a abstração realizada na parte de interesse específico no mundo real. Dentre os modelos existentes que normalmente são implementados pelos SGBDs podemos citar o modelo hierárquico, modelo em redes, modelo relacional (mais utilizado atualmente) além do modelo orientado a objetos.

2.1.1 Modelo hierárquico

Segundo TAKAI, ITALIANO & FERREIRA (2005, p. 6) o modelo hierárquico foi o primeiro modelo de dados. Nesse modelo de dados, os dados são estruturados em hierarquias ou árvores. Os nós das hierarquias contêm ocorrências de registros, onde cada registro é uma coleção de campos (atributos), cada um contendo apenas uma informação. O registro da hierarquia que precede a outros é o registro-pai, os outros são chamados de registros-filhos.

Uma ligação é uma associação entre dois registros. O relacionamento entre um registro-pai e vários registros-filhos possui cardinalidade 1:N (um para muitos, ou seja um pai pode ter vários filhos). Os dados organizados conforme este modelo podem ser acessados como uma sequência hierárquica com uma navegação do topo para as folhas e da esquerda para a direita. Um registro pode estar associado a vários registros diferentes, desde que seja replicado. (TAKAI, ITALIANO & FERREIRA, 2005, p. 6) A replicação possui

desvantagens: pode causar inconsistência de dados quando houver atualização e o desperdício de espaço é inevitável. Características:

- cada pai pode ter vários filhos.
- cada filho pode ter apenas um pai.
- duas entidades podem possuir apenas um relacionamento.
- qualquer recuperação de dados deve obrigatoriamente percorrer a estrutura da árvore. (pai à filho à neto...).

Como exemplos de banco de dados que utilizam o modelo hierárquico podemos citar: IMS e System 2.



Vamos tomar como exemplo, uma agência bancária como pai de um relacionamento pode ter vários clientes e um cliente pode ter várias contas, porém se um cliente pertencer a mais de uma agência, seus dados terão que ser replicados, o que pode causar inconsistência e desperdício de espaço.

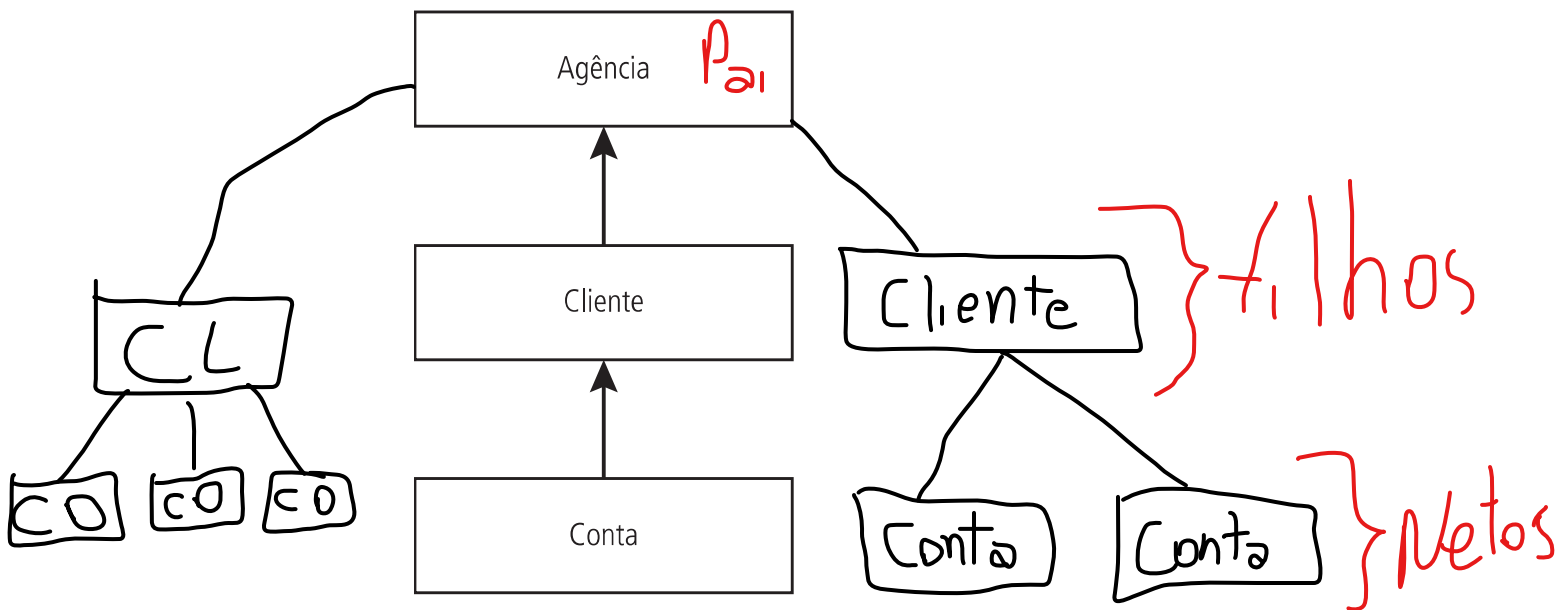


Figura 2.1: Exemplo modelo hierárquico

Fonte: Elaborada pelo autor

2.1.2 Modelo em rede

O modelo em rede surgiu como uma extensão ao modelo hierárquico, eliminando o conceito de hierarquia e permitindo que um mesmo registro estivesse envolvido em várias associações. No modelo em rede, os registros são organizados em grafos onde aparece um único tipo de associação que define uma relação 1:N (um para muitos) entre 2 tipos de registros: proprietário e membro (TAKAI , ITALIANO & FERREIRA, 2005, p. 7).

De modo contrário ao modelo hierárquico, em que qualquer acesso aos dados passa pela raiz, o modelo em rede possibilita acesso a qualquer nó da rede sem passar pela raiz, pois o modelo em rede permite a existência de entidades pais com muitos filhos e de entidades filhos com muitos pais.

Os modelos hierárquicos e em rede são orientados a registros isto é, a qualquer acesso à base de dados como inserção, consulta, alteração ou remoção é feito em um registro de cada vez (TAKAI , ITALIANO & FERREIRA, 2005, p. 7).

O modelo em rede apresenta como principal vantagem a possibilidade de uma modelagem mais próxima da realidade, porém não se firmou no mercado pelo surgimento do modelo relacional que veremos a seguir.

Vamos tomar como exemplo a Figura 2.2, um equipamento pode ter várias bombas e motores, e estes dois podem sofrer manutenção mecânica. Assim neste modelo perde-se a restrição hierárquica.

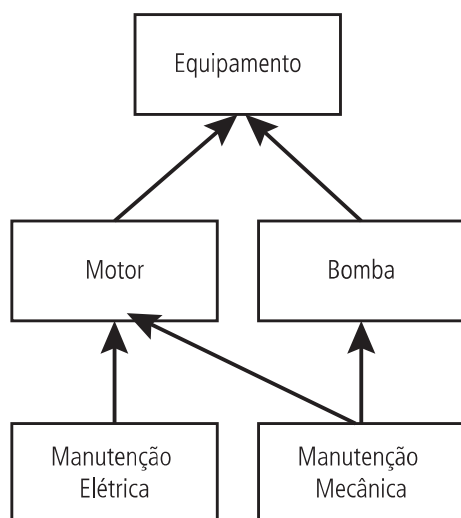


Figura 2.2: Exemplo modelo em rede

Fonte: Elaborada pelo autor

São exemplos de SGBDs em rede: DBMS10, IDSII, DMSII e IMAGE.

2.1.3 Modelo relacional

Os SGBDs que apresentaram maior sucesso e confiabilidade no mercado são os que adotaram o Modelo Relacional como base para sua construção. O Modelo Relacional, de modo geral, permite que os dados sejam “vistos” como tabelas. Ora, esta é uma maneira muito natural de armazenar e recuperar dados. Geralmente, quando fazemos uma relação de dados, tendemos a organizá-los em linhas e colunas, no formato de tabelas. Esta é uma importante razão para o sucesso das planilhas eletrônicas no mundo dos negócios.

O modelo relacional surgiu devido às necessidades de **aumentar a independência de dados** nos sistemas gerenciadores de banco de dados, proporcionando um conjunto de funcionalidades apoiadas em álgebra relacional para armazenamento e recuperação de dados além de permitir processamento dedicado. O modelo relacional revelou-se ser o mais **flexível** e adequado ao solucionar os vários problemas que se colocam no nível da concepção e implementação da base de dados. A estrutura fundamental do modelo relacional é a relação (tabela). Uma relação é constituída por um ou mais atributos (campos) que traduzem o tipo de dados a armazenar. Cada instância do esquema (linha) é chamada de tupla (registro). Este modelo não possui caminhos pré-definidos para se fazer acesso aos dados como nos modelos anteriores. O modelo relacional implementa estruturas de dados organizadas em relações, porém, para trabalhar com essas tabelas, algumas restrições precisaram ser impostas para evitar redundância, perda de dados e incapacidade de representar parte da informação. Essas restrições são: integridade referencial, chaves e integridade de junções de relações (TAKAI, ITALIANO & FERREIRA, 2005, p. 8).



Vamos tomar como exemplo, um aluno pode cursar várias disciplinas e uma disciplina pode ser cursada por vários alunos criando assim uma relação muitos para muitos entre as tabelas Aluno e Disciplina.

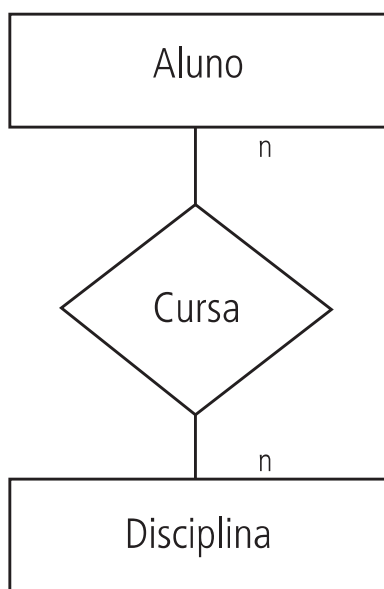


Figura 2.3: Exemplo de Modelo Entidade Relacionamento (MER)

Fonte: Elaborada pelo autor

Atualmente, mesmo que informalmente, o Modelo Relacional é visto no mercado quase como um sinônimo de Sistemas de Banco de Dados. Normalmente, quando nos referimos a banco de dados, os usuários já imaginam um conjunto de tabelas relacionadas.

Em nossa disciplina utilizaremos um SGBD que utiliza o Modelo Relacional como base para sua construção. Através das regras e conceitos deste modelo, “enxergaremos” nosso banco de dados implementado computacionalmente como um conjunto de tabelas inter-relacionadas.

2.1.4 Modelo de dados orientado à objeto

Representam os dados como coleções que obedecem a propriedades. São modelos geralmente conceituais dispondo de poucas aplicações reais. Neste Modelo não seria interessante a existência de uma tabela de funcionários e dentro dela alguma referência para cada registro, de forma a podermos saber onde (em que departamento) o funcionário está alocado. Um conjunto de regras disponibilizaria em separado os funcionários da fábrica, que, no entanto, estariam agrupados aos demais, para o sistema de folha de pagamento.

2.2 Modelagem de dados

Utilizaremos para nossa modelagem o modelo Entidade Relacionamento, pois este é um modelo de dados conceitual de alto nível, cujos conceitos fo-

ram projetados para estar mais próximo possível da visão que o usuário tem dos dados, não se preocupando em representar como estes dados estarão realmente armazenados de maneira física. O modelo ER é utilizado principalmente durante o processo de projeto conceitual de banco de dados.

Quando pensamos no Modelo Entidade Relacionamento temos três conceitos básicos:

- a) **Conjunto de Entidades:** uma entidade pode ser vista como uma pessoa, “coisa” ou “objeto” no mundo real que é distinguível de todos os outros objetos, como por exemplo, um cliente de um banco.



Figura 2.4: Entidade Cliente

Fonte: Elaborada pelo autor

Já um conjunto é um grupo de entidades do mesmo tipo que compartilham os mesmos atributos, como por exemplo, um conjunto de clientes bancários, animais ou pessoas.

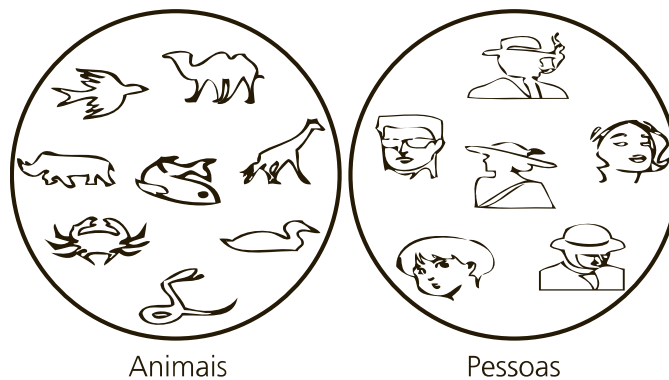


Figura 2.5: Conjunto de Entidades mundo real

Fonte: Elaborada pelo autor

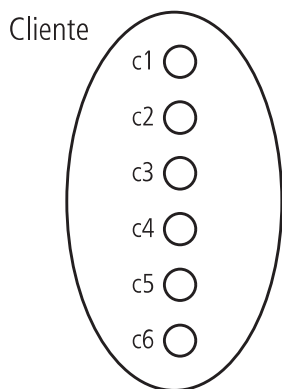


Figura 2.6: Conjunto de Entidades Banco de Dados

Fonte: Elaborada pelo autor

característ dos dados ligados à entidade

b) Conjunto de Atributos: uma entidade é representada por um conjunto de atributos. Possíveis atributos do conjunto de entidades cliente, podem ser: nome-cliente, cpf, rua e cidade-cliente. Possíveis atributos do conjunto de entidade **CONTA** são: número-conta e saldo. Para cada atributo, existe um conjunto de valores permitidos chamado **domínio** = *char, int, float*. daquele atributo. O domínio do atributo nome-cliente pode ser o conjunto de todas as cadeias de texto (*strings*) de certo tamanho.

Por exemplo, podemos limitar o tamanho do nome de uma pessoa a 50 caracteres, então este é seu domínio. Como também, o domínio do atributo número-conta pode ser o conjunto de todos os inteiros positivos. O atributo idade de uma entidade pessoa poderia ter como domínio os números inteiros entre 0 e 150.



Formalmente, um atributo é uma função que mapeia um conjunto de entidades em um domínio.

c) Conjunto de Relacionamentos: um **relacionamento** é uma associação entre diversas entidades. Por exemplo, podemos definir um relacionamento que associa o cliente José Silva à conta 401. Isto especifica que José Silva é um cliente com conta bancária número 401. Um **conjunto de relacionamentos** é uma coleção de relacionamentos do mesmo tipo. Esse relacionamento pode ser representado pelo diagrama mostrado na Figura 2.7 a seguir.



Figura 2.7: Exemplo relacionamento entre cliente e conta

Fonte: Elaborada pelo autor

Nosso projeto conceitual de Banco de Dados será feito através da modelagem de dados usando o Modelo Entidade-Relacionamento (MER). Este modelo fornece as regras e conceitos para a criação de um Diagrama Entidade-Relacionamento (DER), que deverá representar o banco de dados em questão, como por exemplo, um banco com vários clientes e contas, uma loja de produtos de beleza, de um consultório odontológico, uma indústria de peças, uma locadora de vídeo, uma escola, enfim, o negócio a que este banco de dados deve servir. Desta maneira, usaremos dois modelos teóricos para a construção de um banco de dados:

- d) o modelo entidade relacionamento para criação do projeto conceitual do banco de dados.
- e) o modelo relacional para a implementação em um ambiente computacional pré-existente. Este ambiente computacional pré-existente será o nosso Sistema Gerenciador de Banco de Dados Relacional - MySQL.

Com intuito de realizar a transferência entre modelos serão aplicadas as regras de **mapeamento** do Modelo Entidade Relacionamento ME-R para o Modelo Relacional MRel. Ou seja, para “converter” o diagrama conceitual de nosso banco de dados em um conjunto de tabelas relacionadas que possa ser implementado em um SGBD Relacional, usaremos um conjunto de passos. Este processo é chamado **mapeamento** do MER para o MRel.

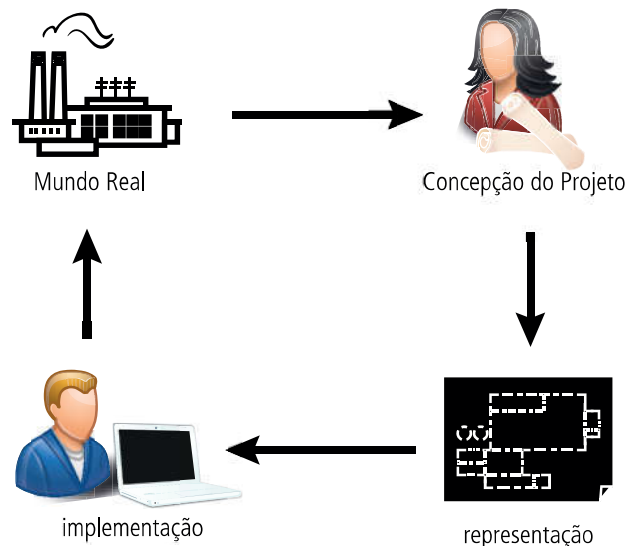


Figura 2.8: Representação mundo real para modelagem banco de dados

Fonte: Elaborada pelo autor

Para realizarmos o mapeamento dos dados entre os modelos, primeiramente utilizaremos os requisitos coletados para construção e realizaremos a modelagem utilizando o Modelo Entidade Relacionamento, em seguida “converteremos” os diagramas gerados em tabelas sobre o modelo relacional para implementarmos em um SGBD.

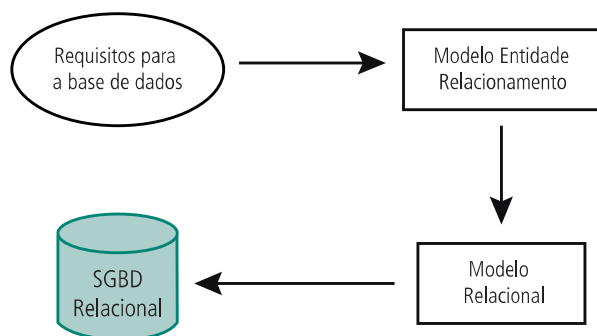


Figura 2.9: Implementação de um BD

Fonte: Elaborada pelo autor

Agora veremos como construir um diagrama Entidade Relacionamento DER para o projeto de um banco de dados. Este será o primeiro passo para a construção de nosso banco de dados. O Modelo Entidade Relacionamento MER é composto por Entidades e Relacionamentos sem nos esquecermos dos atributos.

2.2.1 Representações

A estrutura lógica geral de um banco de dados pode ser expressa graficamente por um diagrama ER, que consiste nos seguintes componentes:

1. Retângulos que representam conjuntos de entidades;
2. Elipses que representam atributos;
3. Losangos que representam relacionamentos entre conjuntos de entidades;
4. Linhas que ligam atributos a conjuntos de entidades e conjuntos de entidades a relacionamentos. Alguns autores chamam as linhas de arestas, em analogia às teorias de grafos e redes.

2.2.2 Entidades e atributos

Como descrito acima, a representação para um conjunto de entidades é um retângulo e, para cada atributo, uma elipse, como por exemplo, um conjunto de entidades Aluno pode possuir os atributos **nome**, **prontuário** e **data de nascimento** (Figura 2.10).

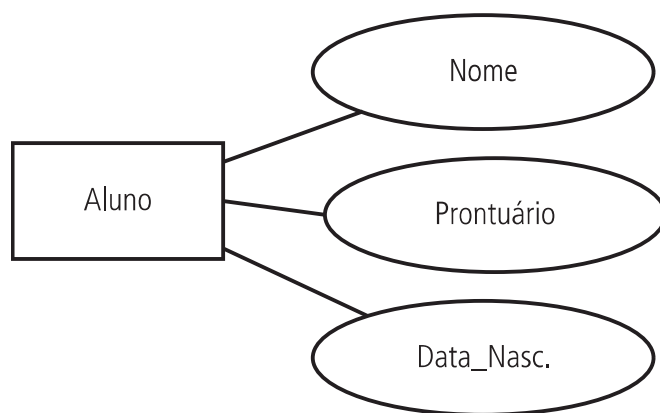


Figura 2.10: Entidade com seus atributos

Fonte: Elaborada pelo autor

Este conjunto de entidades ALUNO possui os conjuntos de atributos Nome, Prontuario e Data_Nasc. Uma entidade desse conjunto poderia ser {'Iago', 10123, '2000-12-16'}.

2.2.3 Chave de um conjunto de entidades

É importante poder especificar como entidades e relacionamentos são identificados. Conceitualmente, entidades e relacionamentos individuais são distintos, mas em uma perspectiva de banco de dados a diferença entre eles precisa ser expressa em termos de seus atributos. Uma restrição importante sobre entidades é a aplicação de uma chave única. Um conjunto de entidades sempre possui um atributo cujo valor é diferente e válido para cada entidade. Tal atributo é chamado atributo-chave e seu valor é usado para identificar cada entidade de modo unívoco, único, como por exemplo, o atributo CPF de uma pessoa é sempre único, ou o prontuário de um aluno é sempre único. A chave de um atributo será sempre sublinhada.

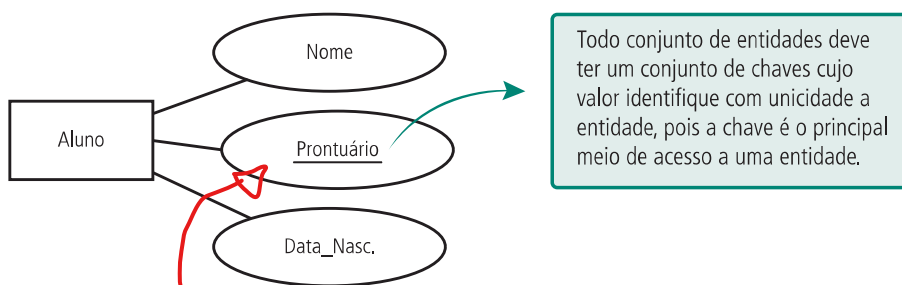


Figura 2.11: Atributo chave de uma entidade

Fonte: Elaborada pelo autor

É possível ocorrer situações onde é preciso mais de um atributo para identificar cada entidade do conjunto. Por exemplo, o RG (registro geral) dos brasileiros não identifica um único cidadão. É possível, que um mesmo número de RG possa ocorrer em duas unidades diferentes da federação, ou seja, um mesmo número de RG poderia ser emitido em São Paulo e Minas Gerais. Assim, para ser identificador, o RG precisa ser composto com o valor do Órgão Emissor, no caso de São Paulo, SSP_SP. Estes dois atributos (RG e Órgão Emissor) se, juntos, identificam cada brasileiro. Nestes casos, quando é necessário mais de um atributo, a chave do conjunto de entidades será composta.

chave composta

2.2.4 Relacionamentos

Nenhuma informação armazenada no Banco de Dados existe isoladamente. Todos os elementos pertencentes ao mundo real (restrito) modelado de alguma forma estão associados a outros elementos. Normalmente estas associações representam ações físicas ou alguma forma de dependência entre os elementos envolvidos. Um relacionamento é uma associação entre diversas entidades.

Considere um conjunto de relacionamentos TRABALHA entre os conjuntos de entidades: EMPREGADO e DEPARTAMENTO que é apresentado na Figura 2.18. Este relacionamento associa cada empregado com o departamento em que ele trabalha. Cada instância de relacionamento em TRABALHA associa uma entidade “empregado” a uma entidade “departamento”. Cada instância de relacionamento conecta uma entidade EMPREGADO a uma entidade DEPARTAMENTO. Na Figura 2.18, os empregados e1, e3 e e6 trabalham para o departamento d1; e2 e e4 trabalham para d2; e e5 e e7 trabalham para d3.



A representação dos conjuntos de entidades e de relacionamentos apresentadas acima no Modelo Entidade Relacionamento é representada na Figura 2.12.



Figura 2.12: Modelo ER para relacionamento TRABALHA

Fonte: Elaborada pelo autor

O conjunto de relacionamentos é, portanto, representado por um losango. Enquanto que para os conjuntos de entidades os atributos são obrigatórios, para os conjuntos de relacionamentos, eles são optativos. O conjunto de entidade só faz sentido quando especificamos seus atributos.

Já um conjunto de relacionamentos (CR), nem sempre precisa possuir atributos. Sua existência justifica-se apenas pela função de relacionar uma ou mais entidades. Em geral, os atributos dos conjuntos de relacionamentos, quando existem, especificam dados sobre tempo (data, horário), quantidades, valores, enfim, atributos relativos a transações, ações, ocorrências, que caracterizam os relacionamentos.

2.2.5 Grau de relacionamento

O grau de um conjunto de relacionamentos indica o número de conjuntos de entidades participantes. Um tipo de relacionamento de grau dois é chamado **binário**, de grau três de **ternário**, de grau quatro **quaternário**, acima disso, n-ário. A quantidade de Entidades envolvidas em um Relacionamento pode ser determinada por sua semântica. Desta forma, podem-se categorizar os graus de relacionamento em:

a) Unário: é o grau de Relacionamento que envolve um único Tipo de Entidade.

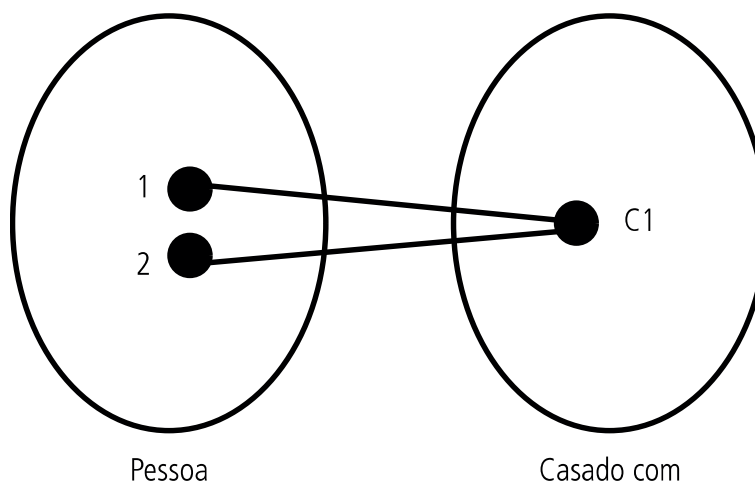


Figura 2.13: Exemplo de relacionamento Unário

Fonte: Elaborada pelo autor

Binário: é o grau de Relacionamento que envolve dois Tipos de Entidades.

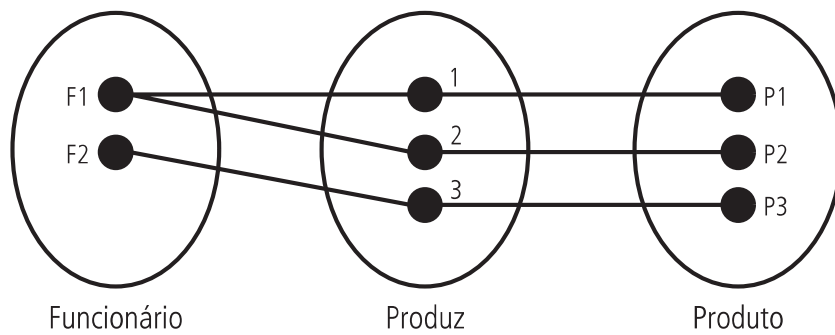


Figura 2.14: Exemplo de relacionamento binário

Fonte: Elaborada pelo autor

b) Ternário: é o grau de Relacionamento que envolve três Tipos de Entidades.

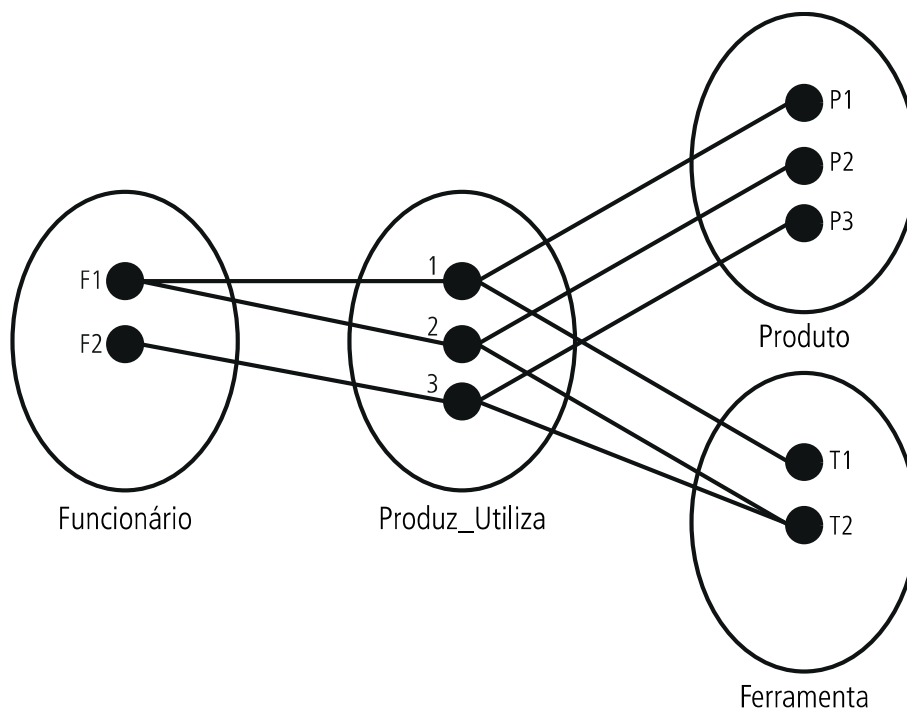


Figura 2.15: Exemplo de relacionamento ternário

Fonte: Elaborada pelo autor

- c) **Quaternário**: é o grau de Relacionamento que envolve quatro Tipos de Entidades.

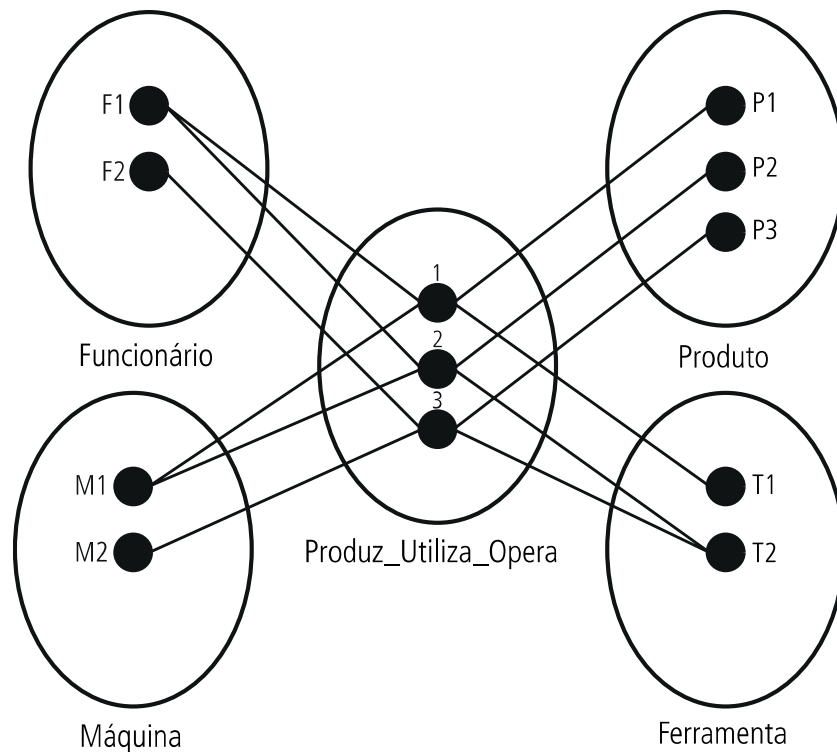


Figura 2.16: Exemplo de relacionamento quaternário

Fonte: Elaborada pelo autor

A quantidade de Entidades envolvidas em cada Relacionamento é determinada pela Cardinalidade do grau de Relacionamento, ou seja, podemos estabelecer a quantidade mínima e máxima de Entidades envolvidas com cada Entidade relacionada.

- A **Cardinalidade Mínima** determina a quantidade mínima de Entidades relacionadas por um número representativo, ou seja, 0 (zero) 1, 2, ..., N (muitos).
- A **Cardinalidade Máxima** determina a quantidade máxima de Entidades relacionadas por um número representativo, ou seja, 1, 2, ..., N (muitos).

2.2.6 Restrições de relacionamento

Os relacionamentos entre entidades possuem certas restrições que limitam as combinações possíveis das entidades que dele participam. Uma destas

restrições é a razão de **cardinalidade**. A quantidade de Entidades envolvidas em cada Relacionamento é determinada pela Cardinalidade do Tipo de Relacionamento, ou seja, pode-se estabelecer a quantidade mínima e máxima de Entidades envolvidas com cada Entidade relacionada.

Até agora vimos, de modo geral, como construir um diagrama de entidade relacionamento (DER) para projetar um banco de dados. Aprendemos os quatro principais construtores deste modelo: os conjuntos de entidades (CE), os conjuntos de relacionamentos (CR), os atributos de entidades e relacionamentos. Agora, veremos um dos mais importantes conceitos do MER: a restrição de razão de cardinalidade do relacionamento. O entendimento deste conceito será imprescindível para o processo de mapeamento do diagrama entidade relacionamento DER para o Modelo Relacional Mrel, ou seja, para transformar o projeto conceitual do banco de dados em um conjunto de tabelas.

Para o grau de Relacionamento Binário (que são mais comuns) podemos citar que as Cardinalidades são as seguintes:

a) UM-PARA-MUITOS (1 : N) ou alternativamente (N : 1): vamos ver um exemplo. No relacionamento binário TRABALHA, os conjuntos de entidades DEPARTAMENTO : EMPREGADO foram representados a seguir como tendo razão de cardinalidade 1:N (um para muitos) conforme DER a seguir.

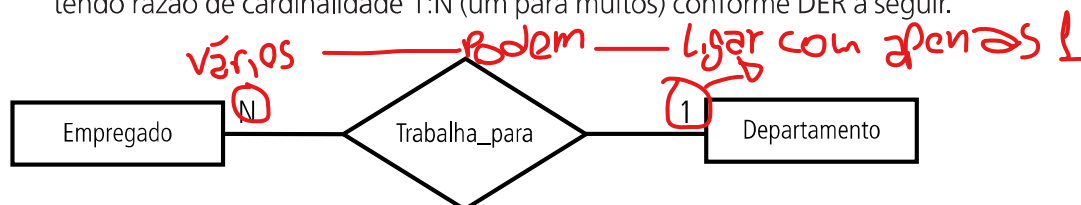


Figura 2.17: Exemplo de Diagrama ER Um-Para-Muitos

Fonte: Elaborada pelo autor

Já a Figura 2.18 apresenta o diagrama de ocorrências para a cardinalidade entre as entidades.

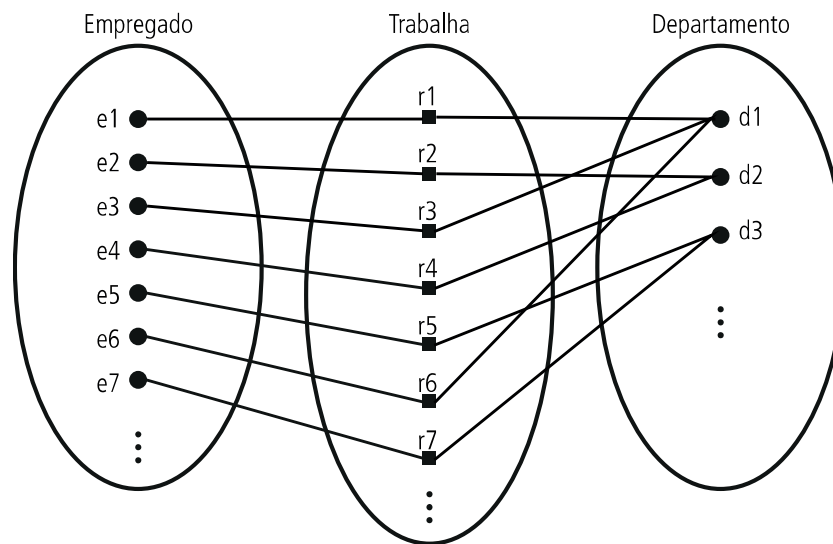


Figura 2.18: Diagrama de ocorrências para o relacionamento Um-Para-Muitos

Fonte: adaptado de ELMASRI & NAVATHE, 2005

Desta maneira, na empresa a qual este diagrama representa, cada entidade EMPREGADO pode estar relacionada a apenas um DEPARTAMENTO (um empregado pode trabalhar apenas para um departamento), mas cada entidade DEPARTAMENTO pode estar relacionada a inúmeras entidades EMPREGADO (um departamento pode conter vários empregados).

não repete dos dois lados
b) UM-PARA-UM (1 : 1): vejamos um exemplo de cardinalidade 1:1. No caso do relacionamento **GERENCIA**:

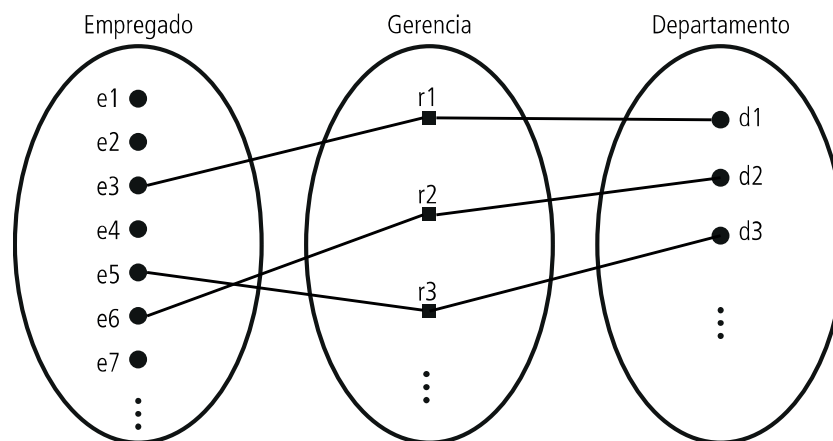


Figura 2.19: Exemplo de diagrama de ocorrências para o relacionamento Um-Para-Um

Fonte: ELMASRI & NAVATHE, 2005

Uma entidade DEPARTAMENTO está relacionada a apenas um EMPREGADO, que gerencia esse departamento. Este relacionamento é 1:1, pois cada empregado gerente pode gerenciar apenas um departamento e, dado um departamento, este departamento pode ter apenas um gerente. Podemos ver que, apenas uma seta sai de um empregado ligando-o a apenas um departamento, e vice-versa. A representação deste relacionamento no DER seria:

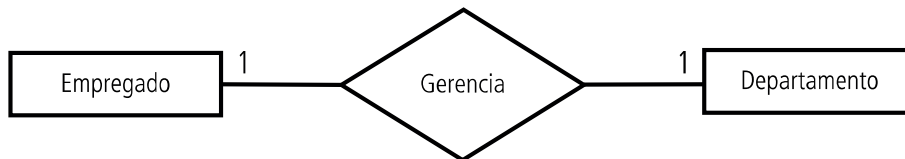


Figura 2.20: Exemplo diagrama ER para o relacionamento Um-Para-Um

Fonte: Elaborada pelo autor

É importante salientarmos que esta restrição de cardinalidade de UM-PARA-UM é definida pela realidade da empresa que se deseja representar no banco de dados. Se em outra empresa, for possível que um mesmo empregado gerencie diversos departamentos, a razão de cardinalidade 1:1 já não seria adequada a esta representação e sim 1 : N.

c) MUITOS-PARA-MUITOS (N : M): Para exemplificar a razão de cardinalidade N:M (muitos para muitos), tomemos o relacionamento TRABALHA-EM entre EMPREGADO e PROJETO (ELMASRI & NAVATHE, 2005). Vamos considerar que um empregado pode trabalhar em diversos projetos e que diversos empregados podem trabalhar em um projeto.

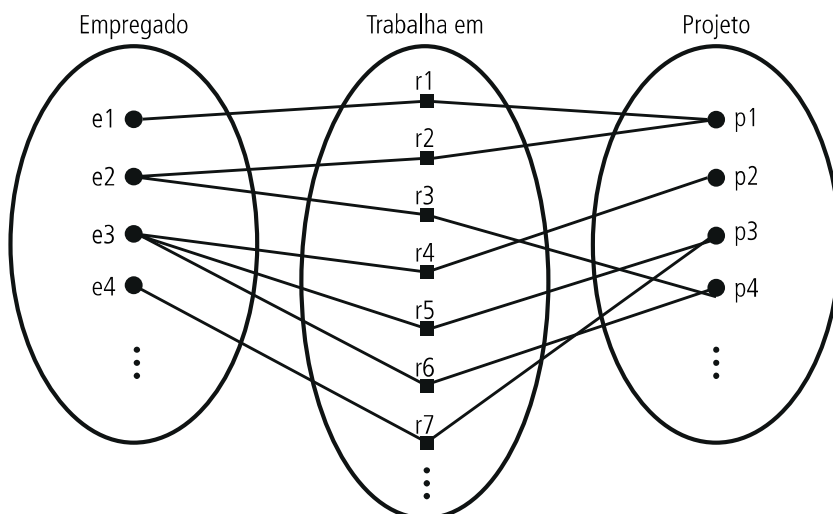


Figura 2.21: Exemplo de diagrama de ocorrência para o relacionamento binário Muitos-Para-Muitos.

Fonte: ELMASRI & NAVATHE, 2005

A partir da Figura 2.21 apresentada acima, podemos ver que o empregado e2 trabalha nos projetos p1 e p4; por sua vez, no projeto p4 trabalha além do empregado e2, também o empregado e3. O empregado e3 trabalha em 3 projetos; o projeto p2 conta com apenas 1 empregado. Vejamos abaixo como seria a representação deste relacionamento no DER:



Figura 2.22: Exemplo de diagrama ER binário Muitos-Para-Muitos

Fonte: Elaborada pelo autor



Para efetuar a modelagem utilizando o modelo entidade relacionamento, faça o download do software DIA no site http://dia-installer.de/index_en.html, nele podemos construir nossos diagramas e realizarmos a exportação para imagens.

Os relacionamentos de grau maior que dois (binários), ou seja, ternários, quartenários e acima, possuem também razão de cardinalidades. Mas, como este conceito não interferirá na criação de um banco de dados simples, não é objeto de estudo de nosso curso. Para saber mais a respeito, veja a bibliografia deste caderno.

d) AUTORELACIONAMENTO: Vimos até agora o conceito de razão de cardinalidade para relacionamentos binários. É bom lembrar que os relacionamentos unários também possuem este conceito. No relacionamento unário os elementos de uma entidade se relacionam a outros elementos dessa mesma entidade. Podemos verificar este tipo de relacionamento no exemplo abaixo:

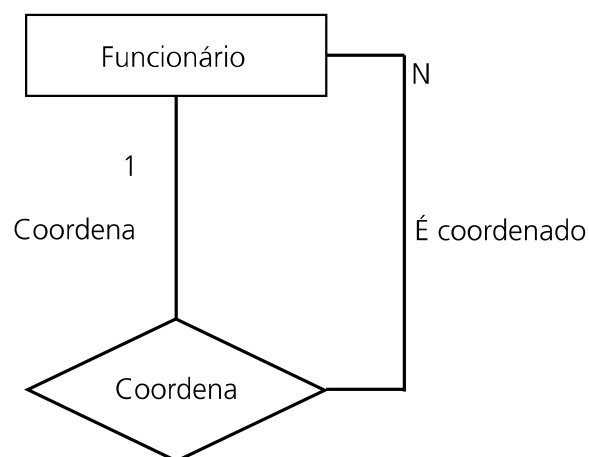


Figura 2.23: Exemplo de relacionamento unário

Fonte: Elaborada pelo autor

Com esta cardinalidade, um Coordenador (que é um funcionário) coordena vários funcionários e Funcionário é coordenado por um só Coordenador.

Resumo

Nesta aula vimos alguns conceitos importantes como os principais modelos de dados possíveis para implementação de um banco de dados. Apresentamos os conceitos de modelagem presentes em um modelo conceitual de alto nível, o modelo Entidade-Relacionamento (ER). Definimos então os conceitos básicos de entidades e atributos no modelo ER. Discutimos também de forma breve, conceitos do modelo ER como o conjunto de entidades e relacionamentos, atributos, atributos-chave, tipos de relacionamentos. Conhecemos as notações para representarmos os esquemas ER na forma de diagramas. Vimos por fim, como aplicar restrições estruturais (cardinalidade) em relacionamentos, especialmente no tipo binário.