

MODELAGEM CONCEITUAL DE DADOS

Projeto de banco de dados

É muito importante compreender como ocorre todo o processo de construção de um banco de dados, iniciando-se com o trabalho do analista de sistema em conjunto com os usuários finais e dando como resultado o primeiro modelo conceitual de dados, até as etapas seguintes, que gradativamente vão passando pelo projeto, até a implementação do banco de dados num SGBD.

Uma das principais características da tecnologia de banco de dados é a possibilidade de abstração dos dados, onde um modelo que esconde detalhes de como os dados estão fisicamente armazenados pode ser criado, facilitando assim a tarefa de programar aplicações para manipular esses dados.

Formalmente, podemos definir um modelo de dados como um conjunto de conceitos usados para descrever a estrutura do BD e certas restrições, que o sistema deve garantir (ELMASRI e NABATHE, 2018, p.11). Em outras palavras, uma notação gráfica é criada para representar dados do mundo real, de forma a desenhar ou projetar uma estrutura para o banco de dados. A estrutura do banco de dados é definida durante o trabalho de projeto do BD. Além das estruturas dos dados, nesse projeto também podem ser colocadas informações referentes a algumas restrições impostas ao BD, que são extraídas dos usuários finais, a partir do trabalho de análise do sistema.

Por exemplo, imagine um sistema de banco de dados que será desenvolvido para uma empresa que pretende armazenar os dados relativos aos seus funcionários e aos departamentos onde estes atuam. Após o trabalho de projeto, poderíamos chegar à definição da seguinte estrutura para os dados dos funcionários:

FUNIONARIOS	CPF_Func	Nome_Func	Endereco_Func	... mais algumas colunas ...	Codigo_Dep_Func



Para esse exemplo, poderíamos ter a seguinte restrição: todo funcionário dessa empresa obrigatoriamente tem que estar vinculado a um departamento. Ou seja, quando os dados dos funcionários forem inseridos no BD, o campo “Codigo_Dep_Func”, que determina qual é o departamento ao qual o funcionário está vinculado, não poderá estar em branco (esse é um campo obrigatório).

Tanto as estruturas dos dados quanto às restrições impostas a esses dados são definidas num modelo de dados. Existem diferentes categorias de modelos de dados. São elas:

- **Modelo Conceitual (resultado da etapa de projeto conceitual):** esse modelo é construído com base em conceitos tais como entidades ou objetos. É a primeira etapa do processo de modelagem de bancos de dados. Descreve a estrutura dos dados de maneira abstrata sem se preocupar em como será a sua futura implementação física.
- **Modelo Lógico (resultado da etapa de projeto lógico ou mapeamento do modelo de dados):** é um modelo gerado a partir do modelo conceitual, numa etapa intermediária do processo, quando o primeiro modelo é mapeado para o modelo físico (ou de implementação). Nessa etapa, um SGBD específico a ser utilizado já deve ser definido para determinar qual será a linguagem de programação utilizada na próxima etapa. Vale ressaltarmos que se houver algum tipo de erro cometido na etapa anterior de modelagem conceitual, o modelo lógico também terá esse erro refletido.
- **Modelo Físico (resultado da etapa de projeto físico):** esse modelo descreve os aspectos físicos de implementação. É a criação propriamente dita do banco de dados, utilizando uma linguagem específica de um SGBD.

A Figura ilustra de maneira simplificada as principais fases do projeto de um banco de dados. Nela, podemos observar todo o processo que se inicia com a definição do Mini Mundo, que é a parte do mundo real que será modelado, até o último passo que é a criação dos programas de aplicação que serão voltados aos usuários finais e irão utilizar os dados gerenciados pelo SGBD.

Ao longo do processo, cada etapa vai evoluindo para passos mais avançados, desde aquele mais próximo das pessoas, como o levantamento e a análise de requisitos feita pelos analistas, projetistas e usuários, até a construção dos modelos:



conceitual, lógico e físico gradativamente. Algumas dessas etapas iniciais são realizadas independentemente do modelo de implementação e do SGBD, e outras, as finais, dependem da escolha por um SGBD específico.

Na Figura, também surgem alguns conceitos, como o de **esquema** de um BD, que é a descrição da estrutura de um BD e pode ser textual, por exemplo, através da definição de comandos em uma linguagem de programação, ou gráfico, por exemplo, através do desenho de símbolos que representam tabelas ou conjuntos de dados.

Os modelos conceituais descrevem o esquema conceitual do BD usando, por exemplo, o modelo Entidade Relacionamento (ER). Os modelos lógicos ou de implementação descrevem os esquemas externos, que usam, por exemplo, o modelo Relacional. Nesses modelos, um SGBD já deve estar sendo definido para ser utilizado posteriormente, pois a depender dessa escolha, o projeto será diferente. Por exemplo, se o SGBD escolhido for o SQLServer da Microsoft, que é originalmente relacional, o projeto lógico do BD será criado baseando-se na estrutura de relações ou tabelas; mas se por outro lado o SGBD escolhido for o PostgreSQL, que tem como teoria a Orientação a Objetos, o projeto lógico do BD será criado baseando-se na estrutura de objetos, classes etc.

Outro conceito da área de modelagem em banco de dados é o de **instância** de um banco de dados, que se refere aos dados atuais armazenados num determinado momento, também chamado **estado** do BD. Ou seja, o estado do BD num determinado momento é como se fosse tirada uma foto dos dados contidos no BD naquele momento específico. Todos os dados que estão cadastrados num banco de dados num determinado instante, ou as suas instâncias, determinam o seu estado naquele momento.



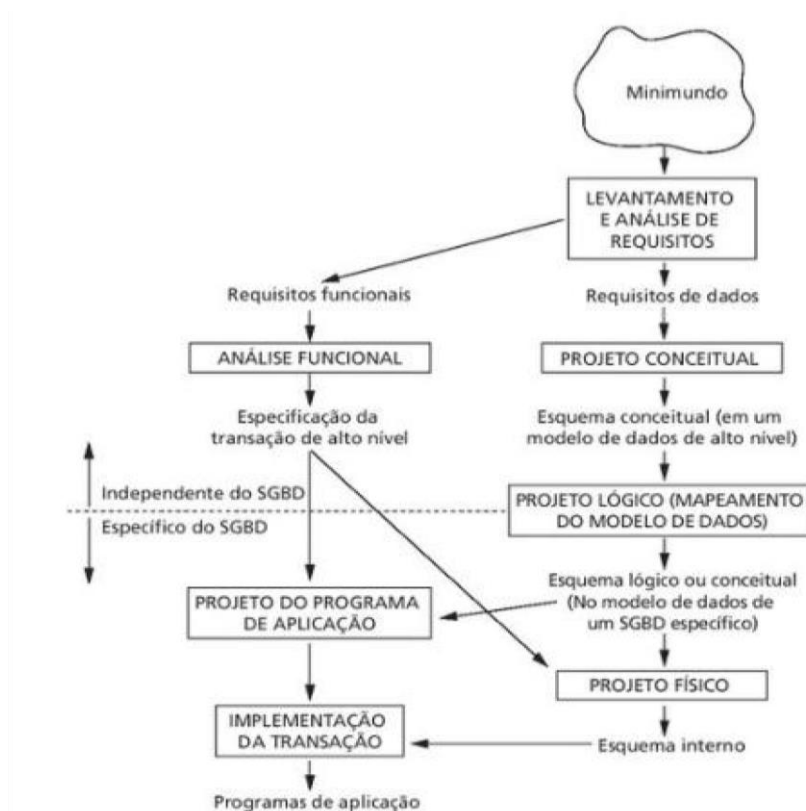


Figura: Um diagrama simplificado para ilustrar as principais fases do projeto de um banco de dados (ELMASRI e NAVATHE, 2018, p. 57).

Então, podemos refletir que o processo de construção de um banco de dados é gradativo, que se inicia com a intervenção dos analistas de sistemas e os usuários finais do sistema, que juntos definem os requisitos de dados. A partir dessa definição inicial, o primeiro modelo conceitual é construído e, posteriormente, é utilizado como base na construção dos modelos posteriores (projetos lógico e físico). Se ocorrer um erro de modelagem na etapa inicial desse processo, muito provavelmente todo o processo estará comprometido. Por esse motivo, considera-se a etapa inicial de modelagem conceitual de dados tão importante.

Vale também ressaltarmos a importância de que nesse processo nenhuma etapa seja ignorada, ou seja, não se deve pular nenhuma etapa, por mais experiente que seja a equipe de desenvolvimento de BD. Todas as etapas são importantes. Uma dessas etapas é a de mapeamento do modelo de dados, que traduz o modelo conceitual, num modelo lógico.



A Figura mostra um exemplo ilustrativo do mapeamento de modelos de dados: do modelo conceitual (ER) e seu equivalente no modelo lógico (Relacional).

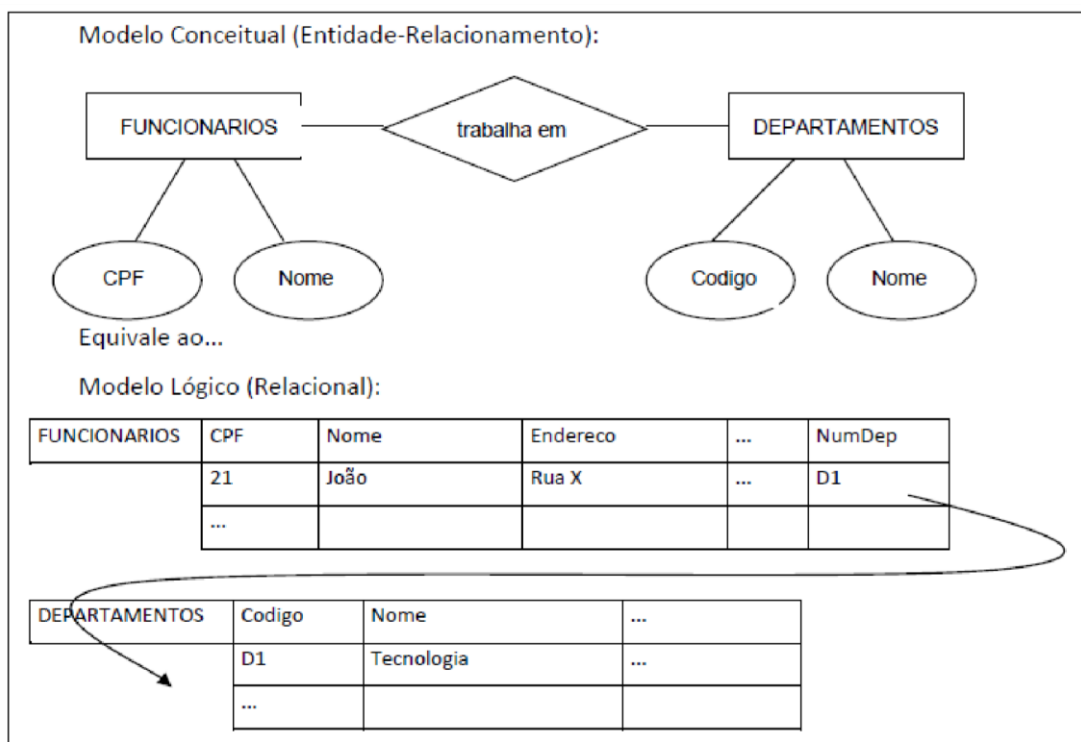


Figura: Exemplos de modelos conceitual e lógico.

Principais conceitos do modelo ER

Segundo Ramakrishnan e Gehrke, o modelo de dados idade- relacionamento é utilizado para descrever os dados envolvidos no cotidiano de uma empresa do mundo real em termos de objetos ou em idades e seus relacionamentos. Esse modelo é amplamente utilizado para desenvolver um projeto inicial de BD e fornece conceitos úteis que nos possibilitam mover de uma descrição informal do que os usuários desejam de seu BD, para uma descrição mais detalhada e precisa do que pode ser implementado num SGBD. Como vimos, o processo de construção de um BD é gradativo e a modelagem conceitual de dados é uma das etapas iniciais. Do resultado dessa modelagem depende o sucesso ou não de todo o restante do processo, fazendo com que a modelagem conceitual seja então uma das principais tarefas a serem desenvolvidas por pessoas dessa área.



A partir da modelagem, faz-se o projeto que é a base para a posterior implementação. Sem essa tarefa, é impossível o trabalho do administrador de BD. A seguir, serão apresentados conceitos sobre modelagem e projetos de BD, utilizando o modelo conceitual Entidade-Relacionamento (ER).

Tanto o projeto conceitual quanto posteriormente o projeto físico, para serem desenvolvidos, precisa de informações sobre as especificações das operações básicas feitas pelos usuários. Os projetistas e analistas de BD trabalham em conjunto com os usuários no levantamento dos requisitos de dados.

O modelo conceitual Entidade-Relacionamento (ER) é formado por desenhos gráficos, que possuem algum significado e representam elementos do mundo real a serem modelados. Os dois principais conceitos desse modelo, como podem ser observados por seu nome, são os conceitos de Entidade e de Relacionamento.

A Figura apresenta a principal simbologia utilizada no modelo ER e seu significado.

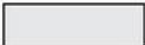




Símbolo	Significado
	Entidade
	Entidade fraca
	Relacionamento
	Relacionamento de identificação
	Atributo

Figura: Notação do modelo ER (parte 1)

Fonte: Adaptada de Elmasri e Navathe (2018, p. 76).

Entidade ou Tipo de entidade: é representado graficamente por um retângulo, é o conjunto formado por instâncias de dados, ou seja, é um conjunto de dados relacionados a um conceito único, que serão gravados no BD.



Entidade fraca é representada graficamente por um retângulo de traço duplo. É um tipo de entidade, mas com a característica de que os dados de suas instâncias são dependentes de dados de outra entidade, ou seja, os dados dessa entidade só existem se existirem dados numa outra entidade.

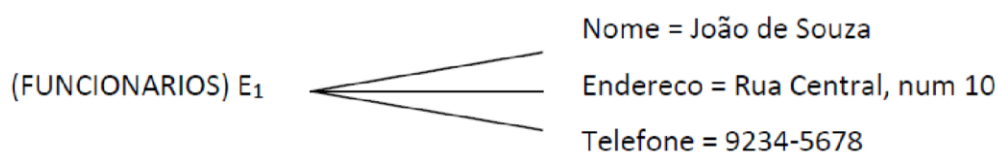
Relacionamento ou Tipo de relacionamento: é representado graficamente por um losango. Refere-se a ações que interagem com os elementos das entidades.

Relacionamento de identificação: representado graficamente por um losango de traço duplo, é um relacionamento que liga algum tipo de entidade fraca, que é identificada por outra entidade.

Atributos: representados graficamente por uma circunferência, são características comuns às instâncias que formam entidades.

Em outras palavras, as entidades são elementos ou objetos ou “coisas” do mundo real que possuem uma existência independente e são de interesse para uma determinada aplicação. As entidades são, por exemplo, os clientes de uma loja, os produtos que ela comercializa os funcionários que nela trabalham etc., ou seja, todos os elementos da loja no mundo real, que para a aplicação que será desenvolvida possuem algum tipo de importância e seus dados precisam ser cadastrados no banco de dados.

Os atributos são propriedades usadas para descrever cada uma das entidades. Por exemplo, na entidade dos Funcionários:



Os atributos podem ser dos seguintes tipos: **simples**, o que chamamos somente de atributos e geralmente são formados por um só conjunto de caracteres ou números; ou podem ser atributos **compostos**, que são formados por vários atributos simples, por exemplo: Endereço = rua + número + bairro + cidade + UF + CEP. Os atributos também podem ser monovalorados, que é o tipo mais comum, quando cada atributo possui uma instância, ou pode ser do tipo multivalorado, quando um atributo pode possuir várias instâncias, por exemplo, um departamento



de uma empresa localizado em diversas cidades é representado pelo uso de traço duplo no atributo.

A Figura apresenta de forma gráfica como esses diferentes tipos de atributos são representados no modelo ER.

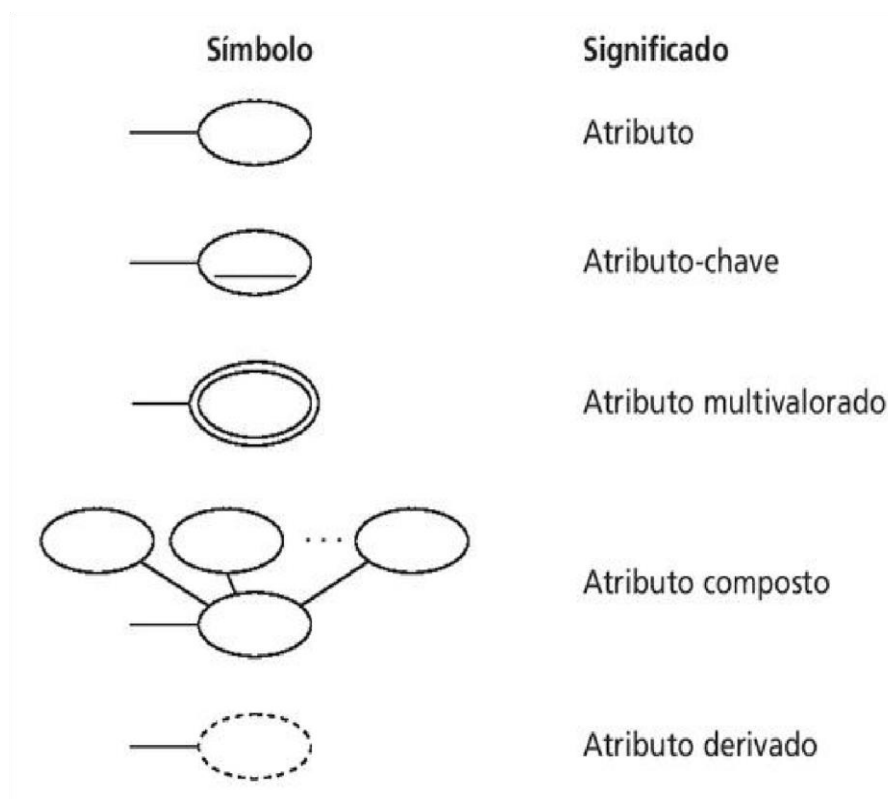


Figura: Notação do modelo ER (parte 2)

Fonte: adaptada de Elmasri e Navathe (2018, p. 76).

A Figura também apresenta o conceito de atributo derivado (representado com tracejado), que se refere aos atributos cujos valores não ficam armazenados no BD, mas sim esses dados são calculados durante a execução da aplicação, ou sempre que necessário. No caso do exemplo, seria a quantidade de Funcionários de um determinado departamento, que é uma conta que seria feita num momento.

Outro conceito é o de atributo-chave. As **chaves** são identificadores de um tipo de entidade, ou, em outras palavras, um atributo de um tipo de entidade que possui um valor único para cada entidade é chamado de chave. Por exemplo, o número de CPF de um funcionário, por ser um valor que não pode se repetir para mais de uma pessoa, quando dado, conseguimos identificar o funcionário. A chave é representada pelo nome do atributo sublinhado.



Uma chave pode ser um atributo simples, ou um atributo composto. Se for composto, temos o conceito de **chave composta**, que é formada pela combinação de vários atributos. Um tipo de entidade pode possuir uma chave simples, ou uma chave composta, ou várias chaves simples (distintas), ou várias chaves compostas. O conceito de chave foi introduzido em 1976 por Peter Chen, para identificar uma instância, porém a chave deve ser vista de uma forma mais ampla, como uma restrição de integridade. Para o conceito de chave, devemos entender que a instância é única. A restrição de integridade nesse caso é a garantia de que se um atributo é chave, não pode haver no BD mais de um atributo com o mesmo valor para aquele atributo. Por exemplo, se numa escola temos que a matrícula dos alunos é um valor único, este é chave. Se ocorrer no BD dois alunos com o mesmo número de matrícula, esse valor estaria repetido. Sendo assim, estaria ferindo a restrição de integridade do BD.

Além desses conceitos apresentados na Figura, há o conceito de **domínio** de um atributo, que se refere ao conjunto de valores que cada atributo de um tipo de entidade possui, ou seja, o domínio define os valores que podem ser assumidos por esse atributo, para cada entidade individualmente. Por exemplo, suponha que o atributo “nome” possua o domínio definido para um conjunto de caracteres alfabéticos. Isso significa que só poderão ser colocados no banco de dados nomes formados por no máximo 200 letras. Outro exemplo é o atributo CPF formado por um conjunto de caracteres numéricos de tamanho fixo 11. Nesse caso, só poderemos cadastrar CPFs com tamanho exato de 11 números.

Dois atributos podem ter o mesmo domínio, mais amplo, como por exemplo: na entidade FUNCIONARIOS, os atributos Telefone Residencial e Telefone Comercial podem possuir o mesmo domínio amplo dos números de telefone; e Data Nascimento e Data Admissão possuem o mesmo domínio amplo das datas.

Já estudamos aspectos e características referentes aos conceitos de entidades e de atributos. Façamos agora dos relacionamentos.

Relacionamentos ou tipos de relacionamentos são associações entre duas ou mais entidades distintas (ou instâncias da entidade), com um determinado significado. Graficamente representados por losangos ligando as entidades e seus nomes, geralmente são verbos, já que significam uma ação entre elementos de uma entidade e seu relacionamento com elementos da outra entidade.



Por exemplo: FUNCIONARIO João **TRABALHA_EM** DEPARTAMENTO
Pesquisa

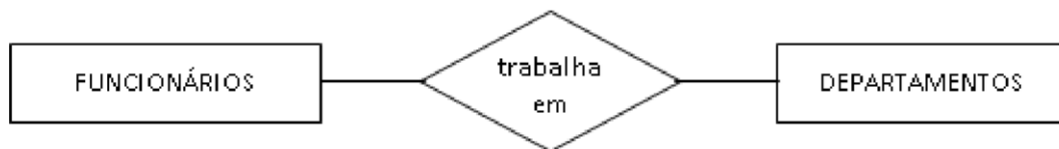


Figura: Exemplo simplificado de relacionamento.

As entidades participantes de um relacionamento atuam com um determinado **papel** nesse relacionamento. O significado desse papel é dado por um nome a ele atribuído. Um relacionamento é dito **recursivo**, também conhecido como autor relacionamento, quando este relaciona uma única entidade, através de dois papéis distintos. Por exemplo:

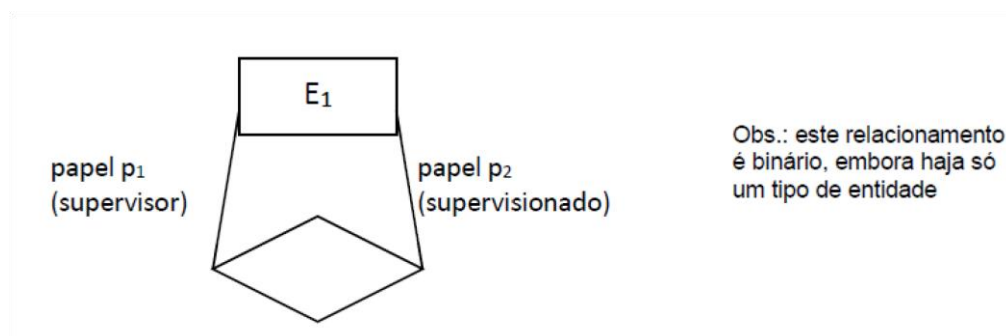


Figura: Exemplo de autor relacionamento ou relacionamento recursivo.

Além das estruturas básicas já explicadas do modelo ER, existem algumas restrições impostas sobre tipos de relacionamento. As restrições mais importantes são a relação de **cardinalidade** e a relação de **participação**. Juntas, as restrições de cardinalidade e de participação formam as **restrições estruturais** do modelo ER, que determinam a quantidade mínima e máxima de elementos das entidades que participam do relacionamento. A relação de **cardinalidade** especifica o número de instâncias de um tipo de entidade que pode participar do relacionamento. Um



relacionamento pode ser: 1 para 1 (1:1), 1 para N (1:N), N para 1 (N:1) ou N para N (N:N).

A Figura apresenta a forma gráfica de representação dos conceitos de participação, cardinalidade e restrição estrutural.

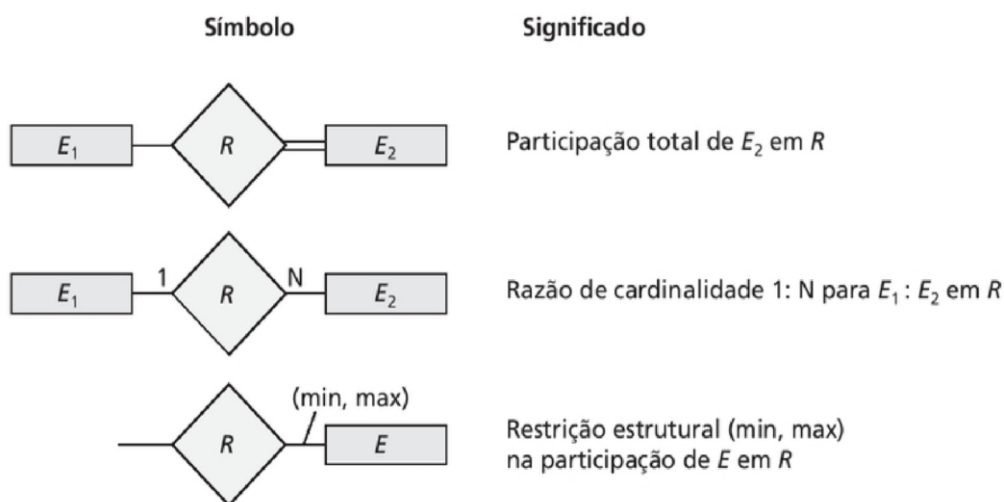


Figura: Notação do modelo ER (parte 3)

Fonte: adaptada de Elmasri e Navathe (2018, p. 76).

No relacionamento **1:1** apenas um elemento da entidade pode se relacionar com apenas 1 elemento da outra entidade envolvida no relacionamento. Por exemplo, supondo que exista um relacionamento chamado “chefia” entre a entidade FUNCIONARIOS e a entidade DEPARTAMENTOS, onde apenas um funcionário pode gerenciar um único departamento e um departamento pode ter apenas um único chefe. O relacionamento “chefia” é do tipo 1:1.

No relacionamento **1: N** ou **N:1**, ou um para muitos, um elemento da entidade pode se relacionar com vários elementos da outra entidade envolvida no relacionamento, por exemplo, supondo que exista um relacionamento chamado “trabalha_em” entre a entidade FUNCIONARIOS e a entidade DEPARTAMENTOS, onde um funcionário pode trabalhar em apenas 1 departamento, porém um departamento possui vários (N) funcionários trabalhando nele. O relacionamento “trabalha_em” é do tipo: N.

No relacionamento **N: N**, ou muitos para muitos, vários elementos da entidade podem se relacionar com vários elementos da outra entidade envolvida no relacionamento. Por exemplo, supondo que exista um relacionamento chamado



“realizam” entre a entidade FUNCIONÁRIOS e a entidade PROJETOS, onde um funcionário pode realizar vários (N) projetos e cada projeto pode ser realizado por vários (N) funcionários. O relacionamento “realiza” é do tipo N: N.

A relação de **participação** especifica se a existência das instâncias nas entidades depende de elas estarem associadas ou não à outra entidade. Pode ser chamada de **obrigatória** (total) ou **opcional** (parcial).

Quando todos os elementos da entidade são obrigados a participarem de um relacionamento, este é chamado de **total** ou **obrigatório**. Por exemplo, se dizemos que obrigatoriamente todos os funcionários da empresa têm que estar relacionados trabalhando em um departamento, nesse caso não pode haver sequer um funcionário cadastrado no banco de dados, sem a informação de qual departamento este trabalha.

Por outro lado, se uma entidade pode ter elementos que não participam de determinado relacionamento, este é dito **opcional** ou **parcial**. Por exemplo, se dizemos que nem todos os funcionários são obrigados a estarem envolvidos num projeto, nesse caso teremos funcionários que realizam projetos e outros que não realizam então esse relacionamento é parcial.

Agora que os principais conceitos do modelo ER foram apresentados, é hora de colocar a mão na massa e ver como esse modelo é utilizado na prática.

