

Primeiro vamos começar com a definição de dado. Segundo a biblioteca Wikipédia [1] dado é um valor discreto que pode representar quantidades, qualidades, fatos, estatísticas e qualquer outra unidade básica que posteriormente podem ser interpretadas. Já o Banco de Dados é um repositório ou recipiente para uma coleção de arquivos de dados computadorizados onde os usuários podem realizar operações sobre esses dados, como a edição e visualização de dados [2,3].

Elmarsri e Navathe [3] afirmam que um banco de dados possui algumas vantagens:

- Natureza autodescritiva dos dados;
- Isolamento entre dados e programa;
- Suporte para múltiplas visualizações dos dados;
- Acesso concorrente aos dados e recuperação de falhas.

Elmarsri e Navathe [3] afirmam que há situações em que não é necessário o uso de um banco de dados, são elas:

- Investimentos iniciais altos em hardware, software e treinamento;
- O banco de dados e suas aplicações são simples, bem definidas e sem previsão de mudanças;
- Acesso de múltiplos usuários aos dados não é necessário.

Existem diversas maneiras de descrever um banco de dados e estes níveis podem variar em função do nível de abstração que se deseja enxergar. Provavelmente um modelo de dados para um usuário não conterá informações específicas sobre o armazenamento destes dados em meios físicos [4]. Em termos gerais vamos descrever dois modelos de BD aqui neste material. São eles:

Modelo conceitual

Registra quais dados podem aparecer no BD, porém não registra como estes dados estão armazenados a nível de SGBD. A forma mais difundida deste tipo de modelagem é o sistema Entidade - Relacionamento (ER).

Modelo lógico

Um modelo lógico é uma descrição de um banco de dados no nível de abstração visto pelo usuário do SGBD. Assim, o modelo lógico é dependente do tipo particular de SGBD que está sendo usado. Por exemplo em um SGBD relacional esse formato é o de tabelas.

A Figura 01 apresenta o modelo conceitual de representação de um banco de dados. Já a Figura 02 apresenta um modelo lógico.

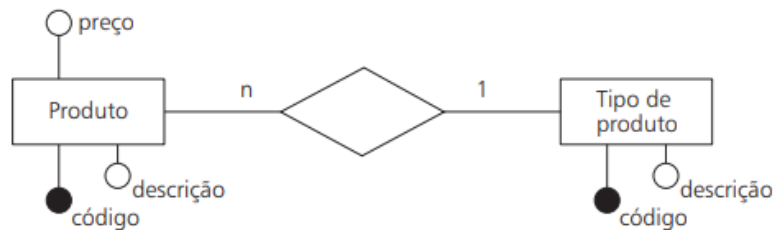


Figura 01. Exemplo do modelo relacional [4].

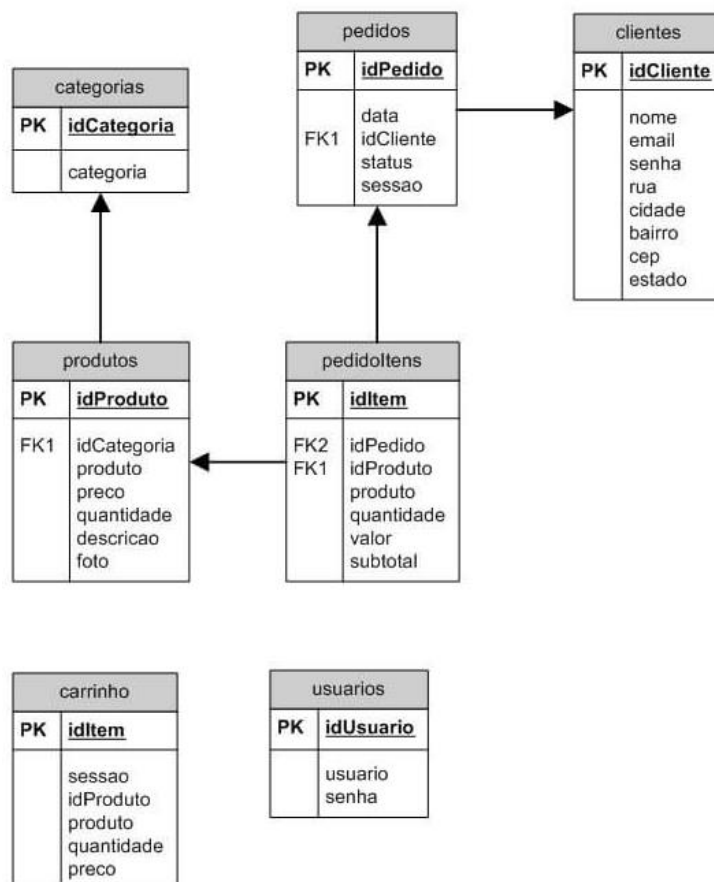


Figura 02. Exemplo do modelo lógico relacional [5].

Após o desenvolvimento destes modelos o projeto físico ou modelo físico do BD pode ser produzido. Nesta etapa o modelo do banco de dados é enriquecido com detalhes que influenciam no desempenho do banco de dados, mas não interferem em sua funcionalidade. Nesta etapa o projeto já entra na fase final do seu desenvolvimento [4].

O modelo conceitual Entidade-Relacionamento (ER)

A abordagem **ER** foi criada por Peter Chen em 1976. A Figura 03 apresenta um modelo de Entidade - Relacionamento (ER).

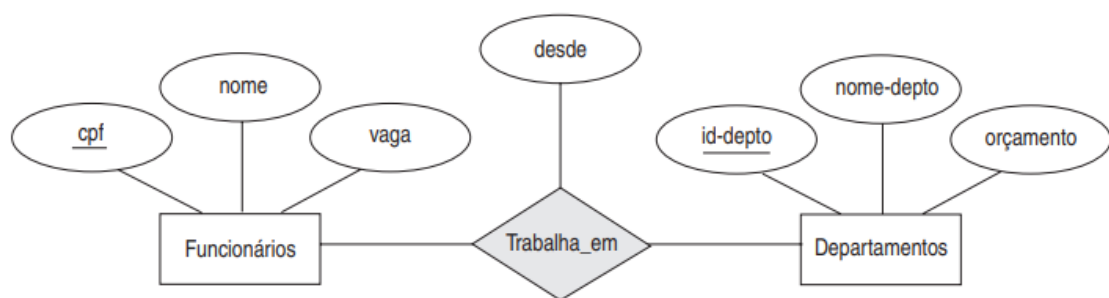


Figura 03. O conjunto de relacionamentos Trabalha_em [6].

O conceito ER é formado por:

Entidade

Como dito anteriormente a entidade é o objeto ao qual deseja-se representar, normalmente a entidade dará origem a uma tabela no banco de dados. A entidade é representada por um retângulo no modelo conceitual. Um exemplo de entidade seria o **carro**.

Relacionamento

Além da descrição dos objetos que formam o banco de dados é necessário estabelecer a regra de associação entre estes objetos e para isso dá-se o nome de relacionamento. A regra de associação é representada por meio de um losango conforme modelo da Figura 03. Estas regras de associação entre entidades podem ser binárias (grau dois), ternárias (grau três) ou n-árias (grau n)

Em relação ao modelo relacional pode-se dizer que as entidades são efetivamente construídas através de seus atributos. Por exemplo atributos de um objeto carro poderiam ser: (a) N° de chassi; (b) Cor, (c) Motor e quaisquer outros elementos que auxiliem ou

sejam necessários na descrição de um carro para um determinado modelo de negócio. Os atributos podem ser representados por meio de círculo ou elipses no modelo conceitual.

Os atributos podem ser compostos ou atômicos. Atributos compostos podem ser divididos em subpartes menores, que representam atributos mais básicos, com significados independentes. Por exemplo, o atributo Logradouro da entidade FUNCIONARIO pode ser subdividido em Logradouro, Cidade, Estado e Cep, com os valores “Rua das Flores, 751”, “São Paulo”, “SP” e “07700110”. Os atributos não divisíveis são chamados atributos simples ou atômicos [3].

Atributos de valor único *versus* multivalorados. A maioria dos atributos possui um valor único para uma entidade em particular; tais atributos são chamados de valor único. Por exemplo, Idade é um atributo de valor único de uma pessoa. Em alguns casos, um atributo pode ter um conjunto de valores para a mesma entidade — por exemplo, um atributo Cores para um carro, ou um atributo Formacao_academica para uma pessoa. Os carros com uma cor têm um único valor, enquanto os carros com duas cores possuem dois valores de cor. De modo semelhante, uma pessoa pode não ter formação acadêmica, outra pessoa pode ter, e uma terceira pode ter duas ou mais formações; portanto, diferentes pessoas podem ter distintos números de valores para o atributo Formacao_academica. Esses atributos são chamados de multivalorados [3].

Atributos armazenados *versus* derivados. Em alguns casos, dois (ou mais) valores de atributo estão relacionados — por exemplo, os atributos Idade e Data_nascimento de uma pessoa. Para uma entidade de pessoa em particular, o valor de Idade pode ser determinado pela data atual (hoje) e o valor da Data_nascimento dessa pessoa. O atributo Idade, portanto, é chamado de atributo derivado e considerado derivável do atributo Data_nascimento, que é chamado, por sua vez, de atributo armazenado [3].

Além disso é válido comentar que alguns atributos podem ter um caráter identificador especial, sendo que estes distinguem a ocorrência da entidade no BD. Há este atributo é dado o nome de chave. Pode haver mais de uma chave candidata; se houver, designamos uma delas como a chave primária. Normalmente a chave primaria é demarcada de forma especial no diagrama ER.

Em relação aos relacionamentos costumamos nos referir a cardinalidade destes relacionamentos. Estes podem ser classificados de 3 formas: 1:1, 1:n e m:n. Vamos treinar algumas leituras de cardinalidades em função das figuras.



Figura 04 – Relacionamento de cardinalidade 1:n (Um para muitos) [4].

Nesta Figura 04 a leitura ideal desta relação seria: “n” alunos inscrevem-se e “um” curso.



Figura 05 - Relacionamento de cardinalidade n:n (muitos para muitos) [4].

Nesta Figura 05 a leitura ideal desta relação seria: “n” engenheiros alocam-se e “n” projetos.

O modelo ER pode ser estendido e isso nos ajudará a representar com maior grau de precisão situações mais complexas. Na especialização um conjunto de subclasses pode ser definida a partir de uma entidade. Essa especialização pode ser definida em função da superclasse. Vejamos o exemplo da entidade FUNCIONARIO.

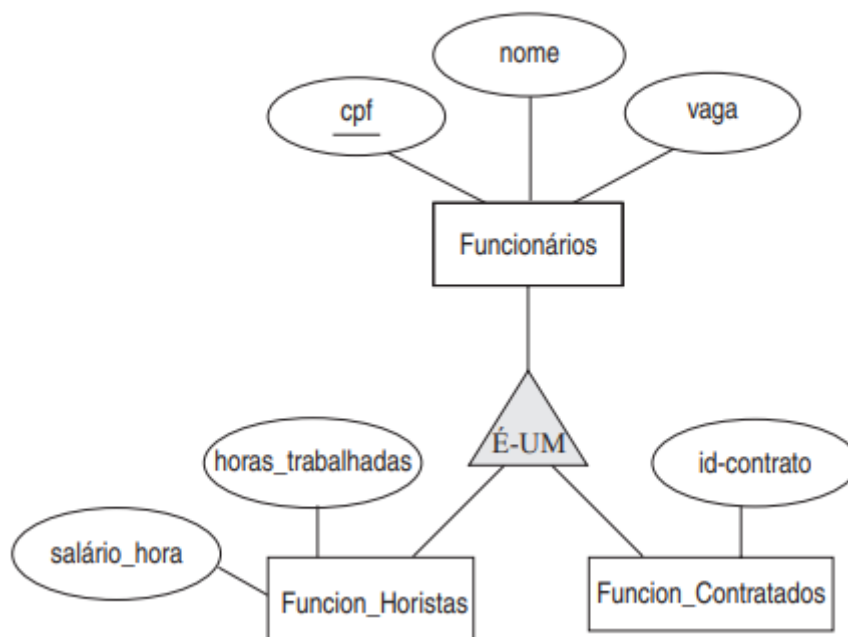


Figura 06. Hierarquia de classe [6].

Neste sentido a entidade FUNCIONARIO (superclasse) compartilha informações com as entidades criadas a partir da especialização (subclasses). Logo todo FUNC_HORISTA herda os atributos cpf, nome e vaga e na sua especialização possui atributos SALARIO_HORA e HORAS_TRABALHADAS.

Além disso o processo inverso pode ser realizado que é chamado de generalização que tenta suprimir as diferenças entre diversos tipos de entidades. Podemos ver esse processo sendo realizado na Figura 07. A letra d na criação da especialização indica uma restrição de disjunção (lembra o operador OU em lógica de programação), ou seja, um veículo pode ser CARRO OU CAMINHAO.

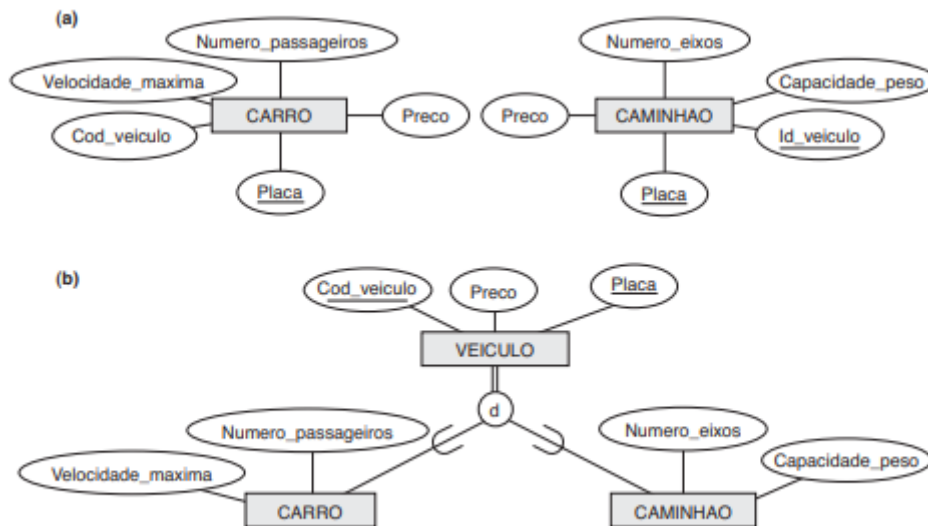


Figura 07. Generalização. (a) Dois tipos de entidade, CARRO e CAMINHAO. (b) Generalizando CARRO e CAMINHAO na superclasse VEICULO [3].

Essa letra indica o vínculo ou restrição entre a superclasse e subclasse. No caso outras modalidades podem ser aplicadas como por exemplo a restrição o que indica a sobreposição (*overlapping* em inglês). Nesta condição uma entidade superclasse pode ser membro de mais do que uma subclasse.

O modelo relacional

O modelo relacional se aproxima do modelo ER porém já está em um estado mais próximo ao modelo real do banco de dados. Então nesta seção conceitos do modelo conceitual aparecem, porém com uma representação diferente. A Figura 09 apresenta um modelo relacional.

<i>id-aluno</i>	<i>nome</i>	<i>login</i>	<i>idade</i>	<i>média</i>
50000	Dave	dave@cs	19	3,3
53666	Jones	jones@cs	18	3,4
53688	Smith	smith@ee	18	3,2
53650	Smith	smith@math	19	3,8
53831	Madayan	madayan@music	11	1,8
53832	Guldu	guldu@music	12	2,0

Figura 09. Uma instância A1 da relação Alunos [6].

O esquema poderá ser representado pelo seguinte bloco:

Alunos (id-aluno: **string**, nome: **string**, login: **string**, idade: **integer**, média: **real**)

Logo a relação FUNCIONARIO possui grau 5.

Outro conceito importante do modelo relacional são as chaves. Estas auxiliam na relação entre as linhas do banco é a chave. Há três tipos de chaves, são elas:

- a) Chave primária;
- b) Chave alternativa;
- c) Chave estrangeira.

Uma chave primária é uma coluna ou uma combinação de colunas cujos valores distinguem uma linha das demais dentro de uma tabela. Por exemplo na Figura 09 a coluna id-aluno é a chave primária.

Uma chave estrangeira é uma coluna ou uma combinação de colunas, cujos valores aparecem necessariamente na chave primária de uma tabela. A chave estrangeira é o mecanismo que permite a implementação de relacionamentos em um banco de dados relacional. A Figura 10 apresenta um exemplo de chave estrangeira.

Dept				
CodigoDepto		NomeDepto		
D1		Compras		
D2		Engenharia		
D3		Vendas		

Emp				
CodEmp	Nome	CodigoDepto	CategFuncional	CPF
E1	Souza	D1	—	132.121.331-20
E2	Santos	D2	C5	891.221.111-11
E3	Silva	D2	C5	341.511.775-45
E5	Soares	D1	C2	631.692.754-88

Figura 10. Chave estrangeira [4].

No banco de dados da Figura 4.3, a coluna CodigoDepto da tabela Emp é uma chave estrangeira em relação a chave primária da tabela Dept. Isto significa que, na tabela Emp, os valores do campo CodigoDepto de todas as linhas devem aparecer na coluna de mesmo nome da tabela Emp. A interpretação desta restrição é que todo empregado deve estar associado a um departamento.

A chave alternativa é aplicada em alguns casos, quando mais de uma coluna ou combinações de colunas podem servir para distinguir uma linha das demais. Uma das colunas (ou combinação de colunas) é escolhida como chave primária. As demais colunas ou combinações são denominadas chaves alternativas.

Normalização é o processo de modelar o banco de dados projetando a forma como as informações serão armazenadas a fim de eliminar, ou pelo menos minimizar, a redundância no banco. Tal procedimento é feito a partir da identificação de uma anomalia em uma relação, decompondo-as em relações melhor estruturadas [7].

Na primeira forma normal todos os valores das colunas são atômicos e as colunas não tem atributos repetidos.

Referências

- [1] Wikipédia. Dados. Wikipédia - A Enciclopédia Livre 2022. <https://pt.wikipedia.org/wiki/Dados>.
- [2] Date CJ. Introdução a Sistemas de Bancos de Dados. 1ª edição. GEN LTC; 2004.

- [3] Elmasri R, Navathe SB. Sistemas de Banco de Dados. 1ª edição. Pearson Universidades; 2019.
- [4] Heuser CA. Projeto de Banco de Dados. 6ª ed. Bookman; 2008.
- [5] Utilidade Pública: Tecnologia, Educação e Cidadania. Modelagem de dados: modelo conceitual, modelo lógico e físico. Blog n.d. <https://www.luis.blog.br/modelagem-de-dados-modelo-conceitual-modelo-logico-e-fisico.html>.
- [6] Ramakrishnan R, Gehrke J, Taniwake C, Tortello JEN, Sousa EPM de. Sistemas de Gerenciamento de Bancos de Dados. 3ª edição. AMGH; 2007.
- [7] Machado D. Normalização em Bancos de Dados. Medium 2015. <https://medium.com/@diegobmachado/normalização-em-banco-de-dados-5647cdf84a12>.