

A faint, stylized line drawing of a person with glasses holding a large document. The background features a large semi-circle and several diamond shapes.

Projeto de banco de dados: modelagem conceitual

Seja bem-vindo ao mundo do projeto de banco de dados! Você conhecerá suas etapas e aprenderá a focar particularmente na modelagem conceitual. Explorará os fundamentos essenciais para criar representações visuais claras e abstratas de sistemas de informação, aprendendo a desenvolver modelos conceituais robustos que servem como base para a criação de bancos de dados eficientes.

Profa. Nathielly de Souza Campos | Prof. Sidney Nicolau Venturi Filho

Propósito

É recomendável que você reproduza os exemplos práticos usando uma ferramenta para modelagem de dados como a brModelo, que pode ser baixada gratuitamente para essa tarefa.

Objetivos

- Identificar as etapas de um projeto de banco de dados.
- Reconhecer os elementos do diagrama de entidade e relacionamento.
- Compreender a modelagem de entidades e relacionamentos.
- Compreender a modelagem de atributos.

Introdução

Queremos construir um banco de dados. Por onde começamos?

Em primeiro lugar, é necessário esclarecer que, ao construirmos um banco de dados, estamos automatizando algum tipo de negócio, ou mesmo parte dele. Segundo Elmasri e Navathe (2019), um banco de dados representa algum aspecto do mundo real, às vezes chamado de minimundo ou de universo de discurso. É fundamental conhecermos como o negócio funciona.

Veremos que a construção de um banco de dados é uma atividade dividida em fases bem definidas. Ao longo delas, costumamos usar modelos de dados, que servem para que o usuário tenha facilidade para entender a organização da estrutura do banco de dados sendo construído. Perceberemos que isso ocorre porque o modelo não possui informações muito detalhadas a respeito da representação física dos dados.

A etapa de projeto conceitual servirá para a construção de diagrama de entidade e relacionamento (DER), em que há dois conceitos essenciais: entidades e relacionamentos. Trabalharemos alterações em um DER com objetivo de comportar novos requisitos de dados e perceberemos que a construção desse diagrama é um processo incremental e sempre sujeito a revisões.

Por último, voltaremos nossa atenção para a modelagem de atributos para fecharmos o ciclo de aprendizagem.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Levantamento de requisitos e projeto conceitual

Abordaremos duas fases cruciais do desenvolvimento de sistemas, começando pelo essencial: levantamento de requisitos. Além disso, exploraremos o projeto conceitual de bancos de dados.

Convidamos você a participar ativamente, desvendando os intrincados processos do levantamento de requisitos e do projeto conceitual, essenciais para a construção de sistemas de informação robustos.

Vamos começar! Assista ao vídeo e compreenda o levantamento de requisitos, fundamental para a construção de sistemas eficientes, e o projeto conceitual de bancos de dados, em que traduziremos esses requisitos em estruturas conceituais sólidas.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Projeto de banco de dados

De maneira simplificada, projetar um banco de dados envolve as seguintes fases: levantamento de requisitos, projeto conceitual, projeto lógico e projeto físico.

Ao construir um banco de dados para uma corporação, devemos ter em mente que há colaboradores desempenhando diversas tarefas associadas ao negócio em questão. Assim, temos que adquirir conhecimento sobre o funcionamento das rotinas e tarefas para capturarmos as necessidades associadas à gestão de dados. Veremos que esse conhecimento ocorre na fase de levantamento de requisitos.

Levantamento de requisitos

Ao longo da etapa de **levantamento de requisitos**, o profissional de dados entrevista usuários para entender sobre o funcionamento do negócio e documentar os requisitos de dados de maneira completa e detalhada.

Imagine que você foi convidado a participar de um projeto que objetiva construir um banco de dados para controlar inscrições de alunos em uma escola de treinamentos na área de Tecnologia de Informação.

Após realizar entrevistas junto aos colaboradores, você identifica os seguintes requisitos de dados da escola:

- A escola planeja diversos cursos. Cada um deles possui nome, descrição, carga horária e é identificado por um código único.
- A escola armazena o nome, a data de nascimento, o CPF, o e-mail e um telefone de cada cliente, que é identificado por um código único.
- Quando um cliente faz inscrição em determinado curso, é necessário armazenar a data. Caso seja cancelada a inscrição, é preciso saber quando ocorreu esse evento. Um cliente pode fazer diversos cursos.

Projeto conceitual

Com os requisitos de dados em mãos, usaremos um modelo de dados gráfico para formalizar entendimento mais preciso a respeito dos requisitos de dados. Essa atividade ocorrerá na próxima fase do projeto de banco de dados: **projeto conceitual**.

O projeto ou esquema conceitual envolve construir um modelo de dados de alto nível a partir dos requisitos de dados que contêm os principais objetos e seus relacionamentos, mapeados na etapa de levantamento de requisitos. Nesta etapa, não há preocupação em saber detalhes sobre como os dados devem ser armazenados.

O projeto conceitual usa um diagrama gráfico, conhecido por **Diagrama de Entidade e Relacionamento (DER)**, que possui três elementos essenciais:

Entidades

É representada por um retângulo com o seu nome.

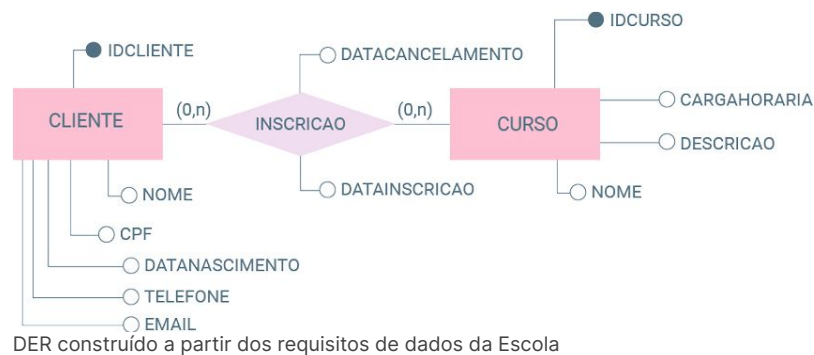
Relacionamentos

É representada por um losango ligado por linhas aos retângulos das entidades participantes do relacionamento.

Atributos

São expressos graficamente ligados à entidade ou ao relacionamento ao qual fazem parte.

A seguinte imagem representa um DER construído a partir dos requisitos de dados obtidos na etapa de levantamento de requisitos. Confira!



A partir do diagrama, é possível concluir que o modelo possui duas entidades (CLIENTE e CURSO) cuja função é armazenar os dados dos clientes e dos cursos da escola. Além disso, essas entidades possuem uma relação entre si, de maneira que um cliente pode fazer inscrição em um ou mais cursos. De acordo com Elmasri e Navathe:

“

O esquema conceitual de alto nível (DER) pode ser utilizado como uma referência para garantir que todos os requisitos de dados dos usuários sejam atendidos e que não estejam em conflito.

—
(ELMASRI; NAVATHE, 2019, n. p)

Isso acontece porque a representação dos requisitos de dados a partir do DER permite um aprendizado mais preciso a respeito do funcionamento do negócio sendo modelado, quando comparado aos requisitos de dados.

Outra notação para DER: diagrama de classes UML

Ao longo da sua atuação profissional, você perceberá que não há uma notação-padrão para representação dos conceitos do modelo de entidade e relacionamento. Normalmente, a notação depende de preferência dos profissionais ou mesmo de regras estabelecidas pela empresa de desenvolvimento. As ferramentas **CASE** fazem uso de várias notações. Por exemplo, a utilizada na ferramenta brModelo é muito próxima da notação original para modelos de entidade e relacionamento.

CASE

Do inglês Computer Aided Software Engineering – engenharia de software auxiliada por computador.

Em projetos de software, é comum o uso da **UML** para visualização e documentação dos seus componentes. De certa forma, um diagrama de classes da UML pode ser considerado uma notação alternativa para representar os conceitos de um DER. No diagrama de classes UML, cada classe é representada por uma caixa que possui três seções:

UML

Do inglês, Unified Modeling Language – Linguagem de Modelagem Unificada.

1

Superior

Exibe o nome da classe.

2

Central

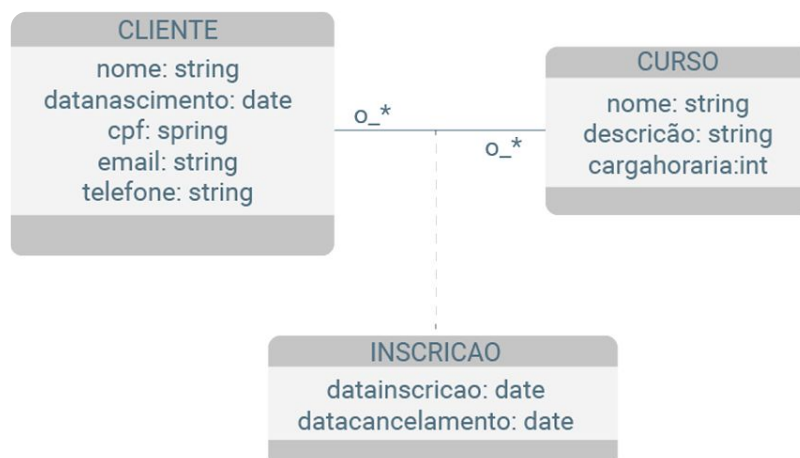
Exibe os atributos. Além disso, o desenvolvedor pode, se desejar, adicionar informações sobre o tipo de dados de algum atributo, colocando um sinal de dois pontos ":" e em seguida o nome do tipo de dados.

3

Inferior

Inclui as operações associadas aos objetos da classe, a serem designadas numa etapa posterior, quando do projeto das aplicações do banco de dados.

Na terminologia da UML, o relacionamento entre classes é chamado de **associação**. Assim, uma associação é representada por uma linha que conecta as classes participantes. Além disso, atributos dos relacionamentos são colocados em uma caixa conectada à associação por uma linha tracejada. A próxima imagem mostra como o DER construído a partir dos requisitos de dados da escola pode ser exibido sob a forma de diagrama de classes UML.



Esquema conceitual Escola na notação do diagrama de classes UML

Após as etapas de levantamento de requisitos e criação do DER, estamos quase prontos para conhecermos a construção do banco de dados propriamente dito. Construiremos um modelo de dados de mais baixo nível, que vai depender da escolha do SGBD. Faremos isso na próxima fase: o **projeto lógico**.

Atividade 1

O projeto de banco de dados é dividido em várias etapas. Uma delas envolve a construção de um modelo que descreve a estrutura do banco de dados, com seus objetos, atributos e relacionamentos inerentes ao funcionamento do negócio alvo da modelagem, independentemente de limitações tecnológicas, implementação ou até mesmo dispositivos de armazenamento. Esse modelo é denominado:

A

Conceitual

B

Lógico

C

Físico

D

Prescritivo

E

Descritivo



A alternativa A está correta.

O modelo conceitual, construído na etapa de projeto conceitual, tem foco na criação de uma visão abstrata do banco de dados, facilitando o entendimento do modelo por parte do usuário final. Essa visão oculta detalhes de implementação, por exemplo, a respeito de como os dados serão armazenados.

Projeto lógico

Assista ao vídeo e aprenda a traduzir o projeto conceitual em estruturas de dados tangíveis, utilizando técnicas específicas para representar de maneira eficaz as relações e entidades identificadas.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

O projeto lógico, também conhecido por **modelo de dados de baixo nível**, objetiva transformar o modelo conceitual em um modelo lógico, que depende do tipo de SGBD escolhido. Existem diversos modelos lógicos, por exemplo:

- Rede
- Hierárquico
- Relacional
- Orientado a objeto
- Grafos
- Chave-valor
- XML

No entanto, atualmente, o mais popular é o relacional. Como exemplos de SGBD que fazem uso do modelo relacional, podemos citar: Oracle, MySQL, PostgreSQL, SQLite e Sql Server.



Saiba mais

É importante acompanharmos as tendências de mercado sobre o uso de tecnologias de banco de dados. No Explore +, há indicação para pesquisar sobre o portal que tem um ranking com atualização mensal sobre o uso de SGBDs. Confira.

O modelo relacional de banco de dados surgiu na década de 1970 e representa os dados em estruturas chamadas tabelas. Cada tabela possui nome e coluna(s) que compõe(m) a sua estrutura. Nossa tarefa é converter o modelo conceitual para o lógico relacional. Para isso, utilizaremos regras bem definidas, que dependem dos elementos do DER.

No dia a dia, a conversão DER para o modelo lógico relacional é realizada com o auxílio de alguma ferramenta de modelagem. No entanto, todo profissional de tecnologia da informação precisa conhecer os princípios utilizados nessa conversão. A imagem em sequência exibe as tabelas originadas das entidades do DER do nosso exemplo, construído na etapa de projeto conceitual.



Tabelas originadas do DER da Escola

No modelo relacional, as entidades de um DER são representadas sob o formato de tabelas, por isso, no exemplo, aparecem as tabelas CLIENTE e CURSO. Em especial, a mesma decisão foi tomada para representar o relacionamento INSCRICAO. Perceba que nesse ponto do projeto ainda não definiremos as características dos atributos, tais como tipos de dados e tamanho. Basta apenas que eles estejam vinculados às suas tabelas.

Além da representação visual do projeto lógico, as tabelas podem ser expressas com o uso de representação textual.

Veja, a seguir, a descrição que corresponde às tabelas originadas das entidades do DER!

CLIENTE

idcliente, nome, datanascimento, CPF, email, telefone

CURSO

idcurso, nome, cargahoraria, descricao

INSCRICAO

idcurso, idcliente, datainscricao, datacancelamento

Observe que, com base na representação textual, podemos dizer que:

- Um cliente é caracterizado por um identificador, além de possuir as propriedades nome, data de nascimento, CPF, e-mail e telefone.
- Um curso possui um identificador, além das propriedades nome, carga horária e descrição.

Uma inscrição associa um cliente a determinado curso, além de possuir as propriedades data de inscrição e data de cancelamento.

Estamos finalizando a nossa jornada nas fases de um projeto de banco de dados. É chegada a hora de construir o projeto físico.

Atividade 2

As etapas do projeto de banco de dados envolvem a modelagem conceitual, lógica e física, cada uma delas com a sua própria finalidade. Qual é a finalidade principal do modelo lógico em um banco de dados?

A

Descrever a estrutura física do banco de dados.

B

Representar os procedimentos de segurança aplicados ao banco de dados.

C

Definir a estrutura organizacional da equipe de administração do banco de dados.

D

Descrever como os dados serão organizados, relacionados e acessados no banco de dados.

Documentar as decisões de negócios que levaram à criação do banco de dados.



A alternativa D está correta.

A principal finalidade de um modelo lógico em um banco de dados é descrever como os dados serão organizados, relacionados e acessados. Ele fornece uma representação abstrata dos dados, identificando tabelas, relacionamentos, chaves primárias, chaves estrangeiras e outros elementos essenciais para o entendimento da estrutura lógica do banco de dados. Diferentemente do modelo físico, que se preocupa com a implementação real no nível do sistema de gerenciamento de banco de dados (SGBD), o modelo lógico foca na representação conceitual dos dados e nas relações entre eles. As demais opções (a., b., c., e.) estão relacionadas a aspectos distintos da administração do banco de dados, mas não representam a finalidade principal do modelo lógico.

Projeto físico

Assista ao vídeo e entenda a transformação da modelagem lógica em uma estrutura concreta, definindo detalhes como tipos de dados, índices e otimizações para garantir eficiência na implementação do banco de dados.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Durante o projeto físico, definimos os detalhes de implementação dos objetos do banco de dados. No caso das tabelas, escolhemos os tipos de dados e tamanho das colunas, e especificamos se elas são opcionais ou obrigatórias.

Os relacionamentos são definidos por uma restrição especial em alguma(s) coluna(s) da tabela em questão. Esse tipo de restrição é denominado **chave estrangeira**. Em geral, o projeto físico é realizado com o auxílio de alguma ferramenta gráfica de modelagem. Há inclusive ferramentas que funcionam on-line, muitas vezes com a política de oferecer acesso limitado a diversos recursos.



Dica

Para o nosso exemplo, escolhemos a ferramenta on-line denominada Vertabelo por ser bastante funcional e sem custos para fins educacionais.

Ao iniciar o design do modelo, escolhemos o SGBD PostgreSQL como produto-alvo da modelagem. A imagem a seguir representa o modelo físico enriquecido com detalhes de implementação compatíveis com o SGBD escolhido. Confira!



Modelo físico do estudo de caso Escola

Observe que, diferentemente do modelo lógico, cada coluna de tabela no modelo está especificada com detalhes relativos ao tipo de dados, além de restrições em algumas colunas indicadas pelos marcadores **FK**, **PK** e **N**.

FK, PK, N

FKDo inglês foreign key, isto é, chave estrangeira.PKDo inglês primary key, traduzido como chave primária.NDo inglês null, que significa nulo.

Na criação do esquema do banco de dados, nós utilizamos uma linguagem declarativa, denominada linguagem de consulta estruturada SQL. A parte da SQL que fornece essas funcionalidades é denominada Linguagem de Definição de Dados. Observe a seguir um script SQL compatível com o modelo escola!

```

1 CREATE TABLE CLIENTE (
2     idcliente int NOT NULL,
3     nome char(90) NOT NULL,
4     datanascimento date NOT NULL,
5     CPF char(12) NOT NULL,
6     email char(50) NOT NULL,
7     telefone char(12) NULL,
8     PRIMARY KEY (idcliente) );
9 CREATE TABLE CURSO (
10    idcurso int NOT NULL,
11    nome char(90) NOT NULL,
12    cargahoraria int NOT NULL,
13    descricao char(120) NOT NULL,
14    PRIMARY KEY (idcurso) );
15 CREATE TABLE INSCRICAO (
16    idcurso int NOT NULL,
17    idcliente int NOT NULL,
18    datainscricao date NOT NULL,
19    datacancelamento date NULL,
20    PRIMARY KEY (idcurso,idcliente),
21    FOREIGN KEY (idcliente) REFERENCES CLIENTE (idcliente),
22    FOREIGN KEY (idcurso) REFERENCES CURSO (idcurso) );

```

Script DDL compatível com o estudo de caso Escola

Um script DDL é um conjunto de comandos que, no contexto do nosso exemplo, servirão para criar as tabelas do banco de dados escola.

Vamos entender o propósito desse código? Observe que:

- A declaração de cada tabela inicia com o comando CREATE TABLE nometabela, conforme linhas 1, 9 e 15.
- Todo cliente possui um identificador único (idcliente especificado na linha 8).
- Todo curso possui um identificador único (idcurso especificado na linha 14).

- Cada inscrição possui um identificador único, nesse caso composto por um par de colunas (idcurso, idcliente especificado na linha 20).
- As linhas 21 e 22 garantem que a inscrição será processada envolvendo necessariamente um cliente e um curso previamente existentes no banco de dados.

Ao longo deste conteúdo, estudamos as fases de um projeto de banco de dados. Percebemos que esta é uma atividade que envolve as seguintes tarefas: levantar requisitos de dados; construir um modelo de entidade e relacionamento; construir um modelo lógico; implementar o modelo físico.

As tarefas de um projeto de banco de dados não são estáticas, visto que os requisitos de dados podem evoluir, por exemplo, a partir da necessidade de adaptar o negócio a algum tipo de legislação, ou mesmo para tornar o trabalho do usuário mais eficiente.

Atividade 3

O projeto de banco de dados envolve atividades diversas, cada uma com a sua própria finalidade e correspondendo a uma etapa do projeto. Quanto a essas etapas, assinale a alternativa correta.

A

Projeto conceitual – É construído um modelo de dados em função de características técnicas de implementação existentes em algum SGBD, por exemplo, SQL Server.

B

Projeto lógico – São especificados esquemas lógicos. Além disso, essa etapa é responsável por criar um modelo físico de dados a partir do modelo conceitual independente do SGBD escolhido.

C

Projeto físico – Descreve detalhes sobre tipos de dados e tamanho de colunas. É dependente da escolha de um SGBD.

D

Projeto físico – Possui total independência de um SGBD específico.

E

Projeto lógico – Depende exclusivamente do SGBD a ser utilizado.



A alternativa C está correta.

Na etapa de projeto físico, a partir da escolha de um SGBD, são definidos detalhes de mais baixo nível, dependentes das funcionalidades ofertadas pelo SGBD escolhido, visto que o resultado dessa etapa corresponde ao banco de dados físico.

Entidades e relacionamentos

Adentrando agora ao mundo da modelagem visual, exploraremos o diagrama entidade-relacionamento (DER). O DER é uma ferramenta fundamental na concepção e comunicação de estruturas de banco de dados, pois proporciona uma visão intuitiva das relações entre diferentes componentes.

Convidamos você a se dedicar a esse processo de modelagem. Mãos à obra!

Neste vídeo, compreenda o diagrama entidade-relacionamento (DER), que traduz nossas ideias em representações gráficas, destacando entidades, atributos e suas interconexões.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Entidade

De acordo com Heuser (2009), a entidade corresponde a uma representação do conjunto de objetos da realidade modelada sobre os quais se deseja manter informações no banco de dados. Em um DER, a entidade é representada por um retângulo e dentro dele definimos o nome da entidade. Considere o seguinte exemplo!



Exemplo de representação gráfica de entidade

Relacionamento

Veja que o retângulo representa o conjunto de todos os alunos sobre os quais há interesse em manter informações no banco de dados.

Heuser (2009) afirma que a propriedade de entidade que especifica as associações entre objetos é o relacionamento, o qual corresponde a um conjunto de associações entre ocorrências de entidades. Em um DER, representamos o relacionamento por meio de um losango ligado por linhas conectadas às entidades envolvidas. Vamos observar o exemplo a seguir.



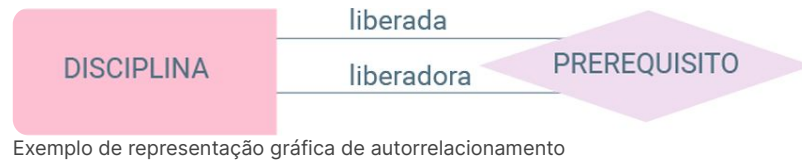
Exemplo de representação gráfica de relacionamento

Veja que há duas entidades, além do relacionamento POSSUI. Todo relacionamento pressupõe a existência dos objetos das entidades participantes.

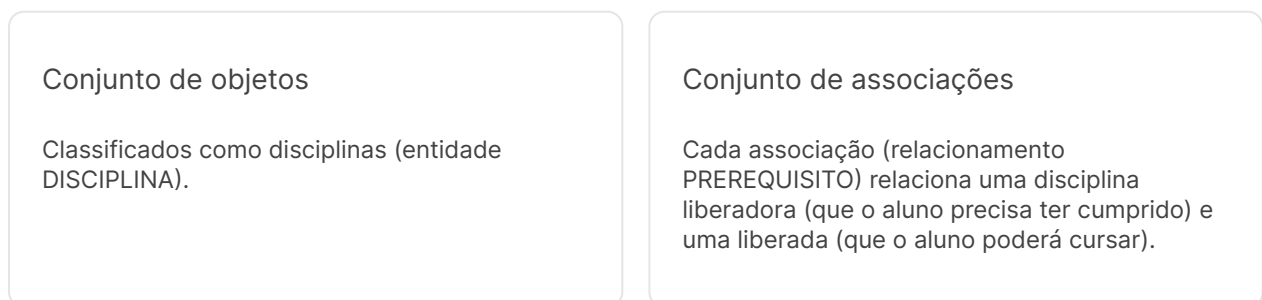
Autorrelacionamento

Há casos em que um relacionamento envolve ocorrências de uma mesma entidade (autorrelacionamento). Em especial, é importante diferenciar o papel que cada ocorrência da entidade cumpre no contexto do relacionamento em questão.

Suponha que, para cursar a disciplina Cálculo II, seja necessário ter conhecimentos em Cálculo I. Esse tipo de situação é conhecido por pré-requisito. A imagem em sequência apresenta um autorrelacionamento envolvendo pré-requisitos a partir da entidade DISCIPLINA. Confira!



O modelo contempla dois tipos de conjunto. São eles:



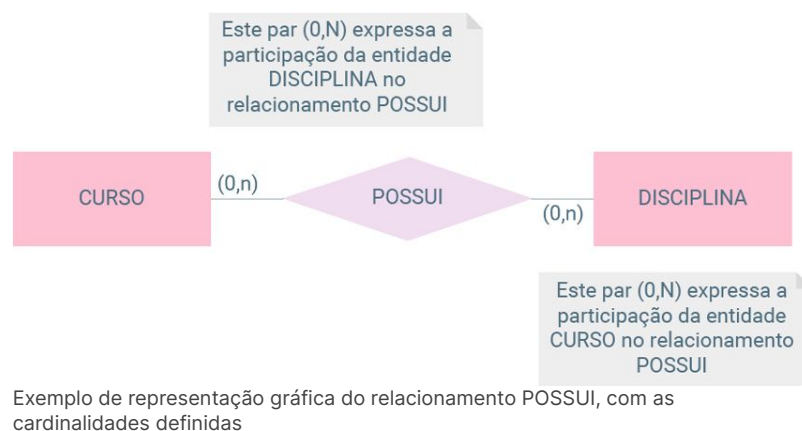
Em nosso exemplo, Cálculo II é a disciplina liberada e Cálculo I, a liberadora.

Cardinalidade de relacionamentos

Até o momento, identificamos um relacionamento (POSSUI) entre as entidades CURSO e DISCIPLINA. No entanto, surgiram quatro importantes perguntas:

1. Toda disciplina, para existir no banco de dados, tem de estar associada a algum curso?
2. Uma disciplina pode estar associada a, no máximo, quantos cursos?
3. Todo curso, para existir no banco de dados, deve estar associado a alguma disciplina?
4. Um curso pode estar associado a, no máximo, quantas disciplinas?

Expressaremos essas respostas no DER, usando o conceito de cardinalidade em relacionamentos. A cardinalidade é um par ordenado sob a forma (mínima, máxima): 0 ou 1 para a mínima e 1 ou N para a máxima, com N representando valores maiores que a unidade. Vejamos um exemplo.



Podemos então responder:

Pergunta 1

Não (cardinalidade mínima 0 expressa ao lado da entidade CURSO).

Pergunta 2

Vários (cardinalidade máxima n expressa ao lado da entidade CURSO).

Pergunta 3

Não (cardinalidade mínima 0 expressa ao lado da entidade DISCIPLINA).

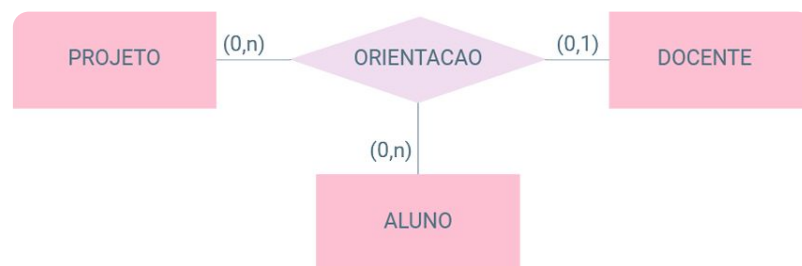
Pergunta 4

Várias (cardinalidade máxima n expressa ao lado da entidade DISCIPLINA).

Por convenção, cada par ordenado da cardinalidade diz respeito à participação da entidade localizada no lado oposto do relacionamento em questão.

Relacionamento ternário

Vamos modelar orientações de alunos em projetos, realizadas por docentes. Há três tipos de informações: projeto, aluno e docente. Estamos, portanto, diante de um relacionamento ternário. Confira na próxima imagem a parte do DER contemplando esse requisito de dados.



Exemplo de relacionamento ternário

Cada ocorrência do relacionamento ORIENTACAO vincula três ocorrências de entidade: um projeto, um aluno a ser orientado e um docente orientador. Em um relacionamento ternário, especificamos cada par de cardinalidade com base na relação existente entre o par de cardinalidade restante. Veja a seguir o que expressa cada par de cardinalidade!

1

Cardinalidade máxima 1

Expressa no modelo ao lado da entidade DOCENTE, diz respeito ao par (ALUNO, PROJETO). Um aluno participante de um projeto pode ser orientado por no máximo um docente.

2

Cardinalidade máxima N

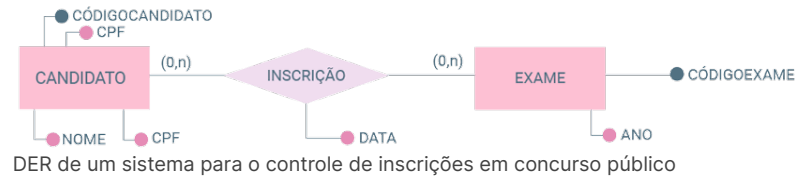
Expressa no modelo ao lado da entidade ALUNO, diz respeito ao par (DOCENTE, PROJETO). Um docente participante de um projeto pode orientar diversos alunos.

3 Cardinalidade máxima N

Expressa no modelo ao lado da entidade PROJETO, diz respeito ao par (ALUNO, DOCENTE). Um aluno e um docente podem participar de vários projetos.

Atividade 1

Durante projeto conceitual de um sistema para o controle de inscrições em concurso público, foi desenvolvido o DER apresentado na imagem a seguir. A respeito deste DER, assinale a alternativa correta.



A

A fim de adicionar ao modelo informações do EXAME para, posteriormente, armazenar dados referentes às provas, a alternativa mais indicada é criar um atributo simples na entidade CANDIDATO.

B

O atributo DATA está representado incorretamente no modelo, uma vez que não é permitido modelar atributo em relacionamento.

C

De acordo com o modelo apresentado, um CANDIDATO pode inscrever-se em mais de um EXAME.

D

No modelo, há um relacionamento ternário, pois estão envolvidos três objetos: CANDIDATO, INSCRICAO e EXAME.

E

No modelo apresentado, temos um relacionamento um para vários.



A alternativa C está correta.

A proposição da alternativa C está correta, pois a informação sobre o número de vezes em que um CANDIDATO pode fazer inscrição em exames é definido pela cardinalidade máxima n, expressa ao lado da entidade EXAME.

Atributo

Avançando na nossa jornada sobre banco de dados, nosso foco agora será a modelagem de atributos. Aprofundaremos a compreensão dos dados, detalhando características específicas de cada entidade.

Vamos juntos explorar as técnicas de modelagem de atributos!

Aprenda, neste vídeo, a definição precisa de atributos, considerando tipos de dados, restrições e particularidades que enriquecem a representação dos elementos no banco de dados.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Entidades e relacionamentos podem ter propriedades, que são especificadas pelos atributos. Segundo Heuser (2009), atributo corresponde a um dado que é associado a cada ocorrência de uma entidade ou de um relacionamento.

Vamos especificar algumas propriedades para as entidades CURSO e DISCIPLINA:

- Todo curso possui um código único, nome e, opcionalmente, data de criação.
- Toda disciplina possui um código único, nome e carga horária.

Observe a imagem que apresenta a parte do DER contemplando esses requisitos de dados!



Percebemos que os atributos CODIGOCURSO e CODIGODISCIPLINA são únicos em suas respectivas entidades. Veja o que essa unicidade significa na prática!

CODIGOCURSO

Todo curso possui valor para o atributo CODIGOCURSO diferente dos demais.

CODIGODISCIPLINA

Toda disciplina possui valor para o atributo CODIGODISCIPLINA diferente dos demais.

Esse tipo especial de atributo é conhecido por atributo identificador e sua representação gráfica é dada por um traço com uma das extremidades contendo um círculo preenchido. De acordo com os requisitos de dados, DATADECRIACAO é um atributo opcional, ou seja, não obrigatório. Sua representação gráfica é dada por um traço com uma das extremidades contendo um círculo pontilhado. Os demais atributos são obrigatórios.

Cardinalidade em atributo

No DER anterior, ao lado do atributo DATAACRIACAO, há um par de cardinalidade com valor (0,1). A cardinalidade 0 expressa que o atributo é opcional. A cardinalidade 1 expressa que o atributo é monovalorado. Cada combinação de cardinalidade tem um significado especial. Vejamos!

Cardinalidade Mínima	Cardinalidade Máxima	Significado
0	1	Opcional, Monovalorado
0	N	Opcional, Multivalorado

Cardinalidade Mínima	Cardinalidade Máxima	Significado
1	1	Obrigatório, Monovalorado
1	N	Obrigatório, Multivalorado

Propriedade de atributo de acordo com a cardinalidade
Nathielly de Souza Campos

Atributo obrigatório e monovalorado

Na construção de um DER, a maioria dos atributos é monovalorado e obrigatório. Assim, adotaremos a convenção de, nesses casos, não expressar no modelo a cardinalidade (1,1) por motivos de legibilidade. Assim, de agora em diante, quando não houver cardinalidade expressa em atributos de um DER, considere que eles são monovalorados e obrigatórios.

Atributo composto

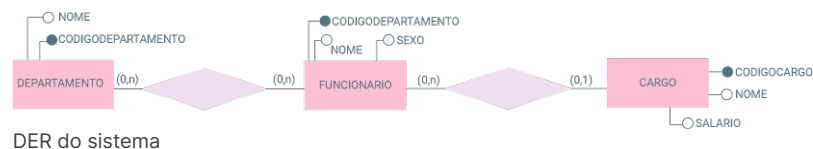
Em modelagem, é comum surgirem atributos mais complexos, que podem ser subdivididos em partes menores; eles são conhecidos por atributos compostos. Por exemplo, um atributo endereço pode ser subdividido em logradouro, complemento, CEP e cidade.

Modelo de entidade e relacionamento estendido

O modelo de entidade e relacionamento estendido traz novos componentes semânticos. Estudaremos a especialização/generalização, além da entidade associativa.

Atividade 2

Durante a modelagem conceitual de um sistema, foi criado o DER apresentado na imagem a seguir. Quanto às características dos atributos, das entidades e dos relacionamentos do modelo, assinale a alternativa correta.



- Todo funcionário está alocado em um departamento.
- Não pode existir mais de um cargo com o mesmo salário.
- SALARIO é considerado um atributo obrigatório em CARGO.
- CODIGOCARGO é considerado um atributo obrigatório na entidade CARGO.
- O atributo nome da entidade FUNCIONARIO é do tipo composto.

A

III e IV

B

III e V

C

I, III e IV

D

I e II

E

II, III e IV



A alternativa A está correta.

A proposição III está correta, pois, neste estudo, convencionamos que todo atributo sem a cardinalidade explícita será considerado obrigatório e monovalorado. A proposição IV está correta, pois todo atributo identificador por padrão é obrigatório.

Especialização/generalização

Explorando a complexidade e a flexibilidade da modelagem em banco de dados, direcionamos nosso olhar para a estrutura de generalização e especialização.

A generalização nos permite capturar a essência de semelhanças entre entidades, enquanto a especialização detalha características únicas.

Vamos ao nosso estudo! Veja, neste vídeo, como representar hierarquias e categorias dentro do sistema, criando uma abordagem robusta para lidar com entidades relacionadas.

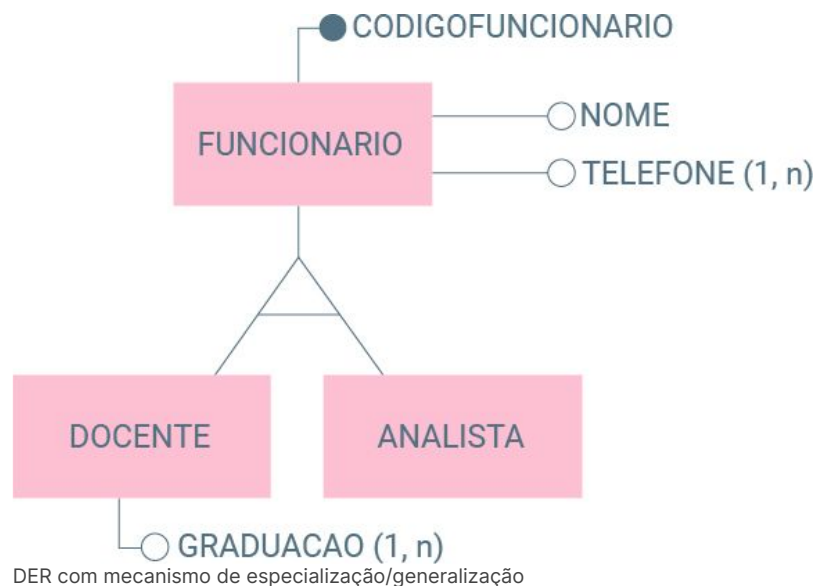


Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Imagine que, além dos docentes, seja necessário gerenciar outros funcionários da instituição, como os analistas. Podemos deixar previsto que a IES pode ter funcionários que não são analistas nem docentes. Queremos, ainda, saber a formação de graduação de cada docente. Surge, então, uma hierarquia, visto que docente é um subtipo de funcionário. Funcionário é um objeto mais genérico, estando, portanto, na posição superior da hierarquia.

O mecanismo de especialização/generalização é representado por um triângulo, com a entidade mais genérica localizada na parte superior e a(s) entidade(s) especializada(s) na parte inferior. Vamos observar o DER na imagem a seguir com os novos requisitos de dados.



Podemos perceber que todo funcionário possui um código único, além de nome e pelo menos um telefone, e que há duas entidades especializadas: DOCENTE e ANALISTA. A entidade DOCENTE possui um atributo obrigatório GRADUACAO.

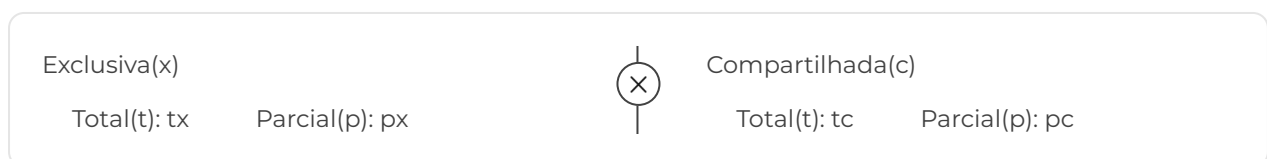
No mecanismo de especialização/generalização há o uso de herança de propriedades: cada entidade especializada herda as propriedades da entidade mais genérica. Assim, todo docente herda as propriedades de funcionário.

Classificações para especialização / generalização

Observe que perguntas surgem quando analisamos o DER apresentado anteriormente.

- Pode existir funcionário que não seja nem docente nem analista?
- Pode existir funcionário que seja docente e analista?

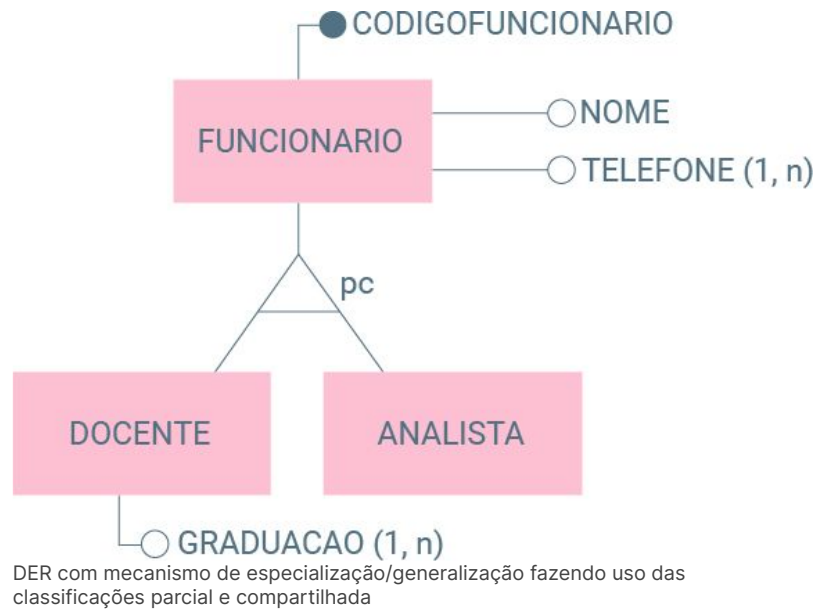
A classificação total/parcial responde ao primeiro questionamento: se a resposta for não, a especialização é total; caso contrário, parcial. O segundo é respondido com auxílio da classificação exclusiva/compartilhada: se a resposta for não, a especialização é exclusiva; caso contrário, compartilhada. Veja as combinações das classes expressas:



Vamos convencionar respostas às perguntas realizadas:

- Pode existir funcionário que não seja nem docente nem analista? Sim
- Pode existir funcionário que seja docente e analista? Sim

Agora vamos observar o DER na imagem seguinte, já com a informação (pc) sobre a classificação do mecanismo de especialização/generalização.



No exemplo, **parcial** significa que pode existir funcionário não especializado, ou seja, não classificado como docente ou analista. Por fim, compartilhado significa que no contexto da IES pode existir funcionário que atue como docente e na função de analista.

Atividade 3

Além dos elementos básicos da modelagem conceitual, entidades, relacionamentos e atributos, existem extensões que visam atender a necessidades específicas de modelagem. A respeito dessas extensões, podemos afirmar que a principal finalidade da relação de especialização/generalização é

A

reduzir a quantidade de dados armazenados no banco de dados.

B

melhorar o desempenho das consultas.

C

organizar os dados em tabelas relacionadas.

D

representar hierarquias de classes e suas relações no mundo real.

E

garantir a integridade referencial entre as tabelas.



A alternativa D está correta.

A relação de especialização/generalização é utilizada para representar hierarquias de classes no mundo real. Esse conceito é frequentemente associado ao modelo de dados orientado a objetos, no qual classes podem ser organizadas em uma hierarquia com uma classe mais geral (superclasse) e classes mais específicas (subclasses) herdando características da superclasse. Essa abordagem ajuda a modelar a relação "é-um" entre as classes e permite uma representação mais fiel da estrutura da informação no contexto do domínio do problema. As demais opções (a., b., c., e.) não representam adequadamente a finalidade da relação de especialização/generalização.

Entidade associativa

Agora, voltamos nossa atenção para a estrutura de agregação e a entidade associativa. A entidade associativa é fundamental na captura de informações adicionais sobre essas relações, enriquecendo a compreensão do sistema.

Aprenda, neste vídeo, a modelar relacionamentos complexos entre entidades, utilizando agregações para representar conexões mais elaboradas.

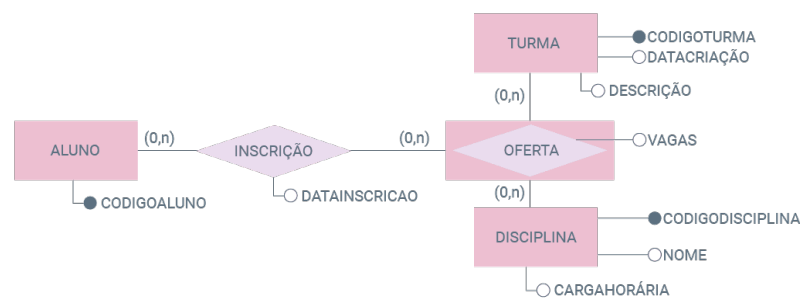


Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

É comum a necessidade de vincular entidade a algum relacionamento. Ao modelarmos a inscrição de alunos em disciplinas, criaremos turmas para serem associadas às disciplinas que serão liberadas para inscrição. Suponha que toda turma possui código, descrição e data de criação. Além disso, pode estar associada a diversas disciplinas. Uma disciplina pode ser ofertada em várias turmas. Ao ofertar uma disciplina, é necessário saber o número de vagas e quando o aluno fez inscrição.

Haverá um relacionamento (OFERTA) entre TURMA e DISCIPLINA. OFERTA deverá estar associada a ALUNO, via relacionamento INSCRIÇÃO. Como resolver esse impasse? O objeto entidade associativa surgiu como alternativa de modelagem em situações dessa natureza. Ela é representada por um losango desenhado dentro de um retângulo. Observe o seguinte modelo!



DER com os novos requisitos de dados, fazendo uso de entidade associativa

É necessário enxergar OFERTA sob duas perspectivas. São elas!

Relacionamento

OFERTA possui atributo VAGAS, útil no planejamento das turmas e disciplinas que serão ofertadas para inscrição.

Entidade

OFERTA útil para identificar a turma e a disciplina escolhida pelo aluno quando do momento de uma inscrição.

Atividade 4

As extensões do DER permitem que o modelo apresente uma semântica mais evoluída, representando situações específicas do mundo real. Entre as extensões, a entidade associativa tem a finalidade principal de:

A

melhorar o desempenho das consultas.

B

evitar a duplicação de dados.

C

representar uma tabela intermediária em relacionamentos muitos-para-muitos.

D

organizar os dados em tabelas relacionadas.

E

garantir a integridade referencial entre as tabelas.



A alternativa C está correta.

Uma entidade associativa permite que uma entidade estabeleça relação com o relacionamento entre as outras duas entidades. Quando duas entidades têm um relacionamento direto muitos-para-muitos, é comum introduzir uma entidade associativa para armazenar as informações associadas a esse relacionamento. Ela contém chaves estrangeiras das entidades relacionadas, servindo como uma ponte para resolver esse tipo de relacionamento complexo. Isso ajuda a evitar a duplicação de dados e a organizar as informações de maneira mais eficiente. As demais opções (a., b., d., e.) não representam adequadamente a finalidade principal de uma entidade associativa.

DER na prática

Agora que vimos os fundamentos da modelagem conceitual, vamos realizar um estudo de caso para aplicar nossos conhecimentos. Vamos modelar uma empresa de limpeza. Entenda o seu minimundo!

A firma Serviços Domésticos presta serviços de limpeza e deseja um sistema autorizado que cuide, principalmente, de alocação dos empregados aos períodos de serviço feitos pelos clientes.

O cliente telefona para a firma e faz um pedido de serviço. Nesse momento, a atendente verifica no cadastro se ele já é um cliente da firma; caso não seja, ela o cadastra, solicitando: CNPJ, razão social, endereço e telefone, se for pessoa jurídica, ou CPF, nome, endereço e telefone, se for pessoa física. A firma atribui um código próprio para identificar qualquer um dos seus clientes.

O cliente telefona para a firma e faz um pedido de serviço. Nesse momento, a atendente verifica no cadastro se ele já é um cliente da firma; caso não seja, ela o cadastra, solicitando: CNPJ, razão social, endereço e telefone, se for pessoa jurídica, ou CPF, nome, endereço e telefone, se for pessoa física. A firma atribui um código próprio para identificar qualquer um dos seus clientes. Estando o cliente devidamente cadastrado, a atendente abre um bloco de pedidos e preenche um pedido de serviços, anotando: o nome do cliente, a data da abertura, a data para realização dos serviços, o local onde deverá ser realizado e uma descrição dos serviços que o cliente solicitou naquele local, com a respectiva metragem quadrada de cada um. A atendente consulta uma tabela de serviços (códigos do serviço, descrição, valor por m2, duração por m2) e anota no

pedido a duração e o valor unitário e total dos serviços. Após o preenchimento do pedido, a atendente informa ao cliente o número do seu pedido, o valor total a ser pago e a duração máxima para o término dos serviços.

Todo empregado está habilitado a executar pelo menos um tipo de serviço de limpeza e somente pode ser alocado em serviços em que está habilitado, definindo, dessa forma, “qual empregado” vai executar “qual serviço do pedido”. Um empregado realiza apenas um serviço em um pedido.

Assista ao vídeo e veja como modelar uma empresa de limpeza por meio do nosso estudo de caso.



Conteúdo interativo

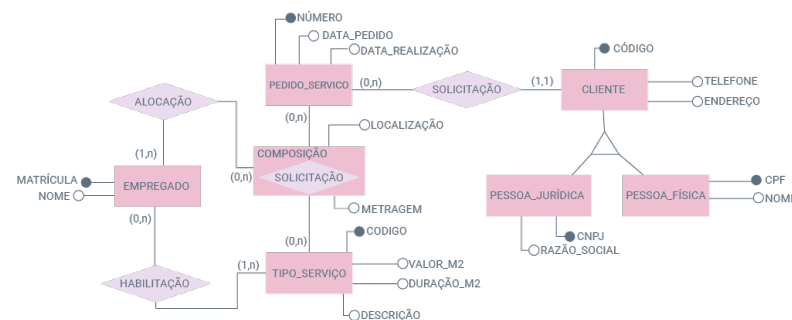
Acesse a versão digital para assistir ao vídeo.

Roteiro de prática

Para realizarmos a modelagem conceitual a partir do mundo real, devemos seguir os seguintes passos:

1. Identificar as entidades existentes.
2. Definir as entidades tipo e representá-las no DER.
3. Identificar as associações entre as entidades, ou seja, os seus relacionamentos.
4. Definir os relacionamentos tipo e representá-los no DER.
5. Estabelecer as cardinalidades dos relacionamentos.
6. Definir a cardinalidade entre os relacionamentos tipo e representá-las no DER.
7. Verificar se existem extensões como especialização/generalização e entidade associativa e representá-las no DER.
8. Identificar e representar os atributos.

Confira a DER de solução!



Resultado DER de solução

Atividade 5

Imagine que você está desenvolvendo um sistema para uma biblioteca digital que armazena diferentes tipos de documentos, como livros, artigos e teses. Além disso, o sistema precisa gerenciar informações sobre os autores desses documentos. Diante dessa situação, explique em que contexto você optaria por utilizar uma estrutura de generalização/especialização na modelagem do sistema, e como isso contribuiria para a eficiência e a flexibilidade do sistema.

Chave de resposta

A utilização de uma estrutura de generalização/especialização seria apropriada no contexto de modelagem de dados para os autores dos documentos na biblioteca digital. Nesse caso, poderíamos ter uma classe geral chamada Autor, que abrange características comuns a todos os tipos de autores, como nome e

biografia. Em seguida, poderíamos ter subclasses específicas, como `AutorLivro`, `AutorArtigo` e `AutorTese`, que herdam as características da classe principal `Autor` e adicionam atributos específicos, como ISBN para autores de livros, ISSN para autores de artigos, e assim por diante.

A estrutura de generalização/especialização é apropriada quando há uma entidade genérica que pode ser especializada em subtipos mais específicos. No caso da biblioteca digital, a entidade genérica é o `Autor`, e os subtipos específicos são os autores de livros, artigos e teses. Essa abordagem facilita a modelagem, pois evita a redundância de dados comuns a todos os autores, como nome e biografia, enquanto permite a inclusão de atributos específicos para cada tipo de autor.

Além disso, essa estrutura proporciona flexibilidade, pois o sistema pode lidar com diferentes tipos de autores de maneira uniforme, simplificando operações como busca, atualização e exclusão. Também facilita a manutenção do sistema, uma vez que mudanças nas características comuns dos autores podem ser feitas na classe principal `Autor` e refletirão automaticamente nos subtipos. Em resumo, a estrutura de generalização/especialização melhora a eficiência, reduz a redundância e torna o sistema mais flexível e fácil de manter.

Processo de modelagem de entidades e relacionamentos

Abordaremos agora o processo central de modelagem de entidades e relacionamentos. Os relacionamentos delineiam as conexões vitais entre essas entidades, formando a espinha dorsal do nosso design.

Vamos à modelagem!

Saiba, neste vídeo, a identificar entidades importantes para nosso sistema, compreendendo suas características e interações.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

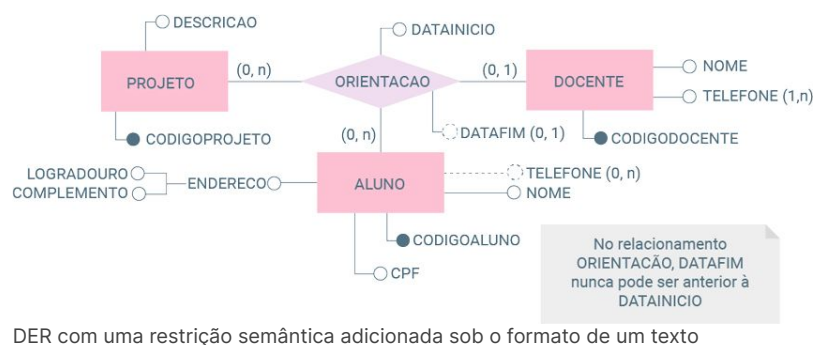
Objetivos ao construir um DER

Um DER deve capturar as partes mais importantes do negócio sendo modelado, pois pessoas diferentes precisam ter o mesmo entendimento do modelo. Manter um DER atualizado facilita a vida dos profissionais de TI, já que, com a documentação atualizada, a curva de aprendizado sobre a organização pode ser minimizada.

Apesar de ser relativamente simples construir um DER, é comum identificarmos propriedades desejáveis no banco de dados que precisam ser registradas mas não podem ser expressas diretamente no modelo. Na linguagem técnica de banco de dados, regras que devem ser obedecidas pelo SGBD em relação ao banco de dados são conhecidas por restrições de integridade.

Há restrições de integridade expressas por meio de algum elemento do próprio DER, por exemplo, ao definirmos cardinalidade para algum atributo. No entanto, há restrições que precisam ser expressas em separado, normalmente usando linguagem natural. Essas, normalmente, são conhecidas como restrições semânticas.

Podemos estabelecer que, no relacionamento de orientação realizado por docentes aos alunos no contexto de um projeto, a data de término da orientação nunca deve ser menor que a data de início. Estamos diante de uma restrição que leva em conta dois atributos de relacionamento definidos no DER. Essa restrição é especificada no DER com o uso de linguagem natural sob a forma de um texto. Entenda como ficou o modelo!



DER com uma restrição semântica adicionada sob o formato de um texto

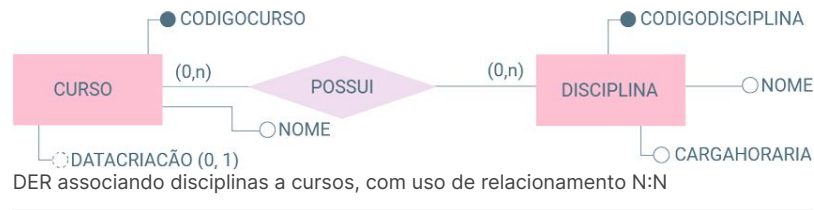
O quanto devo modificar um DER para registrar restrições de integridade? O objetivo fundamental na construção de um DER é projetar um banco de dados e não descrever todas as restrições de integridade.

Modelagem de entidades e relacionamentos

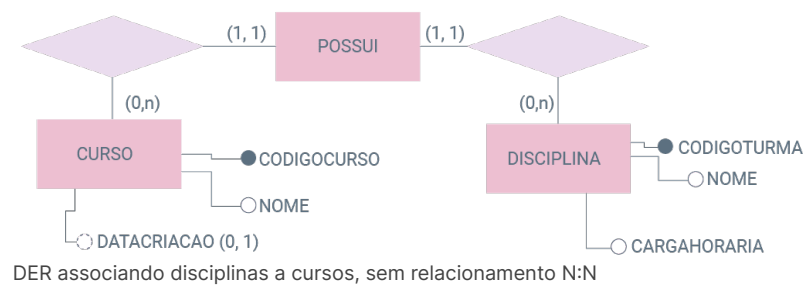
A partir de agora, daremos ênfase no processo de modelagem de entidades e relacionamentos. No entanto, modelos diferentes podem obter o mesmo resultado prático?

Quanto à equivalência entre modelos, Heuser (2009) afirma que dois modelos ER são equivalentes quando ambos geram o mesmo esquema de banco de dados.

Considere o DER expresso na seguinte imagem!



Ao analisarmos o DER, percebemos que a cardinalidade máxima do relacionamento POSSUI é do tipo **N:N**. Sendo assim, um modelo equivalente sem uso de relacionamento **N:N** será criado, onde POSSUI será modelado como entidade, conforme expresso a seguir:



Generalizando, todo relacionamento N:N pode ser transformado em entidade. Para isso, basta acompanhar as etapas a seguir:

1. Representar o relacionamento N:N como uma entidade: foi criada a entidade POSSUI.
2. Relacionar a entidade criada na etapa 1 às entidades participantes do relacionamento original: a entidade POSSUI foi relacionada às entidades CURSO e DISCIPLINA.
3. Adicionar à entidade criada na etapa 1 o(s) atributo(s) – caso exista(m) – do relacionamento original: como não havia atributos no relacionamento original, a entidade POSSUI foi criada sem atributos.
4. A entidade criada na etapa 1 será identificada pelos relacionamentos com as entidades participantes do relacionamento original: esta identificação está representada no modelo com as linhas mais espessas saindo da entidade POSSUI.
5. Estabelecer a cardinalidade (1,1) da entidade criada na etapa 1 para cada relacionamento vinculado a ela: as cardinalidades estão representadas ao lado da entidade POSSUI.

O relacionamento POSSUI só será cadastrado no banco de dados se houver tanto uma disciplina quanto um curso. Essa observação é consequência da definição de relacionamento em modelagem de dados. Assim, quando transformamos um relacionamento em entidade, essa restrição deve ser mantida. De que maneira?

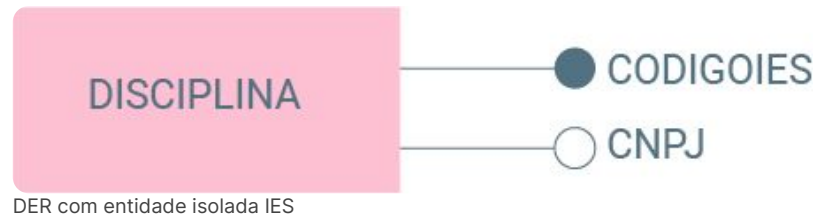
No DER modificado, as linhas mais espessas saindo da entidade POSSUI representam a restrição de que toda ocorrência desta entidade é dependente da existência tanto de uma disciplina quanto de um curso. Alguns autores chamam entidade fraca a uma entidade que depende de outra(s) para existir.

Modelagem de entidade isolada

O DER que estamos modelando é referente ao funcionamento de uma única IES. Em função disso, não há necessidade de informar no banco de dados em qual instituição o aluno está cadastrado ou mesmo à qual IES

os cursos estão relacionados, já que, de acordo com Heuser (2009), uma entidade isolada é uma entidade que não apresenta relacionamento com outras entidades.

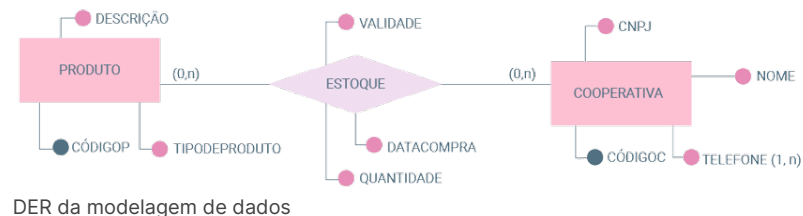
No contexto do nosso DER, uma entidade isolada pode servir para modelar características da IES, tais como código e CNPJ. Portanto, se decidirmos adicionar uma entidade INSTITUICAO ao modelo, esta deverá permanecer isolada, dado que, de acordo com os requisitos de dados, o banco de dados refere-se a uma única IES. Veja que a próxima imagem representa a entidade isolada INSTITUICAO e seus atributo.



Observe que com a entidade INSTITUICAO criada, caso haja necessidade de acrescentar outras características simples à IES, basta adicionar atributos. Por exemplo, poderíamos adicionar informações sobre data de criação, nome da IES, telefone, entre outras.

Atividade 1

Uma cafeteria que comercializa produtos alimentícios realiza compras diárias de uma cooperativa e está enfrentando dificuldades por causa de gestão do prazo de validade de seus produtos. O proprietário vai investir em um sistema que realizará o controle do estoque e do prazo de validade tomando como base a data de compra de cada produto adquirido da cooperativa. O DER parcial a seguir representa a modelagem de dados proposta por um profissional, sendo que o atributo código em cada entidade é único.



A partir das informações, é correto afirmar que

A

o relacionamento entre as entidades PRODUTO e COOPERATIVA é do tipo um-para-muitos.

B

o atributo telefone da entidade COOPERATIVA é do tipo composto.

C

o atributo CODIGOP da entidade PRODUTO é classificado como opcional.

D

é possível construir um novo DER modelando estoque como entidade, sem perda de informação em relação ao modelo original.

E

o atributo NOME da entidade cooperativa é do tipo único.



A alternativa D está correta.

De fato, quando o DER tem relacionamento N:N, é possível construir um modelo equivalente, em que o relacionamento em questão será substituído por uma entidade, mantendo a semântica do modelo original.

Quando manter histórico

Ao adentrarmos na complexidade da gestão de dados, surge o dilema: quando manter históricos? Exploraremos agora esse tema fundamental no banco de dados. A decisão de preservar históricos é estratégica e envolve considerações sobre auditoria, análises temporais e conformidade.

Assista ao vídeo e aprofunde seu entendimento sobre a necessidade de preservar históricos ao longo do tempo, para garantir a integridade e a precisão dos dados.



Conteúdo interativo

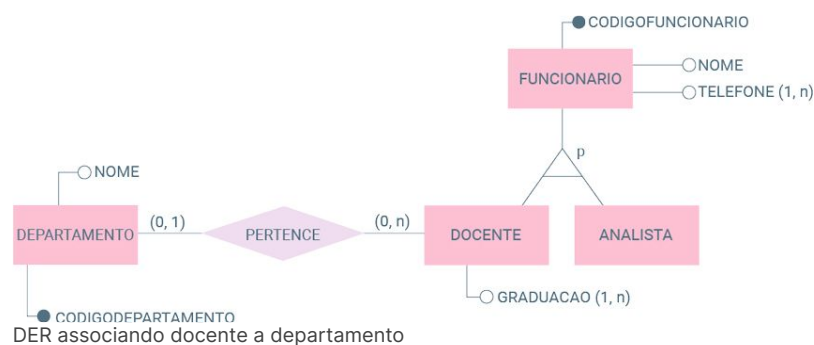
Acesse a versão digital para assistir ao vídeo.

No modelo acadêmico em questão, vamos considerar que surgiu a necessidade de associar docente a departamento. Esta é uma necessidade comum em praticamente toda IES: saber quais professores pertencem a qual departamento.

Vamos supor que, após o surgimento dessa demanda, levantamos os seguintes requisitos de dados:

- Todo departamento é identificado por um código e possui um nome.
- Um docente pode estar associado a, no máximo, um departamento.

Com base nos requisitos, repare no DER construído.



Podemos perceber que o DER construído está de acordo com os novos requisitos de dados, pois as informações dos departamentos estão expressas na entidade DEPARTAMENTO, e a restrição de que um docente está associado a, no máximo, 1 departamento está expressa na cardinalidade máxima 1 ao lado da entidade DEPARTAMENTO.

Os requisitos de dados associados a um projeto de banco de dados podem sofrer alterações ao longo do tempo. Assim, é necessário planejarmos uma estratégia para refletir as restrições impostas pelo novo cenário. Vamos supor, então, o surgimento de novo requisito de dados. Entenda o que deve ser considerado!

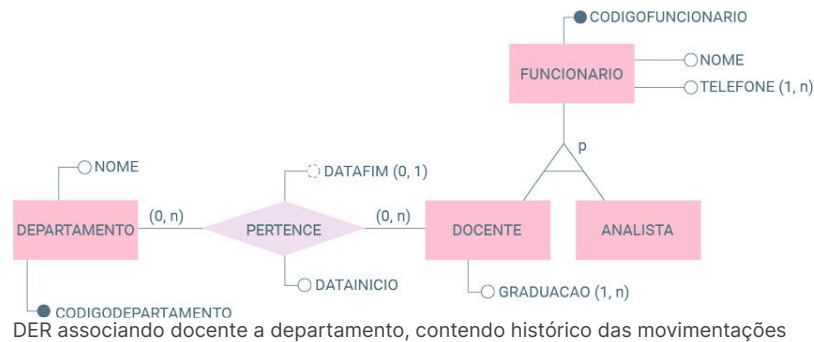
1

O armazenamento, no banco de dados, da movimentação de docentes entre departamentos.

2

A atuação do docente em um departamento possui uma data de início e uma data de fim.

O novo requisito de dados pode ser interpretado como uma necessidade institucional de manter histórico das movimentações de docentes entre departamentos. Veja o DER contemplando o novo requisito!



Podemos perceber que, no novo DER, a cardinalidade máxima N expressa ao lado da entidade DEPARTAMENTO indica que ao longo do seu ciclo profissional um docente pode ter passado por diversos departamentos da IES. Além disso, a atuação do docente em um departamento é caracterizada por uma data de início e uma data de fim.

Perceba que o DER contemplando o novo requisito de dados é semelhante ao DER original. A diferença foi a alteração de uma cardinalidade máxima, além da adição de dois atributos no relacionamento.

Atividade 2

Uma rede de operadoras de planos de saúde necessita de um sistema para controlar seus contratos junto à rede credenciada de hospitais. Cada contrato entre a operadora e um hospital tem duração de dois anos. Após o término desse prazo, um novo poderá ser emitido, caso haja interesse entre as partes. Qual alternativa a seguir representa um DER adequado aos requisitos de dados apresentados?

A

B

C

D

E



A alternativa B está correta.

De fato, de acordo com os requisitos de dados, uma operadora pode realizar diversos contratos com o mesmo hospital e vice-versa. Isso implica que a cardinalidade máxima do relacionamento CONTRATO seja do tipo N:N. Além disso, os atributos DATAINICIO e DATAFIM são propriedades associadas ao relacionamento CONTRATO, permitindo recuperar informações históricas.

Mais sobre modelagem de entidades e relacionamentos

Entrando em território mais detalhado da modelagem, exploraremos agora aspectos avançados do diagrama entidade-relacionamento (DER). Assim, você poderá aprimorar suas habilidades na criação de DERs mais sofisticados, proporcionando uma visão mais completa e refinada das complexas interações dentro do banco de dados. Sigamos adiante!

Descubra, neste vídeo, aspectos avançados do diagrama entidade-relacionamento (DER), aprimorando nossa compreensão sobre como representar de maneira precisa e abrangente as relações entre entidades.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Você deve ter notado que um DER não é construído em uma única etapa. Em projetos de banco de dados reais, a construção de um DER é um processo incremental, ou seja, o modelo vai sendo modificado e enriquecido de forma gradativa à medida que novos requisitos de dados sejam levantados.

Ao longo dos exemplos que construímos, utilizamos uma estratégia que envolveu a modelagem de conceitos mais abstratos que em seguida foram sendo detalhados. Em geral, identificamos algumas entidades e depois definimos seus atributos e relacionamentos. A estratégia de modelagem escolhida é conhecida por estratégia descendente. O passo a passo a seguir resume as etapas utilizadas ao adotarmos esse tipo de estratégia para construção do diagrama de entidade e relacionamento. Acompanhe!

1

Modelo inicial

Identificar entidades, relacionamentos, especializações e cardinalidade máxima de relacionamentos, bem como mapear atributos e identificadores de entidades e relacionamentos.

2

Modelo detalhado

Definir as cardinalidades dos relacionamentos, além de identificar outras restrições de integridade.

3

Validação

Revisar e validar o modelo junto ao usuário.

É importante observarmos que em qualquer etapa é possível retornar à etapa anterior, dado que estamos diante de uma construção que envolve um DER que ocorre de forma gradativa.

Atividade 3

Ao realizarmos a modelagem conceitual de um cenário, necessitamos identificar os seus elementos principais: entidades, relacionamentos e atributos. A respeito da técnica de modelagem, como podemos identificar entidades em um minimundo?

A

Identificando apenas objetos tangíveis no minimundo.

B

Examinando os processos de negócio realizados no minimundo.

C

Identificando somente os eventos importantes ocorridos no minimundo.

D

Analisando as características específicas dos usuários do minimundo.

E

Identificando tanto objetos tangíveis quanto conceitos abstratos relevantes no minimundo.



A alternativa E está correta.

Ao identificar entidades em um minimundo, é crucial considerar tanto objetos tangíveis quanto conceitos abstratos que desempenham um papel significativo no contexto do sistema que está sendo modelado. A modelagem de dados visa representar não apenas os elementos físicos (entidades tangíveis) no ambiente, mas também os conceitos abstratos que descrevem informações importantes no domínio do problema. Portanto, a escolha da opção (e.) é a mais apropriada, pois abrange tanto os objetos tangíveis quanto os conceitos abstratos relevantes no minimundo. As demais opções (a., b., c., d.) não refletem a abordagem abrangente necessária para identificar entidades de maneira adequada.

Modelagem entidade relacionamento na prática

Agora que estudamos com mais detalhes a modelagem das entidades e dos relacionamentos, faremos o estudo de caso da TV a cabo. Entenda seu minimundo a seguir!

Uma empresa de TV a cabo deseja automatizar seu serviço de cobrança. Para isso, todos os requisitos a seguir devem ser preenchidos:

- Um canal deve ser identificado por seu número, e deve conter o nome desse canal.
- Um pacote é um conjunto de canais que é disponibilizado ao cliente mediante uma assinatura. Cada pacote criado pela empresa tem que conter no mínimo um canal, ou vários, e um canal só pode pertencer a um, e somente um, pacote. O pacote deve possuir como informações o nome do pacote, que não se repete, e o preço, além dos canais que ele contém.
- Para um cliente ser cadastrado, ele precisa fazer uma assinatura de ao menos um pacote. Para o cadastro de um cliente, deve ser guardado seu CPF, nome, endereço (logradouro, número, complemento e CEP) e seu e-mail (um cliente pode ter nenhum ou vários e-mails).

- Uma assinatura deve possuir um, e somente um, cliente, um ou vários pacotes, data de ativação da assinatura, data de desativação da assinatura e o endereço de instalação.

Confira, neste vídeo, o estudo de caso sobre uma TV a cabo que deseja automatizar o serviço de cobrança a partir da modelagem das entidades e dos relacionamentos.



Conteúdo interativo

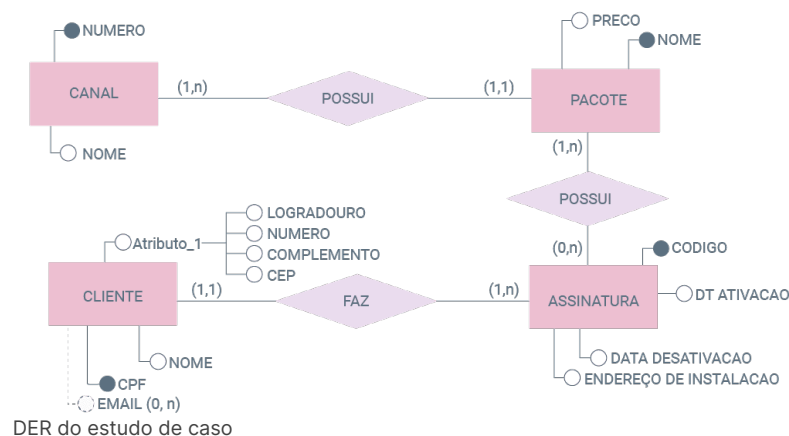
Acesse a versão digital para assistir ao vídeo.

Roteiro de prática

Para resolver um estudo de caso de modelagem a partir de modelo descritivo, no minimundo do enunciado você deve:

1. Identificar no modelo descritivo as entidades tipo e representá-las no DER.
2. Identificar no modelo descritivo os relacionamentos tipo e representá-los no DER.
3. Estabelecer as cardinalidades dos relacionamentos tipo e representá-las no DER.
4. Modelar os atributos.

Observe o DER de solução.



Atividade 4

Você está envolvido no desenvolvimento de um sistema de gerenciamento para uma empresa de logística que realiza o transporte de mercadorias entre diferentes cidades. Inicialmente, modelou-se o sistema com um relacionamento direto entre as entidades Cidade e Transporte, no qual cada transporte é associado a uma ou mais cidades de origem e destino. No entanto, ao longo do desenvolvimento surgiu a necessidade de rastrear informações específicas sobre cada rota de transporte, como distância percorrida, condições climáticas durante a viagem e eventuais paradas intermediárias.

Discuta em que circunstâncias você consideraria transformar o relacionamento entre Cidade e Transporte em uma entidade própria, chamada Rota, e explique como essa mudança impactaria a modelagem do sistema.

Chave de resposta

A transformação do relacionamento entre Cidade e Transporte em uma entidade própria, chamada Rota, seria apropriada quando houvesse a necessidade de rastrear informações específicas sobre cada trajeto de transporte, como distância percorrida, condições climáticas e paradas intermediárias. A entidade Rota permitiria associar detalhes particulares a cada percurso, sem a necessidade de sobrecarregar as entidades Cidade e Transporte com informações específicas de rota.

Devemos entender que, inicialmente, ao modelar o sistema, um relacionamento direto entre Cidade e Transporte pode ser suficiente para representar a movimentação de mercadorias entre diferentes locais. No entanto, ao surgir a necessidade de rastrear informações específicas sobre cada trajeto, como distância, condições climáticas e paradas intermediárias, transformar o relacionamento em uma entidade própria, Rota, torna-se apropriado.

A entidade Rota permitiria associar dados específicos de cada trajeto, evitando a duplicação de informações nas entidades Cidade e Transporte. Essa mudança impactaria positivamente a modelagem do sistema, proporcionando uma estrutura mais organizada e eficiente. Além disso, facilitaria a realização de consultas e relatórios relacionados às informações específicas de cada rota de transporte, contribuindo para uma visão mais completa e detalhada do sistema de logística. Em resumo, a transformação de um relacionamento em uma entidade própria é indicada quando há a necessidade de associar informações específicas e exclusivas a esse relacionamento, melhorando a flexibilidade e a eficiência da modelagem do sistema.

Atributo X Entidade

Atributo X Entidade

Adentramos agora em uma discussão crucial na construção de bancos de dados: a escolha entre utilizar uma entidade ou um atributo para representar informações específicas. Essa decisão estratégica molda a estrutura do sistema e impacta diretamente a eficiência e flexibilidade do modelo.

Vamos em frente!

Entenda, neste vídeo, a escolha entre utilizar uma entidade ou um atributo para representar informações específicas, por meio da análise de circunstâncias em que é mais vantajoso utilizar entidades distintas ou consolidar informações como atributos.

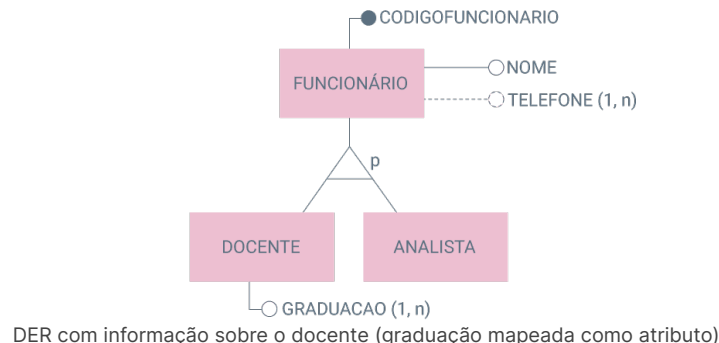


Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Em todos os exemplos desenvolvidos até o momento, caracterizamos entidades e relacionamentos com o auxílio da especificação de atributos. No entanto, em algumas situações, modelar um objeto como atributo pode não ser a melhor alternativa.

No contexto do exemplo acadêmico, é necessário que todo docente tenha pelo menos uma graduação registrada no banco de dados. Veja esse requisito expresso no seguinte modelo.



Precisamos lembrar que a graduação do docente está expressa sob o formato de atributo. A seguir, conheça as diferenças entre as cardinalidades!

Cardinalidade mínima 1

Significa que a informação é obrigatória.

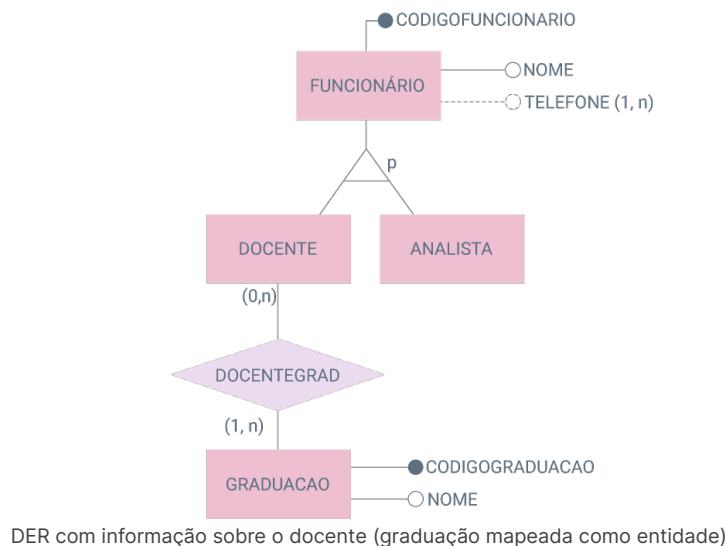


Cardinalidade máxima N

Significa a informação de que é possível registrar no banco de dados diversos cursos associados ao docente.

Se diversos docentes são formados em Ciência da Computação, não faz muito sentido entrar com essa informação no sistema diversas vezes. Mesmo porque, em se tratando de informação textual, daria possibilidade de mais de uma forma de representação para o mesmo item.

Caso, além da informação referente ao nome da graduação, quiséssemos saber o ano em que o docente se formou, estaríamos diante de uma situação em que o mais adequado seria modelar o objeto graduação como uma entidade. Assim, diante do novo cenário, confira o seguinte modelo modificado!



Percebemos que no novo DER:

- O objeto graduação foi mapeado sob o formato de entidade.
- As graduações podem ser cadastradas sem vínculo algum com qualquer docente. Essa conclusão ocorre em função da cardinalidade mínima 0 expressa ao lado da entidade DOCENTE.
- Todo docente necessariamente possui uma graduação. Essa conclusão ocorre em função da cardinalidade mínima 1 expressa ao lado da entidade GRADUACAO.

Finalmente, se estivéssemos interessados em modelar um atributo ANOFIM com objetivo de saber o ano de término da graduação pelo docente, bastaria adicionarmos esse atributo ao relacionamento DOCENTEGRAD.

Uma situação comum durante a construção de um DER é decidir entre modelar um objeto sob o formato de atributo ou especialização. O critério a ser usado na decisão é simples: caso o objeto em questão possua atributo(s) ou mesmo **relacionamento(s)**, usa-se a especialização.

No DER apresentado, a entidade FUNCIONARIO foi especializada. Devemos perceber que a entidade DOCENTE está **relacionada** com a entidade GRADUACAO. Portanto, houve escolha coerente com o critério apresentado.

Atividade 1

Muitas vezes, durante o processo de modelagem temos que decidir se um elemento do minimundo deve ser retratado no DER como entidade, relacionamento ou atributo. A respeito desse aspecto, quando é apropriado utilizar um atributo no lugar de uma entidade em um modelo de banco de dados?

A

Quando o atributo contém informações que podem ser facilmente calculadas a partir de outros dados na base de dados.

B

Sempre que houver a necessidade de armazenar dados complexos e extensos.

C

Quando o atributo representa uma entidade independente, com suas próprias características e relacionamentos.

D

Quando se deseja manter a simplicidade do modelo e o atributo não representa uma entidade distinta.

E

Quando o atributo descreve um relacionamento entre duas entidades.



A alternativa D está correta.

É apropriado utilizar um atributo no lugar de uma entidade para manter a simplicidade do modelo. Em alguns casos, certas informações podem ser tratadas como atributos de uma entidade em vez de criar uma entidade separada. Isso ajuda a simplificar o modelo, especialmente quando a informação não tem uma identidade ou comportamento próprio. As demais opções (a., b., c., e.) não representam as situações específicas em que a escolha por atributo em vez de entidade é apropriada.

Atributo opcional

Entramos agora em uma esfera de decisões imprescindíveis na modelagem de bancos de dados: a determinação de um atributo como obrigatório ou opcional. Essa escolha estratégica molda a rigidez ou a flexibilidade do sistema, influenciando diretamente na integridade dos dados.

Vamos lá!

Compreenda, neste vídeo, a determinação de um atributo como obrigatório ou opcional, a partir da análise de cenários em que a obrigatoriedade é imperativa e aqueles em que a flexibilidade é mais vantajosa.

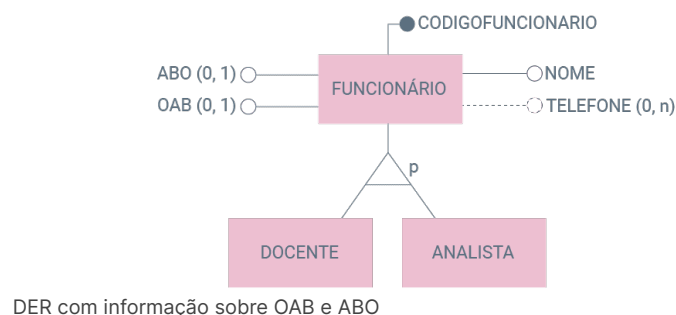


Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

Há situações em que surgem diversos atributos opcionais em uma entidade. Devemos, portanto, estar atentos para perceber se os atributos em questão indicam prováveis entidades especializadas.

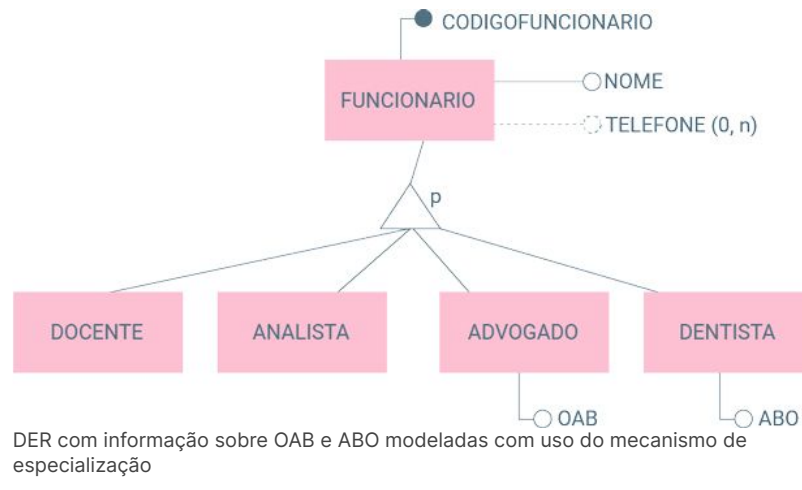
Suponha que, a partir de agora, tenhamos que saber se determinado funcionário tem registro na Ordem dos Advogados do Brasil (OAB) ou na Associação Brasileira de Odontologia (ABO). Assim, diante do novo cenário, considere o seguinte modelo modificado.



DER com informação sobre OAB e ABO

No DER, foram adicionados os atributos OAB e ABO à entidade FUNCIONÁRIO. Devemos perceber que ambos são opcionais, no entanto, não sabemos quais combinações de atributos são válidas. Por exemplo, um funcionário pode ter atributos ABO e OAB?

Se realizarmos uma análise mais criteriosa no modelo, vamos perceber que os atributos ABO e OAB parecem ocultar diferentes categorias: Advogados e odontólogos. Sendo assim, vamos modificar o modelo optando por modelar os atributos ABO e OAB como entidades especializadas de FUNCIONARIO. Logo, confira o modelo, a seguir, que representa as mudanças.



Finalmente, devemos perceber que a estratégia adotada tem a vantagem de modelar a realidade com mais fidelidade, ao mesmo tempo que evitou modelagem de atributos opcionais na entidade FUNCIONARIO.

Atividade 2

Os atributos identificados no minimundo e modelados no DER possuem características como quantidade de valores, obrigatoriedade e unicidade. Na modelagem de banco de dados, o que caracteriza um atributo como sendo opcional?

A

Um atributo é opcional se ele for utilizado apenas em consultas específicas.

B

A opção de tornar um atributo opcional não é aplicável; todos os atributos devem ser obrigatórios.

C

Um atributo é opcional quando seu valor pode ser nulo, ou seja, não é obrigatório para todos os registros.

D

Atributos são sempre opcionais por padrão, e é necessário especificar quando são obrigatórios.

E

A obrigatoriedade de um atributo é determinada pela sua posição na tabela, sendo os primeiros obrigatórios e os últimos opcionais.



A alternativa C está correta.

Quando o valor de um atributo pode ser nulo, ele é caracterizado como opcional. A capacidade de permitir valores nulos em um atributo oferece flexibilidade na modelagem de dados, permitindo que certas informações não sejam necessárias para todos os registros. As demais opções (a., b., d., e.) não refletem corretamente a definição de um atributo opcional em modelagem de banco de dados.

Atributo multivalorado

Outra decisão essencial na modelagem de bancos de dados é a determinação de um atributo como multivalorado ou monovalorado. Vamos analisar os cenários em que a possibilidade de múltiplos valores é vantajosa para o sistema.

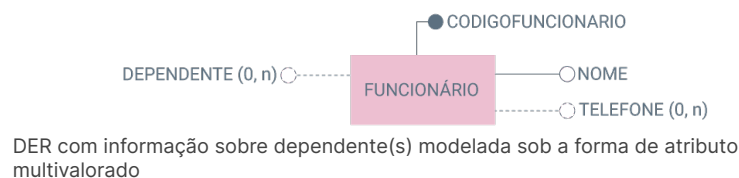
Assista ao vídeo e saiba sobre a determinação de um atributo como multivalorado ou monovalorado, que permite ou restringe a possibilidade de lidar com vários valores diferentes para um mesmo atributo de uma entidade.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

No DER, vamos supor que, a partir de agora, seja necessário modelar que um funcionário pode ter dependente(s). Uma primeira linha de raciocínio para realizar a modelagem desse novo requisito de dados é adicionar um atributo opcional na entidade FUNCIONARIO. Veja!

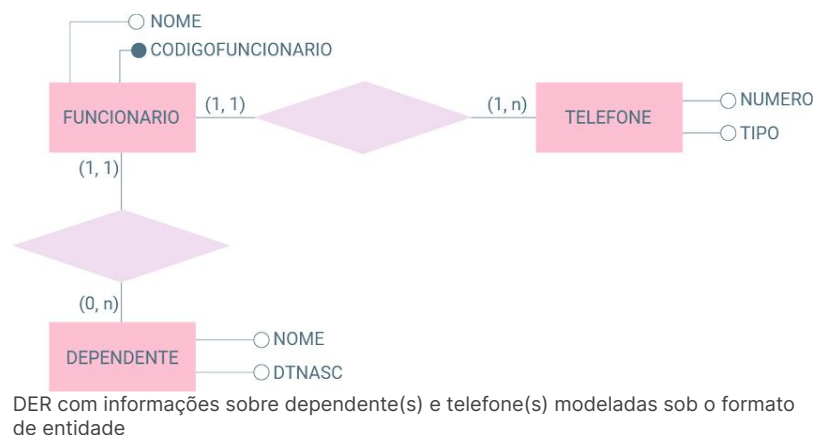


Há dois motivos para evitarmos esse tipo de modelagem:

1. Ao construirmos o modelo físico, vamos perceber que não existe uma implementação direta para atributos multivalorados em um SGBD relacional.
2. Na maioria dos casos, atributos multivalorados escondem atributos e relacionamentos.

É desejável que o banco de dados possa controlar, por exemplo, o nome e a data de nascimento de cada dependente. De forma semelhante, seria igualmente desejável diferenciar o tipo ou a categoria de cada telefone pertencente ao funcionário.

O DER a seguir apresenta tanto a modelagem de dependentes quanto a de telefone sob a perspectiva de entidades relacionadas à entidade FUNCIONARIO:



Podemos observar que, após a decisão de modelar os objetos dependente e telefone como entidades relacionadas à entidade FUNCIONARIO, além de termos eliminado atributos opcionais, conseguimos deixar o modelo mais preciso, legível e com a possibilidade de adicionar novos atributos para as entidades de maneira mais natural.

Atividade 3

Os atributos, sejam das entidades ou dos relacionamentos, têm características que definam sua natureza e determinam aspectos a serem abordados durante sua implementação no banco de dados. Quanto a essas características, o que define um atributo multivalorado?

A

Um atributo é considerado multivalorado quando pode assumir diversos valores para diferentes registros em uma tabela.

B

Atributos multivalorados são aqueles que só podem conter valores booleanos (verdadeiro ou falso).

C

Um atributo é multivalorado quando pode armazenar diferentes tipos de dados, como números e texto, para um mesmo registro.

D

Todos os atributos em um banco de dados são, por padrão, multivalorados, e é necessário especificar quando são monovalorados.

E

Atributos multivalorados são exclusivamente utilizados em relacionamentos muitos-para-muitos entre tabelas.



A alternativa A está correta.

Um atributo multivalorado é caracterizado pelo fato de poder assumir diversos valores para um registro da tabela. Isso significa que, para um único registro, esse atributo pode ter mais de um valor associado. Isso é comumente utilizado quando um dado pode ter várias instâncias associadas a ele. As demais opções (b., c., d., e.) não representam corretamente a definição de um atributo multivalorado na modelagem de banco de dados.

Atributos redundantes e compostos

Agora, analisaremos os atributos redundantes e compostos. Convidamos você a participar ativamente desta reflexão, explorando os *trade-offs* entre a simplificação da estrutura e a preservação da eficiência. Juntos, vamos explorar essas complexidades na busca por uma modelagem de dados que atenda às demandas específicas do ambiente e proporcione um banco de dados coeso e eficaz.

Explore, neste vídeo, os desafios de decidir quando é apropriado utilizar atributos redundantes para otimizar consultas e quando é mais eficiente empregar atributos compostos para manter a consistência e a integridade dos dados.



Conteúdo interativo

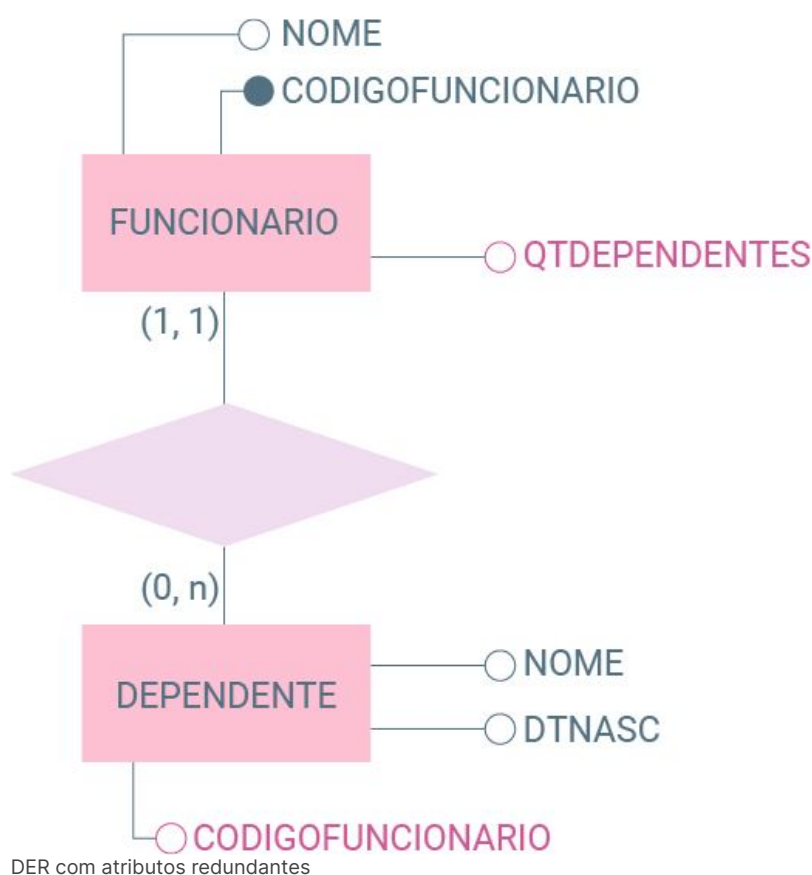
Acesse a versão digital para assistir ao vídeo.

Atributo X Redundância

Atributos redundantes são atributos deriváveis a partir da execução de procedimentos de busca de dados e/ou cálculos sobre o banco de dados (Heuser, 2009).

No exemplo acadêmico, vamos supor que seja necessário sabermos a quantidade de dependentes de cada funcionário. Além disso, precisamos identificar, para cada dependente, o número de matrícula do funcionário responsável.

Confira o uso de atributos redundantes expresso na modelo a seguir, propositalmente em destaque.



DER com atributos redundantes

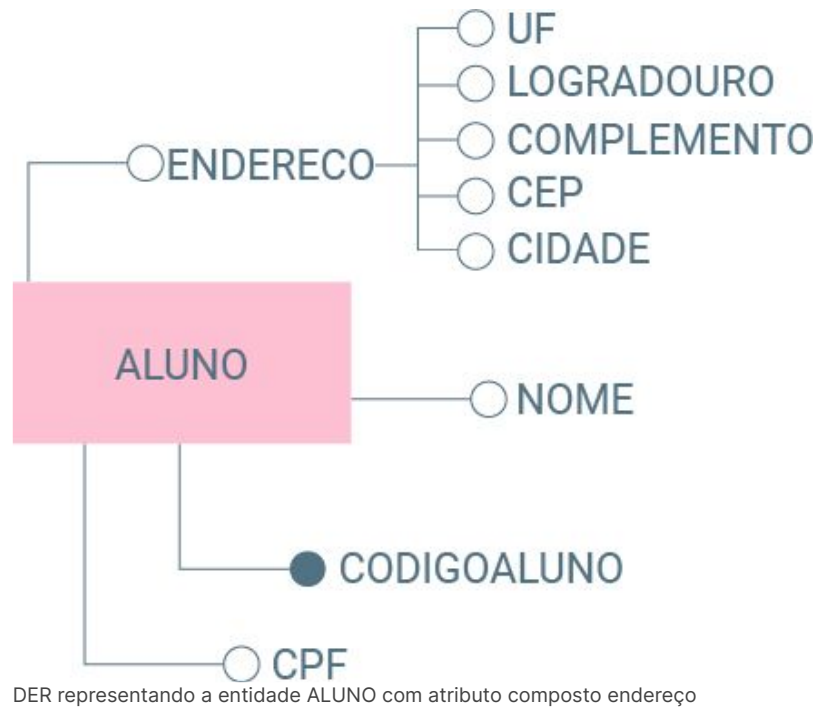
Devemos perceber que não é necessário manter o atributo QTDEPENDENTES na entidade FUNCIONARIO, pois a informação sobre a quantidade de funcionários pode ser obtida a partir de um simples processo de cálculo envolvendo o relacionamento. Também não é necessário manter o atributo CODIGOFUNCIONARIO na entidade DEPENDENTE, pois seu valor pode ser obtido ao acessar a entidade FUNCIONARIO por meio do relacionamento.

Finalmente, atributos redundantes devem ser omitidos do DER, uma vez que esse tipo de modelo não diferencia atributos redundantes dos sem redundância.

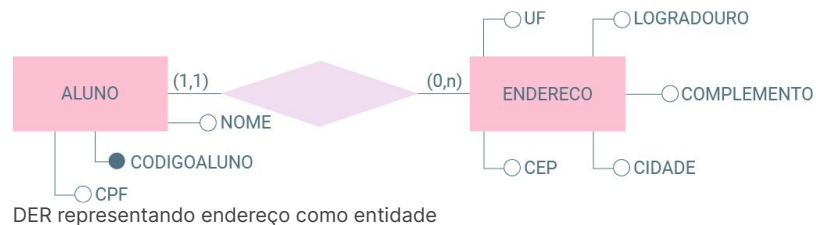
Atributo composto

Pode ser dividido em subpartes ou atributos básicos com significados próprios.

No DER deste estudo, para fins ilustrativos, inicialmente modelamos o endereço do aluno por meio de um atributo composto subdividido em logradouro e complemento. Se adicionarmos mais alguns atributos básicos representativos de outras partes de um endereço, teremos um DER como a seguir.



Como desvantagem, o modelo ficou com um visual bastante denso. Por outro lado, é comum modelarmos o objeto alvo do atributo composto sob o formato de entidade relacionada à entidade principal. Observe essa mudança na imagem a seguir.



Atividade 4

Os atributos de um minimundo necessitam ser representados no DER de modo que mostrem sua característica de unicidade, quantidade de valores, composição etc. Na modelagem de banco de dados, o que caracteriza um atributo composto?

A

Atributos compostos são aqueles que podem conter valores de diferentes tipos, como números e texto, para um mesmo registro.

B

Atributos compostos referem-se exclusivamente a valores calculados a partir de outras informações armazenadas no banco de dados.

C

Um atributo é considerado composto quando é formado por subpartes, cada uma representando uma característica específica do dado.

D

Todos os atributos em um banco de dados são, por padrão, compostos, e é necessário especificar quando são simples.

E

Atributos compostos só são utilizados em tabelas que possuem relacionamentos muitos-para-muitos.



A alternativa C está correta.

Um atributo formado por subpartes que representam características específicas do dado é chamado composto. Em outras palavras, um atributo composto é uma estrutura que consiste em partes menores, e cada parte possui seu próprio significado. Isso é útil quando um dado pode ser subdividido em componentes mais simples. As demais opções (a., b., d., e.) não representam corretamente a definição de um atributo composto na modelagem de banco de dados.

Modelagem de atributos na prática

Agora que já estudamos com mais profundidade os aspectos da modelagem de atributos, veremos um novo estudo de caso mais focado nesse aspecto.

Este é o minimundo: um filme possui código, título, categoria (comédia, suspense etc.). É preciso saber os atores principais de cada filme, sendo que nem todos os filmes têm atores principais. É necessário, ainda, saber o nome real e a idade de cada ator.

Veja, neste vídeo, um estudo de caso sobre filmes, dos quais será necessário saber os atores principais de cada filme, sendo que nem todos os filmes têm atores principais.



Conteúdo interativo

Acesse a versão digital para assistir ao vídeo.

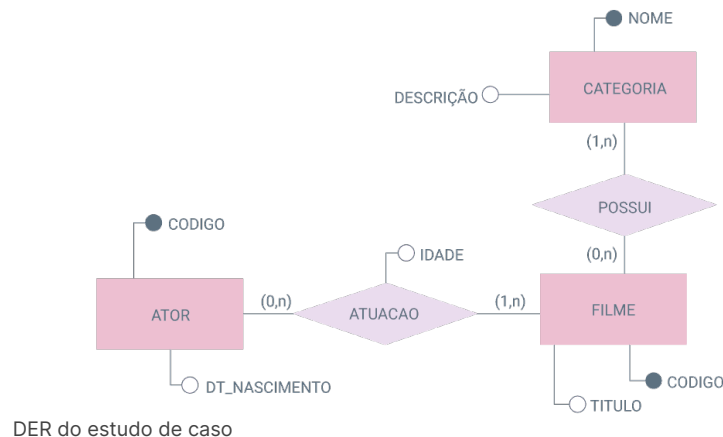
Roteiro de prática

Para realizarmos a modelagem conceitual a partir do mundo real, devemos seguir os seguintes passos:

1. Identificar as entidades existentes.
2. Definir as entidades tipo e representá-las no DER.
3. Identificar as associações entre as entidades, ou seja, os seus relacionamentos.
4. Definir os relacionamentos tipo e representá-los no DER.
5. Estabelecer as cardinalidades dos relacionamentos.
6. Identificar os atributos e suas características.
7. Modelar os atributos e as características de cada um dos atributos do cenário, determinando se esse atributo é:
 - 7.1. Único ou não único.
 - 7.2. Opcional ou obrigatório.
 - 7.3. Simples ou composto.

7.4. Monovalorados ou multivalorado.

Confira o DER de solução!



Atividade 5

Você está envolvido no desenvolvimento de um sistema para uma loja on-line que vende produtos eletrônicos. Inicialmente, você modelou a entidade Fabricante como um atributo na entidade Produto, considerando informações como nome e país de origem do fabricante. Entretanto, ao longo do desenvolvimento, percebeu-se a necessidade de incluir mais detalhes sobre cada fabricante, como histórico de produtos lançados, classificações e avaliações dos clientes para cada fabricante. Nesse cenário, discuta quando seria apropriado transformar o atributo Fabricante em uma entidade própria, e como essa mudança impactaria a estrutura do sistema.

Chave de resposta

A transformação do atributo Fabricante em uma entidade própria seria apropriada quando há a necessidade de rastrear informações mais detalhadas sobre cada fabricante, como histórico de produtos lançados, classificações e avaliações dos clientes. A criação da entidade Fabricante permitiria associar múltiplos produtos a um único fabricante e armazenar informações específicas sobre cada fabricante de maneira organizada.

Inicialmente, ao modelar o sistema, o atributo Fabricante na entidade Produto pode ser suficiente para incluir informações básicas, como nome e país de origem do fabricante. No entanto, ao perceber a necessidade de incluir detalhes mais específicos sobre cada fabricante, como histórico de produtos lançados, classificações e avaliações dos clientes, é apropriado transformar o atributo Fabricante em uma entidade própria.

Ao criar a entidade Fabricante, é possível associar múltiplos produtos a um único fabricante. Cada instância da entidade Fabricante pode conter informações detalhadas sobre o histórico de produtos lançados por aquele fabricante, as classificações e avaliações dos clientes para esse fabricante específico. Essa abordagem melhora a estrutura do sistema, permitindo uma modelagem mais detalhada e organizada das informações relacionadas aos fabricantes.

Essa mudança impacta positivamente a estrutura do sistema ao fornecer uma visão mais completa e flexível das informações relacionadas aos fabricantes. Além disso, facilita a realização de consultas e relatórios mais complexos que envolvem informações específicas de cada fabricante. Em resumo, a transformação de um atributo em uma entidade própria é apropriada quando há a necessidade de incluir

detalhes mais granulares e específicos sobre o atributo em questão, melhorando a flexibilidade e a organização do sistema.

Considerações finais

O que você aprendeu neste conteúdo?

- Etapas de um projeto de banco de dados.
- Elementos do diagrama entidade relacionamento.
- Processo da modelagem entidade relacionamento.
- Técnicas de modelagem de atributos.

Explore +

- Como vimos, um projeto de banco de dados envolve diversas etapas bem definidas. Além disso, a construção de um DER é um processo incremental, sendo possível que o diagrama seja revisado e alterado para atender a novos requisitos de dados. Por isso, é importante você pesquisar o **Guia da Modelagem de Dados: introdução & modelo conceitual**, de Felipe Almeida.
- Pesquise sobre a ferramenta **Vertabelo**, citada no conteúdo.
- Visite o portal **DB-engines** para ver o ranking, atualizado mensalmente, sobre o uso de SGBDs.

Referências

CANDIDO, C. H. **brModelo 3.20**. GitHub, 2019.

ELMASRI, R.; NAVATHE, S. **Sistemas de Banco de Dados**. 7. ed. São Paulo: Pearson, 2019.

HEUSER, C. A. **Projeto de Banco de Dados**. 6. ed. Porto Alegre: Bookman, 2009.