

TEXTO DE APOIO



AULA 1

Tópicos de Banco de Dados

Professor Jamilson Bispo dos Santos



Universidade Presbiteriana
Mackenzie





Sumário



VISÃO ABSTRATA DOS DADOS.....	3
MODELO DE DADOS.....	4
DEFINIÇÃO DO MODELO RELACIONAL.....	5
MODELO ENTIDADE-RELACIONAMENTO.....	8
MAPEAMENTO MER VS. RELACIONAL.....	9
MODELO RELACIONAL.....	10
MODELO FÍSICO.....	11
REFERÊNCIAS BIBLIOGRÁFICAS.....	12

MODELO RELACIONAL

VISÃO ABSTRATA DOS DADOS

Um dos grandes benefícios dos sistemas gerenciadores de banco de dados (BD) é disponibilizar uma visão abstrata dos dados. O sistema é capaz de ocultar as informações de armazenamento e a manutenção dos dados.

A eficiência da recuperação de informações está diretamente relacionada à forma como as estruturas de representação são projetadas (os modelos) e, dada a complexidade, elas devem ser divididas em níveis de abstração.

Os níveis de abstração são:

- **Nível de visão:** possibilita uma visão parcial do banco de dados. Diferentes visões são usadas por diferentes usuários.
- **Nível lógico:** define quais dados serão armazenados e quais são os inter-relacionamentos existentes entre eles. Usado pelos administradores de banco de dados e programadores.
- **Nível físico:** define como os dados estão armazenados e quais são suas estruturas internas (descrição em detalhes das estruturas de dados). Administradores de banco de dados devem ter noções da organização desse nível.

Os bancos de dados sofrem mudança periodicamente, à medida que as informações são inseridas, alteradas ou excluídas. A coleção das informações armazenadas é uma instância do banco de dados. O projeto geral do banco de dados é o esquema. O esquema de banco de dados corresponde às declarações de variável em um programa.

MODELO DE DADOS

Modelos de dados são a base para a construção de um projeto.

Propósitos dos modelos:

- Comunicação
- Categorização
- Descrição
- Especificação
- Investigação
- Desenvolvimento
- Análise

O Modelo de Dados fornece abstração a um BD. São as informações que descrevem a estrutura de uma base de dados. Por estrutura de uma base de dados, entende-se os tipos de dados, os relacionamentos e as restrições aos dados.

Tipos de modelos de dados:

- Hierárquico
- Rede
- Relacional
- Orientado a Objetos

O modelo relacional, atualmente, é o mais utilizado e será discutido com detalhes.

DEFINIÇÃO DO MODELO RELACIONAL

O Modelo Relacional (MR) é um modelo de dados efetivamente usado em aplicações comerciais atualmente. Foi introduzido por E. F. Codd em 1970 no artigo “A Relational Model for Large Shared Data Banks”.

É o modelo que possui a base mais formal entre os modelos de dados, entretanto, é o mais simples e com a estrutura de dados mais uniforme.

Um banco de dados relacional é a coleção de relações ou tabelas de duas dimensões.

Tabela/Relação: Empregos

ID Empregado	Primeiro Nome	Último Nome	E-mail
1	Mariana	Santos	mariana.santos@mackenzie.br
2	Joao	Santos	joao.santos@mackenzie.br

Tabela/Relação: Departamento

ID Departamento	Nome Departamento	ID Empregado Gerente Nome
10	Marketing	1

No exemplo, acima o objetivo é armazenar informações dos funcionários de uma empresa e seus departamentos.

Podemos conceituar da seguinte forma:

- Relação é o conjunto não ordenado de tuplas, não existem tuplas duplicadas.
- Conjunto de relações pode ser visualizada como uma tabela.
- Tuplas é um conjunto ordenado de valores.
- Valores são os atributos.

Codd, em 1985, estabeleceu as 12 regras que determinam o quanto um banco de dados é relacional. Alguns sistemas gerenciadores de banco de dados (SGBD) não fornecem suporte a elas:

1. Regra das informações em tabelas: As informações a serem armazenadas no banco de dados devem ser apresentadas como relações (tabelas formadas por linhas e colunas), e o vínculo de dados entre as tabelas deve ser estabelecido por meio de valores de campos comuns. Isso se aplica tanto aos dados quanto aos metadados (que são descrições dos objetos do banco de dados).
2. Regra de acesso garantido: Para que o usuário possa acessar as informações contidas no banco de dados, o método de referência deve ser o nome da tabela, o valor da chave primária e o nome do campo/coluna.
3. Regra de tratamento sistemático de valores nulos: O SGBD deve ser capaz de tratar valores que não são fornecidos pelos usuários, de maneira que permita a distinção de dados reais. Valores nulos devem ter um tratamento diferente de “valores em branco”.
4. Regra do catálogo relacional ativo: Toda a estrutura do banco de dados (domínios, campos, tabelas, regras de integridade, índices etc.) deve estar disponível em tabelas (também referenciadas como catálogo). Sua manipulação é possível por meio de linguagens específicas. Essas tabelas são, geralmente, manipuladas pelo próprio sistema no momento em que o usuário efetua alterações na estrutura do banco de dados.
5. Regras de atualização de alto-nível: Essa regra diz que o usuário deve ter capacidade de manipular as informações do banco de dados em grupos de registros, ou seja, ser capaz de inserir, alterar e excluir vários registros ao mesmo tempo.
6. Regra de sublinguagem de dados abrangente: Pelo menos uma linguagem deve ser suportada, para que o usuário possa manipular a estrutura do banco de dados (como criação e alteração de tabelas), assim como extrair, inserir, atualizar ou excluir dados, definir restrições de acesso e controle de transações (commit e rollback, por exemplo). Deve ser possível, ainda, a manipulação dos dados por meio de programas de aplicativos.
7. Regra de independência física: Quando for necessária alguma modificação

na forma como os dados estão armazenados fisicamente, nenhuma alteração deve ser necessária nas aplicações que fazem uso do banco de dados, assim como devem permanecer inalterados os mecanismos de consulta e manipulação de dados utilizados pelos usuários finais.

8. Regra de independência lógica: Qualquer alteração efetuada na estrutura do banco de dados, como inclusão ou exclusão de campos de uma tabela ou alteração no relacionamento entre tabelas, não deve afetar o aplicativo que o usa. Da mesma forma, o aplicativo somente deve manipular visões dessas tabelas.
9. Regra de atualização de visões: Uma vez que as visões dos dados de uma ou mais tabelas são, teoricamente, suscetíveis a atualizações, então um aplicativo que faz uso desses dados deve ser capaz de efetuar alterações, exclusões e inclusões. Essas atualizações, no entanto, devem ser repassadas automaticamente às tabelas originais.
10. Regra de independência de integridade: As várias formas de integridade de banco de dados (integridade de entidade, integridade referencial, restrições, obrigatoriedade de valores etc.) precisam ser estabelecidas dentro do catálogo do sistema ou dicionário de dados e ser totalmente independentes da lógica dos aplicativos.
11. Regra de independência de distribuição: Alguns SGBD, notadamente os que seguem o padrão SQL, podem ser distribuídos em diversas plataformas/equipamentos que se encontrem interligados em rede. Essa capacidade de distribuição não pode afetar a funcionalidade do sistema e dos aplicativos que fazem uso do banco de dados.
12. Regra não-subversiva: O sistema deve ser capaz de impedir qualquer usuário ou programador de transgredir mecanismos de segurança, regras de integridade do banco de dados e restrições, utilizando algum recurso de linguagem de baixo nível que eventualmente possa ser oferecido pelo próprio sistema.

MODELO ENTIDADE-RELACIONAMENTO

O modelo entidade-relacionamento (MER) fornece um meio de identificar entidades a serem representadas no banco de dados e verificar como essas entidades são relacionadas.

O MER foi desenvolvido para facilitar o projeto de um banco de dados, permitindo a especificação de um esquema que represente sua estrutura geral.

Uma entidade é algo do mundo real, “coisa” ou “objeto”. É baseada em uma percepção de uma realidade que consiste em uma coleção de objetos.

O MER deve ser intuitivo e representado graficamente por meio das informações modeladas.

Componentes do MER

- Entidades
- Atributos
- Relacionamentos

Entidades

- São uma representação abstrata de um objeto do mundo real.
- Um conjunto de objetos do mundo real que tem as mesmas características é chamada de entidade.

Atributos

- São as propriedades descritas das entidades.
- Um atributo pode ser originário de outras tabelas.

Relacionamentos

- Uma associação entre entidades.

MAPEAMENTO MER VS. RELACIONAL

Os modelos de dados podem ser:

- Modelo Conceitual, uma descrição do banco de dados de forma independente de implementação.
 - Modelo Entidade-Relacionamento
- Modelo Lógico, uma descrição de um banco de dados no nível de abstração visto pelo usuário do SGBD.
 - Modelo Relacional

O mapeamento do MER para o Modelo Relacional é um procedimento executado em seis passos consecutivos que antecedem a criação dos objetos no banco de dados:

1. Mapeamento de todas as entidades “regulares”
2. Mapeamento de todas as entidades “fracas”
3. Mapeamento de todos os relacionamentos binários de cardinalidade 1:1

4. Mapeamento de todos os relacionamentos binários de cardinalidade 1:N
5. Mapeamento de todos os relacionamentos binários de cardinalidade N:N
6. Mapeamento de todos os relacionamentos de grau ≥ 3 (ternários, quaternários etc.)

Exemplos:

MER

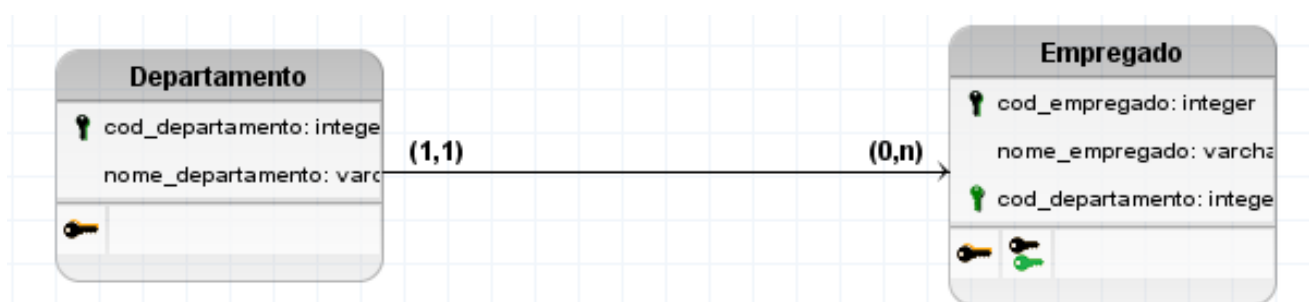
Figura 1 – Modelo Conceitual do relacionamento entre empregado e departamento



Fonte: Elaborada pelo professor

MODELO RELACIONAL

Figura 2 – Modelo Relacional de um relacionamento entre empregado e departamento



Fonte: Elaborada pelo professor

A partir do modelo relacional, é possível gerar um script com os comandos necessários para implementar o modelo físico, que é o de criação das tabelas. Segue exemplo abaixo:

MODELO FÍSICO

```
/* Departamento_Empregado_Logico: */
```

```
CREATE TABLE Empregado (  
  
    cod_empregado INTEGER PRIMARY KEY,  
  
    nome_empregado VARCHAR(30),  
  
    cod_departamento INTEGER  
  
);
```

```
CREATE TABLE Departamento (  
  
    cod_departamento INTEGER PRIMARY KEY,  
  
    nome_departamento VARCHAR(30)  
  
);
```

```
ALTER TABLE Empregado ADD CONSTRAINT FK_Empregado_Departamento  
  
    FOREIGN KEY (cod_departamento)  
  
    REFERENCES Departamento (cod_departamento)  
  
    ON DELETE RESTRICT;
```

Os scripts gerados a partir do modelo relacional geralmente necessitam ser customizados para atender a algumas padronizações existentes na empresa.

REFERÊNCIAS BIBLIOGRÁFICAS

DATE, C. J. Introdução a Sistemas de Banco de Dados. 7. ed. São Paulo: Campus, 2000.

- Capítulo 1: Uma visão geral do gerenciamento de banco de dados.
- Capítulo 2: Arquitetura de sistemas de banco de dados.

ELMASRI, R.; NAVATHE, S. Sistemas de banco de dados. 6. ed. São Paulo: Pearson, 2011 | Biblioteca Virtual Universitária — Pearson.

- Capítulo 7: Modelagem de dados usando o modelo Entidade-Relacionamento (ER) | Páginas 131-148, 151-154.

HEUSER, A. C. Projeto de banco de dados. São Paulo: Artmed, 2009 | Minha Biblioteca.

- Introdução | Páginas 24-26, 29.
- Capítulo 2: Abordagem entidade-relacionamento | Páginas 34-54, 62-64.
- Capítulo 3: Construindo modelos ER | Páginas 72-79, 83-93.

SILBERCHATS, A., KORTH, H. F.; SUDARSHAN, S. Sistemas de Banco de Dados. 3. ed. São Paulo: Makron Books, 1999.

- Capítulo 1, 16 – Apêndices A e B.