

Sistemas de Banco de Dados Projeto, implementação e gerenciamento

Capítulo 2 Modelos de dados

Objetivos

- **Neste capítulo, você aprenderá:**
 - Sobre modelagem de dados e por que esses modelos são importantes
 - Sobre os blocos básicos de construção de modelagem de dados
 - O que são regras de negócio e como influenciam o projeto de bancos de dados
 - Como os principais modelos de dados se desenvolveram
 - Como os modelos de dados podem ser classificados por nível de abstração

Introdução

- Projetistas o programadores e usuários veem os dados de diferentes formas
- Diferentes pontos de vista para o mesmo dado leva a projetos que não refletem a visão da organização
- A modelagem de dados reduz as complexidades do projeto
- Vários graus de abstração de dados ajudam a conciliar opiniões variadas sobre o mesmo dado

Modelagem e Modelos de Dados

- **Modelo de dados:** representação relativamente simples, normalmente gráfica, de estruturas de dados reais mais complexas
- **Modelo:** abstração de um objeto ou evento real de maior complexidade
 - Sua principal função é auxiliar na compreensão das complexidades do ambiente real
- A modelagem de dados é um processo iterativo e progressivo

A Importância dos Modelos de Dados

- Os modelos de dados podem facilitar a interação entre o projetista, o programador de aplicações e o usuário final
- Os usuários finais têm diferentes pontos de vista e necessidades de dados
- A modelagem de dados organiza os dados para diferentes usuários
- A modelagem de dados é uma abstração
 - Não é possível obter os dados necessários a partir do modelo

Blocos Básicos de Construção de Modelos de Dados

- **Entidade:** é algo sobre o qual sejam coletados e armazenados dados
- **Atributo:** é uma característica de uma entidade
- **Relacionamento:** descreve uma associação entre entidades
 - Relacionamento um para muitos (1:M ou 1..*)
 - Relacionamento de muitos para muitos (M:N ou *.*)
 - Relacionamento um para um (1:1 ou 1..1)
- **Restrição:** é uma limitação imposta aos dados

Regras de Negócio

- **Regra de negocio:** descrição breve, precisa e sem ambiguidades de uma política, procedimento ou princípio em uma determinada organização
 - Se aplicam a *qualquer* organização, grande ou pequena que armazene e utilize dados para gerar informações
- As regras de negócio decorrentes de uma descrição detalhada das operações de uma organização ajudam a criar e aplicar ações no interior de seu ambiente organizacional
- Devem ser fornecidas por escrito e atualizadas
- Devem ser de fácil compreensão e amplamente disseminadas
- Descrevem as características principais e particulares dos dados conforme vistos pela empresa

Descobrendo as Regras de Negócio

- Principais fontes de regras de negócio:
 - Gerentes de empresa
 - Elaboradores de políticas
 - Gerentes de departamento
 - Documentações por escrito
 - Manuais de procedimentos, padrões e operações de uma companhia
 - Entrevistas diretas dos usuários finais

Descobrendo as Regras de Negócio (cont.)

- Ajudam a padronizar a visualização dos dados de uma empresa
- Podem constituir uma ferramenta de comunicação entre os usuários e os projetistas
- Permitem que o projetista compreenda a natureza, o papel e o escopo dos dados
- Permitem que o projetista compreenda os processos comerciais
- Permitem que o projetista desenvolva regras e restrições adequadas de participações em relacionamentos, e crie um modelo de dados preciso

Traduzindo Regras de Negócio em Componentes de Modelos de Dados

- Geralmente, um substantivo em uma regra de negócio será traduzido como entidade no modelo
- Um será traduzido como um relacionamento entre entidades
- Relacionamentos são bidirecionais
- Para identificar adequadamente o tipo de relacionamento, deve-se fazer duas perguntas:
 - Quantas instâncias de B são relacionadas a uma instância de A?
 - Quantas instâncias de A são relacionadas a uma instância de B?

TABELA 2.1 Evolução dos principais modelos de dados

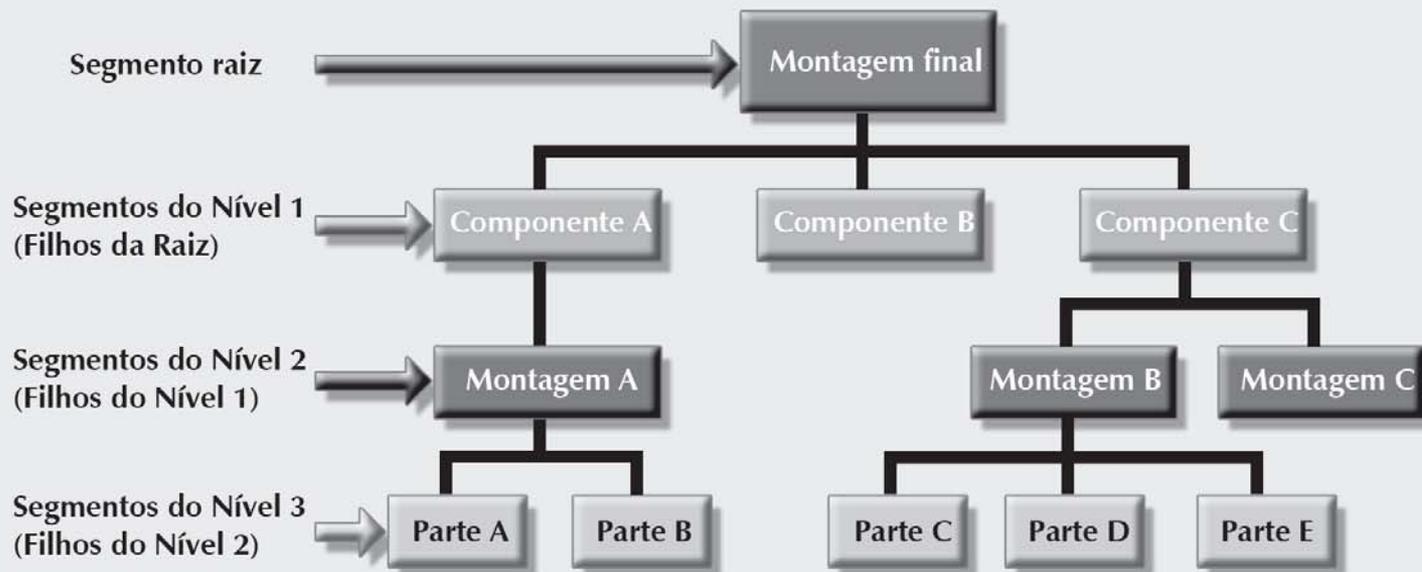
GERAÇÃO	ÉPOCA	MODELO	EXEMPLOS	COMENTÁRIOS
Primeira	década de 1960 e 1970	Sistema de arquivos	VMS/VSAM	Utilizado principalmente em sistemas de mainframe da IBM Gerenciamento de registros, sem relacionamentos
Segunda	década de 1970	Modelo de dados hierárquico e em rede	IMS ADABAS IDS-II	Primeiros sistemas de bancos de dados Acesso navegacional
Terceira	De meados da década de 1970 até o presente	Modelo de dados relacional	DB2 Oracle MS SQL Server MySQL	Simplicidade conceitual Modelagem entidade-relacionamento (ER) e suporte a modelagem relacional de dados
Quarta	De meados da década de 1980 até o presente	Orientado a objetos Relacional estendido	Versant FastObjects.Net Objectivity/DB DB/2 UDB Oracle 10g	Suporte a dados complexos Produtos relacionais estendidos com suporte a warehouse de dados e objetos Bancos de dados na web tornam-se comuns
Próxima geração	Do presente ao futuro	XML	dbXML Tamino DB2 UDB Oracle 10g MS SQL Server	Organização e gerenciamento de dados não estruturados Modelos relacionais e de objetos adicionam suporte a documentos em XML

Modelo Hierárquico

- Foi desenvolvido na década de 1960 para gerenciar grandes quantidades de dados para projetos complexos de fabricação
- Sua estrutura lógica básica é representada por uma estrutura de árvore “de cima para baixo”
- Essa estrutura hierárquica contém níveis ou segmentos:
 - Um **segmento** é equivalente ao tipo de registro em um sistema de arquivos
 - Representa um conjunto de relacionamentos um para muitos (1:M) entre os segmentos

**FIGURA
2.1**

Estrutura hierárquica



Modelo Hierárquico (cont.)

- Muitos recursos do modelo hierárquico constituíram o fundamento dos modelos de dados atuais
- Limitações:
 - Difícil de implementar e de gerenciar
 - Não dispõe de independência estrutural
 - Muitos relacionamentos de dados comuns não se adequam à forma 1:M
 - Não tem padrões de implementação

Modelo em Rede

- Criado para representar relacionamentos de dados complexos com mais eficiência
 - Melhora o desempenho dos bancos de dados
 - Impõe um padrão dos bancos de dados
- **CODASYL** (*Conference on Data Systems Languages*, ou seja, Conferência sobre Linguagens de Sistemas de Dados) criou o **DBTG** (*Database Task Group*, o Grupo de Trabalho sobre Bancos de Dados)
- DBTG foi encarregado de definir especificações-padrão para promover um ambiente que facilitasse a criação de bancos e a manipulação de dados

Modelo em Rede (cont.)

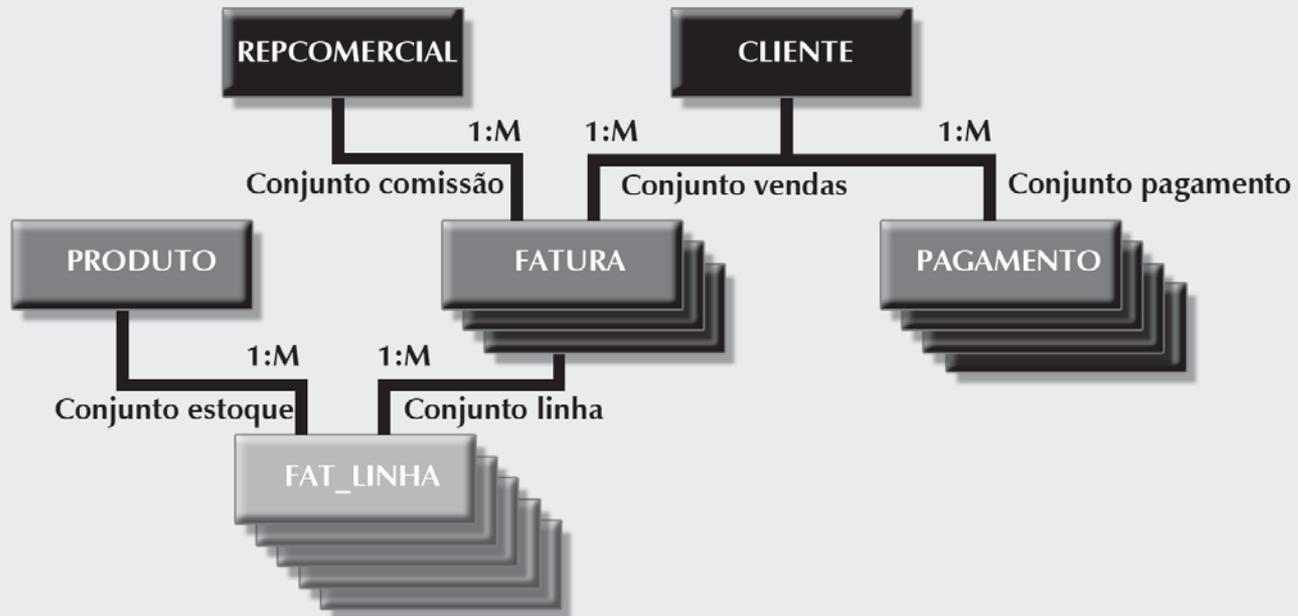
- Esquema
 - Constituía uma organização conceitual do banco como um todo, conforme visto por seu administrador
- Subesquema
 - Parte do banco de dados “vista” pelos aplicativos
- **Linguagem de gerenciamento de dados (DML, de *Data Management Language*)**
 - Define o ambiente em que os dados podem ser gerenciados

Modelo em Rede (cont.)

- Diferente do hierárquico, o modelo em rede permite que um registro tenha mais de um pai
- Coleção de registros em relacionamentos 1:M
- O conjunto é composto de, pelo menos, dois tipos de registros:
 - Proprietário
 - Membro

FIGURA 2.2

Modelo de dados de rede



Modelo em Rede (cont.)

- Desvantagens do modelo em rede:
 - Tedioso
 - A falta de um recurso de consulta *ad hoc* colocava grande pressão para que os programadores gerassem o código necessário para produzir relatórios simples
 - Qualquer alteração estrutural ainda poderia devastar todos os aplicativos que obtinham dados do banco

Modelo Relacional

- Apresentado em 1970 por E. F. Codd (da IBM)
- Tabela (relação)
 - Matriz composta da intersecção de linhas e colunas
 - Cada linha de uma relação é chamada **Tupla**
- Em 1970, o trabalho de Codd foi considerado inovador, porém inviável
- A simplicidade conceitual do modelo relacional era conseguida à custa da sobrecarga do computador

Modelo Relacional (cont.)

- **Sistema de gerenciamento de banco de dados relacionais (SGBDR)**
 - Executa as mesmas funções básicas fornecidas pelos sistemas de SGBD hierárquico e de rede
 - Oculta do usuário as complexidades do modelo relacional
- **Diagrama relacional** é a representação gráfica das entidades, dos atributos e dos relacionamentos de um banco de dados relacional
- Uma tabela relacional armazena um conjunto de entidades relacionadas

FIGURA 2.3

Ligação de tabelas relacionais

Nome da tabela: CORRETOR (seis primeiros atributos) Nome do banco de dados: Ch02_InsureCo

AGENT_CODE	AGENT_LNAME	AGENT_FNAME	AGENT_INITIAL	AGENT_AREACODE	AGENT_PHONE
501	Alby	Alex	B	713	228-1249
502	Hahn	Leah	F	615	882-1244
503	Okon	John	T	615	123-5589

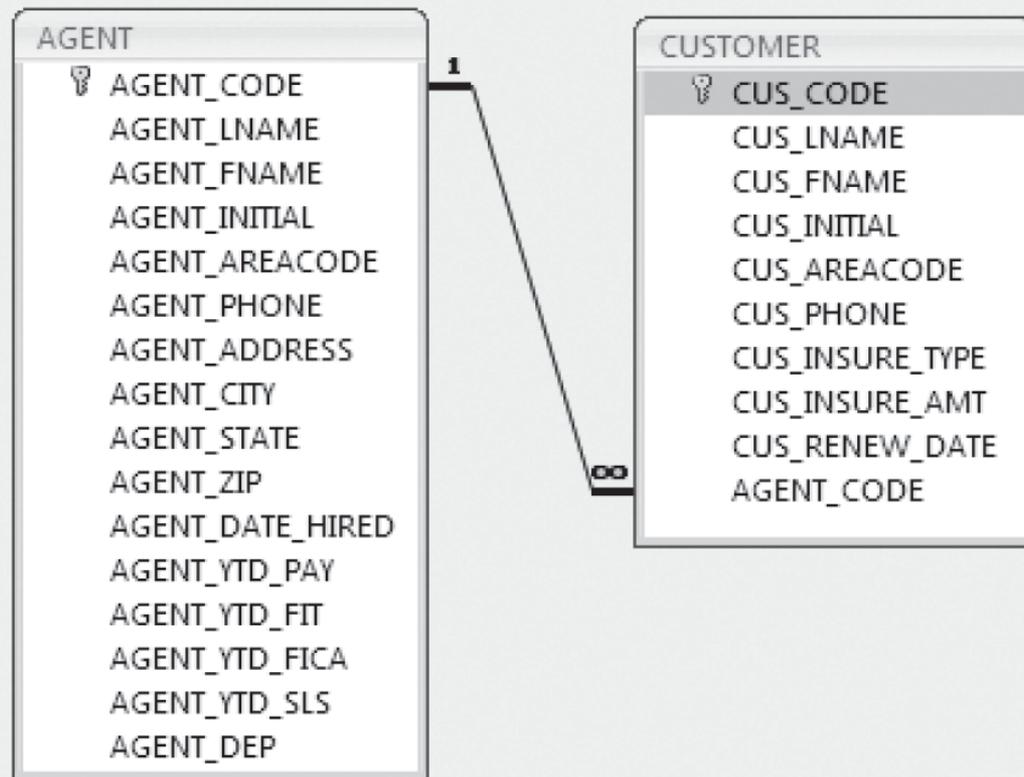
Ligação por meio de AGENT_CODE

Nome da tabela: CLIENTE

CUS_CODE	CUS_LNAME	CUS_FNAME	CUS_INITIAL	CUS_AREACODE	CUS_PHONE	CUS_INSURE_TYPE	CUS_INSURE_AMT	CUS_RENEW_DATE	AGENT_CODE
10010	Ramas	Alfred	A	615	844-2573	T1	100.00	05-Apr-2008	502
10011	Dunne	Leona	K	713	894-1238	T1	250.00	16-Jun-2008	501
10012	Smith	Kathy	vW	615	894-2285	S2	150.00	29-Jan-2009	502
10013	Olowski	Paul	F	615	894-2180	S1	300.00	14-Oct-2008	502
10014	Orlando	Myron		615	222-1672	T1	100.00	28-Dec-2008	501
10015	O'Brian	Amy	B	713	442-3381	T2	850.00	22-Sep-2008	503
10016	Brown	James	G	615	297-1228	S1	120.00	25-Mar-2009	502
10017	vWilliams	George		615	290-2556	S1	250.00	17-Jul-2008	503
10018	Farriss	Anne	G	713	382-7185	T2	100.00	03-Dec-2008	501
10019	Smith	Olette	K	615	297-3809	S2	500.00	14-Mar-2009	503

FIGURA 2.4

Diagrama relacional



Modelo Relacional (cont.)

- SQL (*Structured Query Language*, ou seja, Linguagem Estruturada de Consulta) envolve três partes:
 - Interface do usuário final
 - Permite que o usuário final interaja com os dados
 - Conjunto de tabelas armazenadas no banco de dados
 - Cada tabela é independente uma da outra
 - As linhas de tabelas diferentes são relacionadas com base em valores comuns de atributos comuns
 - Mecanismo de SQL
 - Executa todas as consultas ou solicitações de dados

Modelo Entidade-relacionamento

- Padrão amplamente aceito para a modelagem de dados
- Apresentado por Peter Chen em 1976
- Representação gráfica de entidades e de seus relacionamentos em uma estrutura de banco de dados
- **Diagrama entidade-relacionamento (DER)**
 - Utiliza representações gráficas para modelar os componentes do banco de dados
 - Uma entidade é mapeada para uma tabela relacional

Modelo Entidade-relacionamento (cont.)

- **Instância de entidade (ocorrência de entidade):** cada linha da tabela
- Cada entidade é definida como um conjunto de *atributos* que descrevem suas características particulares
- A **conectividade** identifica os tipos de relacionamento
- Relacionamentos que utilizam as notações de Chen:
 - São representados por um losango
 - O seu nome é escrito dentro do losango
- A notação pé de galinha é utilizada como padrão de projeto neste livro

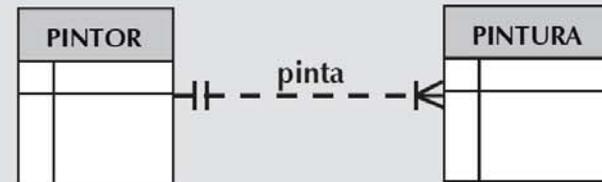
FIGURA 2.5

Notações de Chen e Pés de Galinha (Crow's Foot)

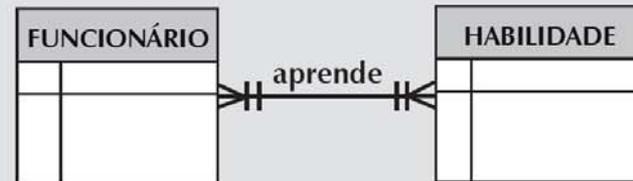
Notação de Chen

Notação Pé de Galinha

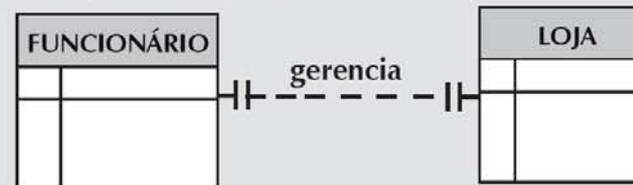
Relacionamento um para muitos (1:M): um PINTOR pode pintar várias PINTURAS; cada PINTURA é criada por apenas um PINTOR.



Relacionamento muitos para muitos (M:N): um FUNCIONÁRIO pode aprender várias HABILIDADES; cada HABILIDADE pode ser aprendida por vários FUNCIONÁRIOS.



Relacionamento um para um (1:1): um FUNCIONÁRIO gerencia uma LOJA; cada LOJA é gerenciada por um FUNCIONÁRIO.



Modelo Orientado a Objetos (OO)

- Dados e relacionamentos são contidos em uma única estrutura conhecida como **objeto**
- Modelo de dados orientado a objetos (**MDOO**), é a base para o sistema de gerenciamento de banco de dados orientados a objetos (**SGBDOO**)
 - Modelo de dados semântico
- Objetos contém operações
- Objetos são autossuficientes: um bloco básico de construção de estruturas autônomas
- Um objeto é uma abstração de uma entidade real

Modelo Orientado a Objetos (OO) (cont.)

- Os atributos descrevem as propriedades de um objeto
- Os objetos que compartilham características similares são agrupados em classes
- As classes organizam-se em uma hierarquia de classes
- **Heranca** é a capacidade de um objeto, no interior da hierarquia de classe, herdar os atributos e métodos das classes superiores
- A **UML** (*Unified Modeling Language*, ou seja, Linguagem de Modelagem Unificada) tem sua base em conceitos de OO que descreve um conjunto de diagramas e símbolos
 - Podem ser utilizados para modelar graficamente um sistema

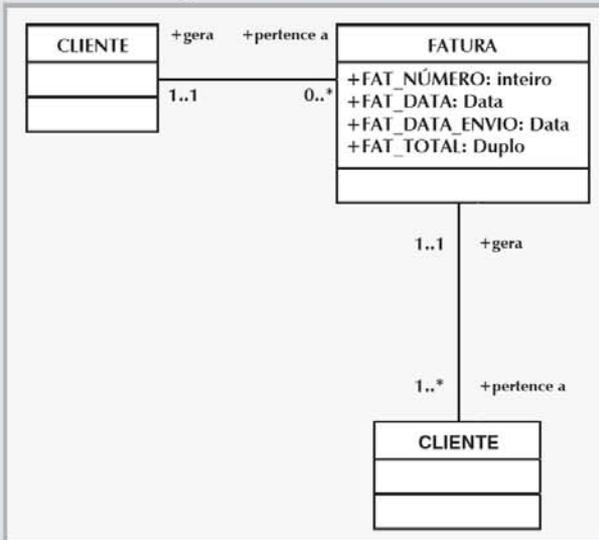
FIGURA 2.6

Comparação dos modelos de OO, UML e ER

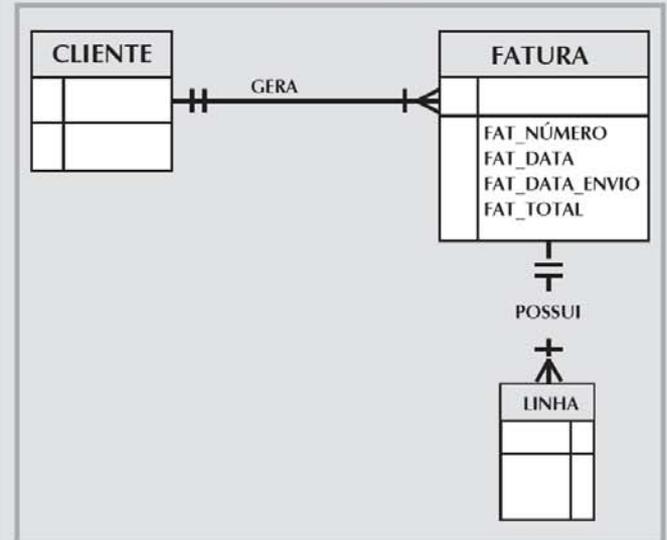
Representação de objetos



Diagrama de classes em UML



Modelo ER



Convergência de Modelos de Dados

- Modelo de dados relacionais estendido (ERDM)
 - Modelo de dados semântico desenvolvido em resposta à complexidade crescente das aplicações
 - Inclui alguns dos melhores recursos do modelo OO
 - É geralmente descrito como um **sistema de gerenciamento de bancos de dados relacionais/objeto (SGBDR/O)**
 - Destina-se principalmente a aplicações comerciais

Modelos de Banco de Dados e Internet

- A internet alterou drasticamente o papel e o escopo do mercado de bancos de dados
- O foco na internet torna os modelos de dados subjacentes menos importantes para o usuário final
- Com o domínio da World Wide Web, ocorre uma necessidade crescente de gerenciamento de dados não estruturados
- Os bancos de dados atuais dão suporte a tecnologias da internet, como XML
 - A importância do suporte a XML não pode ser subestimada, pois é também o protocolo padrão para a troca de dados entre diferentes sistemas e serviços com base na internet

FIGURA 2.7 Desenvolvimento dos modelos de dados

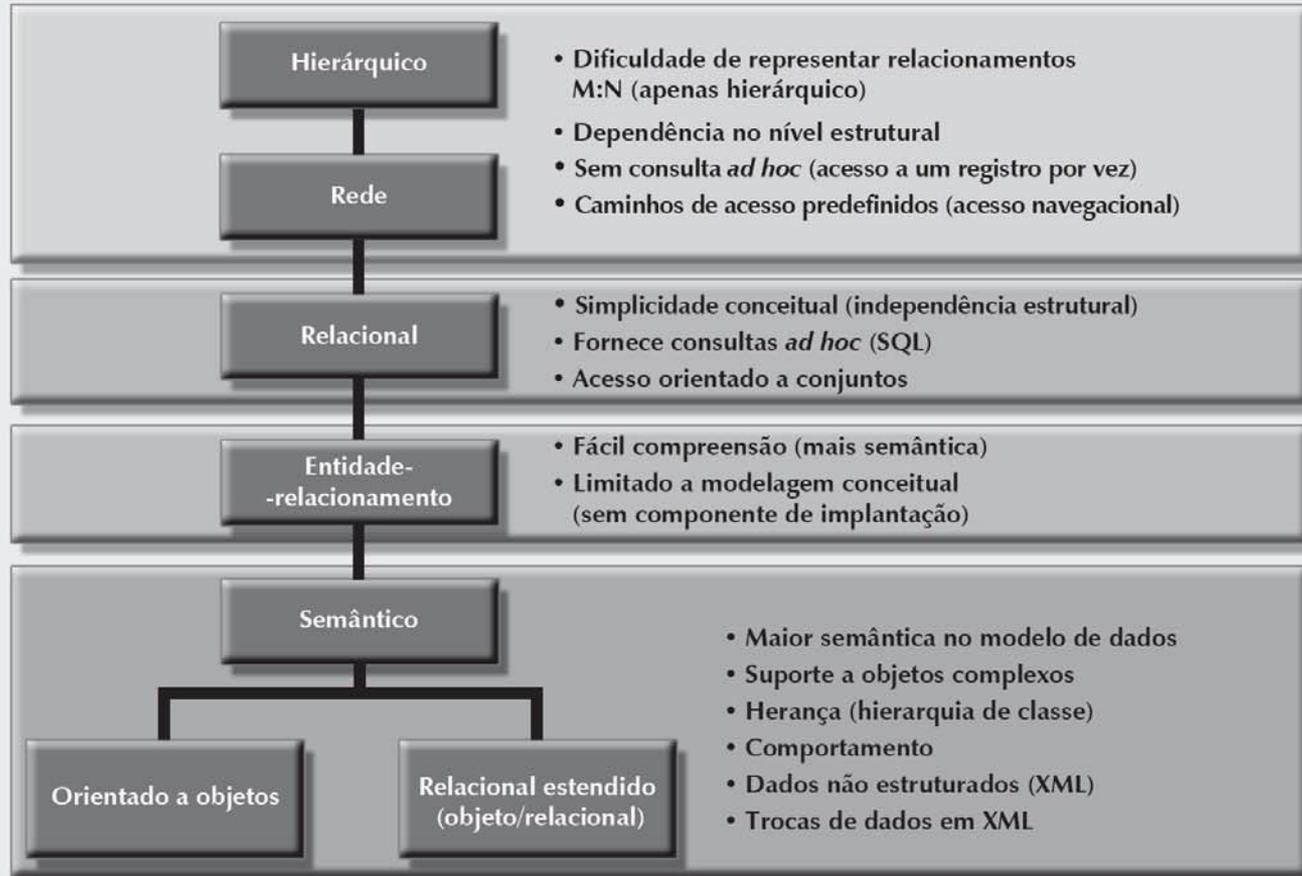
Semântica do modelo de dados

Comentários

menor



maior



Modelos de Dados: um resumo

- Características comuns:
 - Devem mostrar algum grau de simplicidade conceitual sem comprometer a integridade semântica do banco de dados
 - Devem representar a realidade do modo mais preciso possível
 - A representação das transformações reais deve estar em conformidade com as características de consistência e integridade de qualquer modelo
- Cada novo modelo de dados supera uma falha dos modelos anteriores
- Alguns deles são mais adequados do que outros para certas tarefas

TABELA 2.3 Comparação da terminologia básica dos modelos de dados

REALIDADE	EXEMPLO	PROCESSAMENTO DE ARQUIVOS	MODELO HIERÁRQUICO	MODELO EM REDE	MODELO RELACIONAL	MODELO DE ER	MODELO OO
Grupo de fornecedores	Armário arquivo de fornecedores	Arquivo	Tipo de segmento	Tipo de registro	Tabela	Conjunto entidade	Classe
Um fornecedor específico	Global Supplies	Registro	Ocorrência de segmento	Registro atual	Linha (Tupla)	Ocorrência de entidade	Instância do objeto
Nome do contato	Johnny Ventura	Campo	Campo de segmento	Campo de registro	Atributo de tabela	Atributo de entidade	Atributo de objeto
Identificador do fornecedor	G12987	Índice	Campo de sequência	Chave de registro	Chave	Identificador de entidade	Identificador de objeto

Graus de Abstração de Dados

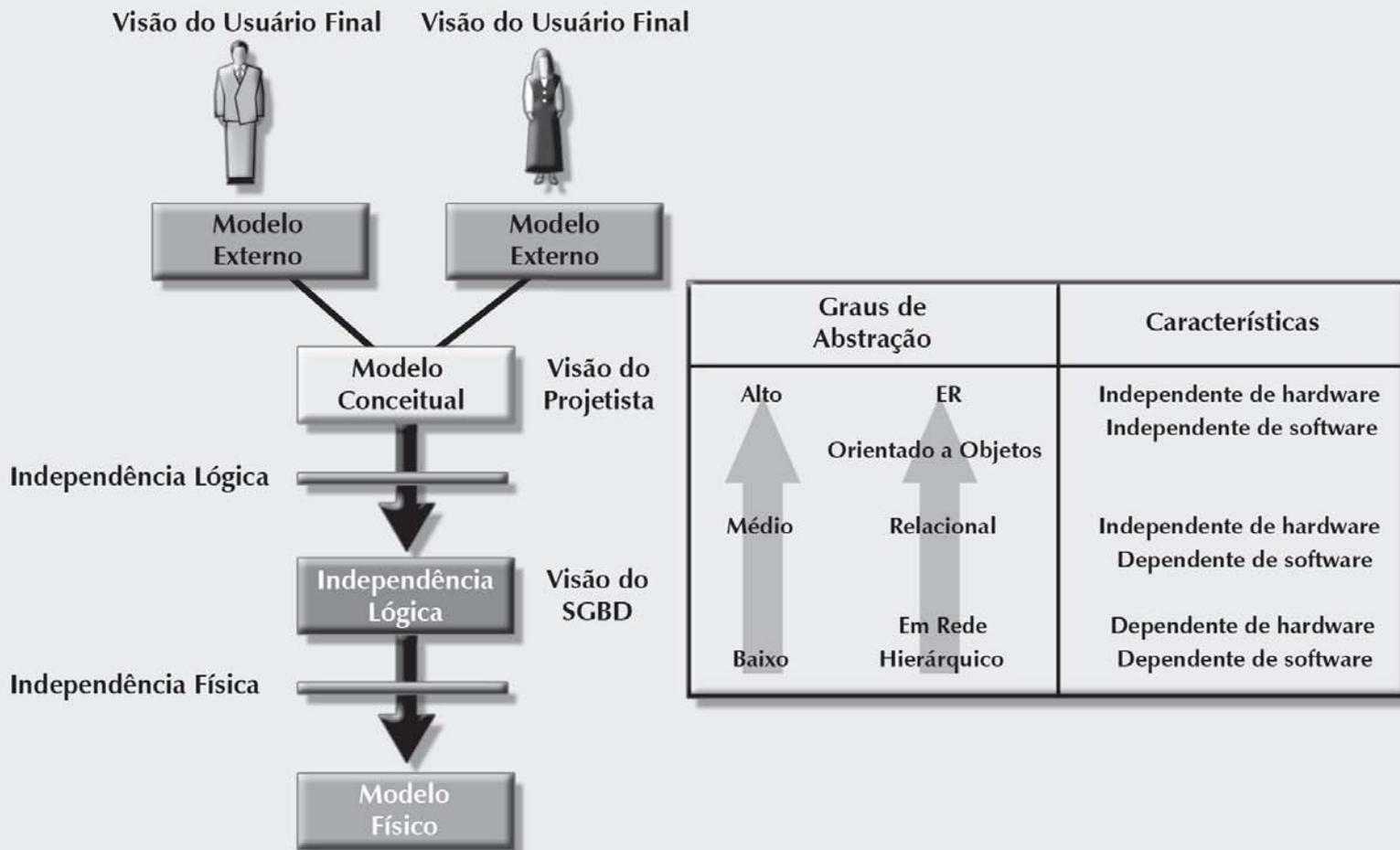
- Um projetista começa com uma visão abstrata do ambiente geral de dados e acrescenta detalhes conforme o projeto se aproxima da implementação
- Comitê de Planejamento e Exigência de Padrões (SPARC)
 - Definiu uma estrutura de modelagem com base em graus de abstração de dados:
 - Externo
 - Conceitual
 - Interno

Modelo Externo

- É a perspectiva dos usuários finais do ambiente de dados
- Os diagramas ER serão utilizados para representar as visões externas
- A representação específica de uma visão externa é conhecida como **esquema externo**
 - Entidades
 - Relacionamentos
 - Processos
 - Restrições

FIGURA 2.8

Níveis de abstração de dados



Modelo Externo (cont.)

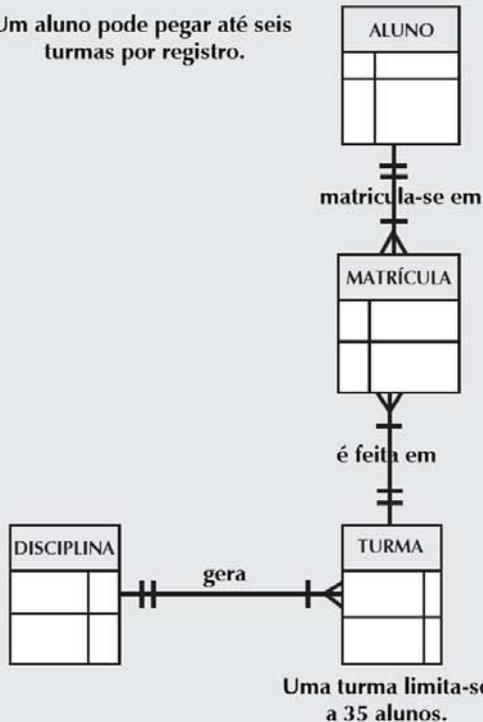
- Facilita a identificação de dados específicos necessários para dar suporte às operações de cada unidade comercial
- Facilita o trabalho do projetista, fornecendo informações sobre a adequação do modelo
- Ajuda a assegurar restrições de *segurança* no projeto do banco de dados
- Torna muito mais simples o desenvolvimento de aplicativos

FIGURA 2.9

Modelos externos da Tiny College

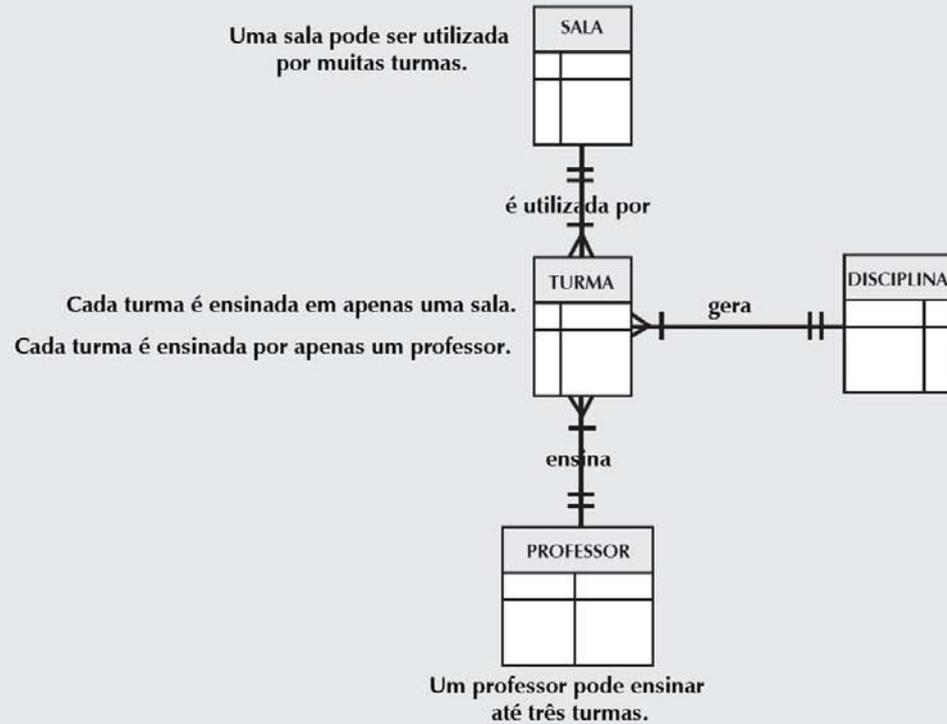
Registro de Alunos

Um aluno pode pegar até seis turmas por registro.



Agendamento de Turmas

Uma sala pode ser utilizada por muitas turmas.

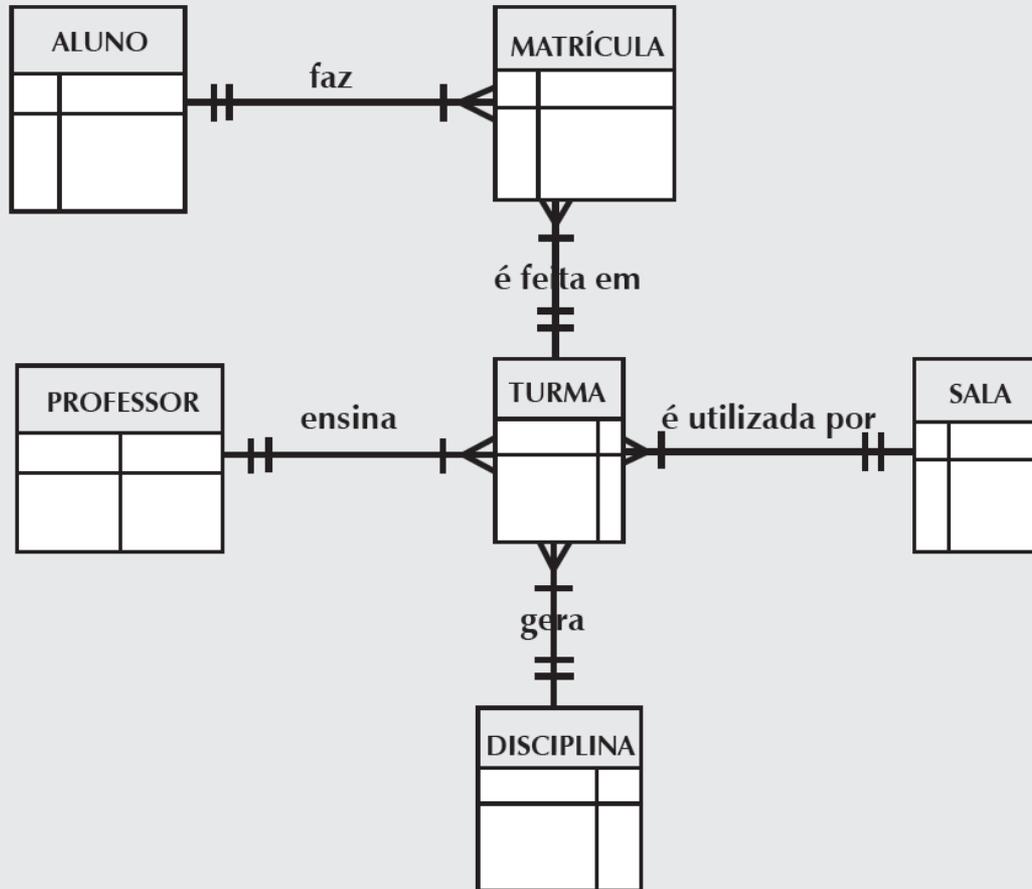


Modelo Conceitual

- Representa uma visão global do banco de dados inteiro conforme visto pela organização como um todo
- Integra todas as visões externas (entidades, relacionamentos, restrições e processos) em uma única visão global de todos os dados da empresa – **esquema conceitual**
- O modelo conceitual mais utilizado é o de ER
- O DER é utilizado para *representar* graficamente o esquema conceitual

**FIGURA
2.10**

**Modelo conceitual para
a Tiny College**



Modelo Conceitual (cont.)

- Fornece uma visão de cima (nível macro) compreendida de modo relativamente fácil sobre o ambiente de dados
- É independente em relação tanto a software como a hardware
 - Não é dependente do software do SGBD utilizado para implantá-lo
 - Não depende do hardware utilizado em sua implantação
 - Alterações de hardware ou do software do SGBD não terão efeito sobre o projeto de banco de dados no nível conceitual

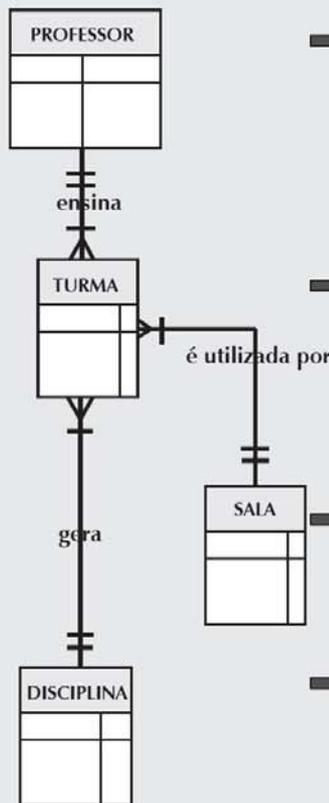
Modelo Interno

- É a representação do banco de dados conforme “visto” pelo SGBD
 - Mapeia o modelo conceitual para as estruturas do modelo relacional
- O **esquema interno** constitui uma representação específica de um modelo interno
- O modelo interno depende do software específico do banco de dados
 - Uma alteração no software de SGBD exige que o modelo interno seja alterado
- **Independência lógica:** quando é possível alterar o modelo interno sem afetar o modelo conceitual

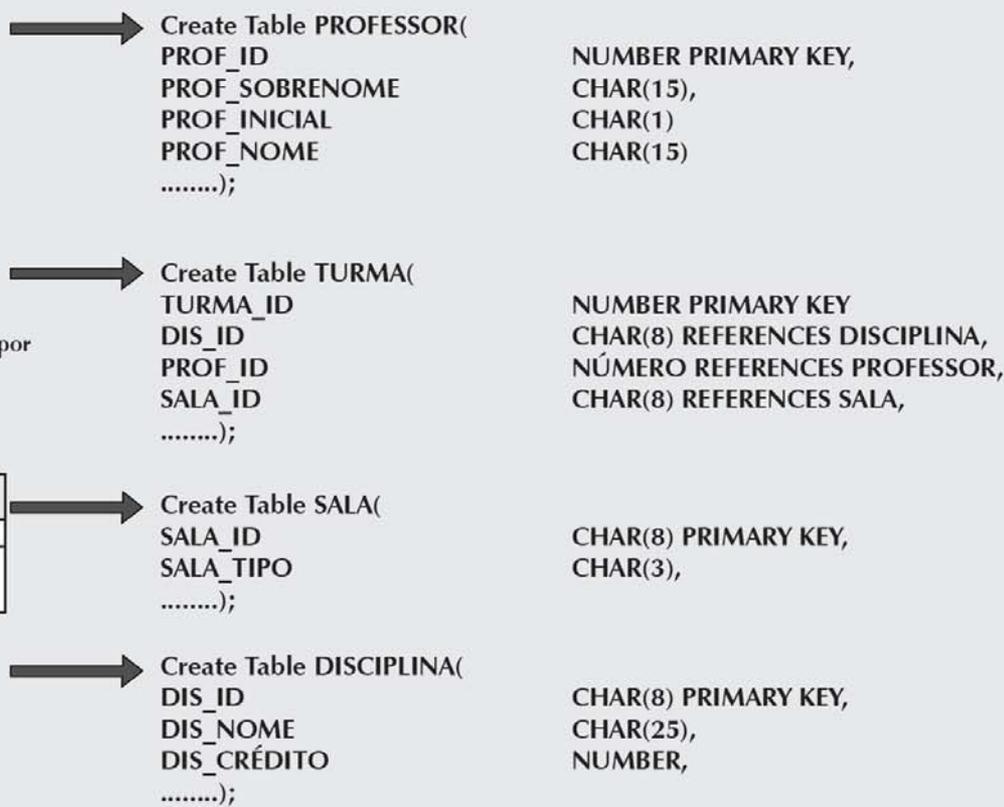
FIGURA 2.11

Modelo interno para a Tiny College

MODELO CONCEITUAL



MODELO INTERNO



Modelo Físico

- Opera nos níveis mais baixos de abstração
 - Descreve o modo como os dados são salvos em meios de armazenamento como discos e fitas
- Exige a definição tanto dos dispositivos de armazenamento físico como dos métodos de acesso
- O modelo relacional atualmente dominante é direcionado amplamente para o nível lógico, não para o físico
- **Independência física:** quando é possível alterar o modelo físico sem afetar o modelo interno, tem-se

TABELA 2.4 Níveis de abstração de dados

MODELO	GRAU DE ABSTRAÇÃO	FOCO	INDEPENDÊNCIA DE:
Externo	Alto	Visões dos usuários finais	Hardware e software
Conceitual		Visão global dos dados (independente do modelo do banco de dados)	Hardware e software
Interno		Modelo específico de banco de dados	Hardware
Físico	Baixo	Métodos de armazenamento e acesso	Nem hardware nem software

Resumo

- Um modelo de dados é uma abstração de um ambiente de dados real e complexo.
- Os componentes básicos de modelagem de dados são:
 - Entidades
 - Atributos
 - Relacionamentos
 - Restrições
- As regras de negócio são utilizadas para identificar e definir os componentes básicos de modelagem em um ambiente específico real

Resumo (cont.)

- Modelo de dados hierárquicos
 - Representa um conjunto de relacionamentos um para muitos (1:M) entre um pai e seus segmentos filhos
- O modelo em rede
 - Utiliza conjuntos para representar relacionamentos 1:M entre tipos de registros
- Modelo relacional
 - Padrão de implementação atual de bancos de dados
 - (ER) é uma ferramenta gráfica popular para a modelagem
 - Complementa o modelo relacional

Resumo (cont.)

- O modelo de dados orientado a objetos (MDOO) utiliza objetos como uma estrutura de modelagem básica
- O modelo relacional adotou muitas extensões orientadas a objetos (OO), tornando-se o modelo de dados relacionais estendido (ERDM)
- Os modelos de dados OO normalmente são representados por diagramas de classe em UML (*Unified Modeling Language*)
- As exigências de modelagem de dados são uma função de diferentes visões de dados (global *versus* local) e do nível de abstração de dados
 - Três níveis de abstração: externo, conceitual e interno