
Banco de Dados

Modelo de Dados

Profa. Fabrícia Damando

fdamando@yahoo.com.br

Introdução

- Os SGBDs evoluíram desses sistemas de arquivos de armazenamento em disco, criando novas estruturas de dados com o objetivo de armazenar informações.
- Com o tempo, os SGBD's passaram a utilizar diferentes formas de representação, ou modelos de dados, para descrever a estrutura das informações contidas em seus bancos de dados.
- Atualmente, os seguintes modelos de dados são normalmente utilizados pelos SGBD's:
 - modelo hierárquico,
 - modelo em redes,
 - modelo relacional (amplamente usado)
 - modelo orientado a objetos.

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

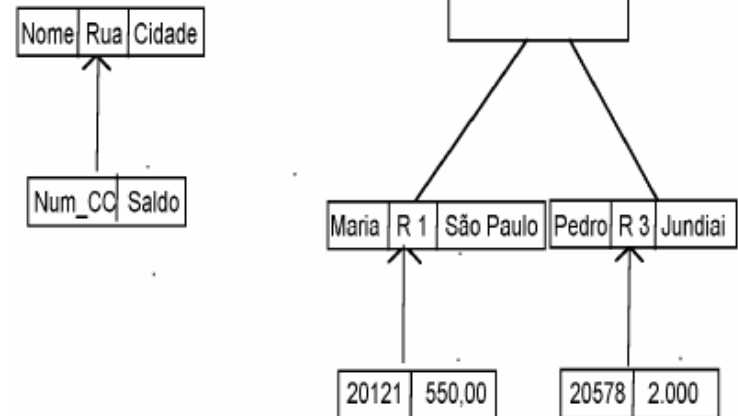
Modelo Hierárquico

- O modelo hierárquico foi o **primeiro** a ser reconhecido como um modelo de dados - década de 1960.
- Possibilitaram a exploração de estrutura de endereçamento físico para viabilizar a **representação hierárquica das informações**
- Os **dados** são **estruturados** em **hierarquias** ou **árvores**.
- Os **nós** das hierarquias contêm **registros**
- Cada registro é uma coleção de campos (atributos)- contendo apenas uma informação.
- O registro da hierarquia que precede a outros é o registro-pai, os outros são chamados de registros-filhos

Modelo Hierárquico

Se lembram de
estruturas de
dados?
Árvores???

- Os dados podem ser acessados segundo uma seqüência hierárquica com uma navegação do **topo** para as **folhas** e da esquerda para a direita.



O sistema comercial mais divulgado no modelo hierárquico foi o *Information Management System da IBM Corp(IMS)*.

Modelo Hierárquico

■ Limitações

- ❑ Difícil implementar e gerenciar
- ❑ Muitos relacionamentos de dados são comuns e não se adequam à forma 1:M
- ❑ Não tem padrão de implementação

Modelo de Rede

- No modelo de rede os registros são organizados em grafos onde aparece um único tipo de associação (set) que define uma relação 1:N entre 2 tipos de registros: proprietário e membro
- O modelo em rede possibilita acesso a qualquer nó da rede sem passar pela raiz.

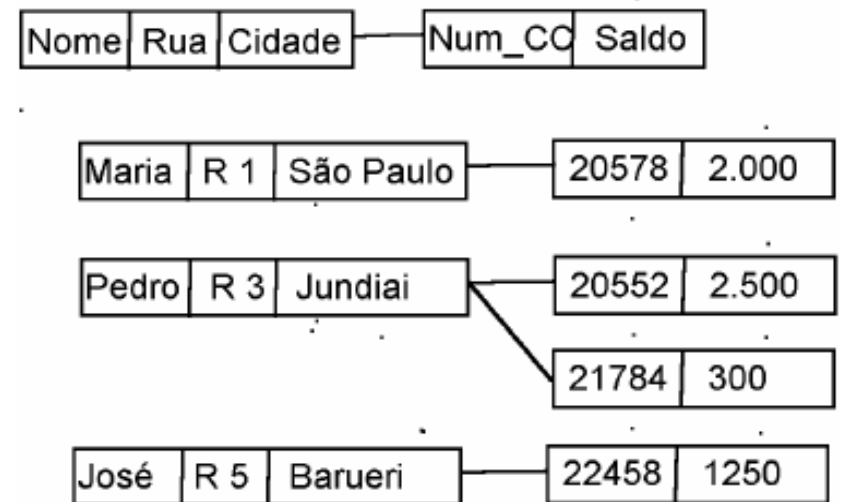
No Modelo em Rede o sistema comercial mais divulgado é o CAIDMS da Computer Associates

Modelo de Rede

- Esquema – parte dos dados vista pelo administrador
- Subesquema - parte dos dados vista pelos aplicativos
- DML – Linguagem de Manipulação (gerenciamento) de dados
- Já trabalha com relacionamento 1:M

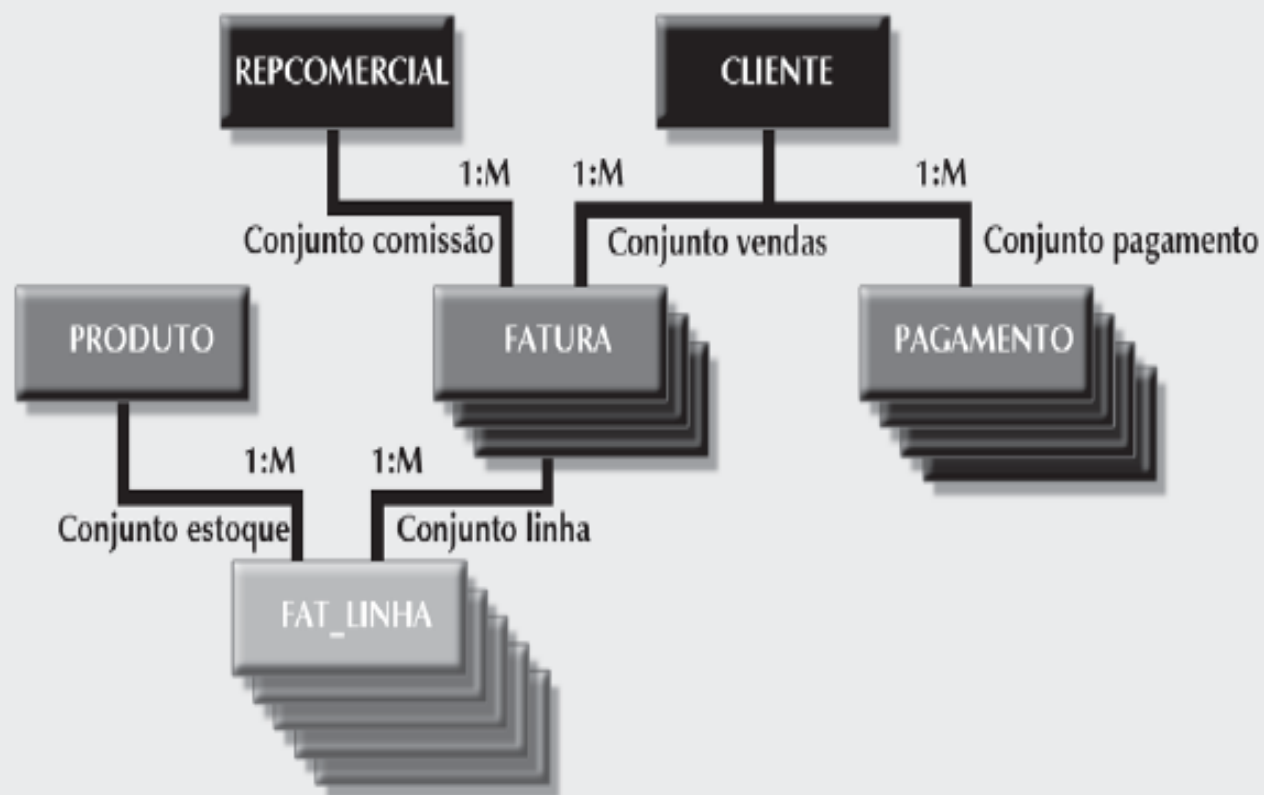
Modelo de Rede

- O diagrama para representar os conceitos do modelo em redes consiste em dois componentes básicos:
 - **Caixas**, que correspondem aos registros
 - **Linhas**, que correspondem às associações.



**FIGURA
2.2**

Modelo de dados de rede



Modelo de Rede

- Limitações
 - Tedioso
 - Falta de recursos de consulta
 - Era preciso gera muitos códigos para produzir relatórios simples

Considerações

Modelos Hierárquico e Rede são Orientados a Registros qualquer acesso à base de dados – inserção, consulta, alteração ou remoção – é feito em um registro de cada vez.

Modelagem de dados

- Entidade – é algo sobre o qual sejam coletados e armazenados dados
- Atributo – é uma característica de uma entidade
- Relacionamento – descreve a associação entre as entidades
 - 1:1
 - 1:M
 - M:N

Modelo Relacional

- Década de 1970 – apresentado por E. Codd (IBM)
- O modelo relacional é baseado a teoria dos conjuntos e álgebra relacional
- O modelo relacional apareceu devido às seguintes necessidades:
 - aumentar a independência de dados nos sistemas gerenciadores de banco de dados;
 - prover um conjunto de funções apoiadas em álgebra relacional para armazenamento e recuperação de dados;

Modelo Relacional

- A estrutura fundamental do modelo relacional é a relação (**tabela**).
- Uma relação é constituída por um ou mais atributos (**campos**) que traduzem o tipo de dados a armazenar.
- Cada instância do esquema (linha) é chamada de tupla (**registro**).

Cod_Cliente	Nome	Rua	Cidade
1	Pedro	A	São Paulo
2	Maria	B	Jundiai

Num_CC	Saldo
20121	1200
21582	1320
21352	652

Cod_Cliente	Num_CC
1	20121
2	21582
2	21352

Modelo Relacional

- O modelo relacional implementa estruturas de dados organizadas em relações.
- Restrições impostas para evitar aspectos indesejáveis, como:
 - Repetição de informação,
 - incapacidade de representar parte da informação
 - perda de informação.
- Essas restrições são:
 - integridade referencial,
 - chaves
 - integridade de junções de relações.

Modelo Relacional

- SGBDR - sistema gerenciador de banco de dados relacional
- Oculta do usuário a complexidade do modelo relacional
- Diagrama relacional
 - Representação gráfica das entidades, dos atributos e dos relacionamentos em um BDR

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	815	882-1244
503	Okon	John	T	815	123-5589

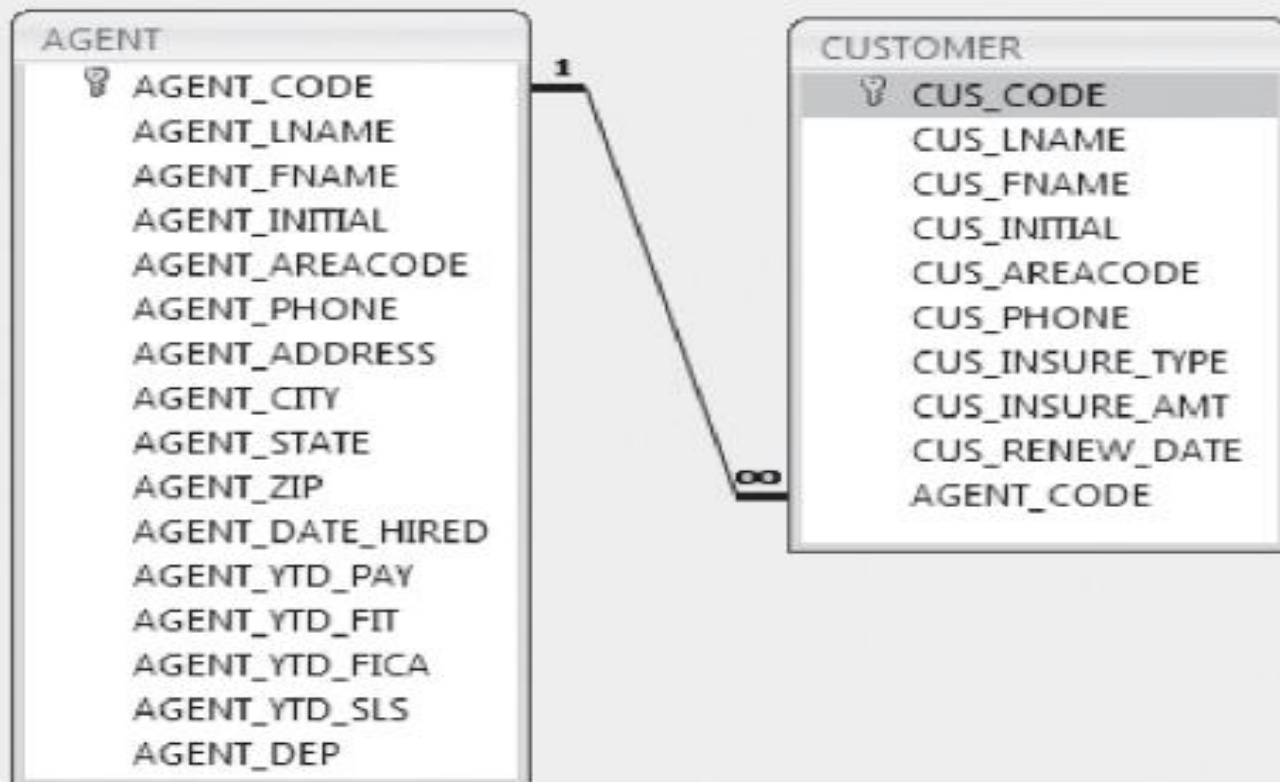
Ligação por meio de AGENT_CODE

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	815	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	W	815	894-2285	S2	150.00	29-Jan-2009	502
10013	Olowski	Paul	F	815	894-2180	S1	300.00	14-Oct-2008	502
10014	Orlando	Myron		815	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	815	297-1228	S1	120.00	25-Mar-2009	502
10017	Williams	George		815	290-2556	S1	250.00	17-Jul-2008	503
10018	Ferriss	Anne	G	713	382-7185	T2	100.00	03-Dec-2008	501
10019	Smith	Olette	K	815	297-3809	S2	500.00	14-Mar-2009	503

**FIGURA
2.4**

Diagrama relacional



Modelo Relacional

- SQL – structured query language - linguagem estruturada de dados
- Envolve 3 partes
 - Interface com o usuário
 - usuário final interaja com os dados
 - Conjunto de tabelas armazenadas no banco
 - Cada tabela é independente uma da outra
 - As linhas da tabela são relacionadas com a base de valores de atributos
 - Mecanismo de SQL
 - Executa todas as consultas ou solicitações de dados

Modelo Entidade – Relacionamento

- Padrão aceito para a modelagem de dados
- Apresentado por Peter Chen em 1976
- Representação gráfica entre de entidades e seus relacionamentos
- Diagrama Entidade-Relacionamento
 - Utiliza representação gráfica para modelar os componentes do BD
 - Uma entidade é mapeada em uma tabela relacional

Modelo Entidade – Relacionamento

- Instância da entidade
 - cada linha de uma tabela
 - cada entidade é definida por um conjunto de atributos que descrevem sua característica particular
- A conectividade indica os tipos de relacionamento
- Relacionamentos são representados por um losango e seu nome é descrito dentro do losango
- Notação pé de galinha é usada - ☹

FIGURA 2.5

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

Notação de Chen

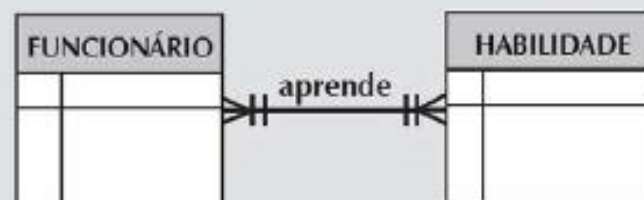
Relacionamento um para muitos (1:M): um PINTOR pode pintar várias PINTURAS; cada PINTURA é criada por apenas um PINTOR.



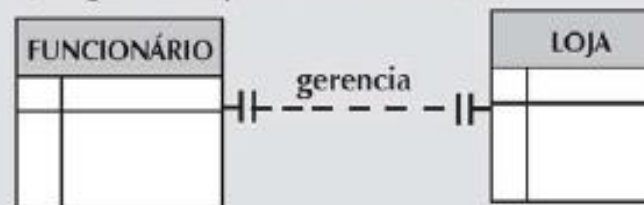
Notação Pé de Galinha



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

- Surgiram em meados de 1980
- Surgiram em função dos limites de armazenamento e representação semântica impostas no modelo relacional.
- Alguns exemplos são:
 - os sistemas de informações geográficas (SIG),
 - os sistemas CAD e CAM

(http://www.rudloff.com.br/conteudo/texto/usinagem/USINAGEM2008-pg_3-revA.pdf), que são mais facilmente construídos usando tipos complexos de dados
- Necessidade em guardar representações das estruturas de dados que utilizam no armazenamento permanente.

Modelo Orientado a Objetos

- Os **atributos** descrevem as **propriedades** de um objeto
- Os **objetos** que compartilham **características similares** são agrupados em **classes**
- **Herança** é a capacidade de um **objeto herdar atributos e métodos** de classes superiores
- A UML (Unified Modeling Language) - linguagem de modelagem unificada - base OO e descreve o conjunto de diagramas

Modelo Orientado a Objetos

- A estrutura padrão para os bancos de dados orientados a objetos foi feita pelo Object Database Management Group (ODMG)

■

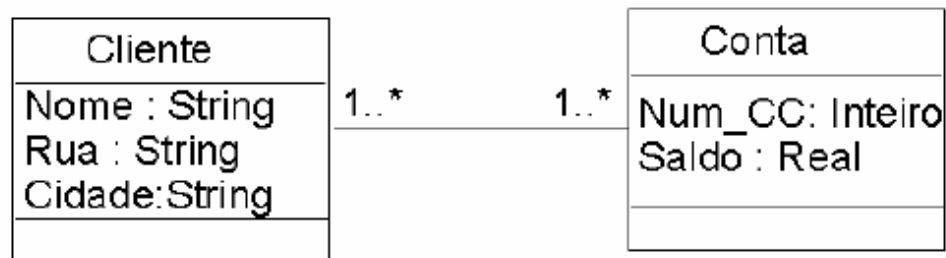


Figura 1.4 - Diagrama UML Cliente - Conta Corrente

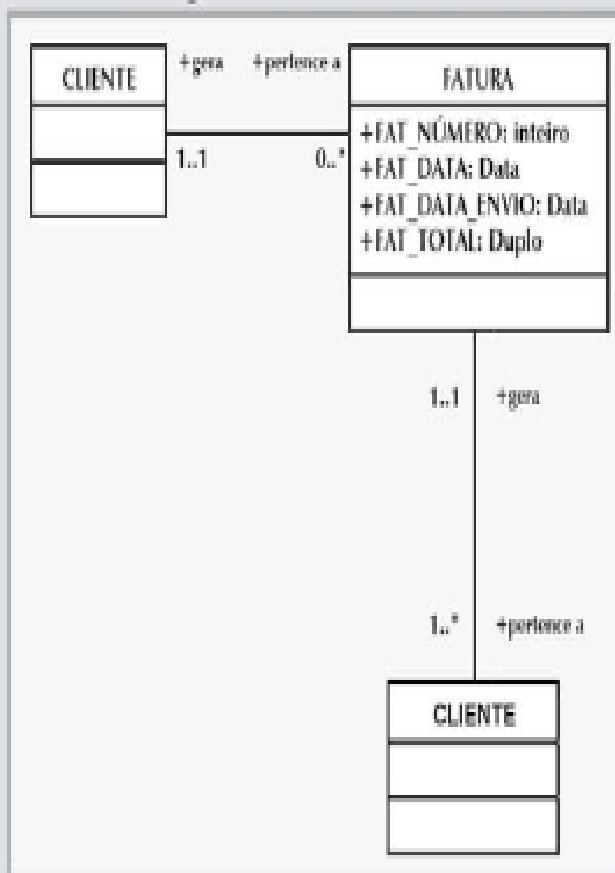
FIGURA
2.6

Comparação dos modelos de OO, UML e ER

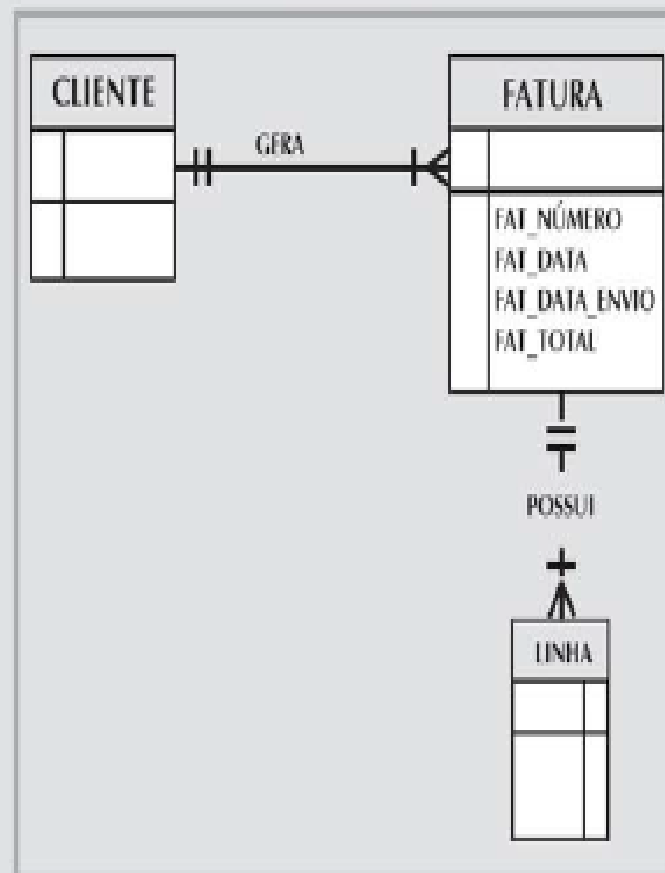
Representação de objetos



Diagrama de classes em UML



Modelo ER



Sistemas Objetos Relacionais

- Sistemas Objeto-Relacional tenta suprir a dificuldade dos sistemas relacionais convencionais
 - que é o de representar e **manipular dados complexos**, visando ser mais representativos em semântica e construções de modelagens.
- A solução é a adição de facilidades para manusear tais dados utilizando-se das facilidades SQL (*Structured Query Language*) existentes.
- Para isso, foi necessário adicionar:
 - extensões dos tipos básicos no contexto SQL;
 - representações para objetos complexos no contexto SQL;
 - herança no contexto SQL
 - sistema para produção de regras.

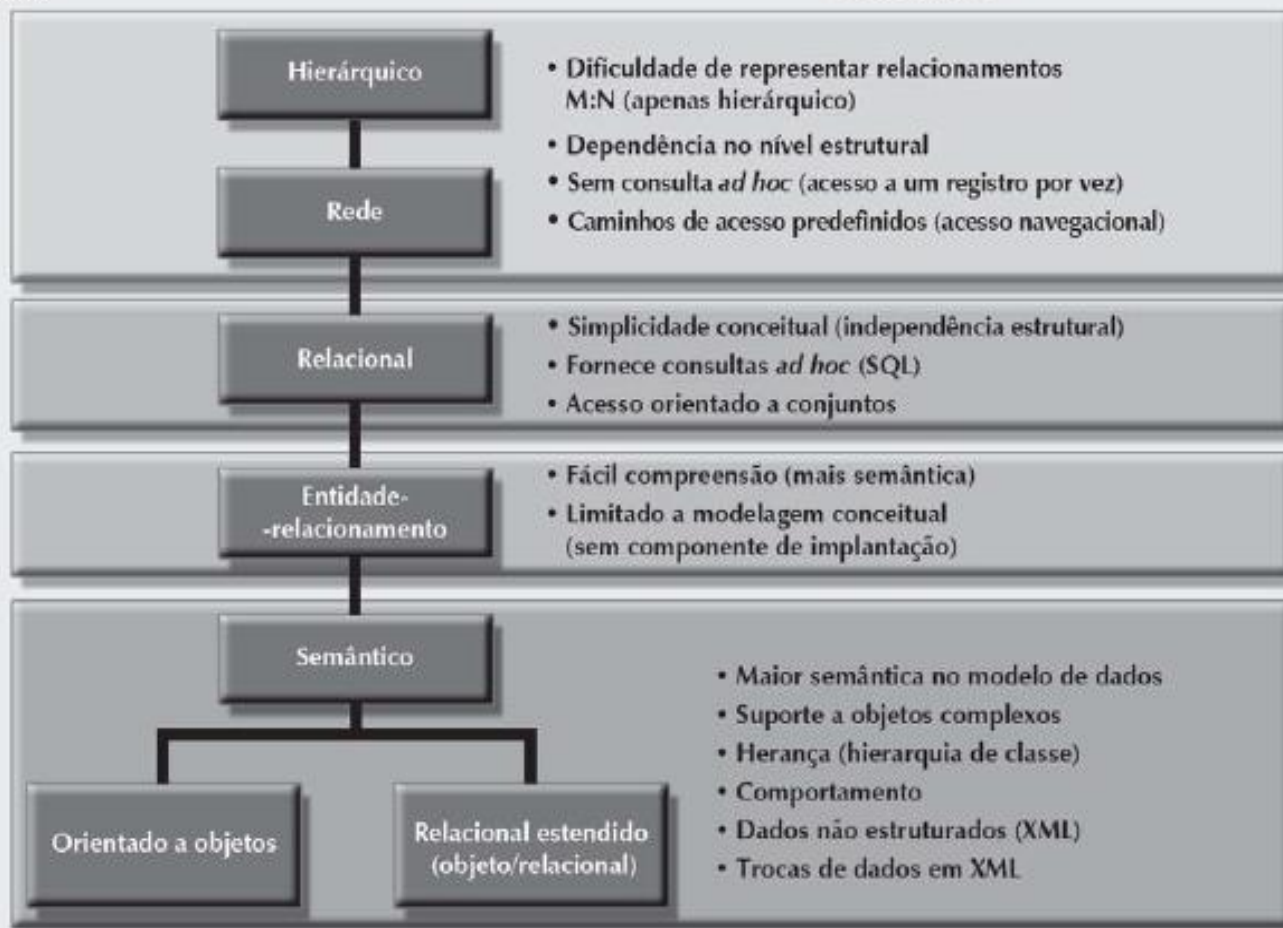
**FIGURA
2.7**

Desenvolvimento dos modelos de dados

Semântica do
modelo de dados

Comentários

menor
↓
maior



Organização de um BD

- Tem que ser **versátil**: isto quer dizer que, dependendo dos usuários ou das aplicações, possam fazer diferentes coisas ou tratem aos dados de formas diferentes.
- Tem que atender com a **rapidez** adequada a cada aplicação ou empresa, atendendo a quem fizer a requisição
- Tem que ter um índice de **redundância** o mais **baixo** possível.
- Ter uma **alta capacidade de acesso** para ganhar o maior tempo possível na realização de consultas.

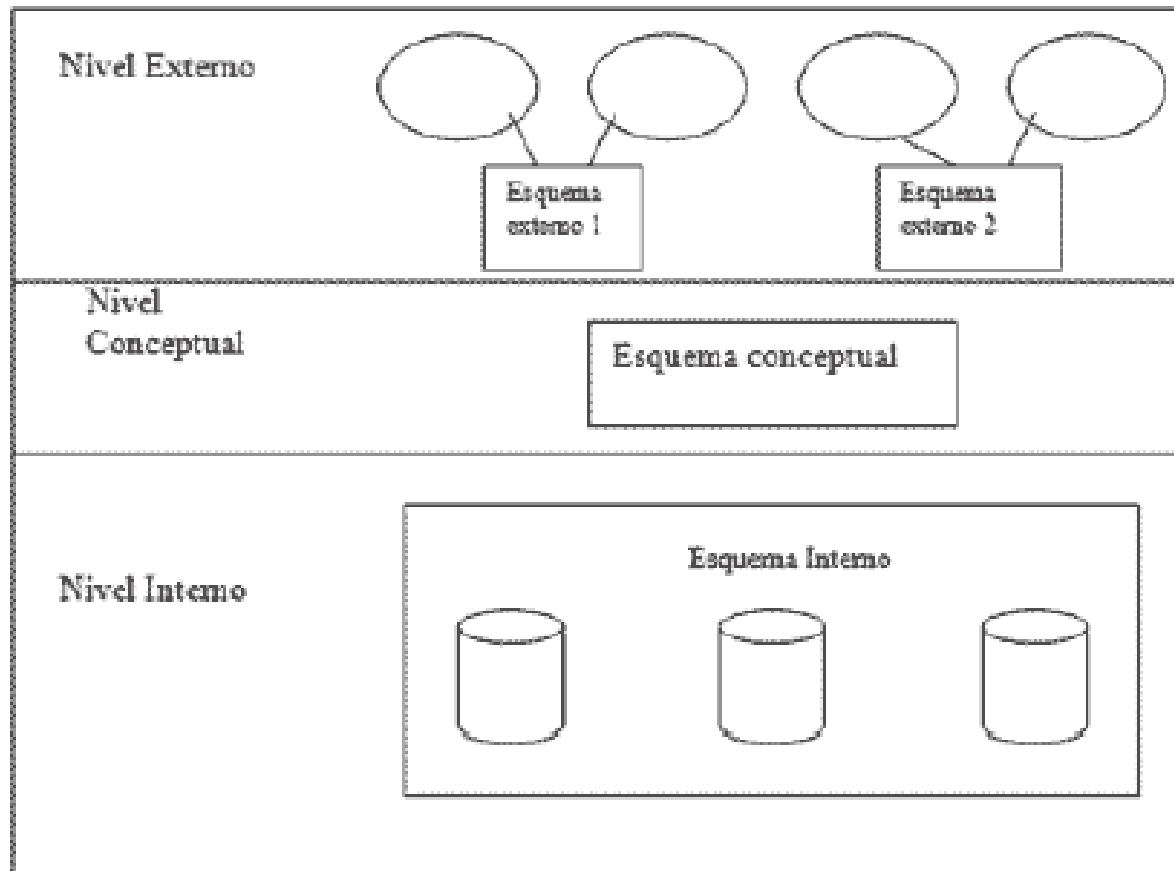
Organização de BD

- Tem que ser possível sua **constante atualização** para não deixar o banco de dados antigo
- Quando fazemos uma mudança na organização física dos dados não deve afetar aos programas, portanto também tem que ter uma independência física dos dados.
- Assim como tem que ter total independência lógica com os dados, isto quer dizer que se fazemos mudanças na estrutura lógica dos dados (agregar novos campos a uma tabela) não devem afetar às aplicações que utilizem esses dados.

Níveis de Visão de usuário

- Três níveis principais segundo a visão e a função que realize o usuário sobre o banco de dados:
 - **Nível externo:** é o mais próximo ao usuário. Neste nível se descrevem os dados ou parte dos dados que mais interessam aos usuários.
 - **Nível conceitual:** Neste nível se representam os dados que vão ser utilizados sem considerar aspectos como o que representamos no nível interno.
 - **Nível Interno:** é o nível mais perto do armazenamento físico dos dados. Permite escrevê-los tal e como estão armazenados no computador. Neste nível se desenham os arquivos que contém a informação, a localização dos mesmos e sua organização, ou seja, criam-se os arquivos de configuração.

Níveis de Visão de usuário



- Responsável por:
 - Definição dos dados
 - Manutenção da integridade dos dados dentro do banco de dados
 - Controle da segurança e privacidade dos dados
 - Manipulação dos dados

Dicionário de dados

- Qualquer operação que o usuário faz contra o banco de dados está controlada pelo administrador.
- O administrador armazena uma descrição de dados no qual chamamos dicionário de dados

- **Dicionário de dados** É um banco de dados onde se salvam todas as propriedades do banco de dados, descrição da estrutura, relações entre os dados, etc.

- O dicionário deve conter:
 - A descrição externa, conceitual e interna do banco de dados
 - As restrições sobre os dados
 - O acesso aos dados
 - As descrições das contas de usuário
 - As permissões dos usuários
 - Os esquemas externos de cada programa

Exemplo de dicionário de dados

TABELA: Tb_Acompanhamento				Acompanhamento dos alunos		
Informações relacionadas ao acompanhamento de alunos pelos responsaveis						
CAMPO LÓGICO	CAMPO FÍSICO	TIPO	PK	FK (Tabela/Campo)	RESTRIÇÕES	OBSERVAÇÕES
Código	Aco_codigo	SMALLINT	PK		NÃO NULO E MAIOR QUE ZERO	Campo auto-incremento
Primeiro trimestre	Aco_priutri	INTEGER			IGUAL OU MAIOR QUE ZERO E MENOR QUE ONZE	Nota do primeiro trimestre
Segundo Trimestre	Aco_segtri	INTEGER			IGUAL OU MAIOR QUE ZERO E MENOR QUE ONZE	Nota do segundo trimestre
Terceiro Trimestre	Aco_tertri	INTEGER			IGUAL OU MAIOR QUE ZERO E MENOR QUE ONZE	Nota do terceiro trimestre
Nota final	Aco_notafinal	INTEGER			IGUAL OU MAIOR QUE ZERO E MENOR QUE ONZE	Media dos 3 trimestres
Situação final	Aco_sitfinal	VARCHAR(15)			APROVADO, REPROVADO OU RECUPERAÇÃO	Situação do aluno para o curso

Administrador BD

Saber trabalhar em equipe
Concentração
Trabalhar sob pressão

- É uma pessoa ou grupo de pessoas responsáveis do controle do sistema de gerenciamento do banco de dados.
- As **principais tarefas** de um administrador são:
 - A definição do esquema lógico e físico do banco de dados
 - A definição das vistas de usuário
 - A atribuição e edição de permissões para os usuários
 - Manutenção e seguimento da segurança no banco de dados
 - Manutenção geral do sistema de gerenciamento do banco de dados
 - Instalação e Gerenciamento do Banco de Dados, bem como dos seus usuários, sempre prezando pela segurança
 - Criar backups periódicos, além de recuperá-los quando for necessário
 - Saber monitorar e ajustar a performance do Banco de Dados, para que ele possa ter o melhor desempenho.

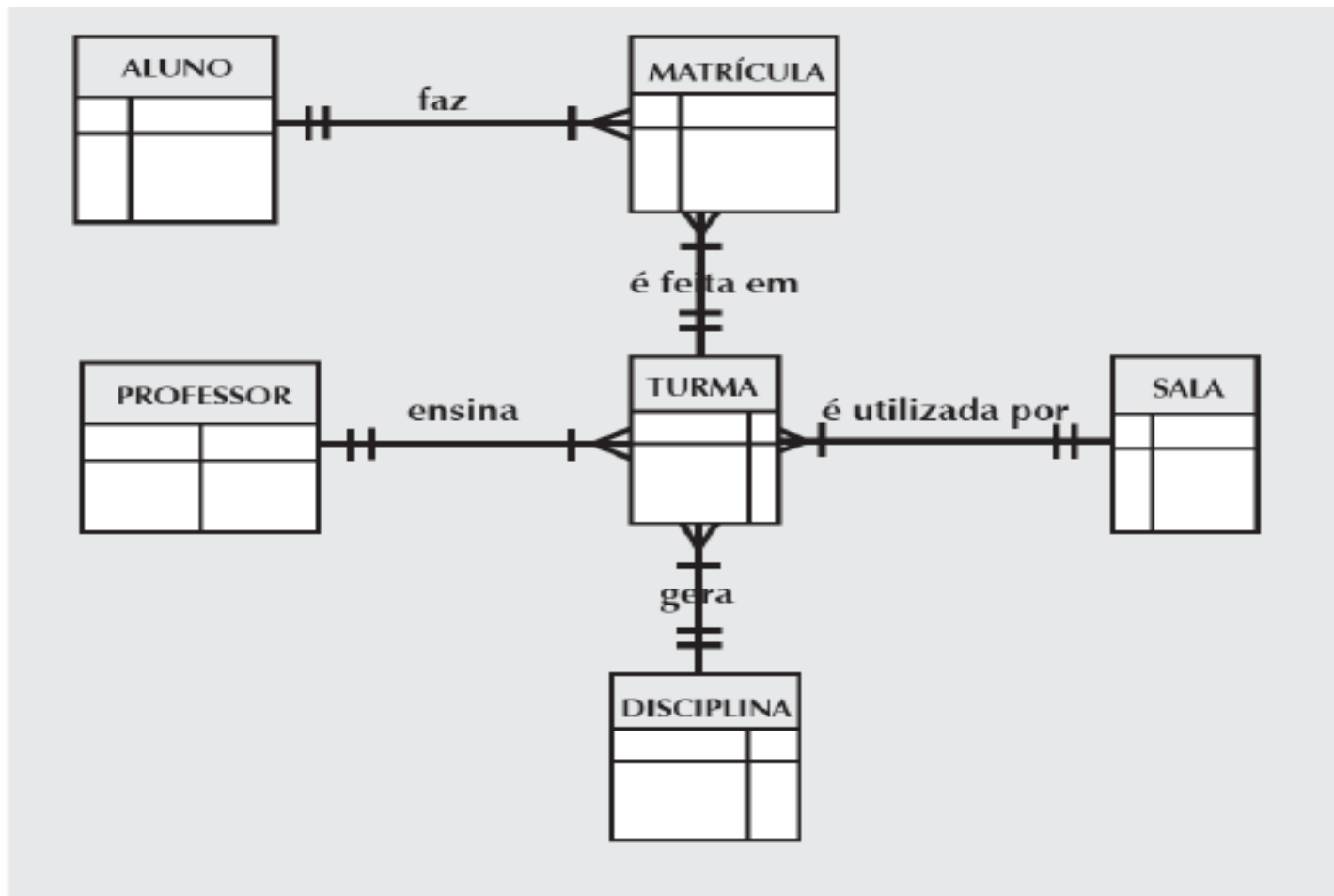
- Fazer uma avaliação completa a nível de *Hardware* do Servidor em que o Banco de Dados ficará hospedado;
- Realizar todo o planejamento e a implementação do Banco de Dados;
- Verificar se todas as tarefas e rotinas agendadas foram executadas com sucesso;
- Monitorar se há espaço em disco para garantir que os dados serão armazenados sem nenhum problema;
- Registrar todas as alterações feitas no servidor, bem como documentar qualquer problema que venha a ocorrer no que diz respeito ao seu desempenho, além de trabalhar nas correções que forem necessárias;
- Buscar por boas práticas em busca de melhorar a performance do Banco de Dados, é o que chamamos de Tunning;
- Criar alertas de notificação, caso ocorram problemas que sejam mais graves;

Linguagens

- SGBD deve proporcionar uma série de linguagens para a definição e manipulação do banco de dados.
- Estas linguagens são as seguintes:
 - Linguagem de definição de dados (DDL).
 - Para definir os esquemas do banco de dados
 - Linguagens de manipulação de dados (DML).
 - Para manipular os dados do banco de dados
 - Linguagem de controle de dados(DCL).
 - Para a administração de usuários e segurança no banco de dados.

Modelo Conceitual

- Representa uma visão geral do BD
- O modelo mais usado para fazer um modelo conceitual é o ER
- O DER (diagrama entidade-relacionamento) é utilizado para representar graficamente o modelo conceitual
- Independe do SGBD

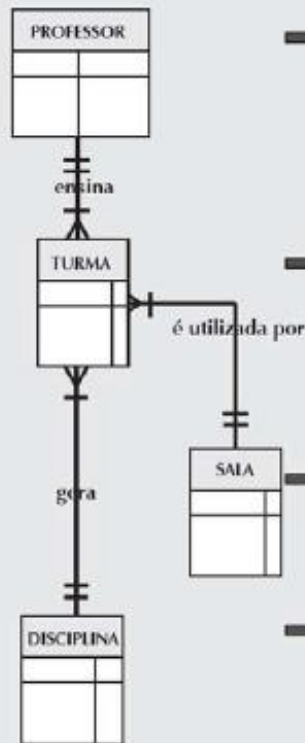


Modelo interno

FIGURA
2.11

Modelo interno para a Tiny College

MODELO CONCEITUAL



MODELO INTERNO

Create Table PROFESSOR(
PROF_ID
PROF_SOBRENOME
PROF_INICIAL
PROF_NOME
.....);

NUMBER PRIMARY KEY,
CHAR(15),
CHAR(1),
CHAR(15)

Create Table TURMA(
TURMA_ID
DIS_ID
PROF_ID
SALA_ID
.....);

NUMBER PRIMARY KEY
CHAR(8) REFERENCES DISCIPLINA,
NÚMERO REFERENCES PROFESSOR,
CHAR(8) REFERENCES SALA,

Create Table SALA(
SALA_ID
SALA_TIPO
.....);

CHAR(8) PRIMARY KEY,
CHAR(3),

Create Table DISCIPLINA(
DIS_ID
DIS_NOME
DIS_CRÉDITO
.....);

CHAR(8) PRIMARY KEY,
CHAR(25),
NUMBER,