



Interativa

**Administração de
Banco de Dados**

Professora conteudista: Cida Atum

Sumário

Administração de Banco de Dados

Unidade I

1 INTRODUÇÃO A BANCO DE DADOS.....	1
1.1 Histórico.....	1
1.2 Definições.....	2
1.3 Importância dos sistemas de bancos de dados nas organizações	3
1.4 Linguagens de banco de dados.....	4
1.5 Níveis da arquitetura de banco de dados.....	6
2 SISTEMAS GERENCIADORES DE BANCO DE DADOS - SGBD.....	7
2.1 Definição.....	7
2.2 Características do SGBD.....	9
2.3 Projeto de banco de dados.....	9

Unidade II

3 MODELAGEM DE DADOS.....	11
3.1 Tipos de modelos de dados	11
3.1.1 Modelo conceitual.....	11
3.1.2 Modelo lógico	11
3.1.3 Modelo físico	12
3.2 Modelo Entidade-Relacionamento (MER)	12
3.2.1 Atributos.....	14
3.2.2 Restrições.....	14
3.2.3 Entidades fortes e entidades fracas	16
3.3 Dicionário de dados.....	18
3.4 Ferramentas CASE.....	19
3.4.1 Definição	19
3.4.2 A ferramenta CASE <i>ErWin</i>	20

Unidade III

4 ADMINISTRAÇÃO DE SGBDs.....	24
4.1 Segurança e administração - controle de acesso	26
4.2 Recuperação (<i>recovery</i>)	28
4.2.1 Recuperação de sistema	29
4.2.2 Recuperação da mídia	30
4.3 Replicação de dados	31
4.4 Formas de melhoria de desempenho	33
4.4.1 Simulação de desempenho.....	33

Unidade IV

5 DEFINIR A MELHOR SOLUÇÃO DE BANCO DE DADOS PARA AS NECESSIDADES

DA EMPRESA.....	35
5.1 O uso das tecnologias.....	35
5.1.1 O <i>software</i> livre	35
5.1.2 Ferramentas de SGBD	37
5.2 Requisitos de <i>software</i> – arquitetura cliente/servidor.....	40

Unidade I

1 INTRODUÇÃO A BANCO DE DADOS

1.1 Histórico

O banco de dados relacional surgiu na empresa IBM entre as décadas de 1960 e 1970, ao desenvolver funções através de pesquisas de automação de escritório. Tal necessidade deve-se ao fato de que as empresas constatavam o baixo custo da informatização em tarefas que exigiam processos repetitivos. Iniciou-se uma época muito valiosa de pesquisas em linguagem de programação industrial com altos investimentos na área.

As pesquisas desenvolvidas nesta época trouxeram os primeiros estudos sobre modelos de banco de dados de rede hierárquicos, além de outras tecnologias empregadas até hoje.

Ted Codd era um destes pesquisadores, e, em 1970, escreveu um artigo técnico sobre Banco de Dados Relacionais, no qual estabelecia o conceito primordial da linguagem: o armazenamento das informações em tabelas onde o usuário poderia acessá-las através de comando em inglês. A complexidade do artigo levou a IBM a organizar um centro de pesquisa que ficou conhecido como *System R*, tendo como principal intuito desenvolver um produto cujo sistema fosse um banco de dados relacional.

O *System R* teve muitas versões que foram sendo utilizadas por várias organizações de peso nos Estados Unidos, até que a evolução do sistema eventualmente tornou-se DB2, e a linguagem desenvolvida para ser utilizada junto ao sistema foi a SQL (*Structured Query Language* - Linguagem de Consulta Estruturada), considerada pela ISO (*International*

Organization for Standardization - Organização Internacional de Padronização) linguagem padrão dos sistemas de banco de dados relacionais.

Apesar do pioneirismo conceitual da IBM, foi mesmo a *Honeywell Information Systems Inc* a primeira empresa a produzir comercialmente o sistema de banco de dados em 1976. A *Honeywell* implementou o sistema criado pela IBM e este foi completamente remodelado.

Na década de 80 os *softwares* de banco de dados relacionais foram sendo evoluídos e os usuários começaram um processo de *feedback* do sistema, devido principalmente ao vertiginoso aumento de sistemas distribuídos e aos computadores-pessoas.

Atualmente a capacidade de armazenamento de dados ultrapassa centenas de *Terabytes*, o que levou consequentemente ao aumento de tamanho destes sistemas. Um dos projetos mais ambiciosos trata-se do desenvolvimento de um banco de dados distribuído que possui uma capacidade de armazenamento em torno de *Hexabytes* (1 *Hexabyte* = 1,000 *Petabytes* = $1 * 10^{18}$ Bytes), este projeto está em fase de andamento pela *CERN* (*European Organization for Nuclear Research* - Organização Européia para a Investigação Nuclear).

O desenvolvimento da linguagem padrão SQL é financiado atualmente pela ANSI (*American National Standards Institute* - Instituto Nacional Americano para Padrões) e pela ISO, que formam um grupo de pesquisas para a contínua evolução da linguagem.

1.2 Definições

Banco de dados é uma coleção de dados inter-relacionados cuja representação refere-se a informações específicas, como por exemplo, acervos de bibliotecas, lista de clientes e fornecedores, controle de RH de uma empresa, etc.

Banco de dados relacional é baseado no modelo relacional e usa um conjunto de tabelas para representar os dados e as relações entre si, a maioria utiliza a linguagem SQL.

1.3 Importância dos sistemas de bancos de dados nas organizações

5 A importância dos sistemas de bancos de dados nas organizações é vista pela crescente valorização dos bancos de dados e dos SGBDs, o que gera consequentes investimentos em técnicas de gerenciamento, monitoramento, *backup* e restauração de dados e em todo o processo que envolve a importância financeira de manter a integridade dos bancos de
10 dados.

Um problema muito real refere-se ao gerenciamento de todas as contas bancárias em sistemas de arquivos permanentes de um determinado banco. Este sistema possui uma série de programas aplicativos necessários para a manipulação por parte
15 dos usuários, que permitem:

- débito e crédito em outra conta;
- um programa para adicionar uma nova conta;
- fazer pagamentos e depósitos;
- calcular aplicações;
- 20 • inserir novas alíquotas.

Esses aplicativos só foram desenvolvidos porque surgiram problemas e necessidades da organização bancária, e isso significa um processo contínuo, pois as aplicações são desenvolvidas conforme vão surgindo as necessidades. Em resumo, os arquivos
25 e os programas são desenvolvidos e acrescentados ao sistema sempre que for preciso.

Este sistema de processamento de arquivos tem deixado de existir gradualmente, pois apresenta uma infinidade de desvantagens, como define Silberschatz (et. al,1999):

- 5 • **difícil acesso:** filtrar uma informação torna-se uma tarefa árdua, já que se pode fazer isso manualmente ou se pode gerar um programa para cada filtro;
- **isolamento:** a informação de formatos diferentes contida em arquivos separados torna difícil desenvolver novos programas aplicativos de recuperação de dados;
- 10 • **redundância e inconsistência:** diversos programadores e diversas linguagens produzem vários tipos de formatos, podendo ser geradas por exemplo, informações em duplicidade, em tabelas diferentes. Inconsistência de dados é gerada pela manutenção de cópias que podem
- 15 estar com valores diferentes;
- **problemas de segurança:** um banco de dados deve manter controles de acessos, dependendo da informação. Se os aplicativos forem adicionados regularmente, este processo de segurança fica restrito;
- 20 • **problemas de integridade:** às vezes os valores dos dados precisam satisfazer algumas restrições, como o saldo nunca estar abaixo de "X" reais, por exemplo. Quando novas restrições forem necessárias, torna-se-á difícil alterar estes programas.
- 25 Estes são alguns dos motivos pelos quais se faz necessária uma séria abordagem da aplicação de gerenciamento de sistemas de banco de dados em uma organização.

1.4 Linguagens de banco de dados

Um sistema de banco de dados fornece uma linguagem de definição de dados para especificar o esquema do mesmo
30 e, uma linguagem de manipulação de dados para expressar as

consultas e a atualização de dados. Na prática, as linguagens de definição de dados e de manipulação de dados são duas linguagens separadas, simplesmente formam partes de uma única linguagem de banco de dados, como a amplamente usada linguagem SQL (DATE, 2003).

Linguagem de Definição de Dados - DDL

Os procedimentos de acesso a um sistema de banco de dados e sua estrutura de armazenamento são definidos por um conjunto de comandos básicos chamados **DDL** (*Data Definition Language* - Linguagem de Definição de Dados), que permite ao usuário definir tabelas novas e seus elementos. Os comandos DDL são:

- **CREATE**: cria uma tabela;
- **DROP**: exclui uma tabela;
- **ALTER**: altera a estrutura da tabela.

Linguagem de Manipulação de Dados - DML

A linguagem **DML** (*Data Manipulation Language*, - Linguagem de Manipulação de Dados) permite o acesso aos dados e/ou manipulá-los. Existem basicamente 4 comandos DML:

- **SELECT**: seleciona dados especificando uma *query* (comando que executa uma busca);
- **INSERT**: insere dados a uma tabela existente;
- **UPDATE**: altera os valores de dados em tabela;
- **DELETE**: remove dados de uma tabela.

A seguir podemos visualizar a completa arquitetura de um sistema de banco de dados (Fig. 1.1):

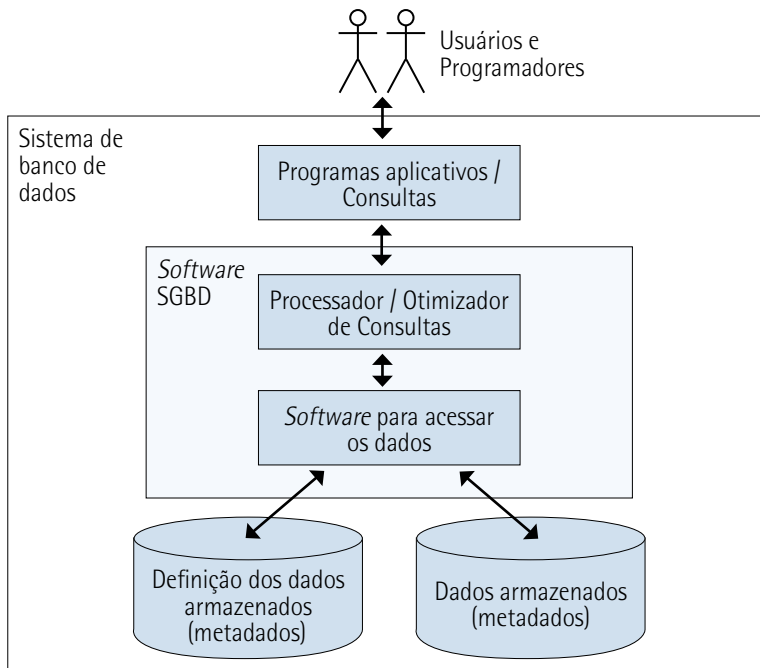


Fig. 1.1 - Diagrama da arquitetura do sistema de banco de dados

1.5 Níveis da arquitetura de banco de dados

De modo geral, como nos mostra DATE (2003:29), a arquitetura se divide em três níveis (Fig. 1.2), a saber:

- O **nível interno** (também conhecido como nível de **armazenamento**) é o mais próximo do meio de armazenamento físico, ou seja, é aquele que se ocupa do modo como os dados são fisicamente armazenados dentro do sistema.
- O **nível externo** (também conhecido como nível **lógico do usuário**) é o mais próximo dos usuários, ou seja, é aquele que se ocupa do modo como os dados são vistos por usuários individuais.
- O **nível conceitual** (também conhecido como nível **lógico de comunidade**, ou às vezes apenas nível **lógico**, sem qualificação), é um nível "indireto" entre os outros dois.

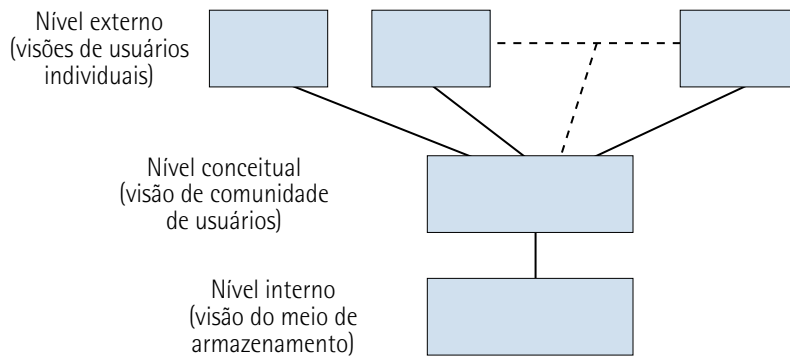


Fig. 1.2 – Os três níveis da arquitetura.

Observe que o nível externo se preocupa com as percepções dos usuários individuais, enquanto o nível conceitual está preocupado com uma percepção da comunidade de usuários. A maior parte dos usuários não estará interessada no banco de dados inteiro, mas somente em alguma parte restrita dele; assim, haverá muitas "visões externas" distintas, cada qual consistindo em uma representação mais ou menos abstrata de alguma parte do banco de dados completo, e haverá exatamente uma "visão conceitual", consistindo em uma representação igualmente abstrata do banco de dados em sua totalidade. Do mesmo modo, haverá exatamente uma "visão interna", representando o modo como o banco de dados está armazenado internamente.

Observe que os níveis externo e conceitual são níveis de **modelo**, enquanto o nível interno é um nível de **implementação**; em outras palavras, os níveis externo e conceitual são definidos em termos de construções voltadas para o usuário, como registros e campos, enquanto o nível interno é definido em termos de construções voltadas para a máquina, como *bits* e *bytes*.

2 SISTEMAS GERENCIADORES DE BANCO DE DADOS - SGBD

2.1 Definição

Sistema Gerenciadores de Banco de Dados (SGBD) é um programa com recursos específicos, que tem o objetivo de manipular as informações contidas nos bancos de dados. Como

exemplo podemos citar o *Ingres*, o *Oracle*, o *Access*, o *MySQL* e o *DBase*.

Um Sistema Gerenciador de Banco de Dados, normalmente referido apenas por "*banco de dados*", refere-se à administração gerencial de um conjunto de dados, estruturando com eficiência as informações e adequadamente às exigências de segurança e armazenamento (MEDEIROS, 2006).

A função do SGBD é facilitar e simplificar o acesso aos dados pelos usuários, gerenciando grupos complexos de informações e fornecendo segurança contra os problemas que venham a ocorrer no sistema e contra a invasão de acessos restritos.

Os componentes funcionais de um banco de dados incluem:

- **gerenciador de arquivos:** gerencia o espaço do armazenamento;
- **gerenciador do banco de dados:** gerencia a interface entre os dados e os programas aplicativos;
- **processador de consultas:** traduz os comandos numa linguagem que o gerenciador do banco de dados possa interpretar;
- **pré-compilador da DML:** converte comandos DML para gerar o código apropriado;
- **compilador da DDL:** converte comandos DDL em um conjunto de tabelas contendo **metadados**.

Além desses componentes, outros são relativamente importantes no desenvolvimento da fase do projeto físico do sistema:

- **arquivos de dados:** é o armazenamento físico do banco de dados;
- **dicionário de dados:** é o armazenamento dos metadados;

- **índices:** o acesso ágil aos dados.

2.2 Características do SGBD

Geralmente um Sistema Gerenciador de Banco de Dados deve ter as seguintes características:

- 5 • **evitar a redundância:** deve poder evitar dados redundantes, evitando a inconsistência das informações;
- **manipulabilidade:** deve facilitar o uso do banco de dados mesmo aos que não o conhecem tecnicamente.
- 10 • **independência física:** o esquema do modelo lógico deve manter-se afastado do nível físico do SGBD para que não haja abstrações, simplificando a interação do usuário com o sistema;
- **independência lógica:** o nível físico do SGBD pode ser alterado independentemente da utilização do usuário;
- 15 • **centralização administrativa:** deve permitir o gerenciamento dos SGBDs de maneira centralizada;
- **rapidez dos acessos:** deve permitir o acesso rápido e ágil aos dados;
- **preservar a integridade:** preservar a coerência entre os dados;
- 20 • **compartilhamento:** deve permitir o acesso simultâneo ao banco de dados;
- **segurança dos dados:** deve prevenir-se de métodos de gerenciamento de acesso.

2.3 Projeto de banco de dados

25 Segundo Silberschatz (et. tal, 1999), um sistema de banco de dados é projetado para gerenciar grandes blocos de informações. Esses grandes blocos não existem isoladamente. Eles são parte da operação de alguma empresa cujo produto final pode ser

informações do banco de dados ou pode ser algum dispositivo ou serviço para o qual o banco de dados desempenha apenas um papel de apoio.

- 5 O projeto de banco de dados envolve principalmente o projeto do esquema de banco de dados. A modelagem requer atenção especial.

- 10 Um modelo de dados serve ao projetista de banco de dados uma estrutura conceitual para especificar, de maneira sistemática, quais são as necessidades, por exemplo, numa instituição financeira, de dados dos usuários do banco de dados, e como o banco será estruturado para satisfazer essas necessidades. Portanto, a fase inicial do projeto de banco de dados é caracterizar completamente as necessidades de dados dos usuários de banco de dados potenciais.

- 15 O projetista de banco de dados precisa interagir extensivamente com especialistas e usuários do domínio para realizar essa tarefa. O resultado dessa fase é uma especificação das necessidades do usuário.

- 20 A seguir, o projetista escolhe um modelo de dados e, aplicando os conceitos do modelo escolhido, traduz essas necessidades em um esquema conceitual do banco de dados. O esquema desenvolvido nessa fase de **projeto conceitual** fornece uma visão geral detalhada da empresa. O projetista revisa o esquema para confirmar se todas as necessidades de dados estão realmente satisfatórias e se não estão em conflito entre si. O projetista também pode examinar o projeto para remover quaisquer recursos redundantes. O foco nesse momento é descrever os dados e suas relações, e não especificar detalhes do armazenamento físico.

- 30 Em termos do modelo relacional, o processo do projeto conceitual envolve decisões sobre quais atributos queremos capturar no banco de dados e como agrupar esses atributos para formar as várias tabelas. A maneira mais utilizada para tratar esse problema é usar o **modelo de entidade/relacionamento**.
