Bancos de Dados e Sistemas de Bancos de Dados são componente essencial na vida moderna. Se parar para pensar, você vai perceber que em um único dia de vida é ator em inúmeras transações computadorizadas sobre bancos de dados.
Pausa para exercício: em um dia típico, quais os BDs onde você é registrado ou consultado ?

A maior parte (ainda) de tais bancos de dados são do tipo tradicional (em que a informação é texto ou uma combinação de texto e números). No futuro teremos além destes, bancos de dados multimídia (com BLOBs de áudio, vídeo, imagens, etc); com georreferenciação (atributos geográficos, devidamente interfaceados com mapas de todo o tipo); em tempo real (com *delay* de atualizações tipicamente de segundos) e o que mais a humanidade vier a inventar.

Como definição genérica, podemos usar: um banco de dados é *uma coleção de dados relacionados*. Mais restritamente pode-se usar a definição: *um banco de dados é uma coleção de dados relacionados a partir de uma fonte da qual os dados são derivados; com algum grau de iteração com eventos do mundo real e um público ativamente interessado em seu conteúdo*. Seu tamanho varia desde centenas de bytes (uma lista de livros em casa) até terabytes (ou petabytes) de informação.

A grande diferença do BD para a tecnologia anterior a ele, é que agora os dados passam a ser o grande acervo das organizações (antes eram os programas). Uma das grandes motivações no desenvolvimento da tecnologia de BD é a constatação de que dados (o que o prof. Horácio Soares chama de depósito) são muito mais estáveis do que os programas que os manuseiam (o Horácio chama de fábrica). Antes do BD, os programas (cada programa) definia seus dados. Bastava uma inconsistência entre um e outro programa, para que a coisa desandasse (e isto sempre foi muito comum).

Um SGDB (sistema gerenciador de bancos de dados) é uma coleção de programas que permite aos usuários criar, usar e manter um banco de dados. Agora, o problema acima está resolvido. Os Bancos de dados têm um componente importante chamado METADADOS (ou seja, dados sobre dados). A descrição (nome, formato, regras de validação, regras de integridade, etc etc) não estão mais nos programas e sim dentro do DB, ou seja, eles acompanham os dados. As principais operações em um SGDB são

criação usando DDL (Data Definition Language), criam-se todos os elementos do DB. Todo este conjunto configura o *dicionário de dados*.

manutenção usando DML (Data Manipulation Language) dados são recuperados, operados e introduzidos no DB, seja através de programas aplicativos (que interfaceiam com o DB) seja através de consultas não estruturadas (*ad hoc*).

garantia sob este título supõe-se operações de validação de direitos, operações de back-up e restore, de log, de tunning etc.

Um exemplo: Seja um banco de dados de uma biblioteca, formado pelos conjuntos de dados (relações): livro, exemplar, leitor, emprestimo, autor. Cada uma destas relações terá a seguinte lista de atributos: livro (nome, autor, editora, idioma, ISBN, preço); exemplar: (edição, registro); leitor (nome, CPF, endereço, CEP, telefone, categoria); emprestimo (registro, CPF-leitor, data de emprestimo, prazo); autor: (nome, ISBN do livro). Diversas decisões precisam ser tomadas (e respeitadas depois):
O que caracteriza um livro ?
O que caracteriza um volume
Quantos empréstimos um leitor pode fazer ? Depende da categoria ?
Pode-se usar CPF para identificar pessoas ?

Um banco de dados bem simples como este, pode responder às seguintes questões:
Quais os livros mais emprestados ?
Quem é o campeão dos atrasos de devolução ?
Qual o autor mais lido ?
Qual o bairro da cidade que mais concentra leitores ?

Na criação de um BD (que pode ser uma parcela da construção de um sistema de informações, ou que ao contrário, pode vir antes da definição dos sistemas de informações) devem ser desenvolvidas as seguintes etapas: Especificação e análise de requisitos (o que o BD deve suportar); Projeto conceitual (a primeira encarnação do BD: muitas vezes usam-se ferramentas automatizadas aqui. Nós deveremos usar o pacote livre chamado dbDesigner. O resultado principal é o MER-modelo de Entidades e Relacionamentos); Projeto Lógico (descrição independente de qual implementação do DB sofrerá) e Projeto Físico (a implementação dos arquivos, índices, ferramentas de apoio etc). Depois disto, vem a fase de população do DB e posteriormente seu uso em condições normais de produção. Vale lembrar que o BD hoje representa um esforço e investimento significativo das organizações, razão pela qual se espera que o mesmo dure anos (ou décadas). Isso introduz a necessidade dele crescer e ser modificado ao longo dos anos sem perder a funcionalidade e completa operação.

Em resumo, as principais características de um BD são

- capacidade auto-descritória e auto-contida
- isolamento entre dados e programas, abstração de dados
- suporte a múltiplas visões dos dados
- compartilhamento de dados
- eventual criptografia e compactação
- capacidade de multiprocessamento

Esta última característica é importante para garantir o controle da concorrência, implementando o conceito de transação. Esta deve ter a característica de isolamento (uma transação nada sabe sobre as demais) e atomicidade (ou todas as operações da transação são feitas, ou nenhuma é feita)

Vale ressaltar que modernamente há a tendência de SGDBs orientados a objetos (ainda pouco usados, abaixo de 5% segundo o Elmasri,2011), os quais implementam a abstração não apenas para os dados, mas também para os procedimentos que os manuseiam.

As vantagens de usar um BD vis-a-vis a modalidade tradicional, são:

- controle da redundância e sua irmã gêmea: a inconsistência.
- restrição de acessos
- processamento eficiente de consultas e acessos
- processos de backup e restore
- múltiplas interfaces aos usuários
- representação de relacionamentos complexos
- restrições de integridade
- automação de ações via regras e gatilhos
- padrões consistentes para a organização
- diminuição de prazos de desenvolvimento
- oferece informações atualizadas
- garante economia de escala

Alguns conceitos derivados do modelo relacional:

entidade Representa um objeto ou conceito do mundo real. Na implementação, a entidade representa uma tabela de dados.

atributo São as propriedades da entidade. Na implementação são as colunas da tabela (mais restritamente, são os cabeçalhos das colunas)

t-uplas As encarnações dos atributos para um determinado item de dados. Aqui a implementação são as linhas da tabela.

relacionamento Nem sempre uma entidade registra um objeto muitas vezes o que é representado é um relacionamento entre dois objetos. Por exemplo, no caso acima, um empréstimo.

Note que se este modelo for bem construído será possível criar um relacionamento 1:1 entre entidades do modelo e arquivos do SGDB e de outro lado entre entidades do modelo e entidades da organização quem têm dados armazenados no BD.

O que uma relação é diferente de um arquivo

ordenação de tuplas Matematicamente, os elementos de um conjunto não guardam ordem. Mas, em um arquivo, um registro segue o outro, há portanto ordem. A ordenação não é importante em uma relação, já que esta descreve fatos de maneira abstrata.

ordenação de valores na tupla Aqui a ordenação é importante, já que segue da definição. Entretanto, o que interessa é o relacionamento entre atributo e valor. Em uma definição alternativa, uma tupla pode ser considerada um conjunto de pares *atributo* e *valor*. Novamente, na prática, a implementação em arquivo implica em algum tipo de ordenação.

valor atômico Valores não podem ser divididos em componentes. Atributos compostos e multivalorados não são permitidos. Isto é chamado **primeira forma normal**.

valores nulos Indicam que o valor de um atributo é desconhecido ou não se aplica a uma tupla.

☞ Para você fazer

Uma locadora de DVDs Descreva em uma lista quais seriam as entidades e atributos de cada entidade capazes de suportar a operação comercial de locadora de DVDs. Assuma operações normais, a saber:

1. Comprar e divulgar DVDs para aluguel.
2. Angariar clientes e seus dependentes, classificando-os conforme o movimento com a empresa
3. propor promoções a clientes fiéis e/ou a DVDs "micados".
4. controlar o acervo, cobrando perdas e atrasos
5. permitir reservas, avisar disponibilidades
6. fechar uma pequena contabilidade informal
7. vender DVDs após um prazo razoável em que ele foi alugado

Não se preocupe em fazer modelos, apenas descreva cada tabela e suas colunas. Garanta que os dados levantados suportam adequadamente todas as operações previstas para a organização Responda em uma folha anexa a esta, ou se preferir no verso desta folha. 1