

MODELO RELACIONAL

O modelo de dados Relacional foi introduzido inicialmente por Ted Cold da IBM Research em 1990, que atraiu atenção imediata devido a sua simplicidade e base matemática. O modelo relacional surgiu devido a algumas necessidades, como: 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; permitir processamento *ad hoc* (Processamento dedicado, exclusivo.). Este Modelo seria mais flexível e apropriado ao solucionar os vários problemas que se colocam no nível da compreensão e implementação da base de dados.

É um modelo padrão usado para aplicações comerciais, devido à sua simplicidade e desempenho.

Um dos SGBD's precursores que implementaram este modelo foi o System R (IBM). Baseado em seus conceitos surgiram: DB2 (IBM), SQL-DS (IBM), Oracle, Informix, Ingres, Sybase entre outros.

O modelo relacional representa os dados num BD como uma coleção de tabelas (relações). Podemos dizer então que um banco de dados relacional é composto por um conjunto de relações. A tabela é composta por linhas (tuplas) e colunas (atributos),

Onde todos os valores de uma coluna são do mesmo tipo de dados. A tuplas são as linhas existentes na tabela, e, portanto não podem existir tuplas duplicadas em uma tabela. Os atributos são representados por cada uma das colunas da tabela. Estão definidas por um nome e podem conter um conjunto de valores.

Em um modelo Relacional existem dois tipos de domínios que são um conjunto de valores que podem tomar um atributo.

A cada domínio está associado um tipo de dados ou formato.

Ex.: Fone: conjunto de 10, dígitos

CPF: conjunto de 7 dígitos

Idade_Empregado: $16 \leq \text{idade} \leq 70$

Departamentos: conjunto de

departamentos de uma empresa.

Domínios gerais: são aqueles que estão compreendidos entre um máximo e um mínimo.

Domínios restringidos: são os que pertencem a um conjunto de valores específicos.

Cada tupla (linha) de uma tabela deverá estar associada a uma chave única que permita identificá-la. A chave poderá ser composta por um ou mais atributos, e esta tem que ser única dentro de sua tabela e não poderá descartar nenhum atributo da mesma para identificar uma tupla.

Existem dois tipos de chaves:

- Chave primária (Primary Key): é o valor ou conjunto de valores que identificam uma fila dentro de uma tabela. Nunca pode ser NULL. Um exemplo claro de chave primária seria o RG, que é único para cada pessoa e não pode ser NULL.
- Chave alheia (Foreign Key): é o valor ou valores de uma tabela que corresponde com o valor de uma chave primária em outra tabela. Esta chave é a que representa as relações entre as tabelas.

MODELO HIERÁRQUICO

O Modelo de Dados Hierárquico surgiu na década de 1960 com a primeira linguagem de banco de dados: a DL/I desenvolvida pela IBM e a North American Aviation.

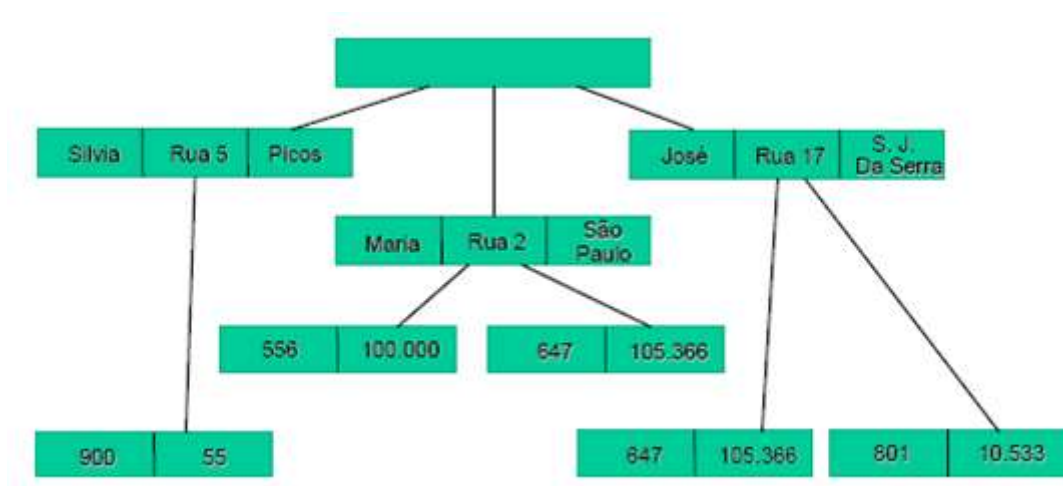
Este modelo organiza os dados de cima para baixo, como uma árvore e é definido como uma coleção de registros conectados por meio de ligações, onde cada registro é uma coleção de campos e cada campo contém um único valor. Cada ligação pode ligar exatamente dois registros. Cada registro é dividido em partes denominadas segmentos. Os segmentos são arranjados em estruturas com um segmento superior ligado a um segmento subordinado em um relacionamento “pai-filho”. Um segmento “pai” pode ter mais de um “filho”, mas um segmento “filho” só pode ter um “pai”. A estrutura da árvore é difícil de manter quando o volume de dados aumenta. Sistema de banco de dados, portanto, hierárquico tiveram que ser descartados quando o trabalho foi para armazenar grandes volumes de dados já que a probabilidade de corrupção de dados e perda foi elevada em tal sistema. Muitos-para-um ou muitos-para-muitos relacionamentos se tornam difíceis de ser implementadas através do relacionamento entre pais e filhos nos casos em que existe um nó de filho que pode ter mais de um nó pai.

Dessa forma, o conteúdo de um registro particular pode ter que ser replicado em vários locais diferentes. A réplica de registro possui duas grandes desvantagens: pode causar inconsistência de dados quando houver atualização, e o desperdício de espaço é inevitável.

Usa-se um diagrama de estrutura de árvore para apresentar o esquema para uma base de dados hierárquica. O diagrama da estrutura de uma árvore consiste de dois componentes básicos que são

caixas (que correspondem ao tipo registro) e linhas (que correspondem às ligações). Seu propósito é especificar a estrutura lógica geral da base de dados.

Apresenta equivalência com diagramas ER, não podem existir ciclos entre os nós (registros) e cardinalidade 1:1 ou 1:n entre pai e filho.



MODELO DE REDE

Os primeiros trabalhos usando este modelo foi em 1964 por Charles Bachman. Esse modelo é uma extensão do modelo hierárquico. Os dados são representados por uma coleção de registros e os relacionamentos por meio de links. É representado por um diagrama constituído por caixas e linhas.

Cada registro filho pode ser ligado a mais de um registro pai criando conexões bastante complexas e são bastante utilizados em sistemas para computadores de grande porte. Em outras palavras, um “filho” pode ter mais de um “pai”. Esta metodologia torna a pesquisa mais rápida e mais flexível, pois não depende de um único nó raiz como vetor de inicialização de pesquisa.

.. Este modelo é composto de uma estrutura mais completa, possui as propriedades básicas de registros, conjuntos e ocorrências, e utiliza a linguagem de definição de BD (DDL) e a linguagem de manipulação de dados (DML), além de permitir evolução mais eficiente do modelo. A estrutura é formada de entidade (registros), atributos (itens de dados), tipo de registro e ocorrência do registro.

Entretanto, o modelo em rede ainda apresenta os mesmos problemas com relação ao projeto de estrutura do modelo hierárquico. Qualquer alteração feita em uma classe de dados implica na criação de uma nova estrutura para suportar aquela alteração. No modelo em rede um dos sistemas mais conhecidos é o CA IDMS da Computer Associates.

Programas de banco de dados são projetados principalmente para que eles são capazes de armazenar e manipular informações. No entanto, a eficiência de um banco de dados depende de como ele pode relacionar fluentemente entre as diferentes unidades de informação, seja no banco de dados ou entre vários bancos de dados. Os dados presentes em um banco de dados precisa estar disponível para os dados de outro banco de dados, a fim de manter a continuidade.