**Aluno:** Bruno Camargo Manso

1. O modelo Conceitual:

Tem a característica de ser independente do *Sistema de Gerenciamento de Banco de Dado*s, responsáveis pela incorporação de funções de definição, alteração e recuperação de dados em um determinado inventário (Oracle, MySQL, SQL Server, PostgreSQL). Assim, o *Modelo Conceitual* não define como esses dados aparecerão no registro, é abstrato e descreve a estrutura de um banco de dados. Tal modelo fora estabelecido órgão American National Standards Institute (ANSI) definiu parâmetros para o desenvolvimento de tecnologias de banco de dados em 3 meios: Interno, conceitual e externo. Sendo o último com enfoque principalmente nos usuários finais, ou aqueles que utilizam aplicativos em geral, assim promovendo a devida manipulação dos dados, que por fim vão gerar dados relevantes a uma entidade. O *Método de Entidade-Relacionamento* ou M.E.R é o modelo usualmente mais praticado. Criado por Peter Chen (1970), é representado pelo D.E.R ou *Diagrama de Entidade-Relacionamento*, possibilita ter uma visão do ‘todo’, porém de maneira simplificada e de alto nível de abstração. Tal modelo sequer precisa de hardware de um servidor de um banco de dados próprio, Tem objetivo de criar entidades e com quem elas relacionam,sempre abstraído de um ambiente empresarial real e assim descrever como os dados serão manipulados por um computador.

1. O Modelo Lógico:

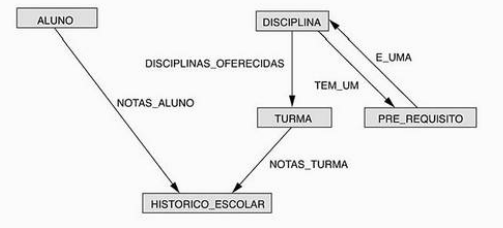
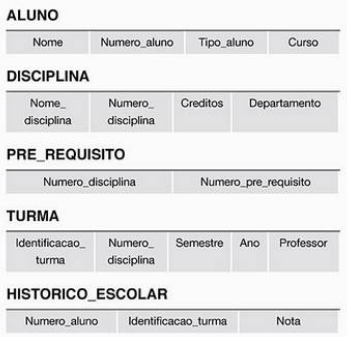
É o oposto do modelo conceitual por depender diretamente de um SGBD para criação de um banco de dados, assim ele consegue descrevê-lo com base em um modelo abstrato. O papel de um projetista, então, será decidir as tabelas relacionais, depois defini-las seguido de suas linhas e colunas que contêm nome de grupos de variáveis e suas atribuições. Tais dados podem ser observados pelo usuário do SGBD como forma de armazenamento interno.

1. O Modelo Hierárquico:

Representa os dados como estruturas em forma de árvores, sendo que cada hierarquia representa uma série de registros relacionados. Tal modelo não requer uma linguagem padrão. Por muitos anos foi utilizado uma DML (Data Manipulation Language) chamada DL/1, que dominou o mercado entre 1965 a 1985 e ainda permanece um sistema utilizado no mundo todo pelo setor de seguros, bancos de dados governamentais, serviços de saúde. Tal DML foi padrão na indústria por longos anos .

1. O Modelo em Rede:

Representam dados com tipos de registro mas também representam dados de um tipo limitado de relacionamento. Também conhecido como CODASYL DBTG, possui uma linguagem que trata um registro por vez, também deve estar contida em uma linguagem de programação hospedeira. O Data Base Task Group em 1971 propôs uma extensão do COBOL para aplicação do modelo. Para ilustrar seu funcionamento, vejamos o modelo abaixo, retirado do livro “Sistemas de Banco de Dados” de Ramez Elmasri e Shamkant B. Navathe:



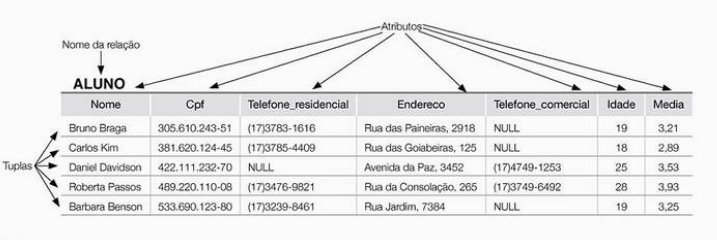
Outro importante SGB criado foi pela empresa IBM (VSAM), chamados IMS funcionavam baseados em modelos de rede, que são até hoje usado em algumas instalações governamentais e industriais, bem como hospitais, bancos etc, apesar de boa parte dos usuários têm migrado para outras plataformas relacionais ao longo dos anos. Marcas como Compaq e HP utilizam tal metodologia (VAX-SGBD).

1. O Modelo Orientado a Objetos :

É um modelo que define um grupo de dados priorizando seus objetos, com suas devidas propriedades e operações. Objetos com a mesma estrutura pertencem à mesma classe relacional, classes essas que são organizadas hierarquicamente ou também chamados de *grafos acíclicos*. Cada classe específica determinadas por procedimentos específicos e pré definidos chamados métodos. A maioria das linguagens de programação vem adotando o modelo orientado a objetos estendido a banco de dados. Linguagem como o Python ou Julia que conseguem processar grandes volumes de dados, com ferramentas que se relacionam como um passe de mágica com outras formas de banco de dados.

1. Modelo de Dados Relacional

Introduzido por Ted Codd da IBM Research em 1970, utiliza-se do conceito de *relação* em matemática, parecida com uma tabela de valores, se baseia em conjuntos e lógica de predicado de primeira ordem, sendo assim um conjunto de relações. Tudo é gravado em um plano de registro. Na tabela de valores cada linha representa uma relação de valores de dados relacionados, normalmente ali se localizam entidades ou relacionamento do mundo real. Os nomes da tabela, em suas colunas ajudam a interpretar o significado dos valores da linha. no exemplo do quadro acima temos o aluno e cada linha representa fatos sobre essa entidade particular. Nome, número, tipo e curso norteiam como interpretar os valores de dados em cada linha, sendo que os valores em uma coluna são do mesmo tipo de dado. Uma linha então é chamada de *tupla*, seu *cabeçalho* de atributo e a tabela de *relação.* O que descreve os tipos de valores são chamados de *domínios.*



1. Modelo de Dados SQL

O Structure Query Language, originalmente chamada de SEQUEL foi criada e implementada pela IBM Research como interface de um sistema de banco de dados chamado de System R. Foi o modelo que consagrou o sucesso dos bancos de dados relacionais, assim diminuíram os temores sobre migração(por exemplo entre sistemas de rede e os hierárquicos) para os sistemas relacionais, tais novos sistemas seguiram então o mesmo padrão linguístico.Um usuário pode escrever comandos em um programa de aplicação de banco de dados que podem ser armazenados em duas diferentes SGBDs relacionais sem precisar alterar sua sublinguagem de banco de dados. Possui regras para embutir instruções SQL em linguagens como Java, COBOL ou C/C++. É uma linguagem abrangente, com instruções para definição de dados, consultas e atualizações. Também usa tabelas, linhas e colunas, assim como relações, tuplas e atributos respectivamente ligados às estruturas citadas. Oferece uma linguagem declarativa de nível alto, de forma que o usuário especifica qual deve ser o resultado. É também baseado em cálculos relacionais porém dando destaque aos cálculos relacional de tupla, todavia com uma sintaxe bem mais simplificada.

1. Existem uma grande gama de ferramentas que tem objetivo de modelagem de dados, desde a produção de diagramas e suas abstrações até a criação de um banco.

Começando pelos softwares de criação de diagramas:

* Draw.io (ferramenta online - free)
* LibreOffice Drawing (parte do pacote LibreOffice, open source)
* LucydChart
* BPMN.io (outra ferramenta online de criação de diagramas)

Os Softwares que tratam diretamente de dados estão presentes nos principais sistemas operativos, entre eles

destacam-se:

* MySQL Benchwork (versão open source de criação e edição de banco de dados)
* PostGreSQL
* Oracle’s Designer
* IBM Relational Rose
* OBDesigner

1. Cardinalidade refere-se a multiplicidade de uma entidade ou mais, aplicado a um relacionamento qualquer. nos diagramas as cardinalidades são representadas por linhas que realizam conexões entre tais entidades. Considera-se como uma restrição existente entre as entidades. O uso de cardinalidade pode ser máxima e mínimas, sendo possível aplicar cardinalidades em relacionamentos binários. Os tipos de cardinalidade podem ser de um pra um , por exemplo um empregado que gerencia apenas uma departamento. o segundo tipo de cardinalidade é quando um funcionário gerencia apenas um departamento, porém este pode ser gerenciado por muitos empregados. O terceiro tipo refere-se a muitos para muitos, considerando um departamento que pode ser gerenciado por vários funcionários.