

BANCO DE DADOS

CLAUDIO RIBEIRO DA SILVA
MARÇO 1996

1 - CONCEITOS GERAIS DE BANCO DE DADOS

1.1 - Conceitos

BANCO DE DADOS - É o arquivo físico, em dispositivos periféricos, onde estão armazenados os dados de diversos sistemas, para consulta e atualização pelo usuário.

S.G.D.B. (SISTEMA GERENCIADOR DE BANCO DE DADOS) - É o software responsável pelo gerenciamento (armazenamento e recuperação) dos dados no Banco de Dados.

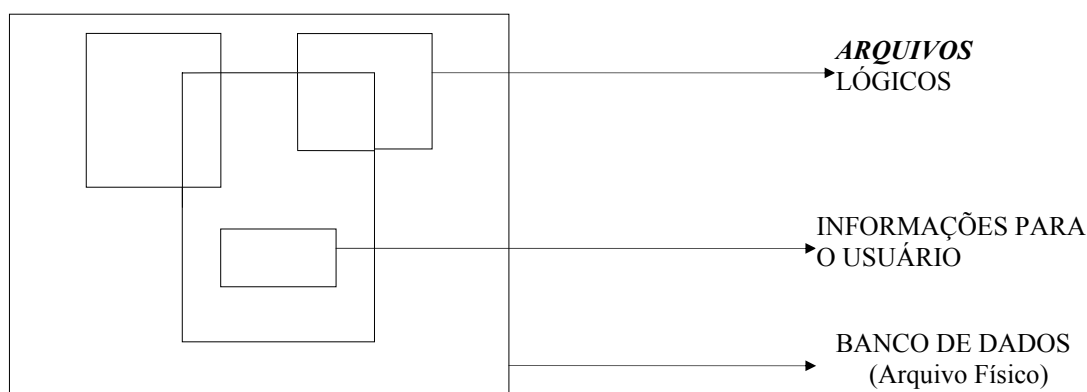
DADO - É o valor do campo quando é armazenado no Banco de Dados. Ex. O valor do campo "nome do cliente" para quem está fazendo a entrada de dados.

CONTEÚDO DO CAMPO - É o valor do campo armazenado no Banco de Dados. Ex. O valor do campo "nome do cliente" sem ser, momentaneamente, utilizado, armazenado na Base de Dados.

INFORMAÇÃO - É o valor que este campo representa para as atividades da empresa. Ex. Resposta a uma consulta. Qual os nomes dos clientes localizados no Rio de Janeiro?

MODELOS DE BANCO DE DADOS - Modelo Relacional, Modelo Hierárquico e Modelo em Rede. Representa a estrutura física no qual o armazenamento dos dados foram projetados. O modelo identifica a estrutura interna de recuperação e armazenamento dos dados no qual o SGBD foi projetado.

1.2 - Representação Física do Banco de Dados



1.3 - Visões do Banco de Dados

- a - **VISÃO INTERNA** - É aquela vista pelo responsável pela manutenção e desenvolvimento do SGBD. Existe a preocupação com a forma de recuperação e manipulação dos dados dentro do Banco de Dados.
- b - **VISÃO CONCEITUAL** - É aquela vista pelo analista de desenvolvimento e pelo administrador das bases de dados. Existe a preocupação na definição de normas e procedimentos para manipulação dos dados, para garantir a sua segurança e confiabilidade, o desenvolvimento de sistemas e programas aplicativos e a definição no banco de dados de novos arquivos e campos. Na visão conceitual, existem 2 (duas) linguagens de operação que são:
- Linguagem de definição dos dados (DDL) - Linguagem que define as aplicações, arquivos e campos que irão compor o banco de dados (comandos de criação e atualização da estrutura dos campos dos arquivos).
 - Linguagem de manipulação dos dados (DML) - Linguagem que define os comandos de manipulação e operação dos dados (comandos de consulta e atualização dos dados dos arquivos).
- c - **VISÃO EXTERNA** - É aquela vista pelo usuário que opera os sistemas aplicativos, através de interfaces desenvolvidas pelo analista (programas), buscando o atendimento de suas necessidades.

UTILIZAÇÃO DAS APLICAÇÕES DESENVOLVIDAS	VISÃO EXTERNA
BANCO DE DADOS	Claudio Ribeiro da Silva

DESENVOLVIMENTO DE APLICAÇÕES

VISÃO CONCEITUAL

UTILIZANDO RECURSOS DO S.G.B.D.

DESENVOLVIMENTO DO S.G.B.D

VISÃO INTERNA

1.4 - Vantagens do Banco de Dados em relação à arquitetura tradicional

1.4.1 - DEFINIÇÕES

SISTEMA TRADICIONAL - São aqueles em que os dados do sistema estão armazenados fisicamente separados um do outro. O acesso é feito pelos programas de aplicação, associando o nome externo dos arquivos e definindo todo o registro independente da utilização dos campos.

SISTEMA DE BANCO DE DADOS - É aquele em que os dados são definidos para o S.G.B.D., através da DDL (linguagem de definição de dados). Fisicamente estão armazenados em um único local, sendo o acesso realizado apenas através do S.G.B.D. Nos programas de aplicação, é necessário apenas definir os campos que serão utilizados pelo programa.

1.4.2 - VANTAGENS DO BANCO DE DADOS

- 1 - **REDUÇÃO OU ELIMINAÇÃO DE REDUNDÂNCIAS** - Possibilita a eliminação de dados privativos de cada sistema. Os dados, que eventualmente são comuns a mais de um sistema, são compartilhados por eles, permitindo o acesso a uma única informação sendo consultada por vários sistemas.
- 2 - **ELIMINAÇÃO DE INCONSISTÊNCIAS** - Através do armazenamento da informação em um único local com acesso descentralizado e, sendo compartilhada à vários sistemas, os usuários estarão utilizando uma informação confiável. A inconsistência ocorre quando um mesmo campo tem valores diferentes em sistemas diferentes. Exemplo, o estado civil de uma pessoa é solteiro em um sistema e casado em outro. Isto ocorre porque esta pessoa atualizou o campo em um sistema e não o atualizou em outro. Quando o dado é armazenado em um único local e compartilhado pelos sistemas, este problema não ocorre.
- 3 - **COMPARTILHAMENTO DOS DADOS** - Permite a utilização simultânea e segura de um dado, por mais de uma aplicação ou usuário, independente da operação que esteja sendo realizada. Deve ser observada apenas o processo de atualização concorrente, para não gerar erros de processamento (atualizar simultaneamente o mesmo campo do mesmo registro). Os aplicativos são por natureza multiusuário.
- 4 - **RESTRICÇÕES DE SEGURANÇA** - Define para cada usuário o nível de acesso a ele concedido (leitura, leitura e gravação ou sem acesso) ao arquivo e/ou campo. Este recurso impede que pessoas não autorizadas utilizem ou atualizem um determinado arquivo ou campo.
- 5 - **PADRONIZAÇÃO DOS DADOS** - Permite que os campos armazenados na base de dados sejam padronizados segundo um determinado formato de armazenamento (padronização de tabela, conteúdo de campos, etc) e ao nome de variáveis seguindo critérios padrões pré-estabelecido pela empresa. Ex. Para o campo "Sexo" somente será permitido armazenamento dos conteúdos "M" ou "F".
- 6 - **MANUTENÇÃO DE INTEGRIDADE** - Exige que o conteúdo dos dados armazenadas no Banco de Dados possuam valores coerentes ao objetivo do campo, não permitindo que valores absurdos sejam cadastrados. Exemplo: Um funcionário que faça no mês 500 horas extras, ou um aluno que tenha nascido no ano de 1860.

-
- 7 - EVITAR NECESSIDADES CONFLITANTES - Representa a capacidade que o administrador de Banco de Dados deve ter para solucionar "prioridades sempre altas" de todos os sistemas, tendo ele que avaliar a real necessidade de cada sistema para a empresa para priorizar a sua implantação.
- 8 - INDEPENDÊNCIA DOS DADOS - Representa a forma física de armazenamento dos dados no Banco de Dados e a recuperação das informações pelos programas de aplicação. Esta recuperação deverá ser totalmente independente da maneira com que os dados estão fisicamente armazenados. Quando um programa retira ou inclui dados o SGBD compacta-os para que haja um menor consumo de espaço no disco. Este conhecimento do formato de armazenamento do campo é totalmente transparente para o usuário. A independência dos dados permite os seguintes recursos:
- a - Os programas de aplicação definem apenas os campos que serão utilizados independente da estrutura interna dos arquivos
 - b - Quando há inclusão de novos campos no arquivo, será feita manutenção apenas nos programas que utilizam esses campos, não sendo necessário mexer nos demais programas. Obs: Nos sistemas tradicionais este tipo de operação requer a alteração no lay-out de todos os programas do sistema que utilizam o arquivo.

2 - ORGANIZAÇÃO DE ARQUIVOS

É a forma com que os registros estão fisicamente organizados e armazenados em um arquivo, possibilitando diferentes modos de acessos e tratamento por parte da aplicação. Os tipos de organizações disponíveis, bem como as maneiras como podem ser acessadas estão descritas abaixo:

2.1 - Seqüencial

Tipo de organização em que os registros estão armazenados lado a lado. O único modo de acesso possível é o seqüencial, ou seja, para acessarmos o registro N, temos que fazer N - 1 acessos. A única técnica utilizada para atualizar um arquivo seqüencial é a técnica de balanced-line.

2.2 - Indexado

Tipo de organização em que os registros são identificados por um índice, chamado de chave do acesso ou chave do registro ou chave principal, que representa o valor com que as demais informações são identificadas no arquivo. Esta chave de acesso deverá ser única para cada registro, ou seja, não é possível ter dois registros com o mesmo valor de chave.

2.2.1 - ESTRUTURA INTERNA

O arquivo de organização indexada possui duas áreas:

1) Área de Índices - Contém a chave de acesso e o endereço de cada registro na área de dados. Todo acesso aos registros será feito através desta área, que é mantida sempre ordenada, pela chave de acesso, pelo Sistema Operacional.

2) Área de Dados - Armazena os demais campos de todos os registros do arquivo, independente de qualquer ordenação ou seqüência.

2.2.2 - MODO DE ACESSO NA ORGANIZAÇÃO INDEXADA

- Seqüencial - É feito através da leitura seqüencial da área de índices do arquivo. Para cada ocorrência desta área (equivalente a um índice), será obtido o endereço físico do registro deste índice na área de dados e com este endereço lido o registro. Na recuperação seqüencial de um arquivo, os dados serão lidos sempre ordenados pela chave.

- Aleatório ou Randômico - Caracteriza-se por não existir uma seqüência lógica na recuperação dos registros, ou seja, após a leitura do último registro, pode ser lido o quinto ou primeiro, ou qualquer outro, uma vez que, esta seqüência é determinada pelo programa de aplicação. No processo de leitura, o Sistema Operacional localiza o valor da chave desejada na área de índices, identificando o endereço do registro correspondente na área de dados, possibilitando o acesso, por este endereço, para a recuperação do registro.

2.2.3 - PRINCIPAIS OPERAÇÕES PARA ATUALIZAÇÃO DOS DADOS

Inclusão - O Sistema Operacional localiza na área de índice o local onde será armazenado a chave a ser incluída, reorganizando se necessário as demais chaves mantendo-as sempre ordenadas. Após este processo, ele procura na área de dados, um endereço disponível para armazenamento do registro. Identificado o endereço, o registro será incluído e o valor do seu endereço, atualizado na área de índices.

Alteração - O Sistema Operacional localiza na área de índices, o valor da chave de acesso que se deseja atualizar, identificando o endereço do registro na área de dados. Após esta operação, ele atualiza os dados do registro no endereço correspondente.

Exclusão - O Sistema Operacional remove fisicamente da área de índices a chave de acesso do registro que será deletado e realiza uma exclusão lógica do registro na área de dados.

3.3 - Seqüencial Indexado

É o tipo de organização em que, semelhante à organização indexada, os registros são identificados por um índice, chamado de chave do acesso ou chave do registro ou chave principal, que representa o valor com que as demais informações do registro será identificado no arquivo. Esta chave de acesso deverá ser única para cada registro, ou seja, não é possível ter dois registros com o mesmo valor de chave.

3.3.1 - ESTRUTURA INTERNA

O arquivo de organização seqüencial-indexada possui duas áreas:

1) Área de Índices - Contém o valor da maior chave de acesso existente em cada bloco de registros da área de dados e o endereço deste bloco nesta área.

2) Área de Dados - Contém, agrupados em blocos, os registros que compõem o arquivo, sendo mantidos ordenados dentro de cada um desses blocos.

3.3.2 - MODOS DE ACESSO NA ORGANIZAÇÃO SEQÜENCIAL INDEXADA

- Seqüencial - É feito através da leitura seqüencial dos dados na área de dados.

- Aleatório ou Randômico - É feito a localização, através do valor da chave de acesso na área de índices, do bloco onde deverá estar armazenado o registro desejado, identificando o seu endereço na área de dados. Após a localização do bloco nesta área, é feita uma procura seqüencial nos registros nele existentes até a localização do registro desejado.

3.3.3 - PRINCIPAIS OPERAÇÕES PARA ATUALIZAÇÃO DOS DADOS

Inclusão - O Sistema Operacional localiza na área de índice o bloco onde será armazenado a chave a ser incluída. Após esta operação ele localiza na área de dados o bloco desejado e inclui o registro na posição correta, isto é, ordenado.

Alteração - O Sistema Operacional localiza na área de índices o endereço do bloco onde poderá estar armazenado a chave desejada. Localizando-o, na área de dados, é feita a procura seqüencial do registro a ser atualizado.

Exclusão - O Sistema Operacional remove fisicamente da área de dados o registro desejado, após a localização do bloco na área de índices.

3.4 - Relativa ou Direta

É o tipo de organização que contém, apenas, uma área de dados para armazenamento dos registros. Estes registros são identificados por uma chave principal, cujo endereço físico para armazenamento é dado através do valor desta chave ou de um valor calculado a partir dela. Um arquivo de organização relativa tem que possuir chave de conteúdo numérico, para possibilitar o cálculo do endereço e localização do registro na área de dados.

Para encontrar o endereço de um registro o Sistema Operacional utiliza uma função que, através de um algoritmo qualquer, transforma o valor da chave em um endereço físico, onde está armazenado o registro solicitado. Eventualmente, esta função gera para dois ou mais valores de chave diferentes o mesmo endereço físico. Neste caso, o primeiro registro incluído ficará na área de dados e os demais em uma área auxiliar, chamada área de colisão, interligados através de uma lista simples. A área de colisão é uma área de dados auxiliar, onde fica armazenado todos os registros cujo cálculo do endereço é igual a de outros já armazenados.

3.4.1 - MODOS DE ACESSO NA ORGANIZAÇÃO RELATIVA

- Seqüencial - É feito através da leitura seqüencial dos dados na área de dados, obedecendo a seqüência das chaves.
- Aleatório - É feito a localização do registro através do endereço físico calculado através de uma função a partir do valor da chave de acesso. Caso haja mais de um registro para o mesmo endereço, o Sistema Operacional fará pesquisa seqüencial dos registro na área de colisão do arquivo.

3.4.2 - PRINCIPAIS OPERAÇÕES PARA ATUALIZAÇÃO DOS DADOS

Inclusão - O Sistema Operacional calcula o endereço do registro, através do valor de sua chave, incluindo-o no local especificado, caso já exista registros neste endereço ele fará a inclusão na área de colisão.

Alteração - O Sistema Operacional calcula o endereço do registro, através do valor de sua chave, atualizando-o no local especificado ou na área de colisão, dependendo de onde ele esteja armazenado.

Exclusão - O Sistema Operacional calcula o endereço do registro, através do valor de sua chave, removendo-o logicamente da área de dados ou da área de colisão, dependendo de onde ele esteja armazenado.

3.5 - Invertido

É o tipo de organização que permite o acesso direto aos registros de dados através de chaves secundária (chave que não é primária), de um arquivo de dados. Esta arquivo é formado pelo conteúdo das chaves secundária e da relação das chaves primárias dos registros que a contém.

Exemplo:

Arquivo Principal: Funcionário

Chave primária ou principal - Cod. funcionário

COD. FUNCIONÁRIO	NOME	SALÁRIO
10	A	110.000
15	D	150.000
20	E	180.000
23	C	110.000
28	F	180.000
32	B	150.000
35	G	180.000

Arquivo invertido por salário:

SALÁRIO	CHAVES PRIMÁRIA
110.000	10, 23
150.000	15, 32
180.000	20, 28, 35

Este tipo de organização auxilia a recuperação de registros através de uma consulta, sendo bastante utilizado pelos Sistemas de Gerenciamento de Banco de Dados (SGBD) existentes. Ele está sempre associado a um arquivo principal, onde está armazenado o registro completo.

O processo de atualização de um arquivo invertido ocorre simultaneamente à do arquivo principal, devendo o mesmo ser feito pela própria aplicação ou pelo SGBD.

3 - NORMALIZAÇÃO DE DADOS

3.1 - Definição

Consiste em definir o formato lógico adequado para as estruturas de dados identificados no projeto lógico do sistema, com o objetivo de minimizar o espaço utilizado pelos dados e garantir a integridade e confiabilidade das informações.

A normalização é feita, através da análise dos dados que compõem as estruturas utilizando o conceito chamado "Formas Normais (FN)". As FN são conjuntos de restrições nos quais os dados devem satisfazê-las. Exemplo, pode-se dizer que a estrutura está na primeira forma normal (1FN), se os dados que a compõem satisfizerem as restrições definidas para esta etapa.

A normalização completa dos dados é feita, seguindo as restrições das quatro formas normais existentes, sendo que a passagem de uma FN para outra é feita tendo como base o resultado obtido na etapa anterior, ou seja, na FN anterior.

Para realizar a normalização dos dados, é primordial que seja definido um campo chave para a estrutura, campo este que permite ir identificar os demais campos da estrutura. Formas Normais existentes:

3.2 - Primeira Forma Normal (1FN)

Consiste em retirar da estrutura os elementos repetitivos, ou seja, aqueles dados que podem compor uma estrutura de vetor. Podemos afirmar que uma estrutura está normalizada na 1FN, se não possuir elementos repetitivos. Exemplo:

Estrutura original:

Arquivo de Notas Fiscais (Num. NF, Série, Data emissão, Cod. do Cliente, Nome do cliente, Endereço do cliente, CGC do cliente, Relação das mercadorias vendidas (onde para cada mercadoria temos: Código da Mercadoria, Descrição da Mercadoria, Quantidade vendida, Preço de venda e Total da venda desta mercadoria) e Total Geral da Nota)

Analisando a estrutura acima, observamos que existem várias mercadorias em uma única Nota Fiscal, sendo portanto elementos repetitivos que deverão ser retirados.

Estrutura na primeira forma normal (1FN):

Arquivo de Notas Fiscais (Num. NF, Série, Data emissão, Código do Cliente, Nome do cliente, Endereço do cliente, CGC do cliente e Total Geral da Nota)

Arquivo de Vendas (Num. NF, Código da Mercadoria, Descrição da Mercadoria, Quantidade vendida, Preço de venda e Total da venda desta mercadoria)

Obs. Os campos sublinhados identificam as chaves das estruturas.

Como resultado desta etapa ocorre um desdobramento dos dados em duas estruturas, a saber:

- Primeira estrutura (Arquivo de Notas Fiscais): Dados que compõem a estrutura original, excluindo os elementos repetitivos.

- Segunda estrutura (Arquivo de Vendas): Dados que compõem os elementos repetitivos da estrutura original, tendo como chave o campo chave da estrutura original (Num. NF) e o campo chave da estrutura de repetição (Código da Mercadoria).

3.3 - Segunda Forma Normal (2FN)

Consiste em retirar das estruturas que possuem chaves compostas (campo chave sendo formado por mais de um campo), os elementos que são funcionalmente dependente de parte da chave. Podemos afirmar que uma estrutura está na 2FN, se ela estiver na 1FN e não possuir campos que são funcionalmente dependente de parte da chave. Exemplo:

Estrutura na primeira forma normal (1FN):

Arquivo de Notas Fiscais (Num. NF, Série, Data emissão, Código do Cliente, Nome do cliente, Endereço do cliente, CGC do cliente e Total Geral da Nota)

Arquivo de Vendas (Num. NF, Código da Mercadoria, Descrição da Mercadoria, Quantidade vendida, Preço de venda e Total da venda desta mercadoria)

Estrutura na segunda forma normal (2FN):

Arquivo de Notas Fiscais (Num. NF, Série, Data emissão, Código do Cliente, Nome do cliente, Endereço do cliente, CGC do cliente e Total Geral da Nota)

Arquivo de Vendas (Num. NF, Código da Mercadoria, Quantidade vendida e Total da venda desta mercadoria)

Arquivo de Mercadorias (Código da Mercadoria, Descrição da Mercadoria, Preço de venda)

Como resultado desta etapa, houve um desdobramento do arquivo de Vendas (o arquivo de Notas Fiscais, não foi alterado, por não possuir chave composta) em duas estruturas a saber:

- Primeira estrutura (Arquivo de Vendas): Contém os elementos originais, sendo excluídos os dados que são dependentes apenas do campo Código da Mercadoria.

- Segundo estrutura (Arquivo de Mercadorias): Contém os elementos que são identificados apenas pelo Código da Mercadoria, ou seja, independentemente da Nota Fiscal, a descrição e o preço de venda serão constantes.

3.4 - Terceira Forma Normal (3FN)

Consiste em retirar das estruturas os campos que são funcionalmente dependentes de outros campos que não são chaves. Podemos afirmar que uma estrutura está na 3FN, se ela estiver na 2FN e não possuir campos dependentes de outros campos não chaves. Exemplo:

Estrutura na segunda forma normal (2FN):

Arquivo de Notas Fiscais (Num. NF, Série, Data emissão, Código do Cliente, Nome do cliente, Endereço do cliente, CGC do cliente e Total Geral da Nota)

Arquivo de Vendas (Num. NF, Código da Mercadoria, Quantidade vendida e Total da venda desta mercadoria)

Arquivo de Mercadorias (Código da Mercadoria, Descrição da Mercadoria, Preço de venda)

Estrutura na terceira forma normal (3FN):

Arquivo de Notas Fiscais (Num. NF, Série, Data emissão, Código do Cliente e Total Geral da Nota)

Arquivo de Vendas (Num. NF, Código da Mercadoria, Quantidade vendida e Total da venda desta mercadoria)

Arquivo de Mercadorias (Código da Mercadoria, Descrição da Mercadoria, Preço de venda)

Arquivo de Clientes (Código do Cliente, Nome do cliente, Endereço do cliente e CGC do cliente)

Como resultado desta etapa, houve um desdobramento do arquivo de Notas Fiscais, por ser o único que possuía campos que não eram dependentes da chave principal (Num. NF), uma vez que independente da Nota Fiscal, o Nome, Endereço e CGC do cliente são inalterados. Este procedimento permite evitar inconsistência nos dados dos arquivos e economizar espaço por eliminar o armazenamento frequente e repetidas vezes destes dados. A cada nota fiscal comprada pelo cliente, haverá o armazenamento destes dados e poderá ocorrer divergência entre eles.

As estruturas alteradas foram pelos motivos, a saber:

- Primeira estrutura (Arquivo de Notas Fiscais): Contém os elementos originais, sendo excluído os dados que são dependentes apenas do campo Código do Cliente (informações referentes ao cliente).

- Segundo estrutura (Arquivo de Clientes): Contém os elementos que são identificados apenas pelo Código do Cliente, ou seja, independente da Nota Fiscal, o Nome, Endereço e CGC dos clientes serão constantes.

Após a normalização, as estruturas dos dados estão projetadas para eliminar as inconsistências e redundâncias dos dados, eliminando desta forma qualquer problema de atualização e operacionalização do sistema. A versão final dos dados poderá sofrer alguma alteração, para atender as necessidades específicas do sistema, a critério do analista de desenvolvimento durante o projeto físico do sistema.

4 - MODELO DE ENTIDADE E RELACIONAMENTO (MER)

4.1 - Definição

Consiste em mapear o mundo real do sistema em um modelo gráfico que irá representar o modelo e o relacionamento existente entre os dados.

ENTIDADE - Identifica o objeto de interesse do sistema e tem "vida" própria, ou seja, a representação abstrata de um objeto do mundo real sobre o qual desejamos guardar informações.

Exemplo: Clientes, Fornecedores, Alunos, Funcionários, Departamentos, etc.

Não são entidades:

- Entidade com apenas 1 elemento;
- Operações do sistema;
- Saídas do sistema;
- Pessoas que realizam trabalhos (usuários do sistema);
- Cargos de direção

INSTÂNCIA DE ENTIDADE - São os elementos da entidade.

Exemplo: Cliente 10, Funcionário João, Aluno Pedro, etc.

ATRIBUTO - Informações que desejamos guardar sobre a instância de entidade.

Exemplo: Nome do aluno, Número da turma, Endereço do fornecedor, Sexo do funcionário, etc.

DOMÍNIO DO ATRIBUTO - Universo de valores que um atributo pode armazenar.

Exemplo:

Conjunto de valores do atributo Sexo do funcionário: M ou F;

Conjunto de valores do atributo Nome do aluno: 40 caracteres alfanumérico.

Conjunto de valores do atributo salário: inteiro maior que 5000

4.2 - Representação Gráfica

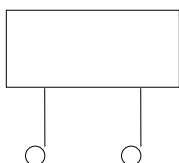
- ENTIDADE



- RELACIONAMENTO



- ATRIBUTO



RELACIONAMENTO - Representa a associação entre os elementos do conjunto de uma entidade com outra entidade.

Exemplo:

O João está matriculado na disciplina de Banco de Dados

onde:

- João - Elemento do conjunto de valores do atributo Nome do aluno da entidade Aluno;

- Banco de Dados - Elemento do conjunto de valores do atributo Nome da disciplina da entidade Disciplina;

- matriculado - Ligação existente entre um aluno e uma disciplina.



4.3 - Cardinalidade de Relacionamentos

Representa a frequência com que existe o relacionamento.

Exemplo:

Relacionamento 1:1 - O João é casado com a Maria.

onde:

- João - Elemento do conjunto de valores do atributo Nome da entidade Homem.

- Maria - Elemento do conjunto de valores do atributo Nome da entidade Mulher.

- casado - Ligação entre um homem e uma mulher, sendo que um homem pode ser casado com uma e apenas uma mulher, assim como uma mulher pode ser casada com um e apenas um homem.



Relacionamento 1:N ou N:1 - O Pedro trabalha no Departamento Pessoal.

onde:

- Pedro - Elemento do conjunto de valores do atributo Nome da entidade Funcionário.

- Depart. Pessoal - Elemento do conjunto de valores do atributo Nome do departamento da entidade Departamento.

- trabalha - Ligação entre um Funcionário e um Departamento, onde um funcionário pode trabalhar em um e somente um departamento e um departamento pode ter vários funcionários.



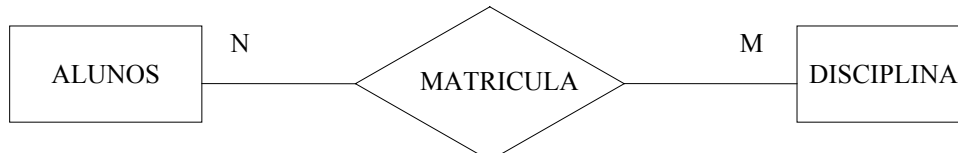
Relacionamento N : M - O Antonio está matriculado na disciplina Banco de Dados.

onde:

- Antonio - Elemento do conjunto de valores do atributo Nome da entidade Aluno.

- Banco de Dados - Elemento do conjunto de valores do atributo Nome da Disciplina da entidade Disciplina.

- matriculado - Ligação existente entre um aluno e uma disciplina, onde um aluno pode estar matriculado em várias disciplinas e cada disciplina pode ter vários alunos matriculados.



4.4 - Atributos do Relacionamento

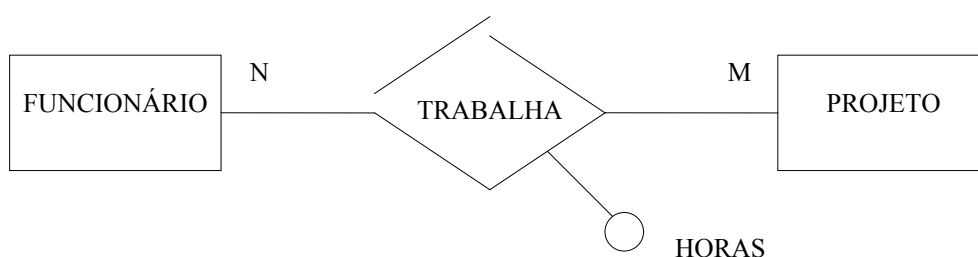
- Quando um determinado relacionamento possui atributos, também conhecido como relacionamento valorado. Esta situação ocorre apenas em relacionamento N : M.

Ex. Pedro trabalha no projeto Alfa 30 horas.

- Pedro - Elemento do conjunto de valores do atributo Nome da entidade Funcionário.

- Alfa - Elemento do conjunto de valores do atributo Nome do Projeto da entidade Projeto.

- trabalha - Ligação existente entre um funcionário e um projeto. Neste caso, este funcionário trabalha 30 horas neste projeto, porém este mesmo funcionário poderá trabalhar outro número de horas em outro projeto, assim como outro funcionário trabalha outro número de horas no mesmo projeto Alfa. Podemos concluir que 30 horas é o atributo que pertence ao Pedro no projeto Alfa.

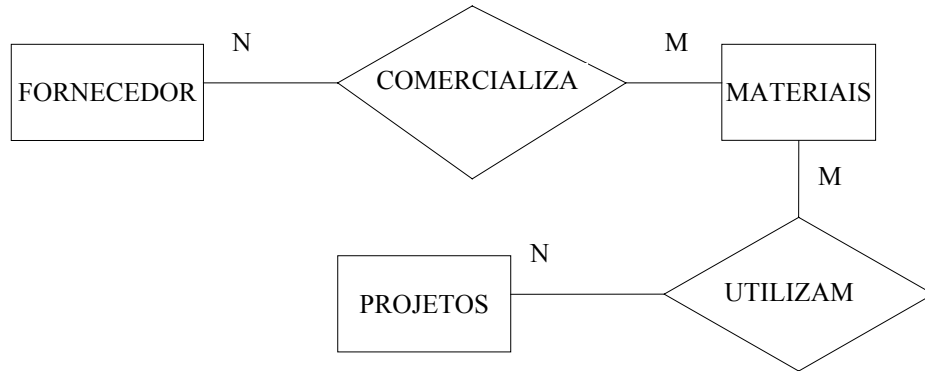


4.5 - Grau do Relacionamento

Indica o número de entidade que se relacionam.

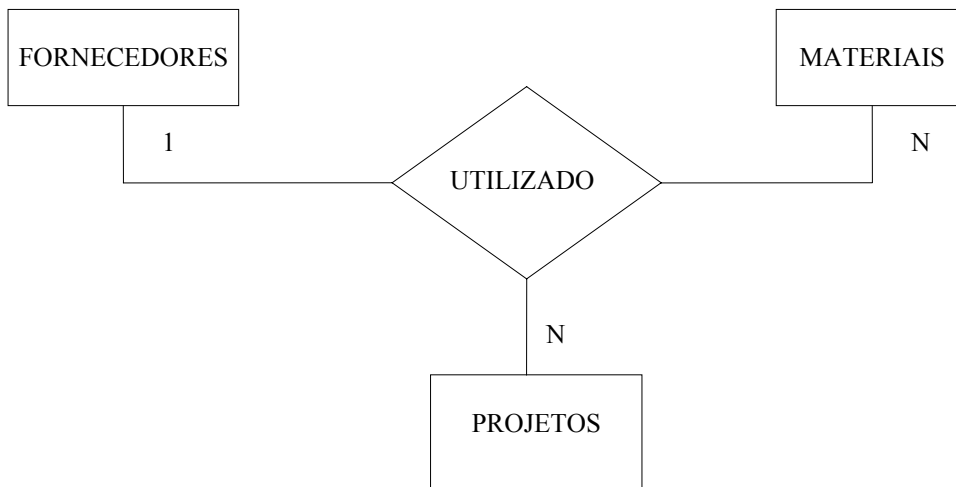
4.5.1 - Relacionamento Binário - Quando existe o relacionamento entre apenas duas entidades.

Ex. Um fornecedor comercializa materiais que são utilizados em diversos projetos.



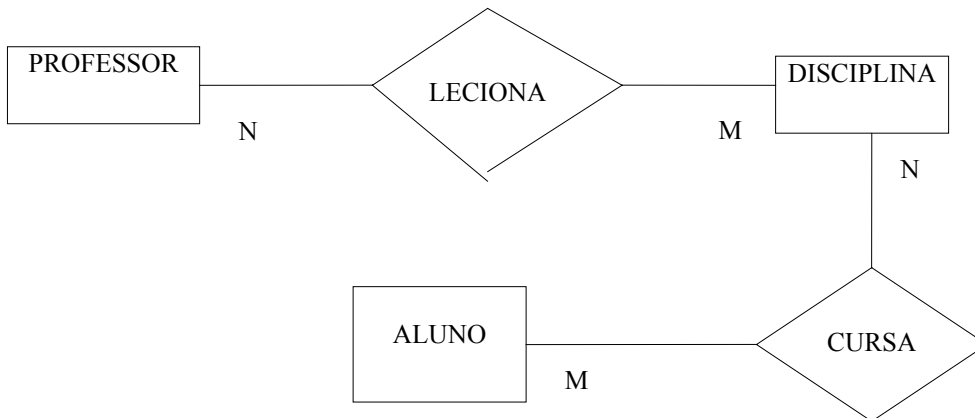
4.5.2 - Relacionamento Ternário - Quando existe o relacionamento entre três entidades.

Ex. Um fornecedor comercializa materiais que são utilizados em projetos específicos.

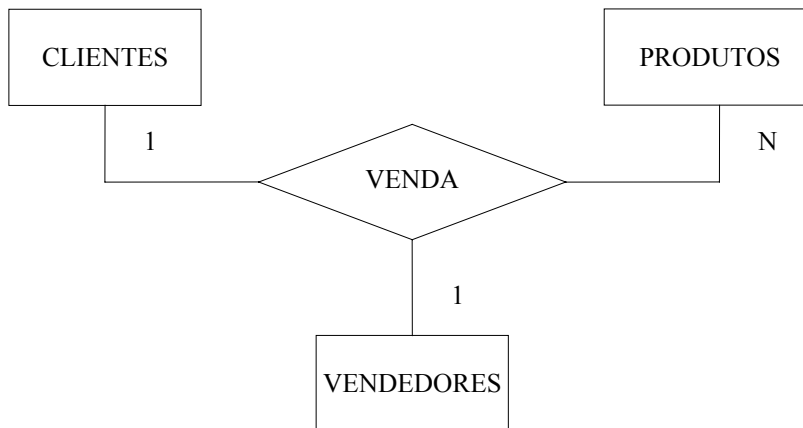


Exemplos de Relacionamento:

- O Professor Alberto leciona Estrutura de Dados e o aluno Pedro cursa Linguagem de Programação



- Pedro comprou 1 Kg. de banana do vendedor Manoel



Obs. Para que haja uma venda, tem que haver um cliente, um produto e um vendedor.

ÍNDICE

1 - CONCEITOS GERAIS DE BANCO DE DADOS	1
1.1 - Conceitos	1
1.2 - Representação Física do Banco de Dados.....	2
1.3 - Visões do Banco de Dados.....	2
1.4 - Vantagens do Banco de Dados em relação à arquitetura tradicional.....	3
2 - ORGANIZAÇÃO DE ARQUIVOS	5
2.1 - Seqüencial	5
2.2 - Indexado	5
3.3 - Seqüencial Indexado	6
3.4 - Relativa ou Direta	6
3.5 - Invertido	7
3 - NORMALIZAÇÃO DE DADOS	8
3.1 - Definição	8
3.2 - Primeira Forma Normal (1FN).....	8
3.3 - Segunda Forma Normal (2FN).....	8
3.4 - Terceira Forma Normal (3FN).....	9
4 - MODELO DE ENTIDADE E RELACIONAMENTO (MER).....	11
4.1 - Definição	11
4.2 - Representação Gráfica	11
4.3 - Cardinalidade de Relacionamentos	12
4.4 - Atributos do Relacionamento.....	13
4.5 - Grau do Relacionamento.....	14