

Conceitos

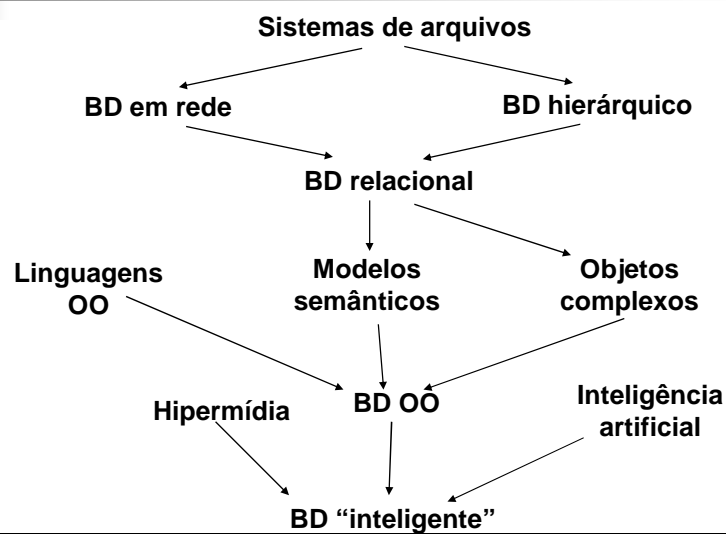
“Um *Banco de Dados* é uma coleção estruturada de dados relacionados a alguns fenômenos reais que estamos tentando modelar.”

Ozsü

“Um *Banco de Dados* é uma coleção de dados relacionados, organizada e armazenada de forma a possibilitar fácil manipulação.”

Elmasri

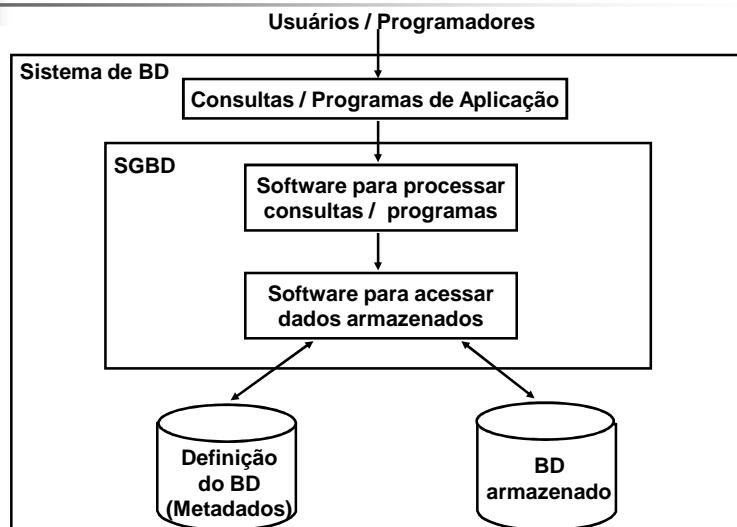
Evolução dos BDs (Khoshafian)



Conceitos

- **SGBD** (Sistema Gerenciador de Banco de Dados)
 - Conjunto de programas para manipulação do Banco de Dados
 - Aplicações específicas para gerenciar as operações de inserção, remoção, atualização e consulta ao BD.
 - Facilidades proporcionadas por um SGBD:
 - Definição do BD
 - Construção do BD
 - Manipulação dos dados

SGBD + BD = *Sistema de Banco de Dados*





Conceitos

- São componentes de um SBD:
 - *Hardware*: Memória secundária, canais de comunicação, etc.
 - *Dados*: Valores fisicamente armazenados no Banco de Dados.
 - *Software*: SGBD, *Front-End*.
 - *Usuários*: Pessoas envolvidas tanto na definição quanto na utilização dos dados.
 - Administradores (*DBAs*)
 - Programadores de Aplicações
 - Usuários Finais



Conceitos

- São objetivos de um SBD:
 - *Rapidez*: consultas on-line para recuperação a informação;
 - *Disponibilidade*: toda a informação da base de dados está totalmente disponível todo o tempo.
 - *Flexibilidade*: Mudanças não previstas tornam-se tratáveis.
 - *Integridade*: Menos duplicação e uso de políticas de atualização resultam em consistência.

Conceitos

- SBD x Sistemas de Arquivos - Vantagens:
 - *Baixa Redundância*: O SBD propaga as atualizações de acordo com regras definidas.
 - *Menos inconsistência*: Dados estruturados através de regras.
 - *Compartilhamento*: Várias aplicações podem acessar o BD ao mesmo tempo.
 - *Segurança*: Permite aplicação de restrições.

SBDs são construídos baseados em **Modelos de Dados**

Modelos de Dados

- Coleção de conceitos construídos com base em abstrações, que são usados para descrever um conjunto de dados e as operações para manipulá-los.
- Compreende uma coleção de elementos de representação com propriedades semânticas e sintáticas definidas.
- Um conjunto de elementos representados segundo um modelo para representar uma porção do mundo real constituem um esquema de dados compreensível pelo gerenciador.

Modelos, Esquemas e Instâncias

- Modelos Conceituais: Representam a realidade com alto nível de abstração. Ex.: Modelo ER
 - Modelos Lógicos: Descrição dos dados da forma como serão processados. Ex.: Modelo Relacional
 - Modelos Físicos: Descrevem como os dados são armazenados fisicamente (tipo de arquivo, formato dos registros, tipos de dados)
-
- Esquema: Descrição do Banco de Dados, baseada em um modelo.
 - Instância: Situação do Banco de Dados em um determinado momento.

Projeto de Bancos de Dados



Coleta e Análise de Requisitos

Requisitos de Dados

Projeto Conceitual

Esquema Conceitual

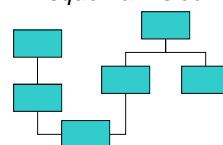
Projeto Lógico

Esquema Lógico

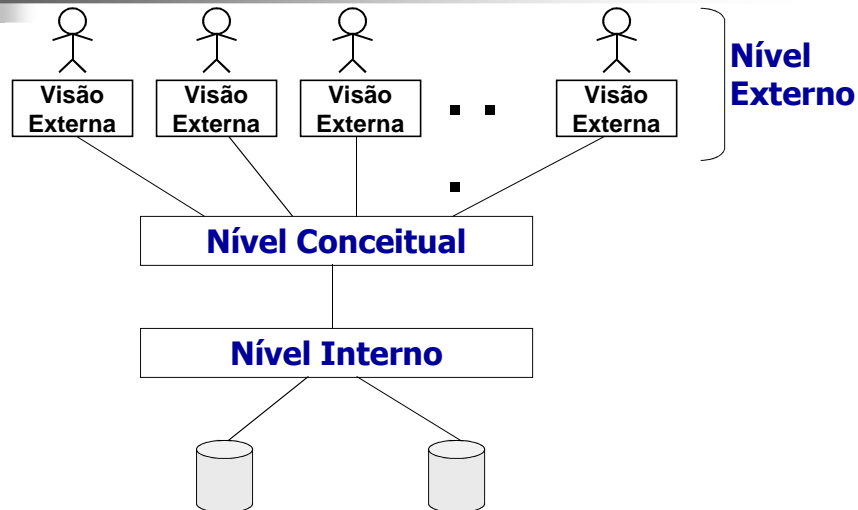
Projeto Físico

Esquema Físico

Dependentes
do SGBD



Arquiteturas em Níveis



Arquiteturas em Níveis

- Nível Externo: Esquemas externos e visões de usuários.
- Nível Conceitual: Descreve a estrutura geral do Banco de Dados, escondendo detalhes físicos.
- Nível Interno: Descreve a estrutura de armazenamento físico do Banco de Dados.

A estrutura de níveis permite a implementação da *independência de dados*.



Independência de Dados

Capacidade de alteração de um nível do Sistema de Banco de Dados sem afetar os outros níveis.

- Independência Lógica de Dados:

Capacidade de alterar o nível conceitual sem exigir mudanças no nível externo.

- Independência Física de Dados:

Capacidade de alterar o nível interno sem exigir mudanças no nível conceitual ou externo.



Projeto Conceitual

- Suportado por Modelos Conceituais ou Semânticos como Entidade-Relacionamento.
- Baseado em mecanismos de abstração:
 - *Classificação:* Alocação de objetos similares, caracterizados por propriedades comuns, em classes de objetos. "é-instância-de"
 - *Agregação:* Construção de objetos compostos a partir de seus objetos componentes. "é-parte-de"
 - *Generalização:* Relacionamento de subconjunto entre os elementos de 2 ou mais classes. "é-um"

Modelo Conceitual: Entidade-Relacionamento

- Apresentado por Peter Chen, em 1976.
- Numerosas modificações e extensões ao modelo original: ER estendido, ER temporal, ER espacial, ER distribuído.
- Uso como modelo conceitual durante o processo de projeto do banco de dados.
- Raras implementações de SGBDs baseados diretamente no modelo.



Modelo Entidade-Relacionamento

- Conceitos Principais:

Entidade: representam classes de objetos do mundo real.
(Aluno, professor)

Relacionamentos: representam associações entre 2 ou mais entidades.

Atributos: representam propriedades das entidades ou dos relacionamentos.



Modelo Entidade-Relacionamento

▪ Cardinalidade

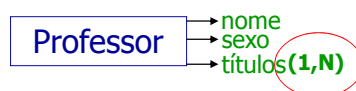
- Para aumentar a precisão, pode ser indicada cardinalidade mínima e máxima.
- Nos Relacionamentos: indica a ocorrência mínima e máxima de cada entidade.



Modelo Entidade-Relacionamento

▪ Cardinalidade

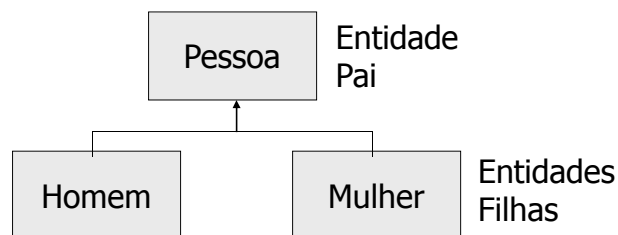
- Nos Atributos: indica a ocorrência mínima e máxima de determinado atributo.



- *Atributo Opcional*: cardinalidade mínima de atributo igual a 0 (zero).
- *Atributo Multivalorado*: cardinalidade máxima de atributo maior que 1 (um).

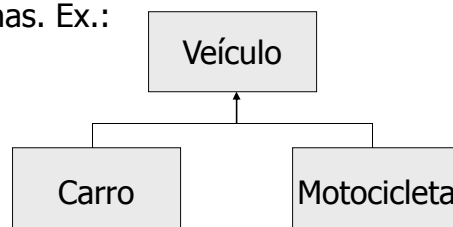
Modelo ER Estendido

- Inclui hierarquias de *generalização* e de *subconjunto*.
- Hierarquia de Generalização: Uma entidade Pai é uma generalização de um grupo de entidades Filhas se cada instância das entidades Filhas é também uma instância da entidade Pai.



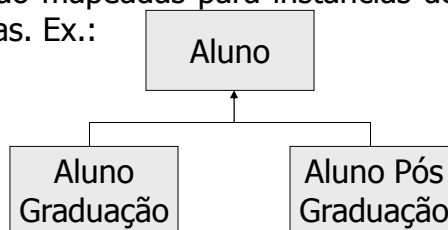
Generalização

- Propriedades de Cobertura
- **Cobertura total (t)**: toda instância da entidade Pai é mapeada para pelo menos uma instância das entidades especializadas (Filhas). Ex.: Pessoa → Homem / Mulher
- **Cobertura parcial (p)**: há instâncias da entidade genérica (Pai) que não são mapeadas para nenhuma instância das entidades Filhas. Ex.:



Generalização

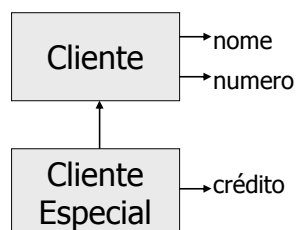
- **Cobertura exclusiva (e):** toda instância da entidade genérica é mapeada para no máximo uma instância das entidades especializadas. Ex.: Veículo → Carro / Motocicleta
- **Cobertura de sobreposição (s):** há instâncias da entidade pai que são mapeadas para instâncias de duas ou mais entidades filhas. Ex.:



- **OBS.:** quando a cobertura não está especificada, admite-se **(t,e)**.

Hierarquia de Subconjunto

- Hierarquia de Subconjunto: É um caso particular da hierarquia de generalização. Uma entidade X é um subconjunto de outra entidade Y se toda ocorrência de X for também uma ocorrência de Y.
- O tipo de cobertura é sempre **(p,e)**.



Modelo Lógico: Relacional

- Apresentado pelo pesquisador da IBM Edward F. Codd em 1970.
- Modelo formal, com embasamento matemático (Álgebra Relacional).
- Dados armazenados em **tabelas** (estruturas simples e uniformes).
- Sucesso comercial.

Modelo Relacional

- Conceitos ligados a representação em tabelas:
 - *Tabela (Relação)*: Conjunto relacionado de dados.
 - *Registro (Tupla)*: Linha da tabela. Instância com valores definidos para os atributos.
 - *Campos (Atributos)*: Coluna da tabela, assume valores dentro de um conjunto possível (domínio).
 - *Chaves (Identificadores)*: Atributos cujos valores distinguem unicamente uma tupla.
 - *Superchave, chave candidata, chave primária.*

Modelo Relacional

▪ Restrições de Integridade

- Garantem a consistência dos dados.
- Podem ser:
 - *De Chave*: Chaves candidatas não podem ter o mesmo valor em tuplas distintas da mesma relação.
 - *De Entidade*: Chaves primárias não podem possuir valor nulo.
 - *Referencial*: A tupla de uma tabela que referencia uma outra tabela deve referenciar um valor válido (tupla existente) = *Chave estrangeira*.

Outras formas de restrições são garantidas através da Normalização.

Normalização

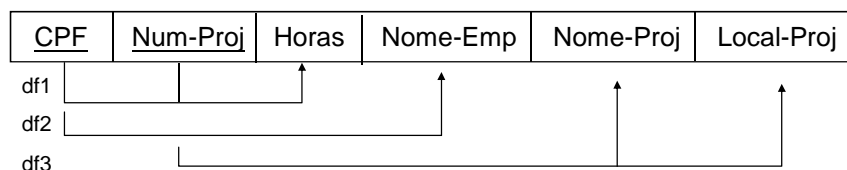
- Visa relacionamentos bem fundamentados, evitando redundância e garantindo consistência.
- É um mecanismo formal de análise de esquemas de relações, analisando suas chaves e as dependências entre seus atributos.
- Projeto conceitual bem realizado resulta naturalmente em esquemas normalizados.

1ª Forma Normal

- Está na 1ª FN uma tabela que não contém atributos compostos ou multivalorados. Ou seja, a 1ª FN define tabelas onde todos seus atributos representam valores atômicos.
- Para esquemas não normalizados (1ª FN), os atributos multivalorados ou compostos devem ser convertidos em novas tabelas, com suas respectivas chaves primárias.

Dependência Funcional (2ª FN)

- É uma definição sobre a semântica dos atributos que compõem uma relação.



df1 : {CPF, Num-Proj} → {Horas}

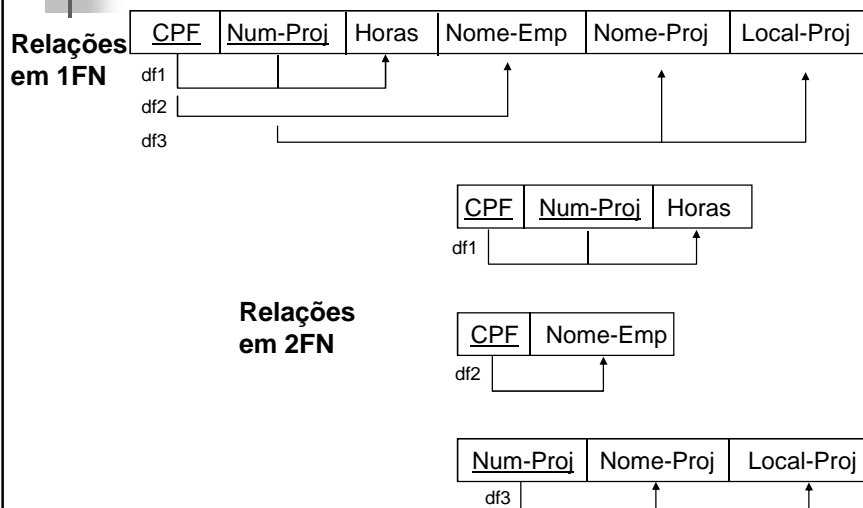
df2 : {CPF} → {Nome-Emp}

df3 : {Num-Proj} → {Nome-Proj, Local-Proj}

2ª Forma Normal

- É aplicada somente em relações com chave primária composta.
- Uma relação está em 2FN se estiver em 1FN e todo atributo não-primário (fora da chave) for totalmente dependente da chave (mesmo de forma transitiva). Nenhum atributo pode depender de apenas parte da chave composta.
- A aplicação da 2FN gera novas relações, que herdam a chave parcial que determinava seus atributos.

2ª Forma Normal

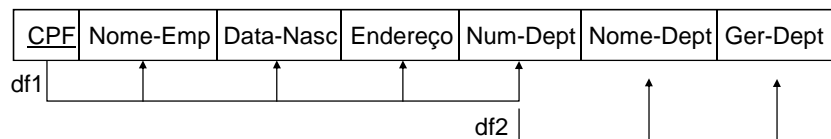


3ª Forma Normal

- Uma relação está em 3FN se estiver em 2FN e nenhum atributo não-primário for transitivamente dependente da chave primária.
- É aplicada sobre atributos que possuem dependência transitiva em relação a outro atributo da entidade que não participe da chave primária, eliminando a dependência transitiva.

3ª Forma Normal

Relações em 2FN



Criar uma nova entidade que contenha os atributos que dependem transitivamente de outro, com sua chave primária sendo o atributo que causou a dependência.

Relações em 3FN

