

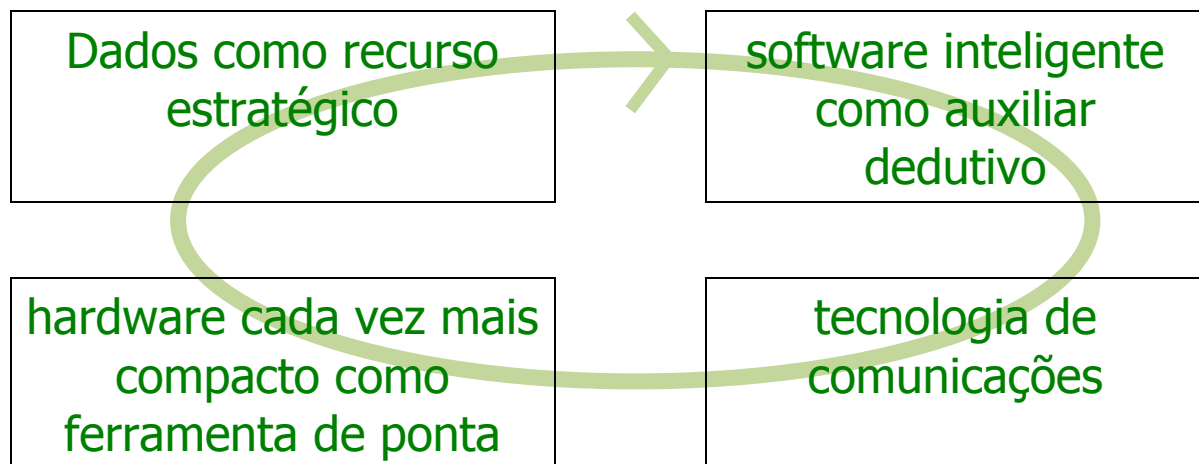
# MODELAÇÃO DE DADOS

## 1. INTRODUÇÃO

- Negócios com necessidade de métodos mais eficazes de tomada de decisão
- Busca de redução de custos e aumento de qualidade e competitividade



✓ Importância da informação



### Dados

- ➡ exigem maior trabalho metodológico e conceptual para o alcance consistente da informação vital para os processos decisórios
- ➡ são produzidos na empresa em função do seu negócio e dos sistemas que o suportam
- ➡ têm uma genética conceptual e fluidez complexa

É tratando bem os dados que se obtêm as melhores informações e consequentemente melhor conhecimento.

Implica na prática a aplicação de três conceitos fundamentais:

a) Dados Modelados:

Devem ser estudados no seu formato, origem, meio, natureza e formação e no seu relacionamento com outros dados.

Os dados não deverão ser propriedade específica de uma área ou grupo, mas, sim pertencer ao património da empresa, mantidos os requisitos de segurança e privacidade definidos na sua área de origem.

b) Dados Resguardados:

Os dados deverão apresentar os requisitos básicos de:

- integridade - garantia de sua veracidade lógica
- segurança - define o seu estado físico e os seus aspetos de privacidade
- documentação - representa a permanência de sua verdade semântica e a garantia de seu entendimento ao longo do tempo

c) Dados Disponibilizados:

Deverá existir um conjunto de ferramentas que permitam o acesso, a atualização, a consolidação, visualização e a simulação de informações para a tomada de decisões

A aplicação destes conceitos



**Dados:** Expressão em estado bruto de factos ou eventos, que sem necessariamente terem uma interpretação são transformados em Informação

## 2. ANÁLISE DE SISTEMAS

Análise de sistemas é, sob vários aspetos, a parte mais difícil do desenvolvimento de um sistema de informação

Analista como intermediário:

- **comunidade dos utilizadores** - percebem com maior ou menor sensibilidade dos seus problemas, mas acham difícil explicá-los e são vagos sobre a forma de serem ajudados
- **programadores** - capazes de desenvolverem as função de processamento, mas sem a informação adequada para saber o que é melhor para a empresa

Analista - deve combinar o que é atualmente possível e o que vale a pena ser feito para a empresa; **deve dar respostas a questões que lhe são colocadas e não às que não lhe são formuladas...** pelo menos ponderar!

Principais problemas que encontra

- a) Difícil aprender o bastante sobre a empresa para conseguir determinar os requisitos do sistema através da visão dos utilizadores.  
Não pode saber o que não lhe foi dito... ou pode?!!!  
- Pode perguntar, o que implica: experiência...
- b) Os utilizadores não sabem o suficiente sobre processamento de dados para saberem o que é viável
- c) Sobrecarga de detalhes sobre o negócio e técnicos, inerentes ao novo sistema

- d) O documento que define os detalhes do novo sistema é um contrato que os utilizadores muitas vezes não entendem pela sua dimensão e conceitos técnicos, acabando por ser aceite na base de que os informáticos sabem o que estão a fazer
- e) Muitas vezes se o documento de especificações faz sentido para os utilizadores, pode não ser útil para os projetistas e programadores: obriga a reanálise redefinindo os dados e a lógica dos processos

## **2.1.- Abordagens top-down, bottom-up, middle-down**

**Top-Down:** baseia-se em observações amplas da empresa

Objetivo: através de uma visão macro da empresa criar as entidades e relacionamentos que fundamentam os seus negócios

Vantagem: identifica o universo de dados da empresa e possibilita uma visão de integração entre as partes.

Dificuldade: mobilização necessária para empreender um trabalho de modelação que se estende às várias unidades funcionais da empresa

**Bottom-up:** orientação aos processos e aos dados produzidos

Dificuldade: agregação dos atributos necessários às visões lógicas dos utilizadores da aplicação.

**Middle-down:** objetivo: os modelos criados por um sistema, devem poder ser consolidados com outros já existentes conseguindo uma integração contínua e gradativa

## 2.2 - Perspetiva histórica das metodologias de modelação

### Análise Estruturada Clássica – Gane & Sarson e DeMarco (76-77)

- abordagem top-down por decomposição funcional gradativa;
- sem integração com Modelo de Dados;
- aplicação típica: sistema com estrutura de dados simples e funcionalidade complexa;
- técnica de modelação utilizada: DFD

### Engenharia de Informações – Finkestein & Martin (80)

- abordagem voltada a dados: modelos construídos em volta de diagramas ER;
- modelo de funções mais a nível de projeto que de análise;
- aplicação típica: sistemas com estruturas de dados complexas e comportamento simples;
- técnicas de modelação utilizadas: Normalização Relacional, Diagrama ER

### Análise Estruturada '80' – Gane(82)

- abordagem top-down como análise estruturada clássica;
- modelo de dados é construído a partir do modelo funcional;
- modelo de dados disponível somente após toda a modelação de funções;
- técnicas de modelação utilizadas: DFD, Normalização Relacional, Diagrama ER

## Análise Estruturada Moderna – McMenamin & Palmer(84), Yourdon(89)

- abordagem bottom-up, a partir de lista de eventos;
- fácil integração com modelo de dados;
- metodologia: modelo conceptual, modelo de dados, lista de eventos, diagrama comportamental (reação do sistema a um evento: um por evento), documentação bottom-up;
- técnicas de modelação utilizadas: DFD, Diagrama ER, Diagrama de Transação de Estados de Eventos, Modelação de Eventos

## Análise Orientada a Objectos – Rumbaugh OMT(90); Coad & Yourdon(OOA, OOD); Schlaer & Mellor, Booch(91,94), Fusion, ...

- conceitos de encapsulamento (objeto reúne uma entidade com os métodos que a tratam) e herança (de atributos e métodos);
- técnicas aplicáveis para aqueles que vão implementar em OO;
- técnicas de modelação utilizadas: DFD, diagrama de objetos, modelação de eventos, state-charts

### 3. MODELAÇÃO

**MODELO** - representação abstrata da realidade, atingido através da percepção do modelador, utilizando uma ou mais metodologias

**MODELAÇÃO** - criação de uma descrição de um "mundo real", para que diferentes observadores o consigam visualizar de um modo não ambíguo, com o objetivo de permitir especificar de forma conceptual o que o software deve fazer.

#### Passos na modelação

REALIDADE → Percepção, denominação, seleção, classificação → **MODELO**

Na prática não é possível descrever todas propriedades de um sistema num único modelo.

As técnicas de modelação descrevem o sistema em perspetivas:

- **Dados (modelo estático)**
  - modelo de dados de um sistema;
  - de forma abstrata: a definição do conjunto de estados do sistema;
  - modelos: Entidade-Relação ER
- **Controlo (modelo dinâmico)**
  - fluxo de controlo do sistema;
  - que atividades são executadas sequencialmente, que atividades ocorrem independentemente;
  - modelos: Fluxograma, Redes de Petri

## - Funções

- sistema é dividido em unidades funcionais;
- fluxo de dados entre unidades funcionais;
- modelos DFD

## Objetivos da modelação:

- Representar o ambiente observado
- Documentar e normalizar
- Fornecer processos de validação
- Garantir processos de relacionamentos entre objetos

## 4. MODELO DE DADOS

### 4.1. Etapas na construção do modelo:

(visão tradicional)

#### 1 - Modelo conceptual

Representa as regras de negócio sem limitações tecnológicas ou de implementação

Temos:

Visão Geral do negócio

Facilidade de entendimento entre utilizadores e analistas

Possui somente as entidades e atributos principais  
(Pode conter relacionamentos N para M....)

#### 2- Modelo Lógico

Leva em conta os limites impostos por algum tipo de tecnologia de SGBD. (base de dados hierárquica, base de dados relacional, etc.)

Características:

Deriva do modelo conceptual e visa a representação do negócio



Possui entidades associativas em lugar de relacionamentos N:M

Define as chaves primárias das entidades

Normalização até a 3a. forma normal

Adequação ao padrão de nomenclatura

Entidades e atributos documentados

### 3- Modelo físico

Leva em consideração limites impostos pelo SGBD (Sistema Gestão de Base de dados) e pelos requisitos não funcionais dos programas que usam os dados.

Características:

Elaborado a partir do modelo lógico

Pode variar segundo o SGBD

Pode ter tabelas físicas (log , parâmetros , ...)

Pode ter colunas físicas (ids, replicação, ... )

Revisões às etapas:

**Modelo conceptual:** se surgir a tentação de identificação de relações N:M, terão de ser investigadas, percebida a sua importância como parte fulcral do negócio e transformadas em entidades – tipicamente as mais relevantes para o negócio.

**Modelo Lógico:** deixa de fazer sentido falar de entidades associativas e normalização.

## 4.2. Etapas na construção do modelo: (revisão)

### 1 - Modelo conceptual

A melhor forma de descrever o negócio é através da linguagem corrente.

Na etapa conceptual far-se-á a recolha de todo o conhecimento que descreva as entidades envolvidas, as suas relações, as suas características e os processos em que estão envolvidas.

Nota: dado o foco na construção do modelo de dados, a parte de processos deverá ser usada para suportar a descrição do negócio e eventuais parametrizações que possam vir a ser necessárias.

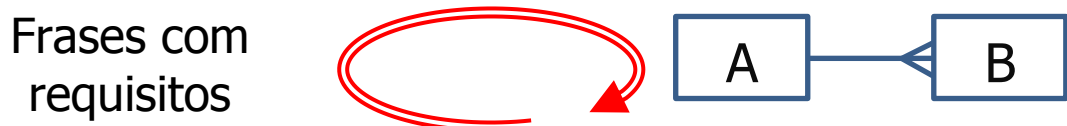
É fundamental nesta etapa perceber de facto todas as relações que possam existir entre entidades. As relações podem surgir de forma explícita, e neste caso a sua caracterização virá com a descrição, ou poderão ser questionadas sobre a informação que nos é transmitida. Este último exercício é fundamental para tornar visíveis algumas das relações do negócio que nem sempre são devidamente transmitidas. É com este exercício que irão ser caracterizadas e percebidas o que na abordagem tradicional são as relações M:N.

Estas relações, na maior parte das vezes, são as que contêm o negócio propriamente dito. Vejamos um exemplo simples de uma organização em que são referidas faturas e recibos, com a informação adicional de que os recibos pagam as faturas, ou até eventualmente sem informação adicional alguma. Dado que os pressupostos do analista e de quem descreve o negócio são muitas vezes diferentes, a relação pode não ser clarificada. Dado que a solução para a qual se procura o modelo não será certamente para criar uma lista de faturas e outra de recibos, será necessário averiguar a forma como a relação ocorre, e identificar ou criar no modelo as entidades necessárias para um suporte adequado.

Da recolha de informação realizada sobre o negócio resultará assim um texto com o conjunto de requisitos.

Dado que o nosso objetivo é a criação de um modelo relacional não adiantará camuflá-lo ou ignorá-lo e as suas características poderão ser introduzidas desde o início. As relações entre entidades são uma forma muito natural de expressar o conhecimento.

Na etapa conceptual, e para melhor suporte do conhecimento extraído pela análise do negócio, deverá fazer-se um exercício simples de tradução:



Traduz-se então cada requisito numa ou mais frases com a sintaxe admissível na linguagem de modelação.

Desta forma teremos uma descrição textual do negócio com uma tradução agregada para a linguagem de modelação.

Fundamental nesta etapa conceptual é que todo o negócio tenha sido percebido. Ou, dito por outras palavras, é fundamental que não haja relações de outro tipo que não o 1:M. Se isto não foi alcançado, o negócio não foi percebido e não deverá avançar-se.

## 2 - Modelo lógico

O modelo lógico terá duas fases. Uma primeira em que se fará a fusão de todos os requisitos resultando assim um esquema global para o negócio e uma segunda fase em que se deverá fazer um exercício de simplificação do modelo.

A simplificação consistirá em analisar o modelo e procurar semelhanças de vários tipos.

Podem procurar-se identidades que sendo diferentes são caracterizadas do mesmo modo e que portanto se poderão fundir numa única. Por exemplo cliente, fornecedor, funcionário, etc...têm praticamente os mesmos atributos, ou atributos muito semelhantes e eventualmente com uma tipificação acessória poderão reduzir-se a uma única. Na realidade muitas das entidades distinguem-se pelo papel que desempenham e não pela sua caracterização.

Podem procurar-se entidades que sendo diferentes desempenham papéis semelhantes e que se poderão também fundir numa única com alguma caracterização adicional. São exemplo disto os inúmeros documentos de uma organização e de todas as relações que podem ser identificadas entre eles. Mesmo limitando o tipo de documentos a 20, se for necessário relacionar cada um com todos os outros, teremos 400 tabelas de relação.

Na prática incomportável de gerir. Se só forem previstas algumas das relações poderá ser difícil evoluir o modelo. Com uma tipificação adequada o mesmo problema resolve-se com 3 tabelas.

Podem também, por exemplo, procurar-se caracterizações em cascata, do tipo, família, sub-família, sub-sub-família, que podem ser resolvidas com uma auto-relação.

Na prática todas estas simplificações deverão traduzir-se em menos tabelas e maior versatilidade sem adulterar o negócio e sem hipotecar as suas funcionalidades => Preferencialmente tornando-o mais versátil.

No caso dos documentos, a existência de uma tabela para a sua tipificação permitirá na prática que se criem documentos de todos os tipos e como está resolvido o problema do relacionamento entre documentos não será necessária mais nenhuma tabela.

No caso das classificações hierárquicas resolvidas com recurso a uma auto-relação, torna-se possível que existam mais dos que os 3 níveis iniciais.

Poderá, por inerência à versatilização, ser necessário introduzir o conceito de parametrização com os benefícios inerentes. Ou seja, poderá ter de se dotar o modelo de tabelas de parâmetros de forma a comportar o negócio da forma que foi descrito e permitir por outro lado suportar outras opções que passam a ser oferecidas. Isto leva-nos ao modelo físico.

### **3 - Modelo físico**

Com a introdução de tabelas de parametrizações e outras necessárias à montagem da solução, esta etapa resumir-se-á a uma implementação concreta do modelo no SGBD escolhido.

Poderão igualmente ser resolvidas de forma prática as ligações entre tabelas, com a criação de alguns campos (IDs) que as simplifiquem, bem como a introdução de outros campos de controlo ligados à implementação.

### 4.3. Conceitos

**Entidade** - qualquer objeto do mundo real do qual se quer guardar informação:

ex: Cliente, produto, Contrato, Vendas, ...

**Atributo** - tudo o que se pode relacionar como propriedade da entidade, unidades de informação atômicas, designadas por colunas (campos)

ex: Código do cliente, referência do produto, data do contrato, valor da venda, ...

**Instância de uma entidade:** ocorrência de uma entidade - registo (*tuple*), corresponde à unidade de informação da entidade

**Domínio:** conjunto de valores possíveis do atributo

**Dicionário de dados:** definição formal dos elementos

**Atributo obrigatório:** aquele que para cada instância da entidade tem de ter um valor

**Atributo opcional:** pode possuir um valor

**Chave primária:** atributo (ou atributos) que identifica de forma exclusiva cada ocorrência de uma entidade e que foi escolhida para a representar.

**Chave candidata:** atributo ou agrupamento de atributos que identificam de forma única cada ocorrência da entidade; pode vir a ser chave primária.  
A chave candidata que não é chave primária é uma chave alternativa.

## **Características da chave primária:**

- a - NÃO PODE haver duas ocorrências de uma mesma entidade com o mesmo conteúdo na Chave Primária;
- b - A chave primária não pode ser composta por atributo opcional, ou seja, por um atributo que aceite nulo;
- c - Os atributos identificadores devem ser o conjunto mínimo que pode identificar cada instância de uma entidade;
- d - Cada atributo identificador da chave deve possuir um tamanho reduzido;
- e- Pode ser gerada automaticamente pelo sistema.

**Chave estrangeira:** atributo de uma relação que é chave primária de outra.

- pode ter o valor nulo
- pode ter valores duplicados

**Relação entre entidades:** associação entre instâncias de entidades devido a regras de negócio. Ocorre entre instâncias de duas entidades mas pode ocorrer entre instâncias da mesma entidade (auto relação)

**Cardinalidade:** indica quantas ocorrências de um entidade participam no mínimo e no máximo da relação

**Cardinalidade mínima:** define se a relação entre duas entidades é obrigatória

**Cardinalidade máxima:** define a quantidade máxima de ocorrências da entidade que pode participar da relação

## Tipos de Cardinalidade

- Um para um
  - temos um lado obrigatório e um lado opcional
  - usadas excepcionalmente:
    - dividir tabelas
    - grande número de campos
    - especificidade de alguns registos
    - isolar parte dos campos por questões de segurança
- Um para N
  - leva a chave primária do lado Um para o lado N
  - o atributo do lado N recebe o nome de chave estrangeira
- N para N
  - leva para o modelo lógico a necessidade de mais uma entidade: Associativa

## Normalização:

Conjunto de regras que visa minimizar as anomalias de modificação de dados e dar maior flexibilidade na sua utilização

- Minimiza as redundâncias e inconsistências
- Facilita a manipulação de dados
- Facilita a manutenção

A versão final dos dados pode sofrer alterações para atender a necessidades específicas do sistema por decisão do analista de desenvolvimento durante o projeto físico.

## 4.4. MODELO ENTIDADE-RELAÇÃO

### 4.4.1. Introdução

- Formalizado nos fins dos anos 1960 por C.W. Bachman



- Coloca ênfase na simplicidade e facilidade de leitura
- Proporciona uma correta representação lógica da organização
- Diversas ferramentas C.A.S.E. integram capacidade de modelação E/R

#### 4.4.2 Benefícios

- Focaliza a atenção na importância das relações
- Utiliza uma sintaxe de diagramas que conduz uma elevada taxa de informação a um estado perceptível
- Representa os requisitos de uma dada realidade de forma compreensível, tanto ao analista como ao programador

#### 4.4.3. Fatores críticos de sucesso no desenho de BD

- Conhecimento profundo dos requisitos o negócio
- Comunicação constante e aberta com os utilizadores finais
- Emprego de uma metodologia estruturada em todas as fases do processo
- Utilização de diagramas de representação

## 5. BASE DE DADOS RELACIONAL

### 5.1. Uma definição:

Coleção de dados operacionais inter-relacionados e armazenados de forma independente dos programas que os utilizam e que servem múltiplas aplicações

- Coleção: agrupamento com repetição
- Operacionais: estratégicos para a tomada de decisões
- Inter-relacionados: uma BD mantém um agrupamento de entidades - factos da realidade em questão, e de relações entre elas
- Independente dos programas: suportes físicos diferentes e autónomos
- Servem múltiplas aplicações: os dados de uma BD podem ser partilhados por diferentes aplicações

### 5.2. SGBD - Sistema gestão de base de dados

#### 5.2.1. Definição

Sistema cujo objetivo é gerir o acesso e a correta manutenção dos dados de uma BD

- Acesso: o SGBD deve disponibilizar uma interface que permita a comunicação com as aplicações - via uma linguagem
- Correta manutenção: devem ser garantidos aspetos de integridade, segurança e concorrência de modo a manter os dados consistentes

## 5.2.2 Funções básicas

### a) Métodos de acesso

**DDL** - Data Definition Language: permite a especificação do esquema da organização, ou seja, entidades com seus atributos e tipos de dados relacionados, as relações entre estas entidades e os índices de acesso associados aos atributos.

Esquema: organização lógica dos dados de uma realidade, adequados ao modelo de dados do SGBD;

**DML** - Data Manipulation Language: permite as operações usuais de manipulação de dados, executados pelas aplicações:

inclusão (*insert*),  
alteração (*update*),  
exclusão (*delete*) e  
consulta (*select*).

### b) Restrições de integridade - asseguram consistência dos dados, devem garantir:

- Estados possíveis da informação,  
ex: um desconto não pode ser superior a 40%, uma venda não pode levar as existências a negativo
- Manutenção das relações entre entidades,  
ex: não pode existir uma consulta sem estar associada a um médico

O SGBD disponibiliza a **DCL** - Data Constraint Language: permite programar:

- testes ( ex: desconto menor que 40%) e
- ações (ex: remover todas as consultas se um médico é removido).

- c) **Segurança:** evita a violação da consistência dos dados por agentes e/ou situações não previstas (falhas)
  - **autorização de acesso:** permitir que apenas agentes autorizados (utilizadores ou aplicações), realizem certas operações sobre certos dados;
  - **recuperação de falhas:** possibilitar o retorno do BD a um estado consistente de seus dados após a ocorrência de uma falha involuntária.
- d) **Controlo da concorrência:** evita conflitos de acesso simultâneo a dados por mais de uma transação - bloqueios (*locks*).
- e) **Independência dos dados:** transparência de gestão e armazenamento.
  - **Física:** a aplicação não se preocupa com detalhes de localização física dos dados ou controlo de integridade e segurança.
  - **Lógica:** garante que uma aplicação tenha condições de especificar a porção da BD a que quer ter acesso, não precisando ter conhecimento do esquema global.

### **5.3. Objetivos e requisitos de uma BD relacional**

A construção de uma BD Relacional deve visar a integração das atividades de gestão e operacionais de uma organização; deve ser alcançada dotando a BD de capacidade para:

- absorver a expansão das atividades da organização, possibilitando sistemas mais complexos;
- prover mecanismos de controlo centralizado das informações e representação de dados;

- comportar as últimas inovações do mercado, tais como sons, vídeos, gráficos;
- facilitar manutenção de dados e oferecer instrumentos para rápida restauração em situações de erro.

Esses pontos são traduzidos por uma série de conceitos que devem ser assegurados pela BD:

**Independência de Dados:** permitir que haja evolução na descrição de dados da empresa, como a criação de uma nova estrutura lógica decorrente de uma nova aplicação, ou a inclusão de um dado novo numa estrutura existente, sem que os sistemas ou aplicações tenham de ser alterados.

**Controlo de Redundância dos Dados:** representado pelo controlo centralizado dos dados partilhados por diversas aplicações; reduz a repetição de dados a um mínimo justificável e aceita apenas por uma questão de desempenho.

**Partilha de Dados:** os dados devem poder ser partilhados por mais de um utilizador.

**Garantia de Integridade dos Dados:** capacidade de evitar que aplicações ou utilizadores concorrentes realizem atualizações sobre os dados, tornando-os inconsistentes. Essa capacidade traduz-se na introdução de recursos no sistema que inibam o acesso a um dado por uma aplicação quando esse mesmo dado já tiver sido lido para modificação por outra aplicação. Outro importante recurso que garante a integridade dos dados

é a possibilidade de desfazer alterações erradas introduzidas por alguma aplicação.

**Segurança de Dados:** abrange conceitos como os de procedimentos de validação e controlo, garantia de integridade e controlo de acesso, que visam resguardar a BD de uma possível perda ou destruição de dados por falha de programa ou equipamento.

**Privacidade dos Dados:** são controlos que devem ser estabelecidos para que os dados possam ser acedidos somente por pessoa autorizada.

**Relações entre Dados:** a relação entre os dados armazenados em arquivos diferentes deve ser controlada de maneira automática - sem intervenção do programador.

**Controlo do Espaço de Armazenamento:** a BD deve ser provida de mecanismos de controlo de acesso do espaço reservado, do espaço real utilizado e da disponibilidade de espaço no SGBD. Além disso, deve incorporar técnicas para a otimização do armazenamento, tais como compressão de dados e reaproveitamento automático de espaços libertados por eliminação de dados.

## 5.4 - Metodologia de desenvolvimento de uma Base de Dados

Fases:

- \* Estratégia: Análise das Necessidades;
- \* Análise: Modelo Conceptual da Base de Dados;
- \* Projeto: Modelo Lógico da Base de Dados;
- \* Construção: Projeto Físico da Base de Dados;
- \* Implementação: Instalação da Base de Dados;
- \* Implantação: Monitorização e Sintonização.

Estratégia - A Análise das Necessidades:

- identificar as deficiências do sistema corrente,
- estabelecer novas metas,
- determinar se é viável o trabalho,
- sugerir melhorias.

Análise - O Modelo Conceptual:

- estrutura as necessidades globais de informação;
- desenho do **DHF** (Design History File, com a documentação de todo o conhecimento adquirido sobre o sistema), do Diagrama de Contexto (**DC** entidades externas que influem no sistema), do **DFD** (representação gráfica de processos mediante funções, fluxos e repositório de dados) e do **ER**;
- balanceamento entre entidades e funções.

Projeto - O Modelo Lógico:

- propõe-se a análise das opções de montagem de uma BD;
- normalização das entidades, atributos e seus relacionamento; escolha das chaves candidatas e primárias;

## Construção - O Projeto Físico

- transforma o Projeto Lógico no formato mais adequado a um SGBD específico. Cria-se a Estrutura Física da Base de Dados para um motor específico.

## A Implementação:

- geração e processamento das definições da BD;
- construção dos programas;
- carga inicial do BD;
- geração da segurança de acesso.

## A monitorização na fase de Implantação

- mede o desempenho da BD, avaliando: utilização da BD pelos sistemas de aplicações; acompanhamento da utilização; análise da evolução do desempenho.

## A fase de sintonização

- apresenta uma crítica construtiva do desempenho através de: análise dos pontos críticos no desempenho; especificação de novas estruturas; otimização da BD central e nos vários servidores da rede.



## 5.5. Regras de Integridade

O Modelo Relacional prevê duas regras gerais de integridade:

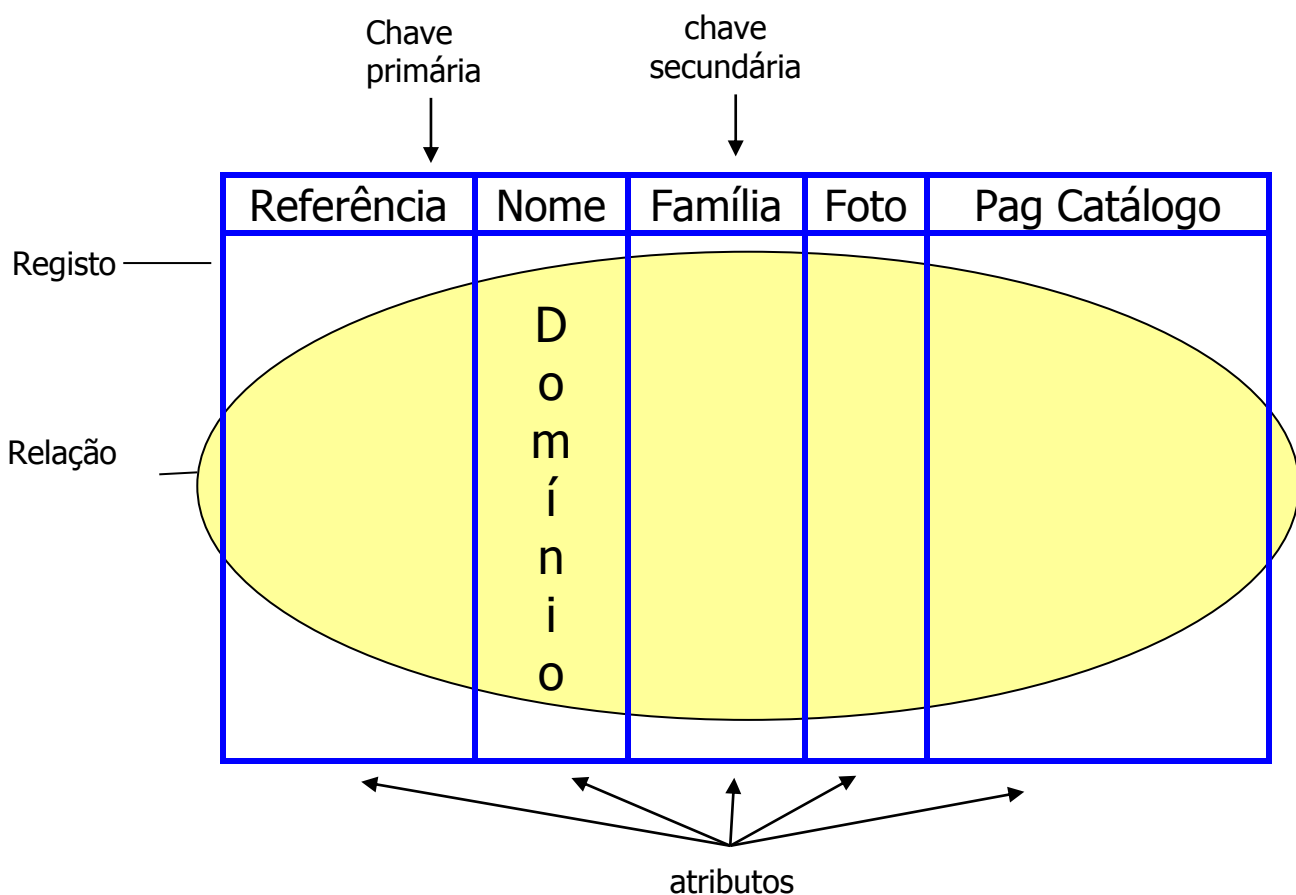
Integridade de entidade - qualquer atributo que faça parte da chave primária de uma relação não pode conter valores nulos - ausência de valores.

Integridade referencial - garante que as referências entre tabelas existam, ou seja, qualquer valor de chave estrangeira deve existir como valor de chave primária na tabela relacionada, ou deve ser nulo.

## 6. ÁLGEBRA RELACIONAL

### Conceitos:

- ❖ Relação: Representada por uma tabela de duas dimensões (linhas e colunas).
- ❖ Registo: Corresponde a uma linha de relação.
- ❖ Atributo: Corresponde a uma coluna da relação.
- ❖ Domínio: Valores possíveis dos atributos.
- ❖ Chave primária: Atributo, ou atributos (*superkey*) que identificam univocamente cada registo.
- ❖ Chave estrangeira: Atributo de uma relação que é chave primária de outra relação.



- ❖ Álgebra Relacional: Conjunto de operações sobre modelos relacionais de dados.

Podem ser agrupadas em duas categorias:

- A) Operações Tradicionais (Teoria dos Conjuntos)
  - União, Intersecção, Diferença, Produto Cartesiano.

## B) Operações Relacionais (Específicas da Álgebra Relacional)

- Seleção, Projeção, Junção, Divisão.

**UNIÃO** de duas relações A e B é o conjunto de todos os registos pertencentes a A ou a B

A = ABCDEFGH, B = FGHIJKL    união: ABCDEFGHIJKL

**INTERSECÇÃO** de duas relações A e B é o conjunto de todos os registos pertencentes a A e a B

intersecção: FGH

**DIFERENÇA** de duas relações A e B é o conjunto de todos os registos de A não pertencentes a B

diferença: ABCD

**PRODUTO CARTESIANO** de duas relações A e B é o conjunto de todos os registos originados pela concatenação de cada registo de A com cada registo de B

A = A,B,C, B=X,Y;

produto cartesiano = AX, AY, BX, BY, CX, CY

**SELEÇÃO** - É a operação usada para construir um subconjunto horizontal de uma relação, cujos registos satisfaçam uma determinada condição

**PROJEÇÃO** - É a operação usada para construir um subconjunto vertical de uma relação, obtida pela escolha de alguns atributos

**JUNÇÃO** de duas relações R1 e R2, que possuem um atributo em comum 'b'. A junção é o subconjunto do produto cartesiano das duas relações, cujos valores dos elementos do atributo comum sejam iguais nas duas relações.

Na relação resultante elimina-se a repetição da coluna 'b'

A1	B1
A2	B2
A3	B3

B1	C1
B2	C2
B3	C3

A1	B1	C1
A2	B2	C2
A3	B3	C3

**DIVISÃO** - Seja R1 uma relação com atributos x e y, e R2 uma relação com atributo z, com y e z definidos sobre o mesmo domínio.

Define-se a operação divisão, como sendo o conjunto dos elementos x, com pares (x,y) pertencentes a R1 para todos os valores y pertencentes a R2.

R1		R2	
x	y	z	
A	X	X	A
A	Y	Z	
A	Z		
B	X		
B	Y		

## 7. CONSTRUÇÃO FÍSICA PARA MODELOS E/R

### Terminologia

Modelo Lógico	Modelo Físico
Modelo dados E/R	Esquema da BD
Entidades	Tabelas
Atributos	Colunas
Relações	Restrições de integridade Índices PK e FK
Ocorrência	Registo

### Elementos da construções física

#### - Nomes

seguir uma convenção

definir nomes únicos e significativos

evitar sinónimos - nomes diferentes para o mesmo significado

evitar homónimos - nomes iguais para significados diferentes

#### - Tipo de dados

tipo de informação a armazenar

capacidade

tipo de operações com valores

precisa de ser ordenado/ agrupado?

(alguns tipos não permitem)

como tem de ser a ordenação?

(1, 10, 100, 2, 20, 200, .... está ordenado, se for texto!)

Tipos genéricos:

text

memo

number

date/time  
currency  
autonumber  
yes/ no  
ole object

- Nulo ou não nulo

não é a mesma coisa que zero ou string vazia  
valor predefinido  
operadores e funções próprias

- Único

- Chaves candidatas

simples	únicas
compostas	não únicas

- Chave primária

- Chaves estrangeiras

- Restrições de integridade: constrangimentos impostos aos atributos

- Restrições de integridade de entidade: a PK não pode ser nula nem se pode repetir

- Restrições de integridade referencial: garante a consistência da informação que estabelece a relação entre tabelas: não pode haver órfãos
  - uma modificação do valor da PK deve ter como reflexo a modificação dos valores de todas as FK correspondentes

- a FK pode ser nula e só aceita valores que existam na PK correspondente

### Tipos de integridade referencial

Reflexa: integridade de uma tabela com ela própria

Cíclica: de pai para filho

Múltipla: integridade de uma tabela com várias outras

- Integridade de domínio: conjunto de valores que uma coluna pode conter:
  - tipo de dados
  - conjunto de valores possíveis (sexo: M ou F ...)
- Integridade aplicacional: restrições impostas pelas regras de negócio
  - stock mínimo
  - só alguns utilizadores podem efetuar algumas operações
- Considerar a parametrização em tabelas específicas de todas as constantes identificadas
- Considerar a parametrização em tabelas específicas de todas as opções de negócio identificadas e suscetíveis de alterarem os processos de negócio
- Outras propriedades dos campos
  - formato
  - máscara de introdução
  - valor predefinido
  - regra de validação (restrição)