

[MAC0313]

# Introdução aos Sistemas de Bancos de Dados Aula 3

## O Projeto Conceitual de Bancos de Dados e o Modelo Entidade-Relacionamento (Parte 1)

Kelly Rosa Braghetto

DCC-IME-USP

8 de agosto de 2017

# Projeto de bancos de dados

Envolve as seguintes etapas:

1. Levantamento e análise dos requisitos
2. Projeto conceitual
3. Projeto lógico
4. Projeto físico

# Fase 1: Levantamento e análise dos requisitos

Nessa fase, o projetista:

- ▶ Registra de forma detalhada e completa os requisitos dos usuários com relação aos dados
- ▶ Identifica também requisitos funcionais conhecidos das aplicações
  - ▶ operações de consulta e modificação que serão realizadas no BD

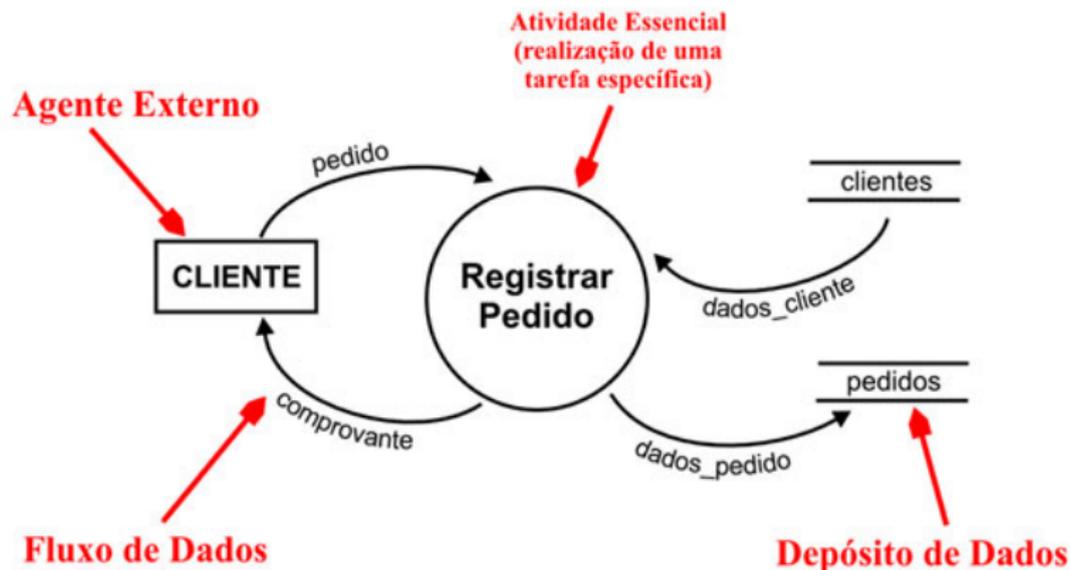
# Fase 1: Levantamento e análise dos requisitos

## Sobre os requisitos:

- ▶ São levantados por meio de entrevistas com os produtores e os usuários dos dados, e podem ser descritos por meio de técnicas para a especificação de requisitos, como os **Diagramas de Fluxos de Dados (DFDs)**
- ▶ Incluem os dados exigidos para processamento, os seus relacionamentos naturais e as informações relevantes para a escolha da plataforma de software para o BD

# Fase 1: Levantamento e análise dos requisitos

## Exemplo de diagrama DFD



Fonte: <http://www.linhadecodigo.com.br/artigo/1293/a-importancia-do-modelagem-de-objetos-no-desenvolvimento-de-sistemas.aspx>

## Fase 2: Projeto conceitual

Fase de criação de um esquema conceitual para o BD, utilizando um modelo de dados conceitual ( = de alto nível).

### Esquema conceitual (definição):

Descrição concisa de requisitos de dados dos usuários, contendo descrições detalhadas sobre os tipos de entidades, relacionamentos e restrições, expressos de acordo com os conceitos existentes no modelo de dados conceitual empregado

## Fase 2: Projeto conceitual

### Características de um esquema conceitual:

- ▶ Fácil compreensão (não envolve detalhes de implementação)
- ▶ Pode ser usado na comunicação com usuários não técnicos

### Exemplos bastante usados:

- ▶ **Modelo Entidade-Relacionamento (ER)**
- ▶ Diagramas de Classe da UML

## Fase 3: Projeto lógico

Essa fase corresponde ao mapeamento do esquema conceitual para um modelo de dados de implementação.

⇒ Passo necessário para a implementação do BD utilizando um SGBD comercial.

Exemplos de modelos de implementação bastante usados:

- ▶ Modelo Relacional
- ▶ Modelo Objeto-Relacional

## Fase 3: Projeto lógico

### Refinamento do esquema:

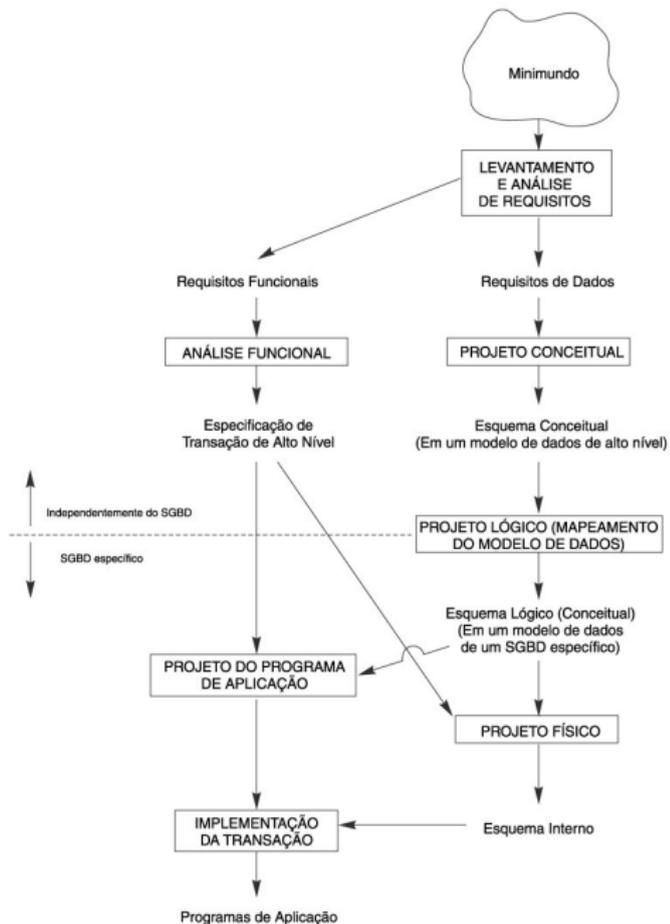
- ▶ O refinamento é uma etapa **opcional** no projeto lógico
- ▶ Ele identifica problemas em potencial no modelo lógico criado e aplica técnicas para melhorar o modelo
- ▶ Exemplo: no modelo relacional, é necessária uma etapa de **normalização** das relações
  - ▶ A normalização tem como objetivo evitar esquemas de BDs em que anomalias de alterações dos dados possam ocorrer

## Fase 4: Projeto físico

Fase na qual são definidas as estruturas de armazenamento interno, índices, caminhos de acesso, organizações de arquivos (para os arquivos do BD) e outros ajustes finos.

- ▶ Finalidade: otimizar o desempenho das operações de consulta e manipulação dos dados
- ▶ Pode até mesmo modificar o projeto de BD resultante das fases anteriores, a fim de satisfazer critérios de desempenho desejados
  - ▶ Exemplo: “denormalização” das relações em *Data Warehouses* (bancos de dados analíticos, em que o desempenho para as operações de consulta é um requisito importante)

## Principais fases do projeto de um banco de dados



# Abstrações de dados

A modelagem conceitual de dados está relacionada à representação dos dados por meio de um conjunto de abstrações.

Abstrações têm:

- ▶ **Estrutura:** do ponto de vista estrutural, abstrações são detalhes que deliberadamente se omitem numa dada visão de implementação para facilitar a visão global do tópico em questão.
- ▶ **Comportamento:** do ponto de vista comportamental, abstrações determinam restrições, regras e operações que devem ser tomadas (automaticamente) na manipulação do conjunto de elementos.

# Modelo Entidade-Relacionamento (ER)

- ▶ Criado por Peter Chen em 1976
- ▶ É simples e de interpretação intuitiva (mesmo para usuários não-especialistas)
- ▶ É modelo mais usado para a modelagem conceitual de BDs
- ▶ Descreve os dados com base em três conceitos principais:
  - ▶ **entidades**
  - ▶ **relacionamentos**
  - ▶ **atributos**

# Entidades e Atributos

- ▶ **Entidade** – “algo” do mundo real, com uma existência independente. Pode ter existência física (ex.: uma pessoa, uma casa, um carro, etc.) ou conceitual (ex.: um projeto, uma disciplina).
- ▶ **Atributos** – propriedades particulares que descrevem uma entidade (ex.: nome da pessoa, endereço, função, etc.). Um dada entidade terá um valor para cada um de seus atributos. Esses valores constituem a maior parte dos dados armazenados em um BD.

# Tipos de atributos no modelo ER

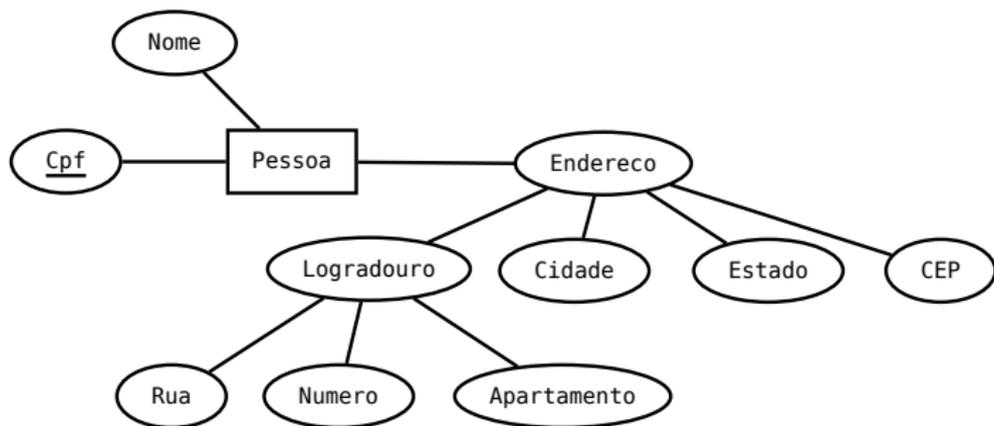
## Simplex (atômicos) × Compostos

- ▶ **Atributos compostos** – podem ser divididos em partes menores. Ex.: o atributo endereço pode ser dividido em Rua, Cidade, Estado e CEP.
  - ▶ O seu valor é dado pela concatenação dos valores dos atributos simples que os compõem.
  - ▶ Devem ser usados quando o usuário algumas vezes se refere ao atributo como um grupo, mas em outras se refere a um de seus componentes.  
(Caso contrário, o endereço, por exemplo, poderia ser definido como um atributo simples).
- ▶ **Atributos simples (ou atômicos)** – os atributos que não são divisíveis.



# Tipos de atributos no modelo ER

## Exemplo de hierarquia de atributos compostos



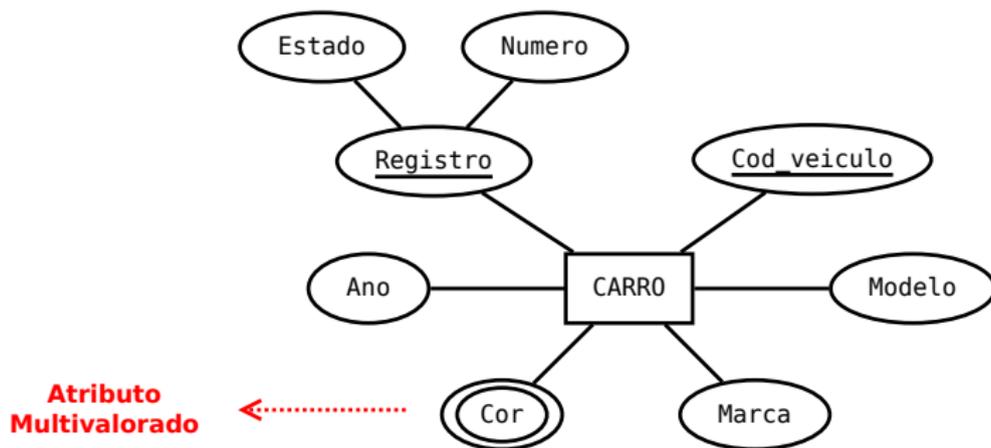
# Tipos de atributos no modelo ER

## Monovalorados × Multivalorados

- ▶ **Monovalorado** – atributo que tem um único valor para uma dada entidade.
  - ▶ **Ex.:** o atributo idade para uma pessoa.
- ▶ **Multivalorado** – atributo para o qual diferentes entidades podem ter diferentes quantidades de valores.
  - ▶ **Ex.:** atributo titulação para uma pessoa; algumas pessoas não possuem título algum, outras possuem 1, ou 2, ...
  - ▶ Um atributo multivalorado pode ter um **limite inferior e superior** pra restringir o número de valores permitidos a cada entidade individual.

# Notação para tipos de entidades e atributos no modelo ER

## Exemplo de atributo multivalorado



# Tipos de atributos no modelo ER

## Armazenados × Derivados

- ▶ **Atributo derivado** – é derivado a partir de outro(s) atributo(s) ou entidade(s) relacionado(s). Exemplos:
  - ▶ o atributo **idade** é derivado a partir da data de nascimento e da data corrente
  - ▶ o atributo **número de empregados** de um departamento pode ser derivado da contagem do número de empregados relacionados (que trabalham) nesse departamento
- ▶ **Atributo armazenado** – é um atributo que não é derivado.

# Notação para tipos de entidades e atributos no modelo ER

## Exemplo de atributo derivado



# Tipos de atributos do modelo ER

## Atributos Complexos

São atributos compostos e multivalorados, que podem ser aninhados de modo arbitrário.

- ▶ **Ex.:** uma pessoa pode ter mais de uma residência, e cada uma delas pode ter múltiplos telefones

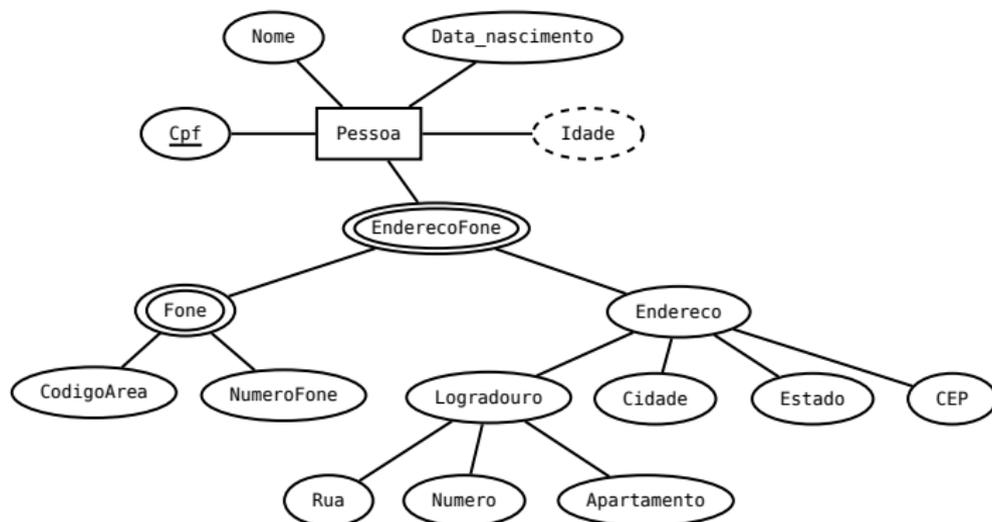
```
{EnderecoFone({Fone(CodigoArea,NumeroFone)},  
               Endereco(Logradouro(Numero,Rua,Apartamento),  
                          Cidade,Estado,CEP))}
```

## Notação

- ▶ {atributo} – atributo multivalorado
- ▶ (atrib1,atrib2,atrib3,...) – atributo composto

# Notação para tipos de entidades e atributos no modelo ER

## Exemplo de atributo complexo



## Domínio de valores dos atributos

- ▶ Cada atributo simples de um tipo de entidade possui um **domínio de valores**, que determina o conjunto de valores válidos que ele pode assumir em cada entidade.
- ▶ Os domínios de valores não são mostrados no diagrama ER.

## Valor NULL (nulo)

- ▶ É um valor especial, usado quando uma entidade **não possui um valor** para um atributo.
- ▶ O NULL serve tanto para indicar que um atributo **não se aplica** a uma dada entidade, quanto para indicar que o valor para um atributo de uma dada entidade é **desconhecido**.
- ▶ “Desconhecido” se aplica a dois casos distintos:
  - ▶ quando é sabido que existe um valor para o atributo, mas ele está faltando (**ex.:** Altura – todo mundo tem!)
  - ▶ quando não é sabido se o valor existe ou não (**ex.:** FoneResidencial – uma pessoa pode ou não ter)

# Tipo de entidade

Um **tipo de entidade** define um conjunto de entidades que possuem os mesmos atributos.

## Conceitos importantes:

- ▶ **Atributo-chave (restrição de exclusividade)** – é um atributo para o qual o valor é distinto para toda entidade do conjunto de entidades do tipo a que pertence.
  - ▶ Os valores do atributo-chave **identificam cada entidade univocamente**.

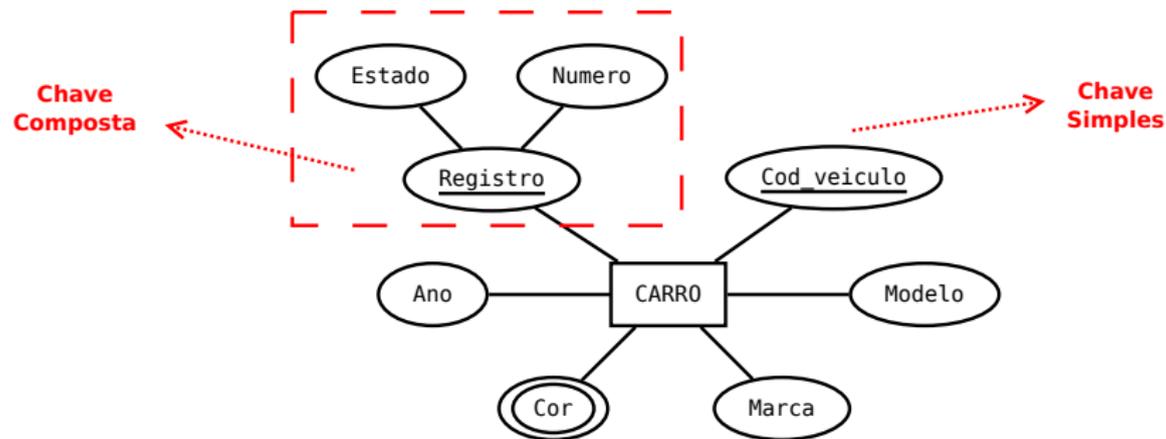
# Atributo-chave de um tipo de entidade

## Conceitos importantes:

- ▶ **Chave composta** – é uma chave formada por diversos atributos, cuja combinação dos valores é distinta para cada entidade. Uma chave composta deve ser mínima, no sentido de que todos os atributos componentes devem ser considerados na composição para garantir a unicidade.
- ▶ Alguns tipos de entidade têm mais de um atributo chave (ex.: os atributos NUSP e CPF para ALUNO). Outros, podem nem ter uma chave – são os chamados **tipos de entidade fraca**.

# Notação para tipos de entidades e atributos no modelo ER

## Exemplo de atributos chave



# Tipos de Entidade

Exemplo de dois tipos de entidade (com chaves simples) e algumas entidades membro de cada um deles

Chave de Empregado: nome

Chave de Empresa: nome

## NOME DO TIPO ENTIDADE:

### EMPREGADO

Nome, Idade, Salario

### EMPRESA

Nome, Sede Administrativa, Presidente

## CONJUNTO DE ENTIDADE: (EXTENSÃO)

$e_1$

(John Smith, 55, 80k)

$e_2$

(Fred Brown, 40, 30K)

$e_3$

(Judy Clark, 25, 20K)

⋮

$c_1$

(Sunco Oil, Houston, John Smith)

$c_2$

(Fast Computer, Dallas, Bob King)

⋮

## Tipos de Entidade

Conjunto de entidades do tipo de entidade CARRO (com uma chave composta e outra simples)

Chaves de Carro: Registro(NumeroRegistro, Estado) ou IdVeiculo

CARRO  
Registro(NumeroRegistro, Estado), IDVeiculo, Marca, Modelo, Ano, {Cor}

car<sub>1</sub> •  
((ABC 123, TEXAS), TK629, Ford Mustang, conversível, 1998, {vermelho, preto})

car<sub>2</sub> •  
((ABC 123, NOVA YORK), WP9872, Nissan Maxima, 4 portas, 1999, {azul})

car<sub>3</sub> •  
((VSY 720, TEXAS), TD729, Chrysler LeBaron, 4 portas, 1995, {branco, azul})

•  
•  
•

## Referências Bibliográficas

- ▶ *Sistemas de Bancos de Dados* (6ª edição), Elmasri e Navathe. Pearson, 2010.  
Capítulo 7
- ▶ *Sistemas de Gerenciamento de Banco de Dados*(3ª edição), Ramakrishnan e Gehrke, 2008.  
Capítulo 2
- ▶ *Projeto e Modelagem de Bancos de Dados*, Teorey, Lightstone e Nadeau, 2007. Capítulo 2

## Cenas dos próximos capítulos...

### Projeto conceitual de BDs:

- ▶ Modelo Entidade-Relacionamento (continuação) – Relacionamentos
- ▶ Modelo Entidade-Relacionamento Estendido (EER)