



# Bases de Dados

## Normalização e Dependências Funcionais - Parte 1

Profa. Elaine Parros Machado de Sousa

# Qualidade do Projeto Lógico

- Como avaliar a **qualidade** do esquema relacional (projeto lógico)?
  - semântica - completude, **consistência**,....
  - implementação – espaço, eficiência de consultas...
- Controle de consistência:
  - 1) no esquema da base de dados
    - estrutura
    - restrições
  - 2) no SGBD
    - *constraints* SQL
    - Procedimentos e *Triggers* (gatilhos)
  - 3) na aplicação
- O que é mais eficaz?



# Qualidade do Projeto Lógico

- Avaliando a qualidade do projeto...
  - **Análise informal**
    - princípios para um bom projeto
  - **Análise formal**
    - Dependências Funcionais
    - Formas Normais
    - Normalização

# Qualidade do Projeto Lógico

- **Análise Informal:**

- **Diretriz 1** => semântica de atributos nas relações
- **Diretriz 2** => redução de valores nulos
  - espaço, interpretação, consultas
- **Diretriz 3** => redução de redundância em tuplas
  - prevenção de anomalias de inserção
  - prevenção de anomalias de remoção
  - prevenção de anomalias de alteração
- **Diretriz 4** => prevenção de geração de tuplas espúrias (tuplas ilegítimas) nas junções

# Qualidade do Projeto Lógico

- **Análise Formal**

- Baseada em **Dependências Funcionais**  $\Rightarrow$  restrições (dependências) entre atributos
  - garantir **consistência** da base de dados:

**NÃO VIOLAR AS DEPENDÊNCIAS FUNCIONAIS**

- **Formas Normais e Normalização**
  - baseadas em Dependências Funcionais
  - avaliação e garantia da qualidade dos esquemas de relação

# Dependências Funcionais

- **Dependência Funcional (DF)** – restrição entre 2 conjuntos de atributos X e Y

$$X \rightarrow Y$$

- **X determina funcionalmente Y** (ou Y depende funcionalmente de X)
  - se  $t_1[X] = t_2[X]$  então  $t_1[Y] = t_2[Y]$

# Dependências Funcionais

- Exemplos

**NUSP**  $\rightarrow$  Nome, Idade, Curso

**Sigla\_Disc**  $\rightarrow$  Nome\_Disc, Créditos

**Sigla\_Disc, NUSP, semestre, ano**  $\rightarrow$  Nota

# Dependências Funcionais

- **Dependência funcional (DF)**
  - propriedade semântica
  - identificada pelo projetista da base de dados
  - deve ser validada na instância da base (nos dados)
  - **nunca deve ser definida (inferida)** a partir dos dados



# Inferindo DFs a partir dos dados....

Alunos = {NUSP, Nome, Curso, Idade}  
{<111, Mario, Comp., 21>,  
<222, Paulo, Eletr., 22>,  
<333, Almir, Fisio., 22>,  
<444, Marta, Comp., 21>,  
<555, Vânia, Eletr., 22>,  
<6666, Mario, Matem., 21>}

- A relação **Alunos** atende **NUSP**→ **Nome**
- Uma DF **NUSP**→ **Nome** teria sentido semântico??

**SIM!!!**

# Inferindo DFs a partir dos dados....

Alunos = {NUSP, Nome, Curso, Idade}  
{<111, Mario, Comp., 21>,  
<222, Paulo, Eletr., 22>,  
<333, Almir, Fisio., 22>,  
<444, Marta, Comp., 21>,  
<555, Vânia, Eletr., 22>,  
<666, Mario, Matem., 21>}

- A relação **Alunos** atende **Curso** → **Idade**
- Uma DF **Curso** → **Idade** teria sentido semântico??

**NÃO!!!!**

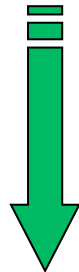
# Garantindo Consistência...

- Intuição...
  - lembrando... **consistência** da base de dados => **não violar as dependências funcionais**
  - ideia: garantir na **construção da base** (no **ESQUEMA**) que as dependências funcionais **não serão violadas**

# Garantindo Consistência

- Modelo Relacional – garantia de consistência na construção da base de dados (i.e. no **ESQUEMA**)

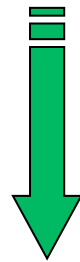
⇒ Qualidade das relações é baseada na análise de **dependências funcionais**



**Formas Normais** (1ª, 2ª, 3ª, 4ª, BCNF, ...)

# Garantindo Consistência

- Uma relação **está em** uma determinada **Forma Normal** quando satisfaz um conjunto de condições baseadas nas **dependências funcionais**
- Colocando uma relação em uma forma normal...



✓ conjunto de DFs para cada relação

✓ condições para Formas Normais

## NORMALIZAÇÃO

# Normalização

- Formas Normais baseadas em dependências funcionais
  - baseadas em chave primária
    - 1ª FN
    - 2ª FN
    - 3ª FN
  - baseadas em chaves candidatas
    - definições genéricas de 2ª FN e 3ª FN
    - FN de Boyce-Codd (BCNF)
- Forma Normal baseada em dependências multivaloradas
  - 4ª FN

**Parte 1**

**Parte 2**

# Normalização

- **Requisito Fundamental:**
  - conhecer todas as dependências funcionais ⇒ **informação semântica fornecida pelo projetista**
  - DFs podem ser inferidas a partir de outras ⇒ **Regras de Inferência**

# Normalização

## Regras de Inferência de DFs:

- **Reflexiva:** se  $Y \subseteq X \Rightarrow X \rightarrow Y$  (**DF trivial**)
- **Transitiva:** se  $X \rightarrow Y, Y \rightarrow Z \Rightarrow X \rightarrow Z$
- **Decomposição:** se  $X \rightarrow YZ \Rightarrow X \rightarrow Y, X \rightarrow Z$
- **Aditiva:** se  $X \rightarrow Y, X \rightarrow Z \Rightarrow X \rightarrow YZ$
- **Aumentativa:** se  $X \rightarrow Y \Rightarrow XZ \rightarrow YZ$
- ....



# Definições iniciais

- Dados os conjuntos de atributos  $X$  e  $Y$ , e um atributo (qualquer)  $a \in X$ :
  - $X \rightarrow Y$  é **dependência funcional total** se  $(X - \{a\})$  não determina  $Y$ 
    - i.e.:  **$Y$  depende, semanticamente, de todo o  $X$**
    - ex:

**Sigla\_Disc, NUSP, semestre, ano  $\rightarrow$  Nota**

DF Total

# Definições iniciais

- Dados os conjuntos de atributos  $X$  e  $Y$ , e um atributo (qualquer)  $a \in X$ :
  - $X \rightarrow Y$  é **dependência funcional parcial** se  $(X - \{a\}) \rightarrow Y$ 
    - i.e.:  $Y$  depende, semanticamente, só de uma parte de  $X$
    - ex:

**Sigla\_Disc, NUSP, semestre, ano  $\rightarrow$  Nome\_Disc**

DF Parcial

# Definições iniciais

- $X \rightarrow Y$  é uma **dependência funcional trivial** se  $Y \subseteq X$
- 

- $X \rightarrow Y$  é uma **dependência funcional transitiva** se existe  $X \rightarrow Z$  e  $Z \rightarrow Y$ , e  $Z$  não é chave candidata
- 

- **Atributo Primário (ou Atributo Principal)** em  $R \Rightarrow$  atributo que faz **parte** de **alguma chave candidata** no esquema de relação  $R$

# 1ª Forma Normal

- **1ª Forma Normal** => todos os atributos da relação devem ser **Atômicos** e **Monovalorados**

- parte da definição formal do Modelo Relacional
- exigida pela maioria dos SGBDRs

# 1ª Forma Normal

- Colocando uma relação na 1ª FN....

⇒ Atributo composto: **Endereço**

Aluno = {Nome, Idade, ~~Endereço~~}



Normalizando...

Aluno = {Nome, Idade, Rua, Nro, Cidade, Estado, CEP}

# 1ª Forma Normal

- Colocando uma relação na 1ª FN....

⇒ Atributo Multivalorado: **Nomes dos Pais**

Aluno = {Nome, ~~Nomes\_Pais~~}



Normalizando...

Aluno = {Nome, Nome\_Pai, Nome\_Mae}

# 1ª Forma Normal

- Colocando uma relação na 1ª FN....

⇒ Atributo Multivalorado: **Alergias**

Aluno = {NUSP, Idade, Alergias}



Normalizando...

Aluno = {NUSP, Idade}

Alergias = {NUSP, Alergia}

# 2ª Forma Normal

- **2ª Forma Normal**
  - relação na 1ª Forma Normal
  - todos os atributos não primários possuem **dependência total**, transitiva ou não, da **chave primária**
    - lembrando: **atributo não primário** é aquele que não faz parte de nenhuma chave candidata...



# 2ª Forma Normal

- DFs identificadas pelo desenvolvedor:

Professor, Sigla → LivroTexto

NúmeroT, Sigla → Sala

Sigla → No.Horas

LivroTexto → LivroExerc

Ministra = {Professor, Sigla, LivroTexto, LivroExerc} **2ª FN: OK!!**

Turma = {NúmeroT, Sigla, Sala, No.Horas} **2ª FN: NÃO!!**

## 2ª Forma Normal: por que **dependência parcial** pode gerar inconsistência?

- DFs :

Professor, Sigla → LivroTexto

NúmeroT, Sigla → Sala

**Sigla → No.Horas**

LivroTexto → LivroExerc

Turma = {NúmeroT, Sigla, Sala, No.Horas}

1, **SCC240**, 5-101, **4** ← OK  
2, **SCC240**, 5-103, **4** ← OK

1, **SCC241**, Lab-5, **4** ← Inconsistência  
2, **SCC241**, Lab-6, **6** ← Inconsistência

# 2ª Forma Normal

- Colocando uma relação na 2ª FN...

Numero, Sigla  $\rightarrow$  Sala

Sigla  $\rightarrow$  No.Horas

Turma = {Numero, Sigla, Sala, No.Horas}

**Parte** da chave que  
determina o grupo de  
atributos **não primários**

Grupo de atributos  
não primários com  
**dependência parcial**

# 2ª Forma Normal

- Colocando uma relação na 2ª FN...

Numero, Sigla  $\rightarrow$  Sala

Sigla  $\rightarrow$  No.Horas

Turma = {Numero, Sigla, Sala, No.Horas}



Normalizando...

**Turma = {Numero, Sigla, Sala}**



**Disciplina = {Sigla, No.Horas}**

# Então...

- DFs :

Professor, Sigla  $\rightarrow$  LivroTexto

NúmeroT, Sigla  $\rightarrow$  Sala

Sigla  $\rightarrow$  No.Horas

LivroTexto  $\rightarrow$  LivroExerc

Turma = {Numero, Sigla, Sala}  $\Rightarrow$  2ª FN

Disciplina = {Sigla, No.Horas}  $\Rightarrow$  2ª FN

Ministra = {Professor, Sigla, LivroTexto, LivroExerc}  $\Rightarrow$  2ª FN

mas... existe ainda uma DF transitiva...

# Por que DF transitiva pode gerar inconsistência?

- DFs :

**Professor, Sigla → LivroTexto**

NúmeroT, Sigla → Sala

Sigla → No.Horas

**LivroTexto → LivroExerc**

Ministra = {Professor, Sigla, LivroTexto, LivroExerc}

Elaine, SCC-240, BD, BD Prática

Elaine, SCC-540, **BD**, **BD Prática**

Cristina, SCC-240, **BD**, **POO Prática**

← Inconsistência

# 3ª Forma Normal

## ■ 3ª Forma Normal:

- relação na 1ª e 2ª Formas Normais
- todos os atributos não primários possuem dependência total, **não transitiva**, da chave primária
  - se existir as DFs  $X \rightarrow Z$  e  $Z \rightarrow Y$ , e se  $Z$  não é chave candidata, então  $X \rightarrow Y$  é **transitiva**

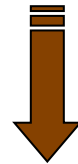
# 3ª Forma Normal

- Colocando uma relação na 3ª FN...

Professor, Sigla  $\rightarrow$  LivroTexto

LivroTexto  $\rightarrow$  LivroExerc

Ministra = {Professor, Sigla, LivroTexto, LivroExerc}



Normalizando...

Ministra = {Professor, Sigla, LivroTexto}



Livro = {LivroTexto, LivroExerc}



# 3ª Forma Normal

Sigla, Número → Sala, Horário

Sigla → LivroTexto, Depto, **ChefeDepto**

Depto → **ChefeDepto**

Turma = {Sigla, Número, Horário, Sala} 3ª FN OK!!

Disciplina = {Sigla, LivroTexto, Depto, ChefeDepto} **NÃO!**

# 3ª Forma Normal

- Colocando uma relação na 3ª FN...

Sigla  $\rightarrow$  LivroTexto, Depto, ChefeDepto

Depto  $\rightarrow$  ChefeDepto

Disciplina = {Sigla, LivroTexto, Depto, ChefeDepto}



Normalizando...

Disciplina = {Sigla, LivroTexto, Depto}

Chefia = {Depto, ChefeDepto}

# Considerações Gerais...

- A 2ª FN e a 3ª FN evitam:
  - **Inconsistência e anomalias** causadas por redundância de informações
  - **Perda de informação** em operações de remoção/alterações na relação

# Considerações Gerais (cont.)...

## ■ Normalização:

- ✓ uma relação por vez
- ✓ **decomposição de relações**
  - aumenta consistência
  - reduz desempenho  $\Rightarrow$  operações de junção

# Considerações Gerais (cont.)...

- Normalização:

- Propriedades desejáveis:

- 1) decomposição **sem perda de junção**  
(sem geração de tuplas ilegítimas)
- 2) decomposição com **preservação de dependências** (possibilidade de avaliar a DF  $\Rightarrow$  atributos na mesma tabela ou relacionados por FK)

# Exemplo...

- **Normalizando para 3ª FN...**

Sigla → LivroTexto, Depto , ChefeDepto

Depto → ChefeDepto

Disciplina = {Sigla, LivroTexto, Depto, ChefeDepto}



Disciplina = {Sigla, LivroTexto, Depto}

Chefia = {Depto, ChefeDepto}

1) SEM PERDA DE JUNÇÃO

2) PRESERVAÇÃO DE DFs

# Sugestão de Leitura

- **ELMASRI, R; NAVATHE, S.B.** – *Sistemas de Banco de Dados*, Addison Wesley
  - 4ª Edição
    - **Capítulo 10** – Dependência Funcional e normalização em um banco de dados relacional
  - 6ª Edição
    - **Capítulo 15** – Fundamentos de dependências funcionais e normalização para bancos de dados relacionais

# EXERCÍCIO

1) Cite e explique as anomalias (inserção, remoção, alteração, geração de tuplas ilegítimas) que podem ocorrer nas relações abaixo. Exemplifique.

a) Piloto\_TipoAviao = {NroLicençaPiloto, CpfPiloto, NomePiloto, EndereçoPiloto, TipoAviao, FabricanteTipoAviao, NroMaxPoltronas}

b) Aluno = {Nusp, Idade, Curso}

Matrícula = {NomeAluno, Disciplina, Curso}



# EXERCÍCIO

- 2) Para a seguinte relação e suas dependências funcionais, indique se as formas normais 1FN, 2FN e 3FN são atendidas (e o porquê).  
Normalize a relação para atender a cada uma dessas formas.

**Peça = {codigo, fornecedor, cidadeForn, estadoForn, qtdEstoque, peso, custoTransporte, valor}**

código, fornecedor  $\rightarrow$  valor;

fornecedor  $\rightarrow$  cidadeForn, estadoForn, custoTransporte;

código  $\rightarrow$  peso, qtdEstoque;

cidadeForn, estadoForn  $\rightarrow$  custoTransporte;