

SCC-540 Bases de Dados

Prof. Moacir P Ponti Jr.

Material da Profa. Elaine Parros Machado de Sousa

Normalização e Dependências Funcionais - Parte 1

Qualidade do Projeto Lógico

- Como avaliar a qualidade os esquemas de relação (projeto lógico)?
 - semântica?
 - implementação?
- Análise informal
 - princípios para um bom projeto
- Análise formal
 - dependência funcional
 - normalização

Qualidade do Projeto Lógico

- **Análise Informal:**
 - semântica de atributos
 - redução de redundância em tuplas
 - prevenção de anomalias de inserção
 - prevenção de anomalias de remoção
 - prevenção de anomalias de alteração
 - redução de valores nulos
 - prevenção de geração de tuplas espúrias (tuplas ilegítimas) nas junções

Qualidade do Projeto Lógico

- **Análise Formal**
 - Baseada em **Dependências Funcionais** \Rightarrow restrições entre atributos
 - garantia de **consistência** da base de dados
 - avaliação da qualidade dos esquemas de relação

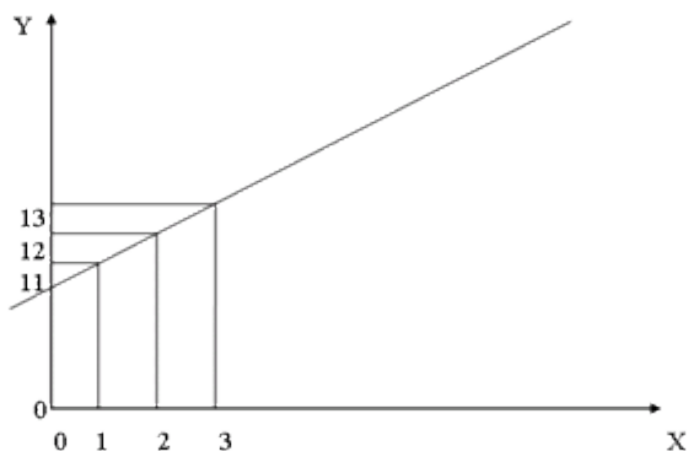
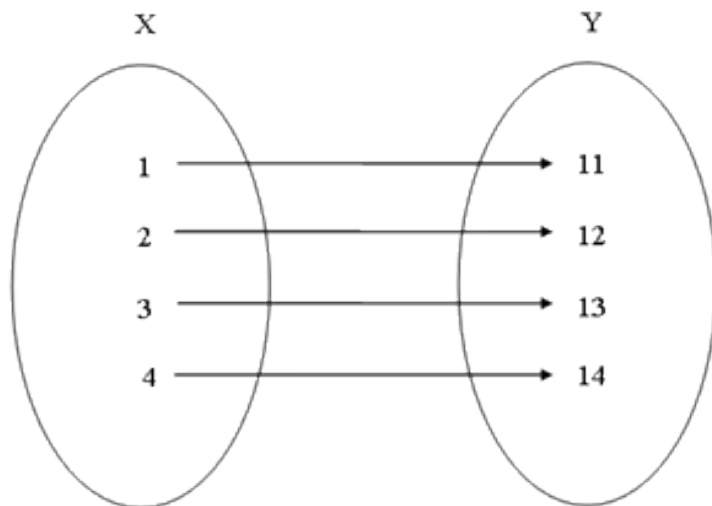
Controlando Consistência

- Controle de consistência:
 - no esquema da base de dados (construção da base)
 - no SGBD
 - no aplicativo
- O que é mais eficiente?

Controlando Consistência

- Modelo Relacional
 - natureza inerentemente formal – conceitos formais para controle de consistência
 - na construção da base:
 - ⇒ definição das relações (tabelas) é baseada na análise de **dependências funcionais**

Dependências funcionais

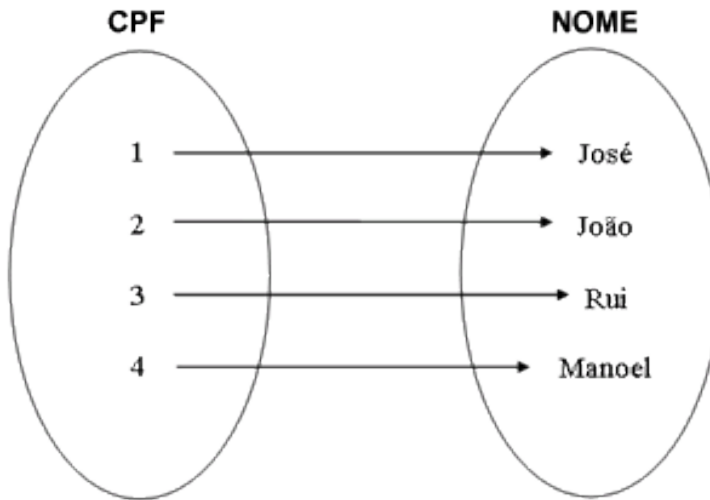


- Existe uma dependência entre os valores dos conjuntos X e Y
- Ela pode ser expressa por:

$$f(x) = x + 10$$

- Assim, y é função de x , já que $y = f(x)$

Dependências funcionais



- Nesse exemplo, temos que:

$$f(\text{CPF}) = \text{NOME}$$

- Podemos interpretar que nome é função de CPF, ou que nome **depende funcionalmente de CPF**:

CPF \rightarrow NOME

Dependências Funcionais

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

$$A \rightarrow B$$

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

Dependências Funcionais

- Exemplos

NUSP \rightarrow Nome, Idade, Curso

Sigla_Disc, NUSP, semestre, ano \rightarrow Nota

Sigla_Disc \rightarrow Nome_Disc, Créditos

Dependências Funcionais

- **Dependência funcional (DF)**
 - propriedade semântica
 - identificada pelo projetista da base de dados
 - pode ser validada na instância da base mas **nunca definida** a partir dela

Dependências Funcionais

Alunos = {Nome, Curso, Idade}
{<Mario, Comp., 21>,
 <Paulo, Eletr., 22>,
 <Almir, Fisio., 22>,
 <Marta, Comp., 21>,
 <Vânia, Eletr., 22>}

A relação **Alunos** atende às seguintes DFs?

Nome \rightarrow Curso

Nome \rightarrow Idade

OK!!!

Dependências Funcionais

Alunos = {Nome, Curso, Idade}
{<Mario, Comp., 21>,
 <Paulo, Eletr., 22>,
 <Almir, Fisio., 22>,
 <Marta, Comp., 21>,
 <Vânia, Eletr., 22>}

A relação **Alunos** atende à seguinte DF?

Curso \rightarrow Idade

OK para esta instância!!!
Mas tem sentido semântico???

Dependências Funcionais

Alunos = {Nome, Curso, Idade}
{<Mario, Comp., 21>,
 <Paulo, Eletr., 22>,
 <Almir, Fisio., 22>,
 <Marta, Comp., 21>,
 <Vânia, Eletr., 22>}

A relação **Alunos** atende à seguinte DF?

Idade \rightarrow Curso

Não! E teria sentido semântico?

Dependências Funcionais e Controle de Consistência

- Controle de consistência
 - necessário conhecer todas as dependências funcionais \Rightarrow **informação semântica fornecida pelo projetista**
 - DFs podem ser inferidas a partir de outras \Rightarrow **regras de inferência**

Dependências Funcionais

- Regras de Inferência de DFs:
 - **Reflexiva:** se $A \subseteq B \Rightarrow B \rightarrow A$ (**DF trivial**)
 - **Aumentativa:** se $A \rightarrow B \Rightarrow AC \rightarrow BC$
 - **Decomposição :** se $A \rightarrow BC \Rightarrow A \rightarrow B, A \rightarrow C$
 - **Aditiva:** se $A \rightarrow B, A \rightarrow C \Rightarrow A \rightarrow BC$
 - **Transitiva:** se $A \rightarrow B, B \rightarrow C \Rightarrow A \rightarrow C$
 - **Pseudo-Transitiva:** se $A \rightarrow B, BC \rightarrow D \Rightarrow AC \rightarrow D$

Controlando Consistência

- Intuição:
 - na construção da base:
 - ⇒ Como garantir consistência?
 - ⇒ **Como garantir que as dependências funcionais não serão violadas?**

Controlando Consistência

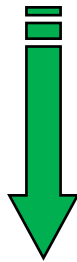
- Modelo Relacional
 - na construção da base:
 - ⇒ definição das relações é baseada na análise de dependências funcionais



Formas Normais

Controlando Consistência

- Uma relação **está em** uma determinada **Forma Normal** quando satisfaz certas propriedades, baseadas nas **dependências funcionais**
- colocar uma relação em uma forma normal



- ✓ conjunto de DFs para cada relação
- ✓ condições para formas normais

NORMALIZAÇÃO

Normalização

- Normalização de Relações
 - baseada nas dependências funcionais
 - garante consistência na construção da base
 - redução de redundância
 - redução de anomalias

Normalização

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

Normalização

- Formas Normais baseadas em dependências funcionais
 - **baseadas em chave primária**
 - 1ª FN
 - 2ª FN
 - 3ª FN

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 $(X - \{a\}) \rightarrow Y$
 - $X \rightarrow Y$ é **dependência funcional total** se $(X - \{a\})$ não determina Y

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 parte de nenhuma chave candidata
- **Atributo Primário** em R – atributo que faz parte de alguma chave candidata no esquema de relação R

1ª Forma Normal

- **1ª Forma Normal**

- parte da definição formal do Modelo Relacional
- todos os atributos da relação devem ser **Atômicos e Monovalorados**

1ª Forma Normal

- Colocando uma relação na 1ª FN....

⇒ Atributos não atômicos: **DataNasc, Endereço**

Aluno = {Nome, Idade, ~~DataNasc.~~, ~~Endereço~~}



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

1ª Forma Normal

- Colocando uma relação na 1ª FN....

⇒ Atributo Multivalorado: **Nomes dos Pais**

Aluno = {Nome, ~~Nomes_Pais~~}



Aluno = {Nome, Nome_Pai, Nome_Mae}

1ª Forma Normal

- Colocando uma relação na 1ª FN....

⇒ Atributo Multivalorado: **Disciplinas**

Aluno = {NUSP, Idade, ~~Disciplinas~~}



Aluno = {NUSP, Idade}

Disciplinas = {NUSP, Disciplina}

Primeira Forma Normal (1FN)

- A 1FN não permite que defina-se atributos que possam possuir dados multivalorados ou compostos. Todos os domínios devem ser atômicos (indivisíveis) – o que impede atributos multivalorados

FORNECEDOR

CodFor

RazaoSoc

Contatos

Fornecedor

CodFornecedor
RazaoSocial
Contatos

Fornecedor

<u>CodCli</u>	RazaoSoc	Contatos
1	ElCom Industria	João
2	Campus Editora	Tatiana, Gabriel
3	Rapido Transporte	
4	Genius Tecnologia	Hélio, Marcelo, Ana
5	MXe Informática	Lilian

Primeira Forma Normal (1FN)

- Como atingir a 1FN:
 - a) Identificar a chave primária da entidade;
 - b) Identificar o grupo repetitivo e excluí-lo da entidade;
 - c) Criar uma nova entidade com a chave primária da entidade anterior e o grupo repetitivo.

Não Normalizado

Fornecedor

(# CodFornecedor
RazaoSocial
Contatos)

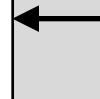
Normalizado pela 1FN

Fornecedor

CodFornecedor
RazaoSocial

Fornecedor_Contato

CodContato
CodFornecedor (FK)
NomeContato



1ª Forma Normal

■ 1ª Forma Normal

- exigida pela maioria dos SGBDRs
- NF² (*Non First Normal Form*) \Rightarrow SGBDs (e a teoria correspondente) que não exigem a 1ª Forma Normal
 - “*Nested Relations*” \Rightarrow relações que não estão em 1ª FN

2ª Forma Normal

Todos os atributos não primários possuem:

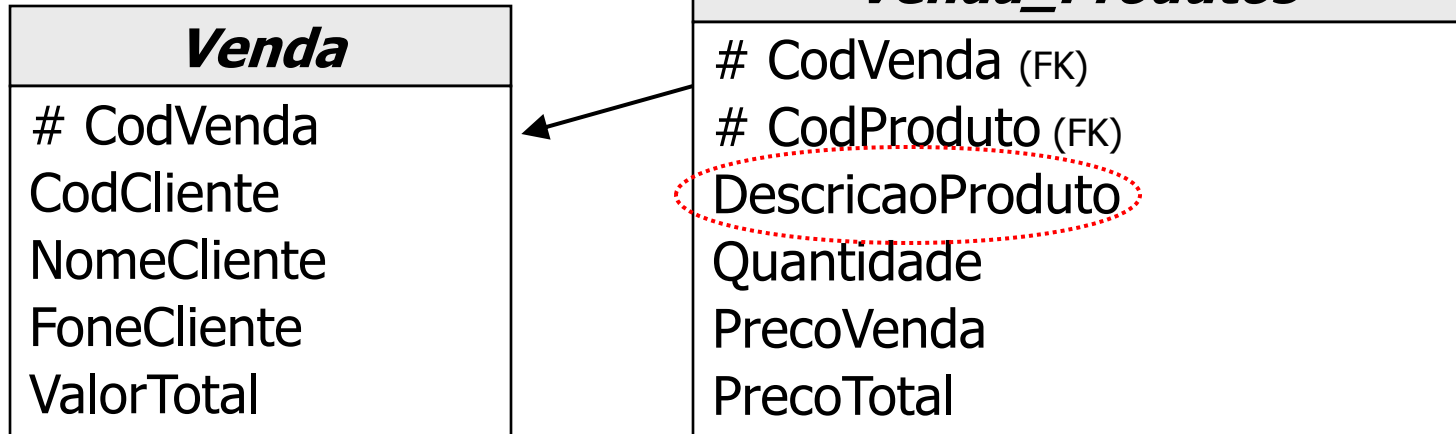
- **dependência total da chave primária, transitiva ou não.**

Segunda Forma Normal (2FN)

- De maneira geral, uma relação (tabela) está na 2FN se está na 1FN, e:
 - Todo atributo não-chave é totalmente dependente da chave-primária da relação, e não só de *parte* da chave-primária
- Esta normalização geralmente é usada em relações que tenham *chaves primárias compostas*

Segunda Forma Normal (2FN)

- Exemplo



- Dependências Funcionais (DFs) em Venda_Produtos:

1 **(CodVenda,CodProduto) → Quantidade**

2 **(CodVenda,CodProduto) → PrecoVenda**

3 **(CodVenda,CodProduto) → PrecoTotal**

4 **CodProduto → DescricaoProduto** →

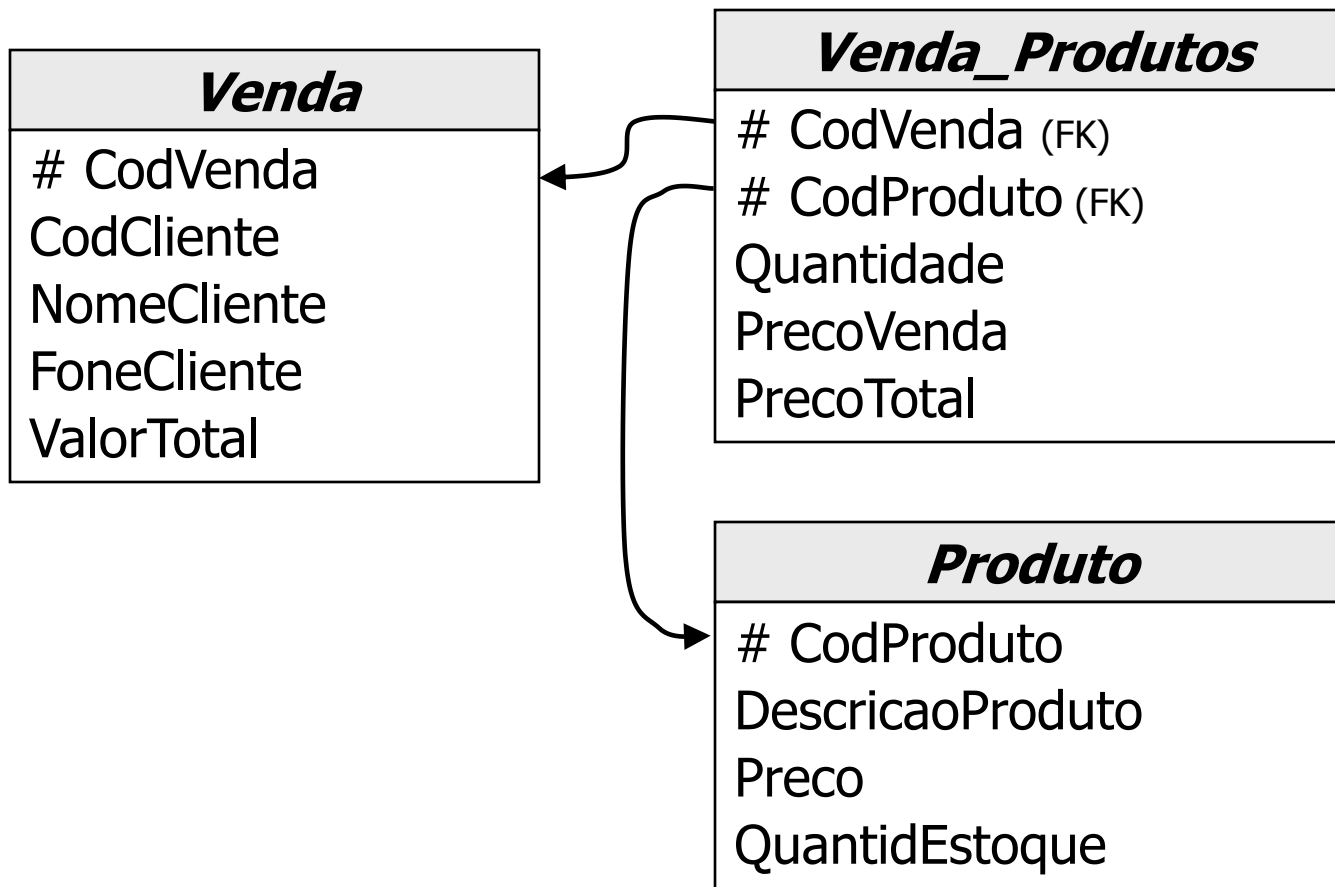
Este atributo é apenas
parcialmente dependente
da chave

Segunda Forma Normal (2FN)

- Para normalizar segundo a 2FN é preciso:
 1. Identificar os atributos não dependentes totalmente da chave primária
 2. Remover estes atributos e transferí-los para outra relação.
 3. Caso não exista uma relação que possa recebê-los, uma nova deve ser criada

Segunda Forma Normal (2FN)

- Resolução:
 - *DescricaoProduto* foi transferido para a relação Produto



2ª Forma Normal

- DFs identificadas pelo desenvolvedor:

Professor, Sigla → LivroTexto

NúmeroT, Sigla → Sala

Sigla → No.Horas

LivroTexto → LivroExerc

Ministra = {Professor, Sigla, LivroTexto, LivroExerc}



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



2ª Forma Normal

■ Colocando uma relação na 2ª FN...

Numero, Sigla → Sala

Sigla → 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}

Disciplina = {Sigla, No.Horas}



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
 - **obs:** se $X \rightarrow Y$ é não transitiva, **então não podem** existir as DFs $X \rightarrow Z$ e $Z \rightarrow Y$

Exemplo

Sigla, Número \rightarrow Sala

Sala \rightarrow Prédio

Sigla, Número \rightarrow Prédio (por transitividade)

Turma = {Sigla, Número, Sala, Prédio} – **2ª FN**

Inconsistência

SFI102, 2, 4, E1
SFI102, 1, 3, E1
SFI104, 1, 12, C2
SFI155, 1, 4, C2
SFI155, 2, 12, C2

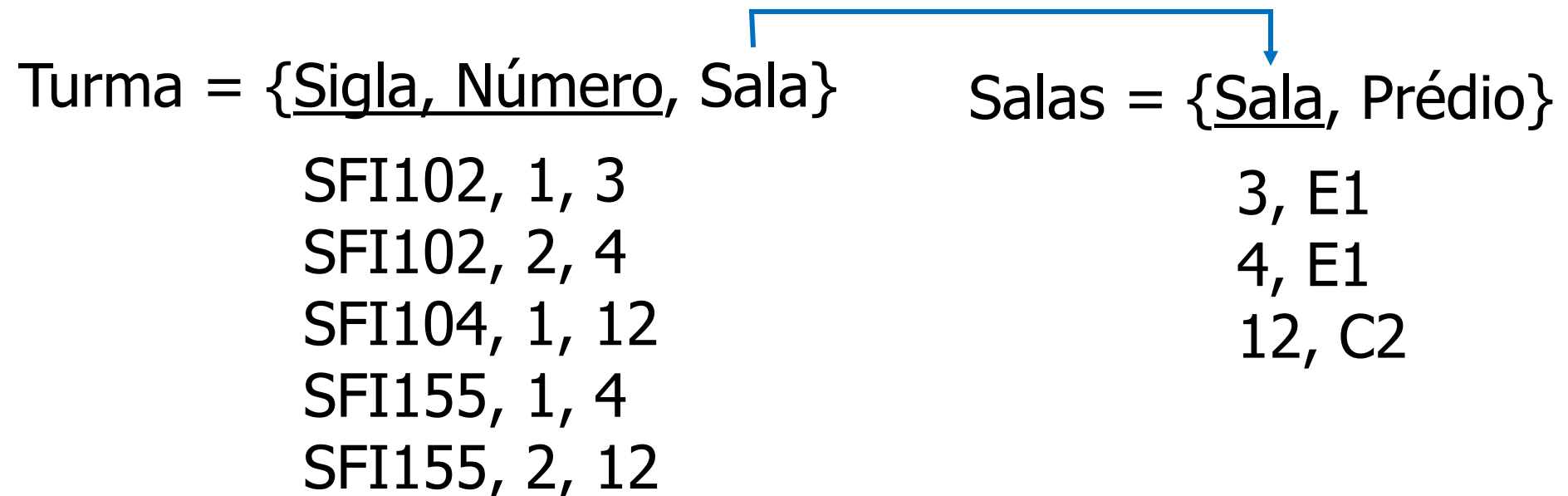
OK

Exemplo - Normalizando....

- **Normalizado segundo a 3ª FN:**

Número, Sigla → Sala

Sala → Prédio



3ª Forma Normal

Sigla, Número → Sala, Horário

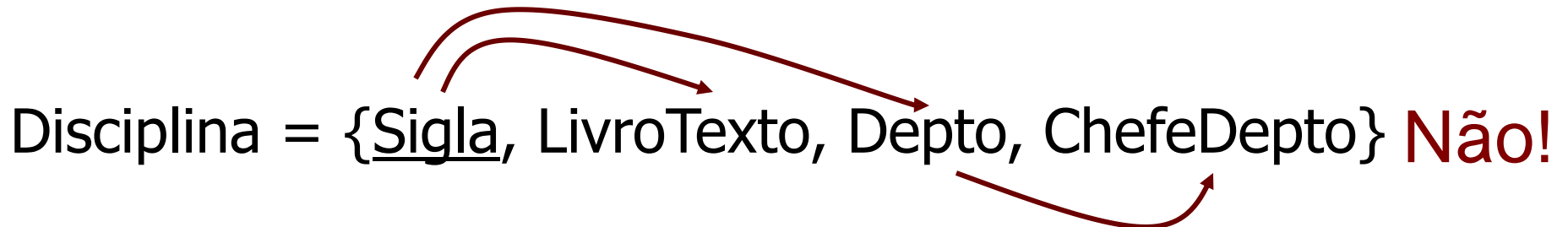
Sigla → LivroTexto, Depto

Depto → ChefeDepto

Turma = {Sigla, Número, Horário, Sala} **OK!!**



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



3ª Forma Normal

- Colocando uma relação na 3ª FN...

Sigla \rightarrow LivroTexto, Depto

Depto \rightarrow ChefeDepto

Disciplina = {Sigla, LivroTexto, Depto, ~~ChefeDepto~~}



Disciplina = {Sigla, LivroTexto, Depto}

Chefia = {Depto, ChefeDepto}

3ª Forma Normal

- Colocando uma relação na 3ª FN...

Professor, Sigla \rightarrow LivroTexto

LivroTexto \rightarrow LivroExerc

Ministra = {Professor, Sigla, LivroTexto, LivroExerc}



Ministra = {Professor, Sigla, LivroTexto}



Livro = {LivroTexto, LivroExerc}

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
 - **forma normal de uma relação** \Rightarrow forma normal mais restrita atendida
 - **forma normal da base de dados** \Rightarrow forma normal mais restrita atendida por todas as relações

Considerações Gerais (cont.)...

- Normalização:

- “quebra” as relações, criando outras \Rightarrow
decomposição

- Propriedades desejáveis:

- decomposição sem perda de junção (sem geração de tuplas ilegítimas)
 - decomposição com preservação de dependências (possibilidade de avaliar a DF)

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