Apresentação baseada em:

- Ramez Elmasri e Shamkant B. Navathe 6° Ed (2010)
- Material da Profa. Dra. Sarajane Marques Peres (UNIVESP)

Banco de Dados

Diretrizes informais para um bom projeto de banco de dados

FACOM – UFMS

Vanessa Borges

vanessa.a.borges@ufms.br

Capítulo 15: Fundamentos de dependências funcionais e normalização para bancos de dados relacionais

Projeto de banco de dados

- Ao projetarmos um modelo para um banco de dados, após adquirir alguma experiência, torna-se natural a **organização de atributos** e relações de uma maneira lógica

- No entanto, ainda assim é necessário atestar tal adequabilidade com o objetivo de **garantir a integridade de modelo e a consistência dos dados**
  - Preservando a informação
  - Minimizando a redundância

Projeto de banco de dados

- **Critérios informais para determinar a qualidade de um esquema de uma relação**
  1. Garantir que a semântica dos atributos seja clara no esquema
  2. Reduzir a informação redundante nas tuplas das relações
  3. Reduzir a ocorrência de valores NULL nas tuplas
  4. Anular a possibilidade de gerar tuplas falsas

Projeto de banco de dados

1. Garantir que a **semântica dos atributos** seja clara no esquema de uma relação
   - **Sempre que agrupamos atributos para formar um esquema de uma relação consideramos que aqueles atributos precisam possuir algum significado no mundo real**

Projeto de banco de dados

1. Garantir que a **semântica dos atributos** seja clara no esquema de uma relação
   - O significado de uma **relação** está fortemente associado à **interpretação dos valores dos atributos em um tupla**
   - Os esquemas deveriam **minimizar a possibilidade de ambiguidade de interpretações**
   - A facilidade em explicar o significa dos atributos em uma relação pode ser tomada como uma medida de qualidade do projeto

FUNCIONARIO

Fnome	Cpf	Datanasc	Endereco	Dnumero
	ChE			

DEPARTAMENTO

Dnome	Dnumero	Cpf_gerente
	ChP	ChE

DEP_LOCALIZACAO

Dnumero	Dlocal
ChE	
ChP	

PROJETO

Projnome	Projnumero	Projlocal	Dnum
	ChP		ChE

TRABALHA_EM

Cpf	Projnumero	Horas
ChE	ChE	
ChP		

Diretrizes de projetos informacionais para esquemas de relação

1. **Garantir que a semântica dos atributos seja clara no esquema**
   - Significado associado aos atributos
   - Como interpretar os valores de atributo armazenados numa tupla da relação

Regra 1:

Projete um esquema de relação de tal modo que **fique fácil explicar o seu significado** (semântica clara). Não combine atributos de vários tipos entidade e tipos relacionamento em uma única relação.

Diretrizes de projetos informacionais para esquemas de relação

1. **Garantir que a semântica dos atributos seja clara no esquema**
   - **Não devemos combinar atributos de múltiplas entidades e relacionamentos em uma única relação**
     - **Exemplos** de mistura de entidades distintas no mundo real **(projeto de baixa qualidade)**:

FUNC_DEP

Fnome	Cpf	Datanasc	Endereco	Dnumero	Dnome	Cpf_gerente

FUNC_PROJ

Cpf	Projnumero	Horas	Fnome	Projnome	Projlocal

- Obs.: As relações acima podem ser usadas como visões (consultas), **mas causam problemas como relações base**

Diretrizes de projetos informacionais para esquemas de relação

2. Reduzir a informação redundante nas tuplas das relações

- A forma como atributos são agrupados em uma relação:
  - Tem efeito importante em relação ao **espaço utilizado para armazenamento**
  - Pode causar **anomalias** de atualização na **inserção**, **exclusão** e **modificação** de valores

Diretrizes de projetos informacionais para esquemas de relação

2. Reduzir a informação redundante nas tuplas das relações

- **Anomalia na <u>inserção</u>:**
  - **Como inserir um departamento que não possui funcionários?**
  - **Como inserir um novo funcionário para o departamento 5?**
    - Como garantir que os valores correspondentes ao departamento 5 estão coerentes?

Redundância

FUNC_DEP

Fnome	Cpf	Datanasc	Endereco	Dnumero	Dnome	Cpf_gerente
Silva, João B.	12345678966	09-01-1965	Rua das Flores, 751, São Paulo, SP	5	Pesquisa	33344555587
Wong, Fernando T.	33344555587	08-12-1955	Rua da Lapa, 34, São Paulo, SP	5	Pesquisa	33344555587
Zelaya, Alice J.	99988777767	19-01-1968	Rua Souza Lima, 35, Curitiba, PR	4	Administração	98765432168
Souza, Jennifer S.	98765432168	20-06-1941	Av. Arthur de Lima, 54, Santo André, SP	4	Administração	98765432168
Lima, Ronaldo K.	66688444476	15-09-1962	Rua Rebouças, 65, Piracicaba, SP	5	Pesquisa	33344555587
Leite, Joice A.	45345345376	31-07-1972	Av. Lucas Obes, 74, São Paulo, SP	5	Pesquisa	33344555587
Pereira, André V.	98798798733	29-03-1969	Rua Timbira, 35, São Paulo, SP	4	Administração	98765432168
Brito, Jorge E.	88866555576	10-11-1937	Rua do Horto, 35, São Paulo, SP	1	Matriz	88866555576

Diretrizes de projetos informacionais para esquemas de relação

2. Reduzir a informação redundante nas tuplas das relações

- **Anomalia na inserção:**
  - A inserção de novos projetos associados a um mesmo funcionário exige a repetição da informação sobre o funcionário
  - A associação de mais de um funcionário a um projeto exige a repetição da informação sobre o projeto

Redundância — Redundância

FUNC_PROJ

Cpf	Projnumero	Horas	Fnome	Projnome	Projlocal
12345678966	1	32,5	Silva, João B.	ProdutoX	Santo André
12345678966	2	7,5	Silva, João B.	ProdutoY	Itu
66688444476	3	40,0	Lima, Ronaldo K.	ProdutoZ	São Paulo
45345345376	1	20,0	Leite, Joice A.	ProdutoX	Santo André
45345345376	2	20,0	Leite, Joice A.	ProdutoY	Itu

Diretrizes de projetos informacionais para esquemas de relação

2. Reduzir a informação redundante nas tuplas das relações

- **Anomalia na <u>remoção</u>:**
  - **Se a tupla referente ao funcionário Brito for excluída do sistema, as informações sobre o departamento Matriz deixarão de existir.**

FUNC_DEP

Redundância

Fnome	Cpf	Datanasc	Endereco	Dnumero	Dnome	Cpf_gerente
Silva, João B.	12345678966	09-01-1965	Rua das Flores, 751, São Paulo, SP	5	Pesquisa	33344555587
Wong, Fernando T.	33344555587	08-12-1955	Rua da Lapa, 34, São Paulo, SP	5	Pesquisa	33344555587
Zelaya, Alice J.	99988777767	19-01-1968	Rua Souza Lima, 35, Curitiba, PR	4	Administração	98765432168
Souza, Jennifer S.	98765432168	20-06-1941	Av. Arthur de Lima, 54, Santo André, SP	4	Administração	98765432168
Lima, Ronaldo K.	66688444476	15-09-1962	Rua Rebouças, 65, Piracicaba, SP	5	Pesquisa	33344555587
Leite, Joice A.	45345345376	31-07-1972	Av. Lucas Obes, 74, São Paulo, SP	5	Pesquisa	33344555587
Pereira, André V.	98798798733	29-03-1969	Rua Timbira, 35, São Paulo, SP	4	Administração	98765432168
Brito, Jorge E.	88866555576	10-11-1937	Rua do Horto, 35, São Paulo, SP	1	Matriz	88866555576

Diretrizes de projetos informacionais para esquemas de relação

2. Reduzir a informação redundante nas tuplas das relações

- **Anomalia na <u>modificação</u>:**
  - **Se atualizarmos o valor de qualquer atributo do departamento 5, será necessário atualizar as tuplas de todos os funcionários daquele departamento.**

Redundância

FUNC_DEP

Fnome	Cpf	Datanasc	Endereco	Dnumero	Dnome	Cpf_gerente
Silva, João B.	12345678966	09-01-1965	Rua das Flores, 751, São Paulo, SP	5	Pesquisa	33344555587
Wong, Fernando T.	33344555587	08-12-1955	Rua da Lapa, 34, São Paulo, SP	5	Pesquisa	33344555587
Zelaya, Alice J.	99988777767	19-01-1968	Rua Souza Lima, 35, Curitiba, PR	4	Administração	98765432168
Souza, Jennifer S.	98765432168	20-06-1941	Av. Arthur de Lima, 54, Santo André, SP	4	Administração	98765432168
Lima, Ronaldo K.	66688444476	15-09-1962	Rua Rebouças, 65, Piracicaba, SP	5	Pesquisa	33344555587
Leite, Joice A.	45345345376	31-07-1972	Av. Lucas Obes, 74, São Paulo, SP	5	Pesquisa	33344555587
Pereira, André V.	98798798733	29-03-1969	Rua Timbira, 35, São Paulo, SP	4	Administração	98765432168
Brito, Jorge E.	88866555576	10-11-1937	Rua do Horto, 35, São Paulo, SP	1	Matriz	88866555576

Diretrizes de projetos informacionais para esquemas de relação

2. Reduzir a informação redundante nas tuplas das relações

Qual ocupa mais espaço de armazenamento?

FUNCIONARIO

Fnome	Cpf	Datanasc	Endereco	Dnumero
Silva, Joao B.	12345678966	09-01-1965	Rua das Flores, 751, São Paulo, SP	5
Wong, Fernando T.	33344555587	08-12-1955	Rua da Lapa, 34, São Paulo, SP	5
Zelaya, Alice J.	99988777767	19-01-1968	Rua Souza Lima, 35, Curitiba, PR	4
Souza, Jennifer S.	98765432168	20-06-1941	Av. Arthur de Lima, 54, Santo André, SP	4
Lima, Ronaldo K.	66688444476	15-09-1962	Rua Rebouças, 65, Piracicaba, SP	5
Leite, Joice A.	45345345376	31-07-1972	Av. Lucas Obes, 74, São Paulo, SP	5
Pereira, André V.	98798798733	29-03-1969	Rua Timbira, 35, São Paulo, SP	4
Brito, Jorge E.	88866555576	10-11-1937	Rua do Horto, 35, São Paulo, SP	1

DEPARTAMENTO

Dnome	Dnumero	Cpf_gerente
Pesquisa	5	33344555587
Administração	4	98765432168
Matriz	1	88866555576

FUNC_DEP

Fnome	Cpf	Datanasc	Endereco	Dnumero	Dnome	Cpf_gerente
Silva, João B.	12345678966	09-01-1965	Rua das Flores, 751, São Paulo, SP	5	Pesquisa	33344555587
Wong, Fernando T.	33344555587	08-12-1955	Rua da Lapa, 34, São Paulo, SP	5	Pesquisa	33344555587
Zelaya, Alice J.	99988777767	19-01-1968	Rua Souza Lima, 35, Curitiba, PR	4	Administração	98765432168
Souza, Jennifer S.	98765432168	20-06-1941	Av. Arthur de Lima, 54, Santo André, SP	4	Administração	98765432168
Lima, Ronaldo K.	66688444476	15-09-1962	Rua Rebouças, 65, Piracicaba, SP	5	Pesquisa	33344555587
Leite, Joice A.	45345345376	31-07-1972	Av. Lucas Obes, 74, São Paulo, SP	5	Pesquisa	33344555587
Pereira, André V.	98798798733	29-03-1969	Rua Timbira, 35, São Paulo, SP	4	Administração	98765432168
Brito, Jorge E.	88866555576	10-11-1937	Rua do Horto, 35, São Paulo, SP	1	Matriz	88866555576

Diretrizes de projetos informacionais para esquemas de relação

2. Reduzir a informação redundante nas tuplas das relações

Regra 2:

Projete os esquemas de relação de modo que nenhuma anomalia de inserção, exclusão ou modificação esteja presente nas relações.
Se houver alguma anomalia, anote-as claramente e cuide para que os programas que atualizam o banco de dados operem corretamente.

- Em geral, é aconselhável usar relações da base **sem anomalias** e especificar **visões que incluem as junções para reunir os atributos frequentemente referenciados** nas consultas importantes.

Diretrizes de projetos informacionais para esquemas de relação

3. Reduzir a ocorrência de valores NULL nas tuplas

- Desperdício de **espaço de armazenamento**
- Levam a problemas de **entendimento do significado** do atributo
- Causam problemas de interpretação de funcionamento de **operações de junção** e de **funções agregadas**

Diretrizes de projetos informacionais para esquemas de relação

3. Reduzir a ocorrência de valores NULL nas tuplas

- **Exemplo:**
  - Deseja-se adicionar um atributo com a **quantidade de escritórios individuais** de cada funcionário
    - Se apenas 15 por cento dos funcionários têm escritórios individuais, há pouca justificativa para incluir um atributo numero_escritorio na relação **FUNCIONARIO**.
      - Em vez disso, uma relação FUNC_ESCRITORIO (fcpf, numero_escritorio) pode ser criada para incluir tuplas apenas para funcionários com escritórios individuais.

FUNCIONARIO

Fnome	Cpf	Datanasc	Endereco	Dnumero
Silva, Joao B.	12345678966	09-01-1965	Rua das Flores, 751, São Paulo, SP	5
Wong, Fernando T.	33344555587	08-12-1955	Rua da Lapa, 34, São Paulo, SP	5
Zelaya, Alice J.	99988777767	19-01-1968	Rua Souza Lima, 35, Curitiba, PR	4
Souza, Jennifer S.	98765432168	20-06-1941	Av. Arthur de Lima, 54, Santo André, SP	4
Lima, Ronaldo K.	66688444476	15-09-1962	Rua Rebouças, 65, Piracicaba, SP	5
Leite, Joice A.	45345345376	31-07-1972	Av. Lucas Obes, 74, São Paulo, SP	5
Pereira, André V.	98798798733	29-03-1969	Rua Timbira, 35, São Paulo, SP	4
Brito, Jorge E.	88866555576	10-11-1937	Rua do Horto, 35, São Paulo, SP	1

Diretrizes de projetos informacionais para esquemas de relação

3. Reduzir a ocorrência de valores NULL nas tuplas

Regra 3:

Ao máximo possível, evite colocar atributos em uma relação da base cujos valores podem ser NULL com frequência.
Se os NULOS forem inevitáveis, garanta que eles se apliquem apenas em casos excepcionais, e não à maioria das tuplas na relação.

Atributos que possuem valor NULL frequentemente podem ser colocados em relações separadas (com uma chave primária).

Diretrizes de projetos informacionais para esquemas de relação

4. Anular a possibilidade de gerar tuplas falsas (ou espúrias)

- Tuplas falsas são tuplas adicionais que são criadas a partir de procedimentos de recuperação de informação mal sucedidos

- **Exemplo:**
  - Sabe-se que FUNC_PROJ é ruim. Imagine que alguém tentou melhorar o projeto criando duas relações para substituir FUNC_PROJ : FUNC_LOCAL e FUNC_PROJ1

FUNC_PROJ

Cpf	Projnumero	Horas	Fnome	Projnome	Projlocal

FUNC_LOCAL

Fnome	Projlocal

ChP

(b)

FUNC_PROJ1

Cpf	Projnumero	Horas	Projnome	Projlocal

ChP

Diretrizes de projetos informacionais para esquemas de relação

4. Anular a possibilidade de gerar tuplas falsas (ou espúrias)

- **Exemplo:**
  - Se tentarmos uma operação JUNÇÃO NATURAL sobre FUNC_PROJ1 e FUNC_LOCAL, o resultado produz muito mais tuplas do que o conjunto original de tuplas em FUNC_PROJ.

FUNC_LOCAL

Fnome	Projlocal
Silva, João B.	Santo André
Silva, João B.	Itu
Lima, Ronaldo K.	São Paulo
Leite, Joice A.	Santo André
Leite, Joice A.	Itu
Wong, Fernando T.	Itu
Wong, Fernando T.	São Paulo
Wong, Fernando T.	Mauá

FUNC_PROJ1

Cpf	Projnumero	Horas	Projnome	Projlocal
12345678966	1	32,5	ProdutoX	Santo André
12345678966	2	7,5	ProdutoY	Itu
66688444476	3	40,0	ProdutoZ	São Paulo
45345345376	1	20,0	ProdutoX	Santo André
45345345376	2	20,0	ProdutoY	Itu
33344555587	2	10,0	ProdutoY	Itu
33344555587	3	10,0	ProdutoZ	São Paulo
33344555587	10	10,0	Computadorização	Mauá
33344555587	20	10,0	Reorganização	São Paulo

Diretrizes de projetos informacionais para esquemas de relação

4. Anular a possibilidade de gerar tuplas falsas (ou espúrias)

- **Exemplo:**

FUNC_PROJ

Cpf	Projnumero	Horas	Fnome	Projnome	Projlocal
12345678966	1	32,5	Silva, João B.	ProdutoX	Santo André
12345678966	2	7,5	Silva, João B.	ProdutoY	Itu
66688444476	3	40,0	Lima, Ronaldo K.	ProdutoZ	São Paulo
45345345376	1	20,0	Leite, Joice A.	ProdutoX	Santo André
45345345376	2	20,0	Leite, Joice A.	ProdutoY	Itu
33344555587	2	10,0	Wong, Fernando T.	ProdutoY	Itu
33344555587	3	10,0	Wong, Fernando T.	ProdutoZ	São Paulo
33344555587	10	10,0	Wong, Fernando T.	Informatização	Mauá
33344555587	20	10,0	Wong, Fernando T.	Reorganização	São Paulo
99988777767	30	30,0	Zelaya, Alice J.	Novosbenefícios	Mauá
99988777767	10	10,0	Zelaya, Alice J.	Informatização	Mauá
98798798733	10	35,0	Pereira, André V.	Informatização	Mauá
98798798733	30	5,0	Pereira, André V.	Novosbenefícios	Mauá
98765432168	30	20,0	Souza, Jennifer S.	Novosbenefícios	Mauá
98765432168	20	15,0	Souza, Jennifer S.	Reorganização	São Paulo
88866555576	20	Null	Brito, Jorge E.	Reorganização	São Paulo

Junção natural entre FUNC_LOCAL e FUNC_PROJ1

Cpf	Projnumero	Horas	Projnome	Projlocal	Fnome
12345678966	1	32,5	ProdutoX	Santo André	Silva, João B.
*12345678966	1	32,5	ProdutoX	Santo André	Leite, Joice A.
12345678966	2	7,5	ProdutoY	Itu	Silva, João B.
*12345678966	2	7,5	ProdutoY	Itu	Leite, Joice A.
*12345678966	2	7,5	ProdutoY	Itu	Wong, Fernando T.
66688444476	3	40,0	ProdutoZ	São Paulo	Lima, Ronaldo K.
*66688444476	3	40,0	ProdutoZ	São Paulo	Wong, Fernando T.
*45345345376	1	20,0	ProdutoX	Santo André	Silva, João B.
45345345376	1	20,0	ProdutoX	Santo André	Leite, Joice A.
*45345345376	2	20,0	ProdutoY	Itu	Silva, João B.
45345345376	2	20,0	ProdutoY	Itu	Leite, Joice A.
*45345345376	2	20,0	ProdutoY	Itu	Wong, Fernando T.
*33344555587	2	10,0	ProdutoY	Itu	Silva, João B.
*33344555587	2	10,0	ProdutoY	Itu	Leite, Joice A.
33344555587	2	10,0	ProdutoY	Itu	Wong, Fernando T.
*33344555587	3	10,0	ProdutoZ	São Paulo	Lima, Ronaldo K.
33344555587	3	10,0	ProdutoZ	São Paulo	Wong, Fernando T.
33344555587	10	10,0	Computadorização	Mauá	Wong, Fernando T.
*33344555587	20	10,0	Reorganização	São Paulo	Lima, Ronaldo K.
33344555587	20	10,0	Reorganização	São Paulo	Wong, Fernando T.

(* são tuplas falsas)

Diretrizes de projetos informacionais para esquemas de relação

4. Anular a possibilidade de gerar tuplas falsas (ou espúrias)

Regra 4:

Projete esquemas de relação de modo que possam ser unidos com condições de igualdade sobre os atributos que são pares relacionados corretamente (chave primária, chave estrangeira) de um modo que garanta que nenhuma tupla falsa será gerada.
Evite relações com atributos correspondentes que não sejam combinações (chave estrangeira, chave primária), pois a junção sobre tais atributos pode produzir tuplas falsas.

Conclusões

Os problemas discutidos até então podem ser detectados sem ferramentas de análise adicionais:

- Anomalias que causam **trabalho redundante** durante a **inserção** e **modificação** em uma relação, e que podem causar **perda acidental de informação durante a exclusão de uma relação**.
- **Desperdício de espaço de armazenamento devido a NULLs** e a dificuldade de realizar seleções, operações de agregação e junções por causa de valores NULL.
- Geração de dados inválidos e falsos durante as **junções em relações** da base com atributos correspondentes que possam **não representar um relacionamento apropriado** (chave estrangeira, chave primária).

A seguir serão apresentados conceitos formais e a teoria que pode ser usada para definir os pontos positivos e negativos dos esquemas de relação individuais com mais precisão.

- Dependência funcional como uma ferramenta para análise
- Especificação de formas normais como estratégia para alcançar um bom projeto é decompor de maneira correta uma relação mal projetada

Apresentação baseada em:

- Ramez Elmasri e Shamkant B. Navathe 6° Ed (2010)
- Material da Profa. Dra. Sarajane Marques Peres (UNIVESP)

Banco de Dados

Dependências funcionais

FACOM – UFMS

Vanessa Borges

vanessa.a.borges@ufms.br

Capítulo 15: Fundamentos de dependências funcionais e normalização para bancos de dados relacionais

Dependências funcionais

- Até agora, a construção foi feita através do **mapeamento ER para Relacional**
  - Considera bom senso do projetista
  - Não tem formalismo adequado

- **Necessário: teoria para determinação de bons esquemas de relação**
  - Medir formalmente por que um esquema de relação é melhor que outro

Dependências funcionais

- Uma dependência funcional (DF) é uma **restrição imposta a dois conjuntos de atributos** de um banco de dados.
  - Trata-se de **uma restrição que é estabelecida pelo projetista do banco de dados**, de acordo com as regras que valem no mundo que real que está sendo modelado
  - É uma ferramenta que permite estabelecer uma **formalização para a avaliação da qualidade** do projeto de banco de dados relacional

Dependências funcionais

- Uma dependência funcional indicada por **X → Y**, entre dois conjuntos de atributos X e Y que são subconjuntos de R, especifica uma **restrição** sobre possíveis tuplas que podem formar um estado da **relação r de R**.
  - **Restrição**:
    - **Para quaisquer duas tuplas t1 e t2 em r, que tenham t1[X] = t2[X], elas também devem ter t1[Y] = t2[Y]**

  - A restrição **X → Y** pode ser lida de diferentes formas:
    - X leva a Y
    - X determina funcionalmente Y (ou X determina Y)
    - Y é determinado funcionalmente por X (ou Y é determinado por X)
    - Y é funcionalmente dependente de X

Dependências funcionais

- Suponha que:
  - Nosso esquema de banco de dados relacional possua os atributos A,B, C e D (relação R)

R

NOME	SOBRENOME	COD	CLASSE
Ana	Silva	555	A
Ana	Costa	111	B
Carlos	Costa	111	C
Bruno	Lima	111	D

Instância do esquema da relação R

SOBRENOME → COD

Para as tuplas nas quais os valores do atributo B são iguais, os valores no atributo C também são iguais

Dependências funcionais

- Suponha que:
  - Nosso esquema de banco de dados relacional possua os atributos A,B, C e D (relação R)

R

NOME	SOBRENOME	COD	CLASSE
Ana	Silva	555	A
Ana	Costa	111	B
Carlos	Costa	111	C
Bruno	Lima	111	D

Instância do esquema da relação R

SOBRENOME → COD

Para os pares de tuplas nas quais os valores do atributo B são iguais, os valores no atributo C também são iguais

Sim! A instância da relação é válida.

Dependências funcionais

- **X determina funcionalmente Y** em um esquema de uma relação R se, e somente se, sempre que duas tuplas de r(R) **concordarem nos seus valores-X**, elas necessariamente **concordam nos seus valores-Y**.
  - Notem que:
    - Afirmar que **X → Y** vale em R **não significa** afirmar que **Y → X** vale em R.

- **Exemplo:**

R

NOME	SOBRENOME	COD	CLASSE
Ana	Silva	555	A
Ana	Costa	111	B
Carlos	Costa	111	C
Bruno	Lima	111	D

SOBRENOME → COD

R

NOME	SOBRENOME	COD	CLASSE
Ana	Silva	555	A
Ana	Costa	111	B
Carlos	Costa	111	C
Bruno	Lima	111	D

COD → SOBRENOME

Dependências funcionais

- **Notem que:**
  - Se X é uma chave **candidata de R**, então **X → Y** para qualquer subconjunto de atributos Y de R, ou seja, **X → R** (X determina todos os atributos de R).
    - **Duas tuplas em qualquer estado válido r(R) não terão o mesmo valor de X**

CLIENTE

id	cpf	pnome	unome
1	123.321.123-11	Claudia	Fernandes
2	123.456.789-00	Moacir	Junqueira
3	221.334.567-88	Jorge	Vilas Boas
4	987.654.321-22	Claudia	Junqueira

Chave candidata

Chave primária

Dependências funcionais:

id → cpf
id → pnome
id → unome
...
id → {cpf, pnome, unome}

cpf → id
cpf → pnome
cpf → unome
...
cpf → {id, pnome, unome}

Dependências funcionais

- Considere os seguintes esquemas (de baixa qualidade).
  - **Quais dependências funcionais gostaríamos de estabelecer?**

FUNC_DEP

Fnome	Cpf	Datanasc	Endereco	Dnumero	Dnome	Cpf_gerente

Cpf → {Fnome, Datanasc, Endereco, Dnumero}
Dnumero → {Dnome, CPF_gerente}

Dependências funcionais

- Considere os seguintes esquemas (de baixa qualidade).
  - **Quais dependências funcionais gostaríamos de estabelecer?**

FUNC_PROJ

Cpf	Projnumero	Horas	Fnome	Projnome	Projlocal

{Cpf, Projnumero} → Horas
Cpf → Fnome
Projnumero → {Projnome, Projlocal}

Dependências funcionais

- Uma dependência funcional **é uma propriedade** do **esquema de relação R** e **não de um estado da relação R particular**
  - Portanto, uma DF **não pode ser deduzida automaticamente** por determinada extensão de relação r, mas deve ser **definida de maneira explícita por alguém que conhece a semântica dos atributos de R**.

ENSINA

Professor	Disciplina	Texto
Silva	Estruturas de Dados	Bartram
Silva	Gerenciamento de Dados	Martin
Neto	Compiladores	Hoffman
Braga	Estruturas de Dados	Horowitz

Professor → Disciplina?

Texto → Disciplina?

Dependências funcionais

- Uma dependência funcional **é uma propriedade** do **esquema de relação R** e **não de um estado da relação R particular**

ENSINA

Professor	Disciplina	Texto
Silva	Estruturas de Dados	Bartram
Silva	Gerenciamento de Dados	Martin
Neto	Compiladores	Hoffman
Braga	Estruturas de Dados	Horowitz

- A partir da análise da relação ENSINA, e assumindo que ela está em um estado válido, podemos dizer que:

A dependência funcional **Professor → Disciplina** **não foi estabelecida** para o esquema ENSINA.

A dependência funcional **Texto → Disciplina** é possível no esquema, mas **não há como confirmar que ela foi estabelecida** para o esquema.

Dependências funcionais

- Não pode haver instância de relações no banco de dados que violem as dependências funcionais estabelecidas pelo projetista de banco de dados

- Devemos garantir que as dependências funcionais **não estão sendo violadas**
  - Primeiro (preferível): estabelecendo um bom projeto de banco de dados
  - Segundo: implementando funções

Apresentação baseada em:

- Ramez Elmasri e Shamkant B. Navathe 6° Ed (2010)
- Material da Profa. Dra. Sarajane Marques Peres (UNIVESP)

Banco de Dados

Normalização

FACOM – UFMS

Vanessa Borges

vanessa.a.borges@ufms.br

Capítulo 15: Fundamentos de dependências funcionais e normalização para bancos de dados relacionais

Projeto relacional

- A maioria dos projetos relacionais práticos assumem uma das duas técnicas a seguir:
  - Realiza um projeto de **esquema conceitual** usando um modelo conceitual como ER ou EER e **mapeia** o projeto conceitual para um **conjunto de relações**
  - **Projeta as relações com base no conhecimento externo** derivado de uma implementação existente de arquivos, formulários ou relatórios

- Independente da técnica utilizada, é útil avaliar a virtude de relações e decompô-las ainda mais, conforme a necessidade, para obter formas normais mais altas, usando a **teoria de normalização**

Normalização

- **Normalização:**
  - É um processo formal, passo a passo, de análise dos atributos de uma relação com o objetivo de **evitar redundância de informação, eliminando as chamadas anomalias de atualização.**
  - Teoria proposta por Codd no início dos anos 70.
  - Baseia-se no conceito de **FORMAS NORMAIS**. Uma relação (tabela) é dita estar em uma determinada forma normal, se ela satisfizer um conjunto específico de restrições.
  - Faz parte do projeto lógico de banco de dados

- **Forma Normal:**
  - Utilizamos **chaves** e **dependências funcionais** (DFs) de uma relação para certificar que um determinado esquema de relação está em uma particular forma normal.
  - 1FN, 2FN, 3FN, FNBC

Normalização

No processo de normalização, esquemas de relação que **não satisfazem certas condições**, são **decompostos** em esquemas de relação menores que satisfaçam tais condições e ainda possuem algumas propriedades desejáveis.

Formas normais

- A forma normal de uma **relação** é a mais alta forma normal cujas condições são atendidas pela relação
- A forma normal de um **projeto de banco de dados** é aquela referente à sua relação em **mais baixa forma normal**

- O processo de normalização estudado será realizado por meio da decomposição das relações (em relações menores)
  - Essa decomposição deve:
    - **Garantir** a propriedade de junção sem perdas (Propriedade necessária)
    - **Procurar preservar** as dependências funcionais (Propriedade desejável)

Primeira forma normal

- Uma **relação R** está na **1FN** se, e somente se **R contém somente atributos atômicos (simples, indivisíveis)**
  - **Não** podem ser atributos **multivalorados**
  - **Não** podem ser atributos **compostos (tabelas aninhadas)**

DEPARTAMENTO

Dnome	Dnumero	Cpf_gerente	Dlocal
Pesquisa	5	33344555587	Santo André, Itu, São Paulo
Administração	4	98765432168	Mauá
Matriz	1	88866555576	São Paulo

Primeira forma normal

- Uma **relação R** está na **1FN** se, e somente se **R contém somente atributos atômicos (simples, indivisíveis)**
  - **Não** podem ser atributos **multivalorados**
  - **Não** podem ser atributos **compostos (tabelas aninhadas)**

DEPARTAMENTO

Dnome	Dnumero	Cpf_gerente	Dlocal
Pesquisa	5	33344555587	Santo André, Itu, São Paulo
Administração	4	98765432168	Mauá
Matriz	1	88866555576	São Paulo

Existem 3 alternativas para colocar essa relação na 1FN

O esquema de relação DEPARTAMENTO não está na 1FN!

Primeira forma normal

- **Como colocar na 1FN? (multivalorado)**
  - **Alternativa 1:**
    - Remover o atributo Dlocal que viola a 1FN e colocá-lo em uma relação separada de DEPARTAMENTO denominada **LOCALIZACAO_DEP**

DEPARTAMENTO

Dnome	Dnumero	Cpf_gerente	Dlocal
Pesquisa	5	33344555587	Santo André, Itu, São Paulo
Administração	4	98765432168	Mauá
Matriz	1	88866555576	São Paulo

Normalização

DEPARTAMENTO

Dnome	Dnumero	Cpf_gerente
Pesquisa	5	33344555587
Administração	4	98765432168
Matriz	1	88866555576

LOCALIZACAO_DEP

Dnumero	Dlocal
1	São Paulo
4	Mauá
5	Santo André
5	Itu
5	São Paulo

Os dois novos esquemas da relação estão na 1FN!

Primeira forma normal

- **Como colocar na 1FN? (multivalorado)**
  - **Alternativa 2:** expandir a chave de modo que haverá uma tupla separada na relação original DEPARTAMENTO para cada local de um departamento.

DEPARTAMENTO

Dnome	Dnumero	Cpf_gerente	Dlocal
Pesquisa	5	33344555587	Santo André, Itu, São Paulo
Administração	4	98765432168	Mauá
Matriz	1	88866555576	São Paulo

Normalização

DEPARTAMENTO

Dnome	Dnumero	Cpf_gerente	Dlocal
Pesquisa	5	33344555587	Santo André
Pesquisa	5	33344555587	Itu
Pesquisa	5	33344555587	São Paulo
Administração	4	98765432168	Mauá
Matriz	1	88866555576	São Paulo

O novo esquemas da relação estão na 1FN mas introduz redundância na relação.

Primeira forma normal

- **Como colocar na 1FN? (multivalorado)**
  - **Alternativa 3:** se o número máximo de valores for conhecido para o atributo, pode-se criar novos atributos para a relação.
    - Exemplo: se for conhecido que **no máximo três locais poderão existir** para um departamento, substituir o atributo Dlocal pelos três atributos atômicos: **Dlocal1**, **Dlocal2** e **Dlocal3**.

DEPARTAMENTO

Dnome	Dnumero	Cpf_gerente	Dlocal
Pesquisa	5	33344555587	Santo André, Itu, São Paulo
Administração	4	98765432168	Mauá
Matriz	1	88866555576	São Paulo

Normalização

Dnome	Dnumero	Cpf_gerente	Dlocal1	Dlocal2	Dlocal3
Pesquisa	5	33344555587	Santo André	Itu	São Paulo
Administração	4	98765432168	Mauá	NULL	NULL
Matriz	1	88866555576	São Paulo	NULL	NULL

O novo esquemas da relação estão na 1FN mas introduz maior quantidade de valor NULL.

A consulta sobre esse atributo torna-se mais difícil.

- Listar os departamentos que têm 'Santo André' como um de seus locais nesse projeto.

Primeira forma normal

- **Como colocar na 1FN? (multivalorado e compostos)**
  - Quando a relação possui relações aninhadas, por exemplo, atributos multivalorados que também sejam compostos

FUNC_PROJ

Cpf	Fnome	Projnumero	Horas
12345678966	Silva, João B.	1	32,5
		2	7,5
66688444476	Lima, Ronaldo K.	3	40,0
45345345376	Leite, Joice A.	1	20,0
		2	20,0
33344555587	Wong, Fernando T.	2	10,0
		3	10,0
		10	10,0
		20	10,0
99988777767	Zelaya, Alice J.	30	30,0
		10	10,0
98798798733	Pereira, André V.	10	35,0
		30	5,0
98765432168	Souza, Jennifer S.	30	20,0
		20	15,0
88866555576	Brito, Jorge E.	20	NULL

FUNC_PROJ(Cpf, Fnome, {PROJS(Projnumero, Horas)})

multivalorado

Normalização

FUNC_PROJ1

Cpf	Fnome

FUNC_PROJ2

Cpf	Projnumero	Horas

Primeira forma normal

- **Como colocar na 1FN? (multivalorado e compostos)**

FUNC_PROJ

Cpf	Fnome	Projnumero	Horas
12345678966	Silva, João B.	1	32,5
		2	7,5
66688444476	Lima, Ronaldo K.	3	40,0
45345345376	Leite, Joice A.	1	20,0
		2	20,0
33344555587	Wong, Fernando T.	2	10,0
		3	10,0
		10	10,0
		20	10,0
99988777767	Zelaya, Alice J.	30	30,0
		10	10,0
98798798733	Pereira, André V.	10	35,0
		30	5,0
98765432168	Souza, Jennifer S.	30	20,0
		20	15,0
88866555576	Brito, Jorge E.	20	NULL

Normalização

Cpf	Fnome
12345678966	Silva, João B.
66688444476	Lima, Ronaldo K.
45345345376	Leite, Joice A.
33344555587	Wong, Fernando T.
99988777767	Zelaya, Alice J.
98798798733	Pereira, André V.
98765432168	Souza, Jennifer S.
88866555576	Brito, Jorge E.

Cpf	Projnumero	Horas
12345678966	1	32,5
12345678966	2	7,5
66688444476	3	40,0
45345345376	1	20,0
45345345376	2	20,0
33344555587	2	10,0
33344555587	3	10,0
33344555587	10	10,0
33344555587	20	10,0
99988777767	30	30,0
99988777767	10	10,0
98798798733	10	35,0
98798798733	20	5,0
98765432168	30	20,0
98765432168	20	15,0
88866555576	20	NULL

Primeira forma normal

- Método para corrigir o problema: a passagem à primeira forma normal por decomposição de tabelas é feita nos seguintes passos:

1. É criada uma tabela na 1FN referente à tabela que contém apenas as colunas com valores atômicos, isto é, sem as tabelas aninhadas. A chave primária da tabela na 1FN é idêntica à chave da tabela com valores atômicos.

2. Para cada tabela aninhada, é criada uma tabela na 1FN composta pelas seguintes colunas:
   a) chave primária de cada uma das tabelas nas quais a tabela em questão está aninhada,
   b) as colunas da própria tabela aninhada.

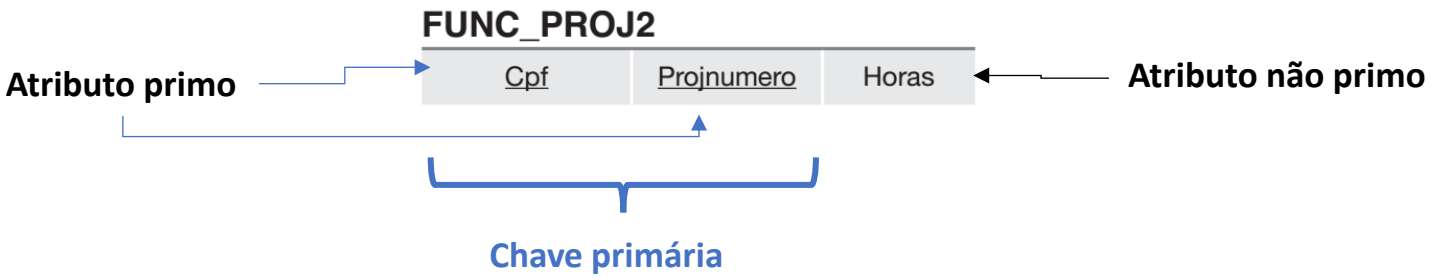
3. São definidas as chaves primárias das tabelas na 1FN que correspondem a tabelas aninhadas.

Heuser, Carlos Alberto. Projeto de banco de dados - V4 - UFRGS (Página 191)

Segunda forma normal – Atributo primo

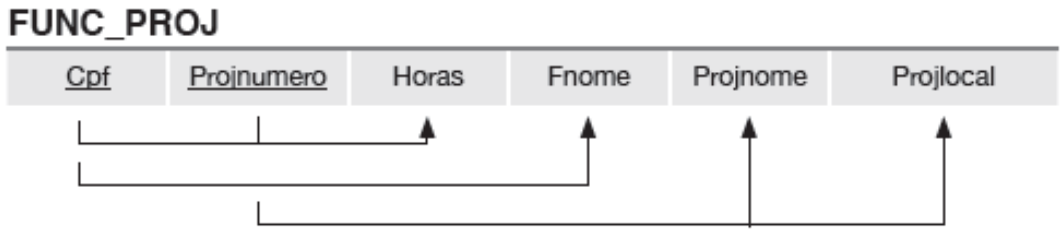
- É baseada no conceito de **atributo primo** e **dependência funcional total**

- **Atributo primo (ou primário)**
  - Um atributo de um esquema de relação R é chamado **ATRIBUTO PRIMO de R** se ele **é membro de alguma chave candidata de R**. Caso contrário ele é chamado ATRIBUTO NÃO PRIMO.



Segunda forma normal – DF Total

- É baseada no conceito de **atributo primo** e **dependência funcional total**
- **Dependência funcional total**
  - A dependência funcional **X → Y** é uma **dependência funcional total** se ao remover qualquer atributo A de X a dependência funcional **deixa de valer**
    - Ou seja, para qualquer **A ϵ X, (X - A) não determina funcionalmente Y**



{Cpf, Projnumero} → Horas
É uma dependência funcional total

Segunda forma normal – DF- Parcial

- É baseada no conceito de **atributo primo** e **dependência funcional total**
- **Dependência funcional parcial**
  - Uma dependência funcional **X → Y** é uma **dependência parcial** se para algum atributo **A ϵ X, (X - A) →Y**

FUNC_PROJ

Cpf	Projnumero	Horas	Fnome	Projnome	Projlocal

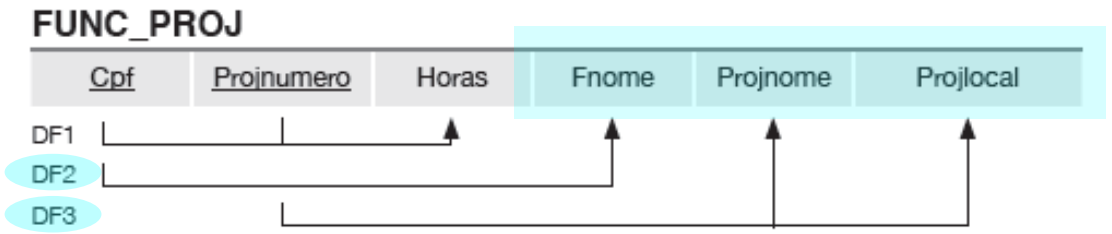
{Cpf, Projnumero} → Fnome

É uma dependência funcional parcial pois Cpf → Fnome

Uma tabela que está na 1FN e que possui apenas uma coluna como chave primária não contém dependências parciais.

Segunda forma normal

- Uma **relação R** está na **2FN** se, e somente se:
  - Está na 1FN
  - **Todo atributo não primo** A em R **tem dependência funcional total da chave primária** (ou de qualquer chave) de R.
    - Em outras palavras:
      - Um esquema de relação R está na 2FN **se todo atributo não primo** A em R **não depende parcialmente de qualquer chave** em R

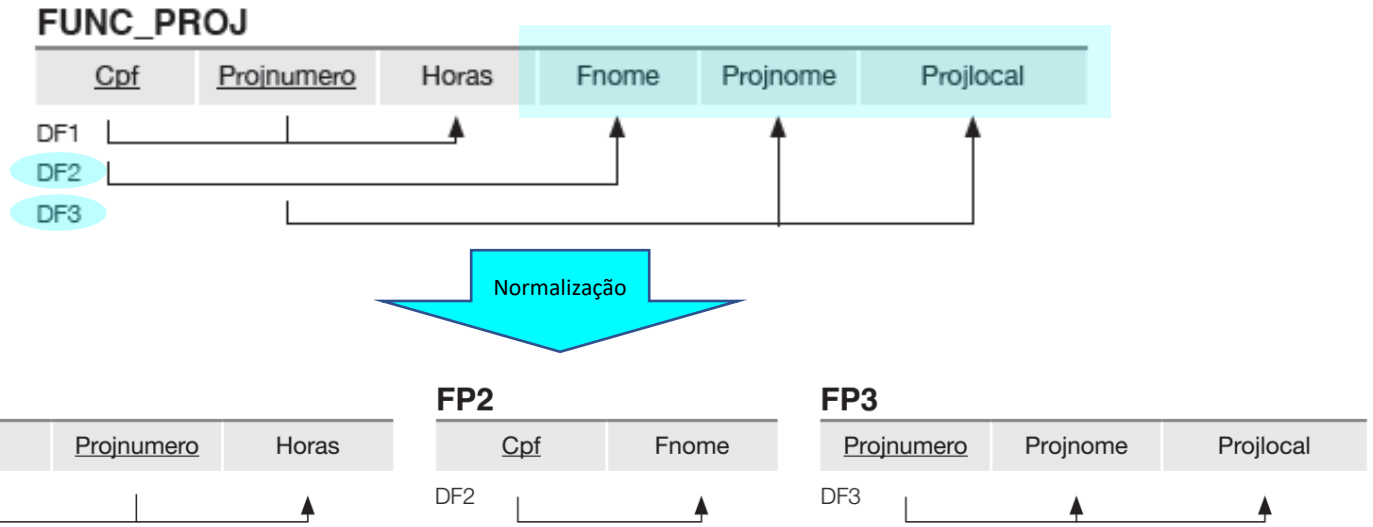


Está na 1FN

Não está na 2FN

Segunda forma normal

- Método para corrigir o problema:
  - Para cada **subconjunto do conjunto de atributos que constitui a chave primária,** começar uma relação com esse sub conjunto como sua chave primária
  - Incluir os atributos da relação original na relação correspondente à chave primária apropriada, isto é**, colocar cada atributo junto com a coleção mínima da qual ele depende**, atribuindo um nome a cada relação



Está na 1FN
Está na 2FN

Terceira forma normal – DF Transitiva

- É baseada no conceito de **dependência funcional transitiva**

- Uma dependência funcional **X → Y** em um esquema de relação R **é uma dependência transitiva se**:
  - Existe um conjunto de atributos Z em R onde **Z não é chave candidata em nenhum subconjunto de qualquer chave de R**, e tanto **X → Z** quanto **Z → Y** valem no esquema

Exemplo:

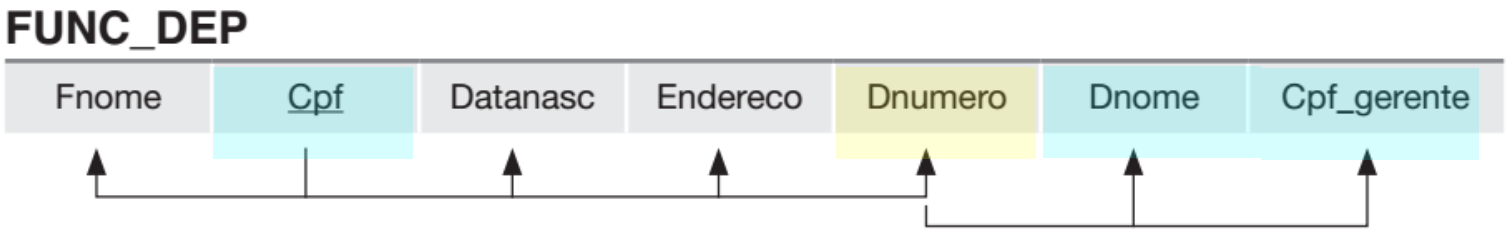
- **Cpf → Cpf_gerente** **(É uma transitiva)**
  - Cpf → Dnumero e Dnumero → Cpf_gerente se mantém
  - Dnumero não é chave candidata
  - Dnumero não é subconjunto de uma chave

Terceira forma normal

- Uma **relação R** está na **3FN** se, e somente se:
  - Está na 2FN
  - Nenhum atributo **não primo de R** é **transitivamente dependente da chave primária** (ou de qualquer chave) de R

Terceira forma normal

- Em outras palavras:
  - Um esquema de relação R **viola a definição geral da 3FN** se uma dependência funcional **X → A**, que se mantém em R, não atender a qualquer condição:
    - Um atributo não primo determina outro atributo não primo. Aqui, em geral, temos uma dependência transitiva que viola a 3FN.
    - Um subconjunto apropriado de uma chave de R determina funcionalmente um atributo não primo. Aqui, temos uma dependência parcial que viola a 3FN (e também a 2FN).



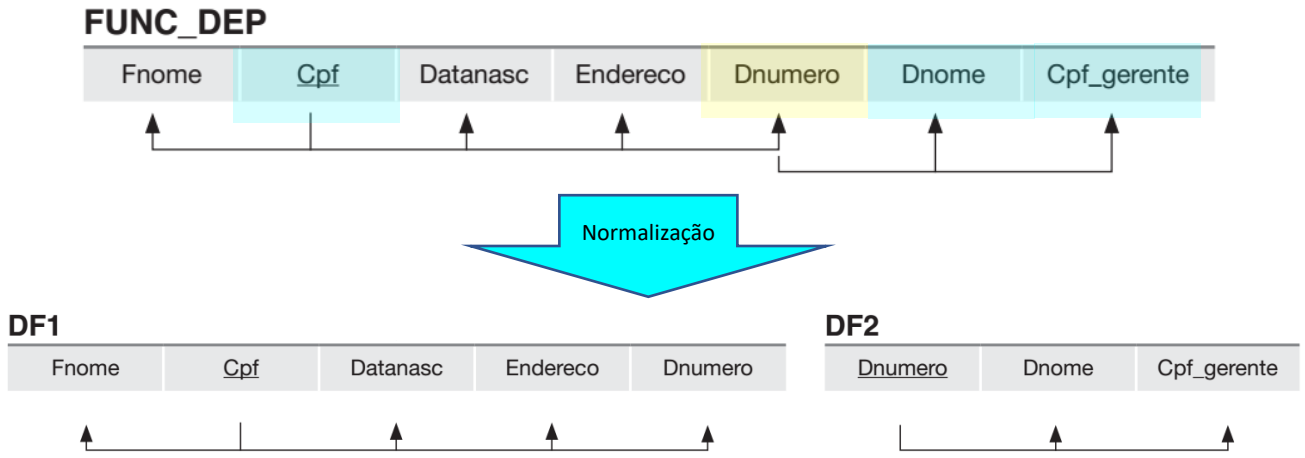
Está na 1FN
Está na 2FN

Não está na 3FN

- FUNC_DEP está na 2FN, pois não existe dependência parcial sobre uma chave
- Porém, FUNC_DEP não está na 3FN devido à dependência transitiva de Cpf_gerente (e também Dnome) em Cpf por meio de Dnumero.

Terceira forma normal

- Método para corrigir o problema:
  - Para cada **determinante que não é uma chave candidata**, remover da relação os atributos que dependem desse determinante.
  - Criar uma nova relação contendo todos os atributos da relação original que dependem desse determinante.
  - Tornar o determinante a chave primária da nova relação.



Está na 1FN
Está na 2FN
Está na 3FN

Exemplo:

Chave candidata

LOTES

Propriedade_num	Nome_cidade	Num_lote	Area	Preco	Imposto

DF1

DF2

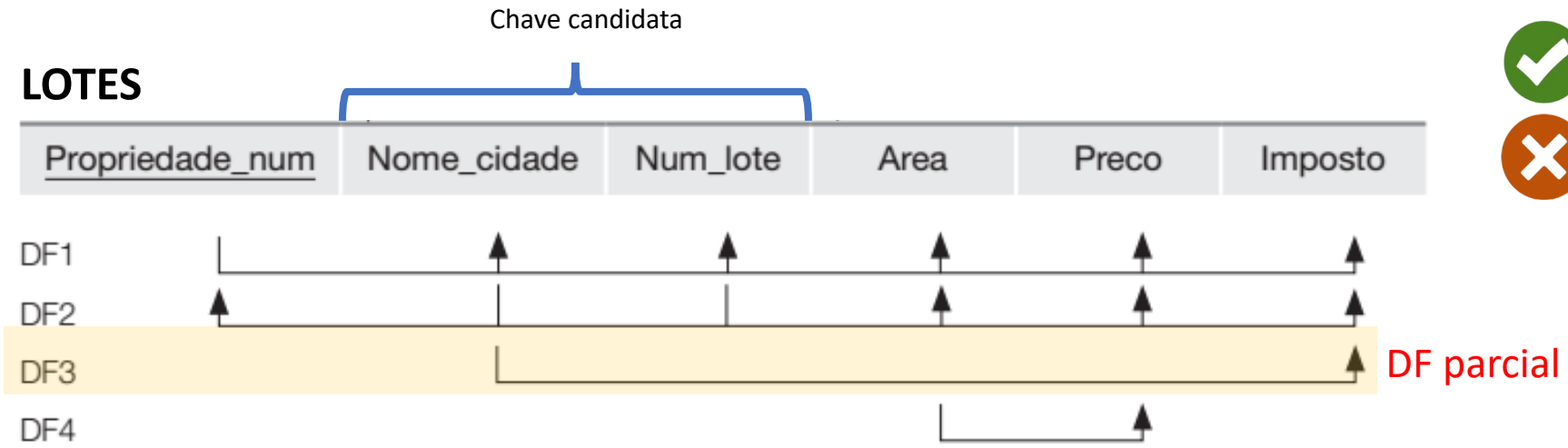
DF3

DF4

Está na 1FN

O teste para 2FN envolve avaliar as dependências funcionais cujos atributos do lado esquerdo façam parte da chave primária.

Exemplo:

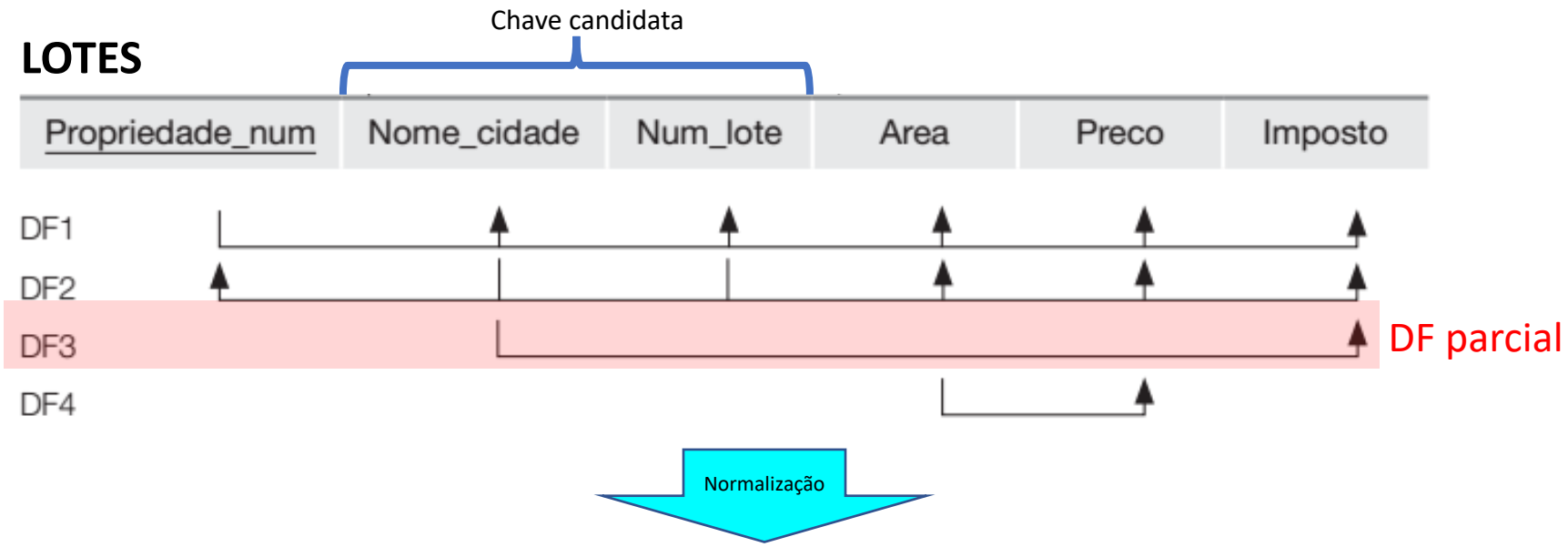


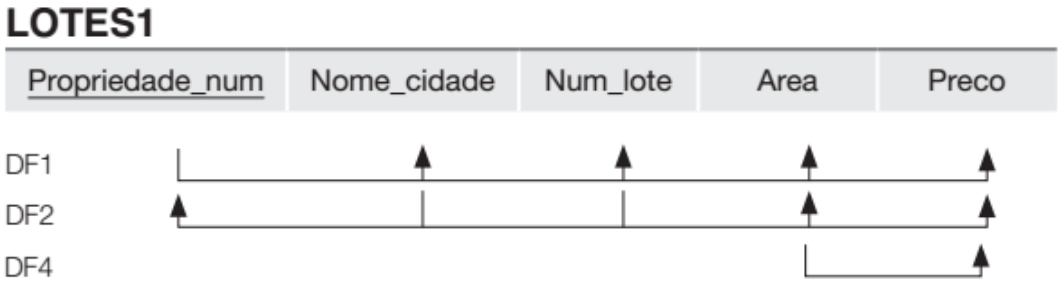
Está na 1FN

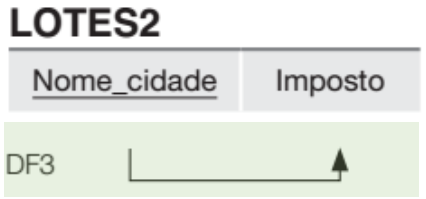
Não está na 2FN

- Com base nas duas chaves candidatas Propriedade_num e {Nome_cidade, Num_lote}, as dependências funcionais **DF1 e DF2 se mantêm**.
- O esquema de relação LOTES viola a definição geral da **2FN porque Imposto é parcialmente dependente da chave candidata** {Nome_cidade, Num_lote}

Exemplo:







Está na 1FN

Está na 2FN

Exemplo:

Chave candidata

LOTES1

Propriedade_num	Nome_cidade	Num_lote	Area	Preco

DF1
DF2
DF4

LOTES2

Nome_cidade	Imposto

DF3

Está na 1FN

Está na 2FN

Não está na 3FN

Atributo não primo determina outro atributo não primo
Não está na 3FN!

Normalização

LOTES1A

Propriedade_num	Nome_cidade	Num_lote	Area

DF1
DF2

LOTES1B

Area	Preco

DF4

LOTES2

Nome_cidade	Imposto

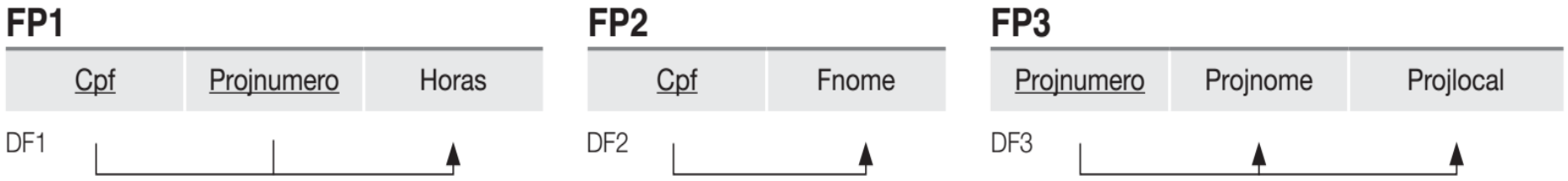
DF3

Formas normais baseadas em chave

Forma normal	Teste	Solução (normalização)
Primeira (1FN)	Relação não deve ter atributos multivalorados ou relações aninhadas.	Formar novas relações para cada atributo multivalorado ou relação aninhada.
Segunda (2FN)	Para relações em que a chave primária contém múltiplos atributos, nenhum atributo não chave deverá ser funcionalmente dependente de uma parte da chave primária.	Decompor e montar uma nova relação para cada chave parcial com seu(s) atributo(s) dependente(s). Certificar-se de manter uma relação com a chave primária original e quaisquer atributos que sejam total e funcionalmente dependentes dela.
Terceira (3FN)	A relação não deve ter um atributo não chave determinado funcionalmente por outro atributo não chave (ou por um conjunto de atributos não chave). Ou seja, não deve haver dependência transitiva de um atributo não chave sobre a chave primária.	Decompor e montar uma relação que inclua o(s) atributo(s) não chave que determina(m) funcionalmente outro(s) atributo(s) não chave.

Forma normal de boyce-codd (FNBC)

- Um esquema de relação R está na FNBC se sempre que uma dependência funcional não trivial **X → A** valem em R, **X é uma superchave de R**
  - **Superchave:**
    - Conjunto de um ou mais atributos que, tomados coletivamente, nos permite identificar unicamente uma tupla da relação
- A FNBC permite que A seja um atributo primo (diferente da 3FN)
    - Isso torna a FNBC mais forte quando comparada a 3NF



Está na FNBC

Forma normal de boyce-codd (FNBC)

- Um esquema de relação R está na FNBC se sempre que uma dependência funcional não trivial **X → A** valem em R, **X é uma superchave de R**

  - **Exemplo 2:**

DF1

Fnome	Cpf	Datanasc	Endereco	Dnumero

DF2

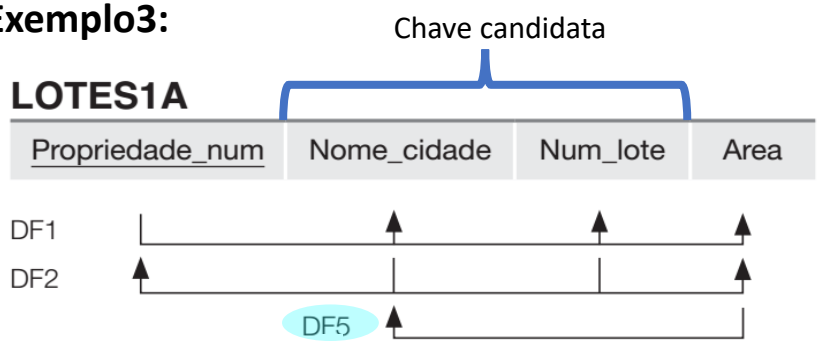
Dnumero	Dnome	Cpf_gerente

Está na FNBC

É comum ao passar para a 2FN ou 3FN as relações atingirem a BCNF, porém nem sempre isso será alcançado.

Forma normal de boyce-codd (FNBC)

- Um esquema de relação R está na FNBC se sempre que uma dependência funcional não trivial **X → A** valem em R, **X é uma superchave de R**
  - **Exemplo3:**



Propriedade_num	Nome_cidade	Num_lote	Area
111	São Paulo	4	100
222	São Paulo	5	100
333	São Paulo	2	50
444	Rio de Janeiro	5	200

Compreendendo o esquema:

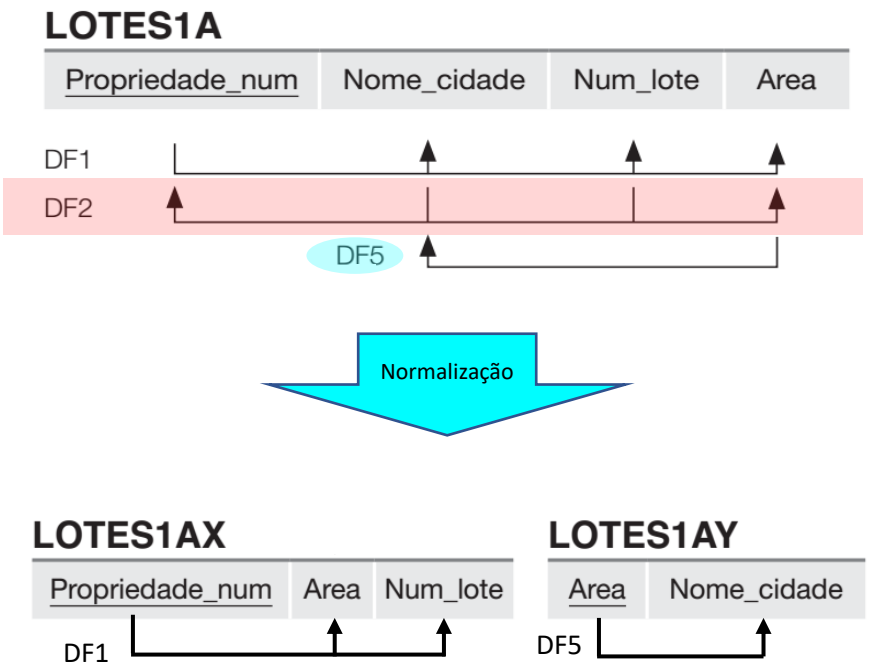
- Pela DF1 sabemos que Propriedade_num é uma chave candidata (escolhida para ser chave primária da relação
- Pela DF2 sabemos que {Nome_cidade, Num lote} é uma chave candidata
- Pela DF5 sabemos que todos os terrenos de mesma área estão localizados na mesma cidade

Está na 3FN (Em DF5 nome_cidade é um atributo primo)

Não está na FNBC pois Area **não é uma superchave (DF5)**

Forma normal de boyce-codd (FNBC)

- **Exemplo3:**



Está na 3FN
As relações estão na FNBC, porém a DF2 foi perdida

Conclusões - Normalização

- Propriedade de decomposição relacional
  - **É preciso assegurar** que cada atributo de R apareça em pelo menos uma relação dentro da decomposição
  - **É uma meta** ter cada uma das relações da decomposição em BCNF ou 3NF
  - **É desejável preservar** as dependências funcionais na decomposição
    - Não é necessário que as DFs especificadas apareçam em relações individuais, é **suficiente que elas possam ser inferidas**.
    - Se as DFs forem quebradas em mais de uma relação, a verificação delas exigirá a execução de operações de junção.
  - **É preciso assegurar** que a decomposição não causará perdas na junção
    - Há casos em que a meta de alcançar uma FN precisa ser sacrificada, pois alcança-la pode levar a uma decomposição com perdas na junção