

Dependências Funcionais e Normalização



Objetivos do projeto relacional

- O projeto de banco de dados relacional produz um conjunto de relações.
- Os objetivos implícitos da atividade de projeto são **preservação da informação** e **redundância mínima**.
 - O projeto relacional precisa preservar os conceitos capturados originalmente no projeto conceitual após o mapeamento do projeto conceitual para lógico.

Diretrizes informais para determinar a qualidade do projeto relacional

- Garantir que a semântica dos atributos seja clara no esquema.
- Reduzir a informação redundante nas tuplas.
- Reduzir os valores NULL nas tuplas.
- Reprovar a possibilidade de gerar tuplas falsas.

Analise esse esquema / estado

ENOME	#CI	DNO	DNOME	DGER
João Silva	620.042	D1	Pessoal	Tereza Costa
Maria Alves	328.345	D2	Brinquedos	Marcia Mendes
Ana Pereira	245.980	D2	Brinquedos	Marcia Mendes
Fabio Costa	567.987	D3	Roupas	Iris Pereira
Jane Lima	456.321	D1	Pessoal	Tereza Costa

- DNO \rightarrow DNOME, DGER
- Problemas:
 - Desperdício de espaço em memória
 - Anomalias de atualização
 - Inserção, Modificação e Deleção.

O que fazer?

- Decompor...

ENOME	#CI	DNO
João Silva	620.042	D1
Maria Alves	328.345	D2
Ana Pereira	245.980	D2
Fabio Costa	567.987	D3
Jane Lima	456.321	D1

DNO	DNOME	DGER
D1	Pessoal	Tereza Costa
D2	Brinquedos	Marcia Mendes
D3	Roupas	Iris Pereira

Transformação ER para Relacional

ER Model

<i>ID</i>	<i>Nome</i>	<i>Endereço</i>	<i>Hobby</i>
1111	Joe	123 Main	{biking, hiking}

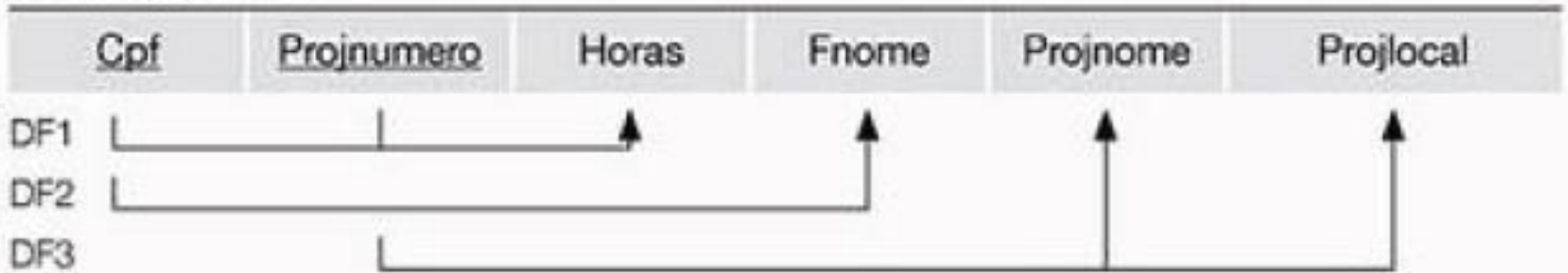
Relational Model

<i>ID</i>	<i>Nome</i>	<i>Endereço</i>	<i>Hobby</i>
1111	Joe	123 Main	biking
1111	Joe	123 Main	hiking
.....			

Redundância

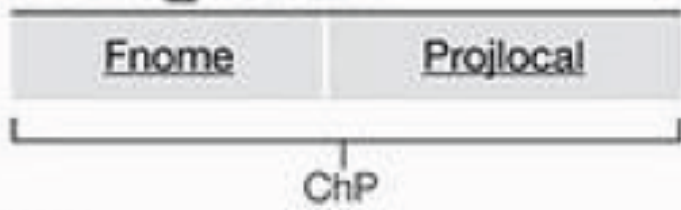
FUNC_PROJ com anomalias de atualização

FUNC_PROJ



Decomposição de FUNC_PROJ

FUNC_LOCAL



FUNC_PROJ1



Estado

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
Zelaya, Alice J.	Mauá
Pereira, André V.	Mauá
Souza, Jennifer S.	Mauá
Souza, Jennifer S.	São Paulo
Brito, Jorge E.	São Paulo

FUNC_PROJ1

Cpf	Projnumero	Horas	Projnome	Projlocalizacao
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
99988777767	30	30,0	Novosbeneficios	Mauá
99988777767	10	10,0	Computadorização	Mauá
98765432168	10	35,0	Computadorização	Mauá
98765432168	30	5,0	Novosbeneficios	Mauá
98765432168	30	20,0	Novosbeneficios	Mauá
98798798733	20	15,0	Reorganização	São Paulo
88866555576	20	NULL	Reorganização	São Paulo

Junção com perdas

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.

Tuplas falsas geradas com asterisco (*).

Problema

- Considere a seguinte relação “Venda”

Num	Cod Produto	Nome	Descrição Produto	Preço
1	5566	Palito de dente	Artefato de madeira utilizado para extrair resíduos alimentares dos espaços entre os dentes. Também utilizados como hastes na improvisação de cotonetes.	R\$1,99
2	9807	Jurubeba	Bebida alcoólica nobre de sabor adocicado altamente apreciada em reuniões sociais.	R\$7,99
3	5566	Palito de dente	Artefato de madeira utilizado para extrair resíduos alimentares dos espaços entre os dentes. Também utilizados como hastes na improvisação de cotonetes.	R\$2,99
4	9807	Jurubeba	Bebida alcoólica nobre de sabor adocicado altamente apreciada em reuniões sociais.	R\$6,99

- Quais as desvantagens da repetição de dados dessa relação?
 - Desperdício de espaço
 - Possibilidade de inconsistência

Solução

- Novas relações para a mesma situação

Venda

Num	Cod Produto	Preço
1	5566	R\$1,99
2	9807	R\$7,99
3	5566	R\$2,99
4	9807	R\$6,99

Produto

Cod Produto	Nome	Descrição Produto
5566	Palito de dente	Artefato de madeira utilizado para extrair resíduos alimentares dos espaços entre os dentes. Também utilizados como hastes na improvisação de cotonetes.
9807	Jurubeba	Bebida alcoólica de nobre sabor adocicado altamente apreciada.

Dependência funcional

- **Dependência funcional** (DF) é uma ferramenta formal para a análise de esquemas relacionais, que permite detectar e descrever problemas em termos precisos.

Dependência funcional

- Uma *dependência funcional* é uma relação de dependência entre atributos de uma relação.
 - **Notação**
 - $A \rightarrow B$ (lê-se “A determina B”)
 - **Exemplo**
 - $\text{Cod Produto} \rightarrow \text{Nome}$
 - $\text{Cod Produto} \rightarrow \text{Descrição Produto}$

Cod Produto	Nome	Descrição Produto
5566	Palito de dente	Artefato de madeira utilizado para extrair resíduos alimentares dos espaços entre os dentes. Também utilizados como hastes na improvisação de cotonetes.
9807	Jurubeba	Bebida alcoólica de nobre sabor adocicado altamente apreciada.

Dependência funcional

- Uma **dependência funcional**, indicada por $X \rightarrow Y$, entre dois conjuntos de atributos X e Y que são subconjuntos de uma relação R , especifica uma **restrição** sobre possíveis tuplas que podem formar um estado de relação r de R .
- A restrição é que, para quaisquer duas tuplas t_1 e t_2 em r que tenham $t_1[X] = t_2[X]$, elas também devem ter $t_1[Y] = t_2[Y]$.
- Isso significa que os valores do componente Y de uma tupla em r dependem dos valores do componente X .
- Uma dependência funcional é **trivial** se $X \supseteq Y$.

Dependência funcional

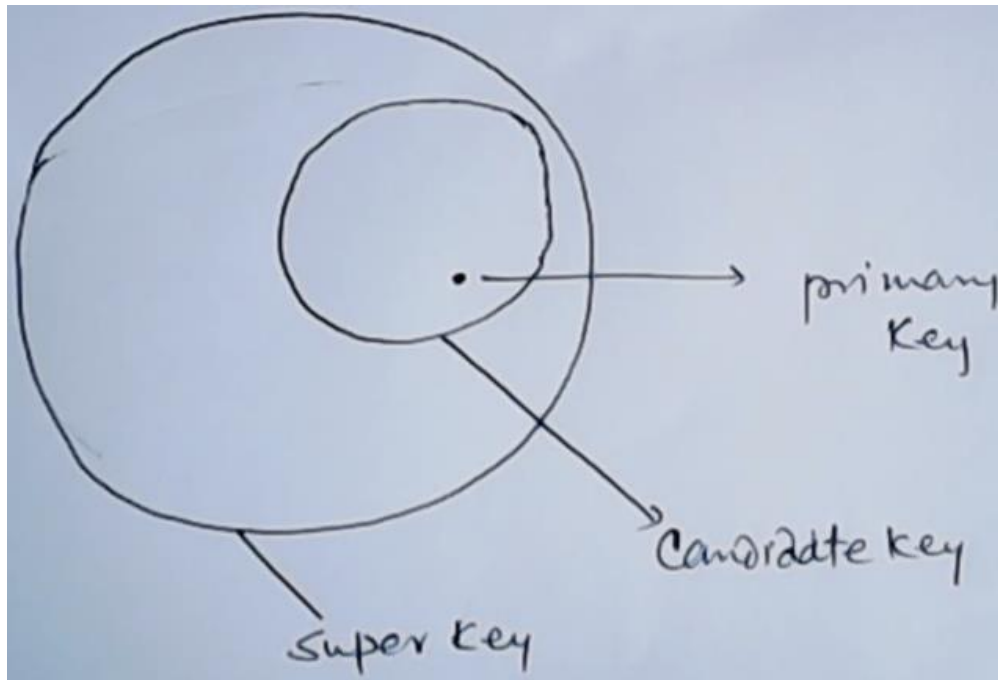
- Em uma tabela relacional, diz-se que uma coluna Y depende funcionalmente de uma coluna X (ou que a coluna X determina a coluna Y) quando, em todas as linhas da tabela, para cada valor de X que aparece na tabela, aparecer o mesmo valor de Y.
- Para denotar a dependência funcional, usa-se o símbolo \rightarrow .
 - Ex: `codigo` \rightarrow `salario`.
 - A expressão denota que a coluna `salario` depende funcionalmente da coluna `codigo`. Diz-se, também, que a coluna `codigo` é o determinante da dependência funcional.

Dependência funcional

- Uma dependência funcional pode ser
 - **Total**
 - $(A,B) \rightarrow C$ e ambos os atributos A e B determinam C.
 - Exemplo: ***(Cod Turma, Matrícula) → Nota Final***
 - **Parcial**
 - $(A,B) \rightarrow C$ e $A \rightarrow C$ ou $B \rightarrow C$
 - Exemplo: ***(Matrícula, CPF) → Nome aluno***
 - **Transitiva**
 - $A \rightarrow C$ quando $A \rightarrow B$ e $B \rightarrow C$
 - Exemplo: ***cod livro → nome categoria***, pois ***cod livro → cod categoria*** e ***cod categoria → nome categoria***

Super chave, **chave candidata** e chave primária

- <https://www.youtube.com/watch?v=CsGalHoQKbs>
- <https://www.youtube.com/watch?v=kzFeh9UJLRo>



Fechamento de atributo

- Fechamento de $A \rightarrow A^+$
- <https://www.youtube.com/watch?v=IUPTC65B9qE>

$R(A, B, C, D)$

$A^+ \rightarrow A B D$

$F(A) = \{ A, B, D \}$

$\left| \begin{array}{l} A \rightarrow B \\ \underline{B} \rightarrow D \\ C \rightarrow B \end{array} \right.$

Determinação de chaves candidatas

- É possível determinar as chaves candidatas de uma relação a partir de suas dependências funcionais.
- https://www.youtube.com/watch?v=YQt_kz3JgXI
- <https://www.youtube.com/watch?v=QykKsqTf-Mo>
- <https://www.youtube.com/watch?v=9fuJUQJd-A8>

Determinação de chaves candidatas

$R(A, B, C, D, E, F)$

$A \rightarrow C$

$C \rightarrow D$

$D \rightarrow B$

$E \rightarrow F$

Atributos que não são determinados por ninguém, devem fazer parte da chave candidata.

$(AE)^+ = AECDBF$

↑

o. key.

Determinação de chaves candidatas

$R(A, B, C, D, E)$

$A \rightarrow B$

$AB \rightarrow C$

$BC \rightarrow E$

$BD \rightarrow E$

$(AD)^+ \rightarrow \underline{A} \underline{D} \underline{B} \underline{C} \underline{E}$

Determinação de chaves candidatas

$R(ABCDEFGH)$

$CH \rightarrow G$

$A \rightarrow BC$

$B \rightarrow CFH$

$E \rightarrow A$

$F \rightarrow EG$

$[A]^{+} \rightarrow \underline{A} \underline{B} \underline{C} \underline{F} \underline{H} \underline{E} \underline{G}$

$[B]^{+} \rightarrow \underline{A} \underline{B} \underline{C} \underline{F} \underline{H} \underline{E} \underline{G} \underline{A}$

$[C]^{+} \rightarrow \underline{A} \underline{C}$

$[E]^{+} \rightarrow \underline{A} \underline{E} \underline{A}$

$[F]^{+} \rightarrow \underline{A} \underline{F} \underline{E}$

$[CH]^{+} \rightarrow \underline{A} \underline{C} \underline{H} \underline{G}$

$[\underline{A}, \underline{B}, \underline{E}, \underline{F}]$

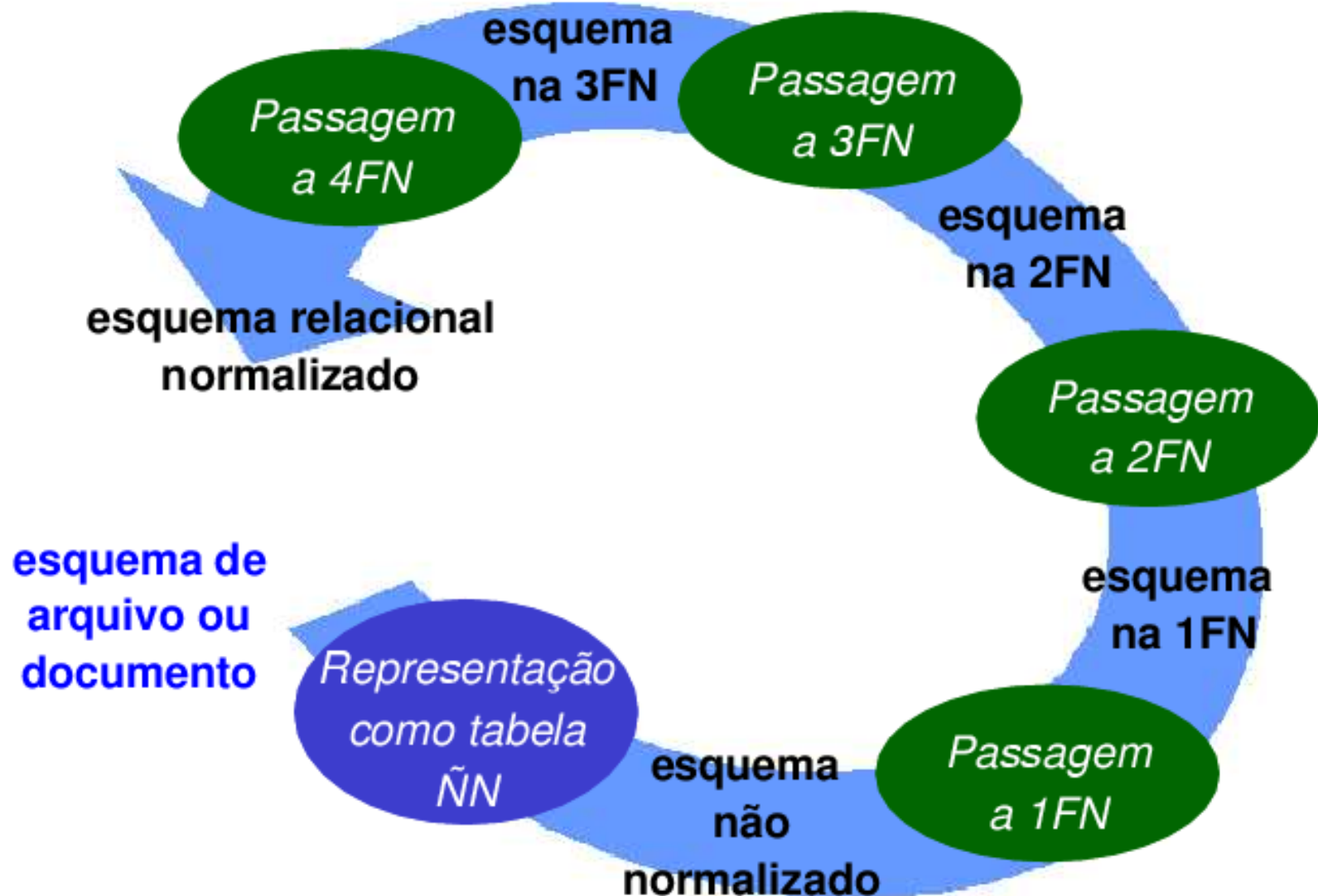
Normalização

- Processo de analisar os esquemas de relação dados com base em suas dependências funcionais e chaves primárias para conseguir as propriedades de **minimização da redundância** e **minimização das anomalias de inserção, exclusão e atualização**.
- Esquemas de relação que não atendem aos testes de forma normal são **decompostos** em esquemas de relação que atendem aos testes.

Por que normalizar?

- Normalizar é um processo para a transformação esquemas de relações ruins em esquemas mais desejáveis.
- Dizemos que uma relação está em uma *forma normal* quando ela evita uma “determinada situação indesejada”.
- Há diversas formas normais.
 - Primeira Forma Normal (1FN)
 - Segunda Forma Normal (2FN)
 - Terceira Forma Normal (3FN)
 - Forma Normal de Boyce-Codd (FNBC ou BCNF)
 - Quarta Forma Normal (4FN)
 - Quinta Forma Normal (5FN)

Passos da normalização



Normalização - Propriedades

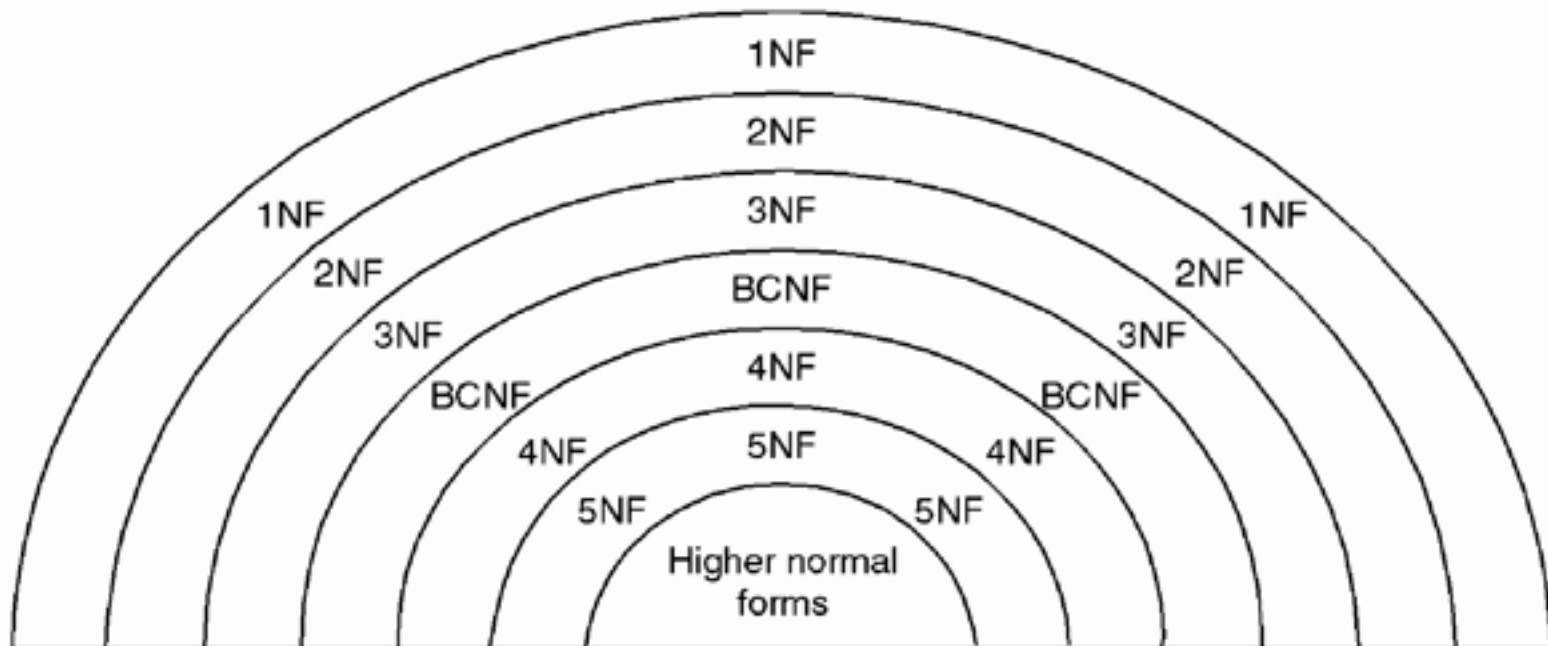
- A normalização deve manter as seguintes propriedades:
 - i. **propriedade de junção não aditiva ou junção sem perdas.**
 - Garante que o problema de geração de tuplas falsas não ocorra após a decomposição.
 - ii. **propriedade de preservação de dependência.**
 - Garante que cada DF seja representada em alguma relação após a decomposição.

Forma Normal (FN)

- Refere-se à condição de forma normal mais alta a que ela atende e, portanto, indica o grau ao qual ela foi normalizada.

Forma Normal (FN)

- Todas as formas normais são aditivas, de modo que se um modelo estiver em 3FN, ele por definição também estará na 2FN e 1FN.



Primeira Forma Normal (1FN)

- A **primeira forma normal (1FN)** afirma que o domínio de um atributo deve incluir apenas valores atômicos (simples, indivisíveis), ou seja, a relação não deve ter atributos multivalorados ou relações aninhadas.
- Uma relação que não esteja na 1FN deve ser decomposta em novas relações para cada atributo multivalorado ou relação aninhada.

Primeira Forma Normal (1FN)

primeira forma normal (1FN)

=

diz-se que uma tabela está na primeira forma normal, quando ela **não contém tabelas aninhadas**

Tabela aninhada

Proj

CódProj	Tipo	Descr	Emp					
			CodEmp	Nome	Cat	Sal	DataIni	TempAI
LSC001	Novo Desenv.	Sistema de Estoque	2146	João	A1	4	1/11/91	24
			3145	Sílvio	A2	4	2/10/91	24
			6126	José	B1	9	3/10/92	18
			1214	Carlos	A2	4	4/10/92	18
			8191	Mário	A1	4	1/11/92	12
PAG02	Manutenção	Sistema de RH	8191	Mário	A1	4	1/05/93	12
			4112	João	A2	4	4/01/91	24
			6126	José	B1	9	1/11/92	12

Proj (CodProj, Tipo, Descr,
(CodEmp, Nome, Cat, Sal, DataIni, TempAI))

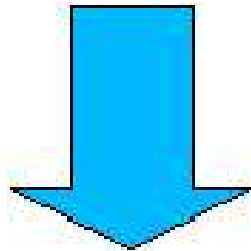
Exemplo

- Proj (cod_proj, tipo, descr, (cod_emp, nome, cat, sal, dt_ini, temp_al))
- O que fazer?
 - Construir uma tabela para cada tabela aninhada (decomposição de tabelas):
 - Proj (cod_proj, tipo, descr)
 - ProjEmp (cod_proj, cod_emp, nome, cat, sal, dt_ini, temp_al)

Exemplo

Uma única tabela

Proj (CodProj, Tipo, Descr,
(CodEmp, Nome, Cat, Sal, DataIni, TempAI))

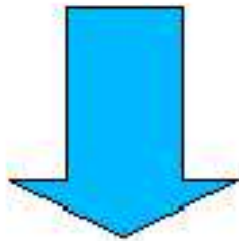


ProjEmp (CodProj, Tipo, Descr, CodEmp, Nome, Cat,
Sal, DataIni, TempAI)

Exemplo

Uma tabela para cada tabela aninhada

Proj (CodProj, Tipo, Descr,
(CodEmp, Nome, Cat, Sal, DataIni, TempAl))



Proj (CodProj, Tipo, Descr)

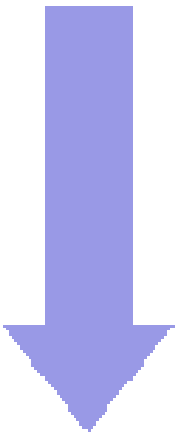
ProjEmp (CodProj, CodEmp, Nome, Cat, Sal, DataIni, TempAl)

Exemplo

$\tilde{N}N$

(CodProj, Tipo, Descr,

(CodEmp, Nome, Cat, Sal, DataIni, TempAl))



1FN

(CodProj, Tipo, Descr)

(CodProj, CodEmp, Nome, Cat,

Sal, DataIni, TempAl)

Exemplo

Proj:

CodProj	Tipo	Descr
LSC001	Novo Desenv.	Sistema de Estoque
PAG02	Manutenção	Sistema de RH

ProjEmp:

CodProj	CodEmp	Nome	Cat	Sal	DataIni	TempAl
LSC001	2146	João	A1	4	1/11/91	24
LSC001	3145	Sílvio	A2	4	2/10/91	24
LSC001	6126	José	B1	9	3/10/92	18
LSC001	1214	Carlos	A2	4	4/10/92	18
LSC001	8191	Mário	A1	4	1/11/92	12
PAG02	8191	Mário	A1	4	1/05/93	12
PAG02	4112	João	A2	4	4/01/91	24
PAG02	6126	José	B1	9	1/11/92	12

Segunda Forma Normal (2FN)

- Uma tabela encontra-se na segunda forma normal quando, além de estar na 1FN, não contém **dependências parciais**.
- **Dependência parcial**
 - Uma dependência (funcional) parcial ocorre quando uma coluna depende apenas de parte de uma chave primária composta.

Segunda Forma Normal (2FN)

- Estar na 1FN e **não** possuir dependências funcionais parciais em relação à chave primária.
- Exemplo: a seguinte relação não está na 2FN

Funcionário_Assistente

CPF_funcionário (PK)	CPF_assistente (PK)	Nome_assistente	Turno
----------------------	---------------------	-----------------	-------

- Na relação Funcionário_Assistente há uma dependência funcional parcial em relação à chave primária:

CPF_assistente → Nome_assistente

Segunda Forma Normal (2FN)

- Como colocar as relações na 2FN?
 - Inicialmente, colocá-las na 1FN.
 - Na relação Funcionário_Assistente, os nomes dos assistentes devem estar em outra relação.

Funcionário_Assistente

CPF_funcionário (PK)	CPF_assistente (PFK)	Turno
----------------------	----------------------	-------

Assistente

CPF_assistente (PK)	Nome_assistente
---------------------	-----------------

Segunda Forma Normal (2FN)

- Uma relação que não está na 2FN deve ser:
 - convertida para a 1FN.
 - Para cada chave parcial com seus atributos dependentes deve-se decompor e montar uma nova relação.
 - Deve-se manter uma relação com a chave primaria original e quaisquer atributos que sejam totais e funcionalmente dependentes dela.

Segunda Forma Normal (2FN)

- Uma relação está na **segunda forma normal (2FN)** se:
 - estiver na 1FN.
 - caso sua chave contenha múltiplos atributos, nenhum atributo não chave deve ser funcionalmente dependente de uma parte da chave primária
 - Ou seja: cada atributo não chave deve ter **dependência funcional total** em relação à chave primária.

Segunda Forma Normal (2FN)

segunda forma normal (2FN)

=

uma tabela encontra-se na segunda forma normal, quando, além de estar na 1FN,
não contém dependências parciais

Dependência parcial

dependência parcial
=
uma dependência (funcional) parcial
ocorre quando uma coluna depende
apenas de **parte de** uma chave primária
composta

ProjEmp (CodProj, CodEmp, Nome, Cat, Sal, DataIni, TempAl)



Dependência total x parcial

- Uma dependência funcional $X \rightarrow Y$ é uma **dependência total** se a remoção de qualquer atributo A de X significar que a dependência não se mantém mais.
- Uma **dependência** funcional $X \rightarrow Y$ é **parcial** se algum atributo A de X puder ser removido e a dependência ainda se mantiver.

Exemplo

ProjEmp:

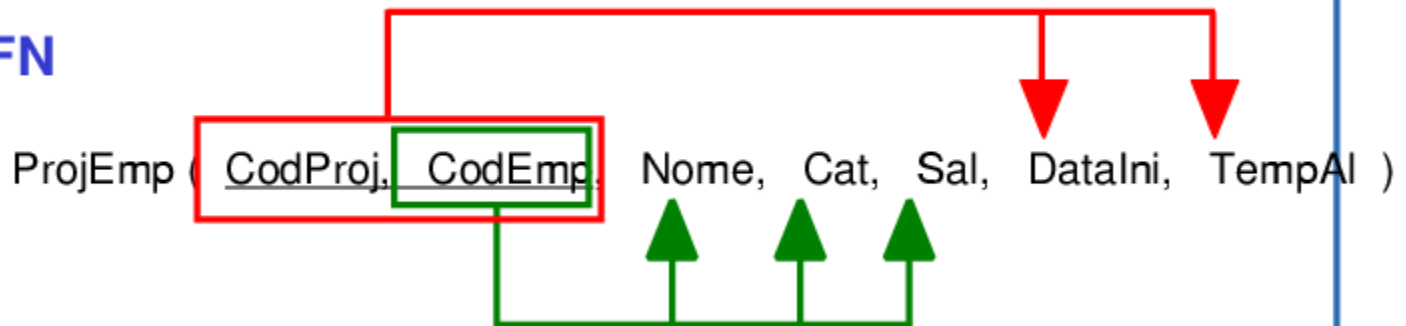
CódProj	CodEmp	Nome	Cat	Sal	DataIni	TempAl
LSC001	2146	João	A1	4	1/11/91	24
LSC001	3145	Sílvio	A2	4	2/10/91	24
LSC001	6126	José	B1	9	3/10/92	18
LSC001	1214	Carlos	A2	4	4/10/92	18
LSC001	8191	Mário	A1	4	1/11/92	12
PAG02	8191	Mário	A1	4	1/05/93	12
PAG02	4112	João	A2	4	4/01/91	24
PAG02	6126	José	B1	9	1/11/92	12

Exemplo

- Tabelas na 1FN:
 - Proj (cod_proj, tipo, descr)
 - ProjEmp (cod_proj, cod_emp, nome, cat, sal, dt_ini, temp_al)
- Passagem à 2FN:
 - Proj (cod_proj, tipo, descr)
 - ProjEmp (cod_proj, cod_emp, dt_ini, temp_al)
 - Emp (cod_emp, nome, cat, sal)

Exemplo

1FN



2FN

ProjEmp (CodProj, CodEmp, DataIni, TempAl)

Emp (CodEmp, Nome, Cat, Sal)

Exemplo

Proj:

CódProj	Tipo	Descr
LSC001	Novo Desenv.	Sistema de Estoque
PAG02	Manutenção	Sistema de RH

ProjEmp:

CódProj	CodEmp	Nome	DataIni	TempAI
LSC001	2146	João	1/11/91	24
LSC001	3145	Sílvio	2/10/91	24
LSC001	6126	José	3/10/92	18
LSC001	1214	Carlos	4/10/92	18
LSC001	8191	Mário	1/11/92	12
PAG02	8191	Mário	1/05/93	12
PAG02	4112	João	4/01/91	24
PAG02	6126	José	1/11/92	12

Emp:

CodEmp	Nome	Cat	Sal
2146	João	A1	4
3145	Sílvio	A2	4
1214	Carlos	A2	4
8191	Mário	A1	4
4112	João	A2	4
6126	José	B1	9

Terceira Forma Normal (3FN)

- Estar na 2FN e **não** possuir **dependências transitivas**.
- **Dependência transitiva**
 - Uma dependência funcional transitiva ocorre quando uma coluna, além de depender da chave primária da tabela, depende de outra coluna ou conjunto de colunas da tabela.
- Exemplo: a seguinte relação não está na 3FN

Produto

Cod_produto (PK)	Cod_categoria	Nome_categoria
------------------	---------------	----------------

- Na relação Produto há uma dependência transitiva:
 $\text{Cod_produto} \rightarrow \text{Cod_categoria} \rightarrow \text{Nome_categoria}$

Terceira Forma Normal (3FN)

- Como colocar as relações na 3FN?
 - Inicialmente, colocá-las na 2FN.
 - Na relação Produto, o nome da categoria deve estar em outra relação.

Produto

Cod_produto (PK)	Cod_categoria (FK)
------------------	--------------------

Categoria

Cod_categoria (PK)	Nome_categoria
--------------------	----------------

Exemplo

- A passagem à 3FN consiste em dividir tabelas de forma a eliminar as dependências transitivas.
- O exemplo a seguir ilustra uma tabela na 2FN:
 - Emp (cod_emp, nome, cat, sal)
 - Passagem à 3FN:
 - Emp (cod_emp, nome, cat)
 - Cat (cat, sal)

Terceira Forma Normal (3FN)

- Uma relação está na **terceira forma normal (3FN)** se estiver na 2FN e não tiver 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.
- Uma relação que não está na 3FN deve ser:
 - convertida para a 2FN.
 - Deve-se decompor e montar uma relação que inclua os atributos não chave que determinam funcionalmente outros atributos não chave.

Exercício

- Informe se as seguintes relações estão na 1FN, 2FN e 3FN e, caso não estejam, transforme-as na 3FN.

Cod_paciente(PK)	Nome	Telefones
------------------	------	-----------

CRM_médico (PK)	Cod_especialidade	Nome_médico	Nome_especialidade
-----------------	-------------------	-------------	--------------------

Cod_paciente(PK)	CRM_médico (PK)	Hora (PK)	Nome_médico	Remédios
------------------	-----------------	-----------	-------------	----------

Resumo formas normais

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.

Até a 3FN temos as dependências funcionais originais preservadas.
O mesmo não pode ser garantido nas demais formas normais, como BCNF e 4FN.

Outras formas normais?

- Para a maioria dos documentos e arquivos, a decomposição até a 3FN é suficiente para obter o esquema de um banco de dados correspondente ao documento.
- Na literatura aparecem outras formas normais que veremos a seguir, como:
 - Forma normal de Boyce/Codd (3,5FN).
 - 4FN.
 - 5FN.

Forma Normal de Boyce-Codd

- **Cada atributo** deve representar um fato sobre a chave, a chave inteira, e nada mais do que a chave.

Chris Date

- 3FN:
 - **Cada atributo não chave** deve representar um fato sobre a chave, a chave inteira, e nada mais do que a chave.
- BCNF aplica a regra da 3FN para incluir os atributos atributos chave.

Forma Normal de Boyce-Codd

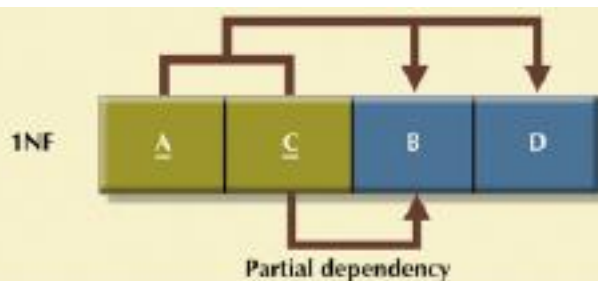
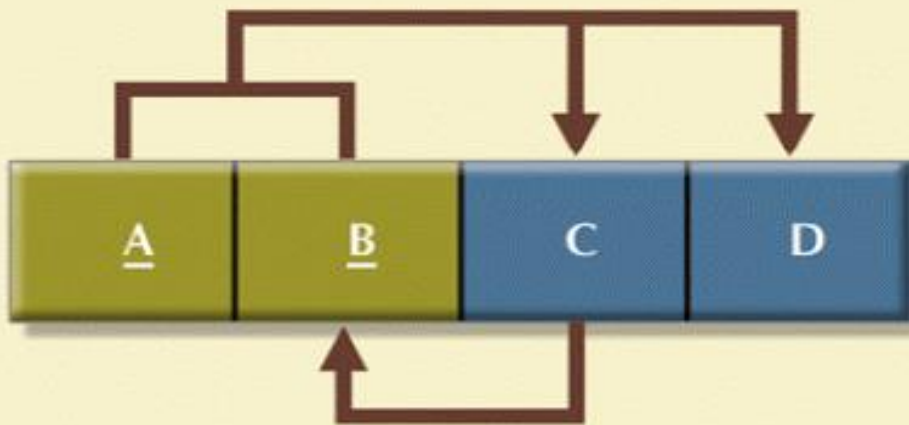
- 1FN, 2FN e 3FN não levam em conta se as dependências funcionais permanecem em outras chaves candidatas da relação.
- Baseia-se em dependências funcionais que levam em conta todas as chaves candidatas de uma relação.
- Todo determinante na tabela é uma chave candidata.
- Quando há somente uma chave candidata, 3FN e BCNF são equivalentes.
- BCNF só pode ser violada quando a tabela contém mais de uma chave candidata.

Forma Normal de Boyce-Codd

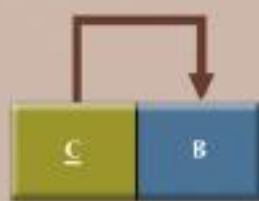
- Uma tabela está na FNBC se:
 - Estiver na 3FN
 - Todo determinante na tabela é uma chave candidata.
- Ocorre violação de BCNF quando um atributo não chave é o determinante de um atributo chave.

Forma Normal de Boyce-Codd

Relação em 3FN, mas não em BCNF.

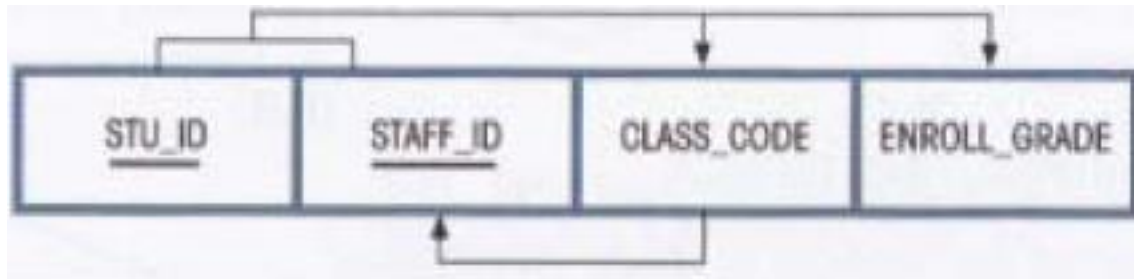


3NF and BCNF

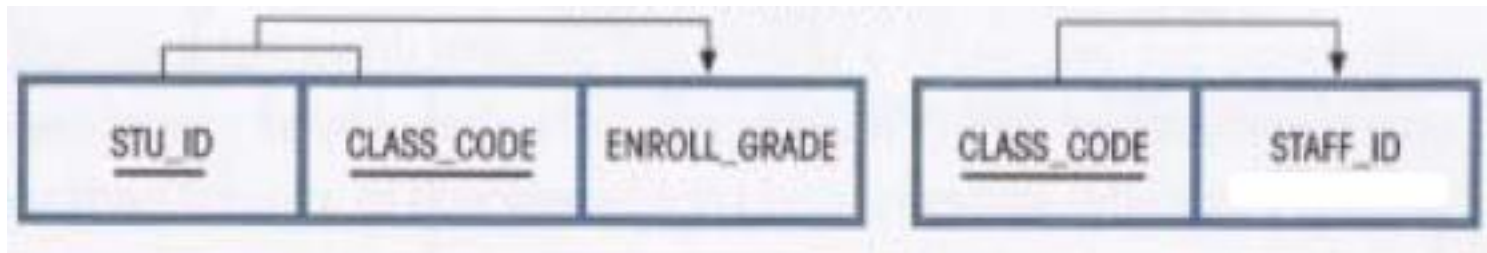


3NF and BCNF

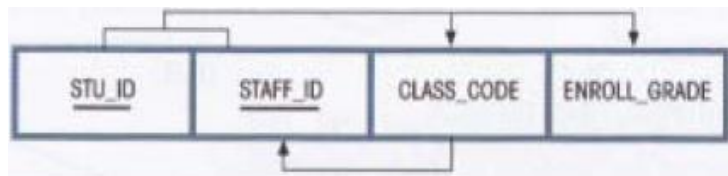
Forma Normal de Boyce-Codd



Relação em 3FN, mas não em BCNF.



Forma Normal de Boyce-Codd



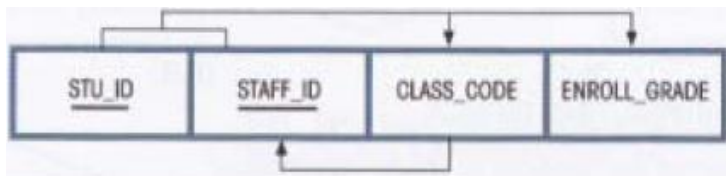
<u>aluno_id</u>	<u>prof_id</u>	turma	nota
125	25	21334	A
125	20	32456	C
135	20	28458	B
135	25	27563	C
144	20	32456	B

Relação em 3FN, mas não em BCNF.

Problema:

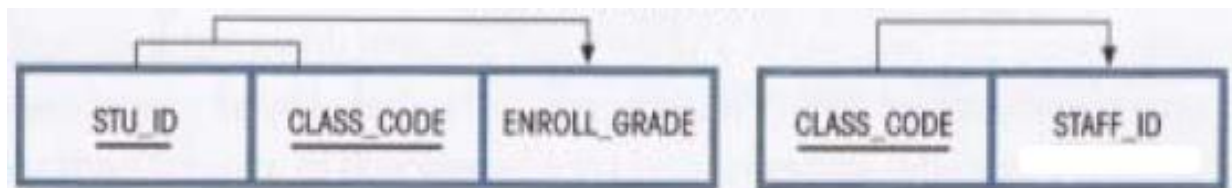
- Se um outro professor for dar aula para a turma 32456, duas tuplas devem ser atualizadas.
- Se o aluno 135 abandonar o curso, perdemos o dado que relaciona o professor com a turma.

Forma Normal de Boyce-Codd



<u>aluno_id</u>	<u>prof_id</u>	turma	nota
125	25	21334	A
125	20	32456	C
135	20	28458	B
135	25	27563	C
144	20	32456	B

Relação em 3FN, mas não em BCNF.



<u>aluno_id</u>	<u>turma</u>	nota
125	21334	A
125	32456	C
135	28458	B
135	27563	C
144	32456	B

<u>turma</u>	<u>prof_id</u>
21334	25
32456	20
28458	20
27563	25

Forma Normal de Boyce-Codd

- Violada quando:
 1. uma tabela tem múltiplas chaves candidatas compostas e;
 2. um atributo de uma chave candidata tem uma dependência funcional de parte de outra chave candidata.

Forma Normal de Boyce-Codd

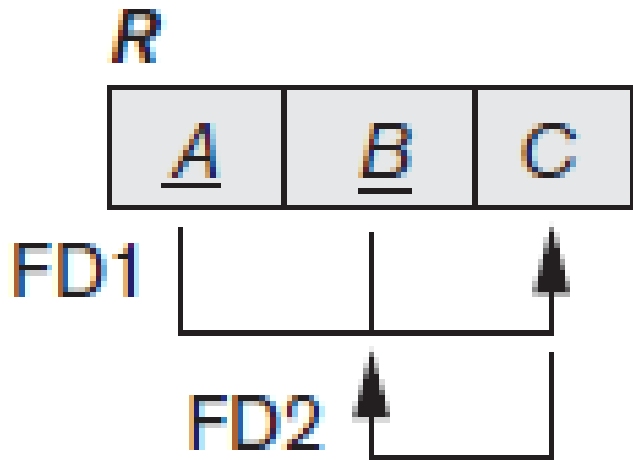
- Condições para violação da BCNF
 1. Deve haver múltiplas chaves candidatas.
 2. As chaves são compostas de múltiplos atributos.
 3. Há atributos comuns entre as chaves.

Forma Normal de Boyce-Codd

- O que fazer em caso de violação
 - Se houver dependências não triviais entre atributos de chaves candidatas, deve-se separá-los em tabelas distintas.

Forma Normal de Boyce-Codd

- Uma relação está na BCNF se todos os atributos são funcionalmente dependentes da chave, de toda a chave e nada mais do que a chave.



Relação em 3FN, mas não em FNBC.

Chaves candidatas:

- A, B
- A, C

Decomposição em BCNF:

- R1(A, C)
- R2(C, B)

Quarta forma normal (4FN)

- Uma tabela se encontra na quarta forma normal, quando, além de estar na 3FN, não contém dependências multivaloradas.
- Uma coluna ou conjunto de colunas depende multivaloradamente de uma coluna (determinante) da mesma tabela, quando um valor do atributo determinante identifica repetidas vezes um conjunto de valores da coluna dependente.
- As dependências multivaloradas normalmente surgem devido à mistura de atributos multivalorados independentes em uma única relação.

Quarta forma normal (4FN)

quarta forma normal (4FN)
=
uma tabela encontra-se na quarta forma normal, quando, além de estar na 3FN, **não contém mais de uma dependência multi-valorada**

Exemplo

- Util(cod_proj, cod_emp, cod equip)
- Redudância: para cada equipamento usado no projeto é necessário informar todos os seus empregados. Assim, a informação de quais equipamentos são usados em um projeto está armazenada redundantemente.
- Portanto, a tabela Util não está em 4FN e deve ser decomposta em duas tabelas:
 - ProjEmp (cod_proj, cod_emp)
 - ProjEquip (cod_proj, cod equip)

Exemplo

CodProj	CodEmp	CodEquip
1	1	1
1	2	1
1	3	1
1	1	2
1	2	2
1	3	2
2	2	2
2	2	4
3	3	1
3	4	1
3	3	3
3	4	3
3	3	5
3	4	5
4	2	5


Verificar:

- ⇒ Quantas vezes cada empregado do projeto 1 é informado?
- ⇒ Quantas vezes cada equipamento usado no projeto 1 é informado?

Dependências multivaloradas

CodProj	CodEmp	CodEquip
1	1	1
1	2	1
1	3	1
1	1	2
1	2	2
1	3	2
2	2	

 CodProj \twoheadrightarrow CodEmp

 CodProj \twoheadrightarrow CodEquip

Deixando na 4FN...

ProjEmp (CodProj, CodEmp)

ProjEquip (CodProj, CodEquip)

Exemplo

FUNC

<u>Fnome</u>	<u>Projnome</u>	<u>Dnome</u>
Silva	X	João
Silva	Y	Ana
Silva	X	Ana
Silva	Y	João

FUNC_PROJETOS

<u>Fnome</u>	<u>Projnome</u>
Silva	X
Silva	Y

FUNC_DEPENDENTES

<u>Fnome</u>	<u>Nome_dependente</u>
Silva	João
Silva	Ana

Quinta Forma Normal (5FN)

- A quinta forma normal é baseada na dependência de junção e identifica uma restrição peculiar que faz que uma relação seja decomposta em vários componentes, de modo que sempre produzam a relação original de volta, após uma junção.
- Na prática, a maioria dos projetos comerciais seguiu as formas normais até a FNBC.
- A necessidade de decomposição para a 5FN raramente surge na prática, e as dependências de junção são difíceis de detectar para a maioria das situações práticas, tornando a 5FN de valor mais teórico.

Quinta Forma Normal (5FN)

- Decomponha a relação em múltiplas relações que devem ser sem perda e manter as dependências da relação original (dependência de junção).
- Uma relação está na 5FN se:
 - Estiver na 4FN.
 - Se houver dependência de junção, a relação deve ser decomposta.

Quinta Forma Normal (5FN)

- Fábrica produz Componente que é entregue ao Projeto.

Factory	Component	Project
GM	Engine	MPC
GM	Gear Box	125A
Honda	Engine	125A
GM	Engine	125A

Quinta Forma Normal (5FN)

Factory	Component	Project
GM	Engine	MPC
GM	Gear Box	125A
Honda	Engine	125A
GM	Engine	125A

Original

- Decomposição

R1 - Fábrica + Componente

Factory	Component
GM	Engine
GM	Gear Box
Honda	Engine

R2 - Componente + Projeto

Component	Project
Engine	MPC
Gear Box	125A
Engine	125A

Junção (R1, R2)

Factory	Component	Project
GM	Engine	MPC
GM	Engine	125A
GM	Gear Box	125A
Honda	Engine	MPC
Honda	Engine	125A

Quinta Forma Normal (5FN)

Factory	Component	Project
GM	Engine	MPC
GM	Gear Box	125A
Honda	Engine	125A
GM	Engine	125A

Original

- Decomposição

R1 - Fábrica + Componente

Factory	Component
GM	Engine
GM	Gear Box
Honda	Engine

R2 - Componente + Projeto

Component	Project
Engine	MPC
Gear Box	125A
Engine	125A

R3 - Fábrica + Projeto

Factory	Project
GM	MPC
GM	125A
Honda	125A

Factory	Component	Project
GM	Engine	MPC
GM	Engine	125A
GM	Gear Box	125A
Honda	Engine	125A

Junção(R1, R2, R3) recupera a relação original.

A tabela original possui dependência de junção, devendo ser decomposta em R1, R2 e R3.

Quinta Forma Normal (5FN)

- Uma fábrica produz todos os componentes.
- Ela fornece pelo menos um dos componentes para cada projeto.

Factory	Component	Project
GM	Engine	MPC
GM	Gear Box	125A
Honda	Engine	125A
Honda	Gear Box	MPC
GM	Engine	125A

Quinta Forma Normal (5FN)

Factory	Component	Project
GM	Engine	MPC
GM	Gear Box	125A
Honda	Engine	125A
Honda	Gear Box	MPC
GM	Engine	125A

Original

Não há dependência de junção.
Portanto, a relação original está na 5FN.

• Decomposição

R1 - Fábrica + Componente

Factory	Component
GM	Engine
GM	Gear Box
Honda	Engine
Honda	Gear Box

R2 - Componente + Projeto

Component	Project
Engine	MPC
Gear Box	125A
Engine	125A
Gear Box	MPC

R3 - Fábrica + Projeto

Factory	Project
GM	MPC
GM	125A
Honda	125A
Honda	MPC

Factory	Component	Project
GM	Engine	MPC
GM	Engine	125A
GM	Gear Box	125A
GM	Gear Box	MPC
Honda	Engine	MPC
Honda	Engine	125A
Honda	Gear Box	125A
Honda	Gear Box	MPC

Junção(R1, R2)

Junção(R1, R2, R3)

Factory	Component	Project
GM	Engine	MPC
GM	Engine	125A
GM	Gear Box	125A
GM	Gear Box	MPC
Honda	Engine	MPC
Honda	Engine	125A
Honda	Gear Box	125A
Honda	Gear Box	MPC

Referências

- Elsmari, R., Navathe, Shamkant B. “Sistemas de Banco de Dados”. 6ª Edição, Pearson Brasil, 2011. → Capítulo 15
- Silberschatz, A., Korth, H., Sudarshan, S. “Sistema de Banco de Dados”. 5ª Edição, Editora Campus, 2006.
- Slides Prof. José Maria (UFC).
- Material CEDERJ.
- <http://web.archive.org/web/20080805014412/http://www.datamodel.org/NormalizationRules.html>
- <http://yukidatabase.blogspot.com.br/2012/08/boyce-codd-normal-form.html>
- <https://www.youtube.com/watch?v=CxUQ8V3uyg4>

Obrigado!
Dúvidas, comentários, sugestões?

Prof. Manoel Ribeiro
manoel@opencare.com.br



OPENCARE[®]
Inovação e Tecnologia