

# Sistemas de Banco de Dados Projeto, implementação e gerenciamento

## *Capítulo 5*

Normalização das tabelas do banco de dados

# Objetivos

- **Neste capítulo, você aprenderá:**
  - O que é normalização e qual o papel executado no processo de projetos de bancos de dados
  - Sobre as formas normais 1NF, 2NF, 3NF, BCNF e 4NF
  - Como transformar formas normais mais baixas em formas normais mais altas
  - Que a normalização e a modelagem ER devem ser utilizadas de forma simultânea na produção de um bom projeto
  - Que algumas situações exigem desnormalização para gerar informações de modo eficiente

# Tabelas de Bancos de Dados e Normalização

- **Normalização**

- Processo para avaliar e corrigir estruturas e tabelas de modo a minimizar as redundâncias de dados, reduzindo, assim, a probabilidade de anomalias
- Atua por meio de uma série de estágios chamados formas normais
  - Primeira forma normal (1NF)
  - Segunda forma normal (2NF)
  - Terceira forma normal (3NF).

# Tabelas de Bancos de Dados e Normalização (cont.)

- **Normalização** (cont.)
  - 3NF é melhor que 2NF, que, por sua vez, é melhor que 1NF
  - Para a maioria das finalidades dos projetos de bancos de dados comerciais, 3NF é o mais alto que se precisa ir no processo de normalização
  - Não se deve assumir que o nível mais alto de normalização seja sempre o mais desejável

# Tabelas de Bancos de Dados e Normalização (cont.)

- **Desnormalização** – produz uma forma normal mais baixa
  - O preço a pagar pela melhora de desempenho decorrente da desnormalização é a maior redundância de dados

# Necessidade de Normalização

- Exemplo: empresa de construção que gerencie vários projetos
  - Cobra seus clientes pelas horas gastas em cada contrato
  - A taxa de cobrança horária depende da posição do funcionário
  - Periodicamente, gera-se um relatório que contém as informações exibidas na Tabela 5.1

TABELA 5.1 Exemplo de layout do relatório

NÚMERO DO PROJETO	NOME DO PROJETO	NÚMERO DO FUNCIONÁRIO	NOME DO FUNCIONÁRIO	CLASSIFICAÇÃO DE CARGO	TARIFA/HORA	HORAS COBRADAS	TARIFA TOTAL
15	Evergreen	103	June E. Arbough	Engenheiro Eletricista	\$ 85,50	23,8	\$ 2.034,90
		101	John G. News	Projetista de Banco de Dados	\$105,00	19,4	\$ 2.037,00
		105	Alice K. Johnson *	Projetista de Banco de Dados	\$105,00	35,7	\$ 3.748,50
		106					
		102	William Smithfield	Programador	\$ 35,75	12,6	\$ 450,45
			David H. Senior	Analista de Sistemas	\$ 96,75	23,8	\$ 2.302,65
				Subtotal			<b>\$10.573,50</b>
18	Amber Wave	114	Annelise Jones	Projetista de Aplicações	\$ 48,10	25,6	\$ 1.183,26
		118	James J. Frommer	Suporte Geral	\$ 18,36	45,3	\$ 831,71
		104	Anne K. Ramoras *	Analista de Sistemas	\$ 96,75	32,4	\$ 3.134,70
		112	Darlene M. Smithson	Analista SSD	\$ 45,95	45,0	\$ 2.067,75
				<b>Subtotal</b>			<b>\$ 7.265,52</b>
22	Rolling Tide	105	Alice K. Johnson	Projetista de Banco de Dados	\$105,00	65,7	\$ 6.998,50
		104	Anne K. Ramoras	Analista de Sistemas	\$ 96,75	48,4	\$ 4.682,70
		113					
		111	Delbert K. Joenbrood	Projetista de Aplicações	\$ 48,10	23,6	\$ 1.135,16
		106	Geoff B. Wabash	Suporte Escriturário	\$ 26,87	22,0	\$ 591,14
			William Smithfield	Programador	\$ 35,75	12,8	\$ 457,60
				<b>Subtotal</b>			<b>\$13.765,10</b>
25	Starflight	107	Maria D. Alonzo	Programador	\$ 35,75	25,6	\$ 915,20
		115	Travis B. Bawangi	Analista de Sistemas	\$ 96,75	45,8	\$ 4.431,15
		101	John G. News *	Projetista de Banco de Dados	\$105,00	56,3	\$ 5.911,50
		114	Annelise Jones	Projetista de Aplicações	\$ 48,10	33,1	\$ 1.592,11
		108					
		118	Ralph B. Washington	Analista de Sistemas	\$ 96,75	23,6	\$ 2.283,30
		112	James J. Frommer	Suporte Geral	\$ 18,36	30,5	\$ 559,98
			Darlene M. Smithson	Analista SSD	\$ 45,95	41,4	\$ 1.902,33
				<b>Subtotal</b>			<b>\$17.595,57</b>
				Total			<b>\$49.199,69</b>

Observação: \* indica o líder do projeto

**FIGURA  
5.1**

**Exemplo de layout do relatório**

Nome da tabela: RPT\_FORMA

PROJ_NUM	PROJ_NAME	EMP_NUM	EMP_NAME	JOB_CLASS	CHG_HOUR	HOURS
15	Evergreen	103	June E. Arbough	Eng <sup>o</sup> Eletricista	\$ 84,50	23,8
		101	John G. News	Projetista de Bancos de Dados	\$ 105,00	19,4
		105	Alice K. Johnson *	Projetista de Bancos de Dados	\$ 105,00	35,7
		106	William Smithfield	Programador	\$ 35,75	12,6
		102	David H. Senior	Analista de Sistemas	\$ 96,75	23,8
18	Amber Wave	114	Annelise Jones	Projetista de Aplicações	\$ 48,10	25,6
		118	James J. Frommer	Suporte Geral	\$ 18,36	45,3
		104	Anne K. Ramoras *	Analista de Sistemas	\$ 96,75	32,4
		112	Darlene M. Smithson	Analista SSD	\$ 45,95	45,0
22	Rolling Tide	105	Alice K. Johnson	Projetista de Bancos de Dados	\$ 105,00	65,7
		104	Anne K. Ramoras	Analista de Sistemas	\$ 96,75	48,4
		113	Delbert K. Joenbrood	Projetista de Aplicações	\$ 48,10	23,6
		111	Geoff B. Wabash	Suporte Escriturário	\$ 26,87	22,0
		106	William Smithfield	Programador	\$ 35,75	12,8
25	Starflight	107	Maria D. Alonzo	Programador	\$ 35,75	25,6
		115	Travis B. Bawangi	Analista de Sistemas	\$ 96,75	45,8
		101	John G. News *	Projetista de Bancos de Dados	\$ 105,00	56,3
		114	Annelise Jones	Projetista de Aplicações	\$ 48,10	33,1
		108	Ralph B. Washington	Analista de Sistemas	\$ 96,75	23,6
		118	James J. Frommer	Suporte Geral	\$ 96,75	30,5
		112	Darlene M. Smithson	Analista SSD	\$ 45,95	41,4



# Necessidade de Normalização (cont.)

- A estrutura do conjunto de dados da Figura 5.1 não manipula os dados muito bem
- A estrutura da tabela *parece* funcionar e o relatório é gerado com facilidade
- Infelizmente, esse relatório pode produzir resultados diferentes, dependendo de qual anomalia de dados ocorreu
- O ambiente relacional é especialmente adequado para ajudar o projetista a superar esses problemas

# Processo de Normalização

- Cada tabela representa um único assunto
- Nenhum item de dados será armazenado *desnecessariamente* em mais de uma tabela
- Todos os atributos não primários de uma tabela são dependentes da chave primária
- Todas as tabelas estão livres de anomalias de inserção, atualização e exclusão

**TABELA 5.2 Formas normais**

FORMA NORMAL	CARACTERÍSTICA	SEÇÃO
Primeira forma normal (1NF)	Formato de tabela, sem grupos repetidos e com PK identificada	Conversão para a primeira forma normal
Segunda forma normal (2NF)	1NF sem dependências parciais	Conversão para a segunda forma normal
Terceira forma normal (3NF)	2NF sem dependências transitivas	Conversão para a terceira forma normal
Forma normal de Boyce-Codd (BCNF)	Todo determinante é uma chave candidata (caso especial de 3NF)	Forma normal de Boyce-Codd (BCNF)
Quarta forma normal (4NF)	3NF sem dependências multivaloradas independentes	Quarta forma normal (4NF)

# Processo de Normalização (cont.)

- O objetivo da normalização é garantir que todas as tabelas estejam, pelo menos, na terceira forma normal (3NF)
- Formas normais de nível superior provavelmente não serão encontradas em um ambiente comercial
- O processo de normalização trabalha em uma relação por vez
- Progressiva separação da relação (tabela) em um conjunto de novas relações (tabelas) baseadas nas dependências identificadas

**TABELA 5.3** Conceitos de dependência funcional

CONCEITO	DEFINIÇÃO
<p><b>Dependência funcional</b></p>	<p>O atributo <b>B</b> é dependente, de modo totalmente funcional, do atributo <b>A</b> se cada valor de <b>A</b> determina um e somente um valor de <b>B</b>.</p> <p>Exemplo: PROJ_NUM → PROJ_NAME</p> <p>(leia-se PROJ_NUM determina funcionalmente PROJ_NAME)</p> <p>Nesse caso, PROJ_NUM é conhecido como atributo determinante e PROJ_NAME, como atributo dependente.</p>
<p>Dependência funcional (definição generalizada)</p>	<p>O atributo <b>A</b> determina o atributo <b>B</b> (ou seja, <b>B</b> é funcionalmente dependente de <b>A</b>) se todas as linhas da tabela que correspondem em valor ao atributo <b>A</b> também correspondem em valor ao atributo <b>B</b>.</p>
<p>Dependência totalmente funcional (chave composta)</p>	<p>Se o atributo <b>B</b> é funcionalmente dependente de uma chave composta <b>A</b>, mas não de qualquer subconjunto dessa chave composta, o atributo <b>B</b> apresenta dependência totalmente funcional em relação a <b>A</b>.</p>

# Conversão para a Primeira Forma Normal

- **Grupo de repetição**
  - Um grupo de várias entradas do mesmo tipo pode existir para qualquer ocorrência *única* de atributo de chave
- Uma tabela relacional deve conter os grupos de repetição
- A normalização de sua estrutura reduzirá essas redundâncias
- A normalização é um procedimento em três etapas

# Conversão para a Primeira Forma Normal (cont.)

- **Etapa 1:** Elimine os grupos de repetição
  - Elimine os nulos, assegurando que cada grupo de repetição contenha um valor de dados adequado
- **Etapa 2:** Identifique a chave primária
  - Deve identificar *exclusivamente* qualquer valor de atributo
  - Deve ser composta
- **Etapa 3:** Identifique todas as dependências
  - As dependências podem ser representadas com a ajuda de um diagrama

**FIGURA  
5.2****Tabela na primeira forma normal**

Nome da tabela: DATA\_ORG\_1NF

PROJ_NUM	PROJ_NAME	EMP_NUM	EMP_NAME	JOB_CLASS	CHG_HOUR	HOURS
15	Evergreen	103	June E. Arbough	Elect. Engineer	84.50	23.8
15	Evergreen	101	John G. News	Database Designer	105.00	19.4
15	Evergreen	105	Alice K. Johnson *	Database Designer	105.00	35.7
15	Evergreen	106	William Smithfield	Programmer	35.75	12.6
15	Evergreen	102	David H. Senior	Systems Analyst	96.75	23.8
18	Amber Wave	114	Annelise Jones	Applications Designer	48.10	24.6
18	Amber Wave	118	James J. Frommer	General Support	18.36	45.3
18	Amber Wave	104	Anne K. Ramoras *	Systems Analyst	96.75	32.4
18	Amber Wave	112	Darlene M. Smithson	DSS Analyst	45.95	44.0
22	Rolling Tide	105	Alice K. Johnson	Database Designer	105.00	64.7
22	Rolling Tide	104	Anne K. Ramoras	Systems Analyst	96.75	48.4
22	Rolling Tide	113	Delbert K. Joenbrood *	Applications Designer	48.10	23.6
22	Rolling Tide	111	Geoff B. Wabash	Clerical Support	26.87	22.0
22	Rolling Tide	106	William Smithfield	Programmer	35.75	12.8
25	Starflight	107	Maria D. Alonzo	Programmer	35.75	24.6
25	Starflight	115	Travis B. Bawangi	Systems Analyst	96.75	45.8
25	Starflight	101	John G. News *	Database Designer	105.00	56.3
25	Starflight	114	Annelise Jones	Applications Designer	48.10	33.1
25	Starflight	108	Ralph B. Washington	Systems Analyst	96.75	23.6
25	Starflight	118	James J. Frommer	General Support	18.36	30.5
25	Starflight	112	Darlene M. Smithson	DSS Analyst	45.95	41.4

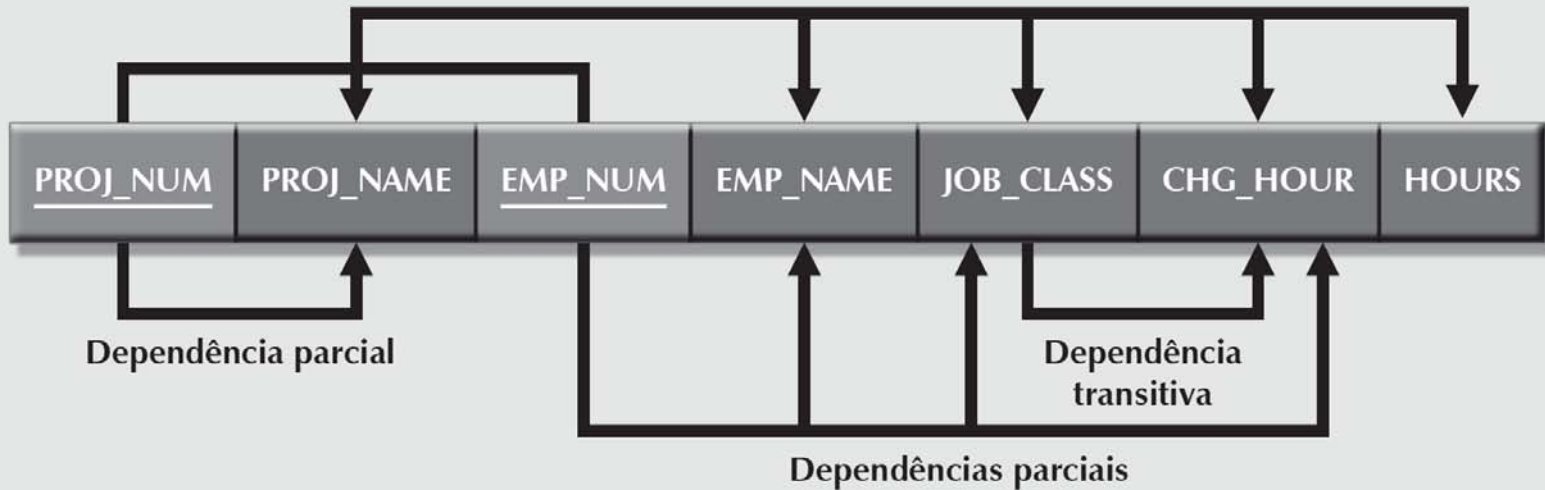


# Conversão para a Primeira Forma Normal (cont.)

- **Diagrama de dependência:**
  - Apresenta todas as dependências encontradas em uma determinada estrutura de tabela
  - São muito úteis para obter uma visão “de cima” de todos os relacionamentos entre atributos de uma tabela
  - Seu uso torna menos provável a omissão de uma dependência importante

**FIGURA 5.3**

**Diagrama de dependência da primeira forma normal (1NF)**



1NF (PROJ\_NUM, EMP\_NUM, PROJ\_NAME, EMP\_NAME, JOB\_CLASS, CHG\_HOURS, HOURS)

DEPENDÊNCIAS PARCIAIS:

(PROJ\_NUM → PROJ\_NAME)

(EMP\_NUM → EMP\_NAME, JOB\_CLASS, CHG\_HOUR)

DEPENDÊNCIA TRANSITIVA:

(JOB CLASS → CHG\_HOUR)

# Conversão para a Primeira Forma Normal (cont.)

- O termo **primeira forma normal (1NF)** descreve um formato de tabela em que:
  - Todos os atributos de chave estão definidos
  - Não há grupos de repetição na tabela
  - Todos os atributos são dependentes da chave primária
- Todas as tabelas relacionais satisfazem as exigências de 1NF
- Algumas tabelas contêm dependências parciais
  - Dependências baseadas em apenas uma parte da chave primária
  - Devem ser aplicadas com precaução

# Conversão para a Segunda Forma Normal

- **Etapa 1:** Apresente cada componente de chave em uma linha separada
  - Apresente cada componente da chave em uma linha separada. Em seguida, escreva a chave original (composta) na última linha
  - Cada componente se tornará a chave de uma nova tabela

# Conversão para a Segunda Forma Normal (cont.)

- **Etapa 2:** Distribua os respectivos atributos dependentes
  - Determine os atributos dependentes de outros
  - Nesse ponto, a maioria das anomalias foi eliminada

**FIGURA 5.4**

**Resultado da conversão para segunda forma normal (2NF)**

Nome da tabela: PROJETO

PROJETO (PROJ\_NUM, PROJ\_NAME)



Nome da tabela: FUNCIONÁRIO

FUNCIONÁRIO (EMP\_NUM, EMP\_NAME, JOB\_CLASS, CHG\_HOUR)



DEPENDÊNCIA TRANSITIVA  
(JOB\_CLASS → CHG\_HOUR)

Dependência transitiva

Nome da tabela: DESIGNAÇÃO

DESIGNAÇÃO (PROJ\_NUM, EMP\_NUM, ASSIGN\_HOURS)



# Conversão para a Segunda Forma Normal (cont.)

- A tabela está na **segunda forma normal (2NF)** quando:
  - Está em 1NF
  - Não inclui dependências parciais;
    - Nenhum atributo é dependente apenas de uma parte da chave primária

# Conversão para a Terceira Forma Normal

- **Etapa 1:** Identifique todos os novos determinantes
  - Para todas as dependências transitivas, apresente seu determinante como PK de uma nova tabela
  - O **determinante** é qualquer atributo cujo valor determine outros valores na mesma linha
- **Etapa 2:** Identifique os atributos dependentes
  - Identifique os atributos dependentes de cada determinante apresentado na Etapa 1
    - Apresente a dependência
  - O nome da tabela reflète seu conteúdo e função



# Conversão para a Terceira Forma Normal (cont.)

- **Etapa 3:** Remova os atributos dependentes das dependências transitivas
  - Elimine todos os atributos dependentes no(s) relacionamento(s) transitivo(s) de cada tabela que apresente esse relacionamento
  - Trace um novo diagrama de dependência para mostrar todas as tabelas definidas nas Etapas 1-3
  - Verifique as novas tabelas, assim como as modificadas na Etapa 3
    - Cada tabela deve ter um determinante
    - Nenhuma tabela deve conter dependências inadequadas

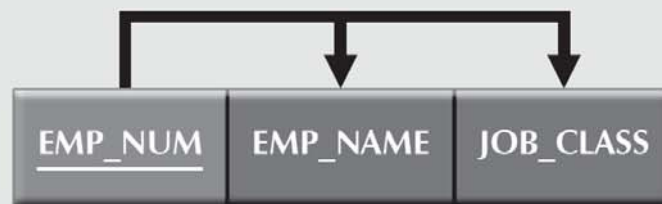
**FIGURA 5.5**

Resultado da conversão para a terceira forma normal (3NF)



Nome da tabela: PROJETO

PROJETO (PROJ\_NUM, PROJ\_NAME)



Nome da tabela: FUNCIONÁRIO

FUNCIONÁRIO (EMP\_NUM, EMP\_NAME, JOB\_CLASS)



Nome da tabela: CARGO

CARGO (JOB\_CLASS, CHG\_HOUR)



Nome da tabela: DESIGNAÇÃO

DESIGNAÇÃO (PROJ\_NUM, EMP\_NUM, ASSIGN\_HOURS)

# Conversão para a Terceira Forma Normal (cont.)

- A tabela está na **terceira forma normal (3NF)** quando:
  - Está em 2NF
  - Não contém dependências transitivas

# Aprimoramento do Projeto

- As estruturas da tabela são “limpas” para eliminar as problemáticas dependências transitivas e parciais
- Não se pode tomar a normalização, por si mesma, como uma garantia de bom projeto
- O valor da normalização se deve à sua ajuda na eliminação de redundâncias de dados

# Aprimoramento do Projeto (cont.)

- Diferentes tipos de questões que devem ser tratados para produzir um bom conjunto de tabelas normalizadas:
  - Avaliação das atribuições de PK
  - Avaliação das convenções de nomenclatura
  - Refinamento da atomicidade de atributos
  - Identificação de novos atributos
  - Identificação de novos relacionamentos
  - Refinamento de chaves primárias conforme necessário para a granularidade dos dados
  - Manutenção da precisão histórica
  - Avaliação por meio de atributos derivados

**FIGURA 5.6**

**FIGURA 5.6 Banco de dados concluído (continuação)**

Nome da tabela: FUNCIONÁRIO



Nome da tabela: FUNCIONÁRIO

EMP_NUM	EMP_LNAME	EMP_FNAME	EMP_INITIAL	EMP_HIREDATE	JOB_CODE
101	News	John	G	08-Nov-00	502
102	Senior	David	H	12-Jul-89	501
103	Arbough	June	E	01-Dec-97	503
104	Ramoras	Anne	K	15-Nov-88	501
105	Johnson	Alice	K	01-Feb-94	502
106	Smithfield	William		22-Jun-05	500
107	Alonzo	Maria	D	10-Oct-94	500
108	Washington	Ralph	B	22-Aug-89	501
109	Smith	Larry	W	18-Jul-99	501
110	Olenko	Gerald	A	11-Dec-96	505
111	Wabash	Geoff	B	04-Apr-89	506
112	Smithson	Darlene	M	23-Oct-95	507
113	Joebrood	Delbert	K	15-Nov-94	508
114	Jones	Annelise		20-Aug-91	508
115	Bawangi	Travis	B	25-Jan-90	501
116	Pratt	Gerald	L	05-Mar-95	510
117	Williamson	Angie	H	19-Jun-94	509
118	Frommer	James	J	04-Jan-06	510

# Considerações sobre Chaves Surrogates

- Quando a chave primária for considerada inadequada, os projetistas utilizam chaves surrogates
- As entradas da Tabela 5.4 são inadequadas, pois duplicam registros existentes
  - Não há violação de integridade referencial ou de entidade

TABELA 5.4 Entradas duplicadas da Tabela Cargo

JOB_CODE	JOB_DESCRIPTION	JOB_CHG_HOUR
511	Programador	\$35,75
512	Programador	\$35,75

# Formas Normais de Nível Superior

- As tabelas em 3NF funcionarão, de forma adequada, em um banco de dados transacional de negócios
- Há ocasiões em que as formas normais superiores são úteis
- Dois casos especiais de 3Nf:
  - Forma normal de Boyce-Codd (BCNF)
  - Quarta forma normal (4NF)



# Forma Normal de Boyce-Codd (BCNF)

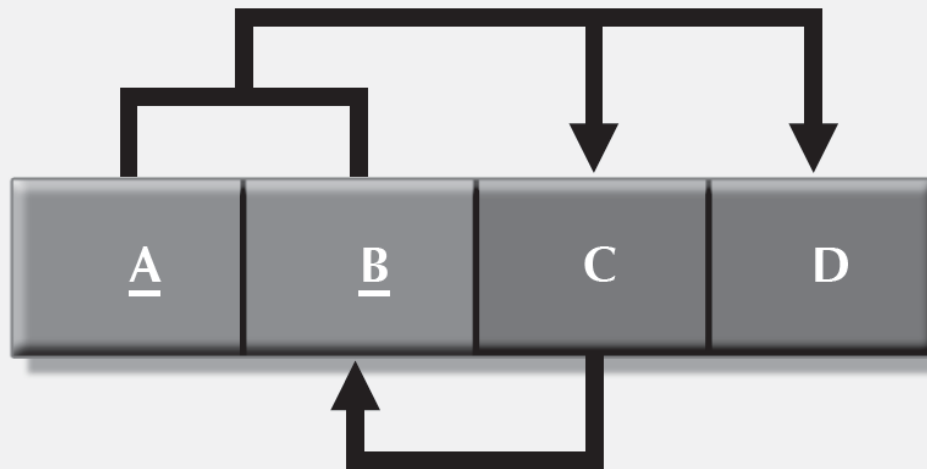
- Uma tabela está na **forma normal de Boyce-Codd (BCNF)** quando todos os seus determinantes são chaves candidatas
  - Uma chave candidata possui as mesmas características de uma chave primária key, mas por algum motivo, não foi escolhida para tal
- Quando uma tabela contém apenas uma chave candidata, 3NF e BCNF são equivalentes
- BCNF pode ser violada somente quando a tabela contiver mais de uma chave candidata

# Forma Normal de Boyce-Codd (BCNF) (cont.)

- A maioria dos projetistas considera BCNF como um caso especial de 3NF
- Uma tabela está em 3NF quando apresenta as características de 2NF, mas nenhuma dependência transitiva

**FIGURA  
5.7**

**Tabela em 3NF, mas não  
em BCNF**



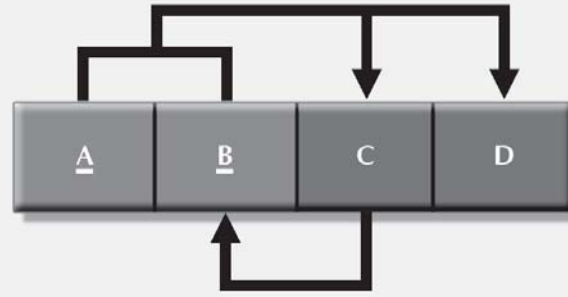
**TABELA 5.5 Exemplo de dados para uma conversão em BCNF**

STU_ID	STAFF_ID	CLASS_CODE	ENROLL_GRADE
125	25	21334	A
125	20	32456	C
135	20	28458	B
144	25	27563	C
144	20	32456	B

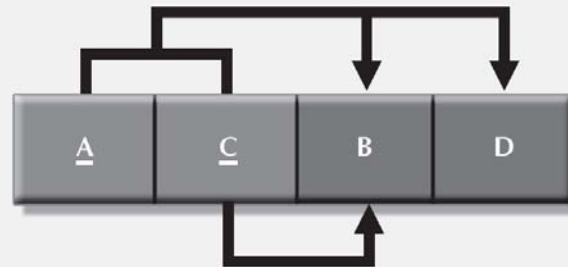
FIGURA 5.8

Decomposição para BCNF

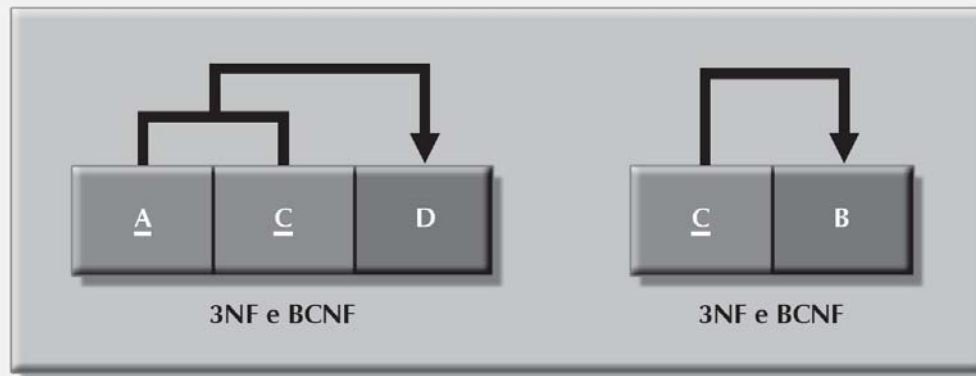
3NF, mas não BCNF



1NF



Dependência parcial



3NF e BCNF

3NF e BCNF

# Quarta Forma Normal (4NF )

- A tabela está na **quarta forma normal (4NF)** se estiver em 3NF e não contiver conjuntos múltiplos de dependências com vários valores
- A discussão sobre 4NF é de essência acadêmica se o projetista se assegurar de que suas tabelas estejam em conformidade com as duas regras seguintes:
  - Todos os atributos devem ser dependentes da chave primária, mas independentes de todos os outros
  - Nenhuma linha pode conter dois ou mais fatos com vários valores sobre uma entidade

**FIGURA  
5.10**

**Tabelas com dependências com vários valores**

Nome do banco de dados: Ch05\_Service

Nome da tabela: VOLUNTÁRIO\_V1

EMP_NUM	ORG_CODE	ASSIGN_NUM
10123	RC	1
10123	LWV	3
10123		4

Nome da tabela: VOLUNTÁRIO\_V3

EMP_NUM	ORG_CODE	ASSIGN_NUM
10123	RC	1
10123	RC	3
10123	LWV	4

Nome da tabela: VOLUNTÁRIO\_V2

EMP_NUM	ORG_CODE	ASSIGN_NUM
10123	RC	
10123	LWV	
10123		1
10123		3
10223		4

**FIGURA 5.11**

**Conjunto de tabelas em 4NF**

Nome do banco de dados: Ch05\_Service

Nome da tabela: FUNCIONÁRIO

EMP_NUM	EMP_LNAME
10121	Rogers
10122	O'Leery
10123	Panera
10124	Johnson

Nome da tabela: PROJETO

PROJ_CODE	PROJ_NAME	PROJ_BUDGET
1	BeThere	1023245.00
2	BlueMoon	20198608.00
3	GreenThumb	3234456.00
4	GoFast	5674000.00
5	GoSlow	1002500.00

Nome da tabela: ORGANIZAÇÃO

ORG_CODE	ORG_NAME
RC	Red Cross
UWV	United Way
WF	Wildlife Fund

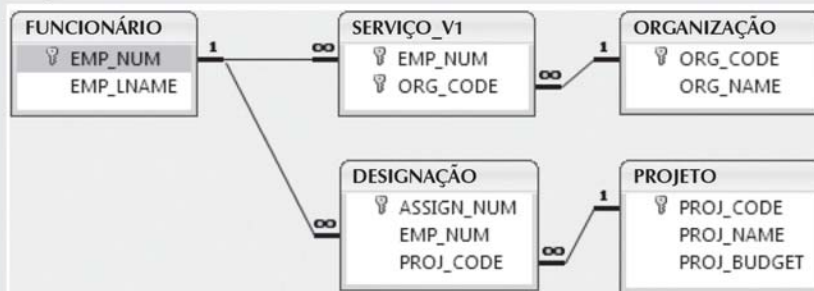
Nome da tabela: DESIGNAÇÃO

ASSIGN_NUM	EMP_NUM	PROJ_CODE
1	10123	1
2	10121	2
3	10123	3
4	10123	4
5	10121	1
6	10124	2
7	10124	3
8	10124	5

Nome da tabela: SERVIÇO\_V1

EMP_NUM	ORG_CODE
10123	RC
10123	UWV
10123	WF

Diagrama relacional





# Normalização e Projeto do Banco de Dados

- A normalização deve fazer parte do processo do projeto
- Certifique-se de que as entidades propostas atendam à forma normal necessária *antes* que as estruturas de tabela sejam criadas
- Muitos dos bancos de dados reais com os quais você vai se deparar foram projetados de forma inadequada, ou carregados de anomalias
- Pode ocorrer que ao solicitarem um novo projeto ou a modificação de bancos de dados existentes

# Normalização e Projeto do Banco de Dados (cont.)

- Deve-se criar um DER por meio de um processo iterativo
  - Identifique as entidades relevantes, seus atributos e relacionamentos
  - Utilize os resultados para identificar entidades e atributos adicionais
- Procedimentos de normalização
  - Foca as características de entidades específicas (representa uma visualização micro das entidades do DER)
- É difícil separar o processo de normalização do de modelagem ER

# FIGURA 5.12

## DER inicial da empresa contratante

FUNCIONÁRIO	
PK	<u>EMP_NUM</u>
	EMP_LNAME EMP_FNAME EMP_INITIAL JOB_DESCRIPTION JOB_CHG_HOUR

PROJETO	
PK	<u>PROJ_NUM</u>
	PROJ_NAME

**FIGURA  
5.13**

**DER modificado da empresa contratante**

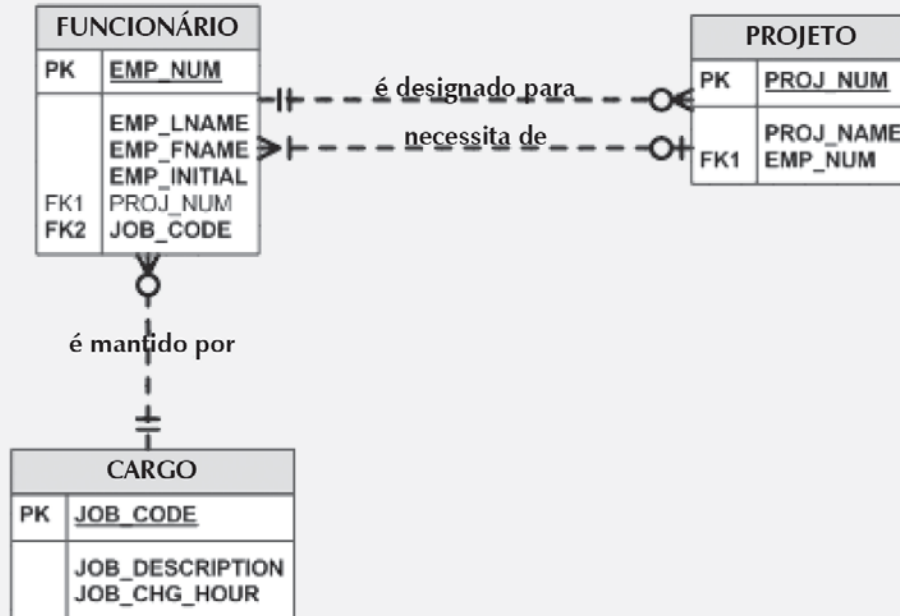


Cada FUNCIONÁRIO possui uma classificação (principal) de CARGO .  
Qualquer classificação de CARGO pode ser mantida por vários FUNCIONÁRIOS.

Algumas classificações de CARGO ainda não receberam nenhuma atribuição de pessoal.  
Portanto, FUNCIONÁRIO é opcional para CARGO.

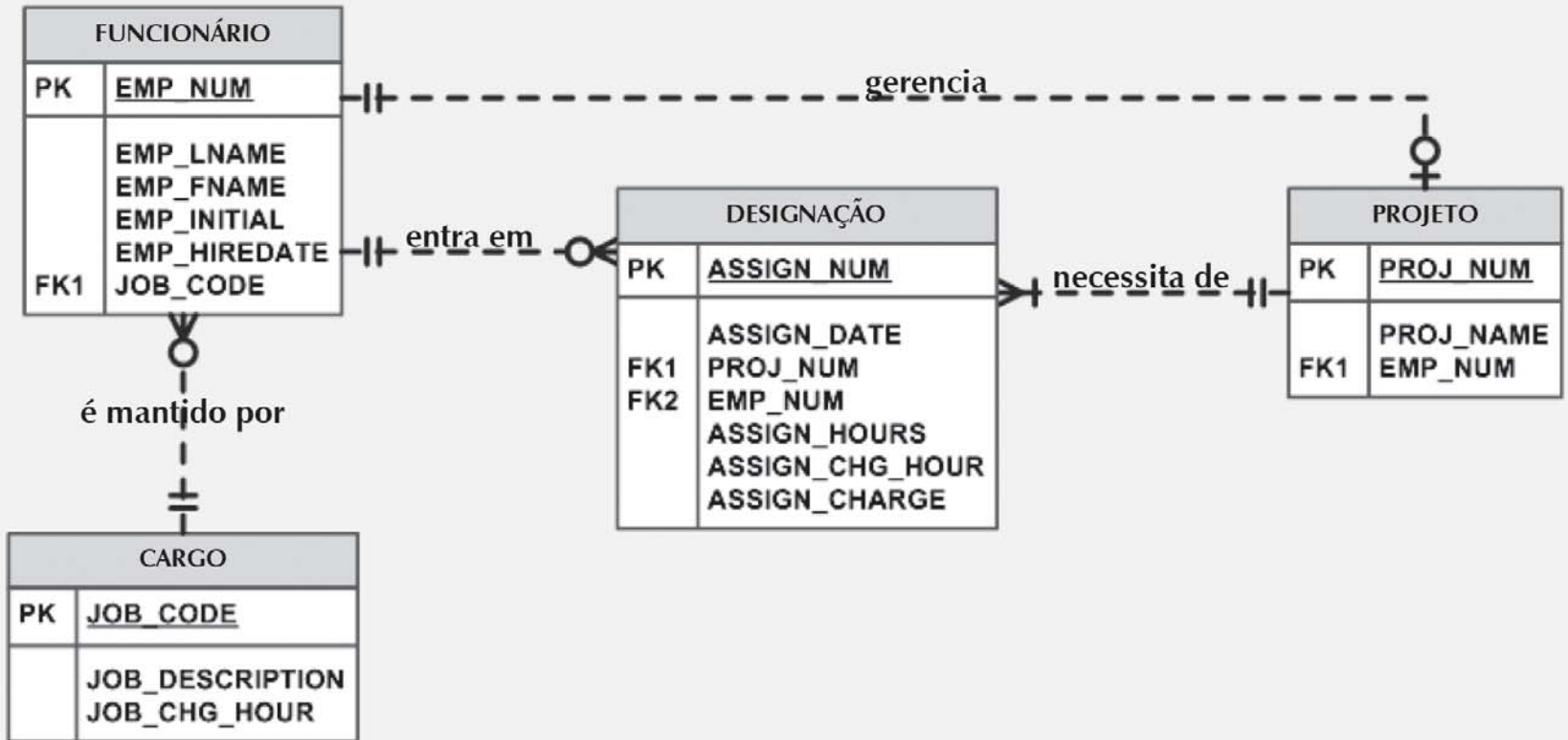
**FIGURA 5.14**

**Representação de um relacionamento M:N incorreto**



**FIGURA 5.15**

DER final da empresa contratante



**FIGURA 5.16** Banco de dados implementado

Nome da tabela: FUNCIONÁRIO

EMP_NUM	EMP_LNAME	EMP_FNAME	EMP_INITIAL	EMP_HIREDATE	JOB_CODE
101	News	John	G	08-Nov-00	502
102	Senior	David	H	12-Jul-89	501
103	Arbough	June	E	01-Dec-97	503
104	Ramoras	Anne	K	15-Nov-88	501
105	Johnson	Alice	K	01-Feb-94	502
106	Smithfield	vWilliam		22-Jun-05	500
107	Alonzo	Maria	D	10-Oct-94	500
108	vWashington	Ralph	B	22-Aug-89	501
109	Smith	Larry	vV	18-Jul-99	501
110	Olenko	Gerald	A	11-Dec-96	505
111	vVabash	Geoff	B	04-Apr-89	506
112	Smithson	Darlene	M	23-Oct-95	507
113	Joebrood	Delbert	K	15-Nov-94	508
114	Jones	Annelise		20-Aug-91	508
115	Bawangi	Travis	B	25-Jan-90	501
116	Pratt	Gerald	L	05-Mar-95	510
117	vWilliamson	Angle	H	19-Jun-94	509
118	Frommer	James	J	04-Jan-06	510

Nome da tabela: CARGO

JOB_CODE	JOB_DESCRIPTION	JOB_CHG_HOUR
500	Programmer	35.75
501	Systems Analyst	96.75
502	Database Designer	105.00
503	Electrical Engineer	84.50
504	Mechanical Engineer	67.90
505	Civil Engineer	55.78
506	Clerical Support	26.87
507	DSS Analyst	45.95
508	Applications Designer	48.10
509	Bio Technician	34.55
510	General Support	18.36

Nome da tabela: PROJETO

PROJ_NUM	PROJ_NAME	EMP_NUM
15	Evergreen	105
18	Amber Wave	104
22	Rolling Tide	113
25	Starflight	101

Nome da tabela: DESIGNAÇÃO

ASSIGN_NUM	ASSIGN_DATE	PROJ_NUM	EMP_NUM	ASSIGN_HOURS	ASSIGN_CHG_HOUR	ASSIGN_CHARGE
1001	04-Mar-08	15	103	2.6	84.50	219.70
1002	04-Mar-08	18	118	1.4	18.36	25.70
1003	05-Mar-08	15	101	3.6	105.00	378.00
1004	05-Mar-08	22	113	2.5	48.10	120.25
1005	05-Mar-08	15	103	1.9	84.50	160.55
1006	05-Mar-08	25	115	4.2	96.75	406.35
1007	05-Mar-08	22	105	5.2	105.00	546.00
1008	05-Mar-08	25	101	1.7	105.00	178.50
1009	05-Mar-08	15	105	2.0	105.00	210.00
1010	06-Mar-08	15	102	3.8	96.75	367.65
1011	06-Mar-08	22	104	2.6	96.75	251.55
1012	06-Mar-08	15	101	2.3	105.00	241.50
1013	06-Mar-08	25	114	1.8	48.10	86.58
1014	06-Mar-08	22	111	4.0	26.87	107.48
1015	06-Mar-08	25	114	3.4	48.10	163.54
1016	06-Mar-08	18	112	1.2	45.95	55.14
1017	06-Mar-08	18	118	2.0	18.36	36.72
1018	06-Mar-08	18	104	2.6	96.75	251.55
1019	06-Mar-08	15	103	3.0	84.50	253.50
1020	07-Mar-08	22	105	2.7	105.00	283.50
1021	08-Mar-08	25	108	4.2	96.75	406.35
1022	07-Mar-08	25	114	5.8	48.10	278.98
1023	07-Mar-08	22	106	2.4	35.75	85.80

# Desnormalização

- Embora a criação dessas relações seja um objetivo importante do projeto de banco de dados, trata-se de apenas um entre vários objetivos importantes.
- O bom projeto de bancos de dados também leva em consideração as necessidades e a velocidade de processamento (e de relatório)
- O problema da normalização é que, conforme as tabelas sejam decompostas para atender a suas exigências:
  - O número de tabelas no banco de dados se amplia



## Desnormalização (cont.)

- A junção de grande número de tabelas reduz a velocidade do sistema.
- Os conflitos entre eficiência de projeto, necessidades de informações e desempenho costumam ser resolvidos por meio de comprometerimentos que incluam desnormalização

# Desnormalização (cont.)

- As tabelas não normalizadas tendem a apresentar as seguintes deficiências:
  - As atualizações de dados são menos eficientes, por ter de lidar com tabelas maiores
  - A indexação é mais trabalhosa
  - Tabelas não normalizadas não resultam em estratégias simples para a criação de tabelas virtuais conhecidas como *visualizações*

# Resumo

- A normalização é uma técnica utilizada para projetar tabelas em que as redundâncias de dados sejam minimizadas
- As três primeiras formas normais (1NF, 2NF e 3NF) são encontradas com mais frequência
- Uma tabela está em 1NF quando:
  - Todos os seus atributos de chave são definidos
  - Todos os atributos restantes são dependentes da chave primária

# Resumo (cont.)

- Uma tabela está em 2NF quando está em 1NF e não contém dependências parciais
- Uma tabela está em 3NF quando está em 2NF e não contém dependências transitivas
- Uma tabela que não esteja em 3NF pode ser separada em novas tabelas até que todas atendam às exigências dessa forma normal
- A normalização é uma parte importante – mas apenas uma parte – do processo de projeto

## Resumo (cont.)

- Uma tabela em 3NF pode conter dependências com vários valores que produzam muitos valores nulos ou dados redundantes
  - Pode ser necessário converter uma tabela 3NF para a quarta forma normal (4NF), separando a tabela para remover tais dependências
- Às vezes as tabelas são desnormalizadas, produzindo menos E/S e aumentando a velocidade de processamento