

## Normalização

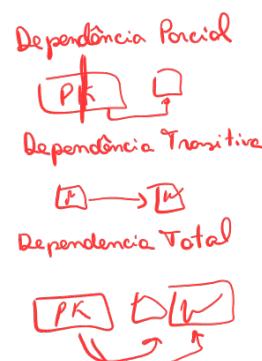
Base de Dados - 2022/23  
Carlos Costa

1

1

## Introdução

- Já estudámos aspectos de desenho conceptual de base de dados e respectivo mapeamento para o modelo relacional.
- No entanto, nunca apresentámos um processo formal de analisar se determinado grupo de atributos de um esquema de relação é melhor do que outro.
- O desenho de uma base de dados relacional resulta num conjunto de relações. Existe um objectivo implícito nesse processo de desenho:
  - Preservação da informação
    - Todos os conceitos capturados pelo desenho conceptual que são mais tarde mapeados para o desenho lógico.
  - Minimizar a redundância dos dados
    - Minimizar o armazenamento duplicado de dados em relações distintas, reduzindo a necessidade de múltiplos updates e consequente problema 2 de consistência entre múltiplas cópias da mesma informação.



2

## Desenho de BD - Esquemas de Relação

### Análise de Qualidade:

- Critérios Informais
- Critérios Formais (matemáticos)
  - Dependências Funcionais, Multivalue e Junção
- Processo de Normalização (classe A++, A, B, C, ...)
  - Formas Normais
    - Baseadas em critérios formais

3

3

## Critérios Informais

- Clareza da semântica dos atributos da relação
- Redundância de informação no tuplo
- Redução dos NULLs nos tuplos
- Junção de relações baseada em PK e FK

4

4

## Semântica dos atributos da relação

- O desenho de um esquema de relação deve ser fácil de explicar.
- Verificar se existe uma semântica clara entre os atributos de uma relação.
  - Evitar que uma relação corresponda a uma mistura de atributos de diferentes entidades e relacionamentos.
  - Exemplos de mau desenho:

EMP\_DEPT

Ename	Ssn	Bdate	Address	Dnumber	Dname	Dmgr_ssn

EMP\_PROJ

Ssn	Pnumber	Hours	Ename	Pname	Plocation

5

5

## Redundância de Informação no Tuplo

- O objectivo é reduzir ao máximo o espaço ocupado por uma relação.
- No mau exemplo anterior verificámos que também há **duplicação desnecessária de informação**.

Duplicação dos dados do departamento sempre que introduzimos um novo funcionário

Update de dados departamento...  
... Update todos os tuplos!!!

EMP\_DEPT

Ename	Ssn	Bdate	Address	Dnumber	Dname	Dmgr_ssn
Smith, John B.	123456789	1965-01-09	731 Fondren, Houston, TX	5	Research	333445555
Wong, Franklin T.	333445555	1965-12-08	638 Voss, Houston, TX	5	Research	333445555
Zelaya, Alicia J.	999887777	1968-07-19	3321 Castle, Spring, TX	4	Administration	987654321
Wallace, Jennifer S.	987654321	1941-06-20	291 Berry, Bellaire, TX	4	Administration	987654321
Narayan, Ramesh K.	666884444	1962-09-15	975 FireOak, Humble, TX	5	Research	333445555
English, Joyce A.	453453453	1972-07-31	5631 Rice, Houston, TX	5	Research	333445555
Jabbar, Ahmad V.	987987987	1969-03-29	980 Dallas, Houston, TX	4	Administration	987654321
Borg, James E.	8888665555	1937-11-10	450 Stone, Houston, TX	1	Headquarters	8888665555

Como introduzir um funcionário sem departamento?

Uso de NULLs...

Como introduzir um novo departamento?

Uso de NULLs...

Ssn=NULL  
!!!Integridade da Entidade???

EMPLOYEE

Ename	Ssn	Bdate	Address	Dnumber
Smith, John B.	123456789	1965-01-09	731 Fondren, Houston, TX	5
Wong, Franklin T.	333445555	1965-12-08	638 Voss, Houston, TX	5
Zelaya, Alicia J.	999887777	1968-07-19	3321 Castle, Spring, TX	4
Wallace, Jennifer S.	987654321	1941-06-20	291 Berry, Bellaire, TX	4
Narayan, Ramesh K.	666884444	1962-09-15	975 FireOak, Humble, TX	5
English, Joyce A.	453453453	1972-07-31	5631 Rice, Houston, TX	5
Jabbar, Ahmad V.	987987987	1969-03-29	980 Dallas, Houston, TX	4
Borg, James E.	8888665555	1937-11-10	450 Stone, Houston, TX	1

Transformar em 1 coluna!  
chave estrutural  
Nova entidade!

DEPARTMENT

Dname	Dnumber	Dmgr_ssn
Research	5	333445555
Administration	4	987654321
Headquarters	1	8888665555

6

6

**Redução dos NULLs nos tuplos**

*Em relações 1:1  
não importavam  
para a obrigação!  
(queremos reduzir  
os NULLs)*

- Há situações em que temos uma grande quantidade de atributos numa relação:
  - Muitos dos atributos não se aplicam a todos os tuplos da relação.
- Consequência: existência de muitos NULLs nesses tuplos
  - Desperdício de espaço
  - Difícil interpretação do seu sentido desses atributos (Null pode ter vários significados)
- Recomendação: Criar outra relação para esses atributos.

Exemplo:

- Imaginando que queremos incluir o número do gabinete na relação Employee mas só 15% dos funcionários têm esse número.
- Solução: criar uma nova relação **EMP\_OFFICES(Essn, Office\_number)** só com tuplos de funcionários com gabinete.

7

7

**Junção de Relações baseada em PK e FK**

*Cuidado com a junção  
em casos que não estão  
ligados ...*

- Devemos evitar esquemas de relação que estabeleçam relacionamentos entre duas relações baseados em atributos que não a chave primária e estrangeira.
- Mau exemplo:

**EMPL\_LOCS  $\bowtie$  EMP\_PROJ1**

Ssn	Pnumber	Hours	Pname	Plocation	Ename	
123456789	1	32.5	ProductX	Bellaire	Smith, John B.	
*	123456789	1	32.5	ProductX	Bellaire	English, Joyce A.
*	123456789	2	7.5	ProductY	Sugarland	Smith, John B.
*	123456789	2	7.5	ProductY	Sugarland	English, Joyce A.
*	123456789	2	7.5	ProductY	Sugarland	Wong, Franklin T.
666884444	3	40.0	ProductZ	Houston	Narayan, Ramesh K.	
*	666884444	3	40.0	ProductZ	Houston	Wong, Franklin T.
*	453453453	1	20.0	ProductX	Bellaire	Smith, John B.
453453453	1	20.0	ProductX	Bellaire	English, Joyce A.	
*	453453453	2	20.0	ProductY	Sugarland	Smith, John B.
453453453	2	20.0	ProductY	Sugarland	English, Joyce A.	

Ssn	Pnumber	Hours	Pname	Plocation
123456789	1	32.5	ProductX	Bellaire
123456789	2	7.5	ProductY	Sugarland
666884444	3	40.0	ProductZ	Houston
453453453	1	20.0	ProductX	Bellaire
453453453	2	20.0	ProductY	Sugarland
333445555	2	10.0	ProductY	Sugarland
333445555	3	10.0	ProductZ	Houston
333445555	10	10.0	Computerization	Stafford
999999999	99	10.0	Renovations	Houston

**Temos situações de junção errada de tuplos:  
\* spurious tuples**

8

8

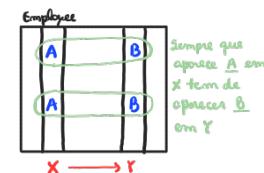
## Dependências Funcionais

9

9

## Dependências Funcionais (DP)

- Considerando a relação:
  - $R(A_1, A_2, \dots, A_n)$
  - Subconjunto de atributos  $X, Y \subseteq R$
- Dependência Funcional:  $\overbrace{X}^{\text{Chave primária}} \rightarrow \overbrace{Y}^{\text{Todos os atributos}}$  Y depende de X
  - tuplos:  $t_1, t_2 \in R$
  - $t_1[X] = t_2[X] \Rightarrow t_1[Y] = t_2[Y]$  Restrição
- Formalismo de análise de esquemas relacionais.
  - Permite descrever restrições dos atributos que os tuplos devem respeitar em todo o momento (invariantes).
  - Permite detectar e descrever problemas com precisão.



10

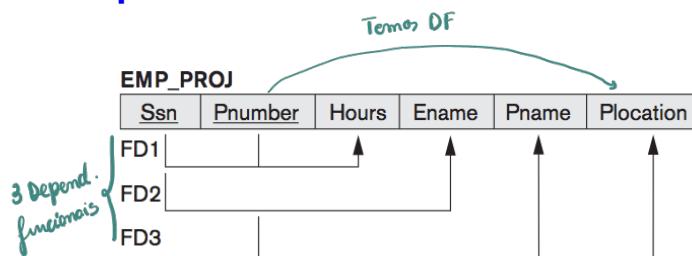
## Dependências Funcionais

- $X \rightarrow Y$  ... por outras palavras:
  - $Y$  é funcionalmente dependente de  $X$ .
  - Os valores da componente  $X$  do tuplo define de forma única a componente  $Y$  do respectivo tuplo.
- Uma DP é uma propriedade do esquema de relação  $R$  que não pode ser inferido de uma qualquer instância de  $R$ , i.e.  $r(R)$ .
  - Deve ser definida por alguém que conhece a semântica dos atributos da relação.

11

11

## Dependências Funcionais - Exemplo



- Pela semântica dos atributos da relação **EMP\_PROJ** podemos inferir as seguintes **DF**:
  - $Ssn \rightarrow Ename$
  - $Pnumber \rightarrow \{Pname, Plocation\}$
  - $\{Ssn, Pnumber\} \rightarrow Hours$

O  $Ssn$  determina de forma única o nome do funcionário.  
 O número do projecto determina de forma única o seu nome e localização.  
 O  $Ssn$  e o número do projecto determinam de forma única o número de horas que um funcionário trabalha para o projecto.

FD: Functional Dependency

12

 deti

## Tipos de Dependências Funcionais

- **Dependência Parcial**
  - atributo depende de parte dos atributos que compõem a chave da relação.

↳ Primary Key  
↳ depende de parte das chaves  
Pnumber → Placeion
- **Dependência Transitiva**
  - atributo que não faz parte da chave da relação depende de um atributo que também não faz parte da chave da relação.

↳ chave  
 $X \rightarrow Y$   
 $Y \rightarrow Z$  Transitive  
↳ Nem um, nem outro!  
↳ Nonhun fog parte da chave da relação

↳ Ambos, X e Y  
não fazem parte da chave primária  
↳ um comum...
- **Dependência Total**
  - atributo depende de toda a chave da relação.

↳ depende de TODA a chave!  
↳ O melhor... 14

14

 deti

## Normalização

15

15

## Introdução

- Objectivo: Reducir a Redundância
- DF especifica alguns aspectos semânticos do esquema da relação.
  - ...mas a redundância está associada a DF não desejadas!
- Vamos assumir que:
  - Existe um conjunto de DF associadas a cada esquema de relação;
  - Que cada relação tem uma chave primária definida;
- Processo de Normalização:
  - Formas Normais
    - Conjunto de testes (condições) para validação de cada forma.
    - Cada forma superior tem menos DF que a anterior.

16

16

## Formas Normais

Conjunto de testes para ver se um modelo do BD cumpre um determinado nível!

- O processo de normalização consiste em efetuar um conjunto de testes para certificar se um desenho de BD relacional satisfaz determinada Forma Normal (FN).
  - Relações que não satisfazem os testes de determinada forma normal são decompostas em relações menores.
- Codd propôs três FN baseadas em DF
  - Primeira (1FN), Segunda (2FN) e Terceira (3FN)
  - A 3FN satisfaz as condições da 2FN e esta as da 1FN
- Mais tarde Boyce e Codd propuseram uma definição mais restritiva da 3NF à qual se chamou:
  - Boyce-Codd Normal Form (BCNF)
- Foram ainda propostas a 4FN e 5FN baseadas respectivamente em dependências multivalor e de junção.

17

17

1FN: mínimo...  
 2FN: não parciais e 1FN  
 3FN: não transitivos e 2FN  
 BCNF: todos são totais e 3FN  
 4FN: não há dep. multivalor e BCNF  
 5FN: máximo e não há dep. de junção e 4FN

## Primeira Forma Normal (1NF)

→ O desenho está bom ...

- Definição formal de uma relação básica do modelo relacional:
  - Atributos são atómicos (simples e indivisíveis)
    - Não permite atributos composto ou multivalor
  - Não suporta relações dentro de relações (Nested Relation)
    - Não é possível utilizar uma relação como valor de um atributo de um tuplo.
    - Violia a atomicidade

EMP_PROJ		Projs	
Ssn	Ename	Pnumber	Hours
123456789	Smith, John B.	1 2	32.5 7.5
666884444	Narayan, Ramesh K.	3	40.0
453453453	English, Joyce A.	1 2	20.0 20.0

ERROS!

Nested Relation

Nested Relation

Restrições de integridade:

- Domínio
- Entidade
- Referencial

18

{} significa que o atributo PROJS é multivalor

↑  
Não está na 1FN

EMP\_PROJ(Ssn, Ename, {PROJS(Pnumber, Hours)})

## 1FN - Exemplo 1

(a) DEPARTMENT

Dname	Dnumber	Dmgr_ssn	Dlocations
Research	5	333445555	{Bellaire, Sugarland, Houston}
Administration	4	987654321	{Stafford}
Headquarters	1	888665555	{Houston}

Esquema Relação

Não está na 1FN

Instância

(b) DEPARTMENT

Dname	Dnumber	Dmgr_ssn	Dlocations
Research	5	333445555	{Bellaire, Sugarland, Houston}
Administration	4	987654321	{Stafford}
Headquarters	1	888665555	{Houston}

- Dlocation - atributo multivalor!
- 3 aproximações para converter a relação na 1FN...

19

**1FN - Exemplo 1 (soluções)**

DEPARTMENT			DEPT_LOCATIONS	
Dname	Dnumber	Dmgr_ssn	Dnumber	Dlocation
Research	5	333445555	1	Houston
Administration	4	987654321	4	Stafford
Headquarters	1	888665555	5	Bellaire
			5	Sugarland
			5	Houston

Aproximação 1

**Melhor solução**

Decomposição da Relação

Péssima!

Aproximação 2

Está na 1FN mas...

Problema: Redundância

Dname	Dnumber	Dmgr_ssn	Dlocation
Research	5	333445555	Bellaire
Research	5	333445555	Sugarland
Research	5	333445555	Houston
Administration	4	987654321	Stafford
Headquarters	1	888665555	Houston

Aproximação 3

"Solução de artista..."

Se K é o nº máximo de Dlocation,

Criar K atributos distintos (Dlocation1, Dlocation2,.., DlocationK)

Problemas: NULL values, consultas Dlocation difíceis, ...

Aproximação 2

Está na 1FN mas...

Problema: Redundância

20

20

**1FN - Exemplo 2 (Nested Relation)**

EMP_PROJ			
Ssn	Ename	Pnumber	Hours
Projs			

→ Não está na 1FN

- Ssn é chave primária da relação EMP\_PROJ
- Pnumber é chave parcial da Nested Relation (Projs)
- Solução:
  - Decompor a relação em duas relações na 1FN:

EMP_PROJ1		EMP_PROJ2	
Ssn	Ename	Ssn	Pnumber
↓		↓	

↗ Decompor em 2 relações na 1FN

21

21

**Segunda Forma Normal (2FN)**

- A relação está na 1FN e...
- ...todos os atributos não pertencentes a qualquer chave candidata devem depender totalmente da chave e não de parte dela.
  - i.e. não existem dependências parciais

Exemplo:

- está na 1FN
- dependência total:
  - FD1 ( $\{Ssn, Pnumber\} \rightarrow Hours$ )
- Problema de dependências parciais:
  - FD2 ( $Ssn \rightarrow Ename$ )
  - FD3 ( $Pnumber \rightarrow \{Pname, Plocation\}$ )

*8 Regra*

*Dependências Parciais!*

*{ dependências parciais! }*

Solução: Decompor a relação...

22

**2FN - Exemplo**

**EMP\_PROJ**

Ssn	Pnumber	Hours	Ename	Pname	Plocation
FD1					
FD2					
FD3					

*Remova estes!*

**2NF Normalization**

**EP1**

Ssn	Pnumber	Hours
FD1		

**EP2**

Ssn	Ename
FD2	

**EP3**

Pnumber	Pname	Plocation
FD3		

*Todos estão na 2FN*

- Todas as dependência parciais deram resultado a uma nova relação.
- Verificar se as novas relações só têm dependências totais.

23

**Terceira Forma Normal (3FN)**

- A relação está na 2FN e... *→ Não esquecer*
- ...não existem dependências funcionais entre atributos não chave.
- i.e. não existem dependências transitivas *!*

Exemplo:

- está na 2FN
- Problema de dependências transitivas:**  
 $Ssn \rightarrow Dnumber$  e  
 $Dnumber \rightarrow Dname$   
 $Dnumber \rightarrow Dmgr\_ssn$

Solução: Decompor a relação...

24

24

**3FN - Exemplo**

**EMP\_DEPT**

Ename	Ssn	Bdate	Address	Dnumber	Dname	Dmgr_ss
-------	-----	-------	---------	---------	-------	---------

↑      ↑      ↑      ↑      ↑      ↑      ↑

**3NF Normalization**

Mantém-se como chave estrangeira

**ED1**

Ename	Ssn	Bdate	Address	Dnumber
-------	-----	-------	---------	---------

↑      ↑      ↑      ↑      ↑

**ED2**

Dnumber	Dname	Dmgr_ss
---------	-------	---------

↑      ↑      ↑

As dependências transitivas relativamente a Dnumber deu origem a uma nova relação (ED2) em que Dnumber é a sua chave primária.

Dnumber mantém-se na relação inicial como chave estrangeira.

25

25

## Quadro Resumo: 1FN, 2FN e 3FN

Summary of Normal Forms Based on Primary Keys and Corresponding Normalization

Normal Form	Test	Remedy (Normalization)
First (1NF)	Relation should have no multivalued attributes or nested relations.	Form new relations for each multivalued attribute or nested relation.
Second (2NF)	For relations where primary key contains multiple attributes, no nonkey attribute should be functionally dependent on a part of the primary key.	Decompose and set up a new relation for each partial key with its dependent attribute(s). Make sure to keep a relation with the original primary key and any attributes that are fully functionally dependent on it.
Third (3NF)	Relation should not have a nonkey attribute functionally determined by another nonkey attribute (or by a set of nonkey attributes). That is, there should be no transitive dependency of a nonkey attribute on the primary key.	Decompose and set up a relation that includes the nonkey attribute(s) that functionally determine(s) other nonkey attribute(s).

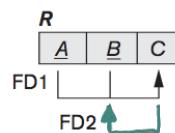
26

26

## Boyce-Codd Normal Form (BCNF)

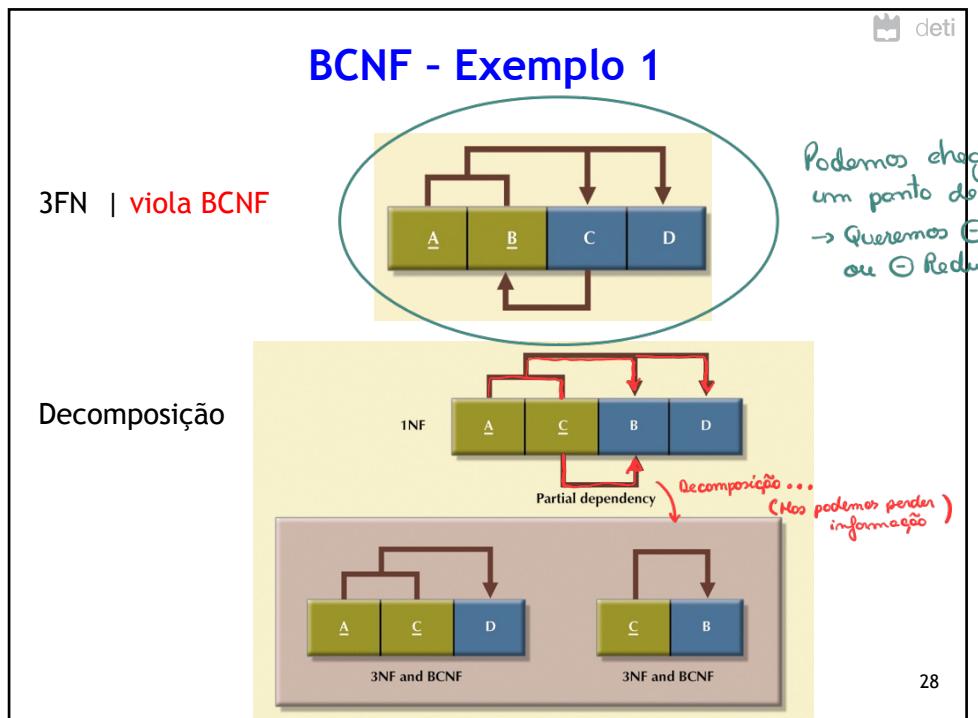
- Usualmente, a 3FN é aquela que termina o processo de normalização.
  - No entanto, em algumas situações a 3FN ainda apresenta algumas anomalias.
- BCNF é mais restritiva que a 3FN
  - BCNF => 3FN
- Definição:
 

Todos os atributos são funcionalmente dependentes da chave da relação, de toda a chave e de nada mais.
- Exemplo:
  - está na 3FN
  - FD2 viola a BCNF

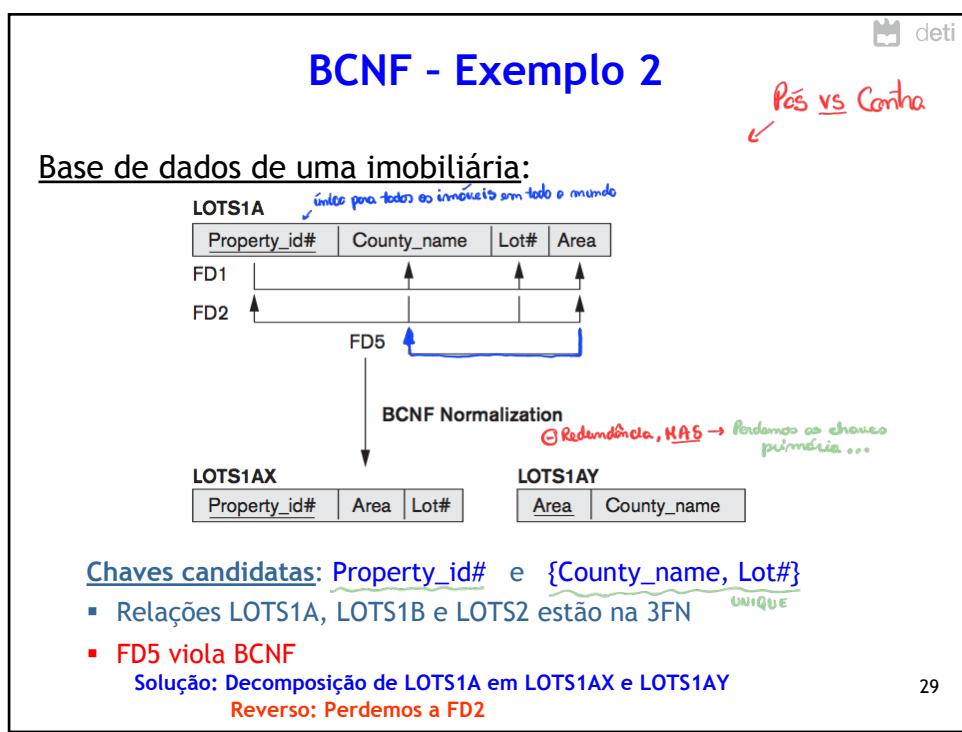


27

27



28



29

**Normalização - Ponto de Equilíbrio**

- Como verificámos no exemplo de BCNF, perdeu-se uma dependência funcional importante (deduzida da semântica dos atributos).
  - Que deverá ser tratada ao nível aplicacional.
- Assim, existe um ponto de equilíbrio no processo de Normalização que tipicamente fica entre a 3FN e a BCNF.

30

30

*São mais raros ...*

**4FN e 5FN**

- Usualmente uma relação na BCNF também se encontra na 4FN e 5FN.
  - 4FN são raros e 5FN ainda mais raros
- Definição 4FN:
  - Está na BCNF
  - Não existem dependências multivalor
- Definição 5FN:
  - Está na 4FN
  - A relação não pode ser mais decomposta sem haver *perda de informação*
  - Não existem dependências de junção

31

31

**Dependências Multivalor**

- Dependência multivalor  $X \twoheadrightarrow Y$  em  $R(X, Y, Z)$
- Garantir a seguinte restrição em qualquer instância  $r(R)$ :
  - Se dois tuplos  $t_1$  e  $t_2$  existem em  $r(R)$  tal que  $t_1[X] = t_2[X]$
  - Então também devem existir dois tuplos  $t_3$  e  $t_4$  em  $r(R)$  com as seguintes características:
    - $t_4[X] = t_3[X] = t_1[X] = t_2[X]$
    - $t_3[Y] = t_1[Y]$  e  $t_4[Y] = t_2[Y]$
    - $t_3[Z] = t_2[Z]$  e  $t_4[Z] = t_1[Z]$
- Exemplo:
  - $X \twoheadrightarrow Y$
  - $X \twoheadrightarrow Z$
- Outras palavras...
 

$X$  multidetermina  $Y$  se, para cada par de tuplos de  $R$  contendo os mesmo valores de  $X$ , existe em  $R$  um par de tuplos correspondentes à troca dos valores de  $Y$  no par original.

32

**4FN: Dependências Multivalor - Exemplo**

**EMP**

Ename	Pname	Dname
Smith	X	John
Smith	Y	Anna
Smith	X	Anna
Smith	Y	John

*Têm de ser verificados  
mos esses de uso ...*

*troca em Z!*

**Dependências Multivalor:**

- Ename  $\twoheadrightarrow$  Pname
- Ename  $\twoheadrightarrow$  Dname

**Solução: decomposição da relação EMP**

*Resolvemos a  
dependência multivalor!*

EMP_PROJECTS	
Ename	Pname
Smith	X
Smith	Y

EMP_DEPENDENTS	
Ename	Dname
Smith	John
Smith	Anna

33

33

## Dependências de Junção

 deti

- Existe uma dependência de junção em R se, dadas algumas projeções de R, apenas se reconstrói R através de algumas junções bem definidas, mas não de todas.
- Muito rara na prática
  - difícil de detectar
- Exemplo: *cada um podia ter vários atributos...*
  - Projetando R em  $(X, Y)$ ,  $(X, Z)$  e  $(Y, Z)$
  - Verificamos que não é possível reconstruir R por junção de qualquer uma das projeções.
  - Só com a junção das 3 projeções é que conseguimos reconstruir R.

$r(R)$		
X	Y	Z
x1	y1	z1
x1	y1	z2
x1	y2	z2
x2	y3	z2
x2	y4	z2
x2	y4	z4
x2	y5	z4
x3	y2	z5

34

*Temos repetidos em XY ou XZ ou YZ*

34

*Só conseguimos reparar a tabela inicial com o junção dos 3 colunas // só operam de 2*

## 5FN: Dependência Junção - Exemplo

 deti

**SUPPLY**

Sname	Part_name	Proj_name
Smith	Bolt	ProjX
Smith	Nut	ProjY
Adamsky	Bolt	ProjY
Walton	Nut	ProjZ
Adamsky	Nail	ProjX
Adamsky	Bolt	ProjX
Smith	Bolt	ProjY

Vamos Criar 3 Projeções de Supply:

- R1(Sname, Part\_name)
- R2(Sname, Proj\_name)
- R3(Part\_name, Proj\_name)

$R_1$	
Sname	Part_name
Smith	Bolt
Smith	Nut
Adamsky	Bolt
Walton	Nut
Adamsky	Nail

$R_2$	
Sname	Proj_name
Smith	ProjX
Smith	ProjY
Adamsky	ProjY
Walton	ProjZ
Adamsky	ProjX

$R_3$	
Part_name	Proj_name
Bolt	ProjX
Nut	ProjY
Bolt	ProjY
Nut	ProjZ
Nail	ProjX

- A relação SUPPLY, com dependência de junção, pode ser decomposta em 3 relações R1, R2 e R3 cada uma na 5FN.
  - Só reconstruímos Supply com a junção das 3 relações R1, R2 e R3.

35

35

deti

## Normalização - Caso de Estudo

### Gestão de Encomendas

36

36

deti

## Esquema de Base de Dados - Início

<b>Encomendas</b>				
<u>num_encomenda</u>	<u>num_cliente</u>	<u>cliente</u>	<u>endereco_cliente</u>	...
...				
<u>data_encomenda</u>	<u>cod_produto</u>	<u>produto</u>	<u>quantidade_prod</u>	

- É notório que o designer não tem conhecimentos de desenho de base de dados...
- Problemas:
  - Mistura de grupos de atributos de entidades (claramente) distintas.
  - Redundância de informação nos tuplos
    - Temos de repetir num\_encomenda, num\_cliente, cliente, endereco\_cliente e data\_encomenda para registar várias linhas de uma encomenda!

37

37

## 1FN

### **Encomendas**

num_encomenda	num_cliente	cliente	endereco_cliente	...
---------------	-------------	---------	------------------	-----

...

data_encomenda	<b>cod_produto</b>	produto	quantidade_prod
----------------	--------------------	---------	-----------------

- Podemos dizer que existe uma segunda relação Linhas\_Encomenda(cod\_produto, produto, quantidade\_prod) na relação Encomendas.
- Por decomposição:

### **Encomendas**

<u>num_encomenda</u>	num_cliente	cliente	endereco_cliente	<u>data_encomenda</u>
----------------------	-------------	---------	------------------	-----------------------

### **Linhos\_Encomenda**

<u>num_encomenda</u>	<u>cod_produto</u>	produto	quantidade_prod
----------------------	--------------------	---------	-----------------

38

38

## 2FN

### **Encomendas**

<u>num_encomenda</u>	num_cliente	cliente	endereco_cliente	<u>data_encomenda</u>
----------------------	-------------	---------	------------------	-----------------------

↑ Dependência transitiva

### **Linhos\_Encomenda**

<u>num_encomenda</u>	<u>cod_produto</u>	<b>produto</b>	quantidade_prod
----------------------	--------------------	----------------	-----------------

↑ Dependência Parcial!

- Verificámos que na segunda relação há uma violação à 2FN:
  - Dep. Parcial: produto só depende de um atributo (cod\_produto) da chave da relação!
- Por decomposição da relação Linhos\_Encomenda:

### **Linhos\_Encomenda**

<u>num_encomenda</u>	<u>cod_produto</u>	quantidade_prod
----------------------	--------------------	-----------------

### **Produtos**

<u>cod_produto</u>	produto
--------------------	---------

↑ Não está na 3FN

39

39

deti

## 3FN

<b>Encomendas</b>				
<u>num_encomenda</u>	<u>num_cliente</u>	<u>cliente</u>	<u>endereco_cliente</u>	<u>data_encomenda</u>

<b>Linhos_Encomenda</b>		
<u>num_encomenda</u>	<u>cod_produto</u>	<u>quantidade_prod</u>

<b>Produtos</b>	
<u>cod_produto</u>	<u>produto</u>

- Verificamos que a relação Encomendas viola a 3FN:
  - Dependência transitiva dos atributos cliente e endereco\_cliente!
  - Problemas: redundância, actualização de dados cliente obriga a actualizar N tuplos, só é possível registar um novo cliente quando existir uma primeira encomenda,...
- Por decomposição da relação Encomendas:

<b>Encomendas</b>		
<u>num_encomenda</u>	<u>num_cliente</u>	<u>data_encomenda</u>

<b>Clientes</b>		
<u>num_cliente</u>	<u>cliente</u>	<u>endereco_cliente</u>

40

40

deti

## BCFN

<b>Encomendas</b>		
<u>num_encomenda</u>	<u>num_cliente</u>	<u>data_encomenda</u>

<b>Linhos_Encomenda</b>		
<u>num_encomenda</u>	<u>cod_produto</u>	<u>quantidade_prod</u>

<b>Clientes</b>		
<u>num_cliente</u>	<u>cliente</u>	<u>endereco_cliente</u>

<b>Produtos</b>	
<u>cod_produto</u>	<u>produto</u>

• Não podia  
 não estar se  
 houvesse dependência  
 entre algo não  
 chave com a chave...

- Já está na BCFN
- Verificamos que todos os atributos só dependem de toda a chave e de nada mais.

41

41

## Resumo

- Qualidade do Desenho de Base de Dados Relacionais
- Critérios Informais
- Dependências Funcionais
- Normalização (Formas Normais)

42

42