

Normalização: Forma Normal de Boyce Codd 3ª Forma Normal

Fernando Lobo

Base de Dados, Universidade do Algarve

1 / 38

Normalização

- O objectivo é obter esquemas relacionais sem redundância e sem anomalias.
- Exemplo:
 - ▶ Filmes1(nome ano duração nomeEstúdio moradaEstúdio)
com DF's:
 - nome ano → duração
 - nome ano → nomeEstúdio
 - nome ano → moradaEstúdio
 - nomeEstúdio → moradaEstúdio

2 / 38

Possível instância de Filmes1

nome	ano	duração	nomeEstúdio	moradaEstúdio
Star Wars	1977	124	Fox	10 Elm St., Los Angeles
Empire Strikes Back	1980	143	Fox	10 Elm St., Los Angeles
Gone With the Wind	1939	181	Paramount	44 Pine St., Los Angeles
Lion King	1994	124	Disney	56 Oak St., Los Angeles
Return of the Jedi	1983	165	Fox	10 Elm St., Los Angeles
Pocahontas	1995	115	Disney	56 Oak St., Los Angeles

3 / 38

Problemas

- 1 Redundância
 - ▶ a morada de um estúdio aparece repetida várias vezes.
- 2 Pode dar origem a anomalias em “updates”.
 - ▶ ex: actualizar a morada no 1º tuplo e não o fazer no 2º e 5º tuplo.
- 3 Pode dar origem a anomalias em “deletes”.
 - ▶ ex: se apagarmos o filme “Gone With the Wind”, o estúdio “Paramount” desaparece da BD!

4 / 38

Decomposição de relações

- Podemos decompor o esquema de uma relação em dois esquemas mais pequenos.
- Dado $R(A_1 A_2 \dots A_n)$, podemos decompor R em 2 relações: $S(B_1 B_2 \dots B_m)$ e $T(C_1 C_2 \dots C_k)$ de tal forma que:
 - 1 $\{A_1, A_2, \dots, A_n\} = \{B_1, B_2, \dots, B_m\} \cup \{C_1, C_2, \dots, C_k\}$
 - 2 $S = \text{projecção de } R \text{ em } \{B_1, B_2, \dots, B_m\}$
 - 3 $T = \text{projecção de } R \text{ em } \{C_1, C_2, \dots, C_k\}$
- A ideia é que possamos reconstituir a relação original juntando as relações decompostas.
 - ▶ A junção é feita igualando os atributos com o mesmo nome.

5 / 38

Exemplo

Filmes1(nome ano duração nomeEstúdio moradaEstúdio)

nome	ano	duração	nomeEstúdio	moradaEstúdio
Star Wars	1977	124	Fox	10 Elm St., Los Angeles
Empire Strikes Back	1980	143	Fox	10 Elm St., Los Angeles
Gone With the Wind	1939	181	Paramount	44 Pine St., Los Angeles
Lion King	1994	124	Disney	56 Oak St., Los Angeles
Return of the Jedi	1983	165	Fox	10 Elm St., Los Angeles
Pocahontas	1995	115	Disney	56 Oak St., Los Angeles

- Pode ser decomposto em:
 - ▶ Filmes2(nome ano duração nomeEstúdio)
 - ▶ Filmes3(nomeEstúdio moradaEstúdio)

6 / 38

Exemplo (cont.)

Filmes2:

nome	ano	duração	nomeEstúdio
Star Wars	1977	124	Fox
Empire Strikes Back	1980	143	Fox
Gone With the Wind	1939	181	Paramount
Lion King	1994	124	Disney
Return of the Jedi	1983	165	Fox
Pocahontas	1995	115	Disney

Filmes3:

nomeEstúdio	moradaEstúdio
Fox	10 Elm St., Los Angeles
Paramount	44 Pine St., Los Angeles
Disney	56 Oak St., Los Angeles

7 / 38

Forma normal de Boyce-Codd (BCNF)

Uma relação R está em **BCNF** sse:

- para todas as DF's de R não triviais $X \rightarrow Y$,
 X for superchave de R .

Basta existir uma DF em que X não seja superchave, para R não estar em BCNF. Se isso acontecer, diz-se que essa DF viola a condição de BCNF.

8 / 38

Exemplo: Filmes1

Filmes1(nome ano duração nomeEstúdio moradaEstúdio)

com DF's:

nome ano \rightarrow duração

nome ano \rightarrow nomeEstúdio

nome ano \rightarrow moradaEstúdio

nomeEstúdio \rightarrow moradaEstúdio

- A chave é {nome,ano} porque:
 - ▶ nome ano \rightarrow duração nomeEstúdio moradaEstúdio
- Filmes1 não está em BCNF porque:
 - ▶ {nomeEstúdio} não é superchave de Filmes1.
- nomeEstúdio \rightarrow moradaEstúdio, viola a condição BCNF.

9 / 38

Outro exemplo: Filmes2

Filmes2(nome ano duração nomeEstúdio)

com DF's:

nome ano \rightarrow duração

nome ano \rightarrow nomeEstúdio

- A chave é {nome,ano} porque:
 - ▶ nome ano \rightarrow duração nomeEstúdio
- Filmes2 está em BCNF porque em todas as DF's, o lado esquerdo contém a chave.
 - ▶ {nome,ano} é superchave de Filmes2.

10 / 38

Outro exemplo: Filmes3

Filmes3(nomeEstúdio moradaEstúdio)

com DF's:

nomeEstúdio → moradaEstúdio

- A chave é {nomeEstúdio} porque:
 - ▶ nomeEstúdio → moradaEstúdio
- Filmes3 está em BCNF porque {nomeEstúdio} é superchave de Filmes3.

Possíveis instâncias de Filmes2 e Filmes3

Filmes2:

nome	ano	duração	nomeEstúdio
Star Wars	1977	124	Fox
Empire Strikes Back	1980	143	Fox
Gone With the Wind	1939	181	Paramount
Lion King	1994	124	Disney
Return of the Jedi	1983	165	Fox
Pocahontas	1995	115	Disney

Filmes3:

nomeEstúdio	moradaEstúdio
Fox	10 Elm St., Los Angeles
Paramount	44 Pine St., Los Angeles
Disney	56 Oak St., Los Angeles

Decomposição de relações para BCNF

- Não é qualquer decomposição que resolve o problema.
- Exemplo: Decompor Filmes1 em:
 - ▶ Filmes4(nome ano duração)
 - ▶ Filmes5(duração nomeEstúdio moradaEstúdio)

13 / 38

Resultado

Filmes4:

nome	ano	duração
Star Wars	1977	124
Empire Strikes Back	1980	143
Gone With the Wind	1939	181
Lion King	1994	124
Return of the Jedi	1983	165
Pocahontas	1995	115

Filmes5:

duração	nomeEstúdio	moradaEstúdio
124	Fox	10 Elm St., Los Angeles
143	Fox	10 Elm St., Los Angeles
181	Paramount	44 Pine St., Los Angeles
124	Disney	56 Oak St., Los Angeles
165	Fox	10 Elm St., Los Angeles
115	Disney	56 Oak St., Los Angeles

14 / 38

Qual o problema?

- 1 Filmes5 continua sem estar em BCNF.
- 2 Mais grave: se juntarmos Filmes4 com Filmes5, vamos obter tuplos que não pertenciam à relação original!

nome	ano	duração	nomeEstúdio	moradaEstúdio
Star Wars	1977	124	Fox	10 Elm St., Los Angeles
Star Wars	1977	124	Disney	56 Oak St., Los Angeles
Empire Strikes Back	1980	143	Fox	10 Elm St., Los Angeles
Gone With the Wind	1939	181	Paramount	44 Pine St., Los Angeles
Lion King	1994	124	Disney	56 Oak St., Los Angeles
Lion King	1994	124	Fox	10 Elm St., Los Angeles
Return of the Jedi	1983	165	Fox	10 Elm St., Los Angeles
Pocahontas	1995	115	Disney	56 Oak St., Los Angeles

15 / 38

Decomposição “Lossless join”

- Uma decomposição de R em R_1 e R_2 diz-se “Lossless join” se ao juntarmos os tuplos de R_1 com os tuplos de R_2 (igualando os atributos com o mesmo nome), obtivermos exactamente os tuplos de R .
- Ex: A decomposição de Filmes1 em Filmes2 e Filmes3 é “Lossless join”.
- Ex: A decomposição de Filmes1 em Filmes4 e Filmes5 não é “Lossless join”.

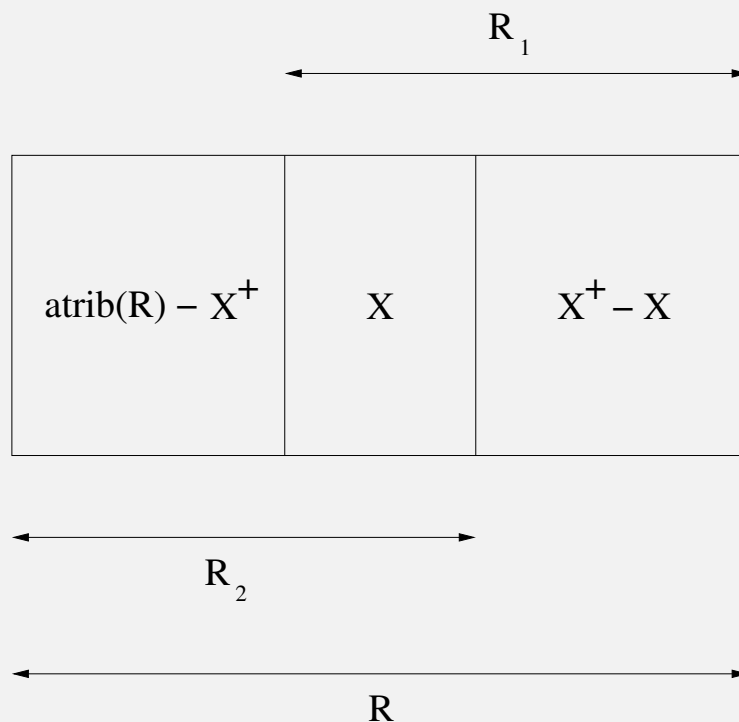
16 / 38

Decomposição para BCNF

- A ideia é usar uma DF $X \rightarrow Y$ que viole a condição de BCNF.
- Calcula-se X^+ .
 - ▶ Não dá todos os atributos, caso contrário X seria superchave.
- Decompõe-se R em duas relações, R_1 e R_2 , com os seguintes esquemas:
 - 1 $R_1 \rightsquigarrow X^+$
 - 2 $R_2 \rightsquigarrow X \cup (\text{atrib}(R) - (X^+))$

17 / 38

Graficamente



18 / 38

Exemplo

Filmes1(nome ano duração nomeEstúdio moradaEstúdio)

com DF's:

nome ano \rightarrow duração

nome ano \rightarrow nomeEstúdio

nomeEstúdio \rightarrow moradaEstúdio

- Escolhe-se uma DF que viole BCNF:
 - ▶ nomeEstúdio \rightarrow moradaEstúdio
- Calcula-se:
 - ▶ nomeEstúdio⁺ = nomeEstúdio moradaEstúdio
- Decompõe-se Filmes1 em:
 - ▶ Filmes2(nomeEstúdio moradaEstúdio)
 - ▶ Filmes3(nome ano duração nomeEstúdio)

19 / 38

Exemplo (cont.)

- Precisamos de verificar se Filmes2 e Filmes3 estão ambas em BCNF.
- Projecta-se as DF's de Filmes1 em Filmes2 e Filmes3.
- Para Filmes2 temos:
 - ▶ nome ano \rightarrow duração
 - ▶ nome ano \rightarrow nomeEstúdio
- {nome,ano} é chave de Filmes2 \Rightarrow Filmes2 está em BCNF.
- Para Filmes3 temos:
 - ▶ nomeEstúdio \rightarrow moradaEstúdio
- {nomeEstúdio} é chave de Filmes3 \Rightarrow Filmes3 está em BCNF.

20 / 38

Outro exemplo

$F_1(\text{nome ano nomeEstúdio nomePresidente moradaPresidente})$

com DF's:

nome ano \rightarrow nomeEstúdio // DF 1
nomeEstúdio \rightarrow nomePresidente // DF 2
nomePresidente \rightarrow moradaPresidente // DF 3

- Chave é {nome,ano}. Porquê?
- DF's 2 e 3 violam BCNF. Vamos decompor usando DF 2.
 - ▶ $\text{nomeEstúdio}^+ = \{\text{nomeEstúdio, nomePresidente, moradaPresidente}\}$
- Decompomos F_1 em F_2 e F_3 :
 - ▶ $F_2(\text{nomeEstúdio nomePresidente moradaPresidente})$
 - ▶ $F_3(\text{nome ano nomeEstúdio})$

21 / 38

Outro exemplo (cont.)

- Agora é necessário ver se F_2 e F_3 estão em BCNF.
- Projecta-se as DF's de F_1 em F_2 e F_3 .
- Para F_2 :
 - nomeEstúdio \rightarrow nomePresidente
 - nomePresidente \rightarrow moradaPresidente
 - ▶ chave é {nomeEstúdio}.
 - ▶ nomePresidente \rightarrow moradaPresidente, viola BCNF.
 - ▶ é necessário decompor F_2 em F_4 e F_5 .
- Para F_3 :
 - nome ano \rightarrow nomeEstúdio
 - ▶ chave é {nome,ano}.
 - ▶ F_3 está em BCNF.

22 / 38

Outro exemplo (cont.)

- Decomposição de F_2 em F_4 e F_5 .
- F_2 (nomeEstúdio nomePresidente moradaPresidente)
DF's:
nomeEstúdio \rightarrow nomePresidente
nomePresidente \rightarrow moradaPresidente
- nomePresidente⁺ = {nomePresidente, moradaPresidente}
- F_4 (nomePresidente moradaPresidente)
 - ▶ DF's: nomePresidente \rightarrow moradaPresidente.
 - ▶ {nomePresidente} é chave. F_4 está em BCNF.
- F_5 (nomeEstúdio nomePresidente)
 - ▶ DF's: nomeEstúdio \rightarrow nomePresidente.
 - ▶ {nomeEstúdio} é chave. F_5 está em BCNF.

23 / 38

Em resumo

F_1 (nome ano nomeEstúdio nomePresidente moradaPresidente)

com DF's:

nome ano \rightarrow nomeEstúdio
nomeEstúdio \rightarrow nomePresidente
nomePresidente \rightarrow moradaPresidente

Foi decomposto em F_3 , F_4 , F_5 :

- F_3 (nome ano nomeEstúdio)
- F_4 (nomePresidente moradaPresidente)
- F_5 (nomeEstúdio nomePresidente)

24 / 38

Decomposição para BCNF

- A decomposição baseada na utilização de uma DF que viole a condição de BCNF, é sempre “lossless join”.
- Qualquer relação com apenas 2 atributos está sempre em BCNF.
- Tentem prová-lo.

25 / 38

3ª Forma Normal: Motivação

- Por vezes a decomposição para BCNF não preserva as DF's da relação original.
- Nesses casos, não devemos decompor para BCNF.
- A situação ocorre tipicamente quando temos DF's com a seguinte estrutura:

$$\begin{aligned}XY &\rightarrow Z \\ Z &\rightarrow Y\end{aligned}$$

26 / 38

Exemplo

CartazFilmes(filme, cinema, cidade)

- cinema \rightsquigarrow cinema onde o filme passa (ex: Forum Algarve).
- cidade \rightsquigarrow cidade onde se encontra o cinema.
- Vamos supor que na mesma cidade não pode passar o mesmo filme em cinemas diferentes.
- Vamos também supor que um cinema pode ter várias salas.
- DF's:

cinema \rightarrow cidade
filme cidade \rightarrow cinema

27 / 38

Exemplo (cont.)

- CartazFilmes tem 2 chaves
 - ▶ {filme,cidade}
 - ▶ {cinema,filme}
- cinema \rightarrow cidade, viola BCNF.
- Somos tentados a decompor CartazFilmes. Obteríamos:
 - ▶ Cartaz₁(cinema,cidade), com DF's: cinema \rightarrow cidade
 - ▶ Cartaz₂(filme,cinema), com DF's: {}
- Mas não conseguimos satisfazer a DF filme cidade \rightarrow cinema em nenhuma das relações resultantes da decomposição!

28 / 38

Exemplo (cont.)

- É possível haver instâncias de Cartaz_1 e Cartaz_2 que satisfaçam as respectivas DF's (i.e., $\text{cinema} \rightarrow \text{cidade}$)
- Mas que a junção viole a DF original: $\text{filme cidade} \rightarrow \text{cinema}$.

29 / 38

Exemplo (cont.)

Cartaz_1 :

cinema	cidade
Forum Algarve	Faro
Santo António	Faro

Cartaz_2 :

filme	cinema
Jesse James	Forum Algarve
Jesse James	Santo António

$\text{Cartaz}_1 \bowtie \text{Cartaz}_2$:

cinema	cidade	filme
Forum Algarve	Faro	Jesse James
Santo António	Faro	Jesse James

30 / 38

3ª Forma Normal

- A 3ª Forma Normal (3FN) relaxa a condição de BCNF.
- Uma relação R está na 3FN se para todas as DF's não triviais $X \rightarrow Y$, se verificar que:
 - ▶ X é superchave de R , ou
 - ▶ Y é membro de uma chave de R
- Se R está em BCNF $\Rightarrow R$ está em 3FN

31 / 38

Exemplo

CartazFilmes(filme, cinema, cidade)

com DF's:

cinema \rightarrow cidade

filme cidade \rightarrow cinema

- está na 3FN porque cidade faz parte de uma chave.
recordem-se que havia 2 chaves:

{filme,cidade} e {cinema,filme}.

32 / 38

Propriedades importantes de uma decomposição

Duas propriedades que uma decomposição deve obedecer:

- ① *Lossless join*: ser possível reconstituir a relação original a partir das relações decompostas.
 - ② *Preservação de dependências*: ser possível verificar as DF's originais nas relações decompostas.
- BCNF garante (1).
 - 3FN garante (1) e (2).

33 / 38

Decomposição para a 3FN

Objectivo: Decompor uma relação R num conjunto de relações que estejam todas na 3FN.

- A decomposição terá a propriedade "*Lossless join*"
- e também irá preservar as dependências funcionais.

34 / 38

Algoritmo de decomposição para a 3FN

Input: uma relação R com um conjunto de DF's F .

Output: uma decomposição de R num conjunto de relações, todas elas na 3FN.

Método:

- 1 $G =$ base mínima para F .
- 2 Para cada DF $X \rightarrow A$ em G , usar XA como esquema de uma das relações decompostas.
- 3 Se nenhum dos esquemas resultantes do passo (2) for uma superchave de R , acrescentar uma relação cujo esquema seja uma chave de R .

35 / 38

Exemplo

$R(ABCDE)$ com DF's $AB \rightarrow C$, $C \rightarrow B$, e $A \rightarrow D$.

- 1 As DF's dadas são uma base mínima (ver justificação nos próximos slides).
- 2 Obtemos as relações $R_1(ABC)$, $R_2(BC)$, $R_3(AD)$.
(R_2 pode ser eliminada porque o seu esquema está contido no esquema de R_1).
- 3 R tem duas chaves: ABE e ACE . Temos de acrescentar uma delas, por exemplo $R_4(ABE)$.

36 / 38

Exemplo (cont.)

Justificação para $\{AB \rightarrow C, C \rightarrow B, A \rightarrow D\}$ ser uma base mínima.

- 1 O lado direito de todas as DFs só tem 1 atributo.
- 2 Não conseguimos inferir nenhuma das DF's a partir das outras duas.
Verificação:

- ▶ será que $AB \rightarrow C$ pode ser inferido de $\{C \rightarrow B, A \rightarrow D\}$?
calcular $(AB)^+$ usando $\{C \rightarrow B, A \rightarrow D\}$.
 $(AB)^+ = ABD$.
 $\Rightarrow AB \rightarrow C$ não pode ser inferido de $\{C \rightarrow B, A \rightarrow D\}$.
- ▶ fazer verificação análoga para provar que:
 - ★ $C \rightarrow B$ não pode ser inferido de $\{AB \rightarrow C, A \rightarrow D\}$.
 - ★ $A \rightarrow D$ não pode ser inferido de $\{AB \rightarrow C, C \rightarrow B\}$.

37 / 38

Exemplo (cont.)

Justificação para $\{AB \rightarrow C, C \rightarrow B, A \rightarrow D\}$ ser uma base mínima.

3. não conseguimos eliminar atributos do lado esquerdo de nenhuma DF e continuar a ter uma base.
 - ▶ a única hipótese seria eliminar A ou B em $AB \rightarrow C$.
 - ▶ se eliminarmos A , temos de demonstrar que $B \rightarrow C$ não pode ser inferido a partir de $\{AB \rightarrow C, C \rightarrow B, A \rightarrow D\}$.
 $B^+ = B$.
 - ▶ fazer verificação análoga para o caso de eliminarmos B .

38 / 38