

Relational Design Theory

Carla Teixeira Lopes

Bases de Dados

Mestrado Integrado em Engenharia Informática e Computação, FEUP

Based on Jennifer Widom and Christopher Ré slides

Agenda

~~Relational Design Overview~~

~~Functional Dependencies~~

~~Closures, Superkeys and Keys~~

~~Inferring Functional Dependencies~~

Normal Forms

Decompositions

Normal Forms

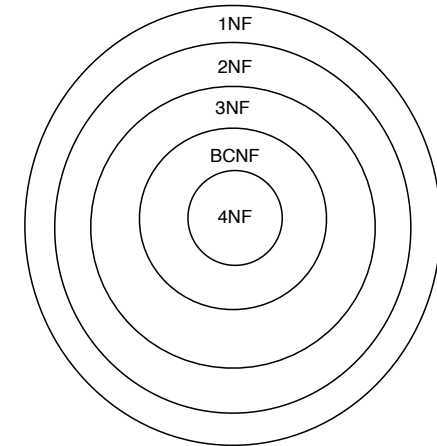
1st Normal Form (1NF)
All tables are flat

2nd Normal Form
disused

Boyce-Codd Normal Form (BCNF)

3rd Normal Form (3NF)

4th and 5th Normal Forms
see text books



DB designs based on
functional
dependencies,
intended to prevent
data anomalies

1st Normal Form (1NF)

The domain of each attribute contains only atomic values and the value of each attribute contains only a single value from that domain

Student	Courses
Mary	{CS145,CS229}
Joe	{CS145,CS106}
...	...



Student	Course
Mary	CS145
Mary	CS229
Joe	CS145
Joe	CS106

2nd Normal Form (2NF)

1NF and no attribute not prime is functionally dependent on a proper subset of a candidate key

An attribute that is member of some key is *prime*

Student-Professor

<u>SID</u>	<u>PID</u>	PName
1	3	Smith
2	2	Bayer

PID->PName



Student-Professor

<u>SID</u>	<u>PID</u>
1	3
2	2

Professor

<u>PID</u>	PName
3	Smith
2	Bayer

Boyce-Codd Normal Form

Relation R with FDs is in BCNF if

For each nontrivial $\bar{A} \rightarrow \bar{B}$, \bar{A} is a (super)key

Why do we have a bad design when this doesn't happen?

\bar{A}	\bar{B}	rest
a		
a		

$\bar{A} \rightarrow \bar{B}$

\bar{A}	\bar{B}	rest
a	b	rest ₁
a	b	rest ₂

redundancy

$\bar{A} \rightarrow \bar{B}$ is a BCNF violation

\bar{A} is not a key \rightarrow Not in BCNF

BCNF? Example #1

Student (SSN, sName, address, HScode, HSname, HScity, GPA, priority)

SSN \rightarrow sName, address, GPA

GPA \rightarrow priority

HScode \rightarrow HSname, HScity

Keys of the relation?

{SSN, HScode}

Does every FD have a key on its left-hand side?

No, none.

BCNF? Example #2

Apply (SSN, cName, state, date, major)

SSN, cName, state \rightarrow date, major

Keys of the relation?

{SSN, cName, state}

Does every FD have a key on its left-hand side?

Yes.

3rd Normal Form (3NF)

2NF and all non-prime attributes are functionally dependent of every candidate key in a non-transitive way

OR

Relation R is in 3NF if, for each nontrivial $\bar{A} \rightarrow \bar{B}$,

\bar{A} is a (super)key **or**

\bar{B} consists of prime attributes only

3NF Example

Bookings (title, theater, city)

theater \rightarrow city

title, city \rightarrow theater



No booking of a movie
in two theaters of the
same city

Keys of the relation?

{title, city}, {theater, title}

BCNF?

FD theater \rightarrow city is a BCNF violation

3NF?

FD theater \rightarrow city has only prime attributes on its right-side

FD title, city \rightarrow theater has a key on its left-hand side and only prime attributes on its right-side

Agenda

Relational Design Overview

Functional Dependencies

Closures, Superkeys and Keys

Inferring Functional Dependencies

Normal Forms

Decompositions

Decomposition of a relational schema

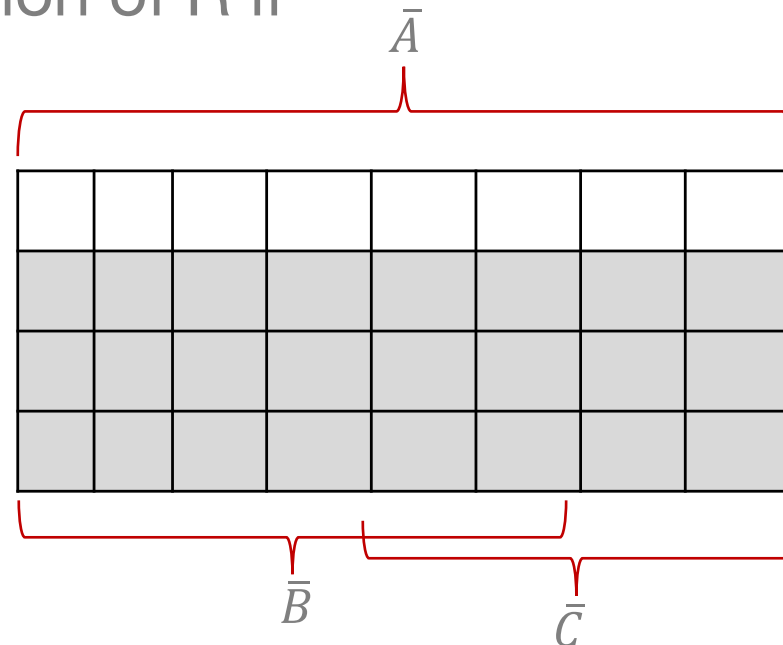
$R (A_1, \dots, A_n)$ \bar{A}
 $R_1 (B_1, \dots, B_n)$ \bar{B}
 $R_2 (C_1, \dots, C_n)$ \bar{C}

R_1 and R_2 are a decomposition of R if

$$\bar{B} \cup \bar{C} = \bar{A}$$

$$R_1 \bowtie R_2 = R$$

Lossless join property



Natural Join (\bowtie)

Student

sID	sName	GPA	HS
12	Mary	3.5	90
23	John	3.8	50

Apply

sID	cName	major	dec
12	Stanford	CS	Y
23	MIT	CS	N



Student \bowtie Apply

sID	sName	GPA	HS	cName	major	dec
12	Mary	3.5	90	Stanford	CS	Y
23	John	3.8	50	MIT	CS	N

Decomposition Example #1

Student (SSN, sName, address, HScode, HSname, HScity, GPA, priority)

S_1 (SSN, SName, address, HScode, GPA, priority)

S_2 (HScode, HSname, HScity)

Is it a correct decomposition?

$$\bar{B} \cup \bar{C} = \bar{A}$$

$$S_1 \bowtie S_2 = Student$$

Decomposition Example #2

Student (SSN, sName, address, HScode, HSname, HScity, GPA, priority)

S_1 (SSN, SName, address, HScode, HSname, HScity)

S_2 (SName, HSname, GPA, priority)

Is it a correct decomposition?

$$\bar{B} \cup \bar{C} = \bar{A}$$

$$S_1 \bowtie S_2 = \text{Student} ?$$

SName and HSname
may not be unique

BCNF decomposition algorithm

Input: relation R + FDs for R

Output: decomposition of R into BCNF relations with “lossless join”

Compute keys for R

Repeat until all relations are in BCNF:

Pick any R' with $\bar{A} \rightarrow \bar{B}$ that violates BCNF

Decompose R' into $R_1(\bar{A}, \bar{B})$ and $R_2(\bar{A}, \text{rest})$

Compute FDs for R_1 and R_2

Compute keys for R_1 and R_2

R'

\bar{A}	\bar{B}	rest



R_1

\bar{A}	\bar{B}

R_2

\bar{A}	rest

BCNF Decomposition Example

Student (SSN, sName, address, HScore, HSname, HScity, GPA, priority)
SSN \rightarrow sName, address, GPA; GPA \rightarrow priority; HScore \rightarrow HSname, HScity
Key: {SSN, HScore}

Pick a BCNF violation

HScore \rightarrow HSname, HScity

Compute FDs and keys for S1

HScore \rightarrow HSname, HScity

Key: {HScore}

S1 is in BCNF

Decompose Student

S1 (HScore, HSname, HScity)

S2 (HScore, SSN, sName, address, GPA, priority)

Compute FDs and keys for S2

SSN \rightarrow sName, address, GPA

GPA \rightarrow priority

Key: {SSN, HScore}

S2 is not in BCNF

} BCNF
violations

BCNF Decomposition Example

Student (SSN, sName, address, HScore, HSname, HScity, GPA, priority)
SSN \rightarrow sName, address, GPA; GPA \rightarrow priority; HScore \rightarrow HSname, Hscity
Key: {SSN, HScore}

Pick a BCNF violation

GPA \rightarrow priority

Compute FDs and keys for S3

GPA \rightarrow priority

Key: {GPA}

S3 is in BCNF

Decompose S2 (HScore, SSN, sName, address, GPA, priority)

S3 (GPA, priority)

S4 (HScore, SSN, sName, address, GPA)

Compute FDs and keys for S4

SSN \rightarrow sName, address, GPA } BCNF violation

Key: {SSN, HScore}

S4 is not in BCNF

BCNF Decomposition Example

Student (SSN, sName, address, HScode, HSname, HScity, GPA, priority)
SSN \rightarrow sName, address, GPA; GPA \rightarrow priority; HScode \rightarrow HSname, Hscity
Key: {SSN, HScode}

Pick a BCNF violation

SSN \rightarrow sName, address, GPA

Decompose S4 (HScode, SSN, sName, address, GPA)

S5 (SSN, sName, address, GPA)

S6 (SSN, HScode)

Compute FDs and keys for S5

SSN \rightarrow sName, address, GPA

Key: {SSN}

S5 is in BCNF

Compute FDs and keys for S6

Key: {SSN, HScode}

S6 is in BCNF

BCNF Decomposition Example

Student (SSN, sName, address, HScode, HSname, HScity, GPA, priority)
SSN \rightarrow sName, address, GPA; GPA \rightarrow priority; HScode \rightarrow HSname, Hscity
Key: {SSN, HScode}

S1 (HScode, HSname, HScity) \longrightarrow Information about high schools

S3 (GPA, priority) \longrightarrow Information about GPA and priorities

S5 (SSN, sName, address, GPA) \longrightarrow Information about students

S6 (SSN, HScode) \longrightarrow Information about the high schools students went

BCNF decomposition algorithm

Input: relation R + FDs for R

Output: decomposition of R into BCNF relations with “lossless join”

Compute keys for R

Repeat until all relations are in BCNF:

Pick any R' with $\bar{A} \rightarrow \bar{B}$ that violates BCNF

Different answers depending on the chosen R'

Extend FD that is used for decomposition (if $A \rightarrow B$ then $A \rightarrow BA^+$)

Decompose R' into $R_1(\bar{A}, \bar{B})$ and $R_2(\bar{A}, \text{rest})$

Compute FDs for R_1 and R_2

See “Projecting a set of FDs” slides

Compute keys for R_1 and R_2

Exercise

Consider the following relation and FDs

Movie (title, year, studioName, president, presAddr)

title, year \rightarrow studioName

studioName \rightarrow president

president \rightarrow presAddr

Decompose into BCNF relations.

Exercise

Movie (title, year, studioName, president, presAddr)

title, year \rightarrow studioName

studioName \rightarrow president

president \rightarrow presAddr

Key: {title, year}

Pick a BCNF violation

studioName \rightarrow president

Decompose Student

S1 (studioName, president)

S2 (studioName, title, year,
presAddr)

Compute FDs and keys for S1

studioName \rightarrow president

Key: {studioName}

S1 is in BCNF

Compute FDs and keys for S2

title, year \rightarrow studioName

studioName \rightarrow presAddr

Key: {title, year}

S2 is not in BCNF

BCNF
violation



Exercise

Movie (title, year, studioName, president, presAddr)

title, year \rightarrow studioName

studioName \rightarrow president

president \rightarrow presAddr

Key: {title, year}

Pick a BCNF violation

studioName \rightarrow presAddr

Decompose S2 (studioName, title, year, presAddr)

S3 (studioName, presAddr)

S4 (studioName, title, year)

Compute FDs and keys for S3

studioName \rightarrow presAddr

Key: {studioName}

S3 is in BCNF

Compute FDs and keys for S4

title, year \rightarrow studioName

Key: {title, year}

S4 is in BCNF

Exercise

Movie (title, year, studioName, president, presAddr)

title, year -> studioName

studioName -> president

president -> presAddr

Key: {title, year}

S1 (studioName, president)

S3 (studioName, presAddr)

S4 (studioName, title, year)

Kahoot time!

Any doubts?

Readings

Jeffrey Ullman, Jennifer Widom, A first course in Database Systems 3rd Edition

Section 3.1 – Functional Dependencies

Section 3.2 – Rules About Functional Dependencies

Section 3.3 – Design of Relational Database Schemas

Section 3.4 – Decomposition: The Good, Bad, and Ugly

Section 3.5 – Third Normal Form