DATABASE SYSTEMS

WEEK 11 LECTURE 1 $ 2

AISHA RIAZ

Topics to Cover

- BCNF (Boyce-Codd Normal Form)
- 4th Normal Form
- 5th Normal Form
- Example

BCNF

- BCNF is an extension to **Third Normal Form** (3NF) and is slightly stronger than 3NF.
- Also known as 3.5 NF.

BCNF

For a table to satisfy the Boyce-Codd Normal Form, it should satisfy the following two conditions:

- It should be in the **Third Normal Form**.
- And, for any dependency A → B, A should be a **super key**.
- The second point sounds a bit tricky, right? In simple words, it means, that for a dependency A → B, A cannot be a **non-prime attribute**, if B is a **prime attribute**.

BCNF

student_id	subject	professor
101	Java	P.Java
101	C++	P.Cpp
102	Java	P.Java2
103	C#	P.Chash
104	Java	P.Java

BCNF

In the table in mentioned in previous slide:

- One student can enroll for multiple subjects. For example, student with **student_id** 101, has opted for subjects - Java & C++
- For each subject, a professor is assigned to the student.
- There can be multiple professors teaching one subject like we have for Java.
- In the table student_id, subject together form the primary key, because using student_id and subject, we can find all the columns of the table.

BCNF

- One more important point to note here is, one professor teaches only one subject, but one subject may have two different professors.
- Hence, there is a dependency between subject and professor here,
  where subject depends on the professor name.

BCNF

- This table satisfies the **1st Normal form** because all the values are atomic, column names are unique and all the values stored in a particular column are of same domain.
- This table also satisfies the **2nd Normal Form** as their is no **Partial Dependency**.
- There is no **Transitive Dependency**, hence the table also satisfies the **3rd Normal Form**.
- But this table is not in **Boyce-Codd Normal Form**.

BCNF

Why this table is not in BCNF?

- In the table above, student_id, subject form primary key, which means subject column is a **prime attribute**.
- But, there is one more dependency, professor → subject.
- And while subject is a prime attribute, professor is a **non-prime attribute**, which is not allowed by BCNF.

Below we have the structure for both the tables.

Student Table

student_id	p_id
101	1
101	2
and so on...	

And, **Professor Table**

p_id	professor	subject
1	P.Java	Java
2	P.Cpp	C++
and so on...		

4^{th} NF

A table is said to have multi-valued dependency, if the following conditions are true,

- For a dependency A → B, if for a single value of A, multiple value of B exists, then the table may have multi-valued dependency.
- Also, a table should have at-least 3 columns for it to have a multi-valued dependency.
- And, for a relation R(A,B,C), if there is a multi-valued dependency between, A and B, then B and C should be independent of each other.

If all these conditions are true for any relation(table), it is said to have multi-valued dependency.

4th NF

s_id	course	hobby
1	Science	Cricket
1	Maths	Hockey
2	C#	Cricket
2	Php	Hockey

4th NF

- As you can see in the table above, student with s_id **1** has opted for two courses, **Science** and **Maths**, and has two hobbies, **Cricket** and **Hockey**.
- You must be thinking what problem this can lead to, right?
- Well the two records for student with s_id **1**, will give rise to two more records, as shown below, because for one student, two hobbies exists, hence along with both the courses, these hobbies should be specified.

4th NF

How to satisfy 4th Normal Form?

To make the above relation satify the 4th normal form, we can decompose the table into 2 tables.

CourseOpted Table

s_id	course
1	Science
1	Maths
2	C#
2	Php

And, **Hobbies Table**,

s_id	hobby
1	Cricket
1	Hockey
2	Cricket
2	Hockey

Now this relation satisfies the fourth normal form.

A table can also have functional dependency along with multi-valued dependency. In that case, the functionally dependent columns are moved in a separate table and the multi-valued dependent columns are moved to separate tables.

If you design your database carefully, you can easily avoid these issues.

4th NF Example

Fourth Normal Form (4NF) - Example

Course-Id	Instructor	Textbook
MGS404	Clay	Hansen
MGS404	Clay	Kroenke
MGS404	Drake	Hansen
MGS404	Drake	Kroenke

- By placing the multi-valued attributes in tables by themselves, we can convert the above to 4NF
- Change to:
  COURSE-INST (Course-Id, Instructor)
  COURSE-TEXT (Course-Id, Textbook)

5th NF

Fifth normal form, also known as project-join normal form, is a level of database normalization designed to reduce redundancy in relational databases recording multi-valued facts by isolating semantically related multiple relationships.

5th NF

A database is said to be in 5NF, if and only if:

- It's in 4NF.
- If we can decompose table further to eliminate redundancy and anomaly, and when we re-join the decomposed tables by means of candidate keys, we should not be losing the original data or any new record set should not arise. In simple words, joining two or more decomposed table should not lose records nor create new records.

5th NF

COURSE
SUBJECT
LECTURER
CLASS

SUBJECT	LECTURER	CLASS
Mathematics	Alex	SEMESTER 1
Mathematics	Rose	SEMESTER 1
Physics	Rose	SEMESTER 1
Physics	Joseph	SEMESTER 2
Chemistry	Adam	SEMESTER 1

5th NF

SUBJECT	LECTURER
Mathematics	Alex
Mathematics	Rose
Physics	Rose
Physics	Joseph
Chemistry	Adam

5NF

CLASS	LECTURER
SEMESTER 1	Alex
SEMESTER 1	Rose
SEMESTER 1	Rose
SEMESTER 2	Joseph
SEMESTER 1	Adam

CLASS	SUBJECT
SEMESTER 1	Mathematics
SEMESTER 1	Physics
SEMESTER 1	Chemistry
SEMESTER 2	Physics

FINDING CANDIDATE KEY

- Let a relation R with attributes ABCD with FDs C → D, C → A and B → C. Find keys for relation R.
  - The core is B. B determines C which determines A and D, so **B is a key**. Therefore B is the key.

- Let a relation R with attributes ABCD with FDs B → C, D → A. Find keys for relation R.
  - The core is BD. B determines C and D determines A, so **BD is a key**. Therefore BD is the key.

- Let a relation R with attributes ABCD with FDs A → B, BC → D and A → C. Find keys for relation R.
  - The core is A. A determines B and C which determine D, so **A is a key**. Therefore A is the key.

Find (candidate) key & check for normal forms [Example]

- Suppose you are given a relation R with four attributes ABCD. For each of the following sets of FDs, do the following: **F = (B → C, D → A)**
  - Identify the candidate key(s) for R.
  - Identify the best normal form that R satisfies (1NF, 2NF, 3NF or BCNF).C

Solution

Candidate Key is **BD**

Relation R is in **1NF but not 2NF**. In above FDs, **there is a partial dependency**

(As per FD B → C, **C depends only on B** but **Key is BD** so **C is partial depends on key (BD)**)

(As per FD D → A, **A depends only on D** but **Key is BD** so **A is partial depends on key (BD)**)

Exercise 2

- Suppose you are given a relation R with four attributes ABCD. For each of the following sets of FDs, do the following: **F = (C → D, C → A, B → C)**
  - Identify the candidate key(s) for R.
  - Identify the best normal form that R satisfies (1NF, 2NF, 3NF or BCNF).

Candidate Key is **B**

Relation R is in **2NF but not 3NF**. In above FDs, **there is a transitive dependency**

(As per FDs **B → C & C → D then B → D** so **D is transitive depends on key (B)**)

(As per FDs **B → C & C → A then B → A** so **A is transitive depends on key (B)**)

Exercise 3

- Suppose you are given a relation R with four attributes ABCD. For each of the following sets of FDs, do the following: **F = (A → B, BC → D, A → C)**
  - Identify the candidate key(s) for R.
  - Identify the best normal form that R satisfies (1NF, 2NF, 3NF or BCNF).

Candidate Key is **A**

Relation R is in **2NF but not 3NF**. In above FDs, **there is a transitive dependency**

(As per FDs **A → B & A → C then A → BC** using union rule) and

(As per FDs **A → BC & BC → D then A → D** so **D is transitive depends on key (A)**)

Exercise 4

- Suppose you are given a relation R with four attributes ABCD. For each of the following sets of FDs, do the following: **F = (ABC → D, D → A)**
  - Identify the candidate key(s) for R.
  - Identify the best normal form that R satisfies (1NF, 2NF, 3NF or BCNF).

Candidate Key are **ABC & BCD**

Relation R is in **3NF but not BCNF**.

In the above FDs, both FDs have **prime attribute (D and A) in dependent (right) side**.

How to normalize database?

1NF

Employee Number	Employee Name	Date of Birth	Department Code	Department Name
1	Raj	1-1-85	1	CE
2	Meet	4-4-86	2	EC
3	Suresh	2-2-85	1	CE

Employee Number	Project Code	Project Description	Project Supervisor
1	1	IOT	Patel
2	2	PHP	Shah
3	1	IOT	Patel
1	2	PHP	Shah

2NF

Employee Number	Employee Name	Date of Birth	Department Code	Department Name
1	Raj	1-1-85	1	CE
2	Meet	4-4-86	2	EC
3	Suresh	2-2-85	1	CE

Project Code	Project Description	Project Supervisor
1	IOT	Patel
2	PHP	Shah

Employee Number	Project Code
1	1
2	2
3	1
1	2

3NF

Employee Number	Employee Name	Date of Birth	Department Code
1	Raj	1-1-85	1
2	Meet	4-4-86	2
3	Suresh	2-2-85	1

Department Code	Department Name
1	CE
2	EC

Project Code	Project Description	Project Supervisor
1	IOT	Patel
2	PHP	Shah

Employee Number	Project Code
1	1
2	2
3	1
1	2

Summary of Normalization

- 1st NF : No repeating groups
- 2nd NF: No partial dependency
- 3rd NF: No transitive dependency
- BCNF: No trivial dependency
- 4th NF: No multi-Valued dependency
- 5th NF: Lossless-join Dependency

NEXT LECTURE

- Transaction Processing