HW4 - Database Normalization

AUTHORS
Tejesh Annavarapu
Srujan Katukam
Anumandla Muralidhar Reddy
Ajaykumar Reddy Rachala

CMSC 608 - Advanced Database

Instructor: Thomas Gyeera

GitHub Repository

👉 AdvancedDatabaseHW4 Repository

Introduction

Database normalization is a systematic approach to organizing data in relational databases to eliminate redundancy and ensure data integrity. This assignment explores different normal forms, functional dependencies, and multi-valued dependencies through a series of problems.

---

1. What conditions would make a 3NF relation violate Boyce-Codd Normal Form (BCNF)?

Answer:

A relation in **Third Normal Form (3NF)** may still violate **Boyce-Codd Normal Form (BCNF)** if: - A **non-trivial functional dependency** exists where a **non-superkey** determines a **candidate key**. - The table has **overlapping candidate keys**, leading to partial dependencies and redundancy issues. - This typically occurs when a **functional dependency exists on a non-superkey attribute**.

Example:

Consider the following relation:

courseID	instructor	department
CSE101	Dr. Smith	CS
CSE102	Dr. Jones	IT

Functional Dependencies:

1. `courseID → instructor`

2. `instructor → department`

Here, `courseID` is a candidate key, but `instructor` (which is not a superkey) determines `department`, violating BCNF. This can cause anomalies such as: - **Update Anomalies**: If an instructor's department changes, multiple rows need to be updated. - **Insertion Anomalies**: A new instructor's department cannot be added unless a course is assigned. - **Deletion Anomalies**: Removing a course may lead to the loss of department information.

BCNF Fix:

To resolve this, we decompose the relation into: - **Course(courseID, instructor)** - **Instructor(instructor, department)**

This ensures that every determinant is a superkey, eliminating redundancy and anomalies.

---

2. Discuss the purpose of Boyce–Codd Normal Form (BCNF) and discuss how BCNF differs from 3NF. Provide an example to illustrate your answer.

Answer:

Purpose of BCNF

BCNF was introduced to address cases where 3NF still allows certain **functional dependencies that cause redundancy**. The main purpose of BCNF is to ensure **every determinant is a superkey**, thereby eliminating anomalies and improving database efficiency.

Differences Between 3NF and BCNF

Feature	3NF	BCNF
Allows non-superkeys as determinants?	Yes, if dependent attributes are prime	No
Eliminates redundancy?	Partially	Completely
Strictness	Less strict	More strict

Example of BCNF Violation in 3NF

projectID	employeeID	department
P1	E1	IT
P2	E2	HR

Functional Dependencies:

1. `{projectID, employeeID} → department`
2. `department → employeeID`

- **3NF:** Acceptable because `department` is a prime attribute.
- **BCNF:** Violated because `department` is not a superkey, leading to redundancy and anomalies.

BCNF Fix:

Decompose into: - **Project(projectID, department)** - **Department(department, employeeID)**

---

3. Describe the concept of multi-valued dependency and discuss how this concept relates to 4NF. Provide an example to illustrate your answer.

Answer:

Definition

A **Multi-Valued Dependency (MVD)** occurs when an attribute is **independent of another attribute but depends on the primary key**. This means that multiple independent values exist for an attribute while maintaining a single primary key.

Explanation of Multi-Valued Dependencies

Unlike functional dependencies, where one attribute determines another, multi-valued dependencies indicate that one attribute is **independent of another** but **still depends on the primary key**.

For example, if a student is enrolled in multiple courses and has multiple hobbies, the courses and hobbies are independent of each other but both depend on the student.

Example of MVD

studentID	course	hobby
S1	Math	Chess
S1	Science	Chess
S1	Math	Painting

Multi-Valued Dependencies:

- `studentID ↠ course`
- `studentID ↠ hobby`

How This Relates to 4NF

- **Third Normal Form (3NF) and Boyce-Codd Normal Form (BCNF) focus on functional dependencies**, but they do not address multi-valued dependencies.
- **Fourth Normal Form (4NF) ensures that multi-valued dependencies are removed**.

4NF Fix:

To remove multi-valued dependencies, we decompose into separate relations: - **Student_Course(studentID, course)** - **Student_Hobby(studentID, hobby)**

This ensures compliance with **Fourth Normal Form (4NF)**, eliminating redundancy and improving data integrity.

---

4. The relation shown in Table 1 lists students enrolled in a postgraduate program at Mzumbe University.

(a) Why is the relation not in 4NF?

Answer:

- The relation contains **multi-valued dependencies**, where each student has multiple sponsors, referees, and supervisors independently:
  - `studentName ↠ refereeName`
  - `studentName ↠ sponsorName`
  - `studentName ↠ supervisorName`

(b) Problems Caused by 4NF Violation

Answer:

- **Data Redundancy**: Same student data is duplicated for multiple sponsors, referees, and supervisors.
- **Insertion Anomalies**: Adding a new sponsor requires duplicating student information.
- **Deletion Anomalies**: Removing a supervisor could remove all related data.

(c) Normalization to 4NF

Answer:

Decompose into separate relations: 1. **Student_Referee (studentName, refereeName)** 2. **Student_Sponsor (studentName, sponsorName)** 3. **Student_Supervisor (studentName, supervisorName)**

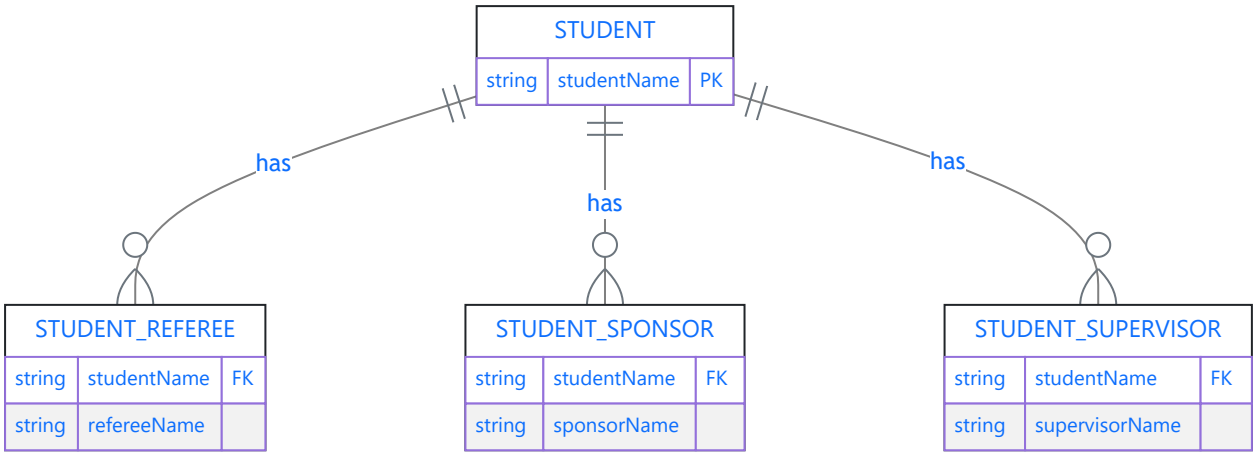
This eliminates multi-valued dependencies, ensuring **4NF compliance**.

---

5. Convert the ERD into a Dependency Diagram in at Least 3NF.

Answer:

ERD Representation Using Crow's Foot Notation



Conclusion

This assignment explores different normal forms and dependency concepts. The ERD representation using **Crow's Foot Notation** visually represents entity relationships, ensuring a **normalized** and **efficient** database design.