5

Chapter 5

Normalization of Database Tables

Database Systems:
Design, Implementation, and Management, Seventh Edition, Rob and Coronel

5

Database Tables and Normalization

- Normalization is a process for assigning attributes to entities. It reduces data redundancies and helps eliminate the data anomalies.
- Probably most valuable as a way of evaluating and correcting DB design
- Normalization works through a series of stages called normal forms:
  - First normal form (1NF)
  - Second normal form (2NF)
  - Third normal form (3NF)
  - Fourth normal form (4NF)
- The highest level of normalization is not always desirable for real-world reasons

Database Tables and Normalization

- Problems with the design based on report Handout
  - Just doesn't fit in a Relational DB – not a table
  - The student number is intended to be part of a primary key, but it contains nulls.
  - The table displays data redundancies.
  - The table entries invite data inconsistencies.
  - The data redundancies yield the following anomalies:
    - Update anomalies.
    - Addition anomalies.
    - Deletion anomalies.

5

The Normalization Process

- Each table represents a single subject
- No data item will be unnecessarily stored in more than one table
- All attributes in a table are dependent on the primary key

The Normalization Process (continued)

TABLE 5.2 Normal Forms

NORMAL FORM	CHARACTERISTIC	SECTION
First normal form (1NF)	Table format; no repeating groups and PK identified	5.3.1
Second normal form (2NF)	1NF and no partial dependencies	5.3.2
Third normal form (3NF)	2NF and no transitive dependencies	5.3.3
Boyce-Codd normal form (BCNF)	Every determinant is a candidate key (special case of 3NF)	5.6.1
Fourth normal form (4NF)	3NF and no independent multivalued dependencies	5.6.2

Database Tables and Normalization

- Conversion to First Normal Form
  - A relational table must not contain repeating groups.
    - (repeating groups involve set of multiple entries in given attribute(s)
  - Repeating groups do not fit in a rectangular table
  - Repeating groups can be eliminated by adding the appropriate entry in at least the primary key column(s).

<Substitute Univ Unnormalized>

5

Prepare for Further Normalization: Identify the Primary Key

- Primary key must uniquely identify all attribute values
- (particularly if you're going to need further normalization) PK may be composite of multiple attributes

Prepare for Further Normalization: Identify all Dependencies

5

- **Remember Functional Dependencies?**
  - A → B means that if you know A then you know B; OR more technically
  - For any given value of A, there is exactly one value of B
- **Dependencies identified through understanding organization and its Business Rules**
- **Dependencies can be depicted with the help of a diagram**
- **Dependency diagram:**
  - Depicts all dependencies found within a given table structure
  - Helpful in getting bird's-eye view of all relationships among a table's attributes
  - Use makes it much less likely that an important dependency will be overlooked

5

Database Tables and Normalization

- Dependency Diagram
  - The primary key components are bold, underlined, and shaded in a different color.
  - The arrows above entities indicate all desirable dependencies, i.e., dependencies that are based on PK.
  - The arrows below the dependency diagram indicate less desirable dependencies -- partial dependencies and transitive dependencies.

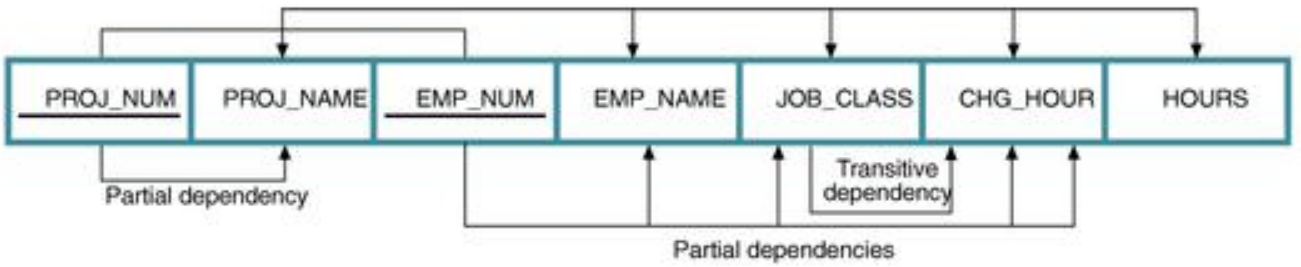

FIGURE 5.4 A DEPENDENCY DIAGRAM: FIRST NORMAL FORM (1NF)

Figure 5.3

Database Tables and Normalization

- 1NF Definition
  - The term first normal form (1NF) describes the tabular format in which:
    - All the key attributes are defined.
    - There are no repeating groups in the table.
    - All attributes are dependent on the primary key.
  - If the table has any partial dependencies or transitive dependencies then You may end up with anomalies during
    - Inserts
    - Deletes
    - Updates

Database Tables and Normalization

- Conversion to Second Normal Form
  - Starting with the 1NF format, the database can be converted into the 2NF format by
    - Writing each key component on a separate line, and then writing the original key on the last line and
    - Writing the dependent attributes after each new key.

PROJECT (PROJ_NUM, PROJ_NAME)
EMPLOYEE (EMP_NUM, EMP_NAME, JOB_CLASS, CHG_HOUR)
ASSIGN (PROJ_NUM, EMP_NUM, HOURS)

Database Tables and Normalization

- 2NF Definition
  - A table is in 2NF if:
    - It is in 1NF and
    - It includes no partial dependencies; that is, no attribute is dependent on only a portion of the primary key.

      (It is still possible for a table in 2NF to exhibit transitive dependency; that is, one or more attributes may be functionally dependent on nonkey attributes.)

5

2NF is Not Good Enough

- **Examine 2NF Current Sections Offered**
  - Definitely In 2NF
  - Problem – data still redundant
  - Anomalies on insert, delete, modify
  - Caused because table is really about more than one thing
- **Transitive dependency is the root of the problem**

Database Tables and Normalization

- 3NF Definition
  - A table is in 3NF if:
    - It is in 2NF and
    - It contains no transitive dependencies.

Database Tables and Normalization

- Conversion to Third Normal Form
  - Create a separate table with attributes in a transitive functional dependence relationship.
    - Any determinant (LHS of FD) gets its own table
    - Any attributes dependent on it (RHS of FD) go in that table
    - Remove dependent attributes from the previous table

PROJECT (PROJ_NUM, PROJ_NAME)
ASSIGN (PROJ_NUM, EMP_NUM, HOURS)
EMPLOYEE (EMP_NUM, EMP_NAME, JOB_CLASS)
JOB (JOB_CLASS, CHG_HOUR)

5

Database Tables and Normalization

- **Boyce-Codd Normal Form (BCNF)**
  - **A table is in Boyce-Codd normal form (BCNF) if every determinant in the table is a candidate key.**

    (A determinant is any attribute whose value determines other values with a row.)
  - **If a table contains only one candidate key, the 3NF and the BCNF are equivalent.**
  - **BCNF is a special case of 3NF.**
  - **Figure 5.9 illustrates a table that is in 3NF but not in BCNF, and how the table can be decomposed to conform to the BCNF form.**
- BCNF doesn't come up very often, DB designers typically aim for 3NF

Normalization and Database Design

- **Normalization should be part of the design process**
  - Many real world DBs have been naively created and suffered from resulting anomalies
- **E-R Diagram provides macro view**
- **Normalization provides micro view of entities**
  - Focuses on characteristics of specific entities
  - May yield additional entities
- **Difficult to separate normalization from E-R modeling**
- **Business rules must be determined for BOTH**

Higher-Level Normal Forms

4NF Definition

- A table is in 4NF if it is in 3NF and has no multiple sets of multivalued dependencies.

Version 1 Table name: FIG5_14A

EMP_NUM	EMP_SERVICE	EMP_ASSIGN
10123	Red Cross	1
10123	United Way	5
10123		12

Version 2 Table name: FIG5_14B

EMP_NUM	EMP_SERVICE	EMP_ASSIGN
10123	Red Cross	
10123	United Way	
10123		1
10123		5
10223		12

Version 3 Table name: FIG5_14C

EMP_NUM	EMP_SERVICE	EMP_ASSIGN
10123	Red Cross	1
10123	Red Cross	5
10123	United Way	12

FIGURE 5.14 TABLES WITH MULTIVALUED DEPENDENCIES

Figure 5.14 Tables with Multivalued Dependencies

5

Denormalization

- **Normalization is only one of many database design goals.**
- **Normalized (decomposed) tables require additional processing, reducing system speed.**
  - More joins of tables
  - More disk accesses
- **Normalization purity is often difficult to sustain in the modern database environment.**
  - The conflict between design efficiency, information requirements, and processing speed are often resolved through compromises that include denormalization.

Denormalization (continued)

- **Unnormalized tables in a production database tend to have these defects:**
  - Risks of inconsistency MUST be managed
    - Application program should ensure that inconsistency does not happen
  - Data updates are less efficient because programs that read and update tables must deal with larger tables
  - Indexing is much more cumbersome
  - Unnormalized tables yield no simple strategies for creating virtual tables known as *views*
- **Use denormalization cautiously**
- **Understand why—under some circumstances—unnormalized tables are a better choice**

5

End Chapter 5