Amity School of Business

BBA, Semester 2

Analysis and Design of Business System

(ADBS)

Arpan Sinha

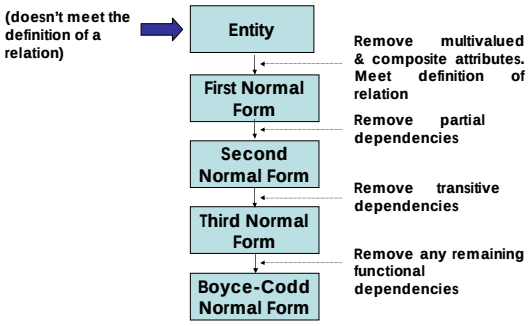
Module-IV

System And Database Design

Types of NF

- First Normal Form (1NF)
- Second Normal Form (2NF)
- Third Normal Form (3NF)
- Boyce-Codd Normal Form (BCNF)

Steps in Normalization



First Normal Form (1NF)

If a table of data meets the definition of a relation, it is in first normal form.

- **Every relation has a unique name.**
- **Every attribute value is atomic (single-valued).**
- **Every row is unique.**
- **Attributes in tables have unique names.**
- **The order of the columns is irrelevant.**
- **The order of the rows is irrelevant.**

Example

- Consider a relation **LIVED_IN** , which keeps the records of person and his residence in different cities
- In this relation, the domain **RESIDENCE** is not simple, e.g. an attribute " Abhishek can have residence in Jamshedpur , Mumbai or Delhi and thus relation is un-normalized.

Relation : LIVED_IN

PERSON	RESIDENCE	
	CITY	DATE-MOVED
Abhishek	Jameshedpur	121202
	Mumbai	070803
	Delhi	050504
	CITY	DATE-MOVED
Thomas	Singapore	100401
	Kolkata	090202
	Bangalore	010105

Un-normalized Relation

Example

Amity School of Business

- Now the relation LIVED_IN is normalized by combining each row in residence with its corresponding value of PERSON and making this combination a row of the relation.
- So non-simple domain RESIDENCE is replaced with simple domains

Relation : LIVED_IN		
PERSON	CITY	DATE-MOVED-IN
Abhishek	Jamshedpur	121202
Abhishek	Mumbai	070803
Abhishek	Delhi	050504
Thomas	Singapore	100401
Thomas	Kolkata	090202
Thomas	Bangalore	010105

Normalized Relation

Example

Amity School of Business

Relation : Patient_Doctor					
Patient-Name	**DOB**	**Doctor Name**	**Contact No.**	**Date-Time**	**Duration-Minutes**
Mathew	**10/02/1957**	**Abhishek**	**657-2145063**	**10/01/10 10:00**	**15**
Ravi	**27/01/1962**	**Sanjay**	**651-2214381**	**10/01/10 11:00**	**10**
Jose	**30/03/1971**	**Thomas**	**011-2324567**	**10/01/10 10:30**	**10**
Jose	**30/03/1971**	**Thomas**	**011-2324567**	**08/03/10 09:00**	**20**
Ravi	**27/01/1962**	**Abhishek**	**657-2145063**	**10/01/10 10:15**	**15**
Mathew	**10/02/1957**	**Thomas**	**011-2324567**	**10/01/10 10:50**	**20**
Ranjan	**02/11/1970**	**Sanjay**	**651-2214381**	**10/01/10 11:10**	**20**
Mathew	**10/02/1957**	**Abhishek**	**657-2145063**	**05/05/10 16:00**	**15**

Relation Patient_Doctor in 1NF

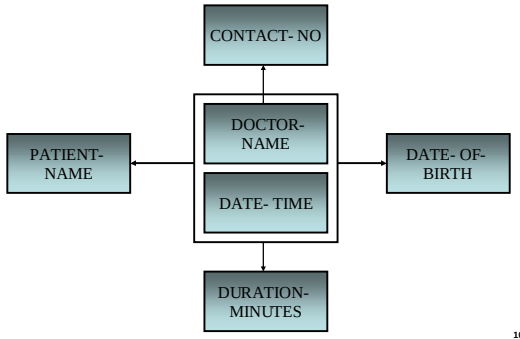
Problems in 1NF

- 1NF contains redundant information. The Relation PATIENT_DOCTOR is in 1NF, even then this has following problems with the structure-
  - A doctor, who does not currently have an appointment with a patient, cannot be represented.
  - Similarly, we cannot represent a patient who does not currently have an appointment with a doctor
  - There is redundant information such as patient's date of birth& doctor's phone number.
  - While deleting the last remaining records containing details of a patient or a doctor, all records of that patient or doctor will be lost.

Therefore the Relation PATIENT_DOCTOR has to be normalized further by separating the information relating to several distinct entities

Amity School of Business

Function Dependency diagram for relation PATIENT_DOCTOR



Amity School of Business

Definition of FD …

• Describes the relationship between attributes in a relation.

For example, if ***A*** and ***B*** are attributes of relation ***R*, *B*** is functionally dependent on ***A*** **(denoted *A* → *B*), if** each value of ***A*** **in *R* is associated with exactly one value of *B* in *R*.**

• Functional dependency is a property of the meaning or semantics of the attributes in a relation.

Second Normal Form (2NF)

A relation R is said to be in second normal form (2NF) if

- It is in 1NF and every non-prime key attributes of R is fully functionally dependent on each relation (Primary) Key of R i.e. no partial dependency is allowed in the relation R.
- In other words, no attributes of the relation should be functionally dependent on only one part of the concatenated primary key.
- Thus, 2NF can be violated only when a key is a composite key, or one that consisting of more than one attribute.
- 2NF is based on the concept of full functional dependency (FFD)
- 2NF is an intermediate step towards higher normal forms. It eliminates the problems of 1NF

Example

Amity School of Business

- It can be observed from the 1NF relational table that a doctor cannot have two simultaneous appointments and thus DOCTOR-NAME and DATE-TIME is a compound key.
- Similarly, a patient cannot have same time from two different doctors. Therefore, PATIENT-NAME and DATE-TIME attributes are also a candidate key.

Example

Amity School of Business

- The Relation could be depicted as:
- **PATIENT_DOCTOR** (PATIENT-NAME, DOB, DOCTOR-NAME, DATE-TIME, CONTACT-NO, DURATION-MINUTES)
- In this relation composite key (DOCTOR-NAME, DATE-TIME), is taken as a primary key.
- But there is a partial dependency as CONTACT-NO is Dependent upon DOCTOR-NAME, and hence the relation is not in 2NF.
- Therefore, to bring the relation in 2NF, the information about doctors and their contact numbers have to be separated from information about patients and their appointments with doctors.

Example

Amity School of Business

- Thus, the relation is decomposed into two table namely PATIENT_DOCTOR and DOCTOR.
- The relation in 2NF can be depicted as:
  - **PATIENT_DOCTOR**(PATIENT-NAME, DOB, DOCTOR-NAME, DATE-TIME, DURATION-MINUTES)
  - **DOCTOR**(DOCTOR-NAME, CONTACT-NO)

Example

Amity School of Business

Relation PATIENT_DOCTOR decomposed into two table for refinement in 2NF

Relation : PATIENT_DOCTOR

Patient-Name	DOB	Doctor Name	Date-Time	Duration-Minutes
Mathew	10/02/1957	Abhishek	10/01/05 10:00	15
Ravi	27/01/1962	Sanjay	10/01/05 11:00	10
Jose	30/03/1971	Thomas	10/01/05 10:30	10
Jose	30/03/1971	Thomas	08/03/05 09:00	20
Ravi	27/01/1962	Abhishek	10/01/05 10:15	15
Mathew	10/02/1957	Thomas	10/01/05 10:50	20
Ranjan	02/11/1970	Sanjay	10/01/05 11:10	20
Mathew	10/02/1957	Abhishek	05/05/05 16:00	15

Example

Amity School of Business

Relation : DOCTOR

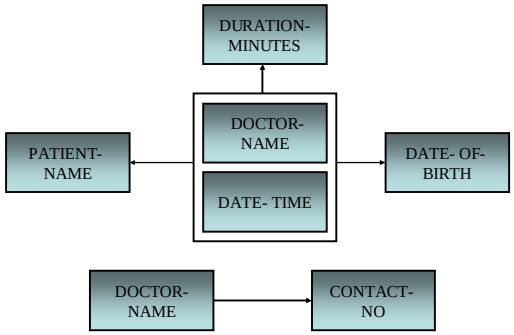
Doctor Name	Contact No.
Abhishek	657-2145063
Sanjay	651-2214381
Thomas	011-2324567
Thomas	011-2324567
Abhishek	657-2145063
Thomas	011-2324567
Sanjay	651-2214381
Abhishek	657-2145063

Problems in 2NF:

- Deleting a record from relation PATIENT_DOCTOR may lose patient's details
- Any changes in the details of the patient may involve changing multiple occurrences because this information is still stored redundantly.

Example

Amity School of Business



FDDs for Relations PATIENT-DOCTOR AND DOCTOR

TRANSITIVE DEPENDENCY

Let A, B and C be three attributes of a relations such that A ⟶ B and B ⟶ C. From these FDs, we may able to derive A ⟶ C , this dependence A ⟶ C is transitive dependency.

Third Normal Form (3NF)

- A relation R is said to be in Third Normal Form (3NF) if
  - It is in 2NF, and The non-prime attributes are Mutually independent & Functionally dependent on the primary (or relation) key.
  - No attributes of the relation should be transitively functionally dependent on the primary key or no non-prime attribute is functionally dependent on another non-prime attribute.
  - This means that a relation in 3NF consists of the primary key and a set of independent nonprime attributes.
  - 3NF is based on the concept of transitive dependency.

Example

- ***The following Relation in 1NF was depicted as:***

  PATIENT_DOCTOR (PATIENT-NAME, DOB, DOCTOR-NAME, DATE-TIME, CONTACT-NO, DURATION-MINUTES)
- ***The relation in 2NF can be depicted as:***

  PATIENT_DOCTOR (PATIENT-NAME, DOB, DOCTOR-NAME, DATE-TIME, DURATION-MINUTES)

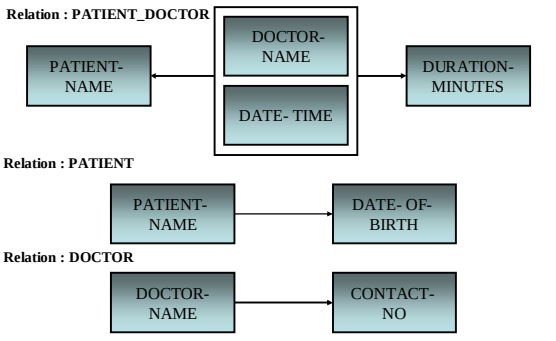
  DOCTOR(DOCTOR-NAME, CONTACT-NO)
- ***Thus the relation in 3NF is depicted as:***

  PATIENT(PATIENT-NAME, DOB)

  PATIENT_DOCTOR (PATIENT-NAME, DOCTOR- NAME, DATE-TIME, DURATION-MINUTES)

  DOCTOR (DOCTOR-NAME, CONTACT-NO)

Example



Relation in 3NF

Boyce-Codd Normal Form (BCNF)

To eliminate the problems and redundancy of 3NF, R.F. Boyce proposed a normal form known as *Boyce Codd Normal Form* (BCNF).

Relation R is in BCNF :

- if for every nontrivial FD: X → Y between attributes X and Y holds in R That means
  - X is super key of R
  - X → Y is a trivial FD, i.e. Y ⊂ X

In Other words, a relation must only have candidate keys as determinants.

Boyce-Codd Normal Form (BCNF)

- 3NF and every determinant is a candidate key.

✓ BCNF is based on the concept of a *determinant.*

✓ A determinant is any attribute (simple or composite) on which some other attribute is fully functionally dependent.

✓ A relation is in BCNF is, and only if, every determinant is a candidate key.

Example

Amity School of Business

Let us consider relation PROJECT_PART

PROJECT_PART(PROJECT-NAME, PART-CODE, VENDER-NAME, QTY)

Relation : PROJECT_PART			
PROJECT-NAME	**PART-CODE**	**QTY**	**VENDER-NAME**
P1	ABC	10	Thomas
P1	BCA	20	John
P2	ABC	30	Thomas
P2	BCA	40	Alok

Relation : PROJECT_VENDOR is in 3NF

Example

Amity School of Business

In this table lists the projects, the parts, the quantities of those parts they use and the vender who supply these parts. There are two assumptions.

- Each project is supplied with a specific part by only one vendor, although a vender can supply that part to more than one project.
- A vendor makes only one part but the same part made by other vendors.

The primary key of the relation PROJECT_PART are PROJECT-NAME and PART-CODE.

Example

Amity School of Business

The relation PROJECT _PART is in 3NF, since there is no transitive functional dependencies on the prime key. However, it is not in BCNF because the attribute VENDER-NAME is the determinant of PART-CODE.

To Convert this relation form 3NF to BCNF by decomposing this single relation PROJECT_PART into two relation PROJECT_VENDOR and VENDOR_PART.

The decompose relations are

PROJECT_VENDOR(PROJECT-NAME, VENDOR NAME, QTY)

VENDOR_PART(VENDOR-NAME, PART-CODE)

Example

Amity School of Business

Relation: PROJECT_VENDOR		
PROJECT-NAME	VENDOR-NAME	QTY
P1	Thomas	10
P1	John	20
P2	Thomas	30
P2	Alok	40

Relation : VENDOR_PART	
VENDOR-NAME	PART-CODE
Thomas	ABC
John	BCA
Alok	BCA

Relation : PROJECT_VENDOR and VENDOR_PART is in BCNF

DBMS – Users

Amity School of Business

- **Database Administrator**

Person having central control over data and programs accessing that data.

Roles and Duties of the Database Administrator include:

- Scheme definition
- Storage structure and access method definition
- Scheme and physical organization modification
- Granting of authorization for data access
- Integrity constraint specification

Amity School of Business

- **Application Programmers**
  computer professionals interacting with the system through application programs written in a host language.
- **Sophisticated users**
  Interact with the system without writing programs. They form requests by writing queries in a database query language.
- **Specialized users**
  Sophisticated users writing special database application programs. These may be CADD systems, knowledge-based and expert systems, complex data systems (audio/video), etc.
- **Naive users**
  Unsophisticated users who interact with the system by using permanent application programs (e.g. automated teller machine).

Benefits of Normalization

- Searching, sorting, and creating indexes is faster, since tables are narrower, and more rows fit on a data page.
- Index searching is often faster, since indexes tend to be narrower and shorter.
- You usually have fewer indexes per table, so data modification commands are faster.
- Fewer null values and less redundant data, making your database more compact.

- Triggers execute more quickly if you are not maintaining redundant data.
- Data modification anomalies are reduced thus maintains data integrity.
- Normalization is conceptually cleaner and easier to maintain and change as your needs change.
- Optimized and faster queries.
- Faster update performance.

The End