

Lecture 7

Steps in Normalization /continuance /

- **The 4th Normal Form (4NF)**
- **The 5th Normal Form (5NF) and the Domain-Key Normal Form (DKNF)**
 - Converting a Table with Partial Dependencies into DKNF Tables
 - Converting a Table with Transitive Dependencies into DKNF Tables
 - Converting into DKNF a Table in Which Not Every Determinant Is a Candidate Key
 - Converting a Table with Multivalued Dependencies into DKNF
 - Single-Theme Tables and the DKNF

Subject: CSE396 “**Database Design and Administration**”

Instructor's full name: Lyazat Kydyrgalievna Naizabayeva

The 4th Normal Form (4NF)

Definition: A table is in 4NF if it is in BCNF and if it has no multi-valued dependencies.

- The 4th Normal Form is concerned with the anomalies that can occur when a table fails to have the property of containing no multivalued dependencies (i.e., the anomalies that can occur when a table does have such dependencies).
- We develop below a table that has these undesirable multivalued dependencies.

- Suppose we have some information about the hobbies of some students at Enormous State University and want to put this information into a database. Suppose, in particular, that Jack Jones's hobbies are surfing the Internet and playing chess;
- Lynn Lee's, photography and stamp collecting; Mary Ruiz's, surfing the Internet and photography; and Lynn Smith's, playing poker.

If we (foolishly) try to put all this information into just one table, here is what we get.

Table_7

LastName	Major	Hobby
Jones	Library and Information Science	Surfing the Internet
Jones	Library and Information Science	Chess
Jones	Public Affairs	Surfing the Internet
Jones	Public Affairs	Chess
Lee	Library and Information Science	Photography
Lee	Library and Information Science	Stamp collecting
Ruiz	Pre-Medicine	Surfing the Internet
Ruiz	Pre-Medicine	Photography
Ruiz	Biochemistry	Surfing the Internet
Ruiz	Biochemistry	Photography
Smith	Pre-Law	Playing poker

- The problem is that Jack Jones, for example, has two majors and two hobbies.
- If we coupled each of his majors with just one of his hobbies (e.g., LIS with chess, or Public Affairs with surfing the Internet), we would imply that Jack plays chess only as an LIS major and surfs the Internet only as a Public Affairs major.
- This would not make sense. (Note that in this relatively small and simple example, it is obvious that such restrictive pairing does not make sense.

- In practice, however, the problems arise in connection with much larger tables, where it may be very difficult to detect that restrictive pairing has occurred.)
- To avoid such false implications, we enter all pairings of majors and hobbies for all the students. Obviously, however, this approach has the problem of redundant information.
- Equally obviously, updating this table presents anomalies; for example, you can work out for yourself what would have to be added to Table7 if Jones took up tennis as a third hobby.

- This situation is an example of the effects of multivalued dependencies.
- A multivalued dependency occurs when
 - (a) a table has at least three attributes,
 - (b) two of the attributes are multivalued, and
 - (c) the values of the multivalued attributes depend on only one of the remaining attributes.

Table7 fits these specifications for the following reasons:

- The LastName attribute determines multiple values of the attributes Major and Hobby, but neither of these latter attributes depends on the other; they are independent.

The notation for multivalued dependency is a double arrow. In this example, we can write: LastName $\twoheadrightarrow$ Major, and LastName $\twoheadrightarrow$ Hobby.

We read these expressions as, "LastName multidetermines Major" and "LastName multidetermines Hobby."

Once again, single-theme tables provide the solution.

We break Table7 down into the following tables.

LastName	Major
Jones	Library and Information Science
Jones	Public Affairs
Lee	Library and Information Science
Ruiz	Pre-Medicine
Ruiz	Biochemistry
Smith	Pre-Law

LastName	Hobby
Jones	Surfing the Internet
Jones	Chess
Lee	Photography
Lee	Stamp collecting
Ruiz	Surfing the Internet
Ruiz	Photography
Smith	Playing poker

- Tables 8 and 9 display, separately, the various students' majors and hobbies; and while doing so, these tables correctly avoid suggesting any connections between particular majors and particular hobbies.

Example2

Professors NAME	Class	Committee
Chris	125	admissions
Chris	125	AP Board
Chris	126	AP Board
Chris	127	admissions
Chris	127	AP Board



Professors NAME	Committee
Chris	admissions
Chris	AP Board

Professors NAME	Class
Chris	125
Chris	126
Chris	127

The 5th Normal Form (5NF) and the Domain-Key Normal Form (DKNF)

Definition: A table is in 5NF, also called "Projection-Join Normal Form" (PJNF), if it is in 4NF and if every join dependency in the table is a consequence of the candidate keys of the table.

Definition: A table is in DKNF if every constraint on the table is a logical consequence of the definition of keys and domains.

The 5th Normal Form is difficult to illustrate in terms of relatively simple examples.

Hence, we will not attempt to illustrate the 5NF property of having every join dependency in the table be a consequence of the candidate keys of the table.

This omission is a minor one, for at least two reasons: First, in practice the 4NF is often regarded as sufficient; and second, the Domain-Key Normal Form (DKNF) subsumes the 5NF.

- The DKNF is important because it offers a complete solution to the problem of avoiding anomalies:
- A set of tables (relations) that is in DKNF is known, as a consequence of a theorem proved by Ronald Fagin in 1981, to be free of anomalies.
- We do not attempt here to reproduce the proof of Fagin's theorem but merely to illustrate how the theorem can be applied in practice.

- The DKNF definition is this: A relation is in DKNF if every constraint on the relation is a logical consequence of the definitions of keys and domains.
- To understand what this definition means, we begin by noting that the central ideas are embodied in the words "constraint," "key," and "domain."
- By "key" Fagin means both primary keys and candidate keys.

- By "domain" Fagin means the set of definitions of the contents of attributes (columns) and any limitations on the kind of data to be stored in the columns, such as a limitation to only numeric data or only logical data; in addition, domain limitations may include such matters as the format (e.g., a limitation on numeric data to being expressed to exactly two decimal digits).
- By "constraint" Fagin means any rule dealing with attributes that is clear enough so that one can decide whether the rule is upheld or broken by any set of the data with which one is dealing.

- There is an important qualification to be attached to the DKNF definition as presented in the preceding paragraph. Fagin excludes constraints that are time-dependent or relate to changes made in data values.
- That means that a time-dependent constraint (or other constraint on changes in value) may exist in a table and may fail to be a logical consequence of the definitions of keys and domains, yet the table may nevertheless be in DKNF.

- As an illustration, some states have a property-tax rule specifying that the assessed value of the primary-residence property owned by a citizen over 65 cannot be increased above the value that was assessed in the year in which the property owner turned 65.
- The existence of such a rule would not, in itself, prevent a table of properties and their assessed values from being in DKNF.

- Achieving DKNF amounts to establishing a set of tables in each of which the constraints follow logically from (i.e., are logical consequences of) the keys and the domain definitions.
- Although there is no direct procedure for converting an arbitrary table into one or more tables each of which is in DKNF, in practice the effort to replace an arbitrary table by a set of single-theme tables achieves the goal.
- To show this, we consider some of the previous examples from the DKNF point of view.

Converting a Table with Partial Dependencies into DKNF Tables

- Here once again is the table, Table3, that we used in our discussion of the problem of partial dependencies. Since we going to use it here, we name this copy of it Table 10.

FirstName	LastName	Major	Level
Jack	Jones	LIS	Graduate
Lynn	Lee	LIS	Graduate
Mary	Ruiz	Pre-Medicine	Undergraduate
Lynn	Smith	Pre-Law	Undergraduate
Jane	Jones	LIS	Graduate

- Let us consider Table 10 from the DKNF point of view. First, we see that the key is composite, consisting of the LastName-FirstName pair of attributes.
- We see also that all other attributes in the table are dependent on this key.
- But there is another significant aspect to this table: the Level attribute is dependent on the LastName attribute, i.e., Level is dependent on just part of the key. (As noted earlier, this partial dependency is contrived, but nevertheless it illustrates the problem of partial dependency.)
- Because Level is dependent on just LastName, the table fails to be one in which all constraints are logical consequences of the key; hence, Table 10 is not in DKNF.

- From the DKNF point of view, therefore, we see that we should take the Level attribute out of Table 10 and put it in some other table, or tables, where it will be a logical consequence of the keys and domains. Clearly, a table that associates just the attributes Major and Level will achieve this.
- We will also need a table that provides the necessary link between the paired attributes, FirstName and LastName, and the attribute Major. In such a table, the attribute Major will be a logical consequence of the keys and domains.

Thus it appears that we need two tables, one containing just **Major** and **Level**, and the other containing **FirstName**, **LastName**, and **Major**. We can indicate this more briefly as **Table A: (Major, Level)** and **Table B: (FirstName, LastName, Major)**.

Here are the tables.

(Table B as described above)

(Table A as described above)

Major	Level
LIS	Graduate
Pre-Medicine	Undergraduate
Pre-Law	Undergraduate

FirstName	LastName	Major
Jack	Jones	LIS
Lynn	Lee	LIS
Mary	Ruiz	Pre-Medicine
Lynn	Smith	Pre-Law
Jane	Jones	LIS

These are single-theme tables, and we arrived at them by steps aimed at achieving DKNF.

Converting a Table with Transitive Dependencies into DKNF Tables

- Here once again is the table, Table4, that we used in our discussion of transitive dependencies. Since we going to use it here, we name this copy of it Table 11.

Author Last Name	Author First Name	Book Title	Subject	Collection or Library	Building
Berdahl	Robert	The Politics of the Prussian Nobility	History	PCL General Stacks	Perry-Castañeda Library
Yudof	Mark	Child Abuse and Neglect	Legal Procedures	Law Library	Townes Hall
Harmon	Glynn	Human Memory and Knowledge	Cognitive Psychology	PCL General Stacks	Perry-Castañeda Library
Graves	Robert	The Golden Fleece	Greek Literature	Classics Library	Waggener Hall
Miksa	Francis	Charles Ammi Cutter	Library Biography	Library and Information Science Collection	Perry-Castañeda Library
Hunter	David	Music Publishing and Collecting	Music Literature	Fine Arts Library	Fine Arts Building
Graves	Robert	English and Scottish Ballads	Folksong	PCL General Stacks	Perry-Castañeda Library

- You will recall from the discussion of this table as Table4 that it exhibits the following transitive dependencies:
 - Book Title $\rightarrow$ Subject,
 - Subject $\rightarrow$ Collection-Library,
 - Collection-Library $\rightarrow$ Building.
- From the DKNF point of view, this means that the primary key, Book Title, is not the only thing that determines the Collection-Library attribute and the Building attribute. In turn, this means that there are constraints that are not logical consequences of the key and, hence, that the table is not in DKNF.

- Reasoning from the DKNF point of view, we would like to have a table in which the Building attribute is a logical consequence of the key; constructing a table containing the Collection-Library and Building attributes, with Collection-Library as key, will accomplish that.
- Again from the DKNF point of view, we would like to have a table in which the Collection-Library attribute is a logical consequence of the key; clearly, a table containing Subject (as key) and Collection-Library suffices.

- The same point of view leads us to desire a table in which the Author First Name and Author Last Name attributes will be a logical consequence of the key; such a table is one that contains Book Title (as key), Author First Name, and Author Last Name.
- Finally, a table that contains Book Title (as key) and Subject will be
 - (1) a table in which the attribute Subject will be a logical consequence of the key and
 - (2) a table that provides the necessary connection between Title and Subject.

Thus from the DKNF point of view, we are led to the same tables as previously:

Author Last Name	Author First Name	Book Title
Berdahl	Robert	The Politics of the Prussian Nobility
Yudof	Mark	Child Abuse and Neglect
Harmon	Glynn	Human Memory and Knowledge
Graves	Robert	The Golden Fleece
Miksa	Francis	Charles Ammi Cutter
Hunter	David	Music Publishing and Collecting
Graves	Robert	English and Scottish Ballads

Book Title	Subject
The Politics of the Prussian Nobility	History
Child Abuse and Neglect	Legal Procedures
Human Memory and Knowledge	Cognitive Psychology
The Golden Fleece	Greek Literature
Charles Ammi Cutter	Library Biography
Music Publishing and Collecting	Music Literature
English and Scottish Ballads	Folksong

Subject	Collection or Library
History	PCL General Stacks
Legal Procedures	Law Library
Cognitive Psychology	PCL General Stacks
Greek Literature	Classics Library
Library Biography	Library and Information Science Collection
Music Literature	Fine Arts Library
Folksong	PCL General Stacks

Collection or Library	Building
PCL General Stacks	Perry-Castañeda Library
Law Library	Townes Hall
Classics Library	Waggener Hall
Library and Information Science Collection	Perry-Castañeda Library
Fine Arts Library	Fine Arts Building

Here we have arrived at these same tables by considering how the information in Table12 (the same information as in Table4) should be re-arranged from the DKNF point of view.

Converting into DKNF a Table in Which Not Every Determinant Is a Candidate Key

SSN	Major	Adviser
123-45-6789	Library and Information Science	Dewey
123-45-6789	Public Affairs	Roosevelt
222-33-4444	Library and Information Science	Putnam
555-12-1212	Library and Information Science	Dewey
987-65-4321	Pre-Medicine	Semmelweis
987-65-4321	Biochemistry	Pasteur
123-54-3210	Pre-Law	Hammurabi

- You will recall from the discussion of this table as Table 6 that one determinant is the pair of attributes, SSN and Major, which determines Adviser; another determinant is the pair, SSN and Adviser, which determines Major; and still another is Adviser alone, which also determines Major. And you will recall that the candidate keys are the pairs, SSN-Major and SSN-Adviser. The third determinant, Adviser, is not a candidate key.
- From the DKNF point of view, we reason as follows: If we choose SSN-Adviser as the key, then Major is determined by, and hence is a logical consequence of, this key. If, instead, we choose SSN-Major as the key, then Adviser is determined by, and hence is a logical consequence of, this alternative key. But in either case, the third constraint, viz., that Adviser determines Major, is not a logical consequence of the key. Hence, the table is not in DKNF.

- In order to move from this table to a set of tables in DKNF, we can argue. from the DKNF point of view, that we need to move Major into a table in which it will be a logical consequence of the key.
- Such a table would obviously need to have Adviser as the key.
- If we put Adviser and Major into such a table, then we will need at least one other table, viz., a table that provides the necessary link between SSN and Adviser, so that we will know who each student's adviser is.
- Once we have put SSN and Adviser into such a table, there is nothing further that needs to be done.

These are the tables presented in Here we have arrived at these same tables by considering how the information in Table13 (the same information as in Table6) should be re-arranged from the DKNF point of view.

Major	Adviser
Library and Information Science	Dewey
Public Affairs	Roosevelt
Library and Information Science	Putnam
Pre-Medicine	Semmelweis
Biochemistry	Pasteur
Pre-Law	Hammurabi
History	Herodotus

SSN	Adviser
123-45-6789	Dewey
123-45-6789	Roosevelt
222-33-4444	Putnam
555-12-1212	Dewey
987-65-4321	Semmelweis
987-65-4321	Pasteur
123-54-3210	Hammurabi

Converting a Table with Multivalued Dependencies into DKNF

LastName	Major	Hobby
Jones	Library and Information Science	Surfing the Internet
Jones	Library and Information Science	Chess
Jones	Public Affairs	Surfing the Internet
Jones	Public Affairs	Chess
Lee	Library and Information Science	Photography
Lee	Library and Information Science	Stamp collecting
Ruiz	Pre-Medicine	Surfing the Internet
Ruiz	Pre-Medicine	Photography
Ruiz	Biochemistry	Surfing the Internet
Ruiz	Biochemistry	Photography
Smith	Pre-Law	Playing poker

- If we analyze Table 14 from the DKNF point of view, the first thing we see is that the key in the table is composite.
- It is the triple, LastName-Major-Hobby.
- But in an intuitive sense, the natural key would be just LastName, since we know that there are just four students involved and that we are trying to present data about their majors and their hobbies.
- The complications arise because some of the students have more than one major and/or more than one hobby.
- Another way of putting it is that the complications of the table arise from the fact that we are trying to display, in just one table, more information than it is practicable to display in a single table.

- From the DKNF point of view, we have two constraints.
- One constraint concerns the natural key, LastName, and the attribute, Major.
- If we set up one table that houses these attributes, then the constraint on Major will be a logical consequence of the key, LastName.
- The other constraint concerns the natural key, LastName, and the attribute, Hobby.
- If we set up a second table that houses these attributes, then the constraint on Hobby will be a logical consequence of the key, LastName.
- Having set up these two tables, we will find that there is nothing further to be done.

These are the tables presented in Here we have arrived at these same tables by considering how the information in Table14 (the same information as in Table7) should be re-arranged from the DKNF point of view.

LastName	Major
Jones	Library and Information Science
Jones	Public Affairs
Lee	Library and Information Science
Ruiz	Pre-Medicine
Ruiz	Biochemistry
Smith	Pre-Law

LastName	Hobby
Jones	Surfing the Internet
Jones	Chess
Lee	Photography
Lee	Stamp collecting
Ruiz	Surfing the Internet
Ruiz	Photography
Smith	Playing poker

Example2

Chair	Subject	Student
mathematics	MATH20	Nursultan
informatics	CS150	Madiyar
informatics	CS103	Gerda
informatics	CS104	Farkhat
chemistry	CH894	Gyuzal
physics	PH654	Elmira

Chair	Student
mathematics	Nursultan
informatics	Madiyar
informatics	Gerda
informatics	Farkhat
chemistry	Gyuzal
physics	Elmira

Subject	Student
MATH20	Nursultan
CS150	Madiyar
CS103	Gerda
CS104	Farkhat
CH894	Gyuzal
PH654	Elmira

Chair	Subject
mathematics	MATH20
informatics	CS150
informatics	CS103
informatics	CS104
chemistry	CH894
physics	PH654

Single-Theme Tables and the DKNF

- What has the preceding discussion shown us?
- We have seen that when we analyze, from the DKNF point of view, tables with various kinds of problems, we find--again and again--that the solutions to the problems consist in turning a complicated, multi-theme table into sets of single-theme tables, tables which satisfy the requirements of the DKNF. If on the other hand, we analyze a complicated, problem-laden table from the point of view of turning it into a set of single-theme tables, we thereby achieve--again and again--a set of tables that satisfy the requirements of the DKNF.
- In short, sets of single-theme tables will almost always be sets of tables in DKNF and, as such, will be sets of tables that avoid the various kinds of anomalies that we want to avoid.