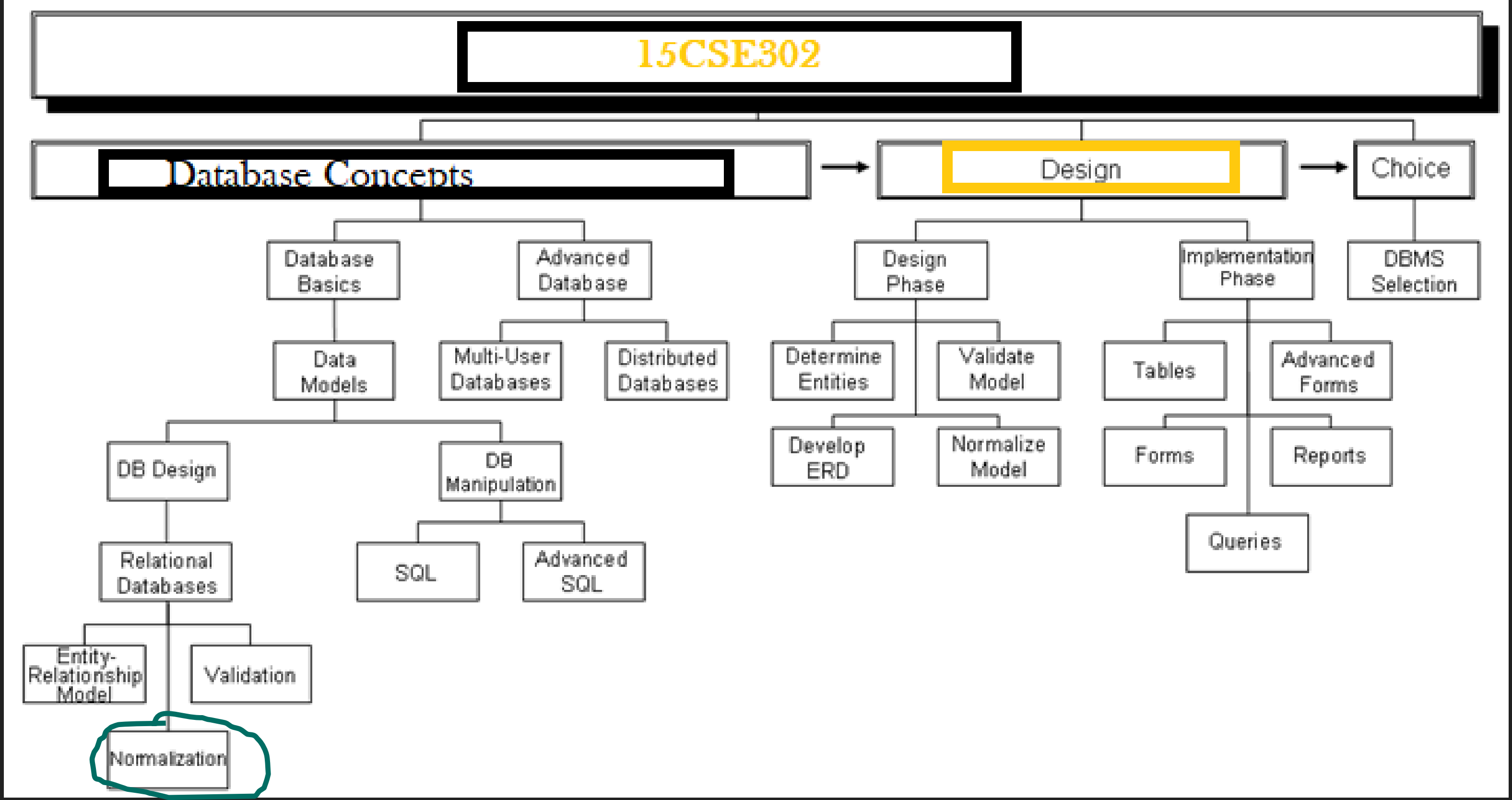
15CSE302 Database Management Systems
Lecture 21 Functional Dependency session 3

B.Tech /III Year CSE/V Semester   L T P C 2 0 2 3

DBMS Team
Dr G Jeyakumar
Bindu K R
Dr Priyanka Kumar
R. Manjusha
Department of CSE
Amrita School of Engineering

Slides Courtesy : Carlos Alvarado,
San Jose State University

Syllabus


15CSE302
Database Concepts
Design
Choice
Database Basics
Advanced Database
Design Phase
Implementation Phase
DBMS Selection
Data Models
Multi-User Databases
Distributed Databases
Determine Entities
Validate Model
Tables
Advanced Forms
DB Design
DB Manipulation
Develop ERD
Normalize Model
Forms
Reports
Queries
Relational Databases
SQL
Advanced SQL
Entity-Relationship Model
Validation
Normalization

Brief Recap of Previous Lecture

- Transitive Dependency
- Third Normal Form

Today we'll discuss

Closure of FDs

Closure of a Set of Functional Dependencies

- Given a set F set of functional dependencies, there are certain other functional dependencies that are logically implied by F.
  - If $A \rightarrow B$ and $B \rightarrow C$, then we can infer that $A \rightarrow$ C etc.
- The set of **all** functional dependencies logically implied by F is the **closure** of F.
- We denote the *closure* of F by $\mathbf{F^+}$.

Closure of a Set of Functional Dependencies

- $R = (A, B, C, G, H, I)$
  $F = \{ A \rightarrow B \quad A \rightarrow C \quad CG \rightarrow H \quad CG \rightarrow I \quad B \rightarrow H\}$
- Some members of F^+
  - $A \rightarrow H$ by transitivity from $A \rightarrow B$ *and* $B \rightarrow H$
  - $AG \rightarrow I$ by augmenting $A \rightarrow C$ with G, to get $AG \rightarrow CG$ and then transitivity with $CG \rightarrow I$
  - $CG \rightarrow HI$ by augmenting $CG \rightarrow I$ to infer $CG \rightarrow CGI$,

    and augmenting of $CG \rightarrow H$ to infer $CGI \rightarrow HI$,

    and then transitivity

Closure of a Set of Functional Dependencies

- To compute the closure of a set of functional dependencies F:

  $F^+ = F$

  repeat

  for each functional dependency f in F^+

  apply reflexivity and augmentation rules on f

  add the resulting functional dependencies to F^+

  for each pair of functional dependencies f_1 and f_2 in F^+

  if f_1 and f_2 can be combined using transitivity

  then add the resulting functional dependency to F^+

  until F^+ does not change any further

- **NOTE:** We shall see an alternative procedure for this task later

Closure of Attribute Sets

- Given a set of attributes α, define the ***closure*** of α **under** *F* (denoted by α^+) as the set of attributes that are functionally determined by α under *F*
- Algorithm to compute α^+, the closure of α under *F*

  result := α;
  while (changes to *result*) **do**
      for each $\beta \rightarrow \gamma$ **in** *F* **do**
        begin
          if $\beta \subseteq$ *result* **then** *result* := *result* $\cup \gamma$
        end

Closure of Attribute Sets

- $R = (A, B, C, G, H, I)$
- $F = \{A \to B \;\; A \to C \;\; CG \to H \;\; CG \to I \;\; B \to H\}$
- $(AG)^+$
  1. result = AG
  2. result = ABCG (A $\to$ C and A $\to$ B)
  3. result = ABCGH (CG $\to$ H and CG $\subseteq$ AGBC)
  4. result = ABCGHI (CG $\to$ I and CG $\subseteq$ AGBCH)
- Is AG a candidate key?

1. Is AG a super key?
   1. Does AG $\to$ R? == Is R $\supseteq$ (AG)+
2. Is any subset of AG a superkey?
   1. Does A $\to$ R? == Is R $\supseteq$ (A)+
   2. Does G $\to$ R? == Is R $\supseteq$ (G)+

In general: check for each subset of size n-1

Uses of Attribute Closure

There are several uses of the attribute closure algorithm:

- Testing for superkey:
    - To test if α is a superkey, we compute α^+, and check if α^+ contains all attributes of R.
- Testing functional dependencies
    - To check if a functional dependency $\alpha \to \beta$ holds (or, in other words, is in F^+), just check if $\beta \subseteq \alpha^+$.
    - That is, we compute α^+ by using attribute closure, and then check if it contains β.
    - Is a simple and cheap test, and very useful
- Computing closure of F
    - For each $\gamma \subseteq R$, we find the closure γ^+, and for each $S \subseteq \gamma^+$, we output a functional dependency $\gamma \to S$.

Closure of Attribute Sets

- Consider a relation R (A , B , C , D , E , F , G) with the functional dependencies-
- $A \longrightarrow BC$
- $BC \longrightarrow DE$
- $D \longrightarrow F$
- $CF \longrightarrow G$

Closure of Attribute Sets

Closure of attribute A-

$A^+ = \{ A \}$

$= \{ A , B , C \}$ (Using $A \rightarrow BC$)

$= \{ A , B , C , D , E \}$ (Using $BC \rightarrow DE$)

$= \{ A , B , C , D , E , F \}$ (Using $D \rightarrow F$)

$= \{ A , B , C , D , E , F , G \}$ (Using $CF \rightarrow G$)

Thus,

$A^+ = \{ A , B , C , D , E , F , G \}$

Closure of Attribute Sets

Closure of attribute A-

$A^+ = \{ A \}$
$= \{ A, B, C \}$ (Using $A \rightarrow BC$)
$= \{ A, B, C, D, E \}$ (Using $BC \rightarrow DE$)
$= \{ A, B, C, D, E, F \}$ (Using $D \rightarrow F$)
$= \{ A, B, C, D, E, F, G \}$ (Using $CF \rightarrow G$)
Thus,
$A^+ = \{ A, B, C, D, E, F, G \}$

Closure of Attribute Sets

Closure of attribute D-

$D^+ = \{ D \}$

$= \{ D , F \}$ (Using $D \rightarrow F$)

We can not determine any other attribute using attributes D and F contained in the result set.

Thus,

$D^+ = \{ D , F \}$

Example

TABLE_BOOK_DETAIL

Book ID	Genre ID	Genre Type	Price
1	1	Gardening	25.99
2	2	Sports	14.99
3	1	Gardening	10.00
4	3	Travel	12.99
5	2	Sports	17.99

Example

TABLE_BOOK

Book ID	Genre ID	Price
1	1	25.99
2	2	14.99
3	1	10.00
4	3	12.99
5	2	17.99

TABLE_GENRE

Genre ID	Genre Type
1	Gardening
2	Sports
3	Travel

Example

Project Code	Project Title	Project Manager	Project Budget	Employee No.	Employee Name	Department No.	Department Name	Hourly Rate
PC010	Pensions System	M Phillips	24500	S10001	A Smith	L004	IT	22.00
PC010	Pensions System	M Phillips	24500	S10030	L Jones	L023	Pensions	18.50
PC010	Pensions System	M Phillips	24500	S21010	P Lewis	L004	IT	21.00
PC045	Salaries System	H Martin	17400	S10010	B Jones	L004	IT	21.75
PC045	Salaries System	H Martin	17400	S10001	A Smith	L004	IT	18.00
PC045	Salaries System	H Martin	17400	S31002	T Gilbert	L028	Database	25.50
PC045	Salaries System	H Martin	17400	S13210	W Richards	L008	Salary	17.00
PC064	HR System	K Lewis	12250	S31002	T Gilbert	L028	Database	23.25
PC064	HR System	K Lewis	12250	S21010	P Lewis	L004	IT	17.50
PC064	HR System	K Lewis	12250	S10034	B James	L009	HR	16.50

Boyce and Codd Normal Form (BCNF)

- **“A relation is in BCNF, if and only if, every determinant is a candidate key.”**

Boyce and Codd Normal Form (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.**
- for any dependency $A \rightarrow B$, A should be a **super key**.

In simple words, it means, that for a dependency

$A \rightarrow B$

A cannot be a **non-prime attribute**, if B is a **prime attribute**.

Boyce and Codd Normal Form (BCNF)

- **Boyce and Codd Normal Form** is a higher version of the Third Normal form.
- This form deals with certain type of anomaly that is not handled by 3NF.
- **A 3NF table which does not have multiple overlapping candidate keys is said to be in BCNF.**

References

- Hillyer Mike, MySQL AB. An Introduction to Database Normalization, http://dev.mysql.com/tech-resources/articles/intro-to-normalization.html, accessed October 17, 2006.

- Microsoft. Description of the database normalization basics, http://support.microsoft.com/kb/283878 , accessed October 17, 2006.

- Wikipedia. Database Normalization. http://en.wikipedia.org/wiki/Database_normalization.html , accessed October 17, 2006.
  https://www.db-book.com/db6/index.html
- https://www.youtube.com/watch?v=mfVCesoMaGA&list=PLroEs25KGvwzmvIxYHRhoGTz9w8
  LeXek0&index=22

Summary

- Normalization basics
- Anomalies

Next Lecture

Functional dependency

Thank You

Happy to answer any questions ! ! !