

W22- CSC 430/530 DBMS/DT

Lecture 16: Normalization for Relational Databases

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

2

Informal Design Guidelines for Relational Databases (2)

- We first discuss informal guidelines for good relational design
 - Then we discuss formal concepts of functional dependencies and normal forms
 - - 1NF (First Normal Form)
 - - 2NF (Second Normal Form)
 - - 3NF (Third Normal Form)
 - - BCNF (Boyce-Codd Normal Form) – Next class.
- } Today

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 10- 6

6

1.1 Semantics of the Relation Attributes

- **GUIDELINE 1:** Informally, each tuple in a relation should represent one entity or relationship instance. (Applies to individual relations and their attributes).
 - Attributes of different entities (EMPLOYEEs, DEPARTMENTs, PROJECTs) should not be mixed in the same relation
 - Only foreign keys should be used to refer to other entities
 - Entity and relationship attributes should be kept apart as much as possible.
- **Bottom Line:** *Design a schema that can be explained easily relation by relation. The semantics of attributes should be easy to interpret.*

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 10- 7

7

Figure 10.1 A simplified COMPANY relational database schema

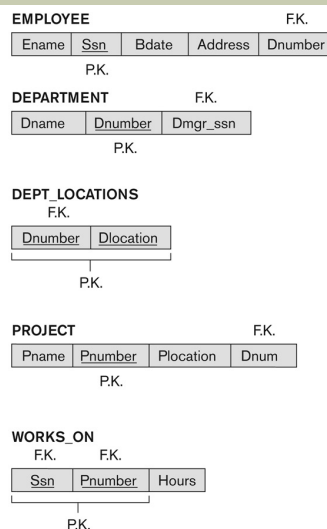


Figure 10.1
A simplified COMPANY
relational database
schema.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 10- 8

8

Guideline to Redundant Information in Tuples and Update Anomalies

■ GUIDELINE 2:

- Design a schema that does not suffer from the insertion, deletion and update anomalies.
- If there are any anomalies present, then note them so that applications can be made to take them into account.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 10- 15

15

1.3 Null Values in Tuples

■ GUIDELINE 3:

- Relations should be designed such that their tuples will have **as few NULL values** as possible
- Attributes that are NULL frequently could be placed in separate relations (with the primary key)
- Reasons for nulls:
 - Attribute not applicable or invalid
 - Attribute value unknown (may exist)
 - Value known to exist, but unavailable

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 10- 16

16

1.4 Spurious Tuples

- Bad designs for a relational database may result in erroneous results for certain JOIN operations
- The "lossless join" property is used to guarantee meaningful results for join operations
 - If R is split into R_1 and R_2 , for this decomposition to be lossless then $R_1 \bowtie R_2 = R$
- GUIDELINE 4:
 - The relations should be designed to satisfy the lossless join condition.
 - No spurious tuples should be generated by doing a natural-join of any relations.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 10- 17

17

Spurious Tuples (2)

- There are two important properties of decompositions:
 - a) Non-additive or losslessness of the corresponding join
 - b) Preservation of the functional dependencies.
- Note that:
 - Property (a) is extremely important and *cannot* be sacrificed.
 - Property (b) is less stringent and may be sacrificed.

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 10- 18

18

Normalization of Relations

- **Normalization:**

- The process of decomposing unsatisfactory "bad" relations by breaking up their attributes into smaller relations

- **Normal form:**

- Condition using keys and FDs of a relation to certify whether a relation schema is in a particular normal form

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 10- 30

30

Practical Use of Normal Forms

- **Normalization** is carried out in practice so that the resulting designs are of high quality and meet the desirable properties
 - The practical utility of these normal forms becomes questionable when the constraints on which they are based are *hard to understand* or to *detect*
- The database designers *need not* normalize to the highest possible normal form
 - (usually up to 3NF, BCNF or 4NF)
- **Denormalization:**
 - The process of storing the join of higher normal form relations as a base relation—which is in a lower normal form

Copyright © 2007 Ramez Elmasri and Shamkant B. Navathe

Slide 10- 32

32

Definitions of Keys and Attributes Participating in Keys (2)

- If a relation schema has more than one key, each is called a **candidate** key.
 - One of the candidate keys is *arbitrarily* designated to be the **primary key**, and the others are called **secondary keys**.
- A **Prime attribute** must be a member of *some* candidate key
- A **Nonprime attribute** is not a prime attribute—that is, it is not a member of any candidate key.