An Introduction to Database Systems

chapter 13. Further Normalization II : Higher Normal Forms

spark@dblab.sogang.ac.kr

13.1 Introduction

- Fourth and fifth normal form

Generalization of FDs

- fourth normal form
  - makes use of a *multivalued dependency*(MVD)

Generalization of MVDs

- fifth normal form
  - makes use of a *join dependency*(JD)

spark@dblab.sogang.ac.kr

13.2 Multivalued dependencies and fourth normal form

- **Unnormalized relation UCTX**
  - *courses, texchers, texts*

UTCX

COURSE	TEACHERS	TEXTS
Physics	Prof. Green Prof. Brown	Basic Mechanics Principles of Optics
Math	Prof. Green	Basic Mechanics Vector Analysis Trigonometry

Assumption : there are no FDs

Each tuple consists of

a *course* name

a repeating group of *teacher* names

a repeating group of *text* names

For a given course,

there can exist any number

of corresponding teachers and

any number of corresponding text

spark@dblab.sogang.ac.kr

13.2 Multivalued dependencies and fourth normal form

- Equivalent normalized form
  - "flattening" the structure

CTX

COURSE	TEACHER	TEXT
Physics	Prof. Green	Basic Mechanics
Physics	Prof. Green	Principles of Optics
Physics	Prof. Brown	Basic Mechanics
Physics	Prof. Brown	Principles of Optics
Math	Prof. Green	Basic Mechanics
Math	Prof. Green	Vector Analysis
Math	Prof. Green	Trigonometry

CTX is in BCNF, since it is "all key"

▶ **Meaning of the normalized version CTX :**

A tuple {COURSE:*c*, TEACHER;*t*, TEXT:*x*} appears in the relation if and only if course c can be taught t and uses text *x*

✱ For a given course, all possible combinations of teacher and text appear ;

```
IF tuples {c, t1, x1}, {c, t2, x2} both appear
Then tuples {c, t1, x2}, {c, t2, x1} both appear also
```


spark@dblab.sogang.ac.kr

13.2 Multivalued dependencies and fourth normal form

- Relation CTX involves a good deal of *redundancy*
  - ⇨ *update anomalies*
  - example :
    - to add the information that the physics course can be taught by a new teacher, it is necessary to create *two* new tuples
  - The difficulties are caused by the fact that teachers and texts are completely independent of one another
  - it is easy to see that matters would be improved if CTX were replaced by its *projections* on {COURSE, TEACHER} and {COURSE, TEXT}

spark@dblab.sogang.ac.kr

13.2 Multivalued dependencies and fourth normal form

- **Two projections CT and CX**
  - **both "all key" and BCNF**

CT

COURSE	TEACHER
Physics	Prof. Green
Physics	Prof. Brown
Math	Prof. Green

CX

COURSE	TEXT
Physics	Basic Mechanics
Physics	Principles of Optics
Math	Basic Mechanics
Math	Vector Analysis
Math	Trigonometry

★ **the decomposition is indeed *correct* and *desirable***

★ **the decomposition cannot be made on the basis of functional dependencies, because there are no FDs in the relation.**

New kind of dependency, *multivalued dependency(MVD)*

spark@dblab.sogang.ac.kr

13.2 Multivalued dependencies and fourth normal form

- **MVD**
  - **there are two MVDs in CTX**

    COURSE $\twoheadrightarrow$ TEACHER

    COURSE $\twoheadrightarrow$ TEXT

  - **MVD A $\twoheadrightarrow$ B**
    - **"B is *multidependent* on A" or "A *determines* B"**

  - **the meaning of MVD *COURSE* $\twoheadrightarrow$ *TEACHER***
    - **Although a course does not have a *single* corresponding teacher(i.e., the functional dependence COURSE → TEACHER does not hold), each course have a *well-defined set* of corresponding teachers.**
    - **"well-defined"**

      for a given course *c* and a given text *x*, the set of teachers *t* matching the pair(*c*,*x*) in CTX depends on the value *c* alone

spark@dblab.sogang.ac.kr

13.2 Multivalued dependencies and fourth normal form

- **Definition of MVD(Multivalued Dependence)**

  Let R be a relation, and let A, B, C be arbitrary subsets of the set of attributes of R.

  We say that B is *multidependent* on A in symbols,

  $A \twoheadrightarrow B$

  if and only if the set of *B-values* matching (A-value, C-value) pair in R depends *only* on the A-value and is independent of the C-value

- **given the relation R{A,B,C}**
  - **the MVD $A \twoheadrightarrow B$ holds if and only if the MVD $A \twoheadrightarrow C$ also holds.**
  - **A single joint statement**

    $A \twoheadrightarrow B \mid C$

    COURSE $\twoheadrightarrow$ TEACHER | TEXT

spark@dblab.sogang.ac.kr

13.2 Multivalued dependencies and fourth normal form

- **Non-loss decomposition**
  - **Theorem(Fagin)**

    Let R{A, B, C} be a relation, where A, B, and C are sets of attributes. Then R is equal to the join of its projections on {A, B} and {A, C} if and only if R satisfies the MVD $A \twoheadrightarrow B \mid C$.

- **Fourth Normal Form**
  - **Relation R is in 4NF if and only if , whenever there exist subsets A and B of the attributes of R such that the (nontrivial) MVD $A \twoheadrightarrow B$ is satisfied, then all attributes of R are also funtionally dependent on A.**

13.2 Multivalued dependencies and fourth normal form

- **4NF**
  - **4NF is strictly stronger than BCNF**
  - **any relation can be nonloss-decomposed into an equivalent collection of 4NF relations**

  - **the only nontrivial dependencies(FDs or MVDs) in R are of the form K→X (i.e., a functional dependency from a candidate key K to some other attribute X)**
  - **Equivalently, R is in 4NF if it is BCNF and all MVDs in R are in fact FDs "out of candidate keys"**

  ∴ Therefore, 4NF implies BCNF
    - **relation CTX in not in 4NF, since it involves an MVD that is not an FD at all, let alone an FD "out of candidate keys"**
    - **two projections CT anc CX are both in 4NF**

spark@dblab.sogang.ac.kr

13.3 Join dependencies and fifth normal form

- **Decomposition process**
  - **replacement of a relation by two of its projections**

- **n-decomposable**
  - **the relation can be nonloss-decomposed into *n* projections but not into any number of projections less than *n***
  - ***example***
    - **a relation that can be nonloss-decomposed into two projections :**

    ⇒ "2-decomposable"

spark@dblab.sogang.ac.kr

13.3 Join dependencies and fifth normal form

SPJ

S#	P#	J#
S1	P1	J2
S1	P2	J1
S2	P1	J1
S1	P1	J1

SP

S#	P#
S1	P1
S1	P2
S2	P1

PJ

P#	J#
P1	J2
P2	J1
P1	J1

JS

J#	S#
J2	S1
J1	S1
J1	S2

Join over p#

SPJ

S#	P#	J#
S1	P1	J2
S1	P2	J1
S2	P1	J1
S2	P1	J2
S1	P1	J1

spurious

Join over J#, P3

Original SPJ

spark@dblab.sogang.ac.kr

13.3 Join dependencies and fifth normal form

- The result of the first join is to produce a copy of the original SPJ relation plus one additional(spurious) tuple

- 3-decomposability of relation SPJ could be more fundamental, time-independent property, if that relation satisfies a certain time-independent constraint
  - "SPJ is equal to the join of its three projections SP, PJ, JS"

  is equivalent to

IF	the pair (s1,p1) appears in SP
AND	the pair (p1,j1) appears in PJ
AND	the pair (j1,s1) appears in JS
THEN	the triple (s1, p1, j1) appears in SPJ

  - because the triple (s1, p1, j1) obviously appears in the join of SP, PJ, and JS

spark@dblab.sogang.ac.kr

13.3 Join dependencies and fifth normal form

- **converse of this statement**

  if (s1,p1,j1) appears in SPJ

  then (s1,p1) appears in projection SP,

  (p1, j1) appears in projection PJ, and

  (j1, s1) appears in projection JS

  is obviously true for any degree-3 relation SPJ.

  since (s1, p1) appears in SP if and only if (s1, p1, j2) appears in SPJ for some j2,

  (p1, j1) appears in PJ if and only if (s2, p1, j1) appears in SPJ for some s2, and

  (j1, s1) appears in JS if and only if (s1, p2, j1) appears in SPJ for some p2,

  IF (s1, p1, j2), (s2, p1, j1), (s1, p2, j1) appear in SPJ

  THEN (s1, p1, j1) appears in SPJ also.

  - **The cyclic nature of that constraint**
  - ***A relation will be n-decomposable for some n > 2 if and only if it satisfies some such cyclic constraint.***

spark@dblab.sogang.ac.kr

13.3 Join dependencies and fifth normal form

- Constraint 3D(for 3-decomposable)
  - time-independent constraint for the relation
- What does Constraint 3D mean in real-world terms ?

  (a) Smith supplies monkey wenches, and

  (b) monkey wrenches are used in the Manhattan project, and

  (c) Smith supplies the Manhattan project,

  then

  (d) Smith supplies monkey wrenches to the Manhattan project
- In chap 1, "the connection trap"

- In the particular case, we are saying *there is no such trap -* because there is an additional real-world constraint in effect, *namely Constraint 3D,* that makes the inference valid in this special case.

spark@dblab.sogang.ac.kr

13.3 Join dependencies and fifth normal form

- *Join Dependency(JD)*
  - **Constraint 3D is satisfied if and only if the relation concerned is equal to the join of certain of its projections**

  - **A JD is a constraint on the relation concerned, just as an MVD or an FD is a constraint on the relation concerned.**

- **Definition of Join Dependency**
  - **Let R be a relation, and let A, B, …, Z be arbitrary subsets of the set of attributes of R. Then we say that R satisfies the JD**

    *** (A, B, …, Z)**

    if and only if R is equal to the join of its projections on A, B, …, Z

spark@dblab.sogang.ac.kr

13.3 Join dependencies and fifth normal form

- Example of update problems in SPJ

SPJ

S#	P#	J#
S1	P1	J2
S1	P2	J1

- If (s2, p1, j1) is inserted, (s1, p1, j1) must also be inserted
- Yet converse is not true

SPJ

S#	P#	J#
S1	P1	J2
S1	P2	J1
S2	P1	J1
S1	P1	J1

- Can delete (s2, p1, j1) without side effects
- If (s1, p1, j1) deleted, another tuple must also be deleted(which?)

spark@dblab.sogang.ac.kr

13.3 Join dependencies and fifth normal form

- **Fagin's Theorem**
  - **R{A, B, C} can be nonloss-decomposed into its projections on {A, B} and {A, C} if and only if the MVDs $A \twoheadrightarrow B \mid C$.**

can be restated as follows

  - **R{A, B, C} satisfies the JD *(AB, AC) if and only if it satisfies the MVDs $A \twoheadrightarrow B \mid C$.**
  - **The MVD is a special case of a JD(generalization of MVDs)**
  - **most general form of dependency**

    (decomposed via projection and reconstructed via join)

spark@dblab.sogang.ac.kr

13.3 Join dependencies and fifth normal form

- **Relation SPJ**
  - **4NF but involve a JD $\Rightarrow$ decompose into the projections specified by a JD**

- **Definition of Fifth Normal Form(5NF)**
  - **A relation R is in 5NF - also called *projection-join normal form(PJ/NF)* - if and only if every join dependency in R is implied by the candidate keys of R**

spark@dblab.sogang.ac.kr

13.3 Join dependencies and fifth normal form

- **Example**
  - **supplier relation S has two candidate keys, S# and SNAME**
  - **relation S satisfies several join dependencies, for example**
    - *** ({S#, SNAME, STATUS}, {S#, CITY})**
    - **This JD is implied by the fact that *S# is a candidate key*.**
  - **Relation S also satisfies join dependency**
    - *** ({S#, SNAME}, {S#, STATUS}, {SNAME, CITY})**
    - **This JD is implied by the fact that *S# and SNAME are both candidate keys***

spark@dblab.sogang.ac.kr

13.4 The Normalization Procedure Summarized

- Technique of nonloss decomposition
  - basic idea

  - Given some first normal form relation R and some list of constraints(FDs, MVDs, and JDs) that apply to R,
  - we systematically reduce R to a collection of smaller relations that are equivalent to R in a certain well-defined sense but are also in some way more desirable than R.

spark@dblab.sogang.ac.kr

13.4 The Normalization Procedure Summarized

- **Overall process as a set of rules**
  1. **Take projections of the original 1NF relation to eliminate *any functional dependencies that are not irreducible***
  2. **Take projections of those 2NF relations to eliminate any *transitive* functional dependencies**
  3. **Take projections of those 3NF relations to eliminate any remaining functional dependencies in which the determinant is *not a candidate key***
  4. **Take projections of those BCNF relations to eliminate any multivalued dependencies that are not also functional dependencies**
  5. **Take projections of those 4NF relations to eliminate any join dependencies that are not implied by the candidate keys**

spark@dblab.sogang.ac.kr

13.4 The Normalization Procedure Summarized

- **Summary**
  1. The process of taking projections at each step must be done in a nonloss way and in a dependency-preserving way
  2. Definitions of BCNF, 4NF, 5NF
     - A relation R is in BCNf if and only if every FD in R is implied by the candidate keys of R
     - A relation R is in 4NF if and only if every MVD in R is implied by the candidate keys of R
     - A relation R is in 5NF if and only if every JD in R is implied by the candidate keys of R
  3. The overall objectives of the normalization process
     - to eliminate certain kinds of redundancy
     - to avoid certain update anomalies
     - to produce a design that is a "good" representation of the real world
     - to simplify the enforcement of certain integrity constraints

spark@dblab.sogang.ac.kr

13.7 Other Normal Forms

- Domain-Key Normal Form(DK/NF)
  - A relation R is said to be in DK?NF if and only if every constraint on R is a logical consequence of the domain constraints and key constraints that apply to R
  - domain constraint
    - constratint to the effect that values of a given attribute are taken form some prescribed domain
  - key constraint
    - constraint to the effect that a certain attribute or attribute combination constitutes a candidate key
- "Restriction-union" normal form

spark@dblab.sogang.ac.kr