8.5 Multivalued Dependencies

- consider the following unnormalized relation, CIT, for courses, instructors for courses, and reference text books for the courses:

course	*instructor*	*text*
S/W Engineering	Bartson	Intro to S/W Engineering
S/W Engineering	Bristow	Data Structures & Algorithms
S/W Engineering	Bartson	Data Structures & Algorithms
S/W Engineering	Bristow	Intro to S/W Engineering
Computer Programming	Johnson	Data Structures & Algorithms
Computer Programming	Johnson	Introduction to C++
Computer Programming	Johnson	Object Oriented Techniques

- assume: for any course there can exist any number of instructors and texts; instructors and texts are independent (the same texts are used for any given offering); a given instructor or given text can be associated with any number of courses.

- We can observe that for a given course, all possible combinations of teacher and text must appear. Hence CIT has a considerable amount of redundancy leading to possible update anomalies. For example, if ‘Computer Programming’ is to be taught by another instructor, then it is necessary to create *three* new tuples, one for each of the three texts.
- It is obvious that no functional dependencies hold in CIT (other than trivial ones). Also, CIT is in BCNF since it is an ‘all key’ relation. But due to redundancy and possible update anomalies it would be desirable to perform some type of decomposition.
- Unfortunately, since there are no nontrivial functional dependencies, the standard normalization techniques based on functional dependencies do not apply.

- Consider the following ‘ad hoc’ decomposition of CIT:

CI

course	instructor
S/W Engineering	Bristow
S/W Engineering	Bartson
Computer Programming	Johnson

CT

course	text
S/W Engineering	Intro to S/W Engineering
S/W Engineering	Data Structures & Algorithms
Computer Programming	Data Structures & Algorithms
Computer Programming	Introduction to C++
Computer Programming	Object Oriented Techniques

- Both CI and CT are in BCNF (both are all key), the original relation CIT can be recovered by the join of CI and CT - a lossless join. Hence a decomposition is possible. These types of problem with BCNF relations (i.e. CIT) were well known but no theoretical basis for their decomposition was developed until around 1977.

  (note: CI $\rightarrow$ CI is “trivial” AND a superkey for CI, similarly for CT)

- The decomposition is possible based on a different type of dependency - **multivalued dependency** (MVD). All functional dependencies are MVDs, but the converse is not true. I.e. there can be MORE MVDs for a given set of attributes than FDs.
- There are two MVDs in CIT:

  course $\rightarrow\rightarrow$ instructor

  course $\rightarrow\rightarrow$ text

  - for example, the 2nd MVD is read as "text is multidependent on course" or "course multidetermines text".
- The interpretation of the 2nd MVD is as follows:

  a course does not have a single specific text, but rather a well defined set of texts. That is. for a given course c and a given instructor i, the set of texts t which match the pair (c, i) in CIT depends on the value of c alone - the value of i does not make any difference.

- More formally, a MVD may be defined as:

  Let R be a relation with X, Y, and Z as arbitrary subsets of attributes of R where R = { X, Y, Z }. Then X multidetermines Y,

  $$X \rightarrow\rightarrow Y$$

  if and only if the set of Y-values matching a given (X-value, Z-value) pair in R depends only on the X-value and is independent of the Z-value.

- Returning to the CIT relation, since C →→ T, then if

  (c1, i1, t1) and (c1, i2, t2) are valid tuples, then so are

  (c1, i1, t2) and (c1, i2, t1) valid tuples.

  i.e. ALL combinations of the "remainder" of the tuples are not only allowed, they MUST appear.

- This can be stated in a more formal manner as:

  $X \rightarrow\rightarrow Y$ holds in R if in any *legal* relation r(R) for all pairs of t_1 and t_2 such that $t_1[X] = t_2[X]$ then there exist t_3 and t_4 in r such that:

  $t_1[X] = t_2[X] = t_3[X] = t_4[X]$

  $t_1[Y] = t_3[Y]$

  $t_2[Y] = t_4[Y]$

  $t_3[R\text{-}Y] = t_2[R\text{-}Y]$

  $t_4[R\text{-}Y] = t_1[R\text{-}Y]$

- $X \rightarrow\rightarrow Y$ is said to be **trivial** if $Y \subseteq X$ or $Y \cup X = R$.
- let D be the set of functional and multivalued dependencies on R, then D^+ is the closure (the set of functional and multivalued dependencies logically implied by D).

8.5.1 Properties of Multivalued Dependencies

⇒ let $\delta,\alpha,\beta,\gamma$ be sets of attributes, $\subseteq R$

Complementation

if $\alpha \rightarrow\rightarrow \beta$ holds, then $\alpha \rightarrow\rightarrow R-\beta-\alpha$ holds.

e.g CIT relation, if $C \rightarrow\rightarrow T$, then if

(c1, i1, t1) and (c1, i2, t2) are valid tuples, then so are

(c1, i1, t2) and (c1, i2, t1).

Similarly, if $C \rightarrow\rightarrow I$, then if

(c1, i1, t1) and (c1, i2, t2) are valid tuples, then so are

(c1, i1, t2) and (c1, i2, t1)

Multivalued Augmentation

if $\alpha \rightarrow\rightarrow \beta$ holds and $\gamma \subseteq R$ and $\delta \subseteq \gamma$, then $\gamma\alpha \rightarrow\rightarrow \delta\beta$ also holds.

Multivalued Transitivity

if $\alpha \rightarrow\rightarrow \beta$ and $\beta \rightarrow\rightarrow \delta$ hold, then $\alpha \rightarrow\rightarrow \delta\text{-}\beta$ holds.

Replication

if $\alpha \rightarrow \beta$ holds, then $\alpha \rightarrow\rightarrow \beta$ holds

Coalescence

if $\alpha \rightarrow\rightarrow \beta$ holds and $\delta \subseteq \beta$ and there is a $\gamma \subseteq R$ and $\gamma \cap \beta = \varnothing$ and $\gamma \rightarrow \delta$ then $\alpha \rightarrow \delta$ holds.

Multivalued Union

if $\alpha \rightarrow\rightarrow \beta$ and $\alpha \rightarrow\rightarrow \delta$ hold, then $\alpha \rightarrow\rightarrow \beta\delta$ holds.

Intersection

if $\alpha \rightarrow\rightarrow \beta$ and $\alpha \rightarrow\rightarrow \delta$ hold, then $\alpha \rightarrow\rightarrow \beta \cap \delta$ holds.

Difference

if $\alpha \rightarrow\rightarrow \beta$ and $\alpha \rightarrow\rightarrow \delta$ hold, then $\alpha \rightarrow\rightarrow \beta\text{-}\delta$ and $\alpha \rightarrow\rightarrow \delta\text{-}\beta$ hold.

♦ consider our previous relation scheme:

⇒ let LIST = (*course, teacher, hour, room, student, grade*), with the following functional dependencies assumed:

Dependency	Description
course→*teacher*	each course only has one teacher
hour, room→*course*	one course/room at same time
hour, teacher→*room*	teacher in only one room at a time
course, student→*grade*	each student has only one grade in each course
hour, student→*room*	student in one at the same time

- list some multivalued dependencies that can be obtained from LIST.
- you may use the properties (axioms) of multivalued dependencies to aid you.

  -All FDs are MVDs, so *course*→*teacher* implies *course*→→*teacher*

  -same for all other FDs.

complementation:

course$\rightarrow\rightarrow$teacher implies course$\rightarrow\rightarrow$(R-teacher-course),

course$\rightarrow\rightarrow$hour, room, student, grade

-FDs are really a special case of MVDs, where the number of values in β is 1.

-going back to the CIT relation, if C $\rightarrow\rightarrow$ T, then if there are 3 teachers teaching a given course and 2 textbooks for the course, one expects 6 tuples to be in CIT for that course.

-if C$\rightarrow$T, then there is only ONE teacher for the course. Hence, it's **guaranteed** that all teachers appear with all textbooks.

The Fourth Normal Form (4NF)

- similar to BCNF but for multivalued dependencies.
- **definition:**

  a relation scheme R is in 4NF with respect to a set of multivalued dependencies (D) if for all multivalued dependencies in D^+ of the form $X \rightarrow\rightarrow Y$ (where X, $Y \subseteq R$) at least one of the following holds:

  - $X \rightarrow\rightarrow Y$ is trivial
  - X is a superkey for R
- a scheme in 4NF is also in BCNF
- **algorithm for 4NF decomposition**:
  - given $R = \{ R_1, R_2, \ldots, R_n\}$ and a set D.
  - if R_i is not in 4NF, then:
    - let $X \rightarrow\rightarrow Y$ be a non trivial dependency that holds on R_i such that X is not a superkey and $X \cap Y = \emptyset$.
    - replace R_i with $(R_i - Y)$ and (X,Y) such that $X \rightarrow\rightarrow Y$ holds and $R_i = (R_i - Y) \cup (X, Y)$.

example:

- consider the previous LIST scheme :

⇒ let LIST = (*course, teacher, hour, room, student, grade*), with the following functional dependencies assumed:

Dependency	Description
$course \rightarrow teacher$	each course only has one teacher
$hour, room \rightarrow course$	one course/room at same time
$hour, teacher \rightarrow room$	teacher in only one room at a time
$course, student \rightarrow grade$	each student has only one grade in each course
$hour, student \rightarrow room$	student in one at the same time

- we also found some additional multivalued dependencies.
- use these multivalued dependencies to obtain a 4NF decomposition of LIST:

LIST = (*course, teacher, hour, room, student, grade*)

course→→teacher holds in LIST:

L2 = (*course, hour, room, student, grade*)

CT = (course, teacher)

hour, room→→course holds in L2

L3 = (hour, room, student, grade)

HRC = (hour, room, course)

CT = (course, teacher)

hour, student$\rightarrow\rightarrow$room holds in L3

-BUT, hour, student is a superkey.

-NO decomposition

Final decomposition:

L3 = (hour, room, student, grade)

HRC = (hour, room, course)

CT = (course, teacher)

-same MVDs as FDs, same decomposition

8.5.2 Lossless Join Decomposition

- let R = {R1, R2} and the set D of functional and multivalued dependencies which hold on R.
- this is a lossless join decomposition if one of the following holds:
    - $R1 \cap R2 \rightarrow\rightarrow R1$
    - $R1 \cap R2 \rightarrow\rightarrow R2$

8.5.3 Dependency Preservation

- let D be the set of functional and multivalued dependencies which hold on R
- the restriction of D to R_i is the set D_i consisting of:
    - all functional dependencies in D^+ that include only attributes of R_i.
    - all multivalued dependencies of the form $X \rightarrow\rightarrow Y \cap R_i$ where $X \subseteq R$ and $X \rightarrow\rightarrow Y$ is in D^+.

- the decomposition is dependency preserving with respect to D if for every set of relations $r_1(R_1)$, …, $r_n(R_n)$, where each r_i satisfies D_i, there exists r(R) that satisfies D for which $r_i = \Pi_{R_i}(r)$ for all i.

-if the schema R is decomposed into R_1 , R_2 , … R_i

-fill each relation $r_i(R_i)$ with data independently such that all MVDs that hold on each $r_i(R_i)$ (D^i, the restriction of D^+ to r_i) are NOT violated.

-IF it is possible that even though each of the multivalued depencies in D^i hold, when computing $r_1 \otimes r_2 \otimes \ldots \otimes r_i$ the resulting tuples do NOT satisfy D, dependencies were not preserved (difficult to determine).

-alternatively: dependency preserving, if the dependencies in D^i only allow data going into each r_i that came from valid data in r (i.e. data that satisfies D)

Example:

- R = (H, I, J, K, L, M)
- D = (H →→I, I →→LM, JK →→L}
- perform a 4NF decomposition on R and try to make use of the dependency preserving definition on previous page.

  In R, H→→I, but H is not a superkey.

  form HI = (H, I) and R2 = (H, J, K, L, M)

  In R2, JK→→L, but JK is not a superkey.

  form JKL = (J, K, L) and R3 = (H, J, K, M)

  In R3, I→→LM does not hold, but, H→→I and I→→LM leads to:

  H→→LM (difference rule). The restriction of H→→LM to R3 is H→→M.

  form HLM = (H, L, M), HI=(H, I), HM = (H, M) and R4 = (H, J, K)

- also prove that (HK) is the superkey

  -bad question, not possible to prove given only MVDs