Multivalued Dependencies & Fourth Normal Form (4NF)

Zaki Malik

October 28, 2008
A New Form of Redundancy

• Multivalued dependencies (MVD’s) express a condition among tuples of a relation that exists when the relation is trying to represent more than one many-many relationship.

• Then certain attributes become independent of one another, and their values must appear in all combinations.
Example

Drinkers(name, addr, phones, beersLiked)

• A drinker’s phones are independent of the beers they like.

• Thus, each of a drinker’s phones appears with each of the beers they like in all combinations.
  – If a drinker has 3 phones and likes 10 beers, then the drinker has 30 tuples
  – where each phone is repeated 10 times and each beer 3 times

• This repetition is unlike redundancy due to FD’s, of which name->addr is the only one.
Tuples Implied by Independence

If we have tuples:

<table>
<thead>
<tr>
<th>name</th>
<th>addr</th>
<th>phones</th>
<th>beersLiked</th>
</tr>
</thead>
<tbody>
<tr>
<td>sue</td>
<td>a</td>
<td>p1</td>
<td>b1</td>
</tr>
<tr>
<td>sue</td>
<td>a</td>
<td>p2</td>
<td>b2</td>
</tr>
<tr>
<td>sue</td>
<td>a</td>
<td>p2</td>
<td>b1</td>
</tr>
<tr>
<td>sue</td>
<td>a</td>
<td>p1</td>
<td>b2</td>
</tr>
</tbody>
</table>

Then these tuples must also be in the relation.
Another Example

The relation is Courses(Number, DeptName, Textbook, Professor).
- Each Course can have multiple required Textbooks.
- Each Course can have multiple Professors.
- Professors uses every required textbook while teaching a Course.

<table>
<thead>
<tr>
<th>Number</th>
<th>DeptName</th>
<th>Textbook</th>
<th>Professor</th>
</tr>
</thead>
<tbody>
<tr>
<td>4604</td>
<td>CS</td>
<td>FCDB</td>
<td>Ullman</td>
</tr>
<tr>
<td>4604</td>
<td>CS</td>
<td>SQL Made Easy</td>
<td>Ullman</td>
</tr>
<tr>
<td>4604</td>
<td>CS</td>
<td>FCDB</td>
<td>Widom</td>
</tr>
<tr>
<td>4604</td>
<td>CS</td>
<td>SQL Made Easy</td>
<td>Widom</td>
</tr>
</tbody>
</table>

The relation is in BCNF since there are no non-trivial FDs.

Is there any redundancy?
Definition of MVD

• A *multivalued dependency* (MVD) \( X \rightarrow\rightarrow Y \) is an assertion that if two tuples of a relation agree on all the attributes of \( X \), then their components in the set of attributes \( Y \) may be swapped, and the result will be two tuples that are also in the relation.
Definition of MVD

A multi-valued dependency (MVD or MD) is an assertion that two sets of attributes are independent of each other.

The *multi-valued dependency* $A_1A_2\ldots A_n \rightarrow B_1B_2\ldots B_m$ holds in a relation $R$ if in every instance of $R$, for every pair of tuples $t$ and $u$ in $R$ that agree on all the $A$’s, we can find a tuple $v$ in $R$ that agrees

1. with both $t$ and $u$ on $A$’s,
2. with $t$ on the $B$’s, and
3. with $u$ on all those attributes of $R$ that are not $A$’s or $B$’s.

<table>
<thead>
<tr>
<th>Number</th>
<th>DeptName</th>
<th>Textbook</th>
<th>Professor</th>
</tr>
</thead>
<tbody>
<tr>
<td>4604</td>
<td>CS</td>
<td>FCDB</td>
<td>Ullman</td>
</tr>
<tr>
<td>4604</td>
<td>CS</td>
<td>SQL Made Easy</td>
<td>Ullman</td>
</tr>
<tr>
<td>4604</td>
<td>CS</td>
<td>FCDB</td>
<td>Widom</td>
</tr>
<tr>
<td>4604</td>
<td>CS</td>
<td>SQL Made Easy</td>
<td>Widom</td>
</tr>
</tbody>
</table>
## Example

<table>
<thead>
<tr>
<th>Number</th>
<th>DeptName</th>
<th>Textbook</th>
<th>Professor</th>
</tr>
</thead>
<tbody>
<tr>
<td>4604</td>
<td>CS</td>
<td>FCDB</td>
<td>Ullman</td>
</tr>
<tr>
<td>4604</td>
<td>CS</td>
<td><strong>SQL Made Easy</strong></td>
<td>Ullman</td>
</tr>
<tr>
<td>4604</td>
<td>CS</td>
<td>FCDB</td>
<td><strong>Widom</strong></td>
</tr>
<tr>
<td>4604</td>
<td>CS</td>
<td><strong>SQL Made Easy</strong></td>
<td><strong>Widom</strong></td>
</tr>
<tr>
<td>2604</td>
<td>CS</td>
<td>Data Structures</td>
<td>Ullman</td>
</tr>
<tr>
<td>2604</td>
<td>CS</td>
<td>Data Structures</td>
<td><strong>Widom</strong></td>
</tr>
</tbody>
</table>

- **Number DeptName → Textbook** is an MD. For every pair of tuples $t$ and $u$ that agree on Number and DeptName, we can find a tuple $v$ that agrees
  1. with both $t$ and $u$ on Number and DeptName,
  2. with $t$ on Textbook, and with $u$ on Professor.

- **Number DeptName → Professor** is an MD. For every pair of tuples $t$ and $u$ that agree on Number and DeptName, we can find a tuple $v$ that agrees
  1. with both $t$ and $u$ on Number and DeptName,
  2. with $t$ on Professor, and with $u$ on Textbook.
Picture of MVD $X \rightarrow\rightarrow Y$

- Does $X \rightarrow Y$ imply $X \rightarrow\rightarrow Y$?
MVD Rules

- Every FD is an MVD
  - If $X \rightarrow Y$, then swapping $Y$’s between two tuples that agree on $X$ doesn’t change the tuples.
  - Therefore, the “new” tuples are surely in the relation, and we know $X \rightarrow \rightarrow Y$.

- Definition of keys depend on FDs and not MDs
Rules for Manipulating MDs

- *Trivial dependencies rule:* If $A \rightarrow B$ is an MD, then $A \rightarrow AB$ is also an MD.
Splitting Doesn’t Hold

• Like FD’s, we cannot generally split the left side of an MVD.

• But unlike FD’s, we cannot split the right side either --- sometimes you have to leave several attributes on the right side.
Another Example

- Consider a drinkers relation: `Drinkers(name, areaCode, phone, beersLiked, manf)`
- A drinker can have several phones, with the number divided between areaCode and phone (last 7 digits).
- A drinker can like several beers, each with its own manufacturer.
Example, Continued

• Since the areaCode-phone combinations for a drinker are independent of the beersLiked-manf combinations, we expect that the following MVD’s hold:

  name -›› areaCode phone
  name -›› beersLiked manf
Example Data

Here is possible data satisfying these MVD’s:

<table>
<thead>
<tr>
<th>name</th>
<th>areaCode</th>
<th>phone</th>
<th>beersLiked</th>
<th>manf</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sue</td>
<td>650</td>
<td>555-1111</td>
<td>Bud</td>
<td>A.B.</td>
</tr>
<tr>
<td>Sue</td>
<td>650</td>
<td>555-1111</td>
<td>WickedAle</td>
<td>Pete’s</td>
</tr>
<tr>
<td>Sue</td>
<td>415</td>
<td>555-9999</td>
<td>Bud</td>
<td>A.B.</td>
</tr>
<tr>
<td>Sue</td>
<td>415</td>
<td>555-9999</td>
<td>WickedAle</td>
<td>Pete’s</td>
</tr>
</tbody>
</table>

But we cannot swap area codes or phones my themselves. That is, neither name ->-> areaCode nor name ->-> phone holds for this relation.
Fourth Normal Form

• The redundancy that comes from MVD’s is not removable by putting the database schema in BCNF.

• There is a stronger normal form, called 4NF, that (intuitively) treats MVD’s as FD’s when it comes to decomposition, but not when determining keys of the relation.
4NF Definition

- A relation $R$ is in 4NF if whenever $X \rightarrowightarrow Y$ is a nontrivial MVD, then $X$ is a superkey.
  - Nontrivial means that:
    1. $Y$ is not a subset of $X$, and
    2. $X$ and $Y$ are not, together, all the attributes.

- Note that the definition of “superkey” still depends on FD’s only.
BCNF Versus 4NF

• Remember that every FD $X \rightarrow Y$ is also an MVD, $X \rightarrow\rightarrow Y$.

• Thus, if $R$ is in 4NF, it is certainly in BCNF.
  – Because any BCNF violation is a 4NF violation.

• But $R$ could be in BCNF and not 4NF, because MVD’s are “invisible” to BCNF.
Decomposition and 4NF

• If $X \rightarrow\rightarrow Y$ is a 4NF violation for relation $R$, we can decompose $R$ using the same technique as for BCNF.

  1. $XY$ is one of the decomposed relations.
  2. All but $Y - X$ is the other.
Example

**Drinkers**(*name*, *addr*, *phones*, *beersLiked*)

FD:    name -> addr
MVD’s: name ->> phones
       name ->> beersLiked

• Key is
  – {name, phones, beersLikedaed}.

• Which dependencies violate 4NF ?
  – All
Example, Continued

• Decompose using name -> addr:

1. Drinkers1(name, addr)
   ▪ In 4NF, only dependency is name -> addr.

2. Drinkers2(name, phones, beersLiked)
   ▪ Not in 4NF. MVD’s name -}> phones and name -}> beersLiked apply.
   ▪ Key ?
     ▪ No FDs, so all three attributes form the key.
Example: Decompose Drinkers2

• Either MVD name -->-> phones or name -->-> beersLiked tells us to decompose to:

  – Drinkers3(name, phones)
  – Drinkers4(name, beersLiked)
Relationships Among Normal Forms

- 4NF implies BCNF, i.e., if a relation is in 4NF, it is also in BCNF.
- BCNF implies 3NF, i.e., if a relation is in BCNF, it is also in 3NF.

<table>
<thead>
<tr>
<th>Property</th>
<th>3NF</th>
<th>BCNF</th>
<th>4NF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eliminates redundancy due to FDs</td>
<td>Maybe</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Eliminates redundancy due to MDs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Preserves FDs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Preserves MDs</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
First Normal Form: each attribute is atomic.