Lecture 08: Multivalued Dependencies

Outline

- Multivalued Dependencies
- Fourth Normal Form (4NF)
- General integrity Constraints
Another look at Heath’s Rule

Given $R(Z, W, Y)$ with FDs $F$

If $Z \rightarrow W \in F^+$, the

$$R = \pi_{Z, W}(R) \bowtie \pi_{Z, Y}(R)$$

What about an implication in the other direction? That is, suppose we have

$$R = \pi_{Z, W}(R) \bowtie \pi_{Z, Y}(R).$$

Q: Can we conclude anything about FDs on $R$? In particular, is it true that $Z \rightarrow W$ holds?

A: No!

We just need one counter example ...

$$R = \pi_{A, B}(R) \bowtie \pi_{A, C}(R)$$

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>$b_1$</td>
<td>$c_1$</td>
</tr>
<tr>
<td>a</td>
<td>$b_2$</td>
<td>$c_2$</td>
</tr>
<tr>
<td>a</td>
<td>$b_1$</td>
<td>$c_2$</td>
</tr>
<tr>
<td>a</td>
<td>$b_2$</td>
<td>$c_1$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>$b_1$</td>
</tr>
<tr>
<td>a</td>
<td>$b_2$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>A</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>$c_1$</td>
</tr>
<tr>
<td>a</td>
<td>$c_2$</td>
</tr>
</tbody>
</table>

Clearly $A \rightarrow B$ is not an FD of $R$. 
A concrete example

<table>
<thead>
<tr>
<th>course_name</th>
<th>lecturer</th>
<th>text</th>
</tr>
</thead>
<tbody>
<tr>
<td>Databases</td>
<td>Tim</td>
<td>Ullman and Widom</td>
</tr>
<tr>
<td>Databases</td>
<td>Fatima</td>
<td>Date</td>
</tr>
<tr>
<td>Databases</td>
<td>Tim</td>
<td>Date</td>
</tr>
<tr>
<td>Databases</td>
<td>Fatima</td>
<td>Ullman and Widom</td>
</tr>
</tbody>
</table>

Assuming that texts and lecturers are assigned to courses independently, then a better representation would in two tables:

<table>
<thead>
<tr>
<th>course_name</th>
<th>lecturer</th>
<th>course_name</th>
<th>text</th>
</tr>
</thead>
<tbody>
<tr>
<td>Databases</td>
<td>Tim</td>
<td>Databases</td>
<td>Ullman and Widom</td>
</tr>
<tr>
<td>Databases</td>
<td>Fatima</td>
<td>Databases</td>
<td>Date</td>
</tr>
</tbody>
</table>

Time for a definition!

**Multivalued Dependencies (MVDs)**

Let $R(Z, W, Y)$ be a relational schema. A multivalued dependency, denoted $Z \rightarrow W$, holds if whenever $t$ and $u$ are two records that agree on the attributes of $Z$, then there must be some tuple $v$ such that

1. $v$ agrees with both $t$ and $u$ on the attributes of $Z$,
2. $v$ agrees with $t$ on the attributes of $W$,
3. $v$ agrees with $u$ on the attributes of $Y$. 
A few observations

Note 1
Every functional dependency is multivalued dependency,

\[(Z \rightarrow W) \implies (Z \rightarrow W)\].

To see this, just let \(v = u\) in the above definition.

Note 2
Let \(R(Z, W, Y)\) be a relational schema, then

\[(Z \rightarrow W) \iff (Z \rightarrow Y),\]

by symmetry of the definition.

MVDs and lossless-join decompositions

Fun Fun Fact
Let \(R(Z, W, Y)\) be a relational schema. The decomposition \(R_1(Z, W), R_2(Z, Y)\) is a lossless-join decomposition of \(R\) if and only if the MVD \(Z \rightarrow W\) holds.
Proof of Fun Fun Fact

Proof of \((Z \rightarrow W) \implies R = \pi_{Z,W}(R) \bowtie \pi_{Z,Y}(R)\)

- Suppose \(Z \rightarrow W\).
- We know (from proof of Heath’s rule) that \(R \subseteq \pi_{Z,W}(R) \bowtie \pi_{Z,Y}(R)\). So we only need to show \(\pi_{Z,W}(R) \bowtie \pi_{Z,Y}(R) \subseteq R\).
- Suppose \(r \in \pi_{Z,W}(R) \bowtie \pi_{Z,Y}(R)\).
- So there must be a \(t \in R\) and \(u \in R\) with \(\{r\} = \pi_{Z,W}({\{t\}}) \bowtie \pi_{Z,Y}({\{u\}})\).
- In other words, there must be a \(t \in R\) and \(u \in R\) with \(t.Z = u.Z\).
- So the MVD tells us that then there must be some tuple \(v \in R\) such that
  - \(v\) agrees with both \(t\) and \(u\) on the attributes of \(Z\),
  - \(v\) agrees with \(t\) on the attributes of \(W\),
  - \(v\) agrees with \(u\) on the attributes of \(Y\).
- This \(v\) must be the same as \(r\), so \(r \in R\).

Proof of Fun Fun Fact (cont.)

Proof of \(R = \pi_{Z,W}(R) \bowtie \pi_{Z,Y}(R) \implies (Z \rightarrow W)\)

- Suppose \(R = \pi_{Z,W}(R) \bowtie \pi_{Z,Y}(R)\).
- Let \(t\) and \(u\) be any records in \(R\) with \(t.Z = u.Z\).
- Let \(v\) be defined by \(\{v\} = \pi_{Z,W}({\{t\}}) \bowtie \pi_{Z,Y}({\{u\}})\) (and we know \(v \in R\) by the assumption).
- Note that by construction we have
  - \(v.Z = t.Z = u.Z\),
  - \(v.W = t.W\),
  - \(v.Y = u.Y\).
- Therefore, \(Z \rightarrow W\) holds.
**Fourth Normal Form**

**Trivial MVD**
The MVD $Z \rightarrow W$ is **trivial** for relational schema $R(Z, W, Y)$ if
- $Z \cap W \neq \{\}$, or
- $Y = \{\}$.

**4NF**
A relational schema $R(Z, W, Y)$ is in 4NF if for every MVD $Z \rightarrow W$ either
- $Z \rightarrow W$ is a trivial MVD, or
- $Z$ is a superkey for $R$.

**Note:** $4NF \subset BCNF \subset 3NF \subset 2NF$

---

**General Decomposition Method Revisited**

**GDM++**
1. Understand your FDs and MVDs $F$ (compute $F^+$),
2. find $R(X) = R(Z, W, Y)$ (sets $Z$, $W$ and $Y$ are disjoint) with either
   - $Z \rightarrow W \in F^+$ or
   - $MVD Z \rightarrow W \in F^+$ violating a condition of desired NF,
3. split $R$ into two tables $R_1(Z, W)$ and $R_2(Z, Y)$
4. wash, rinse, repeat
Summary

We always want the lossless-join property. What are our options?

<table>
<thead>
<tr>
<th></th>
<th>3NF</th>
<th>BCNF</th>
<th>4NF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preserves FDs</td>
<td>Yes</td>
<td>Maybe</td>
<td>Maybe</td>
</tr>
<tr>
<td>Preserves MVDs</td>
<td>Maybe</td>
<td>Maybe</td>
<td>Maybe</td>
</tr>
<tr>
<td>Eliminates FD-redundancy</td>
<td>Maybe</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Eliminates MVD-redundancy</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

General integrity constraints

- Suppose that $C$ is some constraint we would like to enforce on our database.
- Let $Q_{-C}$ be a query that captures all violations of $C$.
- Enforce (somehow) that the assertion that is always $Q_{-C}$ empty.

Example

- $C = Z \rightarrow W$, and FD that was not preserved for relation $R(X)$,
- Let $Q_R$ be a join that reconstructs $R$,
- Let $Q'_R$ be this query with $X \leftrightarrow X'$ and
- $Q_{-C} = \sigma_{W \neq W'}(\sigma_{Z = Z'}(Q_R \times Q'_R))$
create view C_violations as ....

create assertion check_C
    check not (exists C_violations)