Dependencies and Normal Forms

Example 1

Present a real-life example (not an example of ABCD, etc), to demonstrate that the multiple-value dependencies and/or 4NF cannot be ignored in the database design. Explain why.

It is sufficient to given an example to show that without considering the multiple value dependencies, a table schema, even though in BCNF, may have some serious redundancies and/or update anomalies.

Consider the following table $\begin{aligned} &\mathbf{employee}(\ \mathsf{emp_id},\ \mathsf{project},\ \mathsf{dependent}\), \\ &\mathbf{with}\ \mathsf{one}\ \mathsf{multi-value}\ \mathsf{dependency}; \\ &\mathit{emp_id}\ \longrightarrow\ \mathit{project}|\mathit{dependend}. \end{aligned}$

Then the table is in BCNF but with serious redundency and update anomalies.

For an employee may have 4 dependents but work on 5 projects.

Example 2

Consider the following functional dependencies over the attribute set R = ABCDEFGH:

$$A \rightarrow E$$
 $BE \rightarrow D$
 $AD \rightarrow BE$ $BDH \rightarrow E$
 $AC \rightarrow E$ $F \rightarrow A$
 $E \rightarrow B$ $D \rightarrow H$
 $BG \rightarrow F$ $CD \rightarrow A$

- 1. Find a minimal cover of the given set of FDs.
- 2. Find a join loss-less, dependency preserving and 3NF decomposition of R.
- 3. Check if all the resulting relations in the previous step are in BCNF. If you find a schema that is not, decompose it into a lossless BCNF.

(a) The following three steps lead to a minimal cover.

 $\mathbf{Right\text{-}reducing} \ \mathsf{It} \ \mathsf{is} \ \mathsf{done} \ \mathsf{by} \ \mathsf{split} \ \mathsf{AD} \to \mathsf{BE} \ \mathsf{into} \ \mathsf{AD} \to \mathsf{B} \ \mathsf{and} \ \mathsf{AD} \to \mathsf{E}.$

Left-reducing leads to the following set:

$$A \to E \qquad BD \to E$$

$$A \to B \qquad F \to A$$

$$E \to B \qquad D \to H$$

$$BG \to F \ CD \to A$$

$$E \to D$$

By eliminating redundant ones, we obtain the following minimal cover of the given set of FDs:

$$A \to E \qquad BD \to E$$

$$E \to B \qquad F \to A$$

$$BG \to F \ D \to H$$

$$E \to D$$
 $CD \to A$

(b) The following is a join lossless, dependency preserving and 3NF decomposition of the original database schema.

AE, EBD, BGF, FA, DH, CDA, BGC.

Note that BFC is a candicate key that was added later.

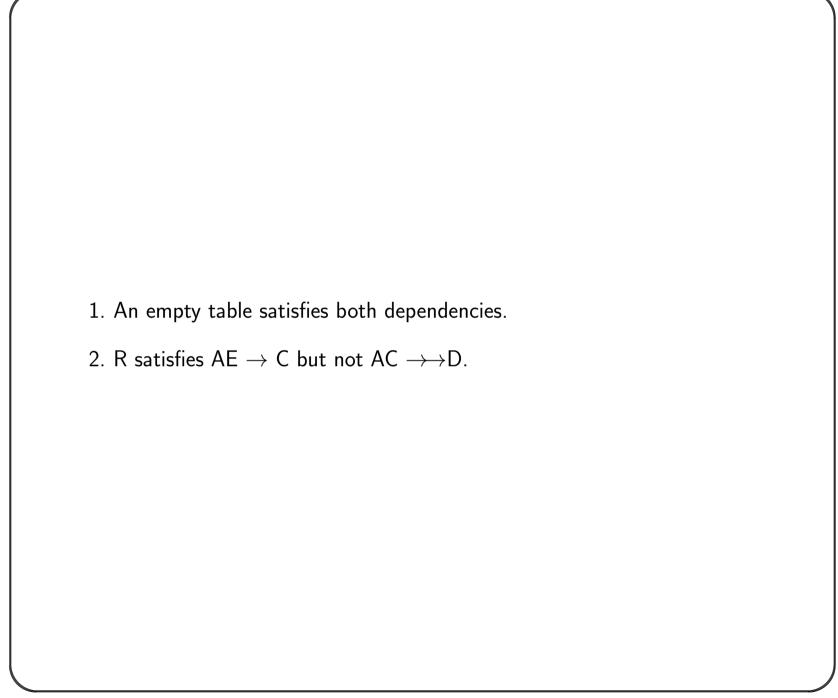
(c) All tables, except CDA, in the above database schema are in BCNF. CDA is not in BCNF because A \rightarrow D but not A \rightarrow C. CDA can be decomposited into {AD, AC}.

Example 3

Consider R = ABCDE. For each of the following instances of R, state whether (1) it violates the FD AE \rightarrow C, and (2) it violates the MVD AC $\rightarrow\rightarrow$ D:

1. an empty table

	Α	В	С	D	Е
2.	а	2	3	4	5
۷.	2	а	3	5	5
	а	3	3	6	5



Example 4

Consider R = ABCDEGHI and the following set F of functional dependencies:

$$\begin{aligned} \mathsf{H} &\to \mathsf{GD} \\ \mathsf{E} &\to \mathsf{D} \\ \mathsf{HD} &\to \mathsf{CE} \\ \mathsf{BD} &\to \mathsf{A} \end{aligned}$$

- 1. Find a join loss-less, dependency preserving and 3NF decomposition of ${\it R.}$
- 2. Indicate whether your database schema is in BCNF with respect to ${\cal F}.$ Explain.

We first find a minimal cover of the FDs, as shown below.

Right reduced	Left Reduced:	Minimal Cover	
$H \to G$	H o G	H o G	
H o D	$H\toD$	H o G $E o D$	
E o D	E o D	H o C	
$HD \to C$	H o C	$H \rightarrow C$	
$HD \to E$	H o E	$BD \to A$	
BD o A	$BD\toA$	BD o A	

Then construct a database $D' = \{HGCE, ED, BDA\}$.

Now, we need to check if D' contains any candidate key.

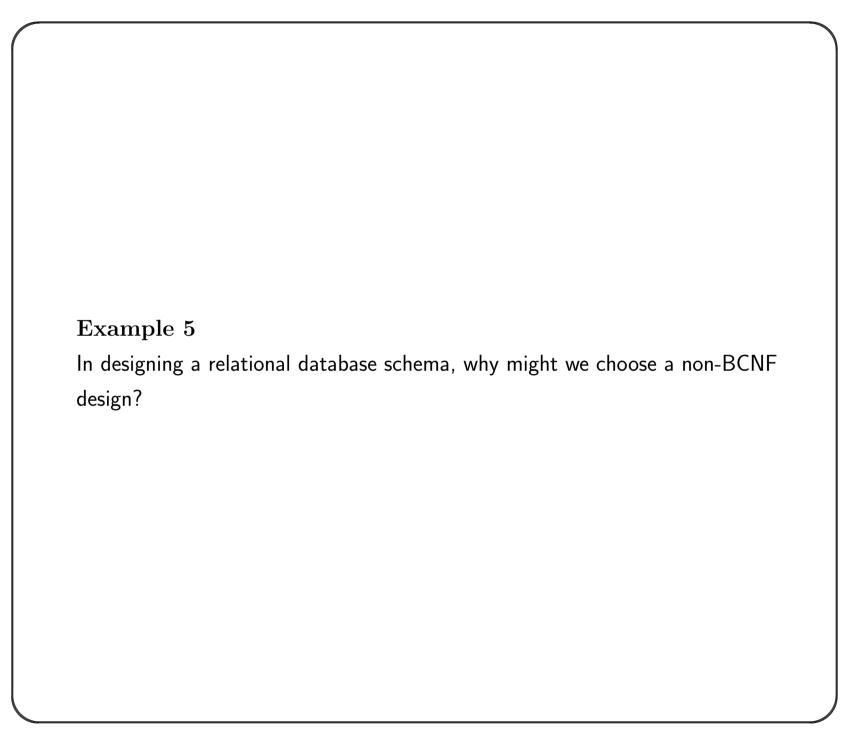
Since no FD in the minimal cover above contains H, B, or I in its right side, any candidate key shall contain these three attributes.

Further, it is not difficult to check that HBI is indeed a candidate key. Therefore, HBI is the only candidate key of R, and shall be added to D^{\prime} . Hence,

$$D = \{HGCE, ED, BDA, HBI\}$$

is a join loss-less, dependency preserving and 3NF decomposition of $\it R.$

(2) D is in BCNF since all the non-trivial FDs X \rightarrow A in held in any relation $R_i \in D$, X is a key of R_i .



This is because in many cases, there exists no database schema that is both BCNF and dependence preserving. If one prefers to have a dependence preserving database schema, then one have to choose a normal form, such as 3NF, that is weaker than BCNF.