# Databases
# TDA357/DIT621

## Exercise 3

## Functional Dependencies, Multivalued Dependencies and Normal Forms

1. Consider this (symbolic) relational schema: $R(a, b, c, d, e)$.

   The following functional dependencies apply to the relation:

   $a \to b$
   $b, c \to d$
   $d, e \to a$

   (a) List 3 different minimal keys of the relation. Your solution should be three sets of attributes (the keys).

   (b) Calculate the following transitive closures. Your solution should be three sets of attributes (the transitive closures), each in alphabetical order.

   $\{b, c\}^+$
   $\{a, c\}^+$
   $\{b, d, e\}^+$

   (c) Assume all the keys you identified have appropriate unique constraints in $R$, show a table (the contents of $R$) that does not respect the functional dependency $b, c \to d$. Use integer values for all the columns.

   In other words: Your solution should be a five column table (for columns $a, b, c, d$, and $e$) with at least two rows, that does not satisfy the functional dependency, but does have unique values for the three keys you found.

   (d) Decompose $R$ into BCNF, list all the intermediate steps. Determine all keys for the resulting schema.

### Solution:

(a) (minimal) keys: (minimal) set of attributes whose transitive closure determines all the attributes.

$c$ and $e$ only appear in LHS so they cannot be derived from other FP, hence they need to be part of every key.

They in itself are not a key: $\{c, e\}^+ = \{c, e\}$.
Keys: $\{a, c, e\}$, $\{c, d, e\}$, $\{b, c, e\}$.

(b) Closures:
$\{b, c\}^+ = \{b, c, d\}$
$\{a, c\}^+ = \{a, c, b, d\}$
$\{b, d, e\}^+ = \{b, d, e, a\}$

(c) $\{a, c, e\}$, $\{c, d, e\}$ and $\{b, c, e\}$ have UNIQUE constrains.
To find values that violate $b, c \to d$ we need 2 rows with the same values of b and c and different values of d. We need to also check that all unique constrains are kept! Column $a$ is essentially irrelevant and $e$ must be different to respect the keys.

| a | b | c | d | e |
|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 5 |
| 1 | 1 | 2 | 4 | 6 |

(d) Additional derived dependencies (just a few examples): $a, c \to d$ and $b, c, e \to a$.

Let $S$ be the set of attributes of the relational schema.
To convert into BCNF, look for a non-trivial FD $X \to Y$ such that $X^+$ doesn't contain all the attributes in $S$ (that is, $X$ is not a superkey).
If not such FD exists, then you are done.
Otherwise, divide $R$ into $R_1(X^+)$ and $R_2(X \cup (S - X^+))$.

We take $a \to b$: $\{a\}^+ = \{a, b\}$ so we split in

- $R_1(a, b)$ which is trivially in BCNF since $\{a\}^+ = R_1$;
- $R_2(a, c, d, e)$.

Now we continue with $R_2$. We cannot use $b, c \to d$ because $b$ is not in the domain of $R_2$ so we take $d, e \to a$: $\{d, e\}^+ = \{d, e, a, b\}$.

- $R_{21}(a, d, e)$, again trivially in BCNF,
  (Observe that $b$ is part of the closure but not of the domain so we cannot put it as part of $R_{21}$);
- $R_{22}(c, d, e)$ which is also in BCNF since there is no violation.

We need to find the keys and references:

- $R_1(\underline{a}, b)$ since $a \to b$;
- $R_{21}(a, \underline{d}, \underline{e})$ since $d, e \to a$,
  reference: $a \to R_1.a$;
- $R_{22}(\underline{c}, \underline{d}, \underline{e})$ since we don't have any FD between the elements,
  reference: $(d, e) \to R_{21}.(d, e)$

OBS: we cannot put $(d, e) \to R_{22}.(d, e)$ as reference in $R_{21}$ since $(d, e)$ is not a key in $R_{22}$.

2. Consider the following domain:

   Each flight is identified by a flight number and a departure time. Each flight has a set of passenger IDs and a set of airport codes it lands at. Furthermore, each passenger has a set of in-flight movies they have purchased and can use on any flight.

   For the relation $R$(flightNo, departure, airport, passenger, movie), identify all MVDs you can find and normalize $R$ to 4NF.

   **Solution:**

   MVD:

   - flightNo, departure $\twoheadrightarrow$ airport
   - flightNo, departure $\twoheadrightarrow$ passenger, movie

   Note: The two above are equivalent and express that the airports a flight lands at are independent from the passengers and the movies they can watch.

   - passenger $\twoheadrightarrow$ movie
   - passenger $\twoheadrightarrow$ airport, flightNo, departure

   Note: These two are also equivalent and express that the movies a passenger can watch are independent from the flights they are booked on and the airports those flights land at.

   Let $S$ be the set of attributes of the relational schema.
   Given a relational schema in BCNF, to convert it into 4NF look for a non-trivial MVD $X \twoheadrightarrow Y$ such that $X^+$ doesn't contain all the attributes in $S$ (that is, $X$ is not a superkey).
   OBS: recall that we compute $X^+$ using FD and not MVD!

   If not such MVD exists, then you are done.
   Otherwise, divide $R$ into $R_1(X \cup Y)$ and $R_2(S - Y)$.

   4NF: We start with $R$(flight, departure, airport, passenger, movie)

   Decomposing after passenger $\twoheadrightarrow$ movie:

   - $R_1$ (passenger, movie)
   - $R_2$ (flightNo, departure, airport, passanger)

   Decomposing after flightNo, departure $\twoheadrightarrow$ airport:

   - $R_{21}$ (flightNo, departure, airport)
   - $R_{22}$ (flightNo, departure, passanger)

   Keys:

   - $R_1$ (<u>passenger</u>, <u>movie</u>)

- $R_{21}$ (flightNo, departure, airport)
- $R_{22}$ (flightNo, departure, passanger)

Logically, the first ones lists what movies passengers have, the second where flights land, and the third what passengers are on what flights.

3. (a) Consider the (symbolic) table:

| a | b | c | d |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 |
| 2 | 0 | 2 | 2 |

Identify two non-trivial functional dependencies that hold on this data and normalize it to BCNF. Provide the resulting schema, and the data in each relation of the schema as a table like the one above.

(b) Briefly explain the concept of lossless join using your result from a) as an example.

(c) Some random person claims that it's impossible to replace the '?' below so that $a \twoheadrightarrow b$ holds but neither $a \rightarrow b$ not $a \rightarrow c$ holds on the table. Either prove the random person wrong (by constructing a table) or write a short and compelling argument for why they are correct.

| a | b | c |
|---|---|---|
| 0 | ? | ? |
| 0 | ? | ? |
| 0 | ? | ? |

**Solution:**

(a) Recall: $X$ determines $Y$, denoted $X \rightarrow Y$, iff for any 2 rows $t, u$ in the table, if $t.X = u.X$ then $t.Y = u.Y$.

That is, if two rows agree on the values of the attributes in $X$, then they should also agree on the values of the attributes in $Y$.

Looking at the table we can see that $c \rightarrow b$ and $a, b \rightarrow c$.

(We have also the non-trivially derived dependencies $a, c \rightarrow b$ and $c, d \rightarrow b$.)

$\{c\}^+ = \{b, c\}$ which gives as $R_1(b, c)$ and $R_2(a, c, d)$, both in BCNF.

The tables and their data becomes then:

| b | c |
|---|---|
| 0 | 0 |
| 1 | 1 |
| 0 | 2 |

| a | c | d |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 2 | 2 | 2 |

(b) Taking the natural join of two relations $R_1$ and $R_2$ gives the original table. Thus no information has been lost in the decomposition.

(c) It is not possible. For $a \twoheadrightarrow b$ to be true, the 6 unknown cells would need to be the Cartesian product of two tables, but the only way to get 3 (pairs of) elements in a Cartesian product is if one of the operands has three elements and the other has 1. (Recall that $|A \times B| = |A| * |B|$ for $A$ y $B$ finite sets.)

Basically it would have to be the Cartesian product of something like $\{0, 1, 2\}$ and $\{0\}$, but then the resulting table would have a column (either $b$ or $c$) with only 0's, and thus either $a \to b$ or $a \to c$ would hold.

4. (Adapted from an exam.)

The questions below all relate to this this table with four columns $(a, b, c, d)$ and five rows:

| $a$ | $b$ | $c$ | $d$ |
|---|---|---|---|
| a0 | b0 | c0 | d0 |
| a0 | b0 | c0 | d1 |
| a1 | b1 | c1 | d1 |
| a2 | b1 | c1 | d0 |
| a0 | b1 | c0 | d1 |

The values (a0, b0, ...) are symbolic, the only important thing is that a0 differs from a1 etc.

(a) Which of the following FDs hold on the table data? Your solution should be one or more of the FDs below, no motivation is required.

$a\ b \to c$
$a\ b \to d$
$c\ b \to a$
$c\ b \to c$

(b) The FD $c\ d \to a$ holds on this table. Explain why it is also a BCNF-violation.

(c) Perform one BCNF decomposition using $c\ d \to a$, and then decompose the table data into the data of both resulting tables.

Your solution should be two tables with column names and table contents (rows). You do not need to mark primary keys.

Hint 1: The natural join of the two tables should be the original tables.
Hint 2: In a correct solution, the tables have different number of rows.

(d) Find one more BCNF-violation that still holds in the tables resulting from part (c). You do not need to perform normalization or motivate your answer, just find a violation.

Your solution should be a single functional dependency.

## Solution:

(a) $a\ b \to c$ and $c\ b \to c$.

(b) $\{c, d\}^+ = \{c, d\}$ is not a superkey, so $c\ d \to a$ is a BCNF-violation.

(c)

| $a$ | $c$ | $d$ |
|---|---|---|
| a0 | c0 | d0 |
| a0 | c0 | d1 |
| a1 | c1 | d1 |
| a2 | c1 | d0 |

| $b$ | $c$ | $d$ |
|---|---|---|
| b0 | c0 | d0 |
| b0 | c0 | d1 |
| b1 | c1 | d1 |
| b1 | c1 | d0 |
| b1 | c0 | d1 |

(d) $a \to c$

5. Consider the following relation and functional dependencies:

$R(a, b, c, d, e)$
$a \to b$
$c \to d$
$e \to a$

Give a real-world example of attributes $a, b, c, d, e$ that would have exactly these dependencies and none else (except of course derived ones). Add any important assumptions you make about the domain.

An attempted example would have the same format as:
Lectures (courseCode, courseTitle, date, room, teacher).

This would reasonably satisfy $a \to b$ (because course code determines the title), and $c \to d$ (because we assume that the course has at most one lecture each date). But it would not satisfy $e \to a$ because a teacher can have several courses. It might also have the unwanted dependency $a, c \to e$ if we assume that each lecture has one teacher.

## Solution:

Seems to be 2 disjunct sets of attributes: $\{a, b, e\}$ and $\{c, d\}$.
$\{c, d\}$: could be personal number and name, or name and phone.
$\{a, b, e\}$: here $e$ is the most specific and $b$ the least specific, so it could be planet, solar system, and galaxy or model, manufacturer and license plate.