Relational Databases | Caltech CS 121
Assignment 7 | March 2023

Dallas Taylor

1 E-R Diagrams

1.1

Many-to-many mapping between A and B

Solution : *There are no non-trivial dependencies, unless you consider a more complex case. Consider having a database that tracks artists, artwork, and medium. Although multiple artists can use multiple mediums and the same medium can be used by multiple artists, there could be a non-trivial dependency such that certain mediums are only used by certain artists. However this is extremely non-trivial.*

1.2

Many-to-one mapping between A and B

Solution : $F = \{a \rightarrow b\}$

1.3

One-to-many mapping between A and B

Solution : $F = \{b \rightarrow a\}$

1.4

One-to-one mapping between A and B

Solution : $F = \{a \rightarrow b, b \rightarrow a\}$

Dallas Taylor

2 Rules

Union

Solution : *The Union rule states that if $a \to b$ holds on R and $a \to g$ holds on R, then $a \to bg$ holds on R. Using only Armstrong's axioms, we can derive:*

$$\begin{aligned} a \to b, \ \ a \to g & \qquad \textit{given} \\ aa \to ag & \qquad \textit{augmentation} \\ ag \to bg & \qquad \textit{augmentation} \\ aa \to bg & \qquad \textit{transitivity} \\ \therefore a \to bg & \qquad \textit{equivalence} \end{aligned}$$

Decomposition

Solution : *The Decomposition rule states that if $a \to bg$ holds on R, then $a \to b$ holds on R and $a \to g$ holds on R. Using only Armstrong's axioms, we can derive:*

$$\begin{aligned} a \to bg & \qquad \textit{given} \\ bg \to b, \ \ bg \to g & \qquad \textit{reflexivity} \\ \therefore a \to b & \qquad \textit{transitivity} \\ \therefore a \to g & \qquad \textit{transitivity} \end{aligned}$$

Pseudotransitivity

Solution : *The Pseudotransitivity rule states that if $a \to b$ holds on R and $gb \to d$, then $ag \to d$ holds on R. Using only Armstrong's axioms, we can derive:*

$$\begin{aligned} a \to b, \ \ gb \to d & \qquad \textit{given} \\ ag \to bg & \qquad \textit{augmentation} \\ \therefore ag \to d & \qquad \textit{equivalence, transitivity} \end{aligned}$$

3 Functional Dependencies

$R = (A, B, C, D, E)$

$F = \{A \to BC, CD \to E, B \to D, E \to A\}$

3.a

Solution : *We can first trivially find all single attribute superkeys by traversing through the functional dependencies:*

$\alpha = A$, $\alpha^+ = A$ *(identity):*

- $A \to BC$, *so* $\alpha^+ = ABC$ *(union)*
- $B \to D$, *so* $\alpha^+ = ABCD$ *(union)*
- $CD \to E$, *so* $\alpha^+ = ABCDE$ *(union)* ✓

$\alpha = B$, $\alpha^+ = B$ *(identity):*

- $B \to D$, *so* $\alpha^+ = BD$ *(union)* ✗

$\alpha = C$, $\alpha^+ = C$ *(identity)* ✗

$\alpha = D$, $\alpha^+ = D$ *(identity)* ✗*:*

$\alpha = E$, $\alpha^+ = E$ *(identity):*

- $E \to A$, *so* $\alpha^+ = AE$ *(union)*
- *same logic as for* A, *so* $\alpha^+ = ABCDE$ *(union)* ✓

So we have that A, E *are single key candidate keys. Now, let us consider having a set of two attributes. We know that we cannot have any set of attributes containing* A *or* E*, as these would not be simple candidate keys.*

$\alpha = BC$, $\alpha^+ = BC$ *(identity):*

- $B \to D$, *so* $\alpha^+ = BCD$ *(union)*
- $CD \to E$, *so* $\alpha^+ = BCDE$ *(union)*
- $E \to A$, *so* $\alpha^+ = ABCDE$ *(union)* ✓

- *Attribute set closures:*

  $B^+ = BD$ *(from above)*

  $C^+ = C$ *(from above)* ✓

$\alpha = BD$, $\alpha^+ = BD$ *(identity):*

- $B \to D$, *so* $\alpha^+ = BD$ *(union)* ✗

$\alpha = CD$, $\alpha^+ = CD$ *(identity):*

- $CD \to E$, *so* $\alpha^+ = CDE$ *(union)*
- $E \to A$, *so* $\alpha^+ = ACDE$ *(union)*
- $A \to BC$, *so* $\alpha^+ = ABCDE$ *(union)* ✓
- *Attribute set closures:*

  $C^+ = C$ *(from above)*

  $D^+ = D$ *(from above)* ✓

Therefore, we find the final set of candidate keys to be $\{A, BC, CD, E\}$

3.b

Describe all functional dependencies that will appear in the closure F^+ of F.

Solution :

- $A \to ABCDE$, *and all dependencies generated from this by applying the Decomposition Rule (above)*
- $E \to ABCDE$, *and all dependencies generated from this by applying the Decomposition Rule (above)*
- $BC \to ABCDE$, *and all dependencies generated from this by applying the Decomposition Rule (above)*
- $CD \to ABCDE$, *and all dependencies generated from this by applying the Decomposition Rule (above)*
- $B \to D$ *(given),* $B \to BD$ *(union)*
- *All trivial dependencies* $\alpha \to \beta$ *where* $\alpha = \{ABCDE\}$ *and* $\beta \subseteq \alpha$ *(identity, decomposition)*

- *All trivial dependencies* $\alpha\beta \rightarrow \gamma$ *where* $\alpha = \{A, BC, CD, E\}$, $\beta \in \{ABCDE\}$, *and* $\gamma \subseteq R$

4 Functional Dependencies

$R(A, B, C, D, E, G)$

$F = \{A \rightarrow E, BC \rightarrow D, C \rightarrow A, AB \rightarrow D, D \rightarrow G, BC \rightarrow E, D \rightarrow E, BC \rightarrow A\}$

4.a

Compute a canonical cover F_c of F.

Solution :

$$\begin{aligned}
F_c = \{A \rightarrow E, BC \rightarrow ADE, C \rightarrow A, AB \rightarrow D, D \rightarrow EG\} &\quad \textit{(union)} \\
BC \rightarrow AB &\quad \textit{(augmentation)} \\
BC \rightarrow D &\quad \textit{(transitivity)} \\
\therefore F_c = \{A \rightarrow E, BC \rightarrow AE, C \rightarrow A, AB \rightarrow D, D \rightarrow EG\} &\quad (BC \rightarrow D \textit{ extraneous}) \\
BC \rightarrow AB &\quad \textit{(augmentation)} \\
BC \rightarrow A &\quad \textit{(decomposition)} \\
\therefore F_c = \{A \rightarrow E, BC \rightarrow E, C \rightarrow A, AB \rightarrow D, D \rightarrow EG\} &\quad (BC \rightarrow A \textit{ extraneous}) \\
BC \rightarrow AB &\quad \textit{(augmentation)} \\
BC \rightarrow D &\quad \textit{(transitivity)} \\
BC \rightarrow E &\quad \textit{(transitivity, decomposition)} \\
\therefore F_c = \{A \rightarrow E, C \rightarrow A, AB \rightarrow D, D \rightarrow EG\} &\quad (BC \rightarrow E \textit{ extraneous})
\end{aligned}$$

There are no more simplifications to be made, so we now have the final canonical cover F_c *of* F.

4.b

Find a candidate key for R.

Solution : *We can find a candidate key for* R *to be* BC.

$(BC)^+$:

- $\alpha^+ = BC$

- $C \to A$, *so* $\alpha^+ = ABC$ *(transitivity, union)*
- $AB \to D$, *so* $\alpha^+ = ABCD$ *(transitivity, union)*
- $D \to EG$, *so* $\alpha^+ = ABCDEG$ *(transitivity, union)* ✓

B^+:

- $\alpha^+ = B$ ✗

C^+:

- $\alpha^+ = C$
- $C \to A$, *so* $\alpha^+ = AC$ *(transitivity, union)*
- $A \to E$, *so* $\alpha^+ = ACE$ *(transitivity, union)* ✗

Therefore, we have shown that we satisfy all requirements for BC *to ba candidate key.*

4.c

Decompose R into a BCNF schema. Prove that each of your final relation-schemas is indeed in BCNF.

Solution :

- $A \to E$, *so we have* $R_1(\underline{A}, E)$ *and* $R_2(A, B, C, D, G)$
- $C \to A$, *so we have* $R_1(\underline{A}, E)$, $R_2(\underline{C}, A)$, *and* $R_3(B, C, D, G)$
- $D \to G$, *so we have* $R_1(\underline{A}, E)$, $R_2(\underline{C}, A)$, $R_3(\underline{D}, G)$, *and* $R_5(B, C, D)$

Since BC *is a candidate key, we can determine that our final relation-schemas are:*

$$R_1(\underline{A}, E), R_2(\underline{C}, A), R_3(\underline{D}, G), \text{ and } R_4(\underline{B}, \underline{C}, D)$$

It is clear that we do not preserve the relations of $AB \to D$ *and* $D \to E$.

We know that our final relation-schemas are in BCNF. This is trivial for R_1, R_2, *and* R_3 *as each of* A, C, D *is on the left hand side of the relation and is a primary key. For* R_4 *this is trivial since* BC *is a candidate key and a primary key.*

4.d

Using the 3NF schema synthesis algorithm, create a 3NF schema for R.

Solution : $F_c = \{A \rightarrow E, C \rightarrow A, AB \rightarrow D, D \rightarrow EG\}$

1. $A \rightarrow E$, *so we have* $R_1(\underline{A}, E)$
2. $C \rightarrow A$, *so we have* $R_2(\underline{C}, A)$
3. $AB \rightarrow D$, *so we have* $R_3(\underline{A}, \underline{B}, D)$
4. $D \rightarrow EG$, *so we have* $R_4(\underline{D}, E, G)$

Candidate key BC *is not in any schema, therefore we have:* $R_5(\underline{B}, \underline{C})$

5 (actually 6) Functional Dependencies

R(course_id, section_id, dept, units, course_level, instructor_id, term, year, meet_time, room, num_students)

$F = \{$

$$\begin{aligned} \{\text{course\_id}\} &\rightarrow \{\text{dept, units, course\_level}\} \\ \{\text{course\_id, section\_id, term, year}\} &\rightarrow \{\text{meet\_time, room, num\_students, instructor\_id}\} \\ \{\text{room, meet\_time, term, year}\} &\rightarrow \{\text{instructor\_id, course\_id, section\_id}\} \end{aligned}$$

$\}$

Shorthand labeling:

A. course_id
B. section_id
C. dept
D. units
E. course_level
F.
G. instructor_id
H. term
I. year
J. meet_time
K. room
L. num_students

6.a

Find all candidate keys of R.

Solution : *There are two candidate keys for R.*

$\alpha_1 = (ABHI)$

- $\alpha_1^+ = ABHI$ *(identity)*
- $A \rightarrow CDE$, *so* $\alpha_1^+ = ABCDEHI$ *(transitivity, union)*

- $ABHI \to JKLG$, so $\alpha_1^+ = ABCDEGHIJKL$ *(transitivity, union)* ✓

$\alpha_2 = (KJHI)$

- $\alpha_2^+ = HIJK$ *(identity)*
- $KJHI \to GAB$, so $\alpha_2^+ = ABGHIJK$ *(transitivity, union)*
- $A \to CDE$, so $\alpha_2^+ = ABCDEGHIJK$ *(transitivity, union)*
- $ABHI \to JKLG$, so $\alpha_2^+ = ABCDEGHIJKL$ *(transitivity, union)* ✓

Therefore, the candidate keys for R are $\{ABHI, KJHI\}$, which can also be represented as {{course_id, section_-id, term, year}, {room, meet_time, term, year}}

6.b

Identify all canonical covers F_c for the above set of functional dependencies.

Solution : *It is trivial that the the instructor_id is extraneous in in one of the two relations it is involved in, producing two possible canoncial covers given below:*

$$\begin{aligned} F &= \{A \to CDE, ABHI \to JKLG, KJHI \to GAB\} \\ F_c^1 &= \{A \to CDE, ABHI \to JKL, KJHI \to GAB\} \\ F_c^2 &= \{A \to CDE, ABHI \to JKLG, KJHI \to AB\} \end{aligned}$$

For the context of our problem, it will probably be more appropriate to use F_c^2, since it will most likely be more helpful to associate the course_id, section_id, term, and year with an instructor_id versus a room, meet_time, term, and year. You want to know what professor teaches what, and it will likely be more rare to try to directly find the instructor within a certain room.

6.c

Suggest both a normal form and a schema decomposition that would be best for actual use in a course management system. Briefly explain your rationale.

Solution : *BCNF:*

- $A \to CDE$, *so we have* $R_1(\underline{A}, C, D, E)$ *and* $R_2(A, B, G, H, I, J, K, L)$
- $ABHI \to JKLG$, *so we have* $R_1(\underline{A}, C, D, E)$ *and* $R_2(\underline{A}, \underline{B}, \underline{H}, \underline{I}, G, J, K, L)$.

3NF:

1. $A \to CDE$, *so we have* $R_1(\underline{A}, C, D, E)$
2. $ABHI \to JKLG$, *so we have* $R_2(\underline{A}, \underline{B}, \underline{H}, \underline{I}, J, K, L, G)$
3. $KJHI \to AB$, *all of* $ABHIJK$ *already in* R_2

We trivially get the same schema representation. There are only two different schemas. We should enforce all restraints since any of the room, instructor_id, meet_time, and num_students could change for any course over different years, terms, or sections.