Q Compute the closure of following set F of functional dependencies for relational schema $R = (A, B, C, D, E)$

$A \to BC$

$CD \to E$

$B \to D$

$E \to A$

List all candidate keys for R

Soln: (1) $A \to BC$ i.e $A \to B$ and $A \to C$

Since $A \to B$ and $B \to D$

$A \to D$ (decomposition, transitive)

Since $A \to CD$ and $CD \to E$

$A \to E$ (union, decomposition, transitive)

Since $A \to A$ (reflexive)

$\therefore \{A\}^+ \to ABCDE$ from above steps (union) — (I)

(2) Since $E \to A$, $A \to ABCDE$ (I) (~~transitive~~)

$\therefore \{E\}^+ \to ABCDE$ (transitive)

(3) Since $CD \to E$, $E \to ABCDE$ (from 2)

$\therefore \{CD\}^+ \to ABCDE$ (transitive)

(4) Since $B \to D$ and $BC \to CD$, (augmentative)

As $CD \to ABCDE$ (from 3)

$\therefore \{BC\}^+ \to ABCDE$

Date / /

Therefore, candidate keys are A, BC, CD and E.

Q Find minimal cover of set of functional dependencies given

$A \rightarrow C$

$AB \rightarrow C$

$C \rightarrow DI$

$CD \rightarrow I$

$EC \rightarrow AB$

$EI \rightarrow C$

Sol^n

Minimal cover :-

Minimal cover of set of FDs is a minimal set of functional dependencies Fmin that is equivalent to F. There can be many such minimal covers for a set of functional dependencies F

Steps of minimal cover :-

[1] Right hand side (RHS) of all FD's should be single attribute

[2] Remove extraneous attributes

Extraneous Attribute

Consider functional dependencies F and any functional dependency of form $\alpha \rightarrow \beta$

Assume α and β are set of one or more attributes [$A \rightarrow BC$, or $AB \rightarrow CE$]

case (I LHS) :- To find if an attribute A in α is extraneous or not i.e to test if an attribute of LHS of FD is extraneous or not

step 1: Find $(\{\alpha\} - A)^+$ using dependencies of F

step 2: If $(\{\alpha\} - A)^+$ contains all the attributes of β, then A is extraneous

case II RHS: To find if an attribute A in β is extraneous or not. i.e test if an attribute of RHS of FD is extraneous or not

step 1: If α^+ using dependencies in F' where $F' = (F - \{\alpha \to \beta\} \cup (\alpha \to (\beta - A)\}$

step 2: If α^+ contains A, then A is extraneous

Example for finding extraneous attribute

Given: $F = \{P \to Q, PQ \to R\}$ Is Q extraneous in $PQ \to R$?

Soln: As we are looking for LHS attribute (Q) let us use case I discussed above step 1 : Find $(\{\alpha\} - A)^+$ using dependencies of F

α is PQ so find $(PQ - Q)^+$ i.e P^+ (closure of P)

P^+ closure of P attribute.

$P^+ = \{P\}$ reflexive }

$P^+ = \{P, Q\}$ $\because P \rightarrow Q$

$P^+ = \{P, Q, R\}$ $\because PQ \rightarrow R$

Hence P^+ : closure of P is PQR

step 2 : If $(\{\alpha\} - A)^+$ contains all attributes of β, then A is extraneous

$(PQ - Q)^+$ contains R (i.e. closure of P contains R)

So Q is extraneous in $PQ \rightarrow R$

Similarly : Given $F = \{P \rightarrow QR, PQ \rightarrow R\}$ Is R extraneous in $P \rightarrow QR$?

Soln : As we looking for RHS attribute. let us use case II discussed above

step 1 : α^+ using dependencies in F' where $F' = (F - \{\alpha \rightarrow \beta\} \cup (\alpha \rightarrow (\beta - A)\}$

So,

$F' = (F - \{\alpha \rightarrow \beta\} \cup \{\alpha \rightarrow (B - A)\}$

$= (\{P \rightarrow QR, Q \rightarrow R\} - \{P \rightarrow QR\}) \cup \{P \rightarrow (QR-R)$

$F' = \{Q \rightarrow R\} \cup P \rightarrow Q$

$\therefore$ α is P so find P^+ closure of P using F' we found

$F' = \{Q \to R,\ P \to Q\}$

$P^+ = \{P\}$ reflexive

$P^+ = \{P, Q\}$ $\because P \to Q$

$P^+ = \{P, Q, R\}$ $\because$ transitive

so P closure is PQR

Step 2 : If α^+ contains A, then A is extraneous

P^+ contains R. Hence R is extraneous in $P \to QR$

BACK TO MINIMAL COVER FINDING STEPS:-

Step 3 : Remove redundant functional dependencies

$\Rightarrow$ Soln : Step 1 : RHS of all FDs should be single attribute so we write F as F_1

$F_1 = \{A \to C,\ AB \to C,\ C \to D,\ C \to I,\ CD \to I,$
$EC \to A,\ EC \to B,\ EI \to C\}$

Step 2 : Remove extraneous attributes

Using above steps to find extraneous attribute i.e. redundant attributes on LHS of FD.

In set of FDs $AB \to C$, $CD \to I$, $EC \to A$, $EC \to B$, $EI \to C$ have more than one attribute in LHS

To check, we need to find closure of attributes on LHS. {Apply closure finding algorithm}

① $A^+ = ACDI$

② $B^+ = B$

③ $C^+ = CDI$

④ $D^+ = D$

⑤ $E^+ = E$

⑥ $I^+ = I$

From ①, closure of A included attribute C. So B is extraneous in $AB \rightarrow C$ and B can be removed

From ③, closure of C included attribute I So D is extraneous in $CD \rightarrow I$ and D can be removed

No more extraneous attributes are found. Hence we write F_1 as F_2 after removing extraneous attributes from F_1 as follows:

$F_2 = \{A \rightarrow C, C \rightarrow D, C \rightarrow I, EC \rightarrow A, EC \rightarrow B, EI \rightarrow \cancel{B}C\}$

$\boxed{\text{Step 3}}$: Eliminate redundant functional dependency

None of FDs in ~~F3~~ F_2 is redundant

Hence, F_2 is minimal cover

* Equivalence of Two sets of Functional Dependencies :-

① Two ~~set~~ different sets of functional dependencies for a given relation may or may not be equivalent.

② If F and G are two sets of functional dependencies, then following 3 cases are possible :-

case 01 : F covers G ($F \supseteq G$)

case 02 : G covers F ($G \supseteq F$)

case 03 : Both F and G cover each other ($F = G$)

case 01 : Determining whether F covers G :

Following steps are followed :-

step 1 :

(i) Take functional dependencies of set G into consideration

(ii) For each functional dependency $X \rightarrow Y$, find closure of X using functional dependencies of G set

step 2 :

(i) Take functional dependencies of set G into consideration

(ii) ~~If~~ For each functional dependency $X \rightarrow Y$, find the closure of X using functional dependencies of set F.

step 3 :

(i) compare results of step 1 and step 2

(ii) If the functional dependencies of set F has determined all attributes that were determined by functional dependencies of set G then it means $\boxed{\text{F covers G}}$

(iii) Thus we conclude F covers G $(F \supseteq G)$

case 02 : Determining whether G covers F :

following steps are followed :-

step 1 :

(i) Take functional dependencies of set F into consideration

(ii) For each functional dependency $X \to Y$, find closure of X using functional dependencies of set F.

step 2 :

(i) Take functional dependencies of set F into consideration

(ii) For each functional dependency $X \to Y$, find closure of X using functional dependencies of set G.

step 3 :

(i) Compare results of step 1 and step 2

(ii) If functional dependencies of set G has determined all the attributes that were determined

determined by functional dependencies of set F, then it means $\boxed{G \text{ covers } F}$

(iii) Thus, we conclude G covers F ($G \supseteq F$)

case 03: Determining whether both F and G cover each other

(i) If F covers G and G covers F then both F and G covers each other ($F = G$)

PROBLEM:-

A relation R (A, C, D, E, H) is having two functional dependencies sets F and G as sho

Set F: $A \rightarrow C$

$AC \rightarrow D$

$E \rightarrow AD$

$E \rightarrow H$

Set G: $A \rightarrow CD$

$E \rightarrow AH$

Which of following holds true?

(A) $F \supseteq G$

(B) $G \supseteq F$

(C) $F = G$

(D) All of above

Date / /

solution :-

Determining whether F covers G :-

step 1 :

$(A)^+ = \{A, C, D\}$ // closure of LHS $A \to CD$ using set G

$(E)^+ = \{A, C, D, E, H\}$ // closure of LHS of $E \to AH$ using set G

step 2 :

$(A)^+ = \{A, C, D\}$ // closure of LHS $A \to CD$ using set F

$(E)^+ = \{A, C, D, E, H\}$ // closure of LHS $E \to AH$ using set F

step 3 :

Comparing results of step 1 and step 2, we find

(i) Functional dependencies of set F can determine all the attributes which have been determined by functional dependencies of set G

(ii) Thus, we conclude F covers G $(F \supseteq G)$

Determining whether G covers F :

step 1 :

$(A)^+ = \{A, C, D\}$ // closure of $A \to C$ using set F

$(AC)^+ = \{A, C, D\}$ // closure of LHS $AC \to D$ using set F

$(E)^+ = \{A, C, D, E, H\}$ // closure of $E \to AD$, $E \to H$ using set F

Scanned with CamScanner

step 2:

$(A)^+ = \{A, C, D\}$ // closure of $A \to C$ using set G

$(AC)^+ = \{A, C, D\}$ // closure of $AC \to D$ using set G

$(E)^+ = \{A, C, D, E, H\}$ // closure of $E \to AD$ and $E \to H$ using set G

step 3:

comparing results of step 1 and step 2:

① Functional dependencies of ~~step~~ set G can determine all the attributes which have been determined by functional dependencies of set F

② conclude, G covers F i.e $G \supseteq F$

Determining whether both F and G covers each other:

step 1, we conclude F covers G

step 2; we conclude G covers F

step 3 : both F and G cover each other

$F = G$

ANS : ALL OF ABOVE

Q Suppose you are given a relation R with four attributes ABCD. For each of the following sets of FDs, assuming those are only dependencies that hold for R, do following

(i) Identify candidate key(s) for R

(ii) Identify best normal form that R satisfies (1NF, 2NF, 3NF or BCNF)

(iii) If R is not in BCNF, decompose it into a set of BCNF relations that preserve dependencies

1] $C \to D$, $C \to A$, $B \to C$

2] $B \to C$, $D \to A$

3] $ABC \to D$, $D \to A$

4] $A \to B$, $BC \to D$, $D \to A$

5] $AB \to C$, $AB \to D$, $C \to A$, $D \to B$

1] $C \to D$, $C \to A$, $B \to C$

a] candidate key : B

b] R is in 2NF but not in 3NF

c] $C \to D$ and $C \to A$ both cause violations of BCNF. One way to obtain lossless join decomposition is to decompose R into AC, BC and CD

2] B → C, D → A

a] candidate key : BD

b] R is in 1NF but not 2NF

c] Both B → C and D → A causes violations of BCNF. AD, BC and BD is BCNF and lossless join preserving

3] ABC → D, D → A.

a] candidate keys : ABC, BCD

b] R is in 3NF but not BCNF

c] ABCD is not in BCNF since D → A and D is not a key.
However if we split up R as AD, BCD we cannot preserve dependency ABC → D so there is no BCNF decomposition

4] A → B, BC → D, D → A

a] candidate key : A

b] R is in 2NF but not in 3NF (of BC → D)

c] BC → D violates BCNF since BC is not a key. So we split R as BCD, ABC

5) AB → C, AB → D, C → A, D → B

a) candidate keys : AB, BC, CD, AD

b) R is in 3NF, but not in BCNF (∵ C→A

c) C → A and D → B both cause violating

so decompose into AC, BCD but this does not preserve AB → C and AB → D and BCD is still not BCNF because D → B

∴ we need to decompose AC, BD, CD

However we revive lost functional dependencies by adding ABC and ABD, but relations are not in BCNF form

∴ no BCNF decomposition

Scanned with CamScanner

Date / /

* Decomposition in DBMS :-

(1) Decomposition is a process of dividing single relation into two or more sub relations

(2) Decomposition can be completed in following two ways :-
- Lossless join decomposition
- lossy join decomposition

⇒ Determine decomposition is lossy or lossless :-

Consider relation R is decomposed into two sub relations R_1 and R_2

Then,

(i) If all the following condition satisfy, then decomposition is lossless

(ii) If any of these conditions fail, then decomposition is lossy

Condition 01 :

Union of both subrelations must contain all the attributes that are present in original relation R.

$$R_1 \cup R_2 = R$$

Condition 02 :

Intersection of both sub relations must not be null (Common attribute between them must

DOMS Page No.
Date / /

be present)

$$R_1 \cap R_2 \neq \phi$$

condition 03:

Intersection of both sub relations must be a super key of either R_1 or R_2 or both

$$R_1 \cap R_2 = \text{Superkey of } R_1 \text{ or } R_2$$

PROBLEM:—

Consider a relation schema R(A, B, C, D) with functional dependencies $A \rightarrow B$ and $C \rightarrow D$. Determine whether decomposition of R into $R_1(A, B)$ and $R_2(C, D)$ is lossless or lossy

solution:

→ check all conditions one by one

→ if any one condition fail then decomposition is lossy otherwise lossless

condition 01:

$$R_1 \cup R_2 = R_1(A, B) \cup R_2(C, D)$$
$$= R(A, B, C, D)$$

condition 1 satisfies

condition 02:

$$R_1(A, B) \cap R_2(C, D)$$

condition 2 fails

∴ Decomposition is lossy

Lossless join decomposition

Question :-

let $R = \{ssn, ename, pnumber, pname, plocation, hours\}$ and R is decomposed into three relations R_1, R_2 and R_3 as follows :-

$R_1 = EMP = \{ssn, ename\}$

$R_2 = PROJ = \{pnumber, pname, plocation\}$

$R_3 = WORKS\_ON \{ssn, pnumber, hours\}$

Assume that following functional dependencies are holding in relation R

$F = \{ssn \rightarrow ename, pnumber \rightarrow \{pname, plocation\},$
$\{ssn, pnumber\} \rightarrow hours$

Find whether decomposition into R_1, R_2 and R_3 is lossless join decomposition or not

Soln :- Lossless join decomposition :-

If a relation R is decomposed into relations R_1 and R_2 then decomposition is lossless if **either** of following holds:-

$$(R_1 \cap R_2) \rightarrow R_1$$
$$(R_1 \cap R_2) \rightarrow R_2$$

In our problem, $R_1 \cap R_2 = \{\phi\}$ no common attribute

DOMS Page No.
Date / /

$R_1 \cap R_3 = \{\cancel{ssn}\}$

$= \{ssn, ename\} \cap \{ssn, pnumber, hours\}$

$= \{ssn\}$ —— (1)

i.e ~~$R_1 \cap R_3 \Rightarrow R_1$~~

∵ ssn → ssn, ename

i.e $(\{ssn, ename\} \cap \{ssn, pnumber, hours\}) \rightarrow$

$\{ssn, ename\}$ —— (2)

$\Rightarrow \{ssn\} \rightarrow \{ssn, ename\}$ (from (1) (2) RHS)

∴ $R_1 \cap R_3 \rightarrow R_1$

Decomposition of R_1 & R_3 is lossless

* Decomposition of R_2 and R_3

{pnumber, pname, plocation} ∩ {ssn, pnumber, hours}

= {pnumber} —— (3)

From FD given pnumber → {pname, plocation}

∴ {pnumber, pname, plocation} ∩ {ssn, pnumber, hours}

= {pnumber, plocation, pname} —— (4) (from FD)

∴ $R_2 \cap R_3 \rightarrow R_2$ (from (3) & (4) RHS)

Decomposition of R_2 & R_3 is lossless

Thus we can conclude that decomposition of R into R_1, R_2 and R_3 is **lossless join** decomposition