Database HW4 鄧鵬宇 二資工三

14.24. Consider the universal relation R = {A, B, C, D, E, F, G, H, I, J} and the set of functional dependencies F = {{A, B}→{C}, {A}→{D, E}, {B}→{F}, {F}→{G, H}, {D}→{I, J}}. What is the key for R? Decompose R into 2NF and then 3NF relations.

AB 是 R 的 KEY。

R to 2NF

R1（A，D，E，I，J）F1={ A →DE，　D→IJ }

R2（B，F，G，H）F2={ B →F, F→GH}

R3（A，B，C）F3={ AB→C}

3NF

R1 to R4（A，D，E）and R5（D，I，J）

R2 to R6（B，F）and R7（F，G，H）

R to

{ R3<（A，B，C），F3={AB →C }>，R4<（A，D，E），F4={A →DE }>，R5<（D，I，J），F5={D →IJ }>，R6<（B，F），F6={B →F }>，R7<（F，G，H），F7={F →GH }>}

14.27. Consider a relation R(A, B, C, D, E) with the following dependencies:

AB → C, CD → E, DE → B

Is AB a candidate key of this relation? If not, is ABD? Explain your answer.

AB 只包括了 ABC，所以不是候選碼。

ABD 包括了所有值，而且子集不能滿足要求，所以是候選碼。

14.30

是 1NF。所有屬性不可以再次細分。

不是 2NF。存在函數依賴，Car# → DateSold　Car# → DiscountAmount　Salesman# → Commission%

不是 3NF。存在傳遞函數依賴 Car# → DateSold → DiscountAmount

2NF

CAR_SALE1(Car#, DateSold, DiscountAmount)

CAR_SALE2(Car#, Salesman#)

CAR_SALE3(Salesman#, Commission%)

3NF

CAR_SALES1A(Car#, DateSold)

CAR_SALES1B(DateSold, DiscountAmount)

CAR_SALE2(Car#, Salesman#)

CAR_SALE3(Salesman#, Commission%)

(20%) Suppose you are given a relation $R$ with four attributes $ABCD$. For each of the following sets of FDs, assuming those are the only dependencies that hold for $R$, do the following: (a) Identify the candidate key(s) for $R$. (b) Identify the best normal form that $R$ satisfies (1NF, 2NF, 3NF, or BCNF). (c) If $R$ is not in BCNF, decompose it into a set of BCNF relations that preserve the dependencies.

a. $C \rightarrow D$, $C \rightarrow A$, $B \rightarrow C$
b. $AB \rightarrow C$, $AB \rightarrow D$, $C \rightarrow A$, $D \rightarrow B$

a

(a) R 的 key 是 B

(b) 2NF，存在函數依賴 $B \rightarrow C$，$C \rightarrow D$

(c) R1（BC），R1（DC），R1（AC）

b

(a) R 的 key 是 AB CD AD BC

(b) 3NF，因爲 $C \rightarrow A$ 不是 BCNF。

(c) R1（AC），R2（BCD）

22.27. What implications would a no-steal/force buffer management policy have on checkpointing and recovery?

No-steal 表示在事務提交前，由此更新的 cache（buffer）頁不能寫入磁盤，force 表示更新的頁面在事務提交時寫入磁盤。

在 No-steal 下，所有修改的記憶體緩衝區寫入硬盤的檢查點都無法寫入由未提交事務更新的頁面。

在 force 下，事務完成更新就會被推送至硬盤，如果發生 failure，將需要重做，不過無論如何不需要撤銷因爲未提交的更新不會傳送至硬盤。