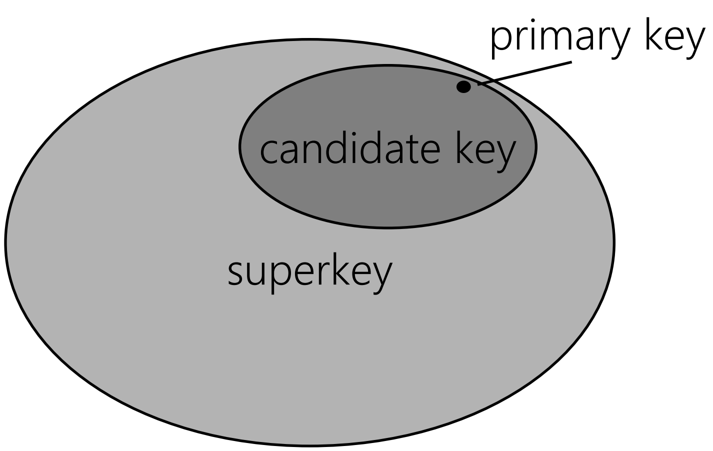
Database Systems

Functional Dependencies and 3NF

Review: Key

- Superkey
  $K \to R$
- Candidate Key
  $K \to R$
  no $K' \subset K$, s.t. $K' \to R$ (minimal最短，不能再分割出key)
- Primary Key
  - 关系表设计中用于识别元组的候选键



Key

- Superkey – 唯一标识关系表中的属性（集）
- Candidate Key (also called key) – A minimal superkey (cannot remove any attribute to make it as a superkey)

- Primary Key – 关系表设计中用于识别元组的候选键

Review: The Closure of FD

- 给定一个FD集合F, 我们可以计算其closure（闭包）
- 阿姆斯特朗公理:
  - Reflexivity
    If $X \supseteq Y$, then $X \rightarrow Y$
  - Augmentation
    If $X \rightarrow Y$, then $XZ \rightarrow YZ$
  - Transitivity
    If $X \rightarrow Y$, $Y \rightarrow Z$, then $X \rightarrow Z$
- 衍生规则:
  - Decomposition
    If $X \rightarrow YZ$, then $X \rightarrow Y$ and $X \rightarrow Z$
  - Union
    If $X \rightarrow Y$ and $X \rightarrow Z$, then $X \rightarrow YZ$
  - Pseudo-transitivity
    If $X \rightarrow Y$ and $WY \rightarrow Z$, then $WX \rightarrow Z$

Review: The Closure of Attributes

Definition:

X, Y are attributes of a relation R:

$X \to Y$ is in $F^+ \Leftrightarrow Y \subseteq X^+$

Algorithm:

- $X^{(0)} := X$
- Repeat
  $X^{(i+1)} := X^{(i)} \cup Z$,
  where Z is the set of attributes such that there exists $Y \to Z$ in F, and $Y \subset X^{(i)}$
- Until $X^{(i+1)} := X^{(i)}$
- Return $X^{(i+1)}$

Example:

Given R = (loan_no, amount, branch_name, customer_name)

- If loan_no $\to$ amount
  then $loan\_no^+$ = {loan_no, amount}
- If we also have loan_no $\to$ branch_name
  then $loan\_no^+$ = {loan_no, amount, branch_name}
- If we also have loan_no $\to$ customer_name
  then $loan\_no^+$ = {loan_no, amount, branch_name, customer_name}

Review: Canonical Cover 规范覆盖 of FD

Definition:

A canonical cover for F is a set of dependencies F_c such that

- 二者等价
- F_c 冗余
- 每个FD的左侧属性都是唯一的

- Algorithm to find canonical cover of F:

  repeat

  消除冗余规则：

  $X_1 \rightarrow Y_1$ and $X_1 \rightarrow Y_2$ with $X_1 \rightarrow Y_1 Y_2$

  Find a functional dependency $X \rightarrow Y$ 包含冗余属性，可能是在X侧，也可能在Y侧

  Delete it from $X \rightarrow Y$

  until F does not change

Review: Normalization

- Decomposition of a relation R with the following goals
  - Lossless (necessary) 无损的
    Information lost?
  - Dependency preservation (desirable) FD保留的
    $(\cup_i F_i)^+ = F^+$?
  - Good form
    1NF, 2NF, 3NF, BCNF

- 一般来说，当且仅当以下FD中至少有一个存在于 F^+ 中时，将 R 分解为 R1 和 R2 才是无损的。
  - $R_1 \cap R_2 \rightarrow R_1$
  - $R_1 \cap R_2 \rightarrow R_2$
  - 即，两个schema的交集属性构成至少其中一个schema的key。

Review: Normalization

- Decomposition of a relation R with the following goals
  - Lossless (necessary) 无损的
    Information lost?
  - Dependency preservation (desirable) FD保留的
    $(\cup_i F_i)^+ = F^+$?
  - Good form
    1NF, 2NF, 3NF, BCNF

2NF:

R is in 2NF if and only if
for each FD: $X \rightarrow \{A\}$ in F^+
Then
$A \in X$ (the FD is trivial), OR
X is not a proper subset of a candidate key for R, OR
A is a prime attribute

3NF:

R is in 3NF if and only if
for each FD: $X \rightarrow \{A\}$ in F^+
Then
$A \in X$ (trivial FD), OR
X is a superkey for R, OR
A is prime attribute for R

- A prime attribute is an attribute that is part of a candidate key

Exercise 1: The Closure of Attributes

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

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

Compute A^+ and B^+:

$A^+ := \{A\}$

$:= \{A, B, C\}$ $A \rightarrow BC$ and $\{A\} \subset A^+$

$:= \{A, B, C, D\}$ $B \rightarrow D$ and $\{B\} \subset A^+$

$:= \{A, B, C, D, E\}$ $CD \rightarrow E$ and $\{C, D\} \subset A^+$

ends because A^+ stops changing

$B^+ := \{B\}$

$:= \{B, D\}$ $B \rightarrow D$ and $\{B\} \subset B^+$

ends because B^+ stops changing

Exercise 2: Candidate Keys

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

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

List all candidate keys of R.

- We have A^+ = {A, B, C, D, E} in Exercise 1,
  then $A \to ABCDE$, it is a candidate key of R.
- Since $E \to A$,
  then $E \to ABCDE$. (transitivity)
- Since $CD \to E$,
  then $CD \to ABCDE$. (transitivity)
- Since $B \to D$,
  then $BC \to CD$, then $BC \to ABCDE$. (augmentation, transitivity)

So A, E, CD, BC are candidate keys of R.

Exercise 3: Compute Canonical Cover

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

F = {AC$\rightarrow$E, ACD$\rightarrow$B, CE$\rightarrow$D, B$\rightarrow$E}

Find the canonical cover of F.

Algorithm:

Repeat

Union

$X_1 \rightarrow Y_1$ and $X_1 \rightarrow Y_2$ replaced with $X_1 \rightarrow Y_1 Y_2$

Find an extraneous attribute

If an extraneous attribute is found in $X \rightarrow Y$, delete it from $X \rightarrow Y$

Until F does not change

Exercise 3: Compute Canonical Cover (cont)

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

F = {AC→E, ACD→B, CE→D, B→E}

Find the canonical cover of F.

First loop:

Union

$Fc^{(1)}$ = {AC→E, ACD→B, CE→D, B→E}

Find an extraneous attribute

Consider ACD→B:

D is extraneous because AC→E and CE→D

Remove D in ACD→B

$Fc^{(1)}$ = {AC→E, AC→B, CE→D, B→E}

Exercise 3: Compute Canonical Cover (cont)

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

$Fc^{(1)} = \{AC \rightarrow E, AC \rightarrow B, CE \rightarrow D, B \rightarrow E\}$

Second loop:

Union

$Fc^{(2)} = \{AC \rightarrow BE, CE \rightarrow D, B \rightarrow E\}$

Find an extraneous attribute

Consider $AC \rightarrow BE$:

E is extraneous because $B \rightarrow E$

Remove E in $AC \rightarrow BE$

$Fc^{(2)} = \{AC \rightarrow B, CE \rightarrow D, B \rightarrow E\}$

Exercise 3: Compute Canonical Cover (cont)

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

$Fc^{(2)} = \{AC \rightarrow B, CE \rightarrow D, B \rightarrow E\}$

Third loop:

Union

$Fc^{(3)} = \{AC \rightarrow B, CE \rightarrow D, B \rightarrow E\}$

Find an extraneous attribute

No extraneous attributes found

Ends because Fc stops changing

$Fc = \{AC \rightarrow B, CE \rightarrow D, B \rightarrow E\}$

Exercise 3: Compute Canonical Cover (cont)

- 去除无关属性的不同顺序可能会导致不同的 F_C
- Example:

R=(A, B, C, D)

FD = $\{A \rightarrow C, BC \rightarrow A, ABC \rightarrow D\}$

- In $ABC \rightarrow D$, A is extraneous or C is extraneous
- If we remove A first, we get $F_c = \{A \rightarrow C, BC \rightarrow AD\}$
- If we remove C first, we get $F_c = \{A \rightarrow C, BC \rightarrow A, AB \rightarrow D\}$

Exercise 4: Normal forms

- R=(A, B, C, D, E)
- FD = {A→BC, CD→E, B→D, E→A}

- Is R in 1NF?
  - Yes. Relational tables are always in 1NF.
- Is R in 2NF?
  - We found candidate keys: A, E, CD, BC.

A→BC	BC are prime attribute
CD→E	E is a prime attribute
B→D	D is a prime attribute
E→A	A is a prime attribute
So R is in 2NF	

2NF:

R is in 2NF if and only if
for each FD: X → {A} in F^+
Then
A ∈ X (the FD is trivial), OR
X is not a proper subset of a candidate key for R, OR
A is a prime attribute
（候选键的一部分）

2NF：防止候选键的真子集决定非主属性

Exercise 4: Normal forms (cont)

- $R=(A, B, C, D, E)$
- $FD = \{A \rightarrow BC, CD \rightarrow E, B \rightarrow D, E \rightarrow A\}$

- Is R in 3NF?
  - We found candidate keys: A, E, CD, BC.

$A \rightarrow BC$	A is a candidate key
$CD \rightarrow E$	CD is a candidate key
$B \rightarrow D$	D is a prime attribute
$E \rightarrow A$	E is a candidate key
So R is in 3NF	

3NF:

R is in 3NF if and only if

for each FD: $X \rightarrow \{A\}$ in F^+

Then

$A \in X$ (trivial FD), OR

X is a superkey for R, OR

A is prime attribute for R

3NF：防止非键决定非主属性

Exercise 4: Normal forms (cont)

- R=(A, B, C, D, E, F)
- FD = {A→B, BC→D, C→E, B→F}
- Is R in 2NF?
  - Candidate key: AC.

A→B	A is a proper subset of candidate key AND B is no proper subset t a prime attribute
BC→D	BC is not a proper subset of candidate key
C→E	C is a proper subset of candidate key AND E is not a prime attribute
B→F	B is not a proper subset of candidate key
A→B or C→E makes R not in 2NF	

2NF:

R is in 2NF if and only if
for each FD: $X \rightarrow \{A\}$ in F^+
Then
$A \in X$ (the FD is trivial), OR
X is not a proper subset of a candidate key for R, OR
A is a prime attribute

2NF：防止候选键的真子集决定非主属性

Exercise 4: Normal forms (cont)

- R=(A, B, C, D, E, F)
- FD = {A→B, BC→D, C→E, B→F}

- Is R in 3NF?
  - 3NF ⊂ 2NF ⊂ 1NF, R is not in 2NF, so R is not in 3NF either.
  - Candidate key: AC.

3NF:

R is in 3NF if and only if

for each FD: $X \rightarrow \{A\}$ in F^+

Then

$A \in X$ (trivial FD), OR

X is a superkey for R, OR

A is prime attribute for R

A→B	A is not a super-key AND B is not a prime attribute
BC→D	BC is not a super-key AND D is not a prime attribute
C→E	C is not a super-key AND E is not a prime attribute
B→F	B is not a super-key AND F is not a prime attribute
Any one of the FD makes R not in 3NF	

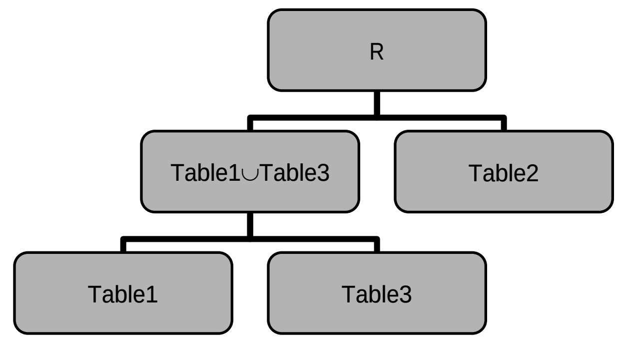
3NF：防止非键决定非主属性

Exercise 5: Decomposition

R = (A, B, C, D, E, F, G, H)

F = {AC→G, D→EG, BC→D, CG→BD, ACD→B, CE→AG}

- A decomposition of R:
  - Table1: (A, B, C, D)
  - Table2: (D, E, G)
  - Table3: (A, C, D, F, H)



- Is it lossless?

  Yes

  A decomposition of R into R1 and R2 is **lossless** if and only if the **common attributes of R1 and R2 is a candidate key for R1 or R2**
  - (Table1 ∪ Table3) ∩ Table2 = D (candidate key of Table2)
  - Table1 ∩ Table3 = ACD (candidate key of Table1)

- 一般来说，当且仅当以下FD中至少有一个存在于 F^+ 中时，将 R 分解为 R1 和 R2 才是无损的。
  - $R_1 \cap R_2 \rightarrow R_1$
  - $R_1 \cap R_2 \rightarrow R_2$
  - 即，两个schema的交集属性构成至少其中一个schema的key。在上述例子中，Position -> T3

Exercise 5: Decomposition (cont)

R = (A, B, C, D, E, F, G, H)

F = {AC→G, D→EG, BC→D, CG→BD, ACD→B, CE→AG}

- A decomposition of R:
  - Table1: (A, B, C, D)
  - Table2: (D, E, G)
  - Table3: (A, C, D, F, H)
- Is it dependency preserving?
  - No (CG→BD is lost)

- Then the decomposition is dependency preserving if and only if

$$(\cup F_i)^+ = F^+$$

即所有子表格FD并集的闭包与原来表格FD闭包一致。

Exercise 6: 3NF

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

F = {AB→C, C→B, A→D}

- 回顾下3NF的分解算法

Algorithm for 3NF Decomposition:

Let R be the initial table with FDs F

Compute the canonical cover F_c of F

S=∅

for each FD X→Y in the canonical cover F_c

S=S∪(X,Y)

if no schema contains a candidate key for R

Choose any candidate key K

S=S ∪ K

Exercise 6: 3NF (cont)

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

F = {$AB \rightarrow C$, $C \rightarrow B$, $A \rightarrow D$}

- List all candidate keys of R.
  - {ABE, ACE}
- Compute 正则覆盖（最小函数依赖集） F_c
  - F and F_c are equivalent （闭包一样的）
  - F_c contains no redundancy
  - Each left hand side of functional dependency in F_c is unique
- We get

  F_c = {$AB \rightarrow C$, $C \rightarrow B$, $A \rightarrow D$}, here F_c is the same as F.

Exercise 6: 3NF (cont)

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

F = {AB→C, C→B, A→D}

- List all candidate keys of R.
  - {ABE, ACE}
- Compute F_c, we get

  F_c = {AB→C, C→B, A→D}
- For each FD, we generate a table:

  R_1 = {A, B, C}, F_1 = {AB→C, C→B}

  R_2 = {A, D}, F_2 = {A→D}

分解结束了吗？

- No need to generate another table for FD {C→B}, because R_1 already contains B and C.

if no schema contains a candidate key for R

Choose any candidate key K

S=S ∪ K

Exercise 6: 3NF (cont)

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

F = {AB→C, C→B, A→D}

- Any table in S contains a candidate key for R?

  No

- Choose any candidate key (ABE) to generate a new table:

  Generate R_3 = {A, B, E}

- So a possible 3NF decomposition of R is:

  R_1 = {A, B, C}, R_2 = {A, D}, R_3 = {A, B, E}

- Lossless?

  Yes

- Dependency preserving?

  Yes