

第5章

关系模式的规范化设计

——模式分解





关系模式的规范化设计

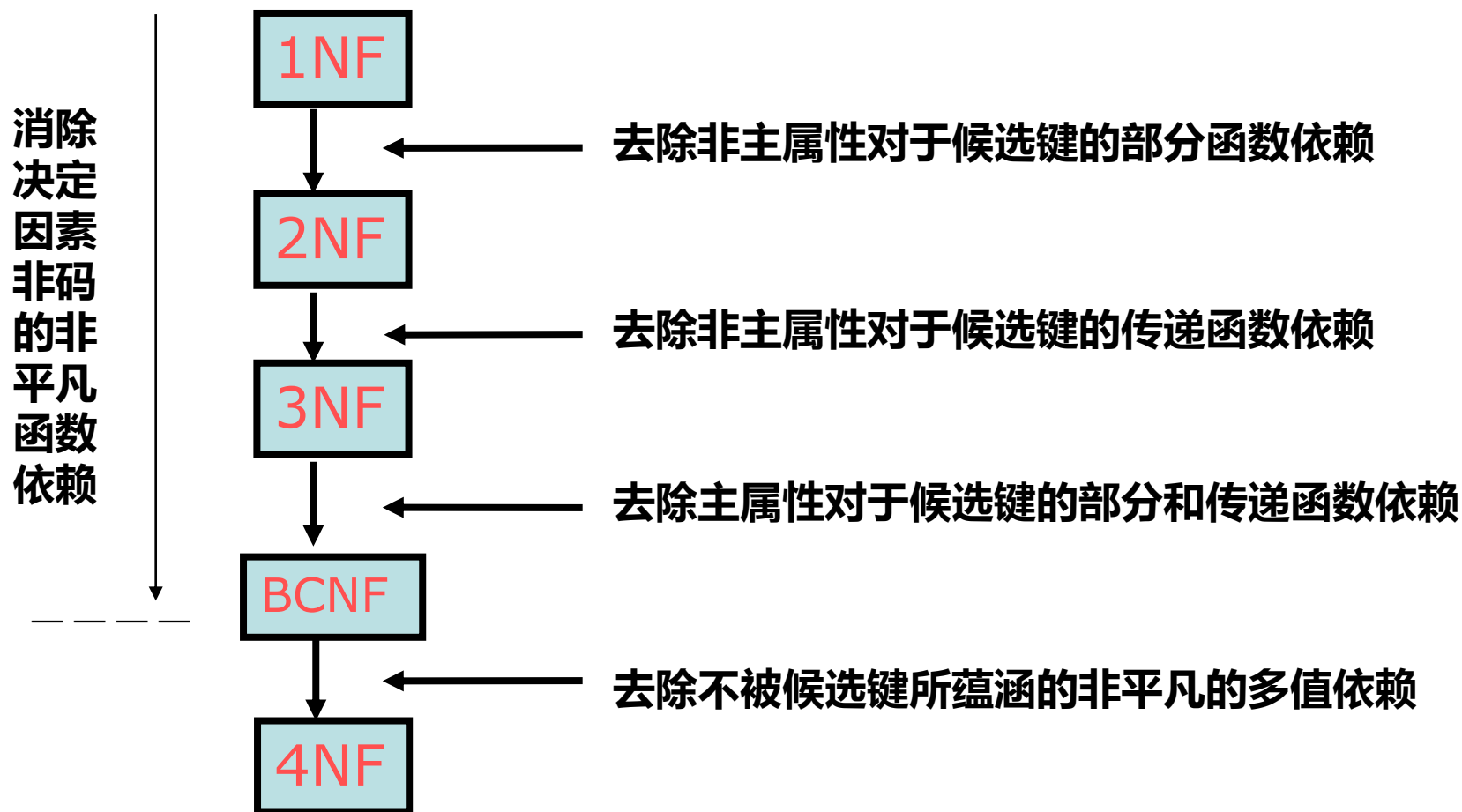
- 所要解决的问题
 - 什么是“好”的关系数据模式
 - 如何评价一个好的关系数据模式
 - 如何设计一个“好”的关系数据模式



关系模式的规范化设计

- 主要内容
 - 关系模式的设计问题
 - 关系模式规范化的基本概念和理论
 - 关系模式分解的理论基础和算法

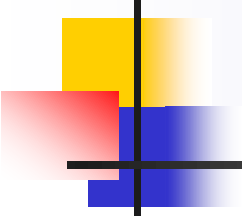
回顾





回顾

- Armstrong公理系统
 - 逻辑蕴涵
 - Armstrong公理系统推理规则
 - FD的逻辑蕴涵
 - 函数依赖集 F 的闭包 F^+
 - 属性集闭包
 - 函数依赖集等价和最小函数依赖集
- 候选码及其求解方法



模式分解

- 主要内容
 - 模式分解的概念
 - 分解的无损连接性和保持函数依赖性
 - 模式分解算法

模式分解的概念

- 分解的概念

设有一关系模式 $R(U, F)$ ，若用一关系模式的集合 $\rho = \{ R_1(U_1, F_1), R_2(U_2, F_2), \dots, R_n(U_n, F_n) \}$ 来取代，其中 $U = \bigcup U_i$ ，并且没有 $U_i \subseteq U_j$ ， $1 \leq i, j \leq n$ ， F_i 是 F 在 U_i 上的投影。则关系模式的集合 ρ 为 R 的一个分解

$$\rho = \{ R_1, R_2, \dots, R_n \}$$

模式分解的概念

- 分解的概念
 - **F在 U_i 上的投影**

$$F_i = \{X \rightarrow Y \mid X \rightarrow Y \in F^+ \wedge XY \subseteq U_i\}$$

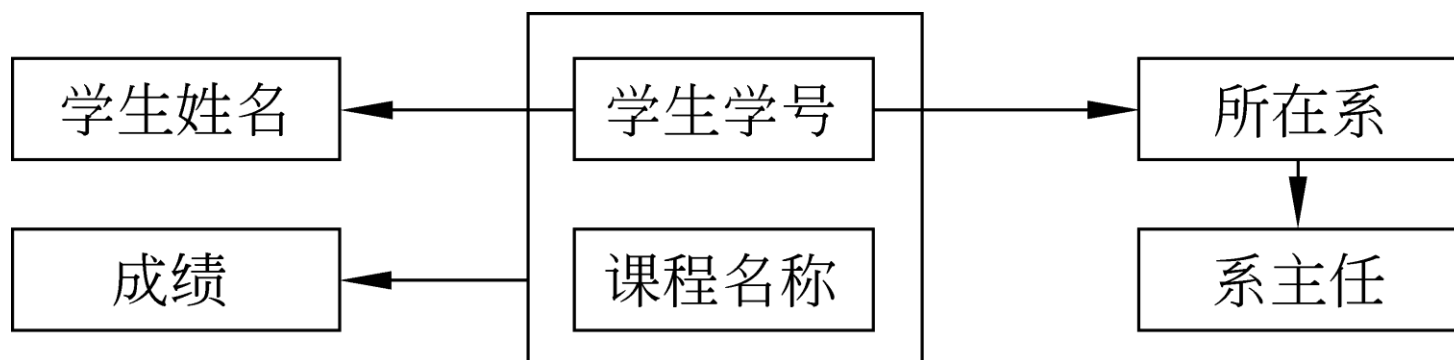
思考：

设有关系模式R (A, B, C, D)，F是R上成立的FD集 $F = \{AB \rightarrow C, D \rightarrow B\}$ ，则F在模式ACD上的投影为 AD \rightarrow C；F在模式AC上的投影为 空。

模式分解的概念

- 分解的概念

【例】 $R(\text{学生学号}, \text{学生姓名}, \text{所在系}, \text{系主任}, \text{课程名称}, \text{成绩})$ ， $F = \{\text{学生学号} \rightarrow \text{学生姓名}, \text{学生学号} \rightarrow \text{所在系}, \text{所在系} \rightarrow \text{系主任}, (\text{学生学号}, \text{课程名称}) \rightarrow \text{成绩}\}$



模式分解的概念

- 分解的概念

R (**学生学号**, 学生姓名, 所在系, 系主任, **课程名称**, 成绩)

1NF



去除非主属性对于键的部分函数依赖

R1 (**学生学号**, **课程名称**, 成绩)

R2 (**学生学号**, 学生姓名, 所在系, 系主任)

2NF



去除非主属性对于键的传递函数依赖

R1 (**学生学号**, **课程名称**, 成绩)

R3 (**学生学号**, 学生姓名, 所在系)

R4 (**所在系**, 系主任)

3NF

模式分解的概念

- 分解的概念

R ({**学生学号**, 学生姓名, 所在系, 系主任, **课程名称**, 成绩} {**学生学号**→学生姓名, **学生学号**→所在系, 所在系→系主任, (**学生学号**, **课程名称**)→成绩}



$\rho_1 = \{$
 R_1 ({**学生学号**, **课程名称**, 成绩}, { (**学生学号**, **课程名称**) → 成绩}),
 R_3 ({**学生学号**, 学生姓名, 所在系}, {**学生学号**→学生姓名, **学生学号**→所在系}),
 R_4 ({**所在系**, 系主任}, {所在系→系主任})
 $\}$

模式分解的概念

- 分解的概念

R ({**学生学号**, 学生姓名, 所在系, 系主任, **课程名称**, 成绩} {**学生学号**→学生姓名, 学生学号→所在系, 所在系→系主任, (学生学号, 课程名称)→成绩}



$\rho_2 = \{R_1$ (学生学号),
 R_2 (学生姓名),
 R_3 (所在系),
 R_4 (系主任),
 R_5 (课程名称),
 R_6 (成绩) }

不具有无损连接性

$U = \bigcup U_i$, 并且没有 $U_i \subseteq U_j$, $1 \leq i, j \leq n$

F_i 是 F 在 U_i 上的投影不存在非平凡的函数依赖。

模式分解的概念

- 分解的概念

R ({**学生学号**, 学生姓名, 所在系, 系主任, **课程名称**, 成绩} { $\text{学生学号} \rightarrow \text{学生姓名}$, $\text{学生学号} \rightarrow \text{所在系}$, $\text{所在系} \rightarrow \text{系主任}$, $(\text{学生学号}, \text{课程名称}) \rightarrow \text{成绩}$ })



$\rho_3 = \{$

R_1 ({**学生学号**, **课程名称**, 成绩}, { $(\text{学生学号}, \text{课程名称}) \rightarrow \text{成绩}$ }),

R_2 ({**学生学号**, 学生姓名, 所在系}, { $\text{学生学号} \rightarrow \text{学生姓名}$, $\text{学生学号} \rightarrow \text{所在系}$ }),

R_3 ({**学生学号**, 系主任}, { $\text{学生学号} \rightarrow \text{系主任}$ }) }

没有保持函数依赖

$U = \cup U_i$, 并且没有 $U_i \subseteq U_j$, $1 \leq i, j \leq n$

F_i 是 F 在 U_i 上的投影。



模式分解的概念

- 分解的概念
 - 一个关系模式的分解并不是唯一的。
 - 理想的分解应该是既具有无损连接性又保持函数依赖，这样的分解是等价分解。
 - 等价分解
 - 分解具有无损连接性，即数据等价
 - 分解保持函数依赖，即语义等价
 - 分解既具有无损连接性又保持函数依赖，即数据等价并且语义等价

无损连接分解

- 无损连接分解

- 设 $\rho = \{ R_1 (U_1, F_1) , \dots , R_k (U_k, F_k) \}$ 是 $R (U, F)$ 的一个分解, r 是 $R (U, F)$ 的一个关系, 则

$\pi_{R_i}(r) = \{ t.U_i | t \in r \}$ 为 r 在子模式 R_i 上的投影,

$m_\rho(r) = \pi_{R_1}(r) \bowtie \pi_{R_2}(r) \bowtie \dots \bowtie \pi_{R_k}(r)$ 是 r 在 ρ 中各关系模式上投影的连接。

- 若对 $R \langle U, F \rangle$ 的任何一个关系 r 均有 $r = m_\rho(r)$ 成立, 则称分解 ρ 具有无损连接性。

分解 ρ 是无损分解

无损连接分解

- 算法：检验一个分解是否是无损分解

– 输入：

关系模式 $R (U, F)$,

$U = \{A_1, A_2, \dots, A_n\}$;

R 上的分解 $\rho = \{ R_1 (U_1, F_1) , R_2 (U_2, F_2) , \dots, R_k (U_k, F_k) \}$ 。

设 F 是最小函数依赖集。

– 输出：

判断 ρ 相对于 F 是否具有无损分解特性。

无损连接分解

- 检验一个分解是否无损的算法
– 步骤

1) 建立一张**k**行**n**列的表，
每行对应分解中的一个关系模式 R_i ，每列对应一个属性 A_j ，若属性 $A_j \in U_i$ ，则在 i 行 j 列处填 a_{ij} ，否则填 b_{ij} 。

	A_1	..	A_n
$R_1 (U_1, F_1)$	a_1		
:			
$R_k (U_k, F_k)$			b_{kn}

$A_1 \in U_1$

$A_n \notin U_k$

无损连接分解

【例】设有关系模式 $R(U, F)$ ， $U=ABCDE$ ， $F=\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$ 。 $\rho=\{R_1(AD), R_2(AB), R_3(BE), R_4(CDE), R_5(AE)\}$ 是 R 的一个分解。试判断 ρ 是否为无损分解。

1) 建立一张**k**行**n**列的表，每行对应分解中的一个关系模式 R_i ，每列对应一个属性 A_j ，若属性 $A_j \in U_i$ ，则在 i 行 j 列处填 a_j ，否则填 b_{ij} 。

	A	B	C	D	E
R1(AD)	a1	b12	b13	a4	b15
R2(AB)	a1	a2	b23	b24	b25
R3(BE)	b31	a2	b33	b34	a5
R4(CDE)	b41	b42	a3	a4	a5
R5(AE)	a1	b52	b53	b54	a5

初始状态

无损连接分解

【例】设有关系模式 $R(U, F)$ ， $U=ABCDE$ ， $F=\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$ 。 $\rho=\{R_1(AD), R_2(AB), R_3(BE), R_4(CDE), R_5(AE)\}$ 是 R 的一个分解。试判断 ρ 是否为无损分解。

2) 把表格看成模式 R 的一个关系，反复检查 F 中的每个函数依赖在表格中是成立，若不成立，则修改表格中的值。

	A	B	C	D	E
R1(AD)	a1	b12	b13	a4	b15
R2(AB)	a1	a2	b23	b24	b25
R3(BE)	b31	a2	b33	b34	a5
R4(CDE)	b41	b42	a3	a4	a5
R5(AE)	a1	b52	b53	b54	a5

初始状态

无损连接分解

【例】设有关系模式 $R(U, F)$ ， $U=ABCDE$ ， $F=\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$ 。 $\rho=\{R_1(AD), R_2(AB), R_3(BE), R_4(CDE), R_5(AE)\}$ 是 R 的一个分解。试判断 ρ 是否为无损分解。

2) 对每一形如 $X \rightarrow Y$ 的FD做如下操作：

检查 X 中的属性所对应的列，找出 X 相等的行，将这些行中对应的 Y 中的属性列所对应的符号改为一**致**。即若其中之一为 a_j ，则都改成 a_j ；否则，将其都改成这些列中行号最小的符号 b_{ij} 。

	A	B	C	D	E
R1(AD)	a1	b12	b13	a4	b15
R2(AB)	a1	a2	b23	b24	b25
R3(BE)	b31	a2	b33	b34	a5
R4(CDE)	b41	b42	a3	a4	a5
R5(AE)	a1	b52	b53	b54	a5

对F的一次扫描

初始状态

无损连接分解

【例】设有关系模式 $R(U, F)$ ， $U=ABCDE$ ， $F=\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$ 。 $\rho=\{R_1(AD), R_2(AB), R_3(BE), R_4(CDE), R_5(AE)\}$ 是 R 的一个分解。试判断 ρ 是否为无损分解。

$A \rightarrow C$

	A	B	C	D	E
$R_1(AD)$	a1	b12	b13	a4	b15
$R_2(AB)$	a1	a2	b23	b24	b25
$R_3(BE)$	b31	a2	b33	b34	a5
$R_4(CDE)$	b41	b42	a3	a4	a5
$R_5(AE)$	a1	b52	b53	b54	a5

无损连接分解

【例】设有关系模式 $R(U, F)$ ， $U=ABCDE$ ， $F=\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$ 。 $\rho=\{R_1(AD), R_2(AB), R_3(BE), R_4(CDE), R_5(AE)\}$ 是 R 的一个分解。试判断 ρ 是否为无损分解。

$A \rightarrow C$

	A	B	C	D	E
R1(AD)	a1	b12	b13	a4	b15
R2(AB)	a1	a2	b13	b24	b25
R3(BE)	b31	a2	b33	b34	a5
R4(CDE)	b41	b42	a3	a4	a5
R5(AE)	a1	b52	b13	b54	a5

无损连接分解

【例】设有关系模式 $R(U, F)$ ， $U=ABCDE$ ， $F=\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$ 。 $\rho=\{R_1(AD), R_2(AB), R_3(BE), R_4(CDE), R_5(AE)\}$ 是 R 的一个分解。试判断 ρ 是否为无损分解。

$A \rightarrow C$

$B \rightarrow C$

	A	B	C	D	E
R1(AD)	a1	b12	b13	a4	b15
R2(AB)	a1	a2	b13	b24	b25
R3(BE)	b31	a2	b33	b34	a5
R4(CDE)	b41	b42	a3	a4	a5
R5(AE)	a1	b52	b13	b54	a5

无损连接分解

【例】设有关系模式 $R(U, F)$ ， $U=ABCDE$ ， $F=\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$ 。 $\rho=\{R_1(AD), R_2(AB), R_3(BE), R_4(CDE), R_5(AE)\}$ 是 R 的一个分解。试判断 ρ 是否为无损分解。

$A \rightarrow C$

$B \rightarrow C$

	A	B	C	D	E
R1(AD)	a1	b12	b13	a4	b15
R2(AB)	a1	a2	b13	b24	b25
R3(BE)	b31	a2	b13	b34	a5
R4(CDE)	b41	b42	a3	a4	a5
R5(AE)	a1	b52	b13	b54	a5

无损连接分解

【例】设有关系模式 $R(U, F)$ ， $U=ABCDE$ ， $F=\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$ 。 $\rho=\{R_1(AD), R_2(AB), R_3(BE), R_4(CDE), R_5(AE)\}$ 是 R 的一个分解。试判断 ρ 是否为无损分解。

$A \rightarrow C$

$B \rightarrow C$

$C \rightarrow D$

	A	B	C	D	E
R1(AD)	a1	b12	b13	a4	b15
R2(AB)	a1	a2	b13	b24	b25
R3(BE)	b31	a2	b13	b34	a5
R4(CDE)	b41	b42	a3	a4	a5
R5(AE)	a1	b52	b13	b54	a5

无损连接分解

【例】设有关系模式 $R(U, F)$ ， $U=ABCDE$ ， $F=\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$ 。 $\rho=\{R_1(AD), R_2(AB), R_3(BE), R_4(CDE), R_5(AE)\}$ 是 R 的一个分解。试判断 ρ 是否为无损分解。

$A \rightarrow C$

$B \rightarrow C$

$C \rightarrow D$

	A	B	C	D	E
R1(AD)	a1	b12	b13	a4	b15
R2(AB)	a1	a2	b13	a4	b25
R3(BE)	b31	a2	b13	a4	a5
R4(CDE)	b41	b42	a3	a4	a5
R5(AE)	a1	b52	b13	a4	a5

无损连接分解

【例】设有关系模式 $R(U, F)$ ， $U=ABCDE$ ， $F=\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$ 。 $\rho=\{R_1(AD), R_2(AB), R_3(BE), R_4(CDE), R_5(AE)\}$ 是 R 的一个分解。试判断 ρ 是否为无损分解。

$A \rightarrow C$
 $B \rightarrow C$
 $C \rightarrow D$
 $DE \rightarrow C$

	A	B	C	D	E
$R_1(AD)$	a1	b12	b13	a4	b15
$R_2(AB)$	a1	a2	b13	a4	b25
$R_3(BE)$	b31	a2	b13	a4	a5
$R_4(CDE)$	b41	b42	a3	a4	a5
$R_5(AE)$	a1	b52	b13	a4	a5

无损连接分解

【例】设有关系模式 $R(U, F)$ ， $U=ABCDE$ ， $F=\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$ 。 $\rho=\{R_1(AD), R_2(AB), R_3(BE), R_4(CDE), R_5(AE)\}$ 是 R 的一个分解。试判断 ρ 是否为无损分解。

$A \rightarrow C$
 $B \rightarrow C$
 $C \rightarrow D$
 $DE \rightarrow C$

	A	B	C	D	E
$R_1(AD)$	a1	b12	a3	a4	b15
$R_2(AB)$	a1	a2	a3	a4	b25
$R_3(BE)$	b31	a2	a3	a4	a5
$R_4(CDE)$	b41	b42	a3	a4	a5
$R_5(AE)$	a1	b52	a3	a4	a5

无损连接分解

【例】设有关系模式 $R(U, F)$ ， $U=ABCDE$ ， $F=\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$ 。 $\rho=\{R_1(AD), R_2(AB), R_3(BE), R_4(CDE), R_5(AE)\}$ 是 R 的一个分解。试判断 ρ 是否为无损分解。

$A \rightarrow C$

$B \rightarrow C$

$C \rightarrow D$

$DE \rightarrow C$

$CE \rightarrow A$

	A	B	C	D	E
$R_1(AD)$	a1	b12	a3	a4	b15
$R_2(AB)$	a1	a2	a3	a4	b25
$R_3(BE)$	b31	a2	a3	a4	a5
$R_4(CDE)$	b41	b42	a3	a4	a5
$R_5(AE)$	a1	b52	a3	a4	a5

无损连接分解

【例】设有关系模式 $R(U, F)$ ， $U=ABCDE$ ， $F=\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$ 。 $\rho=\{R_1(AD), R_2(AB), R_3(BE), R_4(CDE), R_5(AE)\}$ 是 R 的一个分解。试判断 ρ 是否为无损分解。

$A \rightarrow C$

$B \rightarrow C$

$C \rightarrow D$

$DE \rightarrow C$

$CE \rightarrow A$

	A	B	C	D	E
$R_1(AD)$	a1	b12	a3	a4	b15
$R_2(AB)$	a1	a2	a3	a4	b25
$R_3(BE)$	a1	a2	a3	a4	a5
$R_4(CDE)$	a1	b42	a3	a4	a5
$R_5(AE)$	a1	b52	a3	a4	a5

无损连接分解

【例】设有关系模式 $R(U, F)$ ， $U=ABCDE$ ， $F=\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$ 。 $\rho=\{R_1(AD), R_2(AB), R_3(BE), R_4(CDE), R_5(AE)\}$ 是 R 的一个分解。试判断 ρ 是否为无损分解。

$A \rightarrow C$

$B \rightarrow C$

$C \rightarrow D$

$DE \rightarrow C$

$CE \rightarrow A$

	A	B	C	D	E
R1 (AD)	a1	b12	a3	a4	b15
R2 (AB)	a1	a2	a3	a4	b25
R3 (BE)	a1	a2	a3	a4	a5
R4 (CDE)	a1	b42	a3	a4	a5
R5 (AE)	a1	b52	a3	a4	a5

无损连接分解

【例】设有关系模式 $R(U, F)$ ， $U=ABCDE$ ， $F=\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$ 。 $\rho=\{R_1(AD), R_2(AB), R_3(BE), R_4(CDE), R_5(AE)\}$ 是 R 的一个分解。试判断 ρ 是否为无损分解。

3) 若表有变化，则返回步骤2)，否则算法终止。

若某次修改后，有一行全为 a_1, a_2, \dots, a_n ，则算法结束。

	A	B	C	D	E
R1(AD)	a1	b12	b13	a4	b15
R2(AB)	a1	a2	b13	a4	b25
R3(BE)	a1	a2	a3	a4	a5
R4(CDE)	a1	b42	a3	a4	a5
R5(AE)	a1	b52	a3	a4	a5

无损连接分解

【例】设有关系模式 $R(U, F)$ ， $U=ABCDE$ ， $F=\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C, CE \rightarrow A\}$ 。 $\rho=\{R_1(AD), R_2(AB), R_3(BE), R_4(CDE), R_5(AE)\}$ 是 R 的一个分解。试判断 ρ 是否为无损分解。

定理：

ρ 为无损连接分解的充分必要条件是算法终止时，表中有一行为
 a_1, a_2, \dots, a_n 。

	A	B	C	D	E
$R_1(AD)$	a_1	b_{12}	b_{13}	a_4	b_{15}
$R_2(AB)$	a_1	a_2	b_{13}	a_4	b_{25}
$R_3(BE)$	a_1	a_2	a_3	a_4	a_5
$R_4(CDE)$	a_1	b_{42}	a_3	a_4	a_5
$R_5(AE)$	a_1	b_{52}	a_3	a_4	a_5

无损连接分解

- 判断一个关系模式分解成两个关系模式是否为无损分解的方法

定理：

设 $\rho=\{R_1(U_1), R_2(U_2)\}$ 是 $R(U)$ 的一个分解，则
 ρ 为无损分解的充分必要条件为

$$(U_1 \cap U_2) \rightarrow (U_1 - U_2) \in F^+$$

或
$$(U_1 \cap U_2) \rightarrow (U_2 - U_1) \in F^+$$

无损连接分解

证明：用检验分解无损算法构造的如下初始表。表中a和b的下标都已省略

[充分性] 因为 $(U_1 \cap U_2) \rightarrow (U_1 - U_2)$ ，则算法8-2终止时，对应于 $(U_1 - U_2)$ 的 $bb \dots b$ 全为 a ，因此， $\rho = \{R_1(U_1), R_2(U_2)\}$ 是 $R(U)$ 的一个无损连接分解。

[必要性] 若 $\rho = \{R_1(U_1), R_2(U_2)\}$ 是 $R(U)$ 的一个无损连接分解，则算法终止时，表中必有一行全为 a ，假设 R_1 行全为 a ，则 $(U_1 - U_2)$ 所对应的元素都应变为 a 。由于在表中，只有 $(U_1 \cap U_2)$ 所对应的列的元素相同。设 $U_1 - U_2$ 中有一个属性 A_i ，则根据算法将 A_i 所对应的元素都改为 a ，则必有 $S \rightarrow A_i \in F^+$ ，且 $S \subseteq (U_1 \cap U_2)$ 。根据自反律有 $(U_1 \cap U_2) \rightarrow S$ ；根据传递律有 $(U_1 \cap U_2) \rightarrow A_i$ ，即 $A_i \in (U_1 \cap U_2)^+$ 。所以 $(U_1 - U_2) \subseteq (U_1 \cap U_2)^+$ 即 $(U_1 \cap U_2) \rightarrow (U_1 - U_2)$ 是 $(U_1 \cap U_2)$ 所对应的元素都变为 a 的必要条件。

	$U_1 \cap U_2$	$U_1 - U_2$	$U_2 - U_1$
R_1	$aa \dots a$	$aa \dots a$	$bb \dots b$
R_2	$aa \dots a$	$bb \dots b$	$aa \dots a$

无损连接分解

【例5-13】 设有关系模式R (U, F) , $U=\{A,B,C,D,E\}$,
 $F=\{A \rightarrow BC, CD \rightarrow E, B \rightarrow D, E \rightarrow A\}$ 。

R的两个分解为 $\rho_1=\{ R_1(A,B,C), R_2(A,D,E)\}$,
 $\rho_2=\{R_1(A,B,C), R_2(C,D,E)\}$ 。试判断 ρ_1 、 ρ_2 是否为无损分解。

解：对于分解 $\rho_1=\{ R_1 (A,B,C) , R_2 (A,D,E) \}$
因为 $U_1 \cap U_2 = A$, $U_1 - U_2 = BC$, 且 $A \rightarrow BC \in F$
所以 $U_1 \cap U_2 \rightarrow U_1 - U_2$, ρ_1 为无损分解。

对于分解 $\rho_2=\{ R_1 (A,B,C) , R_2 (C,D,E) \}$
因为 $U_1 \cap U_2 = C$, $U_1 - U_2 = AB$, $U_2 - U_1 = DE$,
且 $C \rightarrow AB \notin F^+$, $C \rightarrow DE \notin F^+$ (如何来判断?)
即 $U_1 \cap U_2 \rightarrow U_1 - U_2 \notin F^+$ 或 $U_1 \cap U_2 \rightarrow U_2 - U_1 \notin F^+$,
所以 ρ_2 为有损分解。

保持函数依赖分解

- 保持函数依赖
 - 对于关系模式 $R(U, F)$ ，设 $\rho = \{ R_1(U_1, F_1), R_2(U_2, F_2), \dots, R_n(U_n, F_n) \}$ 是 R 的一个分解，若 $F^+ = (\cup F_i)^+$ ，则称分解 ρ 保持函数依赖。

引理5.3 $F^+ = G^+$ 的充分必要条件是 $F \subseteq G^+$ 且 $G \subseteq F^+$

判断分解 ρ 是否保持函数依赖的方法：

令 $G = \cup F_i$ ，即只需判断 $F \subseteq G^+$ 且 $G \subseteq F^+$ 是否成立。
而要判定 $F \subseteq G^+$ 或 $G \subseteq F^+$ ，只需逐一对 F 或 G 中的函数依赖 $X \rightarrow Y$ 考察 Y 是否属于 X_G^+ 或 X_F^+ 就行了。

分解的保持函数依赖性

【例】 设关系模式 $R(ABCD)$, $F=\{A\rightarrow B, B\rightarrow C, C\rightarrow D, D\rightarrow A\}$, 试判断分解 $\rho=\{R_1(AB), R_2(BC), R_3(CD)\}$ 是否保持函数依赖。

解答:

1) 要先根据分解的定义, 确定分解后的函数依赖, 然后再加以判断。利用Armstrong公理的传递律, 可将 ρ 分解为

$$\begin{aligned}\rho = \{ & R_1(\{A,B\}, \{A\rightarrow B, B\rightarrow A\}), \\ & R_2(\{B,C\}, \{B\rightarrow C, C\rightarrow B\}), \\ & R_3(\{C,D\}, \{C\rightarrow D, D\rightarrow C\})\}\end{aligned}$$

分解的保持函数依赖性

【例】 设关系模式 $R(ABCD)$ ， $F=\{A\rightarrow B, B\rightarrow C, C\rightarrow D, D\rightarrow A\}$ ，试判断分解 $\rho =\{R_1(AB), R_2(BC), R_3(CD)\}$ 是否保持函数依赖。

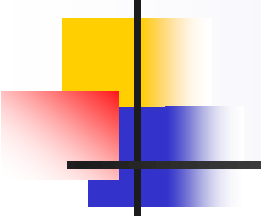
解答：

2) 判断 $D\rightarrow A$ 是否为 G 所覆盖即可

$G=\{A\rightarrow B, B\rightarrow A, B\rightarrow C, C\rightarrow B, C\rightarrow D, D\rightarrow C\}$

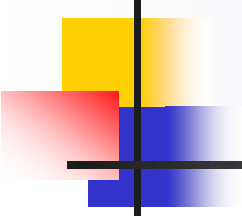
求得 $D_G^+ = \{A, B, C, D\}$

因为 $A \in D_G^+$ ，所以 $D\rightarrow A \in G^+$ ，故 ρ 保持函数依赖。



模式分解算法

- 关系模式的规范化就是将低一级的关系模式分解为若干高一级的范式。
- 对数据库应用设计而言，违反数据等价或依赖等价的分解很难说是一个好的模式设计。
- 在函数依赖范畴内，希望分解的结果能达到BCNF，并且能保持无损连接性和函数依赖性。

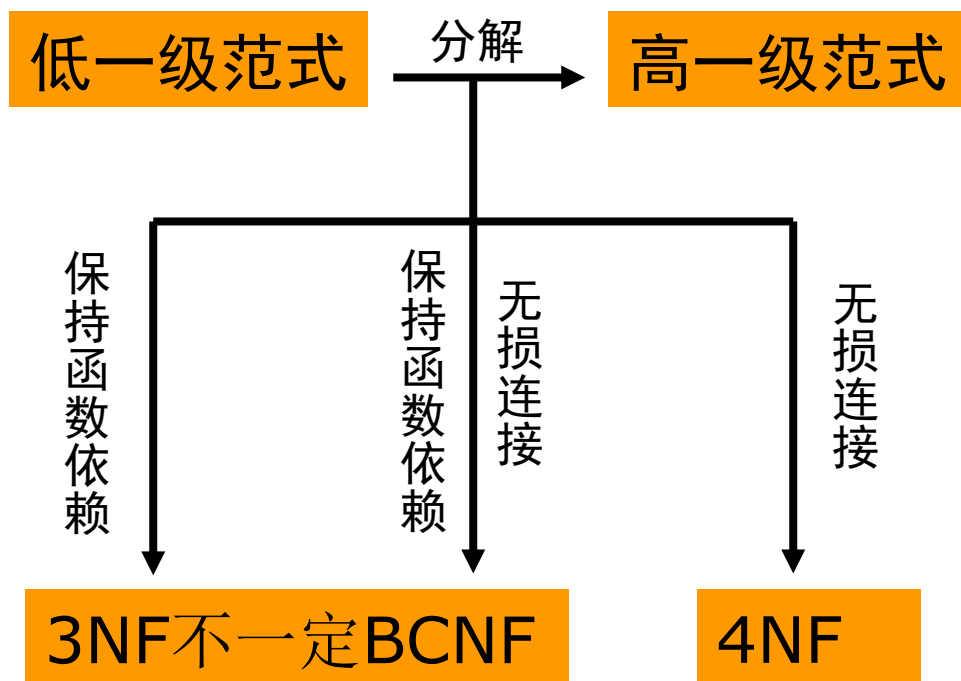


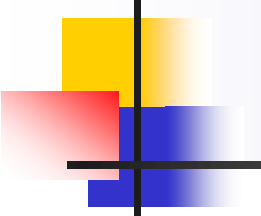
模式分解算法

- 这三个目标有时不能同时满足，一般只能做到如下几点：
 - 若分解保持函数依赖，那么模式总可以分解到3NF，但不一定能达到BCNF；
 - 若分解具有无损连接性，那么模式一定能分解到4NF；
 - 若分解既保持函数依赖，又具有无损连接性，那么模式可以分解到3NF，但不一定能达到BCNF。

模式分解算法

- 关于模式分解的几个重要事实：





模式分解算法

- 数据库设计者在设计关系数据库时，应权衡利弊，尽可能使数据库模式保持最好的特性。
 - 一般尽可能设计成BCNF模式集；
 - 如果设计成BCNF模式集达不到保持函数依赖的特性，那么只能降低要求，设计成3NF模式集，以求达到保持依赖和无损分解的特性。



模式分解算法

- 分解关系模式为满足3NF的一个无损且保持函数依赖的分解算法
 - 输入：关系模式 $R (U, F)$
 - 输出： R 的一个具有无损连接性且保持函数依赖的满足3NF的分解 ρ

模式分解算法

- 分解关系模式为满足3NF的一个无损且保持函数依赖的分解算法

1) 求F的最小覆盖 F_m , 令 $F = F_m$ 。

2) 若有 $X \rightarrow A \in F$, 且 $XA = U$, 输出 $\rho = \{R\}$, 算法结束。

3) 对F按具有相同左部的原则分组, 即F中所有形如

$X \rightarrow A_1, X \rightarrow A_2, \dots, X \rightarrow A_n$ 的函数依赖为一组。每一分组中包含的函数依赖子集 F_i 所涉及的全部属性组成一个属性集 U_i 。若 $U_i \subseteq U_j$ ($i \neq j$) 就去掉 U_i 。

4) 将R中不在F中出现的属性, 单独构成一属性集 U_i 。

模式分解算法

- 分解关系模式为满足3NF的一个无损且保持函数依赖的分解算法
 - 5) 将 U_i 及 F 在 U_i 上的投影 F_i 构成一个关系模式 $R_i(U_i, F_i)$ ，由于 F 是最小函数依赖集， $R_i(U_i, F_i)$ 一定是属于以 X 为候选键的3NF。并且 F_i' 一定被 F_i 所包含，因此各关系模式 $R_i(U_i, F_i)$ 组成的分解 ρ 是保持函数依赖的。
 - 6) 若某个 U_i 中已包含 R 中的候选键，则该分解也就具有无损连接性。若 U_i 中均不包含 R 中的候选键，则在分解 ρ 的结果上再并入一个关系模式 $R_k(K, F_k)$ ，其中 K 为 R 的某一候选键， F_k 为 F 在 K 上的投影。

模式分解算法

【例】 关系模式 $R(U, F)$ ，其中 $U = \{E, G, H, I, J\}$ ， $F = \{E \rightarrow I, J \rightarrow I, I \rightarrow G, GH \rightarrow I, IH \rightarrow E\}$ ，将 R 分解为3NF，并使之具有无损连接性和保持函数依赖特性。

解答：

- (1) F 已为最小函数依赖集。**
- (2) 没有类似 $X \rightarrow A \in F$ ，且 $XA=U$ 的函数依赖。**
- (3) 对 F 按具有相同左部的原则分组，得到每组所对应的属性集 $U_1 = EI$ 、 $U_2 = IJ$ 、 $U_3 = GI$ 、 $U_4 = GHI$ 、 $U_5 = EHI$ 。因为 $U_1 \subseteq U_5$ 、 $U_3 \subseteq U_4$ ，所以去掉 U_1 、 U_3 。**
- (4) R 中的所有属性均在 F 中出现。**

模式分解算法

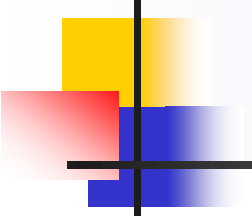
【例(续)】关系模式 $R(U, F)$ ，其中 $U = \{E, G, H, I, J\}$ ， $F = \{E \rightarrow I, J \rightarrow I, I \rightarrow G, GH \rightarrow I, IH \rightarrow E\}$ ，将 R 分解为3NF，并使之具有无损连接性和保持函数依赖特性。

解答：

(5) 将 U_i 及 F 在 U_i 上的投影 F_i 构成一个关系模式 $R_i(U_i, F_i)$ ，则分解 $\rho = \{R_1(\{I, J\}, \{J \rightarrow I\}), R_2(\{G, H, I\}, \{I \rightarrow G, GH \rightarrow I\}), R_3(\{E, H, I\}, \{E \rightarrow I, IH \rightarrow E\})\}$ 保持函数依赖，且分解后的每个关系模式均为3NF。

(6) 可确定 R 的一个候选键 HJ ，由于 U_i 均不包含 HJ ，则 $\rho = \rho \cup \{R_4(\{H, J\})\}$ ，因此将 R 分解为 $\rho = \{R_1(\{I, J\}, \{J \rightarrow I\}), R_2(\{G, H, I\}, \{I \rightarrow G, GH \rightarrow I\}), R_3(\{E, H, I\}, \{E \rightarrow I, IH \rightarrow E\}), R_4(\{H, J\})\}$ 。

分解具有无损连接性， ρ 即为所求。



模式分解算法

算法：分解关系模式为满足BCNF的一个无损分解

- 输入：关系模式 $R (U, F)$
- 输出： R 的一个为BCNF的无损连接分解 ρ 。

模式分解算法

算法：分解关系模式为满足BCNF的一个无损分解

– 方法：

- 1) 令 $\rho = \{R\}$ 。
- 2) 检查 ρ 中每个关系模式是否均属于BCNF。若是，则算法终止。
- 3) 对 ρ 中每一个不满足BCNF的关系模式 S ，做如下操作
 - S 中必有不满足BCNF的非平凡的函数依赖 $X \rightarrow A$ ，其中 X 不是 S 的候选键。
 - 对 S 进行分解，分解为 $S_1(XA)$ 和 $S_2(U_S - A)$ ，其中 U_S 为 S 的属性集，用 S_1 和 S_2 取代 ρ 中的 S 。返回步骤(2)。

由于 U 中属性有限，因而有限次循环后算法一定终止。

模式分解算法

【例】有关系模式 $R\langle U, F \rangle$ ， $U = \{C, T, H, I, S, G\}$ ， $F = \{CS \rightarrow G, C \rightarrow T, TH \rightarrow I, HI \rightarrow C, HS \rightarrow I\}$ ，将R分解成具有无损连接的BCNF。

解答：

1) 令 $\rho = \{R\}$

2) 检查 ρ 中每个关系模式是否均属于BCNF；

H、S是L类属性，且 $HS \xrightarrow{F^+} U$ ，所以HS为 R 的一个候选键。

由此可判断 ρ 中关系模式R不属于BCNF。(存在传递函数依赖)

模式分解算法

【例】有关系模式 $R\langle U, F \rangle$ ， $U = \{C, T, H, I, S, G\}$ ， $F = \{CS \rightarrow G, C \rightarrow T, TH \rightarrow I, HI \rightarrow C, HS \rightarrow I\}$ ，将R分解成具有无损连接的BCNF。

解答：

3) 考虑 $CS \rightarrow G$ ，CS不是R的候选键，将R分解为

$\rho = \{R_1 (\{CSG\}, \{CS \rightarrow G\}),$

$R_2 (\{CTHIS\}, \{C \rightarrow T, TH \rightarrow I, HI \rightarrow C, HS \rightarrow I\})\}$;

R_1 是BCNF， R_2 不是(R_2 的候选键仍为HS)，需对 R_2 进一步分解。

模式分解算法

【例】有关系模式 $R\langle U, F \rangle$, $U=\{C, T, H, I, S, G\}$,
 $F=\{CS \rightarrow G, C \rightarrow T, TH \rightarrow I, HI \rightarrow C, HS \rightarrow I\}$, 将R分解成具有无损连接的BCNF。

解答:

4) 对于 $R_2 (\{CTHIS\}, \{C \rightarrow T, TH \rightarrow I, HI \rightarrow C, HS \rightarrow I\})$

考虑 $C \rightarrow T$, C不是 R_2 的候选键, 将 R_2 分解为

$\rho = \{R_{21} (\{CT\}, \{C \rightarrow T\}) ,$

$R_{22} (\{CHIS\}, \{CH \rightarrow I, HI \rightarrow C, HS \rightarrow I\}) \}$;

$(\because C \rightarrow T, TH \rightarrow I \therefore CH \rightarrow I)$

R_{21} 是BCNF, R_{22} 不是(R_{22} 的候选键仍为HS), 需对 R_{22} 进一步分解。

模式分解算法

【例】有关系模式 $R\langle U, F \rangle$ ， $U=\{C, T, H, I, S, G\}$ ， $F=\{CS \rightarrow G, C \rightarrow T, TH \rightarrow I, HI \rightarrow C, HS \rightarrow I\}$ ，将R分解成具有无损连接的BCNF。

解答：

对于 R_{22} ($\{CHIS\}$, $\{CH \rightarrow I, HI \rightarrow C, HS \rightarrow I\}$)

考虑 $CH \rightarrow I$ ，CH不是 R_{22} 的候选键，将分解为

$\rho=\{R_{221}(\{CHI\}, \{CH \rightarrow I, HI \rightarrow C\})$ ，

$R_{222}(\{CHS\}, \{HS \rightarrow C\})\}$;

R_{221} 和 R_{222} 均是BCNF，不需进一步分解。

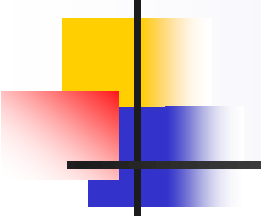
模式分解算法

【例】有关系模式 $R\langle U, F \rangle$, $U=\{C, T, H, I, S, G\}$,
 $F=\{CS \rightarrow G, C \rightarrow T, TH \rightarrow I, HI \rightarrow C, HS \rightarrow I\}$, 将R分解成具有无损连接的BCNF。

解答:

因此, 将R分解为 $\rho = \{R_1 (\{CSG\}, \{CS \rightarrow G\}) , R_{21} (\{CT\}, \{C \rightarrow T\}) , R_{221} (\{CHI\}, \{CH \rightarrow I, HI \rightarrow C\}) , R_{222} (\{CHS\}, \{HS \rightarrow C\}) \}$, 即为所求。

在关系模式 $R_{222} (\{CHS\}, \{HS \rightarrow C\})$ 中, 包含R的一个候选键HS, 利用算法5.3可验证分解 ρ 具有无损连接性。



模式分解算法

【例】 关系模式 $R(U, F)$ 中 $U=ABCDEFG$, $F=\{BG \rightarrow C, BD \rightarrow E, DG \rightarrow C, ADG \rightarrow BC, AG \rightarrow B, B \rightarrow D\}$, 完成下列问题:

- (1) 求 F 的最小函数依赖集;**
- (2) 求解 R 的候选键;**
- (3) 判断 R 最高满足第几范式;**
- (4) 若 R 不满足3NF, 将 R 分解为无损且保持函数依赖的3NF。**

模式分解算法

【例 (续 1)】 关系模式 $R(U, F)$ 中 $U=ABCDEFG$,
 $F=\{BG \rightarrow C, BD \rightarrow E, DG \rightarrow C, ADG \rightarrow BC, AG \rightarrow B, B \rightarrow D\}$ 。

(1) 求 F 的最小函数依赖集;

解答:

① 函数依赖右边属性单一化

将 $ADG \rightarrow BC$ 分解为 $ADG \rightarrow B$ 、 $ADG \rightarrow C$,

$\therefore AG \rightarrow B, DG \rightarrow C,$

$\therefore ADG \rightarrow B$ 、 $ADG \rightarrow C$ 为冗余依赖

$F=\{BG \rightarrow C, BD \rightarrow E, DG \rightarrow C, AG \rightarrow B, B \rightarrow D\}$ 。

模式分解算法

【例 (续 2)】 关系模式 $R(U, F)$ 中 $U=ABCDEFG$,
 $F=\{BG \rightarrow C, BD \rightarrow E, DG \rightarrow C, ADG \rightarrow BC, AG \rightarrow B, B \rightarrow D\}$ 。

(1) 求 F 的最小函数依赖集;

解答:

$F=\{BG \rightarrow C, BD \rightarrow E, DG \rightarrow C, AG \rightarrow B, B \rightarrow D\}$

② 去掉冗余的函数依赖

判断 $BG \rightarrow C$ 是否冗余:

令 $G = \{BD \rightarrow E, DG \rightarrow C, AG \rightarrow B, B \rightarrow D\}$

$BG_G^+ = BGDEC$; $C \in BG_G^+$, $BG \rightarrow C$ 冗余;

$F = \{BD \rightarrow E, DG \rightarrow C, AG \rightarrow B, B \rightarrow D\}$

同理, 判断其他函数依赖不冗余, F 不变。

模式分解算法

【例 (续 3)】 关系模式 $R(U, F)$ 中 $U=ABCDEFG$,
 $F=\{BG \rightarrow C, BD \rightarrow E, DG \rightarrow C, ADG \rightarrow BC, AG \rightarrow B, B \rightarrow D\}$ 。

解答: (1) 求 F 的最小函数依赖集;

$F = \{ BD \rightarrow E, DG \rightarrow C, AG \rightarrow B, B \rightarrow D \}$

③ 去掉各函数依赖左部冗余的属性。

本题需考虑 $BD \rightarrow E, DG \rightarrow C, AG \rightarrow B$ 。

对于 $BD \rightarrow E$, 在决定因素中去掉 B , $D \rightarrow E$, $D_F^+ = \{D\}$, 不包含 E , B 不冗余, F 不变。

在决定因素中去掉 D , $B \rightarrow E$, $B_F^+ = \{BDE\}$, 包含 E , D 多余。
所以用 $B \rightarrow E$ 代替 $BD \rightarrow E$ 。

同理, 对 $DG \rightarrow C, AG \rightarrow B$ 进行判断, 没有冗余属性。

$F_m = \{ B \rightarrow E, DG \rightarrow C, AG \rightarrow B, B \rightarrow D \}$

模式分解算法

【例 (续 4)】 关系模式 $R(U, F)$ 中 $U=ABCDEFG$,
 $F=\{BG \rightarrow C, BD \rightarrow E, DG \rightarrow C, ADG \rightarrow BC, AG \rightarrow B, B \rightarrow D\}$ 。

(2) 求解 R 的候选键;

(3) 判断 R 最高满足第几范式;

解答:

在 $F_m = \{ B \rightarrow E, DG \rightarrow C, AG \rightarrow B, B \rightarrow D \}$ 中

AG 为L类属性, 且 $AG \xrightarrow{F^+} U$, 所以 AG 为 R 的候选键。

R 最高属于2NF。

因为 $F_m = \{ B \rightarrow E, DG \rightarrow C, AG \rightarrow B, B \rightarrow D \}$ 中存在对 AG 的传递依赖 $B \rightarrow D$ 。

模式分解算法

【例 (续 5)】 关系模式 $R(U, F)$ 中 $U=ABCDEFG$,
 $F=\{BG \rightarrow C, BD \rightarrow E, DG \rightarrow C, ADG \rightarrow BC, AG \rightarrow B, B \rightarrow D\}$ 。

(4) 若 R 不满足 3NF, 将 R 分解为无损且保持函数依赖的 3NF。

解答:

$F_m = \{ B \rightarrow E, DG \rightarrow C, AG \rightarrow B, B \rightarrow D \}$

按算法, 对 F_m 按具有相同左部的原则分组, $U_1=BDE$ 、
 $U_2=DGC$ 、 $U_3=AGB$, 因此将 R 分解为

$\rho = \{ R_1 (\{B, D, E\}, \{B \rightarrow E, B \rightarrow D\}) ,$

$R_2 (\{C, D, G\}, \{DG \rightarrow C\}) ,$

$R_3 (\{A, B, G\}, \{AG \rightarrow B\}) \}$

因为候选键 $AG \subseteq U_3$, 则所求分解 ρ 具有无损连接性并
保持函数依赖, 且每个子模式为 3NF。



小结

- 在函数依赖和多值依赖的范畴内讨论了关系模式的规范化，给出了范式的概念。
- 讨论了将低一级的范式分解为若干高一级的范式的理论基础——Armstrong公理以及判断模式的候选键的方法。



小结

- 给出了一组模式与一个关系模式等价的含义，介绍了几种在不同含义下模式分解的算法，以及最终所能达到的“等价”效果，即达到第几范式，是否具有无损连接性，是否保持函数依赖。
 - 总是从一个关系模式出发实行分解。
 - 采用了两种关系运算——投影和自然连接。



小结

- 规范化理论为数据库设计提供了理论的指南和工具
 - **对关系模式进行规范化的主要目的是消除数据冗余和更新异常；**
 - **规范化是通过模式分解来实现的，分解后的关系模式具有较少的列，能描述的信息也相应的减少了，或者说语义更纯了。**



小结

- 并不是规范化程度越高，模式就越好，而必须结合应用环境和现实世界的具体情况合理地选择数据库模式。
 - **有时出于对数据库性能的考虑而允许数据库中**
存在数据冗余，来获得比较好的性能。
 - **比如：**
 - 1) 复制某些数据列到一些表中以便更容易地访问而不必进行多表的连接。
 - 2) 存储派生数据以加快数据处理过程。
 - 3) 撤消某些已分解的关系模式以减少多个表连接的开销。