

数据库系统

哈尔滨工业大学（深圳）

第15讲 关系模式设计之规范形式



关系模式设计之规范形式

如何避免数据库的一致性问题—数据库的规范性设计

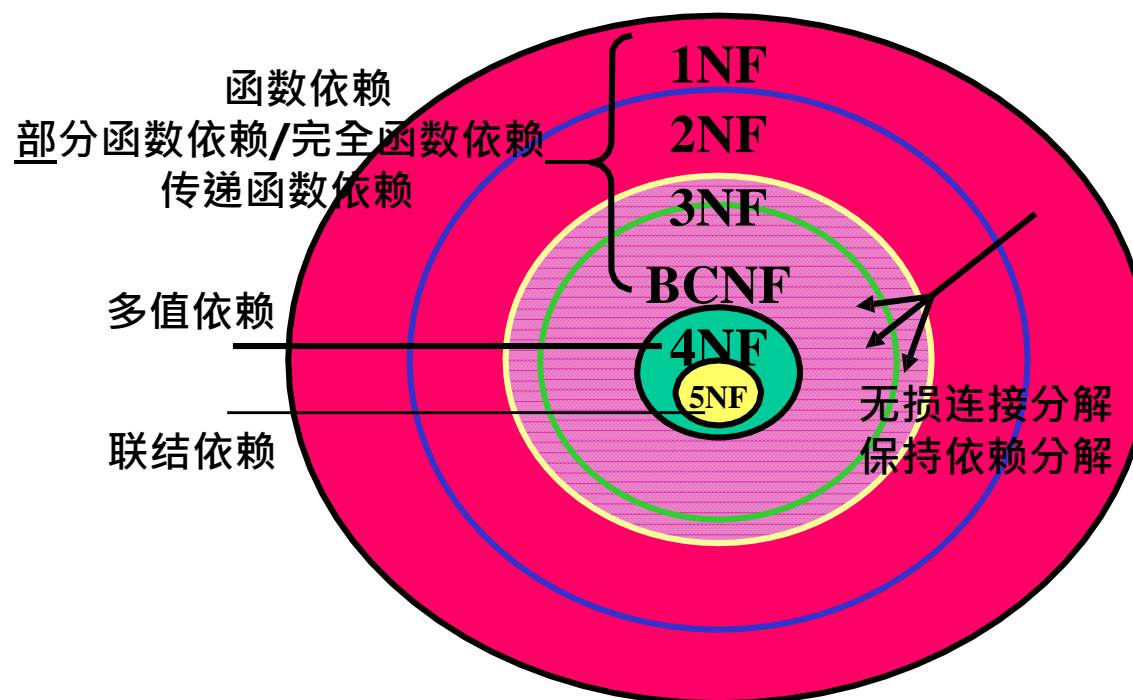
数据库的规范性设计需要分析数据库Table中的属性在取值方面有什么依存关系？数据库设计过程中应遵循什么样的原则

➤数据库设计理论

□数据依赖理论

□关系范式理论

□模式分解理论





关系模式设计之规范形式

基本内容

1. 关系的第1NF和第2NF
2. 关系的第3NF和Boyce-Codd NF
3. 多值依赖及其公理定理
4. 关系的第4NF

重点与难点

- 一组概念：1NF, 2NF, 3NF, BCNF, 4NF；多值依赖
- 熟练应用数据库设计的规范化形式，判断数据库设计的正确性及可能存在的问题



关系的第1范式和第2范式

[Definition] 1NF 若关系模式 $R(U)$ 中关系的每个分量都是不可分的数据项(值、原子), 则称 $R(U)$ 属于第一范式, 记为: $R(U) \in 1NF$ 。

示例: **Star(name, address(street, city))**

Star不属于1NF, 因为属性address仍包含了street和city两个属性, 其分量不是原子。

Students

sid	lname	fname	class	telephone
1	Jones	Allan	2	555-1234
2	Smith	John	3	555-4321
3	Brown	Harry	2	555-1122
5	White	Edward	3	555-3344

符合1NF

1NF要求关系中不能有复合属性、多值属性及其组合

Students

sid	name		class	telephone	enrollment	
	lname	fname			cno	major
1	Jones	Allan	2	555-1234	101	No
					108	Yes
2	Smith	John	3	555-4321	105	No
3	Brown	Harry	2	555-1122	101	Yes
					108	No
4	White	Edward	3	555-3344	102	No
					105	No

不符合1NF

多值属性

Value: structured value collection of values



关系的第1范式和第2范式

不符合1NF的处理

- ✓ 将 非1NF转换为 1NF情况

示例： **Star(name, address(street, city))**

→ **Star(name, address) 或者 Star (name, street, city)**

将复合属性处理为简单属性；将多值属性与关键字单独组成一新的关系

- ✓ 引入新的数据模型处理：**Object-Oriented Data Model**

列对象

Head: structured type

Students							
	name		class	telephone	enrollment		
	lname	fname			cno	major	
1	Jones	Allan	2	555-1234	101	No	
					108	Yes	
2	Smith	John	3	555-4321	105	No	
3	Borwn	Harry	2	555-1122	101	Yes	
					108	No	
4	White	Edward	3	555-3344	102	No	
					105	No	

行对象

结构对象
聚集对象

Value: structured value
collection of values



关系的第1范式和第2范式

[Definition] 2NF

若 $R(U) \in 1NF$ 且 U 中的每一非主属性完全函数依赖于候选键，则称 $R(U)$ 属于第二范式，记为： $R(U) \in 2NF$ 。

第二范式消除了非主属性对候选键的部分依赖。

示例： $R(S\#, SN, SD, CN, G)$

其中， $S\#$:学号, SN :姓名, SD :班级, CN :课程, G :成绩。

函数依赖： $S\# \rightarrow SN$, $S\# \rightarrow SD$, $\{S\#, CN\} \rightarrow G$

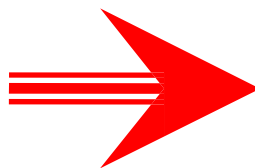
候选键： $\{S\#, CN\}$ ，非主属性： SN, SD, G 。

因为： $\{S\#, CN\} \xrightarrow{p} \{SN, SD\}$ ，所以 R 不属于2NF。

将其分解为 $R_1(S\#, SN, SD)$, $R_2(S\#, CN, G)$, 则 $R_1 \in 2NF$, $R_2 \in 2NF$ 。

学生

学号	姓名	课程号	课程名	成绩
98030101	张三	001	数据库	92
98030101	张三	002	计算机原理	85
98030101	张三	003	高等数学	88
98040202	李四	002	计算机原理	90
98040202	李四	003	高等数学	80
98040202	李四	001	数据库	55
98040203	王五	003	高等数学	56
98030102	周六	001	数据库	54
98030102	周六	002	计算机原理	85
98030102	周六	003	高等数学	48



学生

学号	姓名
98030101	张三
98040202	李四
98040203	王五
98030102	周六

课程

课程号	课程名
001	数据库
002	计算机原理
003	高等数学

选课

学号	课程号	成绩
98030101	001	92
98030101	002	85
98030101	003	88
98040202	002	90
98040202	003	80
98040202	001	55
98040203	003	56
98030102	001	54
98030102	002	85
98030102	003	48



关系的第1范式和第2范式

练习：下列模式是否满足第2范式？怎样使其满足第2范式？

- 学生(学号, 姓名, 班级, 课号, 课程名, 成绩, 教师, 教师职务)
主属性？非主属性？部分依赖？还是完全依赖
- 员工(员工码, 姓名, 出生日期, 联系电话, 最后学历, 毕业学校, 培训日期, 培训内容)
- 图书(书号, 书名, 出版日期, 出版社, 书架号, 房间号)



关系的第1范式和第2范式

练习：下列模式是否满足第2范式？怎样使其满足第2范式？

- 学生(学号, 姓名, 班级, 课号, 课程名, 成绩, 教师, 教师职务)
 - 候选键： $\{\text{学号}, \text{课号}\} \xrightarrow{f} U$ ；非主属性：姓名、课程名
 - 部分依赖： $\{\text{学号}, \text{课号}\} \xrightarrow{p} \text{课程名}$ ； $\{\text{学号}, \text{课号}\} \xrightarrow{p} \text{姓名}$
- 员工(员工码, 姓名, 出生日期, 联系电话, 最后学历, 毕业学校, 培训日期, 培训内容)
 - 候选键： $\{\text{员工码}, \text{培训日期}\} \xrightarrow{f} U$ ；非主属性：姓名、出生日期
 - 部分依赖： $\{\text{员工码}, \text{培训日期}\} \xrightarrow{p} \{\text{姓名}, \text{出生日期}\}$ ；
- 图书(书号, 书名, 出版日期, 出版社, 书架号, 房间号)
 - 候选键： $\text{书号} \xrightarrow{f} U$ ；非主属性：候选键外其他属性
 - 无部分依赖： $\text{书号} \xrightarrow{f} \text{每一个属性}$ ；



关系的第3范式和Boyce-Codd范式

[Definition] 3NF

若 $R(U, F) \in 2NF$ 且 R 中不存在这样的情况：候选键 X ，属性组 $Y \subseteq U$ 和非主属性 A ，且 $A \notin X, A \notin Y, Y \not\subseteq X, Y \not\rightarrow X$ ，使得 $X \rightarrow Y, Y \rightarrow A$ 成立。满足以上条件则称 $R(U)$ 属于第三范式，记为： $R(U) \in 3NF$ 。

示例：Store(Sid, Pid, Did, Mgr)

其中，Sid:商店, Pid:商品, Did:商品部, Mgr:经理。

函数依赖： $\{Sid, Pid\} \rightarrow Did, \{Sid, Did\} \rightarrow Mgr$

候选键： $\{Sid, Pid\}$ ，非主属性： Mgr 。

因为： $\{Sid, Pid\} \rightarrow Did, \{Sid, Did\} \rightarrow Mgr$ ，所以 R 不属于3NF。

将其分解为 $R_1(Sid, Pid, Did), R_2(Sid, Did, Mgr)$,

则 $R_1 \in 3NF, R_2 \in 3NF$ 。

➤第3范式消除了非主属性对侯选键的传递依赖

商店			
商店	商品(编号)	商品部	商品部经理
分店	鞋 01	鞋类部	张三
一分店	鞋 02	鞋类部	张三
一分店	鞋 03	鞋类部	张三
一分店	鞋 04	鞋类部	张三
一分店	鞋 05	鞋类部	张三
一分店	化妆品 01	化妆品部	李四
一分店	化妆品 02	化妆品部	李四
一分店	化妆品 03	化妆品部	李四
一分店	化妆品 04	化妆品部	李四
一分店	化妆品 05	化妆品部	李四
一分店	化妆品 06	化妆品部	李四
二分店	鞋 01	综合部	王三
二分店	鞋 02	综合部	王三
二分店	鞋 03	综合部	王三
二分店	化妆品 01	妇儿部	王四
二分店	化妆品 02	妇儿部	王四
二分店	化妆品 03	妇儿部	王四



关系的第3范式和Boyce-Codd范式

练习：下列模式是否满足第3范式？怎样使其满足第3范式？

●学生(学号，系号，系主任)

●员工(员工码，姓名，部门，部门经理)



关系的第3范式和Boyce-Codd范式

练习：下列模式是否满足第3范式？怎样使其满足第3范式？

●学生(学号，系号，系主任)

□候选键：学号 \xrightarrow{f} U；非主属性：系主任

□传递依赖：学号 \rightarrow 系号，系号 \rightarrow 系主任

□无部分依赖

所以：满足第2NF但不满足第3NF.

●员工(员工码，姓名，部门，部门经理)

□候选键：员工码 \xrightarrow{f} U；非主属性：部门经理

□传递依赖：员工码 \rightarrow 部门，部门 \rightarrow 部门经理

□无部分依赖

所以：满足第2NF但不满足第3NF.

➤关系模式设计如满足第3范式，则一定能满足第2范式；反之则不然。



关系的第3范式和Boyce-Codd范式

关系模式分解成3NF

示例： **$R(A, B, C, D, E, F, G)$**

函数依赖集合{ **$A \rightarrow B, A \rightarrow C, C \rightarrow D, C \rightarrow E, E \rightarrow FG$** }

候选键：**A**；有传递依赖，**R**不满足3NF。

分解规则：

将每一个函数依赖单独组成一个关系

$\rho = \{ R1(A, B), R2(A, C), R3(C, D), R4(C, E), R5(E, F, G) \}$

可以看出：每一个模式都属于3NF

也可以合并一些关系：

$\rho = \{ R12(A, B, C), R34(C, D, E), R5(E, F, G) \}$



关系的第3范式和Boyce-Codd范式

[Definition] BCNF

若 $R(U, F) \in 1NF$, 若对于任何 $X \rightarrow Y \in F$ (或 $X \rightarrow A \in F$), 当 $Y \not\subseteq X$ (或 $A \notin X$)时, X 必含有候选键, 则称 $R(U)$ 属于Boyce-Codd范式, 记为: $R(U) \in BCNF$ 。

示例: 邮编(城市, 街道, 邮政编码)

函数依赖: { 城市, 街道 } \rightarrow 邮政编码; 邮政编码 \rightarrow 城市.

候选键: { 城市, 街道 } $\xrightarrow{f} U$

因不含候选键: 邮政编码 \rightarrow 城市; 所以不满足BCNF

因无传递依赖, 所以满足第3范式;

示例: 选课(学号, 课程号, 教师编号)

假设规定每位教师只开一门课, 则有: { 学号, 课程号 } \rightarrow 教师编号; 教师编号 \rightarrow 课程号. 显然: 该模式满足第3范式但不满足Boyce-Codd范式。



关系的第3范式和Boyce-Codd范式

[定理] 若 $R(U, F) \in \text{BCNF}$, 则 $R(U, F) \in 3\text{NF}$ 。

证明：用反证法证明，设 $R(U, F) \in \text{BCNF}$ ，但 $R(U, F) \notin 3\text{NF}$ ，依据 3NF 定义，则必有一传递依赖存在：

设该传递依赖为 $X \rightarrow Y$ ， $Y \rightarrow A$ ，其中 X 候选键， $A \notin X$ ， $A \notin Y$ ， $Y \not\rightarrow X$ ，显然 $X \not\subset Y$ ，

因 $A \notin Y$ ，则 $Y \rightarrow A$ 将违反 BCNF 的定义 (任一函数依赖都包含候选键，而 Y 不是候选键)。故定理得证。证毕。

➤ 有传递依赖的或者说不满足 3NF 的，一定不满足 BCNF



关系的第3范式和Boyce-Codd范式

关系模式分解成BCNF

示例： **$R(A, B, C, D, E, F, G)$**

函数依赖集合{ **$A \rightarrow B, A \rightarrow C, C \rightarrow D, C \rightarrow E, E \rightarrow FG$** }

候选键：**A**；有不依赖于候选键的其他函数依赖，**R**不满足BCNF。

分解规则：

将左侧不含候选键的函数依赖单独组成一个关系，将包含候选键的组成一关系

$\rho = \{ R1(C, D), R2(C, E), R3(E, F, G), R4(A, B, C) \}$

可以看出： **$R1 \in BCNF; R2 \in BCNF; R3 \in BCNF; R4 \in BCNF;$**

也可以将**R1**和**R2**合并：

$\rho = \{ R12(C, D, E), R3(E, F, G), R4(A, B, C) \}$



多值依赖及其公理定理

[Definition] 多值依赖

对 $R(U)$, 设 $X, Y \subseteq U$, 若对于 $R(U)$ 的任一关系 r , 若元组 $t \in r, s \in r$, $t[X] = s[X]$, 则必有 $u \in r, v \in r$ 使得 :

(1) $u[X] = v[X] = t[X] = s[X]$

(2) $u[Y] = t[Y]$ 且 $u[U-X-Y] = s[U-X-Y]$

(3) $v[Y] = s[Y]$ 且 $v[U-X-Y] = t[U-X-Y]$

均成立, 则称 Y 多值依赖于 X , 或说 X 多值决定 Y , 记作 $X \twoheadrightarrow Y$ 。

	X	Y	Z=U-X-Y	
1	t[X]	t[Y]	t[Z]	t
2	t[X]	s[Y]	s[Z]	s
3	t[X]	t[Y]	s[Z]	u
4	t[X]	s[Y]	t[Z]	v



多值依赖及其公理定理

多值依赖的特性

- 1) 直观地，对于X给定值，Y有一组值与之对应(0或n个)且这组Y值不以任何方式与U-X-Y中属性值相联系，有 $X \twoheadrightarrow Y$ 。
- 2) 若交换t, s 的Y值而得到的新元组仍在r中，则 $X \twoheadrightarrow Y$ 。
- 3) X, Y 可相交，u,v可以与t,s相同。
- 4) 函数依赖是多值依赖的特例。
- 5) 令 $Z=U-X-Y$, 有 $X \twoheadrightarrow Z$, 若 $Z=\phi$, 则必有 $X \twoheadrightarrow Y$ 。

示例： $R = \{ \text{课程名}C, \text{教师名}T, \text{上课时间}H, \text{教室}R, \text{学生名}S, \text{成绩}G \}$ ，则有：

□ $C \twoheadrightarrow HR, T \twoheadrightarrow HR$ ，但不存在 $C \twoheadrightarrow H$ 及 $C \twoheadrightarrow R$ 。

说明：同一门课程或同一教师对同一批学生可以在不同时间不同地点上课。



关系的第4范式和弱第4范式

[Definition] 4NF

设 $R(U) \in 1NF$, D 是其上的一组依赖 (函数依赖, 多值依赖), 对任意 $X \twoheadrightarrow Y \in D$, 若 $Y \neq \phi$, $Y \not\subseteq X$, $XY \neq U$, X 为超键, 则称 $R(U)$ 满足第四范式, 记为: $R(U) \in 4NF$ 。

第四范式消除了非主属性对候选键以外属性的多值依赖。

如果有多值依赖, 则一定依赖于候选键



关系的第4范式和弱第4范式

[定理] 若 $R \in 4NF$, 则必有 $R \in BCNF$ 。

证明： 设 $R \in 4NF$, 对 R 上的任何 $X \rightarrow Y$, $Y - X \neq \phi$,

(1) 当 $XY = U$ 时, $X \rightarrow U$, X 必为超键。

(2) 当 $XY \neq U$ 时, 因 $X \rightarrow Y$, 有 $X \rightarrow \rightarrow Y$, 由第四范式定义 X 必为超键, 再由 $BCNF$ 定义知 $R \in BCNF$ 。



多值依赖及其公理定理

[Armstrong's Axioms A4~A8]关于多值依赖的公理

设 $R(U)$, $X, Y \subseteq U$, 对于 $R(U)$ 的任一关系 r , 有以下规则：

□[A4]多值依赖互补律(Complementation)或对称性：若 $X \twoheadrightarrow Y$, 则 $X \twoheadrightarrow U-X-Y$;

□[A5]多值依赖增广律(Augmentation)：若 $X \twoheadrightarrow Y$ 且 $V \subseteq W$, 则 $WX \twoheadrightarrow VY$;

注意：此条与A2规则是相似的： $X \rightarrow Y$ 且 $V \subseteq W$, 则 $WX \rightarrow VY$;

□[A6]多值依赖传递律(Transitivity)：若 $X \twoheadrightarrow Y$, $Y \twoheadrightarrow Z$, 则 $X \twoheadrightarrow Z-Y$;

注意：此条比A3规则限制要强： $X \rightarrow Y$, $Y \rightarrow Z$, 则 $X \rightarrow Z$ 。多值依赖不存在这种规则，即： $X \twoheadrightarrow Y$, $Y \twoheadrightarrow Z$, 则 $X \twoheadrightarrow Z$ 不一定成立，例如 $C \twoheadrightarrow HR$, $HR \twoheadrightarrow H$ 但是 C 不能多值决定 H 。

□[A7]若 $X \rightarrow Y$, 则 $X \twoheadrightarrow Y$;

□[A8]若 $X \twoheadrightarrow Y$, $Z \subseteq Y$ 且对于某个与 Y 不相交的 W 有 $W \rightarrow Z$, $W \cap Y = \emptyset$, 则有 $X \rightarrow Z$ 。



多值依赖及其公理定理

[定理]Armstrong Axioms系统的规则A1-A8是有效的

A6的证明：用反证法进行。

设一关系 r , 假设 $X \twoheadrightarrow Y$, $Y \twoheadrightarrow Z$, 而 $X \not\rightarrow Z-Y$ 不成立, 按多值依赖定义, 即

对任一关系 r , 有元组 $t \in r$, $s \in r$, 但满足下述条件的 u 不存在($u \notin r$):

$u[X]=t[X]=s[X]$, $u[Z-Y]=t[Z-Y]$ 且 $u[U-X-(Z-Y)]=s[U-X-(Z-Y)]$

下面由 $X \twoheadrightarrow Y$, $Y \twoheadrightarrow Z$ 检验上述 u 是否真的不存在:

由 $X \twoheadrightarrow Y$, 对 $t \in r$, $s \in r$, 有 $v \in r$, 满足:

(1) $v[X]=t[X]=s[X]$

(2) $v[Y]=s[Y]$

(3) $v[U-X-Y]=t[U-X-Y]$

由 $Y \twoheadrightarrow Z$, 对 $v \in r$, $s \in r$, 有 $w \in r$, 满足:

(4) $w[Y]=v[Y]=s[Y]$

(5) $w[Z]=v[Z]$

(6) $w[U-Z-Y]=s[U-Z-Y]$



多值依赖及其公理定理

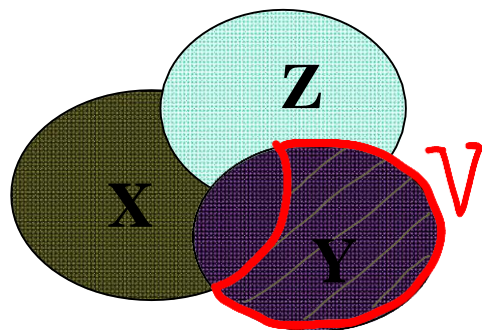
[定理]Armstrong Axioms系统的规则A1-A8是有效的

由 $X \twoheadrightarrow Y$, 对 $t \in r, s \in r$, 有 $v \in r$, 满足:

- (1) $v[X] = t[X] = s[X]$
- (2) $v[Y] = s[Y]$
- (3) $v[U - X - Y] = t[U - X - Y]$

由 $Y \twoheadrightarrow Z$, 对 $v \in r, s \in r$, 有 $w \in r$, 满足:

- (4) $w[Y] = v[Y] = s[Y]$
- (5) $w[Z] = v[Z]$
- (6) $w[U - Z - Y] = s[U - Z - Y]$



A6的证明：用反证法进行(Cont.)

由(1)~(6)可确定如下结论：

$w[X] = t[X]$ (因为 $X = \{Z \cap X\} \cup \{X - Z\}$, 对 $Z \cap X$, 可由(5)(1)得到结果；对 $X - Z$, 可由(4)(6)(1)得到结果)

$w[Z - Y] = t[Z - Y]$ (由(5)(1)(3)可得到结果)

$w[U - X - (Z - Y)] = s[U - X - (Z - Y)]$ (设 $V = U - X - (Z - Y)$, $V = \{V \cap Z\} \cup \{V - Z\}$, 对 $V - Z$, 可由(4)(6)得到结果；对 $V \cap Z$, 由于 $V \cap Z = (Y \cap Z) - X$, 可由(2)(5)得到结果) 所以 w 就是 u , 也就是说 u 是存在的，这与假设 u 不存在相矛盾，所以A6规则是正确的。



多值依赖及其公理定理

[定理]Armstrong Axioms系统的规则A1-A8是有效的

[A8]若 $X \twoheadrightarrow Y$, $Z \subseteq Y$ 且对于某个与 Y 不相交的 W 有 $W \rightarrow Z$, $W \cap Y = \emptyset$, 则有 $X \rightarrow Z$ 。

A8的证明：用反证法。设一关系 r 满足 $X \twoheadrightarrow Y$, $W \rightarrow Z$, $Z \subseteq Y$, $W \cap Y = \emptyset$, 但 $X \rightarrow Z$ 不成立，按函数依赖定义，即对关系 r , 有元组 $t \in r$, $s \in r$, 满足 $t[X] = s[X]$ 但 $t[Z] \neq s[Z]$ 。

下面检验上述当 $t[X] = s[X]$ 时是否有 $t[Z] \neq s[Z]$ ：

由 $X \twoheadrightarrow Y$, 对 $t \in r$, $s \in r$, 有 $u \in r$, 满足：

(1) $u[X] = t[X] = s[X]$

(2) $u[Y] = t[Y]$

(3) $u[U-X-Y] = s[U-X-Y]$ 因为 $W \cap Y = \emptyset$

· 由(1)(3)知 $u[W] = s[W]$, 又因 $W \rightarrow Z$

· 所以可推出 $s[Z] = u[Z]$, 又因 $Z \subseteq Y$,

由(2)可知 $u[Z] = t[Z]$

所以有 $s[Z] = t[Z]$, 与假设 $s[Z] \neq t[Z]$ 相矛盾，所以A8规则是正确的。



多值依赖及其公理定理

[引理7]：由**Armstrong's Axioms**可推出如下结论。

□(a)多值依赖合并律(**Union Rule**)：若 $X \twoheadrightarrow Y$ 且 $X \twoheadrightarrow Z$, 则 $X \twoheadrightarrow YZ$ 。

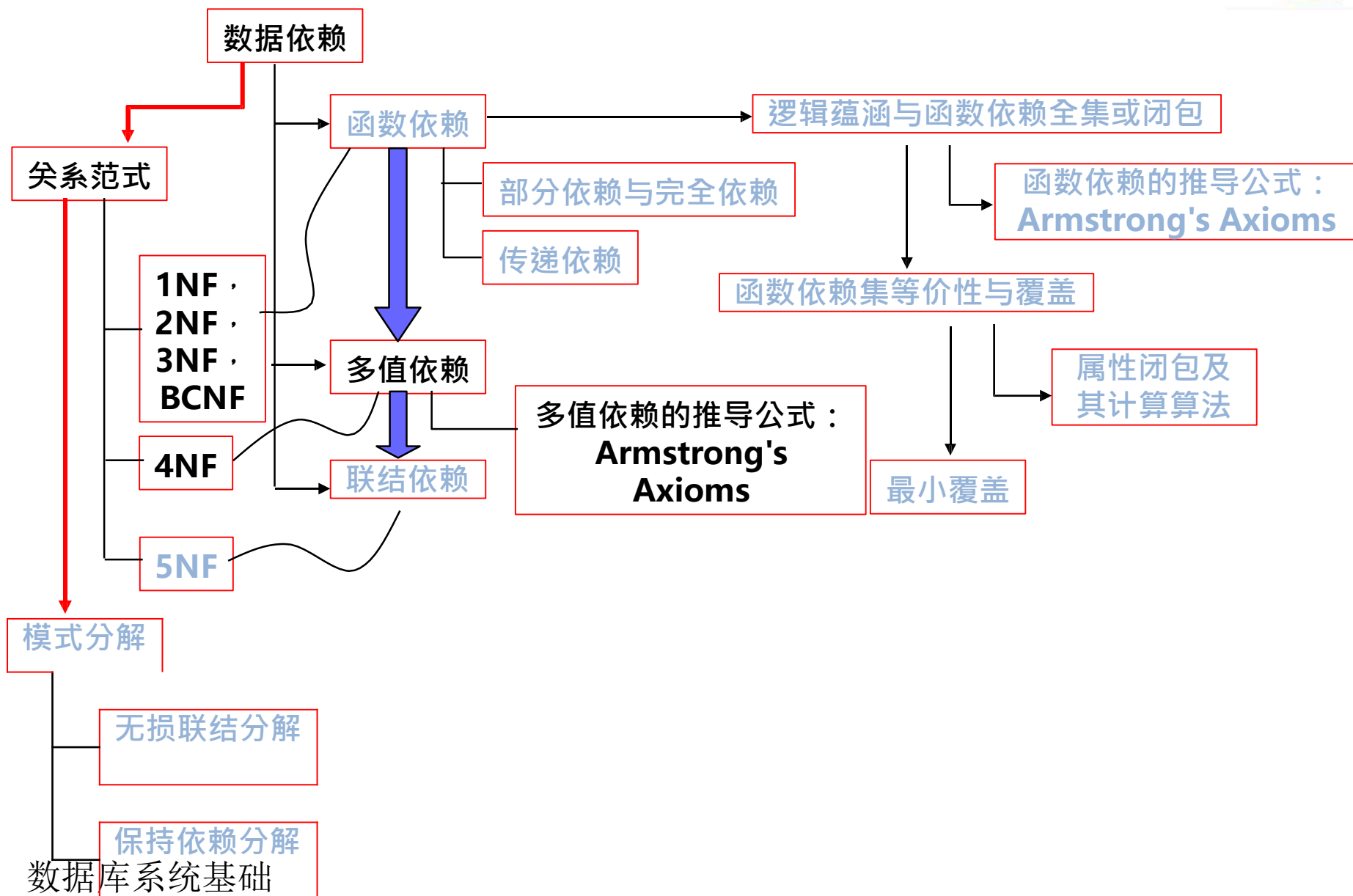
□(b)多值依赖伪传递律(**Pseudo Transitivity**)：若 $X \twoheadrightarrow Y$ 且 $WY \twoheadrightarrow Z$, 则 $X \twoheadrightarrow Z-WY$ 。

□(c)混合伪传递律：若 $X \twoheadrightarrow Y$, $XY \rightarrow Z$, 则 $X \rightarrow Z-Y$

□(d)多值依赖分解律(**Decomposition Rule**)：若 $X \twoheadrightarrow Y$ ， $X \twoheadrightarrow Z$ 则 $X \twoheadrightarrow Y-Z$, $X \twoheadrightarrow Z-Y$, $X \twoheadrightarrow Y \cap Z$ 。

➤证明：(略)

总结



第16章 模式分解存在什么问题



关系模式设计之规范形式

如何避免数据库的一致性问题—数据库的规范性设计

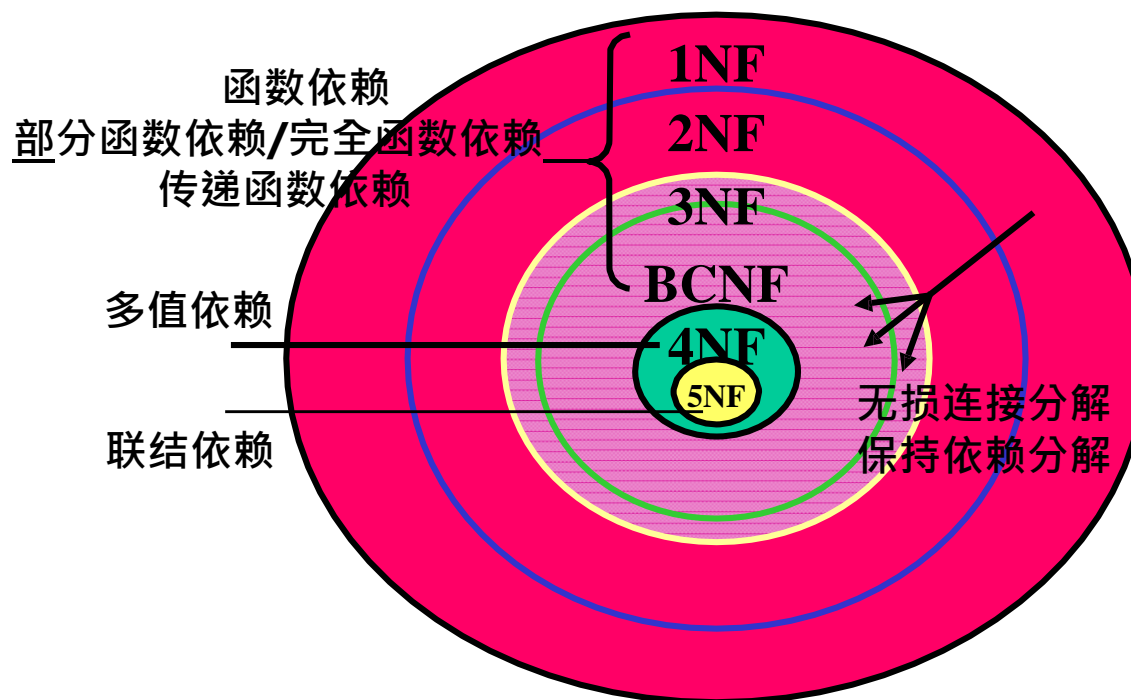
数据库的规范性设计需要分析数据库Table中的属性在取值方面有什么依存 关系？数据库设计过程中应遵循什么样的原则

➤数据库设计理论

□数据依赖理论

□关系范式理论

□模式分解理论





模式分解存在什么问题

1. 模式分解存在什么问题
2. 无损连接分解及其检验算法
3. 保持依赖分解及其检验算法
4. 关系模式无损连接或保持依赖的分解算法
5. 数据库设计需要知道的?

重点与难点

理解模式分解存在的问题是什么，怎样解决

两个概念：无损连接分解和保持依赖分解

关系模式如何进行无损连接或保持依赖地分解

五个算法：2个检验算法，3个分解算法(无损连接分解，保持依赖分解，既无损连接又保持依赖分解)



1. 模式分解存在什么问题

[Definition] 模式分解

关系模式R(U)的分解是指用R的一组子集 $\rho = \{R_1(U_1), \dots, R_k(U_k)\}$ 来代替它。

其中 $U = U_1 \cup U_2 \cup \dots \cup U_k$; $U_i \not\subseteq U_j$ ($i \neq j$)。

注：为便于后面叙述，我们用 R_i 代替 $R_i(U_i)$, R代替R(U)。

对于关系模式R的任一关系r, 它向 ρ 的投影连接记为 $m_\rho(r)$ ：

$$m_\rho(r) = \pi_{R_1}(r) \bowtie \dots \bowtie \pi_{R_k}(r) = \bowtie_{(i=1, \dots, k)} \pi_{R_i}(r)$$

这里： $\pi_{R_i}(r) = \{t[R_i] \mid t \in r, i=1, \dots, k\}$

模式分解需要关注：

R与 ρ 在数据内容方面是否等价：分解的无损连接性；

R与 ρ 在数据依赖方面是否等价：分解的保持依赖性。



1. 模式分解存在什么问题

[引理1] 设 R 为一关系模式， $\rho = \{R_1, \dots, R_k\}$ 是 R 的一个分解， r 是 R 的任一个关系， $r_i = \pi_{R_i}(r)$ ，则有规则成立：

(rule 1) $r \subseteq m_\rho(r)$

(rule 2) 若 $s = m_\rho(r)$ ，则 $\pi_{R_i}(s) = r_i$ (即： $\pi_R(m_\rho(r)) = \pi_R(r)$)

(rule 3) $m_\rho(m_\rho(r)) = m_\rho(r)$

r	A	B	C	D
	1	2	A	D
	2	2	A	E
	2	3	B	C
	1	3	B	D

r1	A	B	C
	1	2	A
	2	2	A
	2	3	B
	1	3	B

r2	B	C	D
	2	A	D
	2	A	E
	3	B	C
	3	B	D

$m_p(r)$	A	B	C	D
1	2	A	D	
1	2	A	E	
2	2	A	D	
2	2	A	E	
1	3	B	C	
1	3	B	D	
2	3	B	C	
2	3	B	D	



1. 模式分解存在什么问题

(1)关于模式分解的一些特性：数据约束的等价性

[示例] $R(C, S, Z)$, C 是城市, S 是街区, Z 是邮政编码

$F = \{ CS \rightarrow Z, Z \rightarrow C \}$

$\rho = \{ R_1(SZ), R_2(CZ) \}$

分解后是否有
约束丢失了?



1. 模式分解存在什么问题

(2)模式分解要注意什么

当模式不符合关系范式时，进行模式分解

示例： $R(A, B, C, D, E, F, G)$

函数依赖集合{ $A \rightarrow B, A \rightarrow C, C \rightarrow D, C \rightarrow E, E \rightarrow FG$ }

候选键：A;有传递依赖，R不满足3NF。

分解规则：

将每一个函数依赖单独组成一个关系

$\rho = \{ R1(A, B), R2(A, C), R3(C, D), R4(C, E), R5(E, F, G) \}$

可以看出：每一个模式都属于3NF

也可以合并一些关系：

$\rho = \{ R12(A, B, C), R34(C, D, E), R5(E, F, G) \}$

分解后的关系的
连接与分解前关
系的等价性？

分解前的约束，
在分解后是否还
存在？



2. 无损连接分解及其检验算法

(1) 无损连接分解的概念

[Definition] 无损连接分解

对于关系模式 $R(U, F)$, U 是属性全集, F 是函数依赖集合, $\rho = \{R_1, \dots, R_k\}$ 是 R 的一个分解, 如果对于 R 的任何满足函数依赖集 F 的关系 r , 有

$$r = m_{\rho}(r),$$

则称 ρ 是 R 相对于 F 的一个无损连接分解, 其中:

$$m_{\rho}(r) = \pi_{R_1}(r) \bowtie \dots \bowtie \pi_{R_k}(r) = \bigbowtie_{(i=1, \dots, k)} \pi_{R_i}(r)$$



无损连接分解及其检验算法(2)无损连接分解的检验算法

➤[Algorithm]无损连接性检验算法

Input : 关系模式 $R=A_1A_2...A_n$, 函数依赖集 F , 分解 $\rho=\{R_1,...,R_k\}$

Output : ρ 是否是无损连接的判断

Method : (1)构造一 k 行 n 列的表, 可称为 R_ρ 表。其中第 j 列对应于 A_j , 第 i 行对应于 R_i , 若 $A_j \in R_i$, 则 R_ρ 表中第 i 行第 j 列位置填写符号 a_j , 否则填写 b_{ij} 。

	A_1	...	A_i	...	A_n
R_1					
...					
R_i			a_i 或 b_{ij}		
...					
R_k					

	A	B	C	D	E
R_1	a_1	b_{12}	b_{13}	a_4	b_{15}
R_2	a_1	a_2	b_{23}	b_{24}	b_{25}
R_3	b_{31}	a_2	b_{33}	b_{34}	a_5
R_4	b_{41}	b_{42}	a_3	a_4	a_5
R_5	a_1	b_{52}	b_{53}	b_{54}	a_5

无损连接分解及其检验算法(2)无损连接分解的检验算法



(2)根据 $\forall (X \rightarrow Y) \in F$, 对 R_ρ 表进行修改 :

给定 $X \rightarrow Y$, 寻找X属性取值相同的行, 用其值重置Y属性值 (a或b) ;

(3)修改后, 如果有一行变成 a_1, a_2, \dots, a_n (全a), 则 ρ 是无损连接分解, 否则为有损连接分解。

	A_1	...	A_i	...	A_n
R_1					
...					
R_i			a_i 或 b_i		
...					
R_k					

	A	B	C	D	E
R_1	a_1	b_{12}	b_{13}	a_4	b_{15}
R_2	a_1	a_2	b_{23}	b_{24}	b_{25}
R_3	b_{31}	a_2	b_{33}	b_{34}	a_5
R_4	b_{41}	b_{42}	a_3	a_4	a_5
R_5	a_1	b_{52}	b_{53}	b_{54}	a_5

为什么可以
用 $X \rightarrow Y$
修改呢?

无损连接分解及其检验算法(3)无损连接分解检验算法的应用示例



示例：已知 $R = \{ ABCDE \}$

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

$\rho = \{ R_1(AD), R_2(AB), R_3(BE), R_4(CDE), R_5(AE) \}$

问： ρ 是否具有无损连接性

解：(1)构造 R_ρ 表。

	A	B	C	D	E
R_1	a_1	b_{12}	b_{13}	a_4	b_{15}
R_2	a_1	a_2	b_{23}	b_{24}	b_{25}
R_3	b_{31}	a_2	b_{33}	b_{34}	a_5
R_4	b_{41}	b_{42}	a_3	a_4	a_5
R_5	a_1	b_{52}	b_{53}	b_{54}	a_5

无损连接分解及其检验算法(3)无损连接分解检验算法的应用示例



(2)用每一个函数依赖修改 R_p 表

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

	A	B	C	D	E
R ₁	a ₁	b ₁₂	b ₁₃	a ₄	b ₁₅
R ₂	a ₁	a ₂	b ₂₃	b ₂₄	b ₂₅
R ₃	b ₃₁	a ₂	b ₃₃	b ₃₄	a ₅
R ₄	b ₄₁	b ₄₂	a ₃	a ₄	a ₅
R ₅	a ₁	b ₅₂	b ₅₃	b ₅₄	a ₅



用 $A \rightarrow C$
修改

	A	B	C	D	E
R ₁	a ₁	b ₁₂	b ₁₃	a ₄	b ₁₅
R ₂	a ₁	a ₂	b ₁₃	b ₂₄	b ₂₅
R ₃	b ₃₁	a ₂	b ₃₃	b ₃₄	a ₅
R ₄	b ₄₁	b ₄₂	a ₃	a ₄	a ₅
R ₅	a ₁	b ₅₂	b ₁₃	b ₅₄	a ₅



用 $B \rightarrow C$
修改

	A	B	C	D	E
R ₁	a ₁	b ₁₂	b ₁₃	a ₄	b ₁₅
R ₂	a ₁	a ₂	b ₁₃	b ₂₄	b ₂₅
R ₃	b ₃₁	a ₂	b ₁₃	b ₃₄	a ₅
R ₄	b ₄₁	b ₄₂	a ₃	a ₄	a ₅
R ₅	a ₁	b ₅₂	b ₁₃	b ₅₄	a ₅



无损连接分解及其检验算法(3)无损连接分解检验算法的应用示例

(2)用每一个函数依赖修改 R_p 表

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

	A	B	C	D	E
R ₁	a ₁	b ₁₂	b ₁₃	a ₄	b ₁₅
R ₂	a ₁	a ₂	b ₁₃	b ₂₄	b ₂₅
R ₃	b ₃₁	a ₂	b ₁₃	b ₃₄	a ₅
R ₄	b ₄₁	b ₄₂	a ₃	a ₄	a ₅
R ₅	a ₁	b ₅₂	b ₁₃	b ₅₄	a ₅



用 $CE \rightarrow A$
修改

	A	B	C	D	E
R ₁	a ₁	b ₁₂	b ₁₃	a ₄	b ₁₅
R ₂	a ₁	a ₂	b ₁₃	b ₂₄	b ₂₅
R ₃	a ₁	a ₂	b ₁₃	b ₃₄	a ₅
R ₄	b ₄₁	b ₄₂	a ₃	a ₄	a ₅
R ₅	a ₁	b ₅₂	b ₁₃	b ₅₄	a ₅



用
 $C \rightarrow D, DE \rightarrow C$
修改

	A	B	C	D	E
R ₁	a ₁	b ₁₂	a ₃	a ₄	b ₁₅
R ₂	a ₁	a ₂	a ₃	a ₄	b ₂₅
R ₃	a ₁	a ₂	a ₃	a ₄	a ₅
R ₄	b ₄₁	b ₄₂	a ₃	a ₄	a ₅
R ₅	a ₁	b ₅₂	a ₃	a ₄	a ₅

(3)检查是否有一行变成 a_1, a_2, \dots, a_n (全a)

有：无损连接；无：有损连接



无损连接分解及其检验算法

(4)分解成两个关系模式的无损连接检验算法

[定理]设 F 是关系模式 R 上的一个函数依赖集合。 $\rho=\{R_1, R_2\}$ 是 R 的一个分解，则：当且仅当 $R_1 \cap R_2 \rightarrow R_1 - R_2$ 或者 $R_1 \cap R_2 \rightarrow R_2 - R_1$

属于 F^+ 时， ρ 是关于 F 无损连接的。

	$R_1 \cap R_2$	$R_1 - R_2$	$R_2 - R_1$
R_1	a_1	a_2	b_{13}
R_2	a_1	b_{22}	a_3

前述算法是对多个关系，这个定理是针对两个关系的



3. 保持依赖分解及其检验算法 (1)保持依赖分解的概念

[Definition] 保持依赖分解

对于关系模式 $R(U, F)$, U 是属性全集, F 是函数依赖集合, $\rho = \{R_1, \dots, R_k\}$ 是 R 的一个分解, 如在 $\pi_{R_i}(F)$ 中的所有依赖之并集 ($i=1, \dots, k$), 逻辑蕴涵 F 的每个依赖, 则称分解 ρ 保持依赖集 F 。

其中 $\pi_{R_i}(F)$ 是 F 在 R_i 上的投影, 即 F 中的任一投影 $X \rightarrow Y$, 如果 X, Y 均包含于 R_i , 则 $X \rightarrow Y \in \pi_{R_i}(F)$ 。

注: (1)保持依赖的分解可能不是无损连接的。

(2)无损连接的分解可能不是保持依赖的。

示例: $R(CSZ), F = \{CS \rightarrow Z, Z \rightarrow C\}$, C 是城市, S 是街区, Z 是邮政编码, $\rho = \{R_1(SZ), R_2(CZ)\}$ 为一无损连接分解, 但却不保持依赖;

示例: $R(ABCD), F = \{A \rightarrow B, C \rightarrow D\}$, $\rho = \{R_1(AB), R_2(CD)\}$ 为一保持依赖分解, 但不是无损连接分解。



[Algorithm]保持依赖性检验算法

Input : 关系模式 $R=A_1A_2...A_n$, R 上的函数依赖集 F , 分解 $\rho=\{R_1,...,R_k\}$

Output : ρ 是否是保持依赖的判断

Method : 令 $G = \cup_{(i=1 \text{ to } k)} \pi_{R_i}(F)$, 只需检查 G 是否覆盖 F 即可。具体算法如下：

□ 首先对每个 $X \rightarrow Y \in F$ 计算 G 中的 X^+_G : (如果 X 不包含于 R_i 则不需计算了)

$Z = X$

WHILE Z 变化 DO

FOR $i = 1$ to k DO

$Z = Z \cup ((Z \cap R_i)^+ \cap R_i)$

□ 判断 G 是否逻辑蕴涵 $X \rightarrow Y$: Z 包含 Y , 则 G 逻辑蕴涵 $X \rightarrow Y$

□ 判断 ρ 是否保持依赖 : 如果 G 逻辑蕴涵 F , 则说 ρ 是保持依赖的分解。



应用示例

Input : $R(A, B, C, D, E)$

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

$\rho = \{ R_1(AC), R_2(BC), R_3(CDE) \}$

Output : ρ 是否是保持依赖的判断

Method : 依据题意

$\pi_{R_1}(F) = \{ A \rightarrow C \}, \pi_{R_2}(F) = \{ B \rightarrow C \}, \pi_{R_3}(F) = \{ C \rightarrow D, DE \rightarrow C \}$

$G = \{ A \rightarrow C, B \rightarrow C, C \rightarrow D, DE \rightarrow C \}$ ，显然不保持依赖。

□对函数依赖 $A \rightarrow C \in F$ 计算 G 中的 X_G^+ ：

$Z = \{A\} \cup \{C\} \cup \{\} \cup \{\} = \{A, C\}$ ， C 包含于 Z 中，所以 $A \rightarrow C$ 被 G 逻辑蕴涵

□对函数依赖 $DE \rightarrow C \in F$ 计算 G 中的 X_G^+ ：

$Z = \{D, E\} \cup \{\} \cup \{\} \cup \{C, D\} = \{C, D, E\}$ ， C 包含于 Z 中，所以 $DE \rightarrow C$ 被 G 逻辑蕴涵

□对函数依赖 $CE \rightarrow A \in F$ 计算 G 中的 X_G^+ ：

$Z = \{C, E\} \cup \{\} \cup \{\} \cup \{D\} = \{C, E, D\}$ ， A 不包含于 Z 中，所以不被 G 逻辑蕴涵

$$Z = Z \cup ((Z \cap R_i)^+_G \cap R_i)$$

左端属性必须是某一个关系模式的属性

求关于 G 的属性闭包，结果属性也必须是此关系模式的属性



4. 关系模式无损连接或保持依赖的分解算法

(1) 关系模式无损连接地分解成BCNF示例

关系模式分解成BCNF

示例： $R(A, B, C, D, E, F, G)$

函数依赖集合{ $A \rightarrow B, A \rightarrow C, C \rightarrow D, C \rightarrow E, E \rightarrow FG$ }

候选键： A ；有不依赖于候选键的其他函数依赖， R 不满足BCNF。

分解规则：

将左侧不含候选键的函数依赖单独组成一个关系，将包含候选键的组成一关系

$\rho = \{ R1(C, D), R2(C, E), R3(E, F, G), R4(A, B, C) \}$

可以看出： $R1 \in BCNF; R2 \in BCNF; R3 \in BCNF; R4 \in BCNF;$

也可以将 $R1$ 和 $R2$ 合并：

$\rho = \{ R12(C, D, E), R3(E, F, G), R4(A, B, C) \}$

ρ 满足无损连接性？ ρ 满足保持依赖性吗？

4. 关系模式无损连接或保持依赖的分解算法



(2) 关系模式无损连接地分解成BCNF算法

[Algorithm]无损连接分解成BCNF的算法。

Input : 关系模式 $R(U, F)$

Output : R 的一个无损连接分解 ρ ， ρ 中的每个关系模式都是 F 在该模式上投影的
BCNF

Method : (1)令 $\rho = \{ R \}$ 。

(2)对每个模式 $s \in \rho$ ，若 $s \notin \text{BCNF}$ ，则 s 上必有 $X \rightarrow A$ 成立且 X 不是 s 的超键且 $A \notin X$ ，用模式 s_1, s_2 替代 s 。 $s_1 = \{A, X\}$ ， $s_2 = \{s - A\}$ (可以发现， $s_1 \in \text{BCNF}$)。

(3)重复步骤(2)，直至 ρ 中全部关系模式达到BCNF。

注：本算法不能保证一关系模式分解成BCNF而又保持依赖。

4. 关系模式无损连接或保持依赖的分解算法



(3) 关系模式保持依赖地分解成3NF示例

关系模式分解成3NF

示例： $R(A, B, C, D, E, F, G)$

函数依赖集合{ $A \rightarrow B, A \rightarrow C, C \rightarrow D, C \rightarrow E, E \rightarrow FG$ }

候选键：A； 有传递依赖，R不满足3NF。

分解规则：

将每一个函数依赖单独组成一个关系

$\rho = \{ R_1(A, B), R_2(A, C), R_3(C, D), R_4(C, E), R_5(E, F, G) \}$

可以看出：每一个模式都属于3NF，且 ρ 是保持依赖的

也可以合并一些关系：

$\rho = \{ R_{12}(A, B, C), R_{34}(C, D, E), R_5(E, F, G) \}$

4. 关系模式无损连接或保持依赖的分解算法



(4)关系模式保持依赖地分解成3NF算法

[Algorithm]保持依赖分解成3NF的算法。

Input : 关系模式 $R(U, F)$, F 是函数依赖集最小覆盖。

Output : R 的一个保持依赖分解 ρ , ρ 中的每个关系模式都是 F 在该模式上投影的3NF。

Method :

(1)把 R 中不出现在 F 中的属性去掉并单独组成一模式。

(2)对 $\forall X \rightarrow A \in F$, 则以 XA 组成一模式; 若有 $X \rightarrow A_1, X \rightarrow A_2, \dots, X \rightarrow A_m$ 都属于 F , 则以 $XA_1A_2 \dots A_m$ 组成一模式(即将 n 个模式合并为一个模式)。

(3)取 ρ 为上述模式之集合, 则 ρ 即为所求之分解。

4. 关系模式无损连接或保持依赖的分解算法



(5)关系模式分解既保持依赖又无损连接

既保持依赖，又无损连接的分解

[定理] 设 σ 是按前述算法构造的 R 的一个第三范式分解， X 是 R 的候选键，则： $\tau = \sigma \cup \{ X \}$ 将是 R 的一个分解，且该分解中的所有关系模式是第三范式的， τ 有保持依赖和无损连接性。

注： τ 并不一定为最小可能关系模式的集合。我们可以依次去掉一个关系模式，只要所要求的性质仍具备，直至求得上述最小集合。

示例： $R(A, B, C, D, E, F, G)$

函数依赖： $A \rightarrow B, A \rightarrow C, C \rightarrow D, C \rightarrow E, E \rightarrow FG$

保持依赖的分解成3NF的集合：

$\rho = \{ R_{12}(A, B, C), R_{34}(C, D, E), R_5(E, F, G) \}$

ρ 满足无损连接性？

4. 关系模式无损连接或保持依赖的分解算法



(6)关系模式分解既保持依赖又无损连接

[Algorithm]无损连接分解成4NF

Input : 关系模式 $R(U, D)$, D 为 R 上的一个依赖集(多值、函数依赖)。

Output : R 的一个无损连接分解 ρ , ρ 中的每个关系模式都是 D 在该模式上投影的4NF。

Method :

(1) 令 $\rho = \{ R \}$ 。

(2)对每个模式 $s \in \rho$, 若 $s \notin 4NF$, 则 s 上必有一依赖 $X \twoheadrightarrow Y$ 成立且 X 不是 s 的超键且 $Y - X \neq \emptyset$, $XY \neq s$, 令 $Z = Y - X$, 显然有 $X \twoheadrightarrow Z$, 此时用模式 s_1, s_2 替代 ρ 中的模式 s , 其中 s_1 由 Y 和 X 构成, s_2 由 $Y - X$ 构成。

(3)重复步骤(2), 直至 ρ 中全部关系模式达到4NF。



4. 关系模式无损连接或保持依赖的分解算法

(7) 连接依赖与第5NF 连接依赖

[Definition] 连接依赖

设 R 为一关系模式， $\rho = \{ R_1, \dots, R_n \}$ 为 R 的一个分解，若对 R 的任一关系 r 均有： $r_{n目} = \pi_{R_1}(r) \bowtie \pi_{R_2}(r) \bowtie \dots \bowtie \pi_{R_n}(r)$ ，则称 R 满足 n 目连接依赖，记为 $JD[R_1, \dots, R_n]$ ，或记为 n -JD。



4. 关系模式无损连接或保持依赖的分解算法

(8)连接依赖与第5NF 关系的第5NF

[Definition] 5NF

当且仅当关系模式R的每个连接依赖均按其候选键进行连接运算时(均由R的候选键所隐含)，则称R是第五范式的，记为 $\mathbf{R} \in 5\text{NF}$ 。

注意：

第五范式消除了不按候选键连接的连接依赖(R的无损连接分解中各模式必含有一个候选键)，但其语义背景抽象。

$5\text{NF} \subseteq 4\text{NF}$ 。第五范式也称投影连接范式，即**PJNF**。



5. 数据库设计需要知道的

数据库设计理论要解决的根本问题

根本的问题

哪些属性被组织成一个关系？

是一个大关系模式呢，还是若干小关系模式？

大关系模式存在什么问题？

学号	姓名	班级	班主任	班主任职称
2003510101	张三	035101	张林	讲师
2003510102	李四	035101	张林	讲师
2003510103	王五	035101	张林	讲师
2003510104	李六	035101	张林	讲师
2003510105	张四	035101	张林	讲师
2003510106	张五	035101	张林	讲师
2003510107	张小三	035101	张林	讲师
2003510108	张小四	035101	张林	讲师
2003510109	李小三	035101	张林	讲师
2003510110	李小四	035101	张林	讲师
2003520201	周三	035202	郑东	副教授
2003520202	赵四	035202	郑东	副教授
2003520203	赵五	035202	郑东	副教授
2003520204	赵六	035202	郑东	副教授
2003520205	钱四	035202	郑东	副教授
2003520206	强五	035202	郑东	副教授
2003520207	梁小三	035202	郑东	副教授
2003520208	梁小四	035202	郑东	副教授
2003520209	王小三	035202	郑东	副教授
2003520210	王小四	035202	郑东	副教授

学号	姓名	班级
2003510101	张三	035101
2003510102	李四	035101
2003510103	王五	035101
2003510104	李六	035101
2003510105	张四	035101
2003510106	张五	035101
2003510107	张小三	035101
2003510108	张小四	035101
2003510109	李小三	035101
2003510110	李小四	035101
2003520201	周三	035202
2003520202	赵四	035202
2003520203	赵五	035202
2003520204	赵六	035202
2003520205	钱四	035202
2003520206	强五	035202
2003520207	梁小三	035202
2003520208	梁小四	035202
2003520209	王小三	035202
2003520210	王小四	035202

班级	班主任	班主任职称
035101	张林	讲师
035202	郑东	副教授



5. 数据库设计需要知道的

关系模式设计的折中

关系模式设计需要折中

遵循关系范式原则，则需要将一个关系模式，拆解成两个或多个小的模式；而查询时，需要将这两个或多个小的模式联结成一个模式；

遵循关系范式原则避免了冗余、插入异常、删除异常等问题，但由于联结运算的低效率，使得查询速度很慢。因此需要折中。

学号	姓名	班级	班主任	班主任职称
2003510101	张三	035101	张林	讲师
2003510102	李四	035101	张林	讲师
2003510103	王五	035101	张林	讲师
2003510104	李六	035101	张林	讲师
2003510105	张四	035101	张林	讲师
2003510106	张五	035101	张林	讲师
2003510107	张小三	035101	张林	讲师
2003510108	张小四	035101	张林	讲师
2003510109	李小三	035101	张林	讲师
2003510110	李小四	035101	张林	讲师
2003520201	周三	035202	郑东	副教授
2003520202	赵四	035202	郑东	副教授
2003520203	赵五	035202	郑东	副教授
2003520204	赵六	035202	郑东	副教授
2003520205	钱四	035202	郑东	副教授
2003520206	强五	035202	郑东	副教授
2003520207	梁小三	035202	郑东	副教授
2003520208	梁小四	035202	郑东	副教授
2003520209	王小三	035202	郑东	副教授
2003520210	王小四	035202	郑东	副教授

学号	姓名	班级
2003510101	张三	035101
2003510102	李四	035101
2003510103	王五	035101
2003510104	李六	035101
2003510105	张四	035101
2003510106	张五	035101
2003510107	张小三	035101
2003510108	张小四	035101
2003510109	李小三	035101
2003510110	李小四	035101
2003520201	周三	035202
2003520202	赵四	035202
2003520203	赵五	035202
2003520204	赵六	035202
2003520205	钱四	035202
2003520206	强五	035202
2003520207	梁小三	035202
2003520208	梁小四	035202
2003520209	王小三	035202
2003520210	王小四	035202

班级	班主任	班主任职称
035101	张林	讲师
035202	郑东	副教授

通常建议：
关系模式符合BCNF

回顾学习

