

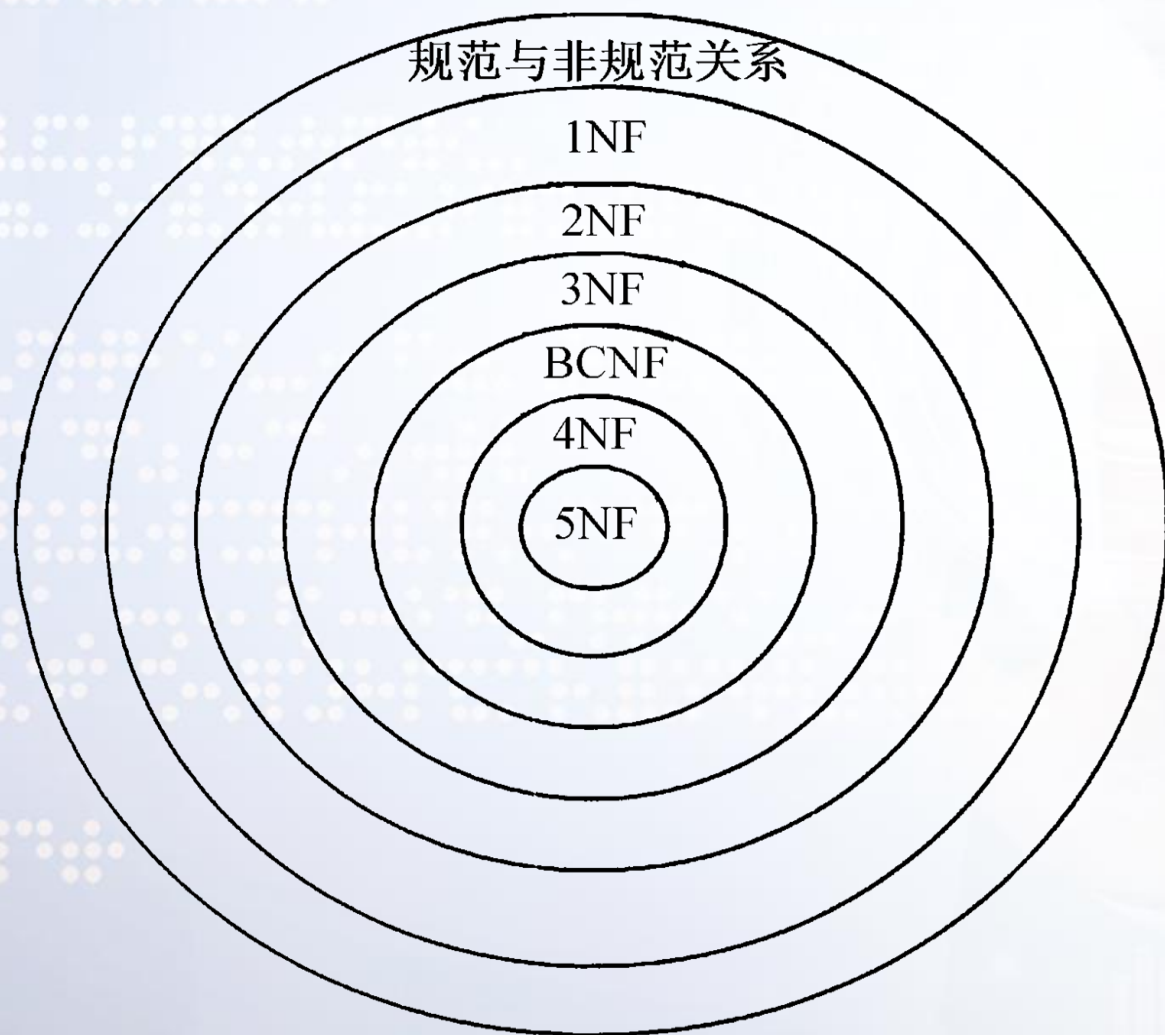
北京林业大学

数据库原理与应用

关系模式的范式



各种范式之间的关系





第一范式

定义4.14

如果关系模式 R 所有的属性均为原子属性，即每个属性都是不可再分的，则称 R 属于第一范式，简称1NF，记作 $R \in 1NF$ 。



第一范式

- ◆ 1NF是关系模式应具备的最起码的条件。
- ◆ 第一范式可能具有大量的数据冗余，存在插入异常、删除异常和更新异常等弊端。
- ◆ 如关系模式SCD属于1NF，它既存在完全函数依赖，又存在部分函数依赖和传递函数依赖。
- ◆ 克服这些弊端的方法是用投影运算将关系分解，去掉过于复杂的函数依赖关系，向更高一级的范式进行转换。

第二范式



第二范式的定义

如果关系模式 $R \in 1NF$ ，且每个非主属性都完全函数依赖于 R 的主码，则称 R 属于第二范式，简称 $2NF$ ，记作 $R \in 2NF$ 。

如：关系模式 $TC (T, C)$

主码 $\rightarrow (T, C)$ ；主属性 $\rightarrow T, C$

不存在非主属性对主码的部分函数依赖，因此 $TC \in 2NF$ 。



第二范式的定义

在关系模式SCD中：
SNO, CNO为主属性；
AGE, DEPT, SN, MN, SCORE均为
非主属性；

由SCD分解的三个关系模式S, D, SC,
其中S的主码为SNO, D的主码为DEPT,
都是单属性, 不可能存在部分函数依赖。



第二范式的定义

从1NF关系中消除非主属性对主码的部分函数依赖，则可得到2NF关系

如果 R 的主码为单属性，或 R 的全体属性均为主属性，则 $R \in 2NF$



2NF规范化

2NF规范化是指把1NF关系模式通过投影分解，转换成2NF关系模式的集合。

[例] 将SCD(SNo, SN, Age, Dept, MN, CNo, Score)规范为2NF。

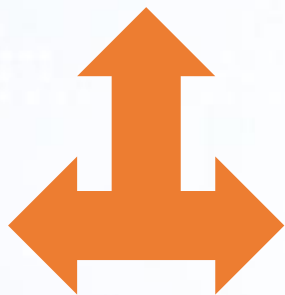
$SNO \rightarrow SN$, $SNO \rightarrow AGE$, $SNO \rightarrow DEPT$,
 $(SNO, CNO) \rightarrow SCORE$



2NF规范化

SCD

学生
SD(SNo,SN,Age,
Dept,MN)



学生与课程联系
SC(SNo,CNo,
Score)

非主属性对主码完全函数依赖。因此，
 $SD \in 2NF$ ， $SC \in 2NF$ 。



2NF的缺点

数据冗余

每个系名和系主任的名字存储的次数等于该系的学生人数

插入异常

当一个新系没有招生时，有关该系的信息无法插入



2NF的缺点

删除异常

某系学生全部毕业而没有招生时，删除全部学生的记录也随之删除了该系的有关信息

更新异常

更换系主任时，仍需改动较多的学生记录

第三范式



3NF的定义

如果关系模式 $R \in 2NF$ ，且每个非主属性都不传递函数依赖于 R 的主码，则称 R 属于第三范式，简称3NF，记作 $R \in 3NF$ 。

如：SC(SNo, CNo, Score)
函数依赖为 $(SNo, CNo) \rightarrow Score$ ，非主属性Score不传递函数依赖于主码 (SNo, CNo) ，因此， $SC \in 3NF$ 。

又如：SD(SNo, SN, Age, Dept, MN)
 $SNo \rightarrow Dept$ 和 $Dept \rightarrow MN \leftrightarrow SNo \xrightarrow{t} MN$
非主属性MN与主码SNo间存在着传递函数依赖，所以 $SD \notin 3NF$ 。



3NF的规范化

算法1 把一个关系模式分解为3NF，使它具有保持函数依赖性。

输入：关系模式 R 和 R 的最小函数依赖集 F_{min}

输出： R 的一个保持函数依赖的分解 $\rho=\{R_1, R_2, \dots, R_k\}$ ，
每个 R_i 相对于 $\Pi_{R_i}(F_{min})$ 是3NF模式。

- ◆ (1)如果 F_{min} 中有一函数依赖 $X \rightarrow A$ ，且 $XA=R$ ，则输出 $\rho=\{R\}$ ，转 (4)。
- ◆ (2)如果 R 中某些属性与 F_{min} 中所有依赖的左部和右部都无关，则将它们构成关系模式，从 R 中将它们分出去，单独构成一个模式。
- ◆ (3)对于 F_{min} 中的每一个函数依赖 $X \rightarrow A$ ，都单独构成一个关系子模式 XA 。若 F_{min} 中有 $X \rightarrow A_1, X \rightarrow A_2, \dots, X \rightarrow A_n$ ，则可以用模式 $XA_1A_2\dots A_n$ 取代 n 个模式 XA_1, XA_2, \dots, XA_n 。
- ◆ (4)停止分解，输出 ρ 。



3NF的规范化

[例]将SD(SNo, SN, Age, Dept, MN)
规范到3NF且具有保持函数依赖性。

- ◆ 求出关系模式SD的最小函数依赖集 $F_{min} = \{SNo \rightarrow SN, SNo \rightarrow Age, SNo \rightarrow Dept, Dept \rightarrow MN\}$
- ◆ 根据算法中 (1), F 中没有满足条件的函数依赖
- ◆ 根据算法中 (2), F 中没有满足条件的函数依赖
- ◆ 根据算法中 (3), 将 R 分解为 $S = \{SNo, SN, Age, Dept\}$,
 $D = \{Dept, MN\}$ 。



3NF的规范化

算法2 把一个关系模式分解为3NF，使它既具有无损连接性又具有保持函数依赖性。

输入：关系模式 R 和 R 的最小函数依赖集 F_{min}

输出： R 的一个分解 $\rho=\{R_1, R_2, \dots, R_k\}$ ， R_i 为3NF， ρ 具有无损连接性和保持函数依赖性

- ◆ (1)根据3NF规范化算法求出保持函数依赖的分解： $\rho=\{R_1, R_2, \dots, R_k\}$ 。
- ◆ (2)判定 ρ 是否具有无损连接性，若是，转 (4) 。
- ◆ (3)令 $\rho=\rho \cup \{X\}=\{R_1, R_2, \dots, R_k, X\}$ ，其中 X 是 R 的候选码。
- ◆ (4)输出 ρ 。



3NF的规范化

[例] 将SD(SNo, SN, Age, Dept, MN)规范到3NF。

◆ 根据3NF的规范化算法求出保持函数依赖的分解：

$\rho = \{S(SNo, SN, Age, Dept), D(Dept, MN)\}$

◆ 判定 ρ 是否具有无损连接性

Dept \rightarrow MN

	SNo	SN	Age	Dept	MN
S(SNo, SN, Age, Dept)	a_1	a_2	a_3	a_4	a_5
D(Dept, MN)	b_{21}	b_{22}	b_{23}	a_4	a_5

$a_1a_2a_3a_4a_5$, ρ 相对于F是无损连接分解



3NF解决了2NF中存在的四个问题:

○ (1) 数据冗余降低了

○ (2) 不存在插入异常

○ (3) 不存在删除异常

○ (4) 不存在更新异常

BC范式



BC 范式

BC 范式的定义

如果关系模式 $R \in 1NF$ ，且所有的函数依赖 $X \rightarrow Y$ ($Y \notin X$)，决定因素 X 都包含了 R 的一个候选码，则称 R 属于 BC 范式，记作 $R \in BCNF$ 。



BC 范式

BCNF 具有如下性质：

如果 $R \in \text{BCNF}$ ，则 R 也是 3NF 。

如果 $R \in 3\text{NF}$ ，则 R 不一定是BCNF。



BC 范式

无部分函数依赖和传递函数
依赖, $SNC \in 3NF$



[例] 设有关系模式SNC (SN_0 , SN, CN_0 , Score)



$SN_0 \leftrightarrow SN$



存在着主属性对主码的部分函数依赖: $(SN_0, CN_0) \xrightarrow{p} SN$, $(SN, CN_0) \xrightarrow{p} SN_0$, 所以SNC不是BCNF。



BCNF 规范化

算法 把一个关系模式分解为BCNF

输入：关系模式 R 和 R 的函数依赖集 F

输出： R 的一个无损连接分解 $\rho = \{R_1, R_2, \dots, R_k\}$ ，
每个 R_i 相对于 $\Pi_{R_i}(F)$ 是BCNF模式。

- (1) 令 $\rho = \{R\}$ 。
- (2) 如果 ρ 中所有模式都是BCNF，则转 (4) 。
- (3) 如果 ρ 中有一个关系模式 S 不是BCNF，则 S 中必能找到一个函数依赖 $X \rightarrow A$ 且 X 不是 S 的候选码，且 A 不属于 X ，设 $S_1 = XA$ ， $S_2 = S - (A - X)$ ，用分解 $\{S_1, S_2\}$ 代替 S ，转 (2) 。
- (4) 分解结束，输出 ρ 。



BCNF 规范化



[例] 将SNC(SNo, SN, CNo, Score)规范到BCNF。



候选码：(SNo, CNo) 和 (SN, CNo)



函数依赖：

$F = \{ SNo \rightarrow SN, SN \rightarrow SNo, (SNo, CNo) \rightarrow Score, (SN, CNo) \rightarrow Score \}$



BCNF 规范化

- (1) 令 $\rho = \{SNC(SNo, SN, CNo, Score)\}$ 。
- (2) 经过前面分析可知, ρ 中关系模式不属于BCNF。
- (3) 用分解 $\{S_1(SNo, SN), S_2(SNo, CNo, Score)\}$ 代替SNC。
- (4) 分解结果为: $S_1(SNo, SN)$ 描述学生实体; $S_2(SNo, CNo, Score)$ 描述学生与课程的联系。