

第6章 关系数据库模式设计

本章主要内容

- 关系模式的设计问题
- 函数依赖
- 关系模式的分解
- 关系模式的范式

四、关系模式的范式

- 范式的概念
- 函数依赖图
- 1NF
- 2NF
- 3NF
- BCNF
- 4NF
- 5NF



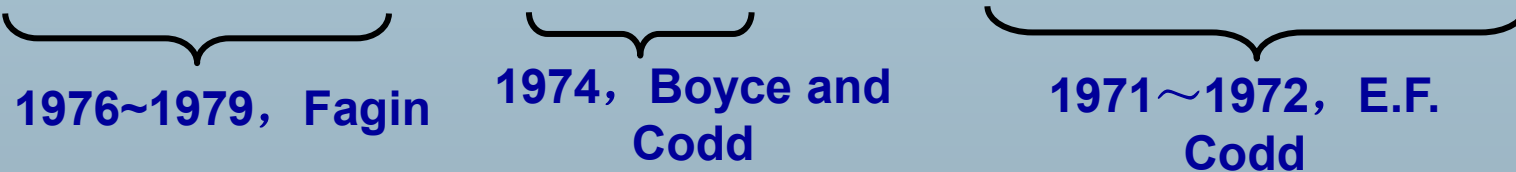
1、范式的概念

■ 范式：满足特定要求的模式

- 不同级别的范式要求各不相同
- 范式可以作为衡量一个关系模式好坏的标准
- 若关系模式R满足范式 xNF ，记作 $R \in xNF$

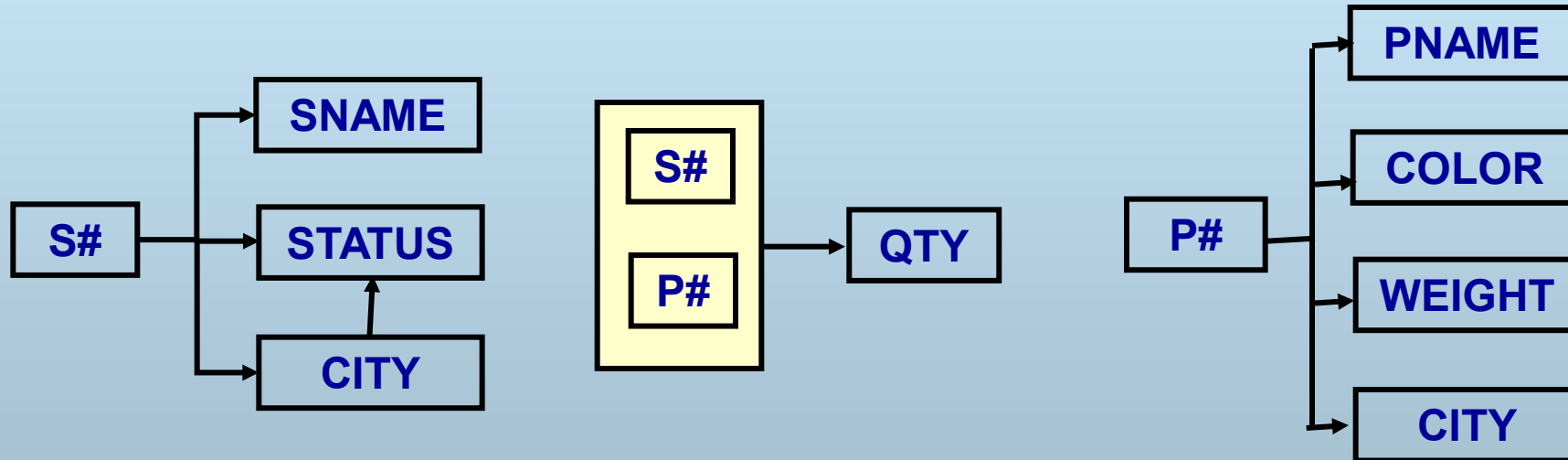
■ 规范化：将低一级范式的关系模式通过模式分解转换为高一级范式的关系模式集合的过程

■ $5NF \subset 4NF \subset BCNF \subset 3NF \subset 2NF \subset 1NF$



2、函数依赖图

- R是关系模式，F是R的一个FD集，F可用函数依赖图表达



箭头表示函数决定关系，每个候选码必定有箭头指出

3、1NF

- 对于关系模式R的任一实例，其元组的每一个属性值都只含有一个值，则 $R \in 1NF$
 - 1NF是关系的基本要求
 - R不满足1NF会带来更新时的二义性
 - 若R中加入“成绩”属性，则{学号,课程} \rightarrow 成绩难以表达

学号	课程
01	数据库
02	{C++, 数据库}

4、2NF

- (假定R只有一个候选码/主码) 当且仅当R属于1NF, 且R的每一个非主属性都完全函数依赖于主码时, $R \in 2NF$
 - 完全函数依赖: 对于函数依赖 $W \rightarrow A$, 若不存在 $X \subset W$, 并且 $X \rightarrow A$ 成立, 则称 $W \rightarrow A$ 为完全函数依赖, 否则为局部函数依赖
 - 主属性: 包含在候选码中的属性
 - 非主属性: 不包含在任何候选码中的属性

(1) 2NF含义

- $R(A,B,C,D,E)$, $\{A,B\}$ 为主码, 则有
- $AB \rightarrow C$, $AB \rightarrow D$, $AB \rightarrow E$
- 但C、D、E都不局部函数依赖于AB
- 即 $A \rightarrow C$ 、 $B \rightarrow C$ 、 $A \rightarrow D$ 、 $B \rightarrow D$ 、 $A \rightarrow E$ 、 $B \rightarrow E$ 中任何一个均不成立

(2) 2NF例子

■ 供应关系

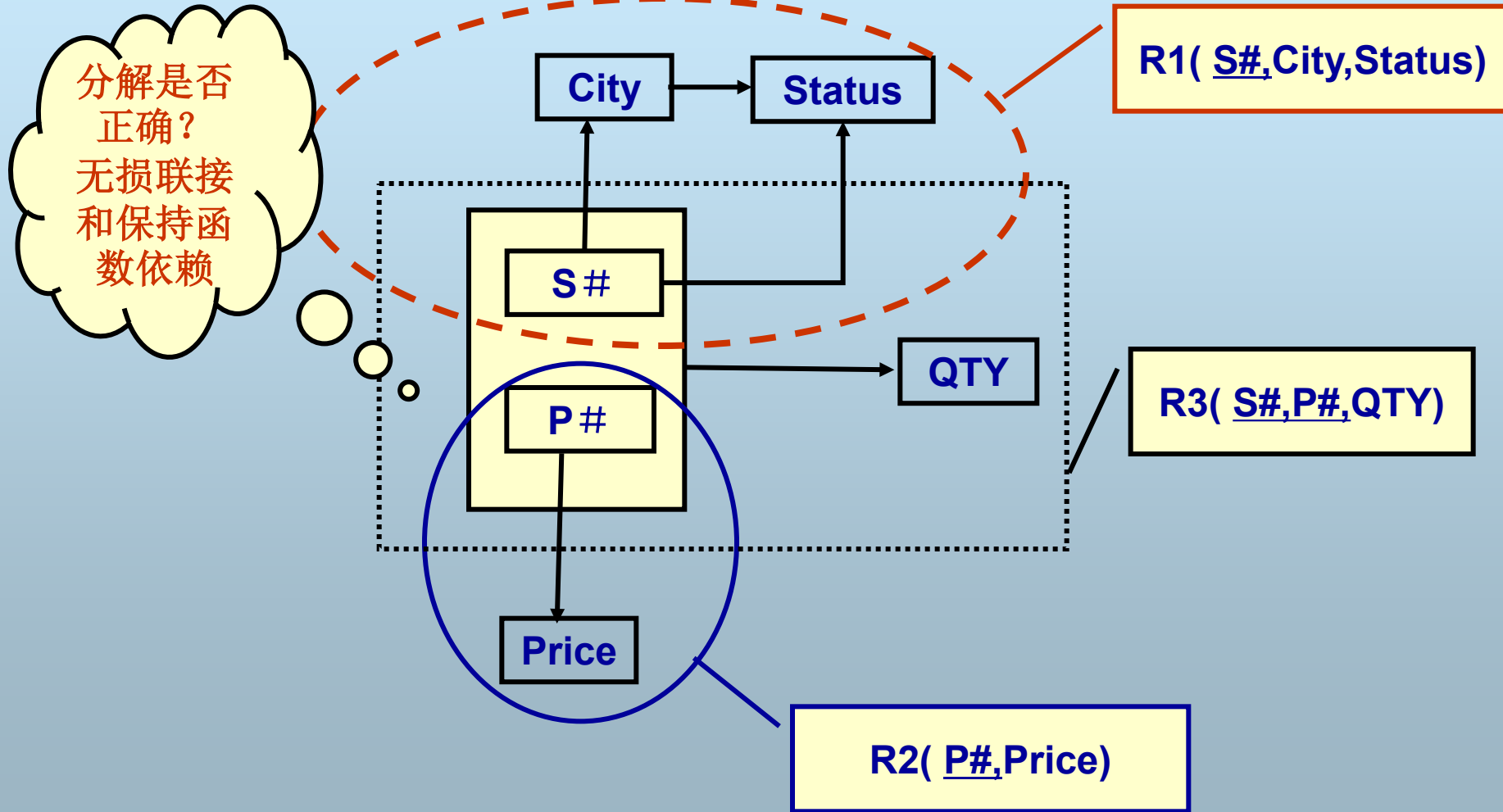
- $R(S\#, P\#, city, status, Price, QTY)$
- $F = \{S\# \rightarrow city, S\# \rightarrow status, P\# \rightarrow Price, city \rightarrow status, \{S\#, P\#\} \rightarrow QTY\}$
- 所以主码为 $\{S\#, P\#\}$
- 但 $city$ 和 $Price$ 都局部函数依赖于主码
- 所以 $R \notin 2NF$

(3) 不满足2NF带来的问题

- **R(S #, P #, city, status, Price, QTY)**
 - **插入异常**: 没有供应零件的供应商无法插入
 - **删除异常**: 删除供应商的供货信息同时删除了供应商的其它信息
 - **更新异常**: 供应商的city修改时必须修改多个元组
 - **数据冗余**: 同一供应商的city被重复存储

(3) 模式分解以满足2NF

■ R(S#, P#, City, Status, Price, QTY)



5、3NF

- (假定R只有一个候选码，且该候选码为主码)当且仅当R属于2NF，且R的每一个非主属性都不传递依赖于主码时，
 $R \in 3NF$
- 传递依赖：若 $Y \rightarrow X$ ， $X \rightarrow A$ ，并且 $X \not\rightarrow Y$ ，A不是X的子集，则称A传递依赖于Y

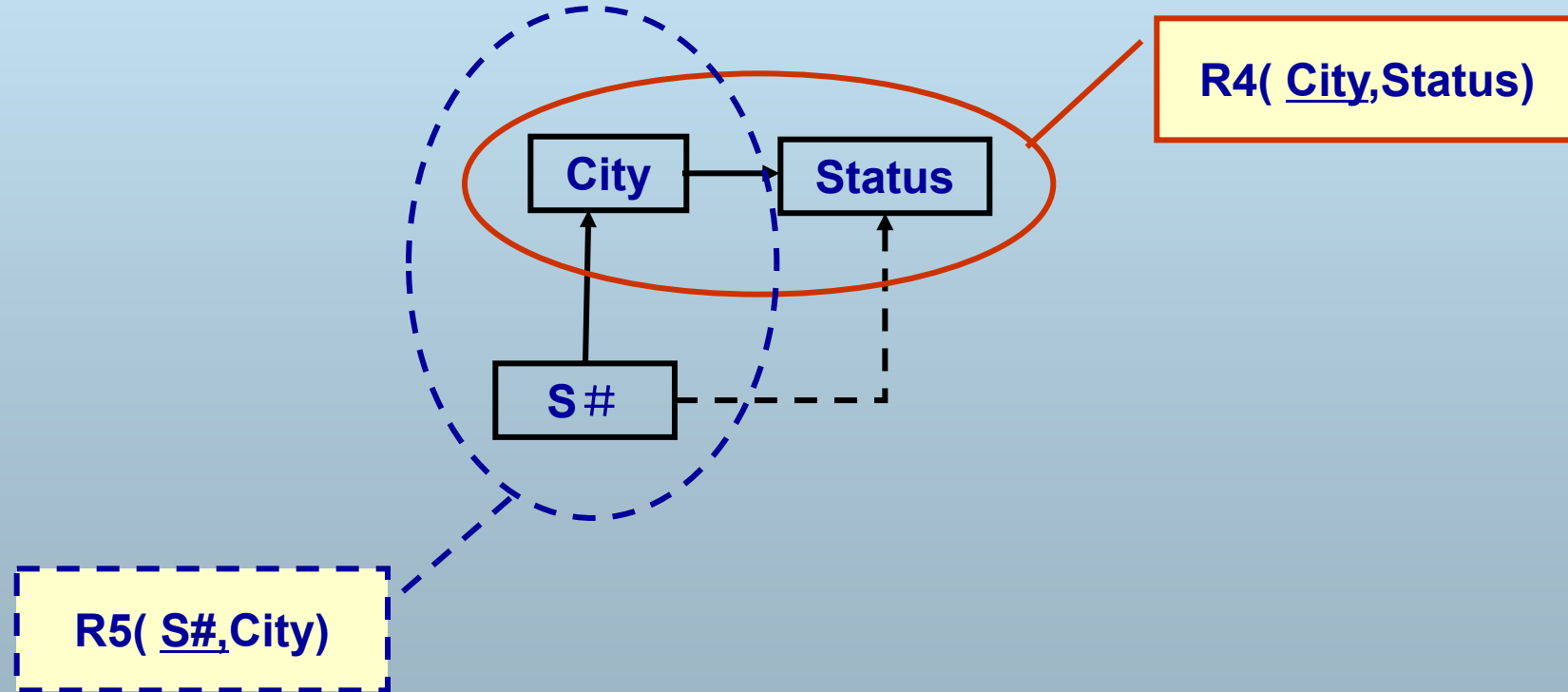
(1) 不满足3NF带来的问题

■ R1(S#,City,Status)

- **插入异常**：不能插入一个具有status但没有供应商的city，例如Rome的status为50，但除非有一个供应商住在Rome否则无法插入
- **删除异常**：删除供应商时会同时删除与该城市相关的status信息
- **更新异常**：一个城市中会有多个供应商，因此status更新时要更新多个元组
- **数据冗余**：同一城市的status冗余存储

(2) 分解2NF到3NF

- R1(S#,City,Status)
- 去掉传递依赖



6、BCNF

■ Boyce/Codd范式

■ 2NF和3NF

- 假设了R只有一个候选码，但如果R有多个候选码并且不同的候选码之间还可能相互重叠，会出现什么情况？
- 2NF和3NF只考虑了非主属性到码的函数依赖

■ BCNF扩充了3NF，可以处理R有多个候选码的情形

- 进一步考虑了主属性到码的函数依赖
- 进一步考虑了主属性对非主属性的函数依赖

(1) 多候选码的例子

- 假设供应商的名字是唯一的
- 供应关系R(S#,SNAME,P#,QTY)存在两个候选码
 - {S#,P#}和{SNAME, P#}
 - R属于3NF, WHY?

{SNAME,P#} → QTY, {S#,P#} → QTY,
S# → SNAME, SNAME → S#

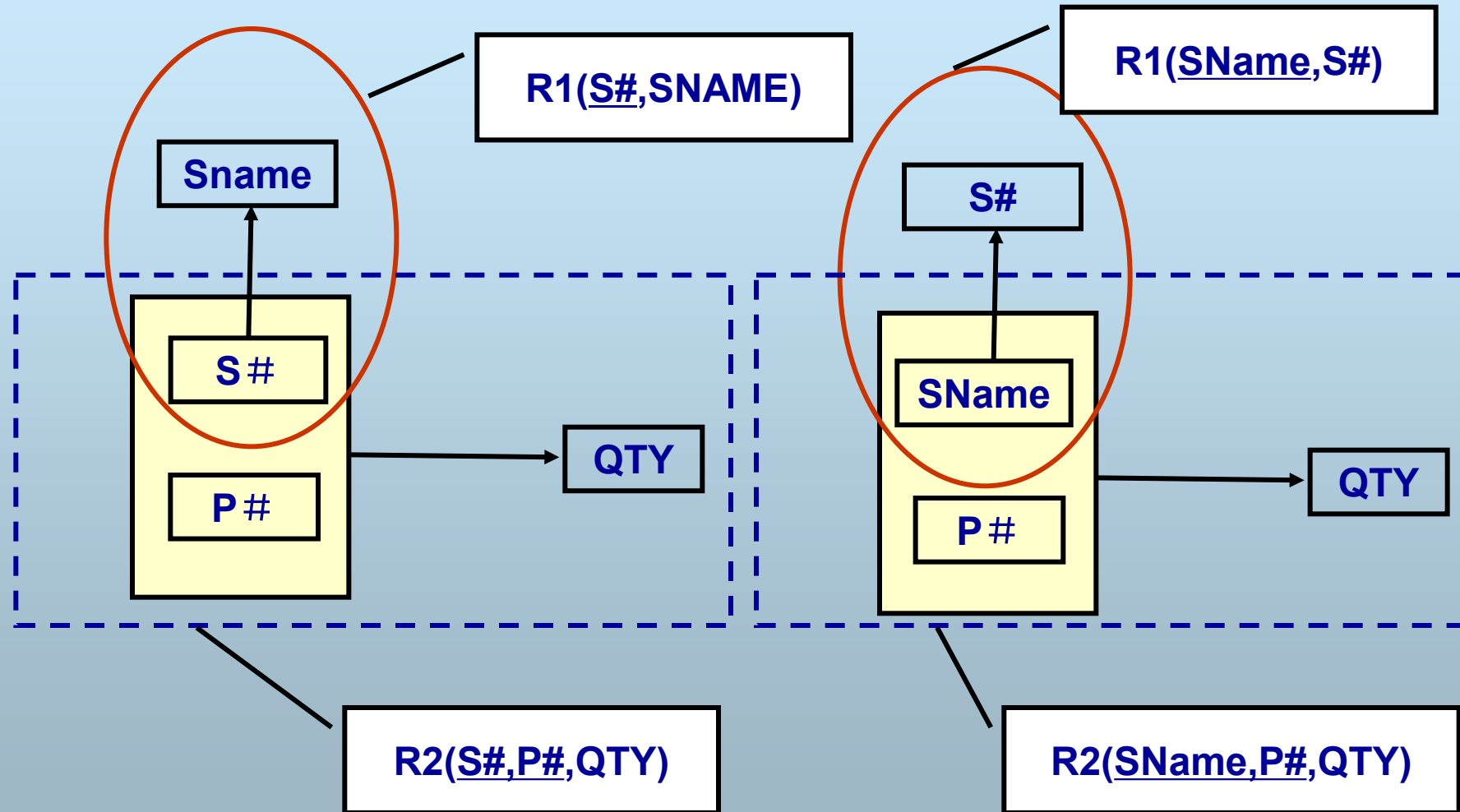
S#	SNAME	P#	QTY
s1	Intel	p1	300
s1	Intel	p2	200
s1	Intel	P3	400
s2	Acer	p1	200

(2) 存在的问题

- **数据冗余：** s1的名字Intel重复存储
- **更新异常：** 修改s1的名字时必须修改多个元组
- **删除异常：** 若s2现在不提供任何零件，则须删除s2的元组，但同时删除了s2的名字
- **插入异常：** 没有提供零件的供应商无法插入

S#	SNAME	P#	QTY
s1	Intel	p1	300
s1	Intel	p2	200
s1	Intel	P3	400
s2	Acer	p1	200

(3) 解决方法 (3NF->BCNF)

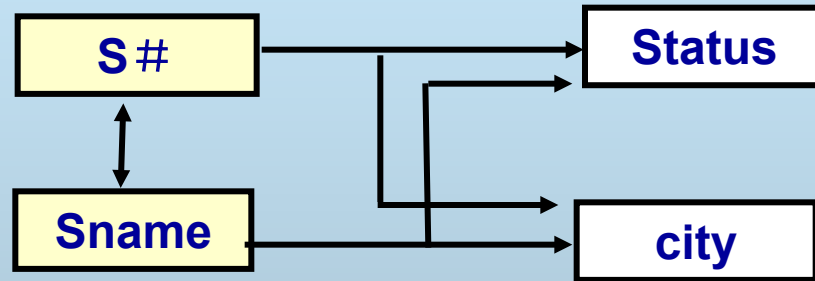


(4) BCNF定义

- 如果关系模式R的所有不平凡、完全的函数依赖的决定因素（左边的属性集）都是候选码，则 $R \in \text{BCNF}$
 - **3NF**: 不允许非主属性到非码的FD，但允许主属性到其它属性的FD
 - **BCNF**: 不允许主属性、非主属性到非码的FD

(5) BCNF例子1

- $R(S\#, SNAME, STATUS, CITY)$
- 设Sname唯一

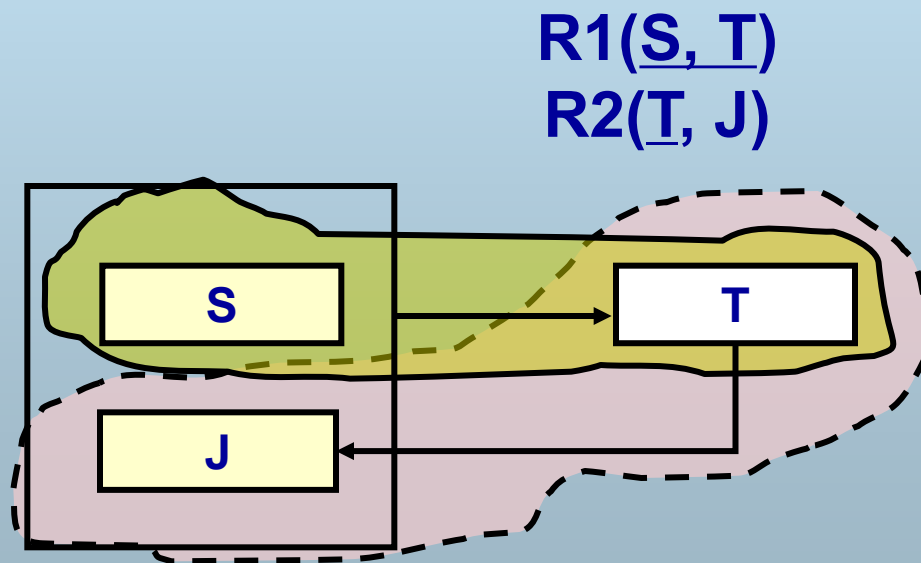


$Sname \rightarrow city,$
 $S\# \rightarrow city,$
 $S\# \rightarrow Sname,$
 $Sname \rightarrow S\#,$
 $Sname \rightarrow Status,$
 $S\# \rightarrow Status$

- BCNF模式的函数依赖图中，箭头都是从候选码中引出，所有不平凡FD的左边都是候选码

(6) BCNF例子2

- $R(S, J, T)$ --- 学号, 课程号, 教师名
- 每个教师只教一门课, 每门课有若干任课教师, 学生选定一门课就对应一个固定的教师
- $T \rightarrow J, \{S, J\} \rightarrow T$
- R属于3NF
- R不属于BCNF



分解到BCNF不一定能保持函数依赖

五、规范化过程总结

- 对**1NF**模式投影，消除非主属性对码的局部函数依赖，产生**2NF**
- 对**2NF**模式投影，消除非主属性对码的传递函数依赖，产生**3NF**
- 对**3NF**模式投影，消除左边不是候选码的函数依赖，产生**BCNF**

五、规范化过程总结

- 整个讨论过程只采用了两种操作：投影和自然联接
 - 以投影来分解
 - 以自然连接来重构

五、规范化过程总结

- **定理1**: 若要求保持函数依赖和无损联接, 则总可以达到**3NF**, 但不一定满足**BCNF**
- **定理2**: 若要求模式分解保持函数依赖, 则总可以分解到满足**3NF**, 但不一定满足**BCNF**
 - **BCNF**可以达到无损连接, 但不一定保持函数依赖

六、模式分解的几个算法

■ 算法1

- 保持函数依赖地分解到3NF的算法

■ 算法2

- 无损并且保持函数依赖分解为3NF的算法

■ 算法3

- 无损分解为BCNF的算法

算法1：保持函数依赖地分解到3NF

1. 求出 $R\langle U, F \rangle$ 的最小函数依赖集（仍记为 F ）
2. 把所有不在 F 中出现的属性组成一个关系模式 R' ，并在 U 中去掉这些属性(剩余属性仍记为 U)
3. 若 F 中存在 $X \rightarrow A$ ，且 $XA=U$ ，则输出 $R(U)$ 和 R' ，算法结束，否则
4. 对 F 按相同的左部分组，将所有 $X \rightarrow A_1, X \rightarrow A_2, \dots, X \rightarrow A_k$ 形式的FD分为一组，并将每组涉及的所有属性作为一个关系模式输出。若某个关系模式 R_i 的属性集是另一个关系模式的属性集的子集，则在结果中去掉 R_i 。设最后得到关系模式 R_1, R_2, \dots, R_k ，则 $p=\{R_1, R_2, \dots, R_k, R'\}$ 一个保持函数依赖的分解，并且满足3NF

例子

- $R(ABCDEF), F = \{A \rightarrow B, AC \rightarrow E\}$
- 求最小FD集 $F = \{A \rightarrow B, AC \rightarrow E\}$
- $R'(DF)$
- 按左部分组: $R1(AB), R2(ACE)$
- $p = \{R'(DF), R1(AB), R2(ACE)\}$

算法2：无损连接且保持函数依赖地分解到3NF

- 首先用算法1求出R的保持函数依赖的3NF分解，设为 $q=\{R_1, R_2, \dots, R_k\}$
- 设X是R的主码，求出 $p=q \cup \{R(X)\}$
- 若X是q中某个 R_i 的子集，则在p中去掉 $R(X)$
- 得到的p就是最终结果

(1) 例子1

- $R(S\#, SN, P, C, S, Z),$
 $F = \{S\# \rightarrow SN, S\# \rightarrow P, S\# \rightarrow C, S\# \rightarrow S, S\# \rightarrow Z, \{P, C, S\} \rightarrow Z, Z \rightarrow P, Z \rightarrow C\}$
- 1. 求出最小FD集: $F = \{S\# \rightarrow SN, S\# \rightarrow P, S\# \rightarrow C, S\# \rightarrow S, \{P, C, S\} \rightarrow Z, Z \rightarrow P, Z \rightarrow C\}$ // $S\# \rightarrow Z$ 冗余
- 2. $q = \{R1(S\#, SN, P, C, S), R2(P, C, S, Z), R3(Z, P, C)\}$
- 3. $R3$ 是 $R2$ 的子集, 所以去掉 $R3$
 $q = \{R1(S\#, SN, P, C, S), R2(P, C, S, Z)\}$
- 4. R 的主码为 $S\#$, 于是
 $p = q \cup \{R(X)\} = \{R1(S\#, SN, P, C, S), R2(P, C, S, Z), R(S\#)\}$
- 5. 因为 $\{S\#\}$ 是 $R1$ 的子集, 所以从 p 中去掉 $R(S\#)$
- 6. $p = \{R1(S\#, SN, P, C, S), R2(P, C, S, Z)\}$ 即最终结果

(2) 例子2

- $R(S\#, SN, P, C, S, Z)$,
 $F = \{S\# \rightarrow SN, S\# \rightarrow P, S\# \rightarrow C, Z \rightarrow S, Z \rightarrow C\}$
- 1. 求出最小FD集: $F = \{S\# \rightarrow SN, S\# \rightarrow P, S\# \rightarrow C, Z \rightarrow S, Z \rightarrow C\}$
- 2. $q = \{R1(S\#, SN, P, C), R2(Z, S, C)\}$
- 3. R的主码为 $\{S\#, Z\}$, 于是
 $p = q \cup \{R(X)\} = \{R1(S\#, SN, P, C), R2(Z, S, C), R(S\#, Z)\}$
- 4. $p = \{R1(S\#, SN, P, C), R2(Z, S, C), R(S\#, Z)\}$ 即最终结果

算法3：无损联接地分解R到BCNF

- 输入： $R\langle U, F \rangle$; 输出： p
- 1. $p := \{R\}$;
- 2. 检查 p 中各关系模式是否都属于BCNF，若是，则算法终止
- 3. 设 p 中 $S(U_s)$ 非BCNF关系模式，则必存在 $X \rightarrow A$ ，其中 X 不是 S 的超码；
 - ① 将 S 分解为 $S_1(XA)$ 和 $S_2(U_s - A)$ ，此分解是无损联接的
// $(\{XA\} \cap \{U_s - A\} = X) \rightarrow (A = \{XA\} - \{U_s - A\})$
 - ② $p := \{p - S\} \cup \{S_1, S_2\}$; // 用 S_1 和 S_2 替换 p 中的 S
 - ③ 转到第2步;
- 4. 由于 U 的属性有限，因此有限次循环后算法终止

例子

- $R(S\#,C\#,G,TN,D)$,
 $F=\{\{S\#,C\#\} \rightarrow G, C\# \rightarrow TN, TN \rightarrow D\}$
- $\rho := \{R\};$
- $TN \rightarrow D$ 不满足BCNF定义, 分解R
 $\rho := \{R1(S\#,C\#,G,TN), R2(TN,D)\}$
- R1中 $C\# \rightarrow TN$ 不满足BCNF, 分解R1为R3和R4
 $\rho := \{R3(S\#,C\#,G), R4(C\#,TN), R2(TN,D)\}$
- ρ 中各模式均满足BCNF, 结束

例子 (续)

- $R(S\#,C\#,G,TN,D)$,
 $F=\{\{S\#,C\#\} \rightarrow G, C\# \rightarrow TN, TN \rightarrow D\}$
- 如果先选择处理 $C\# \rightarrow TN$?
 - $C\# \rightarrow TN$ 不满足BCNF定义, 分解R
 $\rho := \{R1(\underline{S\#}, \underline{C\#}, G, \underline{D}), R2(\underline{C\#}, TN)\}$
 - R1中 $\{S\#,C\#\} \rightarrow G$ 不满足BCNF, 分解R1为R3和R4
 $\rho := \{R3(\underline{S\#}, \underline{C\#}, G), R4(\underline{S\#}, \underline{C\#}, D), R2(\underline{C\#}, TN)\}$
 - ρ 中各模式均满足BCNF, 结束

结论: 无损分解到BCNF的结果不唯一!

本章小结

- 模式设计理论是数据库逻辑设计的理论基础，目的是根据初始的数据库模式构造出合适的数据库模式
- 函数依赖
- 模式分解
 - 无损联接
 - 保持函数依赖
- 规范化理论
 - 1NF、2NF、3NF、BCNF
- 模式分解的算法