

关系数据理论

01 | 关系数据库的规范化理论

02 | 函数依赖和码

03 | 范式

04 | 数据依赖的公理系统

05 | 模式分解的定义及算法

关系数据理论

01 | 关系数据库的规范化理论

02 | 函数依赖和码

03 | 范式

04 | 数据依赖的公理系统

05 | 模式分解的定义及算法

- ① 给定一个关系模式（关系表），可能存在哪几个方面的问题？





例 1: 描述学校教学管理的关系模式如下所示, 分析其主码及存在的问题:

SNO (学号)	SN (姓名)	AGE (年龄)	DEPT (系别)	MN (系主任名)	CNO (课程号)	SCORE (成绩)
08001	李丽	17	计算机	刘伟	C1	85
08001	李丽	17	计算机	刘伟	C2	88
08002	王鹏	18	自动化	王平	C2	67
08002	王鹏	18	自动化	王平	C4	50
08002	王鹏	18	自动化	王平	C5	77
08002	王鹏	18	自动化	王平	C6	72
08005	孙珊	19	自动化	王平	C5	90
08005	孙珊	19	自动化	王平	C6	89
08005	孙珊	19	自动化	王平	C7	95
08110	赵义	20	电信	常乐	C1	75

问题 1：数据冗余

SNO	SN	AGE	DEPT	MN	CNO	SCORE
08001	李丽	17	计算机	刘伟	C1	85
08001	李丽	17	计算机	刘伟	C2	88
08002	王鹏	18	自动化	王平	C2	67
08002	王鹏	18	自动化	王平	C4	50
08002	王鹏	18	自动化	王平	C5	77
08002	王鹏	18	自动化	王平	C6	72
08005	孙珊	19	自动化	王平	C5	90
08005	孙珊	19	自动化	王平	C6	89
08005	孙珊	19	自动化	王平	C7	95
08110	赵义	20	电信	常乐	C1	75

- ✓ 系别 / 系主任名存储次数 = 该系学生人数 × 每个学生选修的课程门数
- ✓ 学生姓名、年龄重复存储多次。
- ✓ 数据的冗余度很大，浪费了存储空间。

问题 2：插入异常

SNO	SN	AGE	DEPT	MN	CNO	SCORE
08001	李丽	17	计算机	刘伟	C1	85
08001	李丽	17	计算机	刘伟	C2	88
08110	赵义	20				

请思考：为什么不能插入这些信息？

- ✓ 如果某个新系没有招生，即尚无学生时，则系名和系主任信息无法插入到数据库中。
- ✓ 如果某个学生尚未选课，他的基本信息不能插入到数据库中。

问题 3：删除异常

SNO	SN	AGE	DEPT	MN	CNO	SCORE
08001	李丽	17	计算机	刘伟	C1	85
08001	李丽	17	计算机	刘伟	C2	88
08002	王鹏	18	自动化	王平	C2	67
08002	王鹏	18	自动化	王平	C4	50
08002	王鹏	18	自动化	王平	C5	77
08002	王鹏	18	自动化	王平	C6	72
08002	王鹏	18	自动化	王平	C7	80

不选课

- 如果某个学生不再选修课程，为保证实体完整性，删除整个元组，丢失了该学生的其它信息。

问题 4：更新异常

SNO	SN	AGE	DEPT	MN	CNO	SCORE
08001	李丽	17	计算机	刘伟	C1	85
08001	李丽	17	计算机	刘伟	C2	88
08002	王鹏飞	18	自动化	张俊	C2	67
08002	王鹏飞	18	自动化	张俊	C4	50
08002	王鹏飞	18	自动化	张俊	C5	77
08002	王鹏飞	18	自动化	张俊	C6	72
08005	孙珊	19	自动化	张俊	C5	90
08005	孙珊	19	自动化	张俊	C6	89
08005	孙珊	19	自动化	张俊	C7	95
08110	赵义	20	电信	张俊	C1	75

- 如果学生改名，则该学生的所有记录都要逐一修改 SN.
- 某系要更换系主任，则属于该系的学生记录都要修改 MN 的内容

- ① 给定一个关系模式（关系表），可能存在哪几个方面的问题？
- ② 衡量一个关系模式“好 / 坏”的依据是？



好的关系模式应具备以下四个条件：

- ① 尽可能少的数据冗余
- ② 没有插入异常
- ③ 没有删除异常
- ④ 没有更新异常

请思考：

- 关系模式中能不能完全消除数据冗余？

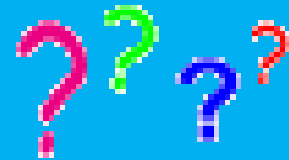
- ① 给定一个关系模式（关系表），可能存在哪几个方面的问题？
- ② 衡量一个关系模式“好 / 坏”的依据是？
- ③ 什么原因导致数据冗余、插入异常、删除异常和更新异常？



不好的关系模式存在：数据冗余、插入异常、删除异常、更新异常等问题。

原因：关系模式中存在某些数据依赖。

什么是关系模式中的数据依赖？



- **数据依赖**是一个关系内部属性与属性之间的约束关系。
用来定义属性值间的相互关联，主要体现于属性值的相等与否。
- **包括：函数依赖、多值依赖等。**

SNO (学号)	SN (姓名)	AGE (年龄)	DEPT (系别)	MN (系主任名)	CNO (课程号)	SCORE (成绩)
08001	李丽	17	计算机	刘伟	C1	85
08001	李丽	17	计算机	刘伟	C2	88
08002	王鹏	18	自动化	王平	C2	67
08002	王鹏	18	自动化	王平	C4	50
08002	王鹏	18	自动化	王平	C5	77
08002			自动化	王平	C6	72

请根据语义分析关系模式 Student 中属性间的依赖关系。

① 学生的学号唯一。

$SNO \twoheadrightarrow SN, SNO \twoheadrightarrow AGE$

关系模式 : Student <U 、 F>

其中 : U={SNO, SN, AGE, DEPT, MN, CNO, SCORE}

**$F=\{SNO \twoheadrightarrow SN, SNO \twoheadrightarrow AGE, SNO \twoheadrightarrow DEPT,$
 $DEPT \twoheadrightarrow MN, (SNO,CNO) \twoheadrightarrow SCORE \}$**

关系数据库的规范化理论

- ① 给定一个关系模式（关系表），可能存在哪几个方面的问题？
- ② 衡量一个关系模式“好 / 坏”的依据是？
- ③ 什么原因导致数据冗余、插入异常、删除异常和更新异常？
- ④ 怎样将一个不好的关系模式转换为好的关系模式？





方法：通过模式分解，将“不好”的关系模式：

Student(SNO, SN, AGE, DEPT, MN, CNO, SCORE,
SNO, SN, AGE, DEPT, MN, CNO, SCORE) → DEPT, DEPT → MN

通过分解关系模式来消除其中不合适的数据依赖，
以解决插入异常、删除异常、更新异常、数据冗余
问题。

② D(DEPT, MN, DEPT → MN)

③ SC(SNO, CNO, SCORE, (SNO, CNO) → SCORE)

SNO	SN	AGE	DEPT
08001	李丽	17	计算机
08002	王鹏	18	自动化
08005	孙珊	19	自动化
08110	赵义	20	电信

DEPT	MN
计算机	刘伟
自动化	王平
电信	常乐

SNO	CNO	SCORE
08001	C1	85
08001	C2	88
08002	C2	67
08002	C4	50
08002	C5	77
08002	C6	72
08005	C5	90
08005	C6	89
08005	C7	95
08110	C1	75

关系数据库的规范化理论

- ① 给定一个关系模式（关系表），可能存在哪几个方面的问题？
- ② 衡量一个关系模式“好 / 坏”的依据是？
- ③ 什么原因导致数据冗余、插入异常、删除异常和更新异常？
- ④ 怎样将一个不好的关系模式转换为好的关系模式？
- ⑤ 对一个 “不好” 的关系模式进行不正确的分解又会导致什么问题？



不正确的模式分解导致的问题

$NL \bowtie DL$ 比 SL 关系多了 3 个元组，无法知道 95002、95004、95005 究竟是哪个系的学生。元组增加了、信息丢失了！——有损分解

SL		
Sno	Sdept	Sloc
195001	计算机	A
195002	信息	B
195003	数学	C
195004	信息	B
195005	物理	B

NL	
Sno	Sloc
195001	A
195002	B
195003	C
195004	B
195005	B

DL	
Sdept	Sloc
计算机	A
信息	B
数学	C
物理	B



$NL \bowtie DL$		
Sno	Sloc	Sdept
195001	A	计算机
195002	B	信息
195002	B	物理
195003	C	数学
195004	B	信息
195004	B	物理
195005	B	信息
195005	B	物理

关系数据库的规范化理论

- ① 给定一个关系模式（关系表），可能存在哪几个方面的问题？
- ② 衡量一个关系模式“好 / 坏”的依据是？
- ③ 什么原因导致数据冗余、插入异常、删除异常和更新异常？
- ④ 怎样将一个不好的关系模式转换为好的关系模式？
- ⑤ 对一个“不好”的关系模式进行不正确的分解又会导致什么问题？
- ⑥ 什么理论用来指导设计“好”的关系模式或对已有的关系模式进行模式分解？



- ✓ 用于：设计“好”的关系模式、对关系模式进行模式分解的有力工具：**关系数据库的规范化理论**。
- ✓ 最早由关系数据库的创始人 E.F.Codd 提出。
- ✓ 研究三方面内容：
 - ① 数据依赖（函数依赖、多值依赖）
 - ② 范式
 - ③ 模式设计（模式分解）

关系数据理论

01 | 关系数据库的规范化理论

02 | 函数依赖和码

03 | 范式

04 | 数据依赖的公理系统

05 | 模式分解的定义及算法

回顾：什么是数据依赖？有哪几类数据依赖？关系模式中的数据依赖可能会导致哪些问题？

① 什么是函数依赖？

② 什么是平凡与非平凡函数依赖？

③ 什么是完全函数依赖和部分函数依赖？

④ 什么是传递函数依赖？

⑤ 如何根据函数依赖集来确定关系模式中的码？

- 数据依赖是一个关系内部属性与属性间的约束关系。用来定义属性值间的相互关联，体现于属性值的相等与否
- 包括：函数依赖、多值依赖等。
- 关系模式中的数据依赖会导致数据冗余、插入异常、删除异常、修改异常

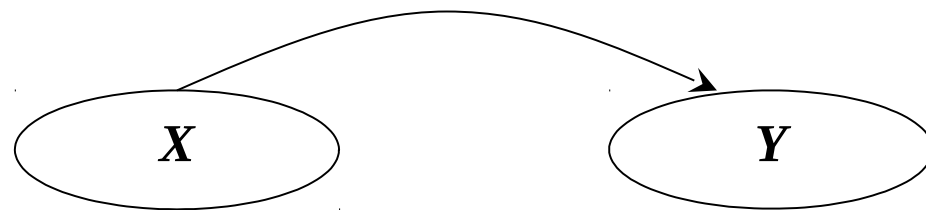


① 函数依赖

- $R(U)$ 是属性集 U 上的关系模式, X 和 Y 是 U 的子集.

若对于 $R(U)$ 任意一个关系 r , r 中不可能存在两个元组在 X 上的属性值相等, 而在 Y 上的属性值不等, 则称 X 函数确定 Y 或 Y 函数依赖于 X , 记作 $X \rightarrow Y$.

- X 称为决定因素.



$X \rightarrow Y$ 函数依赖图

分析下列关系模式中的函数依赖

A	B	C	D
a1	b1	c1	d1
a1	b1	c1	d2
a1	b2	c2	d1
a2	b2	c1	d1
a2	b2	c1	d3

该关系模式中，对于任意两个在 {A,B} 上取值相同的元组，它们在属性 C 上的取值也相同。

函数依赖： $AB \twoheadrightarrow C$

SNO (学号)	SN (姓名)	AGE (年龄)	DEPT (系别)	MN (系主任名)	CNO (课程号)	SCORE (成绩)
08001	李丽	17	计算机	刘伟	C1	85
08001	李丽	17	计算机	刘伟	C2	88
08002	王鹏	18	自动化	王平	C2	67
08002	王鹏	18	自动化	王平	C4	50
08002	王鹏	18	自动化	王平	C5	77
08002	王鹏	18	自动化	王平	C6	72
08005	孙珊	19	自动化	王平	C5	90
08005	孙珊	19	自动化	王平	C6	89
08005	孙珊	19	自动化	王平	C7	95
08110	赵义	20	电信	常乐	C1	75

SNO \square SN

SNO \square AGE

SNO \square DEPT

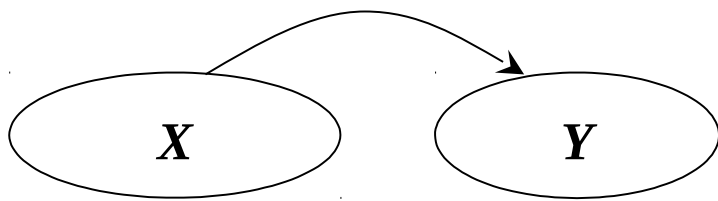
DEPT \square MN

(SNO, CNO) \square SCORE

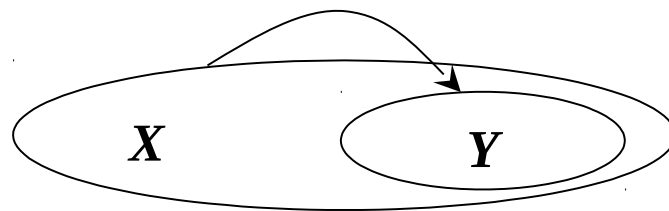
② 平凡函数依赖与非平凡函数依赖

在关系模式 $R(U)$ 中，对于 U 的子集 X 和 Y :

- ✓ 若 $X \rightarrow Y$, 但 $Y \not\subseteq X$, 称 $X \rightarrow Y$ 是非平凡的函数依赖
- ✓ 若 $X \rightarrow Y$, 但 $Y \subseteq X$, 称 $X \rightarrow Y$ 是平凡的函数依赖



非平凡函数依赖



平凡函数依赖

注意：一般不讨论平凡的函数依赖

分析：平凡函数依赖与非平凡函数依赖

SNO	CNO	SCORE
08001	C1	85
08001	C2	88
08002	C2	67
08002	C4	50
08002	C5	77
08002	C6	72
08005	C5	90
08005	C6	89
08005	C7	95
08110	C1	75

非平凡函数依赖： $(SNO, CNO) \rightarrow SCORE$

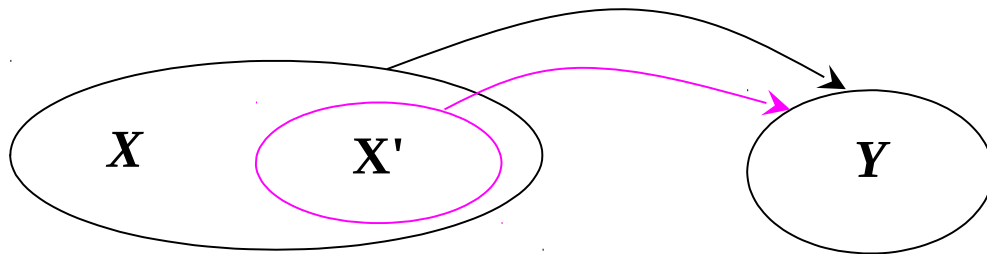
平凡函数依赖： $(SNO, CNO) \rightarrow SNO$

$(SNO, CNO) \rightarrow CNO$

③ 完全函数依赖与部分函数依赖

✓ 在 $R(U)$ 中，如果 $X \rightarrow Y$ ，并且对于 X 的任何一个真子集 X' ，~~都有~~ $X' \not\rightarrow Y$ ，则称 Y 对 X **完全函数依赖**，记作 $X \xrightarrow{F} Y$ 。

✓ $X \rightarrow Y$ ，若存在非空的 $X' \subset X$ ，使 $X' \rightarrow Y$ 成立，则称 Y 对 X **部分函数依赖**，记作 $X \xrightarrow{P} Y$ 。



部分函数依赖

分析：完全函数依赖与部分函数依赖

SNO	SN	AGE	DEPT	MN	CNO	SCORE
08001	李丽	17	计算机	刘伟	C1	85
08001	李丽	17	计算机	刘伟	C2	88
08002	王鹏	18	自动化	王平	C2	67
08002	王鹏	18	自动化	王平	C4	50
08002	王鹏	18	自动化	王平	C5	77
08002	王鹏	18	自动化	王平	C6	72
08005	孙珊	19	自动化	王平	C5	90
08005	孙珊	19	自动化	王平	C6	89
08005	孙珊	19	自动化	王平	C7	95
08110	赵义	20	电信	常乐	C1	75

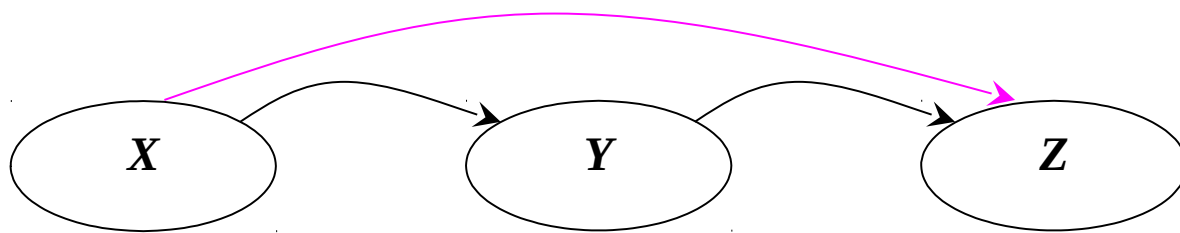
关系模式 Student 的函数依赖集为： $F=\{SNO \rightarrow SN, SNO \rightarrow AGE, SNO \rightarrow DEPT, DEPT \rightarrow MN, (SNO, CNO) \rightarrow SCORE\}$

$(SNO, CNO) \rightarrow SCORE$ 是完全函数依赖

$(SNO, CNO) \rightarrow DEPT$ 是部分函数依赖

④ 传递函数依赖

✓ 在 $R(U)$ 中, 如果 $X \rightarrow Y$ ($\emptyset \subsetneq Y \subsetneq X$), $Y \rightarrow Z$, 则称 Z 对 X **传递函数依赖**。记为: $X \overset{\text{传递}}{\rightarrow} Z$.



传递依赖 $X \rightarrow Z$ 的依赖图

分析：传递函数依赖

SNO	SN	AGE	DEPT	MN	CNO	SCORE
08001	李丽	17	计算机	刘伟	C1	85
08001	李丽	17	计算机	刘伟	C2	88
08002	王鹏	18	自动化	王平	C2	67
08002	王鹏	18	自动化	王平	C4	50
08002	王鹏	18	自动化	王平	C5	77
08002	王鹏	18	自动化	王平	C6	72
08005	孙珊	19	自动化	王平	C5	90
08005	孙珊	19	自动化	王平	C6	89
08005	孙珊	19	自动化	王平	C7	95
08110	赵义	20	电信	常乐	C1	75

由于： $SNO \rightarrow DEPT$ 、 $DEPT \rightarrow MN$
所以： MN 传递函数依赖于 SNO



例：关系模式： $R(\text{队员编号}, \text{比赛场次}, \text{进球数}, \text{球队名}, \text{队长号})$

记录了各个球队队员每场比赛进球数。若规定：每个队员只能属于一个球队、每个球队只有一个队长。

① 函数依赖： $(\text{队员编号}, \text{比赛场次}) \twoheadrightarrow \text{进球数}$

$\text{队员编号} \twoheadrightarrow \text{球队名}$

$\text{球队名} \twoheadrightarrow \text{队长号}$

② 存在传递函数依赖： $\text{队员编号} \twoheadrightarrow \text{队长号}$

③ 存在部分函数依赖： $(\text{队员编号}, \text{比赛场次}) \twoheadrightarrow \text{球队名}$

$(\text{队员编号}, \text{比赛场次}) \twoheadrightarrow \text{队长号}$

⑤ 码的概念

- ✓ 设 K 为 $R\langle U, F \rangle$ 中的属性或属性组合，若 $K \rightarrow U$ ，则 K 称为 R 的**候选码**。
- ✓ 若候选码多于一个，则选定其中一个做**主码**。
- ✓ **主属性**：包含在任何一个候选码中的属性。
- ✓ **非主属性（非码属性）**：不包含在任何码中的属性。
- ✓ 若整个属性组 U 是码，称为**全码**。
- ✓ 关系模式 R 中属性或属性组 X 并非 R 的码，但 X 是另一个关系模式的码，则称 X 是 R 的**外部码（外码）**。

主码与外码一起提供了表示关系间联系的手段。



例：关系模式： $R(\text{队员编号}, \text{比赛场次}, \text{进球数}, \text{球队名}, \text{队长号})$

记录了各个球队队员每场比赛进球数。若规定：每个队员只能属于一个球队，每个球队只有一个队长

$(\text{队员编号}, \text{比赛场次}) \twoheadrightarrow \text{队员编号}$

平凡函数依赖

$(\text{队员编号}, \text{比赛场次}) \twoheadrightarrow \text{比赛场次}$

平凡函数依赖

$(\text{队员编号}, \text{比赛场次}) \twoheadrightarrow \text{进球数}$

完全函数依赖

$(\text{队员编号}, \text{比赛场次}) \twoheadrightarrow \text{球队名}$

部分函数依赖

$(\text{队员编号}, \text{比赛场次}) \twoheadrightarrow \text{队长号}$

部分函数依赖

- ① 主码： $(\text{队员编号}, \text{比赛场次})$
- ② 主属性：队员编号、比赛场次
- ③ 非主属性：进球数、球队名、队长号

关系数据理论

01 | 关系数据库的规范化理论

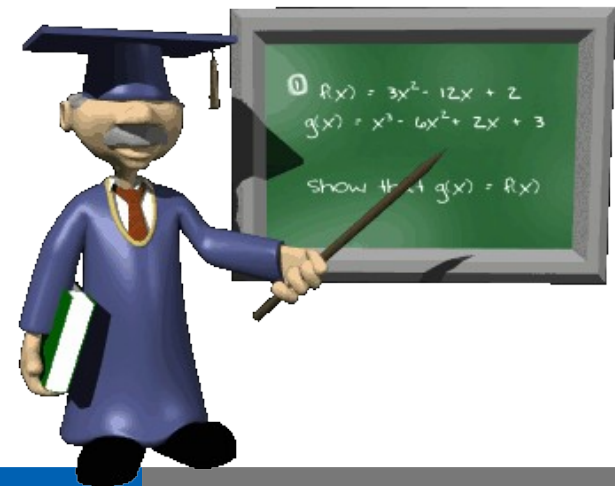
02 | 函数依赖和码

03 | 范式

04 | 数据依赖的公理系统

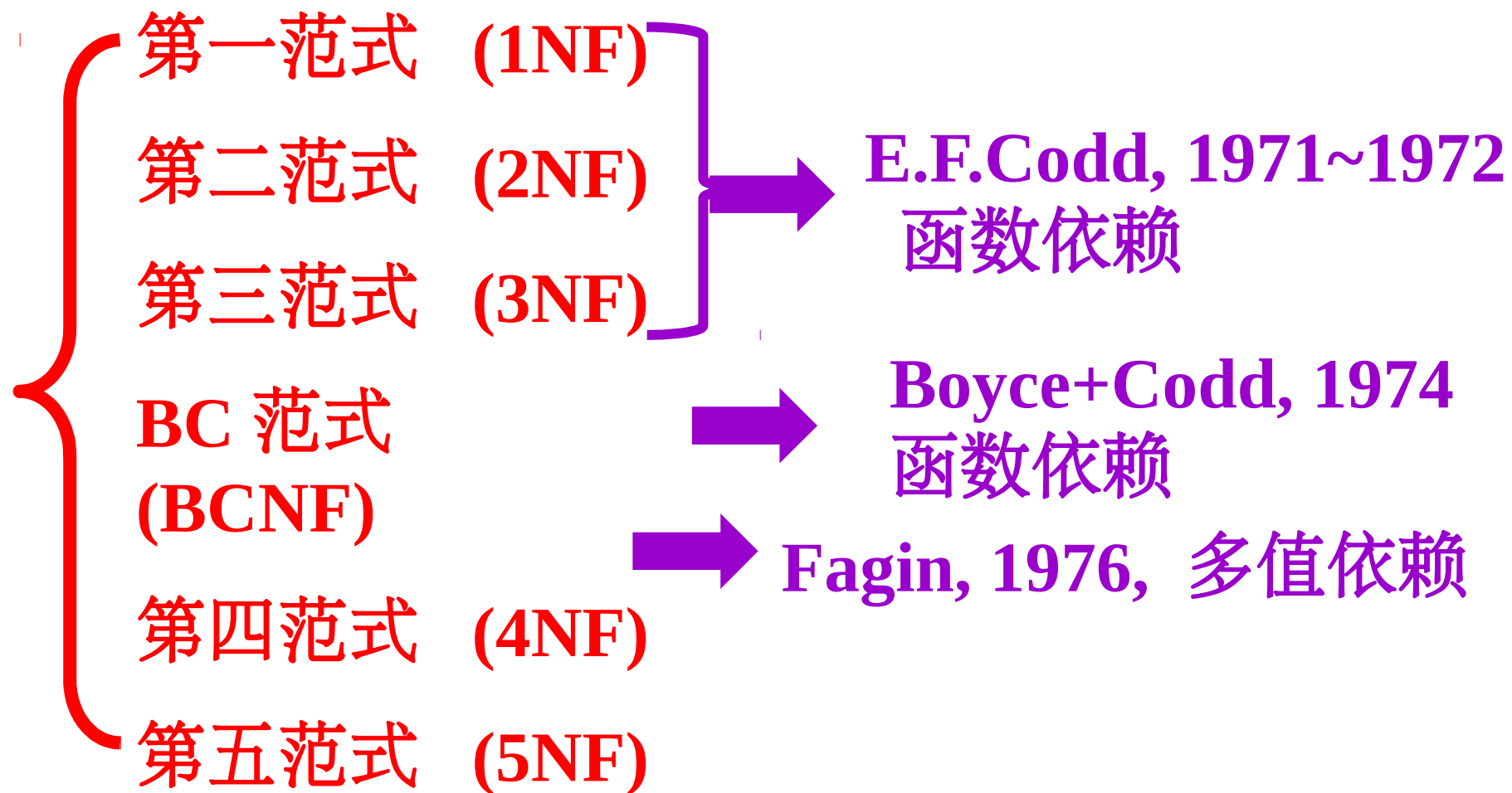
05 | 模式分解的定义及算法

Q1. 什么是范式？



什么是范式?

✓ **范式**：符合某一级别要求的**关系模式集合**。



什么是范式?

- ✓ 各种范式之间存在联系:

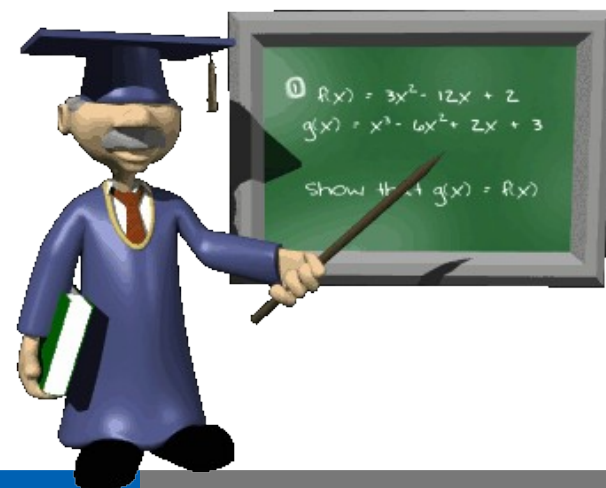
$$1NF \supset 2NF \supset 3NF \supset BCNF \supset 4NF \supset 5NF$$

- ✓ 某关系模式 R 为第 n 范式, 简记为 $R \in nNF$.
- ✓ 一个低一级范式的关系模式, 通过模式分解可以转换为若干个高一级范式的关系模式集合, 这种过程就叫规范化.

范式

Q1. 什么是范式？

Q2. 属于 1NF 的关系模式满足什么要求？存在插入 / 删除 / 修改异常问题吗？



- ✓ 若一个关系模式 R 的所有属性都是不可分的基本数据项，则 $R \in 1NF$.
- ✓ 第一范式是对关系模式的最起码要求，不满足第一范式的数据库模式不能称为关系数据库。

不属于 1NF 的关系模式

部门号	部门名	部门员工			部门经理
		员工 1	员工 2	员工 3	
011	财务部	张强	李丽	王平	石元
012	人事部	董竹	杨聪		赵丽
013	计划部	肖晓	孙立		陈东

属于 1NF 的关系模式

关系模式 S-L-C(Sno, Cno, Grade, Sdept, Sloc)

分别表示：学号、课程号、成绩、系别、宿舍。假设每个系的学生住在同一个地方。

Sno	Cno	Grade	Sdept	SLoc
95001	1	80	计算机系	18号公寓
95001	2	90	计算机系	18号公寓
95003	3	80	化学系	12号公寓
95004	1	91	化学系	12号公寓

问题：属于 1NF 的关系模式一定是好的关系模式吗 ???

插入异常

- ✓ 若一个学生还未选课 (无 Cno), 则学生的固有信息无法插入。
- ✓ 原因: 主属性 Cno 的值不能为空。

Sno	Cno	Grade	Sdept	SLoc
95001	1	80	计算机系	18 号公寓
95001	2	90	计算机系	18 号公寓
				寓
95008			数学系	8 号公寓

删除异常

若某个学生只选修了一门课，现不选修这门课了，当删除选课信息时，学生基本信息也删除。

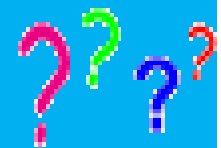
Sno	Cno	Grade	Sdept	Sloc
95001	1	80	计算机系	18 号公寓
95002	1	80	计算机系	18 号公寓
95003	3		化学系	12 号公寓
95004	1	91	化学系	12 号公寓

修改复杂

某个学生从化学系转到计算机系，同时住所改变，若该生选修了 K 门课，则要修改 K 个元组的全部 Sdept、SLoc 信息，修改复杂。

Sno	Cno	Grade	Sdept	SLoc
95001	1	80	计算机系	18 号公寓
95001	2	90	计算机系	18 号公寓
95003	3	80	计算机系	18 号公寓
95003	1	91	计算机系	18 号公寓

什么原因导致插入、删除、修改异常情况？怎样解决？





分析：关系模式 S-L-C(Sno, Cno, Grade, Sdept, Sloc)

(1) 函数依赖集：

$Sno \rightarrow Sdept$

$Sdept \rightarrow Sloc$

$(Sno, Cno) \twoheadrightarrow Grade$

(2) 主码：

(Sno, Cno)

(3) 非主属性：

$Grade$ 、 $Sdept$ 、 $Sloc$

(4) 产生问题的原因：

非主属性 $Sdept$ 、 $Sloc$ 部分函数依赖于码

(5) 解决办法：

模式分解，消除非主属性对码的部分函数依赖

Sno	Cno	Grade	Sdept	SLoc
95001	1	80	计算机系	18 号公寓
95001	2	90	计算机系	18 号公寓
95003	3	80	化学系	12 号公寓
95004	1	91	化学系	12 号公寓
95006	4	92	机械系	12 号公寓
95008	1	99	数学系	8 号公寓

Sno	Cno	Grade	Sdept	SLoc
95001	1	80	计算机系	18号公寓
95001	2	90	计算机系	18号公寓
95003	3	80	化学系	12号公寓
95004	1	91	化学系	12号公寓
95006	4	92	机械系	12号公寓
95008	1	99	数学系	8号公寓

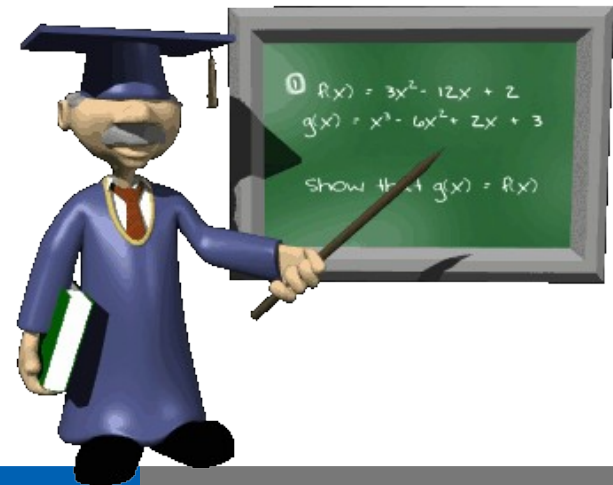
S	C	G
95001	2	90
95003	3	80
95004	1	91
95006	4	92
95008	1	99

95001	计算机系	18号公寓
95003	化学系	12号公寓
95004	化学系	12号公寓
95006	机械系	12号公寓

Q1. 什么是范式？

Q2. 属于 1NF 的关系模式满足什么要求？存在插入 / 删除 / 修改异常问题吗？

Q3. 属于 2NF 的关系模式满足什么要求？还存在插入 / 删除 / 修改异常问题吗？



若关系模式 $R \in 1NF$, 且 R 中每个非主属性都完全依赖于码, 则 $R \in 2NF$.

- ✓采用：投影分解法 将一个 1NF 的关系分解为多个 2NF 关系，消除非主属性对码的部分函数依赖，可在一定程度上减轻原 1NF 关系中存在的插入异常、删除异常、数据冗余度大、修改复杂等问题。



分析下列关系模式是否属于 2NF？如果不属于 2NF，请分解为满足 2NF 的关系模式集。

工程号 Pno	材料号 Ino	数量 C	开工日期 D1	完工日期 D2	价格 Price
P1	I1	4	9805	9911	250
P1	I2	6	9805	9911	210
P1	I3	15	9805	9911	300
P2	I1	6	9811	9912	250
P2	I4	18	9811	9912	380

- ✓ $R \in 1NF$ ， $F=\{Pno \rightarrow D1, Pno \rightarrow D2, (Pno, Ino) \rightarrow C, Ino \rightarrow Price\}$
- ✓ 主码：(Pno, Ino)
- ✓ 非主属性：C、D1、D2、Price
- ✓ 存在非主属性：开工日期 D1、完工日期 D2、价格 Price 对码的部分函数依赖，该关系模式不属于 2NF.



工程号 Pno	材料号 Ino	数量 C	开工日期 D1	完工日期 D2	价格 Price
P1	I1	4	9805	9911	250
P1	I2	6	9805	9911	210
P1	I3	15	9805	9911	300
P2	I1	6	9811	9912	250
P2	I4	18	9811	9912	380

$R1 = \{Pno, Ino, C, (Pno, Ino) \twoheadrightarrow C\}$

工程号 Pno	材料号 Ino	数量 C
P1	I1	4
P1	I2	6
P1	I3	15
P2	I1	6
P2	I4	18

$R2 = \{Pno, D1, D2, Pno \twoheadrightarrow D1, Pno \twoheadrightarrow D2\}$

工程号 Pno	开工日期 D1	完工日期 D2
P1	9805	9911
P2	9811	9912

$R3 = \{Ino, Price, Ino \twoheadrightarrow Price\}$

材料号 Ino	价格 Price
I1	250
I2	210
I3	300
I4	380

分析 2NF 关系模式: SL (Sno , Sdept , Sloc) 是否还存在异常?

① 函数依赖集:

$Sno \rightarrow Sdept$

$Sdept \rightarrow Sloc$

② 主码:

Sno

Sno	Sdept	Sloc
95001	计算机系	18 号公寓
95003	化学系	12 号公寓
95004	化学系	12 号公寓
95006	机械系	12 号公寓
95008	数学系	8 号公寓

插入异常

若数学系尚未招生，尽管系住所已确定，但信息无法保存。

Sno	Sdept	Sloc
95001	计算机系	18 号公寓
95003	化学系	12 号公寓
95004	化学系	12 号公寓
95006	机械系	10 号公寓

删除异常

若化学系仅 2 名学生，但均被退学需删除信息，则会将系名及系地址均删除掉。

SNO	SDEPT	SLOC
95001	计算机系	18 号公寓
95004	化学系	12 号公寓
95006	机械系	10 号公寓

存在问题 3

修改复杂

若计算机系有 K 名学生，若系住址 Sloc 发生变化，则有 K 条记录要全部变化，修改复杂。

结论：将一个 1NF 关系分解为多个 2NF 关系，并不能完全消除关系模式中的各种异常和数据冗余。

Sno	Sdept	Sloc
95001	计算机系	16 号公寓
95003	计算机系	16 号公寓
95004	计算机系	16 号公寓
95006	机械系	10 号公寓
95008	数学系	8 号公寓

什么原因导致异常情况？怎样解决？



原因

2NF 关系模式: SL (Sno , Sdept , Sloc) 中的函数依赖:

$Sno \rightarrow Sdept$ ($Sdept \rightarrow \cancel{Sno}$)

$Sdept \rightarrow Sloc$

可得: $Sno \xrightarrow{\text{传递}} Sloc$

存在非主属性 Sloc 对码 Sno 的传递函数依赖

Sno	Sdept	Sloc
95001	计算机系	18 号公寓
95003	计算机系	18 号公寓
95004	计算机系	18 号公寓
95006	机械系	10 号公寓
95008	数学系	8 号公寓

解决办法

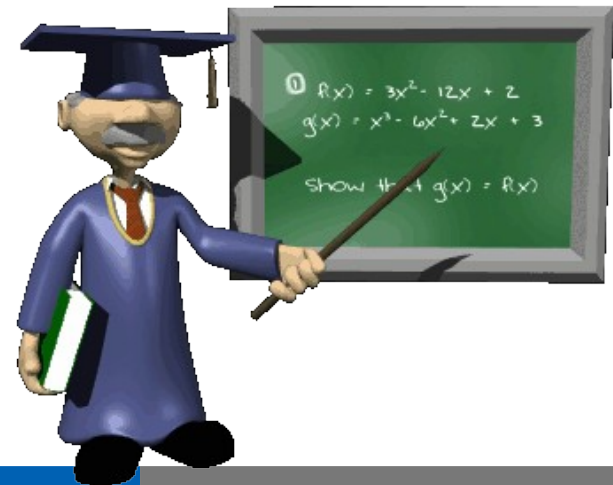
采用投影分解法：消除非主属性对码的传递函数依赖，将 2NF 关系模式分解为 3NF 关系模式集

Sno	Sdept	Sloc
95001	计算机系	18 号公寓
95003	计算机系	18 号公寓
95004	计算机系	18 号公寓
95006	机械系	10 号公寓
95008	数学系	8 号公寓

Sno	Sdept
95001	计算机系
95003	计算机系
95004	计算机系
95006	机械系
95008	数学系

Sdept	Sloc
计算机系	18 号公寓
机械系	10 号公寓
数学系	8 号公寓

- Q1. 什么是范式？
- Q2. 属于 1NF 的关系模式满足什么要求？存在插入 / 删除 / 修改异常问题吗？
- Q3. 属于 2NF 的关系模式满足什么要求？还存在插入 / 删除 / 修改异常问题吗？
- Q4. 属于 3NF 的关系模式满足什么要求？满足 3NF 的关系模式能完全消除各种异常情况吗？



关系模式 $R\langle U, F \rangle$ 中若不存在这样的码 X 、属性组 Y 及非主属性 Z ($Z \subseteq Y$)，使得 $X \rightarrow Y$ 、 $Y \rightarrow Z$ 成立， $Y \rightarrow X$ ，则称 $R\langle U, F \rangle \in 3NF$ 。

结论：若 $R \in 3NF$ ，则每一个非主属性既不部分依赖于码、也不传递依赖于码。

- ✓ 采用投影分解法将一个 2NF 的关系分解为多个 3NF 的关系，可以在一定程度上解决原 2NF 关系中存在的插入异常、删除异常、数据冗余度大、修改复杂等问题。
- ✓ 3NF 关系模式，仍然不能完全消除关系模式中的各种异常情况 and 数据冗余。



关系模式： $STJ (S, T, J)$

每一教师 T 只教一门课 J 、每门课由若干教师教；
某学生 S 选定某门课就确定了一个固定的教师；
某学生选修某教师的课就确定了所选课名称。

S	T	J
95001	陈兰	高数
95001	张松	逻辑
95003	薛梅	高数
95004	薛梅	高数
95004	王竹	物理
95004	张松	逻辑
95006	王竹	物理

① 函数依赖：

$T \rightarrow J$

$(S, J) \rightarrow T$

$(S, T) \rightarrow J$

② 候选码：

(S, J) 、 (S, T)

③ 主属性：

S 、 J 、 T

但 STJ 仍然存在问题！

关系模式 STJ 中不存在非主属性！
即：不存在非主属性对码的部分函数依赖和传递函数依赖。

$STJ \in 3NF$

插入异常

若某老师的课还没安排学生选课，则该老师及相应课信息无法插入数据库

S	T	J
95001	张松	逻辑
95003	薛梅	高数
95004	薛梅	高数
95004	王竹	物理
95004	张松	逻辑
95006	王竹	物理

删除异常

删除学生选课记录时，老师及相应课程信息也随之删除。

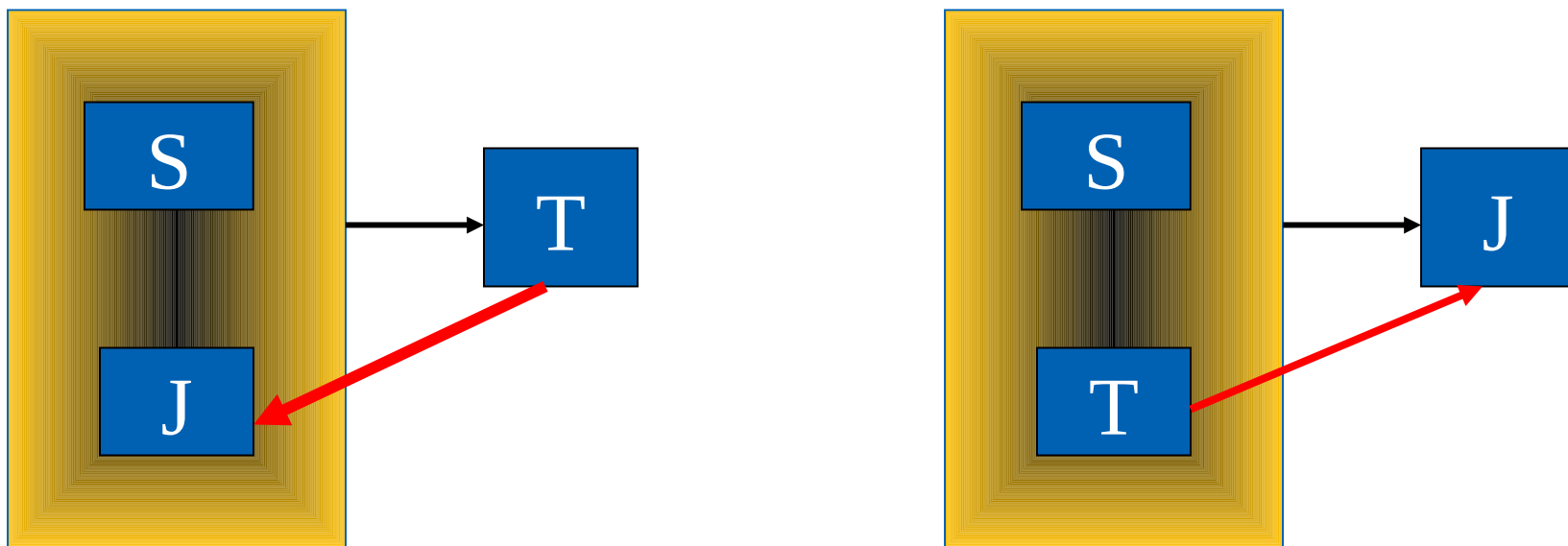
S	T	J
95001	陈兰	高数
95001	张松	逻辑
95004	王竹	物理
95004	张松	逻辑
95006	王竹	物理

不合理的冗余

某老师的课有许多学生选修，课程名存储多次。

S	T	J
95001	陈兰	高数
95002	陈兰	高数
95003	陈兰	高数
95004	陈兰	高数
95005	陈兰	高数
95006	陈兰	高数
95006	张松	逻辑

为何出现上述情况？如何改进？



主属性 J 传递依赖于候选码 (S,J)

主属性 J 部分依赖于候选码 (S,T)

原因：函数依赖 $T \twoheadrightarrow J$ 中决定因素 T 不包含码

解决办法：将 3NF 的关系模式分解为 BCNF 的关系模式集，消除主属性对码的部分函数依赖。

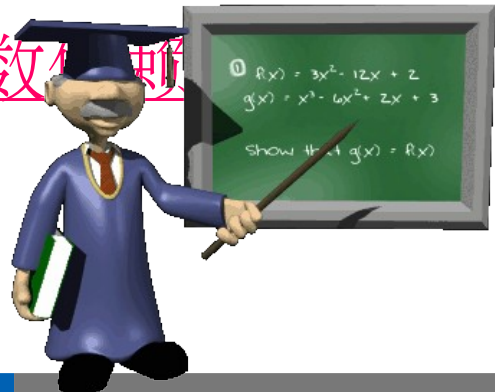


S	T	J
95001	陈兰	高数
95001	张松	逻辑
95003	薛梅	高数
95004	薛梅	高数
95004	王竹	物理
95004	张松	逻辑
95006	王竹	物理

S	T
95001	陈兰
95001	张松
95003	薛梅
95004	薛梅
95004	王竹
95004	张松
95006	王竹

T	J
陈兰	高数
张松	逻辑
薛梅	高数
王竹	物理

- Q1. 什么是范式？
- Q2. 属于 1NF 的关系模式满足什么要求？存在插入 / 删除 / 修改异常问题吗？
- Q3. 属于 2NF 的关系模式满足什么要求？存在插入 / 删除 / 修改异常问题吗？
- Q4. 属于 3NF 的关系模式满足什么要求？满足 3NF 的关系模式能完全消除各种异常情况吗？
- Q5. 属于 BCNF 的关系模式满足什么要求？如果只考虑函数依赖，则模式规范化程度最高可达到哪种范式？



关系模式 $R\langle U, F \rangle \in 1NF$, 若 $X \rightarrow Y$ 且 $\nexists X' \subseteq X$ 时 X 必含有码, 则 $R\langle U, F \rangle \in BCNF$ 。

结论：若关系模式 $R \in 1NF$ ，且 R 中每个决定因素都包含码，则 $R \in BCNF$ 。



分析：关系模式 $R (A, B, C, AB \twoheadrightarrow C, C \twoheadrightarrow A)$ 属于 3NF 吗？
属于 BCNF 吗？

✓ 候选码： AB 、 BC

✓ 主属性： A 、 B 、 C

✓ S 中不存在非主属性对码的部分函数依赖和传递函数依赖，所以 $S \in 3NF$.

✓ 因为 $C \twoheadrightarrow A$ 的决定因素不包含码，所以 S 不属于 BCNF.

分析：关系模式 $R(A, B, C, AB \twoheadrightarrow C, BC \twoheadrightarrow A)$ 属于 3NF 、
BCNF 吗？

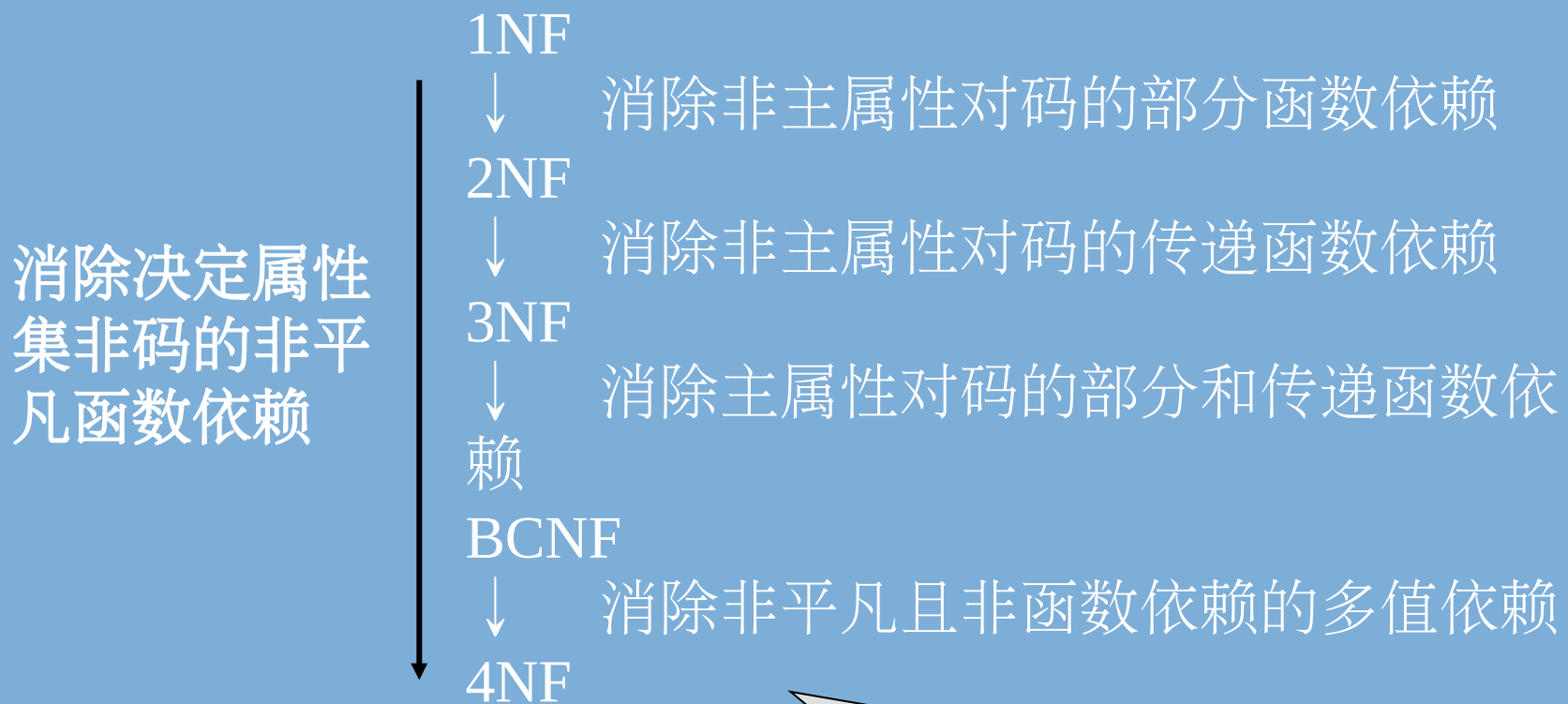
- ✓ 候选码：AB 、 BC
- ✓ 主属性：A 、 B 、 C
- ✓ S 中不存在非主属性对码的部分函数依赖和传递函数依赖，所以 $S \in 3NF$.
- ✓ S 中所有函数依赖的决定因素都包含码，所以 $S \in BCNF$.

1. 若关系 R 的候选码都是由单属性构成的, 那么 R 必定属于 2NF。
2. 在关系模式 R 中, 若没有非主属性, 则 R 必定属于 3NF。
3. 若关系 R 是一个全码关系, 则 R 必定属于 BCNF。
— **BCNF**
4. 任何一个二元关系必定属于 _____。

规范化小结

- ✓ 关系数据库规范化理论是数据库逻辑设计的工具。
- ✓ 目的：尽量消除插入异常、删除异常、修改复杂、数据冗余。
- ✓ 方法：逐步消除数据依赖中不合适的部分。
- ✓ 如果只考虑函数依赖，则关系模式最高规范化程度是 BCNF.
- ✓ 如果考虑多值依赖，则关系模式最高规范化程度为 4NF.

规范化小结



注意：不能说规范化程度越高的关系模式就越好。

在设计数据库模式结构时，必须对现实世界的实际情况和用户应用需求作进一步分析，确定一个合适的、能够反映现实世界的模式。

关系数据理论

01 | 关系数据库的规范化理论

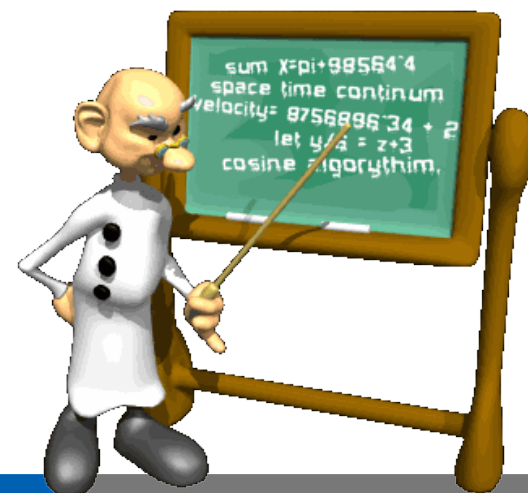
02 | 函数依赖和码

03 | 范式

04 | 数据依赖的公理系统

05 | 模式分解的定义及算法

Q1. 给定关系模式 R 及其函数依赖集 F ，是否还存在其他的函数依赖成立？



逻辑蕴涵

例： 给定 $R = (A, B, C)$, $F = \{A \twoheadrightarrow B, B \twoheadrightarrow C\}$, 则在 R 中必然存在函数依赖：
 $A \twoheadrightarrow C$

证明： 对于关系 r 中任意两个元组 t 和 s ：

$A \twoheadrightarrow B$ ： 若 $t[A] = s[A]$ ，则 $t[B] = s[B]$

$B \twoheadrightarrow C$ ： 若 $t[B] = s[B]$ ，则 $t[C] = s[C]$

故： 若 $t[A] = s[A]$ ，则 $t[C] = s[C]$

即： $A \twoheadrightarrow C$

称为： F 逻辑蕴涵 $A \twoheadrightarrow C$

对于满足函数依赖集 F 的关系模式 $R \langle U, F \rangle$ ，其任何一个关系 r ，若函数依赖 $X \rightarrow Y$ 都成立（即 r 中任意两个元组 t 、 s ，若 $t[X] = s[X]$ ，则 $t[Y] = s[Y]$ ），
则称： F 逻辑蕴含 $X \rightarrow Y$ 。

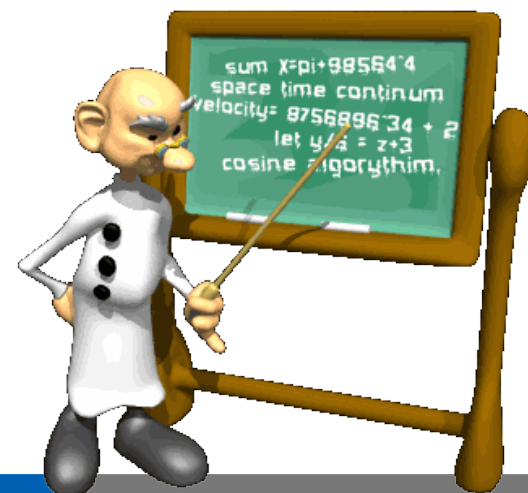
例：

R (队员编号, 比赛场次, 进球数, 球队名, 队长号)

F={ 队员编号 \rightarrow 球队名, 球队名 \rightarrow 队长号, (队员编号, 比赛场次) \rightarrow 进球数 }.

怎样从函数依赖集 **F** 求得 **F** 逻辑蕴涵的函数依赖?

- Q1. 给定关系模式 R 及其函数依赖集 F ，是否还存在其他的函数依赖成立？
- Q2. 怎样从函数依赖集 F 求得 F 蕴涵的函数依赖？有什么推理规则？



Armstrong 公理系统

如： $(\text{队员编号}, \text{比赛场次}) \twoheadrightarrow \text{队员编号}$
 $(\text{队员编号}, \text{比赛场次}) \twoheadrightarrow \text{比赛场次}$
是 F 所蕴含的。

对关系模式 $R \langle U, F \rangle$

A1. 自反律

若 $Y \subseteq X$

为了求出给定关系模式的码、以及从函数依赖集求得蕴涵的函数依赖，需要 Armstrong 推理规则

(1974 年 Armstrong 提出)

A2. 增广律

若 $X \rightarrow Y$ 为 F 所

如： F 中有 $\text{队员编号} \twoheadrightarrow \text{球队名}$ ， $\text{球队名} \twoheadrightarrow \text{队长号}$ ，
则

$\text{队员编号} \twoheadrightarrow \text{队长号}$

是 F 所蕴含的。

A3. 传递律 (Transitivity) :

若 $X \rightarrow Y$ 及 $Y \rightarrow Z$ 为 F 所蕴含，则 $X \rightarrow Z$ 为 F 所蕴含。

Armstrong 公理系统的推论

根据自反律、增广律、传递律规则可以得到：

① 合并规则：由 $X \rightarrow Y$ ， $X \rightarrow Z$ ，有 $X \rightarrow YZ$ 。

已知 $X \rightarrow Z$ ，由增广律知 $XY \rightarrow YZ$ 。又因为 $X \rightarrow Y$ ，可得 $XX \rightarrow XY \rightarrow YZ$ ，故根据传递律得 $X \rightarrow YZ$ 。

② 伪传递规则：由 $X \rightarrow Y$ ， $WY \rightarrow Z$ ，有 $XW \rightarrow Z$ 。

已知 $X \rightarrow Y$ ，根据增广律得 $XW \rightarrow WY$ 。因为 $WY \rightarrow Z$ ，所以 $XW \rightarrow WY \rightarrow Z$ ，通过传递律可知 $XW \rightarrow Z$ 。

③ 分解规则：由 $X \rightarrow Y$ 及 $Z \subseteq Y$ ，有 $X \rightarrow Z$ 。

已知 $Z \subseteq Y$ ，根据自反律知 $Y \rightarrow Z$ ，又因为 $X \rightarrow Y$ ，所以由传递律可得 $X \rightarrow Z$ 。

引理： $X \rightarrow A_1 A_2 \dots A_k$ 成立的充分必要条件是 $X \rightarrow A_i$ 成立 ($i=1,2,\dots,k$)。

例：

R (队员编号, 比赛场次, 进球数, 球队名, 队长号)

F={ 队员编号 \square 球队名, 球队名 \square 队长号, (队员编号, 比赛场次) \square 进球

(队员编号, 比赛场次) \square 队员编号

自反律

(队员编号, 比赛场次) \square 比赛场次

自反

律

(队员编号, 比赛场次) \square (球队名, 比赛场次) 增广律

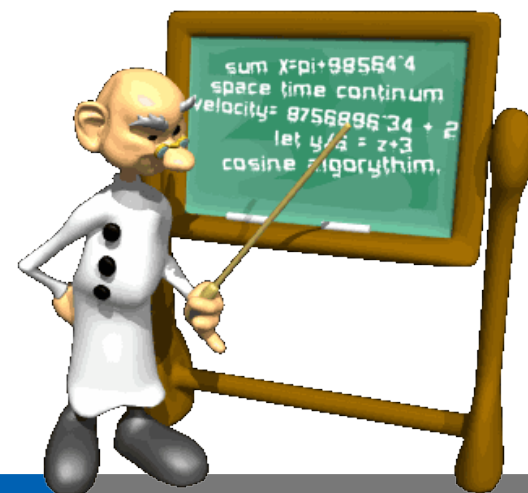
队员编号 \square 队长号

传递律

.....

请思考：能否求出 F 所蕴涵的全体函数依赖？

- Q1. 给定关系模式 R 及其函数依赖集 F ，是否还存在其他的函数依赖成立？
- Q2. 怎样从函数依赖集 F 求得 F 蕴涵的函数依赖？有什么推理规则？
- Q3. F 所蕴涵全体函数依赖叫作什么？能否求出？



在关系模式 $R\langle U, F \rangle$ 中为 F 所逻辑蕴含的全体函数依赖叫作 F 的闭包，记为 F^+

解决办法：不需要计算 F^+ ，如果判断一个函数依赖是否在 F^+ 中，只需通过属性集闭包进行判断。

例：若 $R(X, Y, Z)$, $F=\{X \twoheadrightarrow Y, Y \twoheadrightarrow Z\}$ ，求 F^+ 。

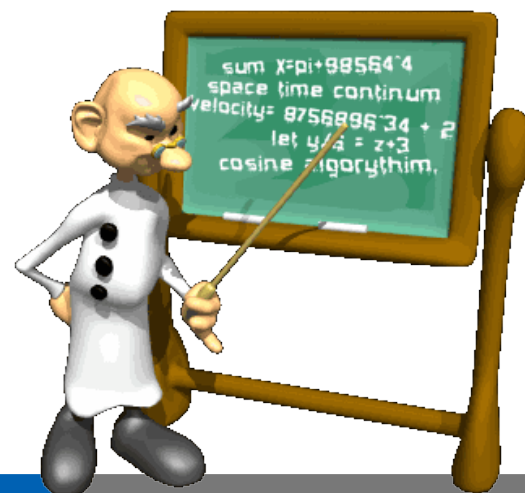
$F^+=\{$ $X \twoheadrightarrow \varnothing, Y \twoheadrightarrow \varnothing, Z \twoheadrightarrow \varnothing, XY \twoheadrightarrow \varnothing, XZ \twoheadrightarrow \varnothing, YZ \twoheadrightarrow \varnothing, XYZ \twoheadrightarrow \varnothing,$
 $X \twoheadrightarrow X, Y \twoheadrightarrow Y, Z \twoheadrightarrow Z, XY \twoheadrightarrow X, XZ \twoheadrightarrow X, YZ \twoheadrightarrow Y, XYZ \twoheadrightarrow X,$
 $X \twoheadrightarrow Y, Y \twoheadrightarrow Z, XY \twoheadrightarrow Y, XZ \twoheadrightarrow Y, YZ \twoheadrightarrow Z, XYZ \twoheadrightarrow Y,$
 $X \twoheadrightarrow Z, Y \twoheadrightarrow YZ, XY \twoheadrightarrow Z, XZ \twoheadrightarrow Z, YZ \twoheadrightarrow YZ, XYZ \twoheadrightarrow Z,$
 $X \twoheadrightarrow XY, XY \twoheadrightarrow XY, XZ \twoheadrightarrow XY, XYZ \twoheadrightarrow XY,$
 $X \twoheadrightarrow XZ, XY \twoheadrightarrow YZ, XZ \twoheadrightarrow XZ, XYZ \twoheadrightarrow YZ,$
 $X \twoheadrightarrow YZ, XY \twoheadrightarrow XZ, XZ \twoheadrightarrow XY, XYZ \twoheadrightarrow XZ,$
 $X \twoheadrightarrow ZYZ, XY \twoheadrightarrow XYZ, XZ \twoheadrightarrow XYZ, XYZ \twoheadrightarrow XYZ$
 $\}$

已知函数依赖集 F ，计算 F^+ 是一个 NP 完全问题！怎么办？



数据依赖的公理系统

- Q1. 给定关系模式 R 及其函数依赖集 F ，是否还存在其他的函数依赖成立？
- Q2. 怎样从函数依赖集 F 求得 F 蕴涵的函数依赖？有什么推理规则？
- Q3. F 所蕴涵全体函数依赖叫作什么？能否求出？
- Q4. 什么是属性集闭包？怎样求属性集闭包？



属性集闭包 X_F^+

已知关系模式 R 的属性集 U , F 是函数依赖集, X 是 U 的子集。则所有用 Armstrong 公理从 F 中导出的函数依赖集 $X \rightarrow A_i$ 中 A_i 的属性集合称为 **X 的属性闭包**, 记为 X_F^+

$$X_F^+ = \{A_i \mid A_i \in U, X \rightarrow A_i \in F^+\}$$

求属性集闭包 X_F^+ 的算法

算法：求属性集 X 关于函数依赖集 F 的闭包 X_F^+

输入： X , F

输出： X_F^+

步骤：

(1) 令 $X^{(0)} = X$, $i=0$

(2) 求 B , 这里 $B = \{ A \mid (\exists V)(\exists W)(V \rightarrow W \in F \wedge V \subseteq X^{(i)} \wedge A \in W) \}$

(3) $X^{(i+1)} = B \cup X^{(i)}$

(4) 判断 $X^{(i+1)} = X^{(i)}$ 吗？

(5) 若相等，或 $X^{(i)} = U$, 则 $X^{(i)}$ 就是 X_F^+ , 算法终止

(6) 若否，则 $i=i+1$, 返回第 (2) 步



例：已知关系模式 $R\langle U, F \rangle$ ，其中： $U=\{A, B, C, D, E\}$ ，

$F=\{AB \rightarrow C, B \rightarrow D, C \rightarrow E, EC \rightarrow B, AC \rightarrow B\}$ 。求 $(AB)^+_F$ 。

① $X^{(0)}=AB$

② 计算 $X^{(1)}$ ：

逐一的扫描 F 集合中各个函数依赖，找左部为 A 、 B 、 AB 的函数依赖，得： $AB \rightarrow C$ 、 $B \rightarrow D$ ， $X^{(1)} = X^{(0)} \cup CD = ABCD$

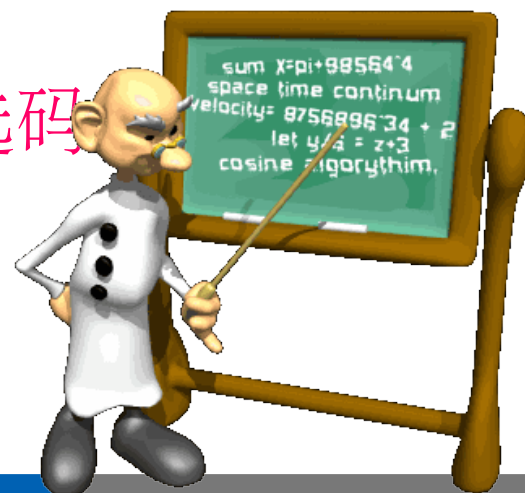
③ $X^{(1)} \neq X^{(0)}$ ，计算 $X^{(2)}$ ：

查找左部为 $ABCD$ 子集的那些函数依赖，得： $AB \rightarrow C$ 、 $B \rightarrow D$ 、 $C \rightarrow E$ 、 $AC \rightarrow B$ ， $X^{(2)} = X^{(1)} \cup CDEB = ABCDE$

④ $X^{(2)}=U$ ，算法终止

数据依赖的公理系统

- Q1. 给定关系模式 R 及其函数依赖集 F ，是否还存在其他的函数依赖成立？
- Q2. 怎样从函数依赖集 F 求得 F 蕴涵的函数依赖？有什么推理规则？
- Q3. F 所蕴涵全体函数依赖叫作什么？能否求出？
- Q4. 什么是属性集闭包？怎样求属性集闭包？
- Q5. 如何根据属性集闭包判断一个属性集 X 是否是候选码？



给定关系模式 $R(U, F)$, $X \subseteq U$, X 是 R 候选码的条件 :

① $X_F^+ = U$

② X 的所有子集的闭包不等于 U 。即对于任意的 $Y \subseteq X$,
不存在 $Y_F^+ = U$



例：已知关系模式 $R\langle U, F\rangle$ ，其中

$U=\{A, B, C, G, H, I\}$ ；

$F=\{A \twoheadrightarrow B, A \twoheadrightarrow C, CG \twoheadrightarrow H, CG \twoheadrightarrow I, B \twoheadrightarrow H\}$ 。

判断 AG 是否是 R 的候选码？

解： (1) $(AG)_F^+ = ABCGHI$

(2) 对于 AG 的子集： $A_F^+ = ABCH$

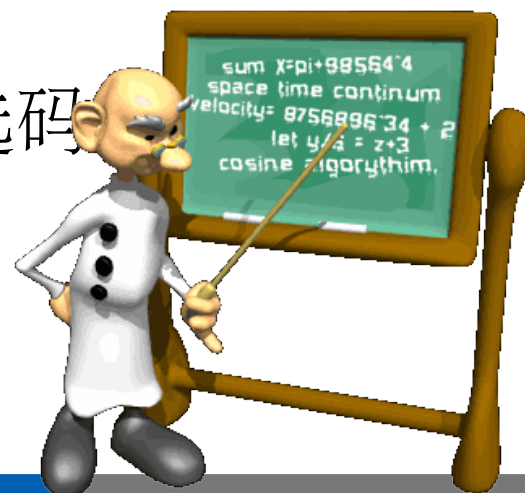
$$G_F^+ = G$$

所以， AG 是关系模式 R 的候选码。

难点：怎样找出关系模式 R 的候选码？候选码是否是唯一？

数据依赖的公理系统

- Q1. 给定关系模式 R 及其函数依赖集 F ，是否还存在其他的函数依赖成立？
- Q2. 怎样从函数依赖集 F 求得 F 蕴涵的函数依赖？有什么推理规则？
- Q3. F 所蕴涵全体函数依赖叫作什么？能否求出？
- Q4. 什么是属性集闭包？怎样求属性集闭包？
- Q5. 如何根据属性集闭包判断一个属性集 X 是否是候选码？
- Q6. 求关系模式 R 的所有候选码？



给定关系 $R(U, F)$ ，其中 $U = \{A_1, A_2, \dots, A_n\}$ ， F 是 R 的函数依赖集。

可将属性分为四类：

- ①L: 仅出现在函数依赖集 F 左部的属性
- ②R: 仅出现在函数依赖集 F 右部的属性
- ③LR: 在函数依赖集 F 左右两边都出现的属性
- ④NLR: 在函数依赖集 F 左右两边都不出现的属性

候选码的求解方法

定理 1 给定关系模式 $R(U, F)$ ，若 $X(X \subseteq U)$ 是 **L** 类属性，则 **X 必为 R 的任一候选码的成员。**

推理 1 给定关系模式 $R(U, F)$ ，若 $X(X \subseteq U)$ 是 **L** 类属性，且 **$X_F^+ = U$** ，则 **X 必为 R 的唯一候选码。**

定理 2 给定关系模式 $R(U, F)$ ，若 $X(X \subseteq U)$ 是 **R** 类属性，则 **X 必不在 R 的任何候选码中。**

定理 3 给定关系模式 $R(U, F)$ ，若 $X(X \subseteq U)$ 是 **NLR** 类属性，则 **X 必为 R 的任一候选码的成员。**

推理 2 给定关系模式 $R(U, F)$ ，若 $X(X \subseteq U)$ 是 **L** 类和 **NLR** 类属性所组成的属性集，且 **$X_F^+ = U$** ，则 **X 必为 R 的唯一候选码。**



例 1：关系模式 $R(HIJKLM)$ ，函数依赖集 $F=\{H \rightarrow I, K \rightarrow I, LM \rightarrow K, I \rightarrow K, KH \rightarrow M\}$ ，请求出 R 的所有候选码。

L 属性：H、L

NLR 属性：J

根据求属性闭包算法， $(HLJ)^+ = HIJKLM = U$

所以：HLJ 是 R 的唯一候选码。

(定理 1、定理 3、推理 1 和推理 2)



例 2：关系模式 $R(HIJKLM)$ ，其函数依赖集 $F=\{H \rightarrow I, K \rightarrow H, LM \rightarrow K, I \rightarrow L, KH \rightarrow M\}$ ，求 R 的候选码。

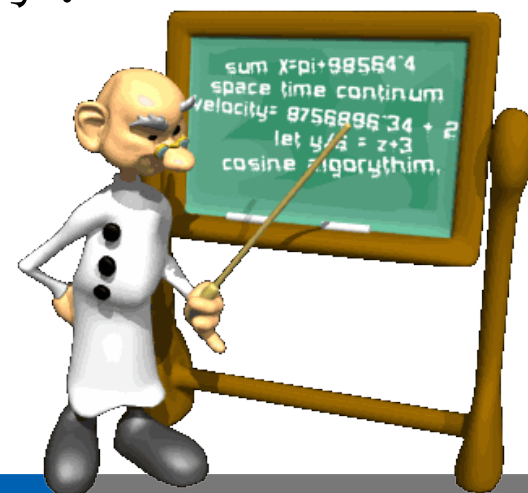
解答：J 是 NLR 属性必在 R 的候选码中，无 L 类属性，其余属性是 LR 类属性，所以候选码可能不唯一。

方法：从 $HIJKLM$ 中逐一去掉冗余属性即可。

- ① 若 H 为冗余属性，去除 H 得： **IJKLM**，因为有 $K \rightarrow H$ ，所以可去除 H 。
- ② 若 I 为冗余属性，去除 I 得： **JKLM**，因为有 $K \rightarrow H$ 、 $H \rightarrow I$ ，所以可去除 I 。
- ③ 若 K 为冗余属性，去除 K 得： **JLM**，因为有 $LM \rightarrow K$ ，所以可去除 K 。
- ④ 若 L 为冗余属性，去除 L 得到 JM ，因为无法从 JM 导出 L ，故 L 保留。

数据依赖的公理系统

- Q1. 给定关系模式 R 及其函数依赖集 F ，是否还存在其他的函数依赖成立？
- Q2. 怎样从函数依赖集 F 求得 F 蕴涵的函数依赖？有什么推理规则？
- Q3. F 所蕴涵全体函数依赖叫作什么？能否求出？
- Q4. 什么是属性集闭包？怎样求属性集闭包？
- Q5. 如何根据属性集闭包判断一个属性集 X 是否是候选码？
- Q6. 怎样求关系模式 R 的所有候选码？
- Q7. 什么是_{最小函数依赖集} F_m ？如何求出？



最小函数依赖集

函数依赖集 F 满足下列条件时，称 F 为最小函数依赖集或最小覆盖：

- ① F 中每个函数依赖的右部都是单属性。
- ② F 中不存在函数依赖 $X \rightarrow A$ ，使得 F 与 $F - \{X \rightarrow A\}$ 等价。
- ③ F 中不存在函数依赖 $X \rightarrow A$ ， X 有真子集 Z 使得： $F - \{X \rightarrow A\} \cup \{Z \rightarrow A\}$ 与 F 等价。



例：对于关系模式 $R\langle U, F \rangle$ ，其中：

$U = \{ Sno, Sdept, Mname, Cno, Grade \}$

$F = \{ Sno \rightarrow Sdept, Sdept \rightarrow Mname, (Sno, Cno) \rightarrow Grade \}$

$F' = \{ Sno \rightarrow Sdept, Sno \rightarrow Mname, Sdept \rightarrow Mname, (Sno, Cno) \rightarrow Grade, (Sno, Sdept) \rightarrow Sdept \}$

分析：F 和 F' 是否是最小函数依赖集

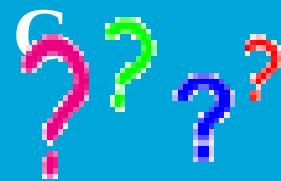
F 是最小依赖集

F' 不是最小依赖集：

$F' - \{ Sno \rightarrow Mname \}$ 与 F' 等价

$F' - \{ (Sno, Sdept) \rightarrow Sdept \}$ 也与 F' 等价

结论：每一个函数依赖集 F 都等价于一个最小函数依赖集



问题：如何求 G，即如何对 F 进行极小化处理？

求最小函数依赖集的算法

算法： 计算最小函数依赖集

输入： 一个函数依赖集 F

输出： F 的一个等价最小函数依赖集

步骤：

(1) 应用分解规则，使 F 中每个函数依赖的右部只含一个属性。

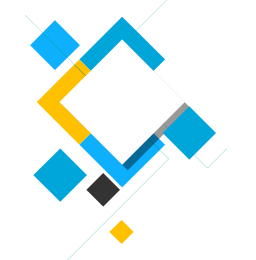
(2) 去掉多余函数依赖。

(3) 去掉各函数依赖左部多余的属性。

逐一检查 F 中各函数依赖 $X \rightarrow Y$ ，若 $Y = A_1 A_2 \dots A_k$, ($k > 2$)，根据分解规则，用 $\{X \rightarrow A_j\}$ 取代 $X \rightarrow Y$

逐一检查 F 中各函数依赖 $X \rightarrow A$ ，令 $G = F - \{X \rightarrow A\}$ ，若 $A \in X_G^+$ ，则从 F 中去掉此函数依赖

逐一检查 F 中各函数依赖 $X \rightarrow A$ ，若 $X = B_1 B_2 \dots B_m$ ，逐一考查 B_i ，若 $A \in (X - B_i)_F^+$ ，则以 $X - B_i$ 取代 X



例：已知 $F=\{A \rightarrow BCE, B \rightarrow C, C \rightarrow A, AD \rightarrow E, BD \rightarrow G\}$ ，
求最小函数依赖集 F_m 。

解答：

- ① 应用分解规则，使 F 中每个函数依赖的右端为单属性。 $F_1=\{A \rightarrow B, A \rightarrow C, A \rightarrow E, B \rightarrow C, C \rightarrow A, AD \rightarrow E, BD \rightarrow G\}$
- ② 去掉各函数依赖左边多余的属性。
 - ✓ 考察 $AD \rightarrow E$ ：由于有 $A \rightarrow E$ ，而 A 是 AD 的子集，故去掉 $AD \rightarrow E$ 。 $F_2 = \{A \rightarrow B, A \rightarrow C, A \rightarrow E, B \rightarrow C, C \rightarrow A, BD \rightarrow G\}$
 - ✓ 考察 $BD \twoheadrightarrow G$ ： $G \notin B_F^+, G \notin D_F^+$ ，保留 $BD \twoheadrightarrow G$ 。



$$F_2 = \{A \rightarrow B, A \rightarrow C, A \rightarrow E, B \rightarrow C, C \rightarrow A, BD \rightarrow G\}$$

③ 去掉多余函数依赖

- ✓ 若去掉 $A \rightarrow B$, $A^+ = ACE$, 因 B 不在 A^+ , 应保留。
- ✓ 若去掉 $A \rightarrow C$, $A^+ = ABEC$, 因 C 在 A^+ , 应去掉 $A \rightarrow C$

$$F_3 = \{A \rightarrow B, A \rightarrow E, B \rightarrow C, C \rightarrow A, BD \rightarrow G\}$$

- ✓ 若去掉 $A \rightarrow E$, $A^+ = ABC$, 因 E 不在 A^+ , 应保留。
- ✓ 若去掉 $B \rightarrow C$, $B^+ = B$, 因 C 不在 B^+ , 应保留。
- ✓ 若去掉 $C \rightarrow A$, $C^+ = C$, 因 A 不在 C^+ , 应保留。
- ✓ 若去掉 $BD \rightarrow G$, $BD^+ = ABCDE$, 因 G 不在 BD^+ , 应保留。

$$F_m = F_3 = \{A \rightarrow B, A \rightarrow E, B \rightarrow C, C \rightarrow A, BD \rightarrow G\}$$

关系数据理论

01 | 关系数据库的规范化理论

02 | 函数依赖和码

03 | 范式

04 | 数据依赖的公理系统

05 | 模式分解的定义及算法



模式分解的定义及算法

Q1 模式分解的 3 个定义

Q2 分解的无损连接性

Q3 分解的保持函数依赖

Q4 模式分解的算法

模式分解的三个定义

例：SL (Sno , Sdept , Sloc)

$F = \{ Sno \rightarrow Sdept , Sdept \rightarrow Sloc \}$

$SL \in 2NF$

存在插入异常、删除异常、冗余度大和修改复杂等问题

SL		
Sno	Sdept	Sloc
95001	CS	A
95002	IS	B
95003	MA	C
95004	IS	B
95005	PH	B

第一种分解方法

SL 分解为下面三个关系模式:

$\rho_1 = \{SN \langle Sno, \varphi \rangle, SD \langle Sdept, \varphi \rangle, SO \langle Sloc, \varphi \rangle\}$

SN
Sno
95001
95002
95003
95004
95005

SD
Sdept
CS
IS
MA
IS
PH

SO
Sloc
A
B
C
B
B

分解后的数据库无法反映数据之间的联系, 丢失了许多信息, 分解 ρ_1 没有意义!

第二种分解方法

$\rho_2 = \{NL < \{Sno, Sloc\}, \{Sdept \rightarrow Sloc\} >\}$

NL	
Sno	Sloc
95001	A
95002	B
95003	C
95004	B
95005	B

DL	
Sdept	Sloc
CS	A
IS	B
MA	C
PH	B



NL ∞ DL 比原来的 SL 关系多了 3 个元组，无法知道 95002、95004、95005 究竟是哪个系的学生。信息丢失了，有损连接！

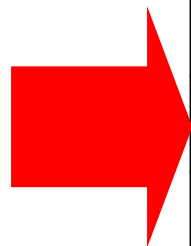
NL ∞ DL		
Sno	Sloc	Sdept
95001	A	CS
95002	B	IS
95002	B	PH
95003	C	MA
95004	B	IS
95004	B	PH
95005	B	IS
95005	B	PH

第三种分解方法

$\rho_3 = \{ND \langle \{Sno, Sdept\}, \{Sno \rightarrow Sdept\} \rangle, NL \langle \{Sno, Sloc\}, \{Sno \rightarrow Sloc\} \rangle\}$

ND	
Sno	Sdept
95001	CS
95002	IS
95003	MA
95004	IS
95005	PH

NL	
Sno	Sloc
95001	A
95002	B
95003	C
95004	B
95005	B



ND \bowtie NL		
Sno	Sdept	Sloc
95001	CS	A
95002	IS	B
95003	MA	C
95004	IS	B
95005	PH	B

存在问题：插入和删除异常还未得到解决！这种分解方法没有保持原关系 SL 中函数依赖 **$Sdept \rightarrow Sloc$** 。

NL \bowtie NL 与原来的 SL 关系一样， ρ_3 具有无损连接性。

4. 第四种分解方法

$\rho_4 = \{ND \langle \{Sno, Sdept\}, \{Sno \rightarrow Sdept\} \rangle, DL \langle \{Sdept, Sloc\},$

ND	
Sno	Sdept
95001	CS
95002	IS
95003	MA
95004	IS
95005	PH

DL	
Sdept	Sloc
CS	A
IS	B
MA	C
PH	B

ND ∞ DL		
Sno	Sdept	Sloc
95001	CS	A
95002	IS	B
95003	MA	C
95004	IS	B
95005	PH	B

原 SL 关系中的函数依赖:
 $Sno \rightarrow Sdept$ 、 $Sdept \rightarrow Sloc$
在 ρ_4 中均保持

NL ∞ NL 与原来的 SL 关系一样
， ρ_4 具有无损连接性。

结论:

- ① 如果一个分解具有无损连接性，则它能够保证不丢失信息。
- ② 如果一个分解保持了函数依赖，则它可以减轻或解决各种异常情况。
- ③ 分解具有无损连接性和分解保持函数依赖是两个互相独立的标准。具有无损连接性的分解不一定能够保持函数依赖。同样，保持函数依赖的分解也不一定具有无损连接性。



模式分解的定义及算法

Q1 模式分解的 3 个定义

Q2 分解的无损连接性

Q3 分解的保持函数依赖

Q4 模式分解的算法

① 分解的无损连接性

设 $\rho = \{ R_1 \langle U_1, F_1 \rangle, R_2 \langle U_2, F_2 \rangle, \dots, R_n \langle U_n, F_n \rangle \}$ 是关系模式 $R \langle U, F \rangle$ 的一个分解, 若 R 与 R_1 、 R_2 、 \dots 、 R_n 自然连接的结果相等, 则称分解 ρ 具有无损连接性。

- 具有无损连接性的分解保证不丢失信息。
- 无损连接性不一定能解决插入异常、删除异常、修改复杂、数据冗余等问题。

② 验证无损连接性的充要条件

✓ 若 $\rho = \{R_1, R_2\}$ 是关系模式 R 的一个分解, U_1 、 U_2 和 U 分别是 R_1 、 R_2 和 R 的属性集合, F 是 R 的函数依赖集合。则 ρ 具有无损连接性的充分必要条件是:

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

$$\text{或 } U_1 \cap U_2 \rightarrow U_2 - U_1 \in F^+$$

② 验证无损连接性的充要条件

例 1：设 $F=\{A \twoheadrightarrow C, B \twoheadrightarrow C\}$ 在 R 上成立，判断分解 $\rho = \{AC, BC\}$ 是否具有无损连接。

解答： $R_1=AC, R_2=BC$

$$R_1 \cap R_2 = C, R_1 - R_2 = A, R_2 - R_1 = B$$

因为： $R_1 \cap R_2 \not\rightarrow R_1 - R_2$ 和 $R_1 \cap R_2 \not\rightarrow R_2 - R_1$ 在 F 中都不满足，

所以 ρ 不具无损连接性。

③ 无损连接性的判断方法

输入：关系模式 $R\langle U, F \rangle$ 及分解 $\rho = \{R_1, R_2, \dots, R_k\}$.

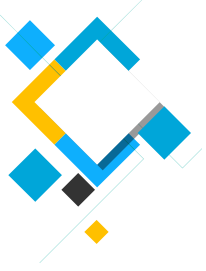
输出：判定 ρ 是否具有无损连接性

方法：

步骤 1：构造一个 k 行 n 列的表。

第 i 行对应关系模式 R_i ，第 j 列对应于属性 A_j 。

如果 $A_j \in R_i$ ，则在第 i 行第 j 列放上符号 a_j ，否则放符号 b_{ij} 。



例：关系模式 $R(ABCDE)$ ， $F=\{AB \rightarrow C, C \rightarrow D, D \rightarrow E\}$

试判断分解 $\rho=\{ABC, CD, DE\}$ 是否具有无损连接性。

步骤 1： 构造一个 3 行 5 列的表。

	A	B	C	D	E
ABC	a1	a2	a3	b14	b15
CD	b21	b22	a3	a4	b25
DE	b31	b32	b33	a4	a5

③ 无损连接性的判断方法

步骤 2：取 F 中的一个函数依赖 $X \rightarrow Y$ ，在表中检查属性 X 所在的列，寻找具有相同符号的行，然后将这些行中的属性 Y 所在的列的符号改为相同的符号：

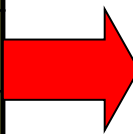
- 如果其中有 a_j ，则将 b_{ij} 改为 a_j
- 若无 a_j ，则改为 b_{mj} ，其中 m 是这些行中最小的行号



例：关系模式 $R(ABCDE)$ ， $F=\{AB \rightarrow C, C \rightarrow D, D \rightarrow E\}$
试判断分解 $\rho=\{ABC, CD, DE\}$ 是否具有无损连接性

(1) 初始表为：

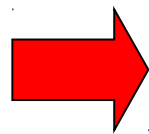
	A	B	C	D	E
ABC	a1	a2	a3	b14	b15
CD	b21	b22	a3	a4	b25
DE	b31	b32	b33	a4	a5



	A	B	C	D	E
ABC	a1	a2	a3	a4	b15
CD	b21	b22	a3	a4	b25
DE	b31	b32	b33	a4	a5

(2) 根据 $AB \rightarrow C$ ，由于 A 和 B 列没有相同符号的行，所以不修改表。
(3) 根据 $C \rightarrow D$ ，由于 C 列有相同符号的行，所以修改 D 列。

(4) 根据 $D \rightarrow E$ ，由于 D 列有相同符号的行，所以修改 E 列。



	A	B	C	D	E
ABC	a1	a2	a3	a4	a5
CD	b21	b22	a3	a4	a5
DE	b31	b32	b33	a4	a5

3、无损连接性的判断方法

步骤 3：

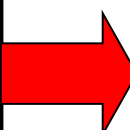
- ✓ 对 F 中每一个函数依赖逐个处理一遍后，若表中的某一行变为 $a_1 a_2 \dots a_n$ ，则可判定该分解具有无损连接性，算法终止。
- ✓ 否则转步骤 2 继续再把 F 扫描一遍，比较扫描前后表有无变化，如有变化则继续修改，直到表不能修改为止。
- ✓ 若在修改中发现有 $a_1 a_2 \dots a_n$ 行，则算法终止；如果最终也没有发现这样的行，则可判定该分解不具有无损连接性。



例：关系模式 $R(ABCDE)$ ， $F=\{AB \rightarrow C, C \rightarrow D, D \rightarrow E\}$
试判断分解 $\rho=\{ABC, CD, DE\}$ 是否具有无损连接性

(1) 初始表为：

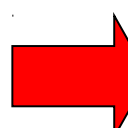
	A	B	C	D	E
ABC	a1	a2	a3	b14	b15
CD	b21	b22	a3	a4	b25
DE	b31	b32	b33	a4	a5



	A	B	C	D	E
ABC	a1	a2	a3	a4	b15
CD	b21	b22	a3	a4	b25
DE	b31	b32	b33	a4	a5

(2) 根据 $AB \rightarrow C$ ，由于 A 和 B 列没有相同符号的行，所以不修改表。
(3) 根据 $C \rightarrow D$ ，由于 C 列有相同符号的行，所以修改 D 列。

(4) 根据 $D \rightarrow E$ ，由于 D 列有相同符号的行，所以修改 E 列。



	A	B	C	D	E
ABC	a1	a2	a3	a4	a5
CD	b21	b22	a3	a4	a5
DE	b31	b32	b33	a4	a5

(5) 由于这时表的第一行为全 a，所以分解 ρ 具有无损连接性



模式分解的定义及算法

Q1 模式分解的 3 个定义

Q2 分解的无损连接性

Q3 分解的保持函数依赖

Q4 模式分解的算法

分解的保持函数依赖

关系模式 $R\langle U, F \rangle$ 的一个分解: $\square = \{R_1\langle U_1, F_1 \rangle, R_2\langle U_2, F_2 \rangle, \dots, R_k\langle U_k, F_k \rangle\}$, 若 $F^+ = (F_1 \cup F_2 \cup \dots \cup F_k)^+$, 则分解 \square 保持函数依赖。



例 1：关系模式 $R(A,B,C,D,E)$ ， $F=\{AB \rightarrow C, C \rightarrow D, D \rightarrow E\}$

试判断分解 $\rho=\{ABC, CD, DE\}$ 是否保持函数依赖。

解答：

F 在 ABC 上的投影 $F_1=\{AB \rightarrow C\}$

F 在 CD 上的投影 $F_2=\{C \rightarrow D\}$

F 在 DE 上的投影 $F_3=\{D \rightarrow E\}$

$F^+=(F_1 \cup F_2 \cup \dots \cup F_k)^+$ ，所以分解 ρ 函数依赖保持



例 2：设 $R(ABCDE)$ ， $F=\{A \rightarrow C, C \rightarrow D, B \rightarrow C, DE \rightarrow C, CE \rightarrow A\}$ ，判断 $\rho=\{AD, AB, BC, CDE, AE\}$ 是否为函数依赖保持。

F 在 AD 上的投影 $F_1=\{A \rightarrow D\}$

F 在 AB 上的投影 F_2 为空集 Φ

F 在 BC 上的投影 $F_3=\{B \rightarrow C\}$

F 在 CDE 上的投影 $F_4=\{C \rightarrow D, DE \rightarrow C\}$

F 在 AE 上的投影 F_5 为空集 Φ

$F_1 \cup F_2 \cup F_3 \cup F_4 \cup F_5 = \{A \rightarrow D, B \rightarrow C, C \rightarrow D, DE \rightarrow C\}$

$(F_1 \cup F_2 \cup F_3 \cup F_4 \cup F_5)^+$ 与 F^+ 不等价，所以 ρ 不保持函数依赖。



模式分解的定义及算法

Q1 模式分解的 3 个定义

Q2 分解的无损连接性

Q3 分解的保持函数依赖

Q4 模式分解的算法

- ① 转换为 3NF 的保持函数依赖的分解算法
- ② 转换为 3NF 既有无损连接又保持函数依赖的分解算法
- ③ 转换为 BCNF 的无损连接的分解算法

算法 1：转换为 3NF 保持函数依赖的分

输入：关系模式 $R(U, F)$

输出：分解 $\rho = \{R_1, \dots, R_k\}$ ， R_i 为 3NF，且 ρ 保持函数依赖

- ① 对 F 进行极小化处理。
- ② 找出不在 F 中出现的属性并将其构成一个关系模式，并把这些属性从 U 中去掉，剩余属性仍记为 U 。
- ③ 若 $X \rightarrow A \in F$ ，且 $XA=U$ ，则输出 $\rho = \{R\}$ ，结束。
- ④ 否则对 F 按相同左部的原则分组，每组函数依赖所涉及到的全部属性形成一个属性集 U_i ，若 $U_i \subseteq U_j (i \neq j)$ 则去掉 U_i 。

最后输出： $R_1(U_1, F_1), \dots, R_k(U_k, F_k)$



例：关系模式 $R(U, F)$ ， $U=(ABCDE)$ ， $F=\{A \rightarrow D, E \rightarrow D, D \rightarrow B, BC \rightarrow D, CD \rightarrow A\}$ ，将 R 分解为 3NF 模式集并保持函数依赖。

① F 已为最小函数依赖集。

② R 中所有属性均在 F 中出现。

③ 不存在函数依赖 $X \rightarrow A \in F$ ，且 $XA=U$ 。

④ 对 F 按具有相同左部的原则分组为

$R_1=AD$ ， $R_2=ED$ ， $R_3=DB$ ， $R_4=BCD$ ， $R_5=CDA$ 。但 $R_1 \subseteq R_5$ ， $R_3 \subseteq R_4$

④ $\rho=\{ED, BCD, CDA\}$

算法 2：转换为 3NF 既有无损连接又保持函数依赖的分解

输入：关系模式 R 和 R 的最小函数依赖集 F_m 。

输出：分解 $\rho=\{R_1,\dots,R_k\}$ ， R_i 为 3NF，且 ρ 具有无损连接性和保持函数依赖。

方法：

- ① 根据算法 1 求出函数依赖保持性分解 $\rho=\{R_1, R_2, \dots, R_k\}$
- ② 判定 ρ 是否具有无损连接性，若是则转④
- ③ 令 $\rho=\rho \cup \{X\}$ ，其中 X 是 R 的候选码
- ④ 输出 ρ

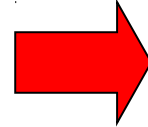


例：设有关系模式 $R(U, F)$, $U=(C, T, H, I, S, G)$, $F=\{CS \rightarrow G, C \rightarrow T, TH \rightarrow I, HI \rightarrow C, HS \rightarrow I\}$, 请将 R 分解为既具无损连接性又保持函数依赖的 3NF 模式集。

解答：①由算法 1 得： $\rho=\{CSG, CT, THI, HIC, HSI\}$.

② 判断 ρ 是否具有无损连接性。

	C	T	H	I	S	G
CSG	a1	b12	b13	b14	a5	a6
CT	a1	a2	b23	b24	b25	b26
THI	b31	a2	a3	a4	b35	b36
HIC	a1	b41	a3	a4	b45	b46
HSI	b51	b52	a3	a4	a5	b56



	C	T	H	I	S	G
CSG	a1	a2	b13	b14	a5	a6
CT	a1	a2	b23	b24	b25	b26
THI	a1	a2	a3	a4	b35	b36
HIC	a1	a2	a3	a4	b45	b46
HSI	a1	a2	a3	a4	a5	a6

③ 分解 ρ 具有无损连接性

输出 $\rho =\{CSG, CT, THI, HIC, HSI\}$.

算法 3：转换为 BCNF 的无损连接的分解

输入：关系模式 R 和 R 的函数依赖集 F

输出：无损连接分解 $\rho = \{R_1, R_2, \dots, R_k\}$ ，每个 R_i 都是 BCNF

方法：

- ① 置初值 $\rho = \{R\}$
- ② 如果 ρ 中所有的 R_i 均已是 BCNF 则转④
- ③ 如果 ρ 中有非 BCNF 的关系模式 R_i ，则 R_i 中必有 $X \rightarrow A$ ， X 不包含 R_i 的码且 A 不属于 X 。设 $R_{i1} = XA$ ， $R_{i2} = R_i - A$ ，用 $\{R_{i1}, R_{i2}\}$ 代替 R_i ，转②
- ④ 分解结束，输出 ρ 。



例：设有关系模式 $R(C, T, H, I, S, G)$ ， $F=\{CS \rightarrow G, C \rightarrow T, TH \rightarrow I, HI \rightarrow C, HS \rightarrow I\}$ ，请将 R 无损连接分解为 BCNF 模式集。

解答：分析得知 R 只有一个候选码 HS 。

① $\rho=\{CTHISG\}$ ， ρ 中的模式不是 BCNF。

② 考虑 $CS \rightarrow G$ ： CS 不包含候选码 HS ，故将 R 分解为：

$R1=\{CSG\}$ ， $R2=R-\{G\}=\{CTHIS\}$

④ 再分解 $CTHIS$ （候选码 HS ），考虑 $C \rightarrow T$ 中 C 不包含候选码，故将 $CTHIS$ 分解为：

$R21=\{CT\}$ ， $R22=\{CHIS\}$

⑤ 再分解 $CHIS$ （候选码 HS ），考虑 $HI \rightarrow C$ 中 HI 不包含候选码，故将其分解为： CHI 和 HIS ，都是 BCNF

⑥ $\rho=\{CSG, CT, CHI, HIS\}$ 。

模式分解的算法——小结

1. 要求分解保持函数依赖，则模式分解总可达到 3NF，但不一定达到 BCNF。
2. 若要求分解具有无损连接性，则分解必可达到 BCNF。
3. 若要求分解既具有函数依赖保持性又具有无损连接性，则模式分解一定可达到 3NF，但不一定达到 BCNF。
4. 并不是规范化程度越高，模式就越好。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。