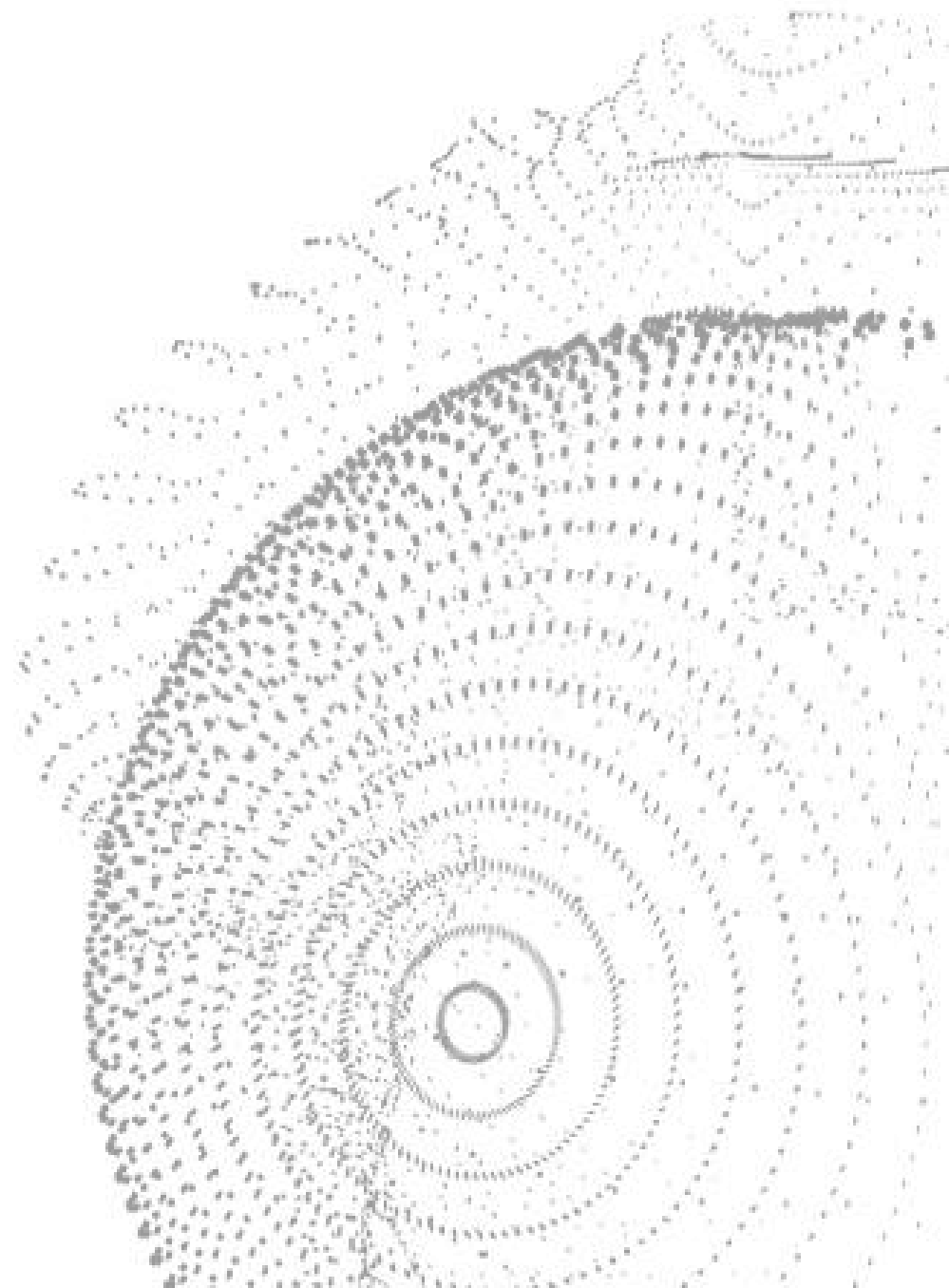


Chapter 07

关系数据库的规范化理论

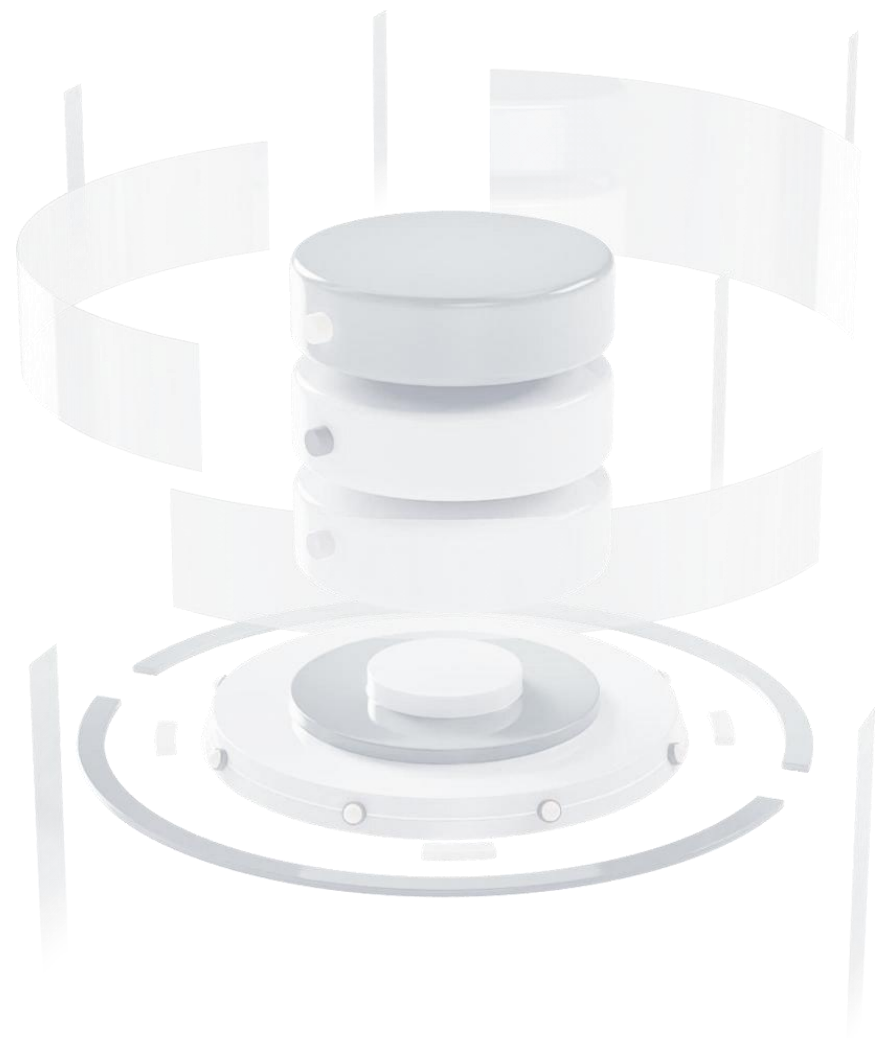
Normalization Theory of Relational Databases



- **在关系数据库系统的建立过程中，
如何构造一个合适的关系数据模式？**

- 关系数据库的设计理论

- 关系数据库的规范化理论



- **7.1 概述**
- **7.2 规范化理论**
 - 7.2.1 函数依赖
 - 7.2.2 与函数依赖有关的范式
 - 7.2.3 多值依赖与第四范式
- **7.3 规范化所引起的一些问题**
 - 函数依赖理论的研究
 - 模式分解的研究：无损联接性，依赖保持性

第7章 目录

-01

概论

-02

规范化理论

-03

规范化所引起的一些问题

- **模式设计**

- 同一个数据库系统可以有多种不同的模式设计方案。

- **如：假设一个学生数据库中有8个属性：S[#], S_n, S_d, S_a, C[#], G, CN, P[#], 可以采用的模式设计方案有多个。如表7-1所示：**

- 方案1：一个关系

SCG(S[#], S_n, S_d, S_a, C[#], G, CN, P[#])

- 方案2：三个关系

S(S[#], S_n, S_d, S_a)

C(C[#], CN, P[#])

SC(S[#], C[#], G)

7.1.2

不同模式设计方案的比较

- 不同的模式设计方案对数据库的影响是否相同？
- 例：
 - 根据方案1所建立的数据库如表7-2所示
 - 根据方案2所建立的数据库如表7-3所示
- 我们从下面三个方面来比较这两个数据库：
 - 数据冗余度
 - 元组插入操作
 - 元组删除操作

7.1.2

不同模式设计方案的比较

S#	Sn	Sd	Sa	C#	Cn	P#	G
0001	王剑飞	CS	17	101	ABC	102	5
0001	王剑飞	CS	17	102	ACD	105	5
0001	王剑飞	CS	17	103	BBC	105	4
0001	王剑飞	CS	17	105	AEF	107	3
0001	王剑飞	CS	17	110	BCF		4
0002	陈 瑛	MA	19	103	BBC	105	3
0002	陈 瑛	MA	19	105	AEF	107	3
0003	方世觉	CS	17	107	BHD	110	4

表7-2 根据方案1所建立的数据库(关系SCG)

7.1.2

不同模式设计方案的比较

S#	Sn	Sd	Sa	关系S
0001	王剑飞	CS	17	
0002	陈 瑛	MA	19	
0003	方世觉	CS	17	
C#	Cn	P#	关系C	
101	ABC	102		
102	ACD	105		
103	BBC	105		
105	AEF	107		
107	BHD	110		
110	BCF			
S#	C#	G	关系SC	
0001	101	5		
0001	102	5		
0001	103	4		
0001	105	3		
0001	110	4		
0002	103	3		
0002	105	3		
0003	107	4		

表7-3 根据方案2所建立的数据库(关系S, C和SC)

- 经过比较发现，表7-2具有如下缺点：
 - 数据冗余度大
 - 插入异常
 - 如果需要新开设一门尚未有学生选修的课程(104, DB, 103)，则无法构造出一个由 $S^{\#}$, S_n , S_d , S_a 等属性值所组成的新元组，在表7-2中就无法执行元组的插入操作。
- 在表7-3中，我们可以直接将元组(104, DB, 103)插入到课程关系C中。

- 删除异常

- 在表7-2中，107号课程仅有0003号学生选修，如果该学生因故退学，就必须将与该学生有关的元组从表7-2中删除掉，这样就必然也将107号这门课程也从数据库中删除掉了。
- 在表7-3中，我们可以仅在学生关系S和选课关系SC中删除0003号学生的元组及其选课信息，但不会误删除掉107号课程，其所对应的元组仍然保留在课程关系C中。

- 因此，不同的模式设计方案有好坏的区分。
-
- 好的设计方案应该是：
 - 既具有合理的数据冗余度，又没有插入和删除等异常现象的出现。

- **在不同的设计结果之间产生区别的原因**
 - 数据库的各属性之间是互相关联的，它们互相依赖、互相制约，构成一个结构严密的整体。
- 要设计出一个好的关系模式，必须从数据库中所有属性的语义上进行分析，从语义上入手分清每个属性的语义含义及其相互之间的依存关系。进而将那些相互依赖密切的属性组合在一起构成一个关系模式，避免对属性的松散组合所引起的‘排它性’，从而可以降低数据冗余度，避免上述异常现象的产生。

- **关系的规范化**

- 在一个关系中，属性与属性之间的内在语义联系有两种：
 - 函数依赖 & 多值依赖
- 关系的规范化
 - 在每个关系中，属性与属性之间一定要满足某种内在的语义联系，这被称为关系的规范化。
 - 根据对属性间所存在的内在语义联系要求的不同，又可以将关系的规范化分为若干个级别，这被称为范式。
- 上述相关的理论被称为关系规范化理论。

第7章 目录

-01

概论

-02

规范化理论

-03

规范化所引起的一些问题

- 数据依赖理论

- 函数依赖 (FD – Functional Dependency)
 - 1970年, E. F. Codd
 - 1972 – 1974年, Codd, Casey, Bernstein, Armstrong
 - 完全/部分FD, 平凡/非平凡FD, 直接/传递FD
 - 键 (key)
 - 1974年, Armstrong公理系统
 - FD的逻辑蕴涵
 - FD的形式化推理规则集
- 多值依赖 (MVD – Multi-Valued Dependency)
 - 1976 – 1978年, Zaniolo, Fagin, Delobel

- 规范化理论

- 规范化的途径

- 竖向规范化

- 采用‘投影’和‘联接’运算
 - 将一个关系模式的属性集分解构成若干个关系模式
 - 有关理论构成了关系数据库的规范化理论
 - 模式分解理论：无损联接性，依赖保持性

- 横向规范化

- 水平规范化
 - 采用‘选择’和‘并’运算
 - 将一个关系的元组集合分解成若干个子集，从而构成若干个与原来的关系具有相同关系模式的子关系
 - 尚未形成一个成熟的规范化理论

7.2.1 函数依赖

- 规范化的途径
 - 函数依赖
 - 完全/部分FD, 平凡/非平凡FD, 直接/传递FD
 - Armstrong公理系统
 - 键 (key)
 - 两个算法
 - 属性集的闭包的计算
 - 关键字的计算

7.2.2 与函数依赖有关的范式

- 范式：1NF，2NF，3NF，BCNF

7.2.3 多值依赖与第四范式

- 多值依赖
 - 与MVD有关的推理规则
- 4NF

7.2.1

函数依赖

- 在一个关系模式 $R(U)$ 中，如果一部分属性 Y 的取值依赖于另一部分属性 X 的取值，则在属性集 X 和属性集 Y 之间存在着一种数据依赖关系，我们称之为函数依赖。

[例1]

在学生关系模式 $S(S\#, S_n, S_d, S_a)$ 中就存在下面的几组依赖关系：

$\{S_n\}$ 的取值依赖于 $\{S\#\}$ 的取值

$\{S_d\}$ 的取值依赖于 $\{S\#\}$ 的取值

$\{S_a\}$ 的取值依赖于 $\{S\#\}$ 的取值

7.2.1

函数依赖

定义7.1: 函数依赖

- 设有关系模式 $R(U)$ ， U 是关系模式 R 的属性集合， X 、 Y 是 U 的子集。若对于任一个符合关系模式 $R(U)$ 的关系 r 中的任一元组 t 在 X 中的属性值确定后，则元组 t 在 Y 中的属性值也必确定，则称 Y 函数依赖于 X ，或者称 X 函数决定 Y ，并记为 $X \rightarrow Y$ 。
- 其中：
 - X 称为决定因素
 - Y 称为依赖因素

7.2.1

函数依赖

定义7.1: 函数依赖 (cont.)

- 假设在关系模式R(U)上存在函数依赖关系: $X \rightarrow Y$
- r 是依据关系模式R(U)建立起来的任意一个关系, 那么关系 r 必满足:
 - 从关系 r 中任取两个元组 t_1 和 t_2 , 如果元组 t_1 在X这组属性上的取值 $t_1[X]$ 等于元组 t_2 在X这组属性上的取值 $t_2[X]$, 即:

$$t_1[X] = t_2[X]$$

- 则它们在Y这组属性上的取值也必定相等, 即:

$$t_1[Y] = t_2[Y]$$

7.2.1

函数依赖

- 函数依赖是语义范畴上的概念，只有根据属性间固有的语义联系才能归纳出与客观事实相符合的函数依赖关系，而不是仅从现有的一个或若干个关系实例中得出的结论。
- 特定的关系实例虽然不能用于函数依赖的发现，但可以用于否定某些函数依赖。

S#	Sn	Sd	Sa
0001	王剑飞	CS	17
0002	陈 瑛	MA	19
0003	方世觉	CS	17

关系S

7.2.1

函数依赖

- 在关系模式 $S(S\#, S_n, S_d, S_a)$ 上，可能存在的函数依赖有很多，例如： $S\# \rightarrow S_n$, $S_n \rightarrow S_d$, $S_d \rightarrow S_n$, $\{S\#, S_n\} \rightarrow S_d$,
- 但是，根据下面的关系 S （依据上述的关系模式而建）：

$S\#$	S_n	S_d	S_a
0001	王剑飞	CS	17
0002	陈 瑛	MA	19
0003	方世觉	CS	17

- 可以否认以下的函数依赖： $S_d \rightarrow S\#$, $S_d \rightarrow S_n$, ...
- 虽然不能否定，但也不能确认以下的函数依赖成立： $S\# \rightarrow S_n$, $S_d \rightarrow S_a$,

7.2.1

函数依赖

[例] 根据下列具体的关系实例，判断其中可能存在哪些函数依赖关系？

T1

A	B
x1	y1
x2	y2
x3	y1
x4	y1
x5	y2
x6	y2

$A \rightarrow B ?$ ✓

$B \rightarrow A ?$ ✗

T2

A	B
x1	y1
x2	y4
x1	y1
x3	y2
x2	y4
x4	y3

$A \rightarrow B ?$ ✓

$B \rightarrow A ?$ ✓

T3

A	B
x1	y1
x2	y4
x1	y1
x3	y2
x2	y4
x2	y3

$A \rightarrow B ?$ ✗

$B \rightarrow A ?$ ✓

7.2.1

函数依赖

- 设有如下所示的关系R

A	B	C	D
a1	b1	c1	d1
a2	b2	c2	d2
a3	b3	c3	d3
a4	b4	c4	d4

- 其中可能成立的函数依赖关系有哪些？
- 其中又有哪些函数依赖关系是不可能成立的？

7.2.1

函数依赖

A	B	C	D
a1	b1	c1	d1
a2	b2	c2	d2
a3	b3	c3	d3
a4	b4	c4	d4

- 首先考虑决定因素和依赖因素都是单个属性的情况:

$A \rightarrow B$ ✓

$A \rightarrow C$ ✗

$A \rightarrow D$ ✗

$B \rightarrow A$ ✗

$B \rightarrow C$ ✗

$B \rightarrow D$ ✗

$C \rightarrow A$ ✗

$C \rightarrow B$ ✓

$C \rightarrow D$ ✗

$D \rightarrow A$ ✓

$D \rightarrow B$ ✓

$D \rightarrow C$ ✓

7.2.1

函数依赖

A	B	C	D
a1	b1	c1	d1
a2	b2	c2	d2
a3	b3	c3	d3
a4	b4	c4	d4

- 也可以按照如下顺序考虑可能存在的函数依赖情况：

$A \rightarrow B$ ✓

$A \rightarrow C$ ✗

$A \rightarrow D$ ✗

$B \rightarrow A$ ✗

$B \rightarrow C$ ✗

$B \rightarrow D$ ✗

$C \rightarrow A$ ✗

$C \rightarrow B$ ✓

$C \rightarrow D$ ✗

$D \rightarrow A$ ✓

$D \rightarrow B$ ✓

$D \rightarrow C$ ✓

7.2.1

函数依赖

A	B	C	D
a1	b1	c1	d1
a2	b2	c2	d2
a3	b3	c3	d3
a4	b4	c4	d4

$A \rightarrow B$

$C \rightarrow B$

$D \rightarrow A$

$D \rightarrow B$

$D \rightarrow C$

- 其次，再考虑决定因素是多个属性的情况：

1. 在FD的左边不需要考虑含有属性 **D** 的情况，why？
2. 在FD的左边不需要考虑含有属性 **B** 的情况，why？

因此只需要考虑下述的FD是否成立： $AC \rightarrow B$ ？ $AC \rightarrow D$ ？

7.2.1

函数依赖

A	B	C	D
a1	b1	c1	d1
a2	b2	c2	d2
a3	b3	c3	d3
a4	b4	c4	d4

 $A \rightarrow B$ $C \rightarrow B$ $D \rightarrow A$ $D \rightarrow B$ $D \rightarrow C$ $AC \rightarrow B ?$ $AC \rightarrow D ?$

- 在上述的FD关系中，我们不用考虑 $AC \rightarrow B$ ，**why**？
- 因此，最后只需要考虑下面的一个FD是否可能成立？

 $AC \rightarrow D \quad \checkmark$

7.2.1

函数依赖

A	B	C	D
a1	b1	c1	d1
a2	b2	c2	d2
a3	b3	c3	d3
a4	b4		

- 该关系上可能存在的函数依赖

$A \rightarrow B$

$D \rightarrow B$

- 思考问题：**为什么没有写出

- 左边含有属性D的其它的那些可能的FD？
- 右边为单个属性B的其它的那些可能的FD？
- 右边为多个属性的那些可能的FD？

a) 形如1)和2)这两种情况的函数依赖，都是可能成立的，并且可以从已写出的这六个函数依赖中推导出来。

b) 在情况3)中，可以用已写出的这六个函数依赖来证明它是否成立。

7.2.1

函数依赖

- 函数依赖反映的是同一个关系中的两个属性子集之间在取值上的依存关系，这种依存关系实际上也是一种数据完整性约束。因此，我们也可以通过数据完整性约束条件的分析来寻找属性之间的函数依赖关系。
- **例3：**有一个学生关系 $R(S^{\#}, S_n, S_d, S_s, C^{\#}, G)$ ，其中 S_s 表示学生所学专业。该关系模式中的语义约束如下：
 - 每个学生均只属一个系与一个专业
 - 每个学生修读之每门课有且仅有一个成绩
 - 各系无相同专业

7.2.1

函数依赖

例3: 学生关系 $R(S^#, S_n, S_d, S_s, C^#, G)$

- 每个学生有唯一的一个学号，每个学生只有一个姓名（基本常识）

$$S^# \rightarrow S_n$$

- 每个学生均只属一个系与一个专业

$$S^# \rightarrow S_d$$

$$S^# \rightarrow S_s$$

- 每个学生修读之每门课有且仅有一个成绩

$$(S^#, C^#) \rightarrow G$$

- 各系无相同专业

$$S_s \rightarrow S_d$$

7.2.1

函数依赖

定义7.2: 平凡/非平凡函数依赖

- 一个函数依赖关系 $X \rightarrow Y$ 如满足 $Y \not\subseteq X$, 则称此函数依赖是非平凡的函数依赖。否则, 我们称其为平凡函数依赖。
- 如无特殊声明, 凡提到函数依赖时总认为指的是非平凡的函数依赖。

7.2.1

函数依赖

定义7.3: 完全函数依赖

- 在关系模式 $R(U)$ 中, 如有 $X \subseteq U$, $Y \subseteq U$, 满足 $X \rightarrow Y$, 且对任何 X 的真子集 X' 都有 $X' \not\rightarrow Y$, 则称 Y 完全函数依赖于 X , 并记作: $X \xrightarrow{f} Y$

定义7.4: 部分函数依赖

- 在关系模式 $R(U)$ 中, 如有 $X \subseteq U$, $Y \subseteq U$, 满足 $X \rightarrow Y$, 但 Y 不完全函数依赖于 X , 则称 Y 部分依赖于 X , 并记作: $X \xrightarrow{p} Y$

7.2.1

函数依赖

例如：在学生 $S(S^{\#}, S_n, S_d, S_s, C^{\#}, G)$ 中

- 我们有： $S^{\#} \rightarrow S_d$

- 同样也会有

$$(S^{\#}, S_n) \xrightarrow{p} S_d$$

$$(S^{\#}, S_a) \xrightarrow{p} S_d$$

- 我们有

$$S^{\#} \rightarrow S_d$$

$$S_s \rightarrow S_d$$

- 同样也会有

$$(S^{\#}, S_s) \xrightarrow{p} S_d$$

7.2.1

函数依赖

定义7.5: 传递函数依赖

- 在关系模式 $R(U)$ 中, 如有 $X \subseteq U$, $Y \subseteq U$, $Z \subseteq U$ 且满足: $X \rightarrow Y$, $Y \not\rightarrow X$, $Y \rightarrow Z$, $Z \not\rightarrow Y$, 则称 Z 传递函数依赖于 X ; 否则, 称为非传递函数依赖 (直接函数依赖)。
- 为了使得函数依赖在表示形式上的简单化, 传递函数依赖与非传递函数依赖在表示形式上没有区别。

7.2.1

函数依赖

例如：在学生关系中增加一个属性‘系的电话号码Dt’

$S(S\#, Sn, Sd, Ss, C\#, G, Dt)$

- 每个系只登记唯一的一个电话号码，则我们有：

$Sd \rightarrow Dt$

- 由 $S\# \rightarrow Sd$ 和 $Sd \rightarrow Dt$ 可得到传递FD: $S\# \rightarrow Dt$
- 由 $S\# \rightarrow Ss$ 和 $Ss \rightarrow Sd$ 可得到传递FD: $S\# \rightarrow Sd$

7.2.1

函数依赖

- **Armstrong公理系统**

- 有关函数依赖的六条推理规则
 - 基本规则 (3条)
 - 自反规则, 增广规则, 传递规则
 - 扩充规则 (3条)
 - 分解规则, 合并规则, 伪传递规则
- FD的逻辑蕴涵概念
- 函数依赖集F的闭包: F^+

7.2.1

Armstrong公理系统

- **基本推理规则**

- 规则 R_1 : 自反规则

- 如果 Y 是 X 的子集, 则: $X \rightarrow Y$

证明:

- 设 t_1, t_2 是关系 R 中的两个元组($t_1 \in R, t_2 \in R$), 且它们在属性集 X 上的值相等($t_1[X] = t_2[X]$)
 - 由于 Y 是 X 的子集, 即 $Y \subseteq X$
 - 因此必有 $t_1[Y] = t_2[Y]$

证毕

7.2.1

Armstrong公理系统

- 基本推理规则

- 规则 R_2 : 增广规则

- 如果 $X \rightarrow Y$, 则: $XZ \rightarrow YZ$

证明:

- 设 $t_1 \in R, t_2 \in R$, 如果 $t_1[XZ] = t_2[XZ]$, 则:
 - $t_1[X] = t_2[X]$ (1)
 - $t_1[Z] = t_2[Z]$ (2)
 - 由(1)及 $X \rightarrow Y$ 得: $t_1[Y] = t_2[Y]$ (3)
 - 由(2)及(3)得: $t_1[YZ] = t_2[YZ]$

证毕

7.2.1

Armstrong公理系统

- 基本推理规则

- 规则 R_3 : 传递规则

- 如果 $X \rightarrow Y$, $Y \rightarrow Z$, 则: $X \rightarrow Z$

证明:

- 设 $t_1 \in R, t_2 \in R$, 如果 $t_1[X] = t_2[X] \dots \dots \dots (1)$

- 由(1)及 $X \rightarrow Y$ 得: $t_1[Y] = t_2[Y] \dots \dots \dots (2)$

- 由(2)及 $Y \rightarrow Z$ 得: $t_1[Z] = t_2[Z]$

证毕

7.2.1

Armstrong公理系统

- 基本推理规则

- 规则 R_4 : 分解规则

- 如果 $X \rightarrow YZ$, 则: $X \rightarrow Y$

证明:

- 由自反规则 R_1 得:

$$YZ \rightarrow Y$$

- 由 $X \rightarrow YZ$, $YZ \rightarrow Y$ 根据传递规则 R_3 得:

$$X \rightarrow Y$$

证毕

7.2.1

Armstrong公理系统

- **基本推理规则**

- 规则 R_5 : 合并规则

- 如果 $X \rightarrow Y$ 且 $X \rightarrow Z$, 则: $X \rightarrow YZ$

证明:

- 使用增广规则 R_2 可作如下推导: :

- 由 $X \rightarrow Y$ 可得: $XX \rightarrow XY$ 即 $X \rightarrow XY$ (1)

- 由 $X \rightarrow Z$ 可得: $XY \rightarrow YZ$ (2)

- 由(1), (2)根据传递规则 R_3 可得:

- $X \rightarrow YZ$

证毕

7.2.1

Armstrong公理系统

- 基本推理规则

- 规则 R_6 : 合并规则伪传递规则

- 如果 $X \rightarrow Y$ 且 $WY \rightarrow Z$, 则: $WX \rightarrow Z$

证明:

- 使用增广规则 $R2_2$, 由 $X \rightarrow Y$ 可得:

- $WX \rightarrow WY$ (1)

- 使用传递规则 R_3 , 由 (1) 及 $WY \rightarrow Z$ 可得:

- $WX \rightarrow Z$

证毕

7.2.1

Armstrong公理系统

练习： 请利用Armstrong公理系统证明下面的推导过程是否成立？如果不成立，请给出具体的例子关系。

1. $\{W \rightarrow Y, X \rightarrow Z\} \Rightarrow \{WX \rightarrow Y\}$
2. $\{X \rightarrow Y\} \text{ and } Z \subseteq Y \Rightarrow \{X \rightarrow Z\}$
3. $\{X \rightarrow Y, X \rightarrow W, WY \rightarrow Z\} \Rightarrow \{X \rightarrow Z\}$
4. $\{XY \rightarrow Z, Y \rightarrow W\} \Rightarrow \{XW \rightarrow Z\}$
5. $\{X \rightarrow Z, Y \rightarrow Z\} \Rightarrow \{X \rightarrow Y\}$
6. $\{X \rightarrow Y, XY \rightarrow Z\} \Rightarrow \{X \rightarrow Z\}$
7. $\{X \rightarrow Y, Z \rightarrow W\} \Rightarrow \{XZ \rightarrow YW\}$
8. $\{XY \rightarrow Z, Z \rightarrow X\} \Rightarrow \{Z \rightarrow Y\}$
9. $\{X \rightarrow Y, Y \rightarrow Z\} \Rightarrow \{X \rightarrow YZ\}$
10. $\{XY \rightarrow Z, Z \rightarrow W\} \Rightarrow \{X \rightarrow W\}$

7.2.1

函数依赖

- **FD的逻辑蕴涵概念**

- 设 F 是关系模式 $R(U)$ 的一个函数依赖集, X, Y 是关系模式 R 的属性子集, 如果从 F 中的已有函数依赖关系利用 Armstrong 公理系统能够推出 $X \rightarrow Y$, 则称 F 逻辑蕴涵 $X \rightarrow Y$, 并记为:

$$F \models X \rightarrow Y$$

- **函数依赖集 F 的闭包 F^+**

- 由被 F 逻辑蕴涵的所有函数依赖关系构成的集合被称为函数依赖集 F 的闭包, 并记为 F^+

$$F^+ = \{ X \rightarrow Y \mid F \models X \rightarrow Y \}$$

7.2.1

函数依赖

- 计算函数依赖集 $F = \{A \rightarrow B, B \rightarrow C\}$ 的闭包 F^+

- F 中的所有函数依赖都是其闭包中的元素，即：

$$A \rightarrow B \in F^+ \quad B \rightarrow C \in F^+$$

- 根据自反规则，下述函数依赖也是其闭包中的元素

$$A \rightarrow A$$

$$B \rightarrow B$$

$$C \rightarrow C$$

$$AB \rightarrow A$$

$$AB \rightarrow B$$

$$AB \rightarrow AB$$

$$AC \rightarrow A$$

$$AC \rightarrow C$$

$$AC \rightarrow AC$$

$$BC \rightarrow B$$

$$BC \rightarrow C$$

$$BC \rightarrow BC$$

$$ABC \rightarrow A$$

$$ABC \rightarrow B$$

$$ABC \rightarrow C$$

$$ABC \rightarrow AB$$

$$ABC \rightarrow AC$$

$$ABC \rightarrow BC$$

$$ABC \rightarrow ABC$$

7.2.1

函数依赖

- 由 $A \rightarrow B$, $B \rightarrow C$ 及传递规则可得到闭包中的下列元素
 $A \rightarrow C$
- 由 $A \rightarrow B$, $A \rightarrow C$ 及合并规则可得到闭包中的下列元素
 $A \rightarrow BC$
- 由 $A \rightarrow B$ 及增广规则可得到闭包中的下列元素
 $A \rightarrow AB$ $AC \rightarrow BC$ $AC \rightarrow ABC$
- 由 $B \rightarrow C$ 及增广规则可得到闭包中的下列元素
 $AB \rightarrow AC$ $B \rightarrow BC$ $AB \rightarrow ABC$
- 由 $A \rightarrow C$ 及增广规则可得到闭包中的下列元素
 $A \rightarrow AC$ $AB \rightarrow BC$
- 由 $A \rightarrow BC$ 及增广规则可得到闭包中的下列元素
 $A \rightarrow ABC$

7.2.1

函数依赖

- 由 $AB \rightarrow B$, $B \rightarrow C$ 及传递规则可得到闭包中的下列元素

$AB \rightarrow C$

- 由 $AC \rightarrow A$, $A \rightarrow B$ 及传递规则可得到闭包中的下列元素

$AC \rightarrow B$

- 由 $AC \rightarrow B$ 及增广规则可得到闭包中的下列元素

$AC \rightarrow AB$

7.2.1

函数依赖

定义7.6: 关键字 (也称为 码 或 key)

- 在关系模式 $R(U, F)$ 中, 如有 $K \subseteq U$ 且满足:

$$K \xrightarrow{f} U$$

则称 K 为 R 的关键字。

- 几种不同的关键字
 - 候选关键字
 - 主关键字
 - 全关键字

7.2.1

函数依赖

定义7.7: 主属性(集)

- 由关系模式 R 的所有关键字中的属性所构成的集合被称为关系模式 R 的主属性集
- 主属性集中的属性被称为关系模式 R 的主属性

定义7.8: 非主属性(集)

- 由主属性集之外的其它属性所构成的集合被称为关系模式 R 的非主属性集
- 非主属性集中的属性被称为关系模式 R 的非主属性

7.2.1

函数依赖

- 如何寻找一个关系模式R(U, F)的关键字？
 - **方法一：** 利用Armstrong公理系统中的推导规则，从已知的函数依赖集F中推导得到如下的函数依赖关系：

$$K \xrightarrow{f} U$$

缺点： 依赖于对推导规则的熟练使用

7.2.1

函数依赖

- 如何寻找一个关系模式 $R(U, F)$ 的关键字？
 - **方法二：**根据关键字的定义及属性集闭包的概念，计算能够满足条件 $(K_F^+ = U)$ 的最小属性集合 K

算法7-1, 7-2

- **方法三：**先计算函数依赖集 F 的最小覆盖（即与 F 相等价的最小函数依赖集），然后根据函数依赖的特性以及关键字的定义来寻找关系的关键字

可缩短算法7-2的计算过程

7.2.1

函数依赖

习题：寻找下述关系模式的关键字

(1) $R(A, B, C, D)$,

$F: \{ B \rightarrow D, AB \rightarrow C \}$

解：

- 由 $B \rightarrow D$ 及增广规则 R_2 可得： $AB \rightarrow AD$ (1)
- 由 (1), $AB \rightarrow C$ 及合并规则 R_5 可得：
 $AB \rightarrow ACD$ (2)
- 由 (2) 及增广规则 R_2 可得：
 $AB \rightarrow ABCD$

解

7.2.1

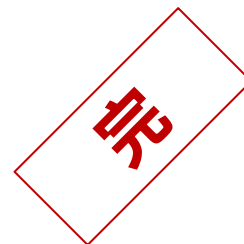
函数依赖

习题：寻找下述关系模式的关键字

(2) $R(A, B, C), \quad F: \{A \rightarrow B, B \rightarrow A, A \rightarrow C\}$

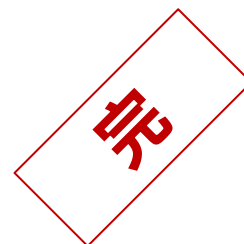
解1：

- 由 $A \rightarrow B, A \rightarrow C$ 及合并规则 R_5 可得： $A \rightarrow BC$
- 由 $A \rightarrow BC$ 及增广规则 R_2 可得： $A \rightarrow ABC$



解2：

- 由 $B \rightarrow A, A \rightarrow C$ 及传递规则 R_3 可得： $B \rightarrow C$
- 由 $B \rightarrow A, B \rightarrow C$ 及合并规则 R_5 可得： $B \rightarrow AC$
- 由 $B \rightarrow AC$ 及增广规则 R_2 可得 $B \rightarrow ABC$



7.2.1

函数依赖

习题：寻找下述关系模式的关键字

(3) $R(A, B, C, D)$,

$F: \{ A \rightarrow C, D \rightarrow B \}$

解：

- 由 $A \rightarrow C$ 及增广规则 R_2 可得： $AD \rightarrow CD$ (1)
- 由 $D \rightarrow B$ 及增广规则 R_2 可得： $AD \rightarrow AB$ (2)
- 由 (1), (2) 及合并规则 R_5 可得： $AD \rightarrow ABCD$

解

7.2.1

函数依赖

习题：寻找下述关系模式的关键字

(4) $R(A, B, C, D)$,

$F: \{ A \rightarrow C, CD \rightarrow B \}$

解：

- 由 $A \rightarrow C$ 及增广规则 R_2 可得： $AD \rightarrow CD \dots\dots\dots (1)$
- 由 (1), $CD \rightarrow B$ 及传递规则 R_3 可得： $AD \rightarrow B$
- 由 $AD \rightarrow B$ 及增广规则 R_2 可得：
 $AD \rightarrow AB \dots\dots\dots (2)$
- 由 (1), (2) 及合并规则 R_5 可得： $AD \rightarrow ABCD$

解

7.2.1

函数依赖

- 属性集的闭包 X_F^+ (可以简写为 X^+)
 - 设 F 是关系模式 $R(U)$ 上的函数依赖集, X 是关系模式 $R(U)$ 的属性子集, 由所有函数依赖于 X 的属性所构成的集合被称为属性集 X 在函数依赖集 F 上的闭包。

$$X^+ = \{ A \mid F \models X \rightarrow A \}$$

7.2.1

函数依赖

算法7-1: 计算属性集 X 在函数依赖集 F 上的闭包 X_F^+
(简写为 X^+)

$X^+ := X;$

repeat

$oldX^+ := X^+;$

for each functional dependency $Y \rightarrow Z$ in F do

if $Y \subseteq X^+$ then $X^+ := X^+ \cup Z;$

until ($oldX^+ = X^+$)

7.2.1

函数依赖

例： 设 $F = \{ (f_1) B \rightarrow CD, (f_2) AD \rightarrow E, (f_3) B \rightarrow A \}$, 请计算 $\{B\}_F^+$?

初始化： $\{B\}_F^+ = \{B\}$

第一遍循环

1. $X = \{B\}_F^+ = \{B\}$
2. f_1 的决定因素是 $\{B\}_F^+$ 的一个子集, 所以
$$\{B\}_F^+ = \{B\}_F^+ \cup \{C, D\} = \{B, C, D\}$$
3. f_2 的决定因素不是 $\{B\}_F^+$ 的子集, 因此 f_2 不影响本次循环的计算结果
4. f_3 的决定因素是 $\{B\}_F^+$ 的一个子集, 所以
$$\{B\}_F^+ = \{B\}_F^+ \cup \{A\} = \{A, B, C, D\}$$
5. $X \neq \{B\}_F^+$, 回到步骤1)开始执行第二遍循环

7.2.1

函数依赖

$$F = \{ (f_1) B \rightarrow CD, (f_2) AD \rightarrow E, (f_3) B \rightarrow A \}$$

第二遍循环

1. $X = \{B\}_F^+ = \{A, B, C, D\}$
2. 跳过在第一遍循环中已经处理过的函数依赖(f_1 和 f_3)
3. f_2 的决定因素是 $\{B\}_F^+$ 的子集, 所以

$$\{B\}_F^+ = \{B\}_F^+ \cup \{E\} = \{A, B, C, D, E\}$$

4. $X \neq \{B\}_F^+$, 回到步骤1)开始执行第三遍循环

7.2.1

函数依赖

$$F = \{ (f_1) B \rightarrow CD, (f_2) AD \rightarrow E, (f_3) B \rightarrow A \}$$

第三遍循环

1. $X = \{B\}_F^+ = \{ A, B, C, D, E \}$
2. F 中的所有函数依赖都已处理过（其依赖因素都已经被并入到 $\{B\}_F^+$ 中
3. 因此在本次循环中 $\{B\}_F^+$ 将不再发生变化，算法执行结束

返回 $\{B\}_F^+$

7.2.1

函数依赖

- **关键字 K 与属性集闭包 X_F^+ 概念之间的关系**

- 设 F 是关系模式 $R(U)$ 上的函数依赖集, K 是关系模式 $R(U)$ 的一个关键字, 则:

- $K_F^+ = U$

- 对于 K 的任意一个真子集 Z , 都有:

$$Z_F^+ \neq U$$

7.2.1

函数依赖

习题7.6 – (1): 寻找下述关系模式的关键字

(4) $R(A, B, C, D)$,

$F: \{ B \rightarrow D, AB \rightarrow C \}$

解: (方法一的计算过程)

- 由 $B \rightarrow D$ 及增广规则 R_2 可得: $AB \rightarrow AD$ (1)
- 由 (1), $AB \rightarrow C$ 及合并规则 R_5 可得:
- $AB \rightarrow ACD$ (2)
- 由 (2) 及增广规则 R_2 可得: $AB \rightarrow ABCD$
- 因为: (对方法一计算结果的验证)

$$\{A\}_F^+ = \{A\} \neq \{A, B, C, D\}$$

$$\{A\}_F^+ = \{B, D\} \neq \{A, B, C, D\}$$

所以 (A, B) 是关系模式 R 的一个关键字。

解

7.2.1

函数依赖

算法7-2: 寻找关系模式 $R(U, F)$ 的关键字 K

1. set $K := R$;
2. for each attribute A in K
 - {
 compute $(K - A)_F^+$;
 if $(K - A)_F^+$ contains all the attributes in R , then
 {
 set $K := K - \{A\}$;
 }
}

7.2.1

函数依赖

习题7.6 – (2): 寻找下述关系模式的关键字

$R(A, B, C), \quad F: \{A \rightarrow B, B \rightarrow A, A \rightarrow C\}$

解1: $K = \{A, B, C\}$

$\because \{K-A\}^+ = \{A, B, C\} = U \quad \therefore K = K-A = \{B, C\}$
 $\because \{K-B\}^+ = \{C\} \neq U \quad \therefore \text{该关键字中必定含有属性 } B$
 $\because \{K-C\}^+ = \{A, B, C\} = U \quad \therefore K = K-C = \{B\}$
最后得到该关系的一个关键字 $\{B\}$

解2: $K = \{A, B, C\}$

$\because \{K-B\}^+ = \{A, B, C\} = U \quad \therefore K = K-B = \{A, C\}$
 $\because \{K-A\}^+ = \{C\} \neq U \quad \therefore \text{该关键字中必定含有属性 } A$
 $\because \{K-C\}^+ = \{A, B, C\} = U \quad \therefore K = K-C = \{A\}$
最后得到该关系的另一个关键字 $\{A\}$

解

解

7.2.1

函数依赖

习题7.6 – (4): 寻找下述关系模式的关键字

$R(A, B, C, D), \quad F: \{B \rightarrow D, AB \rightarrow C\}$

解1: $K = \{A, B, C\}$

$$\because (K - \{A\})^+ = (\{B, C, D\})^+ = \{B, C, D\} \neq U$$

\therefore 该关键字中必定含有属性 A

$$\because (K - \{B\})^+ = (\{A, C, D\})^+ = \{A, B, C, D\} = U$$

$$\therefore K = K - \{B\} = \{A, C, D\}$$

$$\because (K - \{C\})^+ = (\{A, D\})^+ = \{A, B, C, D\} = U$$

$$\therefore K = K - \{C\} = \{A, D\}$$

$$\because (K - \{D\})^+ = (\{A\})^+ = \{A, C\} \neq U$$

\therefore 该关键字中必定含有属性 D

最后得到该关系的一个关键字 $\{A, D\}$

解

7.2.1

函数依赖

习题7.6:

序号	关系模式	函数依赖集	关键字	主属性集	非主属性集
1	R(A,B,C,D)	$B \rightarrow D$ $AB \rightarrow C$	{ A,B }	A,B	C,D
2	R(A,B,C)	$A \rightarrow B$ $A \rightarrow C$	{ A }	A	B,C
3	R(A,B,C,D)	$A \rightarrow C$ $D \rightarrow B$	{ A,D }	A,D	B,C
4	R(A,B,C,D)	$A \rightarrow C$ $CD \rightarrow B$	{ A,D }	A,D	B,C



7.2.1 函数依赖

7.2.2 与函数依赖有关的范式

- 范式及模式分解方法

- 1NF
- 2NF
- 3NF
- BCNF

7.2.3 多值依赖与第四范式

7.2.2

与函数依赖有关的范式

- **定义：第一范式 (1NF)**

- 如果关系模式 $R(U)$ 中的每个属性值都是一个不可分割的数据量，则称该关系模式满足第一范式，并记为： $R \in 1NF$

- **定义7.9：第二范式 (2NF)**

- 设有关系模式 $R(U) \in 1NF$ ，且其每个非主属性都完全函数依赖于关键字，则称关系模式 $R(U)$ 满足第二范式，并记作： $R \in 2NF$

7.2.2

与函数依赖有关的范式

例：有一个学生关系 SCG(S[#], S_n, S_d, S_s, C[#], G), 根据用户给定的语义约束得到如下的函数依赖关系：

- 基本常识：

$$S^{\#} \rightarrow S_n$$

- 每个学生均只属一个系与一个专业；

$$S^{\#} \rightarrow S_d$$

$$S^{\#} \rightarrow S_s$$

- 每个学生修读之每门课有且仅有一个成绩；

$$(S^{\#}, C^{\#}) \rightarrow G$$

- 各系无相同专业。

$$S_s \rightarrow S_d$$

7.2.2

与函数依赖有关的范式

- 关系模式 SCG ($S^\#$, S_n , S_d , S_s , $C^\#$, G), 函数依赖集为:

$S^\# \rightarrow S_n, S^\# \rightarrow S_d, S^\# \rightarrow S_s, (S^\#, C^\#) \rightarrow G, S_s \rightarrow S_d$

该关系模式是否满足第二范式的要求?

- 解题方法如下:

- 找出该关系模式的所有(候选)关键字
- 据此确定该关系的主属性集和非主属性集
- 判断每一个非主属性和关键字之间的函数依赖关系是否满足2NF的要求
 - 如果不存在非主属性对关键字的部分依赖, 那么 $SCG \in 2NF$

7.2.2

与函数依赖有关的范式

- 关系模式 SCG ($S^\#$, S_n , S_d , S_s , $C^\#$, G), 函数依赖集为:

$S^\# \rightarrow S_n, S^\# \rightarrow S_d, S^\# \rightarrow S_s, (S^\#, C^\#) \rightarrow G, S_s \rightarrow S_d$

- 找出该关系模式的所有(候选)关键字

$(S^\#, C^\#)$

- 据此确定该关系的主属性集和非主属性集

主属性集: $\{S^\#, C^\#\}$

非主属性集: $\{S_n, S_d, S_s, G\}$

- 据此确定该关系的主属性集和非主属性集

- 存在非主属性对关键字的部分依赖, 如: $(S^\#, C^\#) \rightarrow S_n$
- 因此, 该关系不满足2NF (只能满足到1NF)

7.2.2

与函数依赖有关的范式

- 采用SCG ($S^\#$, S_n , S_d , S_s , $C^\#$, G)的模式设计方案（仅满足1NF的关系模式）会带来什么样的问题？
 - 数据冗余
 - 因数据冗余而产生更新异常
 - 为什么出现数据冗余现象？
 - 存在非主属性对关键字的部分函数依赖
 - 如何避免上述的数据冗余现象？
 - 按照更高范式的要求设计关系模式：模式分解

7.2.2

与函数依赖有关的范式

S#	Sn	Sd	Ss	C#	G
0001	王剑飞	CS	软件	101	5
0001	王剑飞	CS	软件	102	5
0001	王剑飞	CS	软件	103	4
0002	张 军	CS	应用	101	3
0002	张 军	CS	应用	103	4
0003	陈 瑛	CS	应用	103	3
0003	陈 瑛	CS	应用	104	3
0004	方世觉	CS	软件	103	4

图1: 根据关系模式SCG所建立的数据库 (仅满足1NF)

7.2.2

与函数依赖有关的范式

- 为了使最终的模式设计结果能够满足到2NF，我们可以对关系SCG进行模式分解，将其属性集合分解构成若干个小的关系模式。
 - 随着对原有关系模式的分解，原来在关系模式SCG上存在的函数依赖集合也被分解到若干个小的关系模式上去。
- 模式分解的目标
 - 使得分解得到的每个小的关系模式都能够满足2NF的要求。
 - 从而消除因非主属性对关键字的部分函数依赖而产生的数据冗余现象。

7.2.2

与函数依赖有关的范式

- 模式分解

- 设在关系模式R上成立的函数依赖集为F, Head(R) 是由关系R中的所有属性所构成的属性集合。
- 如果存在一组子关系模式 $\{ R_1, R_2, \dots, R_k \}$ 满足下述的两个条件, 则我们称 $\{ R_1, R_2, \dots, R_k \}$ 是关系模式R的一个分解 (Decomposition)
 - $\text{Head}(R) = \text{Head}(R_1) \cup \text{Head}(R_2) \cup \dots \cup \text{Head}(R_k)$
 - 设子关系 R_i 上的函数依赖集为 F_i ($i=1,2,\dots,k$), 则:

$$F_i = \{ X \rightarrow Y \mid X \rightarrow Y \in F^+ \text{ 且 } (X \cup Y) \subseteq \text{Head}(R_i) \}$$

7.2.2

与函数依赖有关的范式

- 关系的规范化过程就是一个不断进行模式分解的过程。
- 通过模式分解可以消除原有关系模式中那些不好的函数依赖关系
(即不满足更高范式要求的函数依赖关系)，以尽可能地解决原有模式中所存在的数据冗余和各种插入、删除异常现象。

7.2.2

与函数依赖有关的范式

- 模式分解的方法

- 设存在一个关系模式R, 其属性集合为Head(R), F是其函数依赖集。将其分解到满足范式M的步骤如下:

1. 找出所有不满足范式M要求的函数依赖关系
2. 选择一个不符合要求的函数依赖关系作如下的分解:

- 假设 $X \xrightarrow{f} Y \in F^+$ 且不满足范式M的要求, 则我们将关系模式R分解为如下的两个子关系:

- $R_1 (X \cup Y, \{X \rightarrow Y\})$
- $R_2 (\text{Head}(R) - Y, F_2)$, 其中:
 $F_2 = \{ A \rightarrow B \mid A \rightarrow B \in F^+ \text{ 且 } (A \cup B) \subseteq \text{Head}(R_2) \}$

7.2.2

与函数依赖有关的范式

- 模式分解的方法 (cont.)

3. 对于分解得到的子关系模式 R_2 重复上述的步骤1)和步骤2), 直到所有的子关系模式都能满足范式 M 的要求
4. 合并那些具有相同关键字的子关系模式

7.2.2

与函数依赖有关的范式

- 利用前面介绍的分解方法对关系模式SCG进行规范化（分解到满足2NF）

函数依赖集F

$f_1: S^{\#} \rightarrow S_n$

$f_2: S^{\#} \rightarrow S_d$

$f_3: S^{\#} \rightarrow S_s$

$f_4: (S^{\#}, C^{\#}) \rightarrow G$

$f_5: S_s \rightarrow S_d$

关键字K

$(S^{\#}, C^{\#})$

主属性集

$\{S^{\#}, C^{\#}\}$

非主属性集

$\{S_n, S_d, S_s, G\}$

- 不符合2NF要求的函数依赖关系具有以下特征：

- $X \xrightarrow{f} Y$, Y是关系SCG的非主属性，而X则是其某个候选关键字的真子集

7.2.2

与函数依赖有关的范式

• 分解过程(1)

- 在关系SCG($U=\{S\#, S_n, S_d, S_s, C\#, G\}$)中找出所有不满足2NF要求的函数依赖

$$f_1: S\# \rightarrow S_n \quad f_2: S\# \rightarrow S_d \quad f_3: S\# \rightarrow S_s$$

- 从中选择一个函数依赖(f_1)进行模式分解, 分解结果如下:

$$R_1 (U_1 = \{ S\#, S_n \}, F_1 = \{ f_1: S\# \rightarrow S_n \})$$

$$R_0 (U_0 = U - \{ S_n \} = \{ S\#, S_d, S_s, C\#, G \},$$

$$F_0 = \{ f_2: S\# \rightarrow S_d, f_3: S\# \rightarrow S_s, f_4: (S\#, C\#) \rightarrow G, f_5: S_s \rightarrow S_d \})$$

7.2.2

与函数依赖有关的范式

• 分解过程(2)

- 在关系 $R_0(U_0=\{S^\#, S_d, S_s, C^\#, G\}, F_0)$ 中找出所有不满足2NF要求的函数依赖

$$f_2: S^\# \rightarrow S_d \quad f_3: S^\# \rightarrow S_s$$

- 从中选择一个函数依赖(f_2)进行模式分解, 分解结果如下

$$R_2(U_2 = \{ S^\#, S_d \}, F_2 = \{ f_2: S^\# \rightarrow S_d \})$$

$$R_0(U_0 = U - \{ S_d \} = \{ S^\#, S_s, C^\#, G \},$$

$$F_0 = \{ f_3: S^\# \rightarrow S_s, f_4: (S^\#, C^\#) \rightarrow G, \quad)$$

7.2.2

与函数依赖有关的范式

• 分解过程(3)

- 在关系 $R_0(U_0=\{S^\#, S_s, C^\#, G\}, F_0)$ 中找出所有不满足2NF要求的函数依赖

$$f_3: S^\# \rightarrow S_s$$

- 从中选择一个函数依赖(f_2)进行模式分解, 分解结果如下

$$R_3(U_3 = \{ S^\#, S_s \}, F_3 = \{ f_3: S^\# \rightarrow S_s \})$$

$$R_0(U_0 = U - \{ S_s \} = \{ S^\#, C^\#, G \},$$

$$F_0 = \{ f_4: (S^\#, C^\#) \rightarrow G,)$$

7.2.2

与函数依赖有关的范式

- 综合前面的分解结果

$$R_1 (U_1 = \{ S^\#, S_n \}, F_1 = \{ f_1: S^\# \rightarrow S_n \})$$

$$R_2 (U_2 = \{ S^\#, S_d \}, F_2 = \{ f_2: S^\# \rightarrow S_d \})$$

$$R_3 (U_3 = \{ S^\#, S_s \}, F_2 = \{ f_3: S^\# \rightarrow S_s \})$$

$$R_0 (U_0 = U - \{ S_s \} = \{ S^\#, C^\#, G \},$$

$$F_0 = \{ f_4: (S^\#, C^\#) \rightarrow G \})$$

所有的子关系模式都已经满足2NF

7.2.2

与函数依赖有关的范式

- 合并关键字相同的子关系模式后得到如下的分解结果

- $SCG_1 (S^{\#}, C^{\#}, G)$, 其函数依赖集是:

$$\{ f_4: (S^{\#}, C^{\#}) \rightarrow G \}$$

- $SCG_2 (S^{\#}, S_n, S_d, S_s)$, 其函数依赖集是:

$$\{ f_1: S^{\#} \rightarrow S_n,$$

$$f_2: S^{\#} \rightarrow S_d,$$

$$f_3: S^{\#} \rightarrow S_s,$$

$$f_5: S_s \rightarrow S_d \}$$

7.2.2

与函数依赖有关的范式

S#	C#	G
0001	101	5
0001	102	5
0001	103	4
0002	101	3
0002	103	4
0003	103	3
0003	104	3
0004	103	4

关系SCG₁

S#	Sn	Sd	Ss
0001	王剑飞	CS	软件
0002	张 军	CS	应用
0003	陈 瑛	CS	应用
0004	方世觉	CS	软件

关系SCG₂

图2：根据关系模式SCG1和SCG2所建立的数据库（可以满足到 2NF）

7.2.2

与函数依赖有关的范式

S#	Sn	Sd	Ss
0001	王剑飞	CS	软件
0002	张 军	CS	应用
0003	陈 瑛	CS	应用
0004	方世觉	CS	软件

关系SCG₂

- 在关系SCG2中仍然存在数据冗余（**专业Ss与系别Sd之间的对应关系**），以及因此而产生的更新异常现象
- 原因：**存在非主属性对关键字的传递依赖**

7.2.2

与函数依赖有关的范式

- **定义7.10: 第三范式 (3NF)**

- 设有关系模式 $R(U) \in 2NF$ ，且其每个非主属性都不传递函数依赖于关键字，则称关系模式 $R(U)$ 满足第三范式，并记作： $R \in 3NF$
- 不符合3NF要求的函数依赖关系具有如下特征：
 - $X \xrightarrow{f} Y$ ， Y 是关系SCG的非主属性，而 X 并不是关键字，或者 X 只是关键字的一个真子集

7.2.2

与函数依赖有关的范式

- 分解到满足3NF

$SCG_1 (S^\#, C^\#, G), \{ (S^\#, C^\#) \rightarrow G \}$

- 已满足3NF

$SCG_2 (\{ S^\#, S_n, S_d, S_s \}$
 $\{ S^\# \rightarrow S_n, S^\# \rightarrow S_s, S_s \rightarrow S_d \})$

- 关键字: $S^\#$
- 存在非主属性 S_d 对关键字的传递依赖:

$S^\# \rightarrow S_s, S_s \rightarrow S_d \not\models S^\# \rightarrow S_d$

7.2.2

与函数依赖有关的范式

- 将SCG₂分解到满足3NF

SCG₂ (S[#], S_n, S_d, S_s)的分解结果如下:

- SCG₂₁ (S[#], S_n, S_s), 其函数依赖集是:
{ S[#] → S_n, S[#] → S_s }
- SCG₂₂ (S_d, S_s), 其函数依赖集是:
{ S_s → S_d }

7.2.2

与函数依赖有关的范式

S#	C#	G
0001	101	5
0001	102	5
0001	103	4
0002	101	3
0002	103	4
0003	103	3
0003	104	3
0004	103	4

关系SCG₁

S#	Sn	Ss
0001	王剑飞	软件
0002	张 军	应用
0003	陈 瑛	应用
0004	方世觉	软件

关系SCG₂₁

Sd	Ss
CS	软件
CS	应用

关系SCG₂₂

图3: 符合3NF要求的模式设计

7.2.2

与函数依赖有关的范式

- **定义7.11: BCNF**

- 设关系模式 $R(U)$ 满足 1NF，且若 $X \rightarrow Y$ 时 X 必包含有该关系模式的关键字，则称关系模式 $R(U)$ 满足BCNF范式，并记作：

$R \in \text{BCNF}$

7.2.2

与函数依赖有关的范式

例 7.1 设有关系模式 $R(S, C, T)$ ，其中 S, C 含义同前, T 表示教师, R 有下列语义信息:

- 每个教师仅上一门课

$$T \rightarrow C$$

- 每个学生均只属一个系与一个专业;

$$(S, C) \rightarrow C$$

思考题:

1. 关系模式 R 的关键字是什么?
2. 关系模式 R 的主属性集是什么? 非主属性集又是什么?
3. 该关系模式 R 最高能够满足到第几范式?

7.2.2

与函数依赖有关的范式

思考题答案

1. 有两个候选关键字：{ S, C } 和 { S, T }
2. 主属性集是：{ S, C, T }
3. 关系R最高能够满足到第几范式？

$R(\{S, C, T\}, \{T \rightarrow C, (S, C) \rightarrow T\})$

候选关键字：{ S, C } 和 { S, T }

- 其主属性集为{S, C, T}，非主属性集为空（不存在非主属性），因此该关系模式中不存在不满足2NF和3NF范式要求的函数依赖，因此该关系模式一定可以满足到3NF。

7.2.2

与函数依赖有关的范式

- 关系模式R是否能够满足BCNF?

$R(\{S, C, T\}, \{T \rightarrow C, (S, C) \rightarrow T\})$

候选关键字: $\{S, C\}$ 和 $\{S, T\}$

- 存在不满足BCNF要求的函数依赖:

$T \rightarrow C$

因此不满足BCNF

- 分解到满足BCNF范式

$R_1(C, T)$ 函数依赖集: $\{T \rightarrow C\}$

$R_2(S, T)$ 函数依赖集: $\{ \}$

7.2.2

与函数依赖有关的范式

- **定理7-1**

- 如果关系模式 $R(U) \in \text{BCNF}$, 则 $R(U) \in 3\text{NF}$

- **证明：**假设 $R(U) \notin 3\text{NF}$, 则有三种可能性：

1. $R(U) \notin 1\text{NF}$
2. $R(U) \notin 2\text{NF}$, 即：存在一个非主属性部分依赖于关键字
3. $R(U) \notin 3\text{NF}$, 即：存在一个非主属性传递依赖于关键字

7.2.2.1

定理7-1之证明

1. 假设 $R(U) \notin 1NF$

- 由 $R(U) \in BCNF$ 可知: $R(U) \in 1NF$ 。与假设相矛盾。

2. 假设存在一个非主属性 A 部分依赖于关键字 K , 即:

$$K \xrightarrow{p} A \quad (A \notin K)$$

- 由部分依赖的定义可知: 必存在 K 的某个真子集 W , 且满足: $W \rightarrow A$
($A \notin W$)
- 由 $R(U) \in BCNF$ 及 $BCNF$ 定义可知: W 中必含有关键字, 即关键字 K 中含有另一个关键字 W , 且 $W \subset K$, 这与关键字的定义相矛盾。

7.2.2.1

定理7-1之证明

3. 假设假设存在一个非主属性 A 传递依赖于关键字 K , 即存在一个属性集合 W , 并满足:

$$K \rightarrow W, W \not\subseteq K, W \rightarrow K, W \rightarrow A$$

- 由 $W \rightarrow A$ 及 $R(U) \in \text{BCNF}$ 可知: W 中必含有关键字 Z
- 由关键字的定义可知: $Z \rightarrow U$
- 因为 $Z \subseteq W, K \subseteq U$, 故 $W \rightarrow K$ 。
- 这与 $W \rightarrow K$ 相矛盾。

综上所述, 假设不成立, 即 $R(U) \in 3\text{NF}$

证毕

7.2.1 函数依赖

7.2.2 与函数依赖有关的范式

7.2.3 多值依赖与第四范式（略）

- 多值依赖
 - 与MVD有关的推理规则
- 4NF

7.2.3

多值依赖与第四范式

- 一个多值依赖的例子

例子7.2: 设有一个课程关系R，其示意图用表7.5表示。此表表示‘高等数学’这门课(C)的任课教师(T)有3个，它的参考书(L)可以有2本；‘普通物理’这门课(C)的任课教师(T)也有3个，它的参考书(L)可以有3本。

课程名C	教师名T	选用参考书L
高等数学	李华民	高等数学
	王天华	高等数学教程
	林 静	
普通物理	吴铁钢	物理学
	谢晓芳	普通物理
	徐秋芳	普通物理基础

表7.5 关系R的示意图

C	T	L
高等数学	李华民	高等数学
高等数学	李华民	高等数学教程
高等数学	王天华	高等数学
高等数学	王天华	高等数学教程
高等数学	林 静	高等数学
高等数学	林 静	高等数学教程
普通物理	吴铁钢	物理学
普通物理	吴铁钢	普通物理
普通物理	吴铁钢	普通物理基础
普通物理	谢晓芳	物理学
普通物理	谢晓芳	普通物理
普通物理	谢晓芳	普通物理基础
普通物理	徐秋芳	物理学
普通物理	徐秋芳	普通物理
普通物理	徐秋芳	普通物理基础

表7.6 关系 R

- **如果将表7.5用标准的关系形式来表示，如表7.6所示。**

- **缺点**

- **数据大量冗余存储**

- **特点**

- **一个C的值与一组T的值相对应**
- **这组T的值与其它属性(L)的值无关**

• 特点

1) 一个 C 的值与一组 T 的值相对应

‘高等数学’对应 { ‘李华民’, ‘王天华’, ‘林静’ }

‘普通物理’对应 { ‘吴铁钢’, ‘谢晓芳’, ‘徐秋芳’ }

2) 这组 T 的值与该关系中的其它属性 (L) 的值无关

- 即：设 t_1, t_2 是关系 R 中的两个元组 ($t_1 \in R, t_2 \in R$), 且 $t_1[C] = t_2[C]$, 则互换这两个元组在属性 L 上的取值所构成的两个新的元组:

$t_3: (t_1[C], t_1[T], t_2[L])$

$t_4: (t_2[C], t_2[T], t_1[L])$

也必定是关系 R 中的元组 (即: $t_3 \in R, t_4 \in R$)

- 在这里并没有区别 t_1, t_2, t_3, t_4 是不是相同的元组。

7.2.3

多值依赖与第四范式

【定义7-12】 多值依赖(MVD)

设有关系模式 $R(U)$ ， X, Y 是 U 的子集($X, Y \subseteq U$)，若对 $R(U)$ 的任何一个关系 r ，对 X 的一个确定值，存在 Y 的一组值与之对应，且 Y 的这组值又与 $Z=U-X-Y$ 中的属性值不相关，此时称 Y 多值依赖于 X ，并记为： $X \twoheadrightarrow Y$

- 多值依赖产生的原因

- 在一个关系模式中，若存在两个相互独立的属性之间的‘一对多’函数对应关系，如：
 - C 与 T 的‘一对多’函数对应关系
 - C 与 L 之间的‘一对多’函数对应关系
- 而T和L之间又没有任何依赖关系，此时若把它们合并起来构成一个关系，即 $R(C, T, L)$ ，则就会产生多值依赖现象。

7.2.3

多值依赖与第四范式

C	T	L
高等数学	李华民	高等数学
高等数学	李华民	高等数学教程
高等数学	王天华	高等数学
高等数学	王天华	高等数学教程
高等数学	林 静	高等数学
高等数学	林 静	高等数学教程
.....

在表7.6中我们可以看到:

1. 若存在两个元组(第2和第3个元组), 它们在C上的取值相同(高等数学), 但在T上的取值不同(李华民,王天华)

7.2.3

多值依赖与第四范式

C	T	L
高等数学	李华民	高等数学
高等数学	李华民	高等数学教程
高等数学	王天华	高等数学
高等数学	王天华	高等数学教程
高等数学	林 静	高等数学
高等数学	林 静	高等数学教程
.....

那么：

2. 互换它们在L上的取值(高等数学,高等数学教程)，产生的新元组（元组1和元组4）一定也出现在关系R中。

C	T	L
高等数学	李华民	高等数学
高等数学	李华民	高等数学教程
高等数学	王天华	高等数学
高等数学	王天华	高等数学教程
高等数学	林 静	高等数学
高等数学	林 静	高等数学教程
普通物理	吴铁钢	物理学
普通物理	吴铁钢	普通物理
普通物理	吴铁钢	普通物理基础
普通物理	谢晓芳	物理学
普通物理	谢晓芳	普通物理
普通物理	谢晓芳	普通物理基础
普通物理	徐秋芳	物理学
普通物理	徐秋芳	普通物理
普通物理	徐秋芳	普通物理基础

表7.6 关系R

- 在表7.6中我们可以看到：
 - 每一门课程C（高等数学）都对应着一组教师T（李华民，王天华，林静）。
 - 那么对于T的每个取值，都必须重复地与同这个C上的值相关的L上的每个值(高等数学,高等数学教程)相匹配，因此在关系R内出现了一个多值属性，产生了数据冗余。

7.2.3

多值依赖与第四范式

课程名C	教师名T	选用参考书L
高等数学	李华民 王天华 林 静	高等数学 高等数学教程
普通物理	吴铁钢 谢晓芳 徐秋芳	物理学 普通物理 普通物理基础

表7.5 关系R的示意图

- 如果如例7.2那样，用表7.5来表示上述的关系，那么就不会出现前述的数据冗余现象。但表7.5是不符合1NF的要求的。
- 因此，在关系数据库中，因为1NF的要求而产生了多值依赖！

7.2.3

多值依赖与第四范式

- 非平凡的多值依赖

【定义】 设在关系模式 $R(U)$ 中, $X \twoheadrightarrow Y$ 且 $U - X - Y \neq \emptyset$, 则称 $X \twoheadrightarrow Y$ 是非平凡的多值依赖, 否则称为平凡的多值依赖。

【例】 在关系模式 $S(S^\#, S_n, S_d)$ 中, 三个属性分别是学号、姓名和学生所在的系。请问:

1. 在该关系模式中是否存在哪些MVD?
2. 如果存在MVD, 请指出是平凡的MVD, 还是非平凡的MVD?

关系模式：S(S[#], S_n, S_d)

- 所有可能存在的多值依赖如下：

1. $S^{\#} \twoheadrightarrow S_n$ ✓

2. $S^{\#} \twoheadrightarrow S_d$ ✓

3. $S_n \twoheadrightarrow S^{\#}$ ✓

4. $S_n \twoheadrightarrow S_d$ ×

5. $S_d \twoheadrightarrow S^{\#}$ ×

6. $S_d \twoheadrightarrow S_n$ ×

7. $S^{\#} \twoheadrightarrow (S_n, S_d)$ ✓

8. $S_n \twoheadrightarrow (S^{\#}, S_d)$ ✓

9. $S_d \twoheadrightarrow (S^{\#}, S_n)$ ✓

10. $(S^{\#}, S_n) \twoheadrightarrow S_d$ ✓

11. $(S^{\#}, S_d) \twoheadrightarrow S_n$ ✓

12. $(S_n, S_d) \twoheadrightarrow S^{\#}$ ✓

关系模式：S(S[#], S_n, S_d)

- 存在的MVD有：

S[#] →→ S_n

S[#] →→ S_d

S[#] →→ (S_n, S_d)

S_n →→ (S[#], S_d)

S_d →→ (S[#], S_n)

(S[#], S_n) →→ S_d

(S[#], S_d) →→ S_n

(S_n, S_d) →→ S[#]

- 其中：

- 红颜色是非平凡多值依赖
- 其它的为平凡多值依赖

7.2.3

多值依赖与第四范式

习题7.2:

- 在关系 $SC(S^#, C^#, G)$ 中, $S^# \twoheadrightarrow C^#$ 正确吗?
请说明其理由。

7.2.3

多值依赖与第四范式

- 多值依赖的性质

- 在一个关系模式 $R(U)$ 中,
 1. 如有 $X \twoheadrightarrow Y$, 则必有 $X \twoheadrightarrow (U-X-Y)$
 2. 如有 $X \rightarrow Y$, 则必有 $X \twoheadrightarrow Y$

7.2.3

多值依赖与第四范式

【思考题】

在学生关系模式 $S(S\#, S_n, S_s, S_d)$ 中，属性之间的语义约束要求同前。请问在该关系中存在哪些**不是函数依赖的非平凡的MVD**？

【答案】

$$S_s \twoheadrightarrow (S\#, S_n)$$

$$S\# \rightarrow S_n$$

$$S\# \rightarrow S_d$$

$$S\# \rightarrow S_s$$

$$S_s \rightarrow S_d$$

7.2.3

多值依赖与第四范式

- 有关函数依赖(FD)和多值依赖(MVD)的推导规则

关系模式 $R(U)$, X, Y, Z, W 是 U 的子集

- 规则 IR_1 : 自反规则
 - 如果 Y 是 X 的子集, 则: $X \rightarrow Y$
- 规则 IR_2 : 增广规则
 - 如果 $X \rightarrow Y$, 则: $XZ \rightarrow YZ$
- 规则 IR_3 : 传递规则
 - 如果 $X \rightarrow Y, Y \rightarrow Z$, 则: $X \rightarrow Z$
- 规则 IR_4 : 求补规则
 - 如果 $X \twoheadrightarrow Y$, 则 $X \twoheadrightarrow (U-X-Y)$

规则 IR₅ (多值依赖的增广规则)

- 如果 $X \twoheadrightarrow Y$ 且 $Z \subseteq W$, 则 $WX \twoheadrightarrow YZ$

证明:

- 设 $t_1 \in R, t_2 \in R$, 如果 $t_1[WX] = t_2[WX]$, 则: (1)
 $t_1[W] = t_2[W]$ (2)
 $t_1[X] = t_2[X]$ (3)
- 由 (2) 和 $Z \subseteq W$ 得: $t_1[Z] = t_2[Z]$ (4)
- 由 (3) 和 $X \twoheadrightarrow Y$ 得:
 $(t_1[X], t_1[Y], t_2[U-X-Y]) \in R$
 $(t_2[X], t_2[Y], t_1[U-X-Y]) \in R$

- 将 W 中的属性从 (U-X-Y) 中移至 X 之前, 将 Z 中的属性从 (U-X-Y) 中移至 Y 之后得:

$$(t_2[W], t_1[X], t_1[Y], t_2[Z], t_2[U-X-Y-W-Z]) \in R$$

$$(t_1[W], t_2[X], t_2[Y], t_1[Z], t_1[U-X-Y-W-Z]) \in R$$

- 由 (2), (4), (5) 可得:

$$(t_1[W], t_1[X], t_1[Y], t_1[Z], t_2[U-X-Y-W-Z]) \in R$$

$$(t_2[W], t_2[X], t_2[Y], t_2[Z], t_1[U-X-Y-W-Z]) \in R$$

即: $(t_1[WX], t_1[YZ], t_2[U-WX-YZ]) \in R$

$$(t_2[WX], t_2[YZ], t_1[U-WX-YZ]) \in R$$

由 (1) 和 (6) 可知: $WX \twoheadrightarrow YZ$

7.2.3

多值依赖与第四范式

- 规则 IR_6 : 多值依赖的传递规则
 - 如果 $X \twoheadrightarrow Y$, $Y \twoheadrightarrow Z$, 则 $X \twoheadrightarrow (Z - Y)$
- 规则 IR_7 : 转换规则
 - 如果 $X \rightarrow Y$, 则 $X \twoheadrightarrow Y$
- 规则 IR_8 : 结合规则
 - 如果 $X \twoheadrightarrow Y$, 且存在另一个属性集合 W 满足:
 $W \cap Y = \phi \quad W \rightarrow Z \quad Y \supseteq Z \quad \text{则 } X \rightarrow Z$

规则 IR₆ (多值依赖的传递规则)

如果 $X \twoheadrightarrow Y$, $Y \twoheadrightarrow Z$, 则 $X \twoheadrightarrow (Z - Y)$

证明: 假设 X, Y, Z 是关系模式 $R(U)$ 的属性子集, 且 $Y = AB, Z = BC$, 则 $D = U - X - Y - Z$ 是该关系中除 X, Y, Z 三部分属性之外的其它属性所构成的集合。因此我们可以将该关系中的属性分为五个属性子集: X, A, B, C, D

- 如果 t_1, t_2 是关系 R 的任意两个元组且 $t_1[X] = t_2[X]$, 即:
 $(t_1[X], t_1[AB], t_1[C], t_1[D]) \in R \dots\dots\dots (1)$
 $(t_2[X], t_2[AB], t_2[C], t_2[D]) \in R \dots\dots\dots (2)$
 $t_1[X] = t_2[X] \dots\dots\dots (3)$
- 由 $X \twoheadrightarrow Y$ (即 $X \twoheadrightarrow AB$) 和 (1), (2), (3) 得:
 $(t_1[X], t_1[AB], t_2[C], t_2[D]) \in R \dots\dots\dots (4)$
 $(t_1[X], t_2[AB], t_1[C], t_1[D]) \in R \dots\dots\dots (5)$

- 由 $Y \twoheadrightarrow Z$ (即 $AB \twoheadrightarrow BC$) 和 (1), (4) :

$$(t_1[X], t_1[AB], t_1[C], t_1[D]) \in R \dots\dots\dots(1)$$

$$(t_1[X], t_1[AB], t_2[C], t_2[D]) \in R \dots\dots\dots(4)$$

交换它们两者在属性集 **X** 和 **D** 上的取值:

$$(t_1[X], t_1[AB], t_1[C], t_2[D]) \in R \dots\dots\dots(6)$$

$$(t_1[X], t_1[AB], t_2[C], t_1[D]) \in R \dots\dots\dots(7)$$

- 由 $Y \twoheadrightarrow Z$ (即 $AB \twoheadrightarrow BC$) 和 (2), (5) :

$$(t_2[X], t_2[AB], t_2[C], t_2[D]) \in R \dots\dots\dots(2)$$

$$(t_2[X], t_2[AB], t_1[C], t_1[D]) \in R \dots\dots\dots(5)$$

交换它们两者在属性集 **X** 和 **D** 上的取值:

$$(t_2[X], t_2[AB], t_2[C], t_1[D]) \in R \dots\dots\dots(6)$$

$$(t_2[X], t_2[AB], t_1[C], t_2[D]) \in R \dots\dots\dots(9)$$

- 调整表达式(1)和(2)的属性排列次序得:

$$(t_1[X], t_1[C], t_1[ABD]) \in R \dots\dots\dots (10)$$

$$(t_2[X], t_2[C], t_2[ABD]) \in R \dots\dots\dots (11)$$

- 由于 $t_1[X] = t_2[X]$, 因此:

- 将表达式(9): $(t_2[X], t_2[AB], t_2[C], t_2[D]) \in R$ 变换为:

$$(t_1[X], t_1[C], t_2[ABD]) \in R \dots\dots\dots (12)$$

- 将表达式(7): $(t_2[X], t_2[AB], t_2[C], t_2[D]) \in R$ 变换为:

$$(t_2[X], t_{12}[C], t_1[ABD]) \in R \dots\dots\dots (13)$$

- 由 $t_1[X] = t_2[X]$ 及 (10), (11), (12), (13) 得: **$X \twoheadrightarrow C$**

规则 IR₈ (结合规则) 如果 $X \twoheadrightarrow Y$, 且存在另一个属性集合W满足:

$$W \cap Y = \phi, W \rightarrow Z, Y \supseteq Z, \text{ 则: } X \rightarrow Z$$

证明: 将关系模式 R(U) 的属性集合 U 划分为五个属性子集: X, (Y-Z), Z, W, P

- 如果 t_1, t_2 是关系R的任意两个元组且 $t_1[X] = t_2[X]$, 即:

$$(t_1[X], t_1[Y-X], t_1[Z], t_1[W], t_1[P],) \in R \dots\dots\dots (1)$$

$$(t_2[X], t_2[Y-X], t_2[Z], t_2[W], t_2[P],) \in R \dots\dots\dots (2)$$

- 由 $X \twoheadrightarrow Y$ 和 (1), (2) 得:

$$(t_1[X], t_1[Y-X], t_1[Z], t_2[W], t_2[P],) \in R \dots\dots\dots (4)$$

- 由 $W \rightarrow Z$ 和 (2), (4) 得: $t_1[Z] = t_2[Z]$

- 由于从 $t_1[X]=t_2[X]$ 可以推出 $t_1[Z]=t_2[Z]$, 所以有: **$X \rightarrow Z$**

7.2.3

多值依赖与第四范式

定义7.13: 第四范式(4NF)

在关系模式 $R(U)$ 中, 如果 $X \twoheadrightarrow Y$ 是非平凡多值依赖, 则 X 必含有关键字, 此时称关系模式 R 满足第四范式, 并记作 $R \in 4NF$ 。

- 第四范式的特点
 - 只允许出现平凡多值依赖
 - 函数依赖要满足BCNF

【例】 将 $R(C, T, L)$ 分解为:

$R_1(C, T) \qquad C \twoheadrightarrow T$

$R_2(C, L) \qquad C \twoheadrightarrow L$

- 从而消除了原关系模式中的数据冗余和异常现象

7.2.4

小结

- 规范化的目的

- 解决插入、删除及修改异常，降低数据冗余度

- 规范化的方法

- 根据各属性间的依赖关系(函数依赖及多值依赖)来构造关系模式。

- 规范化的实现手段

- 模式分解

习题7.3: 是不是规范化最佳的模式结构是最好的结构？为什么？

关系模式的 逆规范化 问题

第7章 目录

-01

概论

-02

规范化理论

-03

规范化所引起的一些问题

7.3.1

函数依赖理论

1. Armstrong 公理系统
2. 函数依赖的逻辑蕴涵
3. 函数依赖集 F 的闭包 F^+
4. 属性集 X 在函数依赖集 F 上的闭包 X_F^+
5. 函数依赖集的等价
 - 如果两个函数依赖集 F_1 和 F_2 的闭包是相等的, 即 $F_1^+ = F_2^+$, 则称函数依赖集 F_1 等价于函数依赖集 F_2 。

例: $F = \{ A \rightarrow C, AC \rightarrow D, E \rightarrow AD, E \rightarrow H \}$
 $G = \{ A \rightarrow CD, E \rightarrow AH \}$ 请问: F与G是否等价?

分析: 要判断两个函数依赖集是否等价, 只要检查它们的FD是否相互逻辑蕴涵。

1. 首先判断 F 中的函数依赖是否为 G 所逻辑蕴涵

对于F 中的每一个函数依赖 $X \rightarrow A$ 均做如下判断:

如果 $A \in X_G^+$, 则 G 逻辑蕴涵 $X \rightarrow A$

2. 再判断 G 中的函数依赖是否为 F 所逻辑蕴涵

对于G 中的每一个函数依赖 $Y \rightarrow B$ 均做如下判断:

如果 $B \in Y_F^+$, 则F逻辑蕴涵 $Y \rightarrow B$

- 如果F中的每一个FD都是G所逻辑蕴涵的, 且G中的每一个FD也都是F所逻辑蕴涵的, 则F和G逻辑等价。

7.3.1

函数依赖理论

1. Armstrong 公理系统
2. 函数依赖的逻辑蕴涵
3. 函数依赖集 F 的闭包 F^+
4. 属性集 X 在函数依赖集 F 上的闭包 X_F^+
5. 函数依赖集的等价
6. 最小函数依赖集
 - 与函数依赖集 F 相等价的所有函数依赖集中的最小者被称为函数依赖集 F 的最小函数依赖集。
 - ‘最小函数依赖集’也被称为‘最小覆盖’(Minimal Cover)

7.3.1

函数依赖理论

- **最小函数依赖集的判定条件：凡是满足下述三个条件的函数依赖集 F 就是其最小函数依赖集**
 - 对于 F 中的每一个FD关系 $X \rightarrow A$ 均作如下判断：
 1. 依赖因素 A 为单个属性；
 2. 令： $F_1 = F - \{X \rightarrow A\}$, 则 $F_1^+ \neq F^+$;
 - 不存在冗余的函数依赖关系
 3. 对于决定因素 X 的每一个真子集 $Y(Y \subset X)$ 均作如下判断：令 $F_2 = F - \{X \rightarrow A\} \cup \{Y \rightarrow A\}$, 则 $F_2^+ \neq F^+$
 - 不存在部分函数依赖关系

7.3.1

函数依赖理论

- 如果 F 中的每一个FD关系 $X \rightarrow A$ 均符合上述三个条件的要求，则 F 是一个最小函数依赖集。
- 其中：条件1并不是必须的，只是为了方便对条件2和条件3的判断。
- 最小函数依赖集的计算算法
 - 中文算法
 - 英文算法

最小覆盖计算的例子

例1: 设 $F = \{ A \rightarrow C, AC \rightarrow D, E \rightarrow AD, E \rightarrow H \}$
请给出 F 的最小函数依赖集。



例2: $F = \{ ABD \rightarrow AC, C \rightarrow BE, AD \rightarrow BF, B \rightarrow E \}$
Give the minimal cover M for the set F of FDs.



例3: $F = \{ a \rightarrow b, b c \rightarrow d, a c \rightarrow d \}$
Give the minimal cover M for the set F of FDs.



例4: $G = \{ a \rightarrow a c, b \rightarrow a b c, d \rightarrow a b c \}$
Give the minimal cover M for the set G of FDs.

7.3.1

函数依赖理论

例： $F = \{ B \rightarrow CD, AD \rightarrow E, B \rightarrow A \}$
 $G = \{ B \rightarrow CDE, B \rightarrow ABC, AD \rightarrow E \}$

请问： F与G是否等价？

计算G的最小函数依赖集

- 使用Armstrong公理系统来证明F与G的等价
- 使用属性集闭包计算算法来证明F与G的等价



7.3.1

函数依赖理论

- 一个函数依赖集 F 的最小函数依赖集可能有若干个，并不具有唯一性。

例：计算函数依赖集 $F = \{ A \rightarrow B, B \rightarrow A, AB \rightarrow C \}$ 的最小函数依赖集。

解：本题主要是判断：

1. $AB \rightarrow C$ 是否为部分函数依赖？
2. 如果是部分函数依赖，那么可以将其简化为哪一个完全函数依赖？

$F = \{ A \rightarrow B, B \rightarrow A, AB \rightarrow C \}$ 判断 $AB \rightarrow C$ 是否为部分函数依赖？如是则需要将其转化为完全函数依赖。

1. 从其决定因素中去掉属性B，计算 $\{A\}_F^+$ ：

$$\{A\}_F^+ = \{A, B, C\} \quad \text{有: } C \in \{A\}_F^+$$

2. 从其决定因素中去掉属性A，计算 $\{B\}_F^+$ ：

$$\{B\}_F^+ = \{A, B, C\} \quad \text{有: } C \in \{B\}_F^+$$

因此，可以用 $A \rightarrow C$ 和(或) $B \rightarrow C$ 来替换原来的 $AB \rightarrow C$

在算法7-3的步骤(4)中，需要消除G中冗余的函数依赖。根据对函数依赖处理顺序的不同可以得到两个不同（但也是相互等价）的最小函数依赖集：

$$G_1 = \{ A \rightarrow B, B \rightarrow A, A \rightarrow C \}$$

$$G_2 = \{ A \rightarrow B, B \rightarrow A, B \rightarrow C \}$$

7.3.2

模式分解的研究

- **关系数据库的规范化实际上就是一个不断进行模式分解的过程。在分解过程中需要研究下列问题：**
 - 无损联接性 (lossless join)
 - 分解后，原关系中的信息不会被丢失
 - 依赖保持性
 - 原有的函数依赖关系在分解后的关系模式上依然存在
- **在满足“无损联接性”与(或)“依赖保持性”的情况下，一个关系模式可以被分解到满足第几范式？**
 - 相关的一些分解算法

7.3.2

模式分解的研究

- 无损联接性

- 设R是一个关系模式，F是关系模式R上的一个函数依赖集， $\rho = \{ R_1, R_2, \dots, R_K \}$ 是对R的一个分解。
- 如果对R中满足F的每一个关系实例 r 都有：

$$r = \pi_{R_1}(r) \bowtie \pi_{R_2}(r) \bowtie \dots \bowtie \pi_{R_K}(r)$$

其中： $\pi_{R_i}(r)$ 为关系实例r在关系模式 R_i 上的投影，即根据 R_i 中的属性对r执行投影运算后的结果。

- 则称该分解 ρ 相对于F是“无损联接分解”，或称分解 ρ 具有无损联接性。

7.3.2

模式分解的研究

- 对于不具备‘无损联接性’的分解来说，关系 r 与分解后的子关系满足：

$$r \subset \pi_{R_1}(r) \bowtie \pi_{R_2}(r) \bowtie \dots \bowtie \pi_{R_k}(r)$$

一个具有‘无损联接性’的模式分解

A	B	C
A1	100	C1
A2	200	C2
a3	300	c3

ABC



A	B
A1	100
A2	200
a3	300

AB

B	C
100	C1
200	C2
300	C3

BC

A	B	C
A1	100	C1
A2	200	C2
a3	300	c3

$AB \bowtie BC$



一个不具备‘无损联接性’的模式分解

A	B	C
A1	100	C1
A2	200	C2
A3	300	C3
a4	200	C4

ABC
↓

A	B
A1	100
A2	200
A3	300
A4	200

AB

B	C
100	C1
200	C2
300	C3
200	C4

BC

A	B	C
A1	100	C1
A2	200	C2
A2	200	C4
A3	300	C3
A4	200	C2
A4	200	C4

AB ∞ BC



7.3.2

模式分解的研究

- 从关系模式ABC（如下左图）到关系模式AB和BC的分解为何具有‘无损联接性’？

A	B	C
A1	100	C1
A2	200	C2
A3	300	C3

ABC

→

A	B
A1	100
A2	200
A3	300

AB

B	C
100	C1
200	C2
300	C3

BC

- 在关系模式ABC中具有如下的函数依赖关系： $B \rightarrow C$
- 因此从模式ABC到模式AB和BC的分解具有无损联接性：

$$ABC \equiv AB \bowtie BC$$

WHY ?

7.3.2

模式分解的研究

- 在满足依赖关系 $B \rightarrow C$ 的前提下向关系ABC插入一个新的元组(a4, 200, c2)，得到一个新的关系ABC(下图左)。从关系ABC到关系AB和关系BC的分解仍然具有无损联接性

A	B	C
A1	100	C1
A2	200	C2
A3	300	C3
A4	200	C2

ABC



A	B
A1	100
A2	200
A3	300
A4	200

AB

B	C
100	C1
200	C2
300	C3

BC

7.3.2

模式分解的研究

- 如果违反依赖关系 $B \rightarrow C$ 的要求强行向关系ABC插入一个新的元组 (a4, 200, c4)，则新的关系ABC（下图左）将不再具有依赖关系 $B \rightarrow C$ ，因而该分解则不再具有无损联接性（WHY？）

A	B	C
A1	100	C1
A2	200	C2
A3	300	C3
A4	200	C2

ABC



A	B
A1	100
A2	200
A3	300
A4	200

AB

B	C
100	C1
200	C2
300	C3
200	C4

BC

7.3.2

模式分解的研究

定理7.3.2.1

如果R的分解为 $\rho = \{R_1, R_2\}$, F为R所满足的函数依赖集合, 分解 ρ 具有无损联接性的充分必要条件是:

$$R_1 \cap R_2 \rightarrow (R_1 - R_2) \text{ 或 } R_1 \cap R_2 \rightarrow (R_2 - R_1)$$

7.3.2

模式分解的研究

设有一个关系模式 $T(A, B, C)$ ，函数依赖集为 F ，请判断下述的分解 $\rho = \{T_1, T_2\}$ 是否具有无损联接性？

$$F = \{A \rightarrow B\},$$

$$\rho = \{T_1(A, B), T_2(A, C)\}$$

$$F = \{A \rightarrow C, B \rightarrow C\}, \quad \rho = \{T_1(A, B), T_2(A, C)\}$$

$$F = \{A \rightarrow B\},$$

$$\rho = \{T_1(A, B), T_2(B, C)\}$$

$$F = \{A \rightarrow B, B \rightarrow C\}, \quad \rho = \{T_1(A, C), T_2(B, C)\}$$

7.3.2

模式分解的研究

- 依赖保持性

- 设 F 是属性集 U 上的函数依赖集, Z 是 U 上的一个子集, F 在 Z 上的投影用 $\pi_Z(F)$ 来表示
- 设存在关系模式 R 的一个分解 $\rho = \{ R_1, R_2, \dots, R_k \}$, F 是 R 上的函数依赖集。如果:

$$F^+ = (\pi_{R_1}(F) \cup \pi_{R_2}(F) \cup \dots \cup \pi_{R_k}(F))^+$$

- 则称分解 ρ 具有依赖保持性

7.3.2

模式分解的研究

- 在必须同时满足无损联接性和依赖保持性的要求下，一个关系模式最高可以被分解到满足第三范式。

算法7-4:

假设存在一个关系模式R，其函数依赖集为F，利用该算法可以将其分解到满足第三范式，同时该分解满足无损联接性和依赖保持性。

假设S是分解所获得的子关系模式的集合

算法7-4:

1. 计算 F 的最小函数依赖集，并用来代替 F 进行下面的模式分解;
2. $S = \phi$;

7.3.3

规范化设计案例

- 在必须同时满足无损联接性和依赖保持性的要求下，一个关系模式最高可以被分解到满足第三范式。

算法7-4:

假设存在一个关系模式R，其函数依赖集为F，利用该算法可以将其分解到满足第三范式，同时该分解满足无损联接性和依赖保持性。

假设S是分解所获得的子关系模式的集合

算法7-4:

1. 计算 F 的最小函数依赖集，并用来代替 F 进行下面的模式分解;
2. $S = \phi$;

7.3.2

模式分解的研究

算法7-4:

3. 对 F 中的每一个函数依赖 $X \rightarrow Y$ 做如下处理:

- 如果: 对于集合 S 中的每一个子关系模式 Z 都有: $X \cup Y \subseteq Z$
- 则由 X 和 Y 构成一个新的子关系加入到集合 S 中, 即: $S = S \cup \text{Heading}(X \cup Y)$

4. 如果关系 R 的每一个候选关键字 K 都没有出现在分解后的子关系模式中, 即: 对集合 S 中的每一个子关系模式 Z 都有: $K \subseteq Z$

则: 从关系 R 中任选一个候选关键字 K , 并由 K 中的属性单独构成一个子关系模式并加入到集合 S 中去, 即: $S = S \cup \text{Heading}(K)$

计算函数依赖集 $F = \{ A \rightarrow B, B \rightarrow C \}$ 的闭包 F^+

$F^+ = \{$

$A \rightarrow A, A \rightarrow B, A \rightarrow C, A \rightarrow AB, A \rightarrow BC,$
 $A \rightarrow AC, A \rightarrow ABC,$
 $B \rightarrow B, B \rightarrow C, B \rightarrow BC,$
 $C \rightarrow C,$
 $AB \rightarrow A, AB \rightarrow B, AB \rightarrow C, AB \rightarrow AB, AB \rightarrow BC,$
 $AB \rightarrow AC, AB \rightarrow ABC$
 $AC \rightarrow A, AC \rightarrow B, AC \rightarrow C, AC \rightarrow AB, AC \rightarrow BC,$
 $AC \rightarrow AC, AC \rightarrow ABC$
 $BC \rightarrow B, BC \rightarrow C, BC \rightarrow BC,$
 $ABC \rightarrow A, ABC \rightarrow B, ABC \rightarrow C, ABC \rightarrow AB,$
 $ABC \rightarrow BC, ABC \rightarrow AC, ABC \rightarrow ABC$

$\}$

例： 设 $F = \{ (f_1) B \rightarrow CD, (f_2) AD \rightarrow E, (f_3) B \rightarrow A \}$ ，请计算 $\{B\}_F^+$ ？

- $\{B\}_F^+ = \{ B \}$

- **第一遍循环**

1) $X = \{B\}_F^+ = \{ B \}$

2) f_1 的决定因素是 $\{B\}_F^+$ 的一个子集，所以

$$\{B\}_F^+ = \{B\}_F^+ \cup \{C, D\} = \{B, C, D\}$$

3) f_2 的决定因素不是 $\{B\}_F^+$ 的子集，因此 f_2 不影响本次循环的计算结果

4) f_3 的决定因素是 $\{B\}_F^+$ 的一个子集，所以

$$\{B\}_F^+ = \{B\}_F^+ \cup \{A\} = \{A, B, C, D\}$$

5) $X \neq \{B\}_F^+$ 回到步骤1)开始执行第二遍循环

例：设 $F = \{ (f_1) B \rightarrow CD, (f_2) AD \rightarrow E, (f_3) B \rightarrow A \}$ 请计算 $\{B\}_F^+$ ？

- $\{B\}_F^+ = \{B\}$

- 第一遍循环

在当前循环开始之前，记下
闭包 $\{B\}_F^+$ 的计算结果

1) $X = \{B\}_F^+ = \{B\}$

2) f_1 的决定因素是 $\{B\}_F^+$ 的一个子集，所以

$$\{B\}_F^+ = \{B\}_F^+ \cup \{C, D\} = \{B, C, D\}$$

3) f_2 的决定因素不是 $\{B\}_F^+$ 的子集，因此 f_2 不影响本次循环的计算结果

4) f_3 的决定因素是 $\{B\}_F^+$ 的一个子集，所以

$$\{B\}_F^+ = \{B\}_F^+ \cup \{A\} = \{A, B, C, D\}$$

5) $X \neq \{B\}_F^+$ 回到步骤1)开始执行第二遍循环

例：设 $F = \{ (f_1) B \}$

**本次循环：依次处理F中的每一个函数依赖
(仅需要处理那些右边的依赖因素尚未被加入到闭包中的函数依赖)**

• $\{B\}_F^+ = \{B\}$

• **第一遍循环**

1) $X = \{B\}_F^+ = \{B\}$

2) f_1 的决定因素是 $\{B\}_F^+$ 的一个子集，所以

$$\{B\}_F^+ = \{B\}_F^+ \cup \{C, D\} = \{B, C, D\}$$

3) f_2 的决定因素不是 $\{B\}_F^+$ 的子集，因此 f_2 不影响本次循环的计算结果

4) f_3 的决定因素是 $\{B\}_F^+$ 的一个子集，所以

$$\{B\}_F^+ = \{B\}_F^+ \cup \{A\} = \{A, B, C, D\}$$

5) $X \neq \{B\}_F^+$ 回到步骤1)开始执行第二遍循环

例：设 $F = \{ (f_1) B \rightarrow CD, (f_2) AD \rightarrow E, (f_3) B \rightarrow A \}$ ，请计算 $\{B\}_F^+$ ？

- $\{B\}_F^+ = \{B\}$

- **第一遍循环**

1) $X = \{B\}_F^+ = \{B\}$

2) f_1 的决定因素是 $\{B\}_F^+$ 的一个子集

$\{B\}_F^+ = \{B\}$

3) f_2 的决定因素

4) f_3 的决定因素是 $\{B\}_F^+$ 的一个子集，所以

$\{B\}_F^+ = \{B\}_F^+ \cup \{A\} = \{A, B, C, D\}$

5) $X \neq \{B\}_F^+$ 回到步骤1)开始执行第二遍循环

本次循环结束后，根据闭包的计算结果是否发生变化来决定是否需要进行一次循环。

7.0

关键字的计算

- 设有个关系模式 $R(U, F)$ ， U 是关系 R 的属性集合， F 是关系上的最小函数依赖集。根据 F 中的函数依赖，可以将属性集合 U 划分为以下的三个子集：
 - 只在函数依赖的左边出现过的属性的集合 U_L （包括没有出现在任何函数依赖中的属性）
 - 只在函数依赖的右边出现过的属性的集合 U_R
 - 在两边都出现的属性的集合 U_A
- 其中：
 - U_L 中的属性是每一个关键字的组成部分
 - U_R 中的属性不可能出现在任何一个关键字中
 - 在关键字计算算法中，只需要针对 U_A 中的属性进行FOR循环计算就可以了。