

# 数据库原理

## The Theory of Database System

### 第四章 关系规范化理论



中国矿业大学计算机学院



中国矿业大学数据库原理精品课程

# 第四章 关系规范化理论

4.1 问题的提出

4.2 函数依赖和范式

4.3 数据依赖的公理系统

4.4 关系模式的分解方法



## 4.4 关系模式的分解

- 模式分解的概念
- 分解的无损连接性判定
- 分解的函数依赖保持性判定
- 关系模式的分解算法



## 4.4.1 模式分解的概念

**模式分解的定义：** 设  $R(U, F)$  是一个关系模式， $\rho = \{R_1(U_1, F_1), \dots, R_k(U_k, F_k)\}$  是  $R(U, F)$  的一组子集，如果有  $U_1 \cup U_2 \cup \dots \cup U_k = U$ ， $F_i$  是  $F$  在  $U_i$  上的投影。则称  $\rho$  是  $R(U)$  的一个分解。



# “ $F_i$ 是 $F$ 在 $U_i$ 上的投影”的确切定义

定义：

函数依赖集合  $\{X \rightarrow Y \mid X \rightarrow Y \in F^+ \wedge XY \subseteq U_i\}$



# 练习

给定关系模式 $R(U, F)$ ,  $U=ABCD$ ,

$F=\{ A \rightarrow BC, C \rightarrow AD \}$

假设 $R$ 分解为 $R_1=ABC$ ,  $R_2=AD$

试求 $R_1$ 和 $R_2$ 的依赖集 $F_1$ 和 $F_2$ 。

分析:  $F_1=\{ A \rightarrow BC, C \rightarrow A, C \rightarrow B \}$

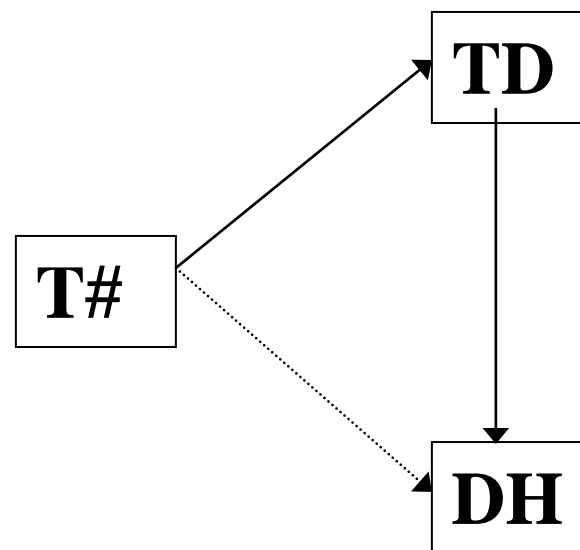
$F_2=\{ A \rightarrow D \}$



## 4.4.2 分解的无损连接性

例：设一关系模式 $R(T\#, TD, DH)$ ，其中 $T\#$ 表示教师编号， $TD$ 表示教师所属系部， $DH$ 表示系主任名。假定每位教师只能在一个系任教，每个系只有一位系主任。

$T\#$	$TD$	$DH$
$T1$	$D1$	$AA$
$T2$	$D1$	$AA$
$T3$	$D2$	$BB$
$T4$	$D3$	$CC$



分解1:

$$\rho_1 = \{R_1(T\#), R_2(TD), R_3(DH)\}$$

分解2:

$$\rho_2 = \{R_1(T\#, TD), R_2(T\#, DH)\}$$

分解3:

$$\rho_3 = \{R_1(T\#, TD), R_2(TD, DH)\}$$

分析：这三种分解那一个最好？





# 分解1:

$R_1$	<u>T#</u>	$R_2$	<u>TD</u>	$R_3$	<u>DH</u>
	T1				AA
	T2	D1			
	T3	D2			BB
	T4	D3			CC

问题：T1是哪一个系的教师？无法回答。  
R1, R2, R3也无法恢复到原来的R。



## 分解2:

R <sub>1</sub>		
	T#	TD
	T1	D1
	T2	D1
	T3	D2
	T4	D3

R <sub>2</sub>		
	T#	DH
	T1	AA
	T2	AA
	T3	BB
	T4	CC

此时，R1，R2的分解是可恢复的，但仍然存在操作异常。原因：TD→DH 在R1，R2中没有体现。



## 分解3:

<b>R<sub>1</sub></b>	<b>T#</b>	<b>TD</b>
	<b>T1</b>	<b>D1</b>
	<b>T2</b>	<b>D1</b>
	<b>T3</b>	<b>D2</b>
	<b>T4</b>	<b>D3</b>

<b>R<sub>2</sub></b>	<b>TD</b>	<b>DH</b>
	<b>D1</b>	<b>AA</b>
	<b>D2</b>	<b>BB</b>
	<b>D3</b>	<b>CC</b>

此时，R1，R2的分解是可恢复的，并且消除了操作异常。



1、分解的无损连接性：如果一个关系模式分解后，可以通过自然连接恢复原模式的信息，这一特性称为分解的无损连接性。

2、判定一个分解的无损连接性的方法：

$\rho = \{R_1(U_1, F_1), \dots, R_k(U_k, F_k)\}$  是  $R(U, F)$  的一个分解， $U = \{A_1, \dots, A_n\}$ ， $F = \{FD_1, FD_2, \dots, FD_p\}$ 。

1) 构造一个  $n$  列  $k$  行的二维表  $T$ 。

$$T_{ij} = \begin{cases} a_j, & \text{如果 } A_j \in R_i \\ b_{ij}, & \text{如果 } A_j \notin R_i \end{cases}$$



2) 根据F中函数依赖修改表T的内容。

**修改规则：**逐个考察F中的每个函数依赖 $X \rightarrow Y$ ，在属性X所在的那些列上找出具有**相同符号的行**，在这些行上使对应于Y的各属性列位置上的符号改为相同，如果其中有一个符号为 $a_j$ ，则把其它符号也**改为 $a_j$** ，否则改为 $b_{mj}$ ，其中 $m$ 是这些行的最小行号。直至在表中发现一行已变成 $a_1a_2...a_k$ ，或表不能再进行修改为止。



3) 如果发现表中有一行已变成 $a_1a_2\dots a_k$ ,  
则表示该分解具有无损连接性, 否则分解不是无损连接的。

**例:** 已知  $R(U, F)$   $U=\{A, B, C, D, E, F\}$ ,  
 $F=\{AB\rightarrow C, C\rightarrow D, A\rightarrow F, D\rightarrow E, D\rightarrow F\}$

$R$ 的一个分解为:

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

判断 $\rho$ 是否具有无损连接性。



# 第一步：建T

A	B	C	D	E	F
<b>a<sub>1</sub></b>	<b>a<sub>2</sub></b>	<b>a<sub>3</sub></b>	<b>b<sub>14</sub></b>	<b>b<sub>15</sub></b>	<b>b<sub>16</sub></b>
<b>b<sub>21</sub></b>	<b>b<sub>22</sub></b>	<b>a<sub>3</sub></b>	<b>a<sub>4</sub></b>	<b>b<sub>25</sub></b>	<b>b<sub>26</sub></b>
<b>b<sub>31</sub></b>	<b>b<sub>32</sub></b>	<b>b<sub>33</sub></b>	<b>a<sub>4</sub></b>	<b>a<sub>5</sub></b>	<b>a<sub>6</sub></b>

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

$$T_{ij} = \begin{cases} a_j, & \text{如果 } A_j \in R_i \\ b_{ij}, & \text{如果 } A_j \notin R_i \end{cases}$$



## 第二步：逐个考察函数依赖，并修改表。

(1) $AB \rightarrow C$ , (2) $C \rightarrow D$ , (3) $A \rightarrow F$ , (4) $D \rightarrow E$ , (5) $D \rightarrow F$

A	B	C	D	E	F
$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$
$b_{21}$	$b_{22}$	$a_3$	$a_4$	$a_5$	$a_6$
$b_{31}$	$b_{32}$	$b_{33}$	$a_4$	$a_5$	$a_6$

因此，该分解具有无损连接性。





## 修改原则：

- 1.有a，有b，向a看齐；
- 2.光有b，但下标不同，则大的向小的看齐。



## 练习

设有关系模式 $R(B, O, I, S, Q, D)$ ，其函数依赖集为：

$F = \{S \rightarrow D, I \rightarrow B, IS \rightarrow Q, B \rightarrow O\}$ ，如果将 $R$ 分解为 $R_1 = SD, R_2 = IB, R_3 = ISQ, R_4 = BO$ ，这样的分解是否具有无损连接性？



# 独立投影法则

如果R只被分解为两个关系模式，则可用更简单的方法进行检验：

**定理：** 设  $\rho = \{R_1(U_1), R_2(U_2)\}$  是  $R(U)$  的一个分解，则  $\rho$  为无损分解的充分必要条件为

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

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



例：  $R(C\#,TN,D)F=\{C\#\rightarrow TN, TN\rightarrow D\}$   $\rho=\{R_1,R_2\}$

$R_1=(C\#,TN), R_2=(TN,D)$

证：  $U_1=\{C\#,TN\}, U_2=\{TN,D\}, (U_1 \cap U_2) = \{TN\},$

$(U_1 - U_2) = \{C\# \}, (U_2 - U_1) = \{D\}$

$\therefore (U_1 \cap U_2) \rightarrow (U_1 - U_2)$  即  $TN \rightarrow C\#$  **不成立**

而  $(U_1 \cap U_2) \rightarrow (U_2 - U_1)$  即  $TN \rightarrow D$  成立

故  $\rho$  是无损分解。

**结论：** 如果两个关系模式间的公共属性集至少包含其中一个关系模式的关键字，则此分解必定具有无损连接性（或连接不失真性）



$$\rho_1 = \{R_1(T\#), R_2(TD), R_3(DH)\}$$

$$\rho_2 = \{R_1(T\#, TD), R_2(T\#, DH)\}$$

$$\rho_3 = \{R_1(T\#, TD), R_2(TD, DH)\}$$

$\rho_1$

T#	TD	DH
a <sub>1</sub>	b <sub>12</sub>	b <sub>13</sub>
b <sub>21</sub>	a <sub>2</sub>	b <sub>23</sub>
b <sub>31</sub>	b <sub>32</sub>	a <sub>3</sub>

$\rho_2$

T#	TD	DH
a <sub>1</sub>	a <sub>2</sub>	b <sub>13</sub>
a <sub>1</sub>	b <sub>22</sub>	a <sub>3</sub>

具有无损连接性

$\rho_3$

T#	TD	DH
a <sub>1</sub>	a <sub>2</sub>	b <sub>13</sub>
b <sub>21</sub>	a <sub>2</sub>	a <sub>3</sub>

具有无损连接性



## 4.4.3 分解的函数依赖保持性

若关系 $R(U, F)$ 的一个分解

$\rho = \{R_1(U_1, F_1), \dots, R_k(U_k, F_k)\}$ 的所有函数依赖

的并集  $(\bigcup_{i=1}^k F_i)$  逻辑蕴涵了 $F$ 中所有函数依赖，

即  $(\bigcup_{i=1}^k F_i)^+ = F^+$ ，则称分解  $\rho$  具有函数依赖保持性。



# 判断分解是否具有函数依赖保持性的方法

例：关系模式 $R(U, F)$   $U=\{A, B, C, D\}$

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

分解 $\rho=\{R_1(A, B), R_2(B, C), R_3(C, D)\}$ 是否具有函数依赖保持性？其中：

$F_1=(A \rightarrow B, B \rightarrow A)$

$F_2=(B \rightarrow C, C \rightarrow B)$

$F_3=(C \rightarrow D, D \rightarrow C)$



# 练习

设有 $R(U, F)$ ，其中， $U=\{A, B, C, D, F\}$ ， $F=\{A \rightarrow C, B \rightarrow C, C \rightarrow D, DF \rightarrow C, CF \rightarrow A\}$ ， $R$ 的一个分解为： $R_1=AB$ ， $R_2=AD$ ， $R_3=AF$ ， $R_4=BF$ ， $R_5=CDF$ ，判断该分解是否具有函数依赖保持性。





## 4.4.4 模式分解的算法

按照上面讨论的模式分解理论，一个模式分解必须满足：

- ①连接不失真性；
- ②依赖保持性；
- ③某一级范式。



# 分解算法1：转换为3NF的保持函数依赖的分解

**输入：** 关系模式R和函数依赖集F

**输出：** 结果为3NF的一个依赖保持分解

**步骤：**

- 1.对R中的函数依赖集F进行极小化处理（处理后的函数依赖集仍记为F）
- 2.找出不在F中出现的属性，把这样的属性构成一个关系模式，把这些属性从U中去掉，剩余的属性仍记为U
- 3.若有 $X \rightarrow A \in F$ ，且 $XA=U$ ，则 $\rho=\{R\}$ ，算法中止



4. 否则，对于F中的每一个 $X \rightarrow A$ ，构成一个关系模式XA。（如果有 $X \rightarrow A_1, X \rightarrow A_2, \dots, X \rightarrow A_n$ （左部相同），则可以用模式 $XA_1A_2 \dots A_n$ 代替n个模式 $XA_1, XA_2, \dots, XA_n$ ）；

5. 算法结束。

例：设有关系模式CTHRSG（C，T，H，R，S，G）及其函数依赖

$F = \{C \rightarrow T, CS \rightarrow G, HT \rightarrow R, HR \rightarrow C, CH \rightarrow R, HS \rightarrow R\}$

请给出3NF的一个依赖保持分解



# 解:

1.进行极小化处理,

$$F=\{C\rightarrow T, CS\rightarrow G, HT\rightarrow R, HR\rightarrow C, HS\rightarrow R\}$$

2.所有属性均在F中出现, 所以没有属性从R中分离出去

3. 根据算法, 本题中六个函数依赖对应于下面的5个关系模式

$$R1=\{C,T\} \quad R2=\{C,S,G\} \quad R3=\{H,R,T\} \quad R4=\{C,H,R\}$$

$$R5=\{H,S,R\}$$

$$\text{所以 } \rho=\{R1,R2,R3,R4,R5\}$$



分解算法2： 结果为3NF， 且具有依赖保持和连接不失真的分解。

(1) 设 $X$ 是 $R(U, F)$ 的码。  $R(U, F)$ 已由前面的算法1分解为 $\rho = \{R_1(U_1, F_1), R_2(U_2, F_2), \dots, R_k(U_k, F_k)\}$ , 令 $\tau = \rho \cup \{R^*(X, F_x)\}$ 。

(2) 若有某个 $U_i$ ,  $X \subseteq U_i$ , 将 $R^*(X, F_x)$ 从 $\tau$ 中去掉

(3)  $\tau$ 就是所求的分解



例：设有关系模式CTHRSG (C, T, H, R, S, G) 及其函数依赖

$F=\{C\rightarrow T, CS\rightarrow G, HT\rightarrow R, HR\rightarrow C, CH\rightarrow R, HS\rightarrow R\}$

给出3NF的一个依赖保持且连接不失真的分解

解：由分解算法1得

$\rho = \{CT, CSG, HRT, CHR, HSR\}$  , KEY=HS

$\tau = \rho \cup \{HS\} = \{CT, CSG, HRT, CHR, HSR, HS\}$

因为HS是HSR的子集

所以 $\tau = \{CT, CSG, HRT, CHR, HSR\}$



# 举例

设有关系模式R (A, B, C, D, E, P), R的函数依赖集为:

$F=\{C \rightarrow B, E \rightarrow D, D \rightarrow B, B \rightarrow D, BC \rightarrow D, DC \rightarrow A\}$ ,  
求R的一个满足3NF的无损连接和函数依赖保持的分解。



(1) 最小函数依赖集

$$F_m = \{C \rightarrow B, E \rightarrow D, D \rightarrow B, B \rightarrow D, C \rightarrow A\}$$

(2) 满足3NF的函数依赖保持的分解为:

$$\{CBA, DE, BD, P\}$$

(3) 候选码 包含CEP

$$(CEP)^+ = (CEDABP)$$

所以唯一候选码是CEP

(4) 无损连接和函数依赖保持的3NF分解为

$$\{CBA, DE, BD, CEP\}$$





# 练习

假定要构造一个数据库，属性集为  $\{A, B, C, D, E, F, G\}$ ，给定的函数依赖集  $F$  如下：

$F = \{BCD \rightarrow A, BC \rightarrow E, A \rightarrow F, F \rightarrow G, C \rightarrow D, A \rightarrow G\}$ 。求：

- (1)  $R$  的一个满足 3NF 的函数依赖保持分解；
- (2)  $R$  的一个满足 3NF 的无损连接和函数依赖保持的分解。

$$F_m = \{BC \rightarrow A, BC \rightarrow E, A \rightarrow F, F \rightarrow G, C \rightarrow D\}$$



# 分解算法3：结果为BCNF的连接不失真分解

1. 设关系模式 $R$ 不属于BCNF， $R$ 可表示为 $R(XYZ)$ ， $XYZ$ 是属性集合， $X \rightarrow Y$ ，且 $X \rightarrow Y$ 是非平凡函数依赖且 $X$ 不是 $R$ 的候选码。用 $R$ 的分解 $R_1(X, Y)$ 和 $R_2(X, Z)$ 代替 $R(XYZ)$ 。

2. 这时 $R_1$ 属于BCNF，如果 $R_2(X, Z)$ 仍不属于BCNF，计算 $F^+$ 在 $R_2(X, Z)$ 上的投影的最小覆盖，继续这个过程，最终可分解成一组BCNF模式集

**注：**BCNF模式集的分解能保证无损分解，但不一定能保证保持函数依赖集



例：设有关系模式R (U, F) , 其中

$U=\{CTHRSG\}$

$F=\{CS\rightarrow G, C\rightarrow T, TH\rightarrow R, HR\rightarrow C, HS\rightarrow R\}$

将其无损连接地分解为BCNF

解：

(1)求所有的候选关键字：本题为HS

(2)分解：首先在F中找出这样一个非平凡的函数依赖 $X\rightarrow A$ ，其中X不包含R的任何候选关键字，把R分解为 $R_1(X,A)$ 和 $R_2(U-A)$

在此选 $CS\rightarrow G$  则 $R=\{CSG, CTHRS\}$ ，显然CSG为BCNF，而CTHRS不属于BCNF



(3) 求 $F^+$ 在CTHRS上的投影的最小覆盖

$$C \rightarrow T \quad TH \rightarrow R \quad HR \rightarrow C \quad HS \rightarrow R$$

求CTHRS的候选关键字为HS

(4) 分解CTHRS，选 $C \rightarrow T$ ，分解为

$$CTHRS = \{CT, CHR\},$$

CT为BCNF，而CHR不是

(5) 再求 $F^+$ 在CHR上的投影的最小覆盖

$$CH \rightarrow R \quad HR \rightarrow C \quad HS \rightarrow R$$

求CHR的候选关键字为HS



(6) 分解CHRS, 选 $CH \rightarrow R$ , 分解为

$$CHRS = \{CHR, CHS\}$$

CHR显然  $\in$  BCNF, CHS是否满足BCNF?

因为对于CHS, HS为关键字所以CHS  $\in$  BCNF

所以本题的解为

$$CTHRSG = \{CSG, CT, CHR, CHS\}$$



**注意：**不是每个BCNF分解都是保持函数依赖的。

银行管理中的一个关系模式：

**R**(分行号，客户编号，帐户负责人编号)，

表示“一个客户在某一支机构有一个银行帐号负责人”

**F**={帐户负责人编号→分行号，

[分行号，客户编号]→帐户负责人编号}



**R的候选码：[分行号， 客户编号]**

由于“帐户负责人编号→分行号” 的决定因素  
不包含码，所以分解为如下两个关系模式：

**R1={帐户负责人编号， 分行号}**

**R2={客户编号， 帐户负责人编号}**

**F1={帐户负责人编号→分行号}**

**F2=∅**

**丢失了函数依赖！**



# 模式分解的几个重要事实

- 若要求连接不失真，分解可达到BCNF。
- 若要求依赖保持，则分解可达到3NF，但不一定能达到BCNF。
- 若同时要求连接不失真和依赖保持，则分解可达到3NF，但不一定能达到BCNF。





## 模式规范化目的

- 1) 消除异常现象。
- 2) 方便用户使用，简化检索操作。
- 3) 加强数据独立性，即当引入一个新数据项时，减少对原有数据结构的修改。
- 4) 使关系模式更灵活，更容易使用非过程化的高级查询语言。
- 5) 更容易进行各种查询统计工作。



# 模式分解

- 连接不失真性;
- 依赖保持性;
- 某一级范式

保持依赖决定分解的好坏  
无损分解决定能否分解



消除  
决定  
因素  
非码  
的非  
平凡  
函数  
依赖

1NF

消除非主属性对码的部分函数依赖

2NF

消除非主属性对码的传递函数依赖

3NF

消除主属性对码的部分和传递函数依赖

BCNF



# 关系规范化问题的处理步骤

1. 确定所有的候选关键字
2. 选定主关键字
3. 确定关系得各个属性中，哪些是主属性，哪些是非主属性
4. 找出属性间的依赖关系（函数依赖）



5.根据应用特点，确定规范化到第几范式

6.分解关系，分解必须是无损的，不得丢失信息

7.分解后的关系，力求相互独立，即对一个关系内容的修改不要影响到分解出来的其他关系

