

# 第六章 关系数据理论

6.1 问题的提出—为什么要学习关系数据理论

**6.2 规范化**

6.3 数据依赖的公理系统（简单介绍）

6.4 模式的分解（简单介绍）

6.5 小结



## 6.2 规范化—关系的规范化理论

6.2.1 函数依赖

6.2.2 码

6.2.3 范式

6.2.4 第二范式 (2NF)

6.2.5 第三范式 (3NF)

6.2.6 BC范式 (BCNF)

\*6.2.7 多值依赖

\*6.2.8 第四范式 (4NF)

6.2.9 规范化小结



## 6.2 规范化—关系的规范化理论

### 6.2.1 函数依赖

### 6.2.2 码

### 6.2.3 范式

### 6.2.4 第二范式 (2NF)

### 6.2.5 第三范式 (3NF)

### 6.2.6 BC范式 (BCNF)

### \*6.2.7 多值依赖

### \*6.2.8 第四范式 (4NF)

### 6.2.9 规范化小结



## 6.2.1 函数依赖

1. 函数依赖
2. 平凡函数依赖与非平凡函数依赖
3. 完全函数依赖与部分函数依赖
4. 传递函数依赖



# 1. 函数依赖

## 定义6.1

设 $R(U)$ 是一个属性集 $U$ 上的关系模式， $X$ 和 $Y$ 是 $U$ 的子集。若对于 $R(U)$ 的任意一个可能的关系 $r$ ， $r$ 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 $X \rightarrow Y$ 。

$X$ 称为这个函数依赖的决定属性组，也称为**决定因素**(Determinant)。

例：  $S(Sno, Sname, Ssex, Sage, Sdept)$

$F = \{Sno \rightarrow Sname, Sno \rightarrow Ssex, Sno \rightarrow Sage, Sno \rightarrow Sdept\}$

$Ssex \not\rightarrow Sage, Ssex \not\rightarrow Sdept$

若 $Y$ 不函数依赖于 $X$ ，则记为 $X \not\rightarrow Y$ 。



# 1. 函数依赖（续）

违背了  $Sno \rightarrow Sname$

Sno	Sname	Ssex	Sage	Sdept
S1	张三	男	20	计算机系
S1	李四	女	21	自动化系
S3	王五	男	20	计算机系
S4	赵六	男	21	计算机系
S5	田七	男	20	计算机系
⋮	⋮	⋮	⋮	⋮



## 如何确定函数依赖（续）

由下面的关系表, 能否得出 **Sname**  $\rightarrow$  **Sno** ?

Sno	Sname	Ssex	Sage	Sdept
S1	张三	男	20	计算机系
S2	李四	女	21	自动化系
S3	王五	男	20	计算机系
S4	赵六	男	21	计算机系
S5	田七	男	20	计算机系
⋮	⋮	⋮	⋮	⋮

函数依赖不是指关系模式R的**某个或某些关系实例r**满足的约束条件, 而是指R的**所有关系实例r**均要满足的约束条件。



# 如何确定函数依赖

- ❖ 函数依赖是语义范畴的概念。只能**根据数据的语义来确定函数依赖**。
  - 如  $Sname \rightarrow Sno$  函数依赖只有在“学生不允许有重名”的条件下成立。
- ❖ 数据库设计者可以对现实世界作强制的规定。
  - 例如设计者**可以强行规定不允许学生有重名**，因而使函数依赖  $Sname \rightarrow Sno$ ,  $Sname \rightarrow Ssex$ ,  $Sname \rightarrow Sage$ ,  $Sname \rightarrow Sdept$  成立。
- ❖ 函数依赖是指**关系模式R在任何时刻的关系实例均要满足的约束条件**。
  - 不是指**某个或某些关系实例 r** 满足的约束条件，而是指R的**所有关系实例 r** 均要满足的约束条件。





## 2. 平凡函数依赖与非平凡函数依赖

- ❖  $X \rightarrow Y$ ,  $Y \not\subseteq X$ , 则称 $X \rightarrow Y$ 是非平凡的函数依赖。
- ❖  $X \rightarrow Y$ , 但 $Y \subseteq X$ , 则称 $X \rightarrow Y$ 是平凡的函数依赖。

例：在关系SC(Sno, Cno, Grade)中，

非平凡函数依赖：  $(\text{Sno}, \text{Cno}) \rightarrow \text{Grade}$

平凡函数依赖：  $(\text{Sno}, \text{Cno}) \rightarrow \text{Sno}$

$(\text{Sno}, \text{Cno}) \rightarrow \text{Cno}$

对于任一关系模式，平凡函数依赖都是必然成立的，它不反映新的语义，因此若不特别声明，**我们总是讨论非平凡函数依赖。**

### 3. 完全函数依赖与部分函数依赖

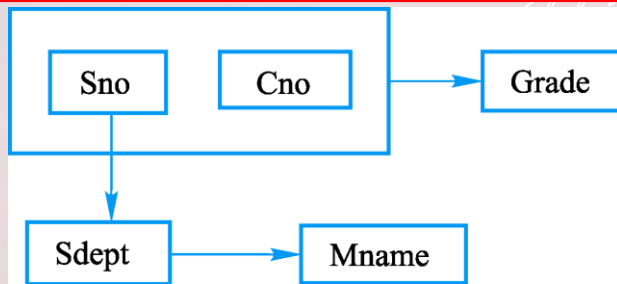
#### 定义6.2

在关系模式 $R(U)$ 中, 如果 $X \rightarrow Y$ , 并且对于 $X$ 的任何一个真子集 $X'$ , 都有 $X' \nrightarrow Y$ , 则称 $Y$ 完全函数依赖于 $X$ , 记作 $X \xrightarrow{F} Y$ 。若 $X \rightarrow Y$ , 但 $Y$ 不完全函数依赖于 $X$ , 则称 $Y$ 部分函数依赖于 $X$ , 记作 $X \xrightarrow{P} Y$ 。

[例] 在关系 $STUDENT(Sno, Sdept, Mname, Cno, Grade)$ 中,

$(Sno, Cno) \xrightarrow{F} Grade$  是完全函数依赖

$(Sno, Cno) \xrightarrow{P} Sdept$  是部分函数依赖, 因为 $Sno \rightarrow Sdept$ ,



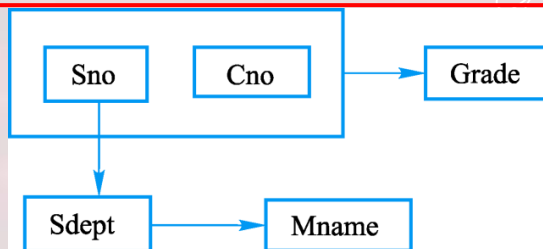
## 4. 传递函数依赖

### 定义6.3

在 $R(U)$ 中, 如果 $X \rightarrow Y$ , ( $Y \not\subseteq X$ ),  $Y \nrightarrow X$ ,  $Y \rightarrow Z$ , 则称 $Z$ 对 $X$ 传递函数依赖 (transitive functional dependency)。记为:  $X \xrightarrow{\text{传递}} Z$ 。

注意: 如果 $Y \rightarrow X$ , 即 $X \longleftrightarrow Y$ , 则 $Z$ 直接依赖于 $X$ 。

[例] 在关系 $STUDENT(Sno, Sdept, Mname, Cno, Grade)$ 中,  
 $Sno \rightarrow Sdept$ ,  $Sdept \rightarrow Mname$ ,  $Sno \xrightarrow{\text{传递}} Mname$



# 思考与讨论

- ❖ 有关系R (C,D,H,S) ,
- ❖ 从 r1、r2 是否可以说  $C \rightarrow DH$  成立？或  $C \nrightarrow DH$  成立。

关系 r 1

C	D	H	S
c1	d1	h1	s1
c1	d1	h1	s1
c1	d1	h1	s2
c2	d2	h2	s3

关系 r 2

C	D	H	S
c1	d1	h1	s1
c1	d1	h2	s1
c1	d1	h1	s2
c2	d2	h2	s3

一般地，从 关系实例，我们能确定某个函数依赖成立吗？  
能确定某个函数依赖不成立吗？



## 6.2.1 函数依赖回顾

**WHAT**

什么是函数依赖

完全函数依赖

部分函数依赖

传递函数依赖

**HOW**

如何确定函数依赖

**WHY**

为什么要学习函数依赖



## 6.2 规范化—关系的规范化理论

6.2.1 函数依赖

6.2.2 码

6.2.3 范式

6.2.4 第二范式 (2NF)

6.2.5 第三范式 (3NF)

6.2.6 BC范式 (BCNF)

\*6.2.7 多值依赖

\*6.2.8 第四范式 (4NF)

6.2.9 规范化小结





## 6.2.2 码

❖ 定义6.4 设K为关系模式R<U,F>中的属性或属性组合。若 $K \xrightarrow{F} U$ ，则K称为R的一个**候选码(Candidate Key)**。

■ 如果U部分函数依赖于K，即 $K \xrightarrow{F} U$ ，则K称为**超码 (Surpkey)**

■ 候选码是最小的超码，即K的任意一个真子集都不是候选码

[例] S(Sno, Sdept, Sage)

Sno  $\rightarrow$  (Sno, Sdept, Sage), Sno是码

(Sno, Sdept)、(Sno, Sage)、(Sno, Sdept, Sage) 是超码

SC(Sno, Cno, Grade)中, (Sno, Cno)是码

❖ 若关系模式R有多个候选码，则选定其中的一个做为**主码(Primary key)**。

[例] S(Sno, Sname, Sdept, Sage), 假设学生无重名

Sno、Sname是候选码，选择Sno为主码。





## 6.2.2 码

### ❖ 主属性与非主属性

- 包含在任何一个候选码中的属性，称为**主属性**（Prime attribute）
- 不包含在任何码中的属性称为**非主属性**（Nonprime attribute）或非码属性（Non-key attribute）

[例] S(Sno, Sdept, Sage), Sno是码, Sno是主属性, Sdept, Sage是非主属性。  
SC(Sno, Cno, Grade)中, (Sno, Cno)是码,  
Sno, Cno是主属性, Grade是非主属性

### ❖ 全码：整个属性组是码，称为**全码**（All-key）

[例] 关系模式 R (P, W, A) P: 演奏者 W: 作品 A: 听众

语义：一个演奏者可以演奏多个作品，某一作品可被多个演奏者演奏，听众可以欣赏不同演奏者的不同作品

R (P, W, A) 码为(P, W, A)，即全码，All-Key。

## 6.2.2 码：外部码——外码

### 定义6.5

关系模式  $R\langle U, F \rangle$ ， $U$ 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 (Foreign key)**也称**外码**。

SC (Sno, Cno, Grade) 中，Sno不是码，但Sno是关系模式S (Sno, Sdept, Sage)的码，则Sno是关系模式SC的外部码。

❖ 主码与外部码一起提供了表示关系间联系的手段



# 思考与讨论

## ❖ 思考题

已知 关系模式  $R\langle U, F \rangle$ ,

$U = \{A, B, C, D, E, G\}$

$F = \{AC \rightarrow B, CB \rightarrow D, A \rightarrow BE, E \rightarrow GC\}$

求关系R的候选码？

## ❖ 如何根据已知的F，求一个关系模式R的候选码？

- 简单情况下，可以用观察、验证的方法
- 一般情况下，使用算法



## 6.2 规范化

6.2.1 函数依赖

6.2.2 码

6.2.3 范式

6.2.4 第二范式 (2NF)

6.2.5 第三范式 (3NF)

6.2.6 BC范式 (BCNF)

\*6.2.7 多值依赖

\*6.2.8 第四范式 (4NF)

6.2.9 规范化小结





# 第六章 关系数据理论

