

Chapter5 规范化设计

规范化设计理论 = 模式设计理论

- 数据依赖（核心）：数据直接的联系。函数依赖
- 范式：关系模式的标准。
- 模式设计方法

5.1 关系模式的设计问题

5.1.1 关系模型的外延和内涵

外延：关系 / 表 / 当前值。

内涵：数据的定义，数据完整性约束的定义。

- 数据的定义：关系，属性，域的定义和说明
- 完整性约束：静态约束（数据依赖，主键，值域），动态约束（定义操作对关系值的影响）

关系模式 = 内涵

5.1.2 泛关系模式与数据库模式

泛关系模式：所有属性组成的关系模式 $R(U)$

泛关系：当前值 r

数据库模式： $\rho = R_1, R_2, \dots, R_n$ ，其中 $R_1 \cup R_2 \cup \dots \cup R_n = R(U)$

数据库： R_i 的实例

5.1.3 冗余和异常

数据冗余：同一个数据重复存储多次。

操作异常：修改异常，插入异常，删除异常

5.2 函数依赖 FD

5.2.1 函数依赖

1. 函数依赖： $X \rightarrow Y$

- X 相等，则 Y 一定相等； Y 相等，则 X 不一定相等
- 例：1对多 $A:B = 1:N$ $B \rightarrow A$ ；1对1 $A:B = 1:1$ $A \rightarrow B$ $B \rightarrow A$

2. 部分依赖： $X \twoheadrightarrow Y$ ，且存在 X 的真子集 x' ，满足 $x' \twoheadrightarrow Y$

3. 完全依赖/左部不可约依赖： $X \rightarrow Y$ ，且不存在 X 的真子集 x' ，满足 $x' \rightarrow Y$

4. 传递依赖

5. 候选键，超键

6. 主属性：候选键中的属性。

a. 如何选择候选键？

- 不在函数依赖右部出现的属性，一定是
- 只在函数依赖右部出现的属性，一定不是

5.2.2 蕴含逻辑

1. 逻辑蕴含 $F \models X \rightarrow Y$

2. 函数依赖集 F 的闭包 $F^+ : \{ \text{被} F \text{蕴含的函数依赖全体} \}$

5.2.3 FD推理规则

1. "Armstrong" 公理

- 自反性：若 $Y \subseteq X \subseteq U$ ，则 $X \rightarrow Y$ 在 R 上成立。
- 增广性：若 $X \rightarrow Y$ 在 R 上成立，且 $Z \subseteq U$ ，则 $XZ \rightarrow YZ$ 在 R 上成立。
- 传递性：若 $X \rightarrow Y$ 和 $Y \rightarrow Z$ 在 R 上成立，则 $X \rightarrow Z$ 在 R 上成立。

2. 常用的推理规则

- 合并性 若 $X \rightarrow Y$ ， $X \rightarrow Z$ ，则 $X \rightarrow YZ$ 。
- 分解性 若 $X \rightarrow Y$ ， $Z \subseteq Y$ ，则 $X \rightarrow Z$ 。
- 伪传递性 若 $X \rightarrow Y$ ， $YW \rightarrow Z$ ，则 $XW \rightarrow Z$ 。

d. 复合性 若 $X \rightarrow Y, W \rightarrow Z$, 则 $XW \rightarrow YZ$ 。

e. 通用一致性 若 $X \rightarrow Y, W \rightarrow Z$, 则 $X \cup (W - Y) \rightarrow YZ$ 。

如果 $A_1 \dots A_n$ 是关系模式 R 的属性集, 那么 $X \rightarrow A_1 \dots A_n$ 成立

3. 充要条件:

的充分必要条件是 $X \rightarrow A_i (i=1, \dots, n)$ 成立。

4. 平凡的FD, 非平凡的FD

定义: 设 F 是属性集 U 上的FD集, X 是 U 的子集, 那么 (相

5. 属性集的闭包 X^+

对于 F) 属性集 X 的闭包用 X_F^+ 表示, 它是一个从 F 集使用FD推理规则推出的所有满足 $X \rightarrow A$ 的属性 A 的集合:

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

6. F^+ 有 $X \rightarrow Y \iff Y \subseteq X^+$

7. 等价的依赖集: 在 U 上依赖集闭包相等, 即 $F = G$, 则 $F^+ = G^+$ 。

8. 最小依赖集 F_{min}^+

a. 消除右端冗余: 右端全分解为单一属性。

b. 消除左端冗余: $AD \rightarrow B$, 用 $D \rightarrow B$ 代替, 求 A^+ , 若能求出 B , 则 A 冗余。

c. 消除FD冗余: 取出一个依赖, 求 F^+ , 判断是否等价。

5.3 关系模式的分解特性

1. 无损分解, 损失分解的判定

a. 分解 = 2 模式, $R_1 \cap R_2 \rightarrow R_1 - R_2 / R_2 - R_1$

b. Chase规则 (表格)

2. 保持函数依赖的判定

a. 求分解模式在 F 上的投影

b. 投影的并 = F , 则保持

5.4 范式

5.4.1 范式的概念

1. 局部依赖, 完全依赖, 传递依赖; 主属性, 非主属性

2. 1NF: 属性不可分。

3. 2NF: 1NF, 且非主属性完全依赖于候选键。

4. 3NF: 1NF, 且非主属性都不传递依赖于候选键。

- a. 对于F中每个非平凡FD $X \rightarrow Y$, 都有X是R的超键, 或者Y的每个属性都是主属性。(左部右部, 至少一边包含码)

5. BCNF: 1NF, 每个属性都不传递依赖于候选键。

- a. 对于F中每个非平凡FD $X \rightarrow Y$, 都有X是R的超键。所有的决定因素都包含码(用于判断)

5.4.2 分解成3NF模式集

算法1: 保持 ρ 的3NF模式集: $U + F_{min}$

1. 所有F中没有出现的属性, 放在一个单独的模式。
2. F中有 $X \rightarrow A$, 且 $\{X, A\} = U$, 则停止
3. 不为a,b, 所有左边相同的, 放在同一分组 $X-A \Rightarrow XA$
4. 分解结束, 输出 ρ

算法2: 保持 ρ , 保持函数依赖的3NF模式集: $U + F_{min}$

1. 算法1得到一个模式集
2. 候选键加入模式集。

5.4.3 分解成BCNF范式

保持无损分解, 不一定保持函数依赖。

1. 置初值: $\rho = \{R\}$ 。
2. ρ 已经满足BC范式, 则结束。
3. 找到左边没有码的依赖 $X \rightarrow A$, 把A从U里面删除, XA单独放在一个模式。继续, 直到都满足BC范式。
4. 结束, 输出 ρ 。

5.4.4 模式设计方法的原则

三个特性

1. ρ 是BCNF模式集, 或3NF模式集
2. 无损分解
3. 保持函数依赖

三个原则

1. 表达性: 无损联接, 保持函数依赖

2. 分离性：范式

3. 最小冗余性：模式个数和模式中属性总数应最少