

## Chapter3 关系数据库系统

### 3.1 关系数据库系统概述

优点：

数据结构简单（二维表）；使用方便；功能强；数据独立性高；理论基础深；

可移植性好；标准化程度高；提供分布式功能；具有开放性；其它方面的功能扩展。

### 3.2 关系数据库系统衡量准则（仅列出来作为参考，应该不会考）

#### 六条准则（1974年IFIP会议）

- 1) 提供高度的数据独立性
- 2) 提供严格的数据视图
- 3) 减轻DBA的工作
- 4) 建立理论基础
- 5) 事务管理与文件管理相结合
  - 为商业及其它行业的服务作准备
- 6) 操作对象是记录集合，而不是单个记录

#### 完全关系型的12条严格标准（1985年）

- 1) 信息准则

关系数据库的所有信息都应能在逻辑一级唯一地用‘表’中的值显式表示

-用表中的‘元组’或元组中的‘属性值’来表示

数据库的结构描述信息也应在逻辑上组织成关系形式

- 2) 确保访问准则

应能保证用逻辑方式依靠表名、关键字值与列名的组合来访问数据库中每一个原子数据

- 3) 空值的关系处理准则

- 4) 基于资源管理的动态联机目录

- 5) 统一易用的数据子语言

- 6) 视图更新准则

- 7) 高级的插入、删除及修改操作

- 8) 物理数据独立性

- 9) 逻辑数据独立性

- 10) 数据完整性准则：提供三类数据完整性约束的定义功能

- 11) 分布独立性：数据分布的改变不影响原有的应用程序

- 12) 无损害原则：对提供低级数据子语言的要求

### 3.3 关系模型数学理论 — 关系代数

#### 3.3.0 关系模型

##### 3.3.0.1 关系数据结构

名词术语之间的对应关系

关系模型	关系数据库管理系统	文件系统
Relation	Table	File of Records
Attribute	Column	Field
Tuple	Row	Record
Schema	Table Heading	Type of Record

Table Heading: a set of named columns about a table

#### 二维表的结构

- 二维表（简称表）的组成
  - 表框架（Frame）
    - 表框架由 $n$ 个命名的‘属性’（Attribute）组成
      - $n$  被称为表的元数（ $n$ 元表）
      - 每个属性有一个取值范围，即值域（Domain）
  - 元组（Tuple）
    - 在表框架中可按行存放数据，其中的每行数据称为‘元组’
      - 一个元组由 $n$ 个元组分量组成，每个元组分量都对应着表框架中的一个属性
    - 一个表框架可存放 $m$ 个元组， $m$ 被称为表的基数（Cardinality）

一个 $n$ 元表框架及框架内的 $m$ 个元组构成一个完整的二维表

#### 二维表的性质

- 1) 元组个数有限性
- 2) 元组的唯一性
- 3) 元组的次序无关性
- 4) 元组分量的原子性
- 5) 属性名唯一性
- 6) 属性的次序无关性
- 7) 分量值域同一性

满足上述7个性质的二维表被称为‘关系’（Relation）

以符合上述条件的二维表为基本数据结构所建立的模型称为‘关系模型’

#### 键（Key）

在二维表中凡能唯一标识元组的属性集称为该表的‘键’，或称‘关键字’

候选键 (Candidate Key)

主键 (Primary key) 在一张二维表的所有候选键中, 被选中的一个候选键被称为该表的'主键' (或称'主关键字')

每一张二维表都至少存在一个'键'

键要求具有最小性, 即它的任意子集都不能做到"唯一标识元组", 但是 superkey 没有该要求; 换言之, 任何可以唯一标识元组的属性集合都可以叫做 superkey。

外键 (Foreign Key)

如果表A中的属性集F是表B的键, 则称该属性集F为表A的'外键' (或称'外关键字')

其中: -表A被称为'引用表', 表B被称为'被引用表' -表A和表B可以是同一张二维表

### 3.3.0.2 关系操纵

查询、增加、删除、修改

关系模型上的五种基本操纵功能: 元组选择、属性指定、两个关系的合并、元组插入、元组删除

#### 空值处理

与空值有关的数据完整性约束

-关系的主键中不允许出现空值

需要定义有关空值的运算

①在算术表达式中如出现空值, 则其运算结果也为空值;

②在逻辑运算表达式中如出现空值, 则其运算结果为逻辑假;

③在统计计算中, **对于'空集'和集合中的'空值'的处理方法**

在作SUM, AVG, MAX, MIN或COUNT统计操作时, 集合中的'空值'元素是不统计在内的

在统计计算中, 对'空集'的处理方法如下:

①对空集作SUM、AVG、MAX或MIN统计操作时, 其统计结果均为空值

②对空集作COUNT统计操作时, 其统计结果为0

-注意 count(<表达式>) 和 count(\*) 的区别

#### 三类数据完整性约束

实体完整性约束 -主键中的属性不能有空值

参照完整性约束 -外键要么取空值, 要么是被引用表中当前存在的某元组上的主键值

用户定义的完整性 -用户自己定义的属性取值约束

### 3.3.1 关系的表示

关系是元组的集合, 元组是元组分量的集合

一个n元关系中的元组是一个由n个元组分量按照属性的排列次序组织起来的n元有序组

因此, 一个n元关系就是一个n元有序组的集合

### 3.3.2 关系操纵的表示

关系上的五种基本操作与关系代数中的五种基本运算之间的对应关系：

关系上的五种基本操作	关系代数中的五种基本运算
元组选择	选择运算
属性指定	投影运算
关系的合并	笛卡儿乘积
元组的插入	并运算
元组的删除	差运算

**相容表：**两个表的表头相同，并且每个相同属性的定义域相同

U、 $\cap$ 、 $-$  要求两个表相容；

	交换律	结合律
并运算	√	√
交运算	√	√
差运算	×	×

#### 投影运算 $\Pi$

➤ 设关系R有n个属性 $A_1, A_2, \dots, A_n$ ，在其中m个属性 $B_1, B_2, \dots, B_m$ 上的投影运算可以表示为：

$$\Pi_{B_1, B_2, \dots, B_m}(R)$$

– 其中： $B_i \in \{A_1, A_2, \dots, A_n\} (i=1, 2, \dots, m)$

➤ 运算结果

– 是一个由 $B_1, B_2, \dots, B_m$ 所组成的m元关系

– 关系R中的每个元组t在 $B_1, B_2, \dots, B_m$ 这m个属性上的取值 $t_1, t_2, \dots, t_m$ 构成结果关系中的一个元组

#### 选择运算 $\sigma$

➤ 根据给定的条件F从关系R中选出符合条件的元组

➤ 结果

– 结果关系的关系模式不变，由属于关系R且满足条件F的元组所组成

可以简写为： $\Pi_A \sigma_F(R)$

– 在没有括号的情况下，其运算顺序为：从右向左

一般先选择再投影，且一般情况下投影运算和选择运算不能相互交换

#### 笛卡儿乘积 $\times$

是两个关系的合并运算。

关系数据库中的四种类型的操作

#### 1) 元组插入

- 设新增加的元组构成关系 $R'$ ，则插入操作可表示为： $R \cup R'$

#### 2) 元组删除

- 设欲删除的元组构成关系 $R'$ （可以经查询操作获得），则删除操作可表示为： $R - R'$

#### 3) 修改操作

- 设欲修改的元组构成关系 $R'$ ，经修改后所生成的新元组构成关系 $R''$ ，则数据库的修改操作可表示为： $(R - R') \cup R''$

#### 4) 查询操作

- 单个关系：选择 + 投影
- 多个关系
  - 先用笛卡儿乘积将多个关系合并为一个关系，然后再执行单个关系上的数据查询操作

### 3.3.3 关系模型与关系代数

主要需要掌握一些典型题目的求解方式

求最大（先求非最大，再做减法）

- 1) 查询所有顾客的编号，其结果构成关系 $R_1$

$$R_1 := \Pi_{cid} (C)$$

- 2) 查询折扣并非最大的顾客（其折扣低于其他某个顾客的折扣，或至少存在一个顾客 $X$ ，而 $X$ 的折扣高于当前顾客的折扣）的编号，查询结果构成关系 $R_2$

令  $S := C$ ，则：

$$R_2 := \Pi_{C.cid} (\sigma_{C.discnt < S.discnt} (C \times S))$$

- 3) 利用减法（**difference**）运算获得享受最大折扣顾客的编号

$$T := R_1 - R_2$$

求所有 (先用投影筛选, 再做除法)

学生 S ( sno, sn, sd, sa )

课程 C ( cno, cn, pno )

选课 SC ( sno, cno, g )

【例3.18】检索修读S4所修读的所有课程的学生姓名

【分析】

1) 结果元组需要满足的条件?

修读了S4所修读过的所有课程

2) S4修读了哪些课程?

3) 如何根据一组课程 (S4所修读过的课程) 查出修读了其中的所有课程的学生元组?

$\Pi_{sn} ( S \bowtie ( \Pi_{sno,cno}(SC) \div \Pi_{cno}(\sigma_{sno='S4'}(SC)) ) )$

### 3.3.4 关系代数中的扩充运算

交 (intersection) 运算

‘交’运算并不是一个基本运算, 其功能可以由‘差’运算来实现

$$R \cap S = R - (R - S) = S - (S - R)$$

除 (division) 运算

➤ 设关系R和S的关系模式(即它们的属性集)分别是 Head(R) 和 Head(S)

➤ 运算条件: Head(S)  $\subset$  Head(R)

– 可以假设为:

▪ Head(R) = { A<sub>1</sub>, A<sub>2</sub>, ..., A<sub>n</sub>, B<sub>1</sub>, B<sub>2</sub>, ..., B<sub>m</sub> }

▪ Head(S) = { B<sub>1</sub>, B<sub>2</sub>, ..., B<sub>m</sub> }

– 其中:

▪ S 被称为 ‘除数关系’

▪ R 被称为 ‘被除数关系’

▪ 结果关系被称为 ‘商’

➤ 结果关系 (令 T = R  $\div$  S)

– 关系模式

▪ Head(T) = Head(R) - Head(S) = { A<sub>1</sub>, A<sub>2</sub>, ..., A<sub>n</sub> }

– 结果元组

▪ 关系S中的所有元组在关系R中所对应的同一个值

▪ 设 x 是结果关系 T 中的一个元组 ( x  $\in$  T ), 则对于关系 S 中的每一个元组 y 必有:

( x, y )  $\in$  R

▪ 由所有符合上述条件的元组 x 构成结果关系的元组集合



## ‘除’运算与基本关系运算的关系

IF

$$\text{Head}(R) = \{ A_1 \dots A_n B_1 \dots B_m \}$$

$$\text{Head}(S) = \{ B_1 \dots B_m \}$$

THEN

$$R \div S = \Pi_{A_1 \dots A_n}(R) - \Pi_{A_1 \dots A_n}((\Pi_{A_1 \dots A_n}(R) \times S) - R)$$

## 联接 (join) 运算

联接 (join) 运算:  $R \bowtie_F S$

➤ 又称θ-联接运算，可以将关系 R 和关系 S 根据联接条件 F 合并为一个关系

$$R \bowtie_F S = \sigma_F(R \times S)$$

## 自然联接 (natural join) 运算

运算条件: R和S有公共属性; 公共属性相同时, 合并R和S中的两个元组。

其他的一些联接运算:

➤ 外联接 (outer join)

$$R \Join S$$

➤ 左外联接 (left outer join)

$$R \Joinleft S$$

➤ 右外联接 (right outer join)

$$R \Joinright S$$

可以通过结果来重构原来的关系R和/或S

例子:

R	<table><tr><th>A</th><th>B</th></tr><tr><td>a1</td><td>b1</td></tr><tr><td>a2</td><td>b2</td></tr><tr><td>a3</td><td>b5</td></tr></table>	A	B	a1	b1	a2	b2	a3	b5	S	<table><tr><th>B</th><th>C</th></tr><tr><td>b1</td><td>c1</td></tr><tr><td>b2</td><td>c2</td></tr><tr><td>b3</td><td>c3</td></tr><tr><td>b4</td><td>c4</td></tr></table>	B	C	b1	c1	b2	c2	b3	c3	b4	c4	$R \bowtie S$	<table><tr><th>A</th><th>B</th><th>C</th></tr><tr><td>a1</td><td>b1</td><td>c1</td></tr><tr><td>a2</td><td>b2</td><td>c2</td></tr></table>	A	B	C	a1	b1	c1	a2	b2	c2
A	B																															
a1	b1																															
a2	b2																															
a3	b5																															
B	C																															
b1	c1																															
b2	c2																															
b3	c3																															
b4	c4																															
A	B	C																														
a1	b1	c1																														
a2	b2	c2																														

$R \Joinleft S$	$R \Joinright S$	$R \Joinfull S$																																													
<table><tr><th>A</th><th>B</th><th>C</th></tr><tr><td>a1</td><td>b1</td><td>c1</td></tr><tr><td>a2</td><td>b2</td><td>c2</td></tr><tr><td>a3</td><td>b5</td><td>null</td></tr><tr><td>null</td><td>b3</td><td>c3</td></tr><tr><td>null</td><td>b4</td><td>c4</td></tr></table>	A	B	C	a1	b1	c1	a2	b2	c2	a3	b5	null	null	b3	c3	null	b4	c4	<table><tr><th>A</th><th>B</th><th>C</th></tr><tr><td>a1</td><td>b1</td><td>c1</td></tr><tr><td>a2</td><td>b2</td><td>c2</td></tr><tr><td>a3</td><td>b5</td><td>null</td></tr></table>	A	B	C	a1	b1	c1	a2	b2	c2	a3	b5	null	<table><tr><th>A</th><th>B</th><th>C</th></tr><tr><td>a1</td><td>b1</td><td>c1</td></tr><tr><td>a2</td><td>b2</td><td>c2</td></tr><tr><td>null</td><td>b3</td><td>c3</td></tr><tr><td>null</td><td>b4</td><td>c4</td></tr></table>	A	B	C	a1	b1	c1	a2	b2	c2	null	b3	c3	null	b4	c4
A	B	C																																													
a1	b1	c1																																													
a2	b2	c2																																													
a3	b5	null																																													
null	b3	c3																																													
null	b4	c4																																													
A	B	C																																													
a1	b1	c1																																													
a2	b2	c2																																													
a3	b5	null																																													
A	B	C																																													
a1	b1	c1																																													
a2	b2	c2																																													
null	b3	c3																																													
null	b4	c4																																													

### 3.3.5 关系代数小结

#### □ 关系的表示

- 元组的集合

#### □ 关系操纵的表示

- 集合上的运算
- 五种基本运算
  - 并  $\cup$ , 差  $-$
  - 投影  $\Pi$ , 选择  $\sigma$ , 笛卡儿乘积  $\times$
- 三种扩充运算
  - 交  $\cap$
  - 除  $\div$
  - $\theta$ -联接  $\bowtie_{\theta}$ , 自然联接  $\bowtie$ , 外联接  $\ltimes$