上海交通大学
SHANGHAI JIAO TONG UNIVERSITY

SJTU

数 据 库 技 术

Database System Technology

郭 捷

(guojie@sjtu.edu.cn)

第二章 关系模型和关系运算理论

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

目 录

关系数据库简介

- 系统而严格地提出关系模型的是美国IBM公司的 E. F. Codd
  - 1970年提出关系数据模型：
    - E.F.Codd, “A Relational Model of Data for Large Shared Data Banks”, 《Communication of the ACM》, 1970
  - 之后，提出了关系代数和关系演算的概念；
  - 1972年提出了关系的第一、第二、第三范式；
  - 1974年提出了关系的BC范式；

关系数据库简介

- 关系数据库应用数学方法来处理数据库中的数据；
- 80年代后，关系数据库系统成为最重要、最流行的数据库系统；

- 典型实验系统
  - System R
  - University INGRES
- 典型商用系统
  - ORACLE
  - SYBASE
  - INFORMIX
  - DB2
  - INGRES

关系模型概述

- 关系数据库系统：
  - 是支持关系模型的数据库系统

- 关系模型的组成：
  - 关系数据结构
  - 关系操作集合
  - 关系完整性约束

关系数据结构

- 单一的数据结构----关系：
  - 现实世界的实体以及实体间的各种联系均用关系来表示；
- 数据的逻辑结构----二维表：
  - 从用户角度，关系模型中数据的逻辑结构是一张二维表；

学号	姓名	性别	专业号	年龄
801	张三	女	01	19
802	李四	男	01	20
803	王五	男	01	20
804	赵六	女	02	20
805	钱七	男	02	19

关系操作集合

1）常用的关系操作：

- 查询
  - 选择、投影、连接、除、并、交、差
- 数据更新
  - 插入、删除、修改
- 查询的表达能力是其中最主要的部分；

关系的三类完整性约束

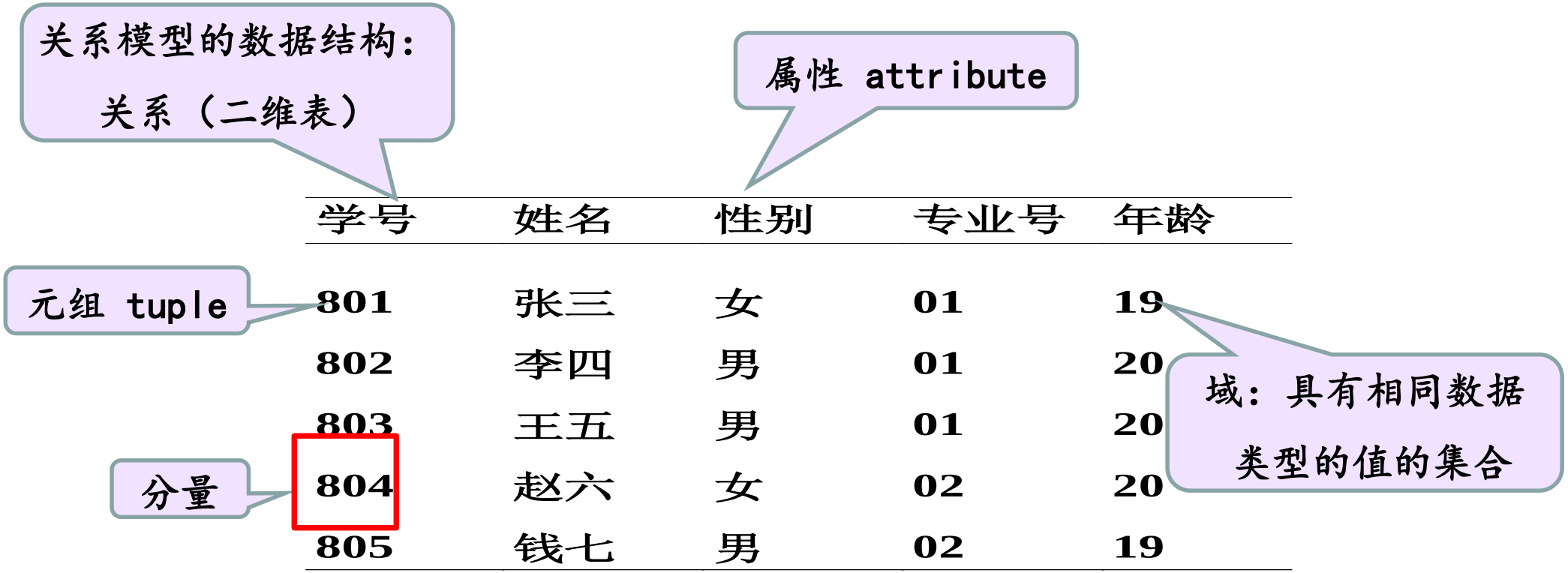
- 实体完整性
  - 通常由关系系统自动支持；
- 参照完整性
  - 早期系统不支持，目前大型系统能自动支持；
- 用户定义的完整性
  - 反映应用领域需要遵循的约束条件，体现了具体领域中的语义约束；
  - 用户定义后由系统支持；

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

目 录

关系数据结构

- 关系模型建立在集合代数的基础上；



学号	姓名	性别	专业号	年龄
801	张三	女	01	19
802	李四	男	01	20
803	王五	男	01	20
804	赵六	女	02	20
805	钱七	男	02	19

学生（学号，姓名，性别，专业号，年龄）

01

关系

关系

1. 域（Domain）

2. 笛卡儿积（Cartesian Product）

3. 关系（Relation）

1.域（Domain）

- 域是一组具有相同数据类型的值的集合。
  - 整数
  - 实数
  - 介于某个取值范围的整数
  - 长度指定长度的字符串集合
  - {‘男’，‘女’}
  - 介于某个取值范围的日期

2.笛卡儿积（Cartesian Product）

- 笛卡儿积：域上的一种集合运算

给定一组域D_1，D_2，…，D_n，这些域中可以有相同的。

例　给出三个域：

D_1=导师集合SUPERVISOR = { 张清玫，刘逸 }

D_2=专业集合SPECIALITY = {计算机专业，信息专业}

D_3=研究生集合POSTGRADUATE = {李勇，刘晨，王敏}

2.笛卡儿积（Cartesian Product）

D_1，D_2，…，D_n的笛卡儿积为：

$$D_1 \times D_2 \times \ldots \times D_n = \{ (d_1, d_2, \ldots, d_n) \mid d_i \in D_i, i = 1, 2, \ldots, n \}$$

- 所有域的所有取值的一个组合；
- 不能重复；

2.笛卡儿积（Cartesian Product）

D_1=SUPERVISOR = { 张清玫，刘逸 }

D_2=SPECIALITY = {计算机专业，信息专业}

D_3=POSTGRADUATE = {李勇，刘晨，王敏}

则D_1，D_2，D_3的笛卡儿积为：

$D_1 \times D_2 \times D_3$ =

{(张清玫，计算机专业，李勇)，(张清玫，计算机专业，刘晨)，
(张清玫，计算机专业，王敏)，(张清玫，信息专业，李勇)，
(张清玫，信息专业，刘晨)，(张清玫，信息专业，王敏)，
(刘逸，计算机专业，李勇)，(刘逸，计算机专业，刘晨)，
(刘逸，计算机专业，王敏)，(刘逸，信息专业，李勇)，
(刘逸，信息专业，刘晨)，(刘逸，信息专业，王敏) }

2.笛卡儿积（Cartesian Product）

- 基数（Cardinal number）
  - 若D_i（i=1，2，…，n）为有限集，其基数为m_i（i=1，2，…，n），则$D_1 \times D_2 \times \cdots \times D_n$的基数$M$为：

$$M = \prod_{i=1}^{n} m_i$$

例如：D_1=SUPERVISOR = { 张清玫，刘逸 }

D_2=SPECIALITY = {计算机专业，信息专业}

D_3=POSTGRADUATE = {李勇，刘晨，王敏}

基数：2×2×3=12，即$D_1 \times D_2 \times D_3$共有2×2×3=12个元组

2.笛卡儿积（Cartesian Product）

- 笛卡儿积的表示方法：
  - 笛卡儿积可表示为一个二维表。

    表中的每行对应一个元组，表中的每列对应一个域。

表　D_1，D_2，D_3 的笛卡尔积

SUPERVISOR	SPECIALITY	POSTGRADUATE
张清玫	计算机专业	李勇
张清玫	计算机专业	刘晨
张清玫	计算机专业	王敏
张清玫	信息专业	李勇
张清玫	信息专业	刘晨
张清玫	信息专业	王敏
刘逸	计算机专业	李勇
刘逸	计算机专业	刘晨
刘逸	计算机专业	王敏
刘逸	信息专业	李勇
刘逸	信息专业	刘晨
刘逸	信息专业	王敏

3.关系（Relation）

1）关系

$D_1 \times D_2 \times \cdots \times D_n$的子集叫作在域$D_1$，$D_2$，…，$D_n$上的关系，表示为

$R(D_1, D_2, \cdots, D_n)$

R：关系名；

n：关系的目或度（Degree）；

3.关系（Relation）

表　D_1，D_2，D_3 的笛卡尔积

SUPERVISOR	SPECIALITY	POSTGRADUATE
张清玫	计算机专业	李勇
张清玫	计算机专业	刘晨
张清玫	计算机专业	王敏
张清玫	信息专业	李勇
张清玫	信息专业	刘晨
张清玫	信息专业	王敏
刘逸	计算机专业	李勇
刘逸	计算机专业	刘晨
刘逸	计算机专业	王敏
刘逸	信息专业	李勇
刘逸	信息专业	刘晨
刘逸	信息专业	王敏

3.关系（Relation）

- D1，D2，…，Dn的笛卡儿积的某个子集才有实际含义！

例： 在前面示例的笛卡儿积中取出有实际意义的元组来构造关系

关系：SAP(SUPERVISOR, SPECIALITY, POSTGRADUATE)

- 关系名，属性名

假设：导师与专业：1:1，导师与研究生：1:n

于是：SAP关系可以包含三个元组：

{ (张清玫，信息专业，李勇)，
(张清玫，信息专业，刘晨)，
(刘逸，信息专业，王敏) }

3.关系（Relation）

2）元组：

关系中的每个元素是关系中的元组，通常用t表示。

3）单元关系与二元关系：

当n = 1时，称该关系为单元关系（Unary relation）。

当n = 2时，称该关系为二元关系（Binary relation）。

3.关系（Relation）

4）关系的表示

关系也是一个二维表，表的每行对应一个元组，
表的每列对应一个域。

SUPERVISOR	SPECIALITY	POSTGRADUATE
张清玫	信息专业	李勇
张清玫	信息专业	刘晨
刘逸	信息专业	王敏

3.关系（Relation）

5）属性：

关系中不同列可以对应相同的域，为了加以区分，必须对每列起一个名字，称为属性（Attribute）。

注意：*n*目关系必有*n*个属性。

3.关系（Relation）

6）码

- 候选码（Candidate key）
  - ✓ 若关系中的某一属性组的值能唯一地标识一个元组，而其子集不能，则称该属性组为候选码。在最简单的情况下，候选码只包含一个属性。
- 全码（All-key）
  - ✓ 在最极端的情况下，关系模式的所有属性组是这个关系模式的候选码，称为全码（All-key）。

3.关系（Relation）

6）码（续）

- 主码（Primary key）
  - 若一个关系有多个候选码，则选定其中一个为主码；
  - 候选码的诸属性称为主属性（Prime attribute）。
  - 不包含在任何候选码中的属性称为非主属性（Non-key attribute）。

3.关系（Relation）

7）三类关系：

- 基本关系（基本表或基表）

  实际存在的表，是实际存储数据的逻辑表示

- 查询表

  查询结果对应的表

- 视图表

  由基本表或其他视图表导出的表，是虚表，不对应实际存储的数据

3.关系（Relation）

注意：1）关系是笛卡儿积的有限子集！无限关系在数据库系统中是无意义的。

2）数学定义：笛卡儿积不满足交换律，即

$$(d1, d2, \cdots, dn) \neq (d2, d1, \cdots, dn)$$

但关系作为关系数据模型的数据结构，满足交换律，即

$$(d_1, d_2, \cdots, d_i, d_j, \cdots, d_n) = (d_1, d_2, \cdots, dj, di, \cdots, d_n) \quad (i, j = 1, 2, \cdots, n)$$

解决方法：为关系的每个列附加一个属性名以取消关系元组的有序性；

基本关系的性质

① 列是同质的（Homogeneous）

每一列中的分量是同一类型的数据，来自同一个域

② 不同的列可出自同一个域

其中的每一列称为一个属性不同的属性要给予不同的属性名

基本关系的性质

③ 列的顺序无所谓

即：列的次序可以任意交换

- 遵循这一性质的数据库产品(如ORACLE)，增加新属性时，永远是插至最后一列；
- 也有许多关系数据库产品没有遵循这一性质，例如FoxPro仍然区分了属性顺序；

基本关系的性质

④ 任意两个元组的候选码不能完全相同

注：由笛卡儿积的性质决定

- 但许多关系数据库产品没有遵循这一性质。

例如：

- Oracle，FoxPro等都允许关系表中存在两个完全相同的元组，除非用户特别定义了相应的约束条件。

基本关系的性质

⑤ 行的顺序无所谓

即：行的次序可以任意交换

- 遵循这一性质的数据库产品(如ORACLE)，插入一个元组时永远插至最后一行；
- 但也有许多关系数据库产品没有遵循这一性质，例如FoxPro仍然区分了元组的顺序；

基本关系的性质

⑥ 分量必须取原子值

即：每一个分量都必须是不可分的数据项。

这是规范条件中最基本的一条！

表　非规范化关系

SUPERVISOR	SPECIALITY	POSTGRADUATE	
		PG1	PG2
张清玫	信息专业	李勇	刘晨
刘逸	信息专业	王敏	

02

关系模式

什么是关系模式

- 关系模式（Relation Schema）是型；
- 关系是值；
- 关系模式是对关系的描述；
  - 元组集合的结构；
    - ✓ 属性构成
    - ✓ 属性来自的域
    - ✓ 属性与域之间的映象关系
  - 元组语义以及完整性约束条件；
  - 属性间的数据依赖关系集合；

定义关系模式

- 关系模式可以形式化地表示为：

$$R(U, D, \mathrm{dom}, F)$$

R	关系名
U	组成该关系的属性名集合
D	属性组U中属性所来自的域
dom	属性向域的映象集合
F	属性间的数据依赖关系集合

定义关系模式

例：

导师和研究生出自同一个域——人，

取不同的属性名，并在模式中定义属性向域的映象，即说明它们分别出自哪个域：

dom（SUPERVISOR-PERSON）
= dom（POSTGRADUATE-PERSON）
= PERSON

定义关系模式

- 关系模式通常可以简记为

  $R(U)$　　或　　$R(A_1, A_2, \cdots, A_n)$

  R　　关系名

  $A_1, A_2, \cdots, A_n$　　属性名

注：域名及属性向域的映象常常直接说明为属性的类型、长度；

关系模式与关系

- 关系模式
  - 对关系的描述；
  - 静态的、稳定的；
- 关系
  - 关系模式在某一时刻的状态或内容；
  - 动态的、随时间不断变化的；
- 关系模式和关系往往统称为关系；
- 通过上下文加以区别；

03

关系数据库

关系数据库

在一个给定的应用领域中，所有实体及实体

之间联系的关系的集合构成一个关系数据库。

关系数据库的型与值

- 关系数据库也有型和值之分；
- 关系数据库的型：称为关系数据库模式，是对关系数据库的描述；
  - 若干域的定义；
  - 在这些域上定义的若干关系模式；
- 关系数据库的值：是这些关系模式在某一时刻对应的关系的集合，通常简称为关系数据库；

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

目 录

关系操作集合

1）常用的关系操作：

- 查询的表达能力是其中最主要的部分；
- 查询
  - 选择、投影、连接、除、交、并、差、笛卡儿积
- 数据更新
  - 插入、删除、修改

关系操作集合

2）关系操作的特点：

- 集合操作方式，即操作的对象和结果都是集合。
  - 非关系数据模型的数据操作方式：一次一记录；
  - 关系数据模型的数据操作方式：一次一集合；

关系操作集合

3）关系数据语言的种类：

- 关系代数（代数方式）：用对关系的运算来表达查询要求；

- 关系演算（逻辑方式）：用谓词来表达查询要求；

关系操作集合

3）关系数据语言的种类

- 关系演算语言：用谓词来表达查询要求；
  - 元组关系演算语言
    - 谓词变元的基本对象是元组变量；
    - 典型代表：APLHA，QUEL；
  - 域关系演算语言
    - 谓词变元的基本对象是域变量；
    - 典型代表：QBE

关系操作集合

3）关系数据语言的种类（续）：

- 关系代数语言
  - 用对关系的运算来表达查询要求；
  - 典型代表：ISBL；
- 具有关系代数和关系演算双重特点的结构化查询语言
  - 典型代表：SQL

关系操作集合

4) 关系数据语言的特点：

- 关系语言是一种高度非过程化的集合操作语言：
  - 存取路径的选择由DBMS的优化机制来完成；
  - 用户不必用循环结构就可以完成数据操作；
- 能够嵌入高级语言中使用；
- 关系代数、元组关系演算和域关系演算三种语言在表达能力上完全等价；

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

目 录

关系的完整性

- 关系模型的完整性规则是对关系的某种约束条件。
- 关系模型中三类完整性约束：
  - 实体完整性
  - 参照完整性
  - 用户定义的完整性

注意：实体完整性和参照完整性是关系模型必须满足的完整性约束条件，被称作是关系的两个不变性，应该由关系系统自动支持。

一、实体完整性

- 实体完整性规则（Entity Integrity）

若属性*A*是基本关系*R*的主属性，则属性*A*不能取空值。空值就是“不存在”或“无意义”的值。

例：

学生(学号，身份证号，姓名，性别，年龄，专业号)

学号属性为主码，则其不能取空值

一、实体完整性

关系模型必须遵守实体完整性规则的原因：

(1) 实体完整性规则是针对基本关系而言的。一个基本表通常对应现实世界的一个实体集或多对多联系。

(2) 现实世界中的实体和实体间的联系都是可区分的，即它们具有某种唯一性标识。

(3) 相应地，关系模型中以主码作为唯一性标识。

一、实体完整性

注意：

实体完整性规则规定基本关系的所有主属性都不能取空值；

例：

选修（学号，课程号，成绩）

“学号、课程号”为主码，则两个属性都不能取空值。

二、参照完整性

1. 关系间的引用

2. 外码

3. 参照完整性规则

1、关系间的引用

- 在关系模型中实体及实体间的联系都是用关系来描述的，因此可能存在着关系与关系间的引用。

例1　学生实体、专业实体以及专业与学生间的一对多联系

学生（学号，姓名，性别，专业号，年龄）

专业（专业号，专业名）

1、关系间的引用

例1：学生（学号，姓名，性别，专业号，年龄）

学号	姓名	性别	专业号	年龄
801	张三	女	01	19
802	李四	男	01	20
803	王五	男	01	20
804	赵六	女	02	20
805	钱七	男	02	19

专业（专业号，专业名）

专业号	专业名
01	信息
02	数学
03	计算机

2、外码（Foreign Key）

- 设*F*是基本关系*R*的一个或一组属性，但不是关系*R*的码。如果F与基本关系*S*的主码K_s相对应，则称F是基本关系*R*的外码（Foreign Key）；
- 基本关系*R*称为参照关系（Referencing Relation）
- 基本关系*S*称为被参照关系（Referenced Relation）或目标关系（Target Relation）。

2、外码（Foreign Key）

例1　学生实体、专业实体以及专业与学生间的一对多联系

学生（学号，姓名，性别，专业号，年龄）

专业（专业号，专业名）

2、外码（Foreign Key）

说明：

- 关系 R 和 S 不一定是不同的关系；
- 目标关系 S 的主码 K_s 和参照关系的外码 F 必须定义在同一个（或一组）域上；
- 外码并不一定要与相应的主码同名；
- 当外码与相应的主码属于不同关系时，往往取相同的名字，以便于识别；

2、外码（Foreign Key）

例3　学生实体及其内部的领导联系(一对多)

学生（学号，姓名，性别，专业号，年龄，班长）

学号	姓名	性别	专业号	年龄	班长
801	张三	女	01	19	802
802	李四	男	01	20	
803	王五	男	01	20	802
804	赵六	女	02	20	805
805	钱七	男	02	19	

3、参照完整性规则

- 若属性（或属性组）*F* 是基本关系 *R* 的外码，它与基本关系 *S* 的主码 K_s 相对应（基本关系 *R* 和 *S* 不一定是不同的关系），则对于 *R* 中每个元组在 *F* 上的值必须为：
  - 或者取空值（*F* 的每个属性值均为空值）；
  - 或者等于*S* 中某个元组的主码值；

3、参照完整性规则

举例：学生（学号，姓名，性别，专业号，年龄）

“专业号”的属性只取下面两类值：

（1）空值，表示尚未给该学生分配专业；

（2）非空值，这时该值必须是专业关系中某个元组的“专业号”值，表示该学生不可能分配到一个不存在的专业中；

3、参照完整性规则

举例：学生（学号，姓名，性别，专业号，年龄，班长）

“班长”属性值可以取两类值：

（1）空值，表示该学生所在班级尚未选出班长，或该学生本人即是班长；

（2）非空值，这时该值必须是本关系中某个元组的学号值

3、参照完整性规则

例2　学生、课程、学生与课程之间的多对多联系：

学生（学号，姓名，性别，专业号，年龄）

课程（课程号，课程名，学分）

选修（学号，课程号，成绩）

三、用户定义的完整性

- 用户定义的完整性是针对某一具体关系数据库的约束条件，反映某一具体应用所涉及的数据必须满足的语义要求。
- 关系模型应提供定义和检验这类完整性的机制，以便用统一的系统的方法处理它们，而不要由应用程序承担这一功能。

三、用户定义的完整性

例：

课程(<u>课程号</u>，课程名，学分)

- 非主属性“课程名”也不能取空值；
- “学分”属性只能取值{1，2，3，4}；

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

目 录

关系代数

关系代数

一种抽象的查询语言，用对关系的运算来表达查询。

关系代数运算的三个要素：

- ✓ 运算对象：关系
- ✓ 运算结果：关系
- ✓ 运算符：四类

关系代数的四类运算符

表：关系代数运算符

运算符		含义	运算符		含义
集合运算符	∪	并	比较运算符	>	大于
	−	差		≥	大于等于
	∩	交		<	小于
	×	笛卡儿积		≤	小于等于
				=	等于
				<>	不等于

关系代数的四类运算符

表：关系代数运算符

运算符	含义		运算符	含义	
专门的关系运算符	σ	选择	逻辑运算符	¬	非
	π	投影		∧	与
	⋈	连接		∨	或
	÷	除			

关系代数的四类运算符

- 集合运算符(并、差、交、广义笛卡儿积)
  - ✓将关系看成元组的集合；
  - ✓运算是从关系的“水平”方向即行的角度来进行；
- 专门的关系运算符(选择、投影、连接、除)
  - ✓不仅涉及行而且涉及列；
- 算术比较符
  - ✓辅助专门的关系运算符进行操作；
- 逻辑运算符
  - ✓辅助专门的关系运算符进行操作；

关系代数的表示记号

（1） R，t∈R，t[A_i]

设关系模式为$R(A_1, A_2, \ldots, A_n)$，

- 它的一个关系设为R；
- $t \in R$ 表示 t 是 R 的一个元组；
- $t[A_i]$则表示元组t中相应于属性A_i的一个分量；

关系代数的表示记号

（2） A，t[A]，$\overline{A}$

- 若$A=\{A_{i1}, A_{i2}, \ldots, A_{ik}\}$，其中$A_{i1}, A_{i2}, \ldots, A_{ik}$是$A_1, A_2, \ldots, A_n$中的一部分，则$A$称为属性列或属性组 。
- $t[A]=(t[A_{i1}], t[A_{i2}], \ldots, t[A_{ik}])$，表示元组$t$在属性列$A$上诸分量的集合。
- $\overline{A}$则表示$\{A_1, A_2, \ldots, A_n\}$中去掉$\{A_{i1}, A_{i2}, \ldots, A_{ik}\}$后剩余的属性组。

关系代数的表示记号

（3） $\overset{\frown}{t_r\ t_s}$

R为n目关系，S为m目关系。$t_r \in R$，$t_s \in S$，$\overset{\frown}{t_r\ t_s}$称为元组的连接(元组的串接)。它是一个$n + m$列的元组，前$n$个分量为$R$中的一个$n$元组，后$m$个分量为$S$中的一个$m$元组

关系代数的表示记号

（4）象集Z_x

给定一个关系$R(X, Z)$，X 和Z 为属性组。当$t[X] = x$ 时，x 在R 中的象集（Images Set）为

$$Z_x = \{t[Z] \mid t \in R,\ t[X] = x\}$$

它表示R 中属性组X上值为x的诸元组在Z上分量的集合。

象集举例

R	
x_1	Z_1
x_1	Z_2
x_1	Z_3
x_2	Z_2
x_2	Z_3
x_3	Z_1
x_3	Z_3

象集举例

- X_1在R中的象集

  $Z_{x1} = \{Z_1, Z_2, Z_3\}$

- X_2在R中的象集

  $Z_{x2} = \{Z_2, Z_3\}$

- X_3在R中的象集

  $Z_{x3} = \{Z_1, Z_3\}$

传统的集合运算

- 并
- 差
- 交
- 广义笛卡儿积

并（Union）

- R 和 S
  - 具有相同的目n（即两个关系都有n个属性）
  - 相应的属性取自同一个域；

- $R \cup S$
  - 仍为n目关系，由属于R或属于S的元组组成

$$R \cup S = \{ t \mid t \in R \vee t \in S \}$$

注意：R ∪ S = S ∪ R

并（Union）

R

A	B	C
a1	b1	c1
a1	b2	c2
a2	b2	c1

S

A	B	C
a1	b2	c2
a1	b3	c2
a2	b2	c1

$R \cup S$

A	B	C
a1	b1	c1
a1	b2	c2
a1	b3	c2
a2	b2	c1

差（Difference）

- R 和S
  - 具有相同的目n；
  - 相应的属性取自同一个域；

- $R - S$
  - 仍为n目关系，由属于R而不属于S的所有元组组成：

$$R - S = \{ t \mid t \in R \wedge t \notin S \}$$

注意： $R - S \neq S - R$

差（Difference）

R

A	B	C
a1	b1	c1
a1	b2	c2
a2	b2	c1

S

A	B	C
a1	b2	c2
a1	b3	c2
a2	b2	c1

R-S

A	B	C
a1	b1	c1

交（Intersection）

- R 和S
  - 具有相同的目n ；
  - 相应的属性取自同一个域；

- $R \cap S$
  - 仍为n目关系，由既属于R又属于S的元组组成：

$$R \cap S = \{ t \mid t \in R \wedge t \in S \}$$

注意： $R \cap S = S \cap R$

交（Intersection）

R

A	B	C
a1	b1	c1
a1	b2	c2
a2	b2	c1

S

A	B	C
a1	b2	c2
a1	b3	c2
a2	b2	c1

$R \cap S$

A	B	C
a1	b2	c2
a2	b2	c1

广义笛卡儿积（Extended Cartesian Product）

- R，n目关系，k_1个元组
- S，m目关系，k_2个元组
- $R \times S = \{ t_r \frown t_s \mid t_r \in R \wedge t_s \in S \}$
  - 列：（$n+m$）列的元组的集合
    - ✓ 元组的前n列是关系R的一个元组；
    - ✓ 后m列是关系S的一个元组；
  - 行：$k_1 \times k_2$个元组

广义笛卡儿积（Extended Cartesian Product）

R

A	B	C
a1	b1	c1
a1	b2	c2
a2	b2	c1

S

A	B	C
a1	b2	c2
a1	b3	c2
a2	b2	c1

$R \times S$

A	B	C	A	B	C
a1	b1	c1	a1	b2	c2
a1	b1	c1	a1	b3	c2
a1	b1	c1	a2	b2	c1
a1	b2	c2	a1	b2	c2
a1	b2	c2	a1	b3	c2
a1	b2	c2	a2	b2	c1
a2	b2	c1	a1	b2	c2
a2	b2	c1	a1	b3	c2
a2	b2	c1	a2	b2	c1

专门的关系运算

- 选择（对关系进行水平分割）
- 投影（对关系进行垂直分割）
- 连接（关系的合并）
- 除

学生—课程数据库

Student（学生）

学号 Sno	姓名 Sname	性别 Ssex	年龄 Sage	所在系 Sdept
201215121	李勇	男	20	CS
201215122	刘晨	女	19	CS
201215123	王敏	女	18	MA
201215125	张立	男	19	IS

(a)

学生—课程数据库

课程号 Cno	课程名 Cname	先行课 Cpno	学分 Ccredit
1	数据库	5	4
2	数学		2
3	信息系统	1	4
4	操作系统	6	3
5	数据结构	7	4
6	数据处理		2
7	PASCAL语言	6	4

(b)

Course
（课程）

学生—课程数据库

学号 Sno	课程号 Cno	成绩 Grade
201215121	1	92
201215121	2	85
201215121	3	88
201215122	2	90
201215122	3	80

SC
（选修关系）

例7　例8

选择（Selection）

1）选择又称为限制（Restriction）

2）选择运算符的含义：

- 在关系R 中选择满足给定条件的诸元组

$$\sigma_F(R) = \{ t \mid t \in R \wedge F(t) = \text{'真'} \}$$

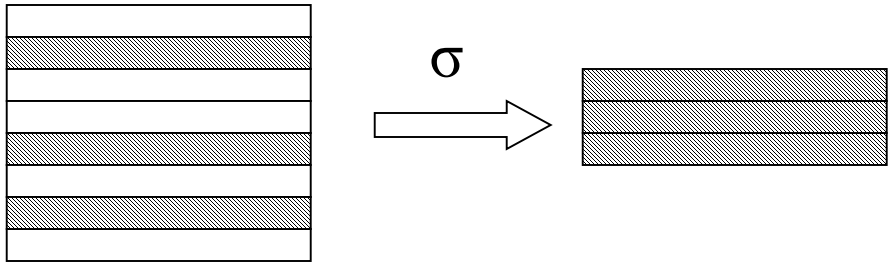
- F：选择条件，是一个逻辑表达式，基本形式为：

  $[\neg(\ ]\ X_1\ \theta\ Y_1\ [\ )][\ \phi\ [\neg(\ ]\ X_2\ \theta\ Y_2\ [\ )]]\ldots$

  - θ：比较运算符（>，≥，<，≤，＝或<>）
  - X_1，Y_1等：属性名、常量、简单函数，属性名也可以用它的序号来代替；
  - Φ：逻辑运算符（ ¬ 、∧或∨）
  - [ ]：表示任选项
  - …：表示上述格式可以重复下去

选择（Selection）

3）选择运算是从行的角度进行的运算：



选出那些满足条件的元组。

选择（Selection）

[例1] 查询信息系（IS系）全体学生

$$\sigma_{Sdept = 'IS'}(Student)$$

或

$$\sigma_{5 = 'IS'}(Student)$$

结果：

Sno	Sname	Ssex	Sage	Sdept
201215125	张立	男	19	IS

选择（Selection）

[例2]　查询年龄小于20岁的学生

$$\sigma_{Sage < 20}(Student)$$

或

$$\sigma_{4 < 20}(Student)$$

结果：

Sno	Sname	Ssex	Sage	Sdept
201215122	刘晨	女	19	CS
201215123	王敏	女	18	MA
201215125	张立	男	19	IS

投影（Projection）

1）投影运算符的含义

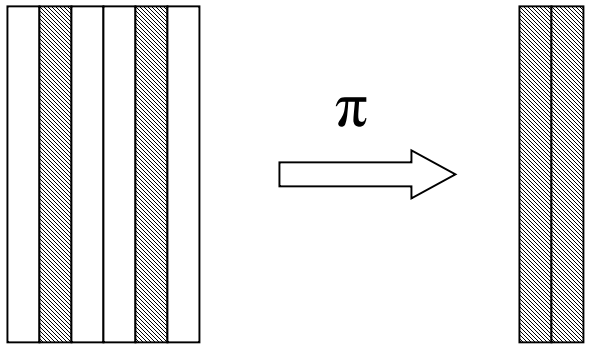
- 从R 中选择出若干属性列组成新的关系：

$$\pi_A(R) = \{ t[A] \mid t \in R \}$$

A：R 中的属性列

投影（Projection）

2）投影操作主要是从列的角度进行运算



- 但投影之后不仅取消了原关系中的某些列，而且还可能取消某些元组（避免重复行）。

投影（Projection）

[例3]　查询学生的姓名和所在系

即求Student关系上学生姓名和所在系两个属性上的投影

$$\pi_{Sname,\ Sdept}(Student)$$

或　　$$\pi_{2,\ 5}(Student)$$

投影（Projection）

结果：

Sname	Sdept
李勇	CS
刘晨	CS
王敏	MA
张立	IS

投影（Projection）

[例4]　查询学生关系Student中都有哪些系

$$\pi_{Sdept}(Student)$$

结果：

取消重复元组

Sdept
CS
IS
MA

综合举例

［例 8］ 查询选修了2号课程的学生的学号。

$$\pi_{Sno}(\sigma_{Cno='2'}(SC))$$

$$=\{201215121, 201215122\}$$

连接（Join）

1）连接也称为 θ 连接

2）连接运算的含义

- 从两个关系的笛卡儿积中选取属性间满足一定条件的元组

$$R \underset{A\theta B}{\bowtie} S = \{ \widehat{t_r t_s} \mid t_r \in R \wedge t_s \in S \wedge t_r[A] \ \theta \ t_s[B] \}$$

  - A和B：分别为R和S上度数相等且可比的属性组；
  - θ：比较运算符 ；

连接（Join）

2）连接运算的含义：

- 连接运算从 R 和 S 的广义笛卡儿积 $R\times S$中选取（R关系）在 A 属性组上的值与（S关系）在 B 属性组上值满足比较关系的元组。

连接（Join）

3）两类常用连接运算

- 等值连接（equijoin）
  - θ 为“＝”的连接运算称为等值连接；
  - 等值连接的含义：
    - 从关系R与S的广义笛卡儿积中选取A、B属性值相等的那些元组，即等值连接为：

$$R \underset{A=B}{\bowtie} S = \{ \widehat{t_r t_s} \mid t_r \in R \wedge t_s \in S \wedge t_r[A] = t_s[B] \}$$

连接（Join）

- 自然连接（Natural join）
  - 自然连接是一种特殊的等值连接
    - 两个关系中进行比较的分量必须是相同的属性组；
    - 在结果中把重复的属性列去掉；

连接（Join）

- 自然连接（Natural join）
  - 自然连接的含义：

    R和S具有相同的属性组B，U为R和S的全体属性集合。

$$R \bowtie S = \{ \widehat{t_r t_s}[U\text{-}B] \mid t_r \in R \wedge t_s \in S \wedge t_r[B] = t_s[B] \}$$

连接（Join）

外连接（Outer Join）

如果把舍弃的元组也保存在结果关系中，而在其他属性上填空值(Null)，这种连接就叫做外连接（OUTER JOIN）。

A	B	C	E
a_1	b_1	5	3
a_1	b_2	6	7
a_2	b_3	8	10
a_2	b_3	8	2
a_2	b_4	12	NULL
NULL	b_5	NULL	2

(a) 外连接

连接（Join）

- 左外连接(LEFT OUTER JOIN 或 LEFT JOIN)

  如果只把左边关系R中要舍弃的元组保留就叫做左外连接。

- 右外连接(RIGHT OUTER JOIN 或 RIGHT JOIN)

  如果只把右边关系S中要舍弃的元组保留就叫做右外连接。

左外连接和右外连接

A	B	C	E
a_1	b_1	5	3
a_1	b_2	6	7
a_2	b_3	8	10
a_2	b_3	8	2
a_2	b_4	12	NULL

(b) 左外连接

A	B	C	E
a_1	b_1	5	3
a_1	b_2	6	7
a_2	b_3	8	10
a_2	b_3	8	2
NULL	b_5	NULL	2

(c) 右外连接

连接（Join）

注意：

- ✓ 选择和投影运算的时间复杂度为n数量级（n为元组个数）；
- ✓ 连接运算的时间复杂度为n×m数量级（n和m分别是两个关系中的元组个数）；
- ✓ 为了减少关系运算的时间复杂度，从而提高效率，通常先做选择运算，再做投影运算，最后做连接运算。

除（Division）

- 给定关系R（X，Y）和S（Y，Z），其中X，Y，Z为属性组。R中的Y与S中的Y可以有不同的属性名，但必须出自相同的域集。
- R与S的除运算得到一个新的关系P(X)，P是R中满足下列条件的元组在X属性列上的投影：元组在X上分量值x的象集Y_x包含S在Y上投影的集合。

$$R \div S = \{ t_r[X] \mid t_r \in R \wedge \pi_Y(S) \subseteq Y_x \}$$

Y_x：x在R中的象集，$x = t_r[X]$

除（Division）

R

A	B	C
a_1	b_1	c_2
a_2	b_3	c_7
a_3	b_4	c_6
a_1	b_2	c_3
a_4	b_6	c_6
a_2	b_2	c_3
a_1	b_2	c_1

S

B	C	D
b_1	c_2	d_1
b_2	c_1	d_1
b_2	c_3	d_2

R÷S
A
a_1

分析

- 在关系R中，A可以取四个值{a1，a2，a3，a4}
  - a_1的象集为 $\{(b_1, c_2), (b_2, c_3), (b_2, c_1)\}$;
  - a_2的象集为 $\{(b_3, c_7), (b_2, c_3)\}$;
  - a_3的象集为 $\{(b_4, c_6)\}$;
  - a_4的象集为 $\{(b_6, c_6)\}$;

- S在(B，C)上的投影为

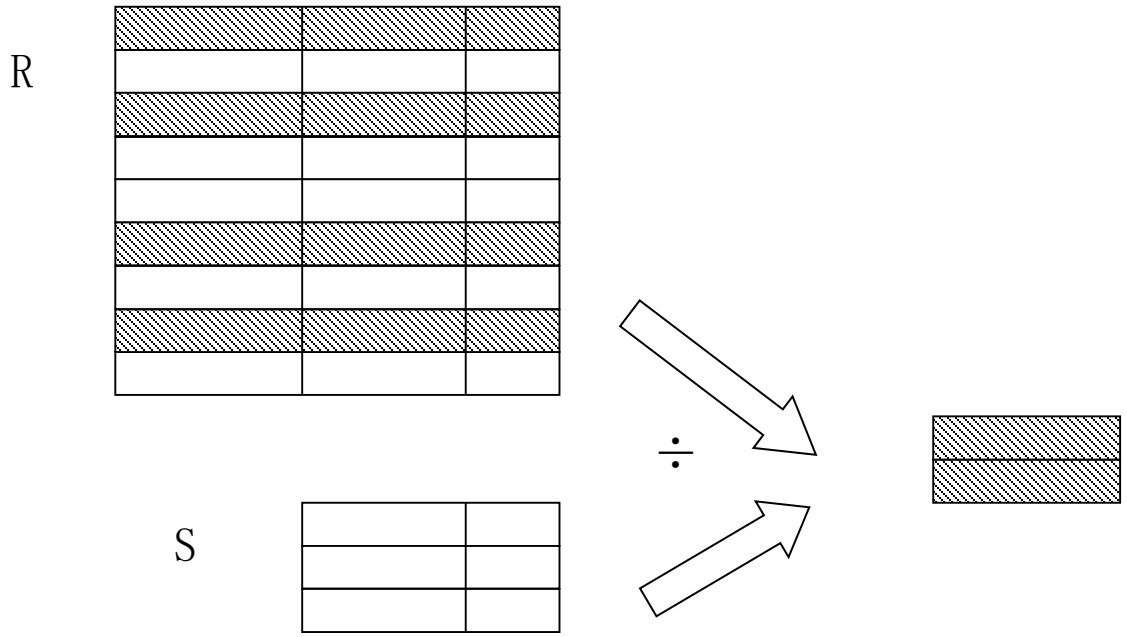
  $\{(b_1, c_2), (b_2, c_1), (b_2, c_3)\}$

  只有a_1的象集包含了S在(B，C)属性组上的投影

  所以 $R \div S = \{a_1\}$

除（Division）

2）除操作是同时从行和列角度进行运算



综合举例

以学生-课程数据库为例

[例7] 查询至少选修1号课程和3号课程的学生号码

首先建立一个临时关系K:

Cno
1
3

然后求：$\pi_{Sno,Cno}(SC) \div K$

综合举例

例 7续　$\pi_{Sno.Cno}(SC) \div K$

201215121象集{1，2，3}

201215122象集{2，3}

$\pi_{Cno}(K)=\{1，3\}$

于是：$\pi_{Sno.Cno}(SC) \div K = \{201215121\}$

Sno	Cno
201215121	1
201215121	2
201215121	3
201215122	2
201215122	3

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

THANK YOU !

饮水思源 爱国荣校