



第二章 关系数据库

主要内容

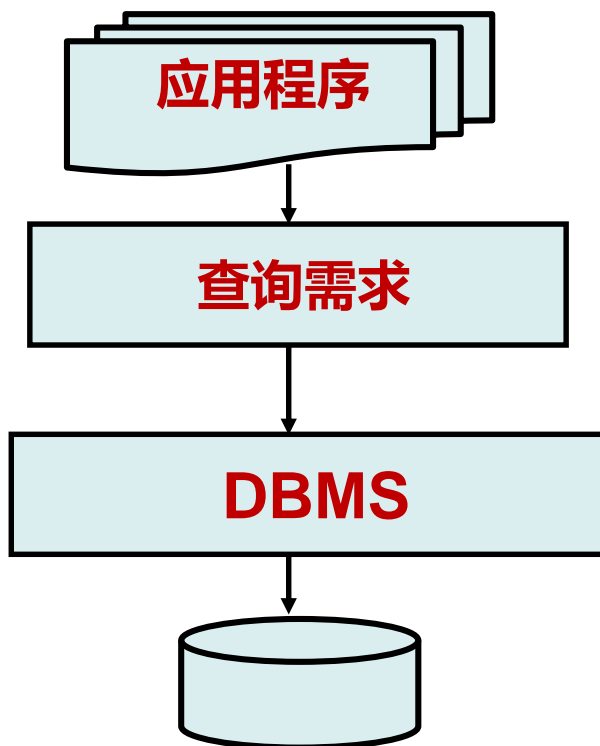
- 如何用离散数学中的代数系统来定义关系数据库？
(2.1、2.2、2.3内容)
- 如何定义代数运算，用数学语言描述查询需求？
(2.4、2.5内容分别介绍了代数语言和谓词语言)



关系代数系统的引入动机

问题：为什么需要代数语言？

关系代数产生动机



如何描述查询需求？

查询需求描述自动化
查询需求描述自然化

- 关系数据库是表集
- 定义完备的表操作
- 用表的运算表达式描述查询需求
- 系统解决表达式与查询语言的转化、执行

2.1 关系数据结构及形式化定义

2.2 关系操作

2.3 关系的完整性

2.4 关系代数

2.5 关系演算



2.1 关系数据结构及形式化定义

- **动机** 能否将一个关系数据库用一个集合代数系统来表示？
- **作用** 应用对DB的查询、处理请求，能够转换成相应的关系操作表达式，再设计DBMS能够自动执行表达式。
- **思考** 关系代数语言虽然用户不友好，但形成一套理论，可以指导数据库研究
用户友好语言可以自动转换为关系代数语言，
例如 SQL



2.1 关系数据结构及形式化定义

2.1.1 关系

2.1.2 关系模式

2.1.3 关系数据库



2.1.1 关系

- 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

- 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

- 建立在集合代数的基础上



关系(续)

1. 域 (Domain)
2. 笛卡尔积 (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。例:

- 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’ }

-



2. 笛卡尔积 (Cartesian Product)

- 笛卡尔积

给定一组域 D_1, D_2, \dots, D_n , 这些域中可以有相同的。

D_1, D_2, \dots, D_n 的笛卡尔积为:

$$D_1 \times D_2 \times \dots \times D_n =$$

$$\{ (d_1, d_2, \dots, d_n) \mid d_i \in D_i, i=1, 2, \dots, n \}$$

- 所有域的所有取值的一个组合
- 不能重复



例如，有三个域：

D1=导师集合 SUPERVISOR=张清玫，刘逸

D2=专业集合 SPECIALITY=计算机专业，信息专业

D3= 研究生集合 POSTGRADUATE=李勇，刘晨，王敏

则D1， D2， D3的笛卡尔积为

**D1XD2XD3={ (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)}**




表 2.1 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



笛卡尔积（续）

- 元组（Tuple）

- 笛卡尔积中每一个元素 (d_1, d_2, \dots, d_n) 叫作一个 n 元组（ n -tuple）或简称元组(Tuple)
- (张清玫, 计算机专业, 李勇)、(张清玫, 计算机专业, 刘晨)等都是元组

- 分量（Component）

- 笛卡尔积元素 (d_1, d_2, \dots, d_n) 中的每一个值 d_i 叫作一个分量
- 张清玫、计算机专业、李勇、刘晨等都是分量



笛卡尔积（续）

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

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



例

- $D_1 = \{\text{Zhang, Li, Wang}\}$, $D_2 = \{21, 22\}$, $D_3 = \{M, F\}$

- $D_1 \times D_2 \times D_3$ 的笛卡尔积为:

D1	D2	D3
Zhang	21	M
Zhang	21	F
Zhang	22	M
Zhang	22	F
Li	21	M
Li	21	F
Li	22	M
Li	22	F
Wang	21	M
Wang	21	F
Wang	22	M
Wang	22	F

3. 关系 (Relation)

1) 关系

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

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

- R : 关系名
- n : 关系的目或度 (Degree)



例. 已知三个域

D1=导师集合S={张清玫,刘逸}

D2=专业集合SP={计算机,信息}

D3=学生集合P={李勇,刘晨,王敏}

计算思考

从笛卡尔集中取出一个子集，选择有意义的结果组成关系

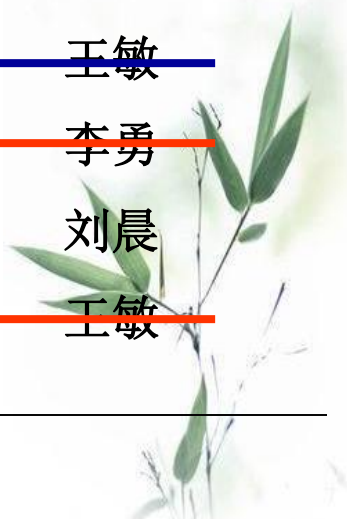
R=(导师，专业，研究生)

•一个研究生只能有一个专业，如李勇和王敏是计算机专业、刘晨是信息专业

现在导师与研究生是什么关系？

限定一个学生只能有一个导师，如张是计算机导师，刘是信息专业导师

导师S	专业SP	研究生P
张清玫	计算机	李勇
张清玫	计算机	刘晨
张清玫	计算机	王敏
张清玫	信息	李勇
张清玫	信息	刘晨
张清玫	信息	王敏
刘逸	计算机	李勇
刘逸	计算机	刘晨
刘逸	计算机	王敏
刘逸	信息	李勇
刘逸	信息	刘晨
刘逸	信息	王敏



关系（续）

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

表 2.2 SAP 关系

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



关系（续）

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) 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）
- n 目关系必有 n 个属性



关系（续）

4) 码

- 候选码

设R是一个关系模式，K是R的子集。若对于R中任意两个不同的元组 $t_1 \neq t_2$, 有 $t_1[K] \neq t_2[K]$, 且K是这样的最小集，则称K是R的候选码

- 候选码可能有多个；
- 唯一确定元组；
- 最小化；

- 主码 若一个关系有多个候选码，则选定其中一个为主码
- 全码 关系模式的所有属性组是这个关系模式的候选码
- 主属性 候选码的诸属性称为主属性，不包含在任何侯选码中的属性称为非主属性或非码属性



候选码

候选码

外码

主码

学号	身份证号	姓名	性别	系别
0101	4201111985	张	男	CS
0102	36020211986	李	女	CS
0203	3671111985	赵	男	MA

系号	系名
CS	计算机
EN	英语
MA	数学系

主码

主码

学号	课号	成绩
0101	CS145	88
0101	CS148	90
0102	CS180	87
0203	CS145	78

课号	课名
CS145	数据库
CS148	操作系统
CS180	数据结构

全码关系

- 学校中某一门课程由多个教师讲授，他们使用相同的一套参考书。每个教员可以讲授多门课程，每种参考书可以供多门课程使用。

Teaching

课程C	教员T	参考书B
物理	李勇	普通物理学
物理	李勇	光学原理
物理	李勇	物理习题集
物理	王军	普通物理学
物理	王军	光学原理
物理	王军	物理习题集
数学	李勇	数学分析
数学	李勇	微分方程
数学	李勇	高等代数
数学	张平	数学分析
数学	张平	微分方程
数学	张平	高等代数
...

Teaching具有唯一候选码(C, T, B)，即全码



关系（续）

5) 三类关系

基本关系（基本表或基表table）

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view

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



关系（续）

6)基本关系的性质

- ① 列是同质的,即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
- ③ 列的顺序无所谓,, 列的次序可以任意交换
- ④ 任意两个元组的候选码不能相同
- ⑤ 行的顺序无所谓, 行的次序可以任意交换
- 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。



关系性质1——同质的列

学号	姓名	性别	年龄	曾用名
9901	张三	男	20	张狗子
9902	李四	女	18	李朋
王五	9903	1	19	王麻子
9904	赵六	2	1981/2/28	赵薇



关系性质2—不同的属性名

姓名2
曾用名

学号	姓名	性别	年龄	姓名
9901	张三	男	20	张狗子
9902	李四	女	18	李朋
9903	王五	男	19	王麻子
9904	赵六	女	20	赵薇



关系性质3—属性无序

学号	姓名	性别	年龄	曾用名
9901	张三	男	20	张狗子
9902	李四	女	18	李朋
9903	王五	男	19	王麻子
9904	赵六	女	20	赵薇

学号	姓名	曾用名	性别	年龄
9901	张三	张狗子	男	20
9902	李四	李朋	女	18
9903	王五	王麻子	男	19
9904	赵六	赵薇	女	20

关系性质4—元组不重复

学号	姓名	性别	年龄	姓名
9901	张三	男	20	张狗子
9902	李四	女	18	李朋
9903	王五	男	19	王麻子
9904	赵六	女	20	赵薇
9901	张三	男	20	张狗子

重复的
元组!

×



关系性质5—元组无序

学号	姓名	性别	年龄
9901	张三	男	20
9902	李四	女	18
9903	王五	男	19
9904	赵六	女	20

学号	姓名	性别	年龄
9904	赵六	女	20
9901	张三	男	20
9903	王五	男	19
9902	李四	女	18



关系性质6—分量是原子

父	母	孩子	
		大	小
李男	丁女	李一	李二
王男	肖女	王一	

父	母	孩子
李男	丁女	李一 李二
王男	肖女	王一

非规范化关系

父	母	大孩	小孩
李男	丁女	李一	李二
王男	肖女	王一	

父	母	孩子
李男 李男 王男	丁女 丁女 肖女	李一 李二 王一

规范化关系



在许多实际关系数据库产品中，基本表并不完全具有这六条性质。

举例

- 列的顺序无所谓，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
 - 但也有许多关系数据库产品没有遵循这一性质，例如**FoxPro**仍然区分了属性顺序
- 任意两个元组不能完全相同，由笛卡尔积的性质决定
 - 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



在许多实际关系数据库产品中，基本表并不完全具有这六条性质。

举例

- 行的顺序无所谓，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



2.1 关系数据结构

2.1.1 关系

2.1.2 关系模式

2.1.3 关系数据库



2.1.2 关系模式

1. 什么是关系模式
2. 定义关系模式
3. 关系模式与关系



1. 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



2. 定义关系模式

关系模式可以形式化表示：

$R(U, D, DOM, F)$

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



例：Student关系模式的定义

– Student(U,D,dom,F)

- $U=\{\text{sno}, \text{name}, \text{age}\}$
- $D=\{\text{CHAR}, \text{INT}\}$
- $\text{Dom}=\{\text{dom}(\text{sno})=\text{dom}(\text{name})=\text{CHAR}, \text{dom}(\text{age})=\text{INT}\}$
- $F=\{\text{sno} \rightarrow \text{name}, \text{sno} \rightarrow \text{age}\}$

• 关系模式通常简写为R(U)，或R(A1,A2,...,An)



定义关系模式 (续)

关系模式通常可以简记为

$R(U)$ 或 $R(A_1, A_2, \dots, A_n)$

- R : 关系名
- A_1, A_2, \dots, 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



3. 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



2.1 关系数据结构

2.1.1 关系

2.1.2 关系模式

2.1.3 关系数据库



2.1.3 关系数据库

- 关系数据库
 - 在一个给定的应用领域中，所有实体及实体间联系的关系的集合构成一个关系数据库。



2.1 关系模型概述

2.2 关系操作

2.3 关系的完整性

2.4 关系代数

2.5 关系演算



2.2.1 基本关系操作

- 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
 - 选择、投影、并、差、笛卡尔积是5种基本操作
- 关系操作的特点
 - 集合操作方式：操作的对象和结果都是集合，一次一集合的方式



扩展的关系代数

- 改名
- 广义投影
- 赋值
- 外连接
- 外部并
- 半连接
- 聚集操作



2.2.2 关系数据库语言的分类

- 关系代数语言
 - 用对关系的运算来表达查询要求

- 关系演算语言
 - 用谓词来表达查询要求

元组关系演算语言

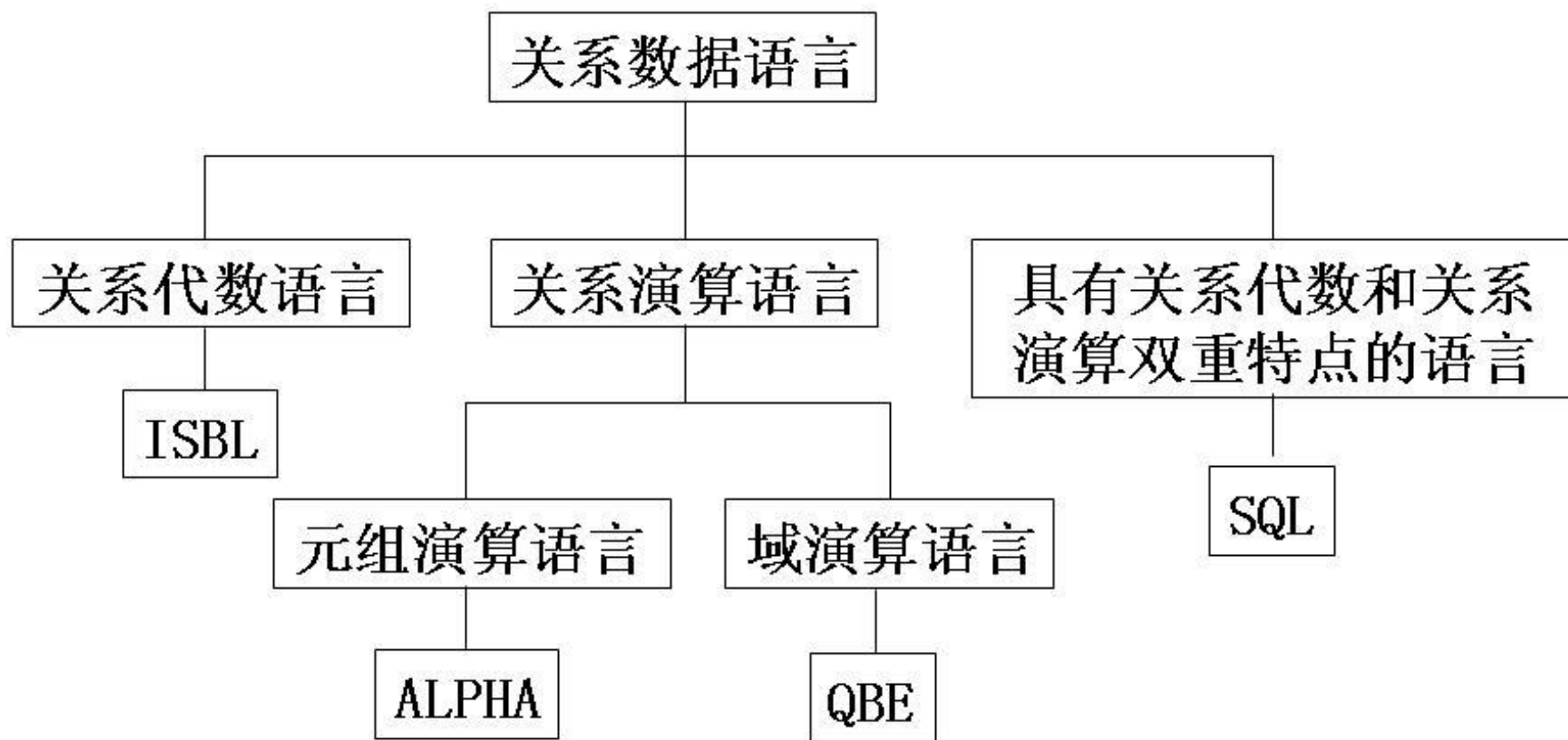
- 谓词变元的基本对象是元组变量

域关系演算语言

- 谓词变元的基本对象是域变量



关系数据库语言的分类



2.1 关系数据结构及形式化定义

2.2 关系操作

2.3 关系的完整性

2.4 关系代数

2.5 关系演算



2.3 关系的完整性

2.3.1 关系的三类完整性约束

2.3.2 实体完整性

2.3.3 参照完整性

2.3.4 用户定义的完整性



2.3.1 关系的三类完整性约束

- 实体完整性和参照完整性

关系模型必须满足的完整性约束条件

称为关系的两个不变性，应该由关系系统自动支持

- 用户定义的完整性

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



2.3 关系的完整性

2.3.1 关系的三类完整性约束

2.3.2 实体完整性

2.3.3 参照完整性

2.3.4 用户定义的完整性



实体完整性

规则1：若属性A是基本关系R的主属性，则属性A不能为空值。

- 基本关系的所有主属性都不能取空值。



The diagram shows a table with three columns: 学号 (Student ID), 课程号 (Course ID), and 成绩 (Score). The first row contains the values S001, C001, and 80. The second row contains S001, an empty cell, and 90. The third row contains an empty cell, an empty cell, and 80. To the left of the table, there are two large 'X' marks. Two horizontal ovals are drawn across the second and third rows, highlighting the empty cells in the primary key columns. This illustrates that these rows violate the entity integrity rule because the primary key attribute (学号) is missing.

<u>学号</u>	<u>课程号</u>	成绩
S001	C001	80
S001		90
		80

实体完整性(续)

实体完整性规则的说明

- (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。
- (2) 现实世界中的实体是可区分的，即它们具有某种唯一性标识。
- (3) 关系模型中以主码作为唯一性标识。
- (4) 主码中的属性即主属性不能取空值。

主属性取空值，说明存在某个不可标识的实体，即存在不可区分的实体，这与第（2）点相矛盾，因此这个规则称为实体完整性



2.3关系的完整性

2.3.1关系的三类完整性约束

2.3.2 实体完整性

2.3.3 参照完整性

2.3.4 用户定义的完整性



2.3.3 参照完整性

1. 关系间的引用
2. 外码
3. 参照完整性规则



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）

- ❖ 学生关系引用了专业关系的主码“专业号”。
- ❖ 学生关系中的“专业号”值必须是确实存在的专业的专业号，即专业关系中有该专业的记录。

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

专业号	专业名
01	信息
02	数学
03	计算机



关系间的引用

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)

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

课程号	课程名	学分
01	数据库	4
02	数据结构	4
03	编译	4
04	PASCAL	2

学号	课程号	成绩
801	04	92
801	03	78
801	02	85
802	03	82
802	04	90
803	04	88

关系间的引用(续)

例3 学生实体及其内部的一对多联系

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	

- ❖ “学号”是主码，“班长”是外码，它引用了本关系的“学号”
- ❖ “班长”必须是确实存在的学生的学号



外码 (Foreign Key)

Student(sno,sname,sage,sdept)

Course(cno,cname)

- 外码

Sc(sno,cno,grade)

- 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Ks相对应，则称F是基本关系R的外码。

F≠K_r

基本关系R(F , ...)

参照关系
Referencing
Relation

基本关系S (K_s, ...)

被参照关系
Referenced Relation
或：目标关系
Target Relation

外码说明

- 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
- 关于取名
 - 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



候选码

候选码

外码

主码

学号	身份证号	姓名	性别	系别
0101	4201111985	张	男	CS
0102	36020211986	李	女	CS
0203	3671111985	赵	男	MA

系号	系名
CS	计算机
EN	英语
MA	数学系

主码

主码

学号	课号	成绩
0101	CS145	88
0101	CS148	90
0102	CS180	87
0203	CS145	78

课号	课名
CS145	数据库
CS148	操作系统
CS180	数据结构

参照完整性举例

例 1：学生实体与专业实体间的关系：

学生（学号，姓名，性别，专业号，年龄）

专业（专业号，专业名）

外码

参照关系

被参照关系

关系参照图

学生关系 $\xrightarrow{\text{专业号}}$ 专业关系

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）

被参照关系

被参照关系

参照关系

关系参照图

学生关系 $\xleftarrow{\text{学号}}$ 选修关系 $\xrightarrow{\text{课程号}}$ 课程关系

参照完整性举例

例 3：学生（学号，姓名，性别，专业号，年龄，班长）

参照关系

被参照关系

外码

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	



参照完整性规则

- 已知基本关系**R**的外码**F**（与基本关系**S**的主码**Ks**相对应），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。

$F \neq K_r$

基本关系**R**(**F** , ...)

基本关系**S** (**Ks**, ...)



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”属性只取下面两类值：
 - 空值，表示尚未给该学生分配专业
 - 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

选修（学号，课程号，成绩）

- “学号”和“课程号”是选修关系中的主属性，按照实体完整性和参照完整性规则，它们只能取相应被参照关系中已经存在的主码值



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：
 - 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；
 - 非空值，这时该值必须是本关系中某个元组的学号值



关系的完整性(续)

2.3.1 关系的三类完整性约束

2.3.2 实体完整性

2.3.3 参照完整性

2.3.4 用户定义的完整性



2.3.4 用户定义的完整性

- 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，以便使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



用户定义的完整性(续)

例:

课程(课程号, 课程名, 学分)

- “课程号” 属性必须取唯一值
- 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}



供应商关系S（主码是“供应商号”）

供应商号	供应商名	所在城市
B01	红星	北京
S10	宇宙	上海
T20	黎明	天津
Z01	立新	重庆

今要向关系P中插入新行，新行的值分别列出如下。哪些行能够插入？

- A、('037', '绿', null)
- B、(null, '黄', 'T20')
- C、('201', '红', 'T20')
- D、('105', '蓝', 'B01')
- E、('101', '黄', 'T11')

零件关系P（主码是“零件号”，外码是“供应商号”）

零件号	颜色	供应商号
010	红	B01
312	白	S10
201	蓝	T20



2.1 关系模型概述

2.2 关系数据结构

2.3 关系的完整性

2.4 关系代数

2.5 关系演算



2.4 关系代数

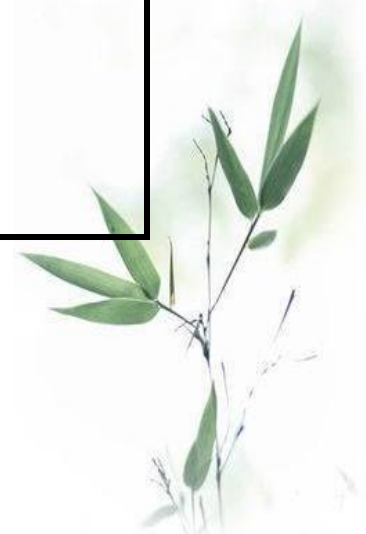
- 概述
- 传统的集合运算
- 专门的关系运算



概述

表2.4 关系代数运算符

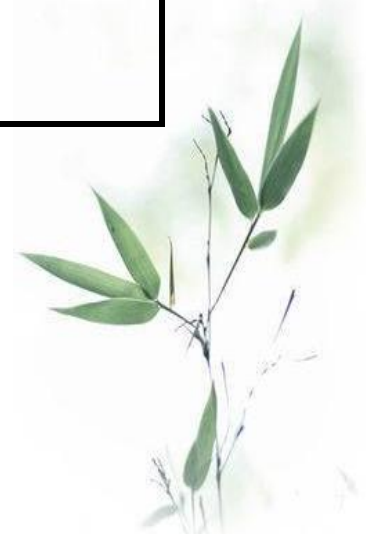
运算符		含义	运算符		含义
集合运算符	\cup	并	比较运算符	$>$	大于
	$-$	差		\geq	大于等于
	\cap	交		$<$	小于
	\times	笛卡尔积		\leq	小于等于
				$=$	等于
				$<>$	不等于



概述(续)

表2.4 关系代数运算符（续）

运算符	含义		运算符	含义	
专门的关 系运算符	σ π \bowtie \div	选择 投影 连接 除	逻辑运算 符	\neg \wedge \vee	非 与 或



2.4 关系代数

- 概述
- 传统的集合运算
- 专门的关系运算



1. 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

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



并运算

R

A	B	C
3	6	7
2	5	7
7	2	3
4	4	3

S

A	B	C
3	4	5
7	2	3

RUS

A	B	C
3	6	7
2	5	7
7	2	3
4	4	3
3	4	5



2. 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

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



差运算

R

A	B	C
3	6	7
2	5	7
7	2	3
4	4	3

S

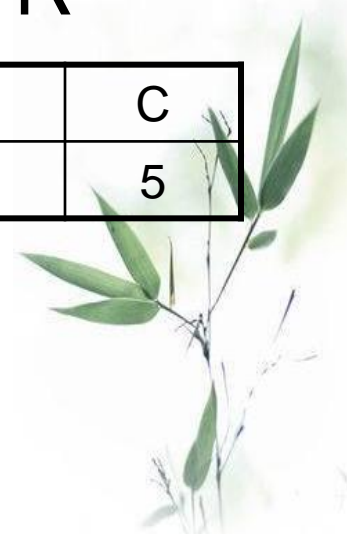
A	B	C
3	4	5
7	2	3

R—S

A	B	C
3	6	7
2	5	7
4	4	3

S—R

A	B	C
3	4	5

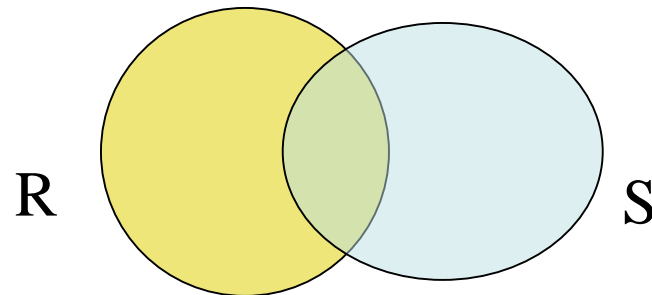


3. 交 (Intersection)

- 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

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

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



交运算

R

A	B	C
3	6	7
2	5	7
7	2	3
4	4	3

S

A	B	C
3	4	5
7	2	3

$R \cap S$

A	B	C
7	2	3



表示记号

- $\widehat{t_r t_s}$

- R 为 n 目关系， S 为 m 目关系

- $t_r \in R$, $t_s \in S$, $\widehat{t_r t_s}$ 称为元组的连接。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

$(n+m)$ 元组 $\widehat{t_r t_s}$



n 元组 t_r

m 元组 t_s



4. 笛卡尔积 (Cartesian Product)

- 严格地讲应该是广义的笛卡尔积 (Extended Cartesian Product)
- R : n 目关系, k_1 个元组
- S : m 目关系, k_2 个元组
- $R \times S$
 - 列: ($n+m$) 列元组的集合
 - 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
 - $R \times S = \{ \overbrace{t_r \ t_s} \mid t_r \in R \wedge t_s \in S \}$



笛卡尔积(续)

R		
A	B	C
a_1	b_1	c_1
a_1	b_2	c_2
a_2	b_2	c_1

S		
A	B	C
a_1	b_2	c_2
a_1	b_3	c_2
a_2	b_2	c_1

$R \times S$					
$R.A$	$R.B$	$R.C$	$S.A$	$S.B$	$S.C$
a_1	b_1	c_1	a_1	b_2	c_2
a_1	b_1	c_1	a_1	b_3	c_2
a_1	b_1	c_1	a_2	b_2	c_1
a_1	b_2	c_2	a_1	b_2	c_2
a_1	b_2	c_2	a_1	b_3	c_2
a_1	b_2	c_2	a_2	b_2	c_1
a_2	b_2	c_1	a_1	b_2	c_2
a_2	b_2	c_1	a_1	b_3	c_2
a_2	b_2	c_1	a_2	b_2	c_1

Relations r , s :

A	B
-----	-----

α	1
β	2

r

C	D	E
-----	-----	-----

α	10	a
β	10	a
β	20	b
γ	10	b

s

$r \times s$:

A	B	C	D	E
-----	-----	-----	-----	-----

α	1	α	10	a
α	1	β	19	a
α	1	β	20	b
α	1	γ	10	b
β	2	α	10	a
β	2	β	10	a
β	2	β	20	b
β	2	γ	10	b



2.4 关系代数

- 概述
- 传统的集合运算
- 专门的关系运算



专门的关系运算(续)

- 选择
- 投影
- 连接
- 除



➤图为学生-课程数据库中的student关系、Course关系、SC关系

SC:

Sno	Cno	Grade
200215121	1	92
200215121	2	85
200215121	3	88
200215122	2	90
200215122	3	80

Student:

Sno	Sname	Ssex	Sage	Sdept
200215121	李勇	男	20	CS
200215122	刘晨	女	19	IS
200215123	王敏	女	18	MA
200215125	张立	男	19	IS

Course:

Cno	Cname	Cpno	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

1.选择(Selection)

- 问题：如何查询信息系(IS)的全体学生信息

选择满足条件：Sdept='IS'的元组

学号 Sno	姓名 Sname	性别 Ssex	年龄 Sage	所在系 Sdept
200215121	李勇	男	20	CS
200215122	刘晨	女	19	IS
200215123	王敏	女	18	MA
200215125	张立	男	19	IS

选择(续)

- 选择又称为限制(Restriction)

- 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:

$X_1 \theta Y_1$

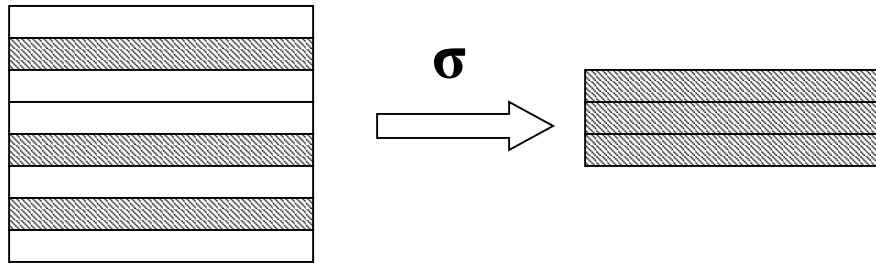
比较运算符, 如 $>$, \geq ,
 $<$, \leq , $=$, $<>$



选择(续)

- 选择运算是从关系 R 中选取使逻辑表达式 F 为真的元组

-----是从行的角度进行的运算



• [例1] 查询信息系

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

或 $\sigma_5='IS' (Student)$

选

学号 Sno	姓名 Sname	性别 Ssex	年龄 Sage	所在系 Sdept
200215121	李勇	男	20	CS
200215122	刘晨	女	19	IS
200215123	王敏	女	18	MA
200215125	张立	男	19	IS

□ 结果:

属性名可以用属性序号代替

Sno	Sname	Ssex	Sage	Sdept
95002	刘晨	女	19	IS
95004	张立	男	19	IS

• [例2] 查询年龄小

$\sigma_{\text{Sage} < 20}(\text{Student})$
或 $\sigma_{4 < 20}(\text{Student})$

学号 Sno	姓名 Sname	性别 Ssex	年龄 Sage	所在系 Sdept
200215121	李勇	男	20	CS
200215122	刘晨	女	19	IS
200215123	王敏	女	18	MA
200215125	张立	男	19	IS

□ 结果:

Sno	Sname	Ssex	Sage	Sdept
95002	刘晨	女	19	IS
95003	王敏	女	18	MA
95004	张立	男	19	IS



• [例] 查询年龄小于

$\sigma_{\text{Sage} < 20 \wedge \text{Ssex} = \text{'女'}}$

或 $\sigma_{4 < 20 \wedge 3 = \text{'女'}}$ (Student)

学号 Sno	姓名 Sname	性别 Ssex	年龄 Sage	所在系 Sdept
200215121	李勇	男	20	CS
200215122	刘晨	女	19	IS
200215123	王敏	女	18	MA
200215125	张立	男	19	IS

□ 结果:

Sno	Sname	Ssex	Sage	Sdept
95002	刘晨	女	19	IS
95003	王敏	女	18	MA

——选择是在关系R中选择满足给定条件的诸元组

2.投影 (Projection)

- **问题：** 如何查询全部学生的姓名及其所在系信息

显示学生关系中的姓名和系属性列

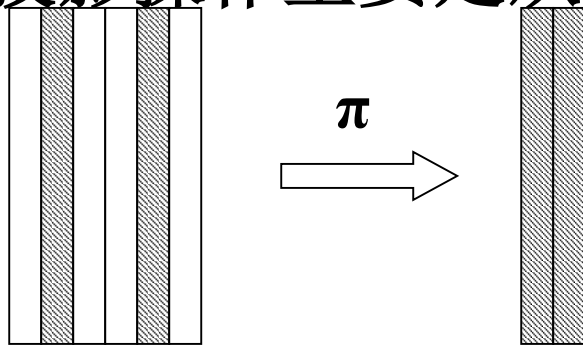
学号 Sno	姓名 Sname	性别 Ssex	年龄 Sage	所在系 Sdept
200215121	李勇	男	20	CS
200215122	刘晨	女	19	IS
200215123	王敏	女	18	MA
200215125	张立	男	19	IS

投影(续)

- **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

$$\pi_A(R) = \{ t[A] \mid t \in R \} \text{ ---其中 } A \text{ 为 } R \text{ 中的属性列}$$

- 投影操作主要是从列的角度进行运算



■ 但投影之后可能取消某些元组(避免重复行)



投影(续)

- [例3] 查询学生的姓名和所在系

即求Student关系上Sname和Sdept两个属性上的投影

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

或 $\pi_{2, 5}(\text{Student})$

属性名
可以用
属性序
号代替

□ 结果:

Sname	Sdept
李勇	CS
刘晨	IS
王敏	MA
张立	IS

学号 Sno	姓名 Sname	性别 Ssex	年龄 Sage	所在系 Sdept
200215121	李勇	男	20	CS
200215122	刘晨	女	19	IS
200215123	王敏	女	18	MA
200215125	张立	男	19	IS

• [例4] 查询学生关系

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

学号 Sno	姓名 Sname	性别 Ssex	年龄 Sage	所在系 Sdept
200215121	李勇	男	20	CS
200215122	刘晨	女	19	IS
200215123	王敏	女	18	MA
200215125	张立	男	19	IS

□ 结果:

Sdept
CS
IS
MA

投影之后取消了原关系中的某些元组
(避免重复行)



➤图为学生-课程数据库中的student关系、Course关系、SC关系

SC:

Sno	Cno	Grade
200215121	1	92
200215121	2	85
200215121	3	88
200215122	2	90
200215122	3	80

Student:

Sno	Sname	Ssex	Sage	Sdept
200215121	李勇	男	20	CS
200215122	刘晨	女	19	IS
200215123	王敏	女	18	MA
200215125	张立	男	19	IS

Course:

Cno	Cname	Cpno	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

选择投影运算复合使用示例

- 复合运用投影、选择、笛卡尔积运算，可以从任意n张表中截取满足条件的子表

例 查询选修了2号课程的学生学号。

$$\pi_{Sno} (\sigma_{Cno = '2'} (SC)) = \{ 200215121, 200215122 \}$$

例 查询选过“数据库”课程学生学号。

$$\pi_{Sno} (\sigma_{sc.cno=course.cno \wedge Cname = '数据库'} (SC \times Course))$$

解：需查询的数据要根据“选课”信息以及“课程”

信息来完成，因此，查询涉及SC和COURSE两张表

step1: 先将“选课”与“课程”表合并为一张表

SC × Course，结果为所有选过任意课程的信息

step2: 再从上一步结果中，对于任意一次选课，所选课

程存在sc.cno=course.cno，且选的课程是“数据库”即Cname=‘数据库’的元组



笛卡尔连接运算存在的问题

- 问题示例 观察上例中表达式 $SC \times Course$ ，其结果中包含所有的选课与所有课程的组合，而实际中，一个学生不可能选所有课程，这样结果元组中许多是无意义的。上例对200215121学生只有第一个元组有意义

SNO	CNO	GRADE	CNO	CNAME	CPNO	CREDIT
200215121	1	92	1	数据库	5	4
200215121	1	92	2	数学	null	2
200215121	1	92	3	信息系统	1	4
200215121	1	92	4	操作系统	6	3
200215121	1	92	5	数据结构	7	4
200215121	1	92	6	数据处理	null	2
200215121	1	92	7	C 语言	6	4
...
200215121	2	92	1	数据库	5	4

连接运算

- **问题分析** 笛卡尔乘运算为了保证数学上的**完整性**，将两张表的‘**所有**’内容合并，若该两张表之间存在关联关系，则合并后结果中将那些**无关联**关系的元组也合并了。

数学上的“完整”在使用中不便，甚至造成“信息丢失”

- **问题解决**
 - 1) 通过在 σ_P 运算中确定适当的P，去除那些无关联的元组。如 `sc.cno=course.cno`
 - 2) 定义扩展的笛卡尔乘运算，在**合并时去除**无关联元组



连接 (续)

- 连接也称为 θ 连接
- 连接运算的含义
 - 从两个关系的笛卡尔积中选取属性间满足一定条件的元组

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

- ◆ 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(续)

- 连接运算从 R 和 S 的广义笛卡尔积 $R \times S$ 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 θ 的元组

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

- 连接后，结果关系的关系模式由 R 和 S 的所有属性组成， R 的属性在前， S 的属性在后



连接(续)

- [例] 计算 SC  Course
SC. Cno < Course. Cno

关系SC

Sno	Cno	Grade
001	1	92
001	3	88
002	2	90
003	6	80

关系Course

Cno	Cname	Cpno	Ccredit
1	操作系统	6	3
3	数据结构	7	4
6	数据处理		2

----从SC × Course中选取SC关系中课程号小于Course关系中课程号值的元组



连接(续)

SC  Course
SC.Cno < Course.Cno

- 计算步骤

(1) 计算笛卡尔积 SC × Course, 结果为

Sno	SC.Cno	Grade	Course.Cno	Cname	Cpno	Ccredit
001	1	92	1	操作系统	6	3
001	1	92	3	数据结构	7	4
001	1	92	6	数据处理		2
001	3	88	1	操作系统	6	3
001	3	88	3	数据结构	7	4
001	3	88	6	数据处理		2
002	2	90	1	操作系统	6	3
002	2	90	3	数据结构	7	4
002	2	90	6	数据处理		2
003	6	80	1	操作系统	6	3
003	6	80	3	数据结构	7	4
003	6	80	6	数据处理		2

连接(续)

SC  Course
SC.Cno < Course.Cno

(2) 从SC × Course中选择满足SC.Cno < Course.Cno的元组

Sno	SC.Cno	Grade	Course.Cno	Cname	Cpno	Ccredit
001	1	92	1	操作系统	6	3
001	1	92	3	数据结构	7	4
001	1	92	6	数据处理		2
001	6	88	1	操作系统	6	3
001	3	88	3	数据结构	7	4
001	3	88	6	数据处理		2
002	2	90	1	操作系统	6	3
002	2	90	3	数据结构	7	4
002	2	90	6	数据处理		2
003	6	80	1	操作系统	6	3
003	6	80	3	数据结构	7	4
003	6	80	6	数据处理		2

连接(续)

SC  Course
SC. Cno < Course. Cno

(3) 结果为

对关系SC和Course进行连接的结果

Sno	SC. Cno	Grade	Course. Cno	Cname	Cpno	Ccredit
001	1	92	3	数据结构	7	4
001	1	92	6	数据处理		2
001	3	88	6	数据处理		2
002	2	90	3	数据结构	7	4
002	2	90	6	数据处理		2



连接(续)----例

计算 $R \bowtie_{1>1} S$

R

S

A	B	D
2	5	8
5	4	1
4	5	8
3	4	9

B	C	D
3	5	8
4	4	1
4	1	8
6	4	1

$R \bowtie_{1>1} S$

A	R. B	R. D	S. B	C	S. D
5	4	1	3	5	8
5	4	1	4	4	1
5	4	1	4	1	8
4	5	8	3	5	8

- ❖ 注意连接运算的执行情况!!
- ❖ 满足条件的元组 ----匹配的元组(保留)
- ❖ 不满足条件的元组 ----不匹配的元组(舍去)



连接(续)

- 常用的两类连接运算
 - 等值连接(equijoin)
 - 自然连接(Natural join)



连接(续)----等值连接

- 等值连接

- θ 为“=”的连接运算

- 含义：从关系 R 与 S 的广义笛卡尔积中选取 A 、 B 属性值相等的那些元组

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



等值连接(续)--例

计算 $R \bowtie_{1=1} S$

R

A	B	D
2	5	8
7	4	1
4	5	8
3	4	9

S

B	C	D
3	5	8
4	4	1
4	1	8
6	4	1

$R \bowtie_{1=1} S$

A	R. B	R. D	S. B	C	S. D
4	5	8	4	4	1
4	5	8	4	1	8
3	4	9	3	5	8

1. $R \times S$

2. $\sigma_{R.A=S.B}(R \times S)$



连接(续)----自然连接

❖ 自然连接：一种特殊的等值连接

■ 两个关系中进行比较的分量须是相同的属性组

■ 在结果中把重复的属性列去掉

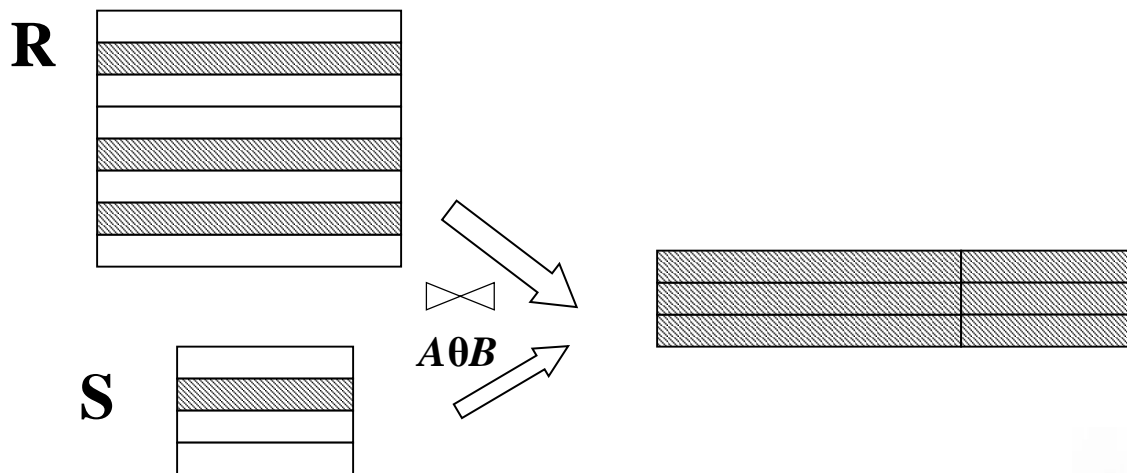
➤ 含义： R 和 S 具有相同的属性组 B

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



连接(续)

- 一般的连接操作是从行的角度进行运算



- ❖ 自然连接还需要取消重复列，所以是同时从行和列的角度进行运算。



自然连接(续)

- 自然连接运算的步骤
 - 计算笛卡尔积 $R \times S$
 - 从 $R \times S$ 中选择那些公共属性A的数值相同 ($\sigma_{R.A=S.A}$) 的元组
 - 去掉S.A(或R.A)，将留下来的R.A(或S.A)改为A，即得所要的结果



连接(续)

- 自然连接例：计算SC \bowtie Course
- 步骤：
 - (1) 计算SC \times Course

Sno	SC. Cno	Grade	Course. Cno	Cname	Cpno	Ccredit
001	1	92	1	操作系统	6	3
001	1	92	3	数据结构	7	4
001	1	92	6	数据处理		2
001	6	88	1	操作系统	6	3
001	3	88	3	数据结构	7	4
001	3	88	6	数据处理		2
002	2	90	1	操作系统	6	3
002	2	90	3	数据结构	7	4
002	2	90	6	数据处理		2
003	6	80	1	操作系统	6	3
003	6	80	3	数据结构	7	4
003	6	80	6	数据处理		2



自然连接例(续)

(2) 选择 $\sigma_{SC.Cno=Course.Cno}$ ($SC \times Course$)

Sno	SC. Cno	Grade	Course. Cno	Cname	Cpno	Ccredit
001	1	92	1	操作系统	6	3
001	3	88	3	数据结构	7	4
003	6	80	6	数据处理		2

自然连接例(续)

(3) 删除重复列SC.Cno，并将留下来的Course.Cno改为Cno，得到结果

进行SC ⋈ Course运算的结果

Sno	Grade	Cno	Cname	Cpno	Ccredit
001	92	1	操作系统	6	3
001	88	3	数据结构	7	4
003	80	6	数据处理		2



自然连接与等值连接

- 自然连接的两个关系要求有公共的属性组B，等值连接则不要求；
- 自然连接中等值的条件一定是公共属性组的值相等 ($R.B = S.B$)，而等值连接不一定；
- 自然连接中等值的条件隐含，不显式地写出来，而等值连接要写出来；
- 自然连接要在结果中去掉重复的一个属性组B，而等值连接则不。



自然连接运算----例

R和S有公共属性B, D

R

A	B	C	D
α	1	α	a
β	2	γ	a
γ	4	β	b
α	1	γ	a
δ	2	β	b

S

B	D	E
1	a	α
3	a	β
1	a	γ
2	b	δ
3	b	ϵ

结果中只有一个B, D

R ~~**S**~~

A	B	C	D	E
α	1	α	a	α
α	1	α	a	γ
α	1	γ	a	α
α	1	γ	a	γ
δ	2	β	b	δ



自然连接与笛卡尔积的区别

- 1) 两者的语义上都有合并两张表的作用

笛卡尔乘是数学意义上的所有可能组合的乘积，而自然连接则是将两张表中有关联关系的元组合并

例如 $SC \times Course$ 是所有可能的选课及所有课程信息

$SC \bowtie Course$ 则是所有的选课及所选课程的信息

- 2) 自然连接有选择 σ_p 的语义

例如 将两个表 SC 、 $Course$ 做自然连接 \bowtie ，可以理解为在 SC 中选择那些选过 “...课程” 的学生

示例 找出选过学分为4的学生号

$\sigma_{credit=4}(Course)$ // 找出学分为4的课程

$SC \bowtie \sigma_{credit=4}(Course)$ // 选出选过4分的选课

$\pi_{Sno} (SC \bowtie \sigma_{credit=4}(Course))$ // 选出选过4分的学号



假设 R_1 、 R_2 和 R_3 是三个关系

R_1

A	B	C
a ₁	b ₁	55
a ₂	b ₂	45
a ₄	b ₂	35

R_2

A	B	C
a ₁	b ₂	55
a ₂	b ₂	45
a ₃	b ₁	35
a ₄	b ₂	35

R_3

A	D
a ₁	101
a ₂	101
a ₂	102
a ₃	102

$R_1 \bowtie R_2$

$R_1.C < R_2.C$

$R_1 \bowtie R_2$

$R_1.C = R_2.C$

$R_1 \bowtie R_3$

