

北京林业大学

数据库原理与应用

元组关系演算

本节目录 CONTENTS

- | 定义
- | 元组关系演算与关系代数的关系
- | 举例说明
- | 总结

定义



定义

元组关系演算中，以元组为单位，通过谓词公式约束所要查找元组的条件，可以表示为：

$$\{t \mid \varphi(t)\}$$

其中： t 为元组变量，即查询的目， φ 称为元组演算的谓词公式，即查询的条件。

$\{t \mid \varphi(t)\}$ 表示使 $\varphi(t)$ 为真的元组 t 的集合。



$\varphi(t)$ 可以通过原子公式、约束变量、自由变量、运算符构成

原子公式分3类：

$R(t)$ ： R 为关系名，表示 t 是 R 中的元组。

$t[i]\theta u[j]$ ：表示“元组 t 的第 i 个分量与元组 u 的第 j 个分量进行比较运算 θ ”，如 $t[2] < u[3]$ 。

$t[i]\theta C$ ：表示“元组 t 的第 i 个分量与常量 C 进行比较运算 θ ”，如 $t[3] > 5$ 。



$\varphi(t)$ 约束变量与自由变量

若元组演算公式中的一个元组变量前有“全称量词”和“存在量词”，则称该变量为约束元组变量，否则称自由元组变量。

在公式 $(\exists t) \varphi(t)$ 和 $(\forall t) \varphi(t)$ 中， φ 称为是量词的辖域。 t 出现在 $(\forall t)$ 或 $(\exists t)$ 的辖域内， t 为约束元组变量，被量词所绑定。任何没有以这种方法显示绑定的变量都称为自由变量。



任意 $\varphi(t)$ 的递归定义



原子公式是公式



设 $\varphi_1(t_1)$ 和 $\varphi_2(t_2)$ 是公式, 则 $\neg \varphi_1(t_1)$, $\varphi_1(t_1) \wedge \varphi_2(t_2)$, $\varphi_1(t_1) \vee \varphi_2(t_2)$ 也是公式



设 $\varphi(t)$ 是公式, t 是 $\varphi(t)$ 中的元组变量, 则 $(\exists t) \varphi(t)$, $(\forall t) \varphi(t)$ 也是公式



有限次使用上述规则得到的式子都是公式



$\varphi(t)$ 中运算符的优先级



公式运算符

- ◆ 算术比较符: $<, >, \leq, \geq, \neq, =$
- ◆ 存在量词 \exists 和全称量词 \forall
- ◆ 逻辑运算符: $\neg, \wedge, \vee, \rightarrow$

高
↓
低

元组关系演算 与关系代数的 关系



元组关系演算与关系代数的关系

并操作

<i>R</i>		
A	B	C
a	b	c
d	a	f
c	b	d

<i>S</i>		
A	B	C
d	a	f
b	g	a

<i>R</i> ∪ <i>S</i>		
A	B	C
a	b	c
d	a	f
c	b	d
b	g	a

$$\{ t \mid R(t) \vee S(t) \}$$



元组关系演算与关系代数的关系

差操作

<i>R</i>		
A	B	C
a	b	c
d	a	f
c	b	d

<i>S</i>		
A	B	C
b	g	a
d	a	f

<i>R-S</i>		
A	B	C
a	b	c
c	b	d

$$\{ t \mid R(t) \wedge \neg S(t) \}$$



元组关系演算与关系代数的关系

选择操作

R

A	B	C
a	b	c
d	a	f
c	b	d

$\sigma_{B='b'}(R)$

A	B	C
a	b	c
c	b	d

$$\{ t \mid R(t) \wedge F \}$$

F表示B='b'



元组关系演算与关系代数的关系

投影操作

R

A	B	C
a	b	c
d	a	f
c	b	d

$\Pi_{A,C}(R)$

A	C
a	c
d	f
c	d

$$\{t^{(2)} | (\exists u) (R(u) \wedge t[1]=u[1] \wedge t[2]=u[3])\}$$



元组关系演算与关系代数的关系

笛卡儿积操作

R

A	B	C
a1	b1	C1
a1	b2	C3

S

D	E
b1	c1
b1	c1
b2	c2

RxS

A	B	C	D	E
a1	b1	c1	b1	c1
a1	b1	c1	b1	c1
a1	b1	c1	b2	c2
a1	b2	c3	b1	c1
a1	b2	c3	b1	c1
a1	b2	c3	b2	c2

$$\{t^{(5)} \mid (\exists u)(\exists v) (R(u) \wedge S(v) \wedge t[1]=u[1] \wedge t[2]=u[2] \\ \wedge t[3]=u[3] \wedge t[4]=v[1] \wedge t[5]=v[2])\}$$

举例说明



元组关系演算举例说明



例:使用教学数据库进行元组关系演算



学生关系: S (学号、姓名、性别、年龄)



课程关系: C (课程号、课程名、先修课程号)



选课关系: SC (学号、课程号、成绩)



元组关系演算举例说明

学生关系： S (学号、姓名、性别、年龄)

课程关系： C (课程号、课程名、先修课程号)

选课关系： SC (学号、课程号、成绩)

查询学生年龄大于等于20岁的学生姓名

$\{ t^{(1)} \mid (\exists u)(S(u) \wedge u[4] \geq 20 \wedge t[1]=u[2]) \}$

学号	姓名	性别	年龄
S1	王小艳	女	18
S2	李明	男	20
S3	司马南	男	18
S4	李昕	女	19
S5	成功	男	21



元组关系演算举例说明

学生关系: S (学号、姓名、性别、年龄)

课程关系: C (课程号、课程名、先修课程号)

选课关系: SC (学号、课程号、成绩)

查询选修了课程名为“操作系统”课程的所有学生的姓名

$$\{t^{(1)} \mid (\exists u)(\exists v)(\exists w) (S(u) \wedge SC(v) \wedge C(w) \wedge w[2] = \text{'操作系统'} \\ \wedge v[2] = w[1] \wedge u[1] = v[1] \wedge t[1] = u[2])\}$$

$$\{t^{(1)} \mid (\exists u)(\exists v)(\exists w) (S(u) \wedge SC(v) \wedge C(w) \wedge w[2] = \text{'操作系统'} \\ \wedge v[2] = w[1] \wedge u[1] = v[1] \wedge t[1] = u[2]) \}$$

$S(u)$	学号	姓名	性别	年龄
u	S1	王小艳	女	18
	S2	李明	男	20
	S3	司马南	男	18
	S4	李昕	女	19
	S5	成功	男	21

$SC(v)$	学号	课程号	成绩
	S1	C1	98
	S1	C4	78
	S2	C2	88
v	S1	C2	98
	S1	C3	89

用 Q 表示

$$(\exists u)(S(u) \wedge u[1] = v[1] \wedge Q) \quad (\exists v)(SC(v) \wedge v[2] = w[1] \wedge P)$$

用 P 表示

$C(w)$	课程号	课程名	先修课程号
	C1	数据库	C2
w	C2	操作系统	C1
	C3	数据结构	C1
	C4	微机原理	C3

$$(\exists w)(C(w) \wedge w[2] = \text{'操作系统'})$$



使用元组关系演算实现查询的注意事项

语句形式化过程需要注意如下问题

- ◆ 准确地从查询语句中提取谓词，即元组变量和元组分量所满足的谓词条件。
- ◆ 涉及某个关系上的全部个体或某个个体时，使用限于该关系的“限定谓词”。
- ◆ 准确确定量词和量词的辖域，当辖域中多于一个谓词时必须注意括号的使用。

总结



总结



域关系演算

本节目录 CONTENTS

- | 定义
- | 域关系演算举例
- | 域关系演算语言
- | 总结

定义



域关系演算的定义

定义 以元组中的域为单位，按照谓词公式所约束的条件查询所需的元组，表示为：

$$\{ \langle x_1, x_2, \dots, x_n \rangle \mid R(x_1, x_2, \dots, x_n) \}$$

其中 x_1, x_2, \dots, x_n 代表域变量，即元组的分量， R 代表由原子构成的公式。

R 的定义如元组关系演算，同样是反复由原子公式、自由变量、约束变量和运算符构成。

域关系演算举例



域关系演算举例



学生关系: S (学号、姓名、性别、年龄)
课程关系: C (课程号、课程名、先修课程号)
选课关系: SC (学号、课程号、成绩)

查询学生年龄大于等于20岁的学生姓名, 学号, 性别和年龄

$\{ \langle \text{姓名}, \text{学号}, \text{性别}, \text{年龄} \rangle \mid (\langle \text{姓名}, \text{学号}, \text{性别}, \text{年龄} \rangle \in S \wedge \text{年龄} \geq 20) \}$



域关系演算举例

学号	姓名	性别	年龄
S1	王小艳	女	18
S2	李明	男	20
S3	司马南	男	18
S4	李昕	女	19
S5	成功	男	21

u



域关系演算举例



学生关系: S (学号、姓名、性别、年龄)
课程关系: C (课程号、课程名、先修课程号)
选课关系: SC (学号、课程号、成绩)

查询学生年龄大于等于20岁的学生姓名

$\{ \langle \text{姓名} \rangle \mid \exists \text{学号, 性别, 年龄} (\langle \text{姓名}, \text{学号}, \text{性别}, \text{年龄} \rangle \in S \wedge \text{年龄} \geq 20) \}$



域关系演算举例

学号	姓名	性别	年龄
S1	王小艳	女	18
S2	李明	男	20
S3	司马南	男	18
S4	李昕	女	19
S5	成功	男	21

\cup

域关系演算语言



域关系演算语言

QBE (Query By Example)

它是一种高度的非过程化，基于屏幕表格的查询语言。

用户通过填写表格，并给出查询事例的方式获取结果。

给出的查询事例是域变量。



域关系演算语言

QBE使用的表格形式

关系名	属性1	...	属性n
操作命令	属性值或查询条件	...	属性值或查询条件

P. (打印或显示)
U. (修改)
I. (插入)
D. (删除)



域关系演算语言

QBE使用步骤

1.出现空白表格，用户在第一行的最左边栏中填写关系名

Student			

2.显示该关系的属性名

Student	Sno	...	Sdept

3.用户在表格上构造查询需要

Student	Sno	...	Sdept
	P.0001		计算机



域关系演算语言

通过例子学习QBE语言

QBE语言例子：全表查询和单表条件查询
显示全部学生信息

S	SNO	...	Dept
P		...	

查询年龄大于18岁的女学生姓名

S	SN	SEX	...	Age
	P.赵亦	=‘女’	...	>18

需要同时满足的条件写在一行中。



域关系演算语言

通过例子学习QBE语言

QBE语言例子：全表查询和单表条件查询

查询年龄大于18岁或者女学生的姓名

S	SN	SEX	Age
	P.赵亦	=‘女’	
	P.钱尔		>18

或关系的条件写在两行中。



域关系演算语言

通过例子学习QBE语言

QBE语言例子：全表查询和单表条件查询
查询选修C1课程的学生姓名

SC	SNO	CNO	...
	<u>S1</u>	C1	

S	SNO	SN	Dept
	<u>S1</u>	P.赵亦	

在进行多表查询的连接过程中，需要给出相同例子的域变量。



域关系演算语言

QBE语言例子

把刘伟教师转到信息系

QBE语言修改

T	TN	SEX	DEPT
U	刘伟		信息

在SC表中插入一条记录

QBE语言插入

SC	SNO	CNO	SCORE
I.	S6	C1	



域关系演算语言

QBE语言例子

删除S1同学选修的C1课程信息

SC	SNO	CNO	SCORE
D.	<u>S1</u>	C1	

QBE语言删除

总结



总结



$$\{ \langle x_1, x_2, \dots, x_n \rangle \mid R(x_1, x_2, \dots, x_n) \}$$