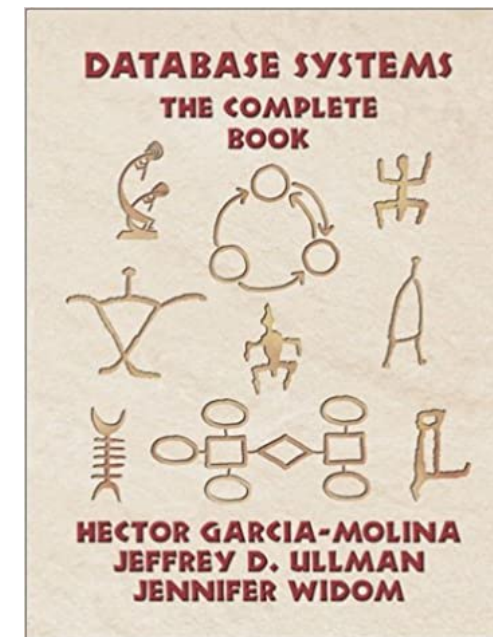
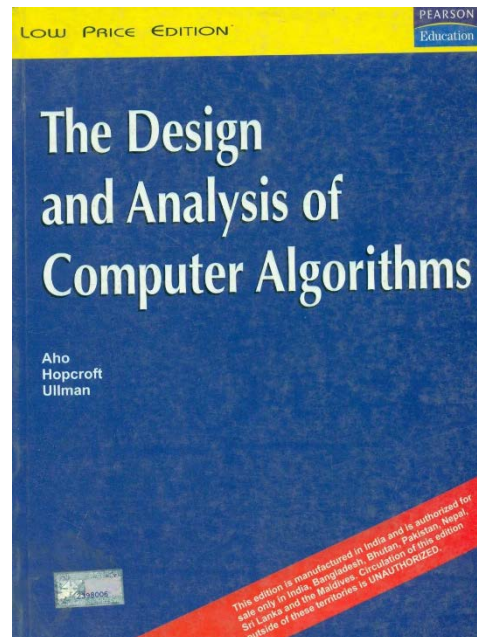




Jeffrey David Ullman Alfred Vaino Aho



第三章 关系模型

关系模型



元组关系演算

表示所有使谓词 P
为真的元组集合

P 是公式
由原子公式和运算符组成

形式化定义
 $\{t \mid P(t)\}$

t 为元组变量

如果元组变量前有全称(\forall)或存在(\exists)量词,
则称其为约束变量, 否则称为自由变量

原子公式

$$s \in R$$

s 是关系 R 中的一个元组

$$s[X] \theta u[Y]$$

$s[X]$ 与 $u[Y]$ 为元组分量，它们之间满足比较关系 θ

$$s[X] \theta c$$

分量 $s[X]$ 与常量 c 之间满足比较关系 θ

公式的递归定义

- 原子公式是公式
- 若 P 是公式，则 $\neg P$ 也是公式
- 若 P_1, P_2 是公式，则 $P_1 \wedge P_2, P_1 \vee P_2, P_1 \Rightarrow P_2$ 也是公式
- 若 $P(t)$ 是公式， R 是关系，则 $\exists t \in R(P(t)), \forall t \in R(P(t))$ 也是公式

公式的等价性

- $P_1 \wedge P_2 \Leftrightarrow \neg (\neg P_1 \vee \neg P_2)$
- $\forall t \in R(P(t)) \Leftrightarrow \neg \exists t \in R(\neg P(t))$
- $P_1 \Rightarrow P_2 \Leftrightarrow \neg P_1 \vee P_2$

p	q	$\neg p$	$p \Rightarrow q$	$\neg p \vee q$
T	T	F	T	T
T	F	F	F	F
F	T	T	T	T
F	F	T	T	T

元组关系演算示例

R

A	B	C
1	2	3
4	5	6
7	8	9

1. $\{t \mid t \in S \wedge t[A] > 2\}$

2. $\{t \mid t \in R \wedge \neg t \in S\}$

3. $\{t \mid t \in S \wedge \exists u \in R(t[C] < u[B])\}$

4. $\{t \mid t \in R \wedge \forall u \in S(t[C] > u[A])\}$

S

A	B	C
1	2	3
3	4	6
5	6	9

5. $\{t \mid \exists u \in R \exists v \in S(u[A] > v[B] \wedge t[A] = u[B] \wedge t[B] = v[C] \wedge t[C] = u[A])\}$

元组关系演算示例

A	B	C
3	4	6
5	6	9

$$\{t \mid t \in S \wedge t[A] > 2\}$$

A	B	C
4	5	6
7	8	9

$$\{t \mid t \in R \wedge \neg t \in S\}$$

A	B	C
1	2	3
3	4	6

$$\{t \mid t \in S \wedge \exists u \in R (t[C] < u[B])\}$$

A	B	C
4	5	6
7	8	9

$$\{t \mid t \in R \wedge \forall u \in S (t[C] > u[A])\}$$

$$\{t \mid \exists u \in R \exists v \in S (u[A] > v[B] \wedge t[A] = u[B] \wedge t[B] = v[C] \wedge t[C] = u[A])\}$$

R.B	S.C	R.A
5	3	4
8	3	7
8	6	7
8	9	7

元组关系演算查询示例

找出工资在800元以上的老师

$$\{ t \mid t \in PROF \wedge t[SAL] > 800 \}$$

找出工资在800元以上的老师的姓名

$$\{ t \mid \exists s \in PROF (t[PNAME] = s[PNAME] \wedge s[SAL] > 800) \}$$

给出计算机系老师的姓名

$$\{ t \mid \exists u \in DEPT \exists s \in PROF (u[DNAME] = \text{“计算机系”} \wedge s[DNO] = u[DNO] \wedge t[PNAME] = s[PNAME]) \}$$

元组关系演算查询示例

求选修了全部课程的学生号

$$\left\{ t \mid \exists u \in S \forall v \in C \left(\exists w \in SC \left(\begin{array}{l} v[CNO] = w[CNO] \wedge \\ u[SNO] = w[SNO] \wedge \\ t[SNO] = u[SNO] \end{array} \right) \right) \right\}$$

$$\{ t \mid \forall v \in C (\exists w \in SC (v[CNO] = w[CNO] \wedge t[SNO] = w[SNO])) \}$$

错误的写法

元组关系演算查询示例

求选修了S1同学所修全部课程的学生号

$$\left\{ t \mid \left(\begin{array}{l} \exists u \in S (t[SNO] = u[SNO]) \wedge \forall w \in C \\ \exists s \in SC (s[CNO] = w[CNO] \wedge s[SNO] = 'S1') \\ \Rightarrow \\ \exists v \in SC (v[CNO] = w[CNO] \wedge v[SNO] = u[SNO]) \end{array} \right) \right\}$$

元组关系演算与关系代数的等价性

- 投影: $\Pi_A(R) = \{ t \mid \exists s \in R (s[A] = t[A]) \}$
- 选择: $\sigma_{F(A)}(R) = \{ t \mid t \in R \wedge F(t[A]) \}$
- 广义笛卡儿积: $R(A) \times S(B) =$
 $\{ t \mid \exists u \in R \exists s \in S (t[A] = u[A] \wedge t[B] = s[B]) \}$
- 并: $R \cup S = \{ t \mid t \in R \vee t \in S \}$
- 交: $R - S = \{ t \mid t \in R \wedge \neg t \in S \}$

关系演算表达式的安全性

元组关系演算有可能会产生**无限关系**，这样的表达式是不安全的，如 $\{t \mid \neg(t \in R)\}$ ，求所有不在 R 中的元组

引入公式 P 的**域**概念，用 $dom(P)$ 表示
 $dom(P)$ = 显式出现在 P 中的值 + 在 P 中出现的关系的元组中出现的值（不必是最小集）
如定义 $dom(t \mid \neg(t \in R)) = \{R \text{ 中出现的所有值的集合}\}$

如果出现在表达式 $\{t \mid P(t)\}$ 结果中的所有值均来自 $dom(P)$ ，则称 $\{t \mid P(t)\}$ 是**安全的**

表达式安全域的例子

R

A	B
A1	B1
A1	B2
A2	B3

$\{t \mid \neg (t \in R)\}$

A	B
A1	B3
A2	B1
A2	B2

$$\text{dom}(\neg (t \in R)) = \{\{A1, A2\}, \{B1, B2, B3\}\}$$



域关系演算形式化定义

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

x_i 代表域变量, P 为由原子构成的公式

原子公式

- $\langle x_1, x_2, \dots, x_n \rangle \in R$: x_i 是域变量或域常量
- $x \theta y$: 域变量 x 与 y 满足比较关系 θ
- $x \theta c$: 域变量 x 与常量 c 满足比较关系 θ

域关系演算的计算例子

R

A	B	C
1	2	3
4	5	6
7	8	9

S

A	B	C
1	2	3
3	4	6
5	6	9

W

D	E
7	5
4	8

$$\{ \langle x, y, z \rangle \mid \langle x, y, z \rangle \in R \wedge x < 5 \wedge y > 3 \}$$

A	B	C
4	5	6

A	B	C
1	2	3
4	5	6
7	8	9
3	4	6

$$\{ \langle x, y, z \rangle \mid \exists u \exists v (\langle z, x, u \rangle \in R \wedge \langle y, v \rangle \in W \wedge u > v) \}$$

B	D	A
5	7	4
8	7	7
8	4	7

$$\{ \langle x, y, z \rangle \mid \langle x, y, z \rangle \in R \vee (\langle x, y, z \rangle \in S \wedge y = 44) \}$$

域关系演算查询示例

找出工资在800元以上的老师

$$\{ \langle a, b, c, d, e \rangle \mid \langle a, b, c, d, e \rangle \in PROF \wedge e > 800 \}$$

找出工资在800元以上的老师的姓名

$$\{ \langle b \rangle \mid \exists a, c, d, e (\langle a, b, c, d, e \rangle \in PROF \wedge e > 800) \}$$

给出计算机系老师的姓名

$$\{ \langle b \rangle \mid \exists l, m, n, s, a, c, d, e (\langle l, m, n, s \rangle \in DEPT \wedge \\ \langle a, b, c, d, e \rangle \in PROF \wedge m = \text{“计算机系”} \wedge l = d) \}$$

采用域关系验算的实际查询系统：QBE

域变量：_X

投影：P.A

某位同学的姓名

sno	sname	age
00801234	P.	

选修C1或C2的同学

sno	cno	grade
P._X	c1	
P._X	c2	

每门课的先修课

cno	cname	pcno
_CX	_NX	_CY
_CY	_NY	

同时选修C1和C2的同学

sno	cno	grade
P._X	c1	
P._Y	c2	

RESULT

P.	_CX	_NX	_CY	_NY
----	-----	-----	-----	-----

CONDITIONS

_X = _Y

关系代数和关系验算的比较

$R(A, B)$ 是二元关系

求 R 中的那些元组，其属性列交换后组成的元组仍在 R 中

A	B
a	b
b	a
a	c

关系代数： $\Pi_{R.A, R.B}(\sigma_{R.A=S.B \wedge R.B=S.A}(R \times \rho_S(R)))$

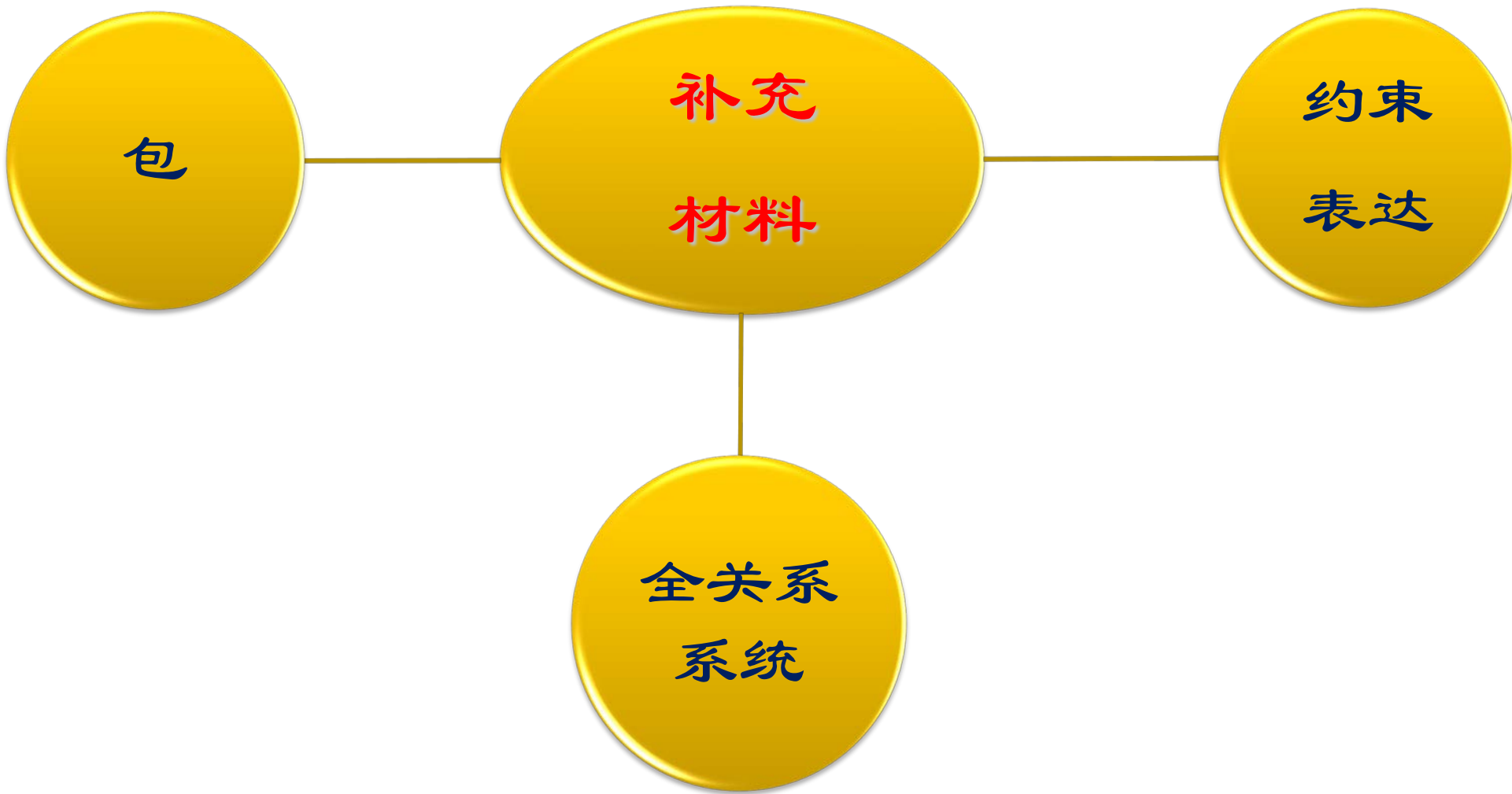
关系代数和关系验算的比较

元组关系验算: $\{t \mid t \in R \wedge \exists u \in R (t[A] = u[B] \wedge t[B] = u[A])\}$

域关系验算: $\{ \langle a, b \rangle \mid \langle a, b \rangle \in R \wedge \langle b, a \rangle \in R \}$

讨论：如何理解

“相比关系演算，关系代数是过程性操作语言”？



包 (bag)

允许重复的集合，或者多集 (multi-set)

$\{1, 2, 2, 3, 3, 3\}$

引入包的原因

- 效率，不用去重
- 合理地计算聚集函数的值

A	B
1	2
3	4
1	2
1	2

$$avg(set(B)) = 3$$

$$avg(bag(B)) = 2.5$$

包的关系代数运算

包的选择、投影、笛卡尔积：和集合上定义是一样的，只需保留重复即可

包的并、交、差：假定一个元组 t 在包 R 中重复出现了 n 次，在包 S 中重复出现了 m 次，则：

- 在 $R \cup S$ 中， t 重复出现 $n + m$ 次
- 在 $R - S$ 中， t 重复出现 $n - m$ 次
- 在 $R \cap S$ 中， t 重复出现 $\min(n, m)$ 次

包的关系代数运算

哪些对包成立，哪些不成立？

- $(R \cup S) \cup T = R \cup (S \cup T)$
- $(R \cap S) \cap T = R \cap (S \cap T)$
- $R \cup (S \cap T) = (R \cup S) \cap (R \cup T)$
- $\sigma_{A \wedge B}(R) = \sigma_A(R) \cap \sigma_B(R)$
- $(R \cap S) - T = R \cap (S - T)$
- $R \cap (S \cup T) = (R \cap S) \cup (R \cap T)$
- $\sigma_{A \vee B}(R) = \sigma_A(R) \cup \sigma_B(R)$



作为约束语言的关系代数

关系代数表达约束的形式

$$R = \varphi \text{ 或者 } R \subseteq S$$

参照完整性

$R(A, B), S(B, C)$, S 中的 B 是参照 R 的 B 的外码



$$\Pi_B(S) \subseteq \Pi_B(R)$$

作为约束语言的关系代数

函数依赖

$R(A, B)$, 要求在 A 上相等的在 B 上也相等

	A	B
违反约束 的情形	1	2
	1	3

表达违反约束的情形不存在

$$\sigma_{R.A=S.A \wedge R.B \neq S.B} (R \times \rho_S(R)) = \phi$$

作为约束语言的关系代数

对于 $R(A, B)$

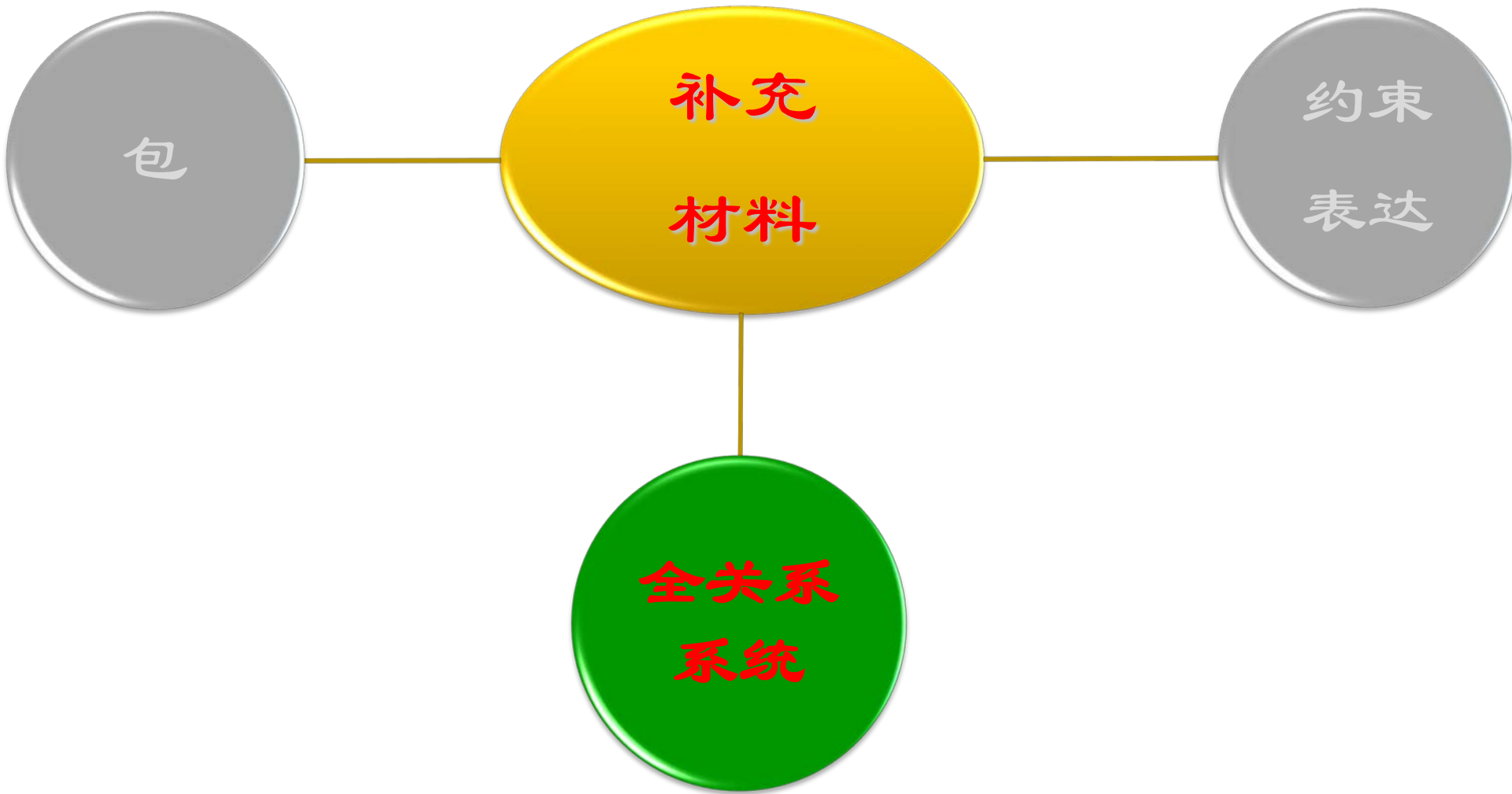
如何确定 A 的唯一性？

	A	B
不存在B取值不同而 A取值相同的两行	1	2
	1	3

对于 $R(A, B, C)$

如何确定 A 的唯一性？

	A	B	C
不存在B或C之一 取值不同而A取 值相同的两行	1	2	3
	1	2	2
	1	3	2



关系系统

表式系统

仅支持关系(即表)数据结构, 不支持集合级操作。
表式系统不能算关系系统

最小关系系统

仅支持关系数据结构和三种关系操作。许多微机关系数据库系统如FoxPro属于此类

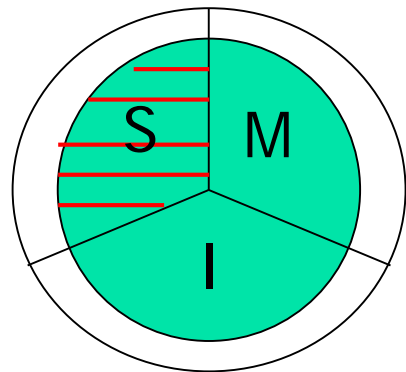
完备关系系统

支持关系数据结构和所有的关系代数操作。90年代初的许多关系数据库管理系统属于这一类

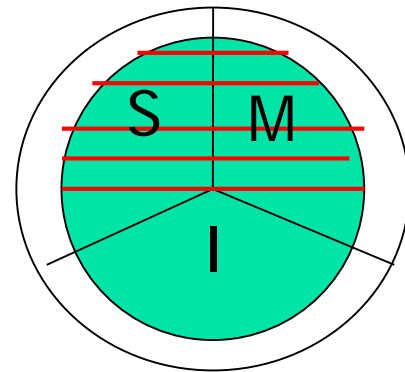
全关系系统

这类系统支持关系模型的所有特征

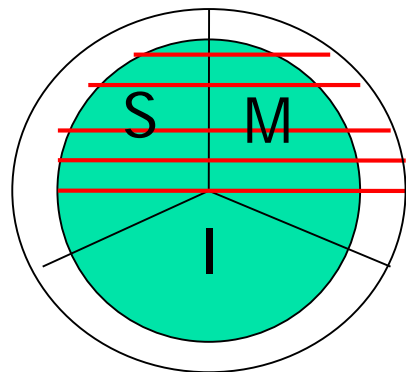
关系系统



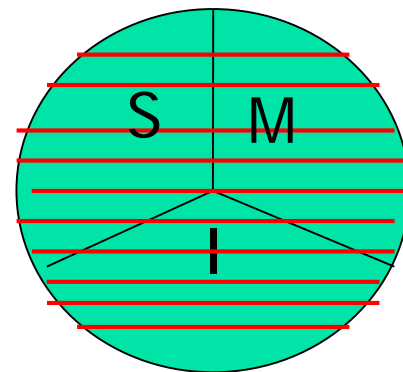
表式系统



最小关系系统



关系完备的系统



全关系系统

全关系系统的十二条准则

准则 0:

一个关系型的DBMS必须能完全通过它的关系能力来管理数据库

准则 1：信息准则

关系型DBMS的所有信息都应在逻辑一级上用一种方法即表中的值显式地表示

准则 2：保证访问准则

依靠表名、主码和列名的组合，保证能以逻辑方式访问关系数据库中的每个数据项（分量值）

以关联寻址代替面向机器的访问方式

全关系系统的十二条准则

准则 3：空值的系统化处理

全关系型DBMS应支持空值概念，
并用系统化的方式处理空值



准则 4：基于关系模型的动态的联机数据字典

数据库的描述在逻辑级上应该和普通数据采用同样的表示方式，使得授权用户可以使用查询一般数据所用的关系语言来查询数据库的描述信息

全关系系统的十二条准则

准则 5：统一的数据子语言

一个关系系统可以有几种语言和多种终端使用方式（如QBE、嵌入式SQL），但必须有一种语言，它的语句可以表示为具有严格语法规定的字符串

准则 6：视图更新准则

所有理论上可更新的视图也应该由系统更新，即对视图的更新要求，存在一个算法可以无二义地把更新要求转换为对基本表的更新序列

全关系系统的十二条准则

准则 7：高级的插入、删除和修改操作

关系系统的操作对象是单一的关系

准则 8：数据物理独立性

无论数据库的数据在存储表示和存取方法上作任何变化，应用程序和终端活动都保持逻辑上的不变性

全关系系统的十二条准则

准则 9：数据逻辑独立性

当对基本关系进行理论上信息不受损害的任何改变时，
应用程序和终端活动都保持逻辑上的不变性

准则 10：数据完整性的独立性

关系数据库的完整性约束条件必须是用数据库语言定义
并存储在数据字典中的，而不是由应用程序加以定义

全关系系统的十二条准则

准则 11：分布独立性

分布独立性是指DBMS具有这样的数据库语言，使得应用程序和终端活动在下列情况下都保持逻辑上的不变性：

- 在第一次引入分布式数据时，即如果原来的DBMS只管理非分布式的数据，而现在引入了分布式数据
- 当数据重新分布时，即如果原来DBMS能管理分布式数据，现在要改变原来的数据分布

全关系系统的十二条准则

准则 12：无破坏准则

如果一个关系系统具有一个低级(一次一记录)语言，则这个低级语言不能违背或绕过完整性准则

- 为获得完整性的独立性，需要让完整性约束条件和数据的逻辑结构相独立
- 不能旁路(bypass)或者关闭约束检查子系统

关系模型总结

笛卡尔积：关联世界一切存在的天罗

关系模型：降身于普罗大众的oracle

关系代数：数据操作的奥斯卡最佳范型奖

遥远天际：关系，关系，关系



最伟大的十个方程

1. 麦克斯韦方程组

2. $e^{i\pi} + 1 = 0$

3. $F = ma$

4. $a^2 + b^2 = c^2$

5. $E = MC^2$

6. 薛定谔方程

7. $1 + 1 = 2$

8. 德布罗意方程

9. 傅里叶变换

10. $C = 2\pi r$