数据库原理及应用

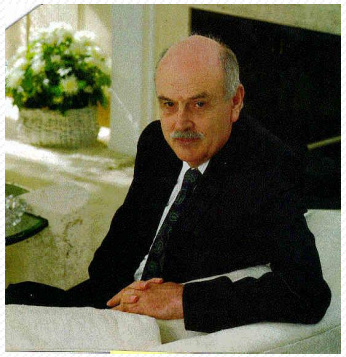
Principle and Application of Database



第二章

关系数据库

关系数据库简介



- 1970年　E.F.Codd
  - 关系数据模型

“A Relational Model of Data for Large Shared Data Banks”,《Communication of the ACM》,1970

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

第二章 关系数据库

2.1 关系数据结构

2.2 关系代数

2.3 关系的完整性

2.4 小结

2.1 关系数据库

-----关系数据结构

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

2.1.1 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（用D表示）。例:
  - 整数
  - 实数
  - 介于某个取值范围的整数
  - 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，…，D_n，允许其中某些域是相同的。

  D_1，D_2，…，D_n的笛卡尔积为：

  $D_1 \times D_2 \times \ldots \times D_n =$

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

  - 所有域的所有取值的一个组合
  - 不能重复

笛卡尔积（续）

- 元组（Tuple）
  - 笛卡尔积中每一个元素（d1，d2，…，dn）叫作一个n元组（n-tuple）或简称元组
- 分量（Component）
  - 笛卡尔积元素（d_1，d_2，…，d_n）中的每一个值d_i叫作一个分量

笛卡尔积（续）

- 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$D_1 \times D_2 \times \ldots \times D_n$的基数$M$为：

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

- 笛卡尔积的表示方法
  - 笛卡尔积可表示为一张二维表
  - 表中的每行对应一个元组，表中的每列对应一个域

笛卡尔积（续）

例如，给出3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}
- D1，D2，D3的笛卡尔积为

笛卡尔积（续）

- D1×D2×D3＝｛

  (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)｝

- 基数为2×2×3＝12

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

3. 关系（Relation）

（1） 关系

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

$R（D_1，D_2，\ldots，D_n）$

- R：关系名
- n：关系的目或度（Degree）

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）单元关系与二元关系

当*n*=1时，称该关系为单元关系（Unary relation）

或一元关系

当*n*=2时，称该关系为二元关系（Binary relation）

关系（续）

（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

（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）
- *n*目关系必有*n*个属性

关系（续）

- *D*1，*D*2，…，*D*n的笛卡尔积的某个子集才有实际含义

例：表2.1 的笛卡尔积没有实际意义

取出有实际意义的元组来构造关系

表2.2 SAP关系

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

关系：SAP(SUPERVISOR，SPECIALITY，POSTGRADUATE)

假设：导师与专业：n:1，　导师与研究生：1:n

主码：POSTGRADUATE（假设研究生不会重名）

关系（续）

（6）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

关系（续）

（7）基本关系的性质

① 列是同质的（Homogeneous）

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

基本关系的性质（续）

⑥ 分量必须取原子值

这是规范条件中最基本的一条

表2.3 非规范化关系

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

小表

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
  - 完整性约束条件

2. 定义关系模式

关系模式可以形式化地表示为：

R（U，D，DOM，F）

R	关系名
U	组成该关系的属性名集合
D	*U*中属性所来自的域
DOM	属性向域的映象集合
F	属性间数据的依赖关系的集合

定义关系模式 （续）

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 （续）

关系模式通常可以简记为

$R(U)$ 或 $R(A_1, A_2, \ldots, A_n)$

- R: 关系名
- $A_1, A_2, \ldots, A_n$: 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笼统称为关系

  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
  - 关系数据库的型: 关系数据库模式，是对关系数据库的描述
  - 关系数据库的值: 关系模式在某一时刻对应的关系的集合，通常称为关系数据库

THANK YOU FOR YOUR WATCHING

