

## 1.2 数据模型

---

1.2.1 两类数据模型

1.2.2 概念模型

1.2.3 数据模型的组成要素

1.2.4 常用的数据模型

1.2.5 层次模型

1.2.6 网状模型

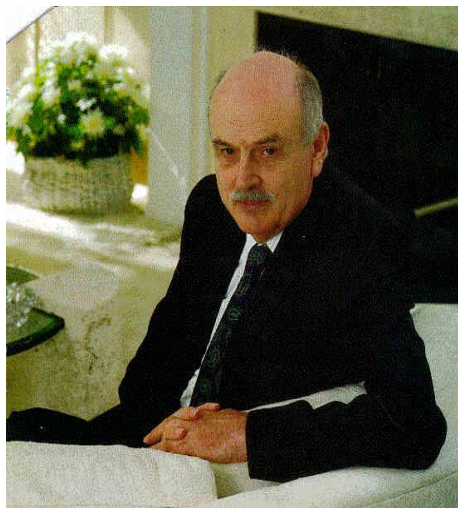
1.2.7 关系模型

## 1.2.7 关系模型

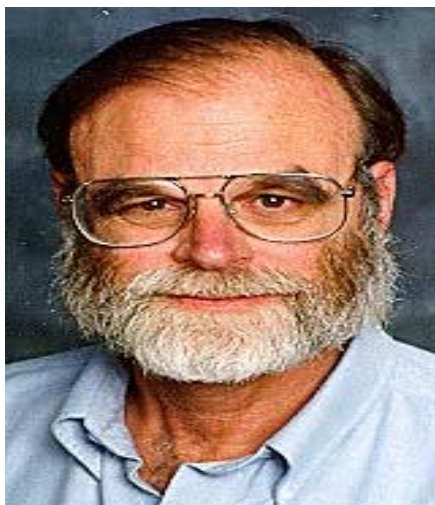
关系数据库系统采用关系模型作为数据的组织方式

1970年美国IBM公司San Jose研究室的研究员E.F.Codd首次提出了数据库系统的关系模型

计算机厂商新推出的数据库管理系统几乎都支持关系模型



Edgar F. Codd  
1922-2003



Jim Gray  
1944-2007

1970年Codd博士提出了关系模型，奠定了关系数据库的理论基础，1981年获得图灵奖

Jim Gray由于事务处理研究方面的原创性贡献以及在将研究原型转化为商业产品的系统实现方面的技术领袖地位，1998年获图灵奖(时任微软研究员)

## 1.2.7 关系模型

---

- 1) 关系模型的数据结构
- 2) 关系模型的数据操纵与完整性约束
- 3) 关系模型的优缺点

# 1) 关系数据模型的数据结构

在用户观点下，关系模型中数据的逻辑结构是一张二维表，它由行和列组成。

学生登记表

学 号	姓 名	年 龄	性 别	系 名	年 级
95004	王小明	19	女	社会学	95
95006	黄大鹏	20	男	商品学	95
95008	张文斌	18	女	法律学	95
...	...	...	...	...	...

# 1) 关系数据模型的数据结构（续）

- 关系（Relation）

一个关系对应通常说的一张表。

- 元组（Tuple）

表中的一行即为一个元组。

- 属性（Attribute）

表中的一列即为一个属性，  
给每一个属性起一个名称即  
属性名。

学生登记表

学 号	姓 名	年 龄	性 别	系 名	年 级
95004	王小明	19	女	社会学	95
95006	黄大鹏	20	男	商品学	95
95008	张文斌	18	女	法律学	95
...	...	...	...	...	...

属性

元组

# 1) 关系数据模型的数据结构（续）

- 码（Key）

表中的某个属性组，它可以唯一确定一个元组。

- 域（Domain）

属性的取值范围。

- 分量

元组中的一个属性值。

- 关系模式

对关系的描述

关系名（属性1，属性2，...，属性n）

学生登记表

域：  
18-20

学 号	姓 名	年 龄	性 别	系 名	年 级
95004	王小明	19	女	社会学	95
95006	黄大鹏	20	男	商品学	95
95008	张文斌	18	女	法律学	95
...	...	...	...	...	...

码

分量

# 1) 关系数据模型的数据结构（续）

在关系模型中，实体和联系都是用关系来表示的。

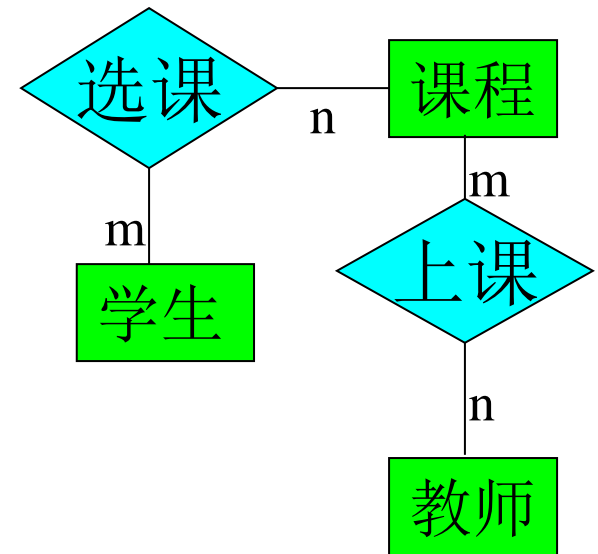
（关系数据库的结构统统是“表”）

- 实体
- 联系

案例：

实体： 学生、教师、课程

联系： 学生选课、教师上课





# 1) 关系数据模型的数据结构（续）

在关系模型中，实体是用关系来表示的

- **实体型**：直接用关系（表）表示。
- **属性**：用属性名表示。

例如：

- 学生（学号，姓名，性别，年龄，系别）
- 课程（课程号，课程名，课时）
- 教师（教师号，姓名，性别，年龄，职称，工资，岗位津贴，系别）

# S（学生表）

SNO 学号	SN 姓名	SEX 性别	AGE 年龄	DEPT 系别
S1	赵亦	女	17	计算机
S2	钱尔	男	18	信息
S3	孙珊	女	20	信息
S4	李思	男	21	自动化
S5	周武	男	19	计算机
S6	吴丽	女	20	自动化

# T（教师表）

TNO 教师号	TN 姓名	SEX 性别	AGE 年龄	PROF 职称	SAL 工资	COMM 岗位津贴	DEPT 系别
T1	李力	男	47	教授	1500	3000	计算机
T2	王平	女	28	讲师	800	1200	信息
T3	刘伟	男	30	讲师	900	1200	计算机
T4	张雪	女	51	教授	1600	3000	自动化
T5	张兰	女	39	副教授	1300	2000	信息

# C（课程表）

CNO 课程号	CN 课程名	CT 课时
C1	程序设计	60
C2	微机原理	80
C3	数字逻辑	60
C4	数据结构	80
C5	数据库	60
C6	编译原理	60
C7	操作系统	60

# 1) 关系数据模型的数据结构（续）

实体间的**联系**也是用**关系**来表示的

- 选课表（学号，课程号，成绩）
- 任课表（教师号，课程号）



联系所涉及实体的码

The diagram illustrates the concept of primary keys in a relationship. It shows two tables: '选课表' (Course Selection Table) and '任课表' (Teaching Table). In '选课表', '学号' (Student ID) and '课程号' (Course ID) are circled with dashed red lines. In '任课表', '教师号' (Teacher ID) and '课程号' (Course ID) are also circled with dashed red lines. Red arrows point from these four circled attributes to a cyan box labeled '联系所涉及实体的码' (Primary Key of the Relationship Involving Entities), indicating that these attributes serve as the primary keys for the relationships.

# 1) 关系数据模型的数据结构（续）

关系必须是规范化的，满足一定的**规范条件**

最基本的规范条件：关系的每一个分量必须是一个不可分的数据项，**不允许表中还有表**

图1.15中工资和扣除是可分的数据项，**不符合关系模型要求**

职工号	姓名	职称	工 资			扣 除		实 发
			基本工资	岗位津贴	业绩津贴	三险	个人所得税	
86051	陈平	讲师	1305	1200	1850	160	112	4083
。 。 。	。 。 。	。 。 。	。 。 。	。 。 。	。 。 。	。 。 。	。 。 。	。 。 。

图1.15 一个工资表（表中有表）实例

# 1) 关系数据模型的数据结构（续）

表1.5 术语对比

关系术语	一般表格的术语
关系名	表名
关系模式	表头（表格的描述）
关系	（一张）二维表
元组	记录或行
属性	列
属性名	列名
属性值	列值
分量	一条记录中的一个列值
非规范关系	表中有表（大表中嵌有小表）

## 2) 关系模型的操纵与完整性约束

数据操作是集合操作，操作对象和操作结果都是关系（若干元组的集合）

- 查询
- 插入
- 删除
- 更新

存取路径对用户隐蔽，用户只要指出“干什么”，不必详细说明“怎么干”



## 2) 关系模型的操纵与完整性约束（续）

### 关系的完整性约束条件

- 实体完整性
- 参照完整性
- 用户定义的完整性

### 3) 关系模型的优缺点

#### 优点

- 建立在严格的数学概念的基础上
- 概念单一
  - ✓ 实体和各类联系都用关系来表示
  - ✓ 对数据的检索结果也是关系
- 关系模型的存取路径对用户透明（隐蔽）
  - ✓ 具有更高的数据独立性，更好的安全保密性
  - ✓ 简化了程序员的工作和数据库开发建立的工作

### 3) 关系模型的优缺点（续）

#### 缺点

- 存取路径对用户透明，查询效率往往不如格式化数据模型
- 为提高性能，必须对用户的查询请求进行优化，增加了开发数据库管理系统的难度

# 1970-1975

## 层次、网状模型 VS 关系模型

Google books

relational debate hierarchical network database

**Databases demystified** 作者: Andrew J. Oppel



★★★★★

2 条评论

写评论

关于此书



目录 ▾

第 17 页 ◀ ▶

issued a research paper titled "A Relational Model of Data for Large Shared Data Banks" in *Communications of the ACM, the Journal of the Association for Computing Machinery, Inc.* The publication can be easily found on the Internet. In 1971, the CODASYL DBTG published their standards, which were over three years in the making. This began five years of heated debate over which model was the best.

The CODASYL DBTG advocates argued the following:

- The relational model was too mathematical.
- An efficient implementation of the relational model could not be built.
- Application systems need to process data one record at a time.

The relational model advocates argued the following:

- Nothing as complicated as the DBTG proposal could possibly be the correct way to manage data.
- Set-oriented queries were too difficult in the DBTG language.
- The network model had no formal underpinnings in mathematical theory.

The debate came to a head at the 1975 ACM SIGMOD (Special Interest Group on Management of Data) conference. Ted Codd and two others debated against Charles

相关图书

# 第一章 绪论

---

1.1 数据库系统概述

1.2 数据模型

**1.3 数据库系统结构**

1.4 数据库系统的组成

1.5 小结

# 1.3 数据库系统的结构

从数据库应用开发人员角度看，数据库系统通常采用三级模式结构，是数据库系统内部的系统结构

从数据库最终用户角度看，数据库系统的结构分为：

- 单用户结构
- 主从式结构
- 分布式结构
- 客户-服务器
- 浏览器-应用服务器 / 数据库服务器多层结构等

## 1.3 数据库系统的结构

---

### 1.3.1 数据库系统模式的概念

### 1.3.2 数据库系统的三级模式结构

### 1.3.3 数据库的二级映像功能与数据独立性

# 1.3.1 数据库系统模式的概念

“型” 和 “值” 的概念

- **型** (Type)

- ✓ 对某一类数据的结构和属性的说明

- **值** (Value)

- ✓ 是型的一个具体赋值

例如

学生记录 (记录**型**):

(学号, 姓名, 性别, 系别, 年龄, 籍贯)

一个记录**值**:

(201315130, 李明, 男, 计算机系, 19, 江苏南京市)



## 1.3.1 数据库系统模式的概念（续）

### 模式（Schema）

- 数据库逻辑结构和特征的描述
- 是型的描述，不涉及具体值
- 反映的是数据的结构及其联系
- 模式是相对稳定的

### 实例（Instance）

- 模式的一个具体值
- 反映数据库某一时刻的状态
- 同一个模式可以有多个实例
- 实例随数据库中的数据不断更新而变动

## 1.3.1 数据库系统模式的概念（续）

例如：在学生选课数据库**模式**中，包含学生记录、课程记录和学生选课记录

- 2013年的一个学生数据库**实例**，包含：
  - ✓ 2013年学校中所有学生的记录
  - ✓ 2013年学校开设的所有课程的记录
  - ✓ 2013年所有学生选课的记录
- 2012年度学生数据库模式对应的实例与 2013年度学生数据库模式对应的实例是**不同**的

## 1.3 数据库系统的结构

---

### 1.3.1 数据库系统模式的概念

### 1.3.2 数据库系统的三级模式结构

### 1.3.3 数据库的二级映像功能与数据独立性

## 1.3.2 数据库系统的三级模式结构

---

- 1) 模式 (Schema)
- 2) 外模式 (External Schema)
- 3) 内模式 (Internal Schema)

## 1.3.2 数据库系统的三级模式结构

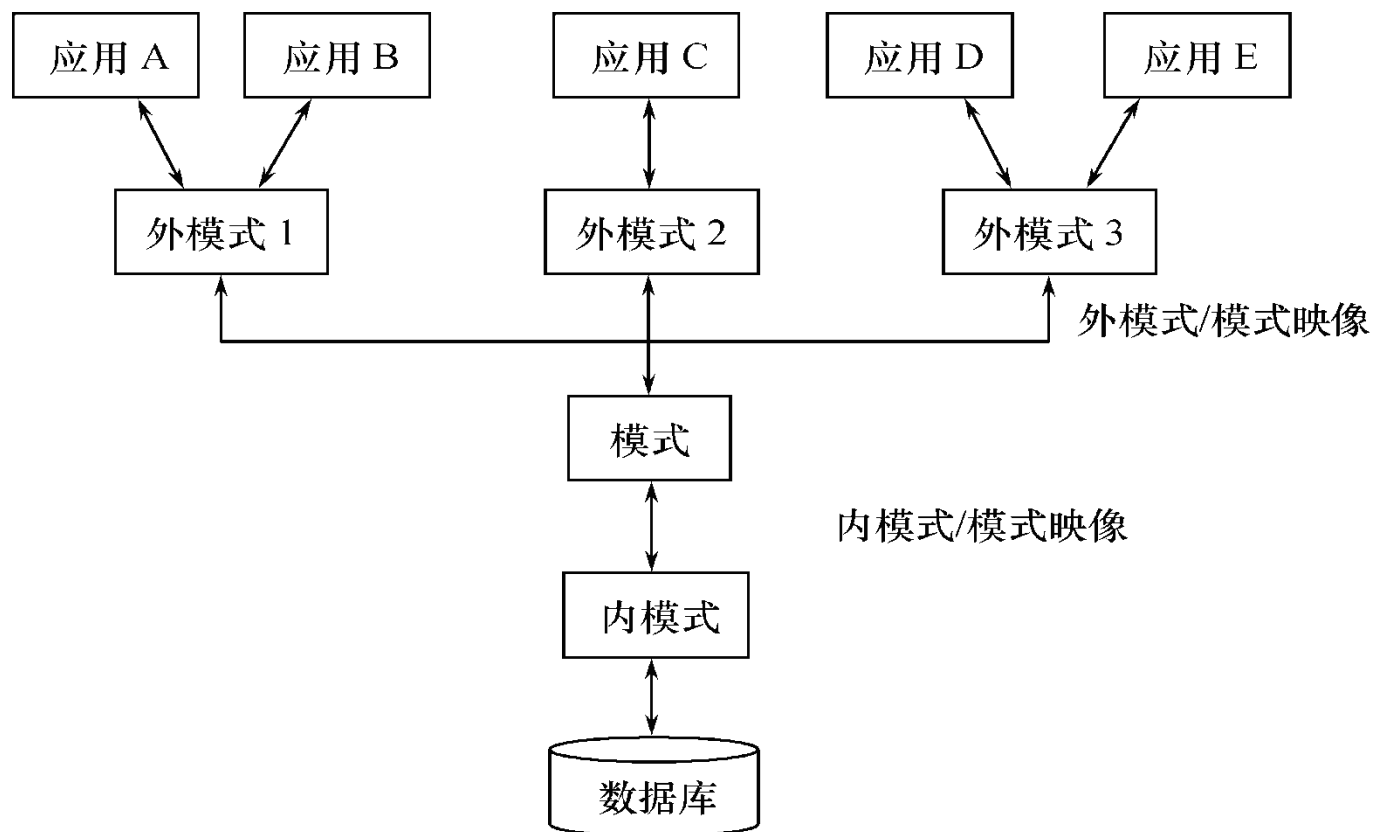


图1.16 数据库系统的三级模式结构 (P28)

# 1) 模式 (Schema)

- 模式 (也称逻辑模式)

- ✓ 数据库中全体数据的逻辑结构和特征的描述
- ✓ 所有用户的公共数据视图

- 一个数据库只有一个模式

- 模式的地位：是数据库系统模式结构的中间层

- ✓ 与数据的物理存储细节和硬件环境无关
- ✓ 与具体的应用程序、开发工具及高级程序设计语言无关

# 1) 模式（续）

## 模式需定义的内容：

- 定义数据的逻辑结构（数据项的名字、类型、取值范围等）
- 定义数据之间的联系
- 定义数据有关的安全性、完整性要求

## 2) 外模式 (External Schema)

外模式 (也称子模式或用户模式)

- 数据库用户 (包括应用程序员和最终用户) 使用的  
局部数据的逻辑结构和特征的描述
- 数据库用户的数据视图, 是与某一应用有关的数据  
的逻辑表示
  - 层次模型中的某个子树
  - 网状模型中的某个子网
  - 关系模型中的部分表、部分元组或部分属性



## 2) 外模式 (续)

### ● 外模式的地位：介于模式与应用之间

#### ➤ 模式与外模式的关系：一对多

✓ 外模式通常是模式的子集

✓ 一个数据库可以有多个外模式。反映了不同的用户的应用需求、看待数据的方式、对数据保密的要求

✓ 对模式中同一数据，在外模式中的结构、类型、长度、保密级别等都可以不同

#### ➤ 外模式与应用的关系：一对多

✓ 同一外模式也可以为某一用户的多个应用系统所使用

✓ 但一个应用程序只能使用一个外模式



## 1.3.2 数据库系统的三级模式结构

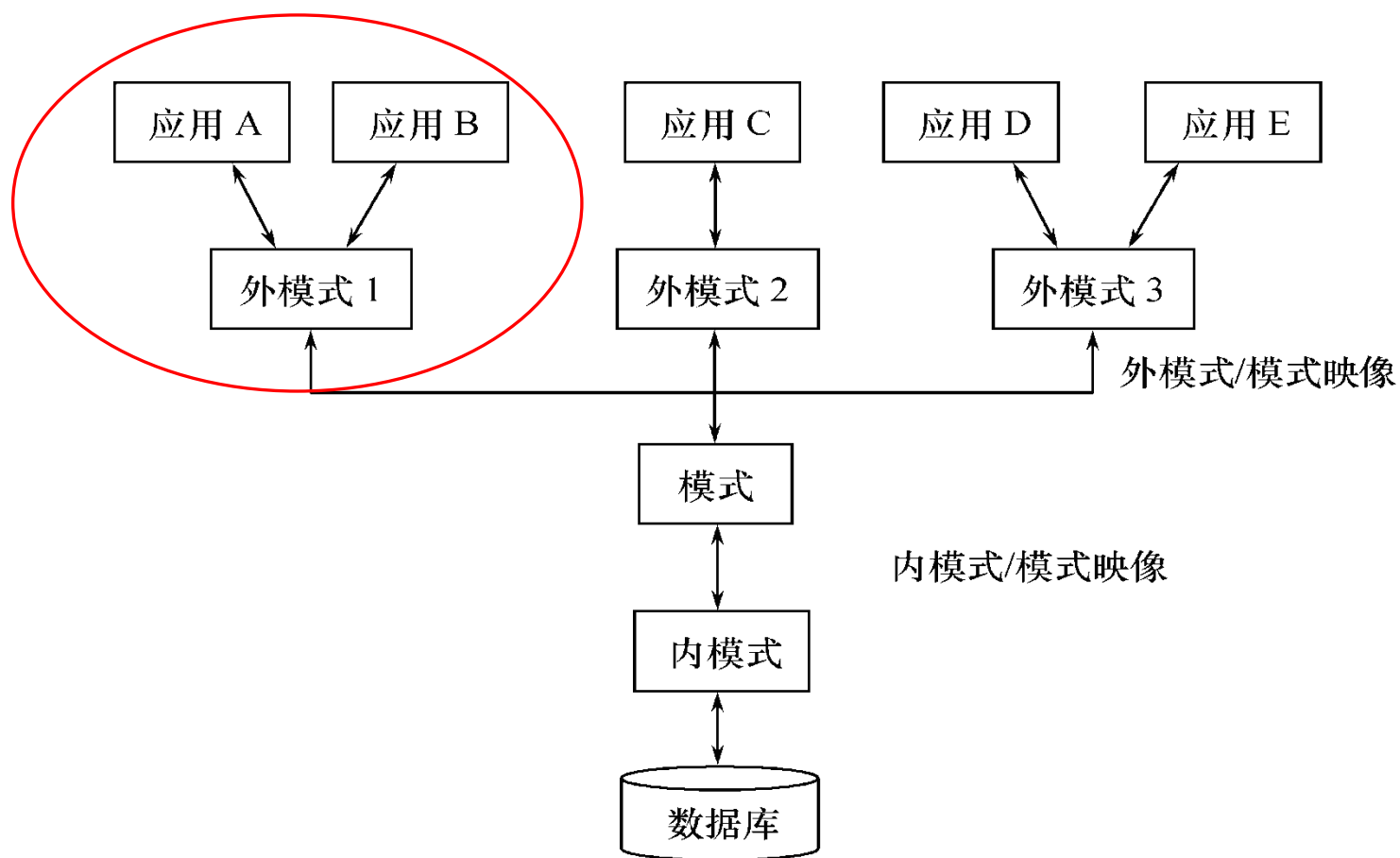


图1.16 数据库系统的三级模式结构 (P28)



## 2) 外模式（续）

### ● 外模式的用途

- 保证数据库安全性的一个有力措施
- 每个用户只能看见和访问所对应的外模式中的数据

“教学管理系统”中不同等级用户能访问的内容不同。

### 3) 内模式 (Internal Schema)

- 内模式 (也称存储模式)

- 是数据物理结构和存储方式的描述

- 是数据在数据库内部的表示方式

- ✓ 记录的存储方式 (例如, 顺序存储, 按照B树结构存储, 按hash方法存储等)

- ✓ 索引的组织方式

- ✓ 数据是否压缩存储

- ✓ 数据是否加密

- ✓ 数据存储记录结构的规定

- 一个数据库只有一个内模式

## 1.3 数据库系统的结构

---

### 1.3.1 数据库系统模式的概念

### 1.3.2 数据库系统的三级模式结构

### 1.3.3 数据库的二级映像功能与数据独立性

# 数据库的二级映像功能与数据独立性

三级模式是对数据的三个抽象级别

二级映像 in 数据库管理系统内部实现这三个抽象层次的联系和转换

1) 外模式 / 模式映像

2) 模式 / 内模式映像

# 1) 外模式 / 模式映像

- 模式：描述的是数据的全局逻辑结构
- 外模式：描述的是数据的局部逻辑结构
- 同一个模式可以有任意多个外模式
- 每一个外模式，数据库系统都有一个外模式 / 模式映像，定义外模式与模式之间的对应关系
- 映像定义通常包含在各自外模式的描述中

# 1) 外模式 / 模式映像 (续)

保证数据的逻辑独立性

- 当模式改变时，数据库管理员对外模式 / 模式映像作相应改变，使外模式保持不变
- 外模式保持不变的好处：应用程序是依据数据的外模式编写的，应用程序不必修改，保证了数据与程序的逻辑独立性，简称数据的逻辑独立性



## 2) 模式 / 内模式映像

- 模式 / 内模式映像定义了数据全局逻辑结构与存储结构之间的对应关系。
  - 例如，说明逻辑记录和字段在内部是如何表示的
- 数据库中模式 / 内模式映像是唯一的
- 该映像定义通常包含在模式描述中

## 2) 模式 / 内模式映像 (续)

### ● 保证数据的物理独立性

- 当数据库的存储结构改变了（例如选用了另一种存储结构），数据库管理员修改模式 / 内模式映像，使模式保持不变。
- 模式保持不变的好处：应用程序不受影响。保证了数据与程序的物理独立性，简称数据的物理独立性。

# 总 结

## ●数据库模式

- 即全局逻辑结构是数据库的中心与关键
- 独立于数据库的其他层次
- 设计数据库模式结构时应首先确定数据库的逻辑模式

# 总 结

## ●数据库的内模式

- 依赖于它的全局逻辑结构
- 独立于数据库的用户视图，即外模式
- 独立于具体的存储设备
- 内模式将全局逻辑结构中所定义的数据结构及其联系按照一定的物理存储策略进行组织，以达到较好的时间与空间效率

# 总 结

## ●数据库的外模式

- 面向具体的应用程序
- 定义在逻辑模式（模式）之上
- 独立于存储模式（内模式）和存储设备
- 当应用需求发生较大变化，相应外模式不能满足其视图要求时，该外模式就得做相应改动
- 设计外模式时应充分考虑到应用的扩充性

# 总 结

## ● 特定的应用程序

- 在外模式描述的数据结构上编制的
- 依赖于特定的外模式
- 与数据库的**模式**和**存储结构****独立**
- 不同的应用程序有时可以共用同一个外模式

# 总 结

## ●数据库的二级映像

- 保证了数据库外模式的稳定性
- 从底层保证了应用程序的稳定性，除非应用需求本身发生变化，否则应用程序一般不需要修改

# 总 结

- 数据与程序之间的独立性：使得数据的定义和描述可以从应用程序中分离出去
- 数据的存取由数据库管理系统管理
  - 简化了应用程序的编制
  - 大大减少了应用程序的维护和修改



# 第一章 绪论

---

1.1 数据库系统概述

1.2 数据模型

1.3 数据库系统结构

**1.4 数据库系统的组成**

1.5 小结

## 1.4 数据库系统的组成

---

- 数据库
- 数据库管理系统（及其开发工具）
- 应用程序
- 数据库管理员

从另一个角度来看  
“组成”

## 1.4 数据库系统的组成

---

**(1) 硬件平台及数据库**

**(2) 软件**

**(3) 人员**

# (1) 硬件平台及数据库

## 数据库系统对硬件资源的要求

- 足够大的内存
- 足够的大的磁盘或磁盘阵列等设备
- 较高的通道能力，提高数据传送率

## (2) 软件

- 数据库管理系统
- 支持数据库管理系统运行的操作系统
- 与数据库接口的高级语言及其编译系统
- 以数据库管理系统为核心的应用开发工具
- 为特定应用环境开发的数据库应用系统

### (3) 人员

- 数据库管理员
- 系统分析员和数据库设计人员
- 应用程序员
- 最终用户

### (3) 人员 (续)

■不同的人员涉及不同的数据抽象级别，具有不同的数据视图，如下图所示

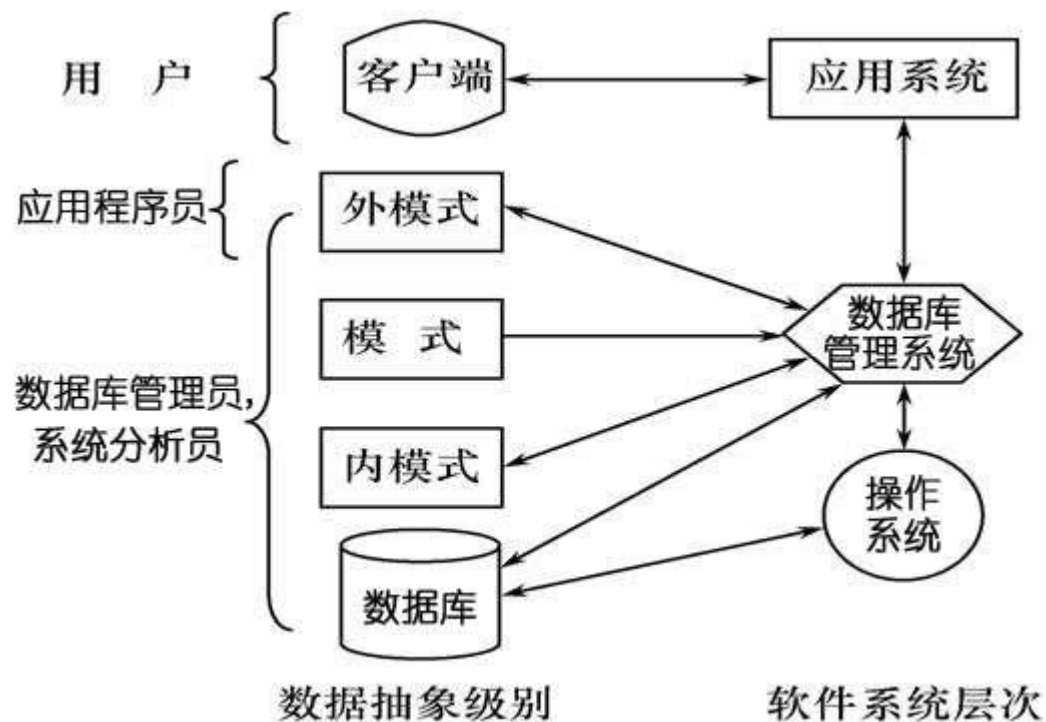


图1.17 各种人员的数据视图

# ① 数据库管理员 (DBA)

具体职责:

- 决定数据库中的信息内容和结构
- 决定数据库的存储结构和存取策略
- 定义数据的安全性要求和完整性约束条件
- 监控数据库的使用和运行
  - ✓ 周期性转储数据库
    - 数据文件
    - 日志文件
  - ✓ 系统故障恢复
  - ✓ 介质故障恢复
  - ✓ 监视审计文件



# ① 数据库管理员（DBA）（续）

## □ 数据库的改进和重组

- 性能监控和调优
- 定期对数据库进行重组，以提高系统的性能
- 需求增加和改变时，数据库须需要重构造

## ② 系统分析员和数据库设计人员

### 系统分析员

- 负责应用系统的需求分析和规范说明
- 与用户及数据库管理员结合，确定系统的硬软件配置
- 参与数据库系统的概要设计

## ② 系统分析员和数据库设计人员（续）

### 数据库设计人员

- 参加用户需求调查和系统分析
- 确定数据库中的数据
- 设计数据库各级模式

### ③ 应用程序员

- 设计和编写应用系统的程序模块
- 进行调试和安装

## ④ 用户

用户是指最终用户（End User）。最终用户通过应用系统的用户接口使用数据库。

### 1. 偶然用户

- 不经常访问数据库，但每次访问数据库时往往需要不同的数据库信息
- 企业或组织机构的高中级管理人员

## ④ 用户（续）

### 2. 简单用户

- 主要工作是查询和更新数据库
- 银行的职员、机票预定人员、旅馆总台服务员

### 3. 复杂用户

- 工程师、科学家、经济学家、科技工作者等
- 直接使用数据库语言访问数据库，甚至能够基于数据库管理系统的应用程序接口编制自己的应用程序

# 第一章 绪论

---

1.1 数据库系统概述

1.2 数据模型

1.3 数据库系统结构

1.4 数据库系统的组成

1.5 小结

## 1.5 小结

### ●数据库系统概述

- 数据库的基本概念
- 数据管理的发展过程
- 数据库系统的特点

### ●数据模型

- 概念模型， E-R 模型
- 数据模型的三要素
- 三种主要数据库模型



## 1.5 小结（续）

- 数据库系统内部的系统结构

- 数据库系统三级模式结构

- 数据库系统两层映像系统结构

- 数据库系统的组成