

# 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 数据库系统结构

## 数据库系统内部的模式结构

从**数据库管理系统**角度看

## 数据库系统外部的体系结构

从**数据库最终用户**角度看

### 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 数据库的二级映象功能与数据独立性

# 1.3.1 数据库系统模式的概念

## ∞ 模式（ Schema ）

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

## ∞ 模式的一个实例（ Instance ）

- 模式的一个具体值
- 反映数据库某一时刻的状态
- 同一个模式可以有很多实例
- 实例随数据库中的数据更新而变动

# 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

# 1.3.1 数据库系统模式的概念 (续)

## ∞ 学生选课数据库模式：

学生表： Student(Sno,Sname,Ssex,Sage,Sdept)

课程表： Course(Cno,Cname,Cpno,Ccredit)

学生选课表： SC(Sno,Cno,Grade)

## ∞ 2003 年的一个学生选课数据库实例

2003 年学校中所有学生的记录

学校开设的所有课程的记录

所有学生选课的记录

## ∞ 2003 年度学生数据库模式对应的实例与 2003 年度的实例是**不同**的

# Student 表

<input type="text"/> <input type="text"/> <b>Sno</b>	<input type="text"/> <input type="text"/> <b>Sname</b>	<input type="text"/> <input type="text"/> <b>Ssex</b>	<input type="text"/> <input type="text"/> <b>Sage</b>	<input type="text"/> <input type="text"/> <input type="text"/> <b>Sdept</b>
<b>200215121</b>	<input type="text"/> <input type="text"/>	<input type="text"/>	<b>20</b>	<b>CS</b>
<b>200215122</b>	<input type="text"/> <input type="text"/>	<input type="text"/>	<b>19</b>	<b>CS</b>
<b>200215123</b>	<input type="text"/> <input type="text"/>	<input type="text"/>	<b>18</b>	<b>MA</b>
<b>200515125</b>	<input type="text"/> <input type="text"/>	<input type="text"/>	<b>19</b>	<b>IS</b>

学生 - 课程数据库 (a)

# Course 表

课程号 Cno	课程名 Cname	先行课 Cpno	学分 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		6	4

PASCAL 语言

学生 - 课程数据库 (b)

An Introduction to Database System

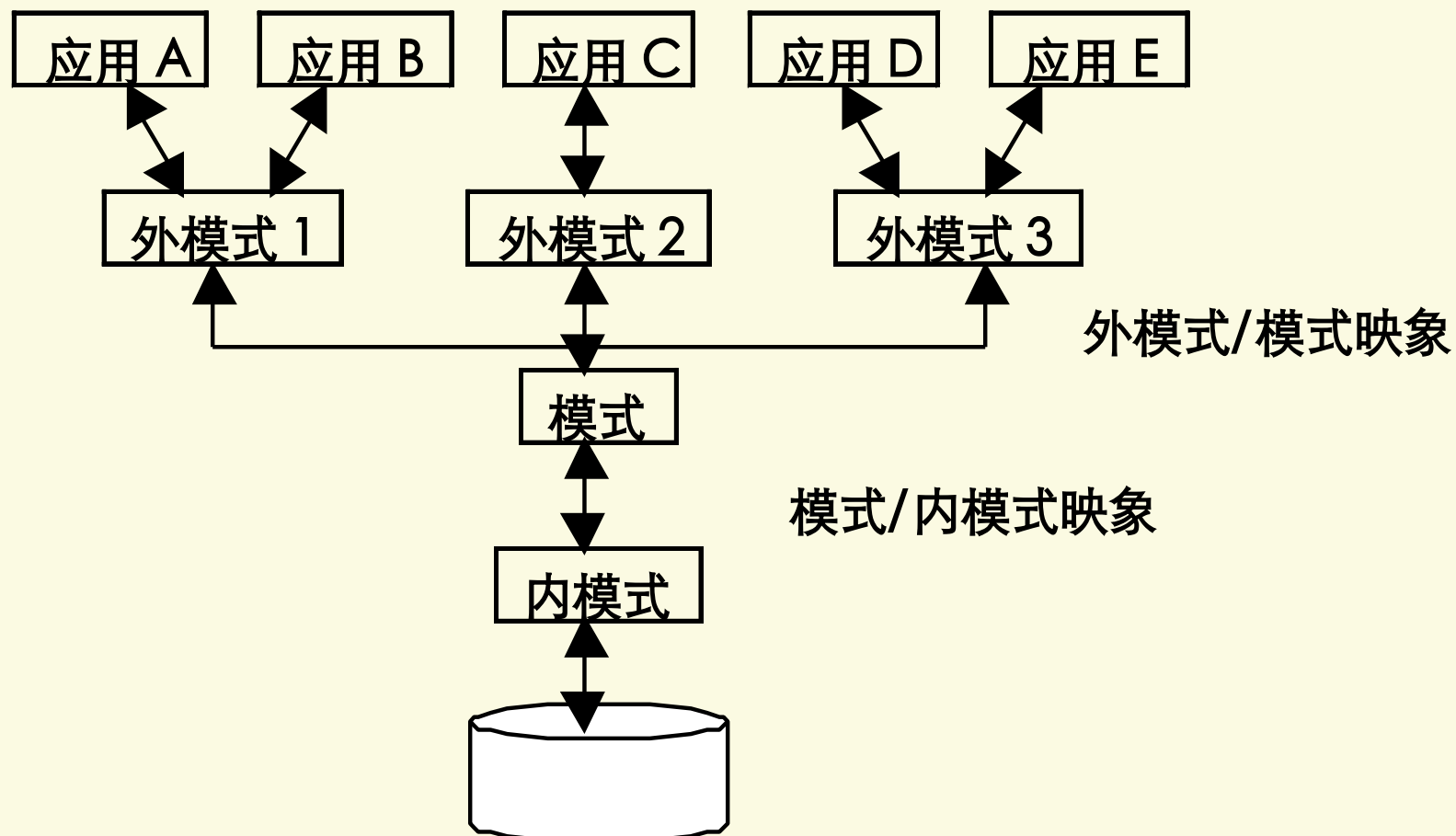
## SC 表

学 号 Sno	课程号 Cno	成绩 Grade
200215121	1	92
200215121	2	85
200215121	3	88
200215122	2	90
200215122	3	80

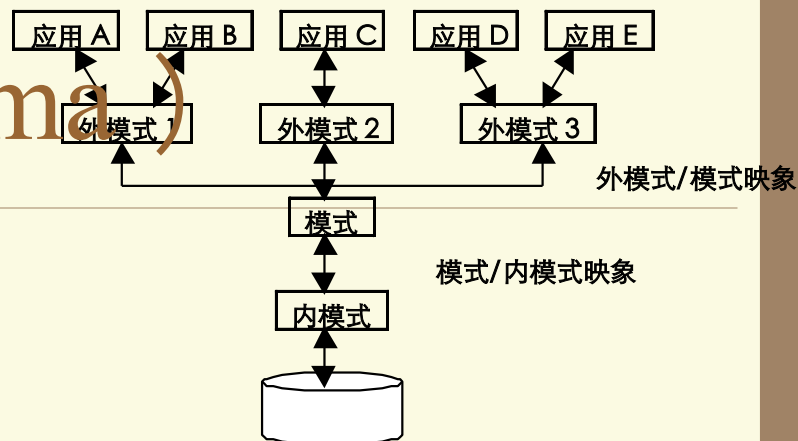
学生 - 课程数据库 (c)



## 1.3.2 数据库系统的三级模式结构



# 1. 模式 (Schema)



## 500 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，综合了所有用户的需求

## 500 一个数据库只有一个模式

## 500 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

## 500 模式的定义

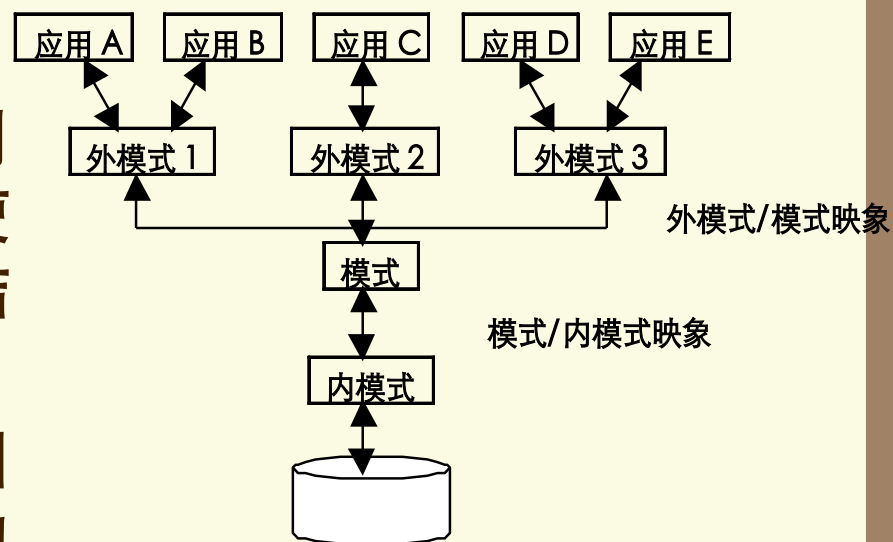
- 数据的逻辑结构 (数据项的名字、类型、取值范围等)
- 数据之间的联系

数据有关的完整性、安全性要求

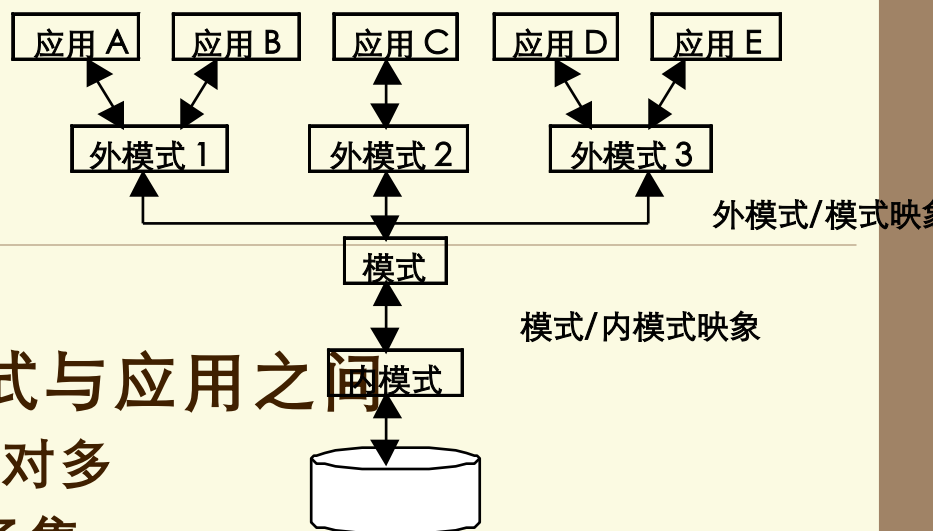
## 2. 外模式 ( External Schema )

50 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



# 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

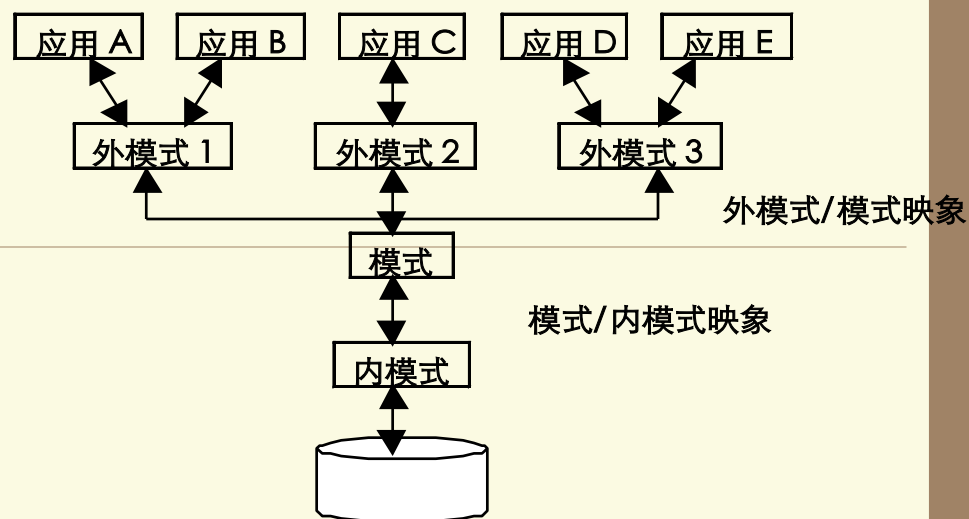
- 外模式通常是模式的子集
- 一个数据库可以有多个外模式。反映了不同用户的应用需求、看待数据的方式、对数据保密的要求
- 对模式中同一数据，在外模式中的结构、类型、长度、保密级别等都可以不同

### — 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。



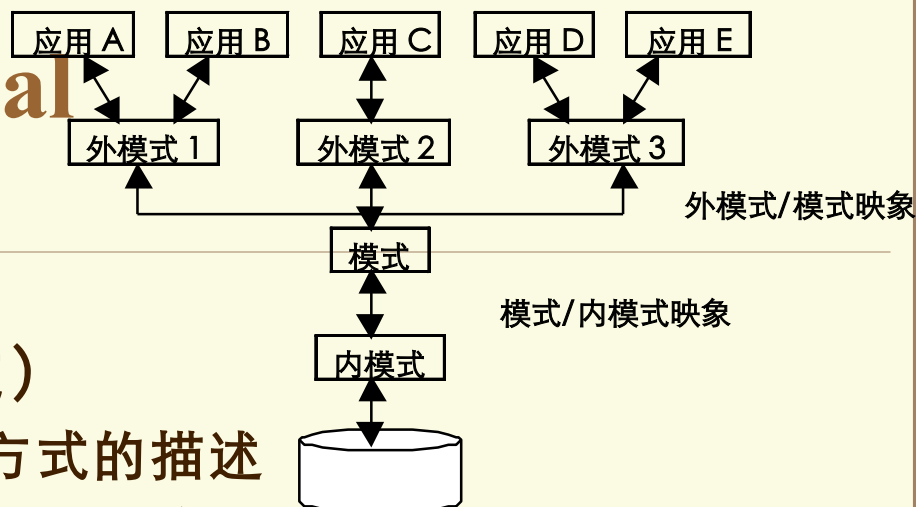
# 外模式 (续)



## 5.0 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
- 保证数据独立性的一个有力措施。

### 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（顺序存储，按照 B 树结构存储，按 hash 方法存储）
  - 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



## 1.3.3 二级映像与数据独立性

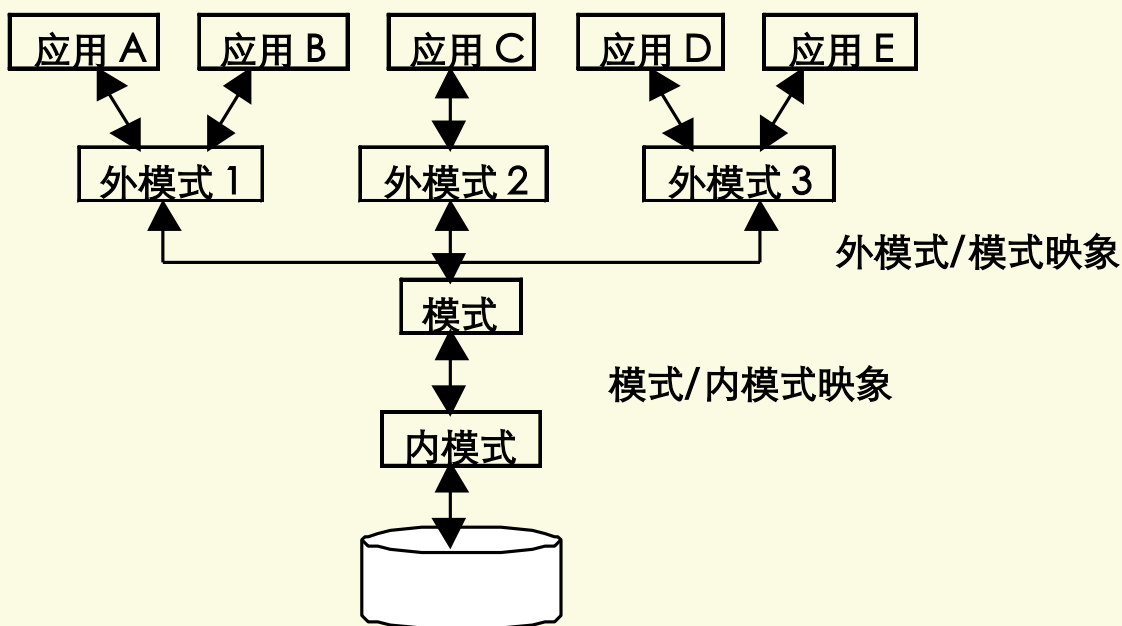
- 三级模式是对数据的三个抽象级别
- 二级映像 在 DBMS 内部实现这三个抽象层次的联系和转换



# 二级映象功能

1. 外模式 / 模式映象

2. 模式 / 内模式映象

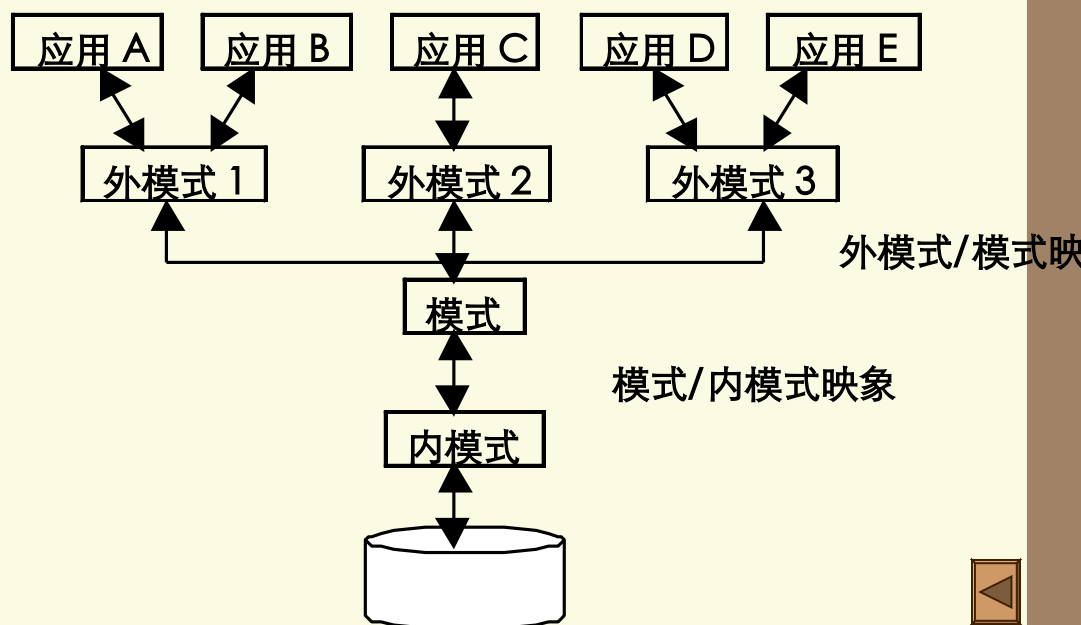




# 1. 外模式 / 模式映象

## 什么是外模式 / 模式映象

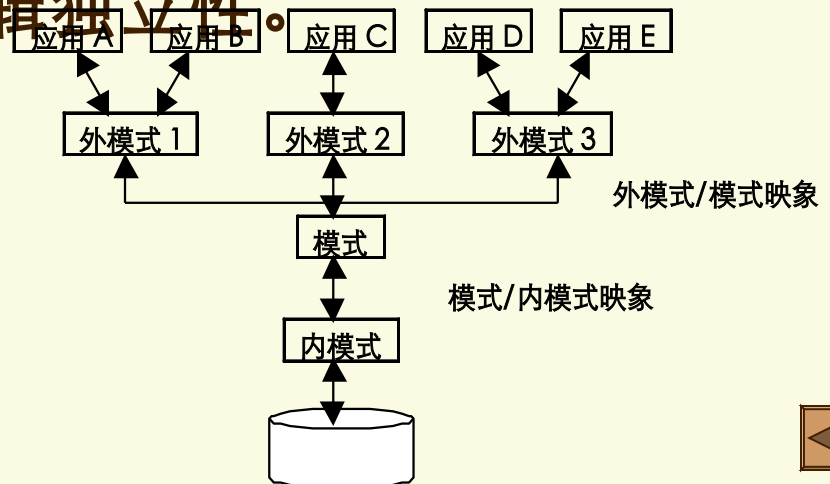
- 定义外模式与模式之间的对应关系
- 每一个外模式都对应一个外模式 / 模式映象
- 映象定义通常包含在各自外模式的描述中



# 外模式 / 模式映像 (续)

∞ 外模式 / 模式映像的用途：保证数据的逻辑独立性

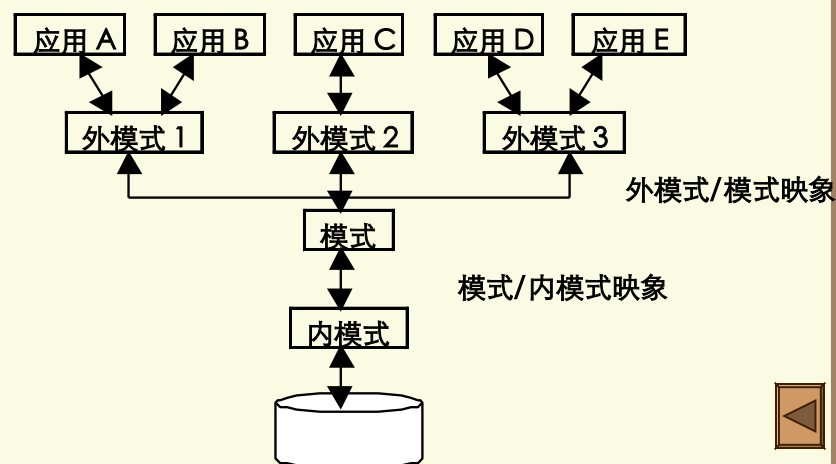
- 当模式改变时，数据库管理员修改有关的外模式 / 模式映像，使外模式保持不变
- 应用程序是依据数据的外模式编写的，从而应用程序不必修改，保证了数据与程序的逻辑独立性，简称数据的逻辑独立性。



## 2. 模式 / 内模式映象

### 什么是模式 / 内模式映象

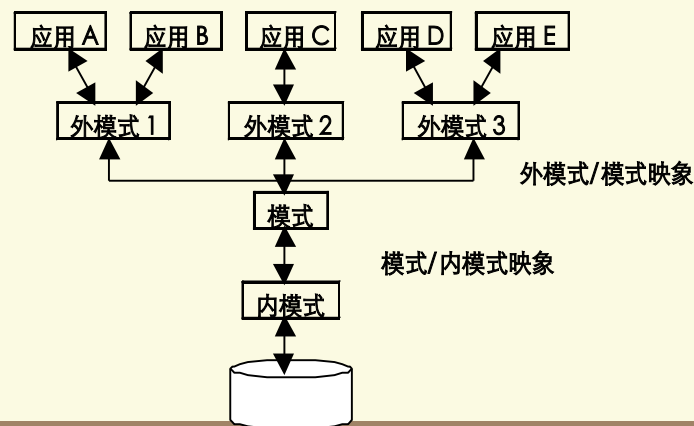
- 模式 / 内模式映象定义了数据全局逻辑结构与存储结构之间的对应关系。例如，说明逻辑记录 and 字段在内部是如何表示的
- 数据库中模式 / 内模式映象是唯一的
- 该映象定义通常包含在模式描述中



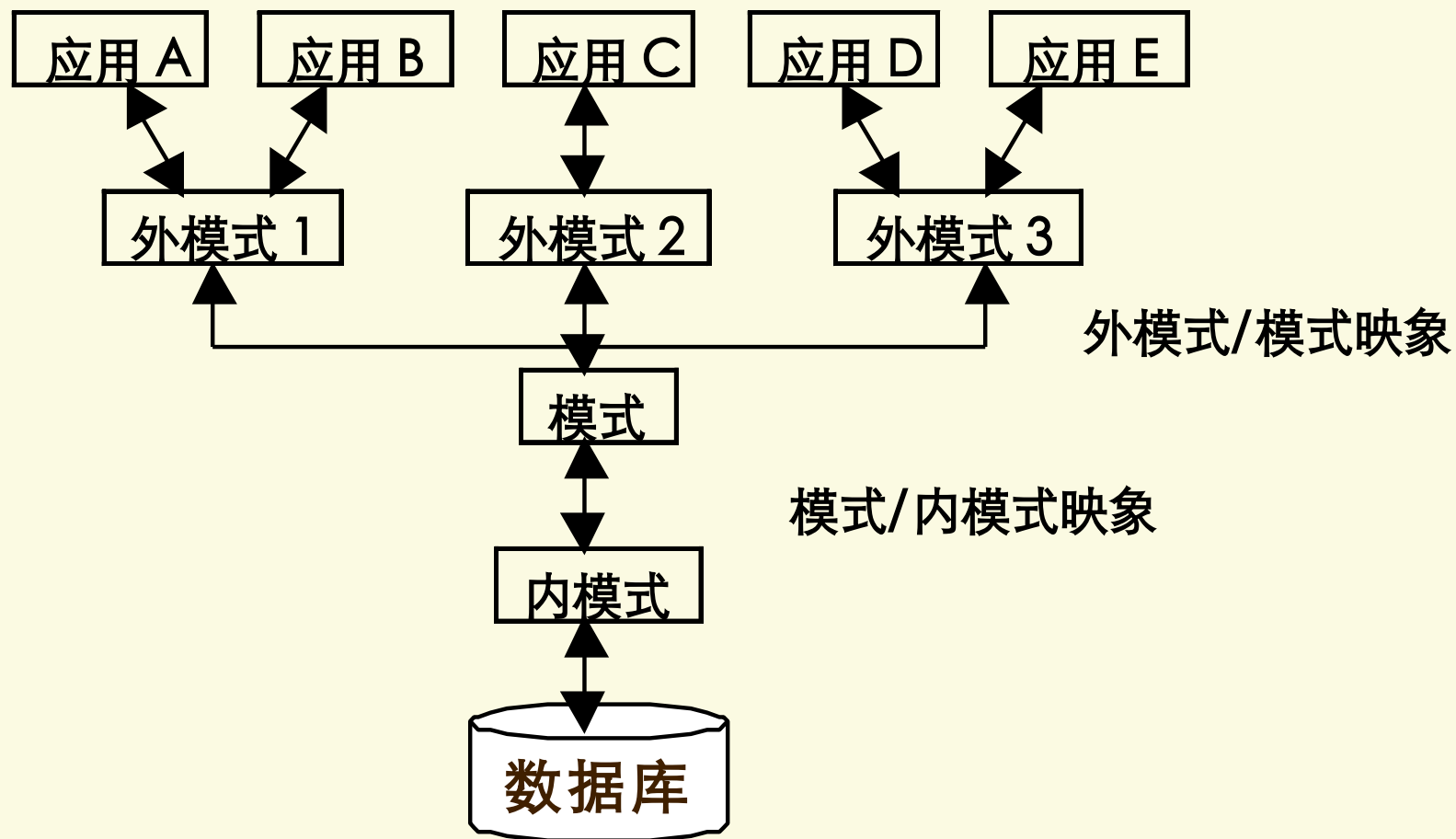
# 模式 / 内模式映像 (续)

50 模式 / 内模式映像的用途：保证数据的物理独立性

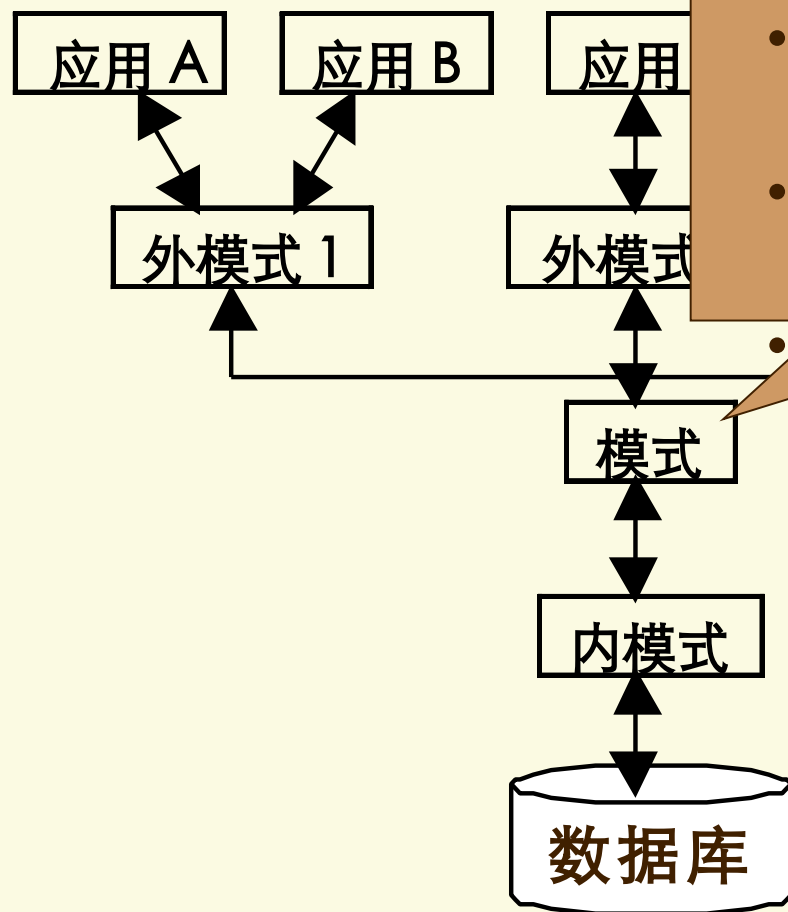
- 当数据库的存储结构改变了（例如选用了另一种存储结构），数据库管理员修改模式 / 内模式映像，使模式保持不变
- 应用程序不受影响。保证了数据与程序的物理独立性，简称数据的物理独立性。



# 小结



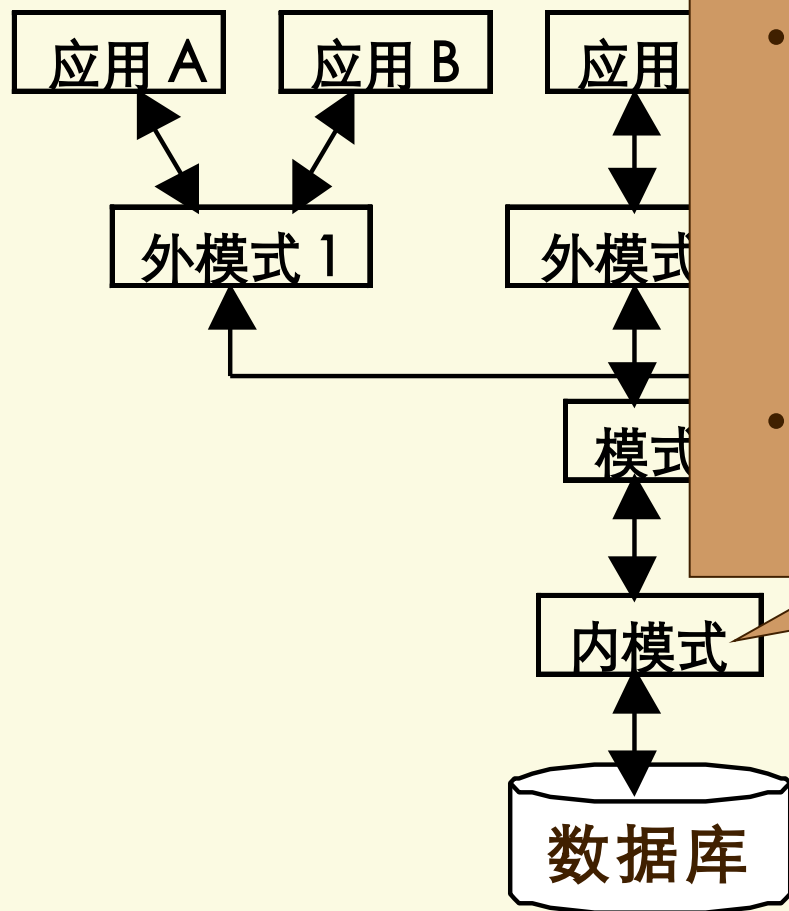
# 小结



## 模式

- 是数据库的中心与关键
- 独立于数据库的其它层次
- 设计数据库模式结构时  
应先确定数据库的逻辑模式  
模式/内模式映象

# 小结



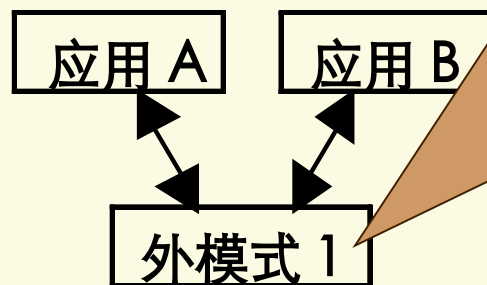
## 内模式

- 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。
- 依赖于全局逻辑结构，但独立于数据库的用户视图即外模式，也独立于具体的存储设备。

# 小结

## 外模式

- 面向具体的应用程序，定义在逻辑模式之上，但独立于存储模式和存储设备
- 设计外模式时应充分考虑到应用的扩充性。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



外模式/模式映象

模式

模式/内模式映象

内模式

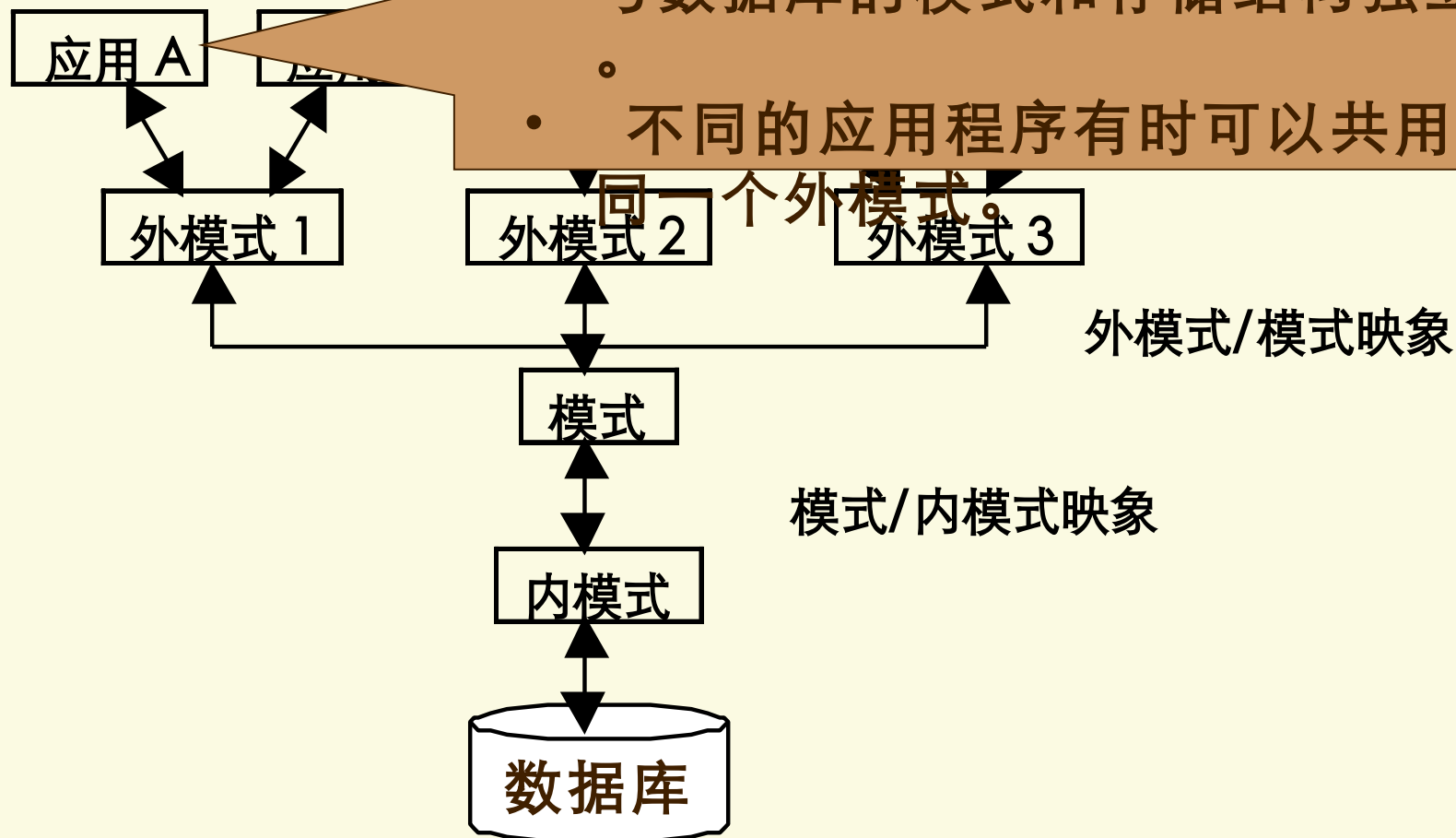




# 小结

## 应用程序

- 在外模式描述的数据结构上编制的，它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



# 小结（续）

---

## ∞ 模式

- 是数据库的中心与关键
- 独立于数据库的其它层次
- 设计数据库模式结构时应首先确定数据库的逻辑模式

# 小结（续）

---

## ∞ 内模式

- 依赖于全局逻辑结构，但独立于数据库的用户视图即外模式，也独立于具体的存储设备。
- 它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。

# 小结（续）

---

## ∞ 外模式

- 面向具体的应用程序，定义在逻辑模式之上，但独立于存储模式和存储设备
- 设计外模式时应充分考虑到应用的扩充性。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

# 小结（续）

---

## ☞ 应用程序

- 在外模式描述的数据结构上编制的，它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。

# 小结（续）

## ∞ 二级映像

- 保证了数据库外模式的稳定性，从而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。
- 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。

## ∞ 数据库特点与数据库的模式结构



