

# 高级数据库系统

2014年秋

# 第一章 引言

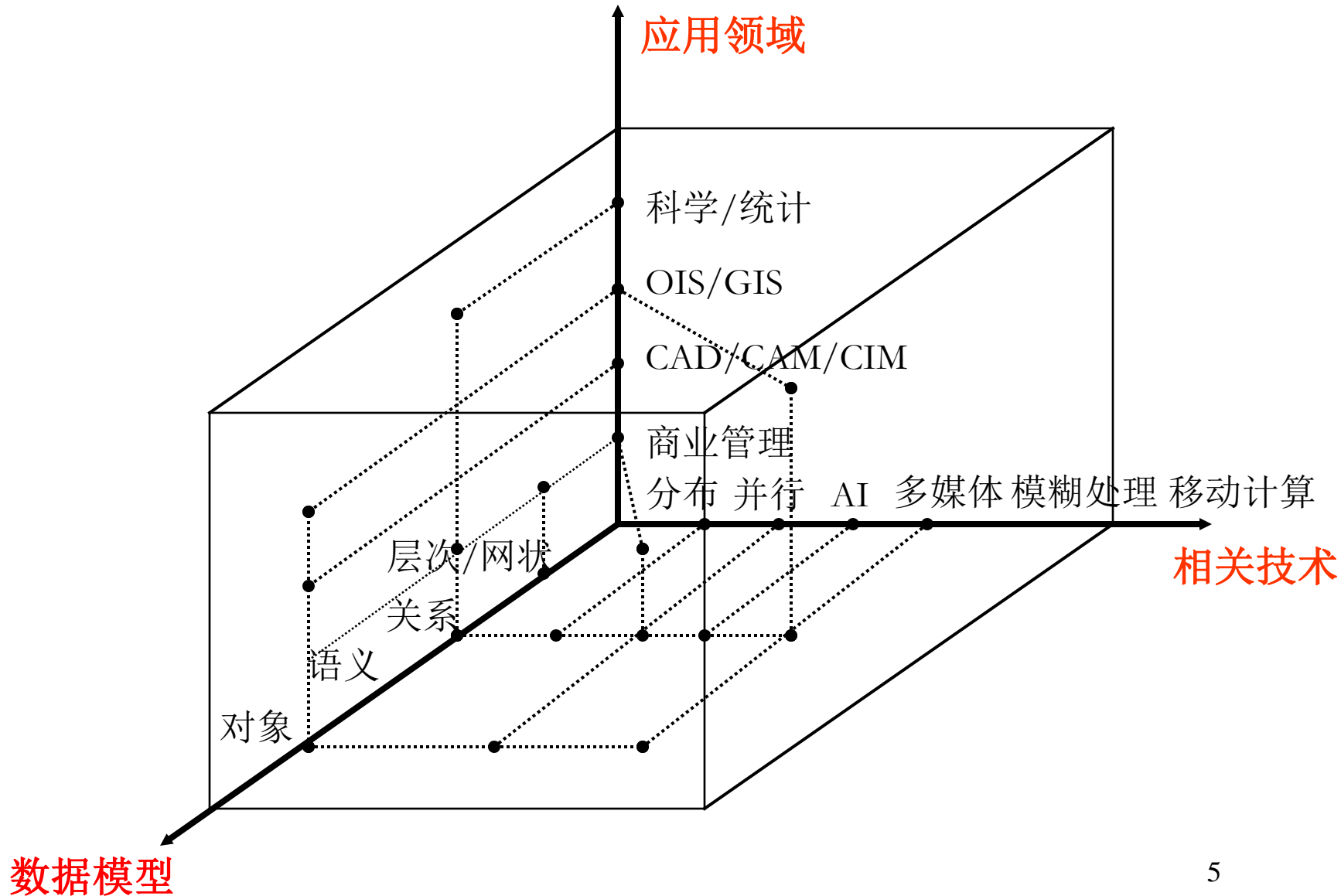
# 目录

- 数据库系统发展
- 关系模型与面向对象模型
- 集中式系统与分布式系统

# 数据库系统发展

- 数据模型
  - 数据结构, 操作
  - 层次, 网状, 关系, OO, 时态, 空间, 图, 时空, 文本, 工程, 生物, 地理, 图像数据库等
- 运行环境
  - 集中式, 分布式, 并行, Cluster, Web, 移动, 嵌入, 数据网格等
- 计算方式
  - 被动/主动, 实时数据库等

# 数据库系统发展及其相互关系示意图



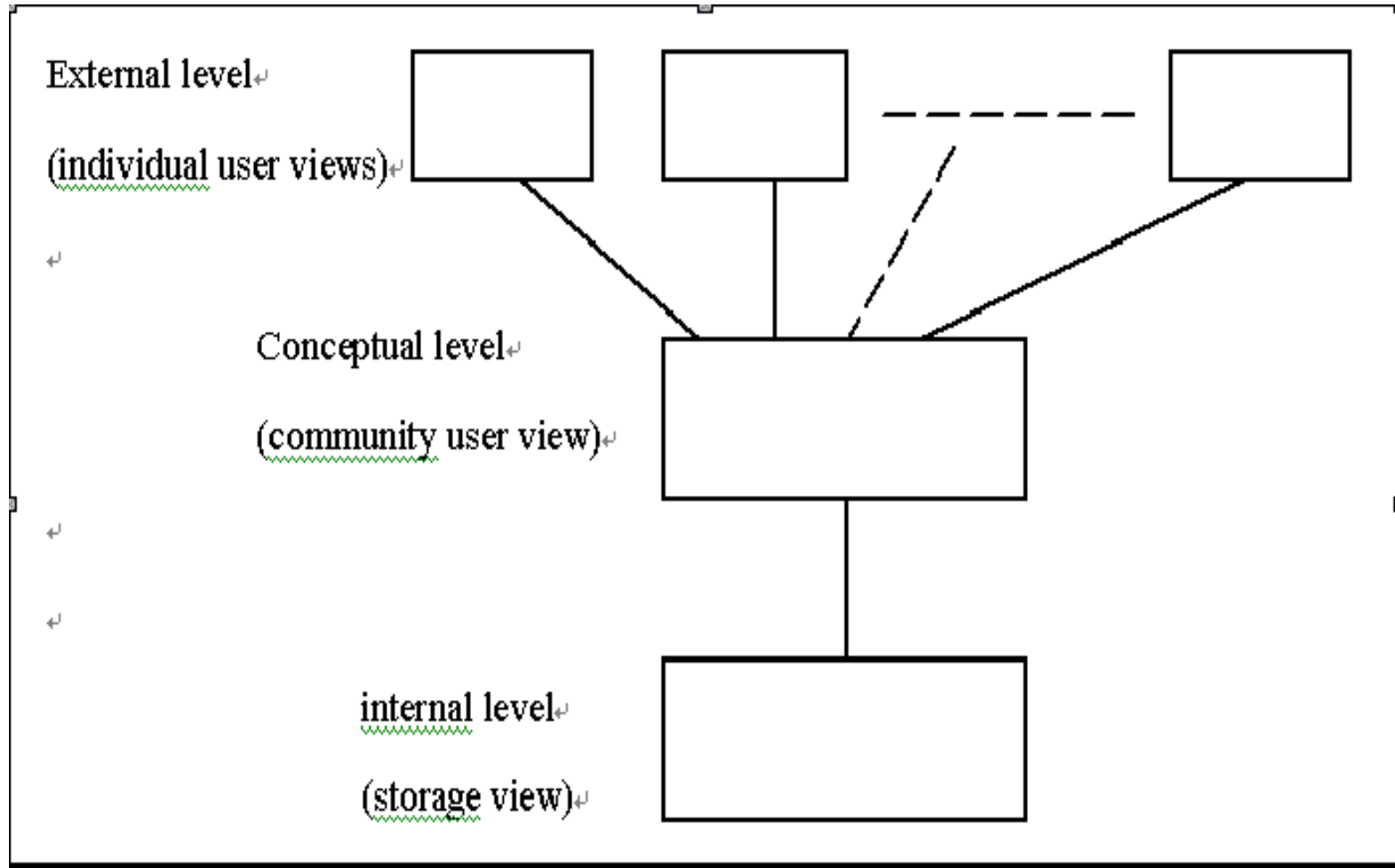
# 数据库系统

- **数据库**-可以存储长时间的信息集合
- **数据库管理系统**-允许用户使用专门的数据定义语言和操作语言,支持大数据量数据的长时间存储,并能正确控制多个用户对数据的立即存取的软件
- **用户**-包括DBA,终端用户,应用程序员等

# 数据库方法优点

- 数据共享
- 减少数据冗余
- 数据一致
- 支持事务处理
- 数据完整性
- 数据安全性
- 并发控制
- 数据标准
- 数据独立（逻辑独立、物理独立）

# 三级体系结构

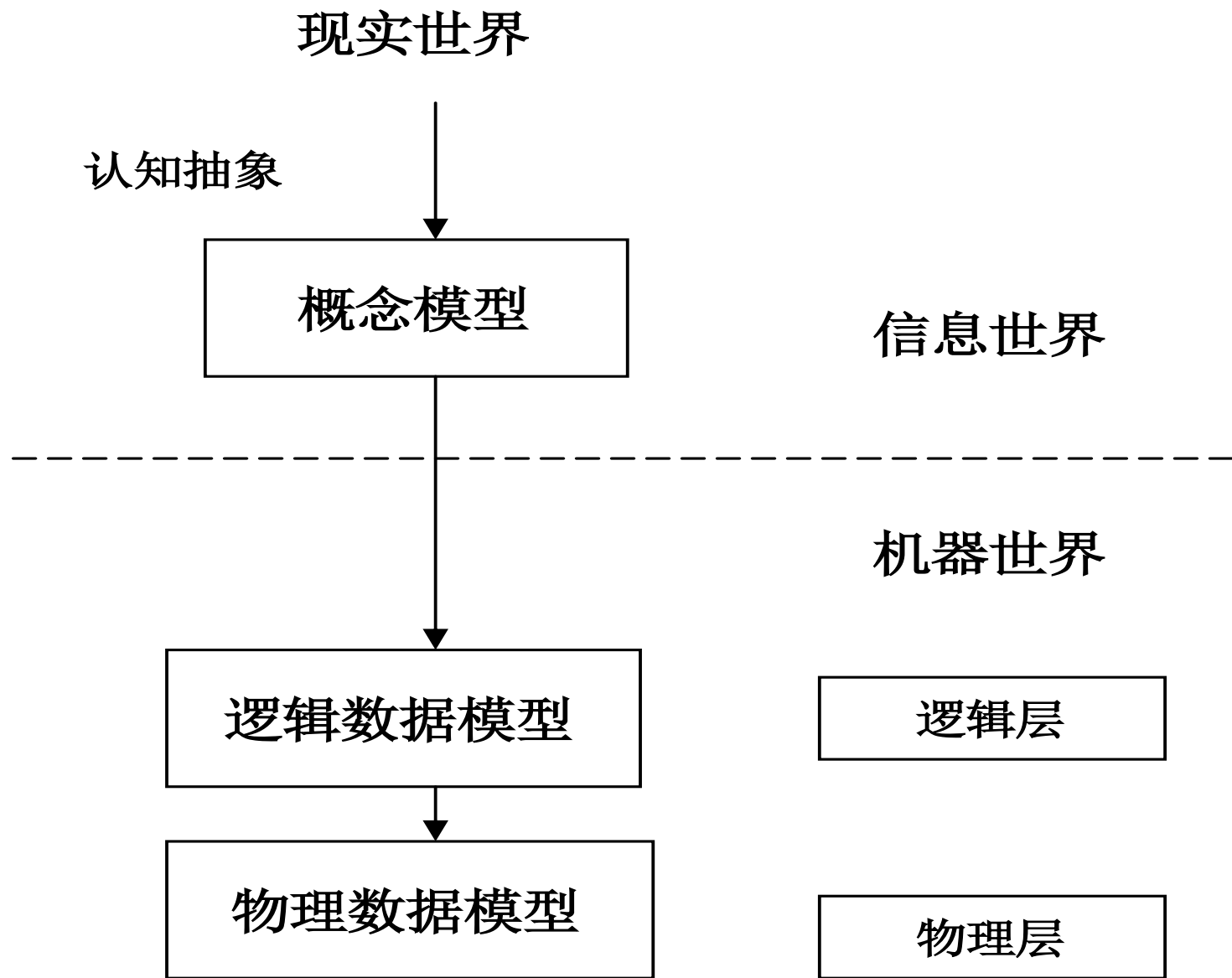




# 数据视图（模型）

# 数据模型

- 数据模型为数据特征的抽象：
  - 真实描述现实世界
  - 易为人所理解
  - 便于在计算机上实现
- DBMS支持的数据模型直接支持用户对现实世界的描述，其能力受实现模型的限制
- 抽象方法：用三层模型逐步抽象



# 三层模型表示

- 物理层 ——对用户透明
  - 数据的表示（存储格式）
  - 数据的索引
  - 高效的操作实现
  - 一致性维护及故障处理

# 三层模型表示—续

- 逻辑层——数据模型
  - 用户通过DBMS看到的现实世界
  - 传统的DBMS模型：层次，网络，关系模型
  - 新型的DBMS模型：O—O模型，对象—关系模型等

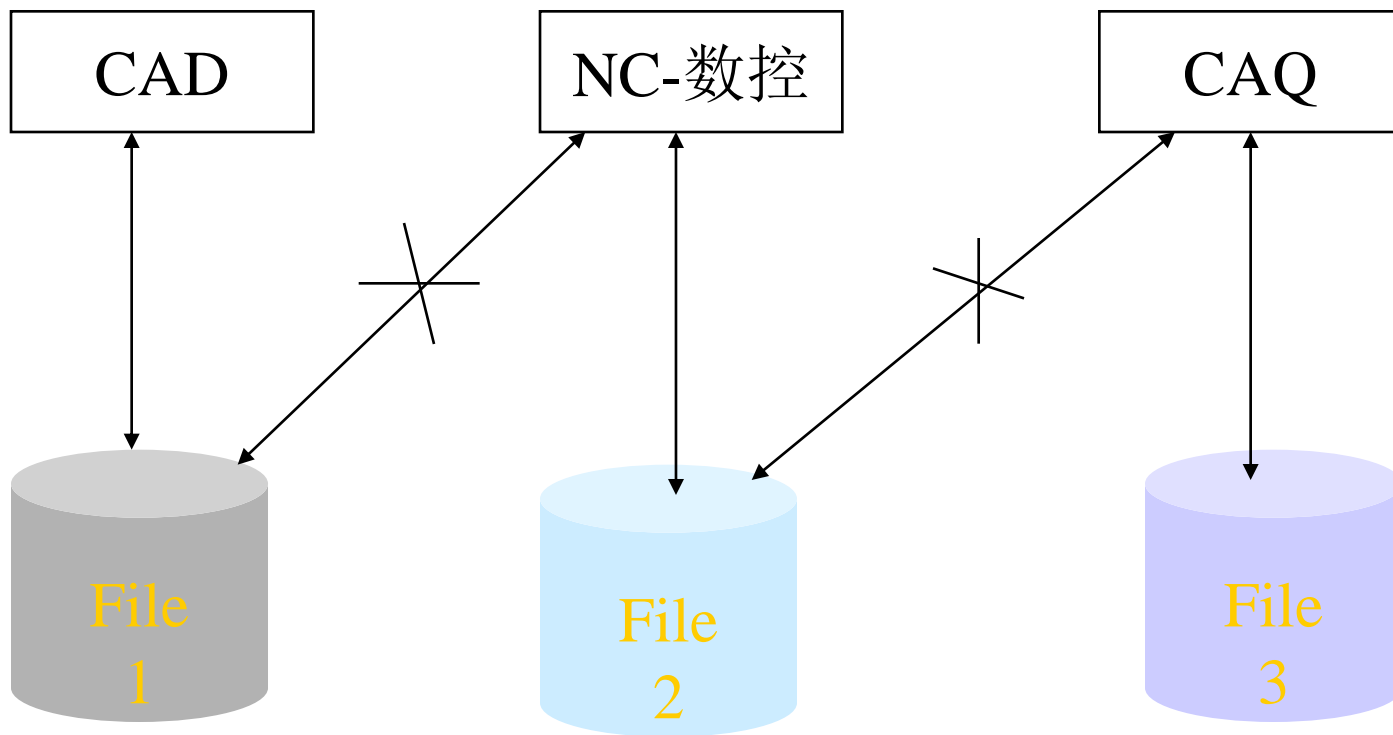
# 三层模型表示—续

- 概念层——直接面对用户
  - 语义表达能力强
  - 易于被用户理解
  - 独立于任何DBMS
  - 易于向逻辑模型转换
  - 语义模型

# 应用需求

- （非传统）应用领域
  - CAD/CAM ;
  - CASE, GIS;
  - 知识库系统;
  - 实时系统;
  - 遥感信息处理系统;
  - 医学影像信息处理系统
  - Web
  - 等等。

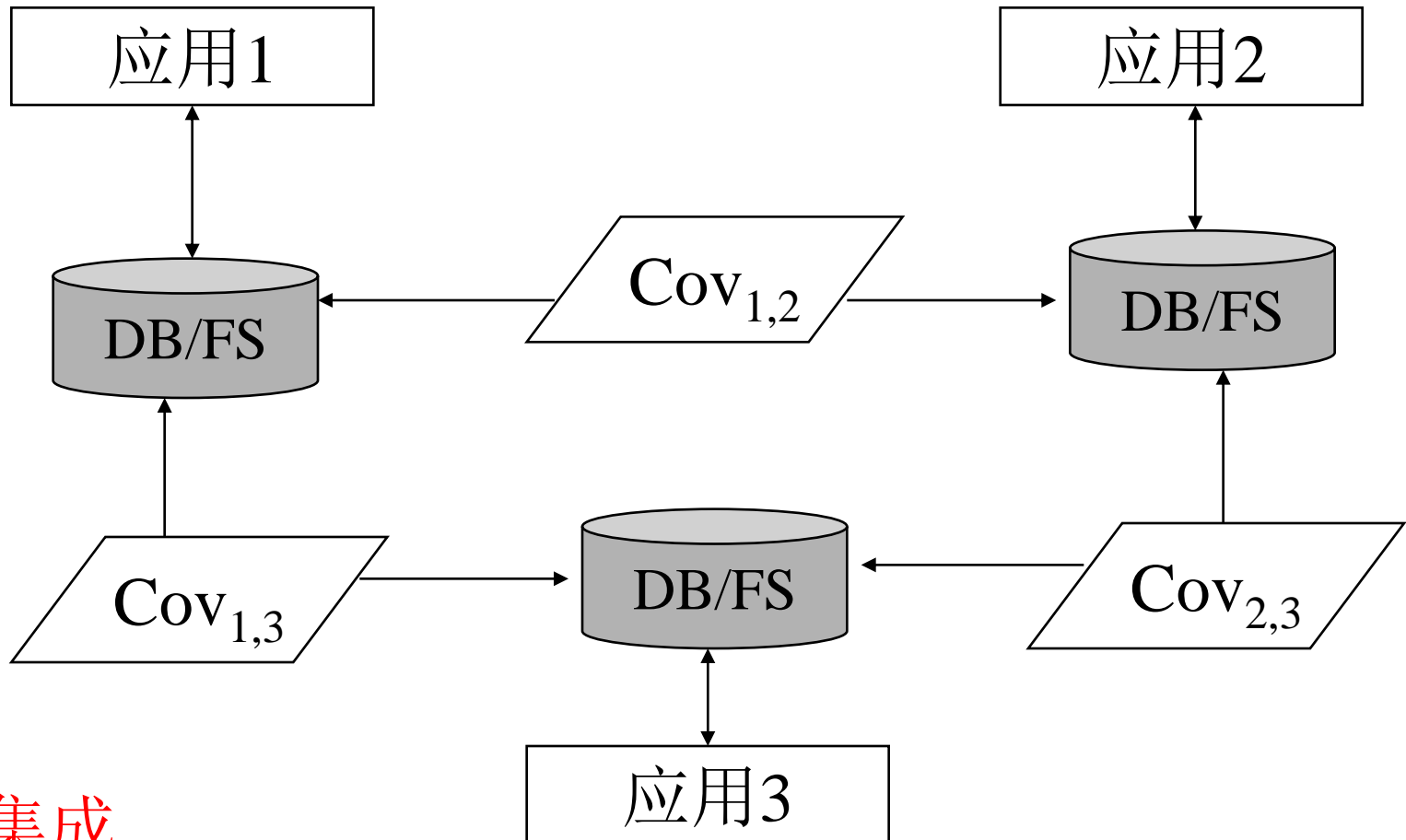
# CAD/CAM中的集成问题



无集成

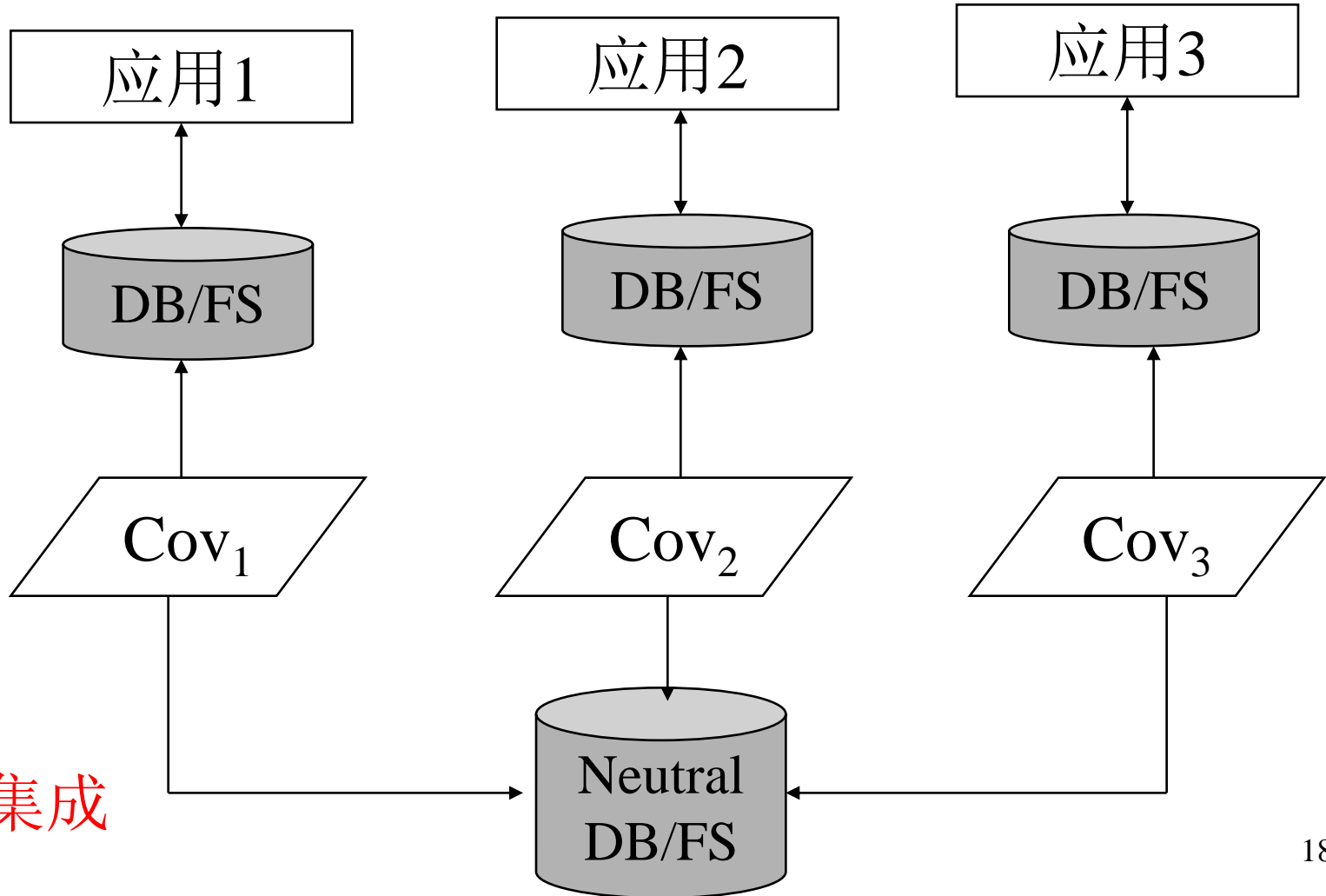


# CAD/CAM中的集成问题-续



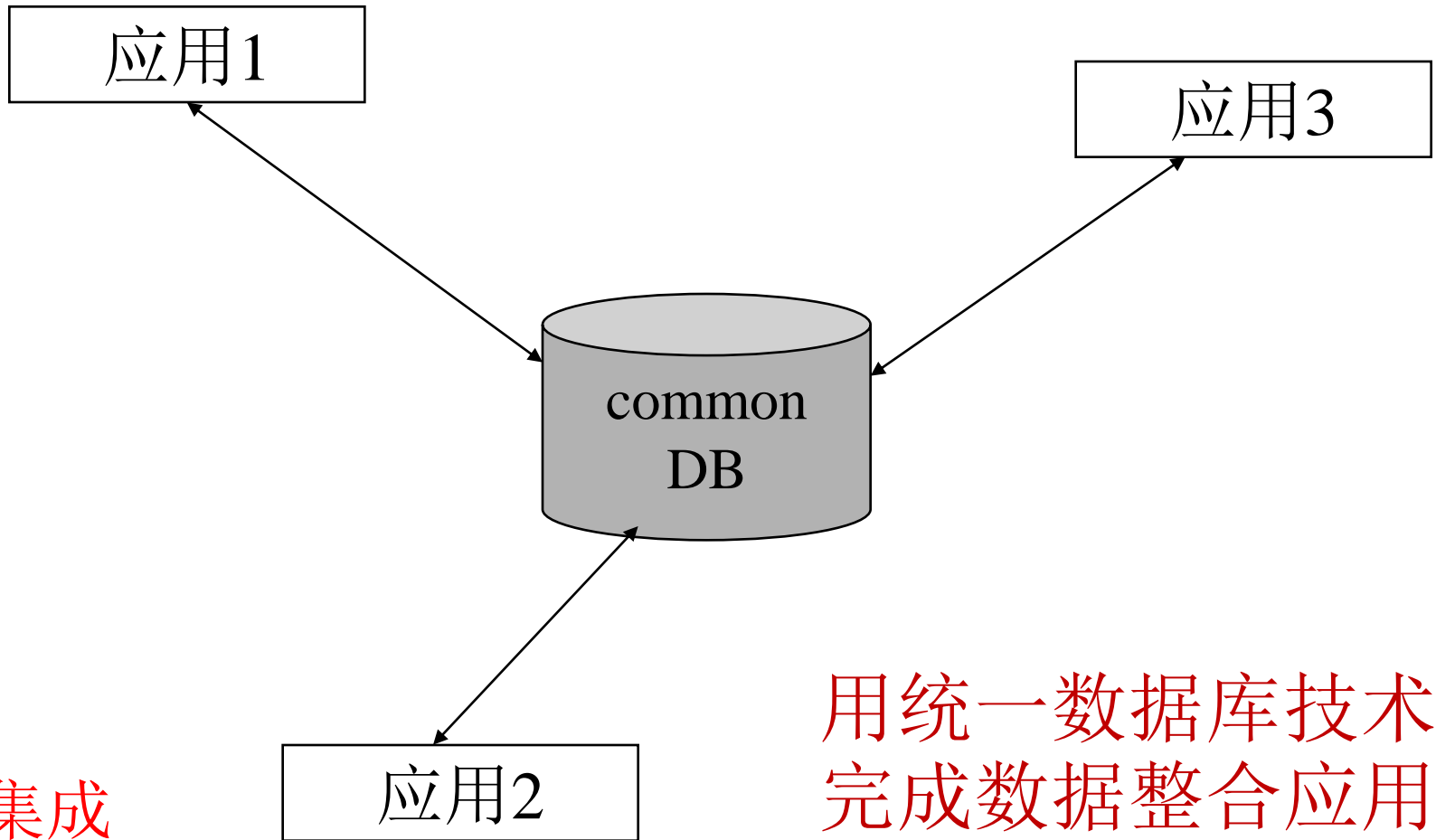
1级集成

# CAD/CAM中的集成问题-续



2级集成

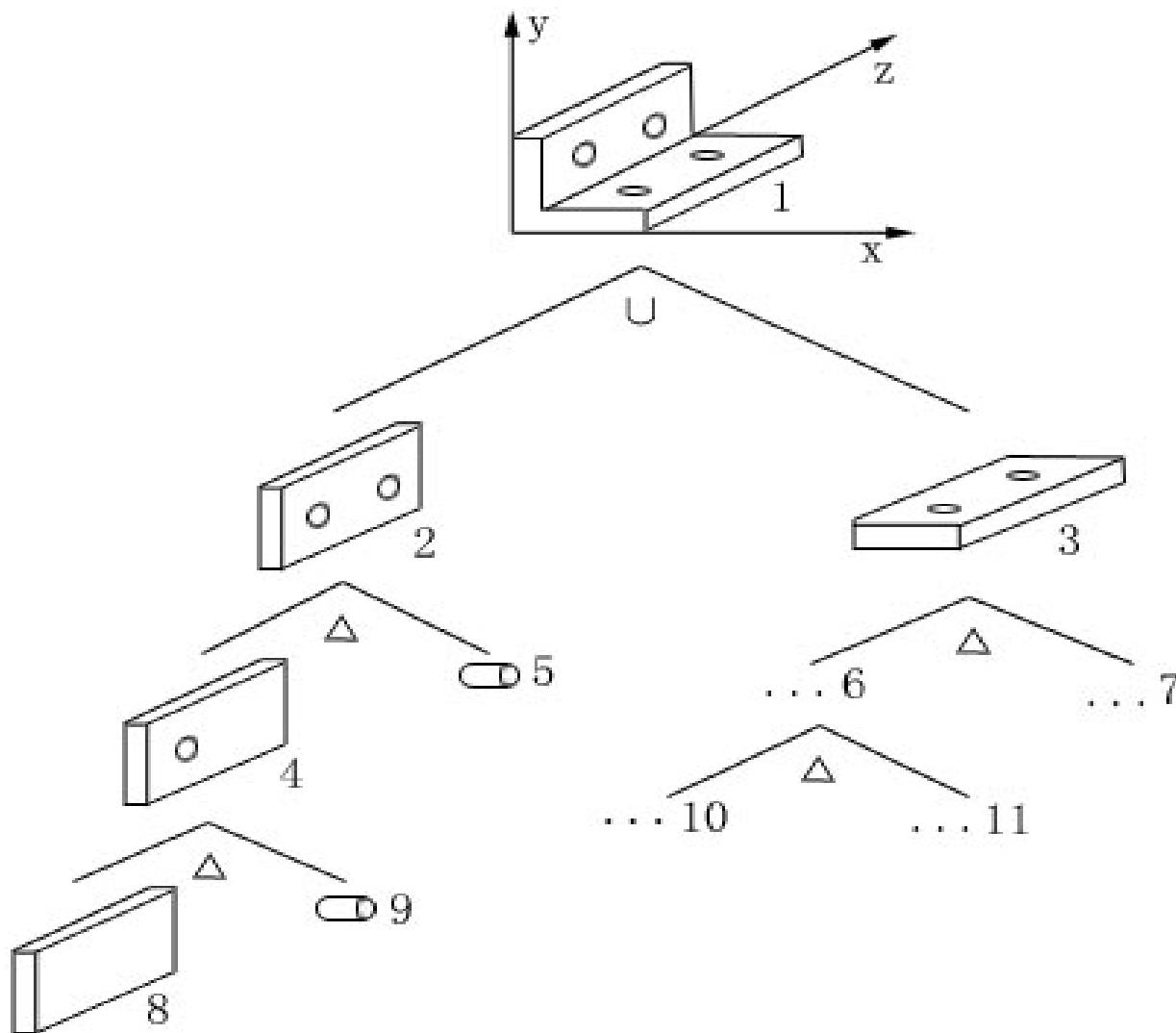
# CAD/CAM中的集成问题-续



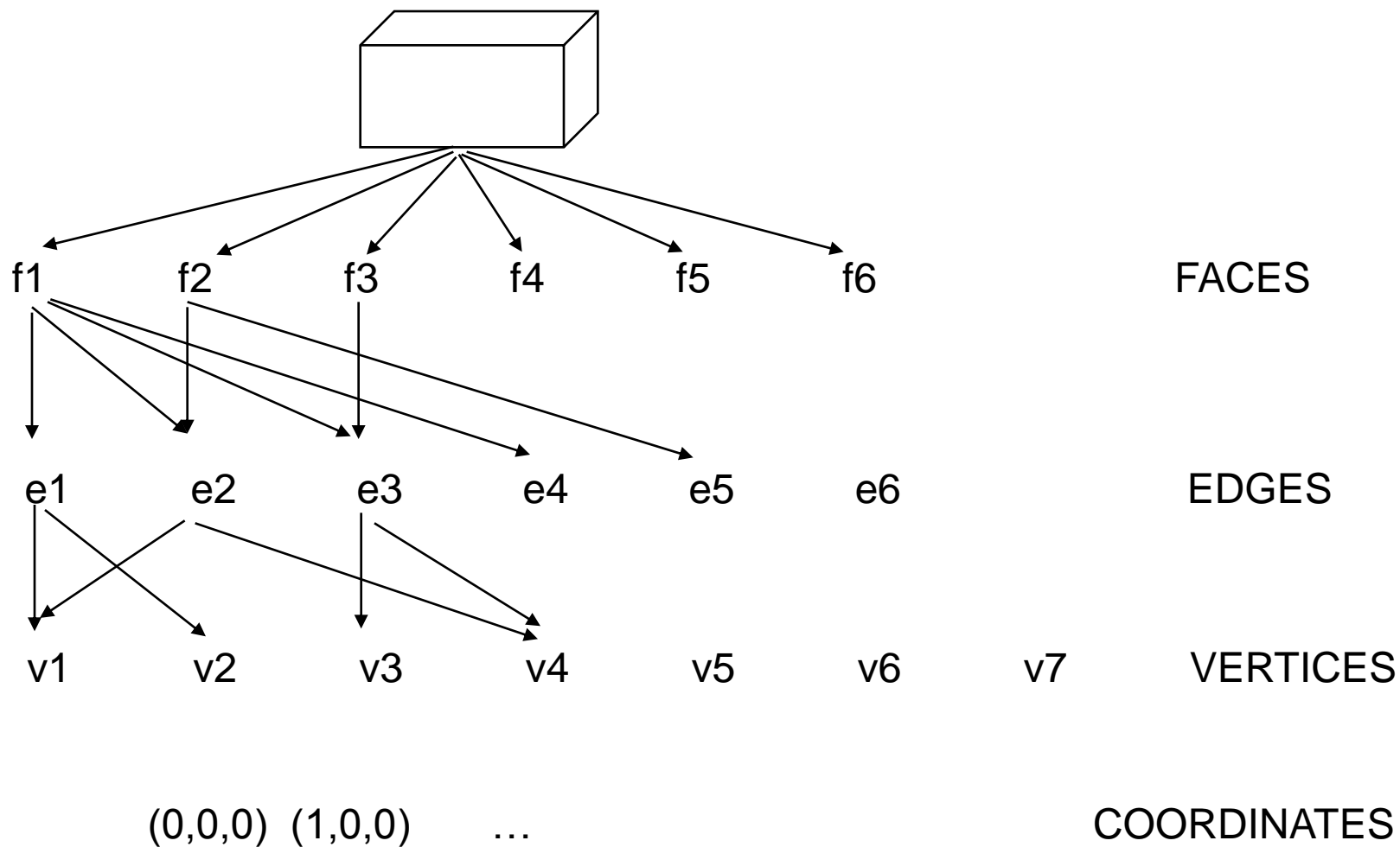
3级集成

用统一数据库技术  
完成数据整合应用

# 例：支架的CSG(constructive solid geometry)树



# 例：立方体Cuboid的边界表示



# 新领域的应用需求

- 复杂的数据类型定义：
    - 抽象数据类型，
    - 无结构超长数据类型，
    - 时间，版本数据类型，
    - 图像，图形数据类型，
- 特点：结构复杂，难于用关系模型描述。

# 新领域的应用需求-续

- 复杂对象的存取结构；
- 对象间复杂多样的语义联系及关联查询；
- 长事务和嵌套事务的处理；
- 对象模型的演进和版本控制；
- 程序设计语言与数据库语言的无缝集成。

# 传统数据模型

- 层次模型 (Hierarchical system 1960s)
  - C.W. Bachman IDS (Integrated Data Store)
  - 树结构模型
  - 指针导航式操作
  - 数据被多个应用共享
- 网状模型 (Network System 1970s)
  - CODASYL DBTG 报告
  - 图结构模型
  - 指针导航式操作



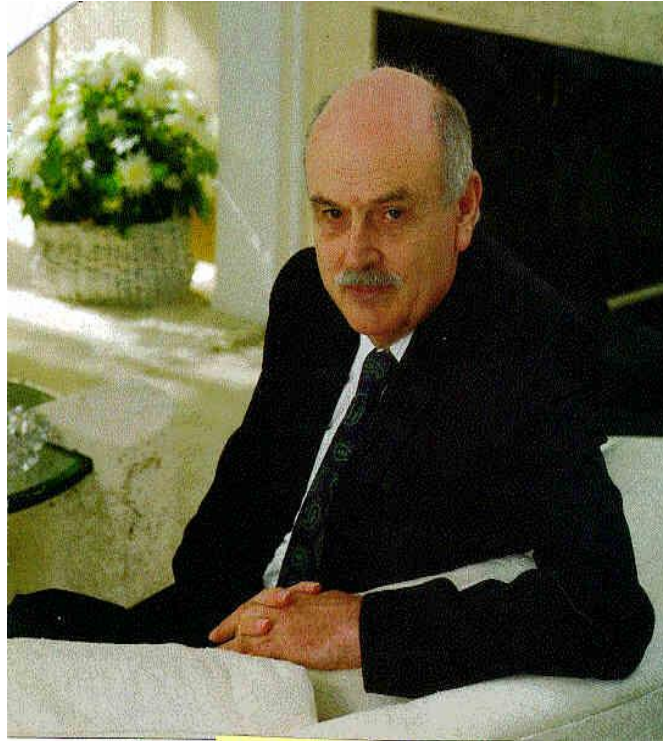
# 传统数据模型-续

- 关系模型 (Relational system 1980s)
  - E.F.CODD 1969-1970提出
  - 二维表结构模型
  - 集合操作
  - 产品： DB2, Ingres, Informix, SQL server, Oracle, Sybase, 等.
- 图灵奖
  - Bachman (network DB) in 1973
  - E.F.Codd (Relational DB) in 1981
  - Jim Gray (Transaction) in 1998

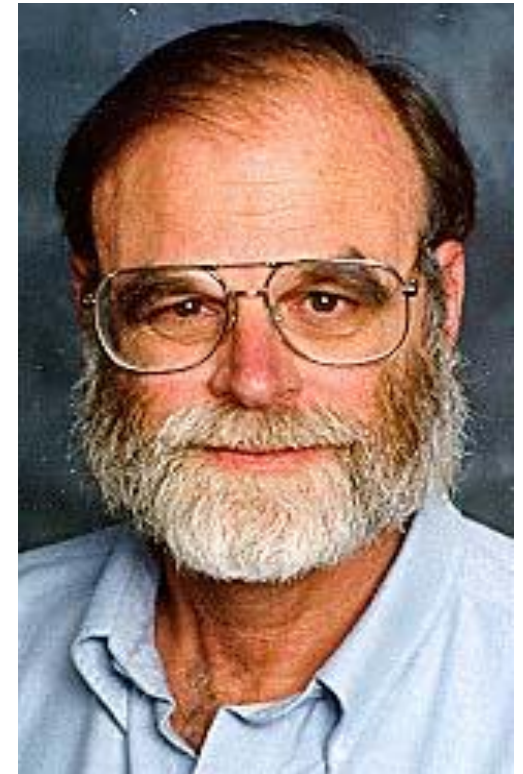
# 三位获奖者



**C.W. Bachman**



**E.F. Codd**



**Jim Gray**

# 传统数据库系统的局限性

- 面向机器的语法数据模型：
  - 强调高度结构化
    - 数据抽象简单
    - 语义表示能力差

# 传统数据库系统的局限性-续

- 直接使用的缺点
  - 数据库结构与外部结构不一致
  - 由联接构造的视图不可更新
  - 关系系统是基于值的
- **数据类型简单，固定**
  - ✓ 简单类型：整形，实型，字串，等等
  - ✓ 固定的类型集合，不支持ADT和嵌套

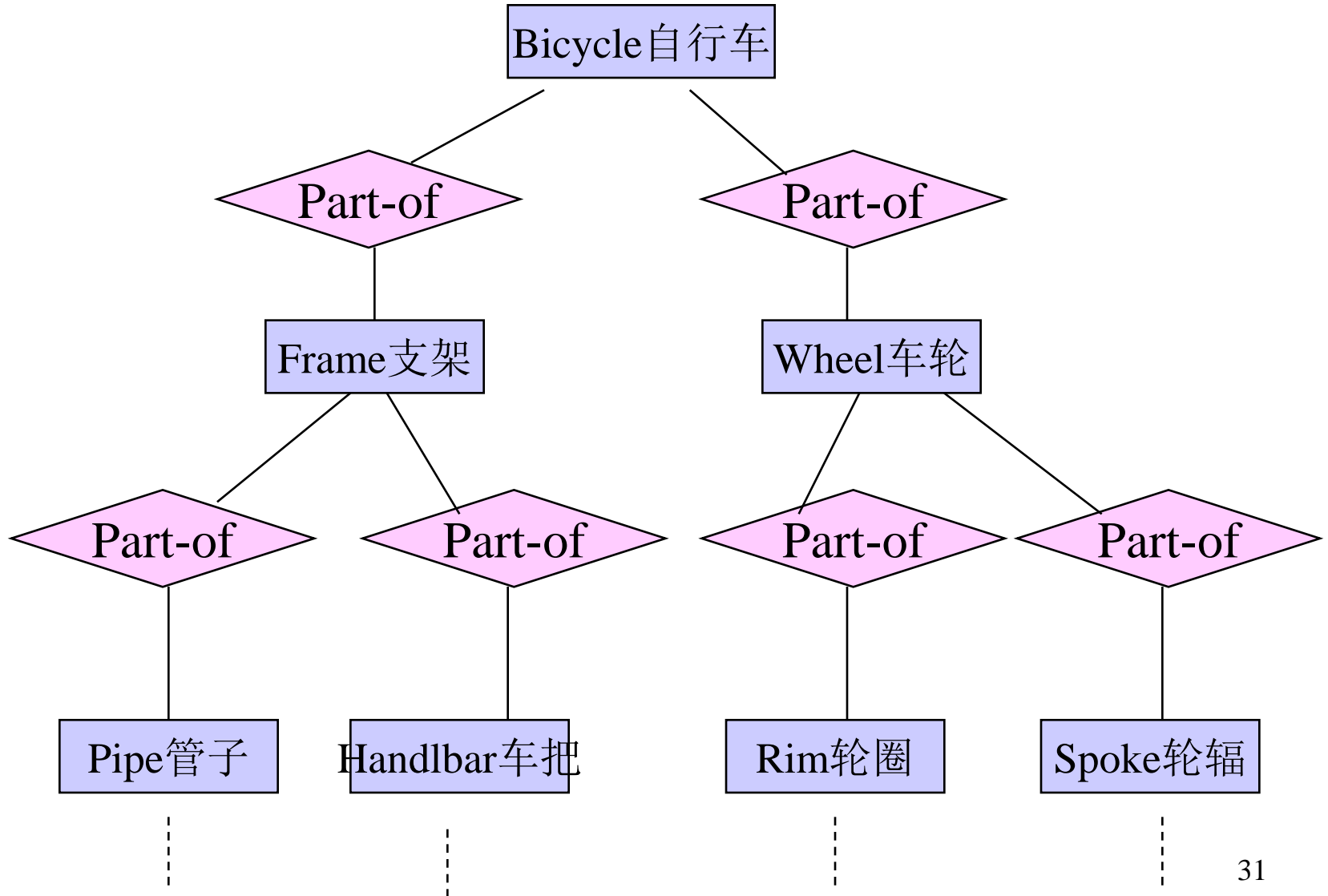
# 传统数据库系统的局限性-续

- **结构与行为完全分离**
  - 不支持抽象行为描述
- 阻抗失配
  - SQL与通用编程语言的失配
    - 编程模式失配
    - 类型系统不匹配
- 被动响应

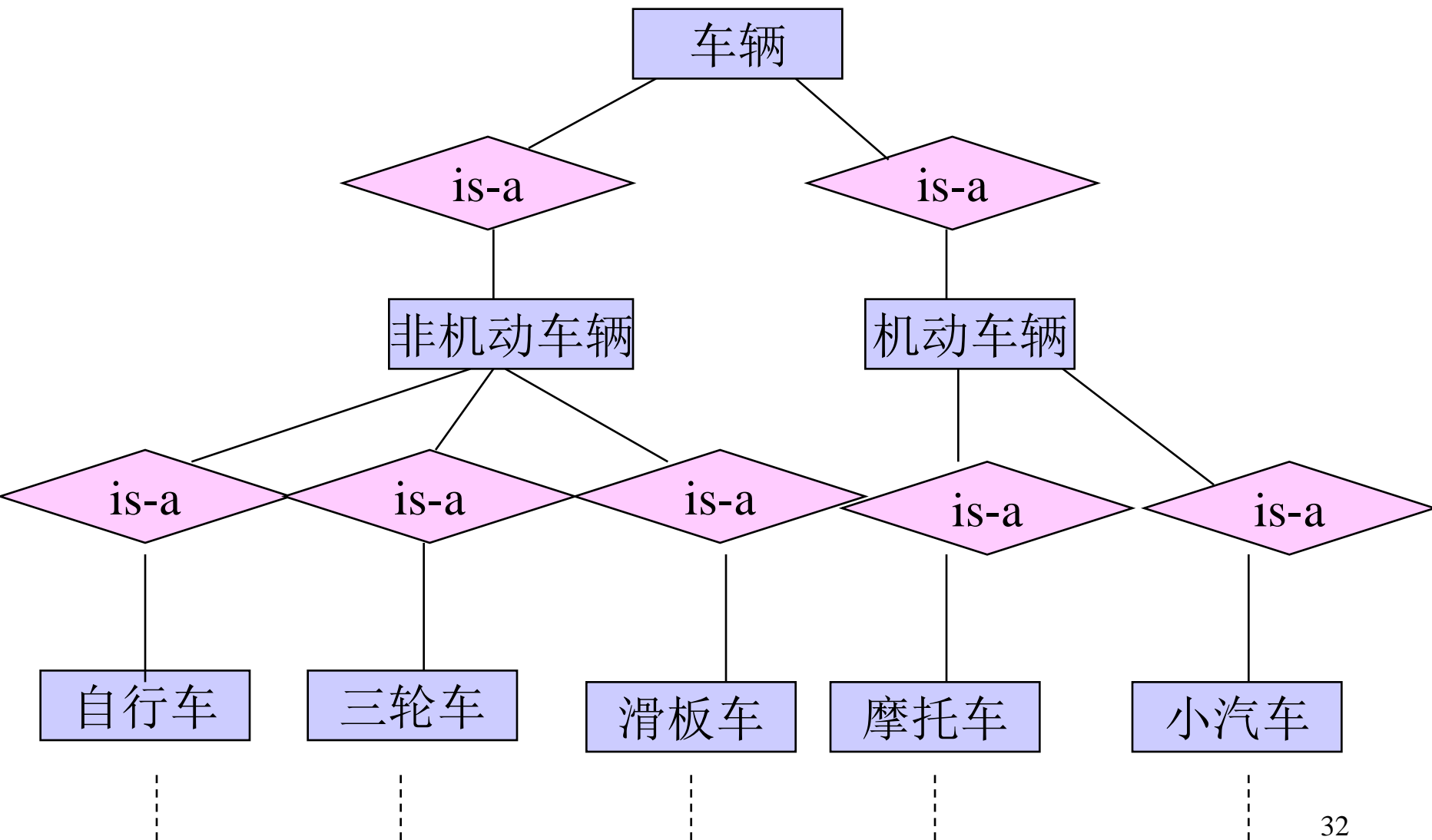
# Smith & Smith 数据抽象

- Aggregation (聚合)
  - Part-of 联系
- Generalization (泛化)
  - Is-a 联系
- Specialization (特化)
  - Is-a 联系

# 聚合 (aggregation)抽象举例

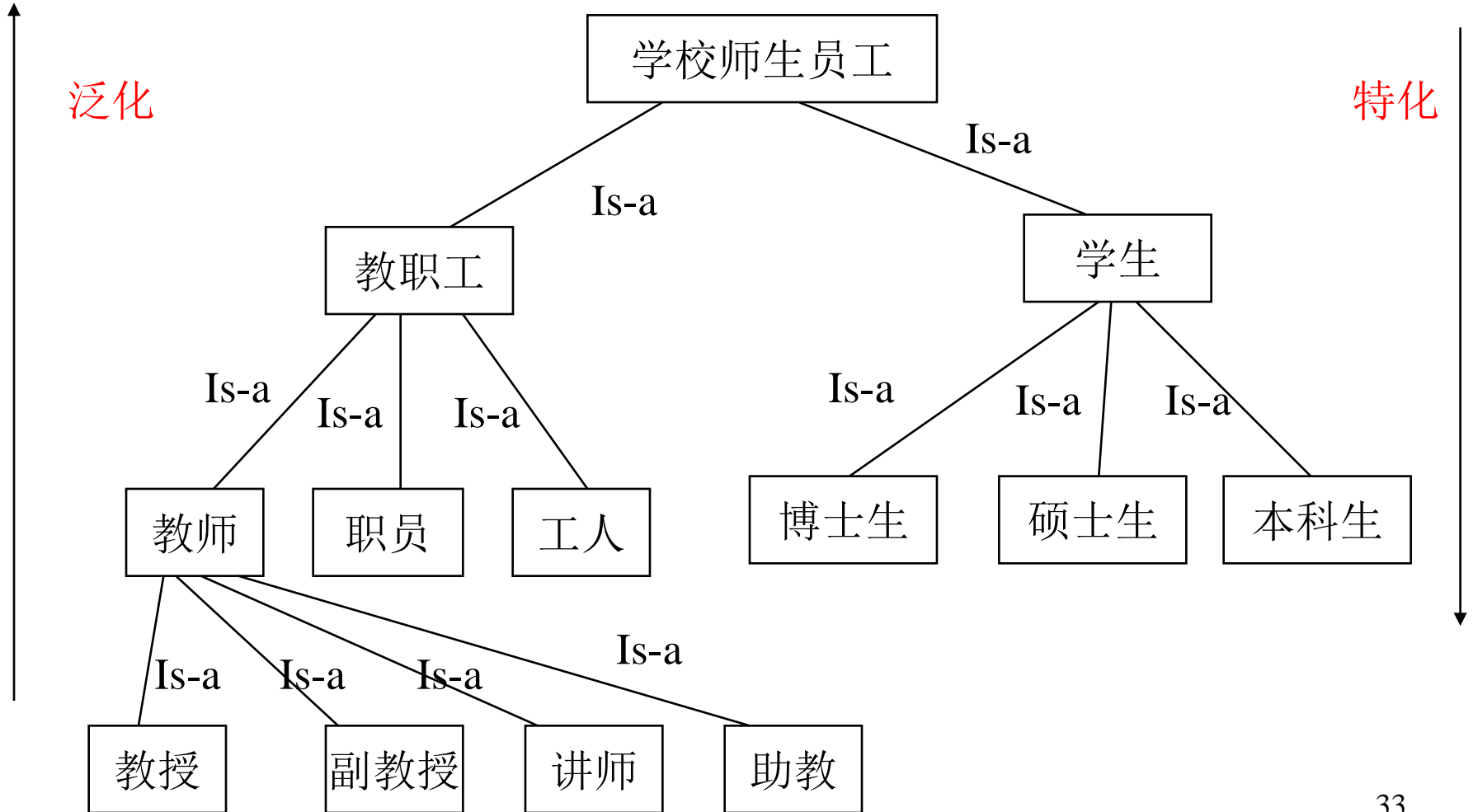


# 泛化（generalization）抽象举例





# 泛化 vs 特化



# 概念模型及其描述工具

- **E/R 模型：实体—联系模型**
  - 用实体描述对象属性
  - 用联系描述对象间的语义关系
- 缺点：
  - 语义表达能力不够强
  - 不能描述实体的行为
- **EER 模型**

# 概念模型及其描述工具-续

- 语义对象模型（Semantic Object Model）
  - 定义了7种对象类型，增强了对对象语义表达能力
  - 可以描述聚合抽象和继承抽象
  - 描述对象间的联系和依赖关系
- 缺点：不能描述对象的行为

# 概念模型及其描述工具-续

- 基于UML的面向对象的模型（Unified Modeling Language）
  - 静态对象模型
    - 描述对象的属性，关联，聚合，继承，等等
  - 动态对象模型
    - 描述对象行为，相互作用，合作和变化，等等
  - 支持概念模型与逻辑模型无缝联接

# 概念模型及其描述工具-续

- 基于XML的对象模型
  - 易学，易扩充
  - 易于快速开发软件
  - 已有多个工具可以支持结构转换
- 问题：XML不是一个面向对象的语言
  - 不提供对象间的联接
  - 不支持属性内容的共享
  - 不支持继承和运算等O—O特征

# 逻辑模型的发展

- 关系模型的扩充 —— 复杂数据模型
  - 结构扩充：嵌套关系模型  $NF^2$
  - 语义扩充：支持继承，可以定义函数域和运算
  - 局限性：
    - 仍然用关键字标识元组
    - 不支持太多的关联语义

# 逻辑模型的发展—续

- **面向对象模型**——O—O模型
  - 对象标识 OID
  - 对象封装
  - 类和类的嵌套
  - 类和类的继承
  - 持久对象概念

# 逻辑模型的发展—续

- 面向对象模型的问题：
  - 索引与封装
  - 导航的计算与集合操作
  - 数学基础不完美
  - 多重继承的多义性
  - 共享对象与持久对象的引用副作用，等等



# 逻辑模型的发展—续

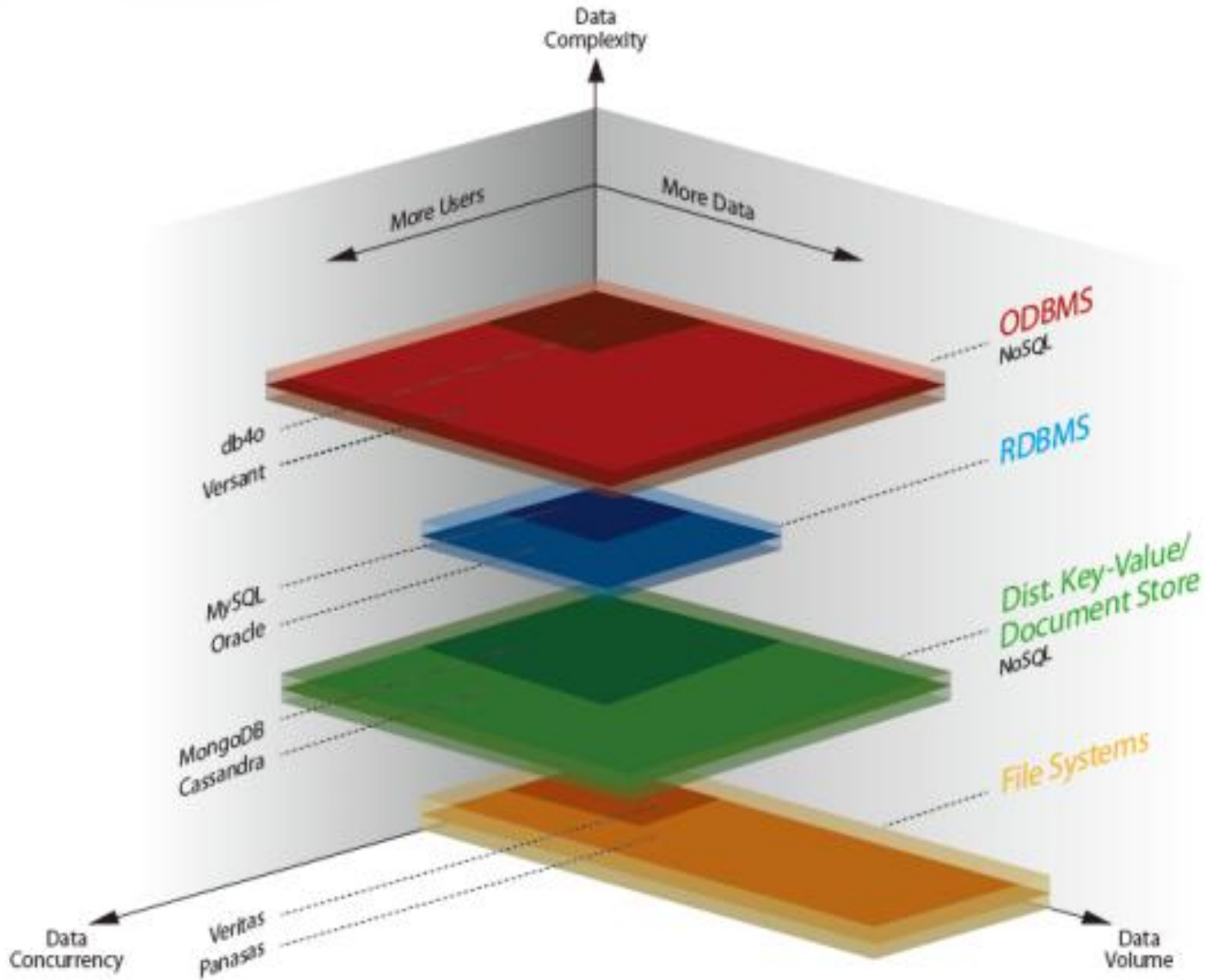
- 对象—关系模型

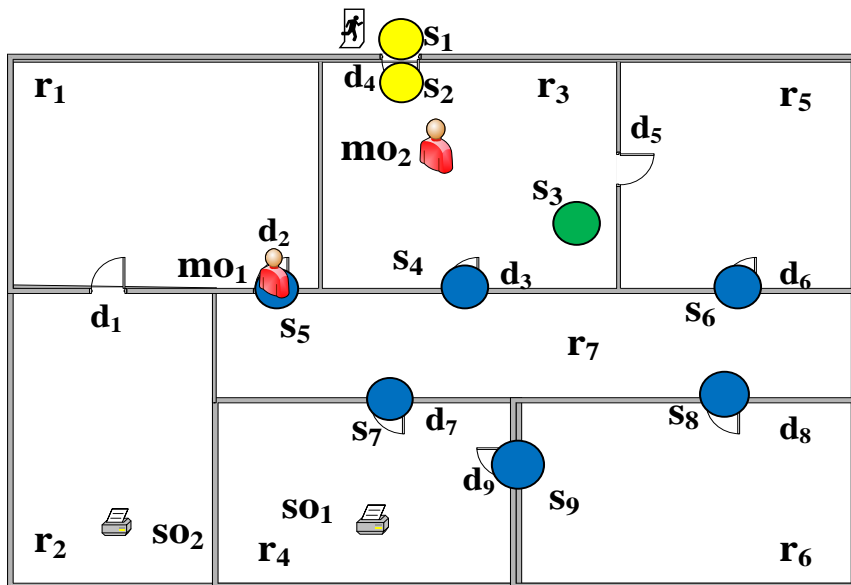
- 在关系的基础上扩充面向对象的概念

- 支持抽象数据类型（允许定义复杂类型）
    - 支持对象行为（允许定义操作）
    - 支持复合对象（允许对象嵌套）
    - 不支持继承概念

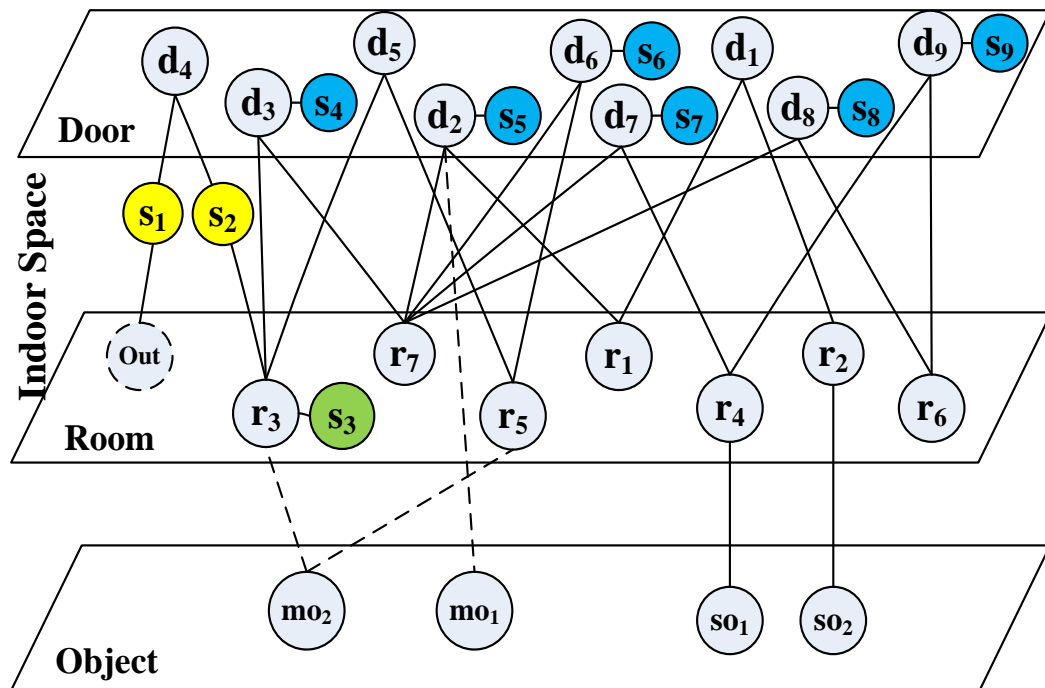
# 面向对象数据库

- 面向对象方法在数据库领域中的应用
- 面向对象数据库的主要特征与要求
  - 具有多种数据类型并能构造抽象数据类型能力
  - 具有构造复杂数据结构与模式能力
  - 具有多种数据操作能力
  - 具有模式演化能力





基于单元房间的室内空间的例子



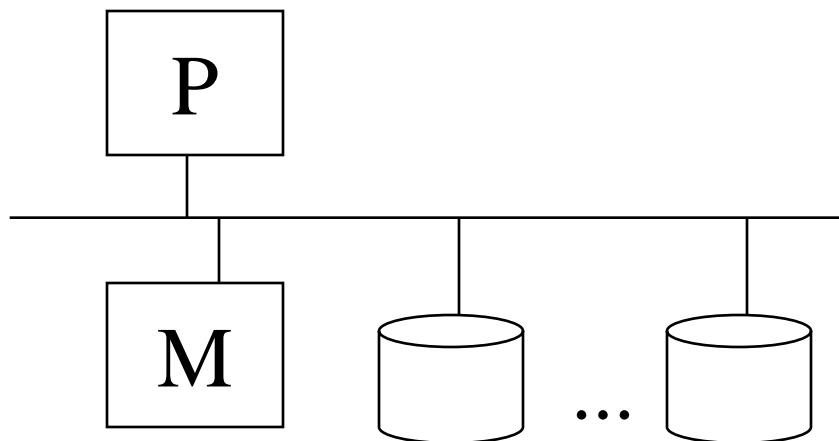
室内移动对象的  
LayeredModel模型示例

# 数据处理

# 事务处理

- ACID性质
- 并发控制
- 恢复管理

# 集中式数据库系统



软件:

应用程序
SQL
查询处理器
事务处理器
文件访问

- 简化

- 单个前端接口
- 集中维护锁
- 如果处理器故障,则系统故障 ...

# 分布式系统

- 计算机网络的发展带来了分布式系统
- 分布式系统是一些独立的计算机集合，但是对这个系统的用户来说，系统就像一台计算机一样
- 硬件角度 单个计算机是自治的
- 软件角度 整个系统是一台计算机
- 特点
  - 高性价比、高速度、高可靠、高可扩展



# 分布式数据库系统

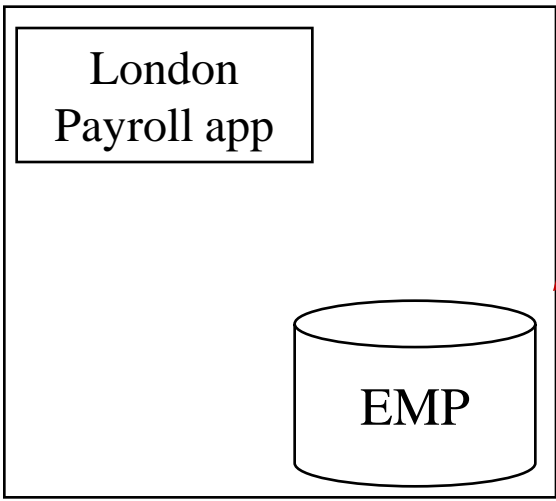
- 定义：物理上分散而逻辑上集中的数据库系统。
- 特点
  - 物理分布性  
数据不在单个Site上, 按全局需求将数据划分成一定的数据子集, 分散存储在各个Site上
  - 逻辑整体性  
各个Site上的数据子集, 相互间有严密的约束规则加以限定, 逻辑上是一个整体
  - 站点自治性  
各个Site上的数据(即LDB)是由本地的DBMS管理, 具有自治处理能力

# 为什么需要分布式数据库

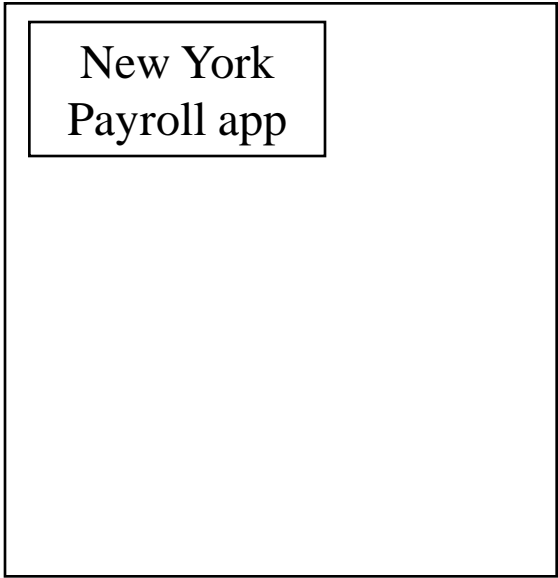
- 举例:
  - 某大公司分别在 London, New York 及 北京 有分公司.
  - 职工数据:  
EMP(ENO, NAME, TITLE, SALARY, ...)
- 问题:
  - 该职工数据表数据应该如何存储

# 职工数据的访问方式

- 大多数情形, 职工数据被其服务的各分公司管理
  - E.g. 工资, 福利, 聘用与解聘
- 定期地, 总公司统一地访问职工数据
  - E.g. 总公司修改影响所有职工的利益计划
  - E.g. 根据公司的全体净利润、销售确定年终分红与公司发展计划等

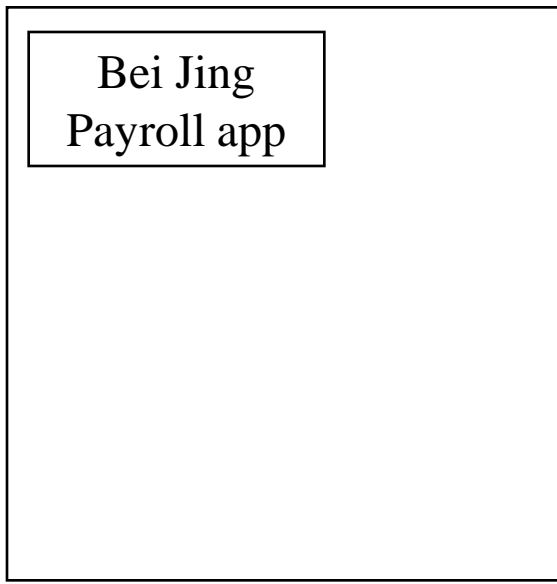


集中式



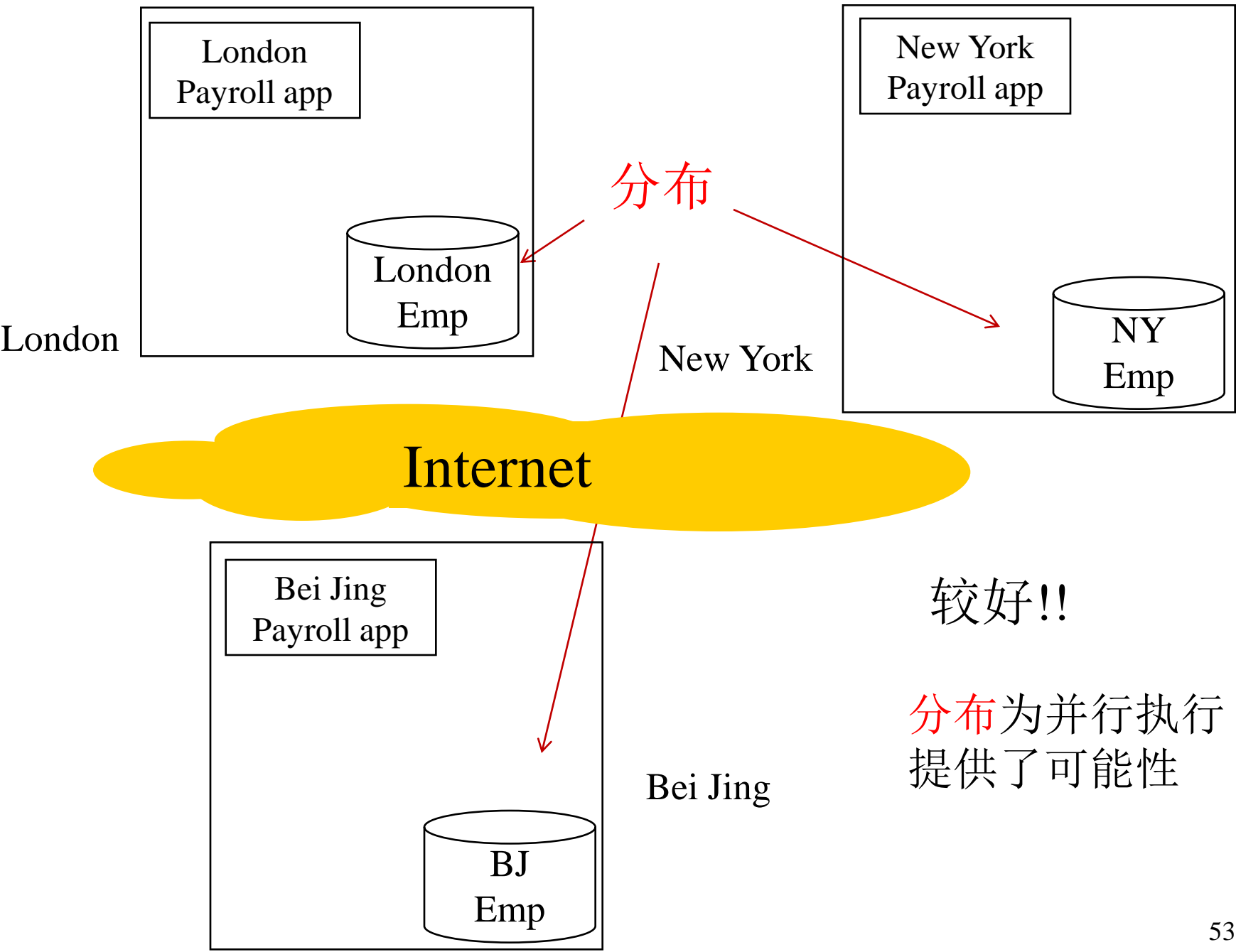
London

New York



纽约和北京的应用运行很慢!

Bei Jing



London  
Payroll app

London  
Emp

New York  
Payroll app

NY  
Emp

分布

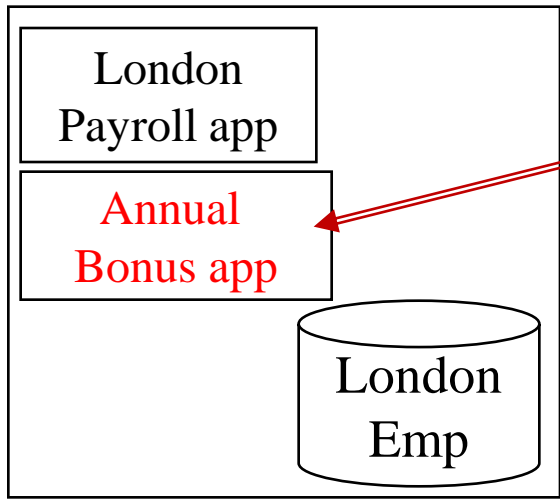
Internet

Bei Jing  
Payroll app

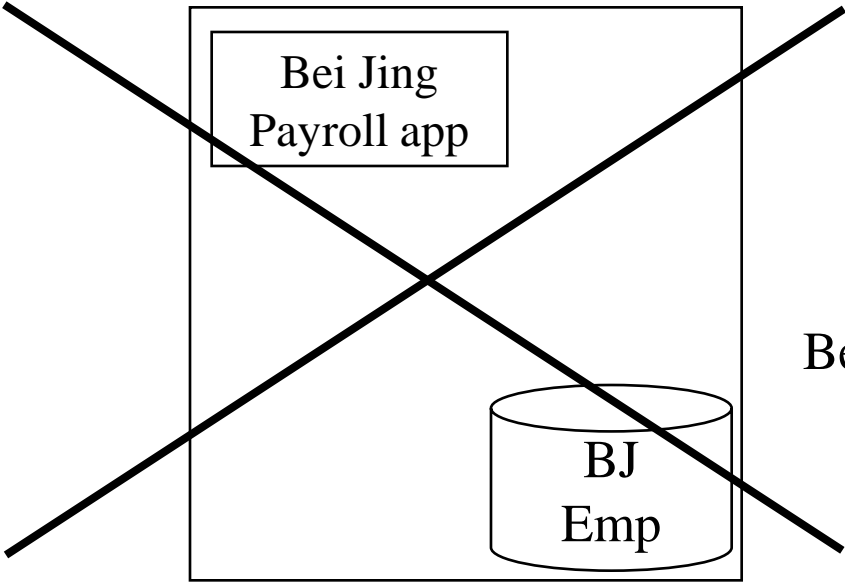
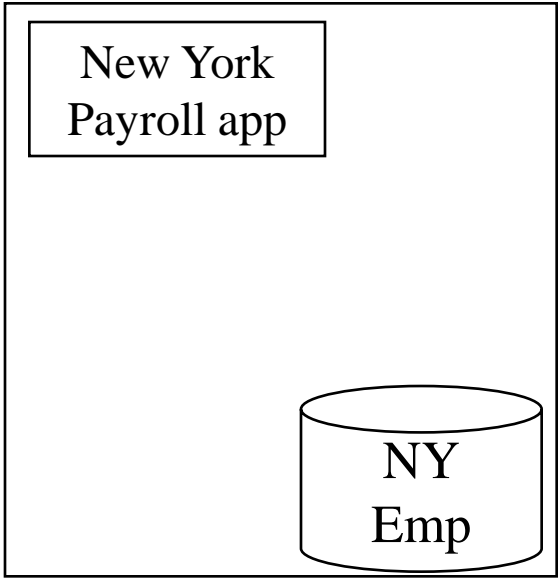
BJ  
Emp

较好!!

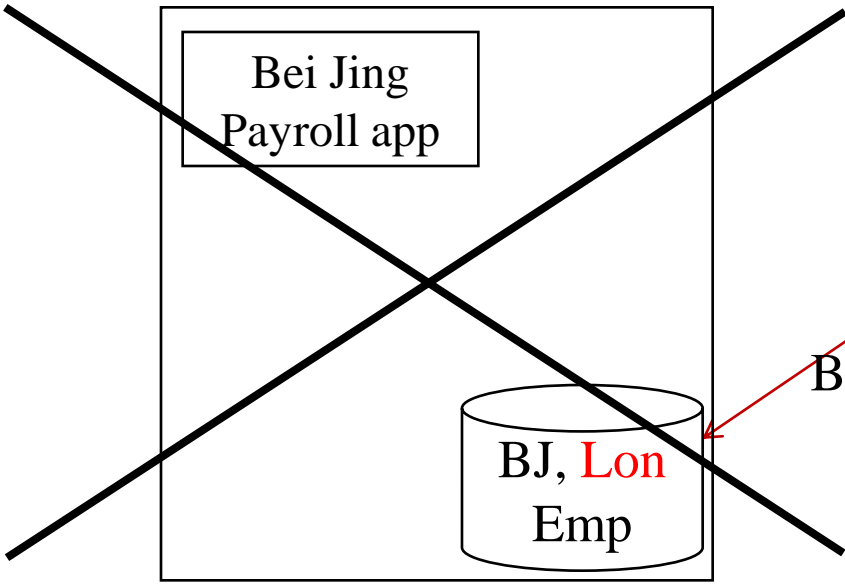
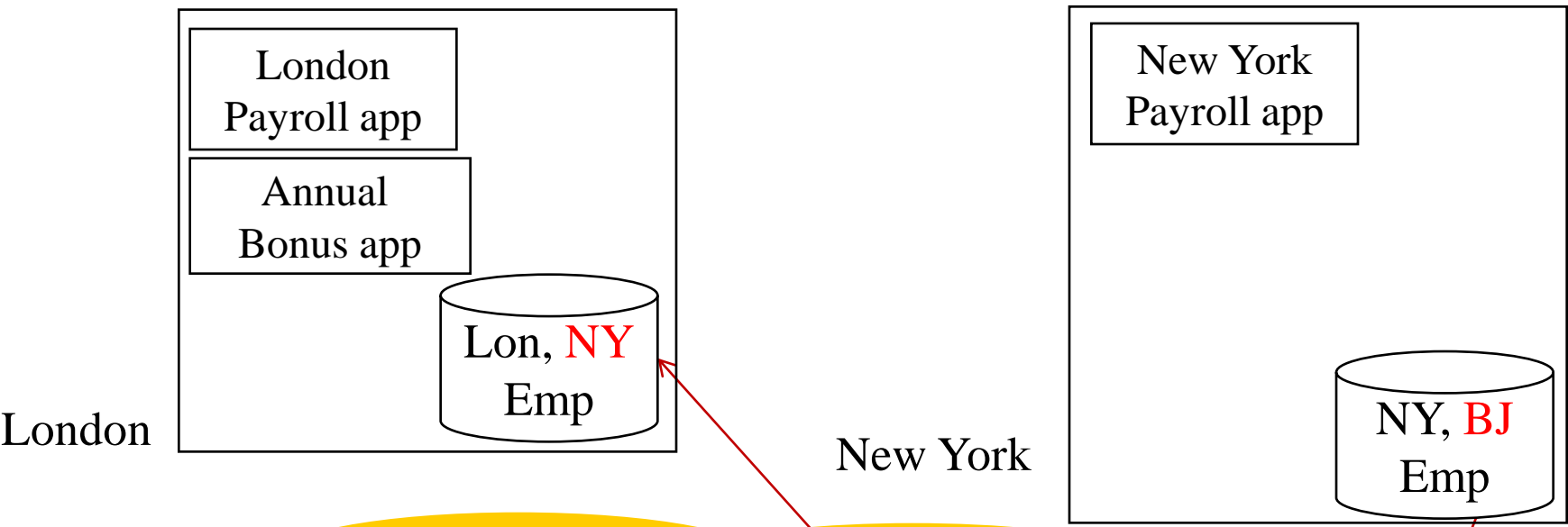
分布为并行执行  
提供了可能性



总公司应用



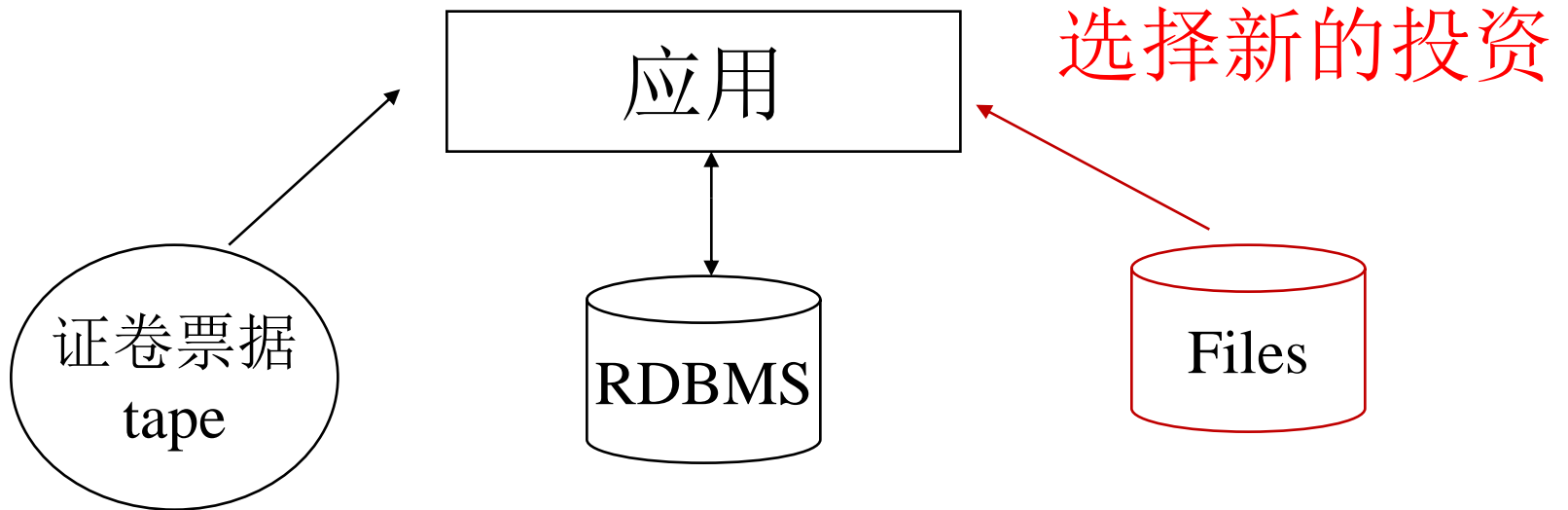
故障, 降低了可用性



复制增加了可用性

Bei Jing

# 异构与自治





# 分布的类型

- 同构型(Homogeneous)
  - 各个Site上的DB数据模型相同,
  - 进一步可分为
    - 同构同质型
    - 同构异质型
- 异构型(Heterogeneous)

各个Site上的DB数据模型不同.

# 分布控制的类型

- 集中型

DDB的全局控制信息位于一个中心Site上

- 分散型

每个Site上包含有全局控制信息的一个Copy

- 可变型

DDB分成两组, 一组Site包含全局控制信息

Copy(主Site), 另一组不含全局Copy(辅Site)

# 分布DB的研究

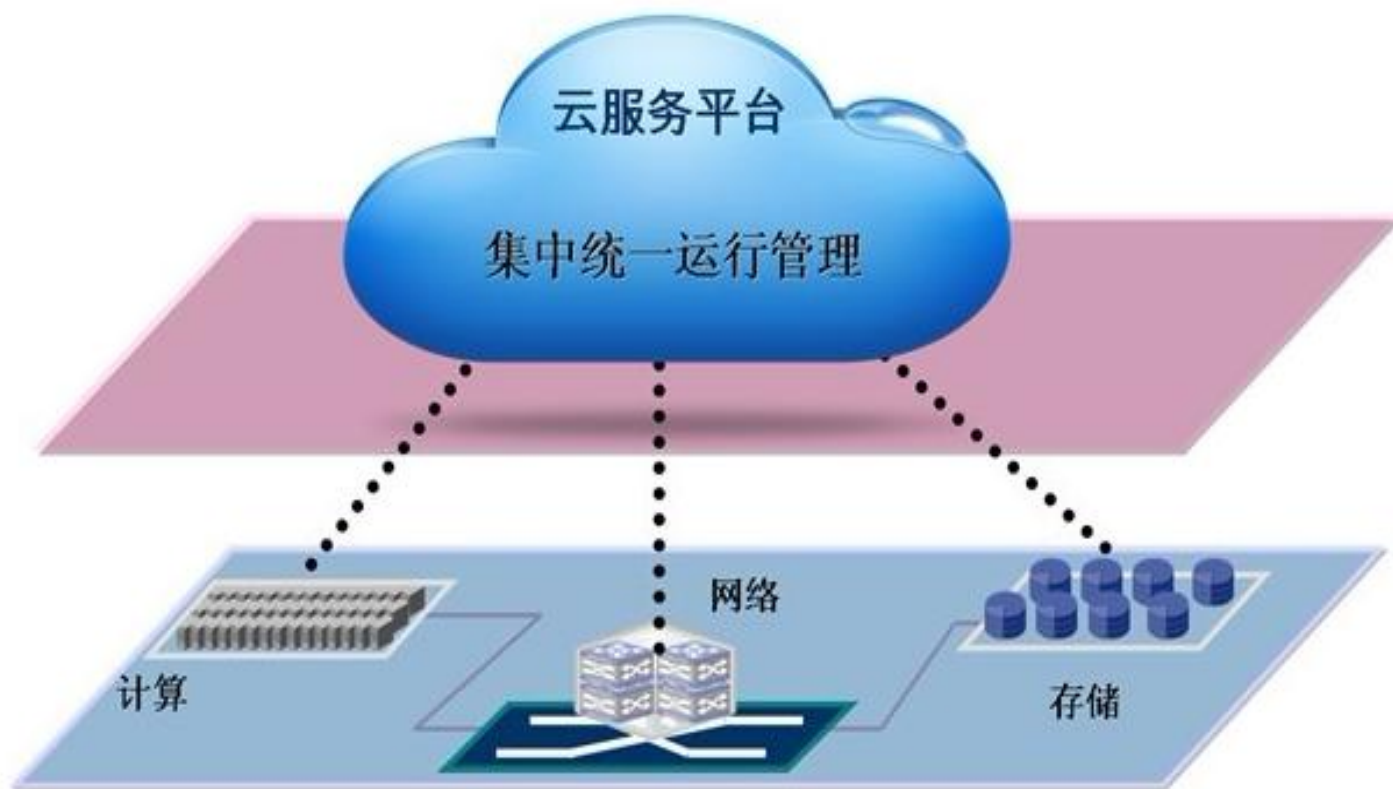
- 因此, 分布式数据库是要考虑在具有多处理器并可能是自治、异构情况下的数据管理
  - 受影响的内容有:
    - 数据组织
    - 查询处理
    - 存取结构
    - 并发控制
    - 恢复处理

# 事务处理

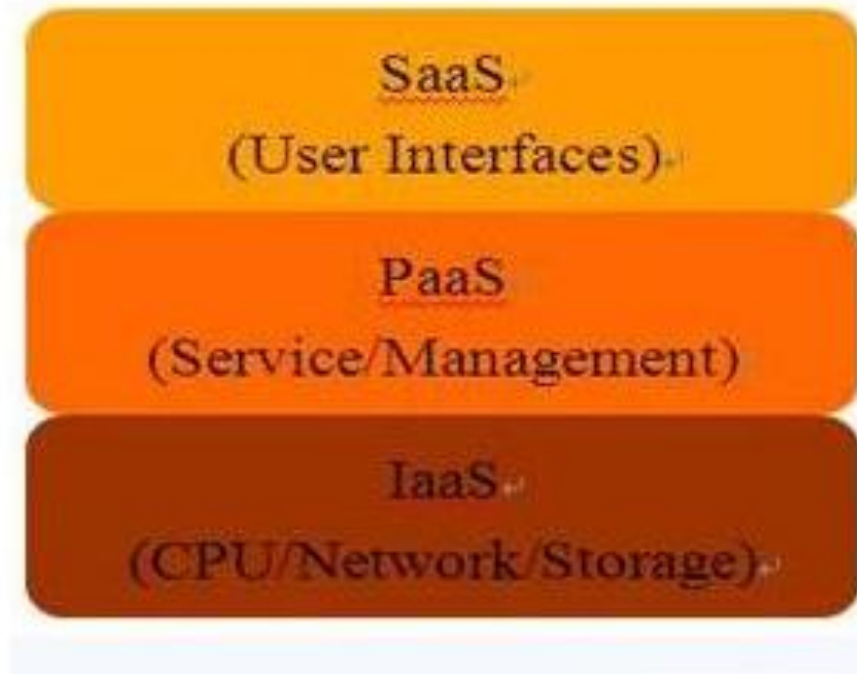
- 事务模型
- 分布事务处理
  - 协调事务执行
    - 多 DBMSs
    - 高性能
- 嵌套事务和长事务
- 并发与恢复

# 其他

- 并行处理
- 分布式计算（如：云）
  - 大数据
  - Hadoop
  - MapReduce
  - ○ ○ ○ ○



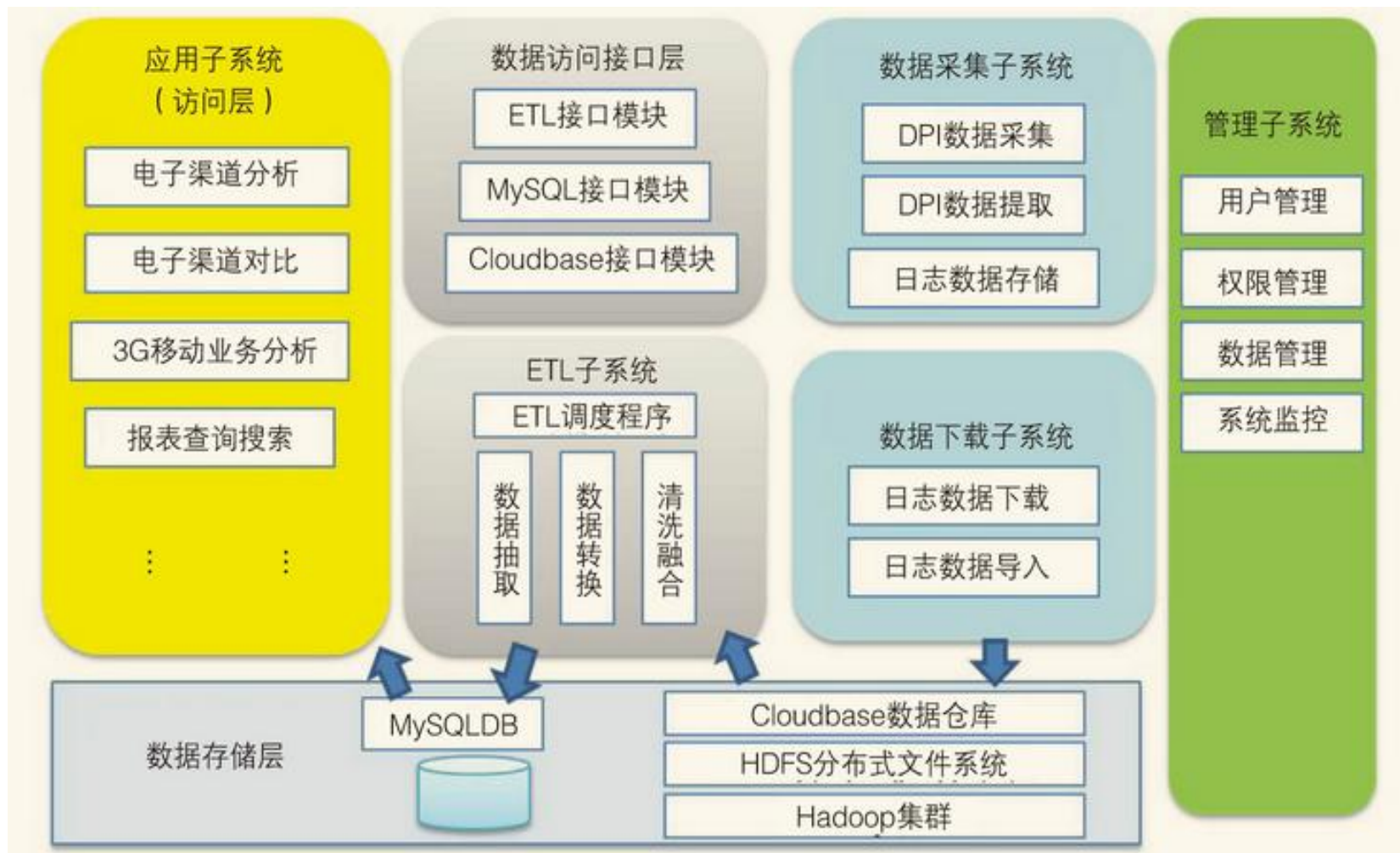
# 架构



# 微软云计算架构









SaaS

分布式数据挖掘

如: Mahout

PaaS

分布式处理

如: MapReduce, JobKeeper

分布式数据库

如: HBase, 数据立方

IaaS

云存储

虚拟化

如: HDFS, cStor

如: VMware, OpenStack



# 课程内容

- 分布式数据库
- 面向对象DB

# 参考教材

- <分布式数据库系统及其应用> 科学出版社  
2011年 邵佩英 第三版
- <分布式数据库系统> 机械出版社 2010年
- <面向对象数据库系统及其应用>科学出版社  
2003年 徐洁磐
- <面向对象数据库管理> ALfons Kemper &  
Guido Moerkotte Prentice Hall 1994 清华大学出  
版社影印 (复印)
- M. Tamer Ozsü and Patrick Valduriez, “Principles  
of Distributed Database Systems,” Second Edition,  
Prentice Hall 1999. 清华大学出版社影印
- 参考文献

- <http://202.38.64.11/~llyue/adddb.html>
- 成绩评定 (参考指标)
  - 课后作业 (15%)
    - ppt中给出
  - 上机作业 (20%)
  - 期末考试 (65%)

# 上机作业

- 面向对象数据库系统的应用
- Versant 系统

# 上机作业

## Travel Reservation System

