

第一章 数据库系统介绍

主要内容

- 1.1 数据库系统应用
- 1.2 数据库与文件系统
- 1.3 数据视图
- 1.4 数据模型
- 1.5 数据库语言
- 1.6 数据库用户和管理员
- 1.7 事务管理
- 1.8 数据库系统结构
- 1.9 应用程序体系结构
- 1.10 数据库系统历史

本章需要掌握的内容

□ 关键概念

- 数据库
- 数据库管理系统
- 数据库系统
- 数据模型三要素及作用

□ 目的和意义

- 数据库技术的作用

□ 应用场景

- 数据库技术的应用场景

1.1 数据库的应用

□ 数据库应用：

- 银行业：交易系统
- 航空业：订票系统
- 大学：招生系统，学生管理系统
- 销售业：客户管理，产品管理
- 在线零售：订单管理
- 制造业：产供销管理
- 人力资源：考勤管理

□ 数据库涉及我们生活的所有方面！

课堂讨论1:

- 你能区分以下四个术语吗?
 - 数据（记录）
 - 数据库
 - 数据库管理系统
 - 数据库系统

数据库技术中常用术语

- Data: 数据
- Databse(DB): 数据库
 - Collection of interrelated data
- Database Management System(DBMS)
 - 数据库管理系统
- Database system(DBS) 数据库系统
 - 相关数据的集合
 - 访问数据的程序的集合
 - 一个方便和高效的使用环境

Data(数据, 记录)

□ 描述某一对象的相关数据集合

如：描述一个学生（姓名，学号，班级，专业）

Database (数据库)

- 一个庞大的相关联的数据集合
- 描述真实企业的数据模型 (e.g., 一个大学数据库)

SID	Name	Age	GPA
546007	Peter	18	5.8
546100	Bob	19	3.65
546500	Bill	20	6.7

Students

CID	CName
CS242	DB
CS207	SW
CS369	OS

Courses

SID	CID	Grade
546007	CS242	6
546007	CS369	5
546100	CS242	7

Enrollment

DBMS: 数据库管理系统

- *Database Management System (DBMS):*

- 系统软件，用于创建、管理和维护数据库

- 常见的DBMS:

- Oracle, MySQL, MS SQL SERVER, ...

- 应用于:

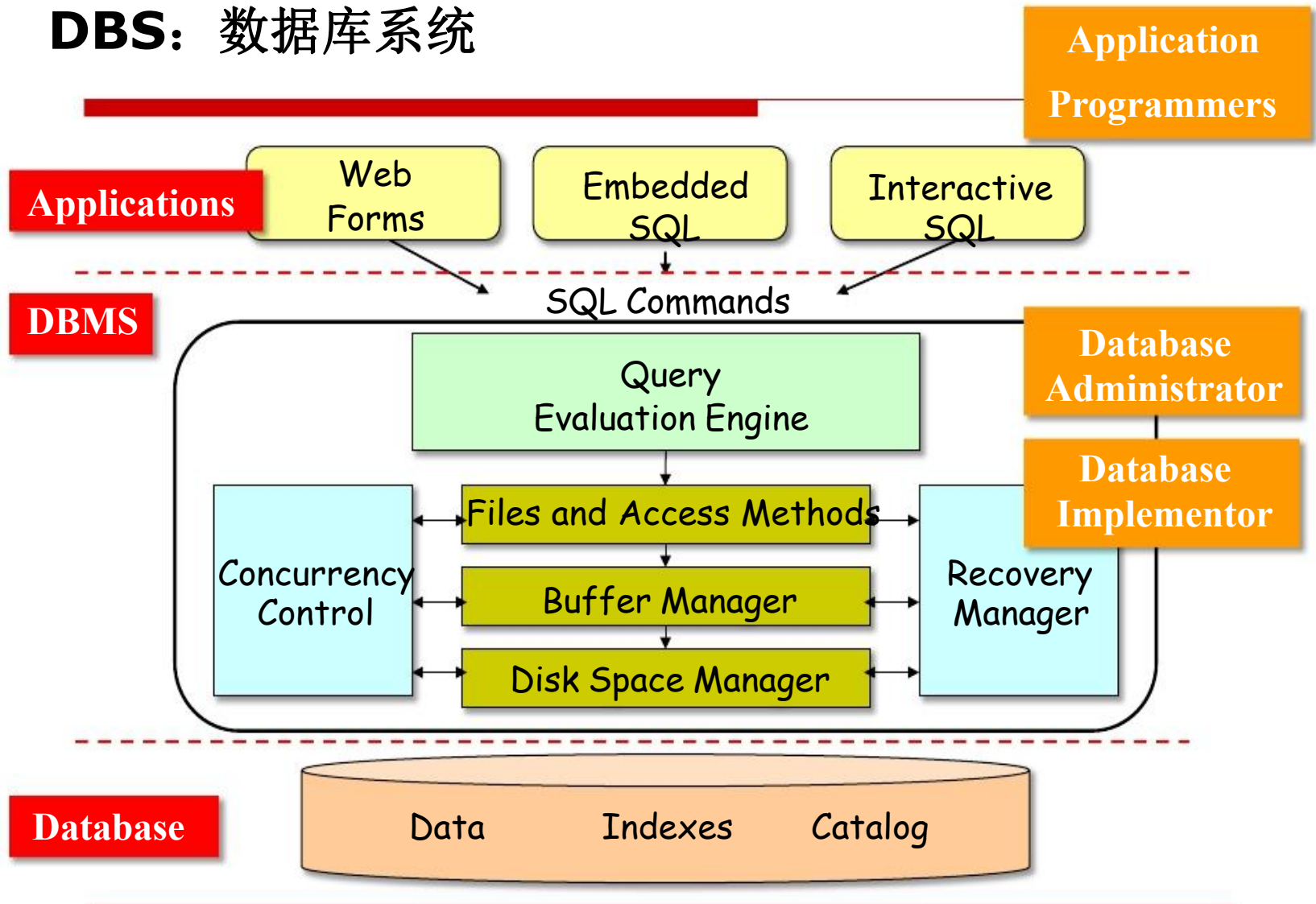
Database System = DB + DBMS + Application Logic

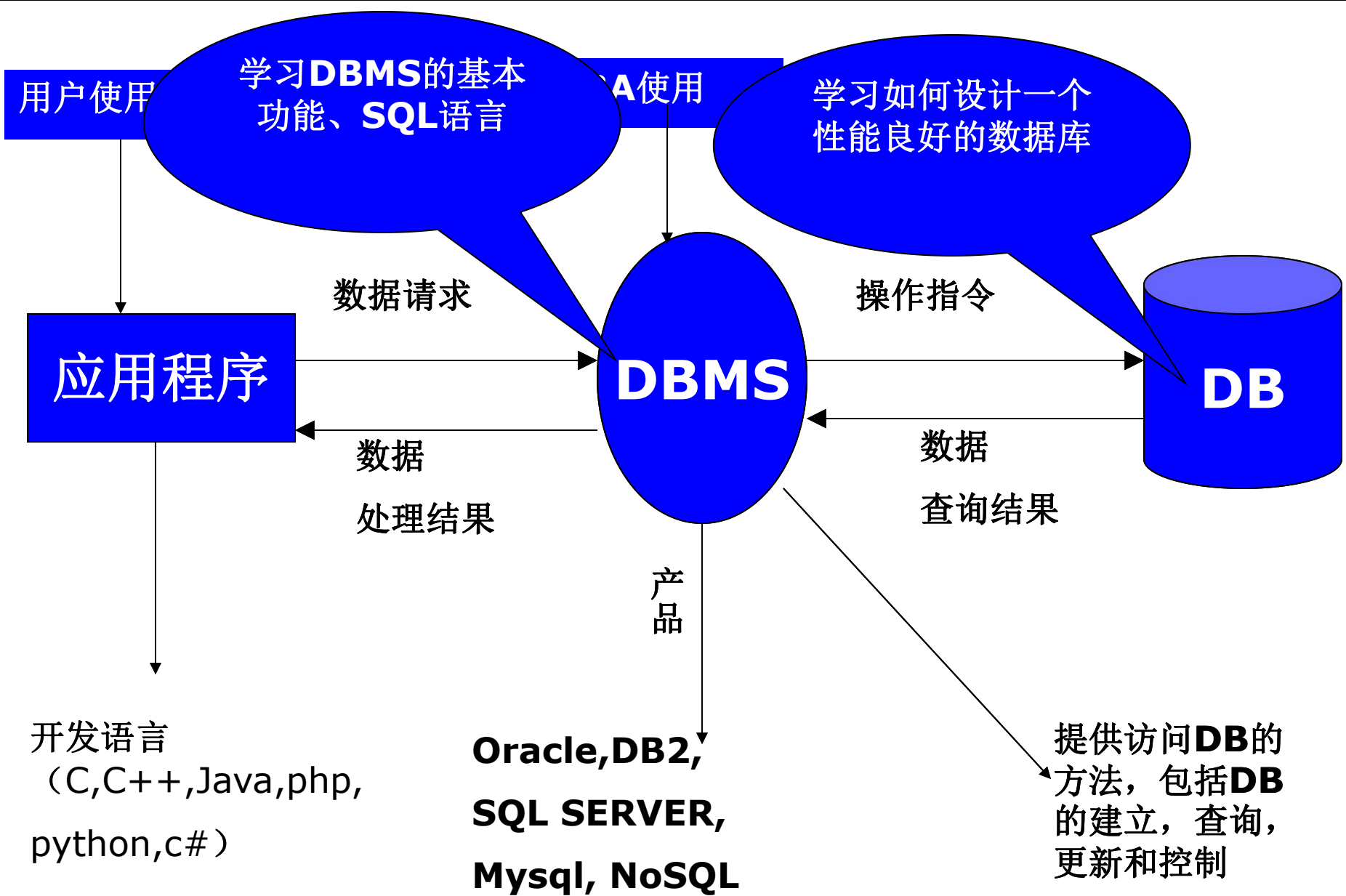
数据库管理系统软件市场占有率排名

■ 访问以下链接：

□ <http://db-engines.com/en/ranking>

DBS: 数据库系统





教材案例：大学数据库

□ 大学管理系统的功能

- 对学生、教师、课程等基本信息的管理
- 学生选修课程、教师教授课程等
- 计算学生的成绩和绩点等

课堂实验：认识数据库

□ ch1-实验"指导手册

- 实验1

- 实验2

1.2 数据库系统的目的

- 早期数据库应用直接建立在文件系统上。

课堂讨论2:

用**Execl**表格、**txt**文件等方式存储数据，有什么优缺点？

正常使用主观题需2.0以上版本雨课堂

郝老师

作答

16

使用文件系统存储数据的弊端：

- 数据冗余和不一致
 - 多文件格式，不同文件中信息重复
- 访问数据困难
 - 对每个新的任务都需要写新的程序
- 数据孤立—多文件多格式
- 完整性问题
 - 约束靠代码实现
 - 难于添加或改变现有约束

使用文件系统的弊端 (cont.)

□ 更新的原子性

- 失败会使数据库处于只执行部分更新的不一致状态
- 如: 银行转账要么完成, 要么根本没有发生

□ 多用户并发访问

- 为了性能需要并发访问

- 不加控制的并发访问会导致不一致性

如：两人同时读余额100，并同时通过各自提现50，更新余额

□ 安全问题

- 难于对某些而不是全部数据提供用户访问

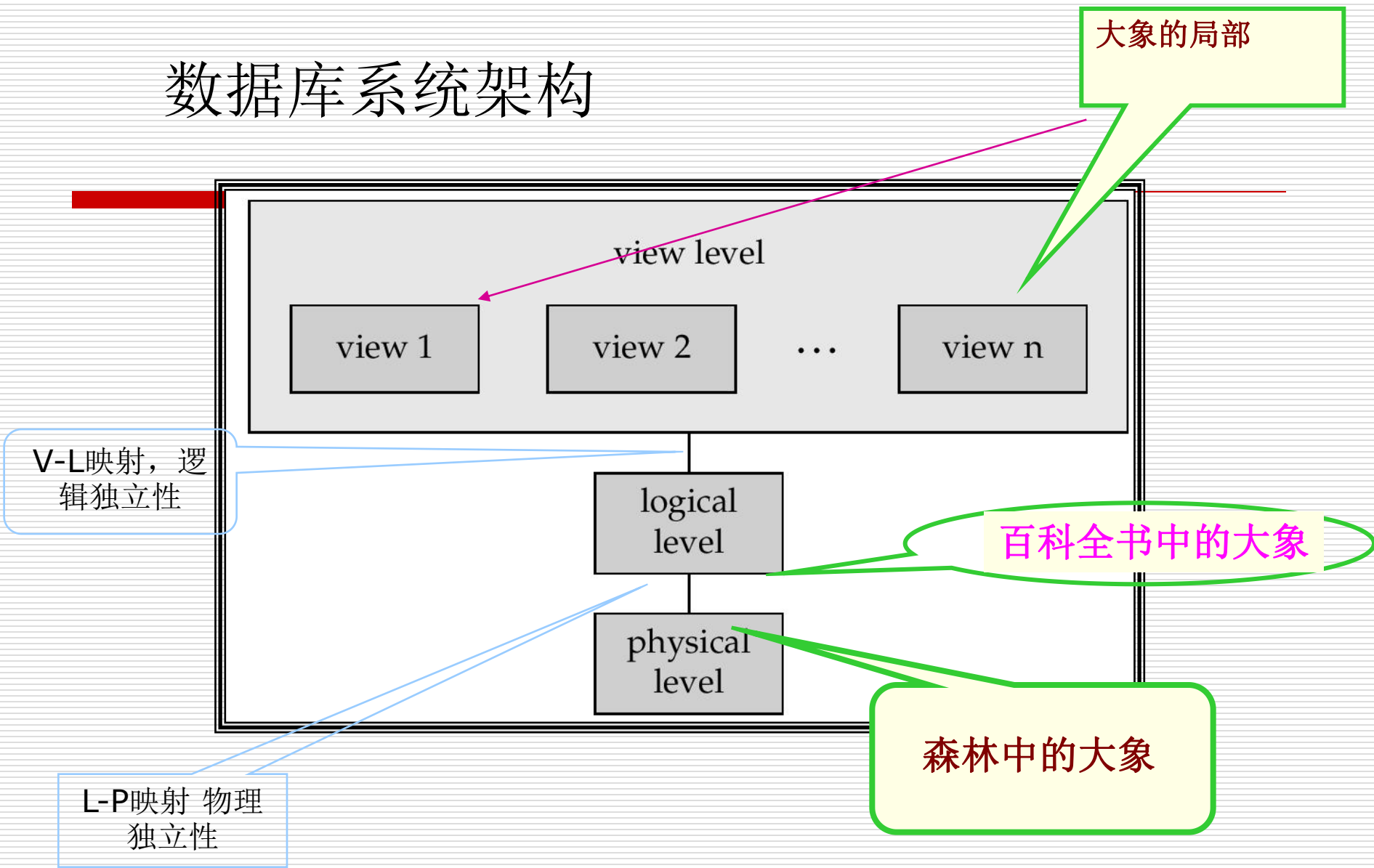
□ 数据库系统提供上述所有问题的解决方案

1.3 数据视图

□ 数据抽象

- 物理层Physical level: 描述一个记录（如客户）
如何被存储
- 逻辑层Logical level: 描述数据库中存储的数据和
数据间的关系
- 视图层View level:
应用程序隐藏数据类型的细节，视图也可以为了安全目的隐藏信息（比如雇员的薪水）

数据库系统架构

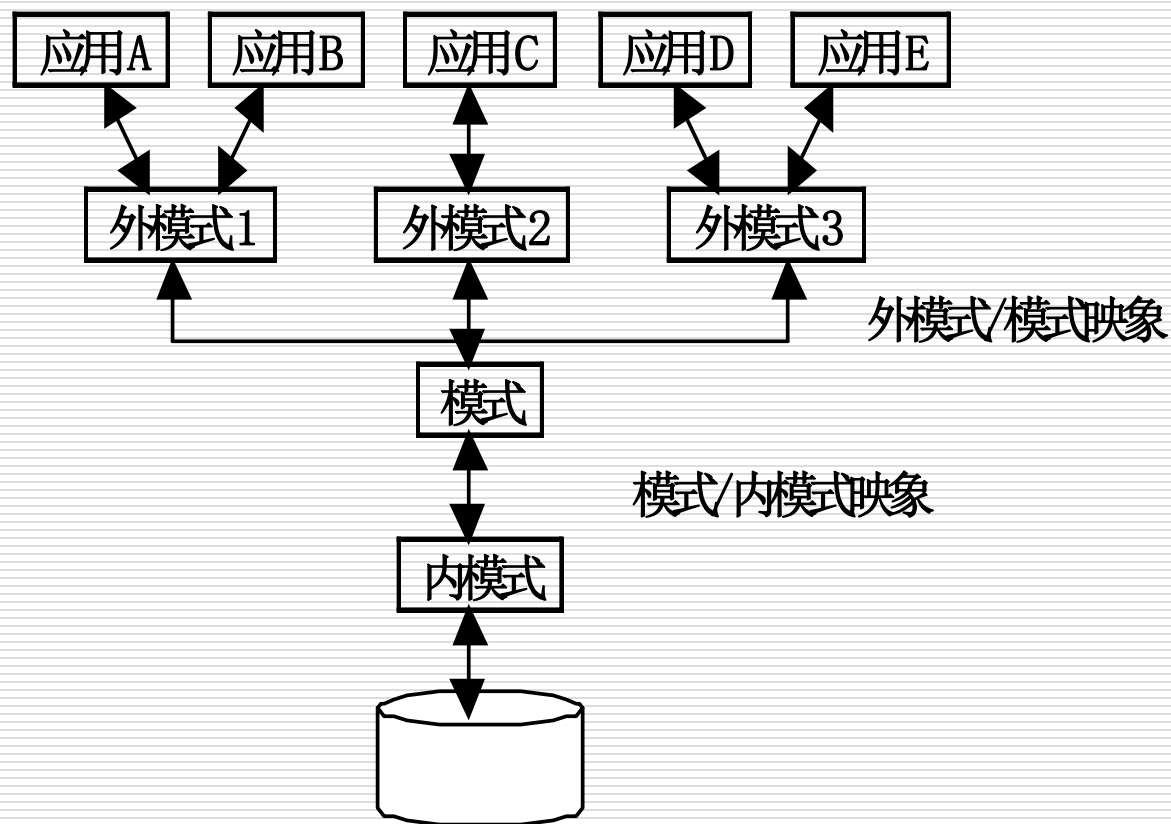


课堂实验：数据抽象

□ ch1-实验"指导手册

■ 实验3

数据库系统的三级模式结构(在很多中文教材中)



课堂讨论3:

数据库系统的三级模式、两层映像结构有什么作用?

正常使用主观题需2.0以上版本雨课堂

郝老师

作答

24

1.3.2 实例与模式

- 类似于程序设计语言中的类型和变量
- **模式 Schema** - 数据库的逻辑结构
 - 物理模式 Physical schema: 物理层的数据库设计
 - 逻辑模式 Logical schema: 逻辑层的数据库设计
- **实例 Instance** - 某个时间点时数据库的实际内容
- **物理数据独立性 Physical Data Independence** -
 - 应用依赖于逻辑模式

1.3.3 数据模型

□ 数据模型是描述下列内容的工具集合：

- 数据 Data
- 数据关系 Data relationships
- 数据语义 Data semantics
- 数据约束 Data constraints

经典的数据模型

- ❑ 关系模型：Relational model
- ❑ 实体联系数据模型：Entity-Relationship data model
(主要是用于数据库设计)
- ❑ 基于对象的数据模型：Object-based data models (面向对象的和对象关系的)
- ❑ 半结构化的数据模型：Semistructured data model (XML)
- ❑ 其他较老的模型：
 - 网络模型Network model
 - 层次模型Hierarchical model

课堂讨论4:

数据模型有什么用?

正常使用主观题需2.0以上版本雨课堂

郝老师

作答

28

数据模型三要素

- 数据结构
 - 数据的组织方式
- 数据操作
 - 如何读取数据
- 数据约束
 - 如何保障数据的正确性

一个简单的数据库：

- 数据结构
 - 一个关系表
- 数据操作
 - 增删查改
- 数据约束
 - 完整性约束

<i>ID</i>	<i>name</i>	<i>dept_name</i>	<i>salary</i>
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

(a) The *instructor* table

<i>dept_name</i>	<i>building</i>	<i>budget</i>
Comp. Sci.	Taylor	100000
Biology	Watson	90000
Elec. Eng.	Taylor	85000
Music	Packard	80000
Finance	Painter	120000
History	Painter	50000
Physics	Watson	70000

(b) The *department* table

小结：模型的分类



1.4 数据库语言

- Data-definition languages(数据定义)
 - Specify the database schema
 - Define Integrity Constraints
 - Example: unique SIDs
 - More...
- Data-manipulation languages(数据操纵)
 - Express database queries and updates
 - To ask questions = Query
 - Example: Which students have GPA > 3.75?
 - To insert, delete and update data

Data Definition Language (DDL)数据定义语言

□ E.g. 建表

```
create table account (  
    account-number char(10),  
    balance integer)
```

DDL例子

□ DDL编译器生成一个表的集合并存储在数据字典中。

-
- 数据字典包含元数据。(i.e.,关于数据的数据)
 - 数据库模式
 - 数据存储和定义语言
 - 指定存储结构和使用的访问方法
 - 完整性约束Integrity constraints
 - 主键Primary key (ID uniquely identifies instructors)
 - 域约束Domain constraints
 - 参照完整性Referential integrity (**references** constraint in SQL)
 - e.g. *dept_name* value in any *instructor* tuple must appear in *department* relation
 - 断言Assertion
 - 授权Authorization

Data Manipulation Language (DML)

- 查 插 删 改
- 访问和操纵那些按某种合适的数据模型组织起来的数据的语言。
 - DML也是查询语言
- 两类语言
 - 过程化的Procedural - 用户指出需要什么数据和如何获得那些数据
 - 陈述式Declarative (nonprocedural) - 用户指出需要什么数据但不需要指定如何获得那些数据
- SQL是使用最广泛的查询语言。

1.5 关系型数据库

□ 一个关系型数据库

- 是建立在关系模型上的
- 使用多个表的集合来表示数据及数据之间的关系

□ 例

属性
Attributes

<i>Customer-id</i>	<i>customer-name</i>	<i>customer-street</i>	<i>customer-city</i>	<i>account-number</i>
192-83-7465	Johnson	Alma	Palo Alto	A-101
019-28-3746	Smith	North	Rye	A-215
192-83-7465	Johnson	Alma	Palo Alto	A-201
321-12-3123	Jones	Main	Harrison	A-217
019-28-3746	Smith	North	Rye	A-201

顾客号

顾客名

街道

城市

账号

<i>customer-id</i>	<i>customer-name</i>	<i>customer-street</i>	<i>customer-city</i>
192-83-7465	Johnson	12 Alma St.	Palo Alto
019-28-3746	Smith	4 North St.	Rye
677-89-9011	Hayes	3 Main St.	Harrison
182-73-6091	Turner	123 Putnam Ave.	Stamford
321-12-3123	Jones	100 Main St.	Harrison
336-66-9999	Lindsay	175 Park Ave.	Pittsfield
019-28-3746	Smith	72 North St.	Rye

(a) The *customer* table

顾客表

账号

<i>account-number</i>	<i>balance</i>
A-101	500
A-215	700
A-102	400
A-305	350
A-201	900
A-217	750
A-222	700

(b) The *account* table

<i>customer-id</i>	<i>account-number</i>
192-83-7465	A-101
192-83-7465	A-201
019-28-3746	A-215
677-89-9011	A-102
182-73-6091	A-305
321-12-3123	A-217
336-66-9999	A-222
019-28-3746	A-201

(c) The *depositor* table

储户

1.6 Database Design

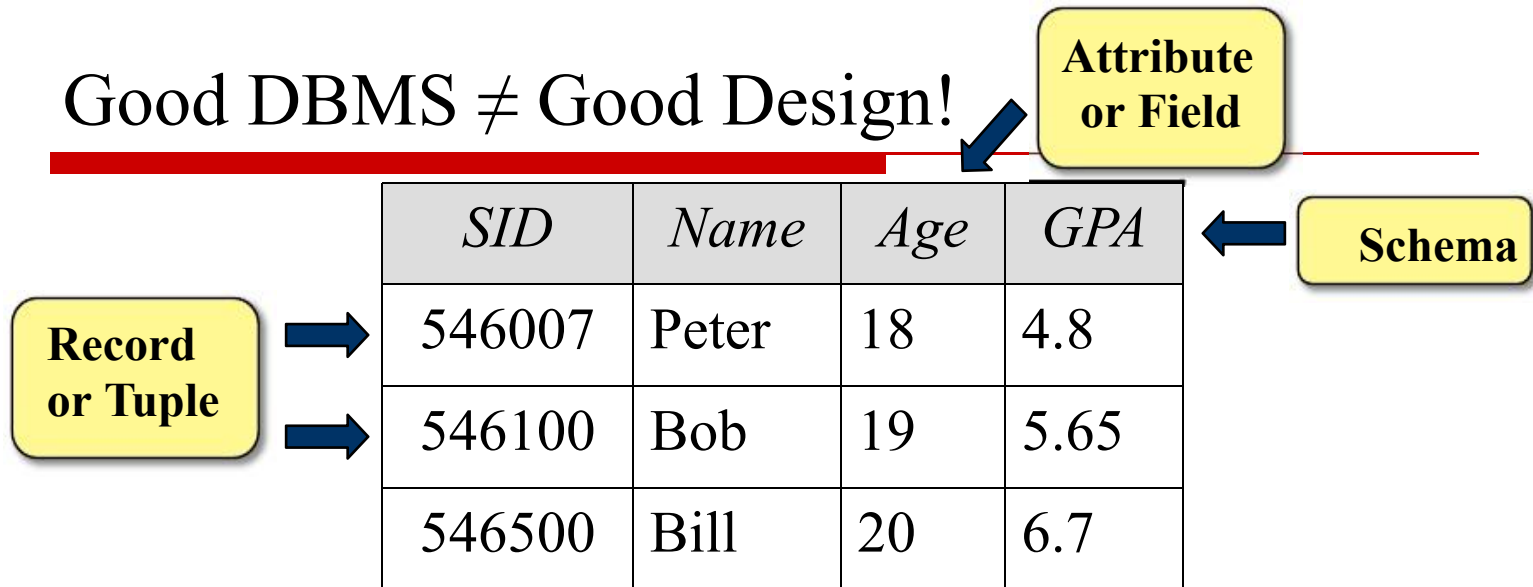
设计数据库的结构的过程

逻辑设计Logical Design – 决定数据库的模式。数据库设计要求我们找到一个好的关系模式集合。

- 业务决定Business decision – 数据库中应该记录什么属性？
- 计算机科学决策Computer Science decision – 我们应该有什么关系模式以及属性应该如何在关系模式间分布？

物理设计Physical Design – 决定数据库的物理布局
physical layout

Good DBMS \neq Good Design!



□ Schema:

■ Students (*ssn*: string, *name*: string, *age*: integer, *gpa*: real)

■ Alternative Schema:

□ Students (*ssn*: integer, *fname*: string, *lname*: string, *dob*: date, *gpa*: real)

1.6 数据库设计

- ❑ The process of designing the general structure of the database
- ❑ 四个层次上的考虑::
 - ❑ Logical Design
 - ❑ 设计好的模式
 - ❑ Business decision
 - ❑ 用户需要存什么（关系,字段）
 - ❑ Computer Science decision
 - ❑ 怎样在计算机中满足用户需求
 - ❑ Physical Design
 - ❑ Deciding on the physical layout of the database

□ Is there any problem with this design?

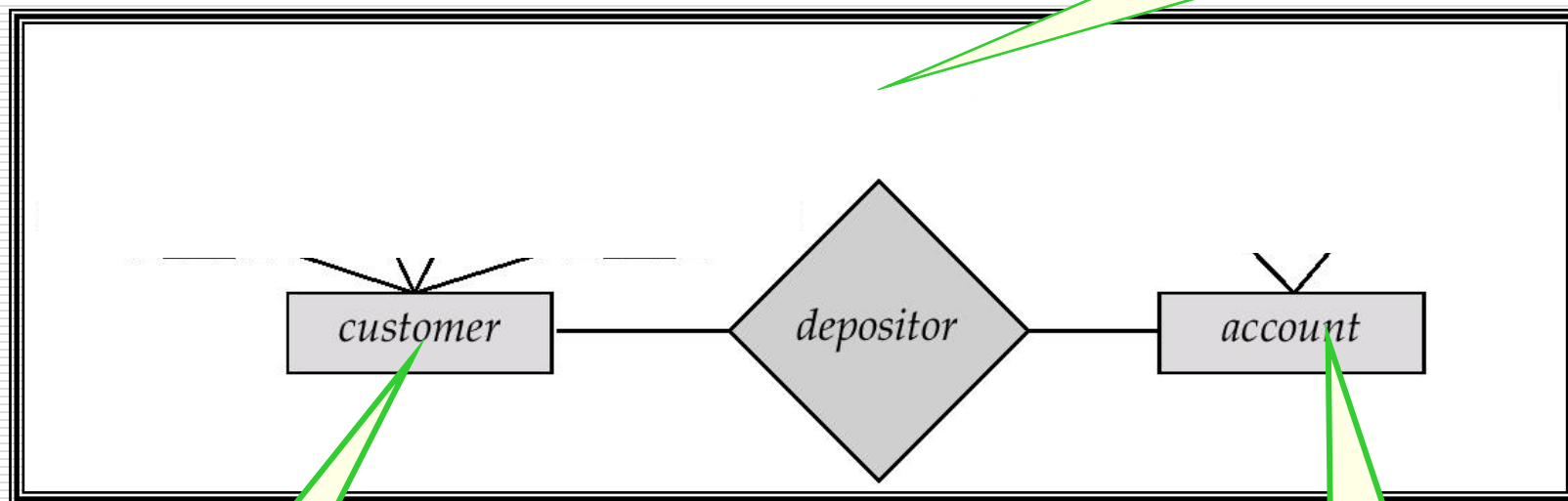
<i>ID</i>	<i>name</i>	<i>salary</i>	<i>dept_name</i>	<i>building</i>	<i>budget</i>
22222	Einstein	95000	Physics	Watson	70000
12121	Wu	90000	Finance	Painter	120000
32343	El Said	60000	History	Painter	50000
45565	Katz	75000	Comp. Sci.	Taylor	100000
98345	Kim	80000	Elec. Eng.	Taylor	85000
76766	Crick	72000	Biology	Watson	90000
10101	Srinivasan	65000	Comp. Sci.	Taylor	100000
58583	Califieri	62000	History	Painter	50000
83821	Brandt	92000	Comp. Sci	Taylor	100000
15151	Mozart	40000	Music	Packard	80000
33456	Gold	87000	Physics	Watson	70000
76543	Singh	80000	Finance	Painter	120000

-
- Normalization Theory (Chapter 8)
 - Formalize what designs are bad, and test for them
 - Entity Relationship Model (Chapter 7)
 - Models an enterprise as a collection of *entities* and *relationships*
 - Entity: a “thing” or “object” in the enterprise that is distinguishable from other objects
 - Described by a set of *attributes*
 - Relationship: an association among several entities
 - Represented diagrammatically by an *entity-relationship*

Entity-Relationship Model 第7章详细讲

□ Example of schema in the entity-relationship model

技巧：先画实体和联系，最后加属性



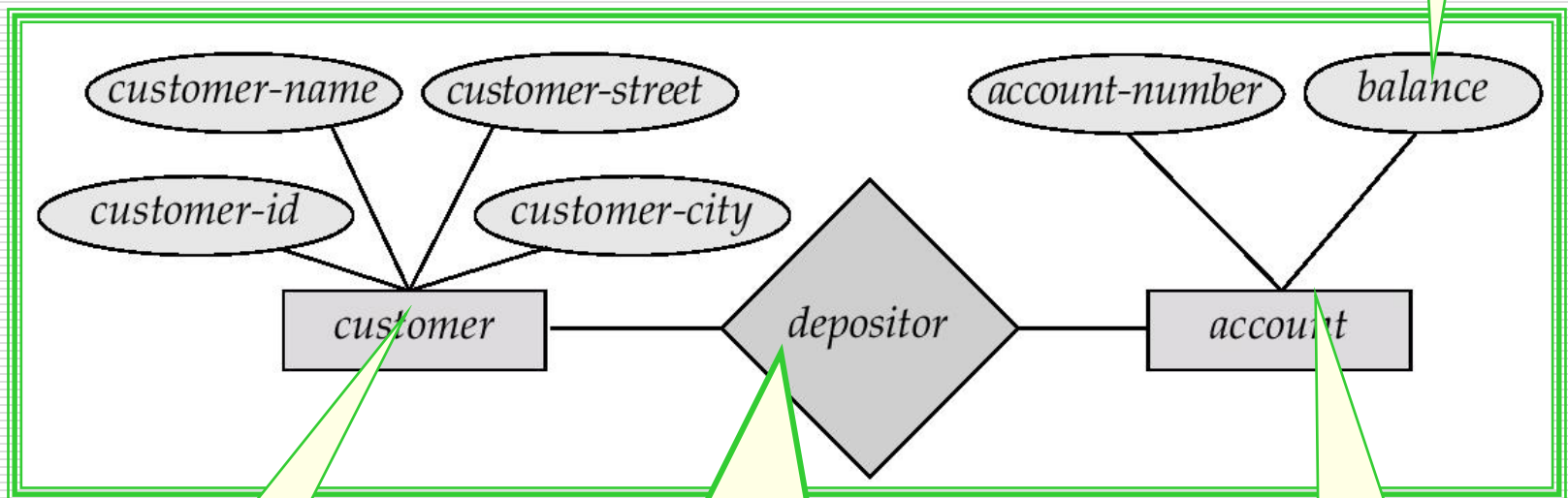
实体
储户

联系：存款
常用动词或动宾结构，有
联系两对象的自然语义

实体
账户

❑ Example of schema in the entity-relationship model

❑ 技巧：先画实体和联系，最后加属性



实体
储户

联系：存款
常用动词或动宾结构，有
联系两对象的自然语义

实体
账户

1.7 Storage Management 存储管理

- Storage manager is a program module that provides the interface between the low-level data stored in the database and the application programs and queries submitted to the database manager.

程序模块，管理员——模块
界面——底层管理
- The storage manager is responsible to the following tasks:
- 存储管理员任务
 - interaction with the file manager 与管理员交互
 - efficient storing, retrieving and updating of data
 - 高效存储，检索，更新

1.8 Transaction Management 事务管理

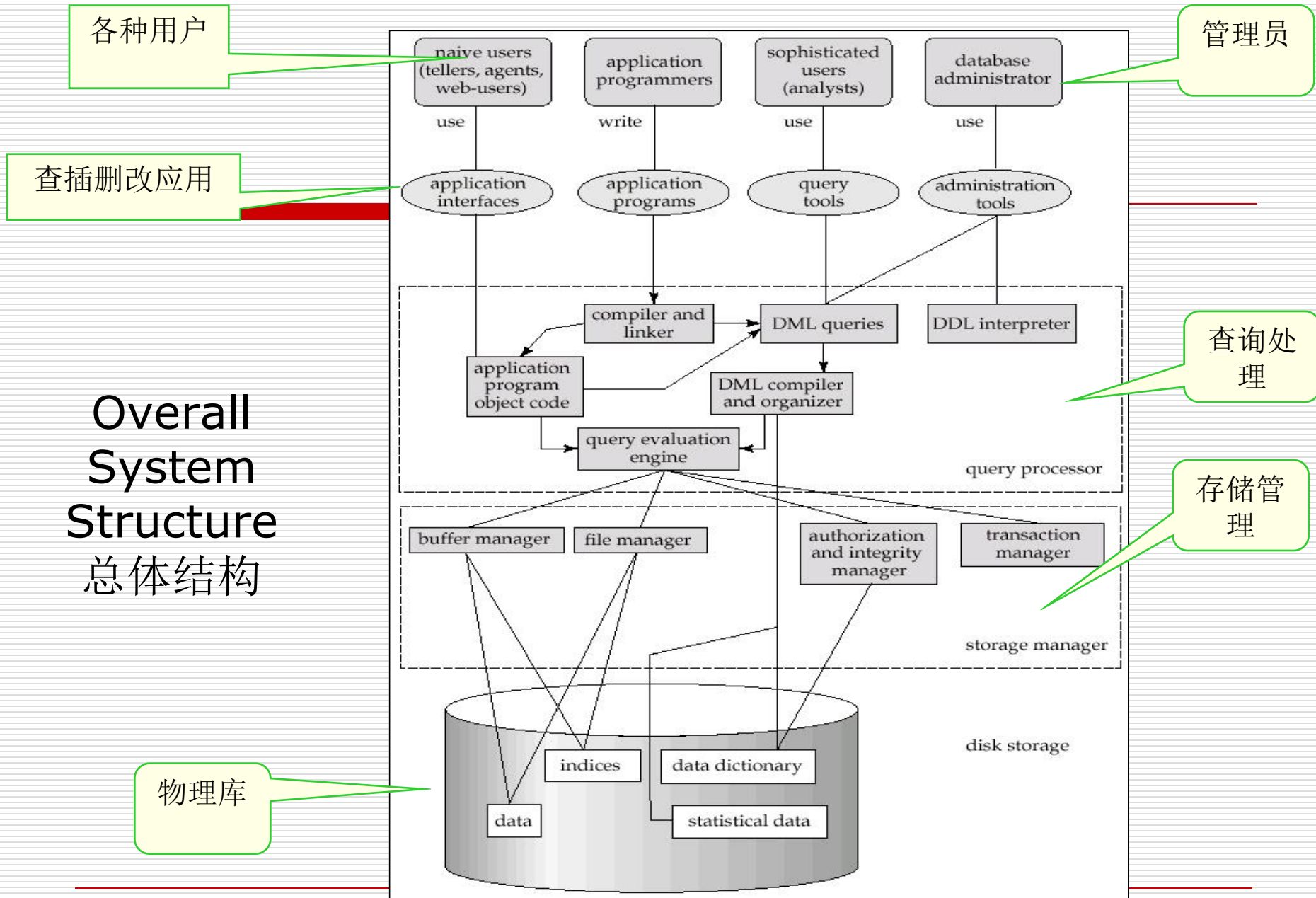
- ❑ 系统失效怎么办？What if the system fails?
- ❑ 如果不止一个用户并发更新相同数据怎么办？What if more than one user is concurrently updating the same data?
- ❑ **事务**是一个数据库应用程序中执行单一逻辑功能的操作集合。
- ❑ **事务管理组件Transaction-management component** 确保数据库不论是系统失败（电源故障或操作系统崩溃）还是事务失败都能保持在一个一致的（正确的）状态。
- ❑ **并发控制管理器Concurrency-control manager** 控制并发事务的交互以确保数据库的一致性。controls the interaction among the concurrent transactions, to ensure the consistency of the database.

1.9 数据库结构

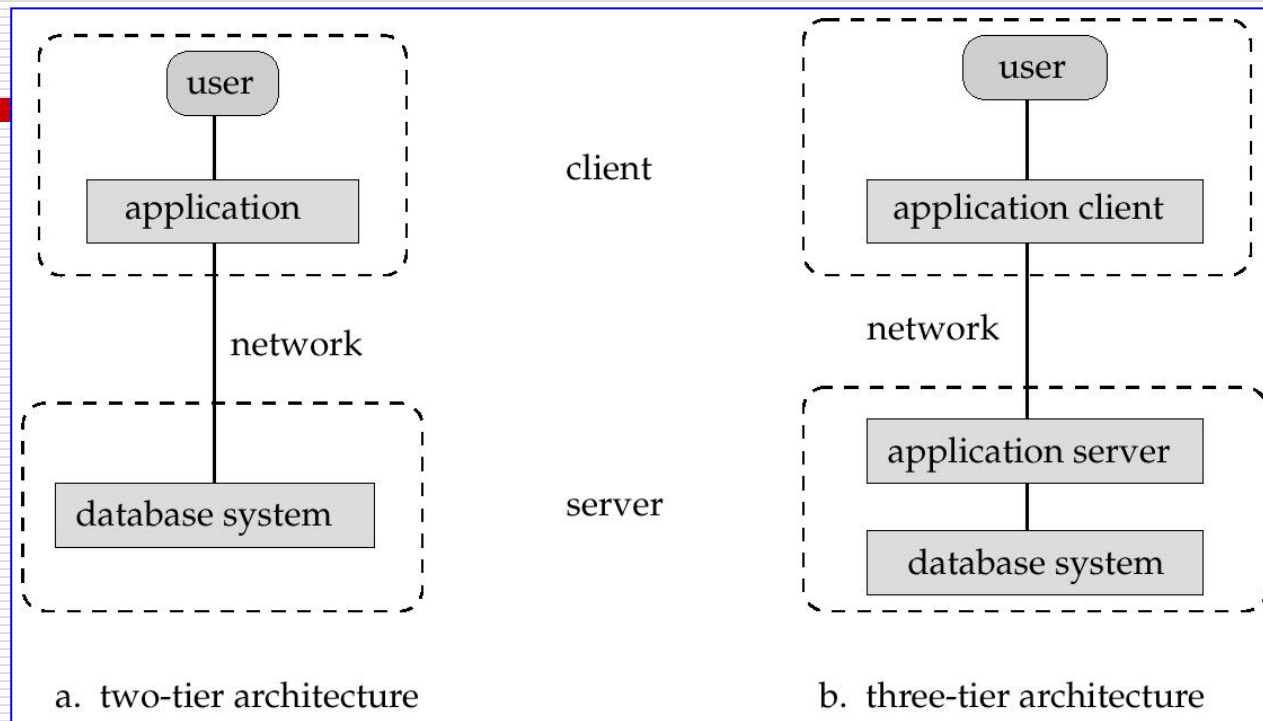
The architecture of a database systems is greatly influenced by the underlying computer system on which the database is running:

- Centralized
- Client-server
- Parallel (multi-processor)
- Distributed

Overall System Structure 总体结构



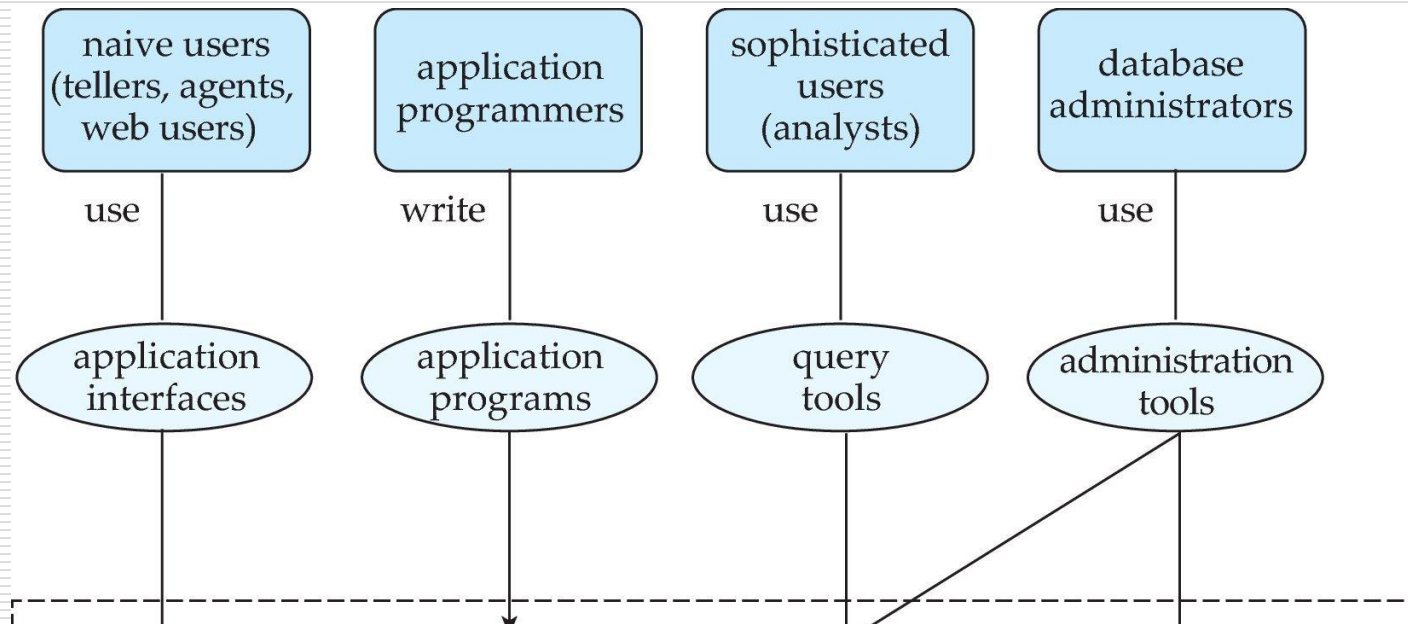
Application Architectures 应用体系



▪ **Two-tier architecture:** E.g. client programs using ODBC/JDBC to communicate with a database 两层结构

▪ **Three-tier architecture:** E.g. web-based applications, and applications built using "middleware" 三层结构

1.12 Database Users 用户



Database Users 用户

用户根据他们期望与系统交互方式的不同而不同

- 应用程序员 - 通过DML调用与系统交互
- 富有经验的用户Sophisticated users - 用数据库查询语言表达请求
- 专业用户 - 编写不适合于传统的数据处理框架的专业数据库应用
- 初级用户 - 通过调用先前已经写好的一个应用程序与系统交互。

Examples, people accessing database over the web, bank tellers

1.12 Database Administrator 数据库管理员

- 协调数据库系统的所有活动;DBA应该对企业信息资源和需求有很好的理解。
- DBA职责：
 - 模式定义Schema definition
 - 存储结构及存取方法定义
 - 模式和物理组织修改
 - 对用户访问数据库授权
 - 指定完整性约束
 - 与用户联系Acting as liaison with users
 - 监控性能和对需求变化做出响应

1.13 数据库系统历史

1950s and early 1960s:

磁带用于数据存储之数据处理：磁带只提供顺序访问。
穿孔卡片用于输入

Late 1960s and 1970s:

硬盘允许直接访问数据

广泛使用网络和层次数据模型

Ted Codd 定义关系数据模型

高性能事务处理High-performance (for the era)
transaction processing

-
- 1980s:
 - 研究性关系原型演化为商用系统
 - SQL 成为 industrial standard
 - Parallel 并行 and distributed 分布式 database systems
 - Object-oriented database systems
 - 1990s:
 - 大型决策支持和数据挖掘应用
 - 数据仓库
 - Emergence of Web commerce
 - Early 2000s:
 - XML and XQuery 标准
 - 自动数据库管理 Automated database administration
 - Later 2000s:
 - Giant data storage systems
 - Google BigTable, Yahoo PNuts, Amazon,
-

本章小结

- ❑ **DBMS**是一个互相关联的数据的集合和一组用以访问这些数据的程序组成。这些数据描述某个特定的企业。
DBMS目标是提供一个可以让人们方便地高效地存取信息的环境。
- ❑ 数据库系统用于存储大量的信息。对数据的管理涉及到信息存储结构的定义，信息操纵机制的提供，提供所存储数据的安全性保证（系统崩溃或用户越权时），并发多用户时避免产生异常结果。
- ❑ 数据库系统的主要目的之一是为用户提供数据的抽象视图，系统隐藏存储和维护的细节。

-
- 数据模型是描述下列内容的工具集合：
 - 数据Data
 - 数据关系Data relationships
 - 数据语义Data semantics
 - 数据约束Data constraints
 - E-R图提供了图形化的方式查看数据联系和约束。
 - 数据库定义语言定义数据库模式；数据操纵语言使用户可以访问和操纵数据。
 - 数据库体系结构应用程序体系结构见图。

本章重点内容

□ 1.2 数据库的目的

□ 1.3 数据视图

Any Question ?