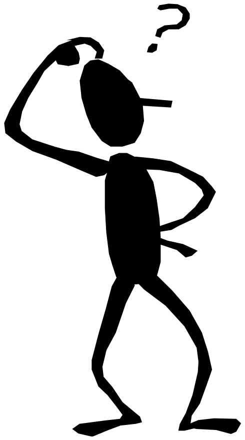
21121350
Database System

Lecture 1: Introduction

Lu Chen (陈璐)
College of Computer Science
Zhejiang University
Spring & Summer 2023
luchen@zju.edu.cn/18868818726

What Is a Database System?



- Database:
  - A very large, integrated collection of data
- Models a real-world enterprise
  - Entities (e.g., teams, companies)
  - Relationships (e.g., The Patriots is playing in The Superbowl)
  - More recently, also includes active components (e.g., business logic)
- A Database Management System (DBMS) is a software system designed to store, manage, and facilitate access to databases.

Lecturer Information

- **Name:** Lu Chen (陈璐), ZJU Plan-100 Professor (PhD Supervisor)

- **Office:** Room 402, Zetong Building, Yuquan Campus,
  Zhejiang University

- **E-mail:** luchen@zju.edu.cn

- **Mobile:** (86) 188-6881-8726

- **Homepage:** https://person.zju.edu.cn/luchen

Lecturer Research Interests

- Metric data management
- Spatial/Spatio-temporal databases
- Indexing and query optimization
- Database usability
- Big data Analytics
- AI interaction with DB technology
- Data Integration and Data Quality

- Everything that looks interesting (mostly database related), especially in the indexing and query processing for various static/moving multi-source heterogeneous data.

TA Information

- **Name:** Danlei Hu (胡丹蕾), Ph.D. Student
- **Mobile:** (86) 188-0115-8711
- **E-mail:** dlhu@zju.edu.cn

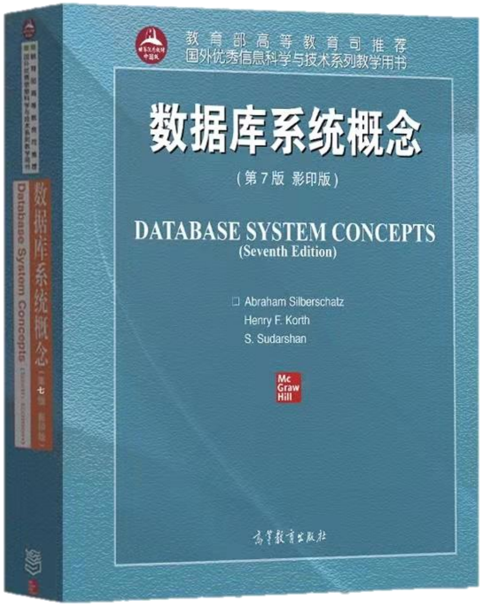
- **PracticeTime:** Fri. 10:00-11:35 (every week)
- **Practice Place:** Computer Center, Zijingang Campus,
  Zhejiang University

Course Textbook, Contents, and References

- Abraham Silberschatz, Henry F. Korth, and S. Sudarshan. Database System Concepts (7th Edition), 高等教学出版社, 2021.
  - Chapters 1, 2, 3, 4, 5, 6, 7, 12, 13, 14, 15, 16, 17, 18, 19

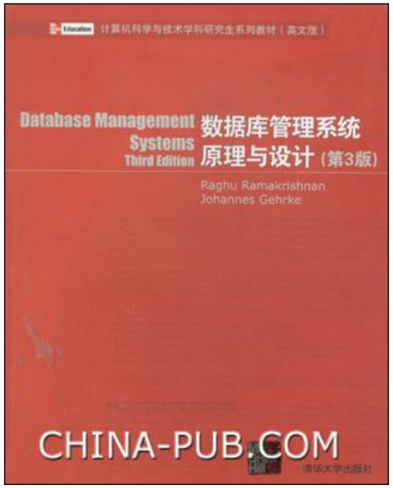
Time: Wednesday. 10:00-12:25

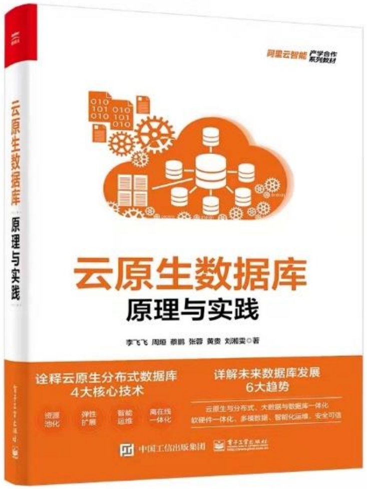
Place: West 1-504, Zijingang Campus, Zhejiang University

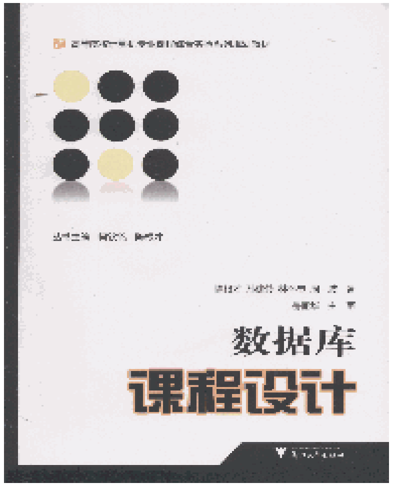


Course Textbook, Contents, and References

- Raghu Ramakrishnan and Johannes Gehrke. Database Management Systems (3rd Edition), 清华大学出版社, 2003.
- 李飞飞, 周烜, 蔡鹏, 张蓉, 黄贵.云原生数据库原理与实践, 中国工信出版集团, 2022. (实验参考书)
- 陈根才, 孙建伶, 林怀中, 周波. 数据库课程设计, 浙江大学出版社, 2007. (实验参考书)


Database Management Systems Third Edition
数据库管理系统原理与设计（第3版）
Raghu Ramakrishnan
Johannes Gehrke
CHINA-PUB.COM
清华大学出版社


云原生数据库
原理与实践
李飞飞 周烜 蔡鹏 张蓉 黄贵 刘雨奕 ◎著
诠释云原生分布式数据库 4大核心技术
详解未来数据库发展 6大趋势
中国工信出版集团



Course Practice/Experiment

- Practice/Experiments (Pending)
  - Place: Computer Center, Zijingang Campus, Zhejiang University
  - IP: 10.214.6.33
  - User: db70 – db79   user700 – user799
  - Password of sa: sa
  - Password of user: userXXX, e.g., user700 for user700

- Projects
  - 数据库系统实验要求 (Available at 学在浙大)
  - 图书管理系统实验要求 (Available at 学在浙大)
  - MINI SQL系统设计与实现 (Available at 学在浙大)

  - Allow selecting other similar systems such as 在线书店管理系统、超市销售管理系统、学生成绩管理系统、人事管理系统、设备管理系统, etc (requirements are similar to图书管理系统实验指导书).

Examination and Grading Policy

- Examination
  - Close book test, but allow taking one A4 page note.

- Grading Policy

Assignments/Exercises	10%
Course quizs	10%
Experiments and Experimental project	30%
Final examination	50%

- The course achievement fails, if the score of final examination is lower than 40.

Miscellaneous

- Course on 学在浙大 (https://courses.zju.edu.cn/course/53464)
  - Lecture slides will be posted shortly before or after the lecture.
  - Many issues discussed in the lectures will be covered in the examinations and assignments.
  - – Try to attend lectures regularly!

- Assignment
  - Paper-based + some programming.
  - Will be collected at the end of class on the due date.
  - Homework handed in by the due time will be graded for full credit, then a 30% late charge will be applied.
  - No late homework is accepted after on-time papers are returned, or the sample solution is made public, whichever occurs first.

- Welcome to discuss with me about any database/research problem.

Outline

- Purpose of Database Systems
  - Database Applications
  - Several Concepts
  - Database System vs. File-Processing System
- View of Data
- Database Language
- Database Design
- Database Users and Administrators
- Transaction Management
- Database Architecture
- History of Database Systems
- Summary

Database Applications

- Data processing and management are the most important fields of computer applications.
  - Knowledge of database systems is essential for computer scientists.

- Database Applications
  - Banking: all transactions
  - Airlines: reservations, schedules
  - Universities: registration, grades
  - Sales: customers, products, purchases
  - Manufacturing: production, inventory, orders, supply chain
  - Human resources: employee records, salaries, tax deductions
  - ……

Of course, you may think out a novel (potential) database application that would make you find/obtain a job/business.

- Databases touch all aspects of our lives although you don't see them.
- Could you imagine a life that is without databases?

Why Study Databases?

- Modeling and design of databases
  - Get/Abstract data models from the real world, and then translate them into the forms suitable for the target Database Management System (DBMS) --- tables, views.

Student Achievement

学号	姓名	专业	DB平时	DB期末	DB总评成绩	OS平时	OS期末	OS总评成绩
3023001093	黄毅照	混合班	85	95	90	…	…	85
3011112340	周朝威	计算机科学与技	80	90	85	…	…	88
3020621034	徐鑫	计算机科学与技	90	90	90			85
3020831035	薄延嵩	计算机科学与技	70	80	75			90
3021131123	胡俊	计算机科学与技	70	70	70			75
3022112002	蒋永丽	计算机科学与技	80	90	85			80
3022112003	顾娉娉	计算机科学与技	90	90	90	…	…	85

......

......

A bad design

Another Design

A good design

Students

Sid	Sname	Ssex	Sage	Specialty
3023001093	黄毅照	M	21	No
3011112340	周朝威	F	20	Cs
3020621034	徐鑫	M	18	Cs
3020831035	薄延嵩	M	19	Cs
3021131123	胡俊	F	22	Cs

Courses

cid	Cname	credit
1	DB	4
2	OS	5
3	English	4
4	Math	4

Enrolled

sid	cid	grade1	grade2	grade3
3023001093	1	90		
3023001093	2	85		
3020621034	1	90		
3020831035	1	75		
3021131123	2	75		

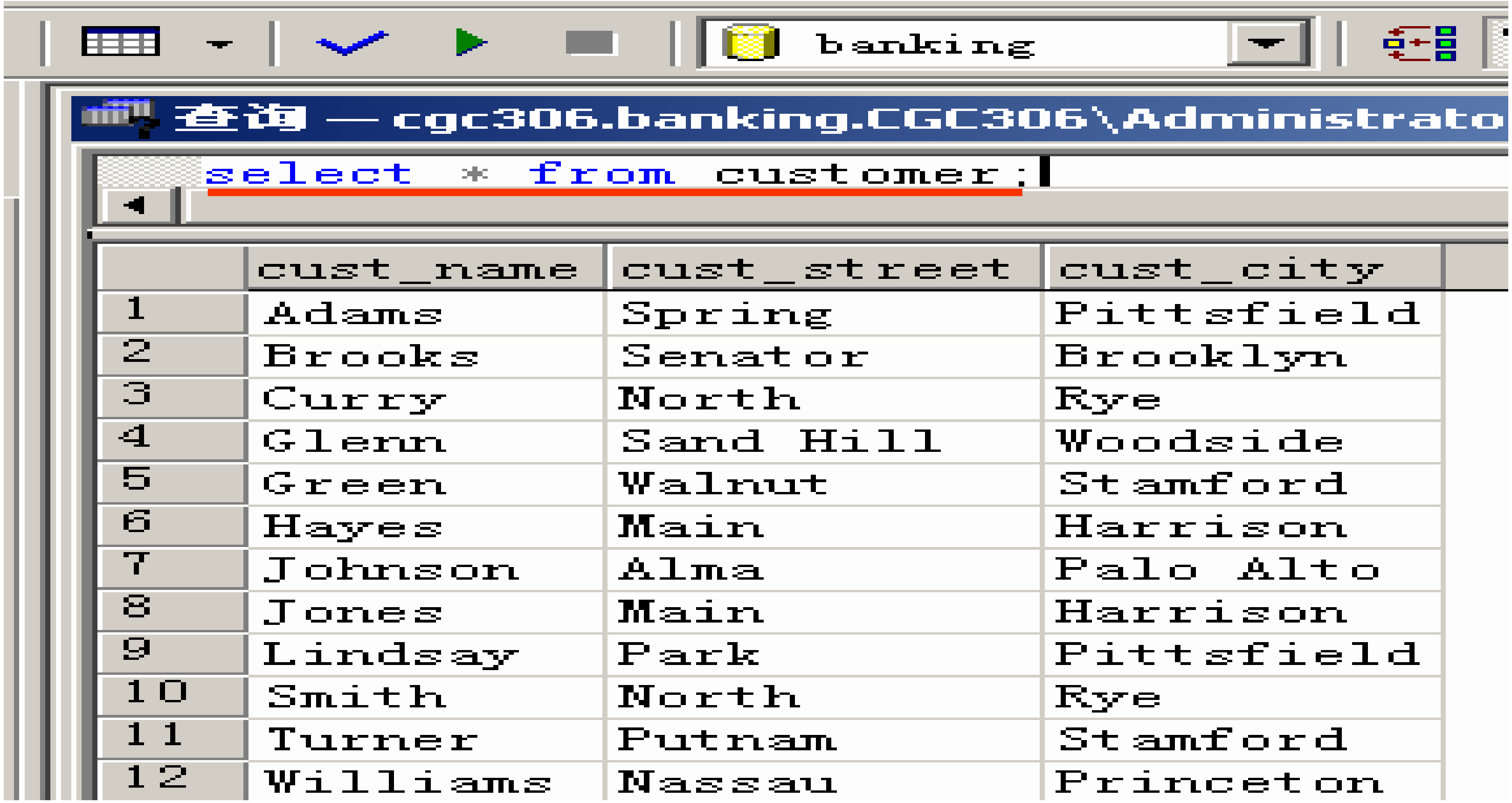
Why Study Databases? (Cont.)

- Modeling and design of databases
  - Get/Abstract data models from the real world, and then translate them into the forms suitable for the target Database Management System (DBMS) --- tables, views.

- Programming
  - Use database --- queries and update of data.
  - SQL = “intergalactic data-speak”.

Database Access Approaches

- Method 1:
  - Access database by using interactive tools (e.g., Query Analyzer of SQL Server, Sql*Plus and Work Sheet of ORACLE, etc.) provided by DBMS.



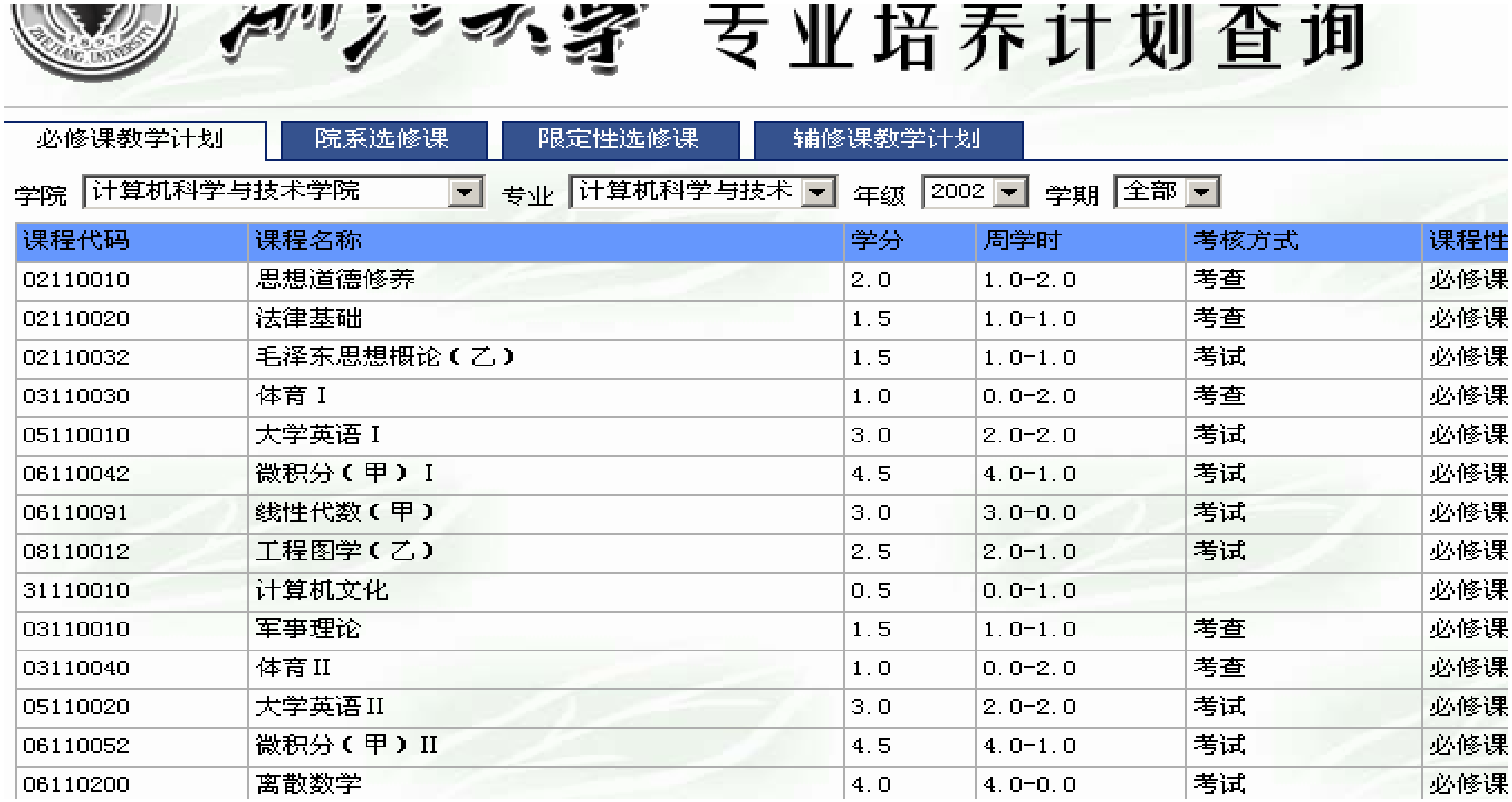
查询 — cgc306.banking.CGC306\Administrato

```
select * from customer;
```

	cust_name	cust_street	cust_city
1	Adams	Spring	Pittsfield
2	Brooks	Senator	Brooklyn
3	Curry	North	Rye
4	Glenn	Sand Hill	Woodside
5	Green	Walnut	Stamford
6	Hayes	Main	Harrison
7	Johnson	Alma	Palo Alto
8	Jones	Main	Harrison
9	Lindsay	Park	Pittsfield
10	Smith	North	Rye
11	Turner	Putnam	Stamford
12	Williams	Nassau	Princeton

Database Access Approaches (Cont.)

- Method 2:
  - Invoke ODBC/JDBC to access database by using development tools (e.g., VC++, PB, Delphi, ASP, JSP, PHP, etc.).



学院 计算机科学与技术学院　专业 计算机科学与技术　年级 2002　学期 全部

课程代码	课程名称	学分	周学时	考核方式	课程性
02110010	思想道德修养	2.0	1.0-2.0	考查	必修课
02110020	法律基础	1.5	1.0-1.0	考查	必修课
02110032	毛泽东思想概论（乙）	1.5	1.0-1.0	考试	必修课
03110030	体育 I	1.0	0.0-2.0	考查	必修课
05110010	大学英语 I	3.0	2.0-2.0	考试	必修课
06110042	微积分（甲） I	4.5	4.0-1.0	考试	必修课
06110091	线性代数（甲）	3.0	3.0-0.0	考试	必修课
08110012	工程图学（乙）	2.5	2.0-1.0	考试	必修课
31110010	计算机文化	0.5	0.0-1.0		必修课
03110010	军事理论	1.5	1.0-1.0	考查	必修课
03110040	体育 II	1.0	0.0-2.0	考查	必修课
05110020	大学英语 II	3.0	2.0-2.0	考试	必修课
06110052	微积分（甲） II	4.5	4.0-1.0	考试	必修课
06110200	离散数学	4.0	4.0-0.0	考试	必修课

Why Study Databases? (Cont.)

- Modeling and design of databases
  - Get/Abstract data models from the real world, and then translate them into the forms suitable for the target Database Management System (DBMS) --- tables, views.
- Programming
  - Use database --- queries and update of data.
  - SQL = “intergalactic data-speak”.
- DBMS implementation
  - How does DBMS work, and how to design a DBMS, i.e., Database System Design.

What Is a Database?

- Database
  - A collection of interrelated data, relevant to an enterprise.
  - A large collection of integrated and persistent data (DB) [R. Ramakrishnan, J. Gehrhe].
  - A collection of information that exists over a long period of time, often many years [Ullman].
  - 长期存储在计算机内、有组织的、可共享的数据集合[萨师煊，王珊].

- Database Management System (DBMS)
  - (Database) + A set of programs used to access, update and manage the data in database.

Characteristics of DBMS

- Efficiency and scalability in data access.
- Reduced application development time.
- Data independence (including physical data independence and logical data independence).
- Data integrity and security.
- Concurrent access and robustness (i.e., recovery).

File-Processing System

- File-processing system is supported by a conventional Operating System (OS).
  - New application programs must be written if necessary, and new data files are created as required.
  - But over a long period of time, data files may be in different formats.
  - Data files are independent each other.

Drawbacks of File-Processing System

- Data redundancy and inconsistency
  - Multiple file formats, duplication of information in different files
- Difficulty in accessing data
  - Need to write a new program to carry out each new task
- Data isolation — multiple files and multiple formats
  - Difficult to retrieve, difficult to share
- Integrity problems
  - Integrity constraints (e.g. account balance > 0) become part of program code
  - Hard to add new constraints or change existing ones

Drawbacks of File-Processing System (Cont.)

- No atomicity of updates
  - Failures may leave database in an inconsistent state with partial updates carried out, e.g., transfer of funds from one account to another should either complete or not happen at all. --- atomicity

- Difficult to concurrent access by multiple users
  - Concurrent access is needed for performance
  - Uncontrolled concurrent access can lead to inconsistency, e.g., two persons are reading a balance and updating it simultaneously.

- Security problems (i.e., Right person uses right data)

- Database systems offer solutions to all the above problems!

The DBMS Marketplace

- Relational DBMS companies
  - Oracle, Sybase – are among the largest database software companies in the world.
  - IBM DB2 -- is by some accounts the largest DBMS vendor in the world.
  - Microsoft SQL-Server, plus Microsoft Access for the cheap DBMS on the desktop, answered by *lite* systems from other competitors.

- Relational database (DB) companies are also challenged by object-oriented DB companies.
- But countered with object-relational systems, which retain the relational core while allowing type extension as in object-oriented (OO) systems.
- Other database products: Foxbase, FoxPro, dBase, Ingres, Paradox, …

The DBMS Marketplace (Cont.)

- Open source databases
  - MySQL (available at http://www.mysql.com): is the most popular open source database for small system on web sites, is a key part of LAMP (Linux, Apache, MySQL, PHP/Perl/Python), and is a fast growing open source enterprise software stack.
  - PostgreSQL (available at http://www.postgresql.org): is a highly scalable, open source object-relational database management system, and is originally developed by the Department of Computer Science, UC Berkeley (called *Postgres*)

Outline

- ❑ Purpose of Database Systems
- ❑ View of Data
  - ➢ Level of Data Abstraction
  - ➢ Schemas and Instances
  - ➢ Physical Independence vs. Logical Indepen
  - ➢ Data Models
- ❑ Database Language
- ❑ Database Design
- ❑ Database Users and Administrators
- ❑ Transaction Management
- ❑ Database Architecture
- ❑ History of Database Systems
- ❑ Summary

Levels of Data Abstraction

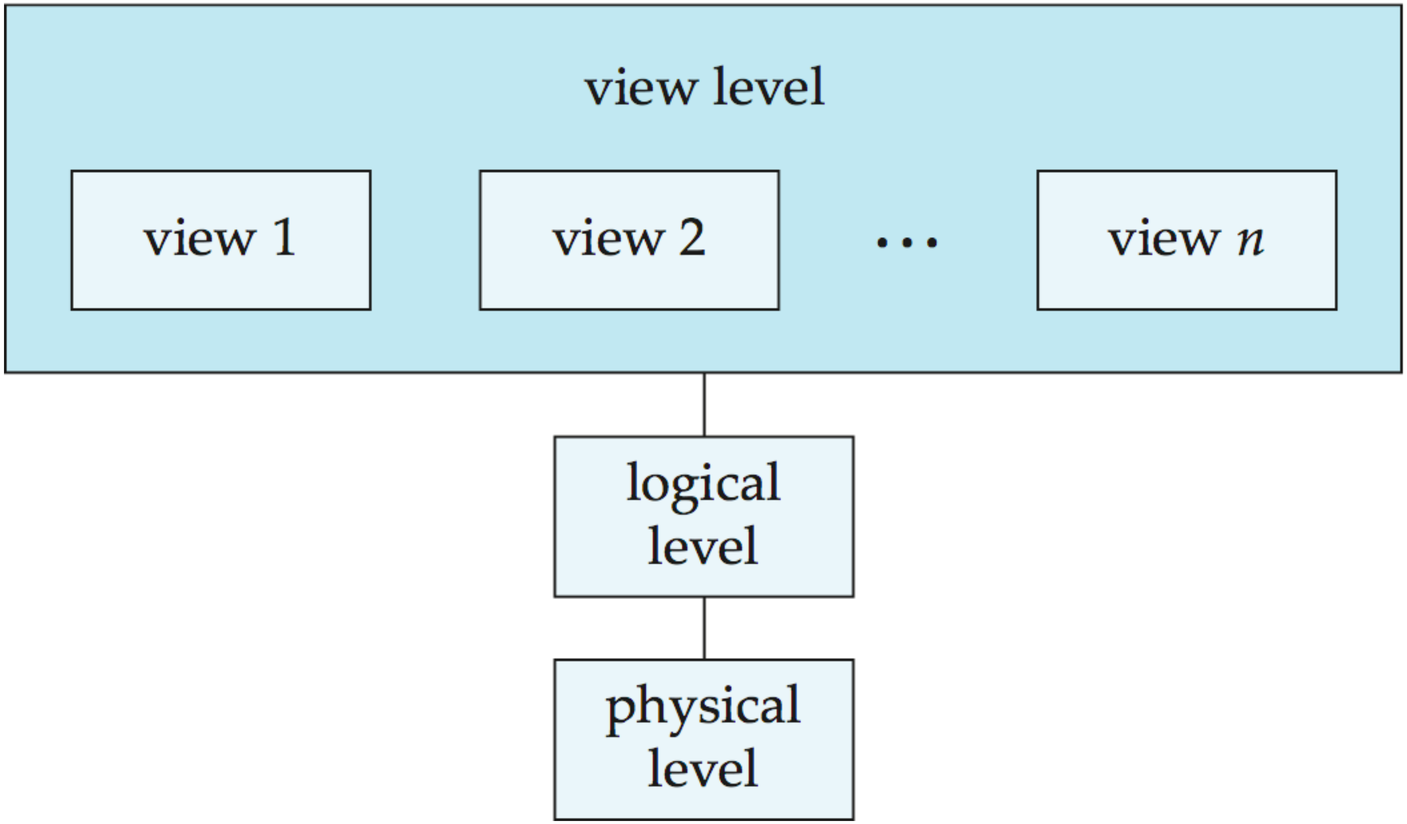
- How to use DB --- Different usage needs different level of abstraction.
  - Physical level: describes how a record is stored.
  - Logical level: describes data stored in database, and the relationships among the data on upper level.
  - Compare physical level with logical: e.g.,

```
type instructor = record
         ID: string;
         name: string;
         dept_name: string;
         salary: integer;
     end
```

  - View level: application programs hide details of data types. Note that views can also hide information (e.g., employee’s salary) for security purposes.

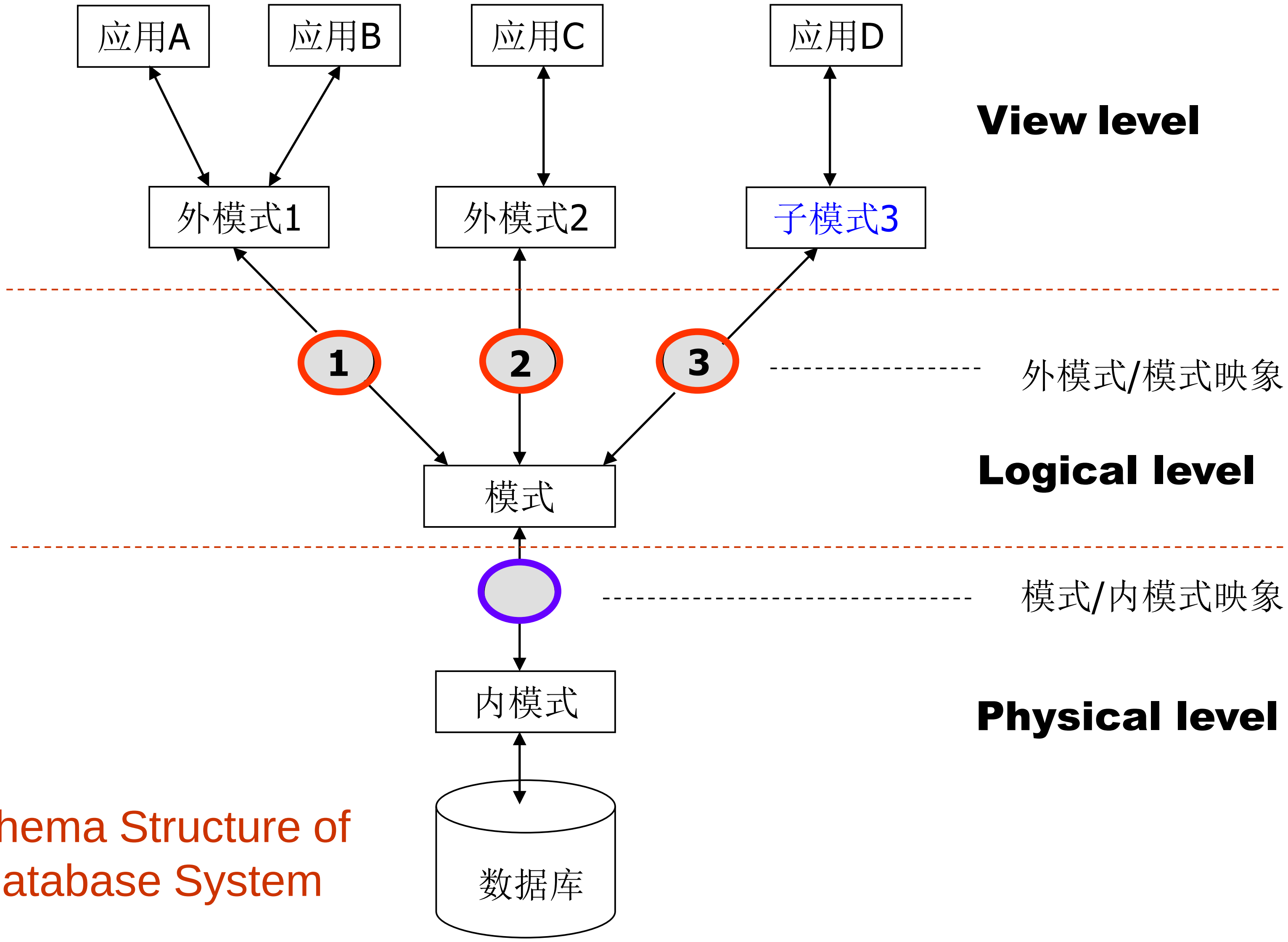
View of Data

- An architecture of a database system



Schemas and Instances

- Schema – the structure of the database on different level
  - Analogous to type information of a variable in a program
  - Physical schema: database structure design at the physical level
  - Logical schema: database structure design at the logical level
  - Subschema: schema at view level
- Instance – the actual content of the database at a particular point in time
  - Analogous to the value of a variable
- Similar to types and variables in programming languages
  - type ↔ schema, variable ↔ instance



Schema Structure of Database System

Physical Independence vs. Logical Independence

- Ability to modify a schema definition at one level without affecting a schema definition at a higher level.

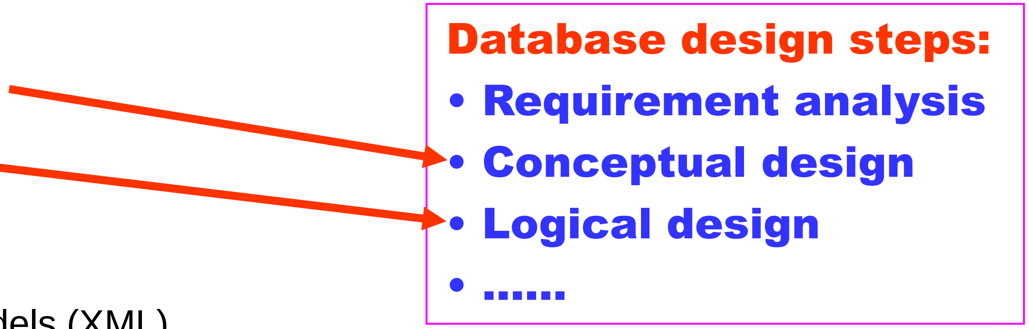
- Physical data independence – the ability to modify the physical schema without changing the logical schema.
  - Applications depend on the logical schema.
  - Applications are insulated from how data is structured and stored.
  - **One of the most important benefits of using a DBMS!**

- Logical data independence – protect application programs from changes in logical structure of data.
  - Logical data independence is hard to achieve as the application programs are heavily dependent on the logical structure of data.

Data Models

- Data model is a collection of conceptual tools for describing
  - data structure
  - data relationships
  - data semantics
  - data constraints

- Different level of data abstraction needs different data model to describe
  - Entity-Relationship model
  - Relational model
  - Other models:
    - Object-oriented model
    - Semi-structured data models (XML)
    - Older models such as network model, hierarchical model, etc.



Database design steps:

- **Requirement analysis**
- **Conceptual design**
- **Logical design**
- **……**

Outline

- ❑ Purpose of Database Systems
- ❑ View of Data
- ❑ Database Language
  - ➢ Data Definition Language (DDL)
  - ➢ Data Manipulation Language (DML)
  - ➢ SQL
- ❑ Database Design
- ❑ Database Users and Administrators
- ❑ Transaction Management
- ❑ Database Architecture
- ❑ History of Database Systems
- ❑ Summary

Database Language

- Database Language
  - Data Definition Language (DDL): Specification notation for defining the database schema.
  - Data Manipulation Language (DML): Language for accessing and manipulating the data organized by appropriate data model.
  - Data Control Language (DCL)

Data Definition Language (DDL)

- Data Definition Language (DDL)
  - Specifies a database scheme as a set of definitions of relational schema.
  - Also specifies storage structure, access methods, and consistency constraints.
  - DDL statements are compiled, resulting in a set of tables stored in a special file called data dictionary.

```
CREATE TABLE account (
                    account_number char(10),
                    balance  integer);
```

- Data dictionary contains metadata (i.e., the data about data) about
  - Database schema
  - Integrity constraints
    - Primary Key
    - Referential integrity
  - Authorization

Data Manipulation Language (DML)

- Data Manipulation Language (DML)
  - Retrieve data from the database
  - Insert / delete / update data in the database
  - DML also known as query language

- Two classes of DMLs
  - Procedural DML – user specifies what data is required and how to get those data (e.g., C, Pascal, Java, etc.).
  - Nonprocedural DML – user specifies what data is required without specifying how to get those data (e.g., SQL, Prolog, etc.).

- SQL is the most widely used query language
  - Set-based, declarative.
  - But procedural extensions are offered by different database systems.

SQL

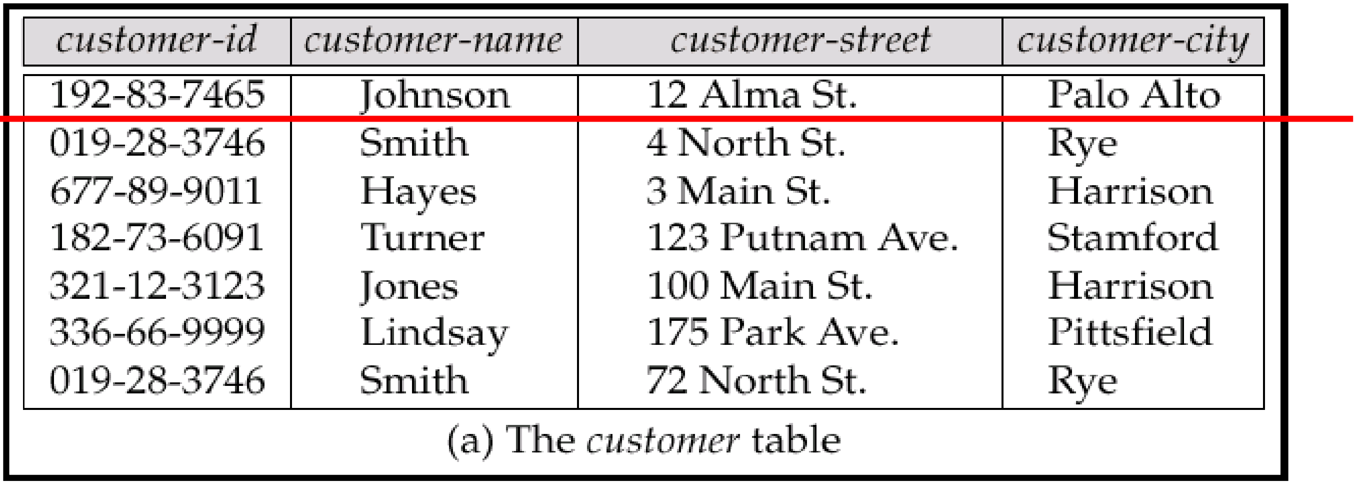
- SQL = DDL+ DML+DCL
- SQL has been widely used
  - Structured Query Language (SQL) --- IBM. System R, Called Structured English QUEry Language (SEQUEL), 1975.
  - Example 1: Find the name of the customer with customer-id 192-83-7465

```
SELECT  customer.customer-name
FROM    customer
WHERE   customer.customer-id = ‘192-83-7465’
```

  - Example 2: Find the balances of all accounts held by the customer with customer-id 192-83-7465

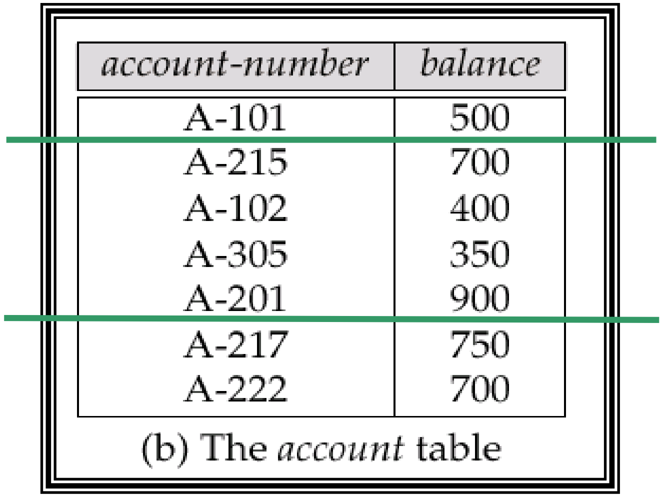
```
SELECT  account.balance
FROM    depositor, account
WHERE   depositor.customer-id = ‘192-83-7465’ and
        depositor.account-number = account.account-number
```

A Sample Relational Database



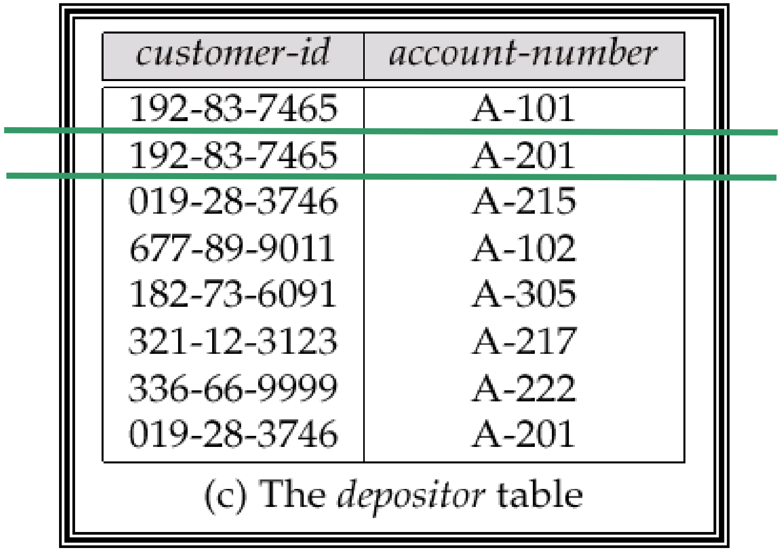
customer-id	*customer-name*	*customer-street*	*customer-city*
192-83-7465	Johnson	12 Alma St.	Palo Alto
019-28-3746	Smith	4 North St.	Rye
677-89-9011	Hayes	3 Main St.	Harrison
182-73-6091	Turner	123 Putnam Ave.	Stamford
321-12-3123	Jones	100 Main St.	Harrison
336-66-9999	Lindsay	175 Park Ave.	Pittsfield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

(a) The *customer* table



account-number	*balance*
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



customer-id	*account-number*
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

SQL (Cont.)

- SQL is the most widely used non-procedural query language.

- Three classes of SQL usage
  - Use it directly in the interactive environment, e.g., Query Analyzer of SQL Server, Sql*Plus and Work Sheet of Oracle, mysql command line client of MySQL, etc.
  - Use it by host language through ODBC / JDBC.
  - Use it by host language with embed-SQL.

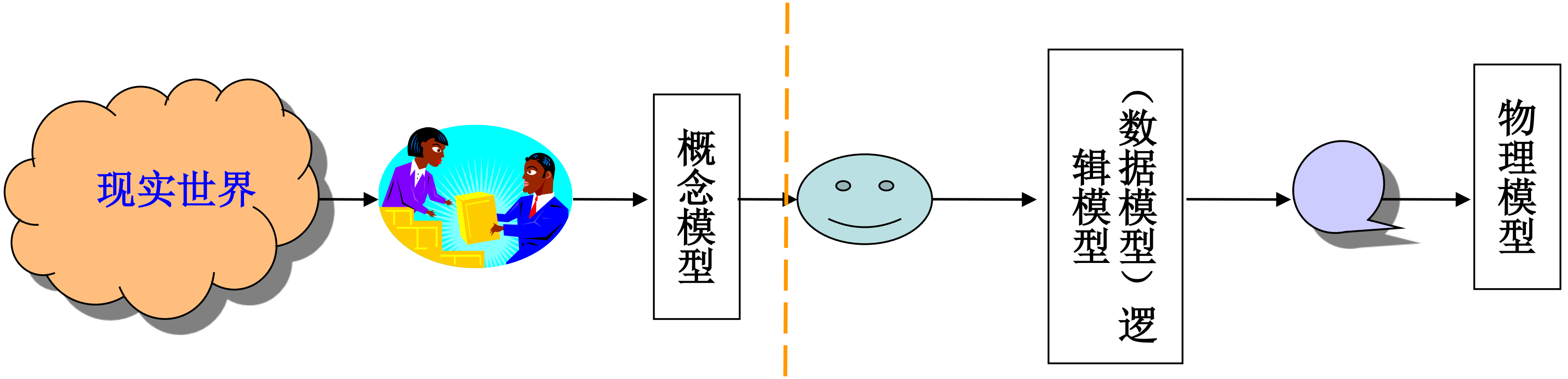
Outline

- Purpose of Database Systems
- View of Data
- Database Language
- Database Design
  - Steps of Database Design
  - Entity-Relationship Model
  - Relational Model
- Database Users and Administrators
- Transaction Management
- Database Architecture
- History of Database Systems
- Summary

Steps of Database Design

- Requirement analysis
  - What data, applications, and operations needed.
- Conceptual database design
  - A high-level description of data, constraints using Entity-Relationship (E-R) model or a similar high level data model.
- Logical database design
  - Convert the conceptual design into a DB schema.
- Schema refinement
  - Normalization of relations: Check relational schema for redundancies and related anomalies.
- Physical database design
  - Indexing, query, clustering, and database tuning.
- Create and initialize the database & Security design
  - Load initial data, testing.
  - Identify different user groups and their roles.

Steps of Database Design (Cont.)



信息世界

E-R 图

机器世界

关系模型
（表结构）

物理存储结构

面向对象模型

--非关系模型--

层次模型

网状模型

1968, IBM IMS

树型结构

(除根结点外，每个结点有且只有一个双亲结点**)**

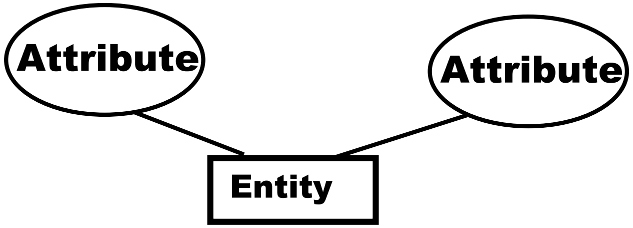
DBTG模型 **| CODASYL**系统

网状结构 **(**一个结点可有多个双亲，允许多个结点无双亲**)**

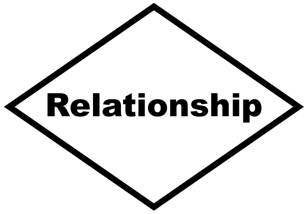
Honeywell IDS/2，**HP image**

Entity-Relationship (E-R) Model

- E-R model of real world
  - Entities (objects)
    - E.g., customers, accounts, bank branch.
    - Entities are described by a set of attributes.



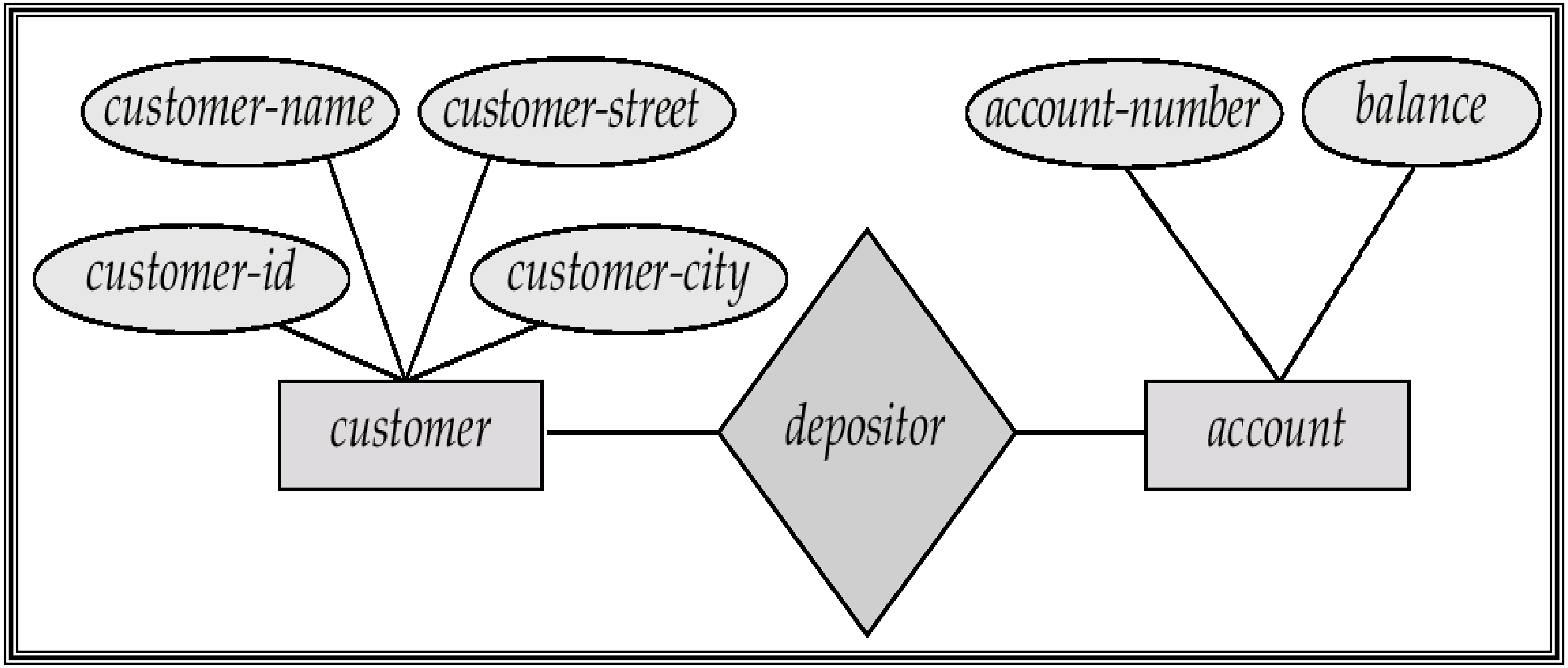
  - Relationships between entities
    - E.g., Account *A-101* is held by customer *Johnson*.
    - Relationship set *depositor* associates customers with accounts



- E-R Model is widely used for database design
  - Database design in E-R model is usually converted to design in the relational model .
  - E-R model was first proposed by Peter Chen.

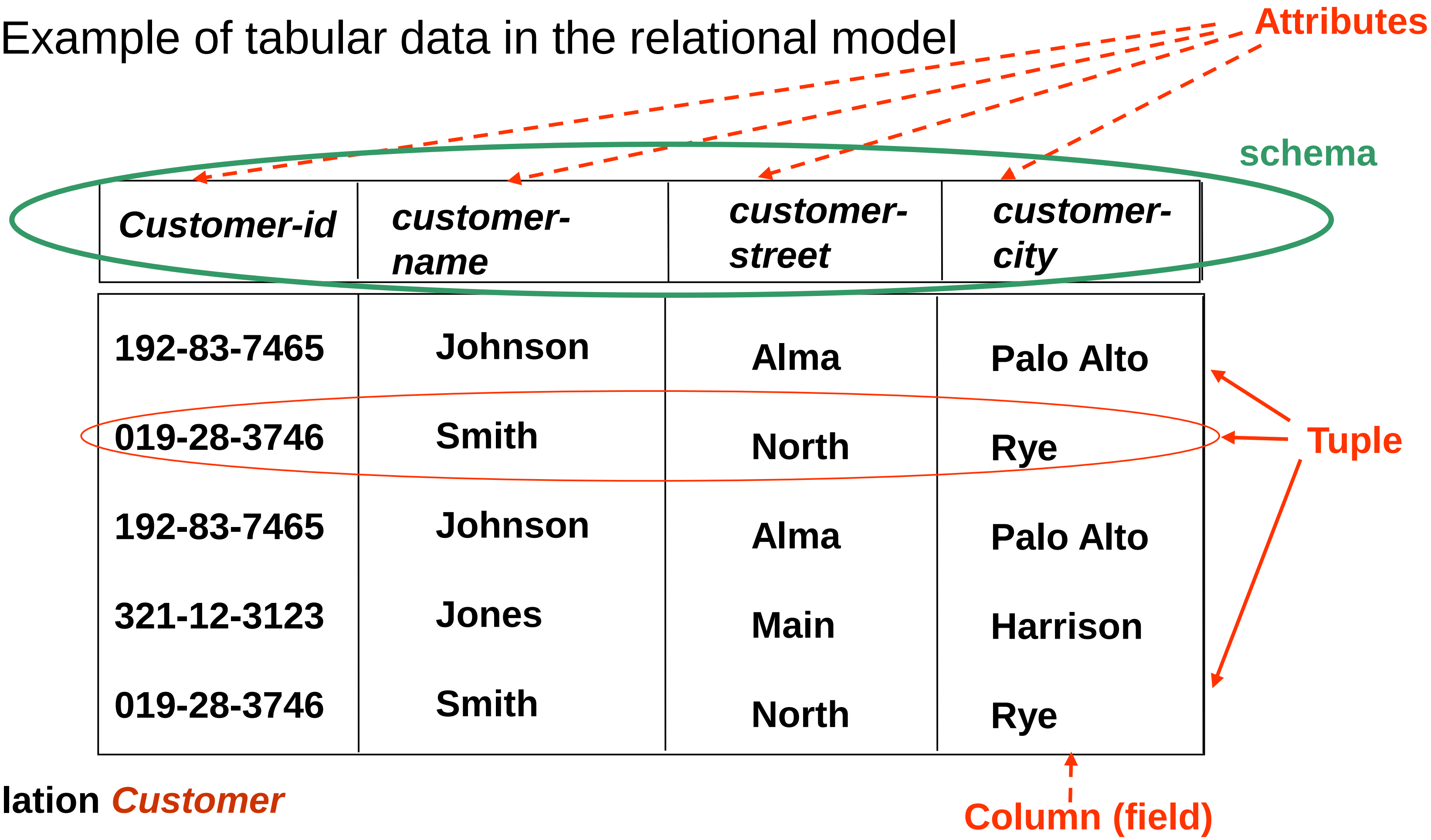
Entity-Relationship (E-R) Model (Cont.)

- Example of schema in the entity-relationship model



Relational Model

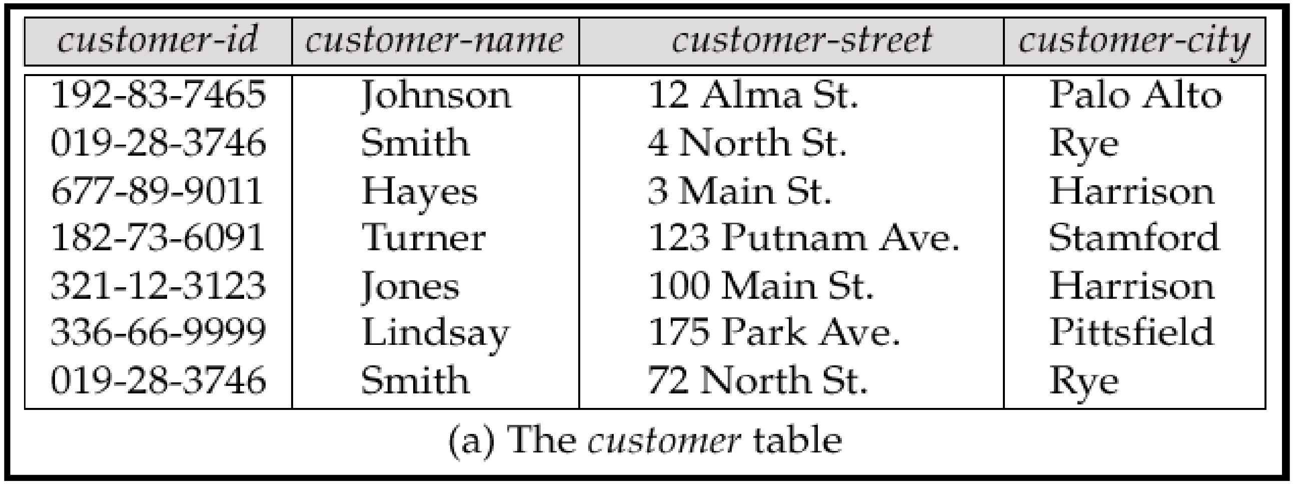
- Transfer E-R diagrams into relational schema
- Example of tabular data in the relational model



Customer-id	***customer-name***	***customer-street***	***customer-city***
192-83-7465	**Johnson**	**Alma**	**Palo Alto**
019-28-3746	**Smith**	**North**	**Rye**
192-83-7465	**Johnson**	**Alma**	**Palo Alto**
321-12-3123	**Jones**	**Main**	**Harrison**
019-28-3746	**Smith**	**North**	**Rye**

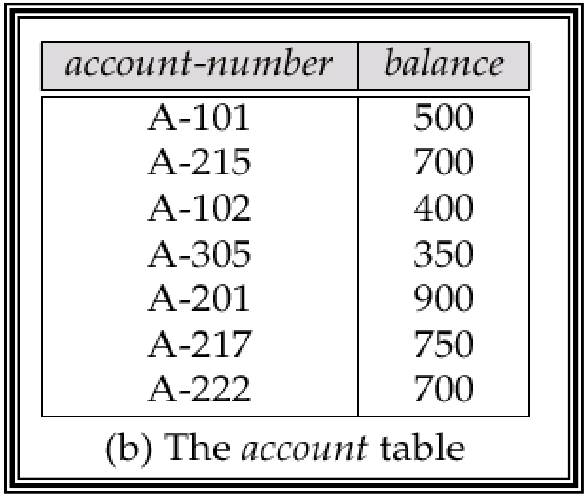
Relation *Customer*

A Sample Relational Database



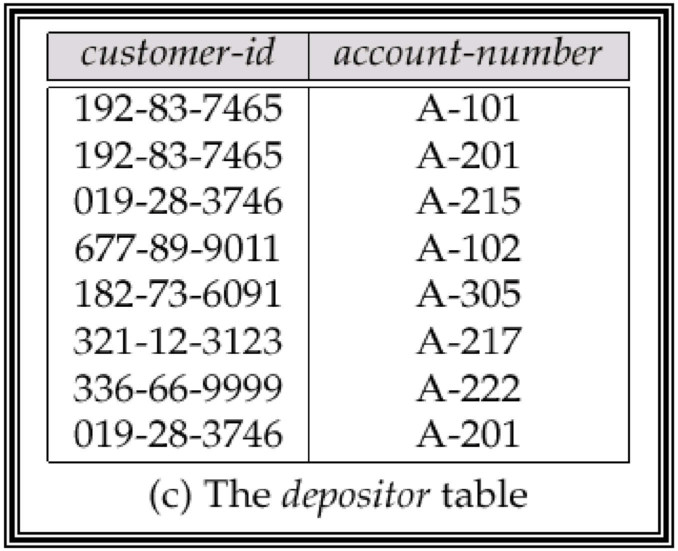
customer-id	*customer-name*	*customer-street*	*customer-city*
192-83-7465	Johnson	12 Alma St.	Palo Alto
019-28-3746	Smith	4 North St.	Rye
677-89-9011	Hayes	3 Main St.	Harrison
182-73-6091	Turner	123 Putnam Ave.	Stamford
321-12-3123	Jones	100 Main St.	Harrison
336-66-9999	Lindsay	175 Park Ave.	Pittsfield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

(a) The *customer* table



account-number	*balance*
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



customer-id	*account-number*
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

Another example: University Database

Students

Sid	Sname	Ssex	Sage	sdept
3023001093	Tom	M	21	Cs
3011112340	Mary	F	20	Cs
3020621034	Jack	M	18	Cs
3020831035	Smith	M	19	Ma
3021131123	Alane	F	22	Is

Courses

cid	Cname	credit
1	DB	4
2	OS	5
3	English	4
4	Math	4

Enrolled

sid	cid	grade
3023001093	1	92
3023001093	2	88
3020621034	1	70
3020831035	1	85
3021131123	2	95

view level (subschema)

Achievement

Sid	Sname	Cname	credit	grade
3023001093	Tom	DB	4	92
3023001093	Tom	OS	5	88
3020621034	Jack	DB	4	70
3020831035	Smith	DB	4	85
3021131123	Alane	OS	5	95

Logical level (schema)

student

Sid	Sname	Ssex	Sage	sdept
3023001093	Tom	M	21	Cs
3011112340	Mary	F	20	Cs
3020621034	Jack	M	18	Cs
3020831035	Smith	M	19	Ma
3021131123	Alane	F	22	Is

course

cid	Cname	credit
1	DB	4
2	OS	5
3	English	4
4	Math	4

Enrolled

sid	cid	grade
3023001093	1	92
3023001093	2	88
3020621034	1	70
3020831035	1	85
3021131123	2	95

Outline

- ❑ Purpose of Database Systems
- ❑ View of Data
- ❑ Database Language
- ❑ Database Design
- ❑ Database Users and Administrators
  - ➢ Database Users
  - ➢ Database Administrators
- ❑ Transaction Management
- ❑ Database Architecture
- ❑ History of Database Systems
- ❑ Summary

Database Users

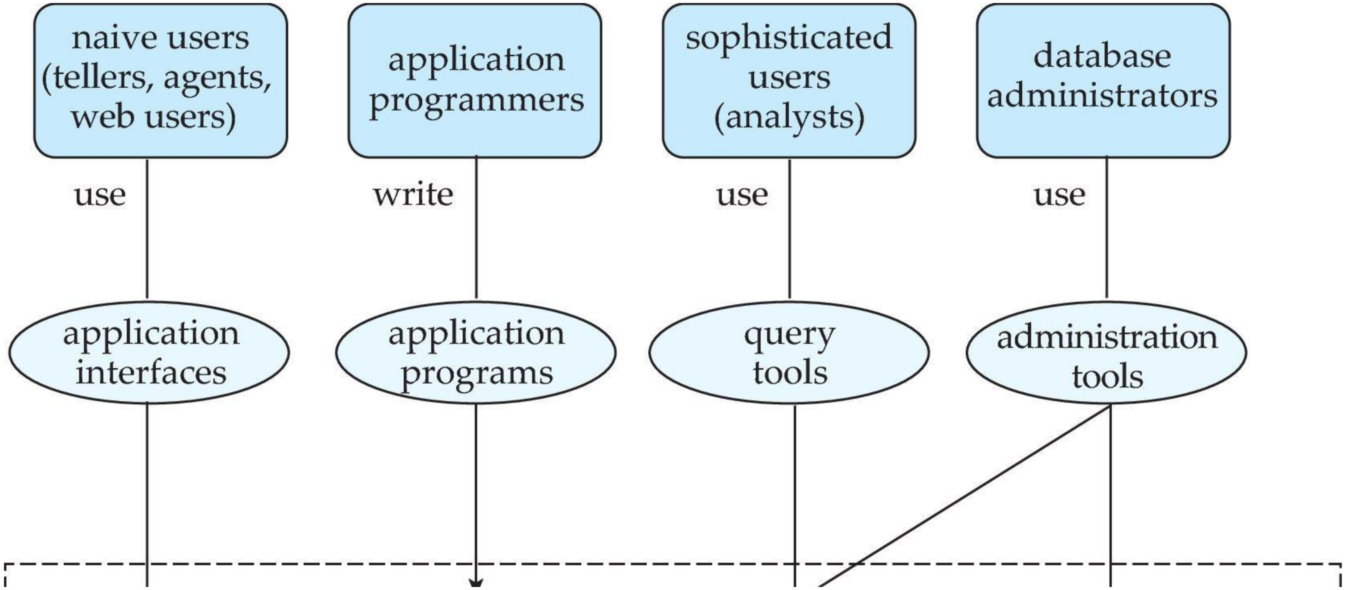
- ❑ Users are differentiated by the way they expect to interact with the system

- ❑ Naive users – invoke one of the permanent application programs that have been written previously by a high level language.
  - ➢ E.g., people accessing database over the web, bank tellers, clerical staff.

- ❑ Application programmers – interact with system via SQL calls.
- ❑ Sophisticated users – form requests in a database query language.
  - ➢ E.g., Online Analytical Processing (OLAP), Data mining.
- ❑ Specialized users – write specialized database applications that do not fit into the traditional data processing framework.
  - ➢ E.g., CAD, Expert System (ES), KDB.

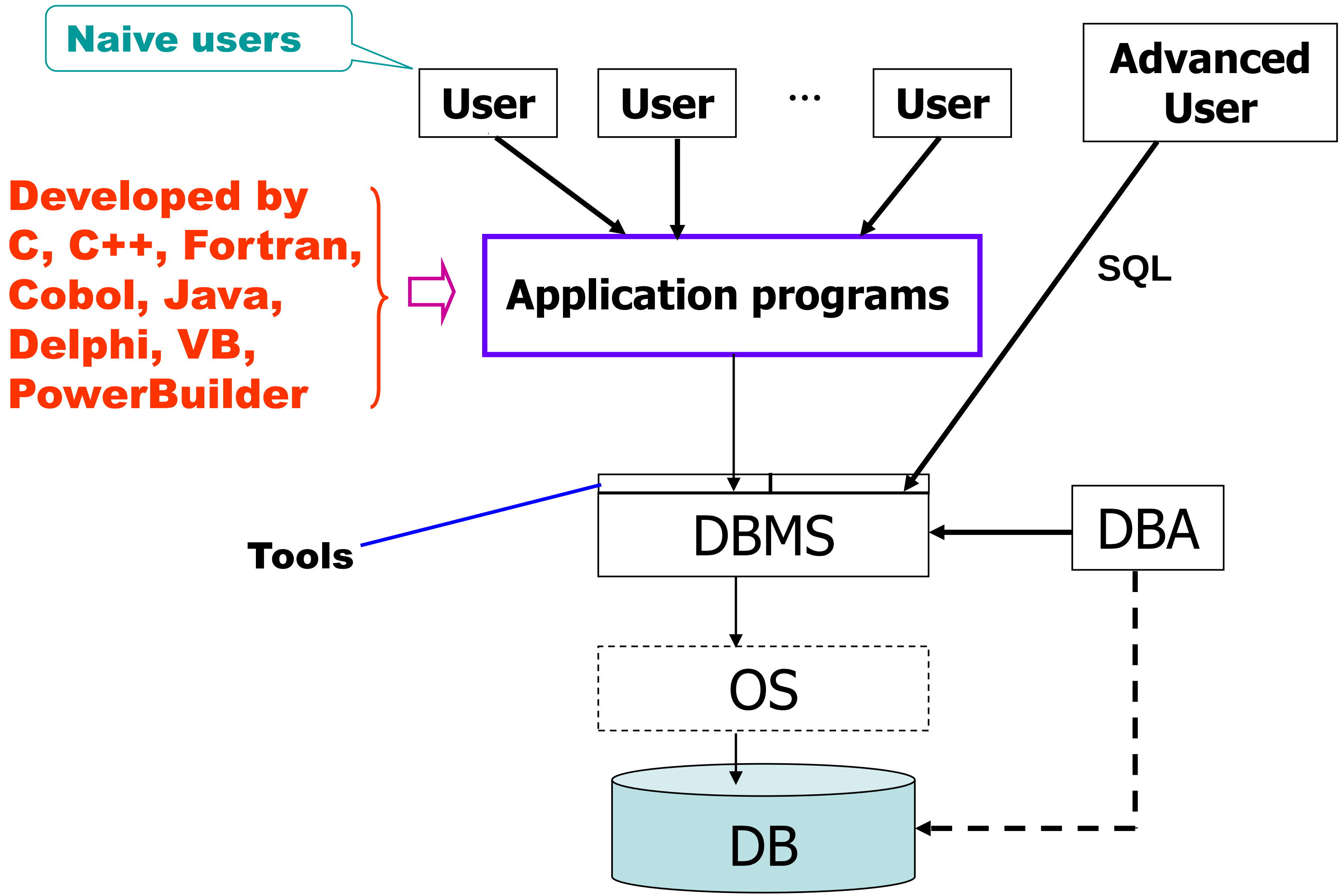
Professional users

Database Users and Administrators



Database

How to Use a Database



Database Administrator

- Database administrator (DBA): A special user having central control over database and programs accessing those data.
- DBA has the highest privilege for the database.
- DBA coordinates all the activities of the database system.
- DBA controls all users authority to the database.
- DBA has a good understanding of the enterprise's information resources and requirements.

Database Administrator (Cont.)

- Database administrator's duties/functions include:
  - Schema definition
  - Storage structure and access method definition
  - Schema and physical organization modification
  - Granting of authorization for data access
  - Routing maintenance
    - Monitoring performance and responding to changes in requirements
    - Security for the database (e.g. periodically backup database, recovery when failure)

Outline

- ❑ Purpose of Database Systems
- ❑ View of Data
- ❑ Database Language
- ❑ Database Design
- ❑ Database Users and Administrators
- ❑ Transaction Management
- ❑ Database Architecture
- ❑ History of Database Systems
- ❑ Summary

Transaction Management

- Concurrent use/access is important, but causes problems/conflict.

- A transaction is a collection of operations that performs a single logical function in a database application.
- Transaction requirements include atomicity, consistence, isolation, durability.
- Transaction-management component ensures that the database remains in a consistent (or correct) state, although system failures (e.g., power failures and operating system crashes) and transaction failures. --- by backup and recovery subsystem
- Concurrency-control manager controls the interaction among the concurrent transactions.

Outline

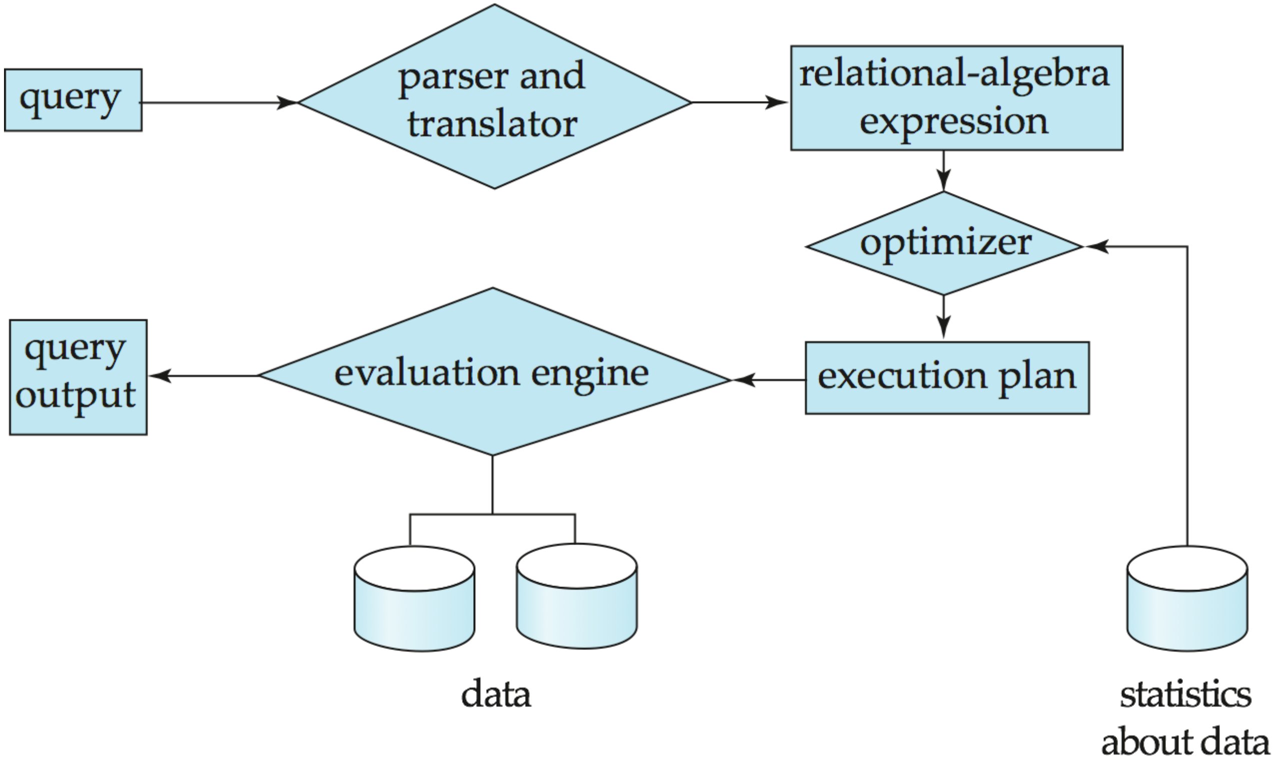
- ❑ Purpose of Database Systems
- ❑ View of Data
- ❑ Database Language
- ❑ Database Design
- ❑ Database Users and Administrators
- ❑ Transaction Management
- ❑ Database Architecture
  - ➢ Storage Manager
  - ➢ Query Processor
  - ➢ Overall System Structure
  - ➢ Application Architecture
- ❑ History of Database Systems
- ❑ Summary

Storage Manager

- 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 system.
- Storage Manager is responsible for the following tasks:
  - Interaction with the file manager
  - Efficient storing, retrieving and updating of data
- Storage Manager includes
  - Transaction manager
  - Authorization and integrity manger
  - File manager (interaction with the file system to process data files, data dictionary, and index files)
  - Buffer manager

Query Processor

- Query Processor includes DDL interpreter, DML compiler, and query processing.
  - Parsing and translation
  - Optimization
  - Evaluation



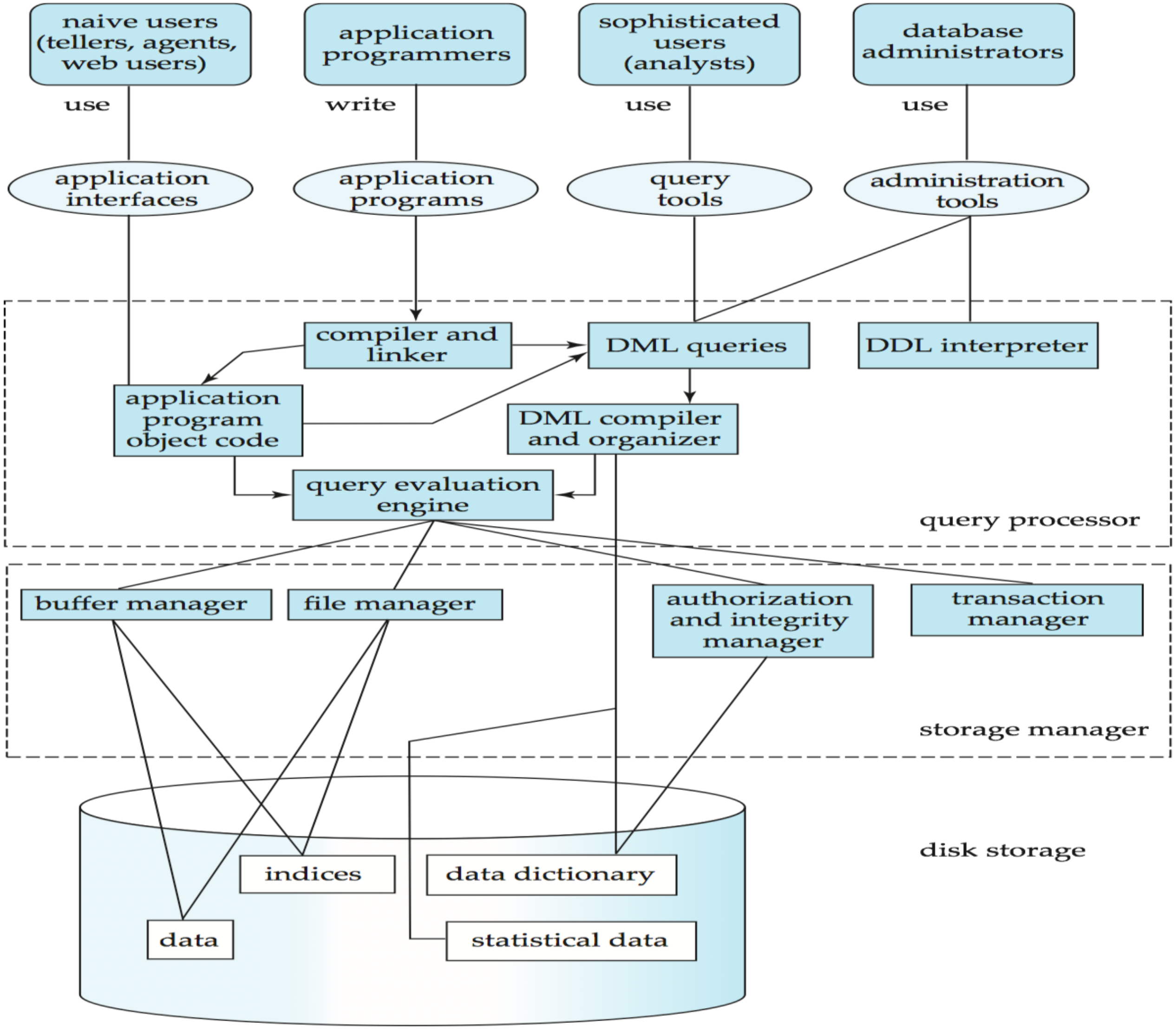
Query Processing Optimization

- Alternative ways of evaluating a given query
  - Equivalent expressions
  - Different algorithms for each operation

- Cost difference between a good and a bad way of evaluating a query can be enormous

- Need to estimate the cost of operations
  - Depends critically on statistical information about relations which the database must maintain
  - Need to estimate statistics for intermediate results to compute cost of complex expressions

Database System Internals

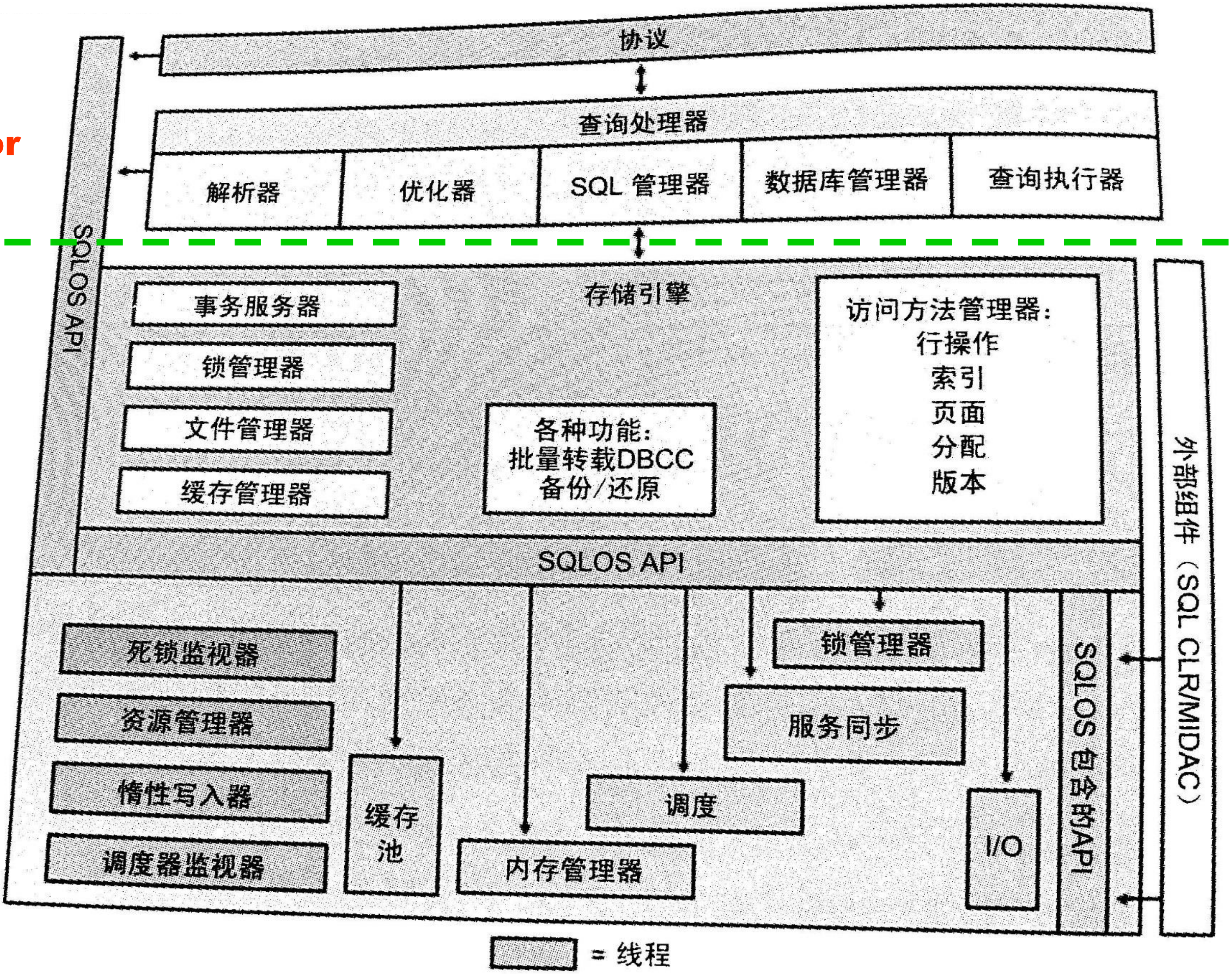


Query processor

Storage manager

Query processor

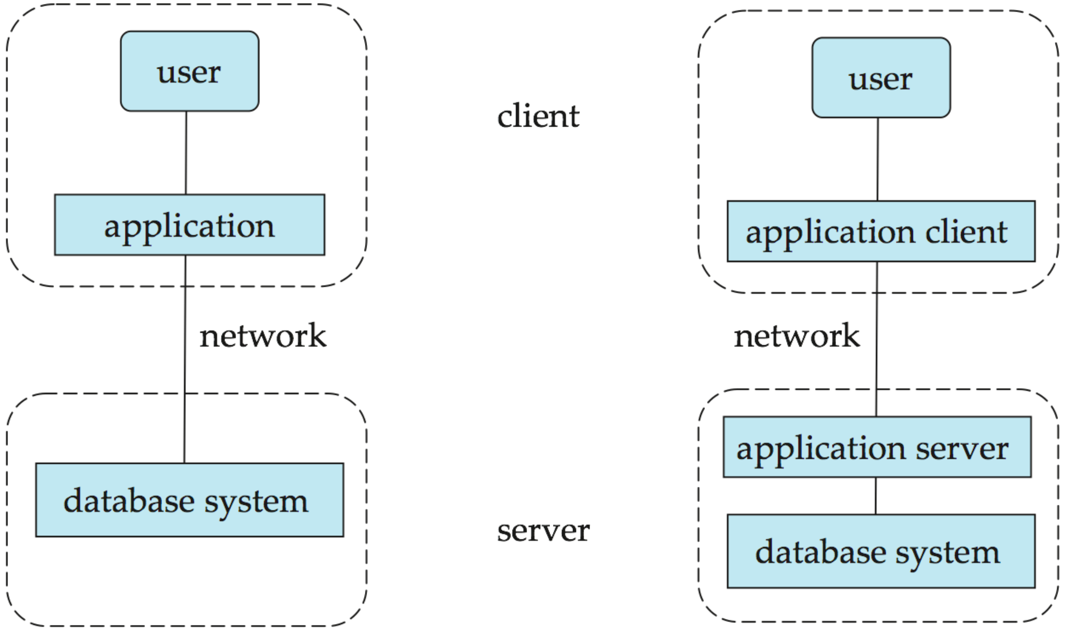
Storage manager



Example：SQL Server 数据库引擎的主要组件

Database Architecture

- The architecture of a database system is greatly influenced by the underlying computer system on which the database is running:
  - Centralized
  - Client-server
  - Parallel (multi-processor)
  - Distributed



(a) Two-tier architecture (b) Three-tier architecture

Outline

- ❑ Purpose of Database Systems
- ❑ View of Data
- ❑ Database Language
- ❑ Database Design
- ❑ Database Users and Administrators
- ❑ Transaction Management
- ❑ Database Architecture
- ❑ History of Database Systems
- ❑ Summary

History of Database Systems

- 1950s and early 1960s:
  - Data processing using magnetic tapes for storage
    - Tapes provided only sequential access
  - Punched cards for input

- Late 1960s and 1970s:
  - Hard disks allowed direct access to data
  - Network and hierarchical data models in widespread use
  - Ted Codd defines the relational data model
    - Would win the ACM Turing Award for this work
    - IBM Research begins System R prototype
    - UC Berkeley begins Ingres prototype
  - High-performance (for the era) transaction processing

History of Database Systems (Cont.)

- 1980s:
  - Research relational prototypes evolve into commercial systems
    - SQL becomes industrial standard
  - Parallel and distributed database systems
  - Object-oriented database systems
- 1990s:
  - Large decision support and data-mining applications
  - Large multi-terabyte data warehouses
  - Emergence of Web commerce
- Early 2000s:
  - XML and XQuery standards
  - Automated database administration
- Later 2000s:
  - Giant data storage systems
    - Google BigTable, Yahoo PNuts, Amazon, …

Outline

- ❑ Purpose of Database Systems
- ❑ View of Data
- ❑ Database Language
- ❑ Database Design
- ❑ Database Users and Administrators
- ❑ Transaction Management
- ❑ Database Architecture
- ❑ History of Database Systems
- ❑ Summary

Summary

- DBMS is used to maintain, query large datasets
- Benefits include recovery from system crashes, concurrent access, quick application development, data integrity, and security
- Levels of abstraction give data independence
- E-R model, and Relational model
- DDL, DML, and SQL
- DBAs hold responsible jobs and are well-paid!

- DB system typical architecture
- DBMS research and development (R&D) is one of the broadest, most exciting areas in computer science (CS)

Q & A

Your questions and suggestions are expected for me.

Thanks a lot!

Questions?

Questions?

Exercises: 1.7, 1.8, and 1.15 (see Page 32)