

CSCD 327: Relational Database Systems

Introduction

Instructor: Dr. Dan Li

Outline

- Course logistics
- Introduction
 - DB related concepts
- Reading assignment
 - course syllabus
 - Textbook 2.1

General Information

- Meeting Time: M-F 9am-9:50am, CEB 204/207
- Office: CEB, Rm 314
- Phone: 359-4373
- Email: DANL@ewu.edu
- Office Hours: M-F 10am-10:50am
- Course Website:
 - Course material will be posted on **Canvas**
- Grader: Jenna Jasa (jjasa@eagles.ewu.edu)

Textbook

- *“A First Course in Database Systems” or “Database Systems The Complete Book”*
 - Required
 - Most course contents are from this book
 - The complete book is also the required textbook for CSCD 427
- Other recommended books
 - SQL The Complete Reference, 3rd Edition (ISBN-10: 0071592555, ISBN-13: 978-0071592550)
 - SQL Bible, 2nd Edition (ISBN 978-0-470-22906-4)
 - SQL Cookbook, 1st Edition (ISBN 978-0-59-600976-2)
 - Database System Concepts, 6th Edition (ISBN 978-0-07-352332-3)
 - Database Management Systems, 3rd Edition (ISBN 978-0-07-246563-1)

Course Structure/Approach

- Lecture-centered
 - Core contents will be delivered through lectures on MWF.
 - Quizzes will not be announced in advance.
 - Be on time.
 - Attendance is required.
- Lab integrated
 - Hands-on exercises will be given through labs on TR.
 - Tools: MySQL database, MySQL Workbench or other preferred
 - Attendance is optional but highly recommended.
 - If no lab, come to room 204 for formal lectures; especially in the last 2–3 weeks of the quarter. Pay attention to announcement.
- Reading Outside of Class
 - Reading assignments will be given regularly.

Assessment & Grading System

	Total Weight
Homework including labs	40%
Two Midterm Exams	50%
Quizzes & In-class participation	10%

93% or above 4.0

60% - 93% $(\% - 53)/10$

Below 60 0.0

Note: No grade will be given in the range 0.1 - 0.6.

Course Policies

- **Academic Integrity**
 - Act in accordance with the ACM Code of Ethics and Professional Conduct
 - The University policy for academic integrity.
- **Disability Accommodation**
 - Reasonable accommodations are available for students who have a documented disability.
- **Makeup Tests**
 - There are NO make-up tests without prior consent from the instructor.
- **Attendance**
 - **Attending lectures is MANDATORY**; tardiness and unexcused absences are considered lack of in-class participation and will be reflected in your grade.
 - Quizzes will NOT be announced in advance.
- **Homework Late Policy**
 - No late work will be accepted without prior consent from the instructor.
- **Electronic Device Usage**
 - All cell phones, PDAs, music players and other electronic devices must be turned off or set to silent mode.
 - Laptops or workstations (if present) are allowed for **note-taking only** during lectures
- **Other University Policies**
 - Please refer to student handbook

Ready to Start?

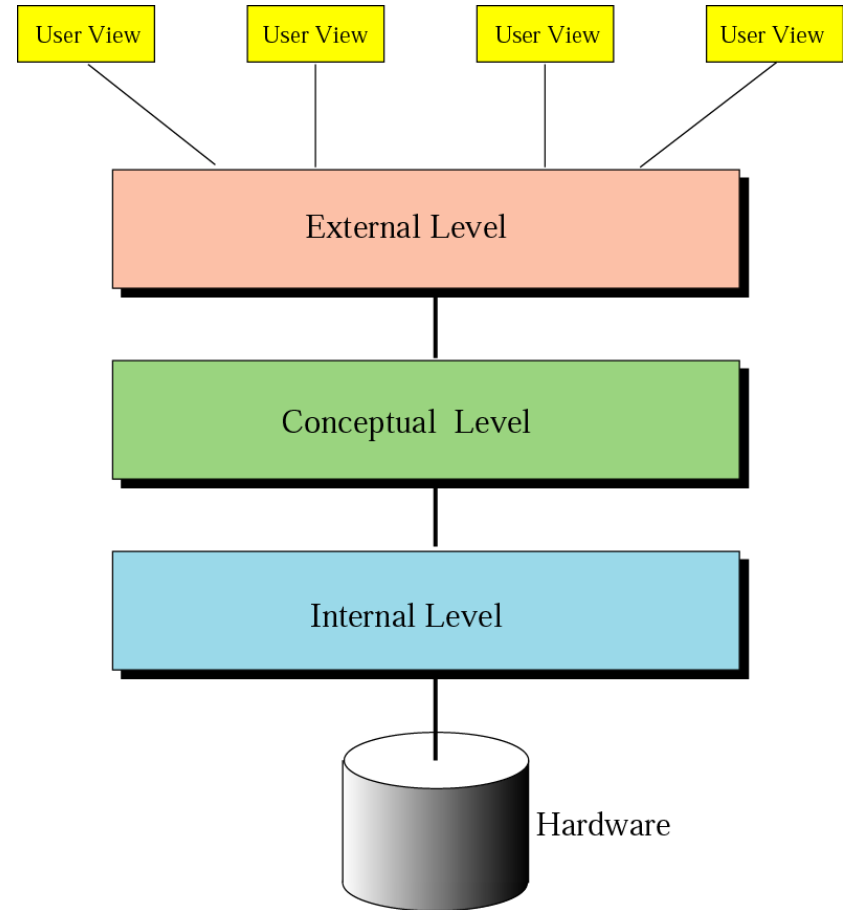
- What is a DB?
- What is a DBMS?
- Database Languages
- Data Models
- Database Users
- Database Architecture
- History of SQL

What Is a Database?

- A large collection of data
- Logically related
 - Doesn't make sense to put student registration and airline reservation into one database.
- Models some real-world enterprise
 - Entities (e.g., students, courses)
 - Relationships (e.g. Alice is taking CSCD 327)
- Usually shared among many people
 - Makes things complicated (e.g., where to store the database, concurrency control, ...)


What is a DBMS?





- DBMS: a software package designed to store and manage databases
- Three-level architecture




External Level

- Interacts directly with the user
- Collects data from a format that is familiar to the user

 Choose a payment method and click **Continue...**

 **Credit Card**

To pay by credit card, please fill out the fields below.

Name on card:

Card Number:

Expiration Date:


Security Code: [\(help\)](#)

Conceptual Level

- Defines the logical view of the data
- Main functions of DBMS are in this level

Choose a payment method and click **Continue...**

☒ VISA ☐ MasterCard ☐ American Express ☐ DISCOVER ☐ Check / MO

 **Credit Card**

To pay by credit card, please fill out the fields below.

Name on card:

Card Number:

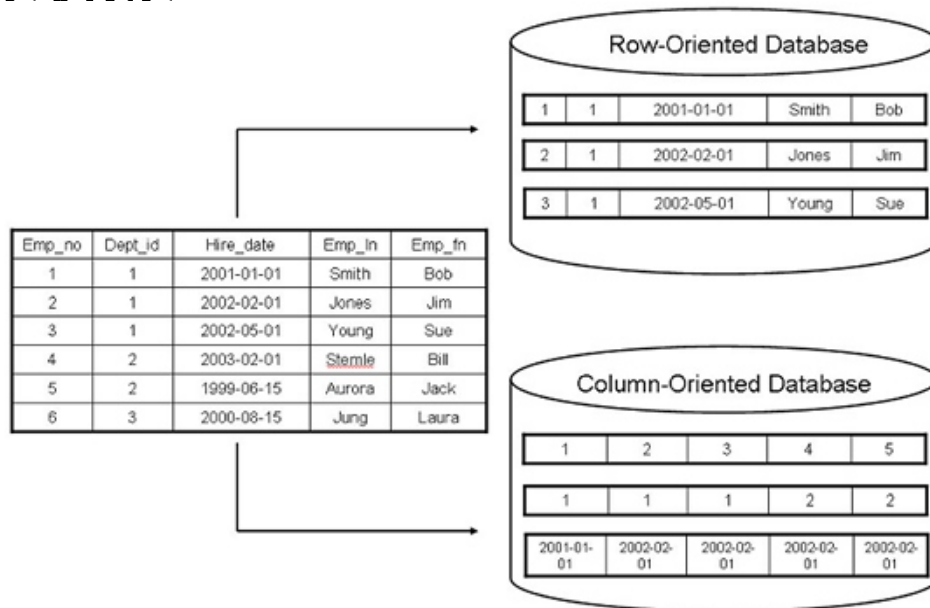
Expiration Date:

Security Code: [help](#)

Attribute	Data Type
Username	Varchar(8)
PaymentMethod	Integer
NameOnCard	Varchar(64)
CardNumber	Integer
ExpirationDate	Date
SecurityCode	Integer

Internal Level

- Determines where and how data are actually stored on the physical storage device



Database Languages

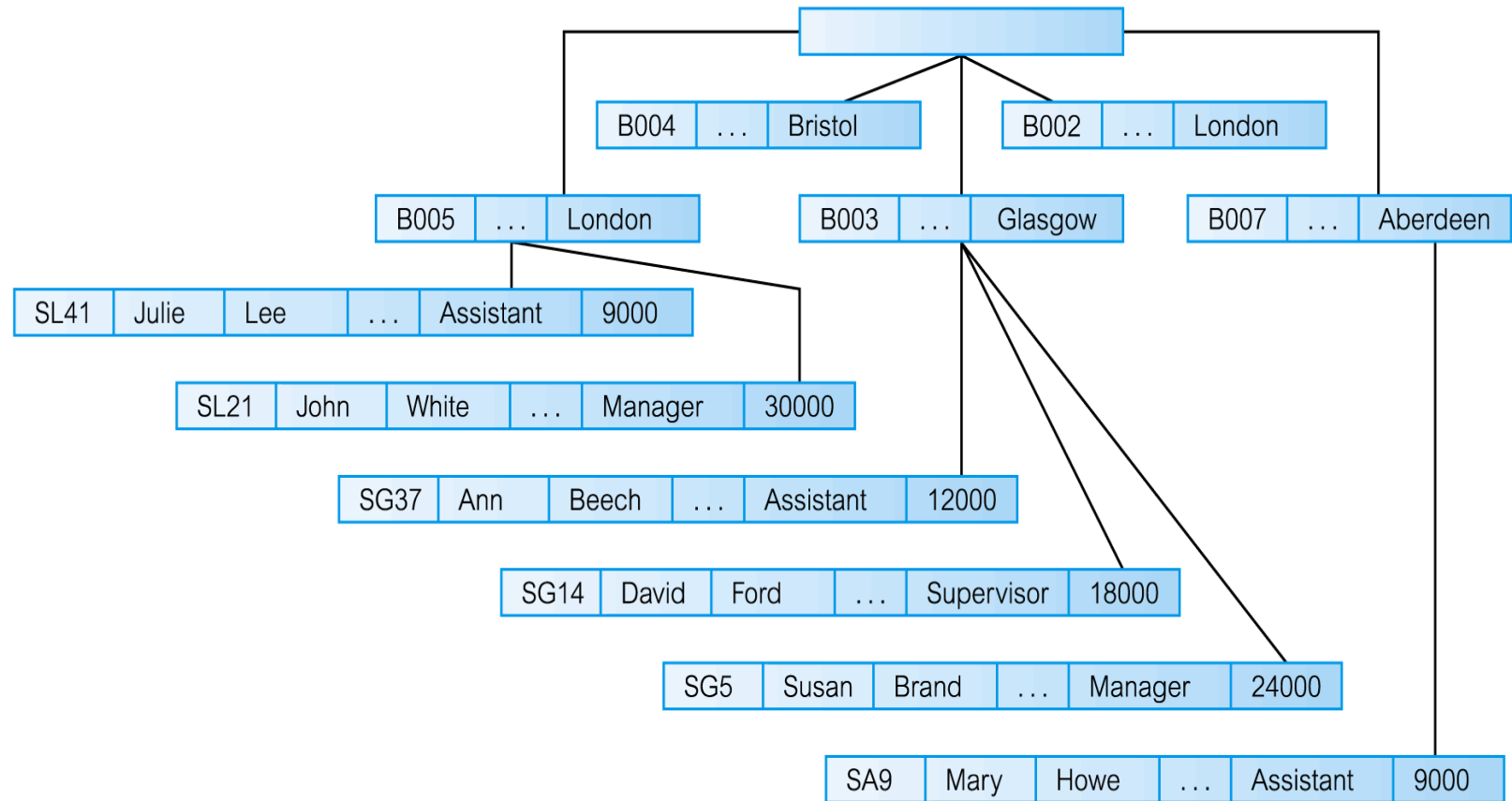
- **DDL** (Data Definition Language)
 - It allows users to define the database.
 - Specify the data type, structure, and constraints on the data to be stored in the database.
- **DML** (Data Manipulation Language)
 - Language for accessing and manipulating the data organized by the appropriate data model
 - DML also known as query language
 - Two classes of languages
 - **Procedural** – user specifies what data is required and how to get those data
 - **Declarative (nonprocedural)** – user specifies what data is required without specifying how to get those data
- Two DB languages will be introduced
 - Relational Algebra (procedural)
 - SQL (Structured Query Language) (declarative)

Database Models

- A database model
 - Defines the logical (conceptual) design of data
 - Hides low-level storage details
 - Describes the relationships between different parts of data
- 3 models as examples
 - Hierarchical model
 - Network model
 - Relational model

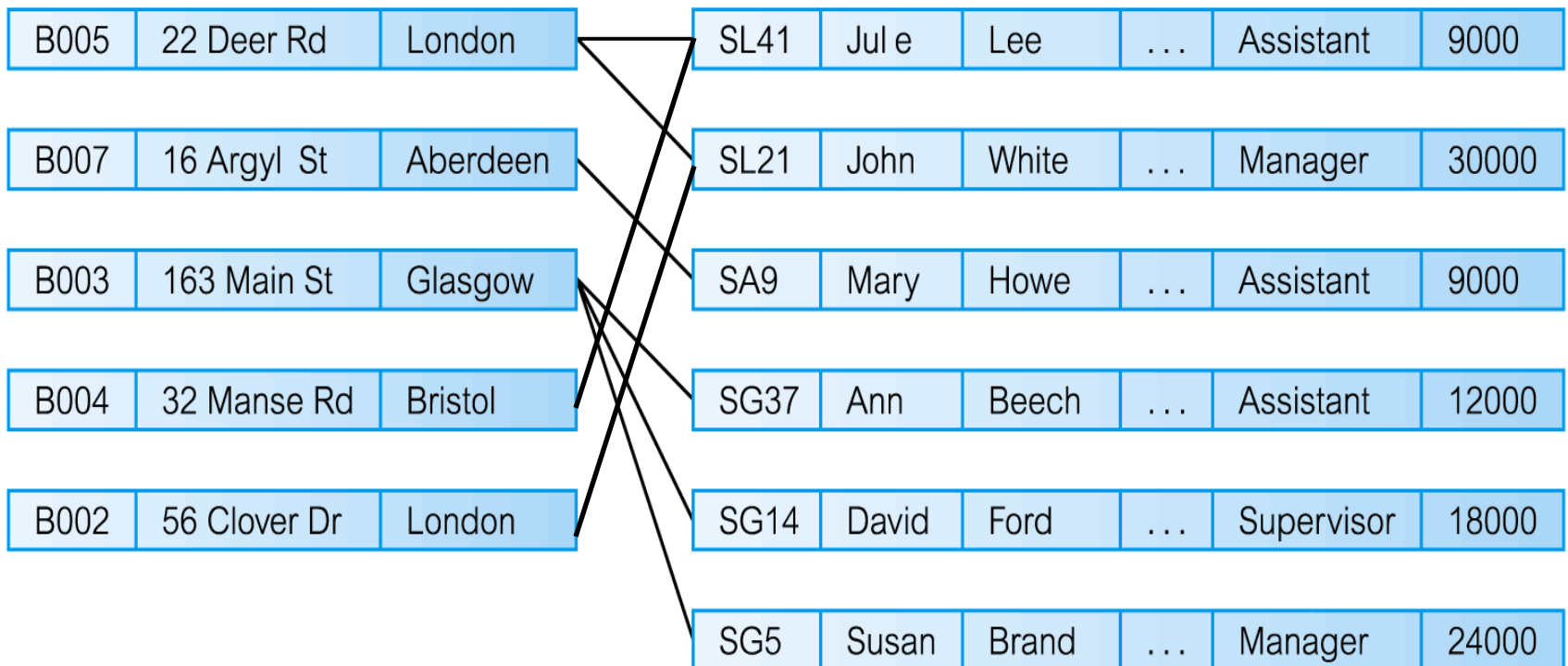
Hierarchical Model

- Data are organized as an upside down tree
- Each entity has only one parent but can have several children



Network Model

- The entities are organized as a graph
- Some entities can be accessed through several paths



Relational Model

- Data are organized as two-dimensional tables called relations
- The tables are related to each other
- The most popular model

Branch

branchNo	street	city	postCode
B005	22 Deer Rd	London	SW1 4EH
B007	16 Argyll St	Aberdeen	AB2 3SU
B003	163 Main St	Glasgow	G11 9QX
B004	32 Manse Rd	Bristol	BS99 1NZ
B002	56 Clover Dr	London	NW10 6EU

Staff

staffNo	fName	lName	position	sex	DOB	salary	branchNo
SL21	John	White	Manager	M	1-Oct-45	30000	B005
SG37	Ann	Beech	Assistant	F	10-Nov-60	12000	B003
SG14	David	Ford	Supervisor	M	24-Mar-58	18000	B003
SA9	Mary	Howe	Assistant	F	19-Feb-70	9000	B007
SG5	Susan	Brand	Manager	F	3-Jun-40	24000	B003
SL41	Julie	Lee	Assistant	F	13-Jun-65	9000	B005

Database Users

- **Users** are differentiated by the way they expect to interact with the system.
 - **End users** – invoke one of the application programs that have been written previously
 - Examples, people accessing database over the web, bank tellers, clerical staff
 - **Application programmers** – interact with system through **DML** calls
 - **Database Administrator (DBA)** – coordinates all the activities of the database system; the DBA has a good understanding of the enterprise's information resources and needs; interact with system through **DDL** and **DML**
 - Schema definition
 - Storage structure and access method definition
 - Schema and physical organization modification
 - Granting user authority to access the database
 - Specifying integrity constraints
 - Monitoring performance and responding to changes in requirements

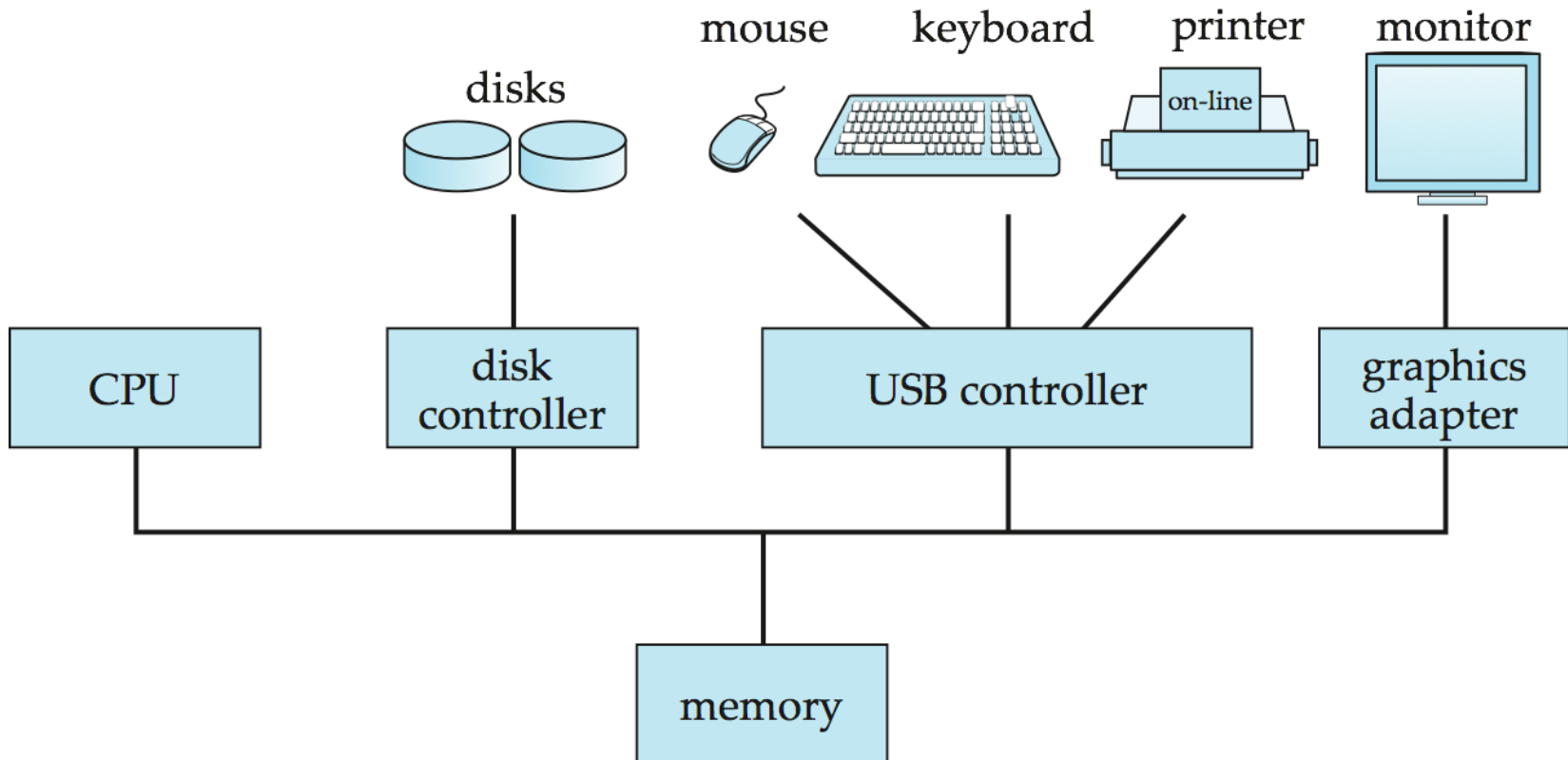
Database Architecture

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

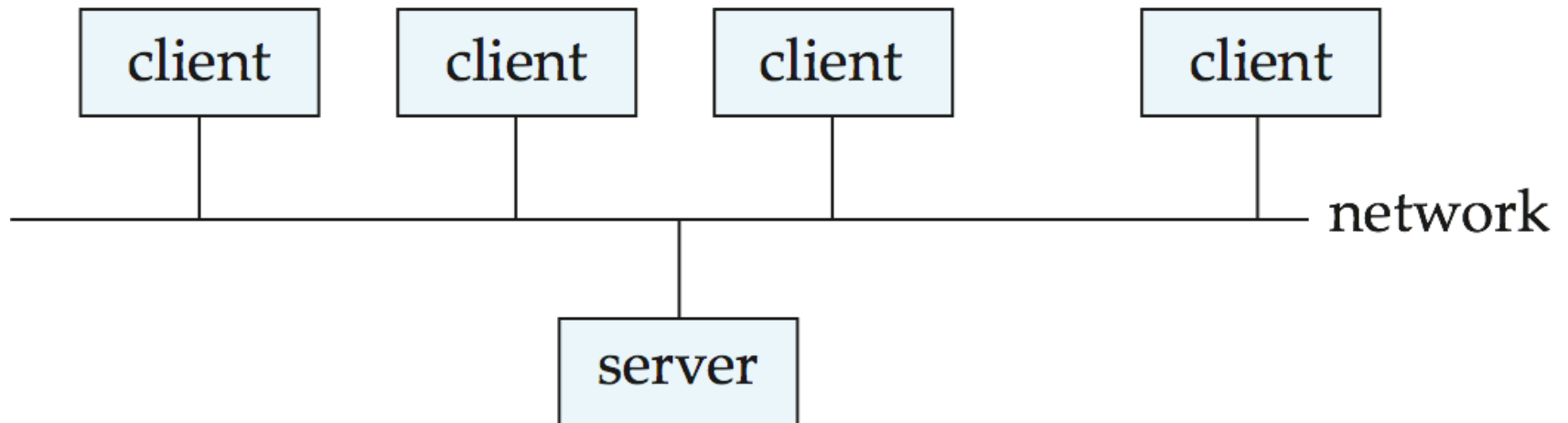
A Centralized Computer System

Run on a single computer system and do **not interact** with other computer systems.



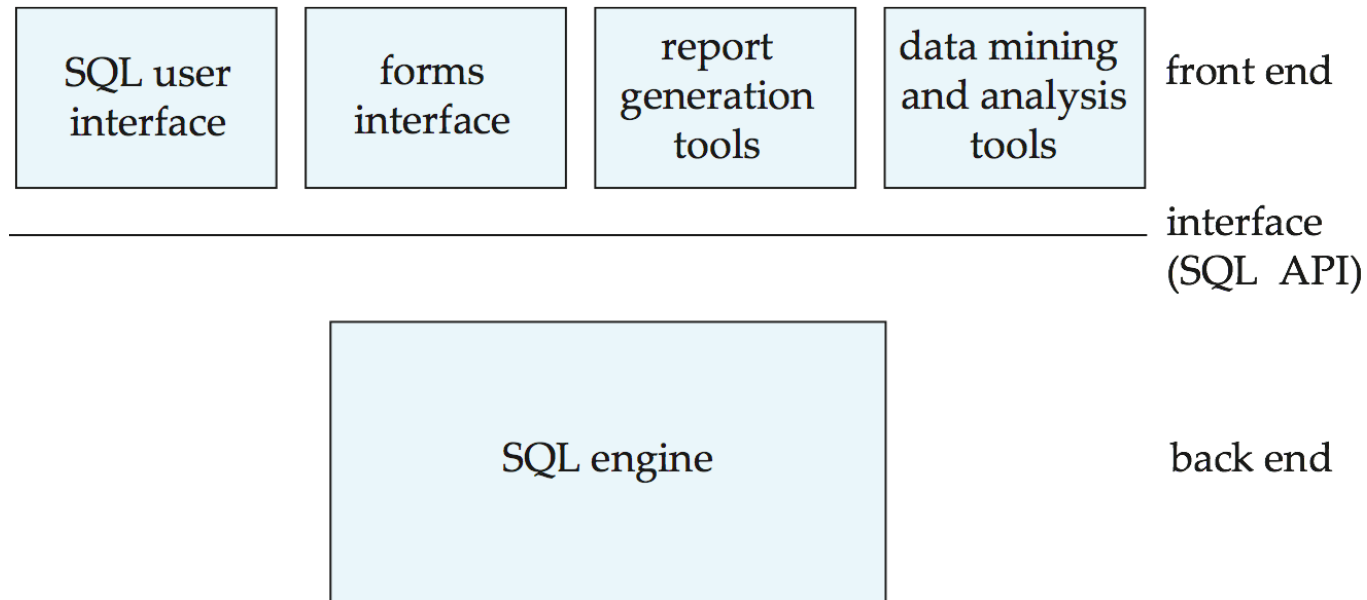
Client-Server Systems

Server systems satisfy requests generated at m client systems, whose general structure is shown below:



Client-Server Systems (Cont.)

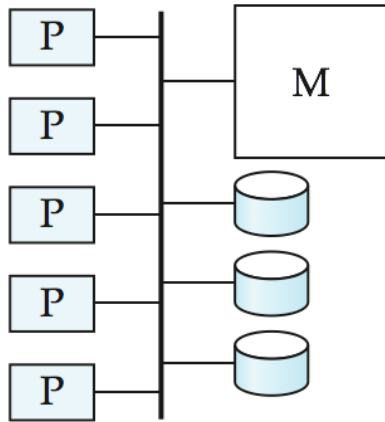
- Database functionality can be divided into:
 - **Back-end**: manages access structures, query evaluation and optimization, concurrency control and recovery.
 - **Front-end**: consists of tools such as *forms*, *report-writers*, and graphical user interface facilities.
- The interface between the front-end and the back-end is through SQL or through an application program interface (API).



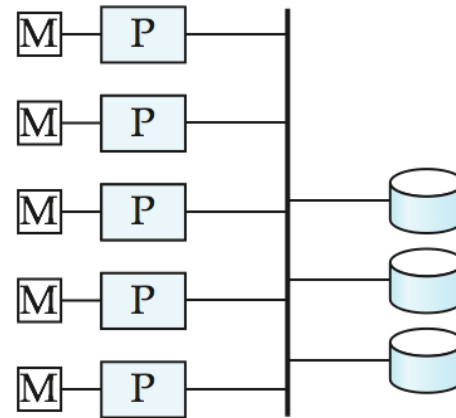
Parallel Database Architectures

- **Shared memory** -- processors share a common memory
- **Shared disk** -- processors share a common disk
- **Shared nothing** -- processors share neither a common memory nor common disk
- **Hierarchical** -- hybrid of the above architectures

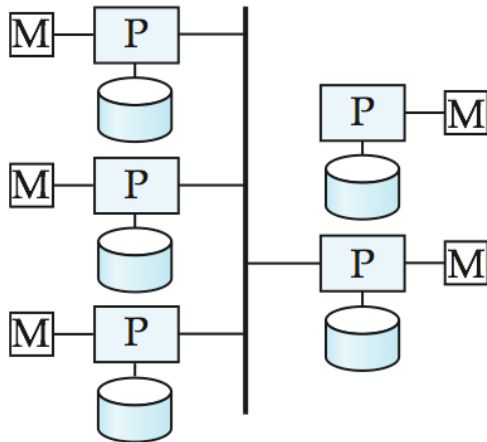
Parallel Database Architectures



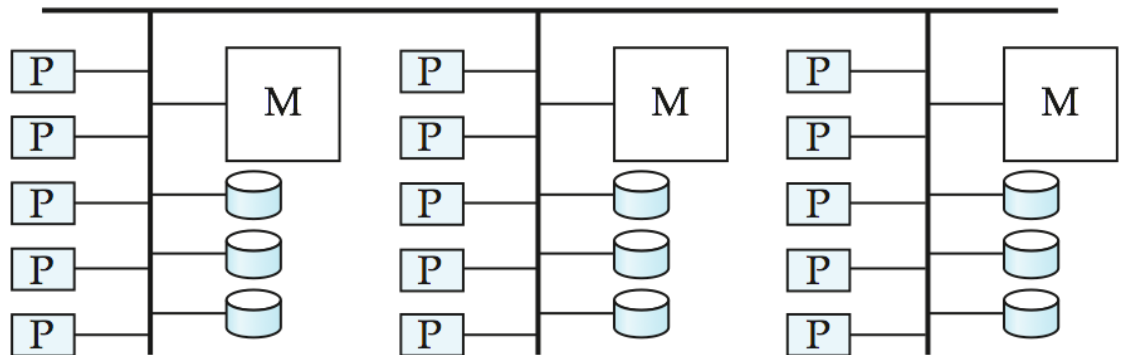
(a) shared memory



(b) shared disk



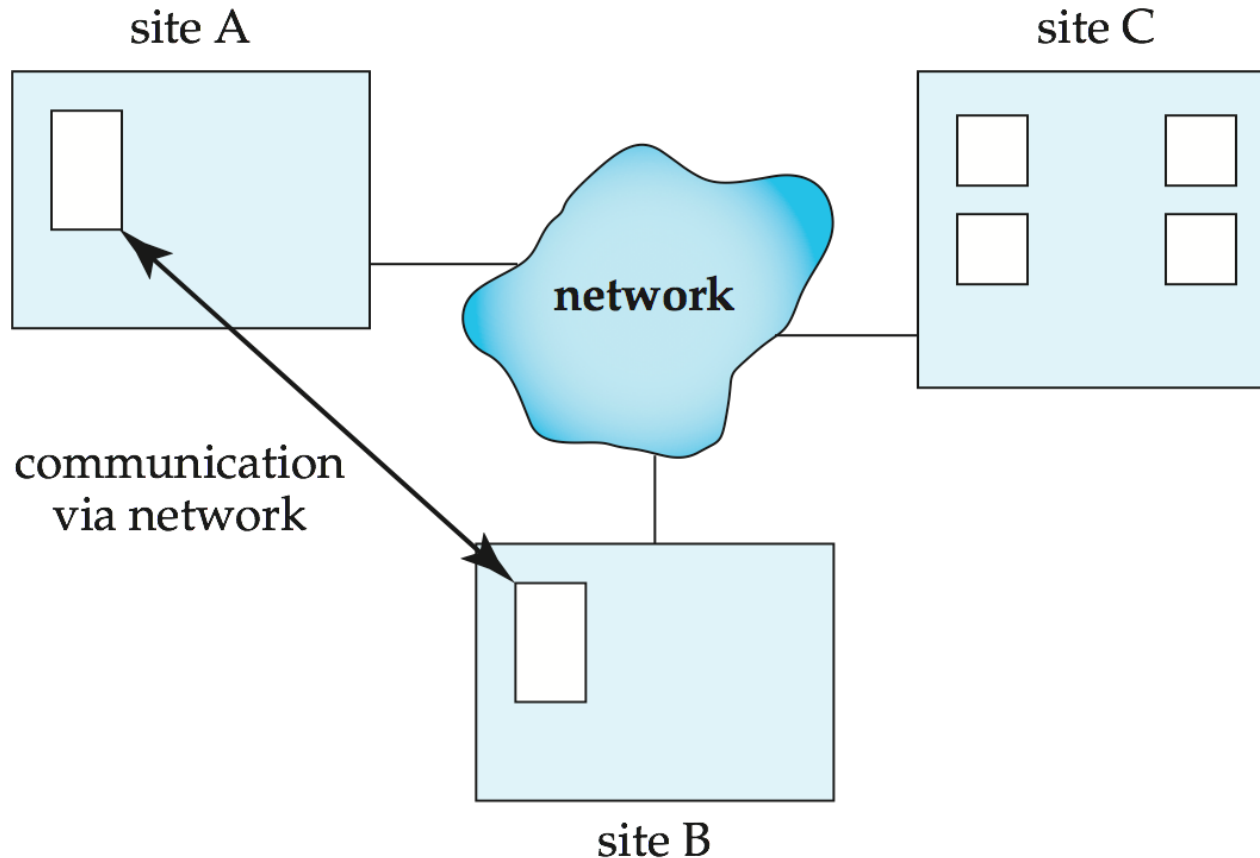
(c) shared nothing



(d) hierarchical

Distributed Systems

- Data spread over multiple machines (also referred to as **sites** or **nodes**).
- Network interconnects the machines
- Data shared by users on multiple machines



History of SQL

- **SQL (Structured Query Language)** is a programming language designed for managing data in relational database management systems (RDBMS).
- SQL was one of the first commercial languages for Edgar F. Codd's relational model, as described in his influential 1970 paper, "A Relational Model of Data for Large Shared Data Banks".
- SQL became a standard of the American National Standards Institute (ANSI) in 1986, and of the International Organization for Standards (ISO) in 1987. (SQL-86/87 or **SQL1**)
- SQL-89 (SQL1.1) slightly revised and expanded SQL 1.
- SQL-92 (**SQL2**) added more features, e.g., with check option for a view, case, and cast expression, built-in join operators, alter tables, subqueries, set operators (union, except, intersect).
- SQL-99 (**SQL3**) incorporated objects and complex data types.
- SQL 2003 included significant extension to the XML parts.
- SQL 2008 added minor changes including BINARY data type.