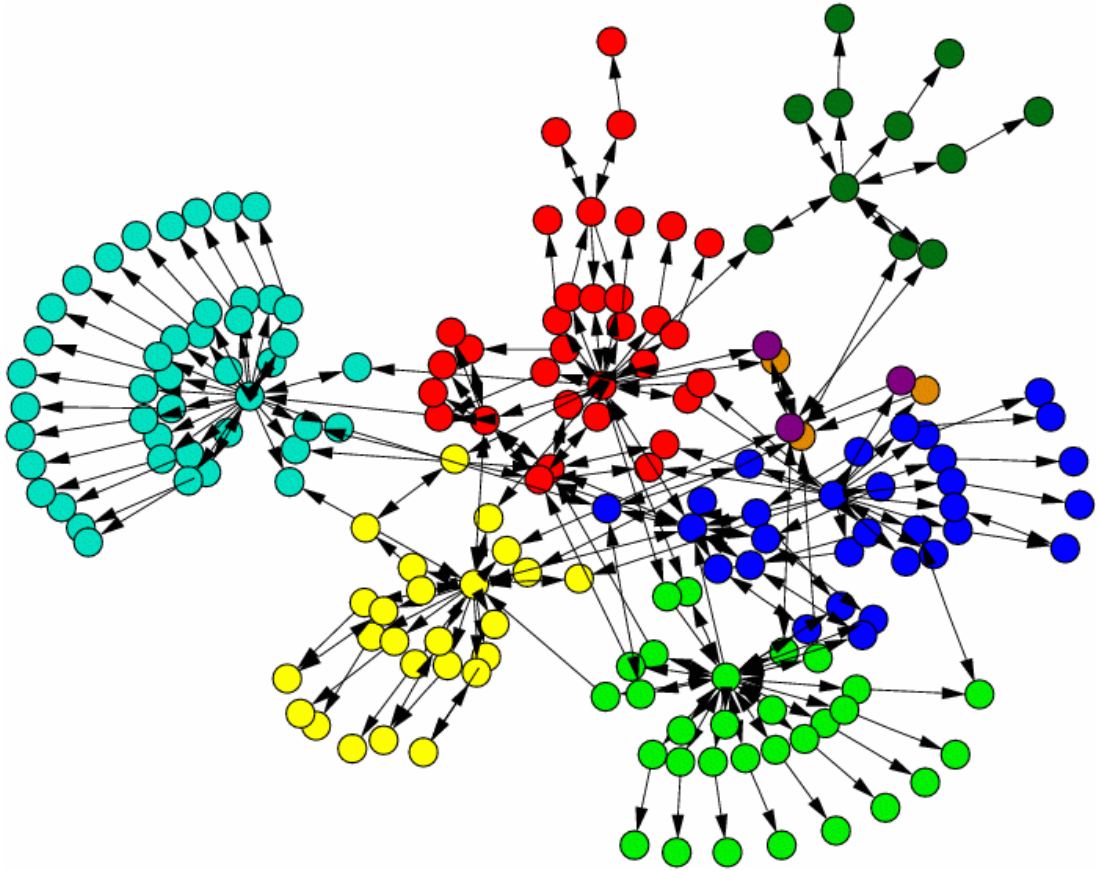
Chapter 12

Information Systems



Chapter Goals

- Define the role of general information systems
- Explain how spreadsheets are organized
- Create spreadsheets for basic analysis of data
- Describe the elements of a database management system
- Describe the organization of a relational database
- Establish relationships among elements in a database
- Write basic SQL statements
- Describe an entity-relationship diagram

Managing Information

- **Information system** Software that helps us organize and analyze data
  - Flexible application software tools that allow the user to dictate and manage the organization of data, and that have basic processing capabilities to analyze the data in various ways
  - Two of the most popular general application information systems are *electronic spreadsheets* and *database management systems*

Spreadsheets

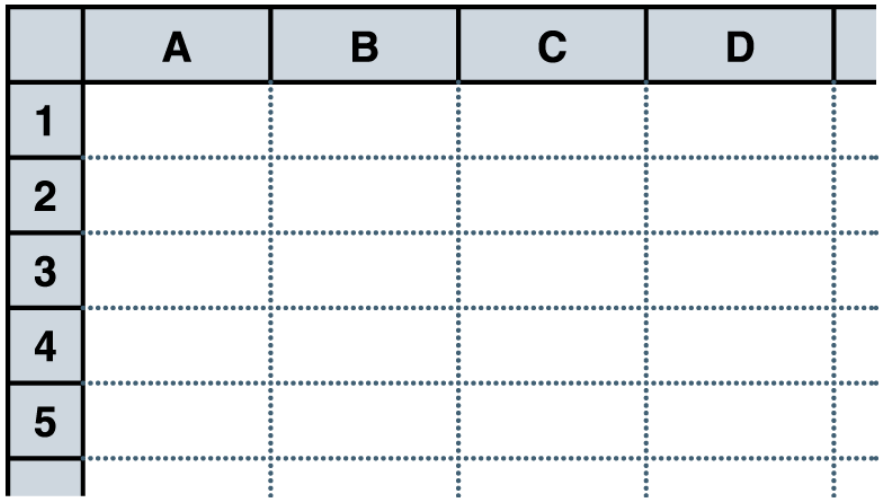


Figure 12.1 **A spreadsheet, made up of a grid of labeled cells**

- **Spreadsheet** A software application that allows the user to organize and analyze data using a grid of labeled **cells**
  - A cell can contain data or a formula that is used to calculate a value
  - Data stored in a cell can be text, numbers, or “special” data such as dates
  - Spreadsheet cells are referenced by their row and column designation

Spreadsheets

- Suppose we have collected data on the number of students that came to get help from a set of tutors over a period of several weeks

	A	B	C	D	E	F	G	H
1								
2				Tutor				
3			Hal	Amy	Frank	Total	Avg	
4		1	12	10	13	35	11.67	
5		2	14	16	16	46	15.33	
6	Week	3	10	18	13	41	13.67	
7		4	8	21	18	47	15.67	
8		5	15	18	12	45	15.00	
9		Total	59	83	72	214	71.33	
10		Avg	11.80	16.60	14.40	42.80	14.27	
11								
12								

Figure 12.1
A spreadsheet containing data and computations

Spreadsheet Formulas

- The power of spreadsheets comes from the formulas that we can create and store in cells
  - When a formula is stored in a cell, the result of the formula is displayed in the cell
  - If we've set up the spreadsheet correctly, we could add or remove tutors, add additional weeks of data, or change any of the data we have already stored and the corresponding calculations would automatically be updated

Spreadsheet Formulas

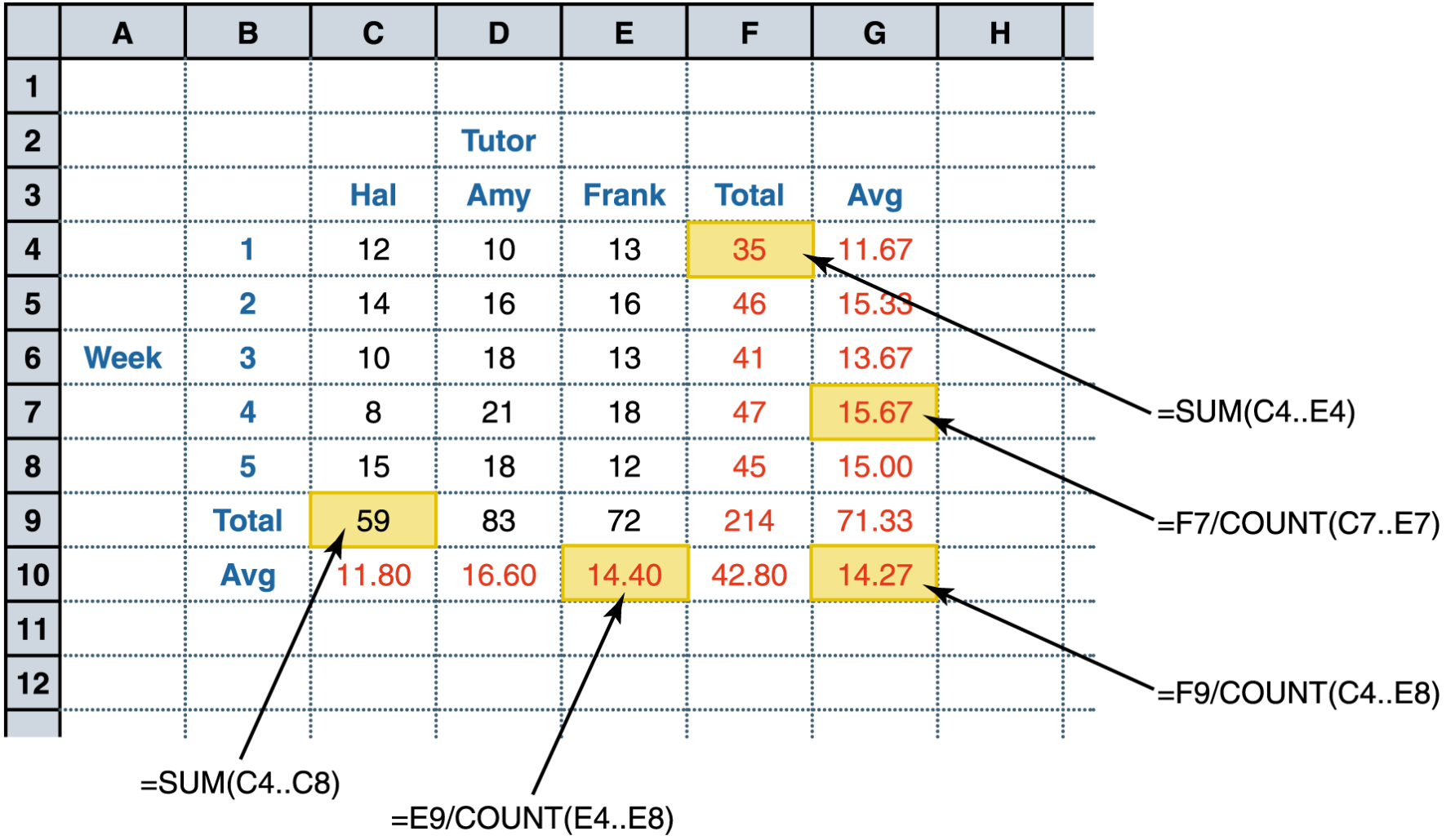


Figure 12.1 **The formulas behind some of the cells**

Spreadsheet Formulas

- Formulas can make use of basic arithmetic operations using the standard symbols (+, 2, *, and /)
- They can also make use of spreadsheet functions that are built into the software
  - Functions often operate on a set of contiguous cells
- A range of cells is specified with two dots (periods) between the two cell endpoints

Spreadsheet Formulas

Function	Computes
SUM(val1, val2, ...) SUM(range)	Sum of the specified set of values
COUNT(val1, val2, ...) COUNT(range)	Count of the number of cells that contain values
MAX(val1, val2, ...) MAX(range)	Largest value from the specified set of values
SIN(angle)	The sine of the specified angle
PI()	The value of PI
STDEV(val1, val2, ...) STDEV(range)	The standard deviation from the specified sample values
TODAY()	Today's date
LEFT(text, num_chars)	The leftmost characters from the specified text
IF(test, true_val, false_val)	If the test is true, it returns the true_val; otherwise, it returns the false_val
ISBLANK (value)	Returns true if the specified value refers to an empty cell

Figure 12.4 **Some common spreadsheet functions**

Circular References

- A circular reference can never be resolved because the result of one formula is ultimately based on another, and vice versa

Cell	Contents
A1	=B7*COUNT(F8..K8)
B7	=A14+SUM(E40..E50)
E45	=G18+G19–D13
D13	=D12/A1

Figure 12.5 **A circular reference situation that cannot be resolved**

Spreadsheet Analysis

- One reason spreadsheets are so useful is their versatility
- Spreadsheet analysis can be applied to just about any topic area
  - Track sales
  - Analyze sport statistics
  - Maintain student grades
  - Keep a car maintenance log
  - Record and summarize travel expenses
  - Track project activities and schedules
  - Plan stock purchases

Spreadsheet Analysis

- Spreadsheets are also useful because of their *dynamic nature*, which provides the powerful ability to do *what-if analysis*
  - *What if the number of attendees decreased by 10%?*
  - *What if we increase the ticket price by $5?*
  - *What if we could reduce the cost of materials by half?*

Database Management Systems

- **Database** A structured set of data
- **Database management system** (DBMS) A combination of software and data, including a physical database, a database engine, and a database schema
  - **Physical database** A collection of files that contain the data
  - **Database engine** Software that supports access to and modification of the database contents
  - **Database schema** A specification of the logical structure of the data stored in the database

Database Management Systems

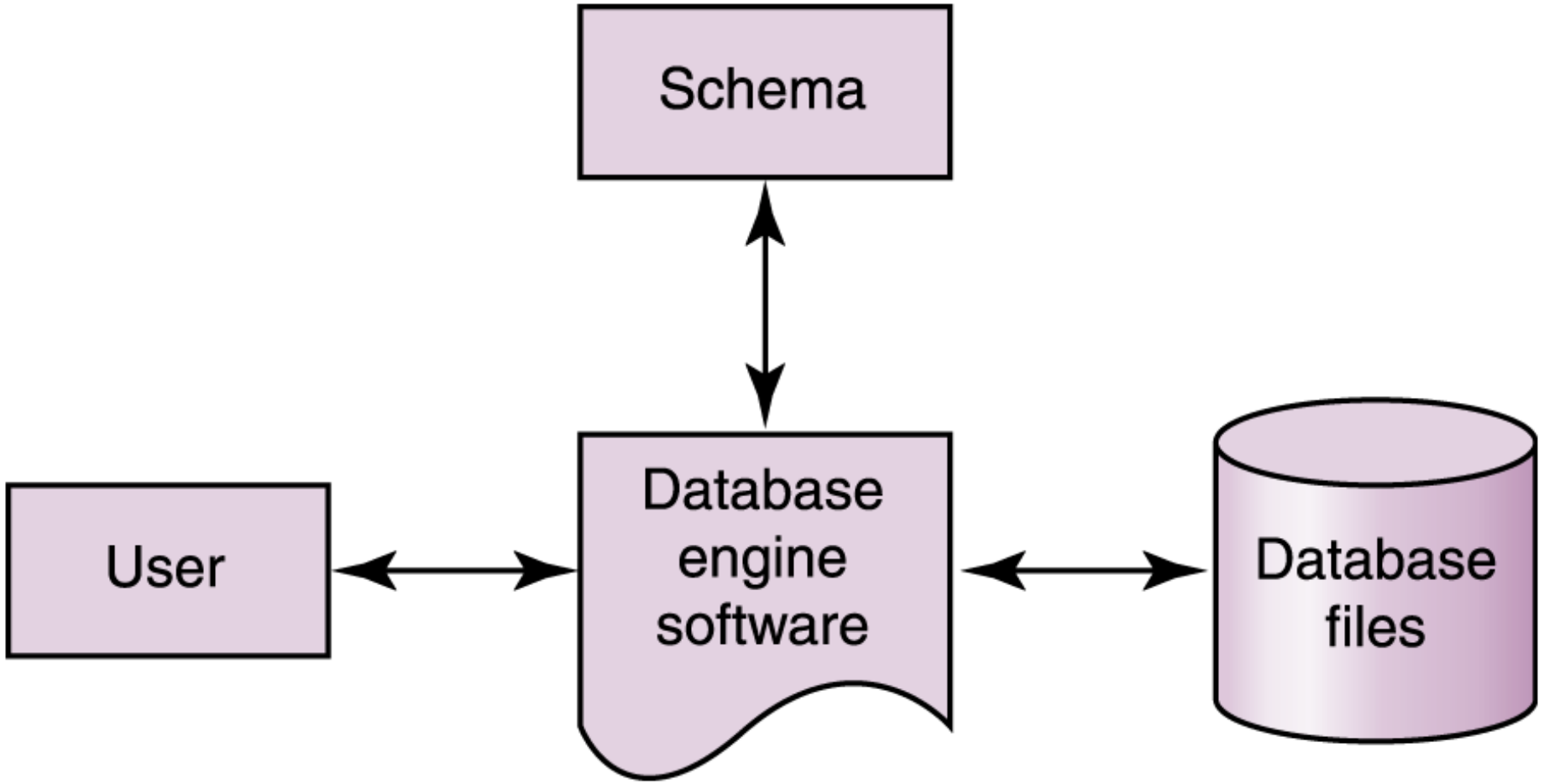


Figure 12.6 **The elements of a database management system**

Database Management Systems

- Specialized database languages allow the user to specify the structure of data; add, modify, and delete data; and **query** the database to retrieve specific stored data
- The database **schema** provides the logical view of the data in the database

The Relational Model

- In a relational DBMS, the data items and the relationships among them are organized into **tables**
  - A table is a collection of **records** - rows
  - A record is a collection of related **fields** - cols
  - Each field of a database table contains a single data value
  - Each record in a table contains the same fields

A Database Table

Movie

MovieId	Title	Genre	Rating
101	Sixth Sense, The	thriller horror	PG-13
102	Back to the Future	comedy adventure	PG
103	Monsters, Inc.	animation comedy	G
104	Field of Dreams	fantasy drama	PG
105	Alien	sci-fi horror	R
106	Unbreakable	thriller	PG-13
107	X-Men	action sci-fi	PG-13
5022	Elizabeth	drama period	R
5793	Independence Day	action sci-fi	PG-13
7442	Platoon	action drama war	R

Figure 12.7 **A database table, made up of records and fields**

A Database Table

- We can express the schema for this part of the database as follows:

  Movie (MovieId:key, Title, Genre, Rating)

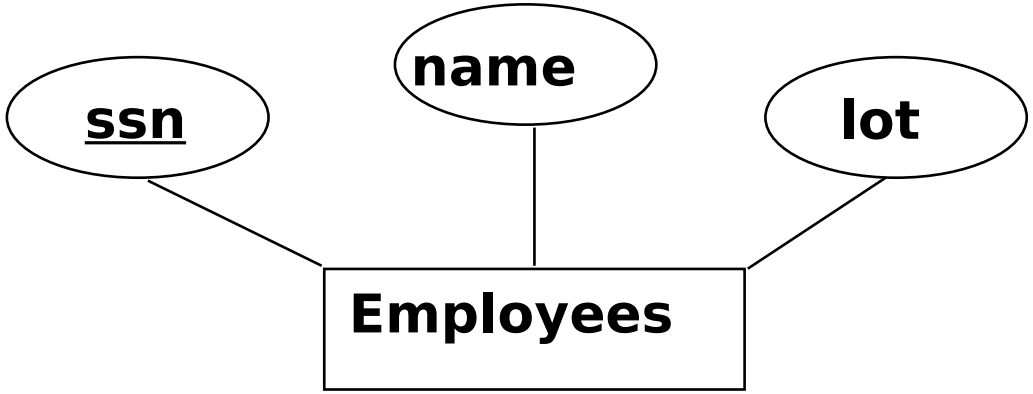
Relationships

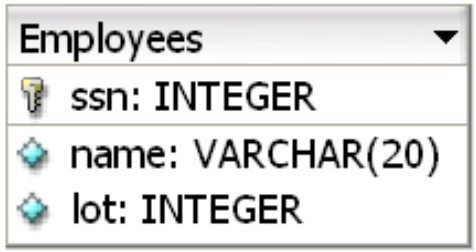
Customer

CustomerId	Name	Genre	CreditCardNumber
101	Dennis Cook	123 Main Street	2736 2371 2344 0382
102	Doug Nickle	456 Second Ave	7362 7486 5957 3638
103	Randy Wolf	789 Elm Street	4253 4773 6252 4436
104	Amy Stevens	321 Yellow Brick Road	9876 5432 1234 5678
105	Robert Person	654 Lois Lane	1122 3344 5566 7788
106	David Coggin	987 Broadway	8473 9687 4847 3784
107	Susan Klaton	345 Easy Street	2435 4332 1567 3232

Figure 12.8 A database table containing customer data

ER Model Basics





ssn	name	lot
123-22-3666	Attishoo	48
231-31-5368	Smiley	22
131-24-3650	Smethurst	35

Relationships

- We can use a table to represent a collection of relationships between objects

Rents

CustomerId	MovieId	DateRented	DateDue
103	104	3-12-2002	3-13-2002
103	5022	3-12-2002	3-13-2002
105	107	3-12-2002	3-15-2002

Figure 12.9 A database table storing current movie rentals

Relational Query Languages

- A major strength of the relational model: supports simple, powerful *querying* of data.
- Queries can be written intuitively, and the DBMS is responsible for efficient evaluation.
  - The key: precise semantics for relational queries.
  - Allows the optimizer to extensively re-order operations, and still ensure that the answer does not change.

Structured Query Language

- **Structured Query Language (SQL)** A comprehensive database language for managing relational databases
- Developed by IBM (system R) in the 1970s
- Need for a standard since it is used by many vendors
- Standards:
  - SQL-86
  - SQL-99 (major extensions, current standard)

Queries in SQL

select *attribute-list* from *table-list* where *condition*

select Title from Movie where Rating = 'PG'

select Name, Address from Customer

select * from Movie where Genre like '%action%'

select * from Movie where Rating = 'R' order by Title

Querying Multiple Relations

- What does the following query compute?

```
SELECT   S.name, E.cid
FROM     Students S, Enrolled E
WHERE    S.sid=E.sid AND E.grade="A"
```

Given the following instances of Enrolled and Students:

sid	name	login	age	gpa
53666	Jones	jones@cs	18	3.4
53688	Smith	smith@eecs	18	3.2
53650	Smith	smith@math	19	3.8

sid	cid	grade
53831	Carnatic101	C
53831	Reggae203	B
53650	Topology112	A
53666	History105	B

we get:

S.name	E.cid
Smith	Topology112

Modifying Database Content

```
insert into Customer values (9876, 'John
   Smith', '602 Greenbriar Court', '2938
   3212 3402 0299')

update Movie set Genre = 'thriller drama'
   where title = 'Unbreakable'

delete from Movie where Rating = 'R'
```

Database Design

- **Entity-relationship (ER) modeling** A popular technique for designing relational databases
- **ER Diagram** Chief tool used for ER modeling that captures the important record types, attributes, and relationships in a graphical form

Database Design

- These designations show the **cardinality constraint** of the relationship

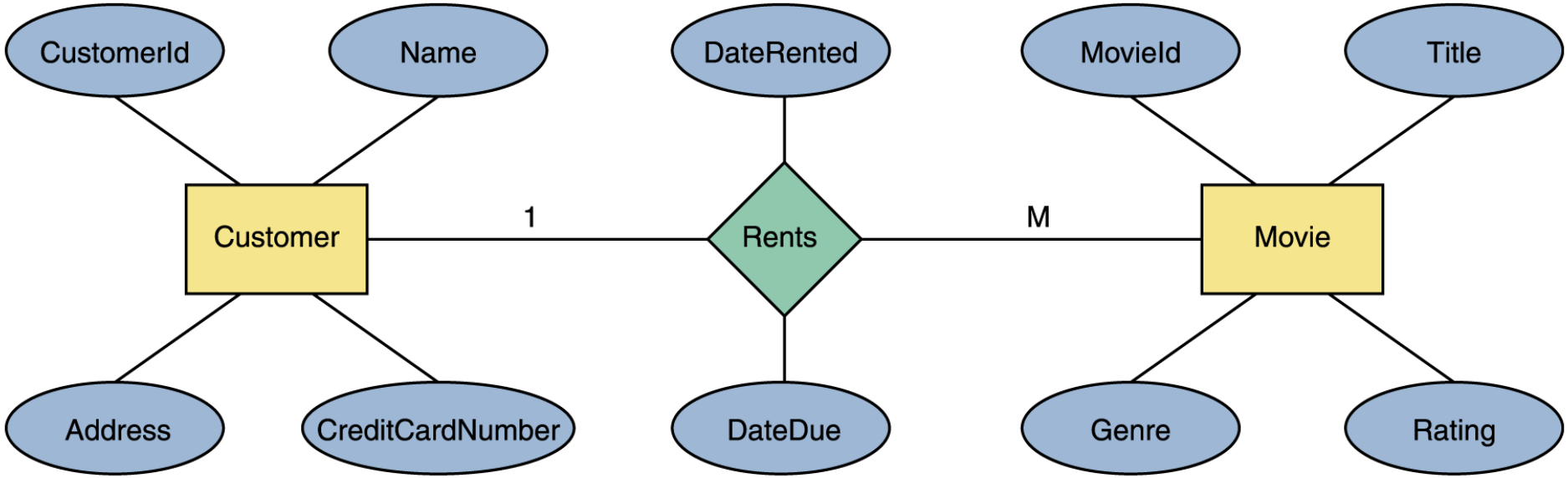


Figure 12.10 **An ER diagram for the movie rental database**

Components of Data-Intensive Systems

Three separate types of functionality:

- **Presentation**
- **Application logic**
- **Data management**

- The system architecture determines whether these three components reside on a single system tier or are distributed across several tiers

The Three Layers

Presentation tier

- Primary interface to the user
- Needs to adapt to different display devices (PC, PDA, cell phone, voice access?)

Middle tier

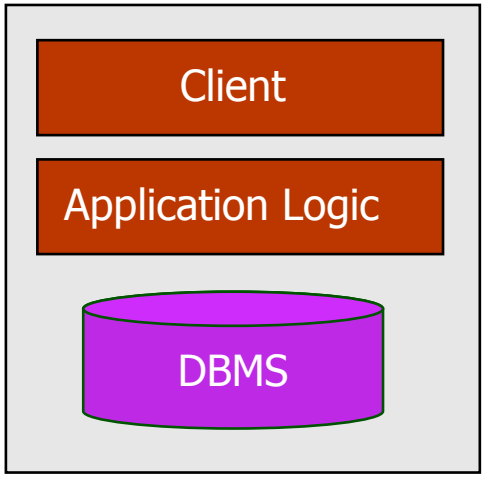
- Implements business logic (implements complex actions, maintains state between different steps of a workflow)
- Accesses different data management systems

Data management tier

- One or more standard database management systems

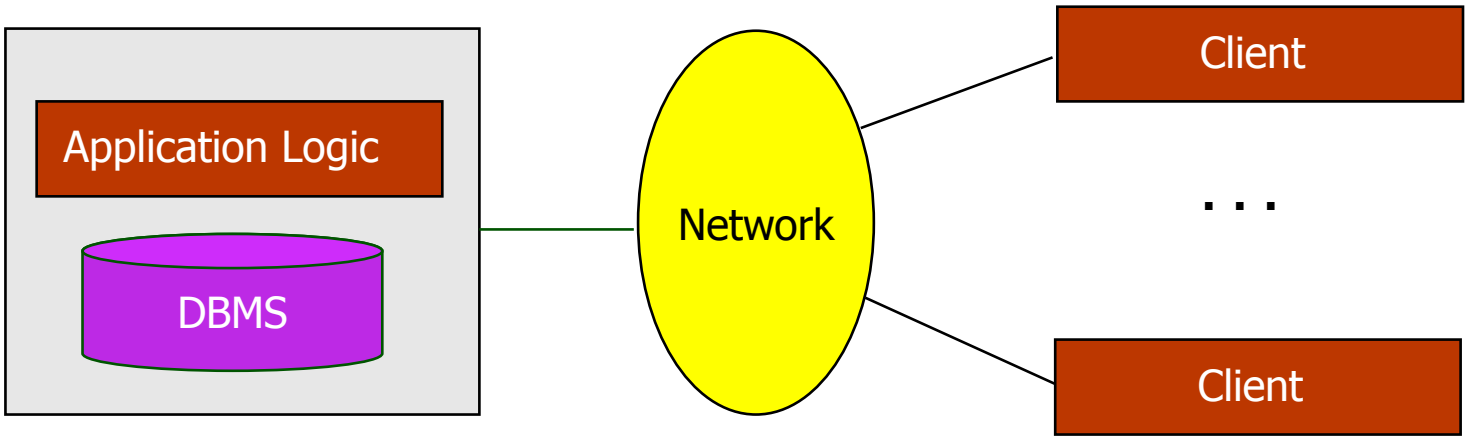
Single-Tier Architectures

- All functionality combined into a single tier, usually on a mainframe
  - User access through dumb terminals
- Advantages:
  - Easy maintenance and administration
- Disadvantages:
  - Today, users expect graphical user interfaces.
  - Centralized computation of all of them is too much for a central system



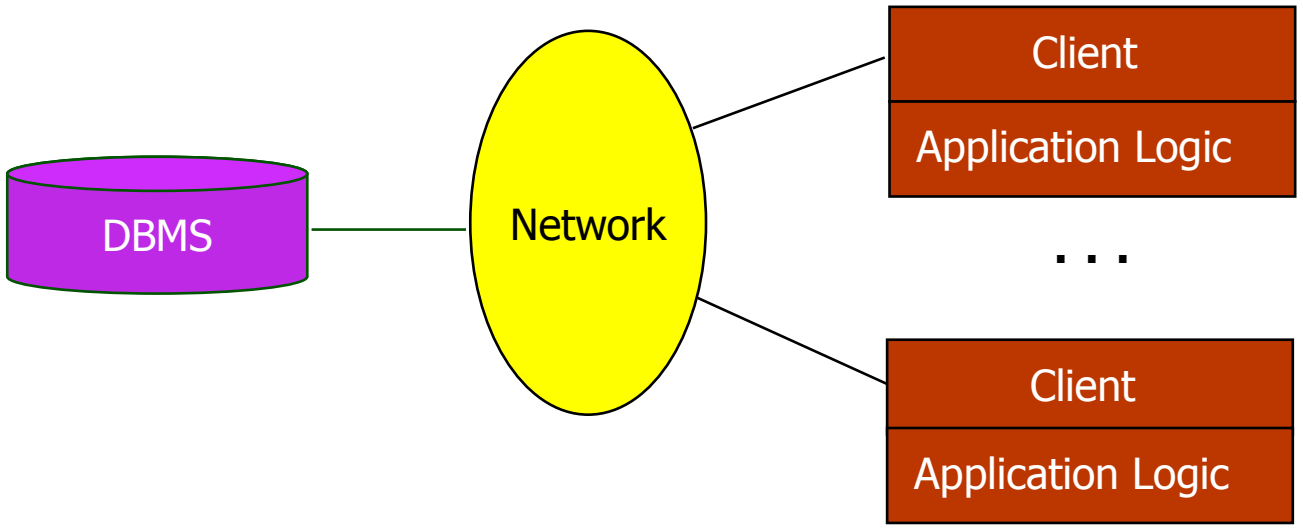
Client-Server Architectures

- Work division: **Thin client**
  - Client implements only the graphical user interface
  - Server implements business logic and data management

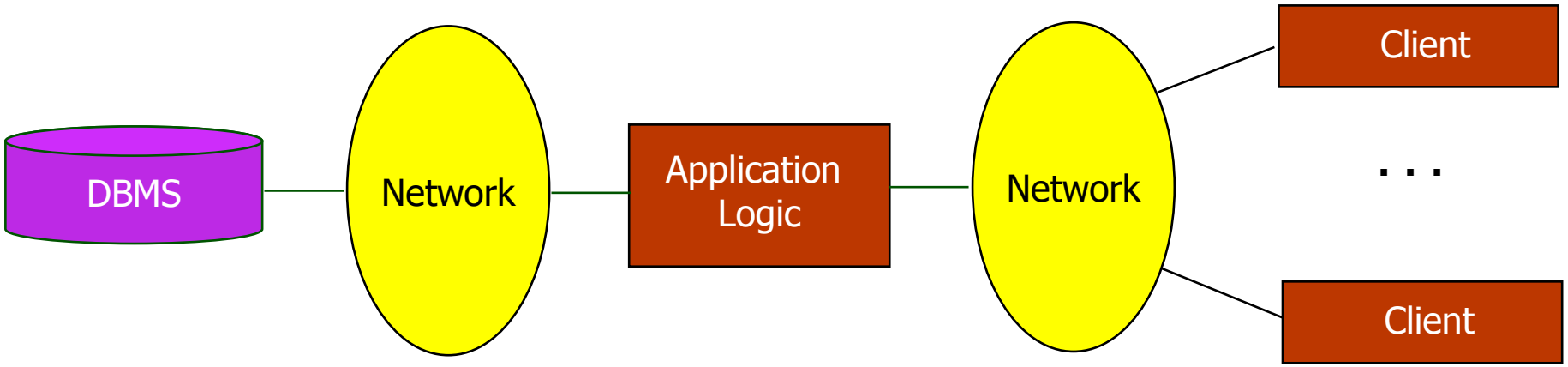


Client-Server Architectures

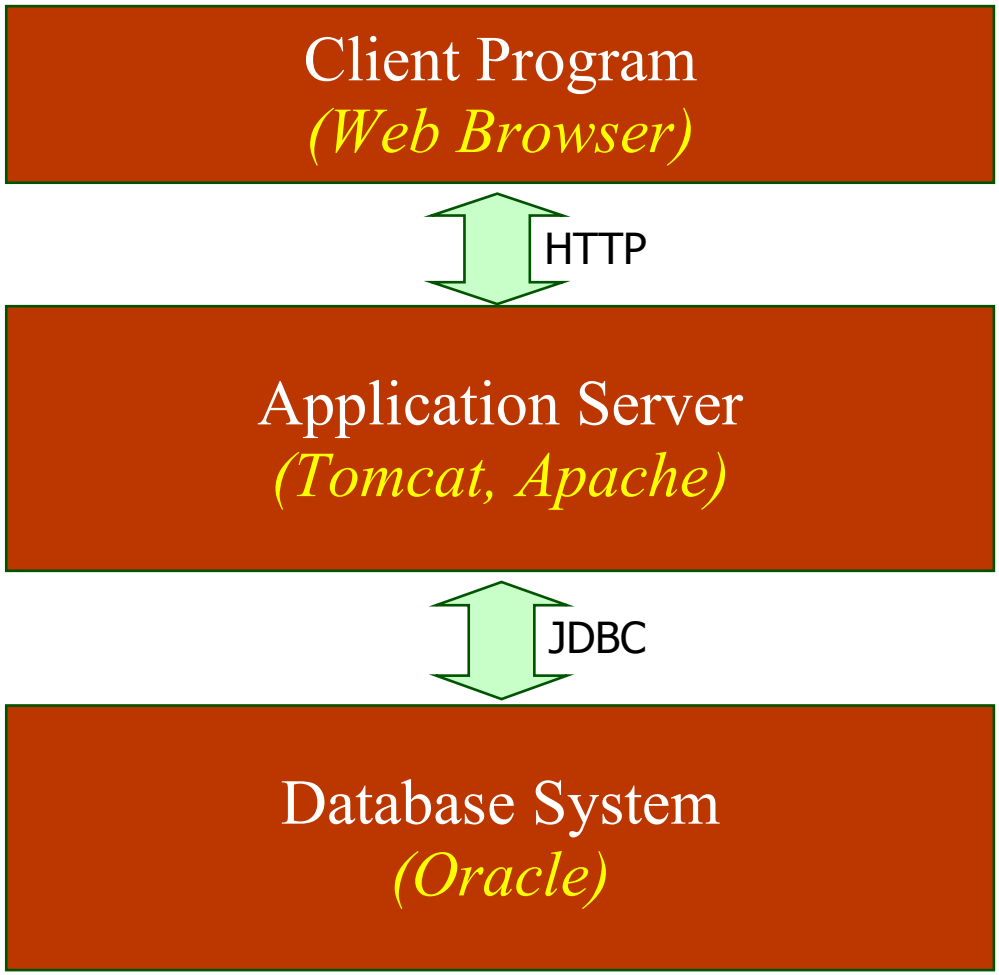
- Work division: **Thick client**
  - Client implements both the graphical user interface and the business logic
  - Server implements data management



Three-Tier Architecture



Technologies



HTML
Javascript
XSLT

JSP
Servlets
Cookies
CGI

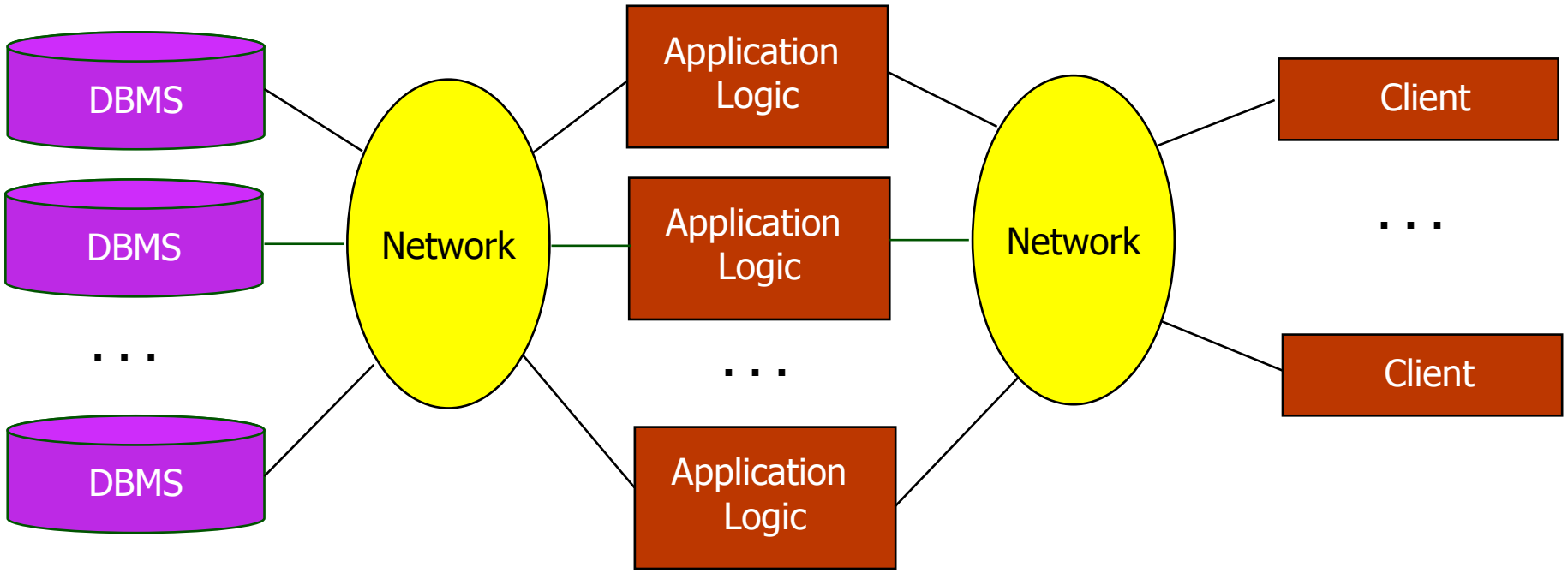
XML

Stored Procedures

Advantages: 3-Tier Architecture

- **Heterogeneous systems**
  - Tiers can be independently maintained, modified, and replaced
- **Thin clients**
  - Only presentation layer at clients (web browsers)
- **Integrated data access**
  - Several database systems can be handled transparently at the middle tier
  - Central management of connections
- **Scalability**
  - Replication at middle tier permits scalability of business logic
- **Software development**
  - Code for business logic is centralized
  - Interaction between tiers through well-defined APIs: Can reuse standard components at each tier

Scalable Three-Tier Architecture



Homework

- **Read Chapter Twelve** - Concentrate on Section 12.3
- **Program Assignment #2** ***– Let Me Know If You Are Having Trouble***
- **Assignment Due 11/20** – but you can email before :-)
- **Workshop Class On 11/20** – program and any other problems

Have A Nice Weekend