Introduction to Databases

Adapted from the slides of : Database systems concepts (Silberschatz et al.) and Database Management Systems (Ramakrishnan et al.)

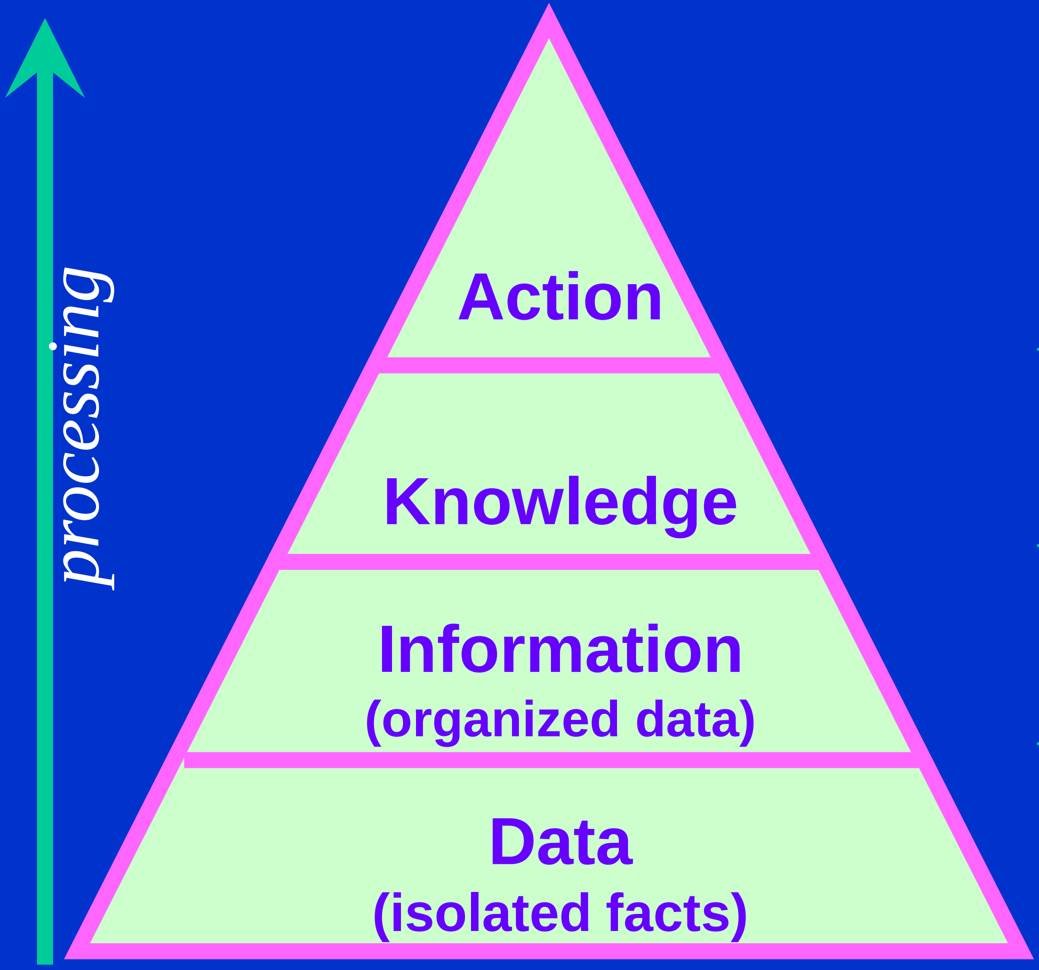
Class Outline

- Introduction
- Database management system
- History of Database
- Levels of database representation?
- Database models
- Features of a DBMS
- Query Languages
- Database Design
- Database Engine
- Types of DBMS

Principles of Information Resource Management

- Organizational resources flow into and out of the organization
- Two types of major organizational resources: Physical resources, Conceptual resources (data & information)
- As scale of organization grows, it becomes increasingly difficult to manage by observation (i.e., reliance on conceptual resources)
- Conceptual resources can be managed just like physical resources or assets (e.g., employees, $$, equipment, widgets, etc.)
- Management of data & information means getting it before it’s needed, protecting it, assuring quality, and getting rid of it when no longer required
- Management of data & information can be achieved only through organizational commitment

Information is a major organizational resource



Survey customers; invest in advertising; cut costs, expand product line

Sales have dropped between July and August

Average/ July is 40

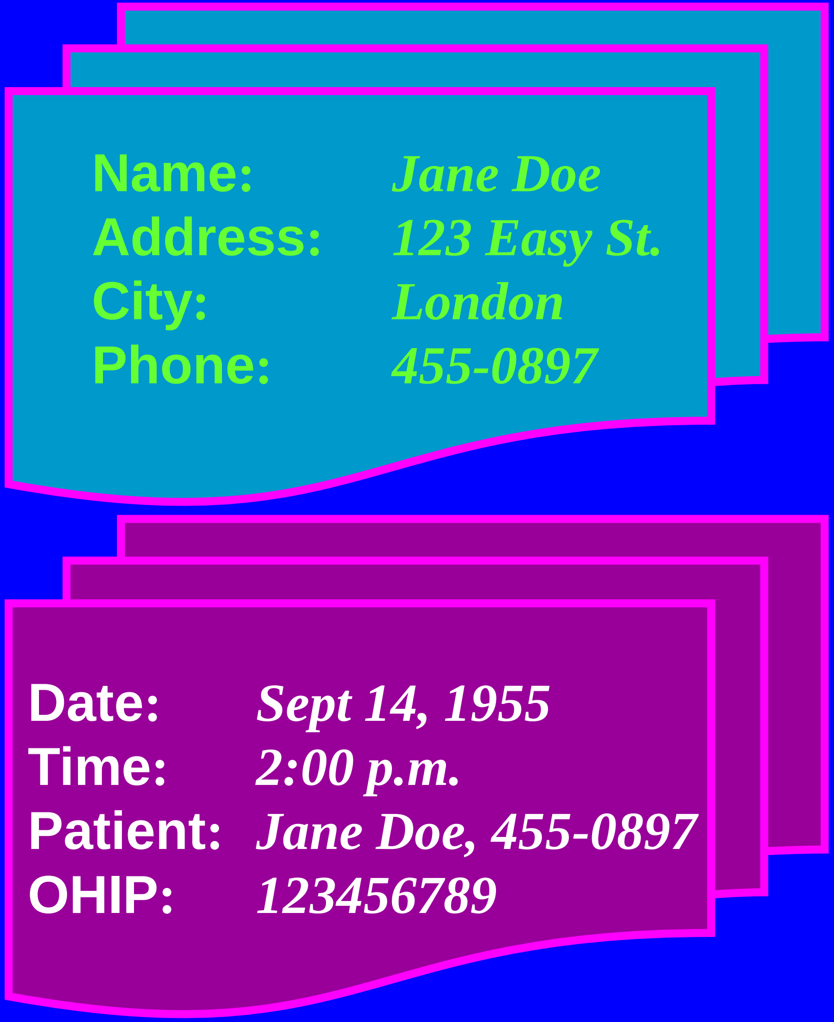
Average/ Aug is 15

John bought 50 in July
John bought 10 in Aug

Jane bought 30 in July
Jane bought 20 in Aug

In the beginning…(in the 1950s)

…There were no databases. Just file (or data processing) systems.

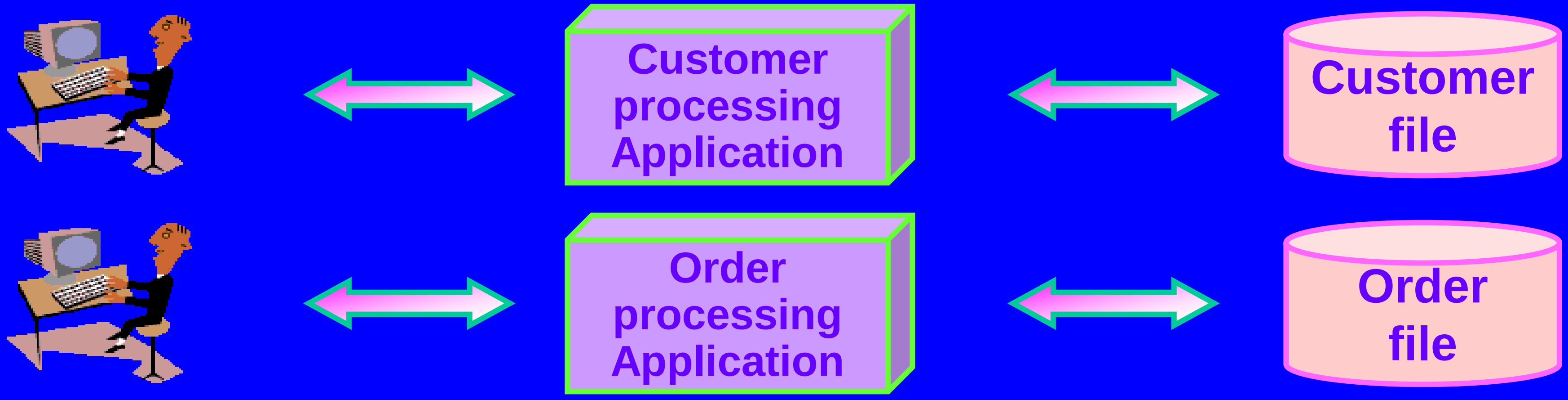


Name:	*Jane Doe*
Address:	*123 Easy St.*
City:	*London*
Phone:	*455-0897*

Date:	*Sept 14, 1955*
Time:	*2:00 p.m.*
Patient:	*Jane Doe, 455-0897*
OHIP:	*123456789*

- File systems were typically organized by function (use)
- The first data management systems performed clerical tasks (transactional processing) such as order entry processing, payroll, work scheduling.
- e.g., files for patients (file folder analogy); each record for a single patient; another file for appointment/ billing information

Limitations of Data File Systems



- Worked adequately if data collection needs were relatively small.
- Problems arose as data files, information needs, and reporting requirements grow in complexity due to:
  - Extensive programming - use of third-generation languages (e.g., COBOL, FORTRAN) in which the programmer must specify *what* is be done as well as *how* it is to be done

Drawbacks of using file systems to store data

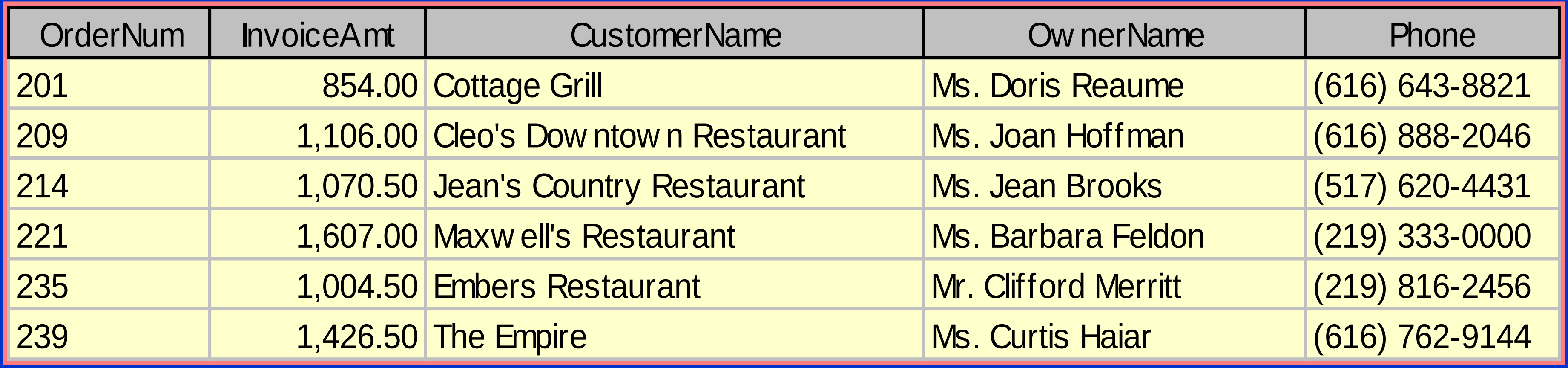
- Data redundancy and inconsistency
  - Multiple file formats, duplication of information in different files
- Difficulty in accessing data
  - Need to write a new program to carry out each new task
- Data isolation
  - Multiple files and formats
- Integrity problems
  - Integrity constraints (e.g., account balance > 0) become "buried" in program code rather than being stated explicitly
  - Hard to add new constraints or change existing ones

Drawbacks of using file systems to store data (Cont.)

- Atomicity of updates
  - Failures may leave database in an inconsistent state with partial updates carried out
  - Example: Transfer of funds from one account to another should either complete or not happen at all
- Concurrent access by multiple users
  - Concurrent access needed for performance
  - Uncontrolled concurrent accesses can lead to inconsistencies
    - Example: Two people reading a balance (say 100) and updating it by withdrawing money (say 50 each) at the same time
- Security problems
  - Hard to provide user access to some, but not all, data

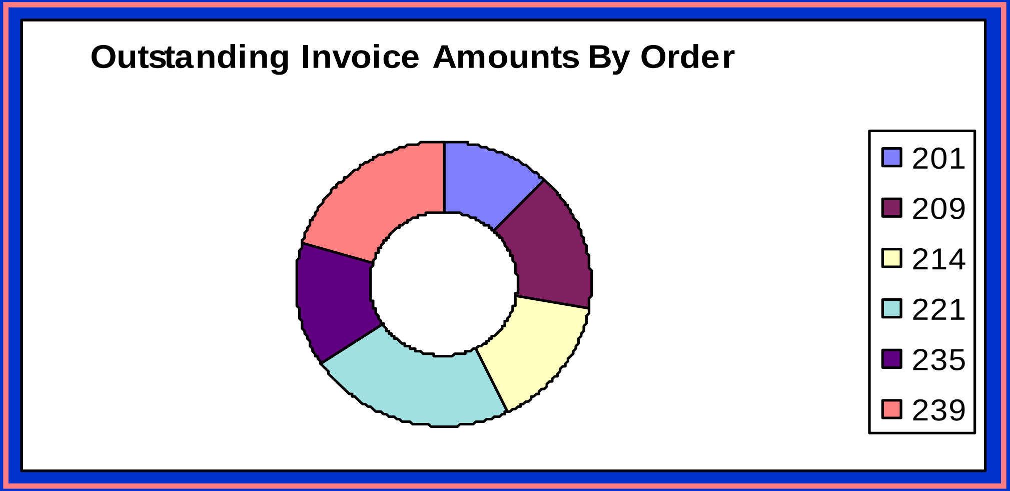
What is a Database?

Organized collection of related information or data stored on a computer disk for easy, efficient use



OrderNum	InvoiceAmt	CustomerName	OwnerName	Phone
201	854.00	Cottage Grill	Ms. Doris Reaume	(616) 643-8821
209	1,106.00	Cleo's Downtown Restaurant	Ms. Joan Hoffman	(616) 888-2046
214	1,070.50	Jean's Country Restaurant	Ms. Jean Brooks	(517) 620-4431
221	1,607.00	Maxwell's Restaurant	Ms. Barbara Feldon	(219) 333-0000
235	1,004.50	Embers Restaurant	Mr. Clifford Merritt	(219) 816-2456
239	1,426.50	The Empire	Ms. Curtis Haiar	(616) 762-9144

data

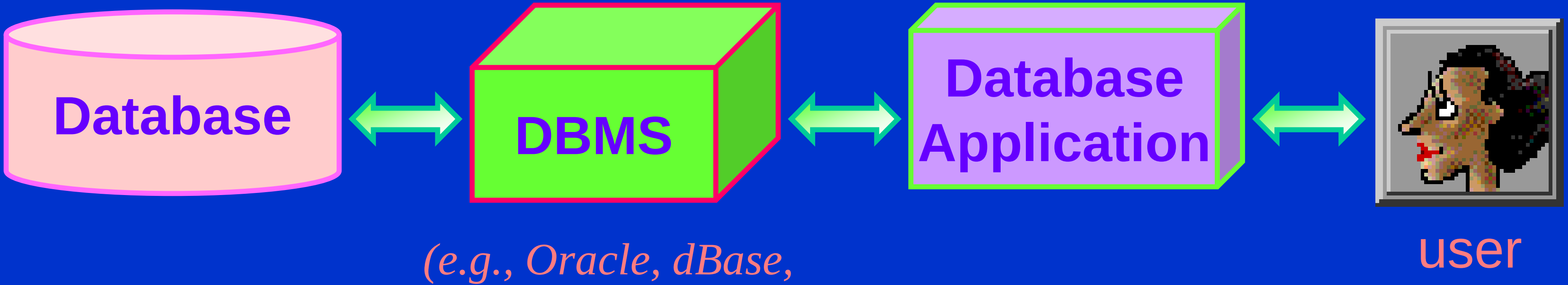


information

What is a Database Management System (DBMS)?

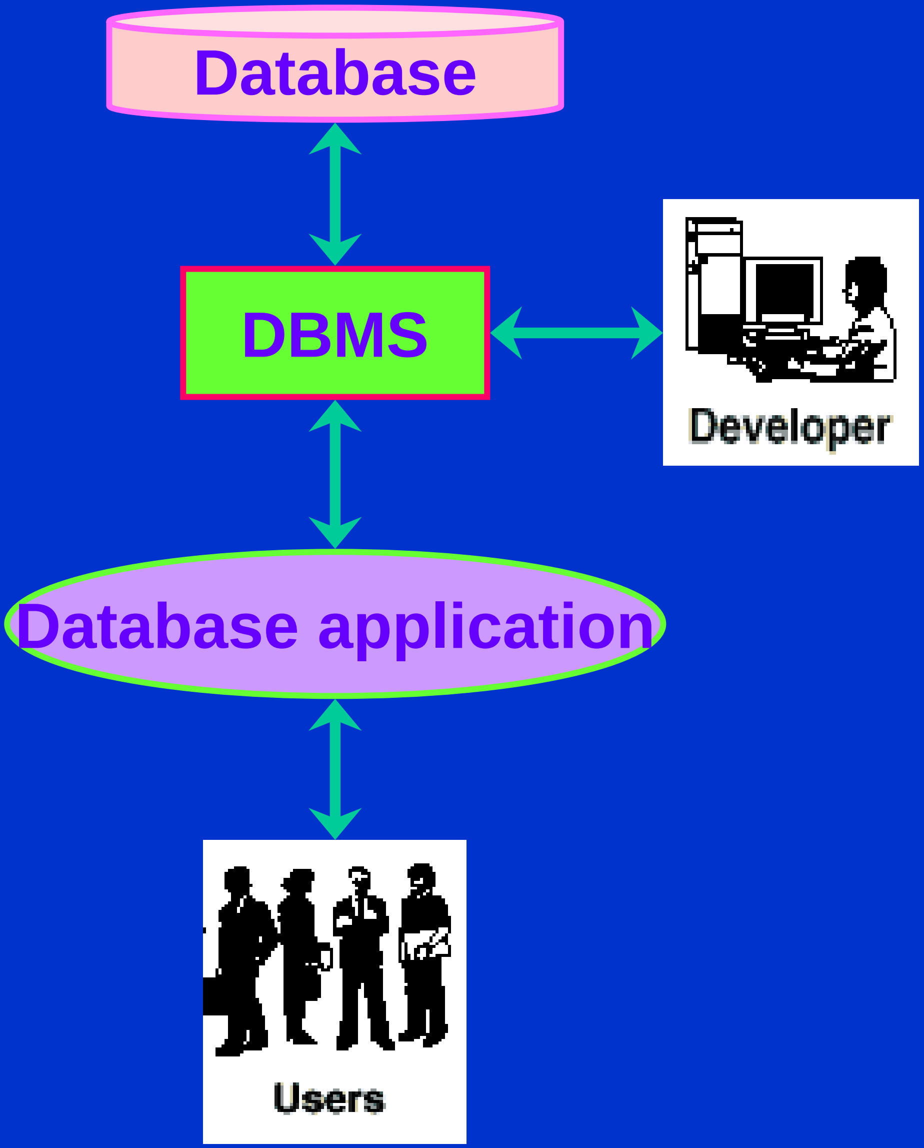
"A set of programs used to define,administer, and process the database and its applications conveniently and efficiently"

Program (or collection of programs) that enables users to create the database. The DBMS manages the storage and retrieval of data, and provides the user with certain functionalities to guarantee that the data will be logically organized and consistently applied.



(e.g., Oracle, dBase, Access, Paradox)

What is a Database Application?



A computer program that performs a specific task of practical value in a business situation

An interface that allows the user to enter and manipulate data;

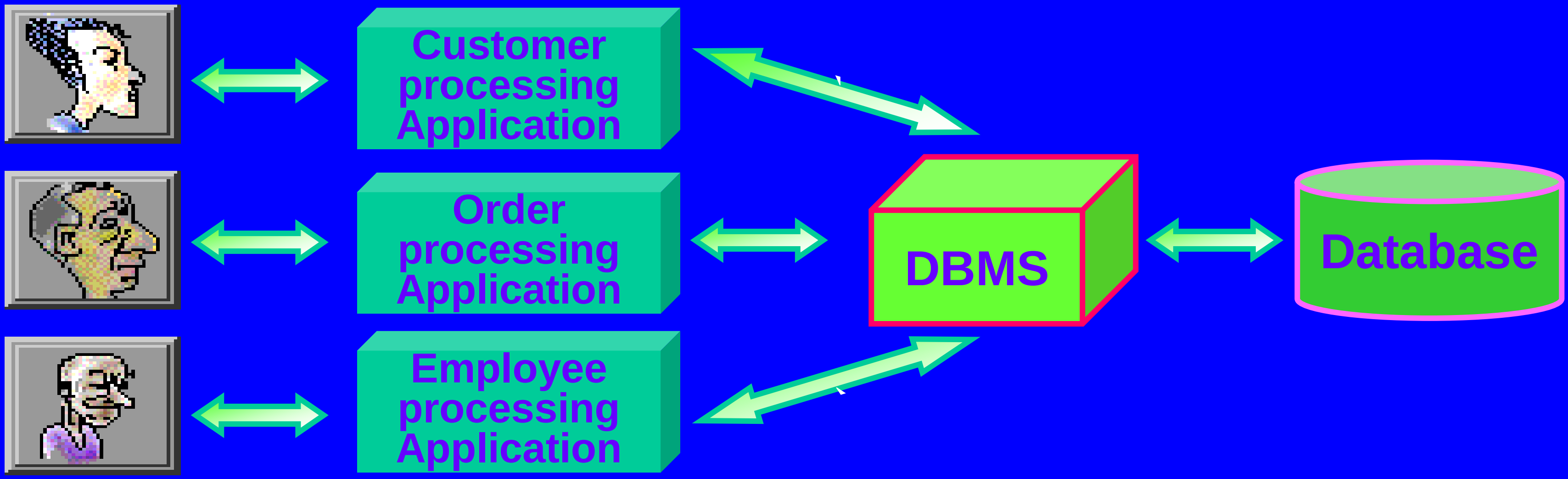
User can request abstract views of data

Created by database designers and developers using a DBMS program or a programming language

Database Management System (DBMS)

- DBMS contains information about a particular enterprise
  - Collection of interrelated data
  - Set of programs to access the data
  - An environment that is both *convenient* and *efficient* to use
- Database Applications:
  - Banking: transactions
  - Airlines: reservations, schedules
  - Universities: registration, grades
  - Sales: customers, products, purchases

Historical Roots of Database Systems

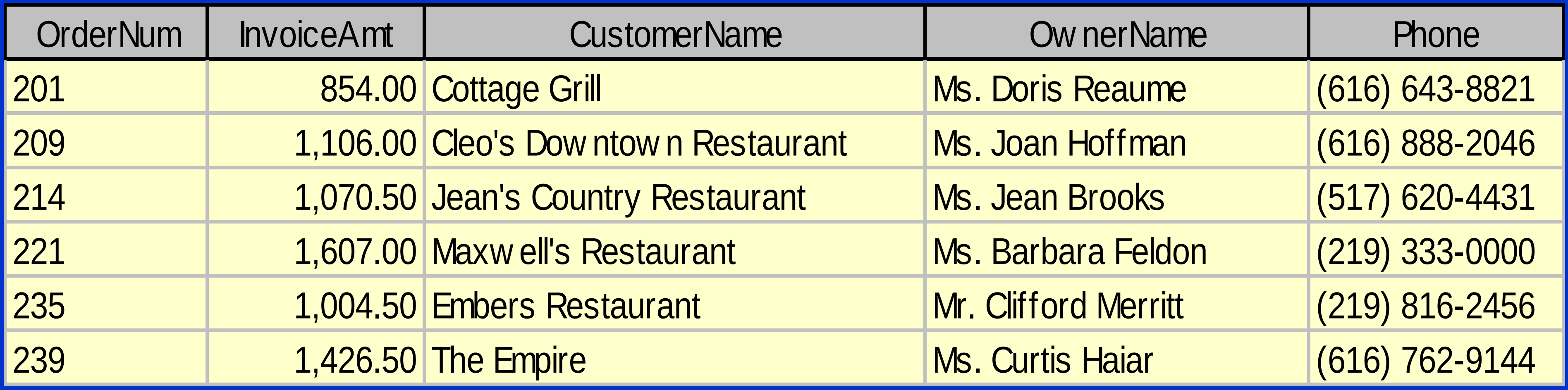

Customer processing Application
Order processing Application
Employee processing Application
DBMS
Database

- Developed to overcome limitations of file systems, developed initially on mainframe computers in late 60s and early 70s - a typical early DBMS cost $100,000 (many are still in use)
- First general databases were created for General Electric Company (GEC) - Integrated Data Store (IDS), designed to run on GEC machines; B.F. Goodrich ported IDS to IBM 360 - became dominant until 1980s
- As PCs gained popularity (1980s), single-user, personal databases developed; at present, most database technology is used in workgroups

Better Definition of a Database

- A collection of users’ data, organized logically and managed by a unifying set of principles, procedures, and functionalities, which help guarantee the consistent application and interpretation of that data

(a) organized collection of related information or data stored on a computer disk for easy, efficient use; represented in tabular format



OrderNum	InvoiceAmt	CustomerName	OwnerName	Phone
201	854.00	Cottage Grill	Ms. Doris Reaume	(616) 643-8821
209	1,106.00	Cleo's Downtown Restaurant	Ms. Joan Hoffman	(616) 888-2046
214	1,070.50	Jean's Country Restaurant	Ms. Jean Brooks	(517) 620-4431
221	1,607.00	Maxwell's Restaurant	Ms. Barbara Feldon	(219) 333-0000
235	1,004.50	Embers Restaurant	Mr. Clifford Merritt	(219) 816-2456
239	1,426.50	The Empire	Ms. Curtis Haiar	(616) 762-9144

When do I use a Database program?

	Word processing	Spreadsheet	Database
Text handling	*excellent*	*fair*	*poor*
Mathematical functions	*poor*	*excellent*	*very good*
Ease of Use	*excellent*	*good*	*fair*
Training Cost	*low*	*moderate*	*high*
Software Cost	*low*	*moderate*	*high*
Volume of data	*low*	*moderate*	*very high*
Multiuser Access	*low*	*moderate*	*very high*

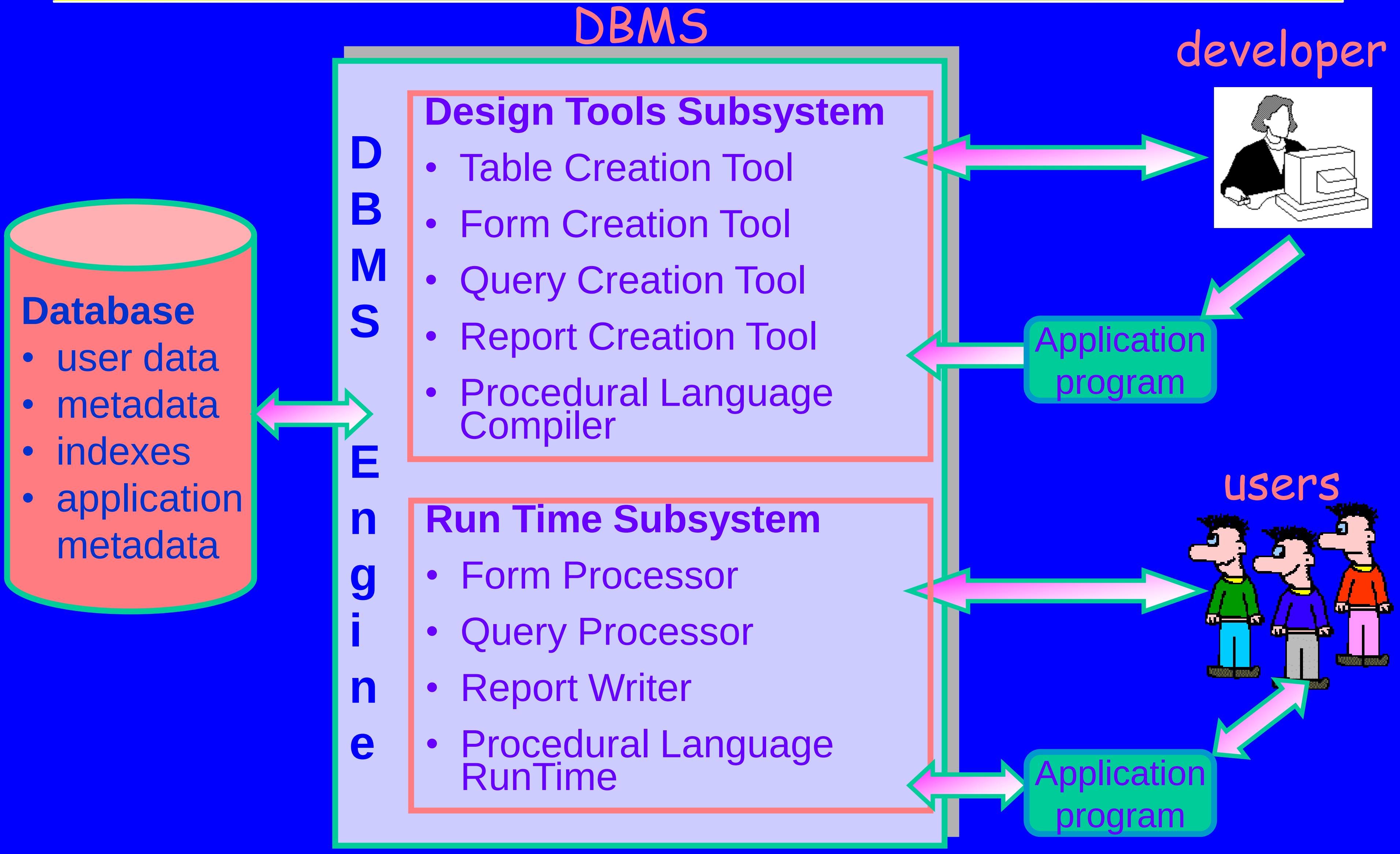
The Database System Environment

- Hardware - physical devices
  - computer, peripherals, network devices

you are here

- Software
  - DBMS (manages the database)
  - operating systems software (manages hardware & software)
  - application programs (user access and manipulate database)
- People
  - system administrators (manage general operations)
  - database designers (architects of database structure)
  - database administrators (ensure the database is functioning)
  - systems analysts & programmers (design & implement database)
  - end users (use application programs)
- Procedures - rules of the company governing use of data
- Data

Features of a DBMS



DBMS

developer

users

DBMS Engine

Design Tools Subsystem

- Table Creation Tool
- Form Creation Tool
- Query Creation Tool
- Report Creation Tool
- Procedural Language Compiler

Run Time Subsystem

- Form Processor
- Query Processor
- Report Writer
- Procedural Language RunTime

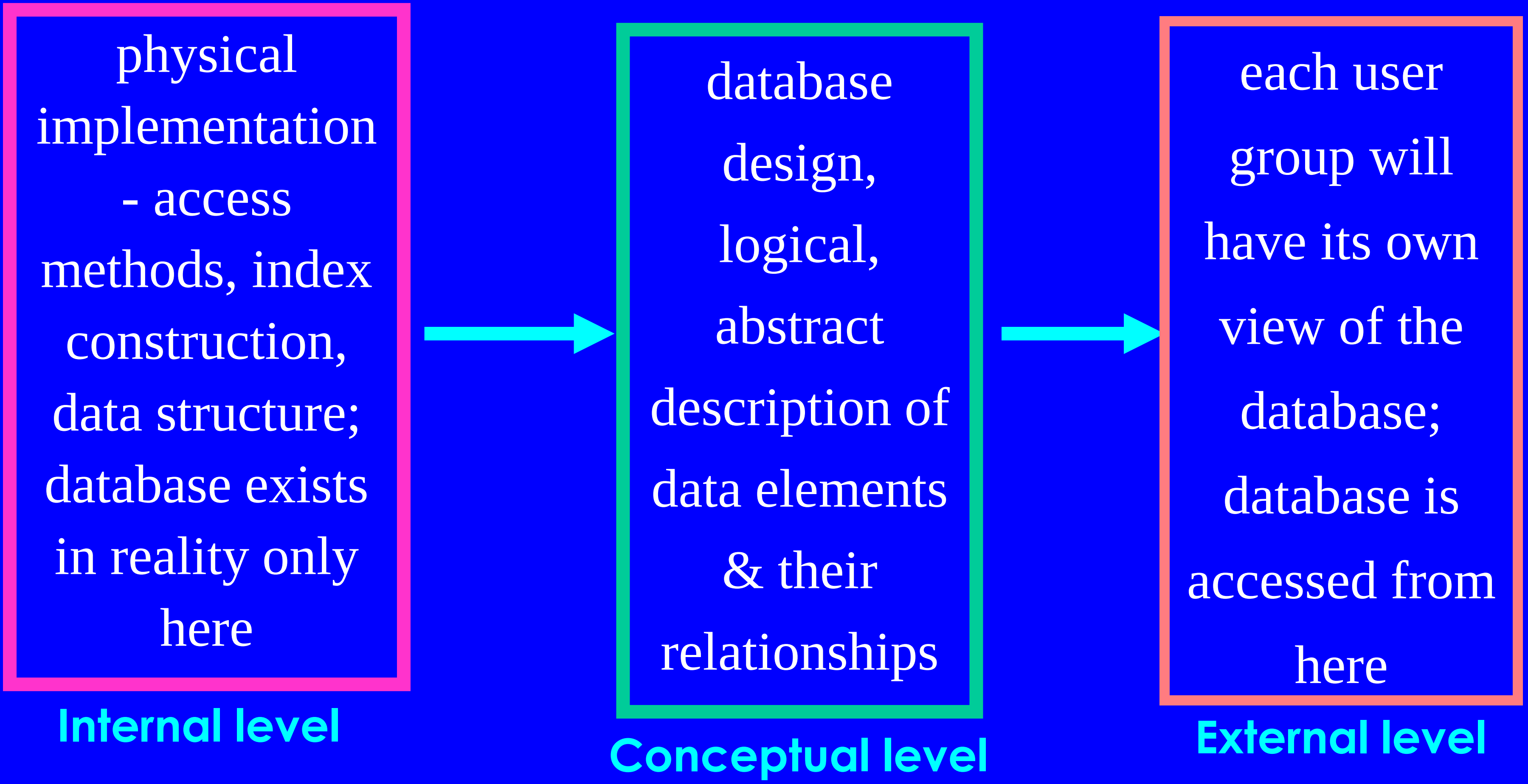
Database

- user data
- metadata
- indexes
- application metadata

Application program

Application program

Three levels of Database Representation



physical implementation - access methods, index construction, data structure; database exists in reality only here

Internal level

database design, logical, abstract description of data elements & their relationships

Conceptual level

each user group will have its own view of the database; database is accessed from here

External level

Levels of Abstraction

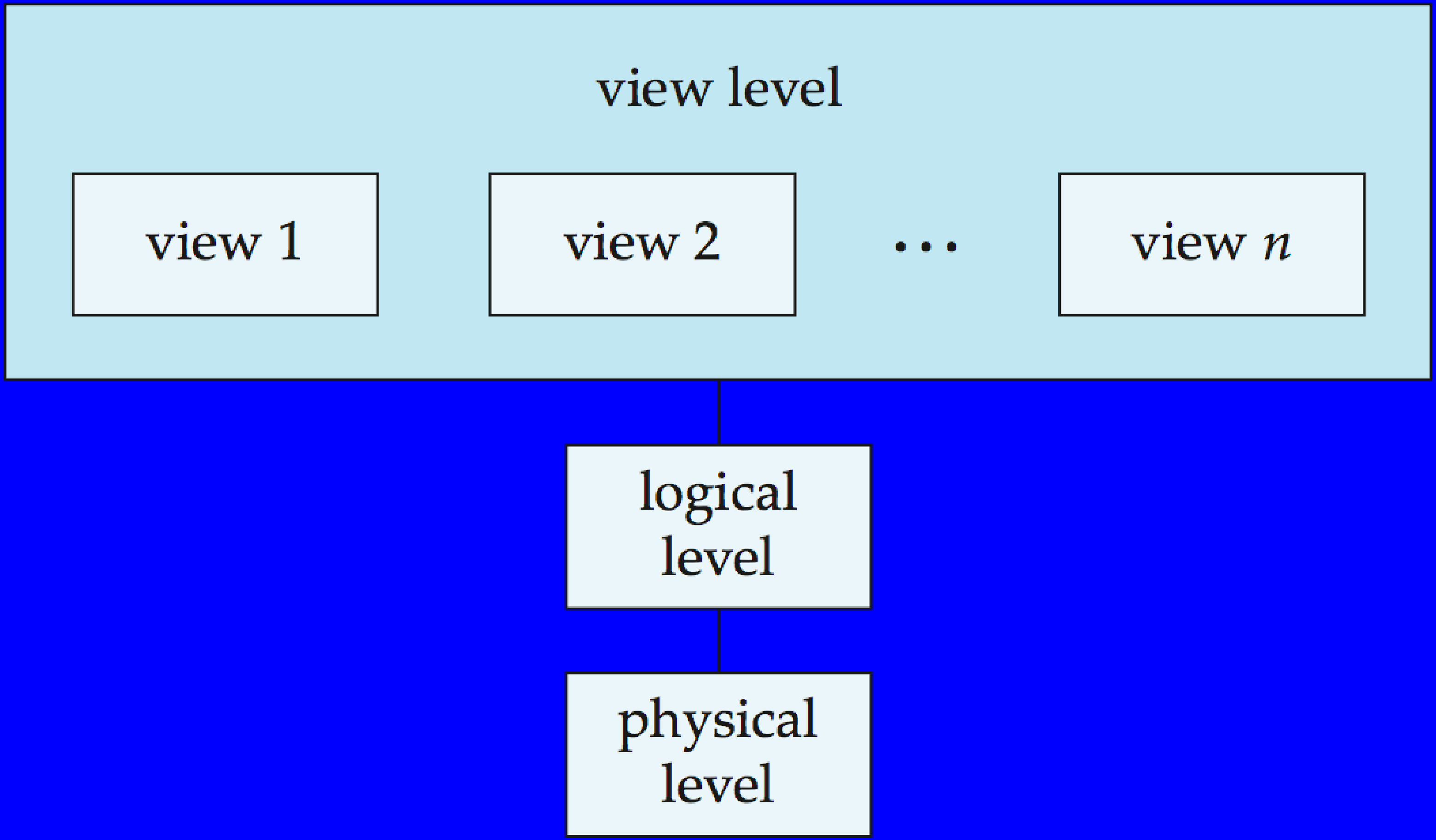
- **Physical level:** describes how a record (e.g., instructor) is stored.
- **Logical level:** describes data stored in database, and the relationships among the data.

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

- **View level:** application programs hide details of data types. Views can also hide information (such as an employee’s salary) for security purposes.

View of Data

An architecture for a database system



Instances and Schemas

- Similar to types and variables in programming languages
- **Logical Schema** – the overall logical structure of the database
  - Example: The database consists of information about a set of customers and accounts in a bank and the relationship between them
    - Analogous to type information of a variable in a program
- **Physical schema**– the overall physical structure of the database
- **Instance** – the actual content of the database at a particular point in time
  - Analogous to the value of a variable
- **Physical Data Independence** – the ability to modify the physical schema without changing the logical schema
  - Applications depend on the logical schema
  - In general, the interfaces between the various levels and components should be well defined so that changes in some parts do not seriously influence others.

Database Schema

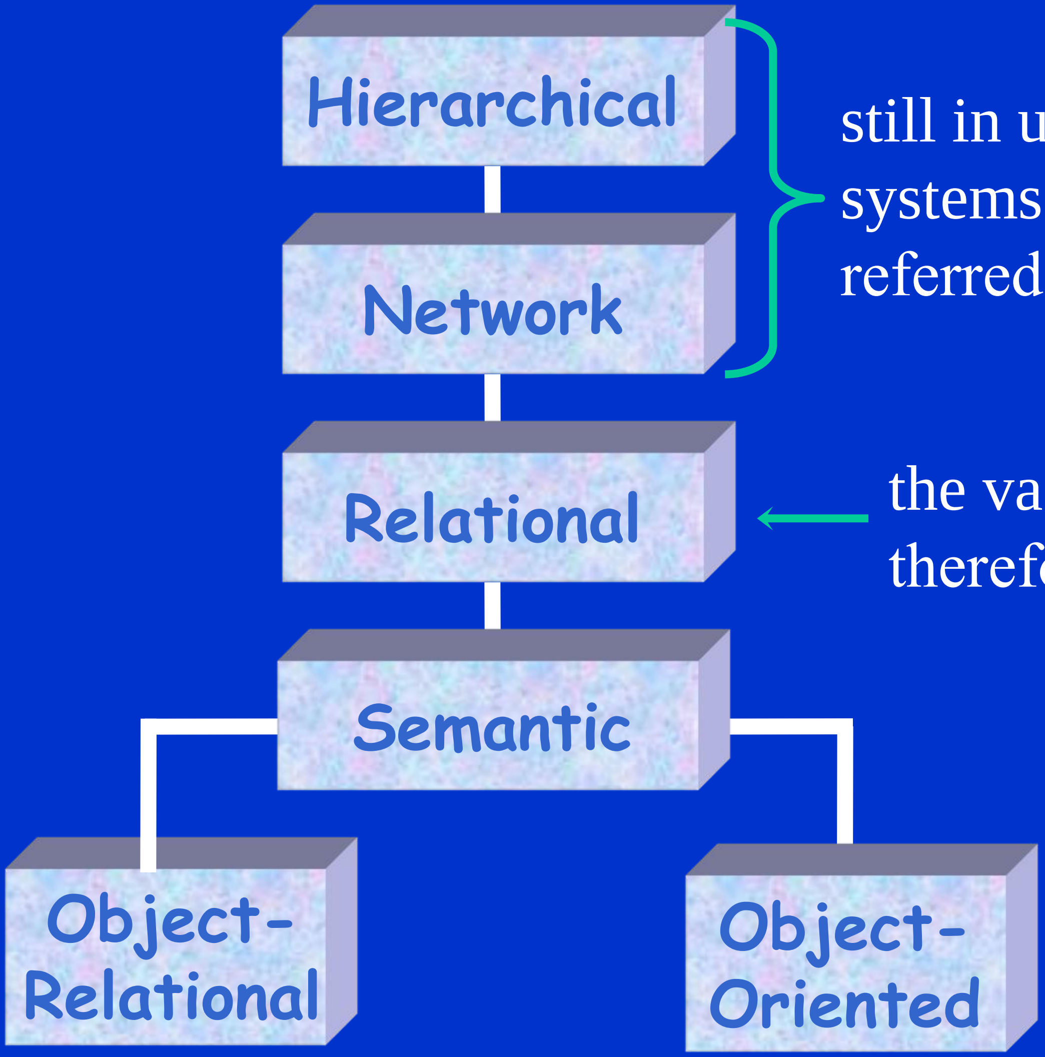
Database schema defines database’s structure, tables, relationships, domains, and constraint rules

- **Tables**
  - BOOK (ISBN, Title, AuthID, PubID, Price)
  - PUBLISHER (PubID, PubName, PubPhone)
  - AUTHOR (AuthID, AuthName, AuthPhone)
- **Relationships**
  - Each book is published by one and only one publisher
  - Each publisher publishes one or more books
- **Domains (set of values in a column)**
  - Physical description (e.g., set of integers 0 < x < 99999)
- **Constraints (business rules)**
  - Price cannot be less than zero; Author phone field cannot be left blank

Data Models

- A collection of tools for describing
  - Data
  - Data relationships
  - Data semantics
  - Data constraints
- Relational model
- Entity-Relationship data model (mainly for database design)
- Object-based data models (Object-oriented and Object-relational)
- Semistructured data model (XML)
- Other older models:
  - Network model
  - Hierarchical model

Evolution of Database Models



Hierarchical

Network

Relational

Semantic

Object-Relational

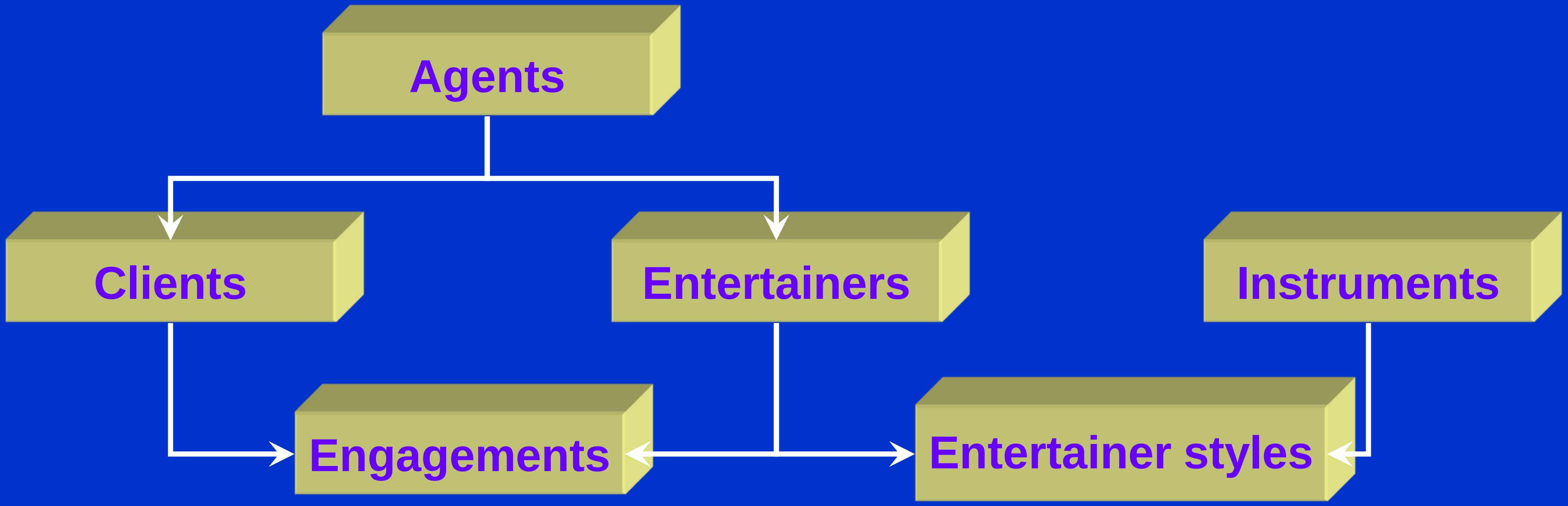
Object-Oriented

still in use in many older (1970s) legacy systems; very few new databases; referred to “navigational systems”

the vast majority currently use this, therefore, our course’s focus is here

Very few new databases are being created using Object-Oriented Programming (not many ODBMS for businesses to implement this model)

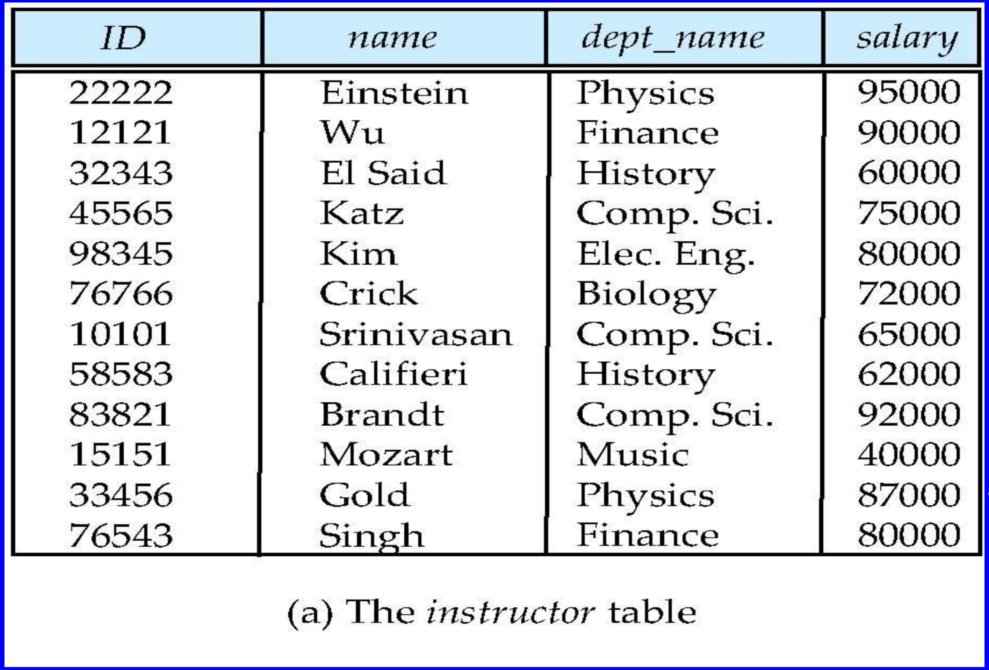
The Relational Database Model



- represented by tables (like spreadsheets)
- tables are NOT linked with physical pointers
- unlike earlier systems, all three types of relationships can be represented
- accommodates the design of larger databases that involve complex relationships and intricate manipulations

Relational Model

- All the data is stored in various tables.
- Example of tabular data in the relational model



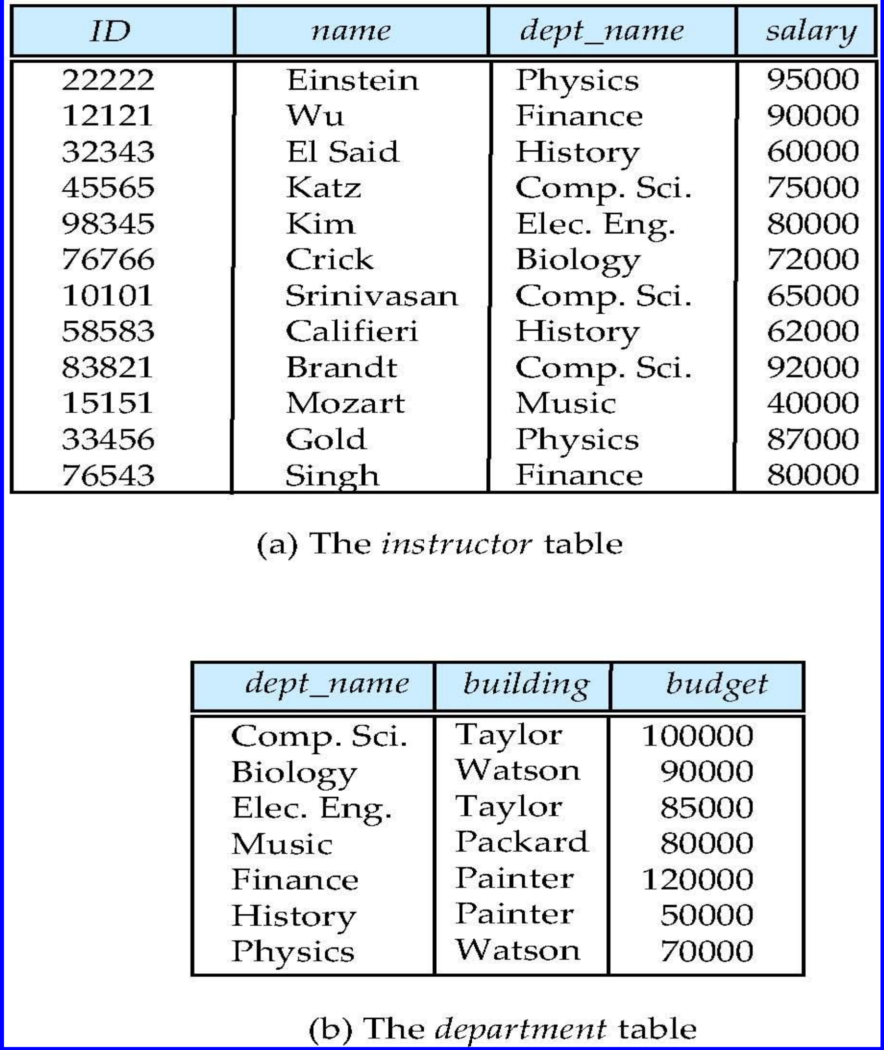
ID	*name*	*dept_name*	*salary*
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music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

(a) The *instructor* table

Columns

Rows

A Sample Relational Database



ID	*name*	*dept_name*	*salary*
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

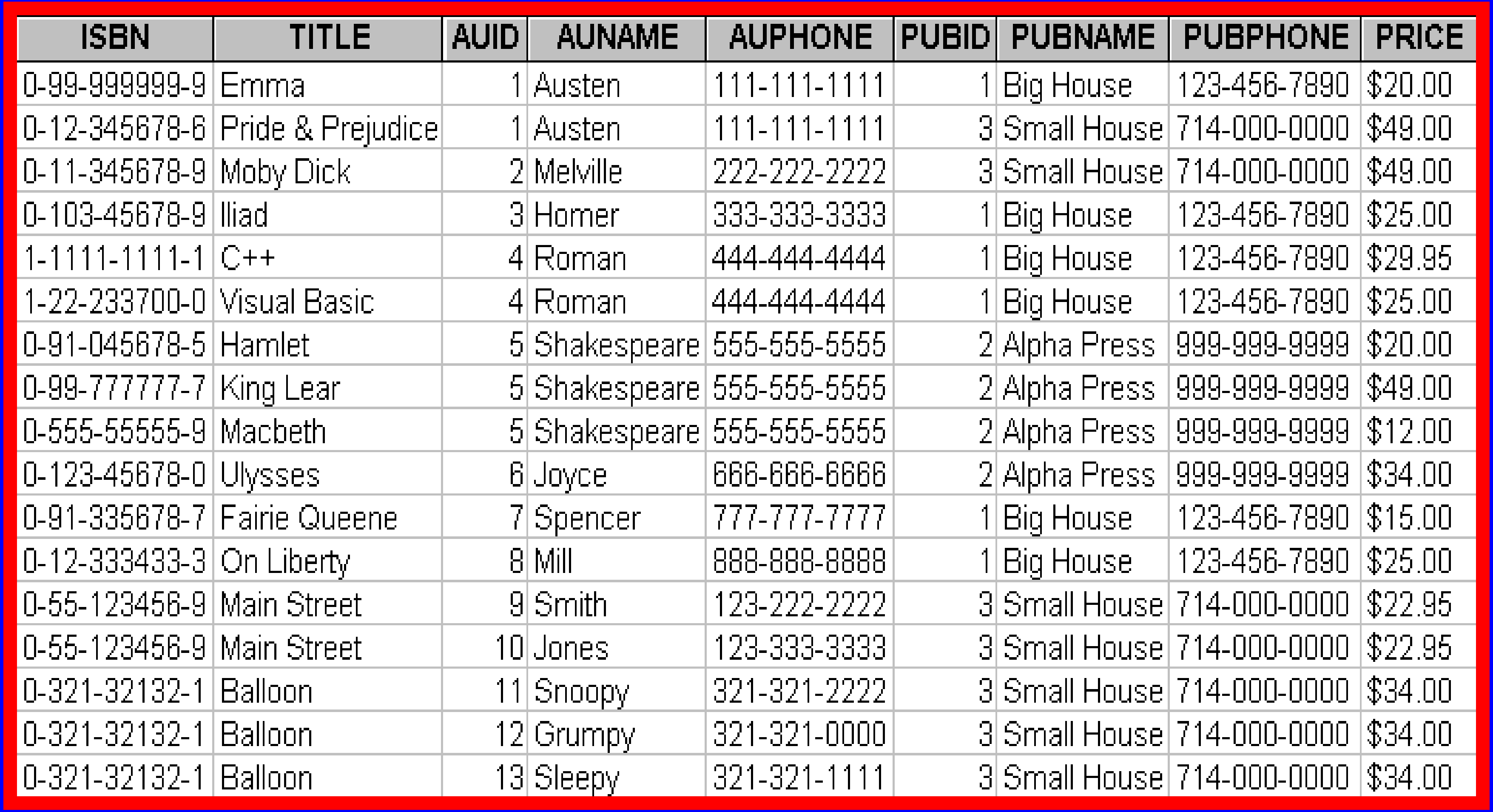
dept_name	*building*	*budget*
Comp. Sci.	Taylor	100000
Biology	Watson	90000
Elec. Eng.	Taylor	85000
Music	Packard	80000
Finance	Painter	120000
History	Painter	50000
Physics	Watson	70000

(b) The *department* table

Table

Users view their data in two-dimensional tables.

table = file = relation



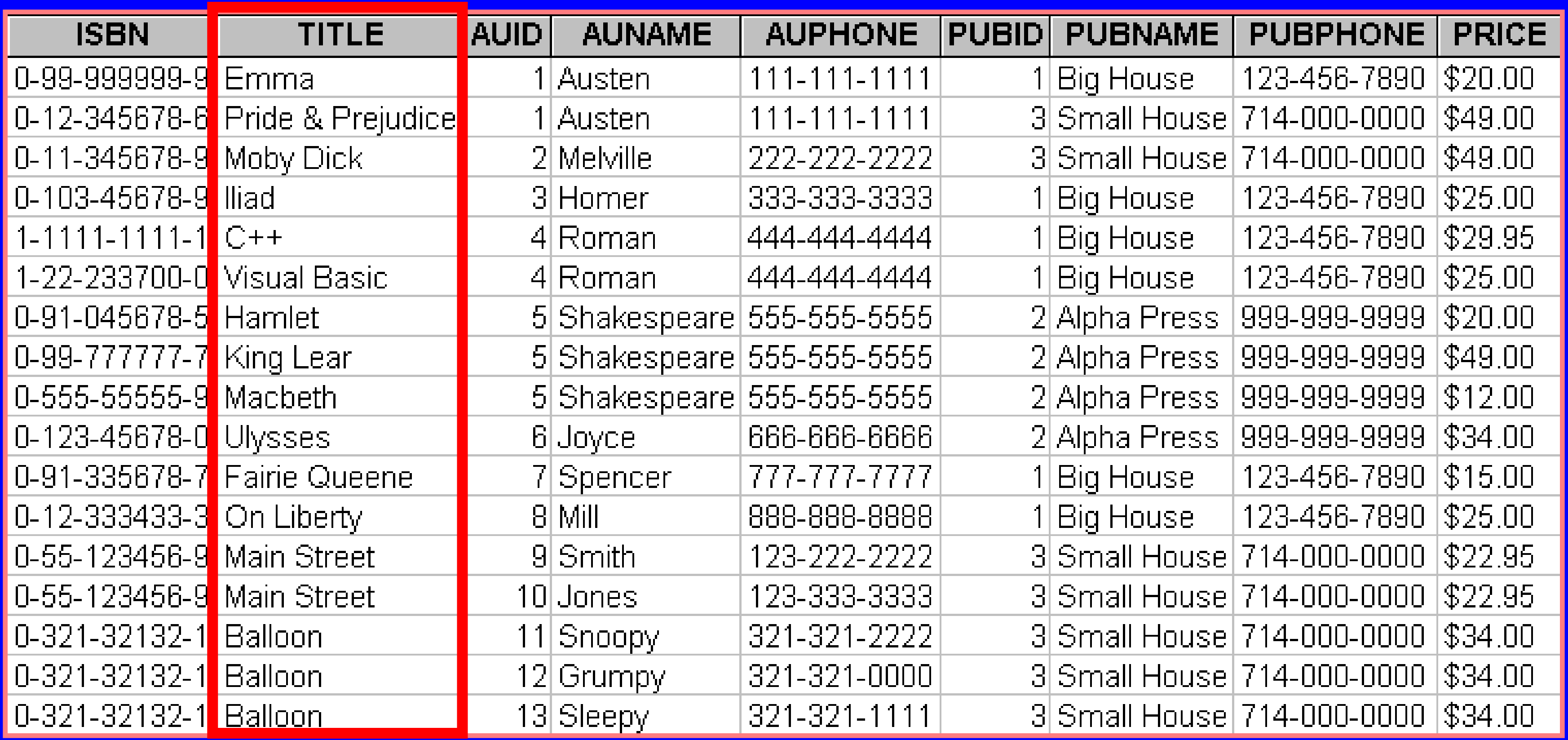
ISBN	TITLE	AUID	AUNAME	AUPHONE	PUBID	PUBNAME	PUBPHONE	PRICE
0-99-999999-9	Emma	1	Austen	111-111-1111	1	Big House	123-456-7890	$20.00
0-12-345678-6	Pride & Prejudice	1	Austen	111-111-1111	3	Small House	714-000-0000	$49.00
0-11-345678-9	Moby Dick	2	Melville	222-222-2222	3	Small House	714-000-0000	$49.00
0-103-45678-9	Iliad	3	Homer	333-333-3333	1	Big House	123-456-7890	$25.00
1-1111-1111-1	C++	4	Roman	444-444-4444	1	Big House	123-456-7890	$29.95
1-22-233700-0	Visual Basic	4	Roman	444-444-4444	1	Big House	123-456-7890	$25.00
0-91-045678-5	Hamlet	5	Shakespeare	555-555-5555	2	Alpha Press	999-999-9999	$20.00
0-99-777777-7	King Lear	5	Shakespeare	555-555-5555	2	Alpha Press	999-999-9999	$49.00
0-555-55555-9	Macbeth	5	Shakespeare	555-555-5555	2	Alpha Press	999-999-9999	$12.00
0-123-45678-0	Ulysses	6	Joyce	666-666-6666	2	Alpha Press	999-999-9999	$34.00
0-91-335678-7	Fairie Queene	7	Spencer	777-777-7777	1	Big House	123-456-7890	$15.00
0-12-333433-3	On Liberty	8	Mill	888-888-8888	1	Big House	123-456-7890	$25.00
0-55-123456-9	Main Street	9	Smith	123-222-2222	3	Small House	714-000-0000	$22.95
0-55-123456-9	Main Street	10	Jones	123-333-3333	3	Small House	714-000-0000	$22.95
0-321-32132-1	Balloon	11	Snoopy	321-321-2222	3	Small House	714-000-0000	$34.00
0-321-32132-1	Balloon	12	Grumpy	321-321-0000	3	Small House	714-000-0000	$34.00
0-321-32132-1	Balloon	13	Sleepy	321-321-1111	3	Small House	714-000-0000	$34.00

Field

The fields within records contain data.

Data within a field must be of the same data type. Each field within a table must have a unique name. Order of fields is unimportant.

column = field = attribute

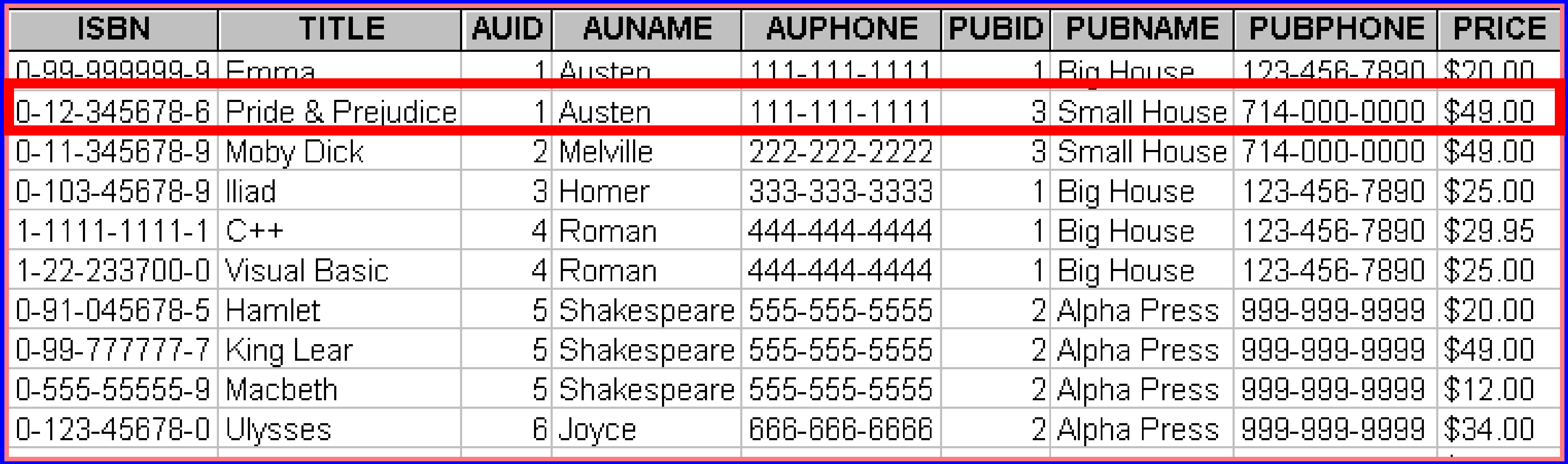


ISBN	TITLE	AUID	AUNAME	AUPHONE	PUBID	PUBNAME	PUBPHONE	PRICE
0-99-999999-9	Emma	1	Austen	111-111-1111	1	Big House	123-456-7890	$20.00
0-12-345678-6	Pride & Prejudice	1	Austen	111-111-1111	3	Small House	714-000-0000	$49.00
0-11-345678-9	Moby Dick	2	Melville	222-222-2222	3	Small House	714-000-0000	$49.00
0-103-45678-9	Iliad	3	Homer	333-333-3333	1	Big House	123-456-7890	$25.00
1-1111-1111-1	C++	4	Roman	444-444-4444	1	Big House	123-456-7890	$29.95
1-22-233700-0	Visual Basic	4	Roman	444-444-4444	1	Big House	123-456-7890	$25.00
0-91-045678-5	Hamlet	5	Shakespeare	555-555-5555	2	Alpha Press	999-999-9999	$20.00
0-99-777777-7	King Lear	5	Shakespeare	555-555-5555	2	Alpha Press	999-999-9999	$49.00
0-555-55555-9	Macbeth	5	Shakespeare	555-555-5555	2	Alpha Press	999-999-9999	$12.00
0-123-45678-0	Ulysses	6	Joyce	666-666-6666	2	Alpha Press	999-999-9999	$34.00
0-91-335678-7	Fairie Queene	7	Spencer	777-777-7777	1	Big House	123-456-7890	$15.00
0-12-333433-3	On Liberty	8	Mill	888-888-8888	1	Big House	123-456-7890	$25.00
0-55-123456-9	Main Street	9	Smith	123-222-2222	3	Small House	714-000-0000	$22.95
0-55-123456-9	Main Street	10	Jones	123-333-3333	3	Small House	714-000-0000	$22.95
0-321-32132-1	Balloon	11	Snoopy	321-321-2222	3	Small House	714-000-0000	$34.00
0-321-32132-1	Balloon	12	Grumpy	321-321-0000	3	Small House	714-000-0000	$34.00
0-321-32132-1	Balloon	13	Sleepy	321-321-1111	3	Small House	714-000-0000	$34.00

Record

- A record is a group of related fields of information about a single instance of one object or event in a database.
- Tables consist of zero, one, or more records.
- Order of rows is unimportant.

row = record = tuple



ISBN	TITLE	AUID	AUNAME	AUPHONE	PUBID	PUBNAME	PUBPHONE	PRICE
0-99-999999-9	Emma	1	Austen	111-111-1111	1	Big House	123-456-7890	$20.00
0-12-345678-6	Pride & Prejudice	1	Austen	111-111-1111	3	Small House	714-000-0000	$49.00
0-11-345678-9	Moby Dick	2	Melville	222-222-2222	3	Small House	714-000-0000	$49.00
0-103-45678-9	Iliad	3	Homer	333-333-3333	1	Big House	123-456-7890	$25.00
1-1111-1111-1	C++	4	Roman	444-444-4444	1	Big House	123-456-7890	$29.95
1-22-233700-0	Visual Basic	4	Roman	444-444-4444	1	Big House	123-456-7890	$25.00
0-91-045678-5	Hamlet	5	Shakespeare	555-555-5555	2	Alpha Press	999-999-9999	$20.00
0-99-777777-7	King Lear	5	Shakespeare	555-555-5555	2	Alpha Press	999-999-9999	$49.00
0-555-55555-9	Macbeth	5	Shakespeare	555-555-5555	2	Alpha Press	999-999-9999	$12.00
0-123-45678-0	Ulysses	6	Joyce	666-666-6666	2	Alpha Press	999-999-9999	$34.00

Evaluation of the Relational database model

Advantages

But #1 problem still is

- mechanisms for minimizing data redundancy and inconsistency
- logical database design is separated from physical aspects
- relatively program-data independent
- management of data for access, manipulation, and security
- flexible mechanisms for generating reports and queries
- program development and maintenance costs are reduced
- data can be accessed in a multiplicity of ways within and amongst organizations

Disadvantages

- ease of use - many untrained people create and use databases without considering its design - usually incorporate many errors

Data Definition Language (DDL)

- Specification notation for defining the database schema

  Example:

```
create table instructor (
        ID          char(5),
        name        varchar(20),
        dept_name   varchar(20),
        salary      numeric(8,2))
```

SQL

- The most widely used commercial language
- To be able to compute complex functions SQL is usually embedded in some higher-level language

SQL

- SQL: widely used non-procedural language
  - Example: Find the name of the instructor with ID 22222

```
select   name
from     instructor
where    instructor.ID = '22222'
```

  - Example: Find the ID and building of instructors in the Physics dept.

```
select instructor.ID, department.building
from instructor, department
where instructor.dept_name = department.dept_name and
        department.dept_name = 'Physics'
```

- Application programs generally access databases through one of
  - Language extensions to allow embedded SQL
  - Application program interface (e.g., ODBC/JDBC) which allow SQL queries to be sent to a database

Database Design

The process of designing the general structure of the database:

- Logical Design – Deciding on the database schema. Database design requires that we find a "good" collection of relation schemas.
  - Business decision – What attributes should we record in the database?
  - Computer Science decision – What relation schemas should we have and how should the attributes be distributed among the various relation schemas?
- Physical Design – Deciding on the physical layout of the database

XML: Extensible Markup Language

- Defined by the WWW Consortium (W3C)
- Originally intended as a document markup language not a database language
- The ability to specify new tags, and to create nested tag structures made XML a great way to exchange **data**, not just documents
- XML has become the basis for all new generation data interchange formats.
- A wide variety of tools is available for parsing, browsing and querying XML documents/data

Database Engine

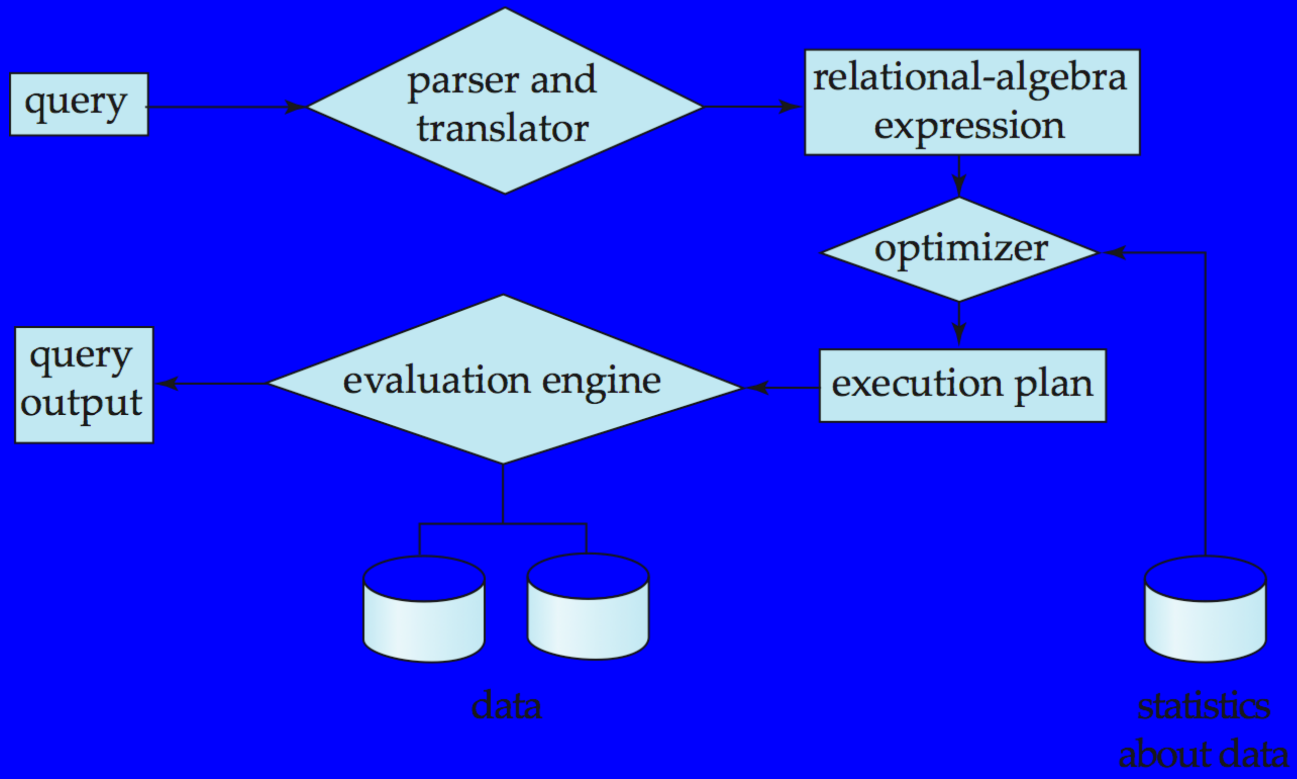
- Storage manager
- Query processing
- Transaction manager

Storage Management

- **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.
- The storage manager is responsible to the following tasks:
  - Interaction with the OS file manager
  - Efficient storing, retrieving and updating of data
- Issues:
  - Storage access
  - File organization
  - Indexing and hashing

Query Processing

1. Parsing and translation
2. Optimization
3. Evaluation



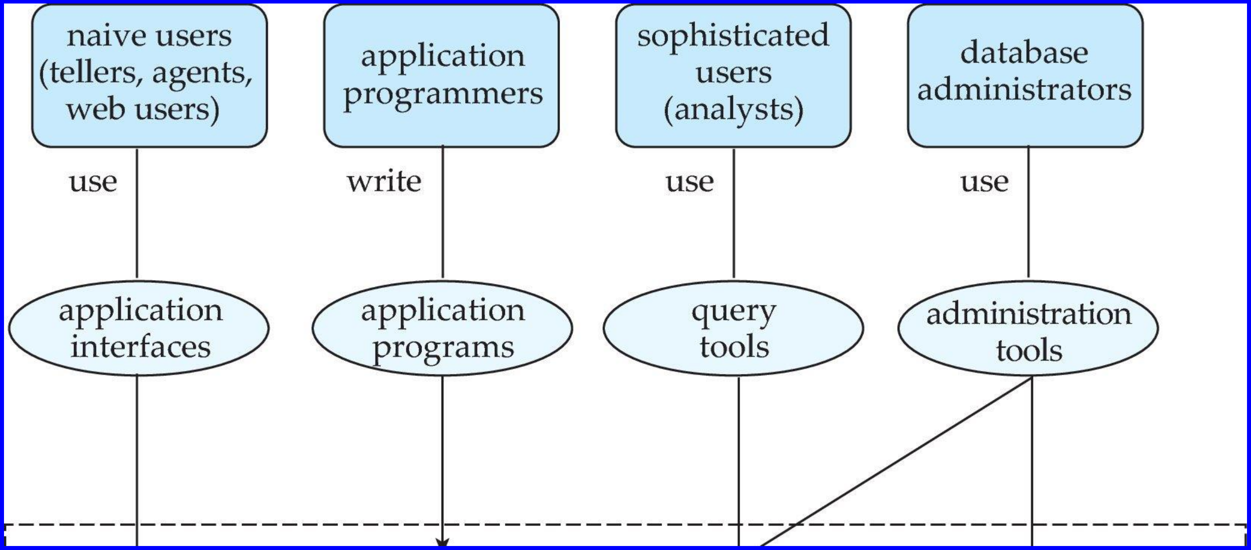
Query Processing (Cont.)

- 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

Transaction Management

- What if the system fails?
- What if more than one user is concurrently updating the same data?
- A **transaction** is a collection of operations that performs a single logical function in a database application
- **Transaction-management component** ensures that the database remains in a consistent (correct) state despite system failures (e.g., power failures and operating system crashes) and transaction failures.
- **Concurrency-control manager** controls the interaction among the concurrent transactions, to ensure the consistency of the database.

Database Users and Administrators



Database

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