INTRODUCTION TO DATABASE SYSTEMS

CHAPTER 1. INTRODUCTION

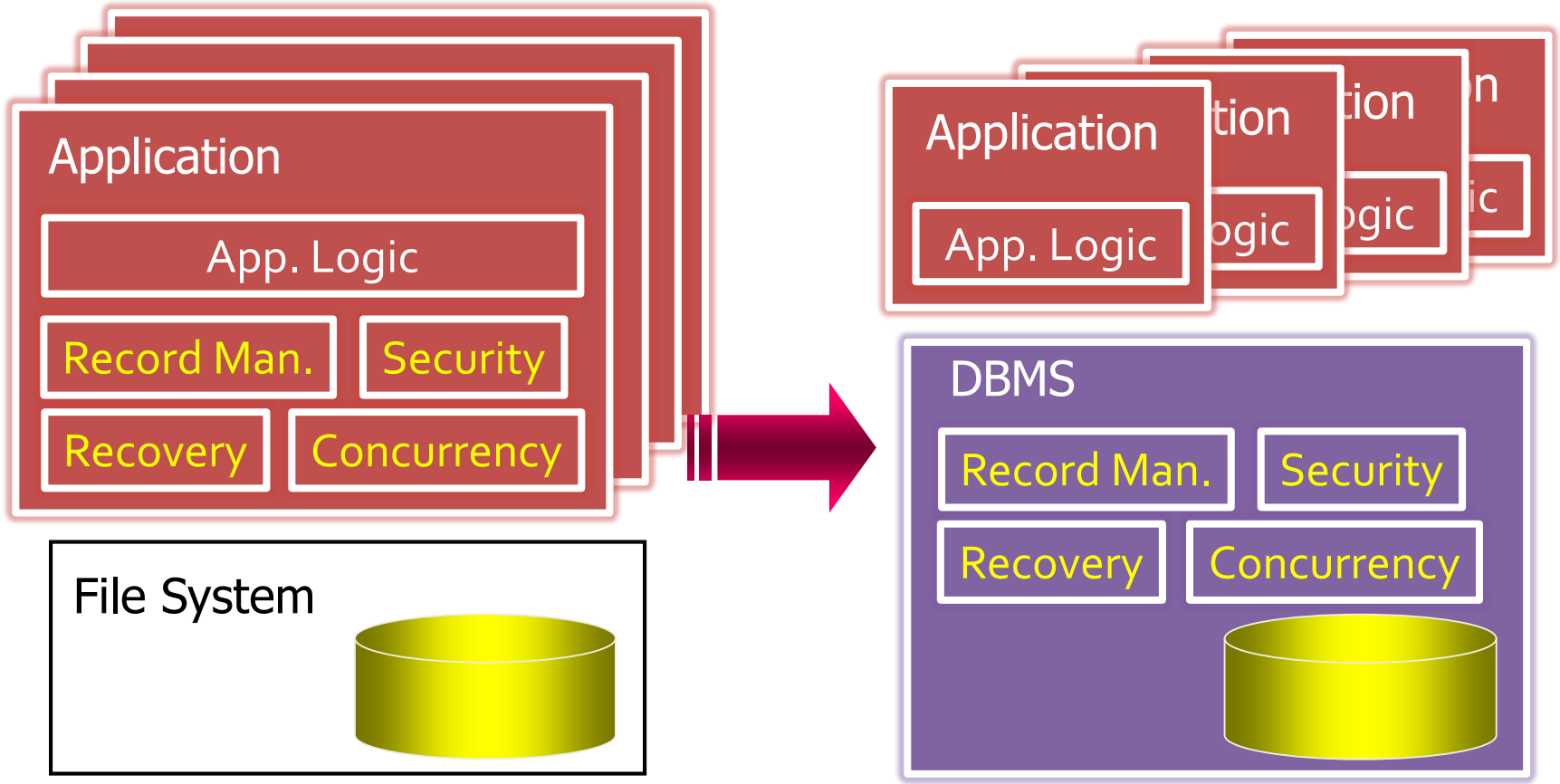
Contents

- Data & Database
- Database Management Systems
- File Systems
- View of Data
- Data Models
- Data Languages
- Database Users
- Overall System Structure

What is a Database?

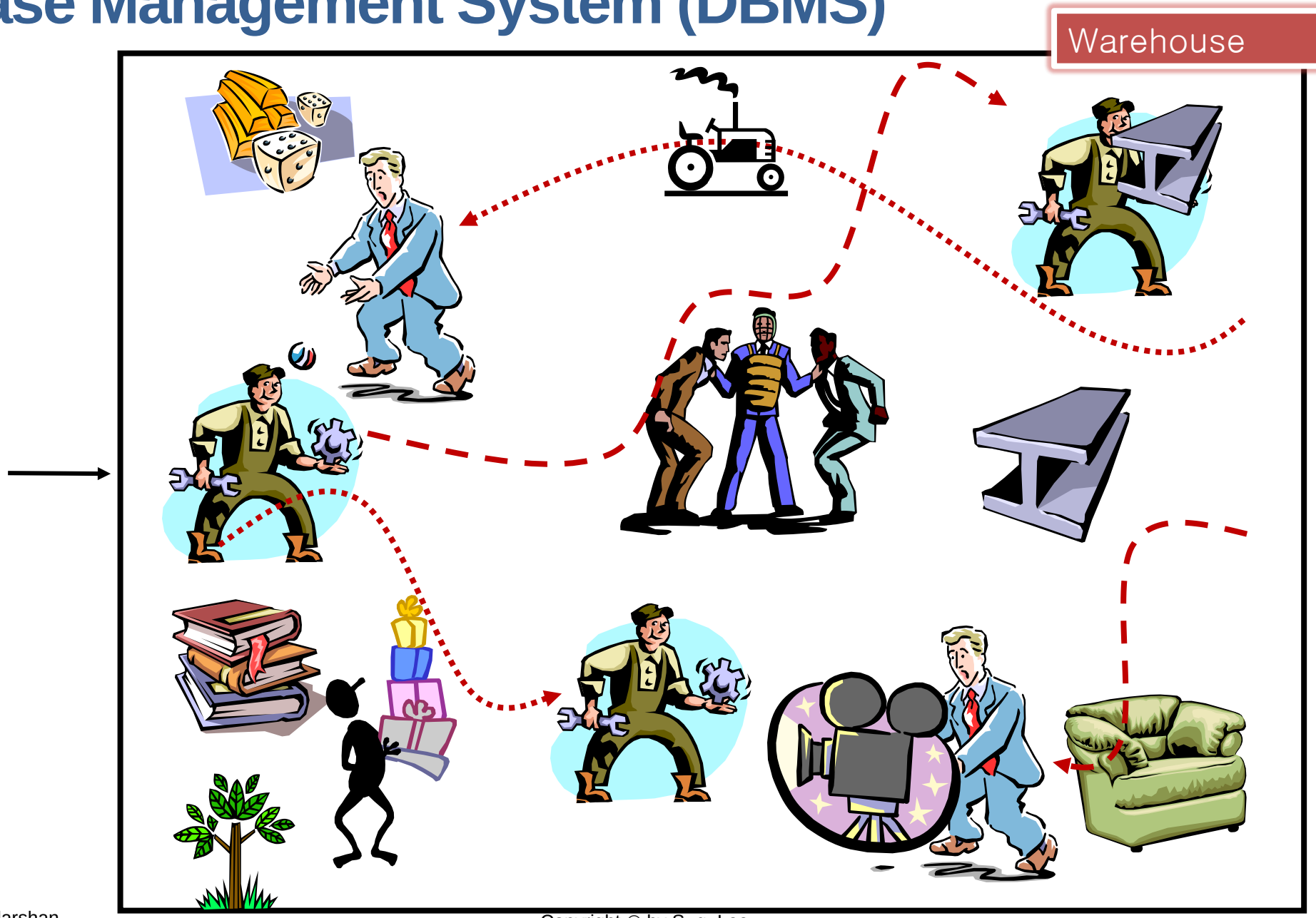
- **Data**
  - A formal description of
    - an entity, event, phenomena, or idea
    - that is worth recording
- **Database**
  - An of
  - data
  - representing the information of interest
  - for various programs that compose the computerized information system of an organization.
  - Data are separated from the programs that use them

DBMS

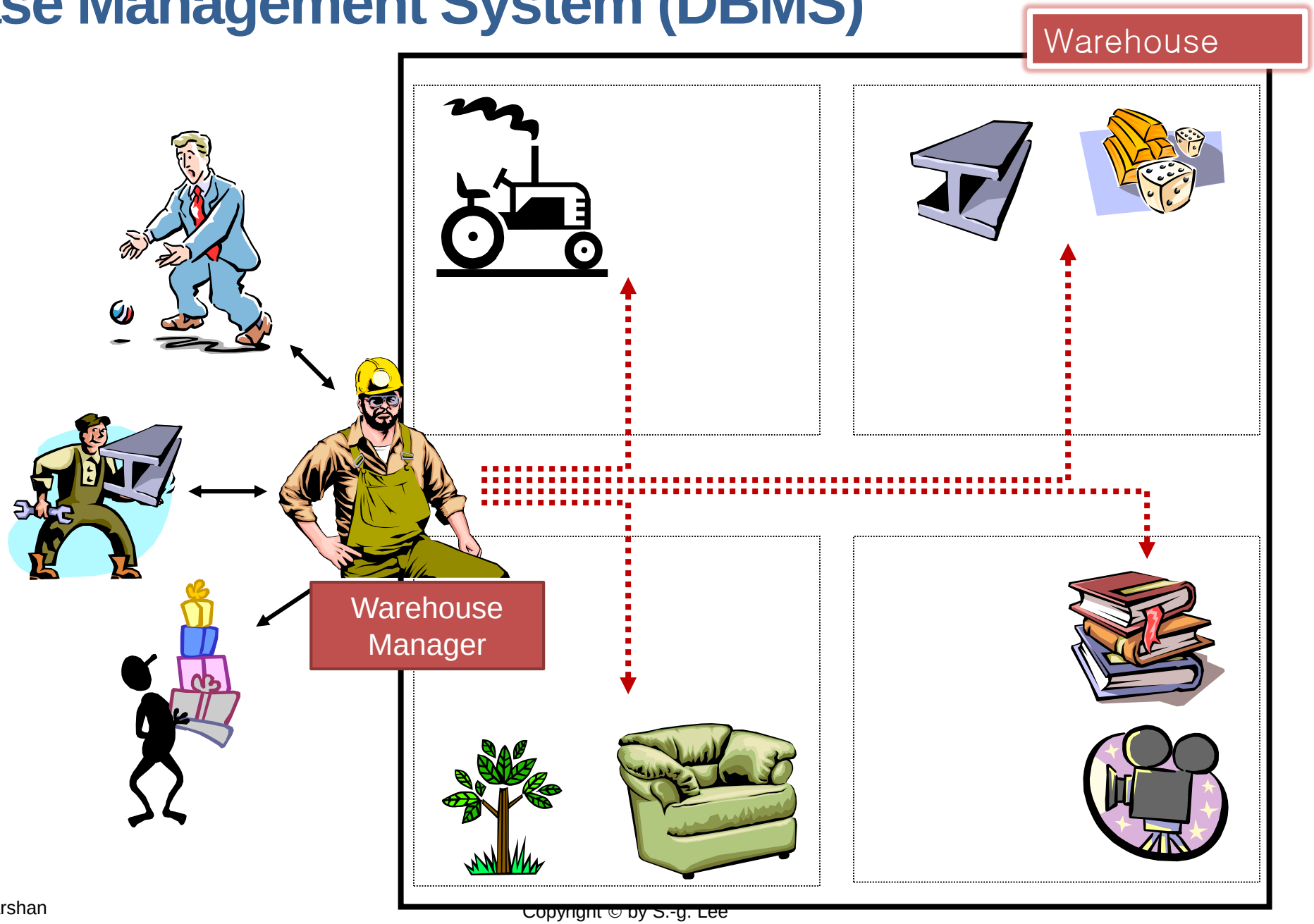


- Database Management System
  - A collection of program modules that store, process, and manage data
  - Abstraction

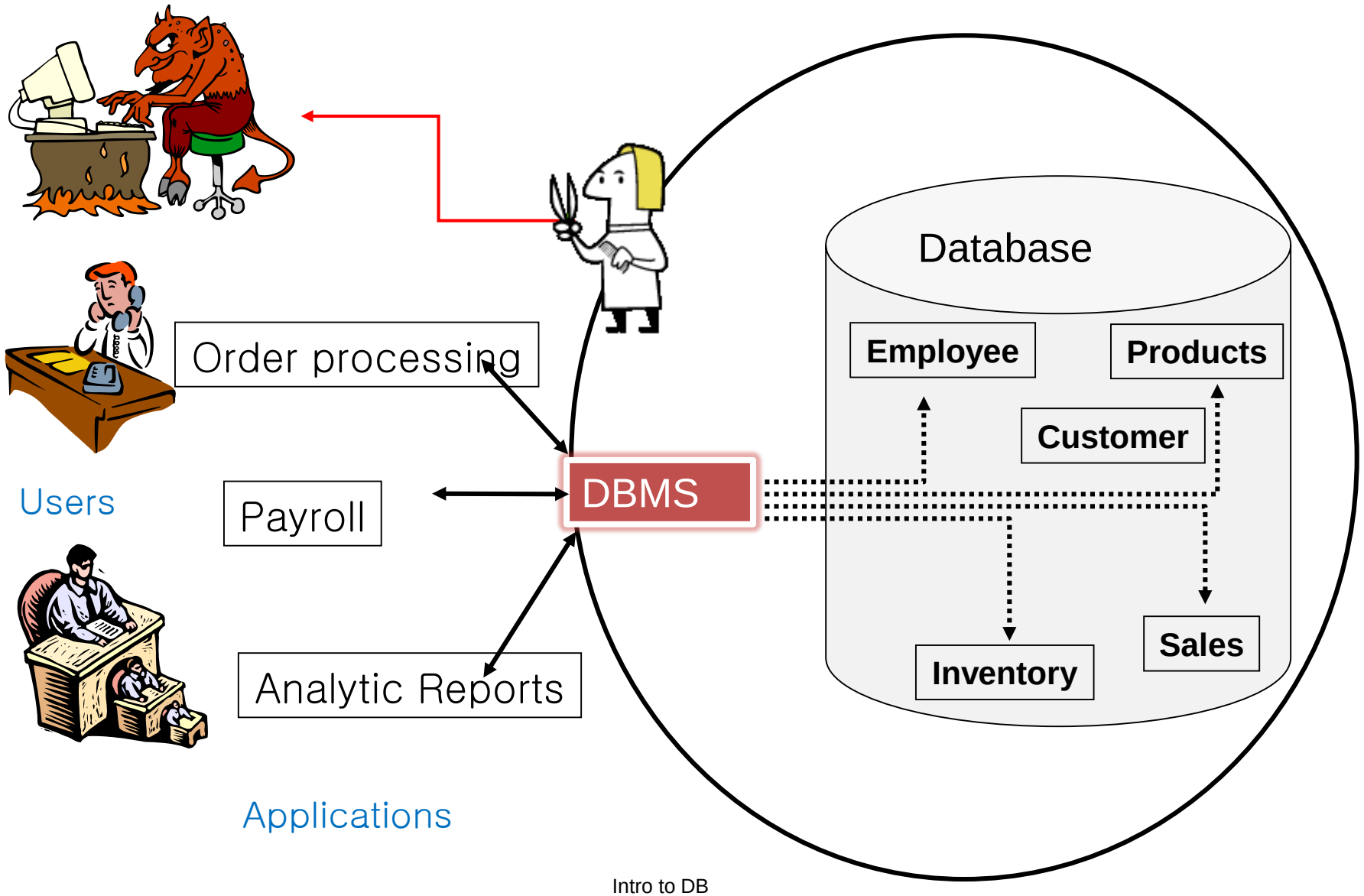
Database Management System (DBMS)



Database Management System (DBMS)



Database Management System (DBMS)



Database Management System (DBMS)

-
- DBMS provides an environment that is both *convenient* and *efficient* to use.
- Database Applications (Information Systems):
  - 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
- Databases touch all aspects of our lives
- Commercial Systems
  - DB2, Oracle, MS SQL Server, MySQL, Hana
  - MS Access

File Systems

- File System
  -
  - Stores programs, data, documents, or anything
  - (in disk)
- In the early days, database applications were built on top of file systems
- 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
  - Integrity problems
    - Integrity constraints (e.g. account balance > 0) become part of program code
    - Hard to add new constraints or change existing ones

File Systems (cont.)

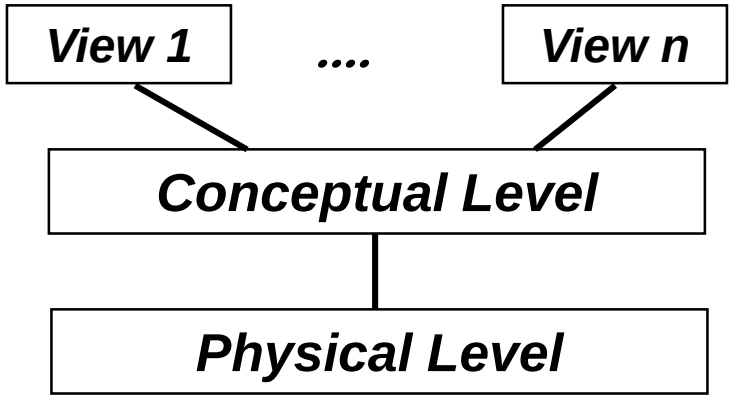
- Drawbacks of using file systems (cont.)
  - Atomicity of updates
    - Failures may leave database in an inconsistent state with partial updates carried out
    - E.g. transfer of funds from one account to another should either complete or not happen at all
  - Concurrent access by multiple users
    - Concurrent accessed needed for performance
    - Uncontrolled concurrent accesses can lead to inconsistencies
      - E.g. two people reading a balance and updating it at the same time
  - Security problems
- Database systems offer solutions to all the above problems

Levels of Abstraction

- **Physical level** describes how a record (e.g., customer) is stored in a physical device.

- **Logical level** describes data stored in database, and the relationships among the data.

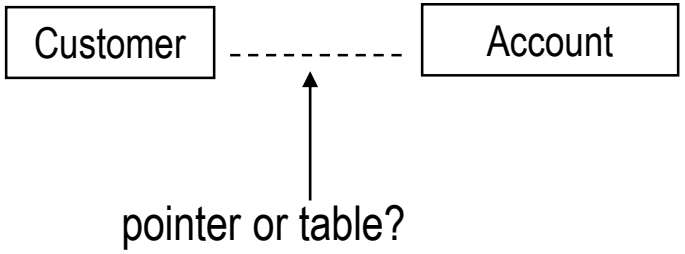
```
type customer = record
        name : string;
        street : string;
        city : integer;
end;
```



- **View level**: application programs hide details of data types.
  - Views can also hide information (e.g., salary) for security purposes.

Data Independence

- ability to modify a schema in one level without affecting a schema definition in the next higher level
- physical data independence:
  - physical level - conceptual level
- logical data independence:
  - conceptual level - view level



Instances and Schemas

- Similar to types and variables in programming languages
- **Schema** – [redacted]
  - e.g., the database consists of information about a set of customers and accounts and the relationship between them)
  - Analogous to type information of a variable in a program
  - **Physical schema**: database design at the physical level
  - **Logical schema**: database design at the logical level
- **Instance** – [redacted] at a particular point in time
  - Analogous to the value of a variable

Instances and Schemas – Examples

- Scheme (schema)
  - the skeletal structure of the data content

Customer

Name	Address	Telephone

Account

No.	Type	Balance

- Instance
  - the actual content of the data at a given time
  - database status

2014/2/20/12:00
Customer

HS Kim	Seoul	323-3232
KS Lee	Busan	323-5454
PL Park	Seoul	553-3235
…		

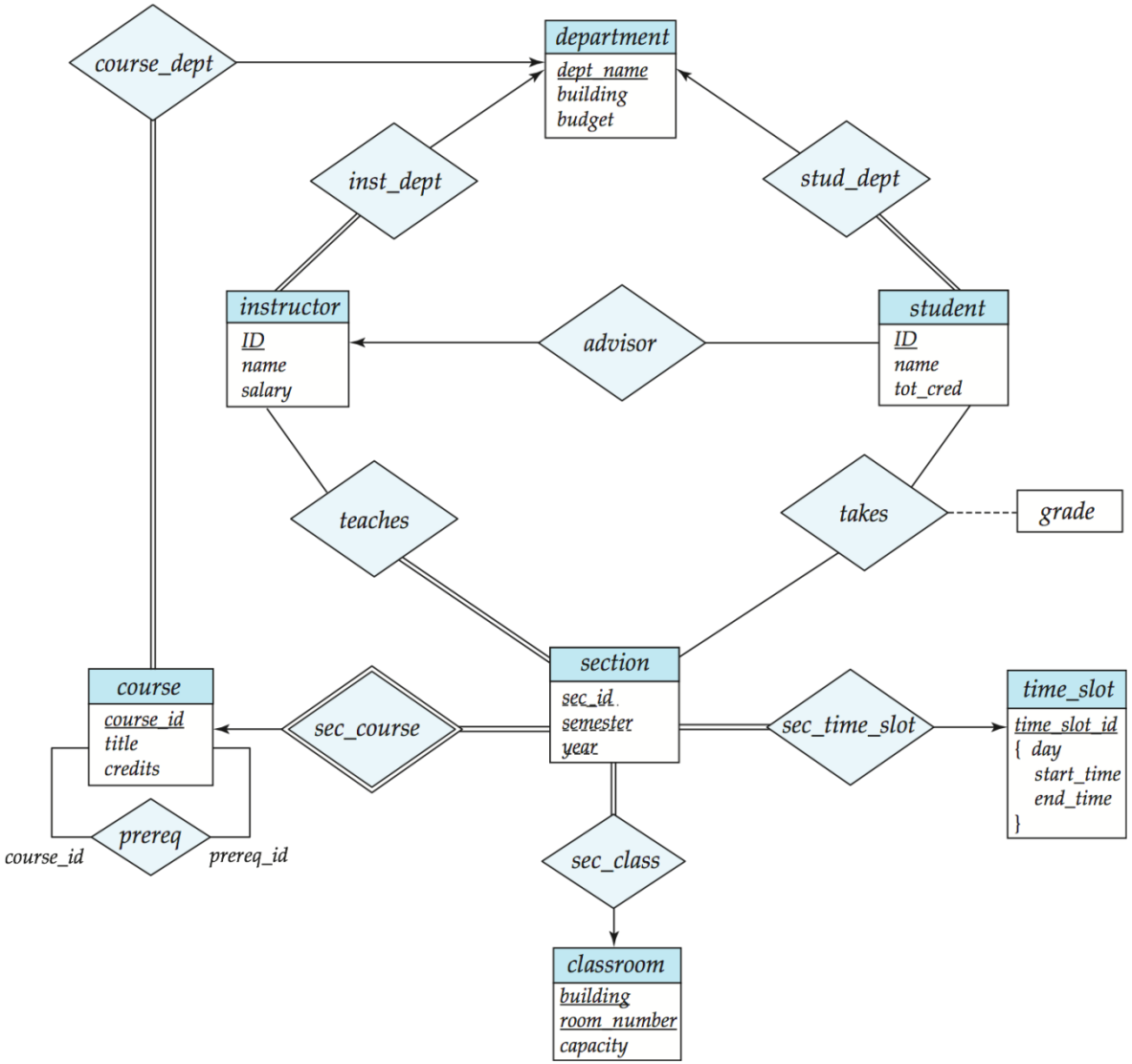
2014/3/3/12:00
Customer

HS Kim	Suwon	323-3232
KS Lee	Busan	323-5454
MH Choi	Seoul	553-3235
KH Na	Yongin	545-5488
…		

Data Models

- A collection of tools for describing
  - data
  - data relationships
  - data semantics
  - data constraints
- Entity-Relationship model
- Relational model
- Other models:
  - object-oriented model
  - semi-structured data models
  - Older models: network model and hierarchical model

Entity-Relationship Model

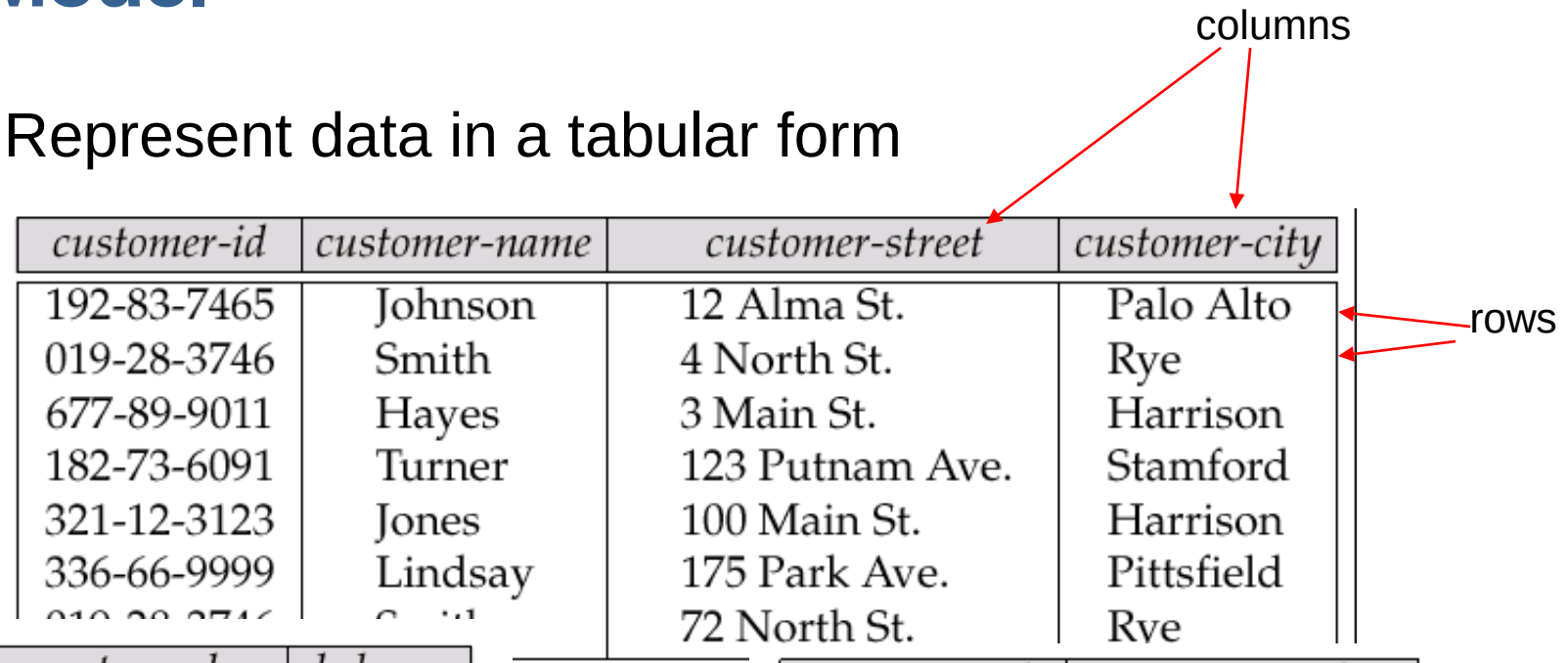


Entity Relationship Model (cont.)

- E-R model of real world
  - 
    - E.g. customers, accounts, bank branch
  - 
    - E.g. Account A-101 is held by customer Johnson
    - Relationship set *depositor* associates customers with accounts
- Widely used for database design
  - Database design in E-R model usually converted to design in the relational model (coming up next) which is used for storage and processing

Relational Model

- Represent data in a tabular form



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
[illegible]	[illegible]	72 North St.	Rye

) The *custom*

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

Database Languages

- Data Definition Language (DDL)
  -
    - create table
    - drop column
- Data Manipulation Language (DML)
  -
    - Retrieve
    - Insert
    - Delete
    - Change
- Query
  -
  - query language: part of DML
  - sometimes “query language = DML”

SQL

- The most widely used language
  - E.g. 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'
```

  - E.g. 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
```

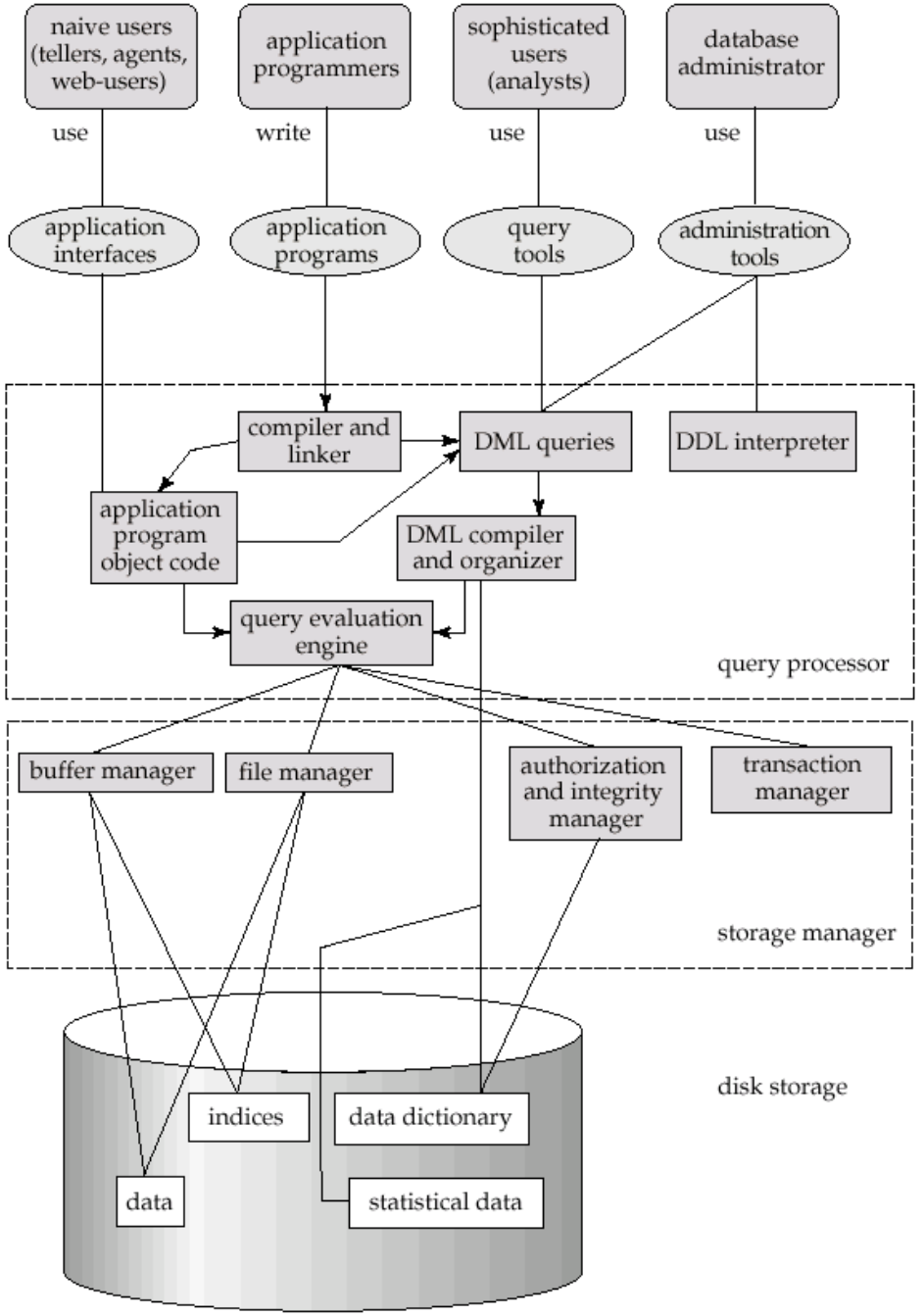
Database Users

- Users are differentiated by the way they expect to interact with the system
- *Application programmers* –
- *Sophisticated users* – form requests in a database query language
- *Specialized users* – write specialized database applications that do not fit into the traditional data processing framework
- – invoke one of the permanent application programs that have been written previously
  - E.g. people accessing database over the web, bank tellers, clerical staff

Database Administrator

- Coordinates all the activities of the database system; the database administrator has a good understanding of the enterprise's information resources and needs.
- Database administrator's duties include:
  -
  - Storage structure and access method definition
  - Schema and physical organization modification
  - Granting user authority to access the database
  - Specifying integrity constraints
  - Acting as liaison with users
  -

Overall System Structure



END OF CHAPTER 1