

CS2102

Database Systems

Slides adapted from Prof. Chan Chee Yong

LECTURE 01

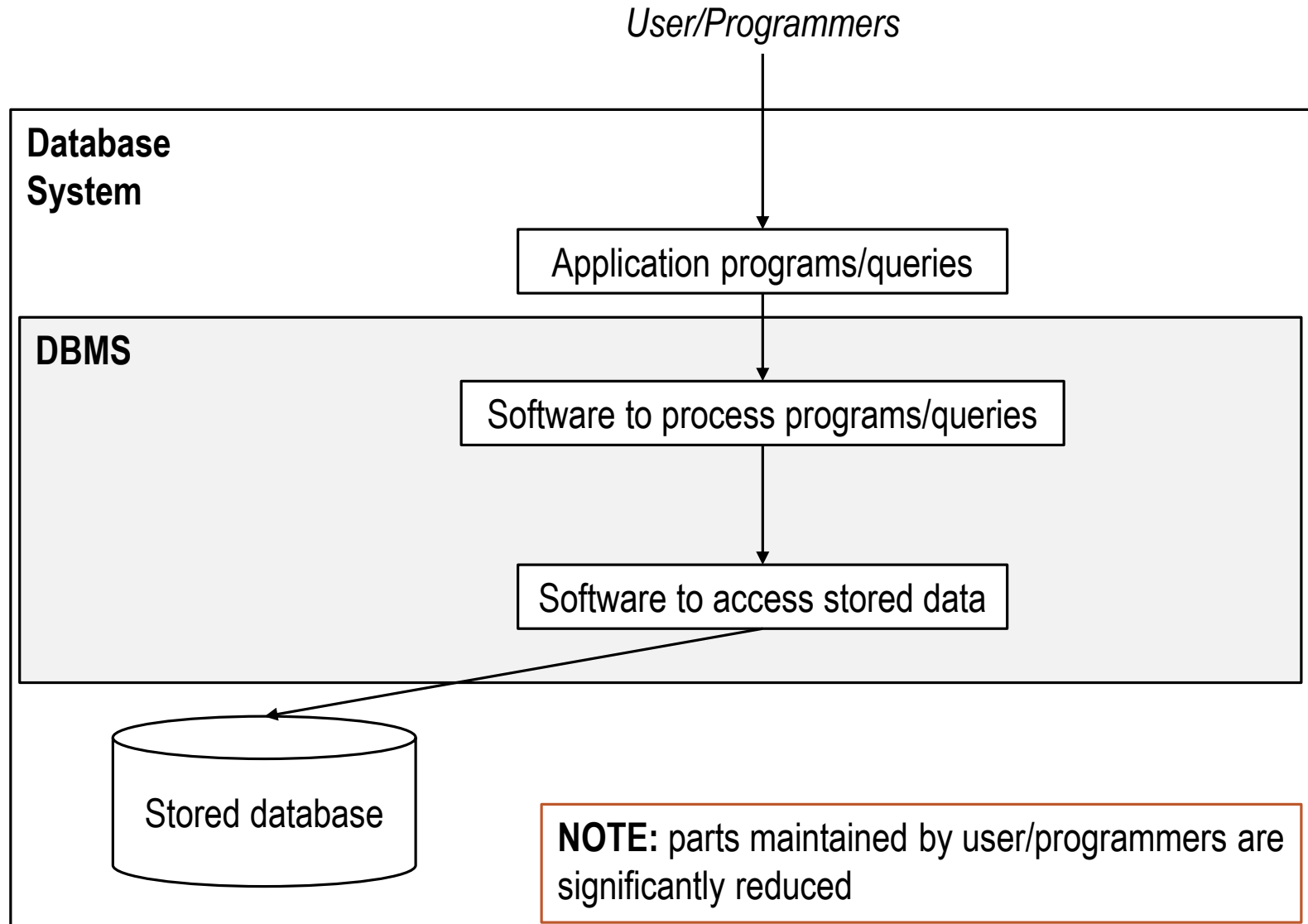
LIGHT INTRODUCTION

Traditional data processing

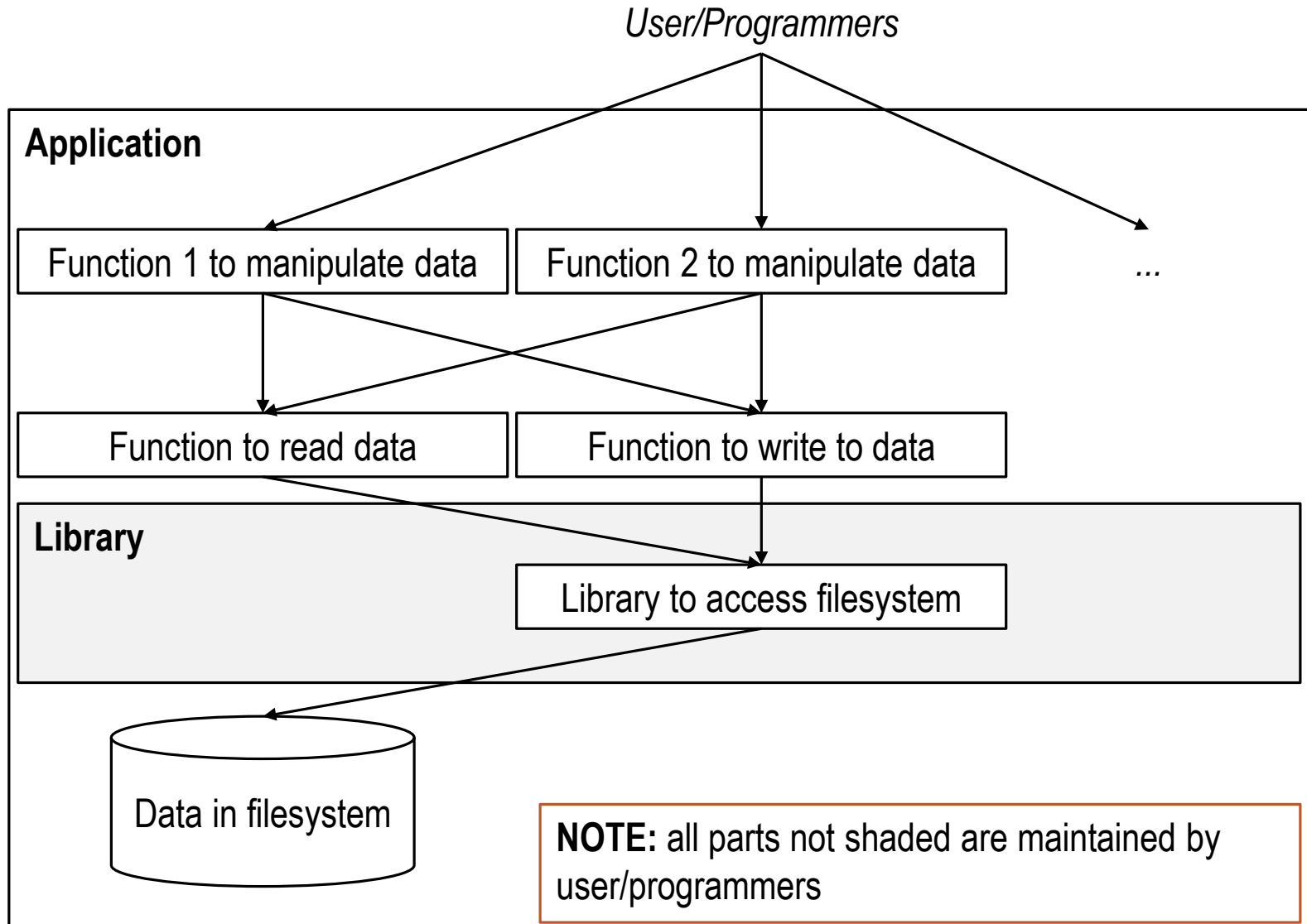
File processing technique

- Pseudo-code
 - initialize some book-keeping information I
 - open data file F
 - while (F is not empty)
 - read next record r from F
 - if (r satisfies some condition) then
 - do something with r
 - update I if necessary
 - do something with I if necessary
 - close file F

Database management system (DBMS)



Traditional data processing



Database management system (DBMS)

What is a DBMS?

- Software for managing data

Advantages of a DBMS

- Data independence
 - Efficient data access
 - Data integrity & security
 - Data administration
 - Concurrent access & crash recovery
 - Reduced application development time
- 

Database management system (DBMS)

Study of DBMS

- Database design
 - How to model the data requirements of applications
 - How to organize data using a DBMS
 - **Topics: relational model, ER model, schema refinement**
- Database programming
 - How to create, query, and update a database
 - How to specify data constraints
 - How to use SQL in applications
 - **Topics: SQL, relational algebra/calculus**
- DBMS implementation
 - How to build a DBMS ([covered in CS3223](#))

Database management system (DBMS)

Describing data in a DBMS

- A DBMS allows users to define and query data in terms of a data model
- A **data model** is a collection of concepts for describing data
- A **schema** is a description of the structure of a database using a data model
- A **schema instance** is the content of the database at a particular time

Data models

Types

- Network model *General Electric's IDS (1964)*
- Hierarchical model *IBM's IMS (1966)*
- Relational model
 - Commercial RDBMS
 - *IBM DB2, Microsoft SQL Server, Oracle, SAP ASE, etc*
 - Open-source RDBMS
 - *MariaDB, MySQL, SQLite, etc*
- Object-oriented model *ObjectStore (1988)*
- Object-relational model *PostgreSQL (1986)*
- etc...

Relational DBMS



(image: Software Engineering Daily)

Relational data model

History

- Introduced by **Edgar Codd** of IBM Research Lab in 1970
- Data is modeled using **relations**
- Relations are simply tables with rows & columns

<i>studentID</i>	<i>name</i>	<i>birthDate</i>	<i>cap</i>
3118	Alice	1999-12-25	3.8
1423	Bob	2000-05-27	4.0
5609	Carol	1999-06-11	4.3

- **Definitions**
 - **Degree/Arity** : *number of columns*
 - **Cardinality** : *number of rows*

Relational data model

Relation schema

- Each relation has a definition called a **relation schema**
 - Schema specifies **attributes** and **data constraints**
 - Data constraints include **domain constraints**
 - Students (*studentID*: integer, *name*: string, *birthDate*: date, *cap* : numeric)
- Each row in a relation is called a **tuple/record**
 - It has one **component** for each attribute of relation
 - Example: (1423, “Bob”, 2000-05-07, 4.0)

<i>studentID</i>	<i>name</i>	<i>birthDate</i>	<i>cap</i>
3118	Alice	1999-12-25	3.8
1423	Bob	2000-05-27	4.0
5609	Carol	1999-06-11	4.3