Database Systems
Lecture 2

Dr. Abid Sohail Bhutta

abidbhutta@cuilahore.edu.pk

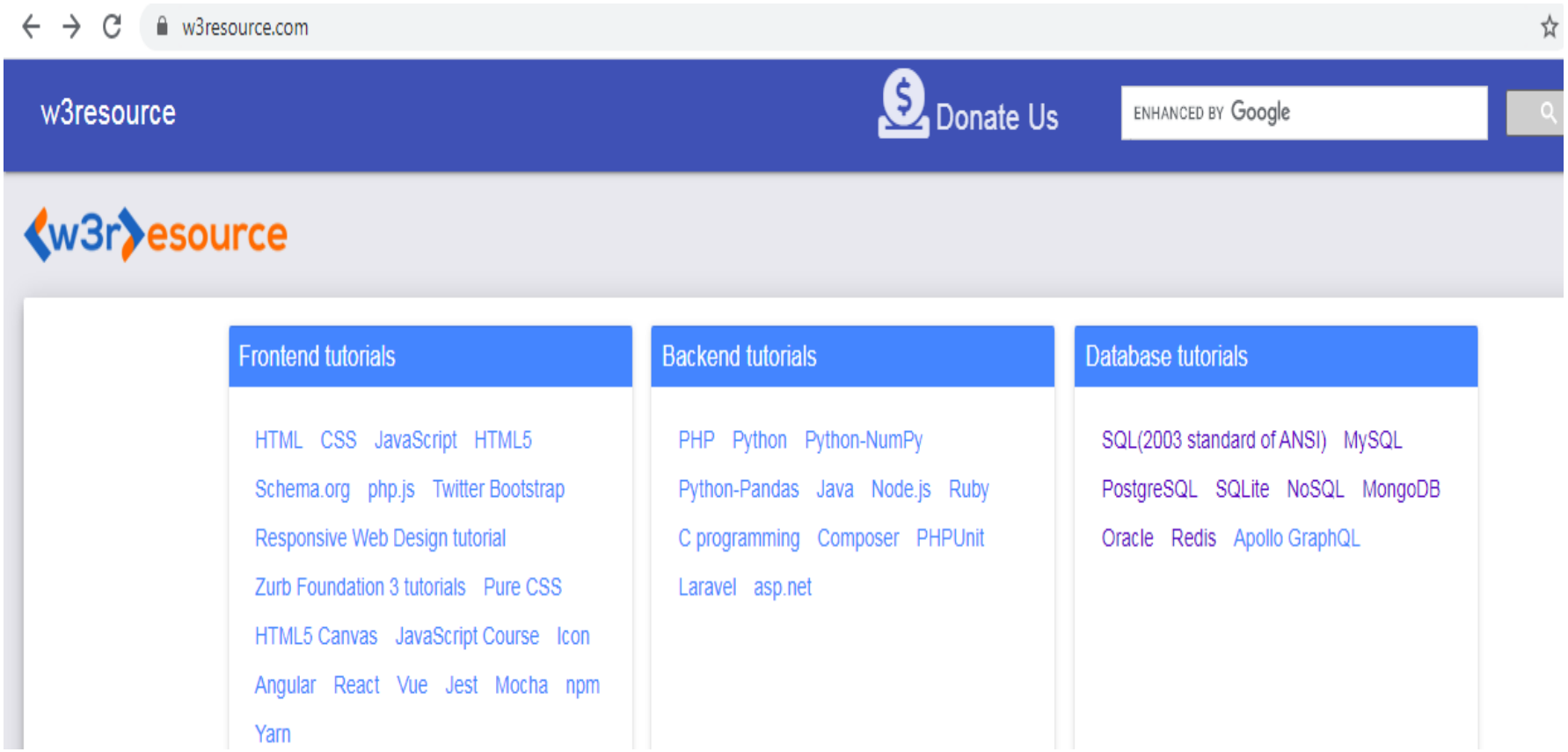
Department of Computer Science,

COMSATS University , Lahore Campus


بسم الله الرحمن الرحيم

Lecture 2

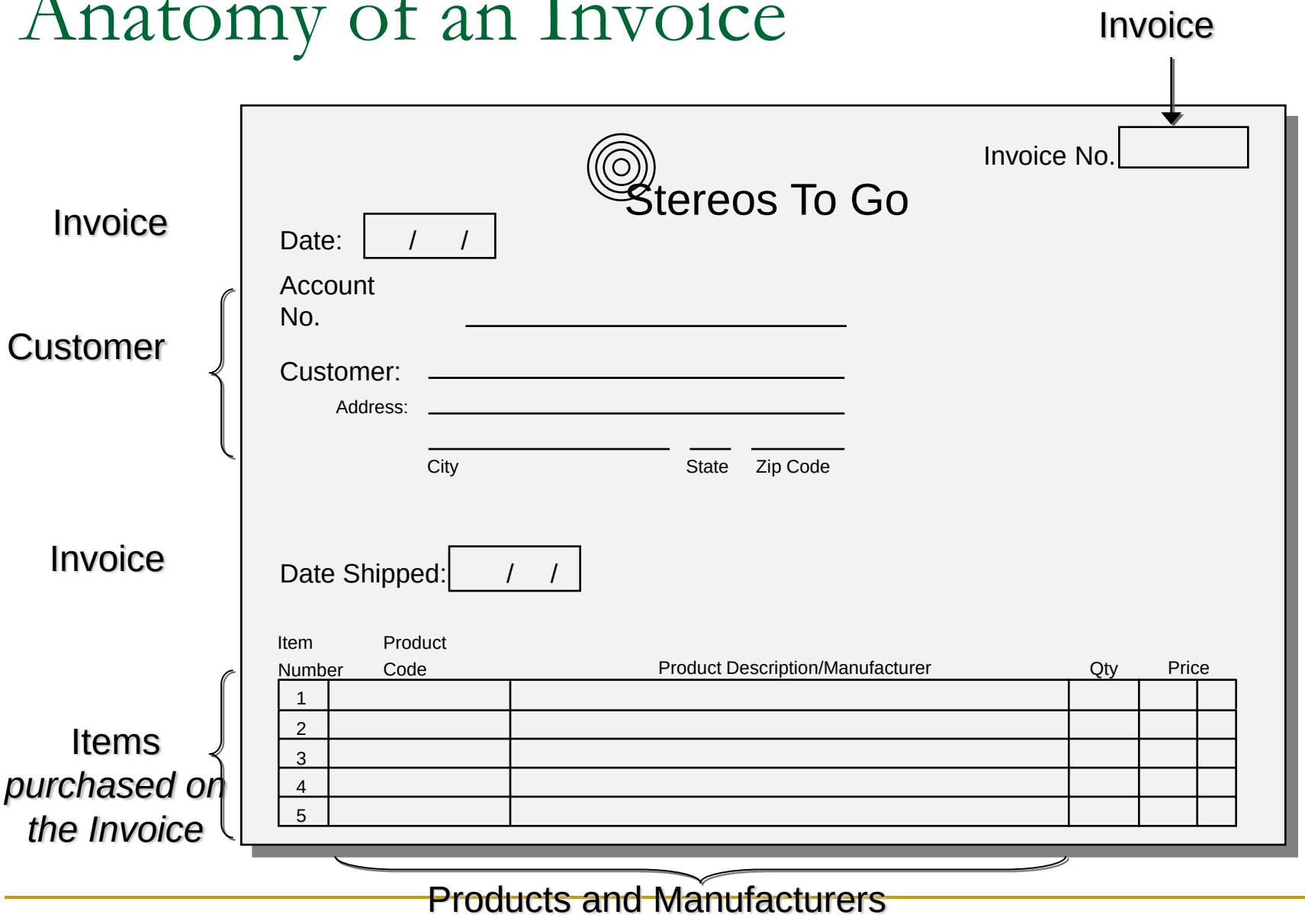
File Processing Systems vs Database Management Systems


w3resource.com
w3resource
Donate Us
ENHANCED BY Google
w3resource
Frontend tutorials
HTML CSS JavaScript HTML5
Schema.org php.js Twitter Bootstrap
Responsive Web Design tutorial
Zurb Foundation 3 tutorials Pure CSS
HTML5 Canvas JavaScript Course Icon
Angular React Vue Jest Mocha npm
Yarn
Backend tutorials
PHP Python Python-NumPy
Python-Pandas Java Node.js Ruby
C programming Composer PHPUnit
Laravel asp.net
Database tutorials
SQL(2003 standard of ANSI) MySQL
PostgreSQL SQLite NoSQL MongoDB
Oracle Redis Apollo GraphQL

Invoice No.

1			
2			
3			
4			
5			

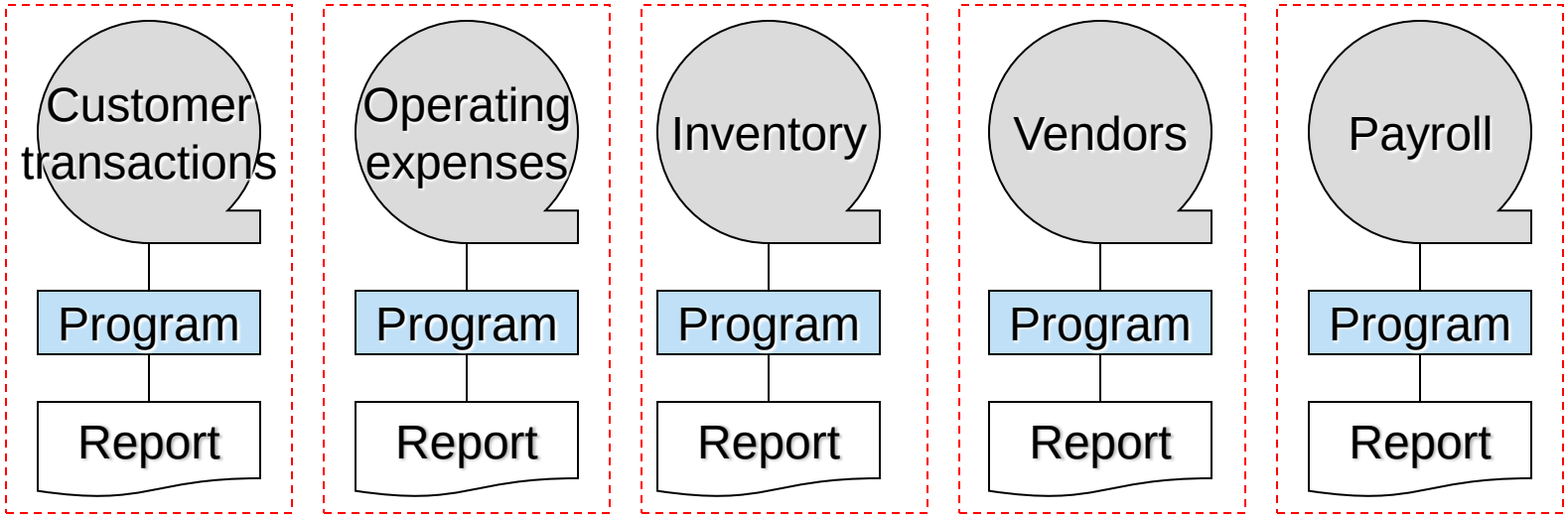
Anatomy of an Invoice



Traditional File-Based System

- Definition:

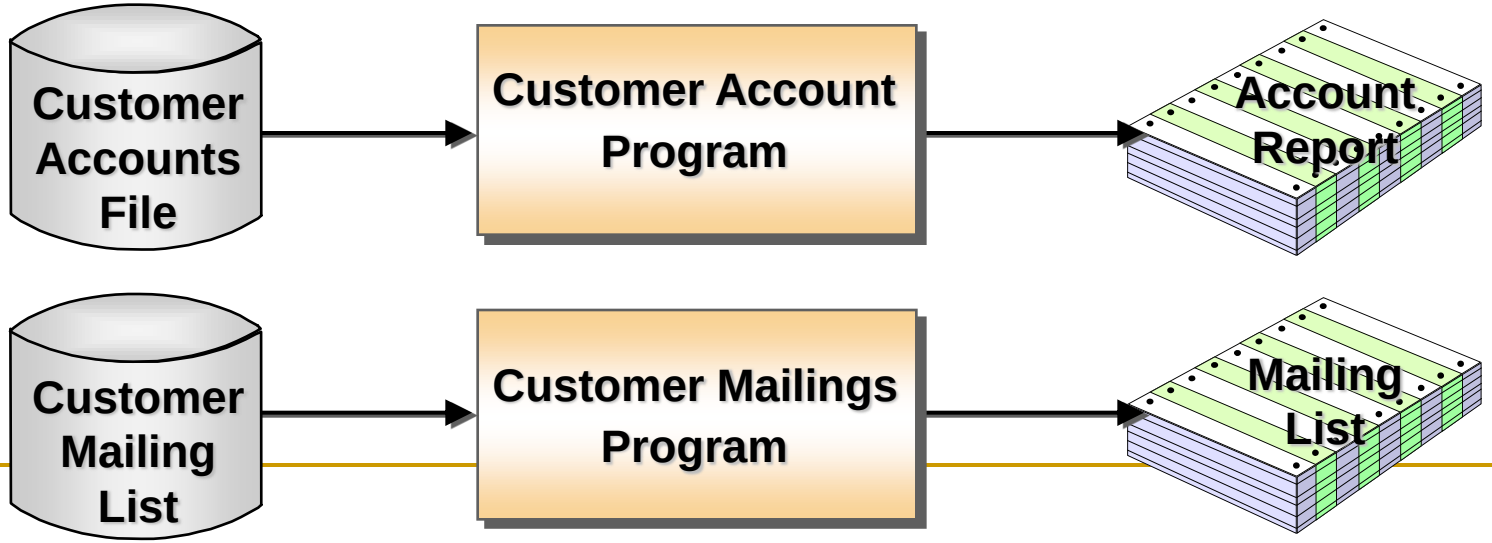
  "A collection of application programs that perform services for the end-users such as the production of reports. Each program defines and manages its own data."



One file, one application

Traditional File-Based System

"A collection of application programs that perform services for the end users such as the production of reports. Each program defines and manages its own data."



File-Based Systems

- Records contain *logically related* data
- Limitations:
  - Separation and isolation of data (*one file, one program*)
  - Duplication of data
    - Loss of data integrity - uncertainty of the *correct* version of data and no consistency
  - Data dependence - application program *defines* the data
  - Incompatibility of file formats
  - Fixed queries of application programs - little flexibility in meeting changing information needs

Data Redundancy

- Customer Order File
  - Invoice number
  - Customer account number
  - Customer name, address, city, state, zip code
  - Order date
  - Product code, product description, price, unit
- Customer Account File
  - Account Number
  - Customer name, mailing address, city, state, zip code
- Customer Mailing List File
  - Customer name, mailing address, city, state, zip code

Data Abstraction

- Separation between the data's structure (definition) and the application programs

COBOL

```
FD Master-File.
01  Master-Record.
      05  ID                          PIC X(10).
      05  Customer-Fname    PIC X(25).
          .
          .
          .
```

Tightly binds the data file and program

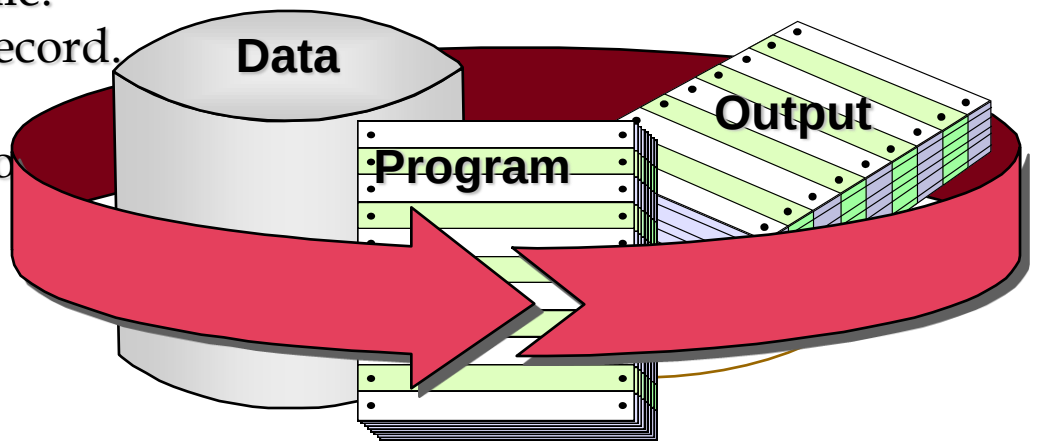
Data Abstraction

- Separation between the data's structure (definition) and the application programs

COBOL

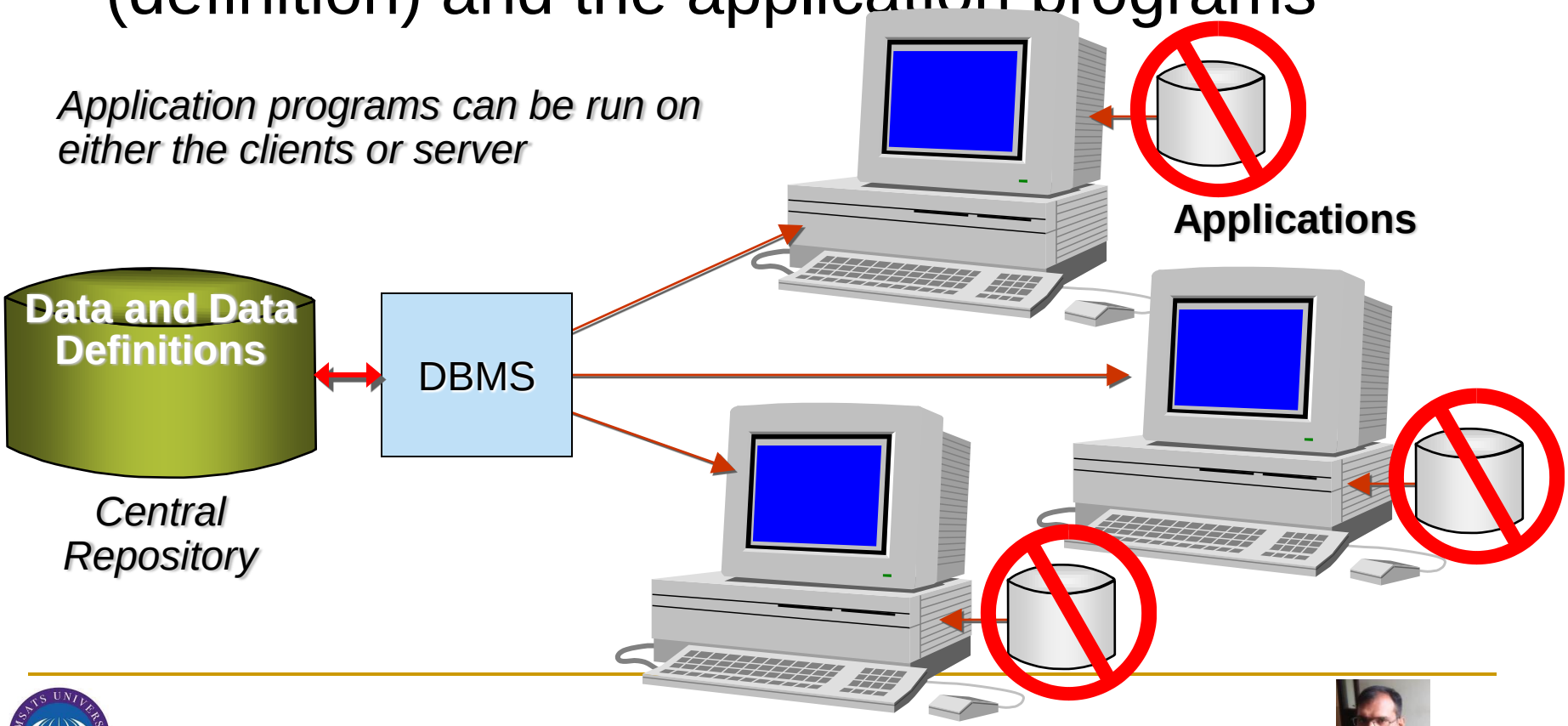
```
FD Master-File.
01  Master-Record.
      05  ID
      05  Custo
            .
            .
            .
```

Tightly binds the data file and program



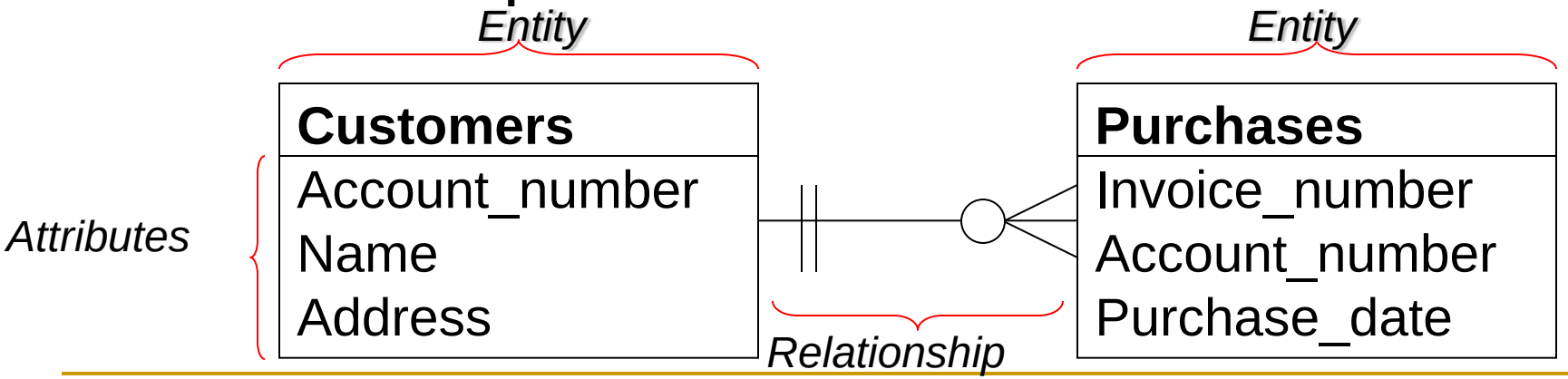
Data Abstraction

- Separation between the data's structure (definition) and the application programs

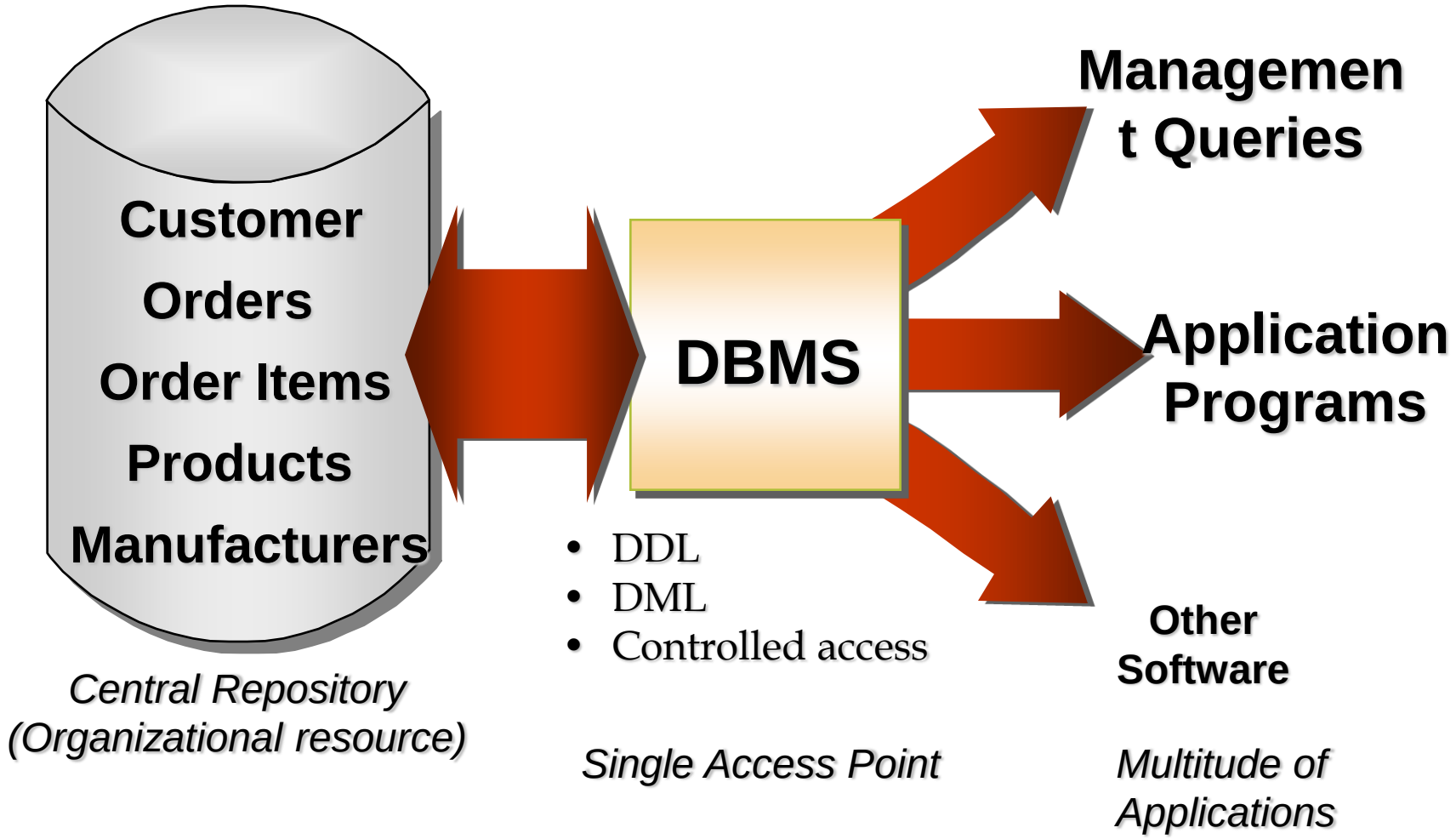


Organizing Data

- Entity - distinct object (i.e., person, place, thing, concept or event)
- Attribute - describes some aspect of the entity (object)
  - Property of the entity
- Relationship - association between entities



Stereos to Go Database



Components of a Database Environment

- Hardware
- Software: DBMS, application program and query software
- Data: Organized in a schema, partitioned into subschemas
- Procedures: Govern the design, access and use of the database
- People: Administrators (DA, DBA), designers (logical and physical), application developers and users (novice and *high-powered*)

Advantages of the Database Approach

- Control of data redundancy
- Data consistency
- Greater informational gain, *more information from the same amount of data*
- Sharing data, *organizational resource (i.e., shared resource)*
- Improved data integrity, *validity and consistency*
- Improved access and security
- Enforcement of standards

Advantages of the Database Approach

- Economy of scale, *centralization and consolidation*
- Balancing of conflicting requirements, *DBA oversees data and data definitions*
- Improved data accessibility and responsiveness
- Increased productivity
- Improved maintenance through data independence
- Increased concurrency
- Improved backup and recovery services

Disadvantages of the Database Approach

- Complexity
- Size

Dedication of resources including technology and people infrastructures

- Cost of DBMS
- Additional hardware costs
- Cost of conversion
- Performance
- Higher impact of failure
- In a production environment, processing can be *slow*