

In the Lecture Series Introduction to Database Systems

502051

First Lecture

We discuss the rationale and motivate and outline the syllabus of the course

Why are we here?

Database is not taught at MIT



Database is just
another application

Database is subsumed
by fundamental
topics

Database Application

Can we devise a definition for a
``Database Application”?

“A database application is a collection of
data and the programs that allow the
manipulation of these data”

But this definition fits any computer
application ☹

Database Application (Examples)

Can you find examples of database applications?

- Banking
- University
- Airline reservations
- My address book
- The e-shop around the corner



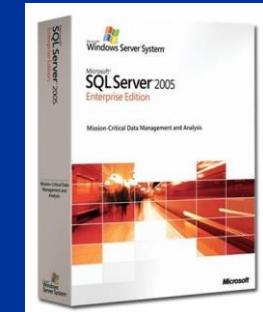
Database Management Systems

- We do not program database applications from scratch.
- We use Database Management Systems (DBMS)
- DBMS are generic platforms for the implementation and management of database applications



Database management Systems

- Example Database management Systems are:
 - Oracle
 - SQL Server
 - Sybase
 - DB2
 - PostgreSQL
 - MySQL
 - SQLite
 - MS Access



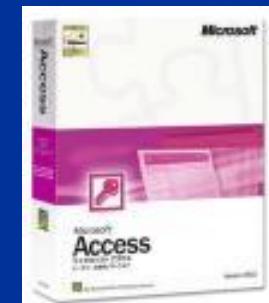
SYBASE®



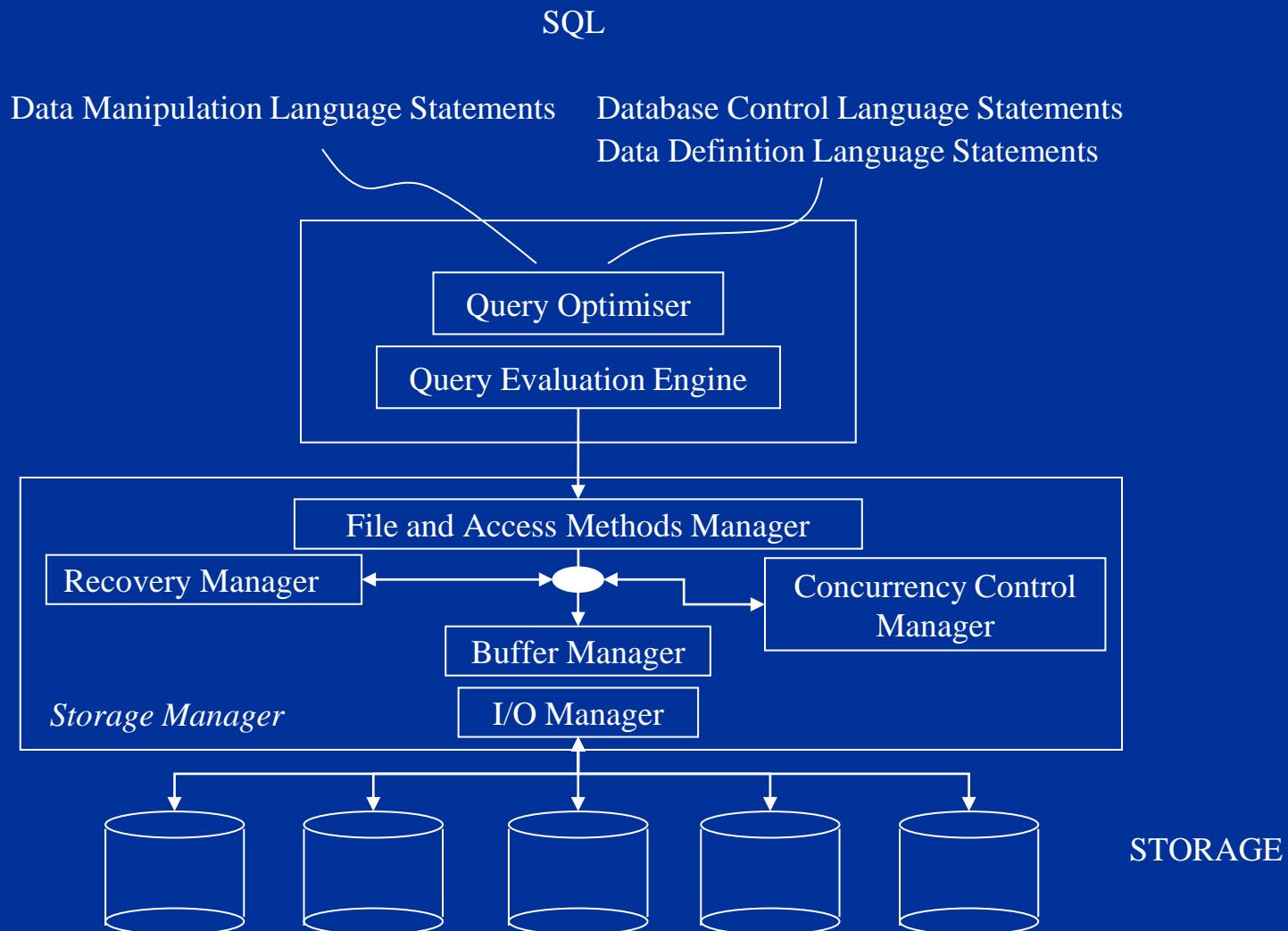
ORACLE®



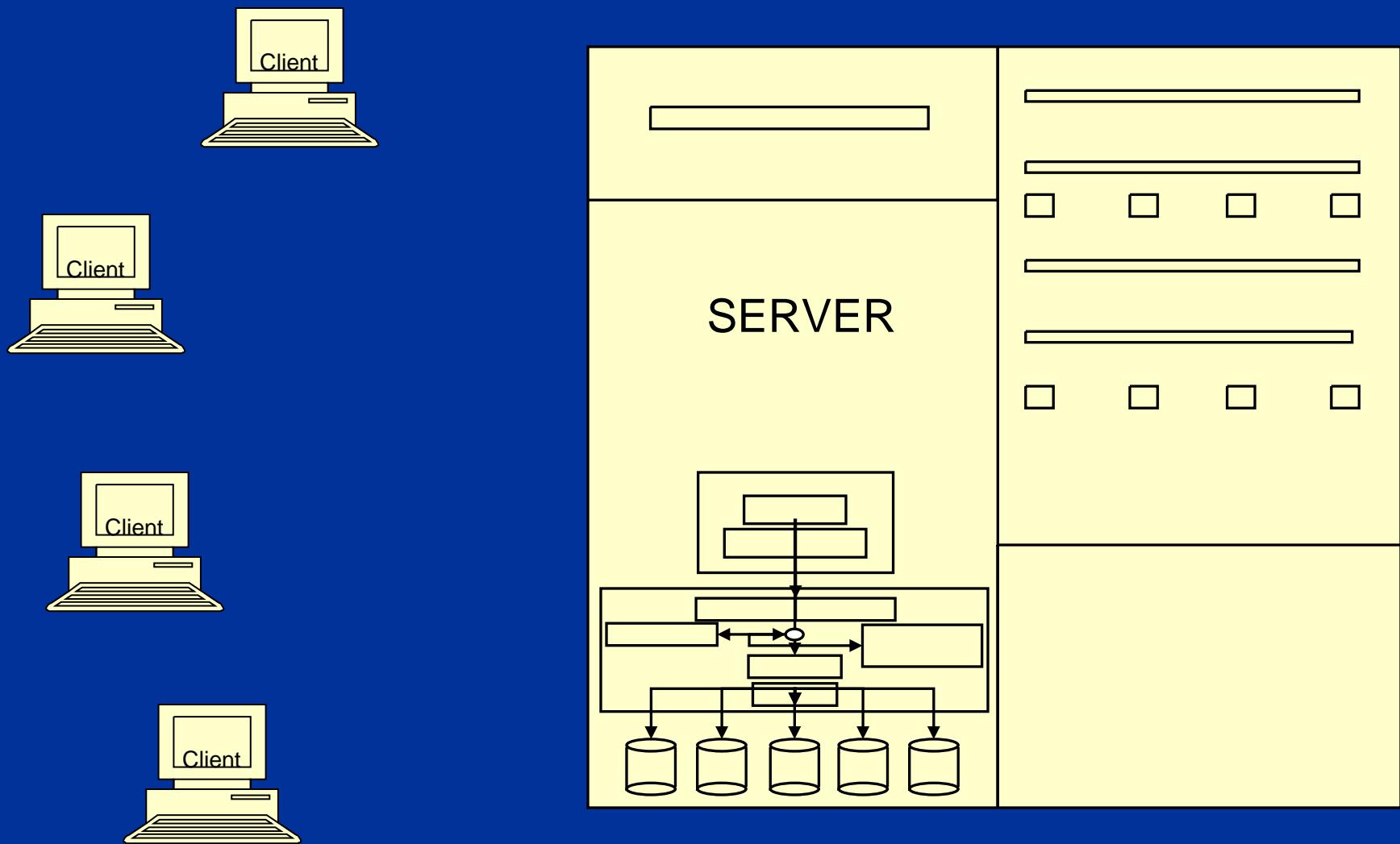
PostgreSQL



DBMS (simplified) Architecture



DBMS Client/Server Architecture



What is Specific about Database Applications?

- ***Database application***
 - Persistent data
 - Homogeneous collections
 - Large amounts of data
 - Structured data
 - Integrity
- Distributed and
- Concurrent
- Access
- Security

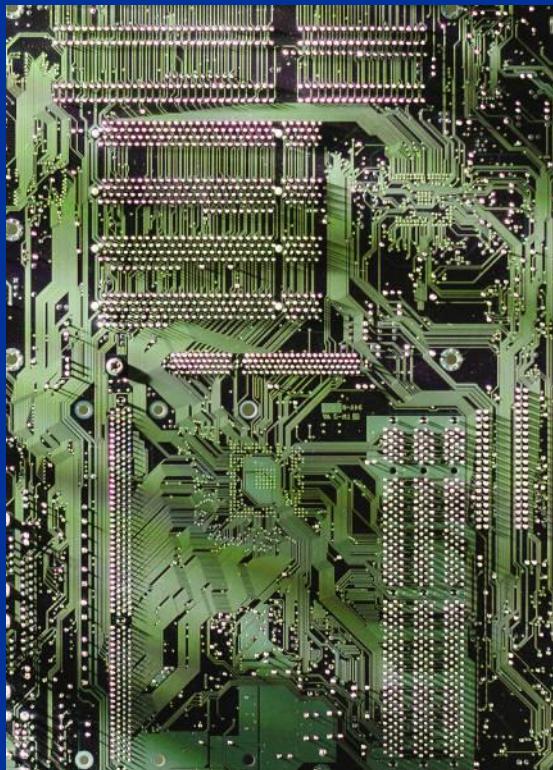
Data must Persist

How can data survive the process that created it, and be reused by other processes?



Data must Persist

Primary memory is
volatile



Secondary and tertiary
memories are
persistent



Data Comes in Large Amounts

- There were 190 million registered voters in the 2014 Indonesian elections
- Where could one store the names, identification numbers, and electoral districts of voters?



Data Comes in Large Amounts

- There were 190 million registered voters in the 2014 Indonesian elections
- How could one sort them by alphabetical order of electoral districts and names?



Data Comes in Large Amounts

When data is to be stored on secondary or tertiary storage, then we need to devise **efficient algorithms** taking into account the dominant cost of **Input/Output operations** (I/Os)

Such algorithms are called **external** algorithms (e.g., *external sort*)

Data Comes in Large Amounts

- There were 190 million registered voters in the 2014 Indonesian elections
- Imagine the original tapes contain duplicate entries
- Think about an algorithm to remove the duplicate entries



Data Comes in Homogeneous Collections

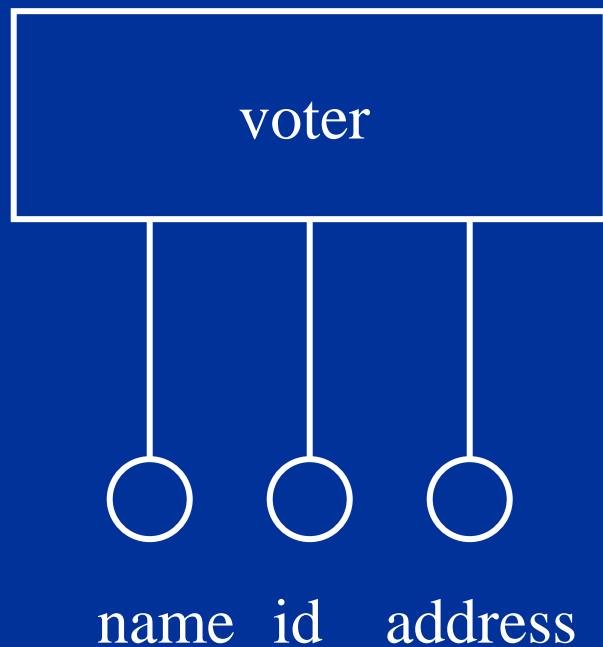


The Good News!

The DBMS implements

- access methods (the loop)
- and indexing and access methods for efficient storage, update, and retrieval

Data is Structured



The Good News!

- The DBMS supports data models

We can design applications around the data by defining the application schema

- The DBMS supports languages for data definition and manipulation

We can program applications using dedicated languages such as SQL

Data is Structured: the Good News!

- DDL: Data Definition Language. It includes statements to define the schema
- DML: Data Manipulation Language. It includes statements for creating, updating, and querying data

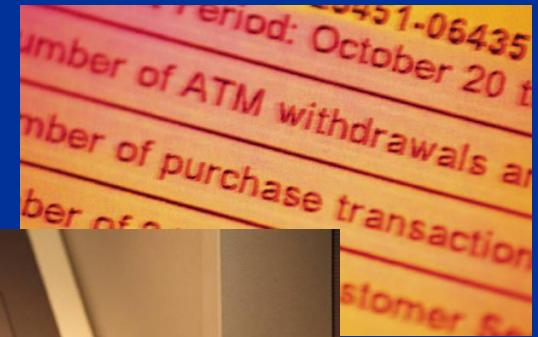
Data is Structured

```
CREATE TABLE voters  
    (first_name char(32),  
     last_name CHAR(32),  
     district CHAR(64),  
     national_id NUMBER)
```

```
SELECT last_name  
FROM voters  
WHERE first_name = 'Bambang'
```

Transactions

A **transaction** is a **logical unit of work**
carried out by a user or an application



In practice, there are different ways to specify the scope of transactions.
One of them is to use BEGIN END blocks in procedural SQL

- **Recovery**
 - **Atomicity:** all actions in a transaction happen or none happen
 - **Durability:** effects of successful transactions last
- **Concurrency Control**
 - **Isolation:** Transactions can be understood independently from each other
 - **Consistency:** If individual transactions would leave the application in a consistent state, a concurrent execution should do the same

Integrity of Data should be Maintained

How to maintain the integrity of data in spite of possible application, system, or media failures?

Consistent States

A consistent state of the database is a state which complies with the business rules as usually defined by integrity constraints

“students who have not passed cs2102 cannot take cs3223”

If the integrity constraints are violated by a transaction, the transaction is aborted and rolled back, otherwise, it is committed.

Distributed and Concurrent Access

How can data be shared by users and processes that are possibly distributed over a network?

Security and Access Control of Data is Critical

How to protect the data
and define and
control access to
data?



Definitions

- DCL: Database Control Language. It include statements to administer access privileges and transactions properties

In Summary

A database application manages **homogeneous collections** containing **large amounts** of **persistent structured data** that are **shared** among **distributed users** and processes and whose **integrity** must be maintained and **security** controlled

Conclusion

- We have **identified the typical requirements** of database applications
- We have **identified Database Management Systems** as the platforms for database applications
- We have **identified the topics to study** in this course: design and programming

Syllabus

- **Design**
 - Entity Relationship Model
 - Relational Model
 - Normalisation with Functional Dependencies
- **Programming**
 - Theory of Query Languages: algebra and calculi
 - SQL
 - SQL and Programming Languages

Credits

The content of this lecture is based
on chapter 1 of the book
“Introduction to database
Systems”

By
S. Bressan and B. Catania,
McGraw Hill publisher

Clipart and media are licensed from
Microsoft Office Online Clipart
and Media

Copyright © 2016 by Stéphane Bressan

