Introduction to Database Systems (1)

SWEN304/ SWEN435

Trimester 1, 2024

Lecturer: Dr Hui Ma

Engineering and Computer Science

TE WHARE WĀNANGA O TE ŪPOKO O TE IKA A MĀUI

VICTORIA

UNIVERSITY OF WELLINGTON

slides by: Pavle Mogin & Hui Ma

People

- Tutors:
  - Kaan Demir demirkaan@ecs.vuw.ac.nz
  - Kun Huang huangkun@myvuw.ac.nz
  - Zhengxin Fang zhengxin.fang@ecs.vuw.ac.nz
  - Micky Snadden snaddemick@myvuw.ac.nz
  - Veryan Straight: straigvery@myvuw.ac.nz
  - William Jordan: jordanwill2@myvuw.ac.nz

- Class representative nominates:
  - SWEN304: Hanning Qu quhann@myvuw.ac.nz,
SWEN435: Quinten Smit <smitquin@myvuw.ac.nz>
  - please fill the form at: http://www.vuwsa.org.nz/class-representatives/

- Discord Server: https://discord.gg/az6usY4Ec7

Outline

- Fundamental assumptions
- Databases (DB) and data
- Database management systems (DBMS)
- Database systems (DBS)

- Reading:
  - Chapter 1 of the textbook
  - Lecture slides make use of material provided on the textbook's companion website

Introduction

- Fundamental Assumptions of Data Management:
  - databases provide data for *multiple* application programs
  - data in databases is accessed and manipulated *concurrently*
  - data in databases is *dynamic*, that is, may change over time
  - data in databases is *persistent*
  - the amount of data in databases can be huge

Introduction

- Our Goals:
  - understand the storage and retrieval of persistent data (principles)
  - understand technology for the management of data in databases (foundations, applications)

Some Immediate Consequences

- Integration of data from various sources:
  - completeness and redundancy freeness
  - utilization of secondary storage

- Data integrity:
  - never violate (static and dynamic) integrity constraints
  - constraints determined by the semantics of the data (and application programs)

Some Immediate Consequences

- Data security / safety:
  - protection against loss of data
  - protection against misuse of data

- Concurrent access to data:
  - synchronization
  - concurrent execution of application programs
  - utilize transactions (serializability)

Basic Terminology

- a database (DB) is a collection of related data that is well structured and stored permanently

- a database management system (DBMS) is a general-purpose software system that facilitates the process of *defining*, *constructing*, *manipulating*, and *sharing* databases among various users and applications.

- a database system (DBS) comprises a DBMS plus one or more databases

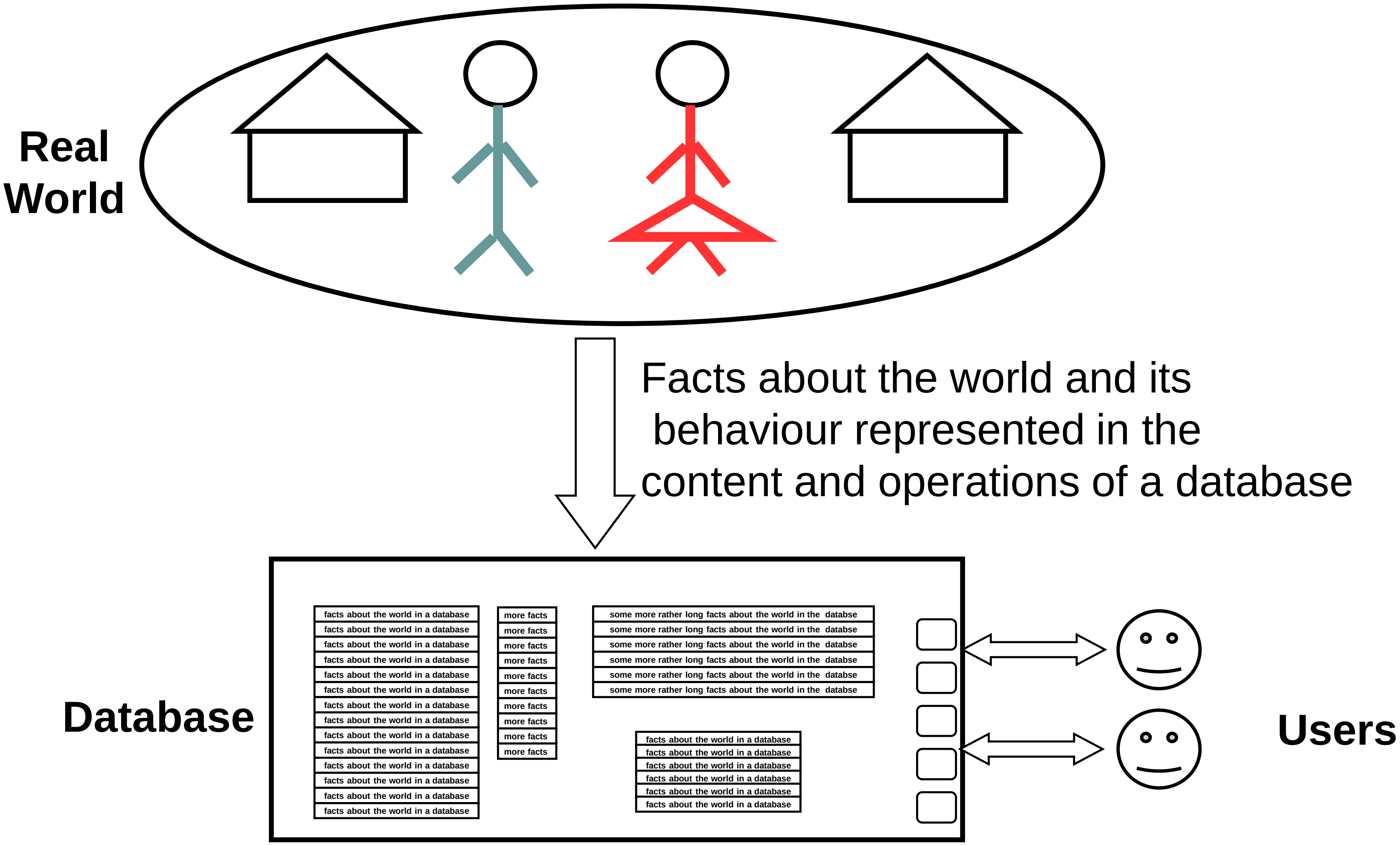
Basic Terminology

- Meta-data
  - Database definition or descriptive information
  - Stored by the DBMS in the form of a database catalog or dictionary
- Manipulating a database
  - Query and update the database of a miniworld
  - Generate reports

Databases

- Essential database characteristics are:
  - Represents an aspect of the real world, called miniworld or the universe of discourse (UoD),
  - Reflects (or should reflect) current state of the UoD,
  - We shall suppose it is well structured (even has a strict regular structure),
  - Has users and applications, and
  - Stored in a permanent (persistent) computer memory,
  - Managed by a Database Management System (DBMS)
- All these characteristics have to be met

What is a Database?


Real World
Database
Users
facts about the world in a database
facts about the world in a database
facts about the world in a database
facts about the world in a database
facts about the world in a database
facts about the world in a database
facts about the world in a database
facts about the world in a database
facts about the world in a database
facts about the world in a database
facts about the world in a database
facts about the world in a database
facts about the world in a database
facts about the world in a database
more facts
more facts
more facts
more facts
more facts
more facts
more facts
more facts
more facts
more facts
some more rather long facts about the world in the databse
some more rather long facts about the world in the databse
some more rather long facts about the world in the databse
some more rather long facts about the world in the databse
some more rather long facts about the world in the databse
some more rather long facts about the world in the databse
facts about the world in a database
facts about the world in a database
facts about the world in a database
facts about the world in a database
facts about the world in a database
facts about the world in a database

Facts about the world and its behaviour represented in the content and operations of a database

Example Commercial Database

- Amazon.com
  - 20 million books, CDs, videos, DVDs, electronices, apparel and other items
  - Occupies over 42 terabytes (1 terabytes = 1024GB)
  - Stored on 200 different computers
  - 15 million visitors access Amazon.com each day
  - the database is continually updated as new books/items are added to the inventory and purchases are transacted
  - 100 people are responsible for keeping the database up-to-date

A Simple Sample Database

- University database: information concerning students, courses, and grades in a university environment

STUDENT

id	lname	fname	major
300111	Smith	Susan	COMP
300121	Bond	James	MATH
300132	Smith	Susan	COMP

COURSE

course_id	cname	points	dept
SWEN304	DB sys	15	Engineering
COMP301	softEng	20	Engineering
MATH214	DisMat	15	Math

GRADE

id	course_id	grade
300111	SWEN304	A+
300111	COMP301	A
300111	MATH314	A
300121	COMP301	B
300132	COMP301	C
300121	SWEN304	B+
300132	SWEN304	C+

Questions for You

1. Is a book (like "Fundamentals of Database Systems") a database?
2. Is an old style library card catalog a database?
3. Is a bank statement a database?
4. Is a spreadsheet, containing contact information, a database?

Definition of Data (Datum)

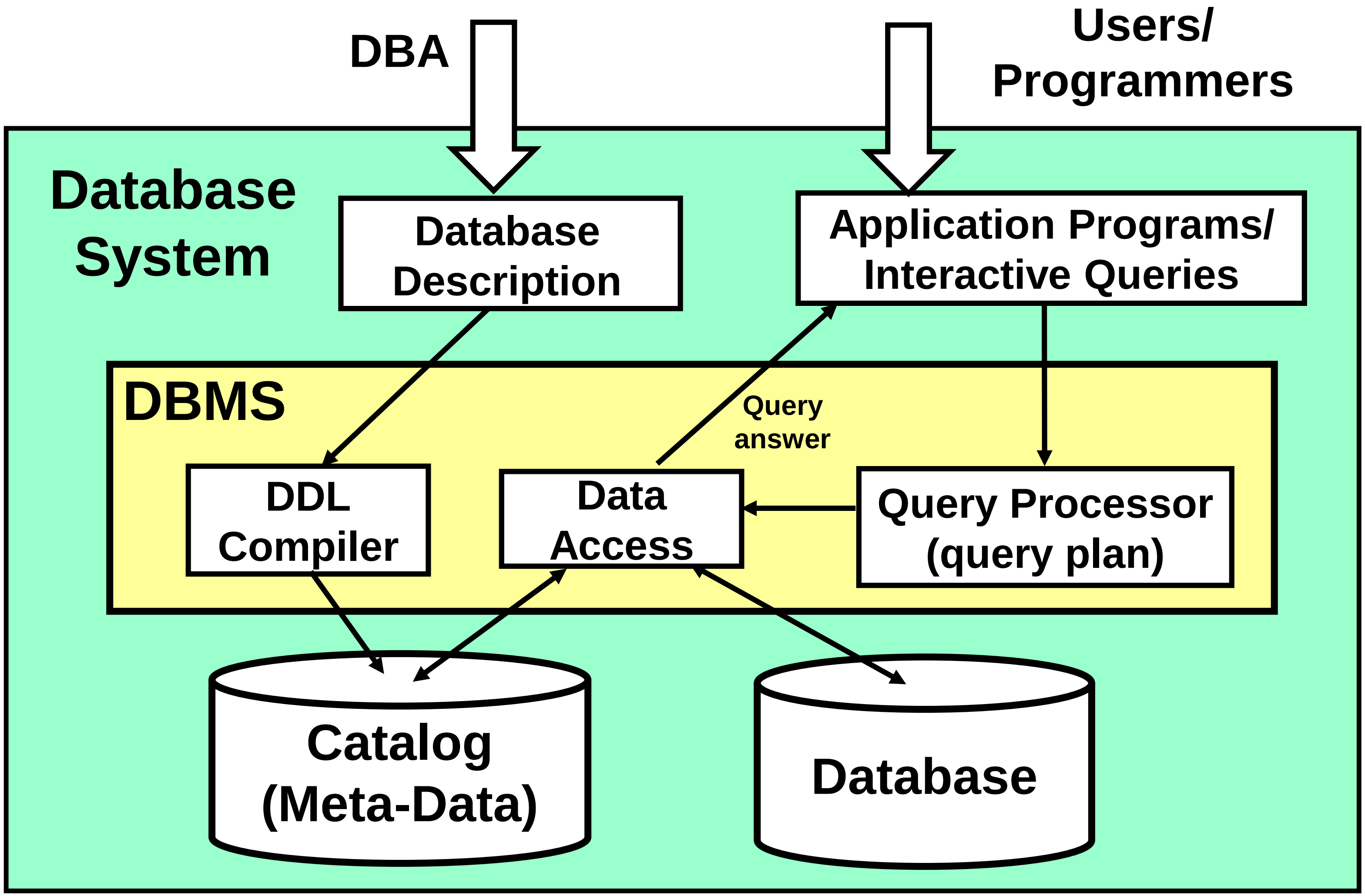
- **Data** is a value of
  - a *property* of an individual UoD object or
  - a *relationship* (between two UoD objects)

  at a particular period of time

- Example

UoD object(s)	James	James & CompSci
Property	Age	Number of Points
Time	July 2008	July 2008
Value	21	240

A Simplified Database System Layout



Typical DBMS Functionality

- **Define** a particular database in terms of its data types, structures, and constraints
- **Construct** or load the initial database contents on a secondary storage medium
- **Manipulating** the database:
  - Retrieval: querying, generating reports
  - Modification: insertions, deletions and updates to its content
- **Processing** and **Sharing** by a set of concurrent users and application programs
  - keeping all data valid and consistent

Typical DBMS Functionality

- Protection or Security measures to prevent unauthorized access

- Maintaining the database and associated programs over the life time of the database application

- Presentation and Visualization of data

Data Definition Example

Defining a table in SQL:

```
CREATE TABLE COURSE (
        course_id   CHR(4)    CONSTRAINT cspk PRIMARY KEY,
        cname       CHR(15)   NOT NULL,
        points      INT       NOT NULL CHECK (Points >= 0),
        dept        CHR(25)
    );
```

Query and Update Examples

- Retrieve a list of all surnames, course names and grades of 'James'

```
SELECT     lname AS SURNAME, cname, grade
 FROM      STUDENT s, GRADE g, COURSE p
 WHERE     FName = 'James'
 AND       s.id = g.id
 AND       p.course_id = g.course_id;
```

- Insert two records into STUDENT

```
INSERT INTO STUDENT (fname, lname, id)
 VALUES    ('Ann', 'Bole', 111111),
           ('Sharon', 'King' 121212);
```

Essential Roles in Data Management

- The database administrator (DBA) ‘owns’ the DBMS and is responsible for
  - authorizing access to the database
  - the maintenance of the physical schema
  - the decision on the physical storage structures and access methods
  - physical optimization and tuning

- The data engineer (or data administrator or database designer) ‘owns’ the database and is responsible for
  - the design of conceptual/logical and external schemata
  - specification of interfaces to application programs (queries, transactions)
  - liaison with current or potential users

Advantages of Using the Database Approach

- Controlling redundancy in data storage and in development and maintenance efforts
  - Data normalization
  - Denomalization: sometimes it is necessary to use **controlled** redundancy to improve the performance of queries
- Sharing of data among multiple users
- Restricting unauthorized access to data

Advantages of Using the Database Approach

- Providing persistent storage for program Objects (in Object-oriented DBMS’s)
  - Complex object in C++ can be stored permanently in an object-oriented DBMS
  - Impedance mismatch problem: object-oriented database system typically offer data structure compatibility
- Providing storage structures for efficient query processing
  - Index
  - Buffering and catch
  - Query processing and optimisation

Advantages of Using the Database Approach

- Providing backup and recovery services
- Providing multiple interfaces to different classes of users
- Representing complex relationships among data
- Enforcing integrity constraints on the database
  - Referential integrity constraint
  - Key or uniqueness constraint
- Drawing inferences and actions using rules
  - E.g. triggers and stored procedures

Summary

- A database is a collection of related data that is well structured and stored permanently
- A data (datum) is a value of an real object’s (or of a relationship between two objects) property in a perceived moment of time
- A DBMS is a set of programs that allows a comfortable database usage:
  - Defining
  - Populating by data,
  - Querying,
  - Preserving consistency,
  - Protecting from misuse,
  - Recovering from failure, and
  - Concurrent using

Plan for the next lecture

- Data models
- Schemas and instances
- The three schema architecture
- Data independence
- Database users and languages

- *Reading:*
    - *chapter 2 of the textbook*