Relational Database Systems

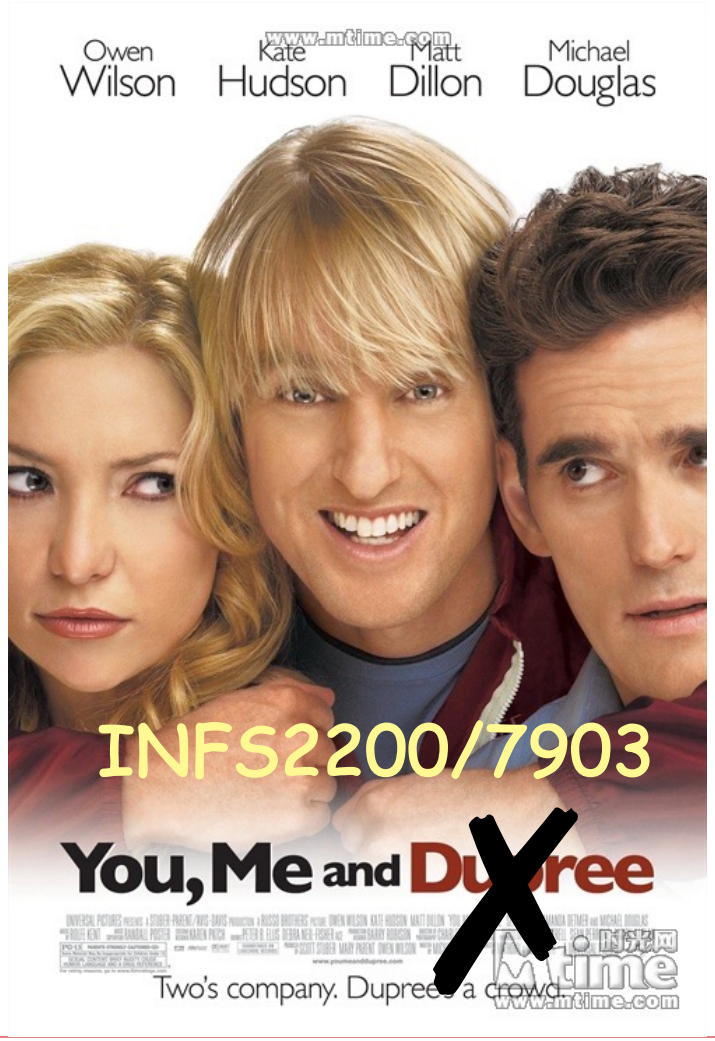
INFS2200/7903

Dr. Wen Hua

School of Information Technology & Electrical Engineering

Today…

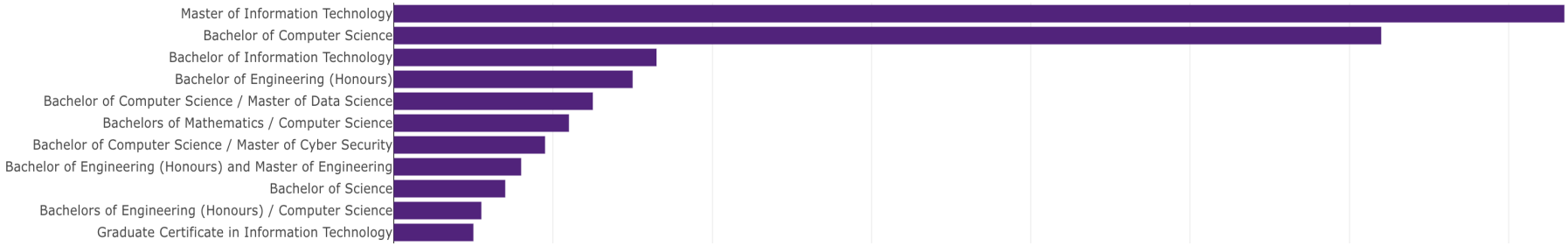
- You
- Me
- INFS2200/7903



Quick Facts About the 2022 Offering

Course	Enrollment
INFS2200 (Undergraduates)	338
INFS7903 (Postgraduates)	164

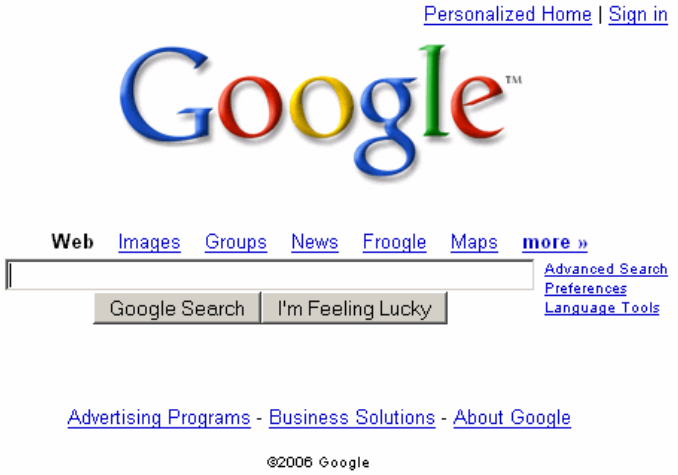
- 18% External Offering
- Multiple Programs:



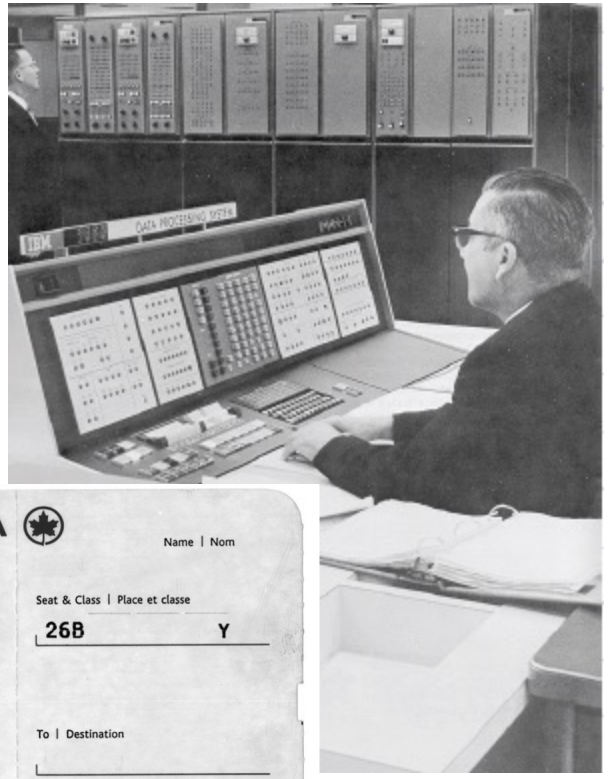
Why INFS2200/7903?

- **Top reasons to take INFS2200/7903**
  - It is a requirement
  - My mates are taking it too
  - Works with my timetable
  - I want to know how database systems work
  - I want a job in database systems
  - I want to do research in database systems

Databases Everywhere



Airline Ticket Reservation - Before



AIR CANADA

Class | Classe
ECONOMY CLASS / CLASSE ECONOMIQUE

Name | Nom

Flight & Date | Vol et date
AC 231

Gate | Porte
A12

Seat | Place
26B

Where not prohibited by law
Sauf où la loi l'interdit

Seat & Class | Place et classe
26B Y

Boarding time
Heure d'embarquement

From | De

To | Destination

To | Destination

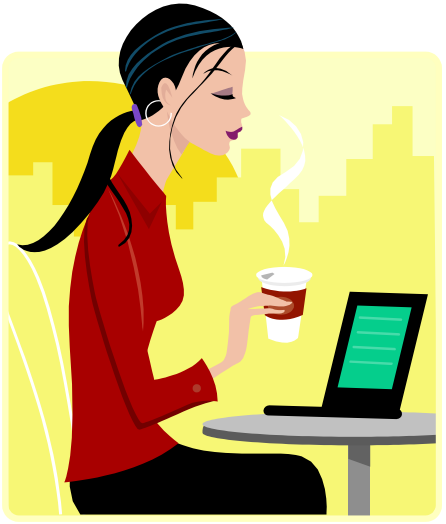
Name | Nom

Airline use | À usage interne
0081A YYC27670

Remarks | Observations

Boarding Pass | Carte d'accès à bord

Airline Ticket Reservation - After



-Users interact with databases on daily basis

-You build those systems!

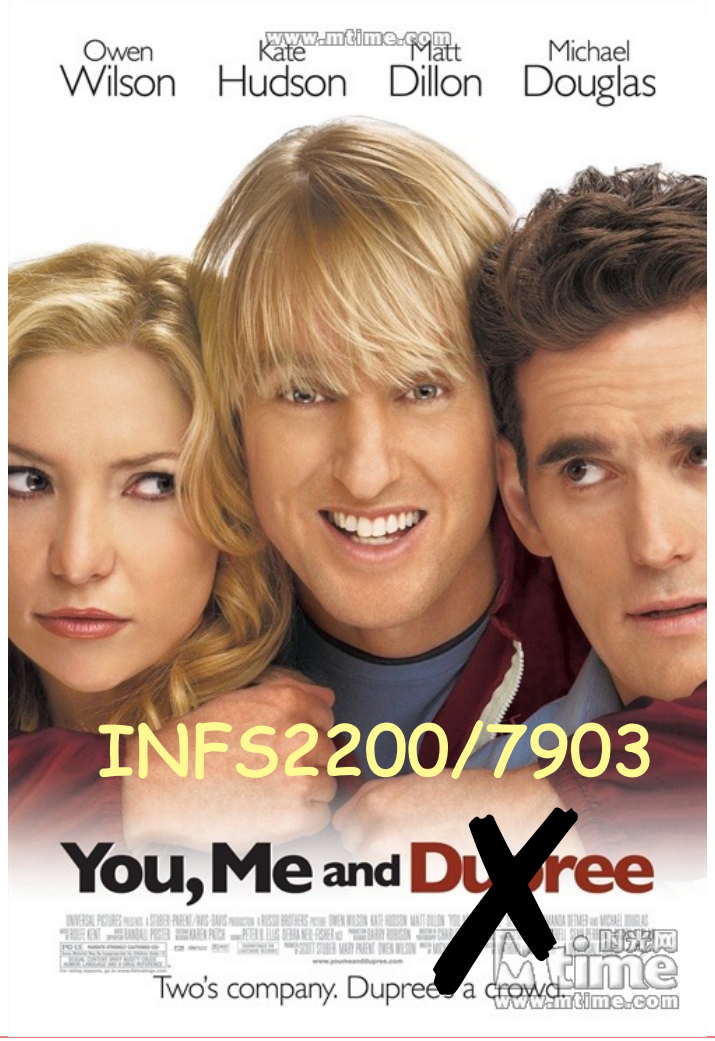
Big Data Job

Big Data Job


I DON'T ALWAYS STORE
MY GROCERY LIST ONLINE
BUT WHEN I DO, I SPIN UP LARGE
EC2 CLUSTERS WITH HADOOP,
HIVE, KAFKA, STORM AND MONGODB
imgflip.com

Today…

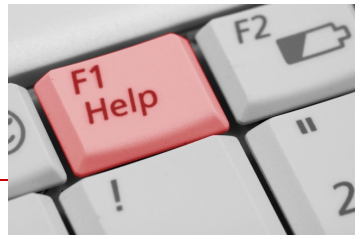
- You
- Me
- INFS2200/7903



Teaching Staff

Dr. Wen Hua	w.hua@uq.edu.au
9 tutors – tutorial and practical sessions	Contact information can be found on Blackboard

Help



- Discussions using **Ed Discussion Board**
  - Emails to the lecturer/tutor for specific questions; all other questions for discussion should be posted to Ed
  - Students are encouraged to help each other on Ed
  - The teaching team will monitor Ed and intervene when required
  - No solutions to exams or assignment can be posted on Ed

- Additional **consultation**:
  - Online consultation via Zoom
  - By email appointment with the lecturer/tutor

Lectures

- **When:**
  - Monday: 2pm-5pm
- **Where:**
  - 50-T203
  - https://uqz.zoom.us/j/89557936974
- **What:**
  - Lecture Notes
  - Ramez Elmasri and Shamkant B. Navathe. **Fundamentals of Database Systems**, 7th Edition

Tutorials & Practicals

- **Tutorials (11 x 1 hour, from week 3)**
  - Continuation of lectures with problem solving
  - Answers will be discussed during the tutorials, but you must be prepared before you come
  - In-person & Zoom (Details on Balckboard)

- **Practicals (11 x 1 hour, from week 3)**
  - Oracle-based
  - Hands-on application of theoretical concepts
  - Assignment preparation & consultation
  - In-lab or Zoom (Details on Blackboard)

COVID-Safe Teaching

- Please only attend the session you're signed up to, unless your course coordinator permits otherwise
- Use hand sanitiser on your way in and out of the classroom
- Use sanitising wipes to wipe down equipment / surfaces before and after use
- We highly encourage you to keep physical distancing, and wear a face mask if close-range discussion is needed
- If you feel unwell, please stay at home and get tested
  - Alternative arrangement is possible, e.g., Zoom attendance

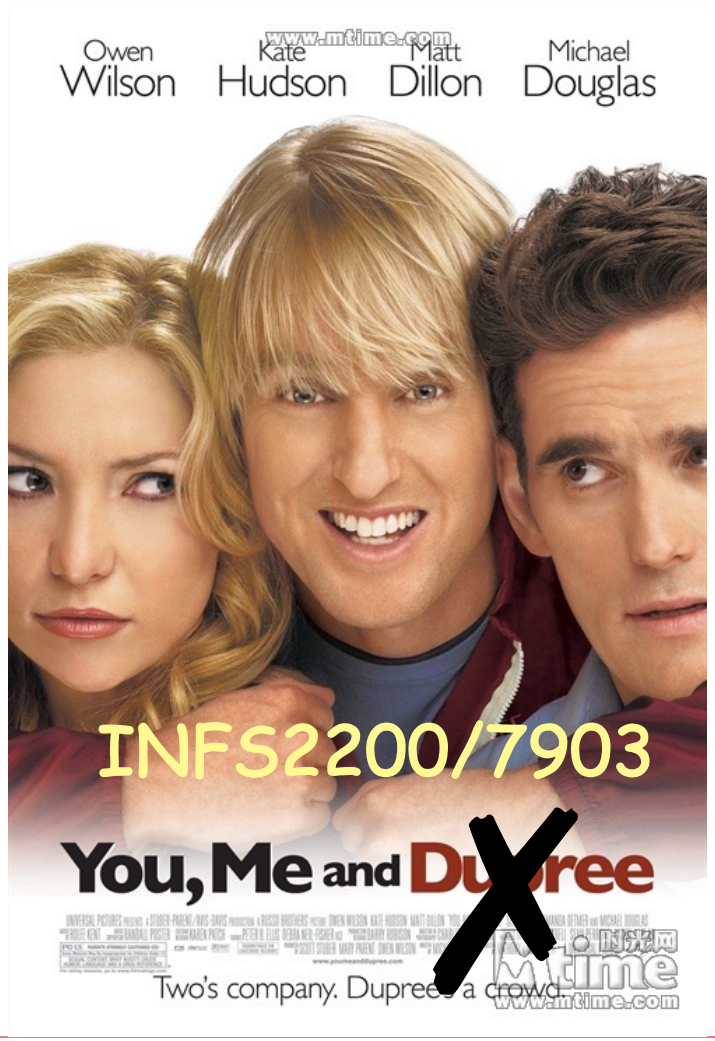
Assessments

- Project & Midterm Exam: Blackboard
- Final Exam
  - External: online invigilated exam
  - Internal: on-campus invigilated exam (online invigilated exam as a back up, if Queensland Health restrictions change)

Assessment	Percentage
Midterm Exam	15%
Project Assignment	25%
Final Exam	60%

Today…

- You
- Me
- INFS2200/7903



UQ INFS Courses

- INFS1200/7900 Information Systems
- INFS2200/7903 Relational Database Systems
- INFS3200/7907 Advanced Database Systems
- INFS3208/7208 Cloud Computing
- INFS3202/7202 Web Information Systems
- INFS4203/7203 Data Mining
- INFS4205/7205 Adv. Techniques for high-dimensional data
- INFS7410 Information Retrieval
- INFS7450 Social Media Analytics

These courses are offered by the **Data Science** Research Group in ITEE: one of the strongest database research groups in the world.

INFS1200/7900

- Introduction to databases
- Basic concepts required for DB design
- Understand the relational model
- Query a database system and write simple applications using MySQL.
- Basic architectures of Database Management Systems (DBMS)

INFS2200/7903

- In-depth understanding of Database Management Systems (DBMS) technology
  - System-Oriented
  - Understand the functions of a DBMS
  - Learn how these functions are implemented
  - Gain practical experience in applying this knowledge using commercial DBMS

INFS1200 vs INFS2200

INFS1200

INFS2200

Overview of Database Management Systems

What is a Database?

- A very large, integrated collection of data
- Models real-world enterprise (e.g., university)
  - **Entities** (e.g., students, courses)
  - **Relationships** (e.g., Bob is taking CSC 242)

SID	*Name*	*Age*	*GPA*
546007	Peter	18	5.8
546100	Bob	19	3.65
546500	Bill	20	6.7

Students

CID	*CName*
CS 242	DB
CS 207	SW
CS 369	OS

Courses

SID	*CID*	*Grade*
546007	CS 242	6
546007	CS 369	5
546100	CS 242	7

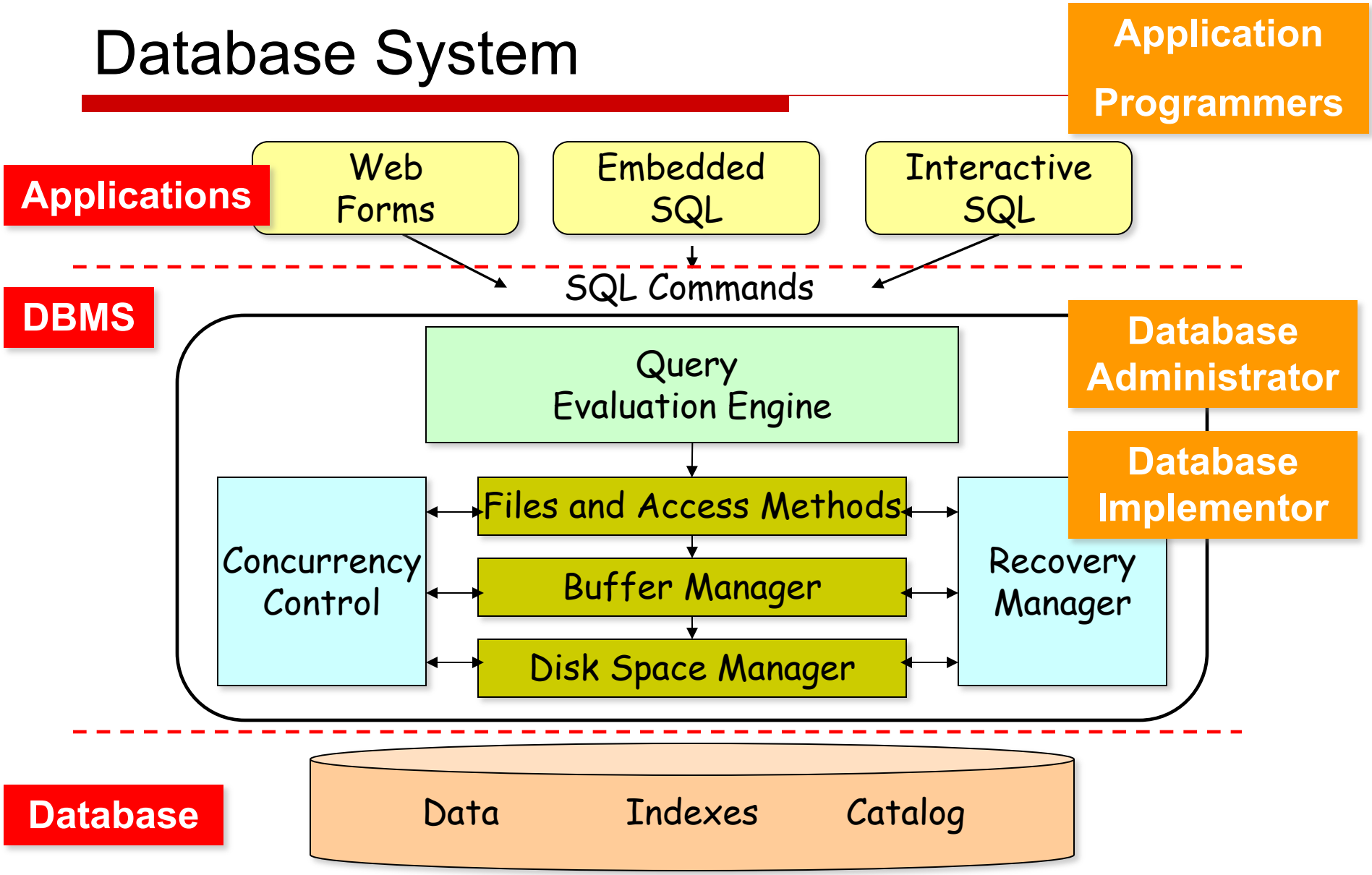
Enrollment

What is a Database Management System?

- ***Database Management System (DBMS):***
  - A software package designed to store and manage databases
- **Systems:**
  - Oracle, IBM DB2, MySQL, …
- **Usage:**

  Database System = DB + DBMS + Application Logic
  - Resource Planning Applications:
    - PeopleSoft, SAP, …
  - Web-based Applications:
    - amazon.com, ebay, orbitz, …

Database System

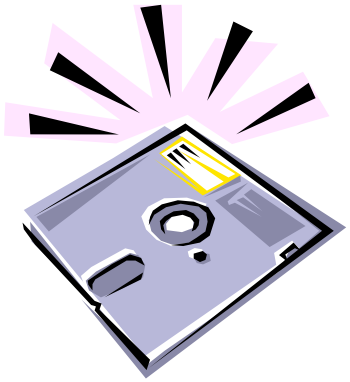


Weekly Plan

Week	Lecture	Tutorial	Practical
1	Course Introduction SQL & View		
2	Integrity Constraint		
3	Storage Management I	T1. SQL & View	P1. SQL
4	Storage Management II	T2. Integrity Constraint	P2. Table Constraint
5	Indexing I	T3. Storage Management I	P3. Trigger
6	Indexing II	T4. Storage Management II	Assignment1
7	**Midterm Exam**	T5. Indexing I	**Assignment1 Due**
8	Query Optimization I	T6. Indexing II	P4. View
9	Query Optimization II	Midterm Exam Discussion	P5. Index
Mid-Semester Break			
10	**Public Holiday**	T7. Query Optimization I	P6. Query Plan
11	Concurrency Control I	T8. Query Optimization II	Assignment2
12	Concurrency Control II	T9. Concurrency Control I	Assignment2
13	Database Recovery Course Review	T10. Concurrency Control II	**Assignment2 Due**
Revision Period			
Examination Period			

Approaches to Management of Data

- Database approach



- File system approach
  - Traditional (flat) files +
    C (Java, ...) programs to access them
  - E.g., use one (or more) UNIX files, with student records and their courses
  - Decide on a layout for the student records, etc…

Advantages of DBMS (vs. File System)

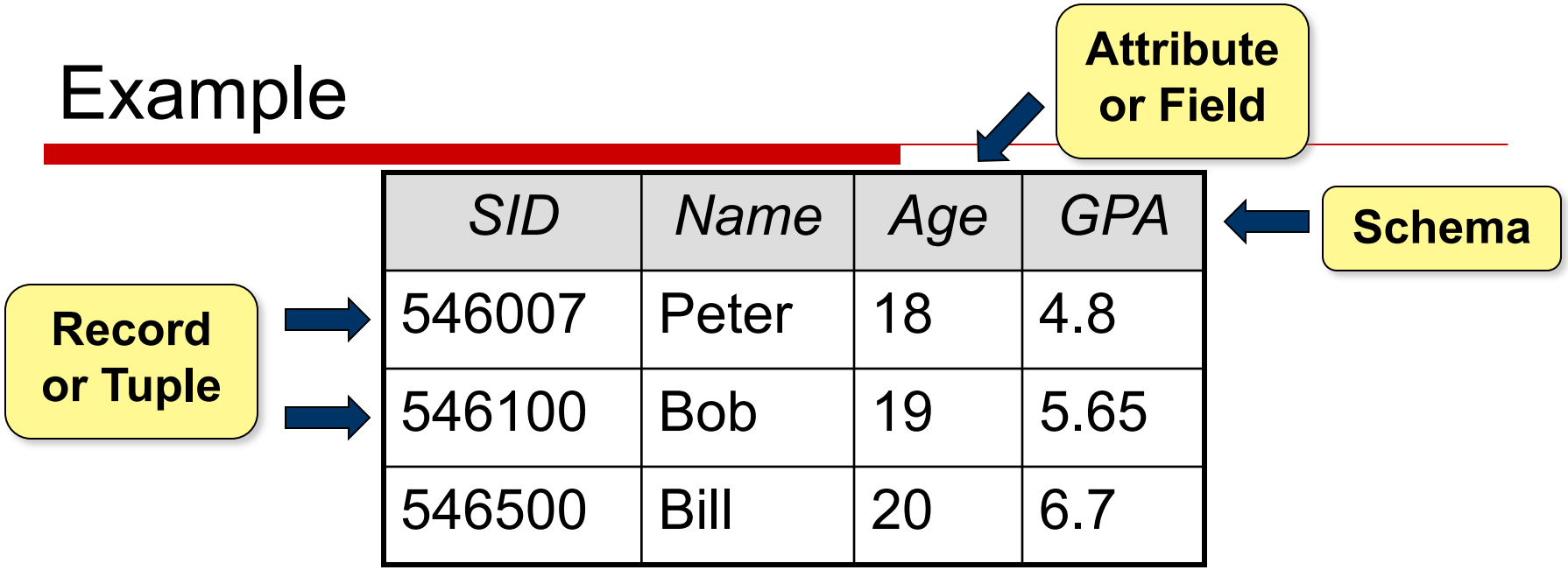
- **Data & Execution Abstraction**
- **Reliability**
- **Efficiency & Performance**

Data Abstraction

- **Data Model:**
  - A collection of high-level data **description** constructs that hide low-level storage details

- **The Relational Model:**
  - Is the most widely used data model today
  - Main construct is a ***relation***: table of records
  - Every relation has a ***schema***:
    - Relation name
    - Names of fields
    - Types of fields

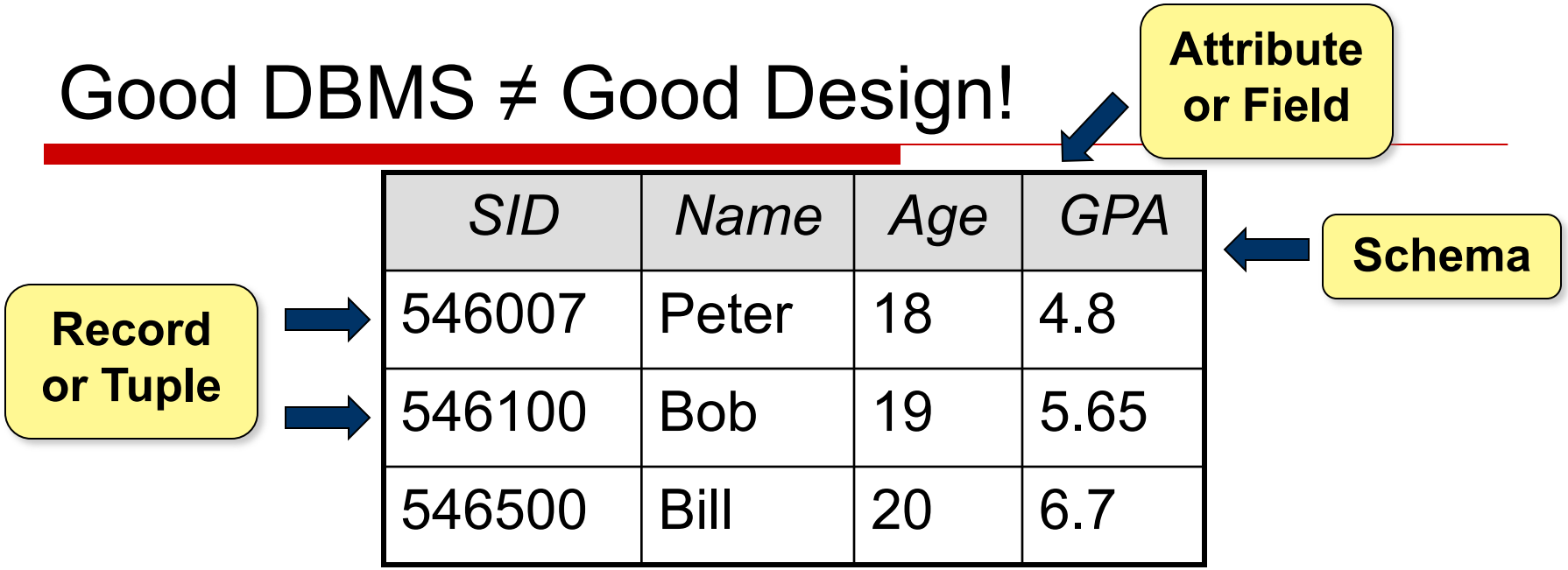
Example



SID	*Name*	*Age*	*GPA*
546007	Peter	18	4.8
546100	Bob	19	5.65
546500	Bill	20	6.7

- Schema:
  - Students (*sid:* `string`, *name*: `string`, *age*: `integer`, *gpa*: `real`)

Good DBMS ≠ Good Design!



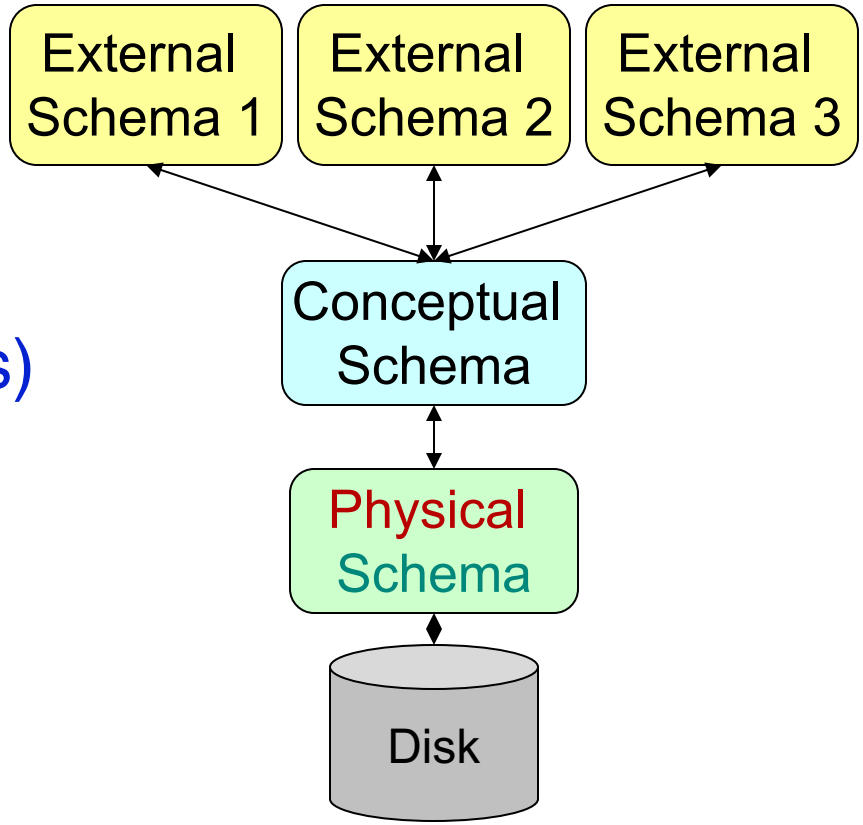
SID	*Name*	*Age*	*GPA*
546007	Peter	18	4.8
546100	Bob	19	5.65
546500	Bill	20	6.7

- Schema:
  - Students (*sid:* `string`, *name*: `string`, *age*: `integer`, *gpa*: `real`)
  - Alternative Schema:
    - Students (*sid*: `integer`, *fname*: `string`, *lname*: `string`, *dob*: `date`, *gpa*: `real`)

INFS1200/7900!

Levels of Data Abstraction in a DBMS

- The data in a DBMS is described at three levels of abstraction:
  1. *Conceptual Schema*
  2. *Physical Schema*
  3. *External Schema* (Views)



- Many external schemas
- plus one conceptual
- plus one physical

Database Languages

- **Data Definition Language (*DDL*):**
  - Define schemas
  - Define **Integrity Constraints**
    - Example: unique *SID*s
  - More…
- **Data Manipulation Language (*DML*):**
  - To ask questions = ***Query***
    - Example: Which students have GPA > 3.75?
  - To insert, delete and update data
- ***SQL:*** Most widely used database language

Execution Abstraction

- *A **transaction** is a **logical unit of work** in DBMSs*
  - *It is the execution of a **program segment** that performs some function or task by accessing shared data (e.g., a db)*
  - Logical grouping of query and update requests needed to perform a task

- Examples:
  - Deposit, withdraw, transfer money (banking transaction)
  - Reserve a seat on a flight (airline reservation)
  - Print monthly payment checks (business transaction)
  - Update inventory (inventory transaction)

Advantages of DBMS (vs. File System)

- **Data & Execution Abstraction**
- **Reliability**
- **Efficiency & Performance**

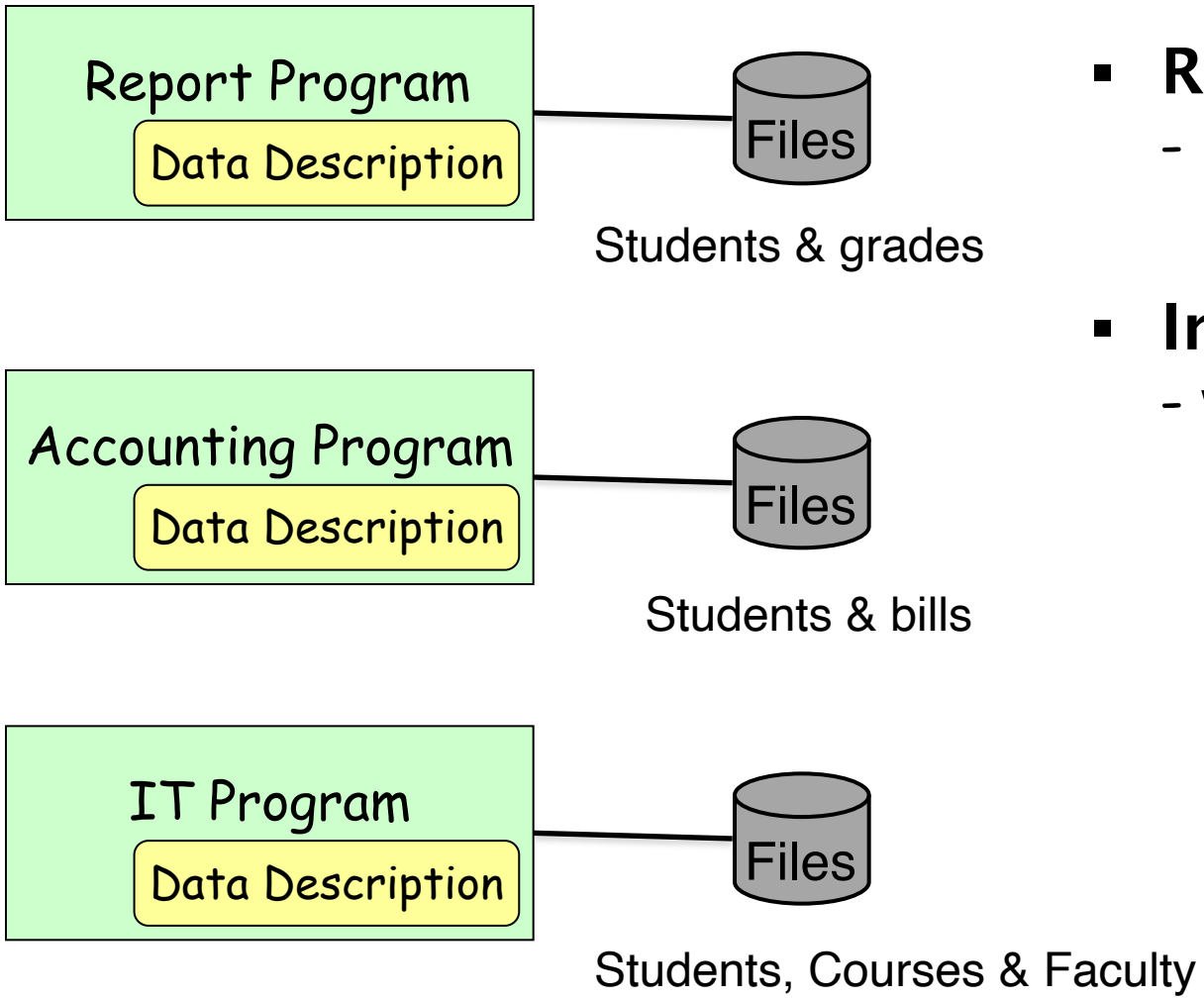
Reliability

- **Enforcing Integrity Constraints, such as:**
  - Data types
  - Value ranges
  - Certain rules on records
- **Backup and Recovery**
- **Restricting Unauthorized access**

Advantages of DBMS (vs. File System)

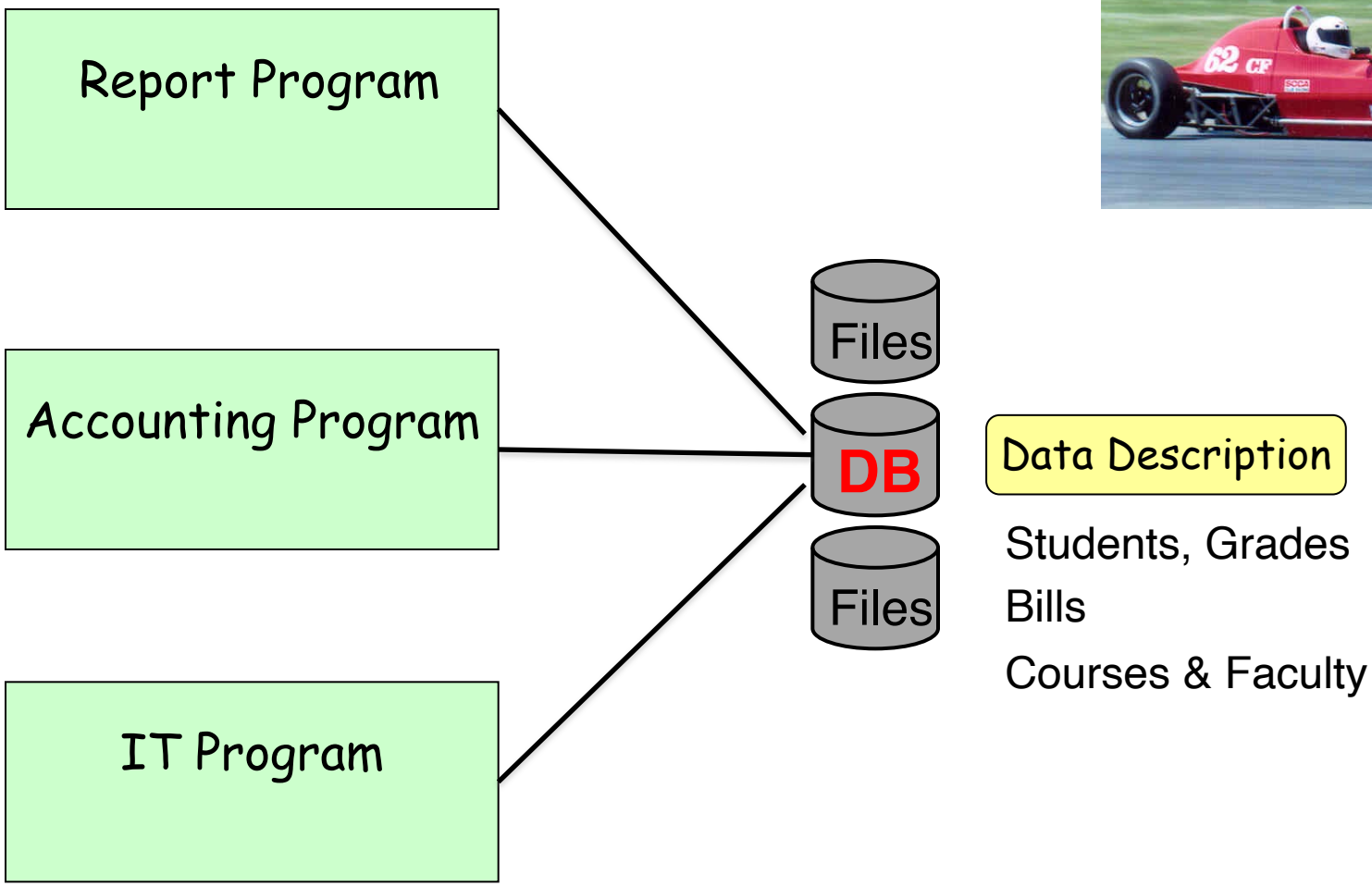
- **Data & Execution Abstraction**
- **Reliability**
- **Efficiency & Performance**

Performance Problems with Files



- **Redundancy**
  - waste of space
- **Inconsistency**
  - waste of effort

No Performance Problems with DBs



Efficiency and Performance

- Space efficiency:
  - Minimizes data redundancy by storing data only once
- Time efficiency (response time):
  - Eliminates the need for multiple updates to keep the replicas consistent and up-to-date
  - Enhances query performance by means of ***optimizations*** and ***access methods***
  - Allows many users (transactions) to access and share the database ***concurrently***

When an SQL-DBMS is Inappropriate?

- Disadvantages:
  - Price to buy (DBMS & Hardware)
  - Additional expertise (SQL/DBA)

- Hence, it is ***over-kill*** when
  - The database has simple structure and/or its size is small
  - The application is simple, special purpose and is not expected to change
  - Concurrent, multiple-user access is not required
  - Can tolerate failures