Information Technology

FIT2094 Databases

Week 1 - Introduction

Lindsay Smith

Lindsay.Smith@monash.edu

algorithm distributed systems **database**
systems **computation** knowledge ma
design e-business **model** data mining **int**
distributed systems **database** software
computation knowledge management an

MONASH University

Faculty of Information Technology
Industry Based Learning Program
Information Session

Information on eligibility and how to apply for the IBL program

WHEN ?   **Tuesday 8 August**

Attend <u>one</u> of two sessions: 11am – Midday or 2pm-3pm

WHERE ?   *Lecture Theatre M3*
                37 Rainforest Walk

RSVP ?   **You do not need to RSVP …**

But please come on time as there is limited seating (~120)

Your Teaching Team

Lecturers

Lindsay Smith (wks 1 – 6)

Aamir Cheema (wks 7 – 12)

Tutors

Vidya Saikrishna (Head Tutor)

Ahmad Choudhry

Tennyson Yuan

Tom Wei

Overview

- Teaching Method
- A summary of topics to be studied in FIT2094
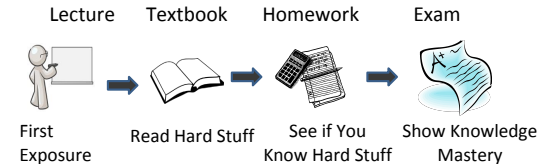
Slide 5

Teaching Method

- Your peers help you to understand the concepts through discussion.
- Lecture includes a series of discussions on concepts.
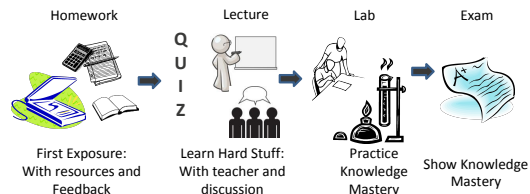- The lecturer guides the discussion.

Peer Instruction

Prof Eric Mazur, Harvard University

Slide 6

Traditional Teaching Method

Lecture	Textbook	Homework	Exam
First Exposure	Read Hard Stuff	See if You Know Hard Stuff	Show Knowledge Mastery

Slide 7

Peer Instruction – Full Picture

Homework	Lecture	Lab	Exam
First Exposure: With resources and Feedback	Learn Hard Stuff: With teacher and discussion	Practice Knowledge Mastery	Show Knowledge Mastery

QUIZ

Slide 8

Discussion Questions – Scenario

- Lecturer shows a question.
- Student answers using the response system. (no discussion – individual vote).
- *If uncertainty*
  - Group discussion (2-3 students) – need to get a consensus.
  - Student answers using the response system (group vote – everyone in the group still needs to vote).
  - Class wide discussion.

Why The Scenario?

- Pose carefully designed question
  - Solo vote: Think for yourself and select answer
    - Checks your understanding and create an opinion to base your discussion during the group discussion, if needed.
  - *If needed*
    - Discuss: Analyze problem in teams of 2-3
      - Practice analyzing, talking about challenging concepts
      - Reach consensus
    - Group vote: Everyone in group votes
      - You must all vote the same
      - Convince your group or get convinced by your group.
    - Class wide discussion.

Let's Practice

Using MARS

1. Visit http://mars.mu on your internet enabled device

2. Log in using your Authcate details (not required if you're already logged in to Monash)

3. Touch the + symbol

4. Enter the code for your unit: F5CQAT

5. Answer questions when they pop up.

Multiple choice questions

Q1: 1 + 1 = ?

Hint: There are 10 types of people in this world. Those who understand binary and those who don't.

- a.   2
- b.   10
- c.   11
- d.   Not sure

Text-based poll

Q2: Write the name of your favourite fruit.

Q3. What database management systems are you familiar with?

a. Oracle
b. MySQL
c. MS Access
d. SQL Server
e. others
f. I am not familiar with any database management systems.

Is it bad to get it WRONG?

NO

It is better to be WRONG and understand why you are WRONG, rather than, getting the RIGHT answer but NOT knowing WHY it is the RIGHT answer!

Why Peer Instruction?

▪ Learn/practice hard concepts in class
▪ Build and test one's understanding in a supportive environment.
▪ Develop critical thinking, communication and reflection skills.
▪ Engage students to take ownership of their learning.

Things are different…

- Pre-lecture activities are crucial.
  - Your lecture experience will depend on your preparation.
- Attending lectures is very important
- My lecture slides are NOT your notes!
  - Create your own notes during pre-lecture reading.
  - Annotate difficult concepts, revisit the annotation after lecture/tutorials.
  - It is better not to take notes during lecture. You should be prepared before the lecture, then think, discuss and ask questions during lectures.

Study Program

Week	Activities	Assessment	
0		No formal assessment or activities are undertaken in week 0	
1	**PART I: The Relational Model** Introduction to Database		
2	Relational Model	Pre-lecture Quiz Questions due weekly prior to the lecture (Weeks 2 to 11)	5%
3	**PART II: Database Design** Conceptual model - E/R Diagram		
4	Logical model - E/R Transformation		
5	Normalisation		
6	Database Design Language (DDL)		
7	**PART III: The SQL Database Language** SQL I	Assignment 1 due	20%
8	Data Manipulation Language (DML)		
9	SQL II		
10	SQL III	Assignment 2 due	10%
11	**PART IV: Web Database Implementation** Database Connectivity and Web Technologies - Querying data		
12	Web Technologies - Manipulating data	SQL Test	15%

Alexandria

FIT2094 2017 S2

FIT2094 2017 S2
FIT Database Teaching Team

FIT Database Teaching Team
Copyright © Monash University. All Rights Reserved.
Updated: 25 July 2017

CONTENTS

1 Introduction to SQL Developer

   1.1 Connecting to Oracle database using SQL Developer

   1.2 Using SQL Developer GUI to manage data

   1.3 Data Anomalies

2 The Relational Data Model FIT2094

3 Conceptual Modelling

   3.1 Building Conceptual Models

   3.2 Using Tools to draw ERD

4 Logical Modelling

   4.1 Logical Modelling – Task B – Rental Model

5 Normalisation

SYLLABUS

Download as ePub
Interactive (2.6 MB)

Download as PDF
Printable (2.9 MB)

Downloads updated: 25 July 2017


Overview

- An overview of relational database management systems (RDBMS)

Let's travel back to 1960s

- Relational databases do not exist yet
- Let's create a database to record the information on Monash students
    - What kind of approaches do we have?
    - What kinds of problems are involved?

1970: Relational model

- An IBM scientist
- Proposed and developed the relational model
- Also proposed normalisation forms
- Resistance from IBM to implement his model
- Turing award (1981)

- Relational model in week 2
- Normalisation in week 5

E.F Codd (1923-2003)

- E. F. Codd, "A Relational Model of Data for Large Shared Data Banks", *Comm. Of ACM*, 1970
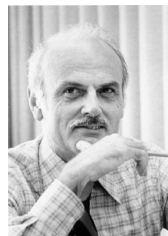
1974: SQL

- Developed at IBM
- Initially called SEQUEL (**S**tructured **E**nglish **QUE**ry **L**anguage)
- Doesn't strictly follow Codd's theory
- Oracle: the first commercially available implementation of SQL in 1979

Donald Chamberlin (1944-)

- SQL in weeks 7, 8, 9 & 10

- D Chamberlin, R Boyce, "SEQUEL: A structured English query language", *ACM SIGFIDET*, 1974
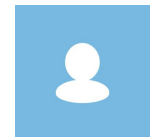
Raymond Boyce (unknown - 1974)

1976: Conceptual model

- Proposed Entity-Relationship Model (ER diagram)
- A systematic process to design a relational database

- Database design process in week 3 & 4

- Peter Chen, "The entity-relationship model—toward a unified view of data", *ACM TODS*, 1976

Peter Chen (1947 -)

1979: Oracle

- Inspired by Codd's ideas
- First commercial release in 1979
- Most popular RDBMS
- Introduced PL/SQL in 1988 (Procedural Language/SQL)

- Oracle SQL in week 7, 8, 9 & 10

Larry Ellison (1944 -)

1981: Transactions management

- Introduced transaction management
- Turing award (1998)
- Presumed lost at sea in 2007

- Transaction management in week 8

- Jim Gray, "The Transaction Concept: Virtues and Limitations ", *VLDB*, 1981
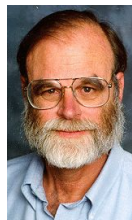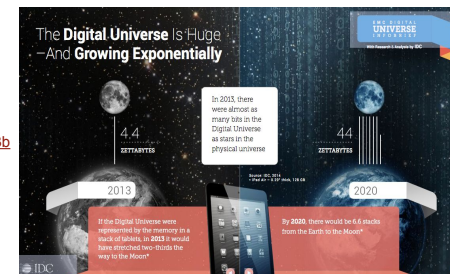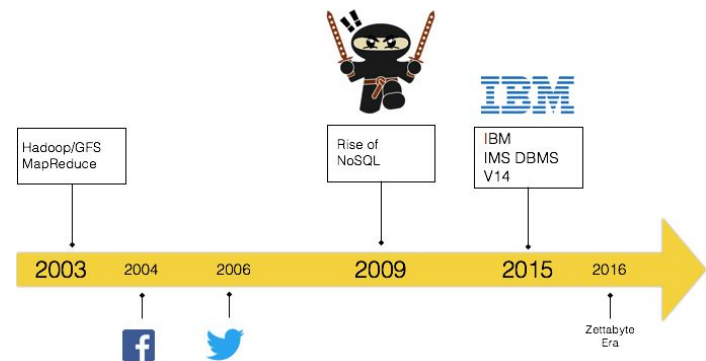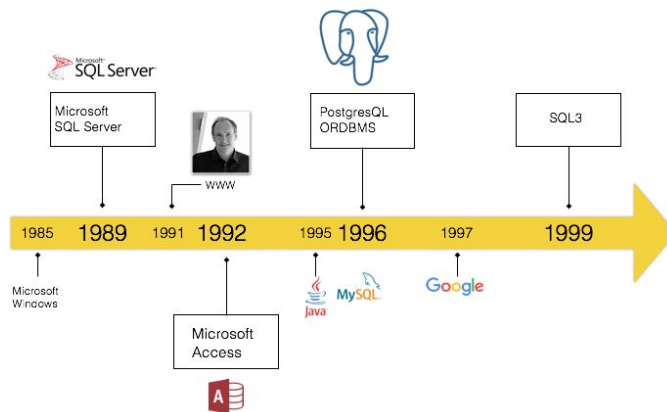
Jim Gray (1944 -)

Data management Today

- Relational databases are still very popular. But …
  - Social Networks (Facebook, Twitter, Foursquare etc.)
  - Multimedia data (YouTube, Pinterest, Facebook etc.)
  - Data streams (Twitter, computer networks)
  - Spatial data (Road networks, Google Earth, Space etc.)
  - Textual data
  - Web data
  - Big Data
  - …
    https://goo.gl/zMxG3b

http://db-engines.com/en/ranking

The Digital Universe Is Huge —And Growing Exponentially

Slide 29 — In Perspective ...

Timeline: Early Computers (1950s), JFK Moon Speech (1961), IBM System/360 (1964), "Database", IBM IMS DBMS (1966), ARPANET / Apollo 11 (1969)

Slide 30

Timeline: Relational Data Model (1970), Unix/C (1972), SQL / 1974, Entity Relationship Data Model (1976), IPv4 (1978), ORACLE (1979), IBM PC (1981), Apple Macintosh (1984)

Slide 31

Timeline: Microsoft Windows (1985), Microsoft SQL Server (1989), WWW (1991), Microsoft Access (1992), Java / MySQL (1995), PostgresQL ORDBMS (1996), Google (1997), SQL3 (1999)

Slide 32

Timeline: Hadoop/GFS MapReduce (2003), Facebook (2004), Twitter (2006), Rise of NoSQL (2009), IBM IMS DBMS V14 (2015), Zettabyte Era (2016)

Rank	DBMS	Type	Introduced
1	ORACLE	commercial Relational	1979
2	MySQL	open-source Relational	1995
3	Microsoft SQL Server	commercial Relational	1989
4	PostgreSQL	open-source Relational	1996
5	mongoDB	open-source NoSQL	2009
6	IBM DB2	commercial Relational	1983
7	CASSANDRA	open-source NoSQL	2008
8	MS Access	commercial Relational	1992
9	SQLite	open-source Relational	2000
10	redis	open-source NoSQL	2009

DB-ENGINES Feb 17

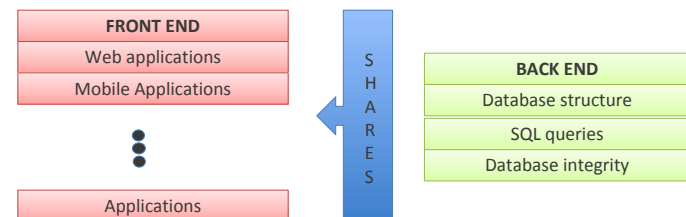## Relational database systems in action: End-users' view

Web Enrolment System

Enrolment / Re-Enrolment
Fees / Scholarships
Student Services
Course Progression
Enrolment Access Dates

network

Student database

Front end application (client)

Student Database is implemented in an Oracle DBMS (server)
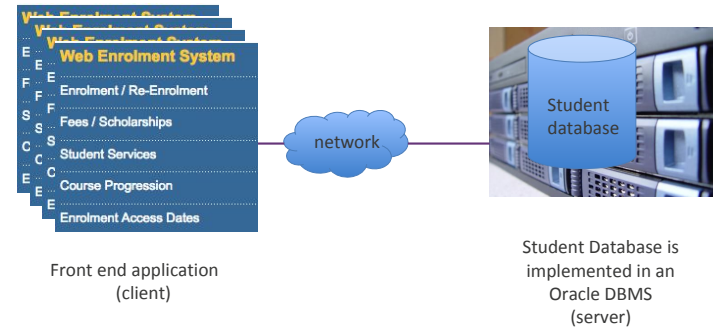
Database Systems in Action Developers' View

network

Student database

Development environment (client, eg SQL Developer, Integrated Development Environment for web scripting)

Student Database (server)
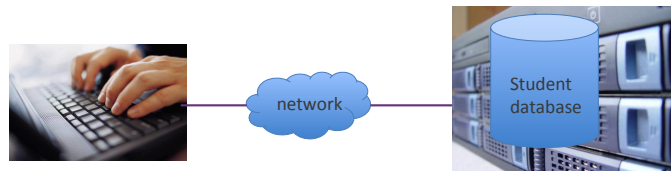
Developing Application with Database

FRONT END
Web applications
Mobile Applications

Applications

S H A R E S

BACK END
Database structure
SQL queries
Database integrity
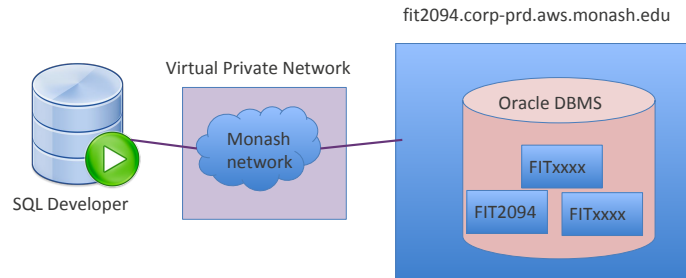
In this unit, we will concentrate on building the back end. Database designer.

Database Systems in FIT2094

fit2094.corp-prd.aws.monash.edu

Virtual Private Network

Monash network

SQL Developer

Oracle DBMS

FITxxxx

FIT2094   FITxxxx

Tutorials start from today