



MONASH  
University

MONASH  
INFORMATION  
TECHNOLOGY

Week 1 - Introduction

Semester 1 2022

Pre-Forum Slides

**Malaysia Campus**



# An overview of **DataBase** **M**anagement **S**ystems (DBMS)



# The challenge

- Let's create a system to record information on Monash students
  - student, unit and enrolment details
- What kind of approaches do we have?
- What kinds of problems are involved?

# Pre-Database Systems

- Manual System
  - recording data on paper/cards stored (filed) in folders/cabinets
  - management (insert/update and delete of data) and reporting are slow and cumbersome
- File Processing Systems
  - recording of data in computer based files

FIGURE 1.7 CONTENTS OF THE CUSTOMER FILE

Database name: Ch01\_Text

C_NAME	C_PHONE	C_ADDRESS	C_ZIP	A_NAME	A_PHONE	TP	AMT	REN
Alfred A. Ramas	615-844-2573	218 Fork Rd., Babo, TN	36123	Leah F. Hahn	615-882-1244	T1	100.00	05-Apr-2016
Leona K. Dunne	713-894-1238	Box 12A, Fox, KY	25246	Alex B. Alby	713-228-1249	T1	250.00	16-Jun-2016
Kathy W. Smith	615-894-2285	125 Oak Ln, Babo, TN	36123	Leah F. Hahn	615-882-2144	S2	150.00	29-Jan-2019
Paul F. Olowski	615-894-2160	217 Lee Ln., Babo, TN	36123	Leah F. Hahn	615-882-1244	S1	300.00	14-Oct-2018
Myron Orlando	615-222-1672	Box 111, New, TN	36155	Alex B. Alby	713-228-1249	T1	100.00	28-Dec-2018
Amy B. O'Brian	713-442-3381	387 Troll Dr., Fox, KY	25246	John T. Okon	615-123-5589	T2	850.00	22-Sep-2018
James G. Brown	615-297-1228	21 Tye Rd., Nash, TN	37118	Leah F. Hahn	615-882-1244	S1	120.00	25-Mar-2019
George Williams	615-290-2556	155 Maple, Nash, TN	37119	John T. Okon	615-123-5589	S1	250.00	17-Jul-2016
Anne G. Farriss	713-382-7185	2119 Elm, Crew, KY	25432	Alex B. Alby	713-228-1249	T2	100.00	03-Dec-2018
Olette K. Smith	615-297-3809	2782 Main, Nash, TN	37118	John T. Okon	615-123-5589	S2	500.00	14-Mar-2019

**C\_NAME** = Customer name  
**C\_PHONE** = Customer phone  
**C\_ADDRESS** = Customer address  
**C\_ZIP** = Customer zip code  
**A\_NAME** = Agent name  
**A\_PHONE** = Agent phone  
**TP** = Insurance type  
**AMT** = Insurance policy amount, in thousands of \$  
**REN** = Insurance renewal date

FIGURE 1.8 CONTENTS OF THE AGENT FILE

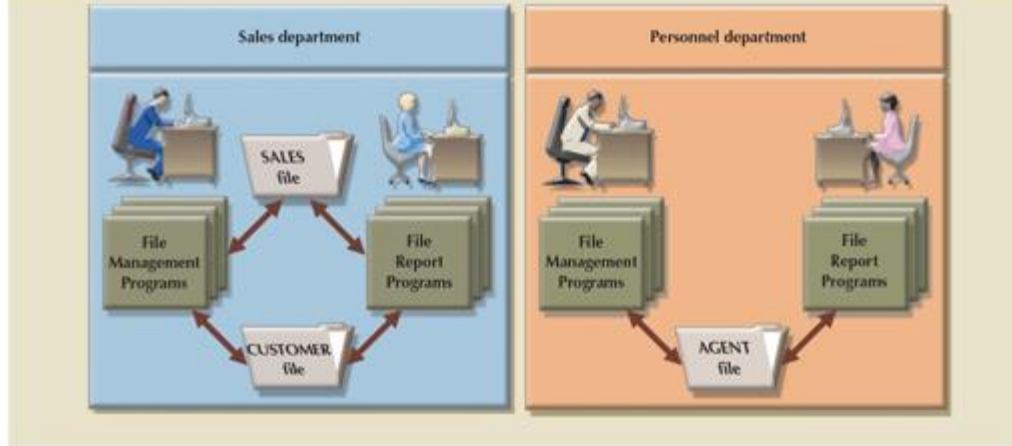
Database name: Ch01\_Text

A_NAME	A_PHONE	A_ADDRESS	ZIP	HIRED	YTD_PAY	YTD_FIT	YTD_FICA	YTD_SLS	DEP
Alex B. Alby	713-228-1249	123 Toll, Nash, TN	37119	01-Nov-2000	26566.24	6641.56	2125.30	132737.75	3
Leah F. Hahn	615-882-1244	334 Main, Fox, KY	25246	23-May-1986	32213.78	8053.44	2577.10	138967.35	0
John T. Okon	615-123-5589	452 Elm, New, TN	36155	15-Jun-2005	23198.29	5799.57	1855.86	127093.45	2

**A\_NAME** = Agent name  
**A\_PHONE** = Agent phone  
**A\_ADDRESS** = Agent address  
**ZIP** = Agent zip code  
**HIRED** = Agent date of hire  
**YTD\_PAY** = Year-to-date pay  
**YTD\_FIT** = Year-to-date federal income tax paid  
**YTD\_FICA** = Year-to-date Social Security taxes paid  
**YTD\_SLS** = Year-to-date sales  
**DEP** = Number of dependents

# Problems with file processing systems

FIGURE 1.9 A SIMPLE FILE SYSTEM



- Data duplication
- Program and data dependence
- Lack of security and limited data sharing (islands of information)
- Lengthy development times, difficulty of getting quick answers
  - Extensive programming needed

# What is a database?

**database**

/ˈdeɪtəbeɪs/ 

*noun*

plural noun: **databases**

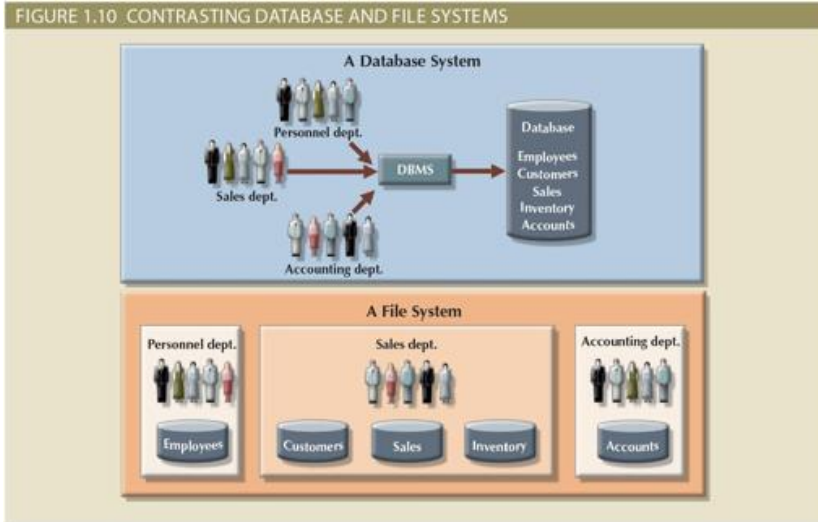
How do we  
structure our data?

We can run various  
queries/questions  
without the need to  
change the structure  
of the database.

a structured set of data held in a computer, especially one that is accessible in various ways.  
"a database covering nine million workers"

# A database

- Logically related data stored in a single logical data repository (the Database)
  - the repository may be stored on one local computer, distributed or in the cloud
  - stores data structures, relationships between structures, and access paths
  - defines, stores, and manages all access paths and components



# Types of database

- Hierarchical
- Network
- Relational \*
- Object Oriented/  
Object Relational
- XML/Hybrid
- No SQL

\* Unit focus

TABLE 2.1

## EVOLUTION OF MAJOR DATA MODELS

GENERATION	TIME	DATA MODEL	EXAMPLES	COMMENTS
First	1960s–1970s	File system	VMS/VSAM	Used mainly on IBM mainframe systems Managed records, not relationships
Second	1970s	Hierarchical and network	IMS, ADABAS, IDS-II	Early database systems Navigational access
Third	Mid-1970s	Relational	DB2 Oracle MS SQL Server MySQL	Conceptual simplicity Entity relationship (ER) modeling and support for relational data modeling
Fourth	Mid-1980s	Object-oriented Object/relational (O/R)	Versant Objectivity/DB DB2 UDB Oracle 12c	Object/relational supports object data types Star Schema support for data warehousing Web databases become common
Fifth	Mid-1990s	XML Hybrid DBMS	dbXML Tamino DB2 UDB Oracle 12c MS SQL Server	Unstructured data support O/R model supports XML documents Hybrid DBMS adds object front end to relational databases Support large databases (terabyte size)
Emerging Models: NoSQL	Early 2000s to present	Key-value store Column store	SimpleDB (Amazon) BigTable (Google) Cassandra (Apache) MongoDB Riak	Distributed, highly scalable High performance, fault tolerant Very large storage (petabytes) Suited for sparse data Proprietary application programming interface (API)

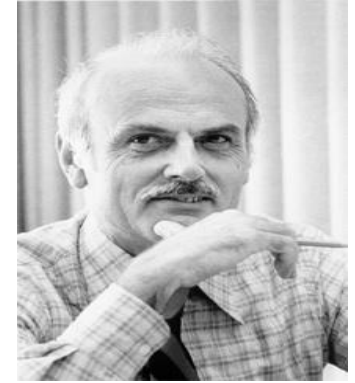


# Who's Who (the founders) in the Relational Database World



# 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 3
- Normalisation in week 4
- E. F. Codd, “A Relational Model of Data for Large Shared Data Banks”, *Comm. Of ACM*, 1970



E.F. Codd (1923-2003)

# 1974: SQL

- Developed at IBM
- Initially called SEQUEL (**S**tructured **E**nglish **Q**Uery **L**anguage)
- Doesn't strictly follow Codd's theory
- Oracle: the first commercially available implementation of SQL in 1979
- SQL in weeks 7, 8, 9 & 10
- D Chamberlin, R Boyce, “**SEQUEL: A structured English query language**”, *ACM SIGFIDET*, 1974



Donald Chamberlin (1944- )



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 2 & 5
- Peter Chen, “The entity-relationship model—toward a unified view of data”, *ACM TODS*, 1976



Peter Chen (1947 - )

# 1979: Oracle

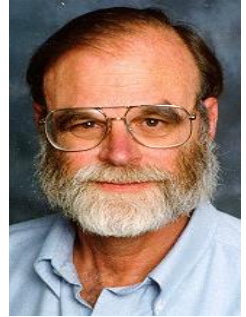
- Inspired by Codd's ideas
- First commercial release in 1979
- Most popular RDBMS
- Introduced PL/SQL in 1988  
(Procedural Language/SQL)
- Oracle SQL in week 7, 8, 9 & 10



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 (1944 - )

- Jim Gray, “The Transaction Concept: Virtues and Limitations”, *VLDB*, 1981

# 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.)
  - Web data
  - Big Data



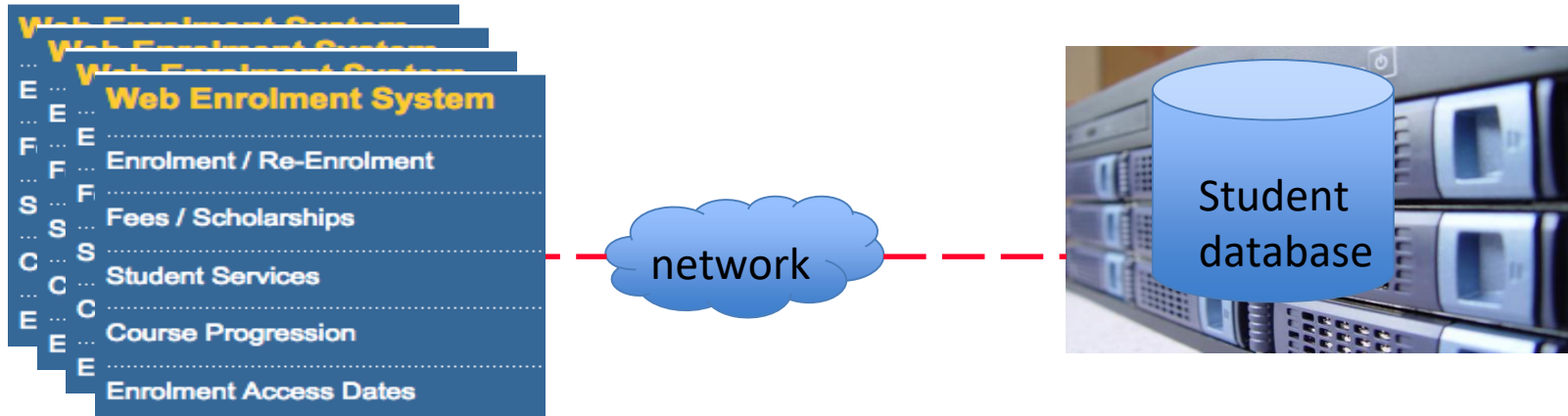
<https://www.domo.com/learn/infographic/data-never-sleeps-8#/>

# Our Database Environment





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

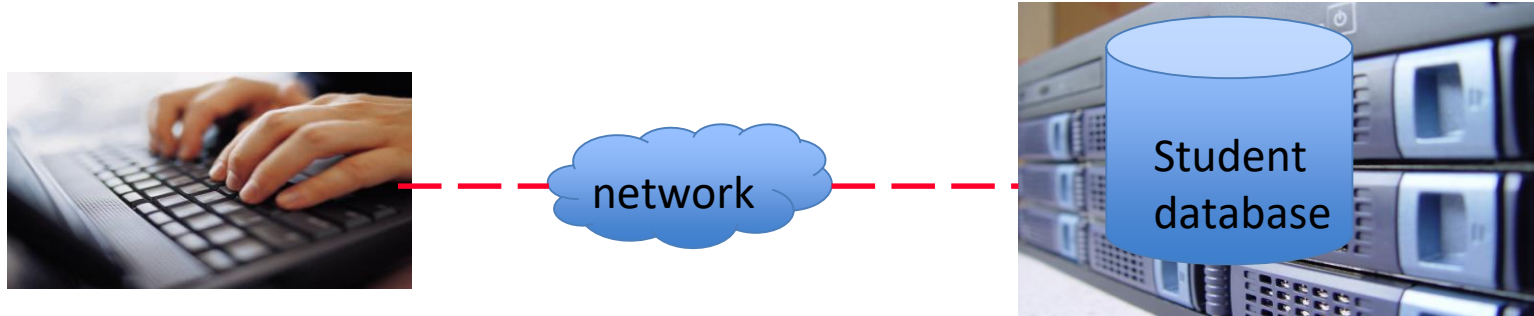


Front end application  
(client)

Student Database is  
implemented in an  
Oracle DBMS (server)

# Database Systems in Action

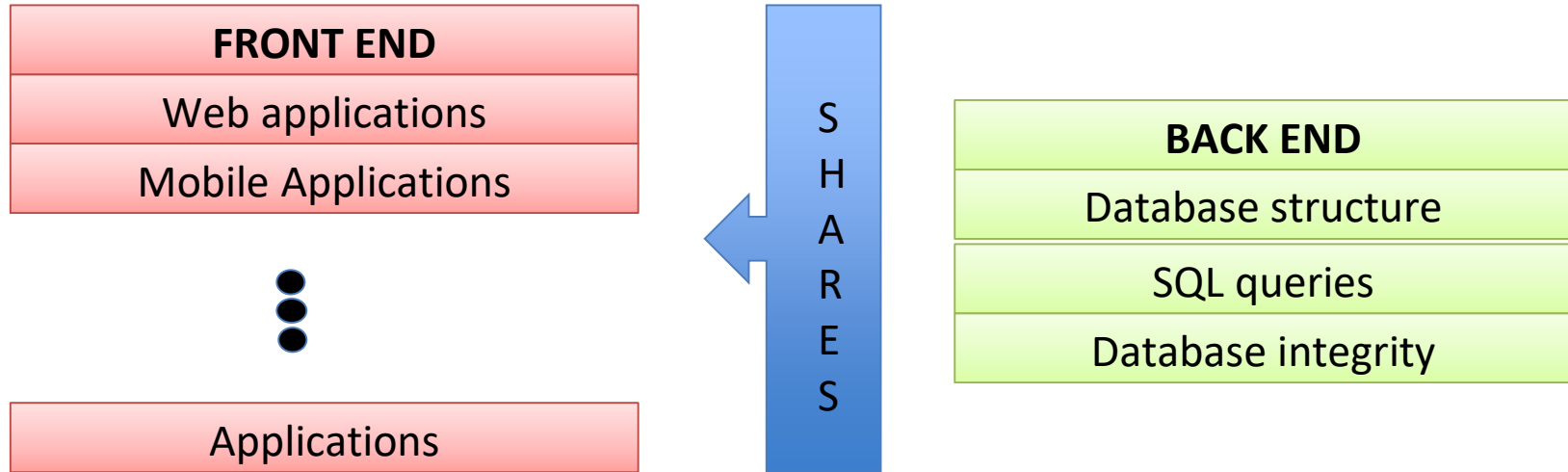
## Developers' View



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

Student Database  
(server)

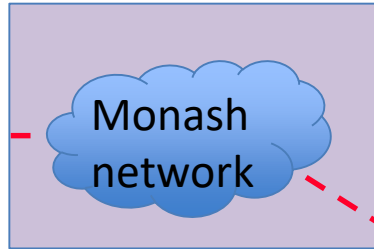
# Developing Application with Database



In this unit, we will concentrate on building the back end.

# Our Database Systems Environment

Local install of SQL Developer  
and Monash Virtual Private  
Network (VPN)



OR

Monash MoVE SQL Developer



[ora-fit.ocio.monash.edu](http://ora-fit.ocio.monash.edu)

