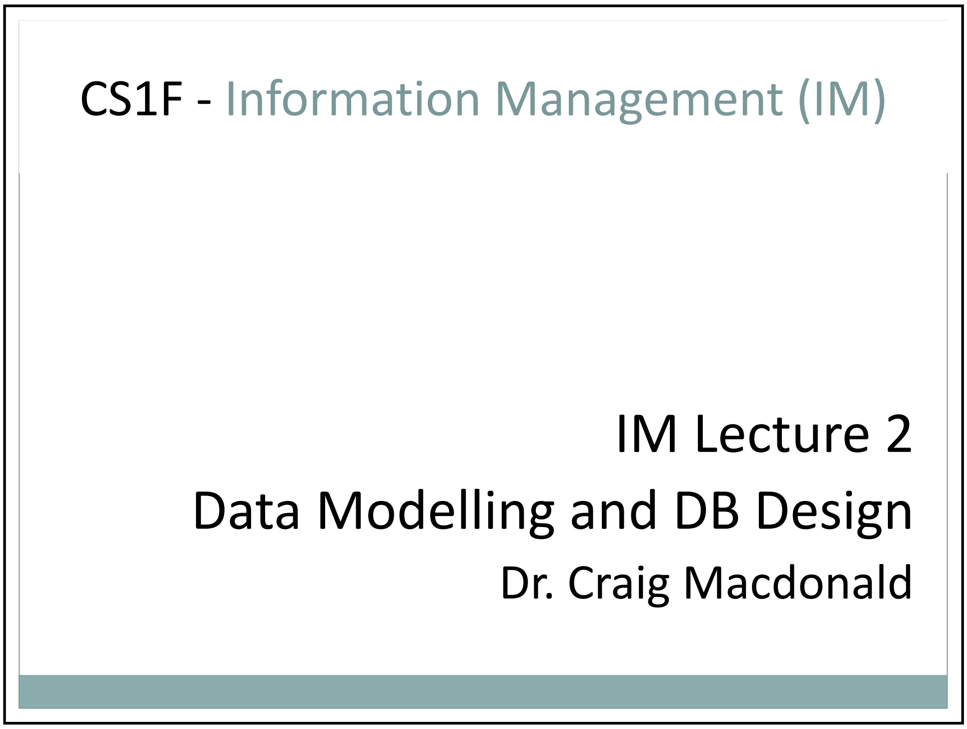
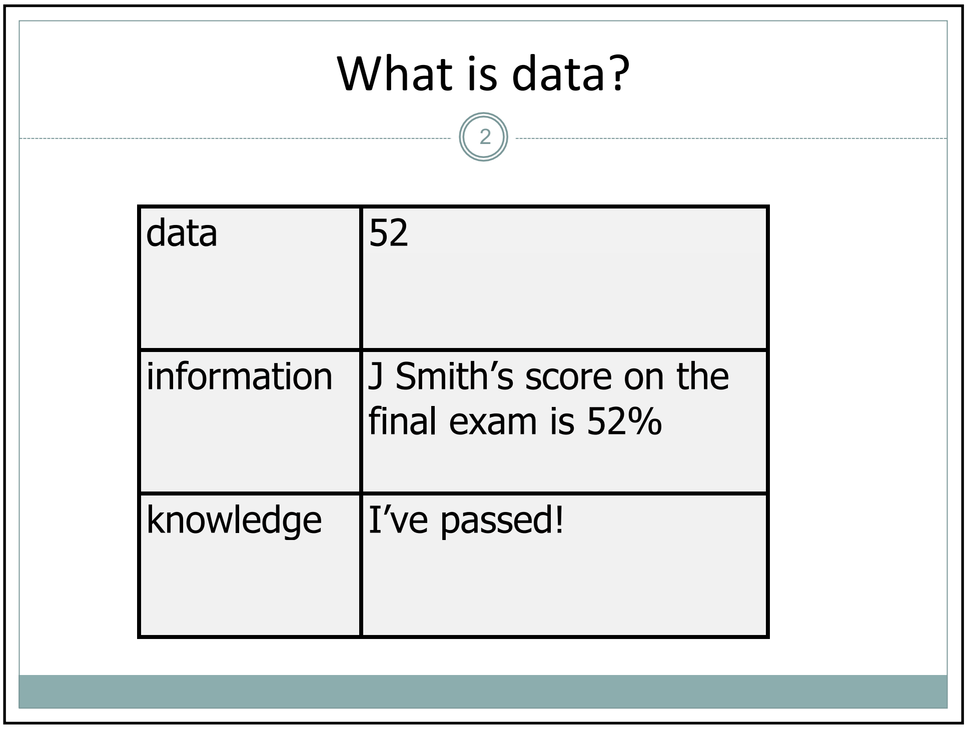


CS1F - Information Management (IM)

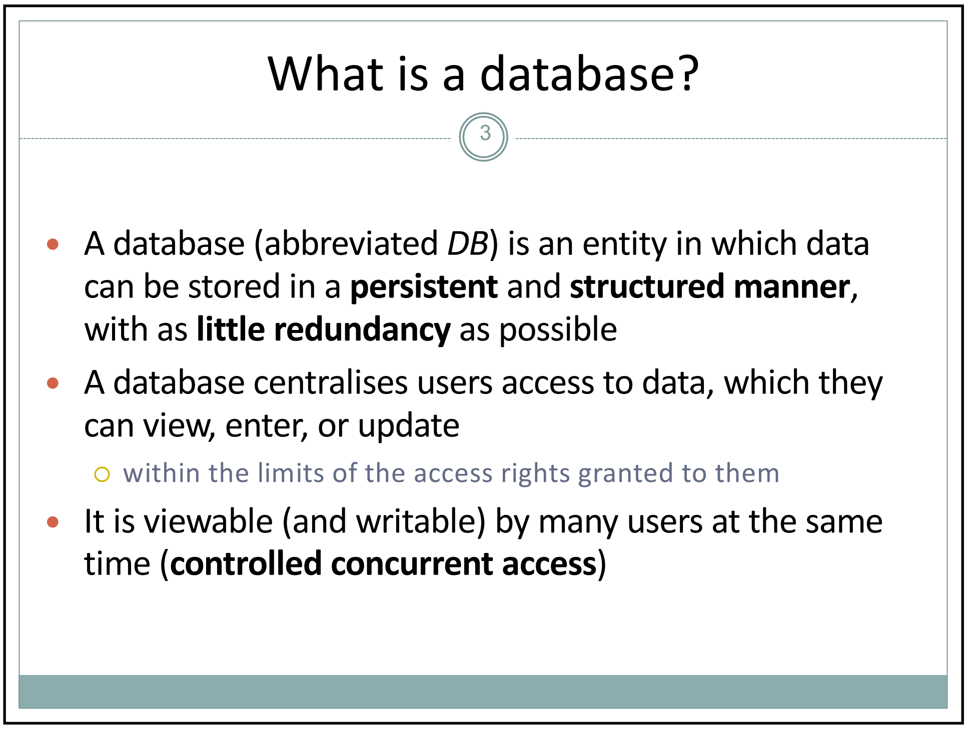
IM Lecture 2
Data Modelling and DB Design

Dr. Craig Macdonald



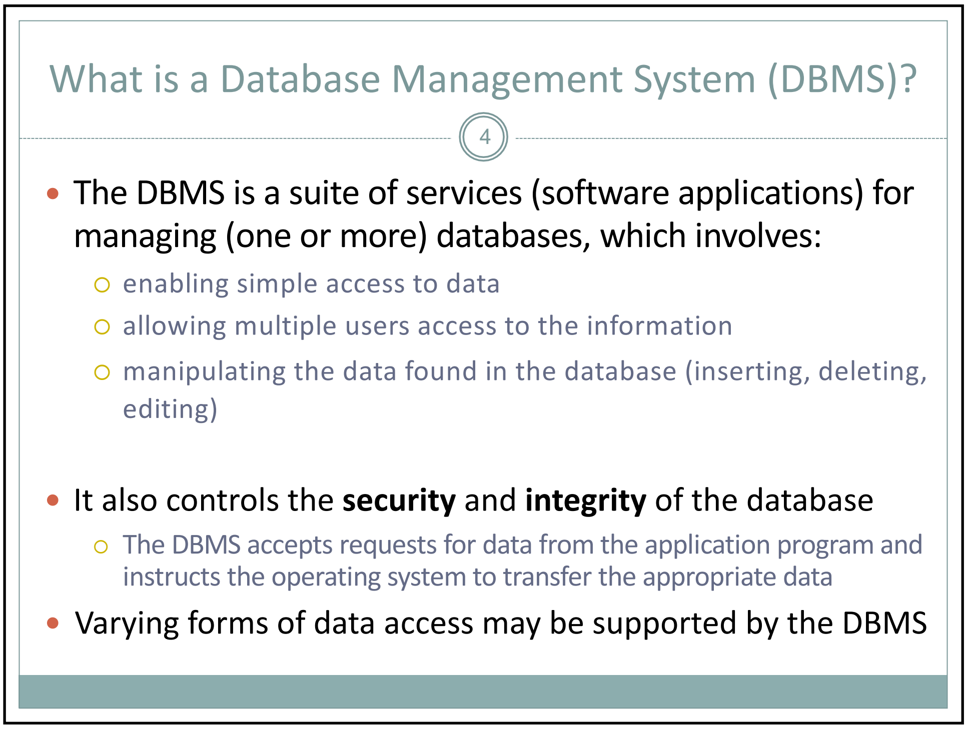
What is data?

data	52
information	J Smith’s score on the final exam is 52%
knowledge	I’ve passed!



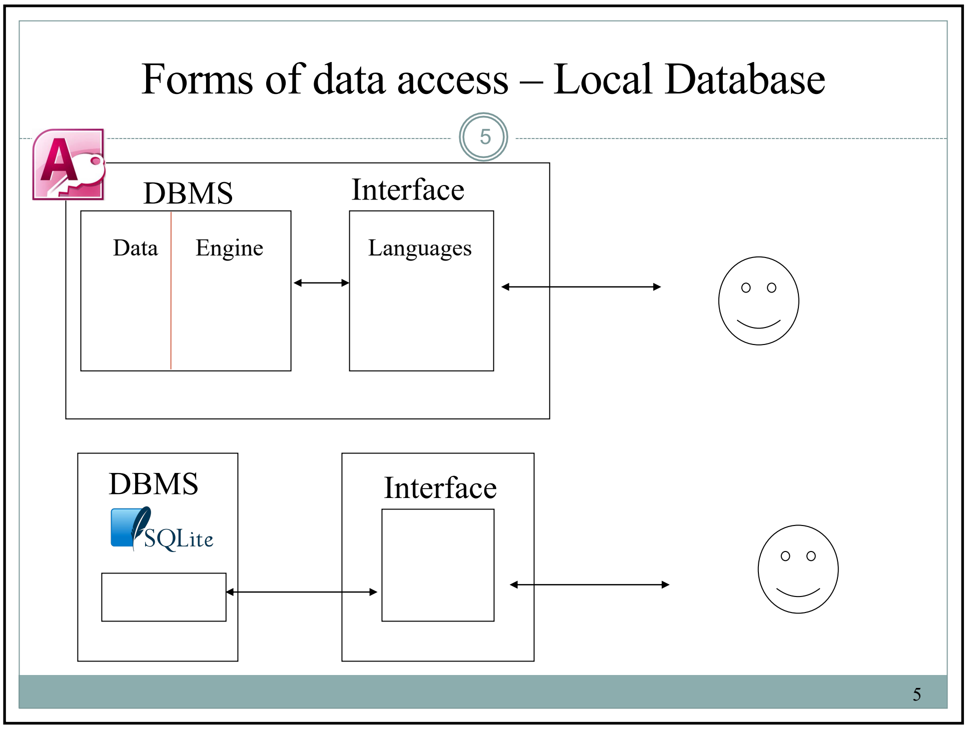
What is a database?

- A database (abbreviated *DB*) is an entity in which data can be stored in a **persistent** and **structured manner**, with as **little redundancy** as possible
- A database centralises users access to data, which they can view, enter, or update
  - within the limits of the access rights granted to them
- It is viewable (and writable) by many users at the same time (**controlled concurrent access**)

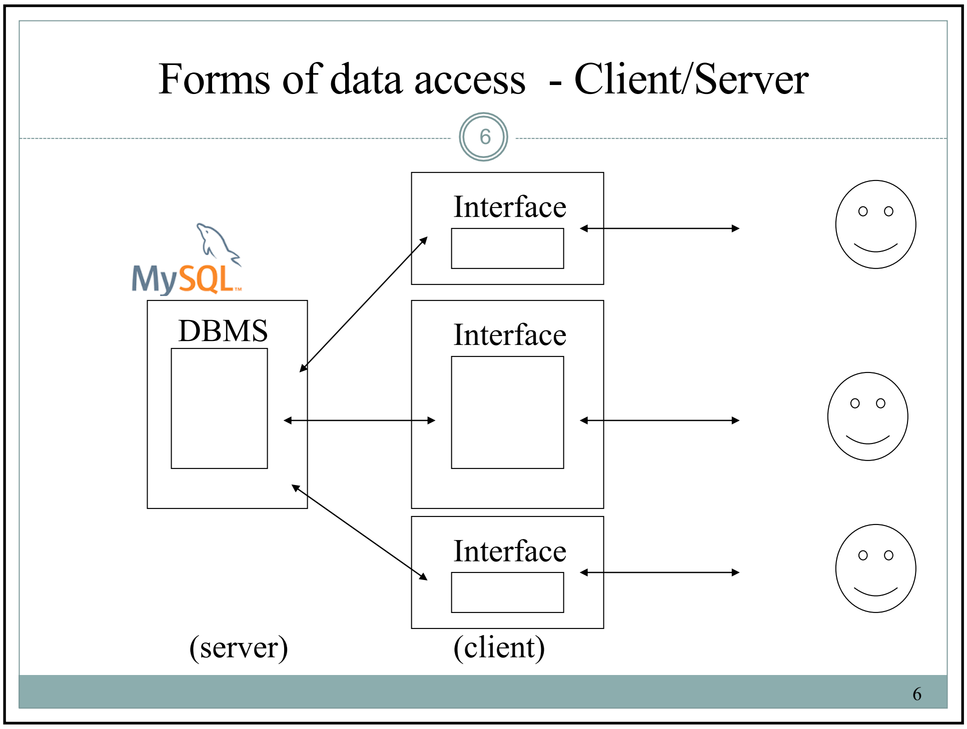


What is a Database Management System (DBMS)?

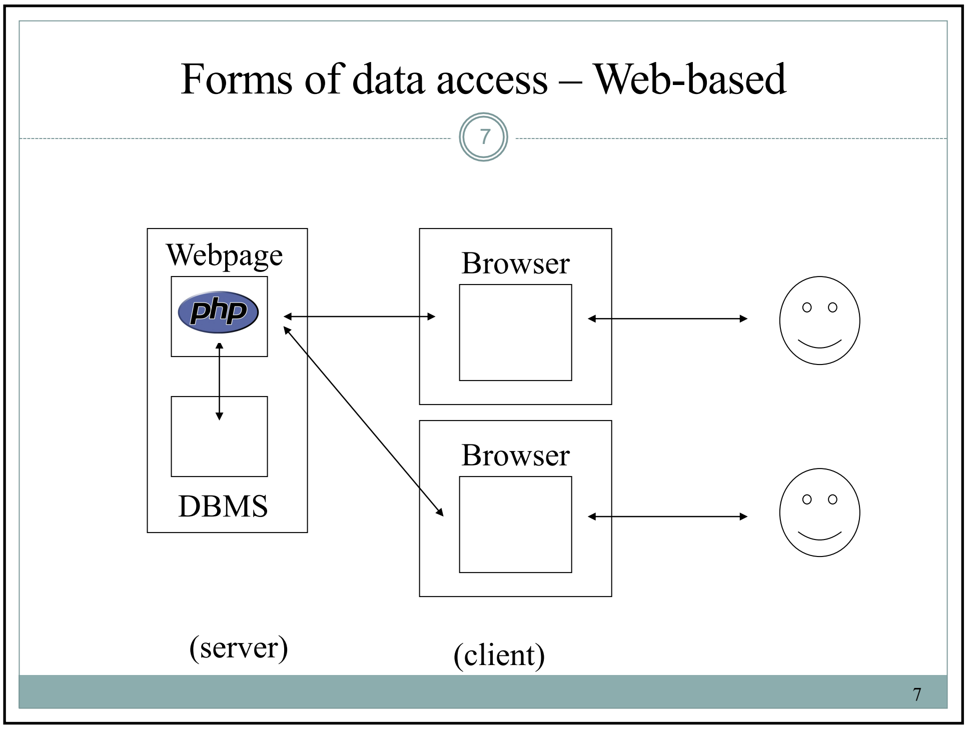
- The DBMS is a suite of services (software applications) for managing (one or more) databases, which involves:
  - enabling simple access to data
  - allowing multiple users access to the information
  - manipulating the data found in the database (inserting, deleting, editing)
- It also controls the **security** and **integrity** of the database
  - The DBMS accepts requests for data from the application program and instructs the operating system to transfer the appropriate data
- Varying forms of data access may be supported by the DBMS



Forms of data access – Local Database



Forms of data access - Client/Server



Forms of data access – Web-based

7

Webpage

php

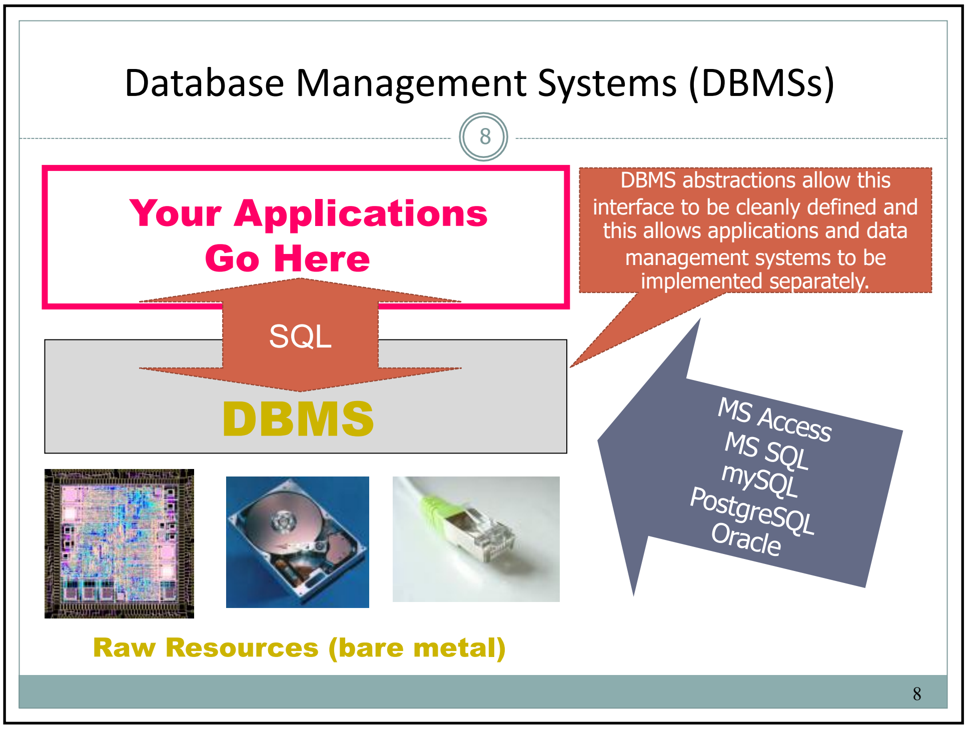
DBMS

Browser

Browser

(server)

(client)



Database Management Systems (DBMSs)

8

Your Applications Go Here

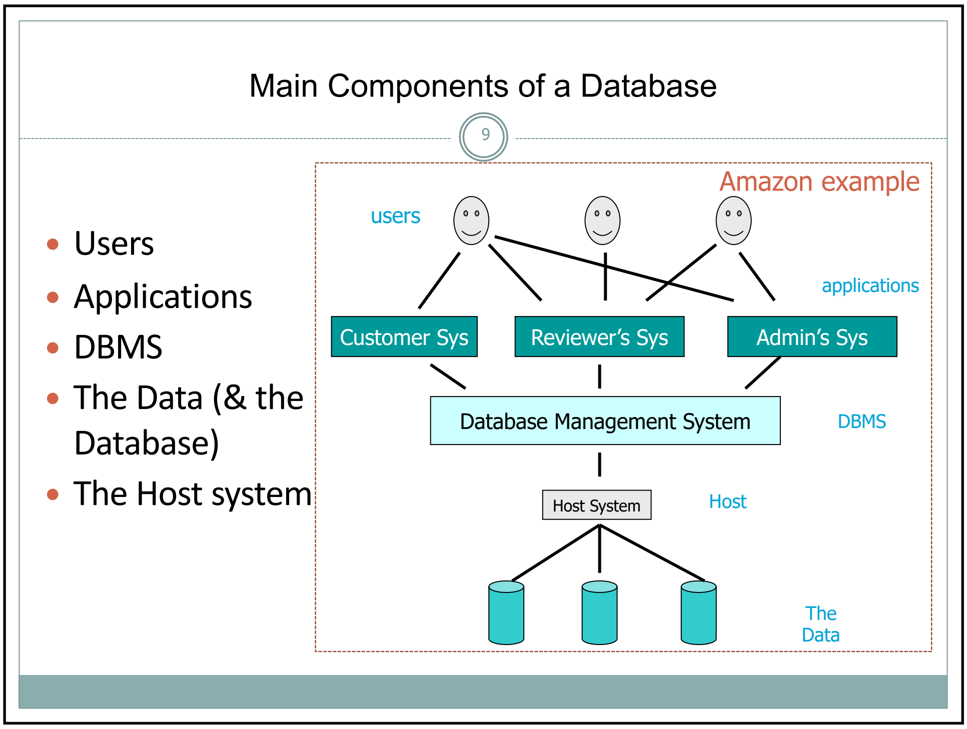
SQL

DBMS

DBMS abstractions allow this interface to be cleanly defined and this allows applications and data management systems to be implemented separately.

MS Access
MS SQL
mySQL
PostgreSQL
Oracle

Raw Resources (bare metal)



Main Components of a Database

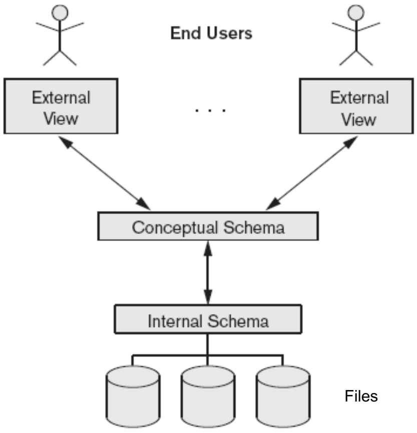
- Users
- Applications
- DBMS
- The Data (& the Database)
- The Host system

Who will use this database? What will they see? What data will we store?

DESIGNING DATABASE SYSTEMS

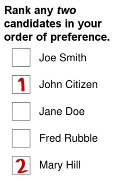
Database Systems: Three-level Architecture

- We can think about a database system at three levels, called “schemas”:
  - External: how users view data
  - Conceptual: how programmers model and implement the database in the database
  - Internal: how the DBMS stores the data
- In designing a database, we take an **external schema,** and *design* a corresponding **conceptual schema.** The DBMS handles the **internal schema,** with hints from the designer.



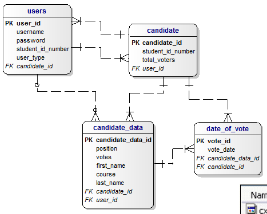
Three-level Architecture: Election Example

External Schema

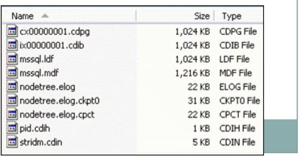

Rank any *two* candidates in your order of preference.

□ Joe Smith
1 John Citizen
□ Jane Doe
□ Fred Rubble
2 Mary Hill

Conceptual Schema



Internal Schema



Name	Size	Type
cx00000001.cdpg	1,024 KB	CDPG File
ix00000001.cdib	1,024 KB	CDIB File
mssql.ldf	1,024 KB	LDF File
mssql.mdf	1,216 KB	MDF File
nodetree.elog	22 KB	ELOG File
nodetree.elog.ckpt0	31 KB	CKPT0 File
nodetree.elog.cpct	22 KB	CPCT File
pid.cdih	1 KB	CDIH File
stridm.cdin	5 KB	CDIN File

Example: Amazon

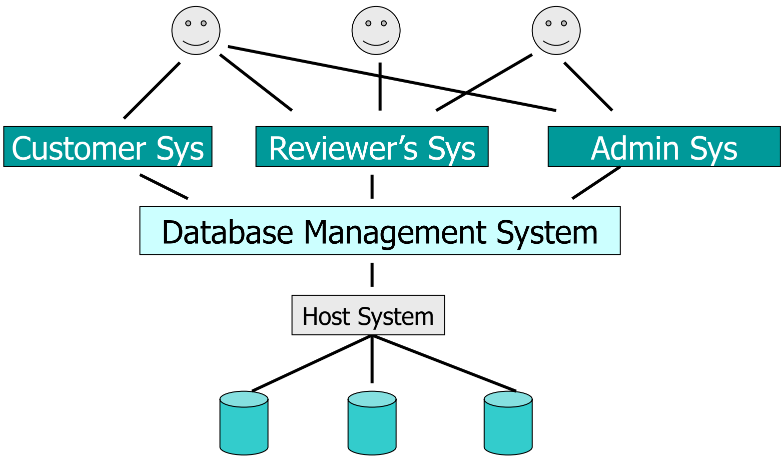
13

- Stores data about products and their related details (name, price, colour, product code…….)

- A customer can view products, search products, buy a product and rate reviews
- An admin person can upload products & edit product info
- A reviewer can write reviews
- ……….

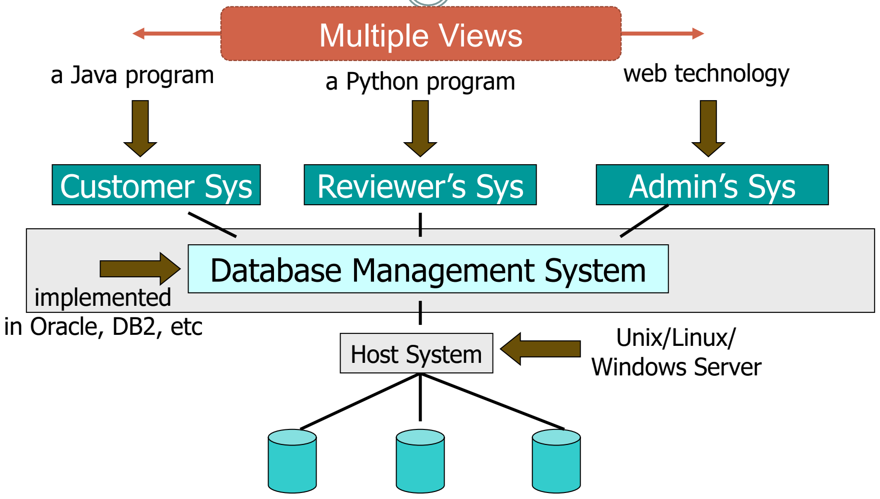
An example Database system - AMAZON

14



An example Database system - AMAZON

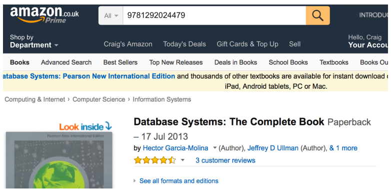
15



Three-level Architecture: Amazon Example

16

External Schema



9781292024479

atabase Systems: Pearson New International Edition and thousands of other textbooks are available for instant download iPad, Android tablets, PC or Mac.

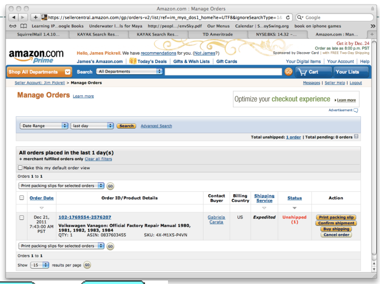
Computing & Internet › Computer Science › Information Systems

Database Systems: The Complete Book Paperback – 17 Jul 2013

by Hector Garcia-Molina (Author), Jeffrey D Ullman (Author), & 1 more

3 customer reviews

See all formats and editions



https://sellercentral.amazon.com/gp/orders-v2/list/ref=im_myo_dos1_home?ie=UTF8&ignoreSearchType=1&

Hello, James Pickrell. We have recommendations for you. (Not James?)

Get it by Dec. 24
Order as late as 8:00 p.m. PST
Sponsored by Discover Card | with FREE Two-Day Shipping

Seller Account: Jim Pickrell > Manage Orders

Manage Orders

Optimize your checkout experience

Date Range: last day

Total unshipped: 1 order | Total pending: 0 orders

All orders placed in the last 1 day(s)
+ merchant fulfilled orders only Clear all filters

□ Make this my default order view

Orders 1 to 1

Print packing slips for selected orders

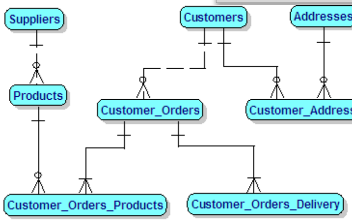
□	Order Date	Order ID/Product Details	Contact Buyer	Billing Country	Shipping Service	Status	Action
□	Dec 21, 2011 7:43:00 AM PST	102-1769554-2576207 Volkswagen Vanagon: Official Factory Repair Manual 1980, 1981, 1982, 1983, 1984 QTY: 1 ASIN: 0837603455 SKU: 4X-M1X5-P4VN	Gabriela Carata	US	Expedited	Unshipped (1)	Print packing slip Confirm shipment Buy shipping Cancel order

Print packing slips for selected orders

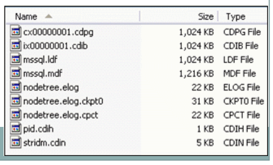
Orders 1 to 1

Show 15 results per page

Conceptual Schema



Internal Schema



Name	Size	Type
cx00000001.cdpg	1,024 KB	CDPG File
ix00000001.cdib	1,024 KB	CDIB File
mssql.ldf	1,024 KB	LDF File
mssql.mdf	1,216 KB	MDF File
nodetree.elog	22 KB	ELOG File
nodetree.elog.ckpt0	31 KB	CKPT0 File
nodetree.elog.cpct	22 KB	CPCT File
pid.cdih	1 KB	CDIH File
stridm.cdin	5 KB	CDIN File

Summary

- Databases are used by people…… to perform particular tasks, obtain views on the data
- Databases therefore need interfaces to allow people access to the data
- Many people may need to access the same database
  - Web pages / mApps are just one way of implementing an interface to a database
- We must consider the needs of the users when designing a database … our next focus.
- Database design: taking external schemas to identify the conceptual scheme. The DBMS will handle the internal schema

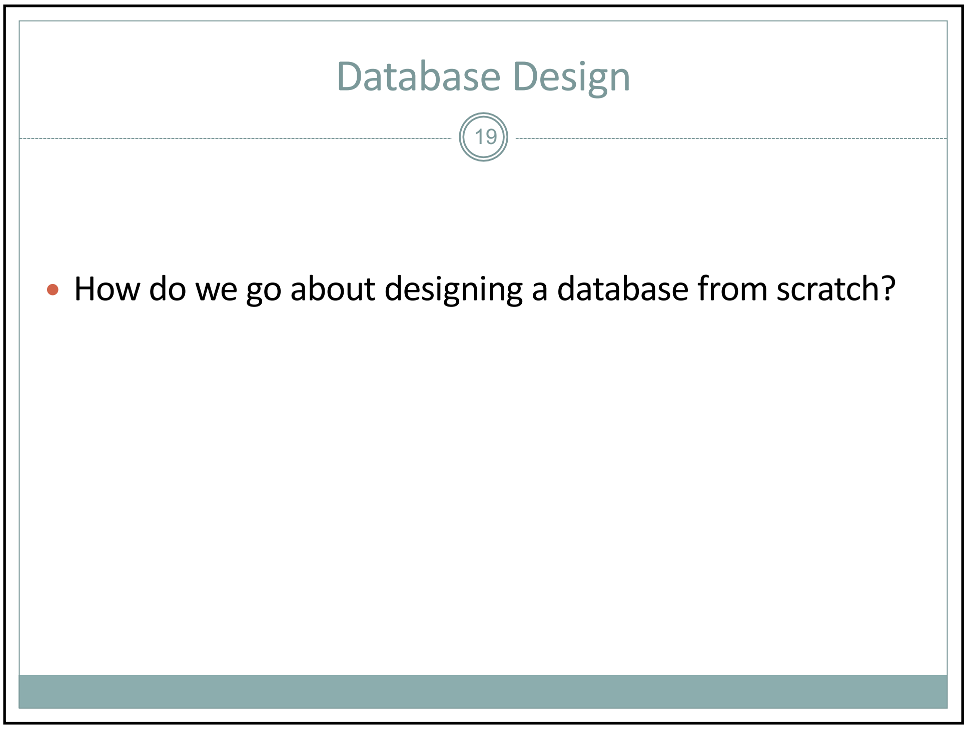
Database design lifecycle

- Requirements analysis
  - User needs; what must database do?
- Conceptual design
  - High-level description; often using E/R model
- Logical design
  - Translate E/R model into (typically) relational schema
- Schema refinement
  - Check schema for redundancies and anomalies
- Physical design/tuning
  - Consider typical workloads, and further optimise

Today

Lecture 3

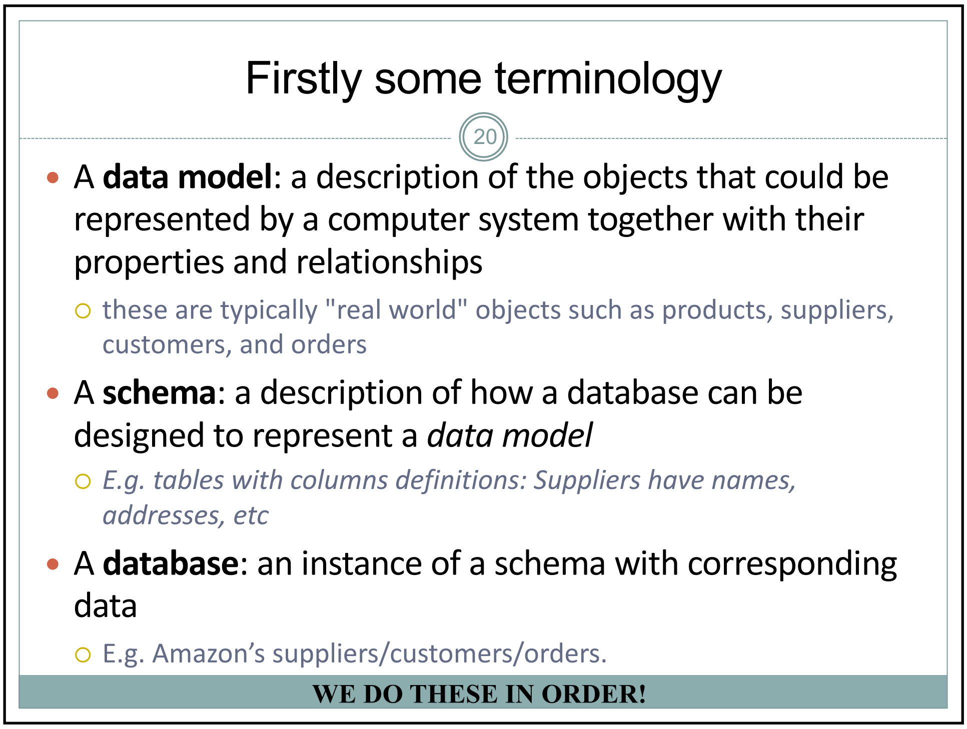
Later years



Database Design

19

- How do we go about designing a database from scratch?

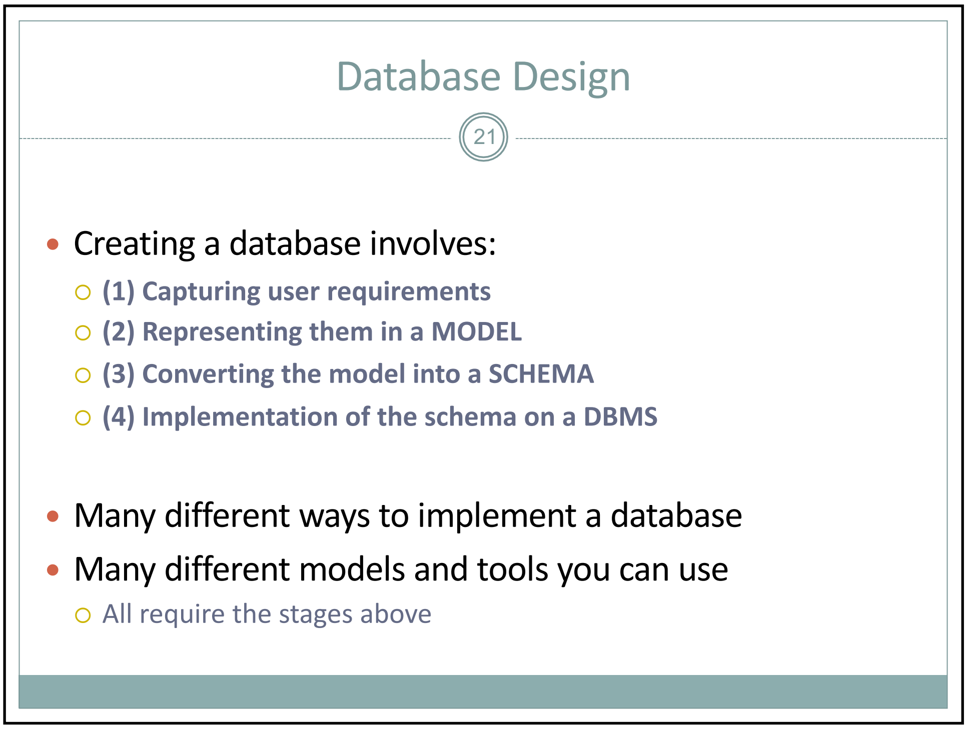


Firstly some terminology

20

- A **data model**: a description of the objects that could be represented by a computer system together with their properties and relationships
  - these are typically "real world" objects such as products, suppliers, customers, and orders
- A **schema**: a description of how a database can be designed to represent a *data model*
  - *E.g. tables with columns definitions: Suppliers have names, addresses, etc*
- A **database**: an instance of a schema with corresponding data
  - E.g. Amazon's suppliers/customers/orders.

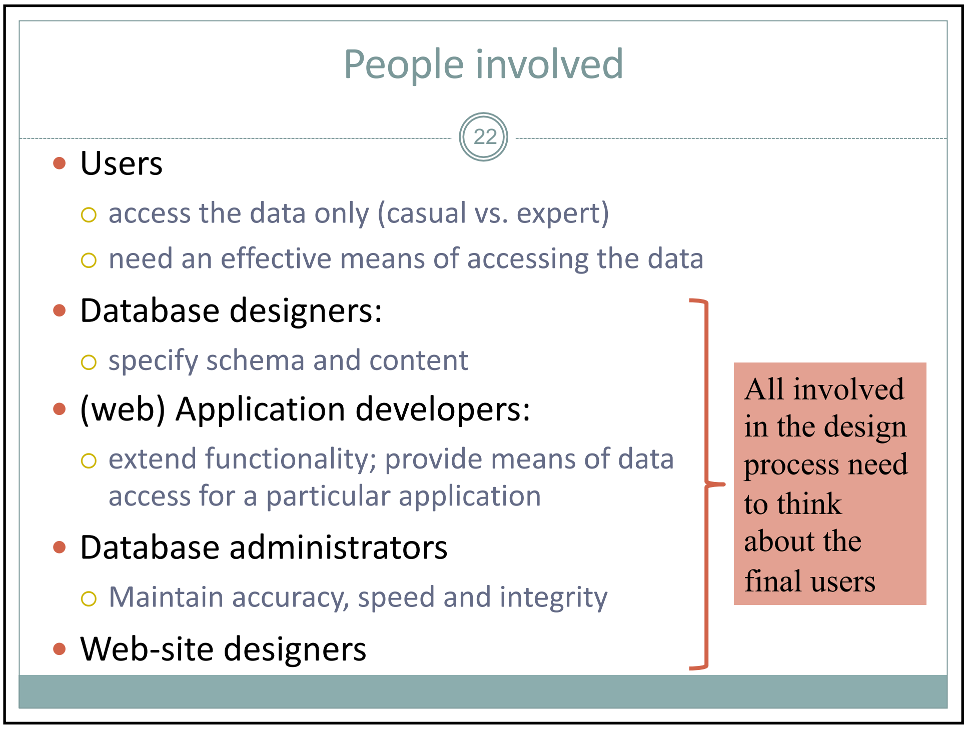
WE DO THESE IN ORDER!



Database Design

21

- Creating a database involves:
  - **(1) Capturing user requirements**
  - **(2) Representing them in a MODEL**
  - **(3) Converting the model into a SCHEMA**
  - **(4) Implementation of the schema on a DBMS**
- Many different ways to implement a database
- Many different models and tools you can use
  - All require the stages above

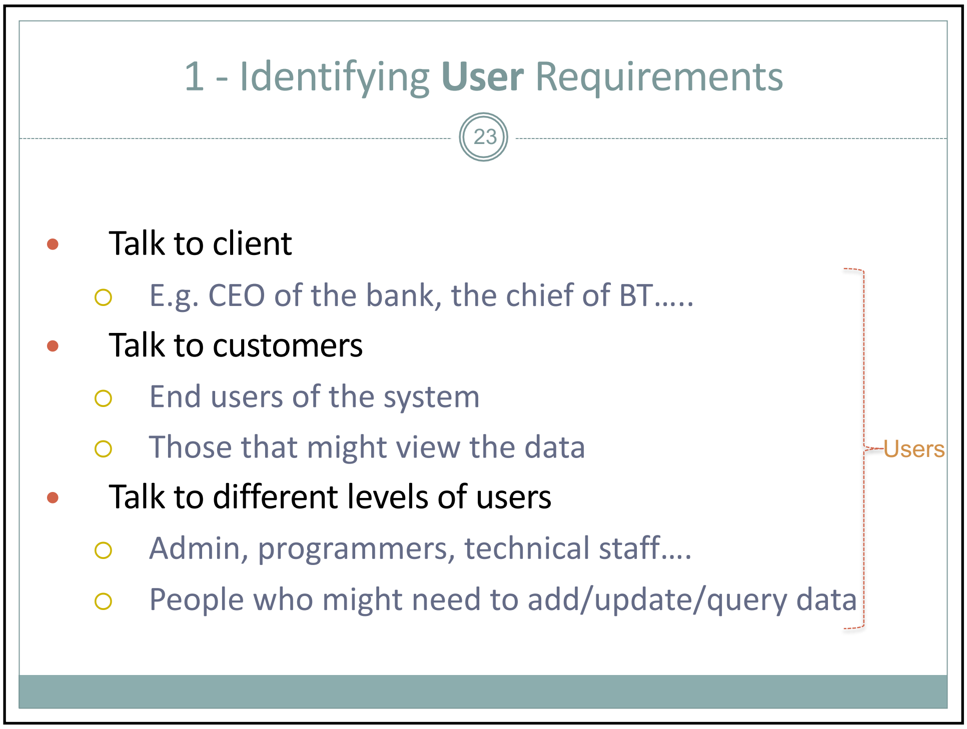


People involved

22

- Users
  - access the data only (casual vs. expert)
  - need an effective means of accessing the data
- Database designers:
  - specify schema and content
- (web) Application developers:
  - extend functionality; provide means of data access for a particular application
- Database administrators
  - Maintain accuracy, speed and integrity
- Web-site designers

All involved in the design process need to think about the final users

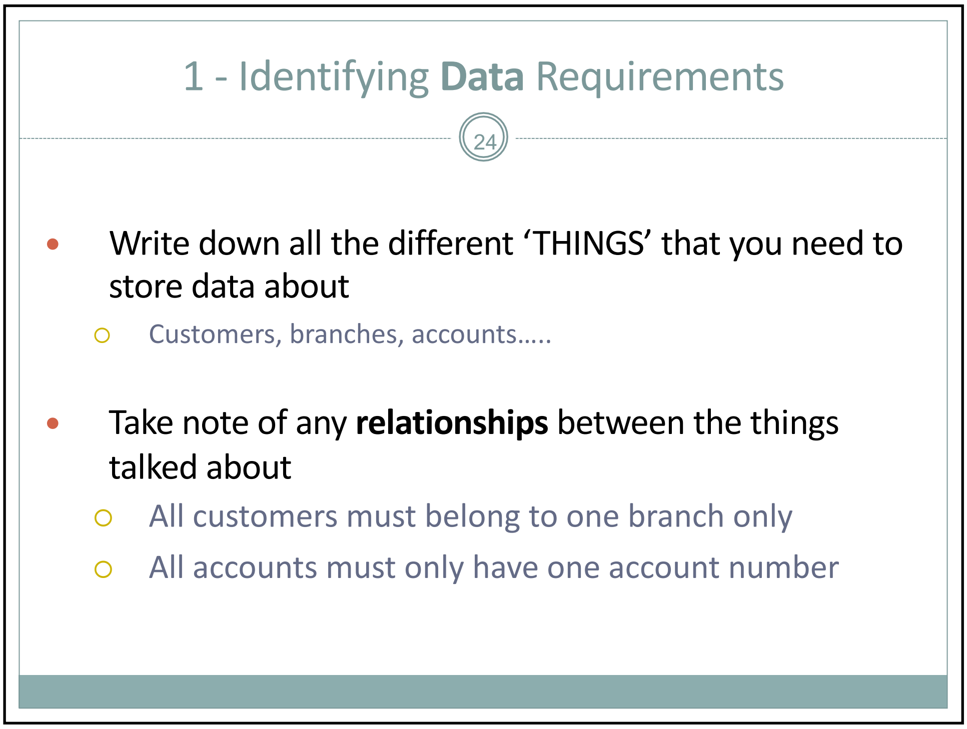


1 - Identifying **User** Requirements

23

- Talk to client
  - E.g. CEO of the bank, the chief of BT…..
- Talk to customers
  - End users of the system
  - Those that might view the data
- Talk to different levels of users
  - Admin, programmers, technical staff….
  - People who might need to add/update/query data

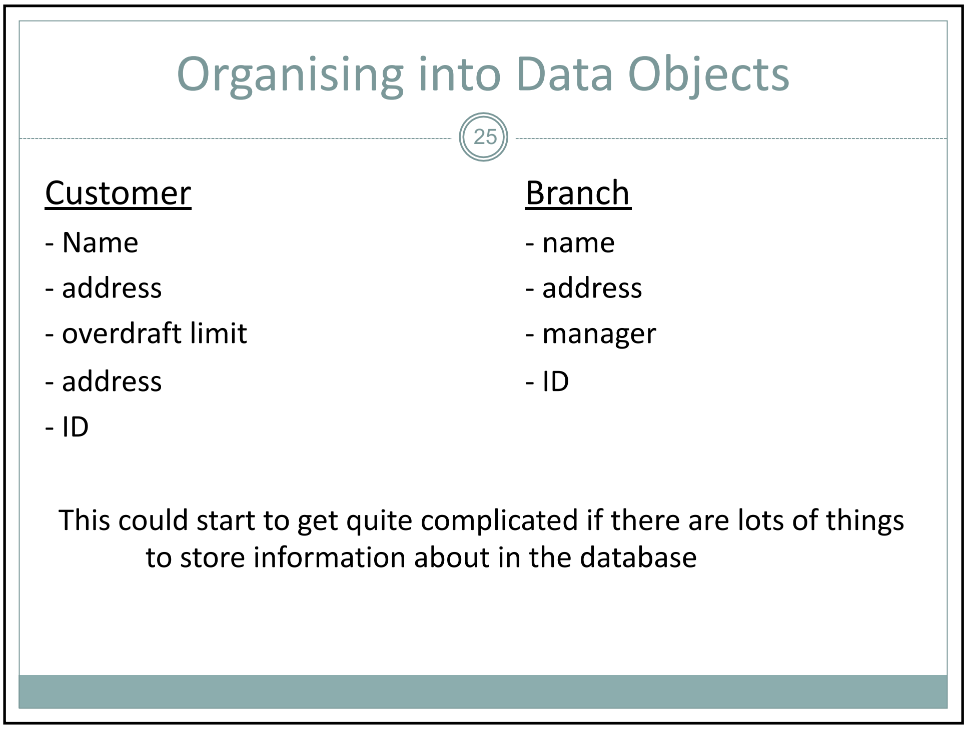
Users



1 - Identifying **Data** Requirements

24

- Write down all the different ‘THINGS’ that you need to store data about
  - Customers, branches, accounts…..
- Take note of any **relationships** between the things talked about
  - All customers must belong to one branch only
  - All accounts must only have one account number



Organising into Data Objects

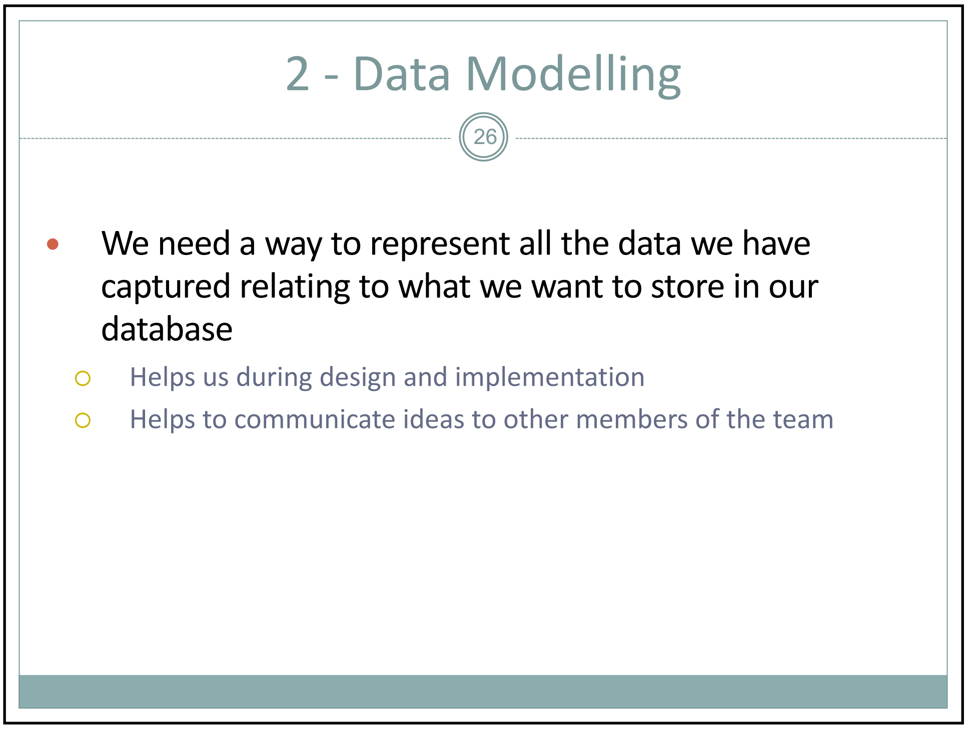
Customer

- Name
- address
- overdraft limit
- address
- ID

Branch

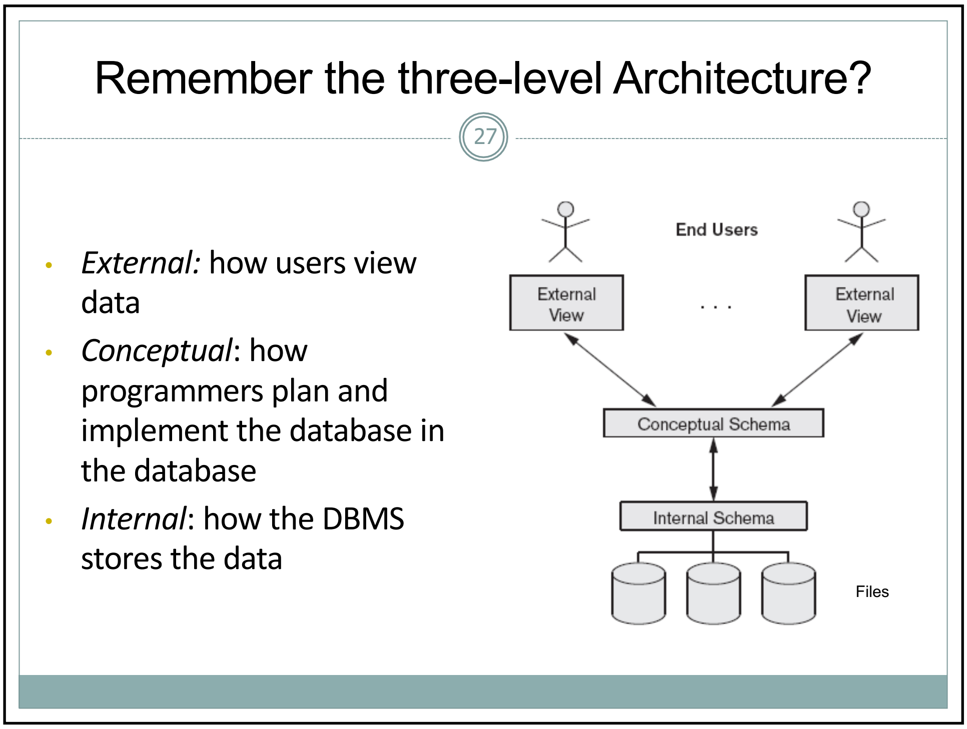
- name
- address
- manager
- ID

This could start to get quite complicated if there are lots of things to store information about in the database



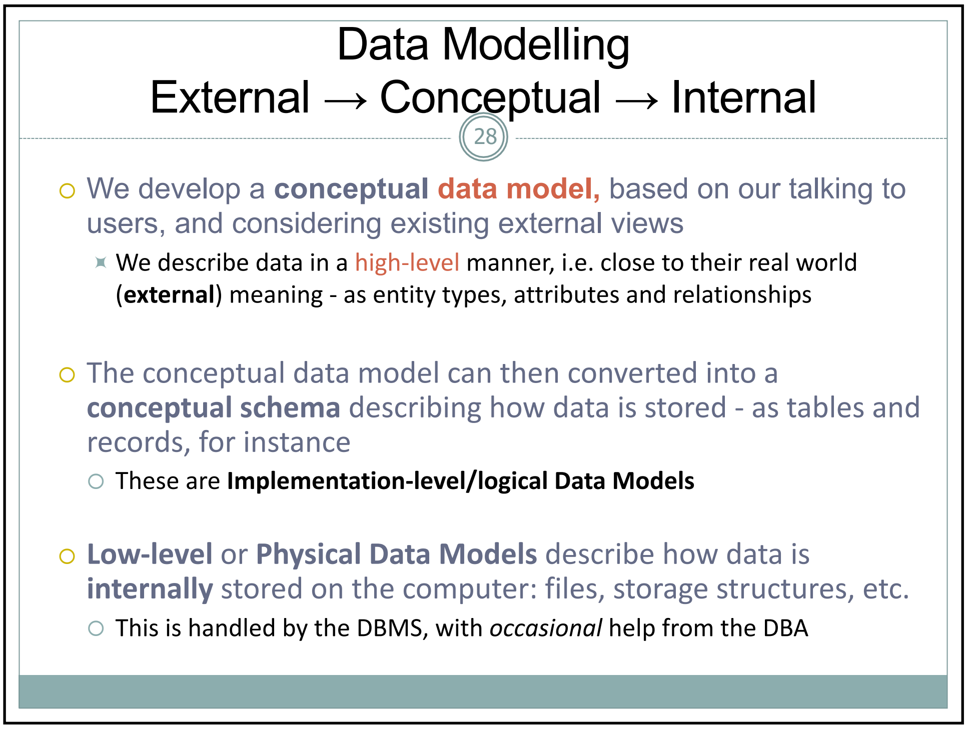
2 - Data Modelling

- We need a way to represent all the data we have captured relating to what we want to store in our database
  - Helps us during design and implementation
  - Helps to communicate ideas to other members of the team



Remember the three-level Architecture?

- *External:* how users view data
- *Conceptual*: how programmers plan and implement the database in the database
- *Internal*: how the DBMS stores the data

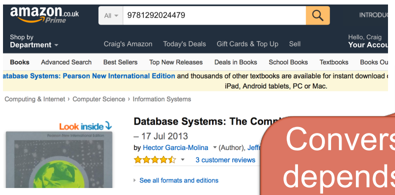


Data Modelling
External → Conceptual → Internal

- We develop a **conceptual data model,** based on our talking to users, and considering existing external views
  - We describe data in a high-level manner, i.e. close to their real world (**external**) meaning - as entity types, attributes and relationships
- The conceptual data model can then converted into a **conceptual schema** describing how data is stored - as tables and records, for instance
  - These are **Implementation-level/logical Data Models**
- **Low-level** or **Physical Data Models** describe how data is **internally** stored on the computer: files, storage structures, etc.
  - This is handled by the DBMS, with *occasional* help from the DBA

Back to the Amazon Example

1. External Schema (aka User Needs)



amazon.co.uk Prime | All | 9781292024479 | INTRODU

Shop by Department | Craig's Amazon | Today's Deals | Gift Cards & Top Up | Sell | Hello, Craig Your Accou

Books | Advanced Search | Best Sellers | Top New Releases | Deals in Books | School Books | Textbooks | Books Ou

atabase Systems: Pearson New International Edition and thousands of other textbooks are available for instant download
iPad, Android tablets, PC or Mac.

Computing & Internet › Computer Science › Information Systems

Look inside

Database Systems: The Comp

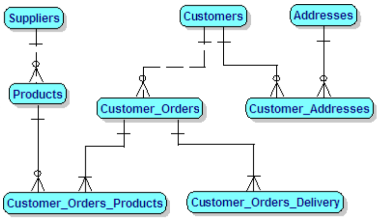
– 17 Jul 2013

by Hector Garcia-Molina (Author), Jeff

3 customer reviews

See all formats and editions

2. Conceptual *Model*

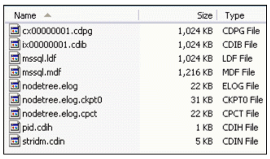


Conversion depends on Database Model used by DBMS

3. Implementation Model: Conceptual *Schema*

```
Supplier(ID, Name, Address, Postcode)
Product(ID, Supplier, Title)
Customer_Orders_Products(
…
```

4. Internal Schema



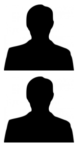
Name	Size	Type
cx00000001.cdpg	1,024 KB	CDPG File
ix00000001.cdib	1,024 KB	CDIB File
mssql.ldf	1,024 KB	LDF File
mssql.mdf	1,216 KB	MDF File
nodetree.elog	22 KB	ELOG File
nodetree.elog.ckpt0	31 KB	CKPT0 File
nodetree.elog.cpct	22 KB	CPCT File
pid.cdih	1 KB	CDIH File
stridm.cdin	5 KB	CDIN File

What are the *concepts* involved?

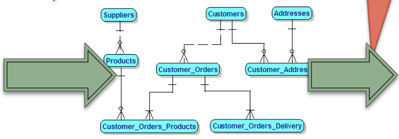
How can we represent these in a database?

Chosen by DBMS

External



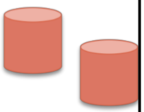
Conceptual Data Model



Implementation/ Logical Model

```
Suppliers(ID, Name, Address)
Products(Id, Supplier, Title)
Customer_Orders_Products()
…
```

Physical Model



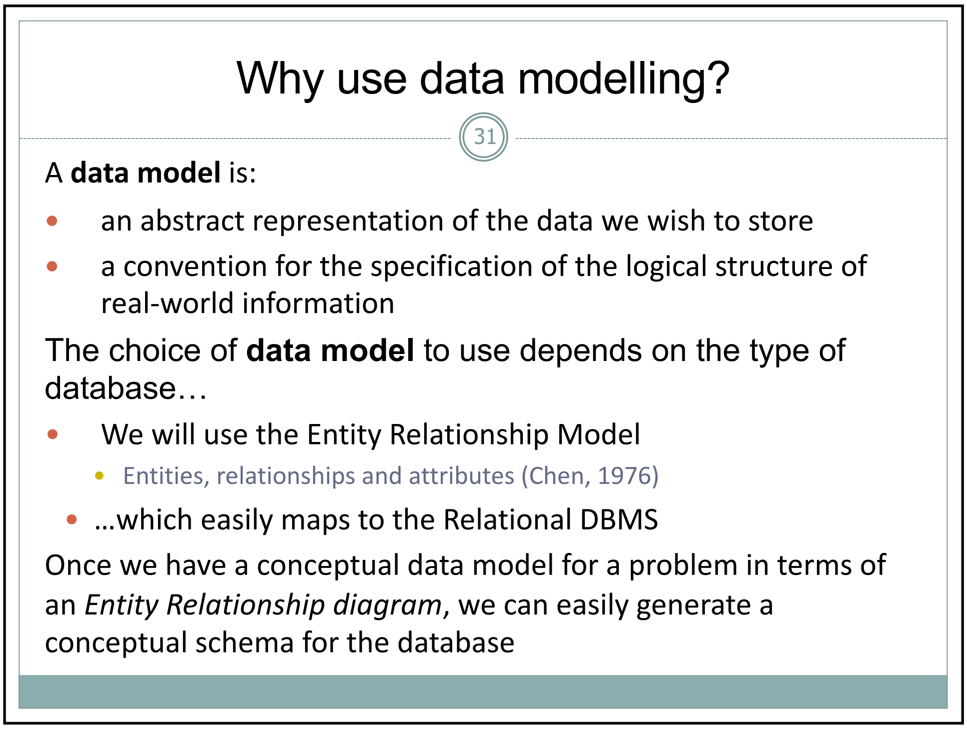
Conceptual **Schema**

Internal **Schema**

Real World (users)

Database Designer

DBMS



Why use data modelling?

31

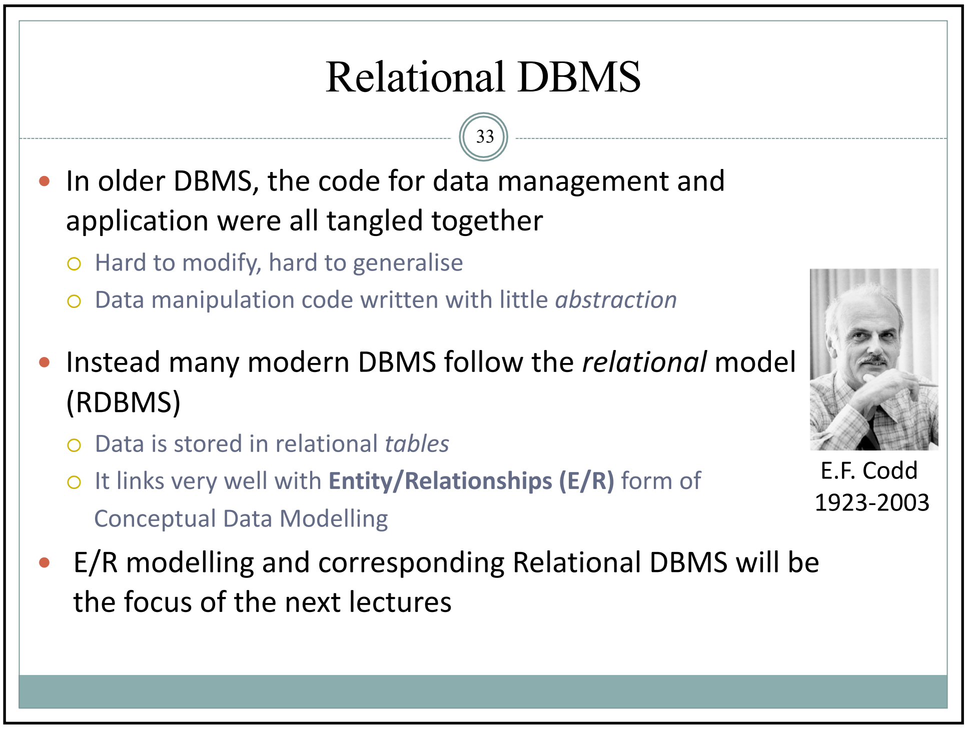
A **data model** is:

- an abstract representation of the data we wish to store
- a convention for the specification of the logical structure of real-world information

The choice of **data model** to use depends on the type of database…

- We will use the Entity Relationship Model
  - Entities, relationships and attributes (Chen, 1976)
- …which easily maps to the Relational DBMS

Once we have a conceptual data model for a problem in terms of an *Entity Relationship diagram*, we can easily generate a conceptual schema for the database

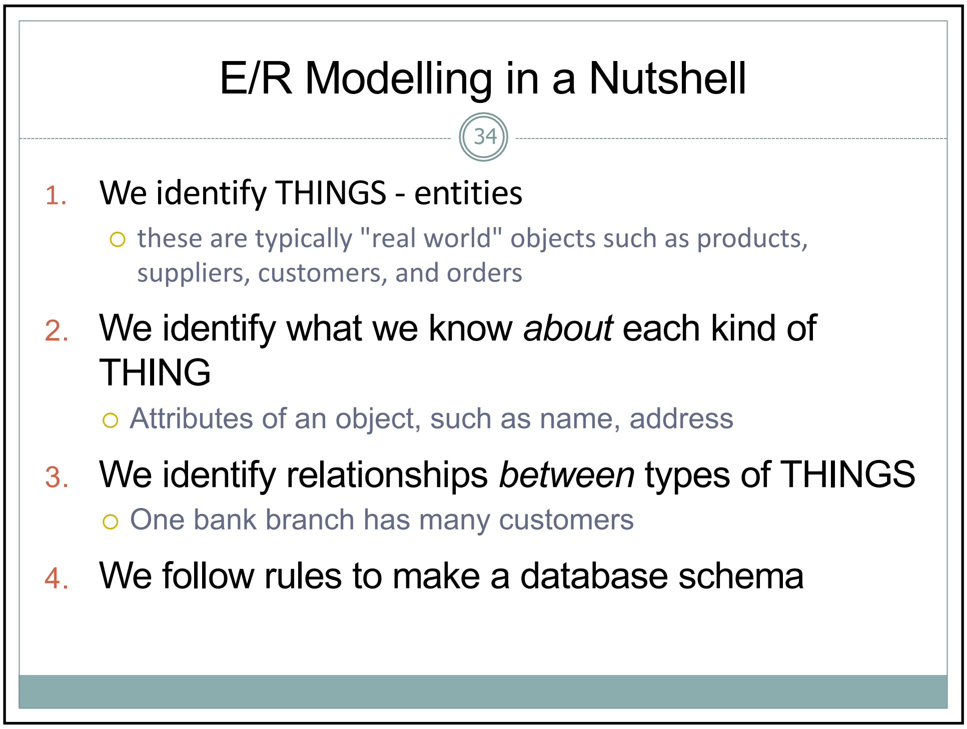


Relational DBMS

33

- In older DBMS, the code for data management and application were all tangled together
  - Hard to modify, hard to generalise
  - Data manipulation code written with little *abstraction*
- Instead many modern DBMS follow the *relational* model (RDBMS)
  - Data is stored in relational *tables*
  - It links very well with **Entity/Relationships (E/R)** form of Conceptual Data Modelling
- E/R modelling and corresponding Relational DBMS will be the focus of the next lectures

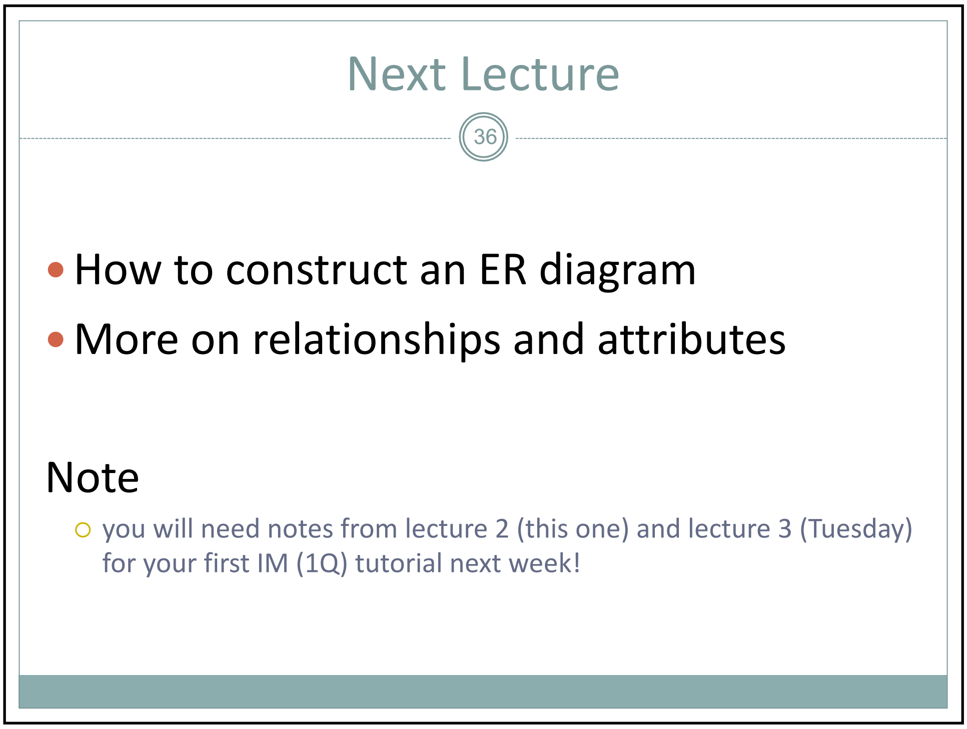
E.F. Codd
1923-2003



E/R Modelling in a Nutshell

34

1. We identify THINGS - entities
   - these are typically "real world" objects such as products, suppliers, customers, and orders
2. We identify what we know *about* each kind of THING
   - Attributes of an object, such as name, address
3. We identify relationships *between* types of THINGS
   - One bank branch has many customers
4. We follow rules to make a database schema



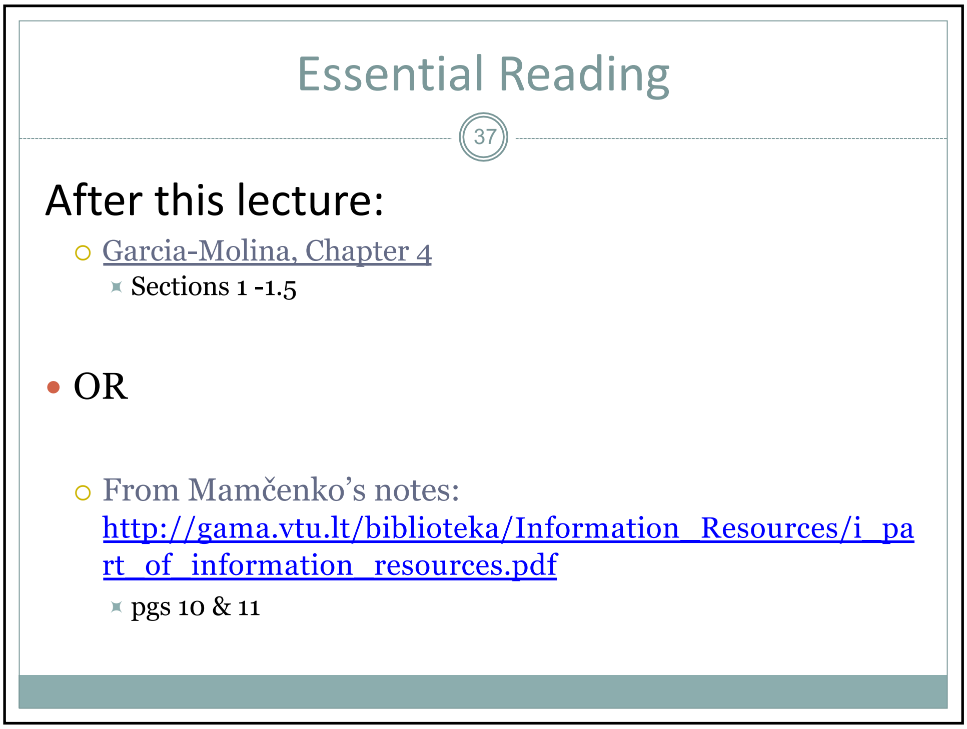
Next Lecture

36

- How to construct an ER diagram
- More on relationships and attributes

Note

- you will need notes from lecture 2 (this one) and lecture 3 (Tuesday) for your first IM (1Q) tutorial next week!



Essential Reading

After this lecture:

- Garcia-Molina, Chapter 4
  - Sections 1 -1.5

- OR

- From Mamčenko’s notes: http://gama.vtu.lt/biblioteka/Information_Resources/i_part_of_information_resources.pdf
  - pgs 10 & 11