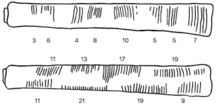
What's in a Database (Course)?

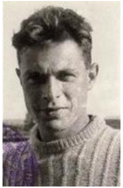
Stéphane Bressan

COM1-03-20
steph@nus.edu.sg

20,000 BCE

The Ichango bones (kept in the collections of the Royal Belgian Institute of Natural Sciences, Brussels, Belgium), discovered in 1950 by Jean de Heinzelin de Braucourt in the village of Ichango on the shores of lake Edward in the now Democratic Republic of Congo, are artefacts suggesting a tally device made and used around 20,000 before our common era.





2,000 BCE

The invention of writing systems has been arguably motivated by the need and will to store and manage data for all kinds of human activities: science, engineering, trade, law, and entertainment.

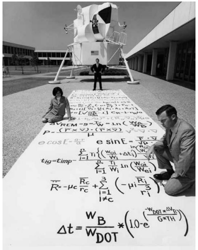
A clay tablet, kept in the collections of Cambridge University, reads “18 jars of pig fat – Balli. 4 jars of pig fat – Nimgir-ab-lah. Fat dispensed (at ?) the city of Zabala. Ab-kid-kid, the scribe. 4th year 10th month”.

20^{th} Century: Sputnik, Yuri Gagarin, Apollo, and Neil Armstrong

"On May 25, 1961, United States President John F. Kennedy challenged American industry to send an American man to the moon and return him safely to earth, thus launching the Apollo program. North American Aviation, in partnership with IBM, fulfilled the requirement for an automated system to manage large bills of material for the construction of the spacecraft in 1965." About the Rice Stadium speech at Rice University, Houston, Texas. From IBM Knowledge Center (`http://www-01.ibm.com`)

IMS

”The first ”READY” message was displayed on an IBM 2740 typewriter terminal at the Rockwell Space Division at NASA in Downey, California, on August 14, 1968. Less than a year later, on July 20, 1969, Apollo 11 landed on the moon’s surface. ICS was subsequently relaunched as Information Management System/360 (IMS/360) [...]”
From IBM Knowledge Center (`http://www-01.ibm.com`)



The IMS Model

IMS is a hierarchical model. A hierarchical database consists of a collection of records connected to one another through links. Each record is a collection of fields (attributes), each of which contains only one data value. A link is an association between precisely two records. In the IMS model, only many-to-one links are allowed. The database forms a tree or a forest. This allows efficient serialization, storage and retrieval on sequential medium such as tape.

DBTG

The List Processing Task Force (LPTF) was founded in 1965 by the Cobol[a][b] Committee, formerly Programming Language Committee, of the Conference of Data Systems Language. In 1967, it was renamed the Data Base Task Group (DBTG)

[a] http://www.csis.ul.ie/cobol

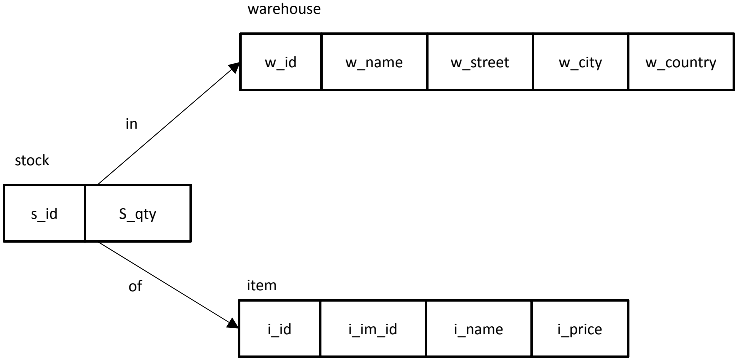
[b] http://blog.codinghorror.com/cobol-everywhere-and-nowhere

The DBTG CODASYL Model

The CODASYL DBTG 1971 report is the first database standard specification.

The DBTG CODASYL Model

DBTG is a network model. In a network database consists of a collection of records connected to one another through links. Each record is a collection of fields (attributes), each of which contains only one data value. A link is an association between precisely two records. In the DBTG model, only many-to-one links are allowed, but the database may form a graph.



warehouse

w_id	w_name	w_street	w_city	w_country

in

stock

s_id	S_qty

of

item

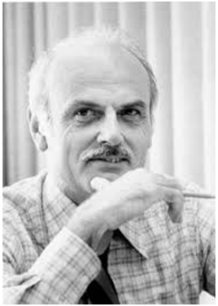
i_id	i_im_id	i_name	i_price

Example Query

Print the total quantity of aspirin in stock.

```
sum := 0;
item.i_name := ''aspirin'';
find any item using i_name;
find first stock within wh;
while DB-status = 0 do
begin
get stock;
sum := sum + stock.s_qty;
find next stock  within wh;
end
print (sum);
```

“Future users of large data banks must be protected from having to know how the data is organized in the machine.”
Edgar F . Codd (”A Relational Model for Large Shared Data Banks” Communication of the ACM, Vol 13, #6)



Example Query

Print the total quantity of aspirin in stock.

```
SELECT SUM(s.s_qty)
FROM stock s, i item
WHERE s_i_id = i_i_id
ANDS i.i_name = 'aspirin'

```

A database needs a *data model* (and the corresponding languages) that prescribes what data elements can be stored and managed how they are organised, and how they and their relative organisation should be interpreted.

Data Models

- Hierarchical Model 1965 (IMS) (see `codex.cs.yale.edu/avi/db-book/db6/appendices-dir/e.pdf`)
- Network Model 1965 (DBTG) (see `codex.cs.yale.edu/avi/db-book/db6/appendices-dir/d.pdf`)
- Relational Model (1NF) 1970s
- Nested Relational Model (NF2) 1970s
- Complex Object 1980s
- Object Model 1980 (OQL, O2)
- Object Relational Model 1990s (SQL)
- XML (DTD), XML Schema 1990s (Xpath, Xquery, XSLT, existdb)
- NoSQL Databases (MongoDB, Neo4J, CouchDB, Cassandra)

Protect the Users

Edgar F. Codd argued that database management required data independence.

Instead of the physical optimised actual representation of the data, the user should interact with a logical model. He proposed the relational model in which the user only creates relations (tables) and manipulates tuples (records, rows) with attributes (columns).

DBMS

We do not program database applications from scratch. We use database management systems. Database management Systems are generic platforms for the implementation and management of database applications.

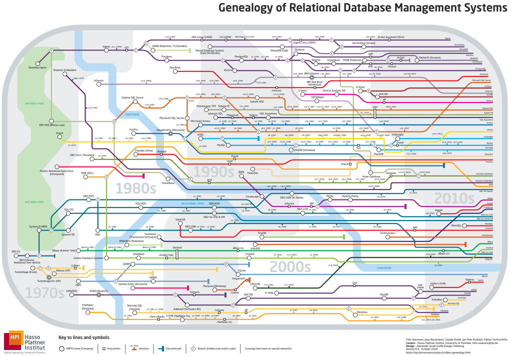
RDBMS

Most popular database management systems since the 1970's are relational. They implement the relational model of Edgar F. Codd. They implement a subset of one version of the international standard for the SQL language.

Popular relational database management systems include Oracle, Microsoft SQL Server, IBM DB2, Sybase, PostgreSQL, MySQL, MariaDB, SQLite, MS Access, SAP Hana, etc.

. . .

The Hasso-Plattner-Institut of the University of Potsdam, Germany, maintains a genealogy of relational database management systems at `hpi.de/naumann/projects/rdbms-genealogy.html`.


Genealogy of Relational Database Management Systems
1970s
1980s
1990s
2000s
2010s
HPI Hasso Plattner Institut
Key to lines and symbols
DBMS name (Company)
Acquisition
Versions
Discontinued
Branch (intellectual and/or code)
Crossing lines have no special semantics

Not only SQL

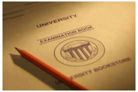
The need for different data models and different performance priorities in modern applications compelled the development of and use of NoSQL systems together with relational database management systems.

Popular NoSQL systems include eXist-db, MongoDB (document databaes), neo4j (graoph database), Couchbase (key-value store), Cassandra (wide-column store), etc., as well as geographical information systems (GIS), scientific databases, multimedia databases, etc.

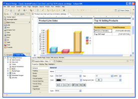
. . .

We are using SQLite and PostgreSQL.

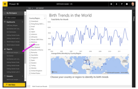
Typical online transaction processing (OLTP) database applications include airline reservation, banking, university, the e-shop around the corner and my address book.

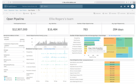


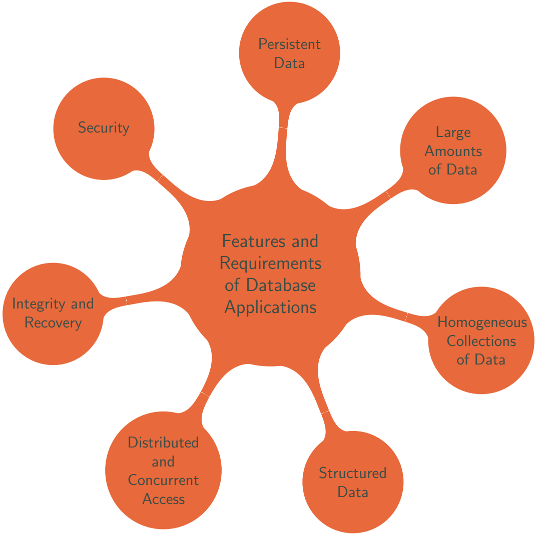
Typical online analytical processing (OLAP) database applications include business analysis and reporting for auditing, forecasting and business intelligence.



https://newsroom.fb.com/news/2018/04/restricting-data-access/
Australia : 311127
Vietnam : 427446
Brazil : 443117
India : 562455
Canada : 622161
Mexico : 789880
United Kingdom : 1079031
Indonesia : 1096666
Philippines : 1175870
United States : 70632350
United States
Philippines
Indonesia
United Kingdom
Mexico
Canada
India
Brazil
Vietnam
Australia


Power BI
Birth Trends in the World
Choose your country or region to identify its birth trends




Features and Requirements of Database Applications
Persistent Data
Large Amounts of Data
Homogeneous Collections of Data
Structured Data
Distributed and Concurrent Access
Integrity and Recovery
Security

Persistence

How can data survive the process that created it, and be reused by other processes?

Primary memory is volatile. Secondary and tertiary memory is persistent.

Every data created by a successful SQL transaction is persistent.

Large Amounts

There were 187 million registered voters in the 2019 Indonesian elections.

Where could one store the names, identification numbers, and voting stations of voters?

There were 187 million registered voters and 805,000 voting stations in the 2019 Indonesian elections.

In order to print and distribute the voters lists one need to sort the voters in alphabetical order of their voting stations and names?

External Algorithms

When data is to be stored on secondary or tertiary storage, then we need to devise efficient algorithms taking into account the dominant cost of Input/Output operations. Such algorithms are called external algorithms (e.g., external sort).

SQL allows to use these algorithms without implementing them

```
SELECT *
FROM voters
ORDER BY district , name
```

There were 185 million registered voters in the 2019 Indonesian elections. Imagine the original tapes contain duplicate entries.

Think about an algorithm to remove the duplicate entries (although you won't need it).

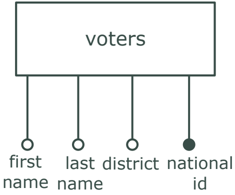
SQL allows to eliminate duplicates a priori with unique and primary key constraints or a posteriori with the keyword `DISTINCT`.

Homogeneous Data

We mostly consider applications in which data comes in homogeneous collections.

```sql
CREATE TABLE persons
CREATE TABLE bottles
CREATE TABLE cars
```

Structured Data



```sql
CREATE TABLE voters (
first_name VARCHAR(32),
last_name VARCHAR(32),
district VARCHAR(64),
national_id NUMBER));
```

Homogenous Collections of Structured Data

Since applications manipulate homogenous collection of structured data we can use a model and a language as simple as the relational model and SQL.

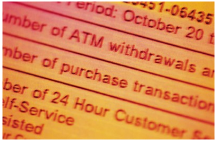
```sql
CREATE TABLE voters (
 first_name VARCHAR(32),
 last_name VARCHAR(32),
 district VARCHAR(64),
 national_id NUMBER));

SELECT last_name
FROM voters
WHERE first_name = 'Bambang';
```

Distributed and Concurrent Access: Transaction

A database operation is an insertion, an update, a deletion or a query.
CRUD is the acronym for CREATE, READ, UPDATE and DELETE.

A database transaction is a logical unit of work carried out by a user or a client application. It consists in the execution of a sequence of operations.


eriod: October 20
umber of ATM withdrawals a
nber of purchase transaction
ber of 24 Hour Customer S
elf-Service
sisted

In some contexts the programmer uses the SQL commands `BEGIN`, `END` or `COMMIT`, `ABORT`, and `ROLLBACK`. In other context they are implicit.

Distributed and Concurrent Access

How to resolve conflicts and maintain the integrity of data and when data is shared by users and processes that are possibly distributed over a network?

In a nutshell, the solution offered by most database management systems is to serialise all the operations in the different transactions to avoid conflicts.

Integrity and Recovery

How to maintain the integrity of data in spite of possible application, system, or media failures?

Relational database management system maintain user-defined integrity constraints.

In a nutshell, the solution used by most database management systems is to log all the transactions and their operations in order to be able to undo and redo the necessary operations and restore a consistent sate of the database before the failure.

Recovery: Atomicity

All actions in a transaction happen or none happen.

Concurrency Control : Consistency

If individual transactions would leave the application in a consistent state, a concurrent execution should do the same.

Concurrency Control : Isolation

Transactions can be understood independently from each other.

Recovery: Durability (Persistence)

effects of successful transactions last

Security

Security is achieved by means of access control policies. Most database management systems implement a role-based access control to control the access to the various database objects: tables, rows, columns, views, stored functions and procedures etc. for the different possible operations: create, update, delete and query or execute. Most database management systems also offer for security security mechanisms such as cryptography, authentication methods, etc. Nevertheless, application developers must be wary of all possible attacks in general and, in particular, of SQL injections.

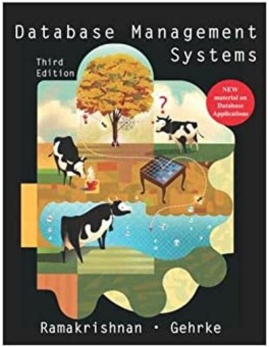
Definition

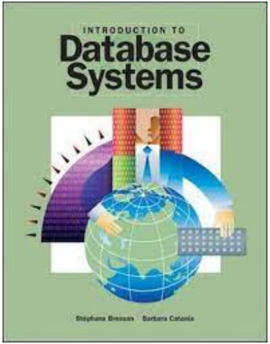
“A database application manages persistent homogeneous collections containing large amounts of structured data that are shared among distributed users and processes and whose integrity must be maintained and security controlled.
Typically, a database application is best designed for, implemented, deployed and maintained with a database management sytem.”

Syllabus

This semester we discuss the design (normalised entity-relationship modelling) and implementation (SQL) of database applications with relational database management systems for online transaction and analytical processing applications in the classical full-stack three-tier model (Web client or app, Web and application server, database server).

Recommended Textbooks


Database Management Systems
Third Edition
NEW material on Database Applications
Ramakrishnan • Gehrke



Database Management Systems by Raghu Ramakrishnan and Johannes Gehrke
Introduction to Database Systems by Stéphane Bressan and Barbara Catania

Tentative Schedule

- 11 Lectures (in-class and Zoom, including one online asynchronous lecture (CNY) and one Q&A lecture (Week 12))
- 2 Tests (Week 7 and 13, in-class)
- 9 Tutorials (in-class)

Assessement

- Tutorial attendance 5% (-1 mark for each absence[a] after two absences)
- Several Projects 45% (some individual, some in group)
- Test 1 (Week 7) 25% (in-class, digital, individual)
- Test 2 (Week 13) 25% (in-class, digital, individual)

[a] Official leave of absence are not counted. Apply for official leave of absence by sending an application with the relevant documentation to the School's Undergraduate Office.