ISAD353: Advanced Databases and Data Management

Seminar 2: From SQL to NoSQL

Marco Palomino

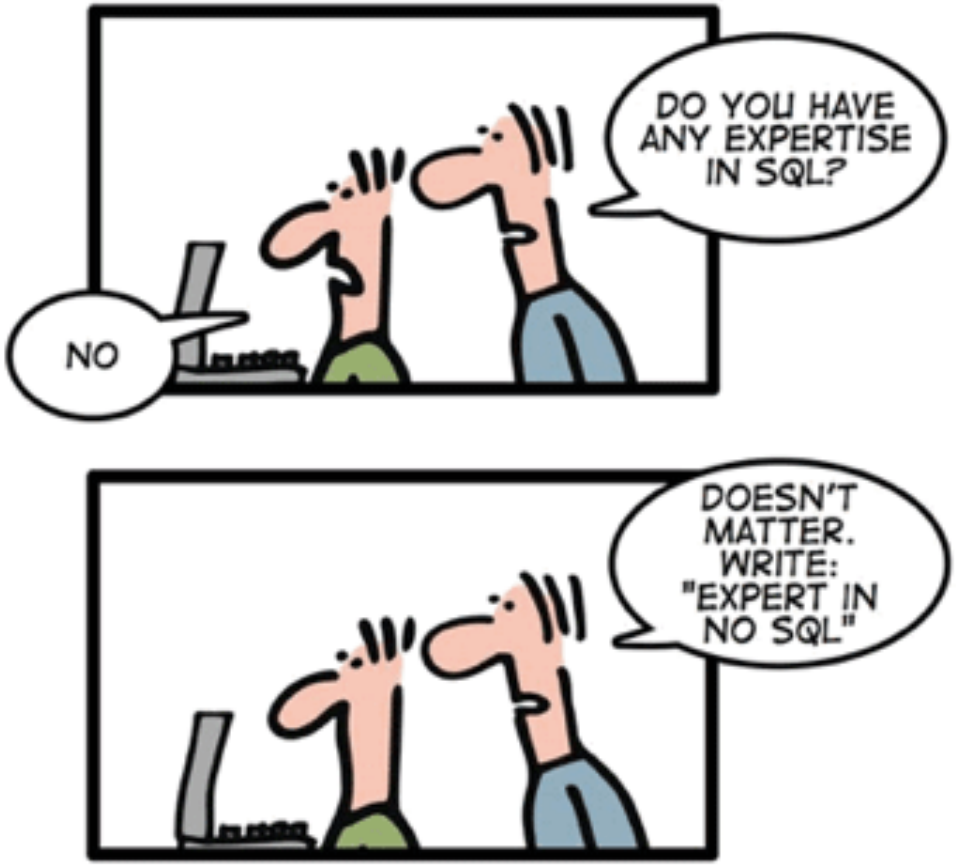
Module Aims

To expose students to the challenges of solutions for managing, processing, analysing and interpreting large amounts of unstructured data within relational and **non-relational database** environments.

Assessed Learning Outcomes

Critically evaluate current and emerging approaches for analysing and interpreting data using data mining and business intelligence techniques.

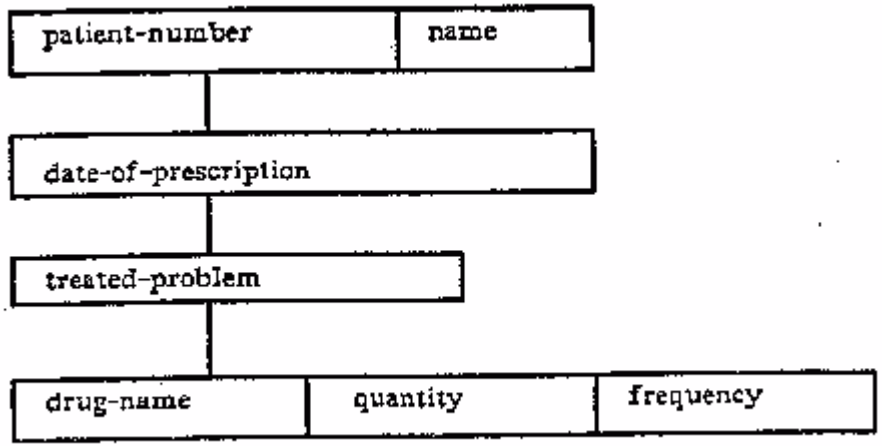
Outline



- Historical background
  - From SQL to object oriented databases
  - Internet
- NoSQL models
- Coursework

The first database

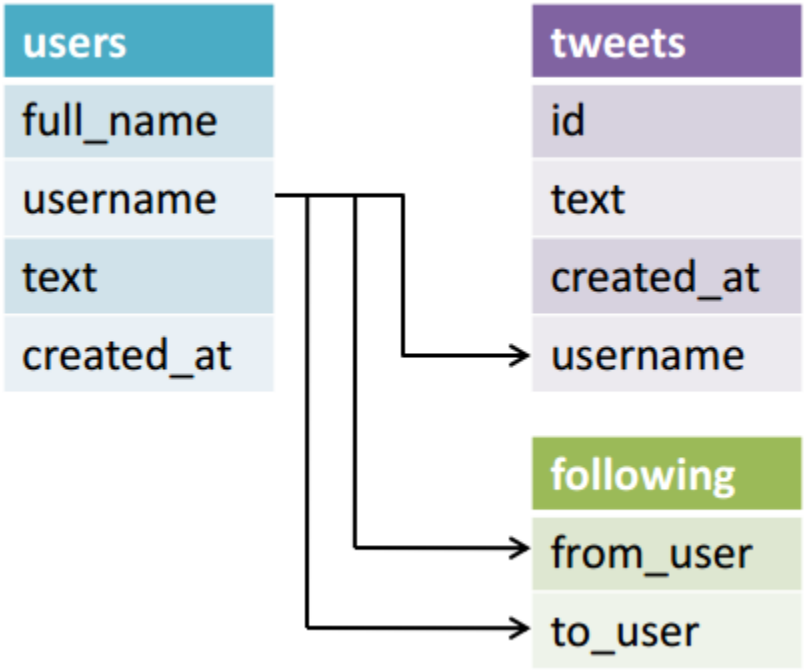
- Computerised databases started in the 1960s.
- The **SABRE** system that was used by **IBM** to help **American Airlines** manage its reservations data.
- **MUMPS** (Massachusetts General Hospital Utility Multi-Programming System), or alternatively **M**: A general-purpose computer programming language that provides **ACID** (Atomic, Consistent, Isolated, and Durable) transaction processing.



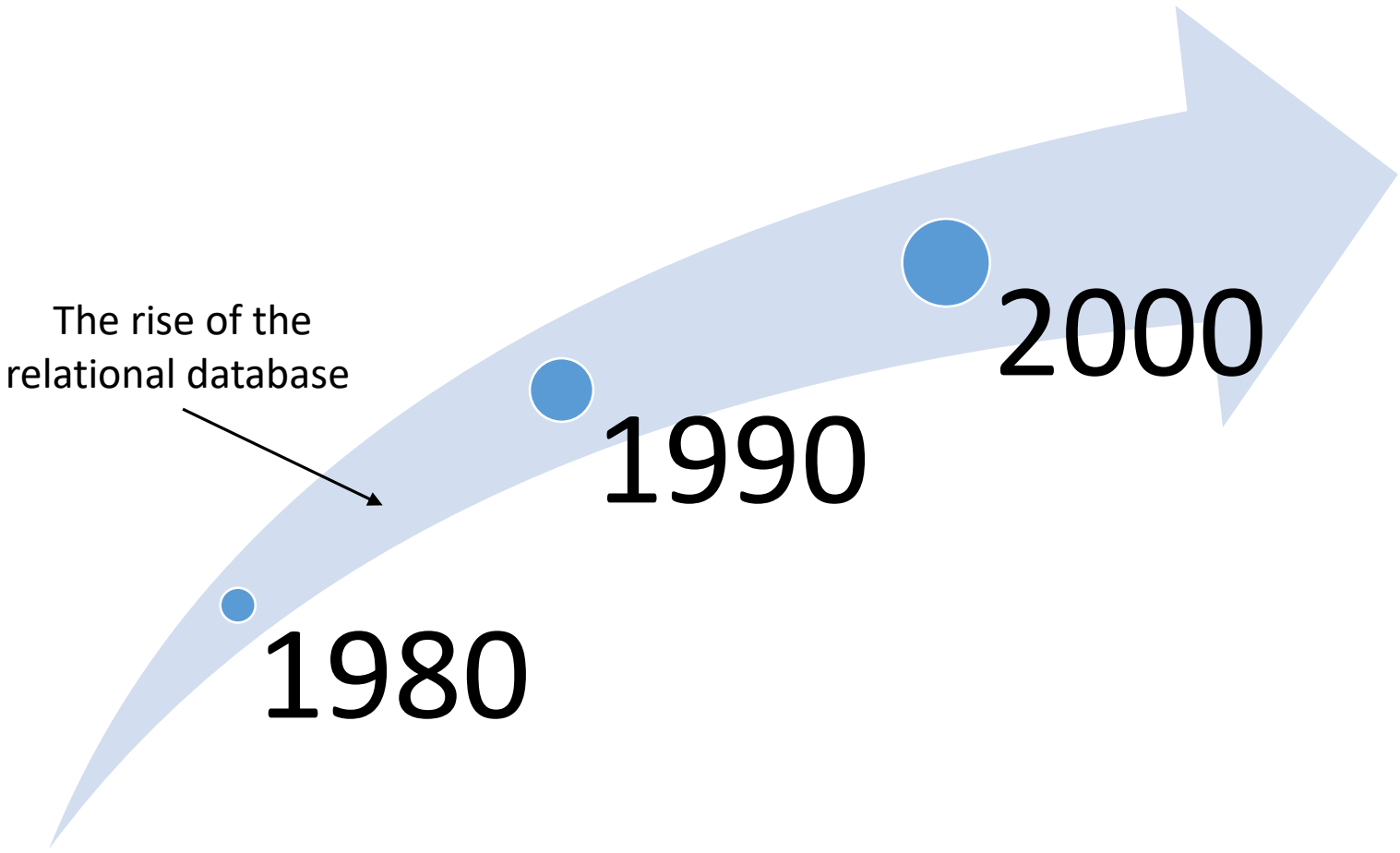
Relational database

- A **relational database** is a digital database whose organisation is based on the **relational model of data**, as proposed by E. F. Codd in 1970.

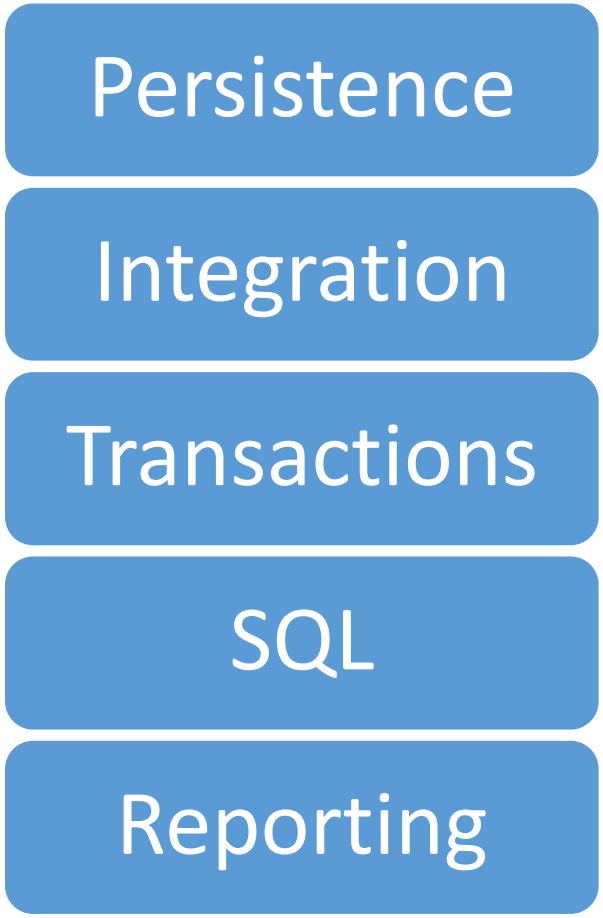
- The relational model organises data into one or more **tables** (or **"relations"**) of **columns and rows**, with a **unique key** identifying each row. Rows are also called **records**.



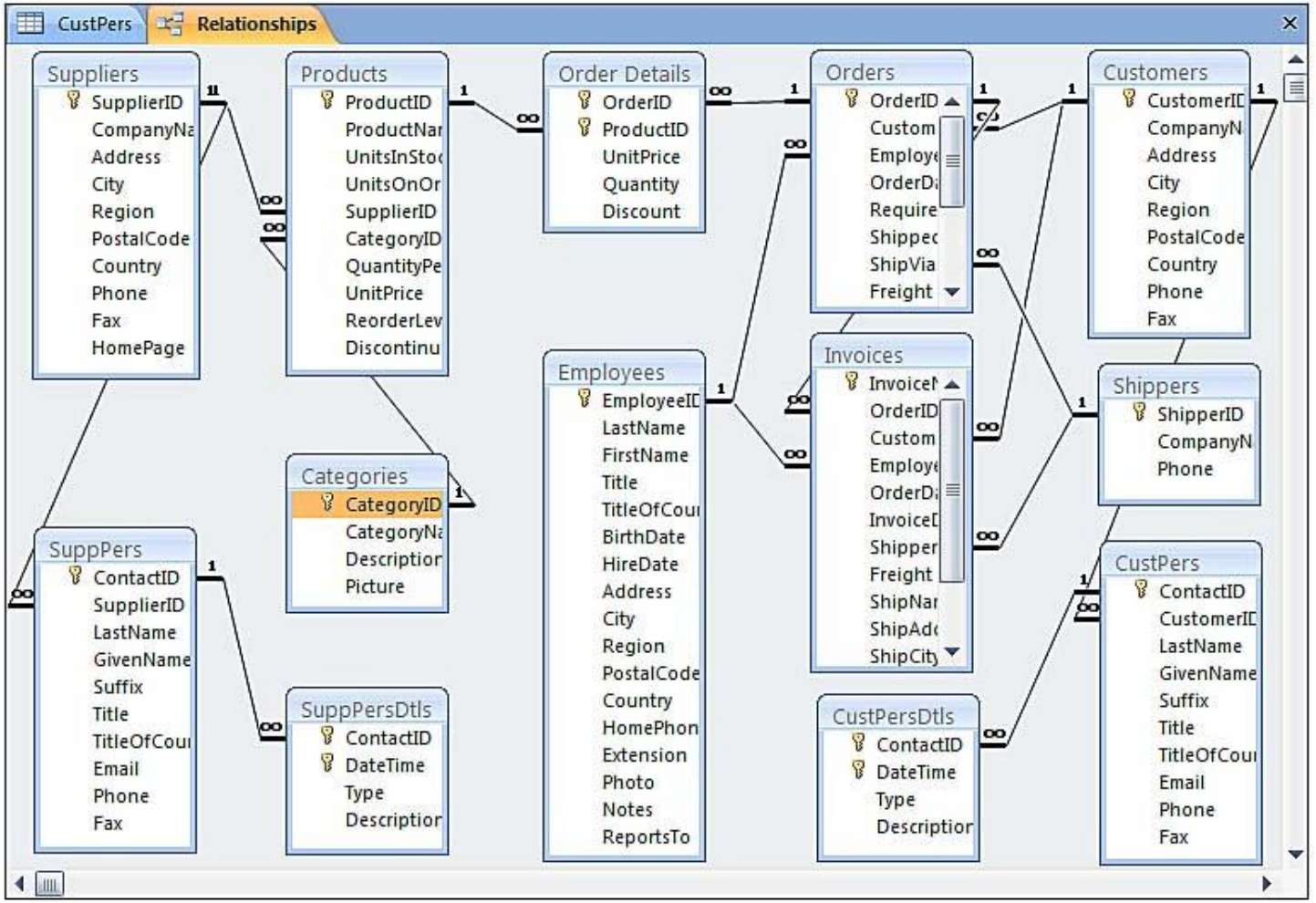
Background (history)



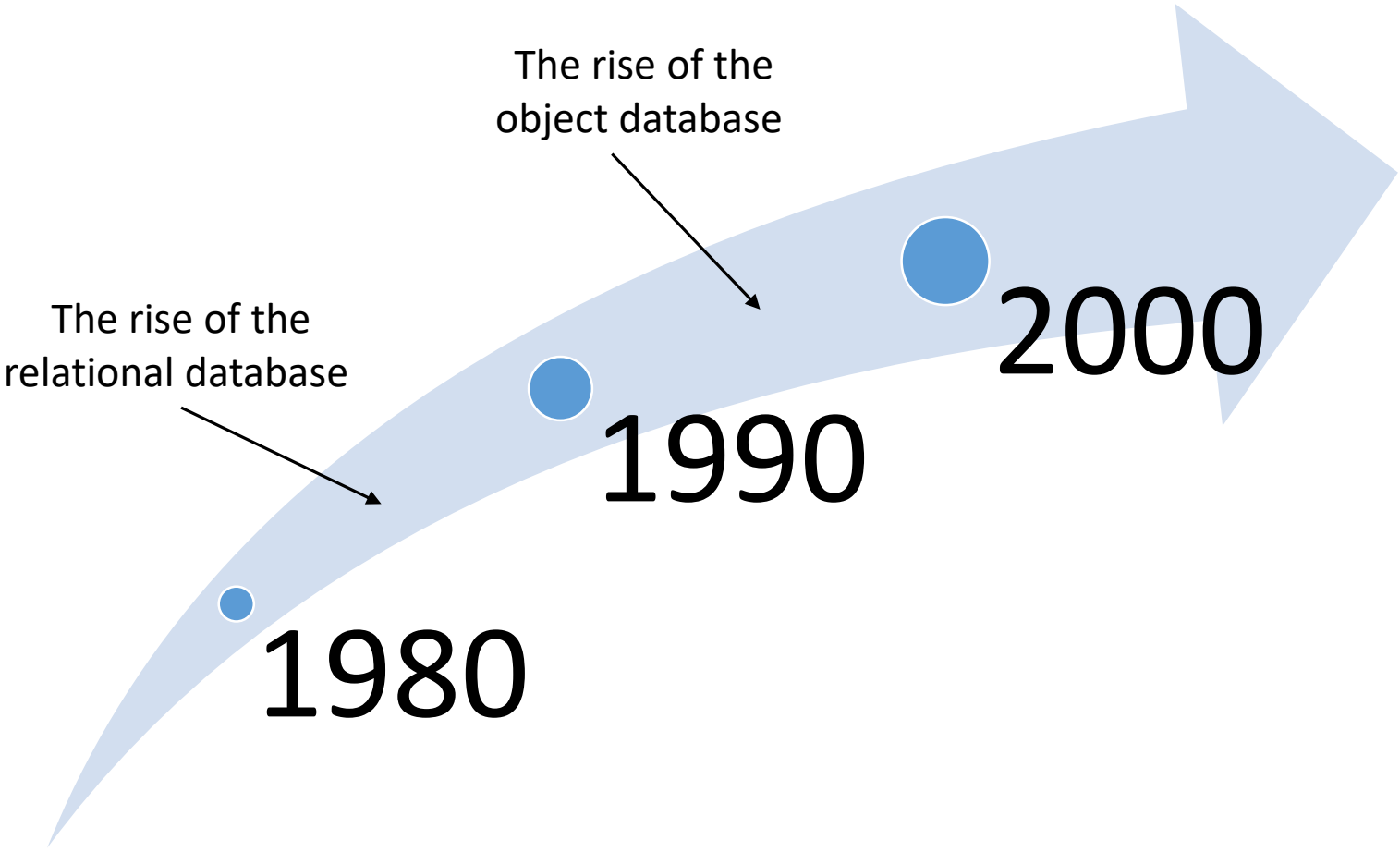
Benefits of the relational model



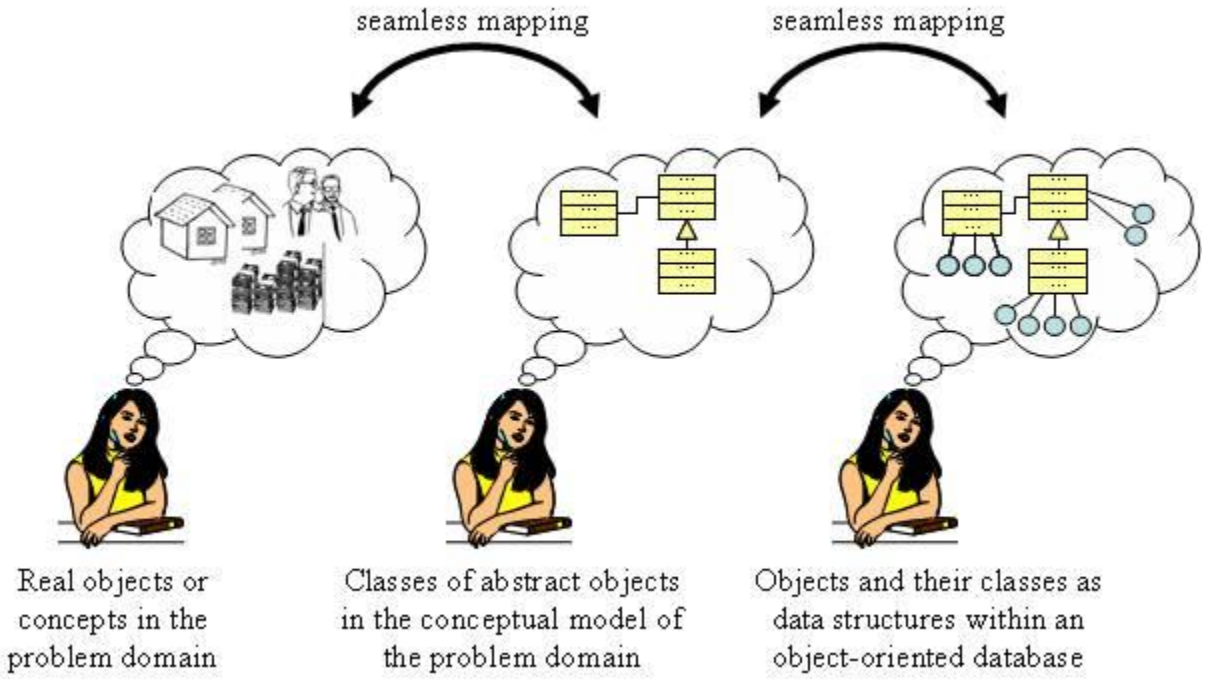
Impedance mismatch



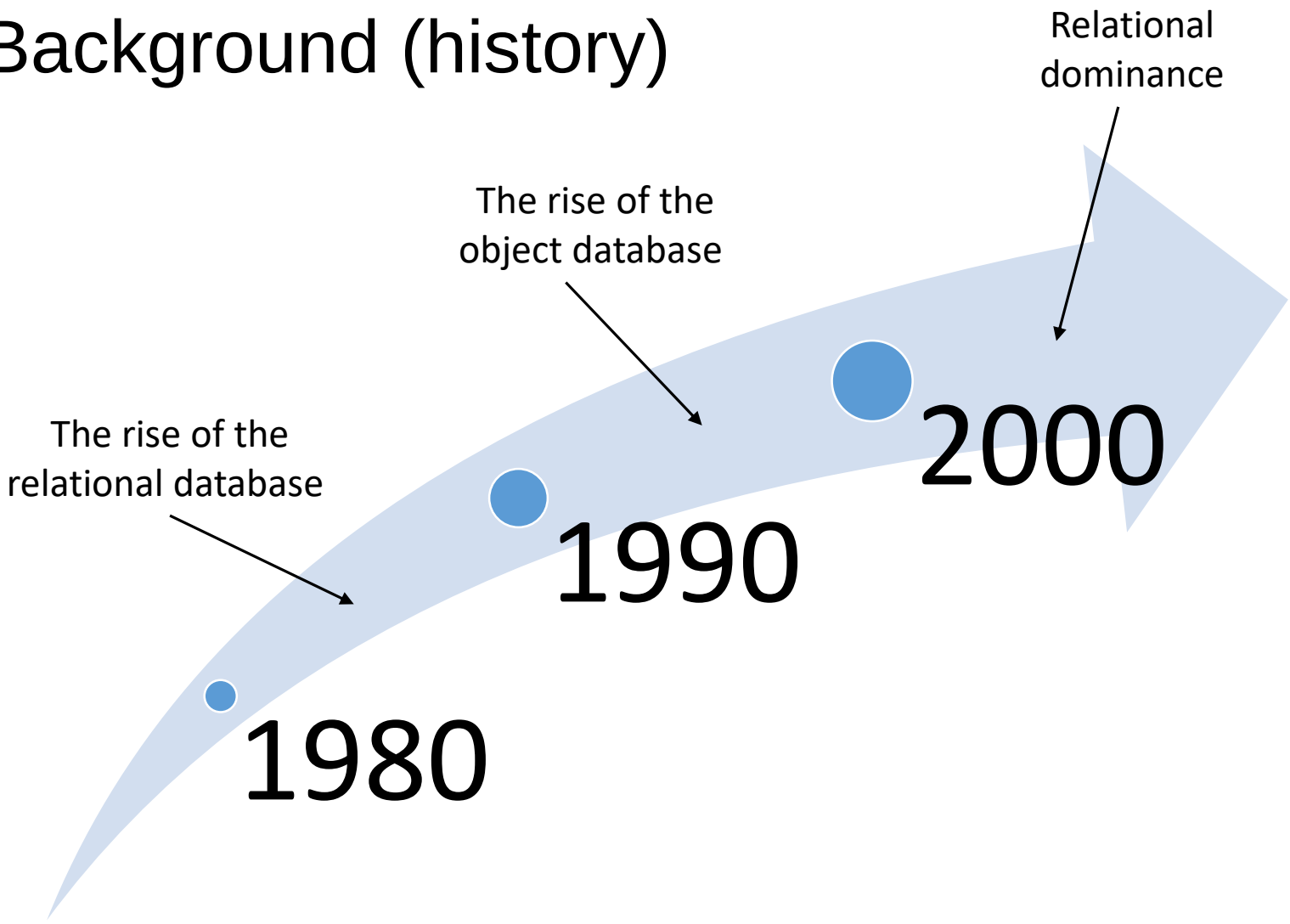
Background (history)



Object databases



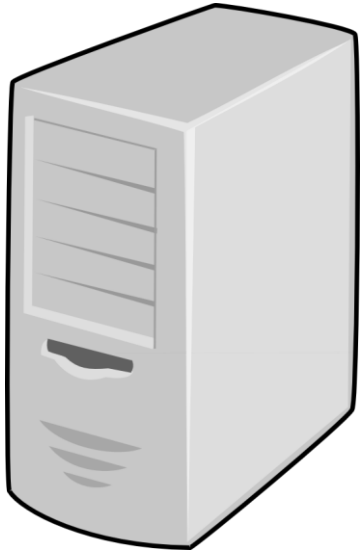
Background (history)



Internet traffic

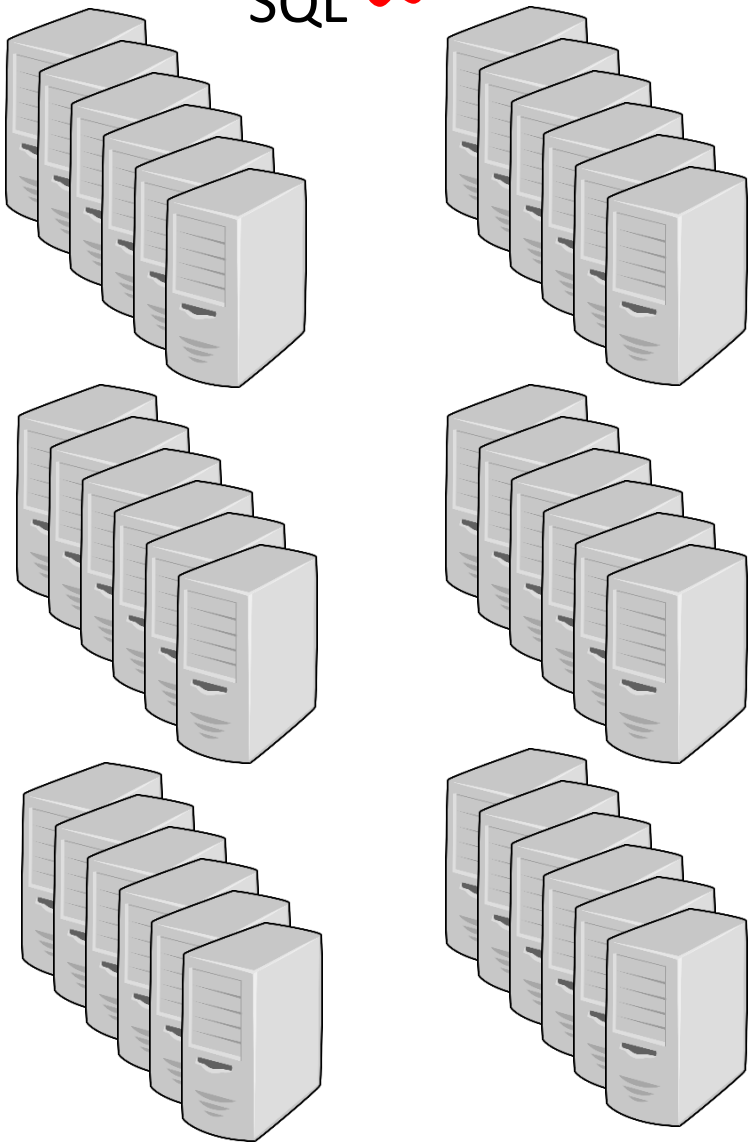
Servers

SQL ✓



Unnatural acts...

SQL ✗



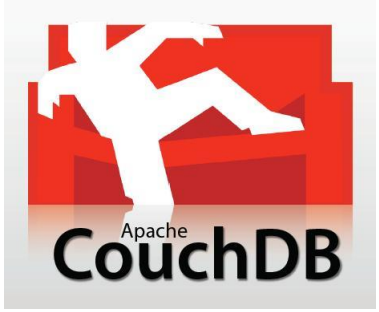
New data storage systems

Google BigTable

NoSQL... ?

- NoSQL was a hashtag (`#nosql`) chosen for a meetup organised by **Johan Oskarsson** in San Francisco in 2009 to discuss new databases. There were presentations delivered by **Voldemort**, **Cassandra**, **Dynomite**, **HBase**, **Hypertable**, **CouchDB**, and **MongoDB**.

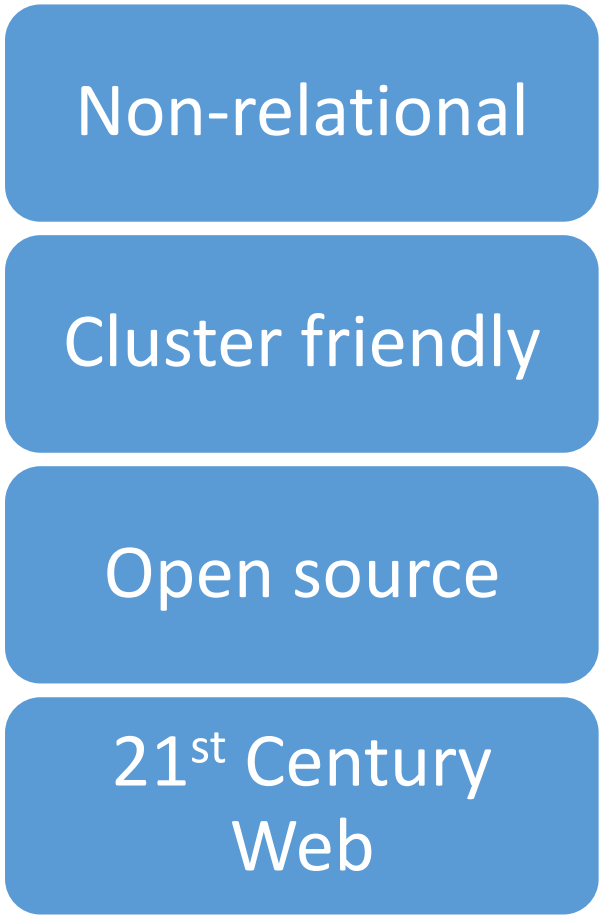
- "NoSQL is an accidental term with no precise definition". [Sadalage & Fowler: NoSQL Distilled, 2012]



NoSQL

- **NoSQL** means **Not Only SQL**, implying that when designing a software solution or product, there is more than one storage mechanism that could be used.

NoSQL features



Apache Hadoop

- **What is Apache Hadoop?**

What Is Apache Hadoop?

"The **Apache Hadoop** software library is a framework that allows for the distributed processing of large datasets across clusters of computers using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage."

http://hadoop.apache.org/

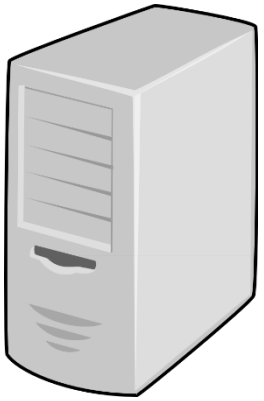
THE APACHE® SOFTWARE FOUNDATION

Origins

Google	Apache
Google File System (GFS)	Hadoop Distributed File System (HDFS)
Google MapReduce	MapReduce
Google BigTable	Apache HBase

Why Hadoop?

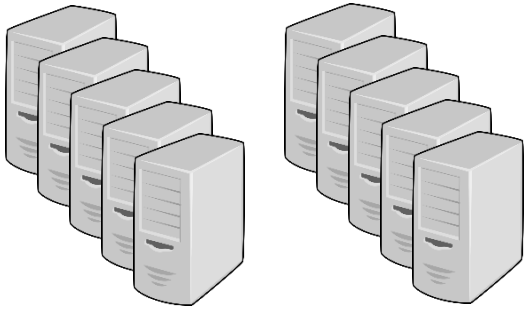
Read 1 TB of data



1 machine

- 4 I/O channels
- Each channel: 100 MB/s

~45 minutes



10 machines

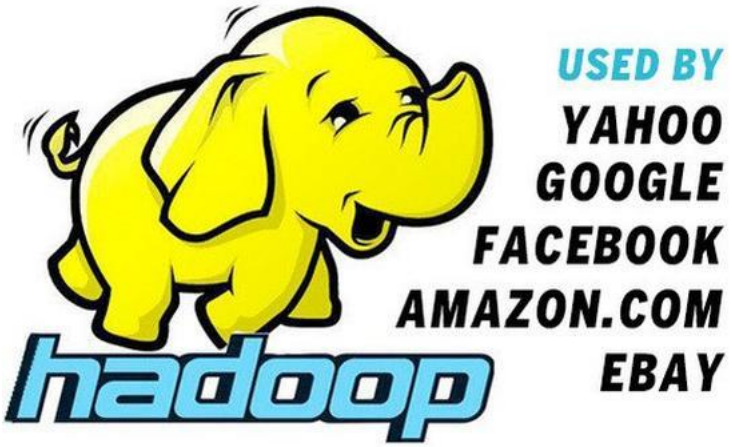
- 4 I/O channels
- Each channel: 100 MB/s

~4.5 minutes

Hadoop

- **Apache Hadoop** is an open-source software framework written in Java for distributed storage and distributed processing of very large data sets on computer clusters built from commodity hardware.

- Apache Hadoop is an open source platform for data storage and processing that
  - ✓Distributed
  - ✓Scalable



Components

- There are two **primary components** at the core of Apache Hadoop:

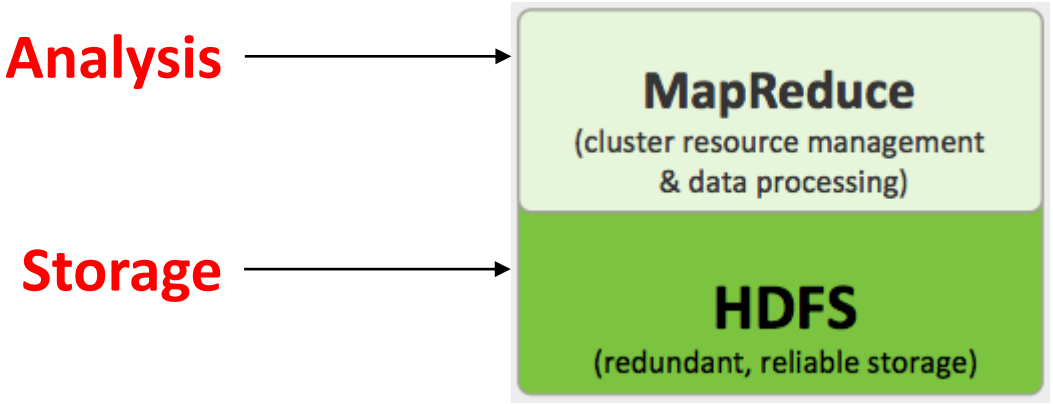
Data storage and management → **HDFS**

Processing and computation → **MapReduce**

- Hadoop Framework = **HDFS + MapReduce**

Hadoop Core Components

- **HDFS**
  - Distributed File System
  - Responsible for storing the data
- **MapReduce**
  - Distributed Data Processing Model
  - Responsible for processing the data in a massive parallel manner



Emerging data models in NoSQL

- **Bigtable:** Google Bigtable
- **Key-value store**, or **key-value database:** Redis, MemcacheDB, Berkeley DB (BDB), HamsterDB...
- **Document-oriented database**, or **document store**: MongoDB, CouchDB, OrientDB, RavenDB, Lotus Notes....
- **Column-family stores:** Cassandra, Hbase, Hypertable, Amazon DynamoDB...
- **Graph database**: Neo4j, HyperGraphDB...