

MY472 - Week 9: Working with Online Databases

Pablo Barbera & Akitaka Matsuo

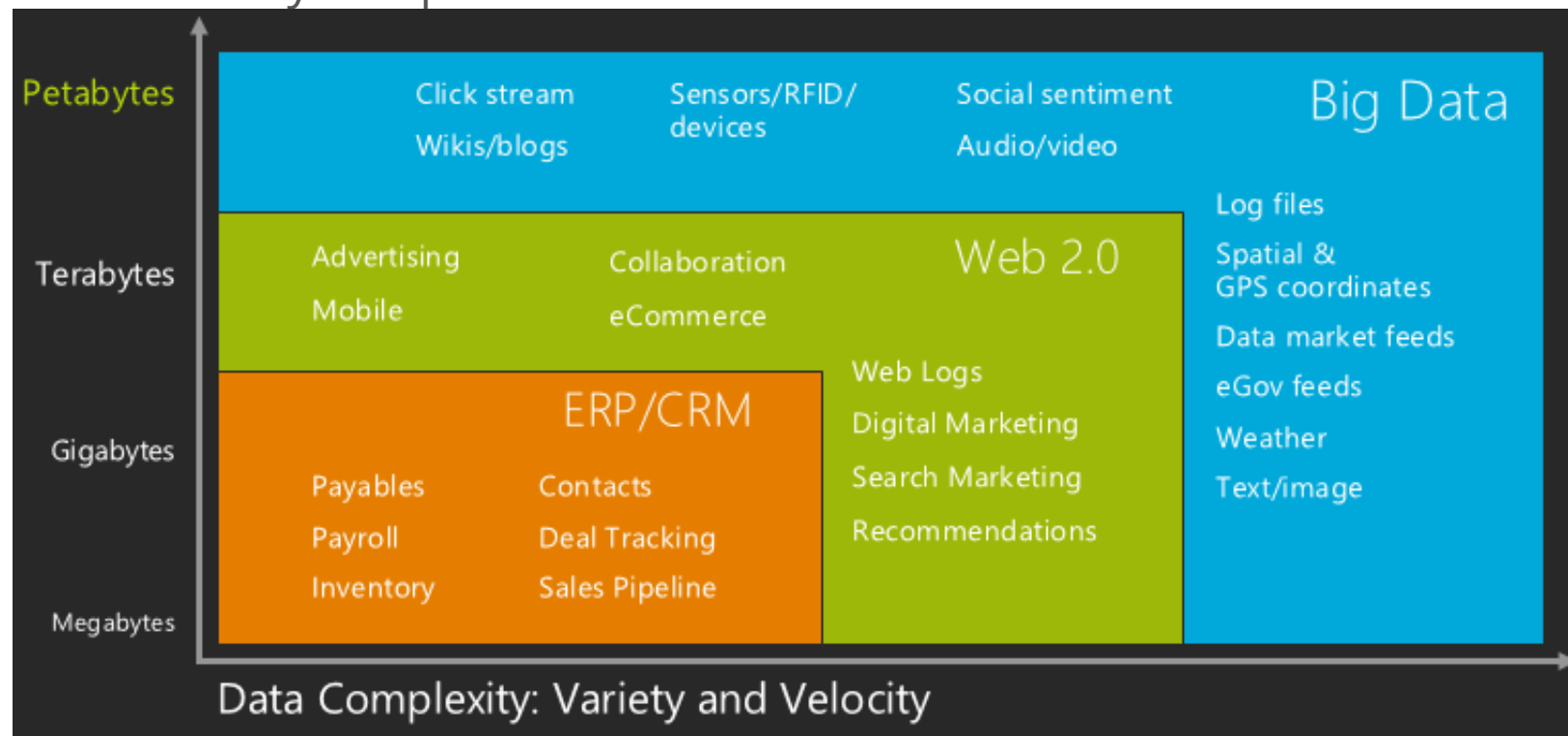
November 27, 2018

Outline

- Database Solution for Big Data
- SQL vs noSQL
- Cloud solutions
 - GCP Bigquery
 - AWS Redshift/DynamoDB
- Examples
 - Mongo DB
 - Google Bigquery

Big Data

- Your data can be really big
 - Gigabytes? Terabytes? Petabytes or more?
- And also very complicated



Database Solutions Big Data

- Different types of databases (SQL vs NoSQL)
- Cloud solution using fully managed services

SQL or noSQL

SQL?

SQL: Structured Query Language

- We have learned how to run query in SQL databases
- Example:

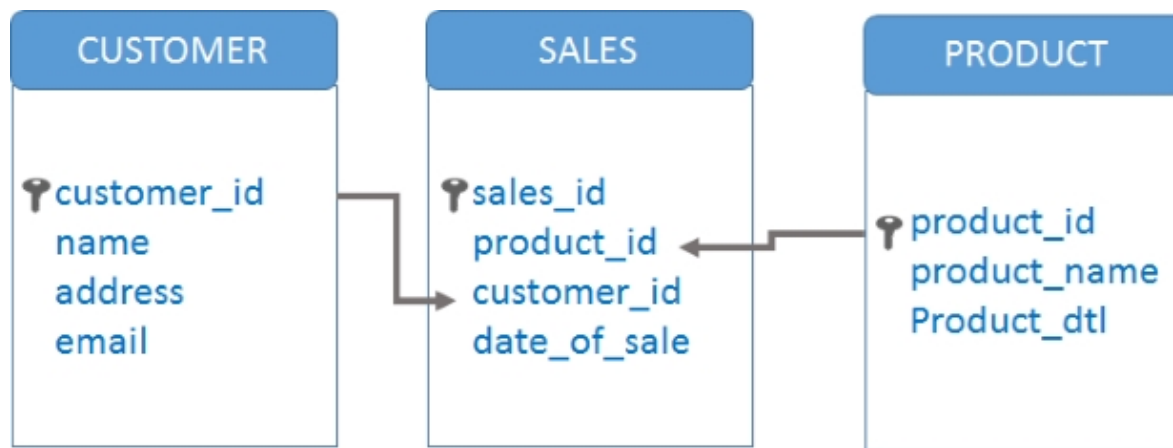
```
SELECT name, party FROM congress;
```

- SQL databases has a strict structure

SQL Structure

It's all about relations

A simple e-commerce example



SQL: Review

- The result of an SQL query is always a table
- There are a number of commands, and many of them are about the relations between tables
SELECT columns
 - **FROM** a table in a database
 - **WHERE** rows meet a condition
 - **GROUP BY** values of a column
 - **ORDER BY** values of a column when displaying results
 - **LIMIT** to only X number of rows in resulting table
 - Always required: **SELECT** and **FROM**. Rest are optional.
- **SELECT** can be combined with operators such as **SUM, COUNT, AVG...**
- To merge multiple tables, you can use **JOIN**

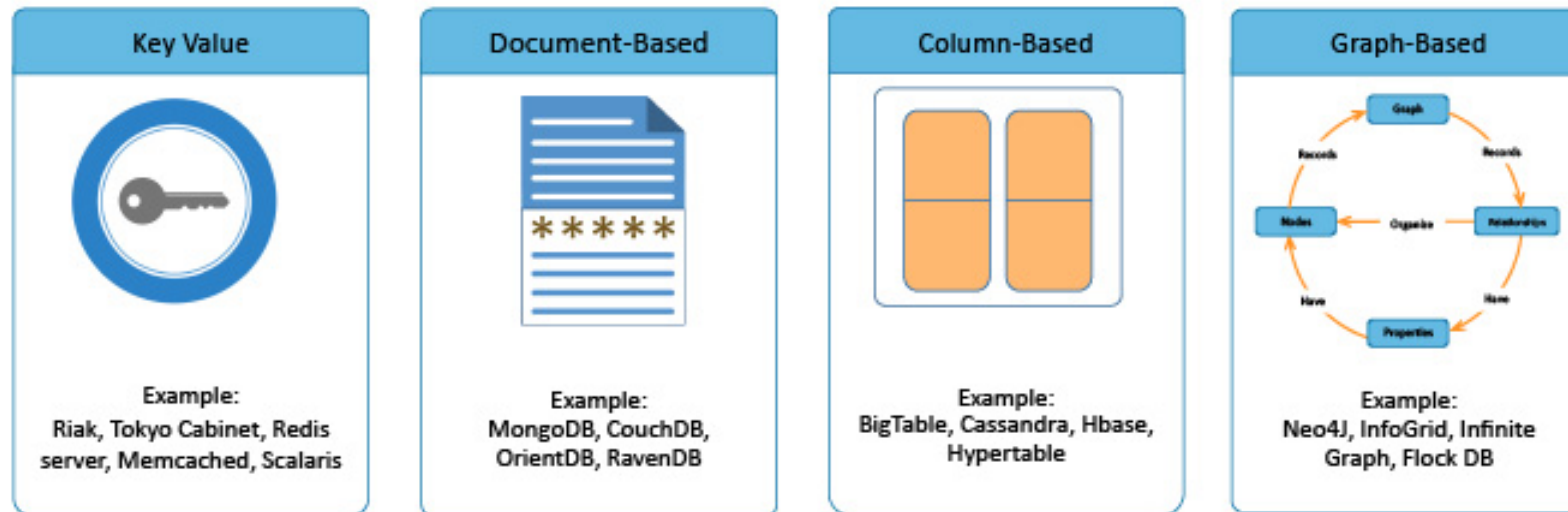
noSQL

A noSQL (originally referring to “non SQL”, “non relational” or “not only SQL”) database provides a mechanism for storage and retrieval of data which is modeled in means other than the tabular relations used in relational databases.

noSQL databases are good for the data with:

- High data velocity – lots of data coming in very quickly
- Data variety – data can be structured, semi-structured and unstructured
- Data volume - total size of data
- Data complexity - stored in many locations

noSQL: types



© Simplelearn. All rights reserved.

simplelearn

From: [Simplelearn](https://www.simplelearn.com/)

noSQL: Pros and Cons

PROS

Massive scalability

High availability

Schema flexibility

Sparse and semistructured data

CONS

Limited query capabilities

Not standardized

Not matured

Developer heavy

MongoDB

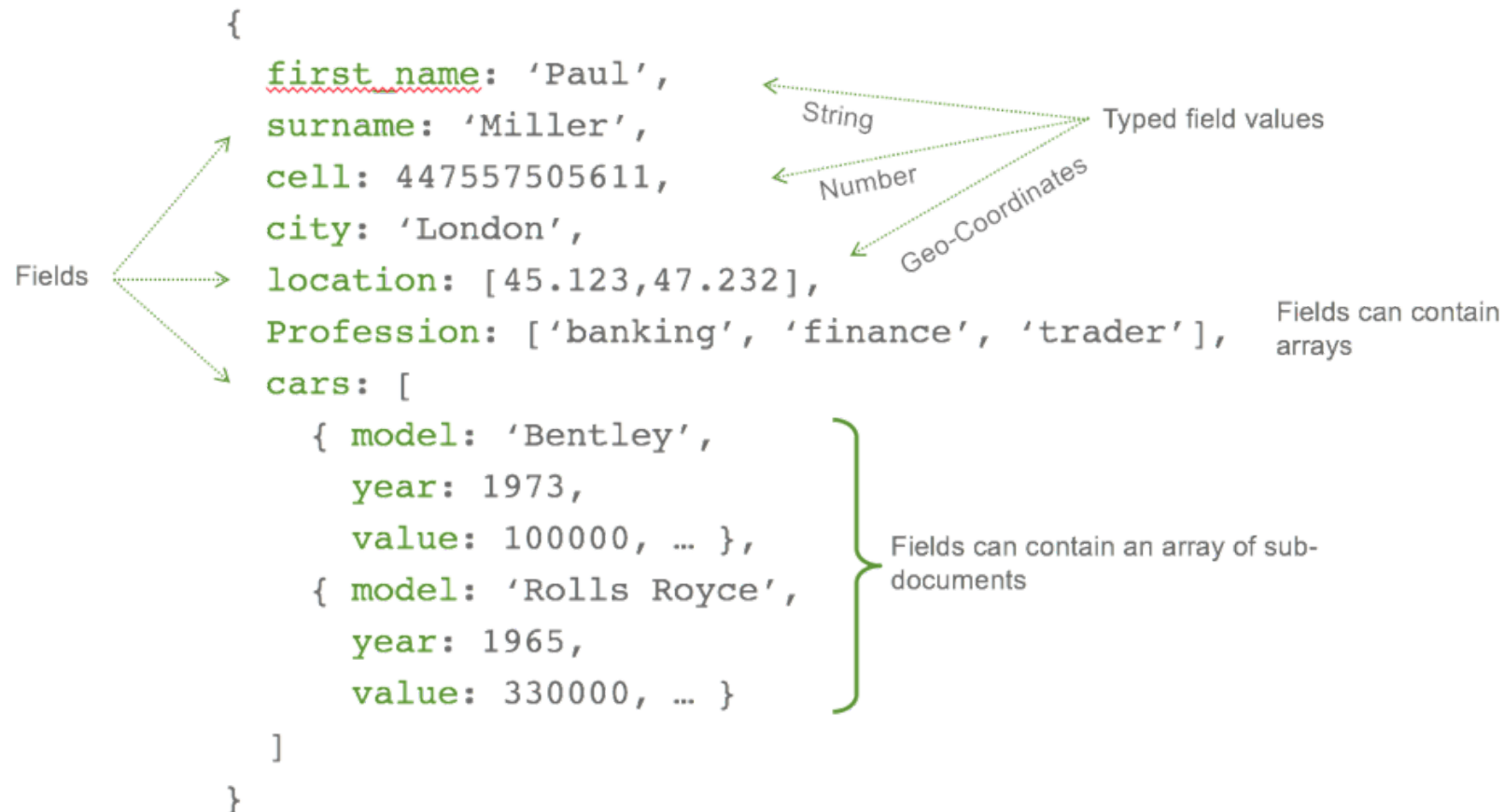
- Document based database
- Concept mapping:

SQL Terms/Concepts	MongoDB Terms/Concepts
database	database
table	collection
row	document or BSON document
column	field

- Each document is constructed as a **BSON** (Binary JSON)

MongoDB documents

A document looks like this:



From datawow.io

MongoDB Example

See `mongodb-demo.rmd`

- We will see the replication of SQL Basics in the last week using MongoDB
- For a simple selection of documents (i.e. rows in SQL), we will use `find()` method
- For a bit more sophisticated query, we will use `aggregate()` method
- Search query is in **BSON**
- For your reference, we will see the equivalent syntax of SQL right above the MongoDB query

MongoDB: JOIN?

- Use \$lookup

```
dbMongo$aggregate([  
  { "$match": { "party": "Republican" } },  
  { "$sort": { "shares_count": -1 } },  
  { "$limit": 10 },  
  { "$lookup": {  
    "localField": "screen_name",  
    "from": "congress", "foreignField": "screen_name",  
    "as": "congress"  
  } }])
```

- This is close to

```
dbGetQuery(db, "SELECT posts.*, congress.*  
FROM posts JOIN congress ON congress.screen_name = posts.screen_name  
WHERE party = 'Republican'  
ORDER BY shares_count DESC LIMIT 10")
```

MongoDB: JOIN?

- This will work, but it is not as powerful as SQL's **JOIN**. In the end, "if you have relational data, use a relational (SQL) database!".

Managed services in the cloud

Services

Database Type	AWS	GCP	Azure
Managed RDS	Amazon RDS	Cloud SQL	Azure SQL
Data Warehousing	Redshift	Bigquery	Snowflake
NoSQL (simple key-value)	DynamoDB	BigTable	Azure Tables
NoSQL (document)	MongoDB on EC2	MongoDB on GCE	DocumentDB

Google Cloud Platform: Bigquery

- GCP's data warehousing
- Integration with other Google data storage solutions (Google Drive, Google Cloud Storage)
- Scalable: same SQL syntax for datasets of *any* size
- Easy to collaborate and export results
- Affordable pricing and cost control
- API access allows integration with R or python
- Excellent documentation

Bigquery pricing

Operation	Pricing	Details
Active storage	\$0.023 per GB	The first 10 GB is free each month. See Storage pricing for details.
Long-term storage	\$0.016 per GB	The first 10 GB is free each month. See Storage pricing for details.
Streaming Inserts	\$0.0126 per 200 MB	You are charged for rows that are successfully inserted. Individual rows are calculated using a 1 KB minimum size. See Streaming pricing for details.
Queries (analysis)	\$9.35 per TB	First 1 TB per month is free, see On-demand pricing for details. Flat-rate pricing is also available for high-volume customers.

Bigquery example

- `Bigquery-demo.rmd`

This week's lab

- SQL topics: JOINS and sub queries
- Problem Set 8 (Assessed Assignment #4)