

MY472 - Week 10

**NoSQL and Working with Online
Databases**

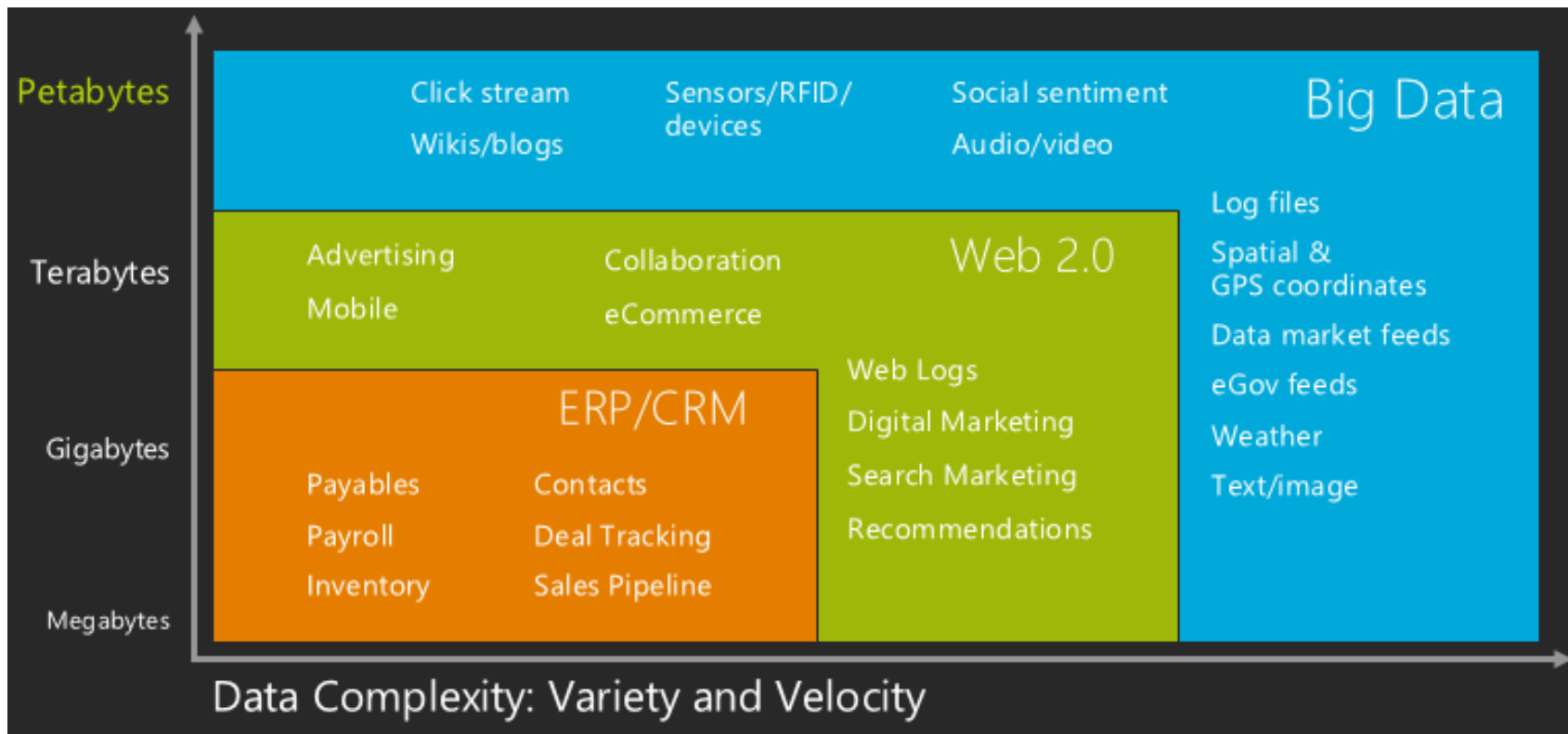
Outline

- Introduction
- SQL vs. noSQL
- Cloud solutions
- Coding session
 - Online database example with SQL: BigQuery
 - NoSQL example: MongoDB

Introduction

Big Data

- Today's data can be very large and also quite complicated



From: [Bigdata Dimension](#)

Database solutions for Big Data

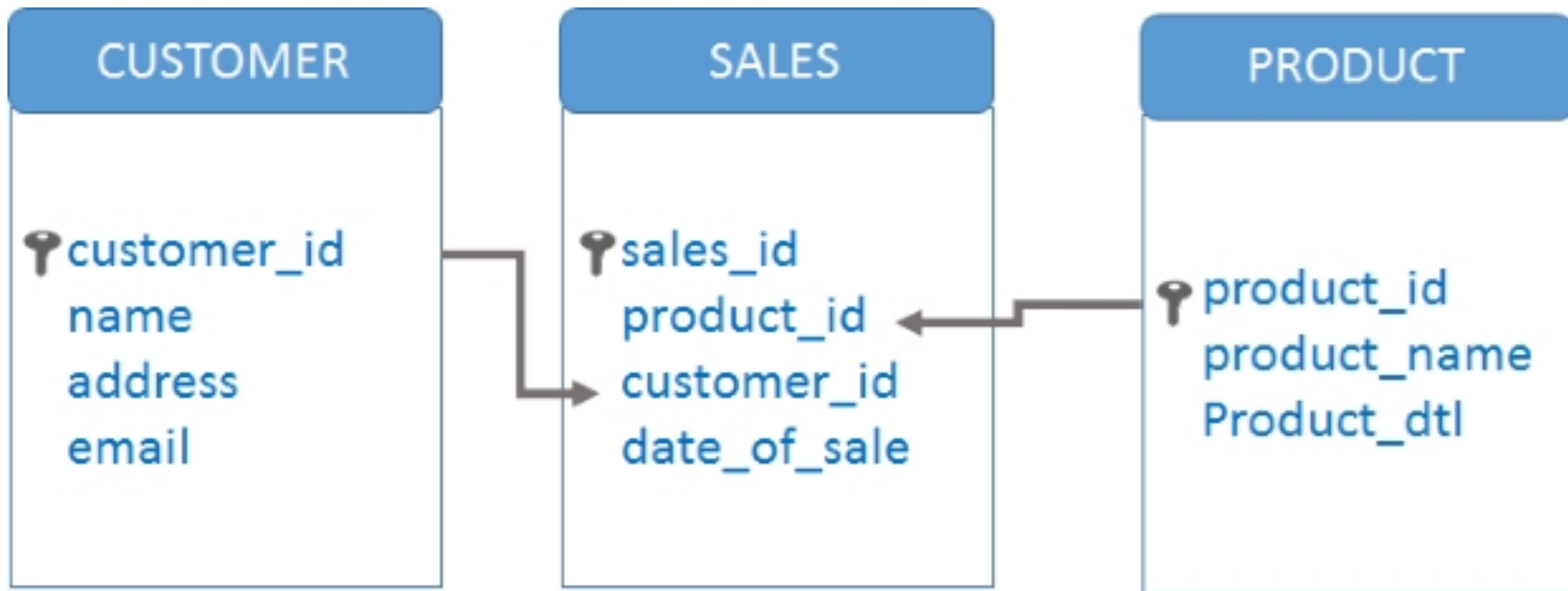
- Different types of databases (SQL vs. NoSQL)
- Cloud solutions using fully managed services

SQL or noSQL?

SQL

- Relational databases have a strict structure

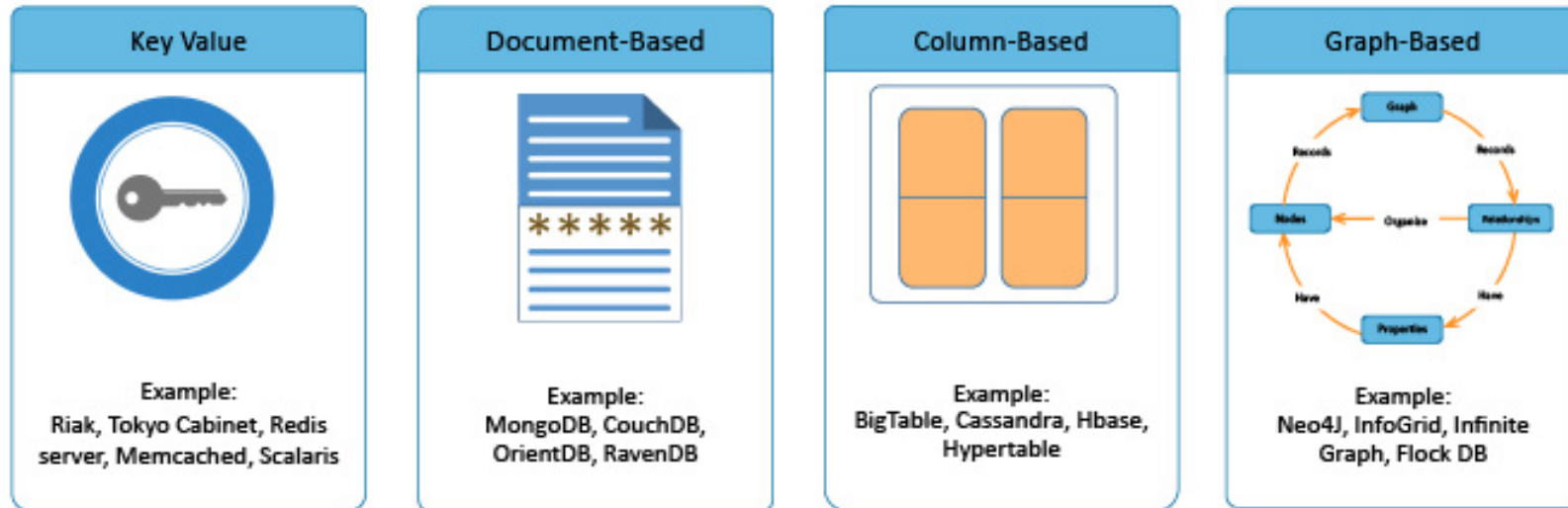
A simple e-commerce example:



noSQL

- Originally referring to “non SQL”, “non relational” or “not only SQL”
- Provides a mechanism for storage and retrieval of data which is modeled in means other than the tabular relations used in relational databases
- No strict structure/schema
- noSQL databases are good for data with
 - High **velocity** – Lots of data coming in very quickly
 - High **variety** – Data can be structured, semi-structured, and unstructured
 - High **volume** – Total size of data
 - High **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

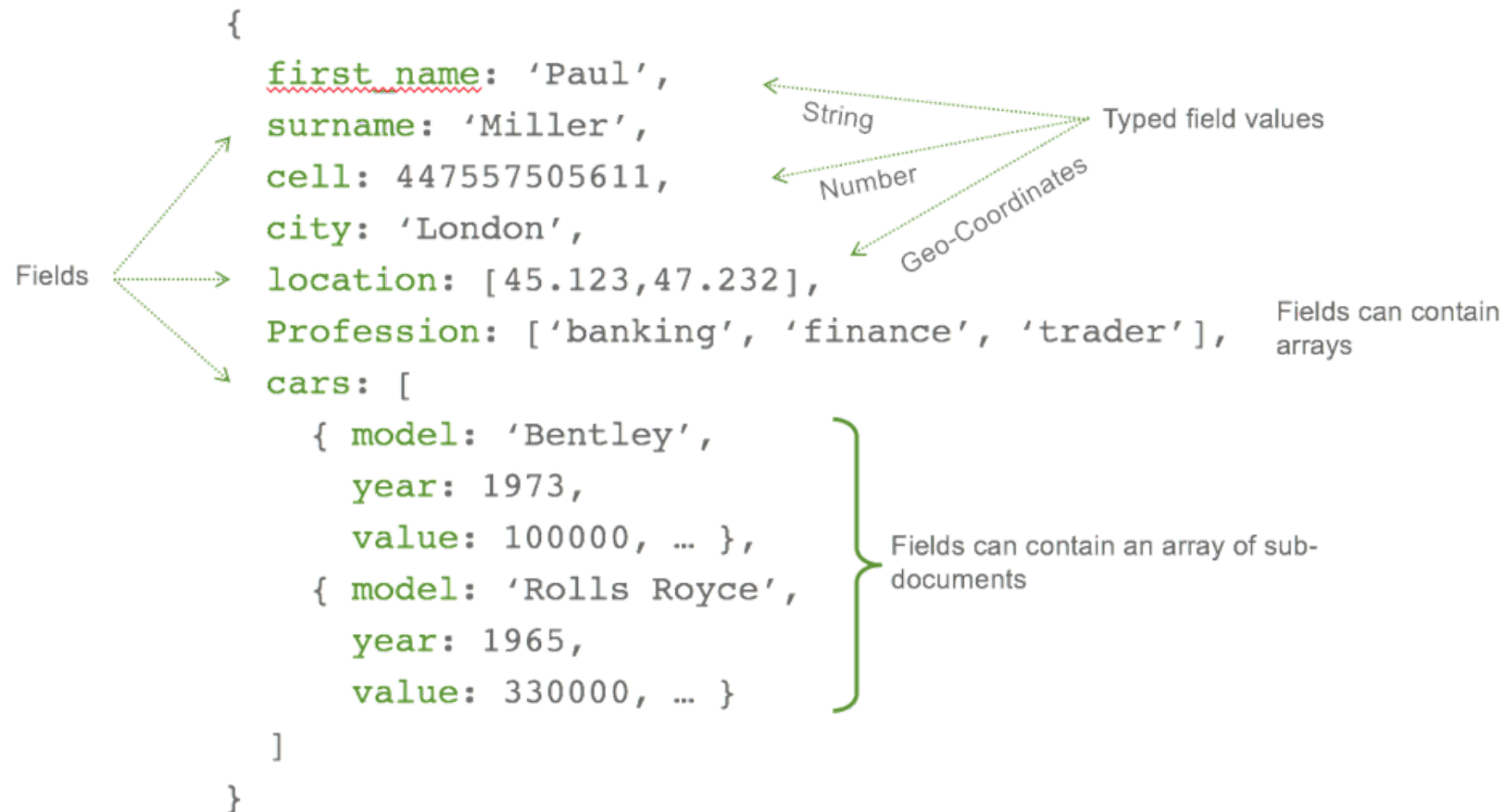
- Document-based database
- Mapping of concepts

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

- [Reference](#)
- Each document is constructed as a **BSON** ([Binary JSON](#))
- Not UTF-8 string encoded like JSON, but binary - machine readable
- Can store more data types: Dates, separate kinds of numerics (int, float, etc.)

MongoDB documents

A document looks like this:



From: datawow.io

MongoDB this week

- We will look at MongoDB as an example of a popular noSQL database this week
- We thereby try to replicate basic queries from last week using MongoDB via R with the package `mongolite`
- For a simple selection of documents (i.e. rows in SQL), we will use its `find()` method
- For a bit more sophisticated queries, we will use the `aggregate()` method
- Search queries are in JSON like notation
- Detailed [documentation](#) of MongoDB commands and operators
- [Resource 1 \(pdf\)](#) and [resource 2 \(website\)](#) for the R package `mongolite`

Cloud solutions

Some exemplary 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

- To create and query online databases, we will look at Google BigQuery's sandbox version as an example
- Database warehouse with other features, used by many financial and commercial companies
- Queried via SQL syntax (API access allows integration with R or Python)
- Scalable to very large databases
- Good [documentation](#)
- Many similar databases exist from other providers

Coding session

Files this week

- 01-bigquery-create-own-database.Rmd
- 02-bigquery-examples.Rmd
- 03-mongodb-demo.Rmd