Lecture 27 - A Brief Discussion of Non-Relational Databases

DSE 511

Drew Schmidt
2022-12-01

Announcements

- Schedule:
  - Dec 1 - more databases
  - Dec 6 - course wrapup
- New homework (last one)
  - Assigned now
  - Due Mon Dec 5
  - No homework on last modeule (databases)
- Questions?

What Is a Database?

- "A database is an organized collection of structured information, or data, typically stored electronically in a computer system" - Oracle
- Usually referring to a DBMS + its stored data
- Data is usually "tabular", but can form complicated hierarchies

Types of Databases

Relational (tables - SQL)

- MySQL
- PostgreSQL
- SQLite
- Oracle

Non-Relational ("NoSQL")

- MongoDB (document-oriented)
- Redis (key/value)
- Apache Cassandra (columnar)

Relational Databases

- We've seen one example (SQLite)
- They're all very similar

Pros

- Ubiquity
- Data integrity
- SQL is a quasi-standard
- Your favorite language has an interface

Cons

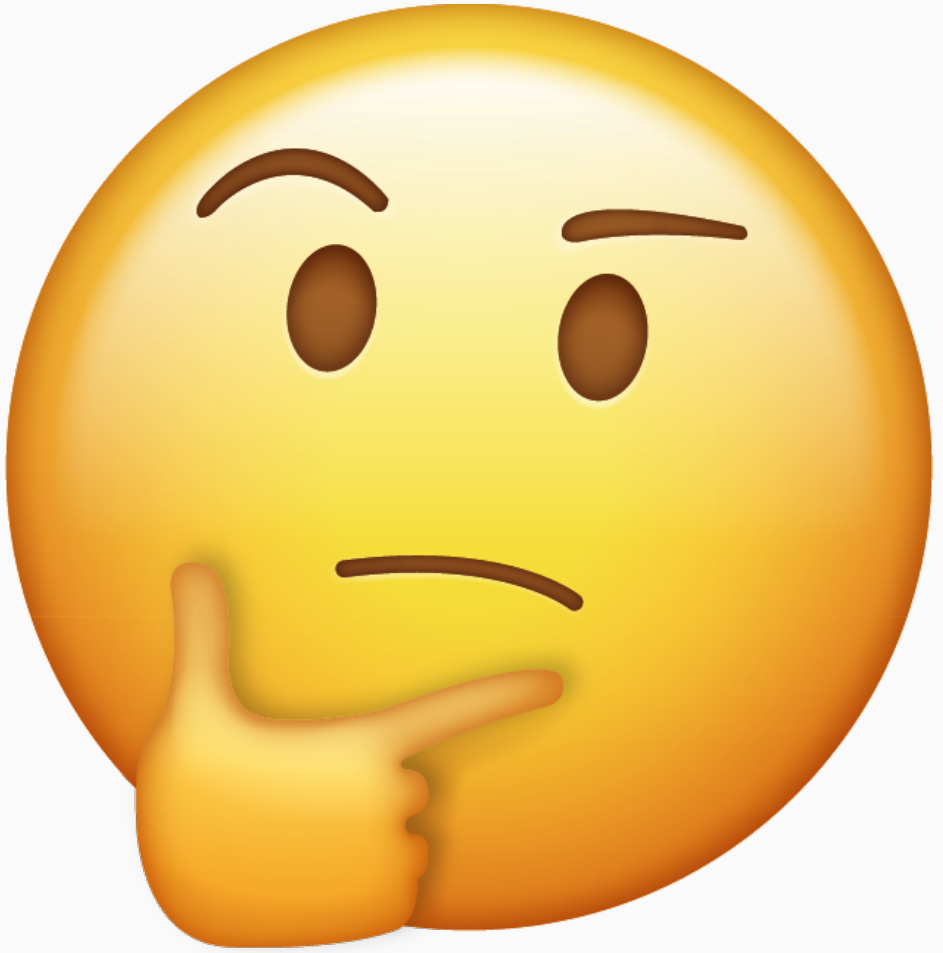
- No table flexibility
  - Can't change columns
  - Can't change types
  - ...
- Scalability
  - Usually no parallelism
  - Table complexity

A Question

Observation: Relational DB's can maintain complex data relationships ("complexity")

Question: If we just want to "store dataframes", could we use something else?

- Yes
- But you lose SQL (or may as well)
- And you *must* run a server

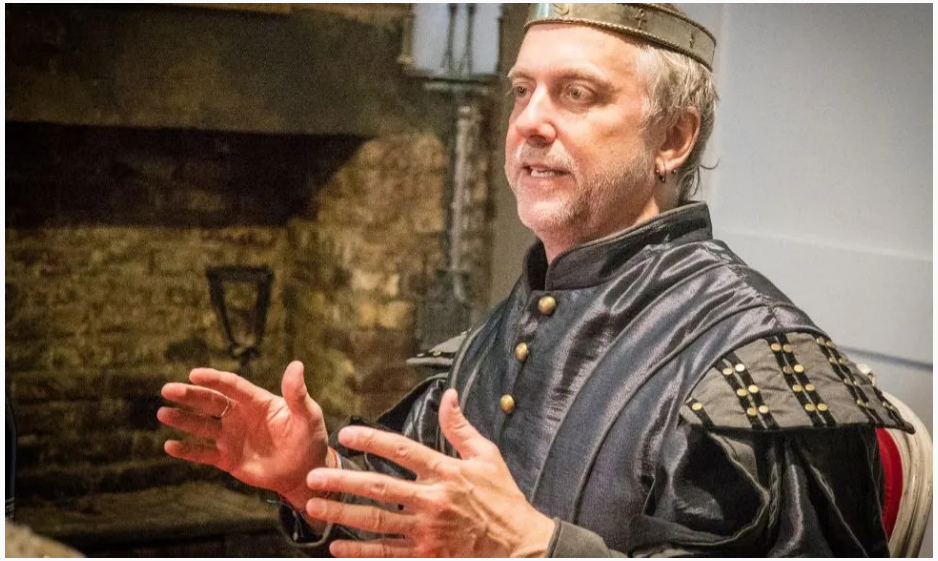


MongoDB

- The poster-child of NoSQL
- A document DB
  - Data is json-like
  - Fields can change
  - Types can vary
  - Data can change over time!
- Items in the DB are just key/value stores
- "Scales" (in the web sense - more next semester)
- Well-supported language bindings

A Note on Terminology

"Shards"



Questions?