Introduction to Database Systems **CSE 414** Lecture 12: NoSQL

CSE 414 - Spring 2018

Class Overview • Unit 1: Intro · Unit 2: Relational Data Models and Query Languages · Unit 3: Non-relational data - NoSQL Json - SQL++ Unit 4: RDMBS internals and query optimization · Unit 5: Parallel query processing · Unit 6: DBMS usability, conceptual design · Unit 7: Transactions 2

• Unit 8: Advanced topics (time permitting)



1

































Key-Value Stores Features

· Data model: (key,value) pairs

- Key = string/integer, unique for the entire data
- Value = can be anything (very complex object)
- Operations
 - get(key), put(key,value)
 - Operations on value not supported

Key-Value Stores Features

- Data model: (key,value) pairs
 - Key = string/integer, unique for the entire data
 - Value = can be anything (very complex object)
- Operations
 - get(key), put(key,value)
 - Operations on value not supported
- Distribution / Partitioning







Flights(fid, date, carrier, flight_num, origin, dest, ...) Carriers(cid, name)

Example

 How would you represent the Flights data as key, value pairs?

How does query processing work?

Flights(fid, date, carrier, flight_num, origin, dest, ...) Carriers(cid, name)

Example

- How would you represent the Flights data as key, value pairs?
- Option 1: key=fid, value=entire flight record

How does query processing work?

Flights(fid, date, carrier, flight_num, origin, dest, ...) Carriers(cid, name)

Example

- How would you represent the Flights data as key, value pairs?
- Option 1: key=fid, value=entire flight record
- Option 2: key=date, value=all flights that day

How does query processing work?

Flights(fid, date, carrier, flight_num, origin, dest, ...) Carriers(cid, name) Example • How would you represent the Flights data as key, value pairs? • Option 1: key=fid, value=entire flight record • Option 2: key=date, value=all flights that day • Option 3: key=(origin,dest), value=all flights between How does query processing work?









JSON - Overview

- JavaScript Object Notation = lightweight textbased open standard designed for humanreadable data interchange. Interfaces in C, C++, Java, Python, Perl, etc.
- The filename extension is .json.

We will emphasize JSon as semi-structured data

















