



॥ त्वं ज्ञानमयो विद्वानमयोऽसि ॥

#NoSQL: The Document Data Model

Saptarshi Pyne

Assistant Professor

Department of Computer Science and Engineering
Indian Institute of Technology Jodhpur, Rajasthan, India 342030

CSL4030 Data Engineering Lecture 9

August 21st, 2023

What we discussed in the last class

- Why do we need to normalize relations?
 - To avoid redundancies and **'anomalies'**
- How can we normalize relations?
 - By 'decomposing' relations into smaller relations while preserving the **'non-additive join'** property
- When do we need to **denormalize** relations?
 - When we have frequent 'join' queries

Today we will discuss

- The #NoSQL movement in 2009
 - Impedance mismatch
 - Push for open source software
- Polyglot persistence
- NoSQL databases
 - The document data model

The document data model

- Most suitable when
 - We have one-to-many relationships that can be represented as a tree
 - We need to load the whole tree together
- Not suitable when
 - We have many-to-many relationships

Schema flexibility in the document data model

- Schema-on-write
 - The relational data model
- Schema-on-read
 - The document data model

Storage Locality in the document data model

- XML documents
- JSON documents
- Binary documents, e.g., MongoDB's BSON

- The latest versions of many relational DBMS software have supports for XML and JSON documents

Examples of document-oriented DBMS software

- MongoDB by MongoDB Inc.
- CouchDB by Apache, an open-source alternative

Reference

- M. KLEPPMANN (2017), Designing Data-Intensive Applications The Big Ideas Behind Reliable, Scalable, and Maintainable Systems, O'Reilly.
 - Chapter 2. Data Models and Query Languages

In the next class, we will discuss

- NoSQL databases
 - The graph-like data model

Thank you