# Towards Benchmarking Multi-Model Databases

Jiaheng Lu
Department of Computer Science, University of Helsinki, Finland
jiaheng.lu@helsinki.fi

## 1. INTRODUCTION

As more businesses realized that data, in all forms and sizes, is critical to making the best possible decisions, we see the continued growth of systems that support massive volume of relational or non-relational forms of data. Unlike traditional database management systems which are organized around a single data model that determines how data can be organized, stored, and manipulated, a multi-model database is designed to support multiple data models against a single, integrated backend. For example, document, graph, relational, and key-value models are examples of data models that may be supported by a multi-model database. Nothing shows the picture more starkly than looking at Gartner Magic quadrant for operational database management systems, which assumes that, by 2017, all leading operational DBMSs will offer multiple data models, relational and NoSQL, in a single DBMS platform. Having a single data platform for managing both well-structured data and NoSQL data is beneficial to users; this approach reduces significantly integration, migration, development, maintenance, and operational issues.

Benchmarking is a common practice for the evaluation of the database systems, as more and more platforms are proposed to deal with multi-model data, it becomes important to have benchmarks that can be used to evaluate performance and usability of the next generation of multi-model database systems. A number of benchmarks have been proposed that can be used to evaluate big data systems (e.g. YCSB, BigBench, TPCx-BB, Bigframe). Unfortunately, those general-purpose big data benchmarks are not designed for the evaluation of multi-model databases. In this abstract, we argue that thorough evaluation of multi-model database systems imposes several new challenges that need to be overcome. First, the input and output of existing multi-model databases are quite diverse. Since there is no standard multi-model query language available now, publicly available implementations of benchmarking data and queries for different systems should be developed, shared, unified and optimized. Second, unlike the relation world, NoSQL systems follow "*data first, schema later or never*" paradigm. For a rigorous evaluation, it must be possible to control (and systematically vary) input schema and the complexity of a schema evolution for multi-model data. And the benchmark must promote productivity



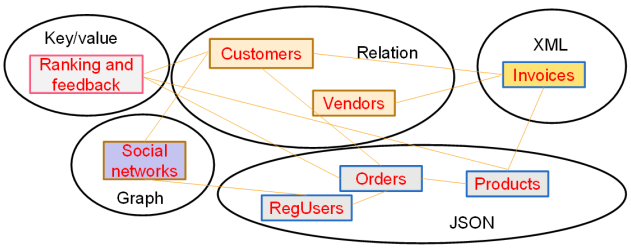


**Figure 1: Data models of UDBMS benchmark**

by enabling the creation of a large number of multi-model data with varied schema using little manual effort. Finally, multi-model databases are supposed to support the cross-model transaction and consistency. Therefore, novel consistency metrics which describe consistency behavior for different models of data must be proposed in a precise way.

## 2. OUR UDBMS BENCHMARK

To realize our vision we propose UDBMS-benchmark, a benchmark system for Unified DataBase Management Systems, with the following new features:

**Multi-model data**: A high level overview of the data model is presented in Figure 1, which includes relational data, XML data, graph data, JSON data and key-value data.

**Multi-model schema evolution**: UDBMS-benchmark automates the schema evolution process for multi-model data. The change of schema can affect the usability of history queries.

**Multi-model transaction and consistency**: One transaction can involve multiple data models. For example, an update of order information may affect JSON files (`Orders`, `Product`), key-value messages (`Feedback`) and XML data (`Invoice`). UDBMS-benchmark develops consistency metrics of ACID and eventual consistency with multi-model data and accurately determines consistency behavior via experiments with actually deployed systems.

**Multi-model data conversion**: An ideal multi-model database should support the model conversion between relation and NoSQL data. Therefore, data generators must support the creation of reasonable gold standard outputs for different transformation tasks.

To sum up, our proposed system involves novel features to give a comprehensive and rigorous evaluation for multi-model databases. Once completed, this benchmark will provide the community with a rich set of examples for multi-model data, query and transaction that can be studied to improve multi-model data management, as well as to ease the evaluation of diverse systems. This benchmark can be downloaded from `http://udbms.cs.helsinki.fi/bench/`