Multi-Model Data Query Languages and Processing Paradigms

Qingsong Guo
University of Helsinki
qingsong.guo@helsinki.fi

Jiaheng Lu
University of Helsinki
jiaheng.lu@helsinki.fi

Chao Zhang
University of Helsinki
chao.z.zhang@helsinki.fi

Calvin Sun
Huawei Canada
steven.yuan1@huawei.com

Steven Yuan
Huawei Canada
steven.yuan1@huawei.com

ABSTRACT

Specifying users' interests with a formal query language is a typically challenging task, which becomes even harder in the context of multi-model data management because we have to deal with data variety. It usually lacks a unified schema to help the users issuing their queries, or has an incomplete schema as data come from disparate sources. Multi-Model DataBases (MMDBs) have emerged as a promising approach for dealing with this task as they are capable of accommodating and querying the multi-model data in a single system. This tutorial aims to offer a comprehensive presentation of a wide range of query languages for MMDBs and to make comparisons of their properties from multiple perspectives. We will discuss the essence of cross-model query processing and provide insights on the research challenges and directions for future work. The tutorial will also offer the participants hands-on experience in applying MMDBs to issue multi-model data queries.

CCS CONCEPTS

• **Information systems** → **Data management systems**; **Query languages**; **Query languages for non-relational engines**.

KEYWORDS

data variety; multi-model database; multi-model data management; query language

ACM Reference Format:
Qingsong Guo, Jiaheng Lu, Chao Zhang, Calvin Sun, and Steven Yuan. 2020. Multi-Model Data Query Languages and Processing Paradigms. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20), October 19–23, 2020, Virtual Event, Ireland*. ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3340531.3412174

1 INTRODUCTION

In the past years, big data was overwhelming in both industry and research communities. A critical issue in big data management is to address the data variety. Data may be presented in various formats – structured, semi-structured, and unstructured – and produced by disparate sources, and hence natively have multiple models. The increasing availability of multi-model data has triggered the development of Multi-Model DataBase (MMDB) systems [8]. This is critical for many applications in which one needs to retrieve data across multiple models.

Take a healthcare data set Mimic II [11] as an example, which encompasses 26,000 patients/days in the intensive care unit (ICU) of Beth Israel Hospital in Boston. This data set includes data collected from disparate sources: (1) real-time data (time series from bedside monitoring devices); (2) a historical archive of waveform data (from previous patients); (3) patient metadata (relational data); (4) doctor's and nurse's notes (text); and (5) prescription information (semi-structured data). Relational data is only a small minority in this data set. If a doctor wants to know the historical treatments of his patients, she/he has to retrieve data from multiple sources.

MMDBs have emerged as a promising approach for dealing with this task. The traditional database systems were typically designed for a single data model. But many of them have recently evolved into multi-model versions. We have found 77 DBMSs listed on the DB-Engines Ranking site (334 DBMSs in total) are now supporting multi-model data. An MMDB typically integrates multiple (at least two) data stores together, so as to accommodate data in the formats that fit the sources best, e.g., key/value pairs, relational tables, graphs, or XML/JSON documents, etc. It also provides a unified query language, so one can store multi-model data in an MMDB and retrieve data of different models in a single query. The considerable research activity devoted to the field resulted in the development of dozens of multi-model query languages, where each concentrates on a set of specific data models.

Scope of the Tutorial. This tutorial is to offer a comprehensive investigation of a wide range of declarative query languages of MMDBs and to make a comparative study of their essential properties. The tutorial will also provide the participants with hands-on experience in multi-model queries over MMDBs. We will begin with the challenges from the "Variety" of big data and the essential issues in multi-model data management. In the past decades, a wealth of data models have been proposed for practical purposes. We will briefly discuss the major data models that are widely adopted by database systems, i.e., the relational model [4] and its extensions [5, 12, 13], graph model [10], and semi-structured model [1, 2]. After that, we will dive into the details of several popular multi-model data query languages such as AsterixDB's SQL++, Marklogic's XQuery, and ArangoDB's AQL [3, 14]. We will also make in-depth comparisons of these languages from four related aspects: (1) the essential semantic difference of these languages, (2) the expressive power defined what queries can be expressed with a given language, (3) the internal representation (e.g., the relational algebra) taken by each language, and (4) the processing paradigms

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
CIKM '20, October 19–23, 2020, Virtual Event, Ireland
© 2020 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-6859-9/20/10.
https://doi.org/10.1145/3340531.3412174

adopted by MMDBs. Finally, we will discuss the open problems in designing query languages for MMDBs and provide insights on the research challenges and directions for future work. In addition, during this tutorial, we will invite the participants to write and run some multi-model queries by using ArangoDB to provide them hands-on experience.

The slides of this tutorial can be downloaded at this site[1]. To the best of our knowledge, this is the first tutorial to discuss state-of-the-art research works and industrial trends on multi-model data query languages. Multi-model data management and MMDBs have attracted a lot of attention during the past decades. Three existing tutorials are related to this topic. The tutorial [7] discussed the general challenges and issues in multi-model data management, and the tutorial [9] compared the two solutions for managing multi-model data, i.e., MMDBs and integrated polystores. The tutorial [6] investigated the query languages and processing paradigms for graph data. In this tutorial, we will not concentrate on the query languages for a single data model and their processing paradigms in this tutorial, which were not surveyed by previous tutorials.

2 TUTORIAL ORGANIZATION

The tutorial is planned for 6 hours and is divided into 6 parts:

Part I: Introduction(15 minutes)

We start the tutorial by introducing data variety and motivating the need of multi-model data management.

- Basics on data variety
- The need and essence of multi-model data management

Part II: Data models (45 minutes)

We will briefly discuss the major data models adopted by database systems and a benchmark for multi-model data.

- The relational model and its extensions
- The semi-structured data models, e.g. XML and JSON
- The graph data models

Part III: Multi-model data query languages (60 minutes)

We will discuss several well-known multi-model data query languages, which fall into three categories.

- The SQL-extensions
- The XML/JSON-extensions
- The graph-extensions

Part IV: Comparison of the query languages (60 minutes)

We will make a comparative study of the query languages from 4 perspectives.

- The semantic difference
- The expressive power
- The internal representation
- The manner of query evaluation

Part V: Open problem and challenges (30 minutes)

We will conclude with a discussion of open problems and challenges in designing multi-model data query languages.

- An algebra for a multi-model query language.
- General approaches for cross-model query processing.

Part VI: Hands-on experience (150 minutes)

We will invite the participants to write and run some multi-model queries by using ArangoDB.

- Generate an E-commence dataset with Unibench [15, 16]
- Hands-on experience for multi-model queries with ArangoDB.

[1] https://www.helsinki.fi/en/node/93817

3 SHORT BIBLIOGRAPHIES

Qingsong Guo is a Postdoctoral Researcher at the University of Helsinki, Finland. His current research interests include multi-model data management and automatic management of big data with deep learning algorithms.

Jiaheng Lu is an Associate Professor at the University of Helsinki. His main research interests lie in the Big Data management and database systems. He has published more than one hundred journal and conference papers. He has published several books on XML, Hadoop and NoSQL databases.

Chao Zhang is a Ph.D. candidate at the University of Helsinki. His research topic is on multi-model database benchmarking and cross-model query optimization.

Calvin Sun is the Chief Database Architect at Huawei Cloud. He has 20+ years working experience in the development of several database systems, ranging from embedded database, large-scale distributed database, to cloud-native database.

Steven Yuan is the director of Huawei Toronto Distributed Scheduling and Data Engine Lab. He leads an over 30 people research team in big data and cloud domain.

REFERENCES

[1] ECMA-404 The JSON Data Interchange Standard. https://www.json.org/json-en.html.

[2] Extensible Markup Language (XML). https://www.w3.org/XML/.

[3] R. Angles, M. Arenas, P. Barceló, A. Hogan, J. L. Reutter, and D. Vrgoc. Foundations of modern query languages for graph databases. *ACM Comput. Surv.*, 50(5):68:1–68:40, 2017.

[4] E. F. Codd. A relational model of data for large shared data banks. *Commun. ACM*, 13(6):377–387, 1970.

[5] E. F. Codd. Extending the database relational model to capture more meaning. *ACM Trans. Database Syst.*, 4(4):397–434, Dec. 1979.

[6] A. Deutsch and Y. Papakonstantinou. Graph data models, query languages and programming paradigms. *Proc. VLDB Endow.*, 11(12):2106–2109, 2018.

[7] J. Lu and I. Holubová. Multi-model data management: What's new and what's next? In *Proceedings of the 20th International Conference on Extending Database Technology, EDBT 2017, Venice, Italy, March 21-24, 2017*, pages 602–605. OpenProceedings.org, 2017.

[8] J. Lu and I. Holubová. Multi-model Databases: A new journey to handle the variety of data. *ACM Computing Surveys*, 52(3), 2019.

[9] J. Lu, I. Holubová, and B. Cautis. Multi-model databases and tightly integrated polystores: Current practices, comparisons, and open challenges. In *CIKM '18*, pages 2301–2302, New York, NY, USA, 2018. ACM.

[10] I. Robinson, J. Webber, and E. Eifrem. *Graph Databases: New Opportunities for Connected Data.* O'Reilly Media, Inc., 2nd edition, 2015.

[11] M. Saeed, M. Villarroel, A. Reisner, G. Clifford, L.-w. Lehman, G. Moody, T. Heldt, T. Kyaw, B. Moody, and R. Mark. Multiparameter Intelligent Monitoring in Intensive Care II (Mimic-II): A Public-Access Intensive Care Unit Database. *Critical care medicine*, 39:952–60, 05 2011.

[12] M. H. Scholl. Extensions to the Relational Data Model. In *Conceptual Modelling, Databases and CASE: An Integrated View of Information Systems Development.* Jon. Wiley & Sons, 1992.

[13] M. H. Scholl, H. Paul, and H. Schek. Supporting flat relations by a nested relational kernel. In *VLDB'87, September 1-4, 1987, Brighton, England*, pages 137–146. Morgan Kaufmann, 1987.

[14] P. T. Wood. Query languages for graph databases. *SIGMOD Rec.*, 41(1):50–60, 2012.

[15] C. Zhang and J. Lu. Holistic evaluation in multi-model databases benchmarking. *Distributed and Parallel Databases*, pages 1–33, 2019.

[16] C. Zhang, J. Lu, P. Xu, and Y. Chen. UniBench: A Benchmark for Multi-model Database Management Systems. In *TPCTC '18, Rio de Janeiro, Brazil, August 27-31, 2018, Revised Selected Papers*, volume 11135 of *Lecture Notes in Computer Science*, pages 7–23. Springer, 2018.