Master Thesis
Efficiently executing data graph queries in SQL

Prof. Dr. Burkhard Freitag
Chair of Information Management, University of Passau
http://www.im.uni-passau.de/

Tutor: Matthias Schmid (Matthias.Schmid@uni-passau.de)

March 21, 2019

Keywords: Database Systems, Graph Databases, Graph Queries, *Cypher*, Query Transformation

1 Introduction

Data graphs are of increasing importance in modern information systems. Therefore, an efficient method that allows to store graph data in a database is highly desirable. In [6] the *SQLGraph* approach to storing data graphs in a relational database has been proposed, which can be proven to be reasonably efficient on average even if compared to native graph database systems.

However, using the *SQLGraph* approach and the corresponding database schema, queries cannot be formulated in a graph-based language and therefore tend to be rather complex. On the other hand, the graph query language *Cypher* [4, 3], which has originally been developed for native graph databases, provides a convenient abstraction level for asking queries against data graphs.

In an ongoing research project at the Chair of Information Management, a first attempt at providing a graph-based query language, which is based on a transformation of *Cypher* into SQL-queries, has been investigated and prototypically implemented. However, the efficiency of the evaluation of the transformed query depends on the way the graph query is formulated. This is partly a consequence of the fact that in the transformed queries many Common Table Expressions (CTE) occur, for which the database optimizer has only very limited capacity.

2 Objectives

In this Master thesis the performance achievable for the transformed queries is to be analyzed and reasons for good or bad performance are to be found. The results of the analysis should be verified using the LDBC social network benchmark [1, 2, 5, 7]. Based on the confirmed results of the performance analysis, the query transformer is to be optimized. The overall objective is the **development of a *Cypher* query transformer that delivers SQL queries which can be evaluated with good average performance** on a relational database system.

The thesis consists of a conceptual part and an implementation part. The implementation shall be based on an existing prototypical implementation of the *Cypher* query transformer and the PostgreSQL database management system.

3 Tasks

Among the subtasks that have to be performed are:

- Study the literature (see below); study the existing system prototype;
- Analyze the performance of the evaluation of the transformed queries; the analysis is to be based on the LDBC benchmark;
- Determine the reasons for good and bad performance of transformed queries;
- Develop an optimized query transformer which avoids the deficiencies of the existing prototype;
- Implement the optimized query transformer;
- Confirm the correctness and efficiency of the proposed transformer using the LDBC benchmark;
- Documentat the proposed solution;
- Demonstrate the proposed solution.

References

[1] P. A. Boncz. LDBC: benchmarks for graph and RDF data management. In *17th International Database Engineering & Applications Symposium, IDEAS '13, Barcelona, Spain - October 09 - 11, 2013*, pages 1–2, 2013.

[2] O. Erling, A. Averbuch, J. Larriba-Pey, H. Chafi, A. Gubichev, A. Prat-Pérez, M. Pham, and P. A. Boncz. The LDBC social network benchmark: Interactive workload. In *Proc. of the 2015 ACM SIGMOD Intl. Conf. on Management of Data, Melbourne, Victoria, Australia, 2015*, pages 619–630, 2015.

[3] N. Francis, A. Green, P. Guagliardo, L. Libkin, T. Lindaaker, V. Marsault, S. Plantikow, M. Rydberg, M. Schuster, P. Selmer, and A. Taylor. Formal semantics of the language cypher. *CoRR*, abs/1802.09984, 2018.

[4] N. Francis, A. Green, P. Guagliardo, L. Libkin, T. Lindaaker, V. Marsault, S. Plantikow, M. Rydberg, P. Selmer, and A. Taylor. Cypher: An evolving query language for property graphs. In *Proceedings of the 2018 International Conference on Management of Data, SIGMOD Conference 2018, Houston, TX, USA, June 10-15, 2018*, pages 1433–1445, 2018.

[5] A. Iosup, T. Hegeman, W. L. Ngai, S. Heldens, A. Prat-Pérez, T. Manhardt, H. Chafi, M. Capota, N. Sundaram, M. J. Anderson, I. G. Tanase, Y. Xia, L. Nai, and P. A. Boncz. LDBC graphalytics: A benchmark for large-scale graph analysis on parallel and distributed platforms. *PVLDB*, 9(13):1317–1328, 2016.

[6] W. Sun, A. Fokoue, K. Srinivas, A. Kementsietsidis, G. Hu, and G. T. Xie. SQLGraph: an efficient relational-based property graph store. In *Proc. of the 2015 ACM SIGMOD Intl. Conf. on Management of Data, Melbourne, Victoria, Australia, 2015*, pages 1887–1901, 2015.

[7] G. Szárnyas, A. Prat-Pérez, A. Averbuch, J. Marton, M. Paradies, M. Kaufmann, O. Erling, P. A. Boncz, V. Haprian, and J. B. Antal. An early look at the LDBC social network benchmark's business intelligence workload. In *Proceedings of the 1st ACM SIGMOD Joint International Workshop on Graph Data Management Experiences & Systems (GRADES) and Network Data Analytics (NDA), Houston, TX, USA, June 10, 2018*, pages 9:1–9:11, 2018.