Compound Analytics of Compound Data within RDBMS Framework – Infobright's Perspective

Dominik Ślęzak[1,2]

[1] Institute of Mathematics, University of Warsaw Banacha 2, 02-097 Warsaw, Poland
[2] Infobright Inc., Krzywickiego 34 lok. 219, 02-078 Warsaw, Poland
slezak@infobright.com

The relational model has been present in research and applications for decades, inspiring a number of RDBMS products based on entirely different architectures, but sharing the same way of understanding and representing the data [4]. Given 40 years of history, it is clear that the relational paradigms should not be blindly followed in all situations [1]. On the other hand, given its popularity, the relational framework is usually the easiest one to accept by database users and the most convenient for interfacing with other tools.

An important trend in database industry relates to *analytical engines* that are optimized for advanced reporting and ad hoc querying. Such engines are usually applied at the level of data marts, especially in market segments where rapid data growth is expected. Originally, they have been technically complex and difficult to maintain. However, they have evolved toward solutions such as, e.g., Infobright's Community/Enterprise Editions (ICE/IEE)[1], capable of handling tens of terabytes of data on a single off-the-shelf box [15].

Infobright's engine is a fully functional RDBMS product with external connectors provided via integration with MySQL, and internals based on columnar storage [8], adaptive compression [17], as well as compact *rough* information that replaces standard database indexes [13]. We refer, e.g., to [3,10,16] for current research on ICE/IEE core technology, and to [2,7,12] for several interesting examples of its usage in academic and commercial projects.

In this talk, we use Infobright's software as a baseline to discuss limitations of relational model with respect to modern database applications. In particular, we investigate some challenges related to *compound analytics* and *compound data.* In both cases, we claim that it would be a mistake to give up too quickly the benefits of a typical RDBMS way of interacting with users. Instead, we present some application-level and technology-level solutions that do not contradict with original relational framework's universality and simplicity.

With regards to compound analytics, as an example, we consider practical inspirations and opportunities for enriching standard SQL language with approximate aspects [5,11], assuming minimum impact on query syntax and maximum easiness of interpreting inexact query answers.

With regards to compound data, we discuss two general approaches to employ domain knowledge about data semantics in order to improve database efficiency:

[1] www.infobright.{org.com}

T.-h. Kim et al. (Eds.): FGIT 2010, LNCS 6485, pp. 39–40, 2010.

1) expressing data hierarchies explicitly at data schema level (see e.g. [6,12]), or 2) doing it independently from both logical and physical modeling layers, taking into account that domain experts may need interfaces other than those designed for database end-users and administrators (see e.g. [9,14]).

References

1. Agrawal, R., et al.: The Claremont report on database research. SIGMOD Rec. 37(3), 9–19 (2008)
2. Apanowicz, C.: Data Warehouse Discovery Framework: The Case Study. In: Zhang, Y., et al. (eds.) DTA/BSBT 2010. CCIS, vol. 118, pp. 159–170. Springer, Heidelberg (2010)
3. Borkowski, J.: Performance debugging of parallel compression on multicore machines. In: Wyrzykowski, R., et al. (eds.) PPAM 2009. LNCS, vol. 6068, pp. 82–91. Springer, Heidelberg (2010)
4. Codd, E.F.: Derivability, redundancy and consistency of relations stored in large data banks. SIGMOD Rec. 38(1), 17–36 (2009) (Originally: IBM Research Report RJ599, 1969)
5. Cuzzocrea, A.: OLAP Data Cube Compression Techniques: A Ten-Year-Long History. In: Kim, T.-h., et al. (eds.) FGIT 2010. LNCS, vol. 6485, pp. 751–754. Springer, Heidelberg (2010)
6. Das, S., Chong, E.I., Eadon, G., Srinivasan, J.: Supporting ontology-based semantic matching in RDBMS. In: Proc. of VLDB 2004, pp. 1054–1065. Morgan Kaufmann, San Francisco (2004)
7. Frutuoso Barroso, A.R., Baiden, G., Johnson, J.: Knowledge Representation and Expert Systems for Mineral Processing Using Infobright. In: Proc. of GRC 2010, pp. 49–54. IEEE, Los Alamitos (2010)
8. Hellerstein, J.M., Stonebraker, M., Hamilton, J.R.: Architecture of a Database System. Foundations and Trends in Databases 1(2), 141–259 (2007)
9. Moss, L.T., Atre, S.: Business Intelligence Roadmap: The Complete Project Lifecycle for Decision-Support Applications. Addison-Wesley, Reading (2003)
10. Ślęzak, D., Kowalski, M.: Intelligent Data Granulation on Load: Improving Infobright's Knowledge Grid. In: Lee, Y.-h., et al. (eds.) FGIT 2009. LNCS, vol. 5899, pp. 12–25. Springer, Heidelberg (2009)
11. Ślęzak, D., Kowalski, M.: Towards Approximate SQL – Infobright's Approach. In: Szczuka, M., et al. (eds.) RSCTC 2010. LNCS, vol. 6086, pp. 630–639. Springer, Heidelberg (2010)
12. Ślęzak, D., Sosnowski, Ł.: SQL-Based Compound Object Comparators – A Case Study of Images Stored in ICE. In: Kim, T.-h., et al. (eds.) ASEA 2010. CCIS, vol. 117, pp. 304–317. Springer, Heidelberg (2010)
13. Ślęzak, D., Synak, P., Wróblewski, J., Toppin, G.: Infobright – Analytic Database Engine using Rough Sets and Granular Computing. In: GRC 2010, pp. 432–437. IEEE, Los Alamitos (2010)
14. Ślęzak, D., Toppin, G.: Injecting Domain Knowledge into a Granular Database Engine – A Position Paper. In: Proc. of CIKM 2010, pp. 1913–1916. ACM, New York (2010)
15. Ślęzak, D., Wróblewski, J., Eastwood, V., Synak, P.: Brighthouse: An Analytic Data Warehouse for Ad-hoc Queries. PVLDB 1(2), 1337–1345 (2008)
16. Synak, P.: Rough Set Approach to Optimisation of Subquery Execution in Infobright Data Warehouse. In: Proc. of SCKT 2008. PRICAI 2008 Workshop (2008)
17. Wojnarski, M., et al.: Method and System for Data Compression in a Relational Database. US Patent Application 2008/0071818 A1 (2008)