In-Memory for the Masses: Enabling Cost-Efficient Deployments of In-Memory Data Management Platforms for Business Applications

Alexander Boehm
SAP SE
Walldorf, Germany
alexander.boehm@sap.com

ABSTRACT

With unrivaled performance, modern in-memory data management platforms such as SAP HANA [5] enable the creation of novel types of business applications. By keeping all data in memory, applications may combine both demanding transactional as well as complex analytical workloads in the context of a single system. While this excellent performance, data freshness, and flexibility gain is highly desirable in a vast range of modern business applications [6], the corresponding large appetite for main memory has significant implications on server sizing. Particularly, hardware costs on premise as well as in the cloud are at risk to increase significantly, driven by the high amount of DRAM that needs to be provisioned potentially.

In this talk, we discuss a variety of challenges and opportunities that arise when running business applications in a cost-efficient manner on in-memory database systems.

PVLDB Reference Format:
Alexander Boehm. In-Memory for the Masses: Enabling Cost-Efficient Deployments of In-Memory Data Management Platforms for Business Applications. *PVLDB*, 12(12): 2273-2274, 2019.
DOI: https://doi.org/10.14778/3352063.3352142

1. INTRODUCTION

Conceptually, we can group the challenges of running business applications in a cost-efficient manner on in-memory database systems into the three different areas discussed below.

1.1 Database Architecture

Looking at the database architecture, there are multiple options that can help to reduce the memory footprint and processing requirements of an in-memory system. First and foremost, data compression [4] reduces the DRAM required for data storage, while fine-grained pipelined query execution reduces the intermediate memory for query processing. We discuss several design choices taken by the HANA system which primarily relies on columnar data storage and dictionary-compression.

1.2 Software / Hardware Codesign

Recent advances in hardware provide new, exciting opportunities and design choices for in-memory systems. In particular, the availability of storage class memory makes it feasible to keep more data at cheaper costs in memory, with some impact on the write frequency and performance. Further, very fast storage systems such as NVM-E allow to offload storage to even cheaper media, with a high impact on the overall TCO, but only limited impact to performance. We discuss some architectural choices taken by the HANA system, and how they can help to leverage modern storage systems in an efficient way [1, 7, 8]. We also highlight some important questions on how to identify data access pattern, and derive appropriate partitioning criteria and data placement strategies.

1.3 Database / Application Codesign

Many applications still treat database management systems as more or less 'dumb' storage systems that are accessed using convenient, application-focused abstraction layers such as Object-Relational Mappers (ORM). While this simplistic view helps to reduce the complexity of the overall architecture, it at the same time reduces the potential for cross-stack optimizations that span both application and database [2]. This leads to suboptimal performance and feature redundancy (e.g. by processing joins and aggregation operations on the application server in lack of efficient push-down possibilities). Similar inefficiencies arise from a TCO perspective when treating an in-memory database system as a black box: Applications end up storing way too much data in-memory, leading to both high DRAM costs as well as issues dealing with data privacy regulations [3]. Another source of inefficiencies are suboptimal physical schema designs that inhibit cost-efficient storage and processing by the database system. We discuss several examples from database/application codesign projects and share the results that could be achieved.

This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License. To view a copy of this license, visit http://creativecommons.org/licenses/by-nc-nd/4.0/. For any use beyond those covered by this license, obtain permission by emailing info@vldb.org. Copyright is held by the owner/author(s). Publication rights licensed to the VLDB Endowment.
Proceedings of the VLDB Endowment, Vol. 12, No. 12
ISSN 2150-8097.
DOI: https://doi.org/10.14778/3352063.3352142

2. REFERENCES

[1] M. Andrei, C. Lemke, G. Radestock, R. Schulze, C. Thiel, R. Blanco, A. Meghlan, M. Sharique, S. Seifert, S. Vishnoi, D. Booss, T. Peh, I. Schreter, W. Thesing, M. Wagle, and T. Willhalm. SAP HANA

adoption of non-volatile memory. *PVLDB*, 10(12):1754–1765, 2017.

[2] A. Böhm. Novel optimization techniques for modern database environments. In *Datenbanksysteme für Business, Technologie und Web (BTW), 16. Fachtagung des GI-Fachbereichs "Datenbanken und Informationssysteme" (DBIS), 4.-6.3.2015 in Hamburg, Germany. Proceedings*, pages 23–24, 2015.

[3] S. Kessler, J. Hoff, and J.-C. Freytag. SAP HANA goes private - from privacy research to privacy aware enterprise analytics. In *Proceedings of the 45th International Conference on Very Large Data Bases, Los Angeles, California, August 26-30, 2019, to appear*, 2019.

[4] C. Lemke, K. Sattler, F. Faerber, and A. Zeier. Speeding up queries in column stores - A case for compression. In *Data Warehousing and Knowledge Discovery, 12th International Conference, DAWAK 2010, Bilbao, Spain, August/September 2010. Proceedings*, pages 117–129, 2010.

[5] N. May, A. Böhm, and W. Lehner. SAP HANA - the evolution of an in-memory DBMS from pure OLAP processing towards mixed workloads. In *Datenbanksysteme für Business, Technologie und Web (BTW 2017), 17. Fachtagung des GI-Fachbereichs ,,Datenbanken und Informationssysteme" (DBIS), 6.-10. März 2017, Stuttgart, Germany, Proceedings*, pages 545–563, 2017.

[6] H. Plattner. A common database approach for OLTP and OLAP using an in-memory column database. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2009, Providence, Rhode Island, USA, June 29 - July 2, 2009*, pages 1–2, 2009.

[7] R. Sherkat, C. Florendo, M. Andrei, and R. Blanco. Native store extension for SAP HANA. In *Proceedings of the 45th International Conference on Very Large Data Bases, Los Angeles, California, August 26-30, 2019, to appear*, 2019.

[8] R. Sherkat, C. Florendo, M. Andrei, A. K. Goel, A. Nica, P. Bumbulis, I. Schreter, G. Radestock, C. Bensberg, D. Booss, and H. Gerwens. Page as you go: Piecewise columnar access in SAP HANA. In *Proceedings of the 2016 International Conference on Management of Data, SIGMOD Conference 2016, San Francisco, CA, USA, June 26 - July 01, 2016*, pages 1295–1306, 2016.