Novel Optimization Techniques for Modern Database Environments

Alexander Böhm
alexander.boehm@sap.com

1 Abstract

Performance and scalability have always been major aspects of the design and implementation of database management systems (DBMS). Consequentially, there is a vast number of publications addressing performance improvements for virtually every aspect of a DBMS, such as query processing, restart recovery, physical data storage, index structures, or buffer management, to only name a few.

In recent years, due to the evolution of hardware and its new capabilities (e.g. vector processing or hardware transactional memory), the topic of hardware/software co-design has gained much attention. Both DBMS researchers and developers have broadened their scope from a pure (system) software programming perspective towards a more hardware-aware view.

In this keynote, we will complement this trend of looking towards the hardware by considering another part of the usual software stack a DBMS is deployed in, and focus towards the application server running the user code. While the interaction and potential synergies between application server and DBMS are often neglected, we will discuss a multitude of optimization opportunities that can be addressed. Examples include, but are not limited to:

- Efficient data transfer between application server and DBMS
- Overhead reduction by data type and memory layout alignment
- Request batching (in the spirit of group commit)
- Code relocation (push-down and pull-up)

We discuss how these techniques allow to improve the overall efficiency of the software stack running the end user’s applications and optimize their perceived performance and experience.
2 Biography

Alexander Böhm is a database architect working on SAP’s HANA in-memory database management system with a strong focus on performance optimization. Prior to joining SAP, he received his PhD from the University of Mannheim, Germany where he worked on the development of efficient and scalable applications using declarative message processing.