Processing-in-Memory Primitives for Data Management (PIMPMe)

Kai-Uwe Sattler

SPP 2377

RESEARCH QUESTION:

Which data management primitives can be offloaded to memory or processed near specific levels of the memory hierarchy?

PROCESSING NEAR MEMORY

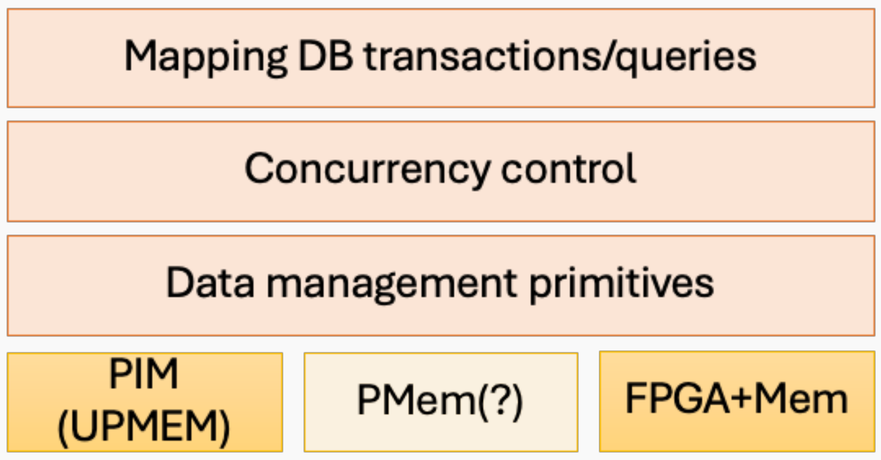
- bringing processing close to memory or storage
- use case: accelerator cards (GPU, FPGA) with device memory
- transfer between host and device memory still needed
- operations of arbitrary complexity

PROCESSING IN MEMORY

- execute operations directly in memory (on the same die)
- without moving data from DRAM
- limited capability (clock rate, operation complexity)

OBJECTIVES

- identifying and prototyping data management primitives for offloading to memory
- using HTM and related technologies for thread-safe data structures and MVCC implementation
- evaluation of cost-based placement (CPU/DRAM vs. PIM)
- effect of offloading in end-to-end scenarios (database queries)
- translate graph queries to programs utilising PIM



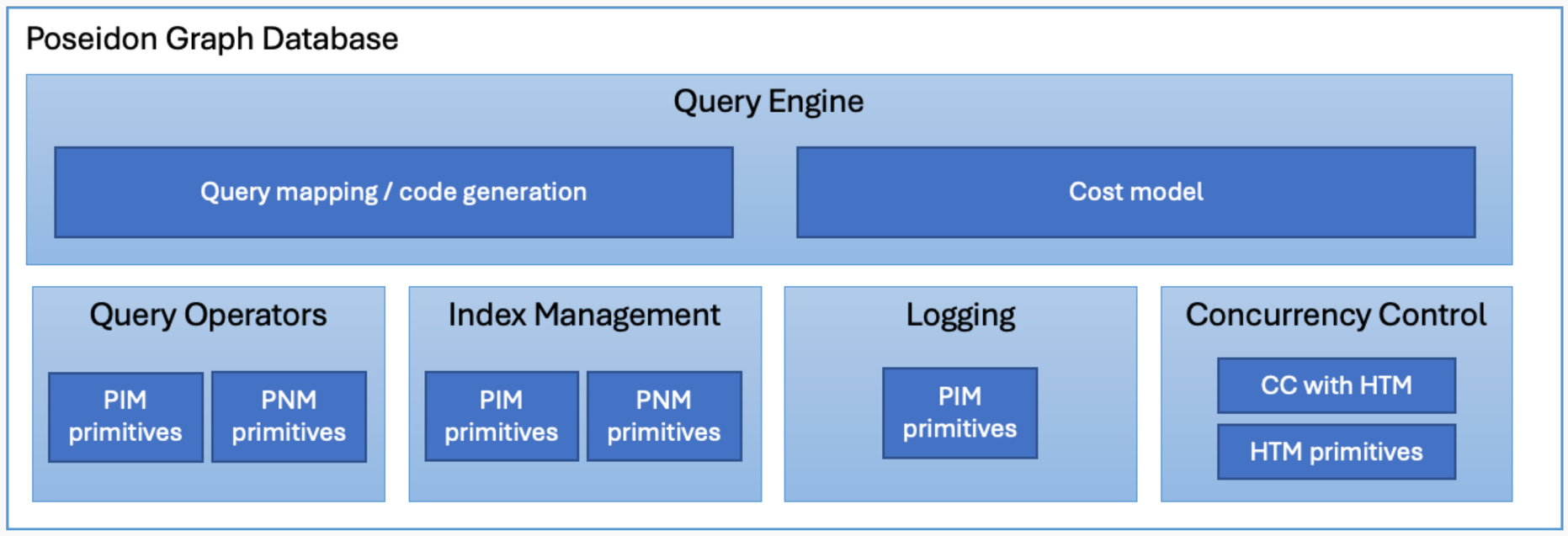
SOFTWARE PLATFORM

- Poseidon graph database
- property graph model
- HTAP workload: Cypher-like queries + transactional updates
- Use PMem as primary storage, i.e. maintain graph directly in PMem
- https://github.com/dbis-ilm/poseidon_core

HARDWARE PLATFORMS

- UPMEM: 2D DRAM array combined with DRAM Processing Units (DPU) running at 350 MHz
- Intel FPGA Acceleration Card D5005: FPGA + device memory

WORK PROGRAM



Technische Universität Ilmenau

Department of Computer Science and Automation
Databases and Information Systems Group
contact: kus@tu-ilmenau.de

funded under
grant no. SA782/31 DFG

TECHNISCHE UNIVERSITÄT ILMENAU