PERSISTENT MEMORY BASED KEY-VALUE STORE FOR DATA ACQUISITION SYSTEMS

Maciej Maciejewski, Grzegorz Jereczek, Jakub Radtke (Intel Corporation)
Danilo Cicalese, Giovanna Lehmann Miotto (CERN)
September 2019

LEGAL NOTICES & DISCLAIMERS

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer or learn more at [intel.com].

Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. Consult other sources of information to evaluate performance as you consider your purchase. For more complete information about performance and benchmark results, visit www.intel.com/benchmarks.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel, the Intel logo, Intel Xeon, Intel, Optane are trademarks of Intel Corporation or its subsidiaries in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

DATA ACQUISION TYPICAL USE CASE

Data stream

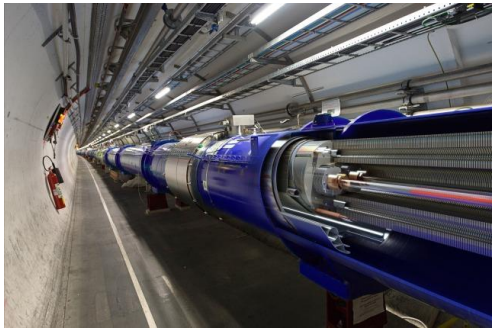
Processing at the edge

Filtered data

Cloud storage

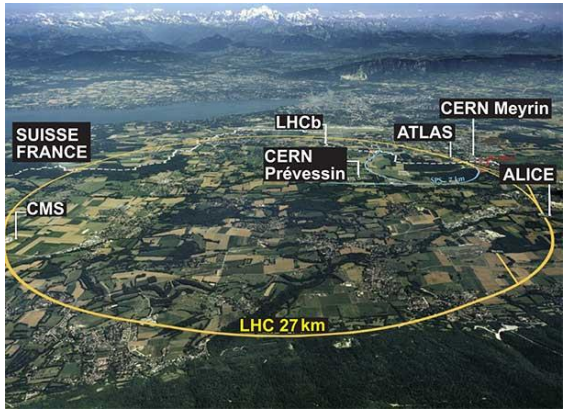
Key – timestamp, camera ID, geo, ...
Value - frame

DAQ DB enables better ways to process data at the edge

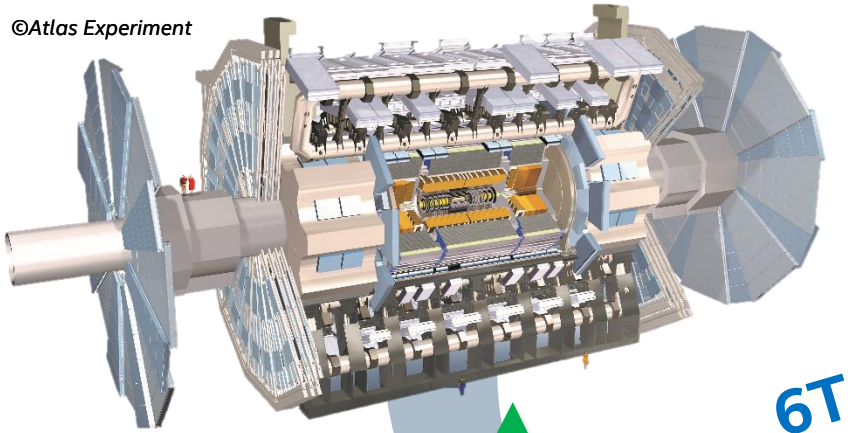


CERN openlab

LHC EXPERIMENTS WILL BE PRODUCING HUNDREDS OF PETABYTES A DAY



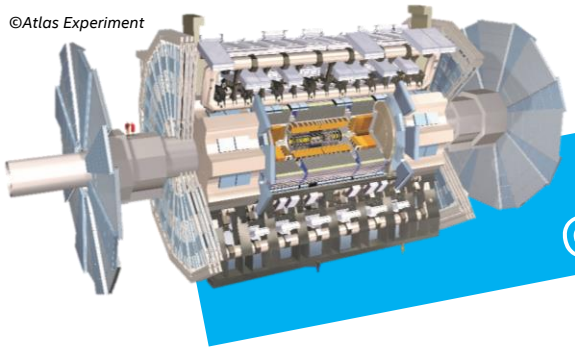
DATA ACQUISITION SYSTEMS (DAQ)



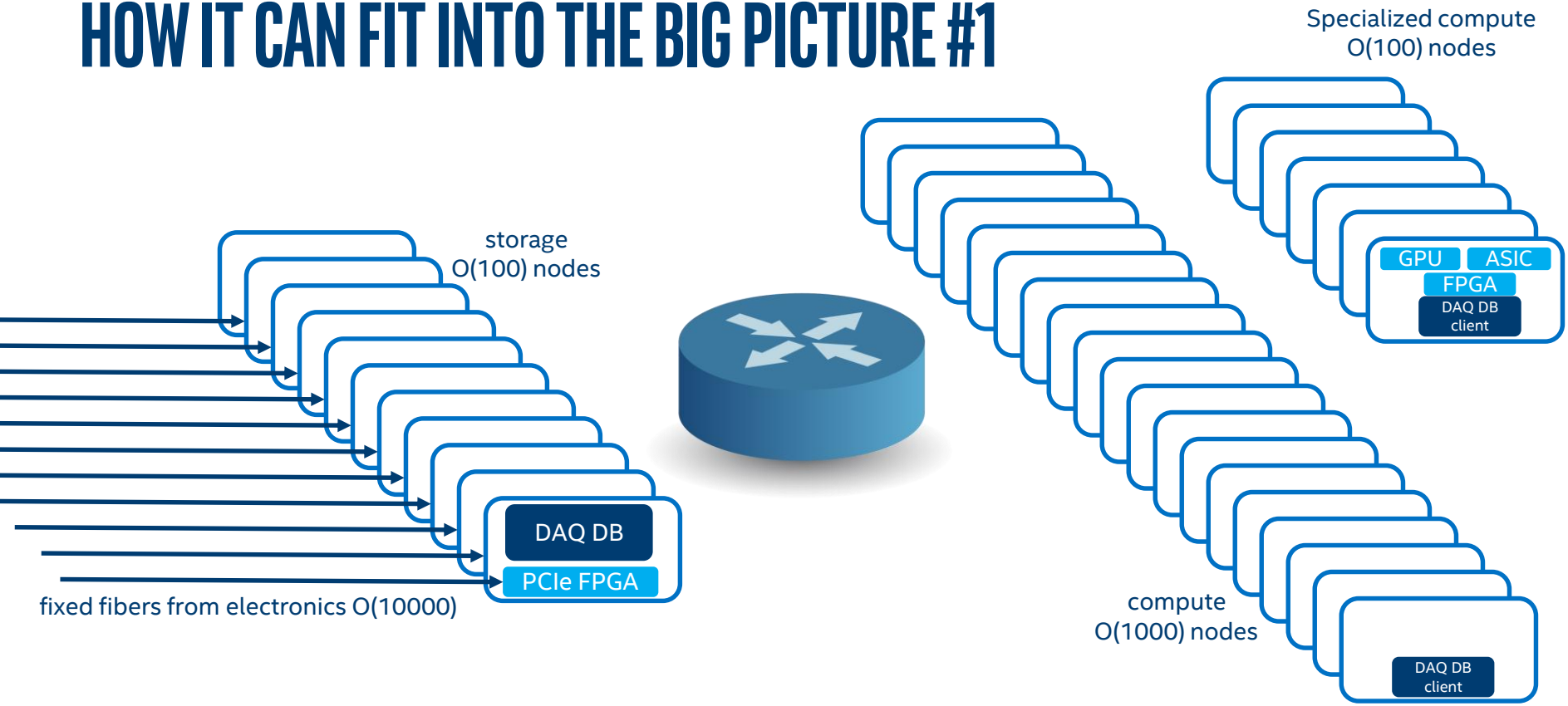
©Atlas Experiment

DAQDB – A KVS FOR DAQ

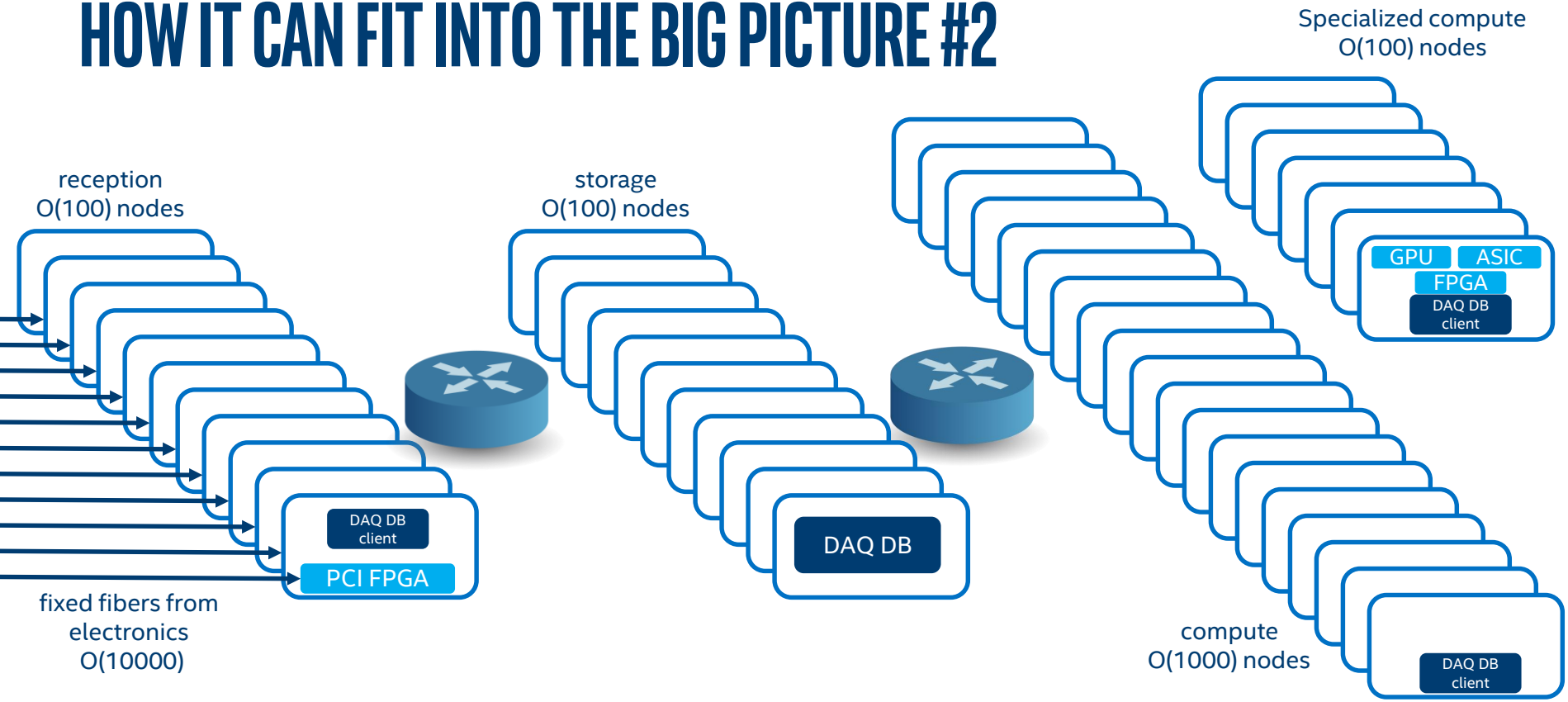
- First-line buffer for fast pre-computing and second-line buffer for longer term storage
- Data structure based on optimized Adaptive Radix Trie
- Distributed locking



HOW IT CAN FIT INTO THE BIG PICTURE #1



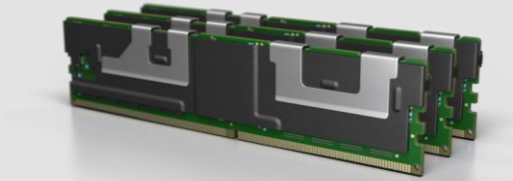
HOW IT CAN FIT INTO THE BIG PICTURE #2



PMDK

Persistent Memory Development Kit

- Optimal performance of persistent memory



intel OPTANE™ DC PERSISTENT MEMORY

SPDK

Storage Performance Development Kit

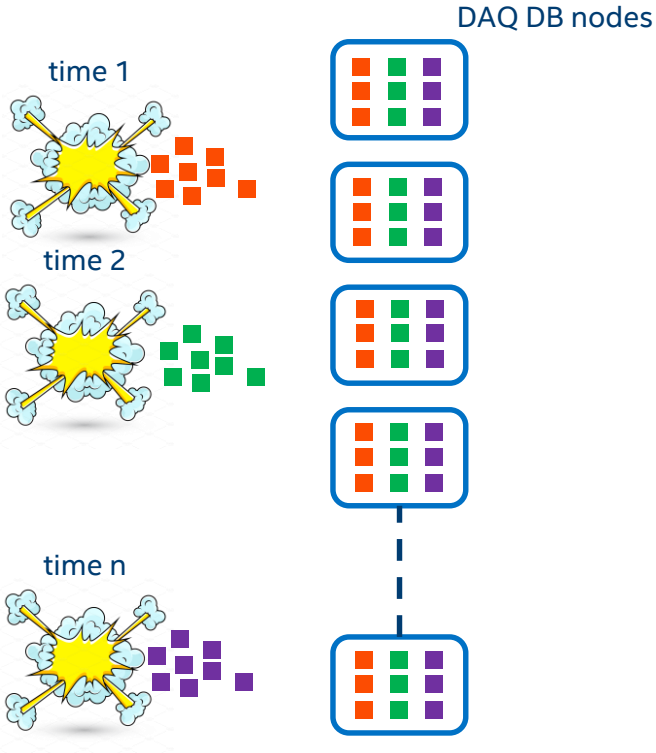
- User-mode access to NVMe devices (SSDs)

intel OPTANE™

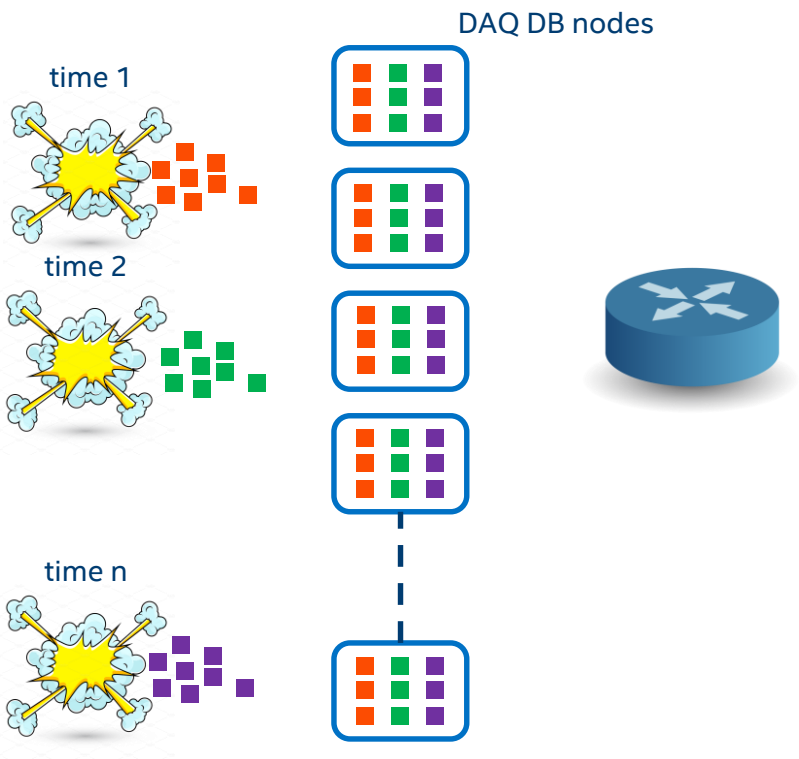
DAQ-SPECIFIC API

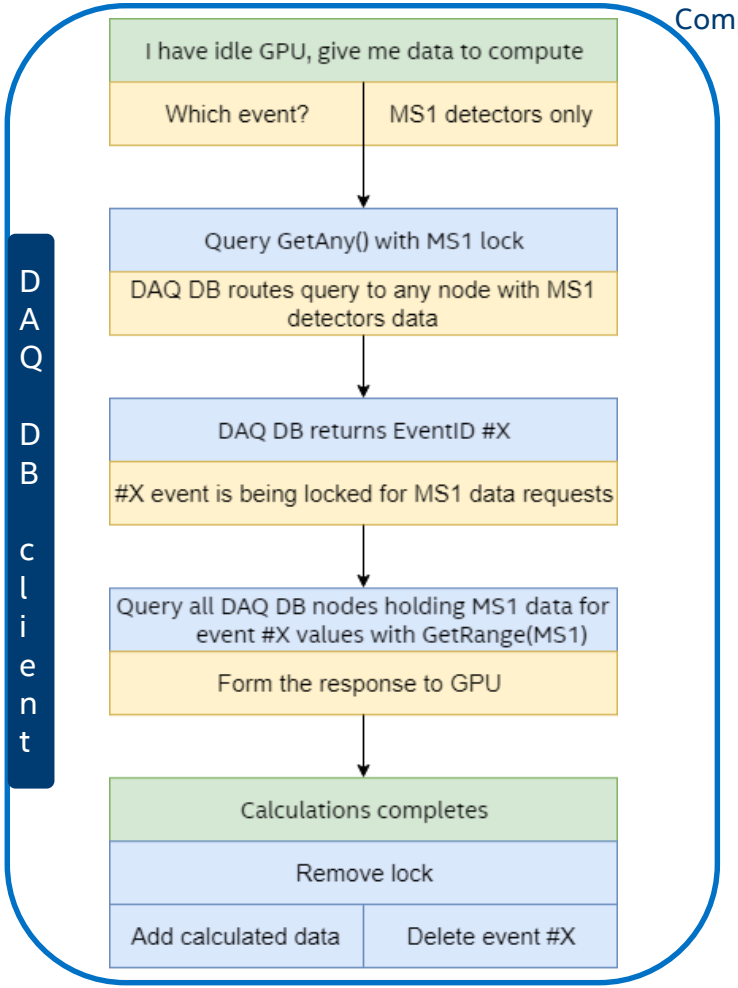
User-defined key structure	struct **MinidaqKey** { uint64_t eventId; uint8_t detectorId; uint16_t subdetectorId; uint16_t runId; }
Asynchronous mode for even higher performance	kvs->**GetRangeAsync**(keyMin, keyMax, cb)
DAQDB memory allocator minimizing copy operations	value = kvs->**Alloc**(key, 10 * 1024)
Range queries with compound keys	kvPairVector = kvs->**GetRange**(keyMin, keyMax)
Distributed locking for next event retrieval	eventKey = kvs->**GetAny**(options=(lock))

GetAny & GetRange EXAMPLE



GetAny & GetRange EXAMPLE





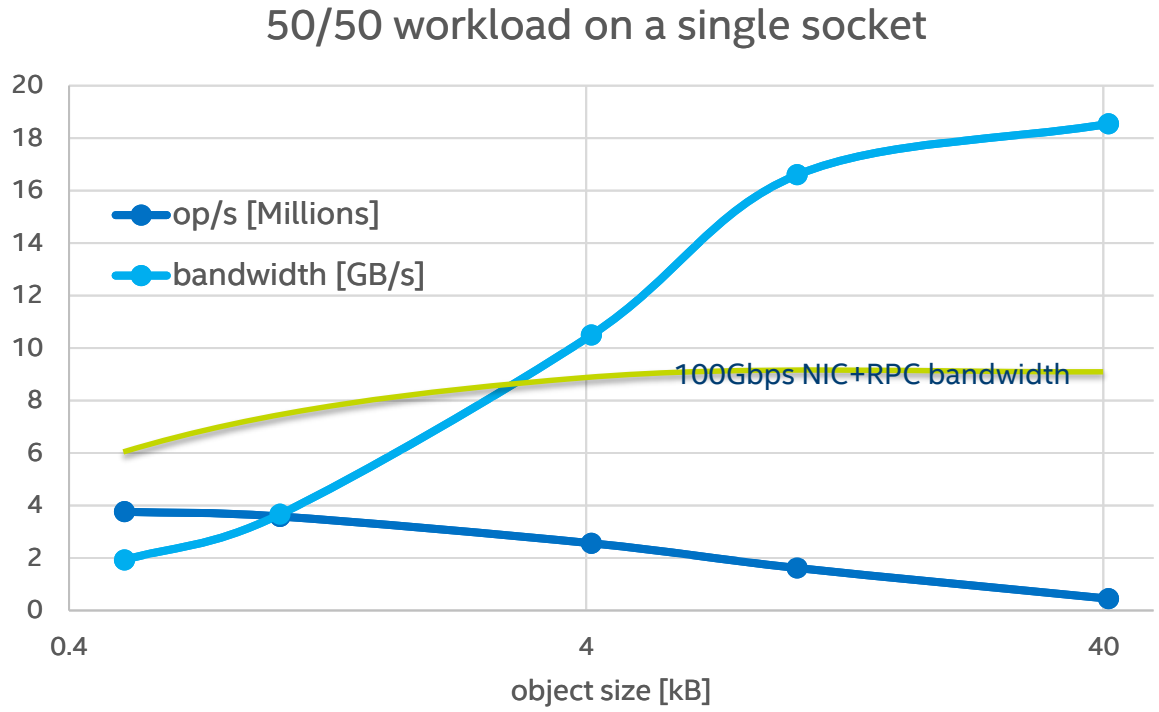
PERFORMANCE

2-2-2 configuration of
512GB Optane DC PMEM +
16GB DDR4

Optane DC PMEM App Direct mode
with full persistency
based on PMDK

CLX 8280: 28 cores @2.8GHz

NIC: Mellanox ConnectX®-5



- Throughput matching 2x100Gbps
- Distributed Range Queries
- RDMA networking

DAQDB

Multi-TB/s hot storage solution of a petascale capacity for DAQ

https://github.com/daq-db/daqdb