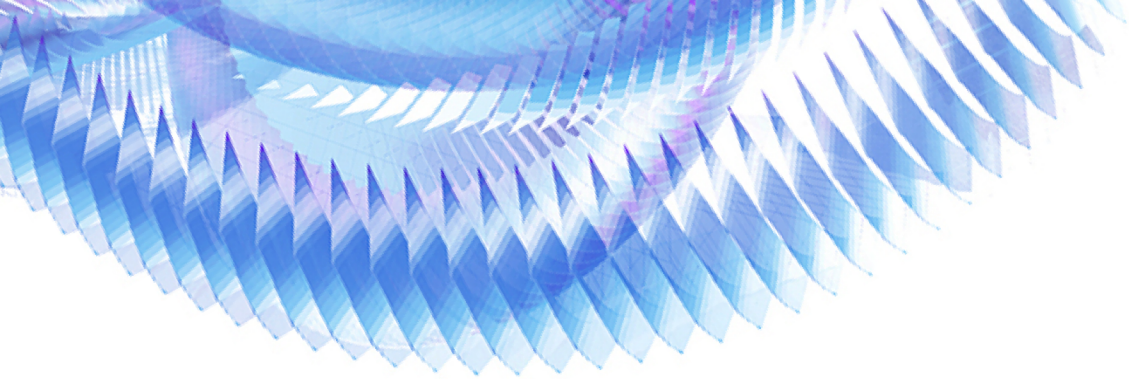
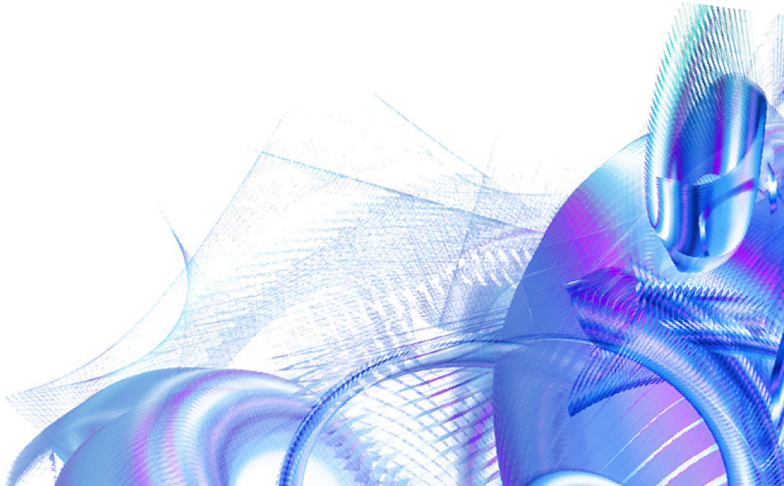


SC19
Denver, CO | hpc is now.

I/O MIDDLEWARE UPDATE

Mohamad Chaarawi, Intel
November, 2019



NOTICES AND 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 product or component can be absolutely secure.

Tests document performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect actual performance. For more complete information about performance and benchmark results, visit http://www.intel.com/benchmarks .

Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products. For more complete information visit http://www.intel.com/benchmarks .

Intel Advanced Vector Extensions (Intel AVX) provides higher throughput to certain processor operations. Due to varying processor power characteristics, utilizing AVX instructions may cause a) some parts to operate at less than the rated frequency and b) some parts with Intel® Turbo Boost Technology 2.0 to not achieve any or maximum turbo frequencies. Performance varies depending on hardware, software, and system configuration and you can learn more at http://www.intel.com/go/turbo.

Intel's compilers may or may not optimize to the same degree for non-Intel microprocessors for optimizations that are not unique to Intel microprocessors. These optimizations include SSE2, SSE3, and SSSE3 instruction sets and other optimizations. Intel does not guarantee the availability, functionality, or effectiveness of any optimization on microprocessors not manufactured by Intel. Microprocessor-dependent optimizations in this product are intended for use with Intel microprocessors. Certain optimizations not specific to Intel microarchitecture are reserved for Intel microprocessors. Please refer to the applicable product User and Reference Guides for more information regarding the specific instruction sets covered by this notice.

Cost reduction scenarios described are intended as examples of how a given Intel-based product, in the specified circumstances and configurations, may affect future costs and provide cost savings. Circumstances will vary. Intel does not guarantee any costs or cost reduction.

Intel does not control or audit third-party benchmark data or the web sites referenced in this document. You should visit the referenced web site and confirm whether referenced data are accurate.

POSIX I/O Support

DAOS File System (libdfs)

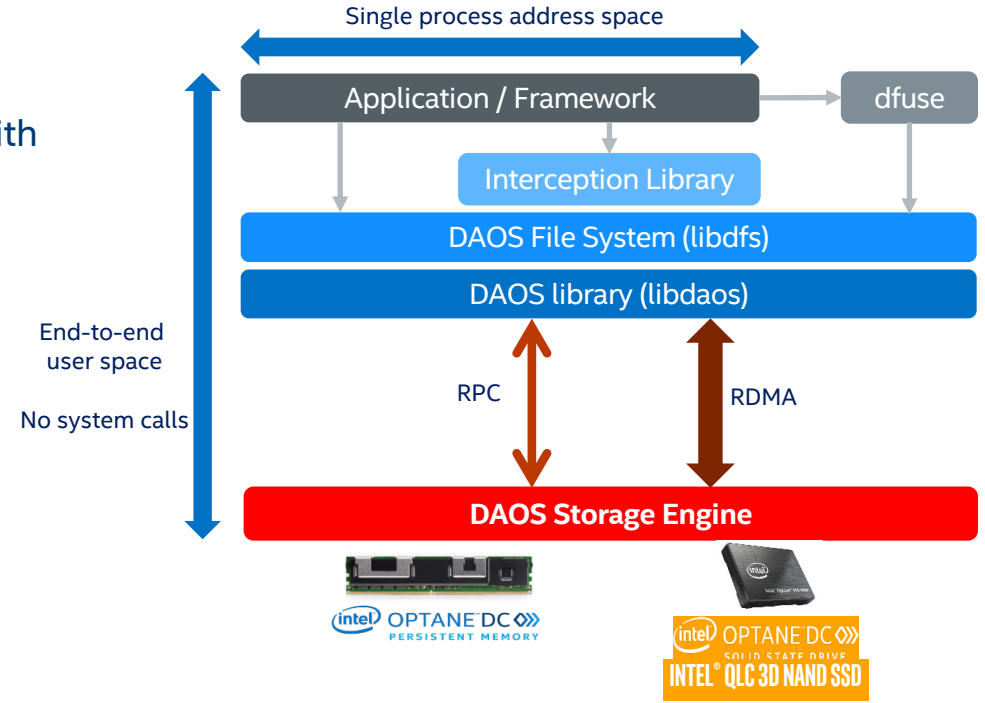
- Encapsulated POSIX namespace
- Application/framework can link directly with libdfs
  - ior/mdtest backend provided
  - MPI-IO driver leveraging collective open
  - TensorFlow, …

FUSE Daemon (dfuse)

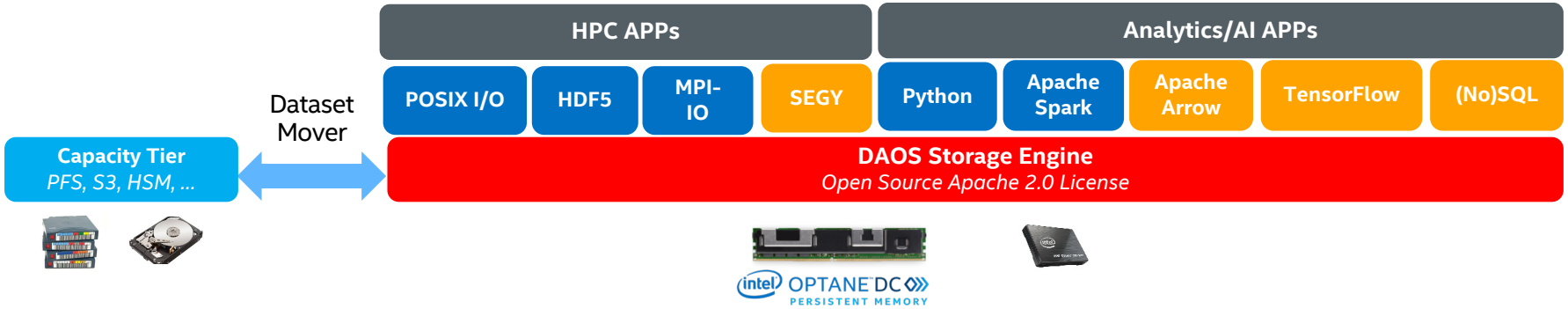
- Transparent access to DAOS
- Involves system calls

I/O interception library

- OS bypass for read/write operations



Application Interface

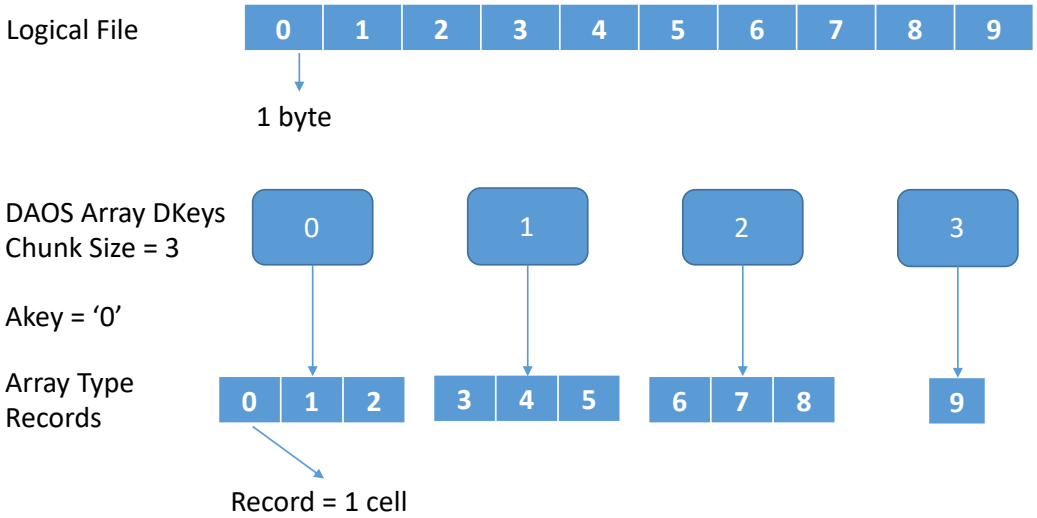


MPI-IO Driver for DAOS

The DAOS MPI-IO driver is implemented within the I/O library in MPICH (ROMIO).

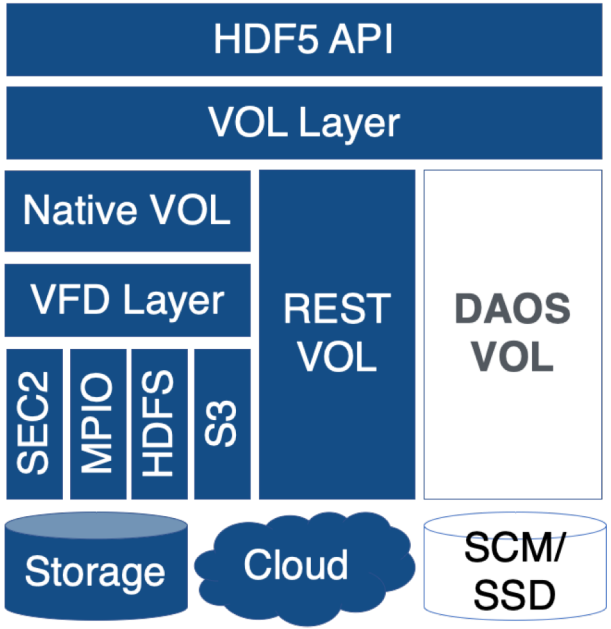
- Added as an ADIO driver
- Portable to Open-MPI, Intel MPI, etc.
- https://github.com/daos-stack/mpich
- daos_adio branch
- PR to mpich master in review

- 1 MPI File = 1 DAOS Array Object

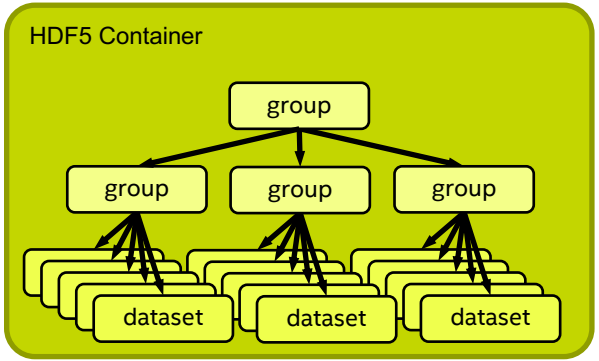


Application works seamlessly by just specifying the use of the driver by appending "`daos:`" to the path.

HDF5®



- Developing an HDF5® VOL Connector
- Minimal / No application code changes (including other middleware I/O libraries (e.g. NetCDF4, PIO, etc.)



Adding new extensions to the HDF5® library that are not available to date without the DAOS VOL connector

- Asynchronous I/O for both metadata and raw data operations
- Container Snapshots
- Query & Indexing API

HDF5 DAOS VOL Connector – Current Status

No longer requires separate version of HDF5

- Compatible with main develop branch of HDF5
- Compatible with upcoming 1.12.x release series of HDF5 with VOL support

Currently supported features

- All HDF5 object types are currently supported except references (new API for references)
- New H5M MAP API to expose K/V interface to HDF5 users
- Variable length datatypes are now also supported
- Chunking is recommended storage layout to get most of DAOS performance

Coming by end of the year

- References, fill values, point selection
- Independent metadata writes (= independent object creation)

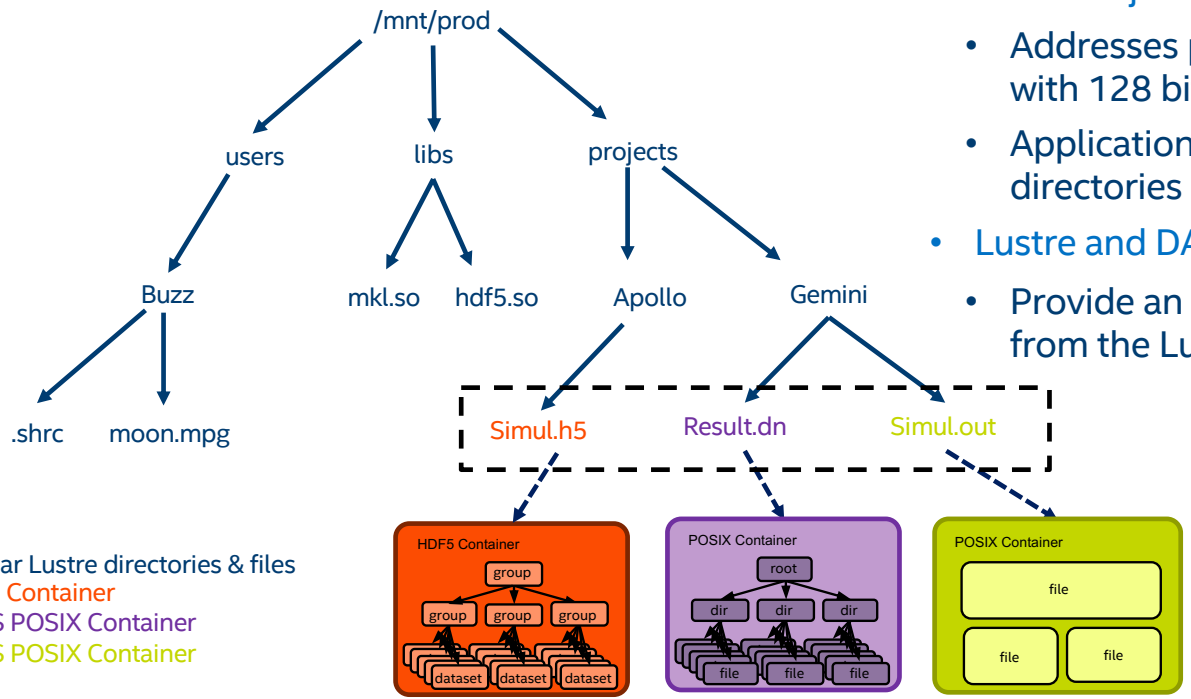
2020:

- Asynchronous I/O
- Tools support (h5dump, h5ls, h5repack etc)

Available from: https://git.hdfgroup.org/projects/HDF5VOL/repos/daos-vol/

- See user's guide for more detailed list of supported features

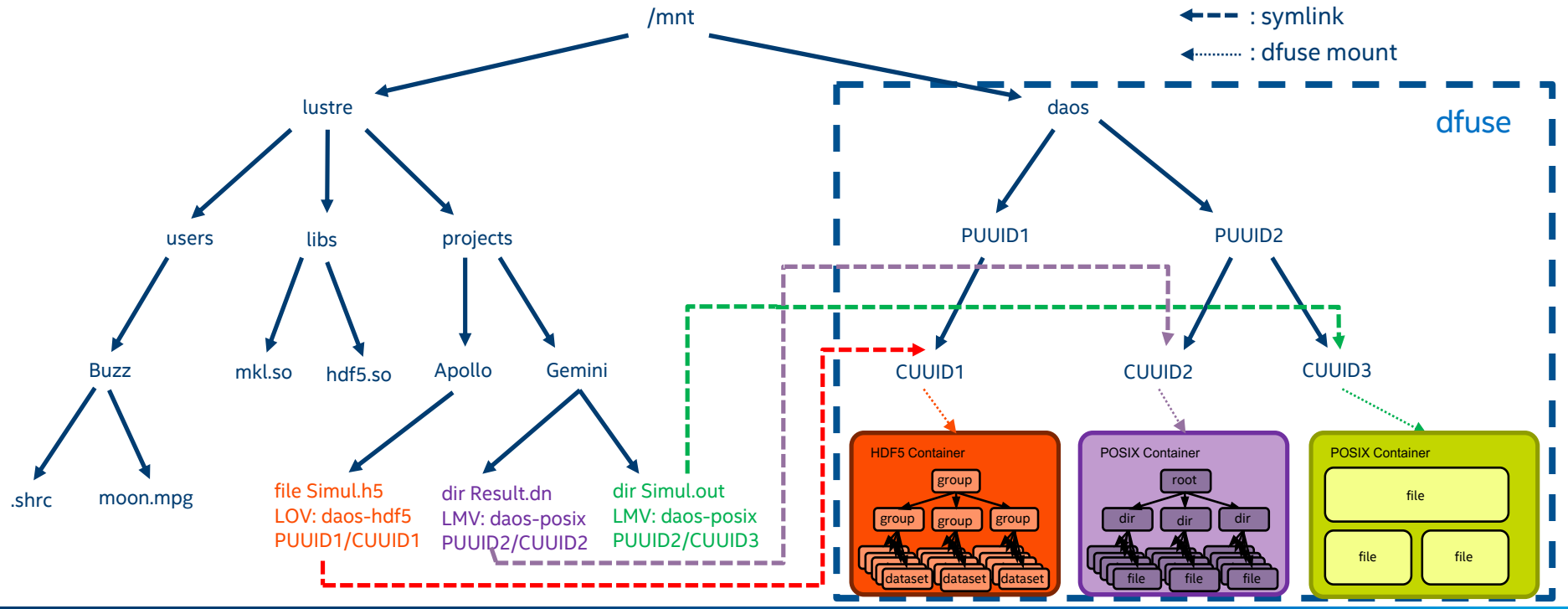
Lustre/DAOS Integration



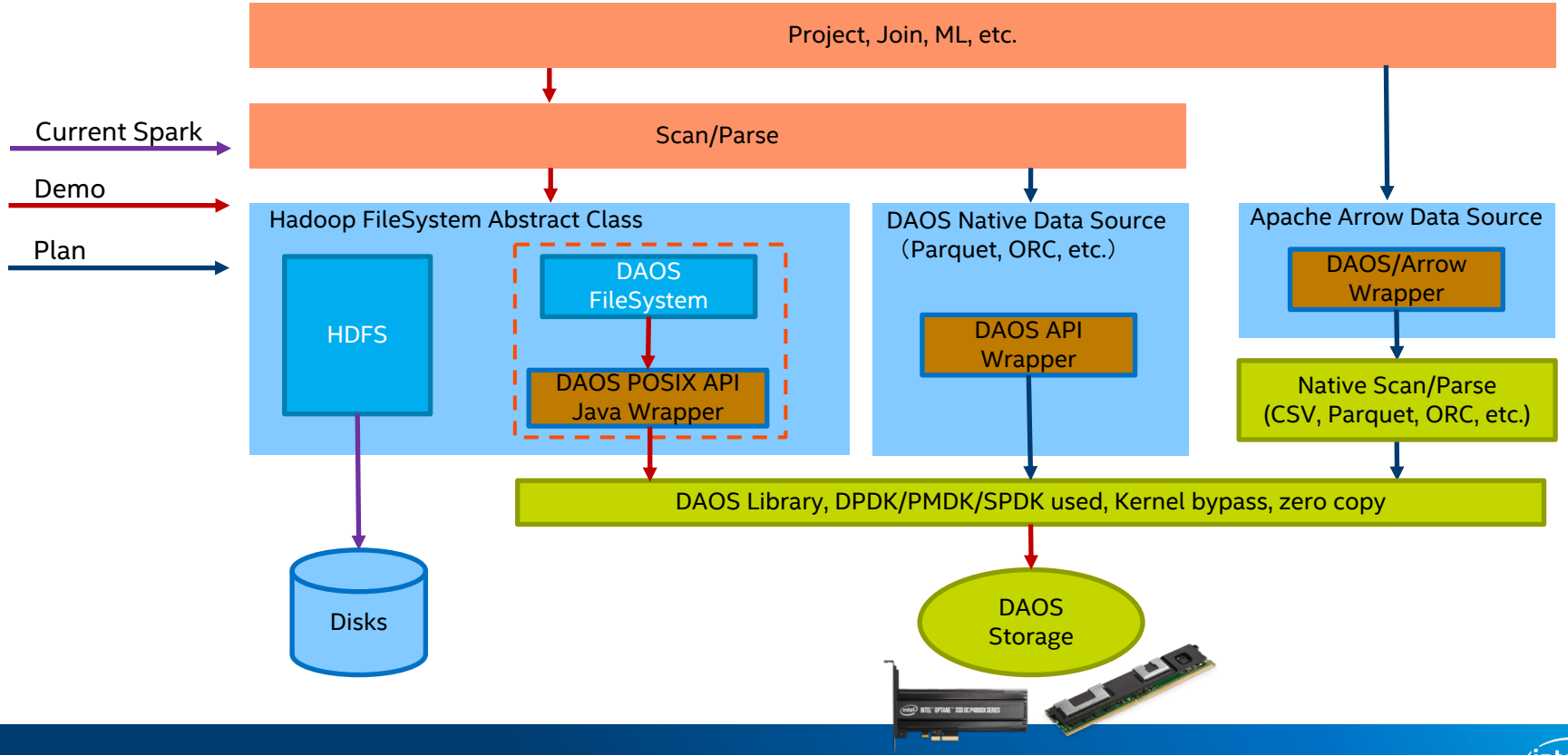
Regular Lustre directories & files
HDF5 Container
DAOS POSIX Container
DAOS POSIX Container

- DAOS Object Store:
  - Addresses pools, containers with uuids, object with 128 bit IDs.
  - Applications/Users are used to access files / directories in a traditional namespace
- Lustre and DAOS tiers:
  - Provide an easy way to look at DAOS containers from the Lustre namespace

Unified Name Space Concept



Spark Input/Output Support



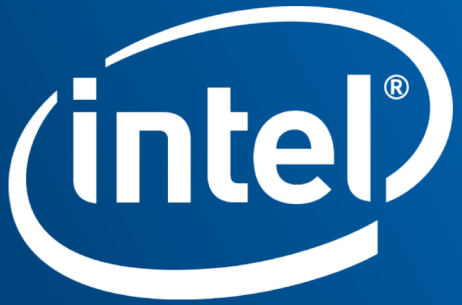
Python Integration

Pythonic bindings called pydaos

- Export key-value store objects
- Integrated with python dictionnaries
  - Support python iterator, direct assignments, …
- Scalable & performant
  - Bulk insert/retrieve
  - Core written in C
- Python 2.7 & 3 support
- Pyprob support

TODO

- Expose snapshots
- Integration with NumPy
- Explore integration with python frameworks like PyTorch, Intel HPAT, …


intel®