CROSS-TIER UNIFIED NAMESPACE UPDATE

Mohamad Chaarawi, Bruno Faccini, Johann Lombardi

LUG - May 16, 2019

DCG/ESAD, Intel

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 computer system 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.

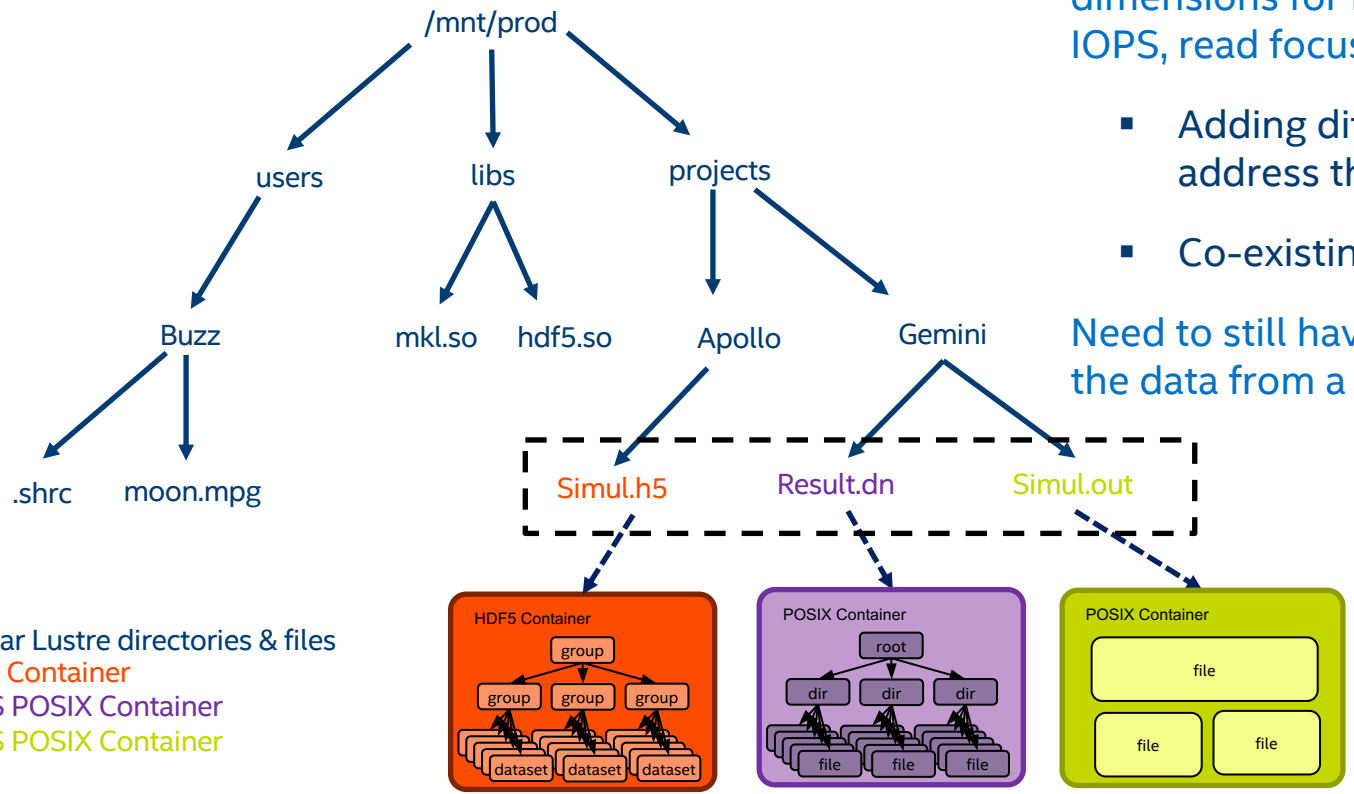
Cost reduction scenarios described are intended as examples of how a given Intel-based product, in the specified circumstances and configurations, may affect future costs and provide cost savings. Circumstances will vary. Intel does not guarantee any costs or cost reduction.

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.
Intel, the Intel logo, and Intel Xeon are trademarks of Intel Corporation in the U.S. and/or other countries.
*Other names and brands may be claimed as property of others.

AGENDA

- DAOS Overview and Storage Architecture
- Unified Namespace with DAOS and Lustre
- Lustre Integration with UNS
- Dataset Migration between Tiers
- Not in Scope:
  - DAOS internals, features, performance
  - A comparison between DAOS, Lustre, other PFS or Object Store

UNIFIED NAMESPACE CONCEPT



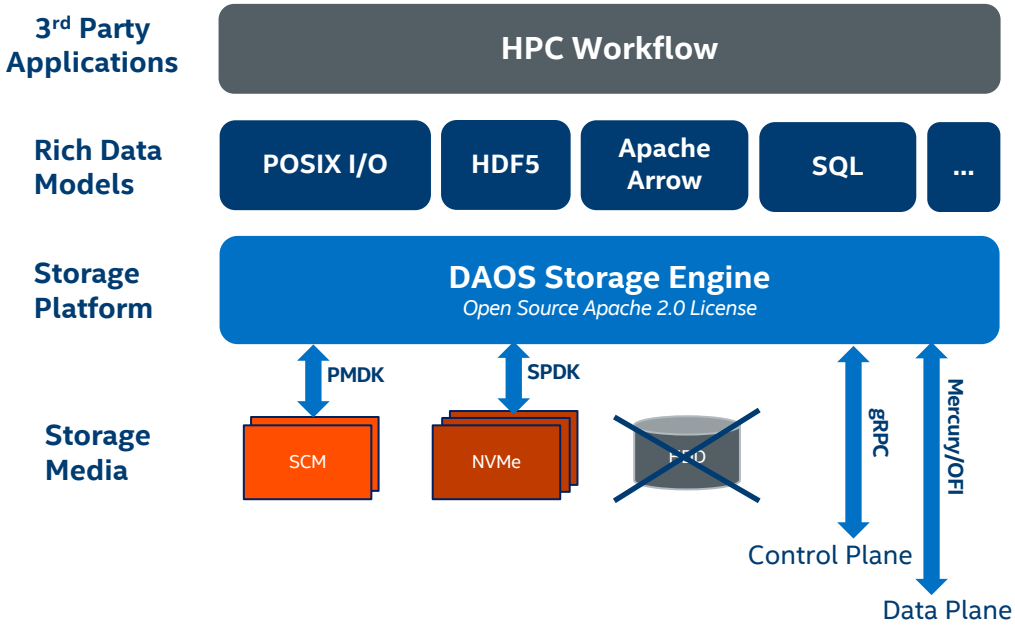
Regular Lustre directories & files
HDF5 Container
DAOS POSIX Container
DAOS POSIX Container

Emerging I/O workloads add new dimensions for I/O requirements (high IOPS, read focused, etc.)

- Adding different storage tier(s) to address those requirements.
- Co-existing with Lustre

Need to still have an easy way to access the data from a user's perspective

DISTRIBUTED ASYNCHRONOUS OBJECT STORAGE



Built natively over **new userspace** PMEM/NVMe software stack

- Not intended for block devices.

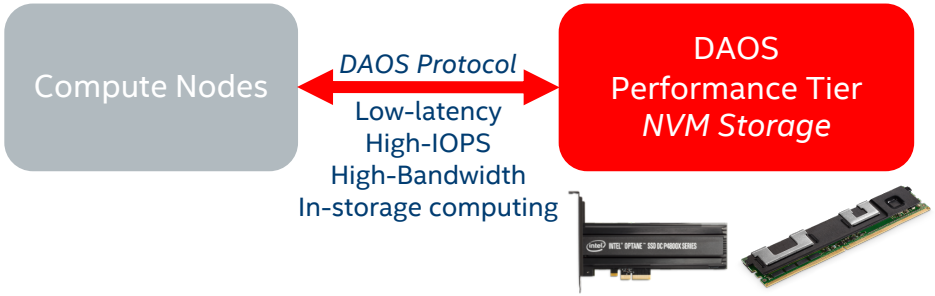
Support for relaxed POSIX semantics and other middleware

DAOS is open source:

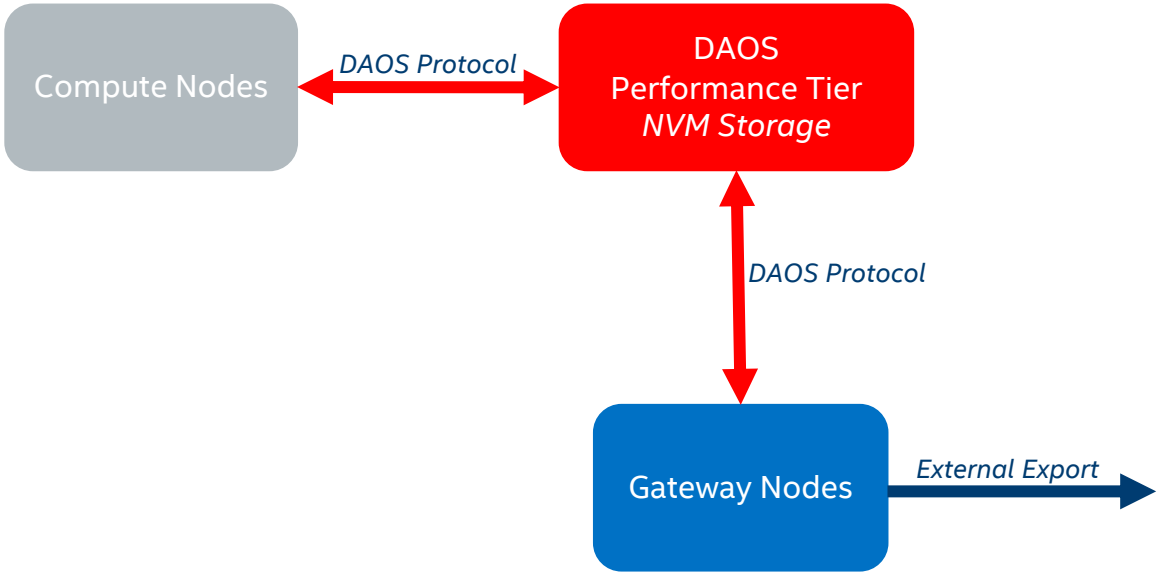
- https://github.com/daos-stack/daos
- http://daos.io

For more details about DAOS internals and features, please ping me offline

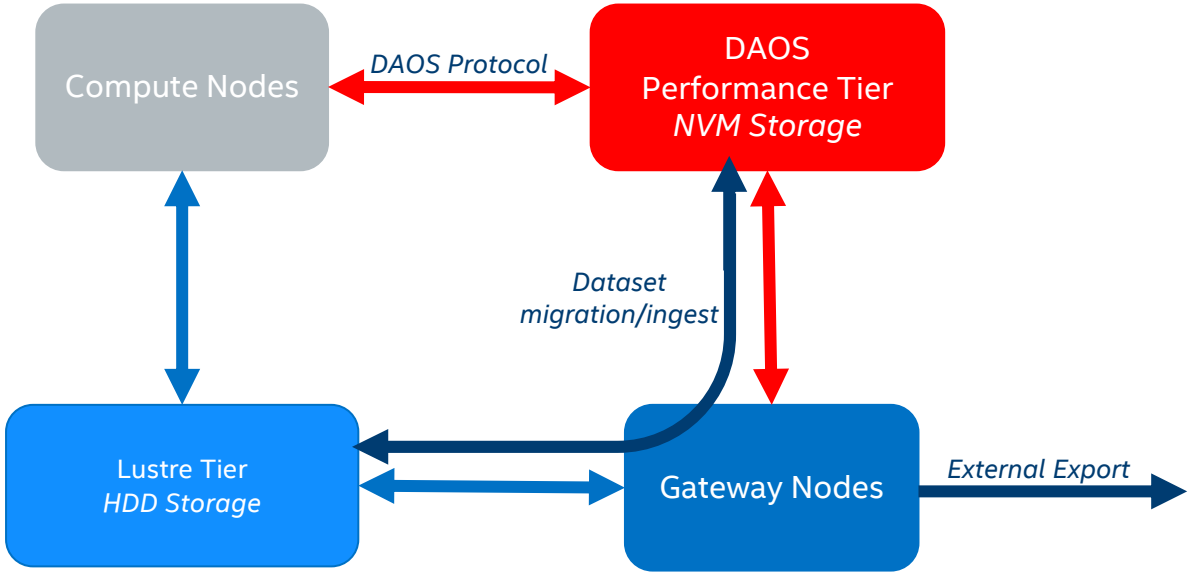
STORAGE ARCHITECTURE



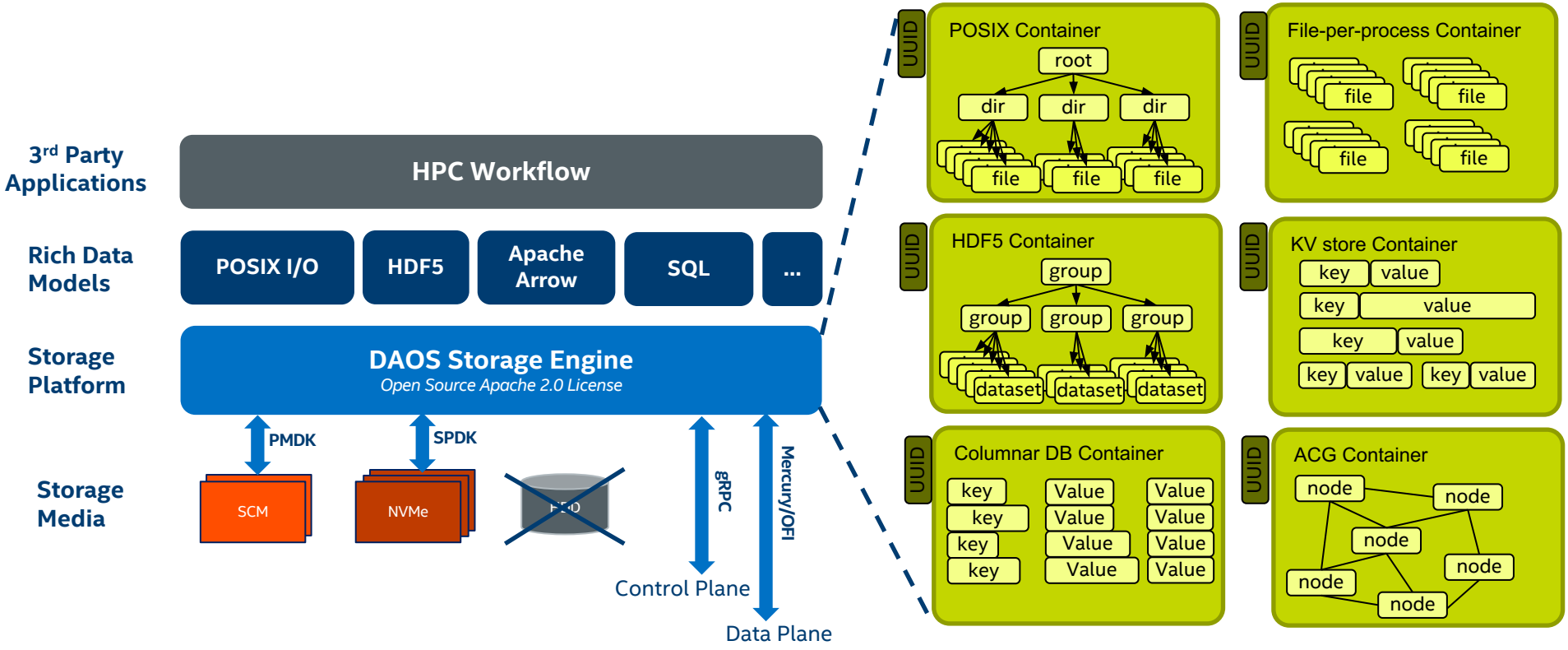
STORAGE ARCHITECTURE



STORAGE ARCHITECTURE

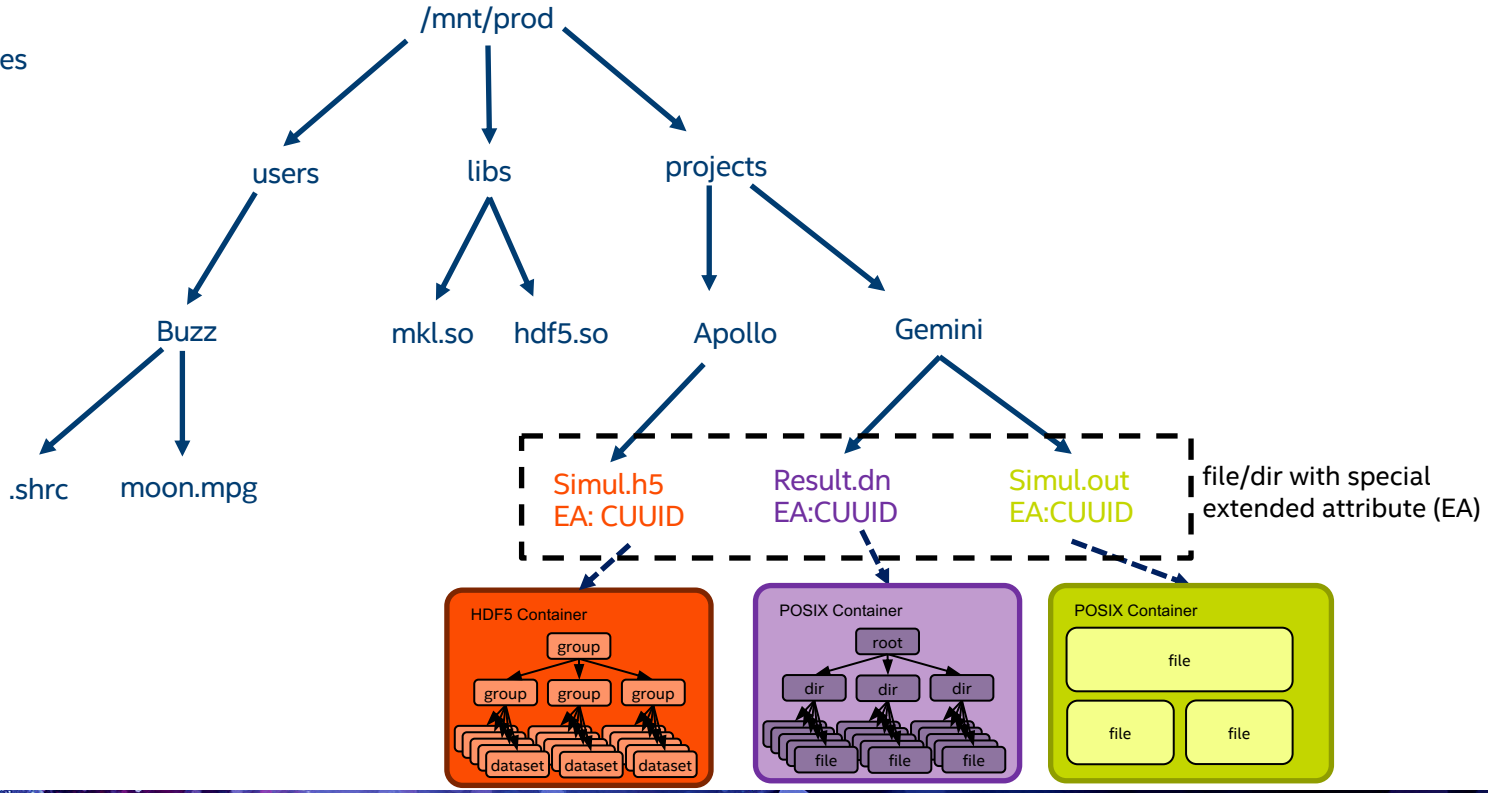


DISTRIBUTED ASYNCHRONOUS OBJECT STORAGE



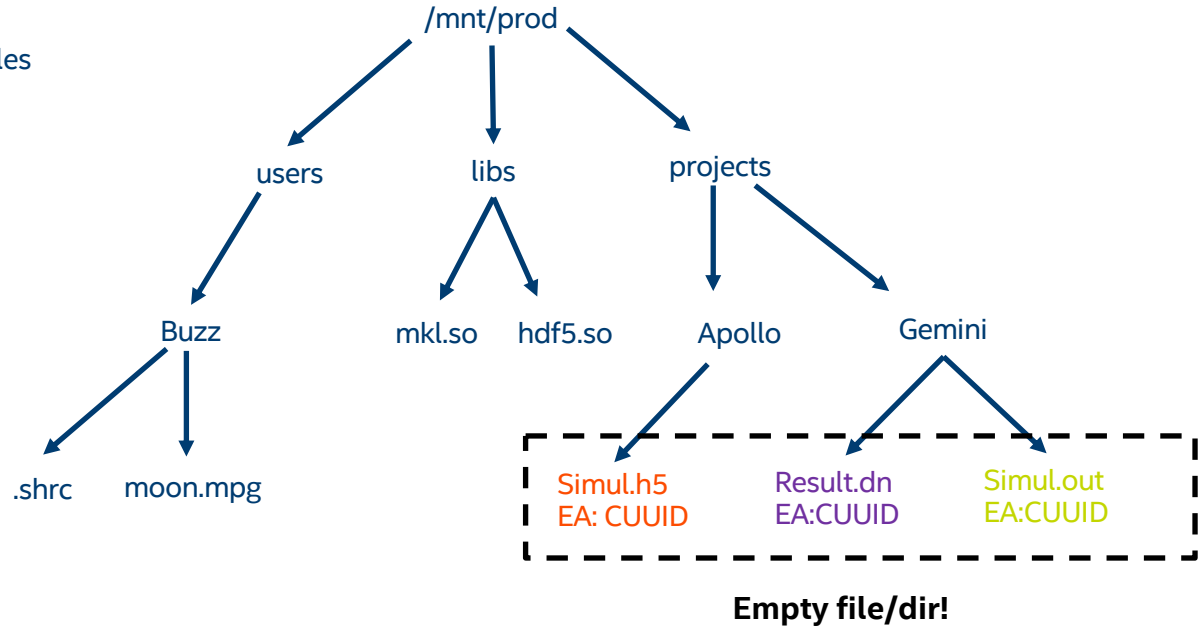
UNIFIED NAMESPACE CONCEPT

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



WHAT'S REALLY STORED IN THE PFS?

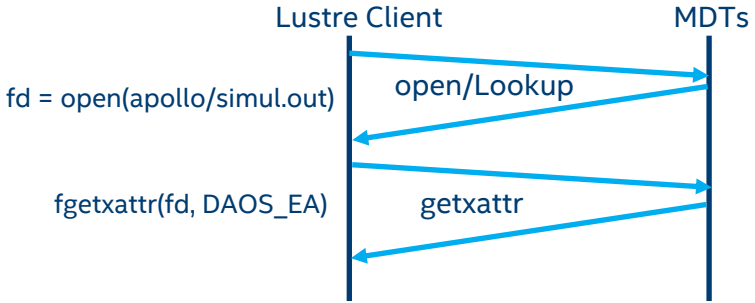
Regular Lustre directories & files
HDF5 Container
DAOS POSIX Container
DAOS POSIX Container



SPECIAL FILE/DIR REPRESENTATION (TO STORE EXTERNAL TIER ATTRIBUTES)

Regular Extended Attribute (EA)

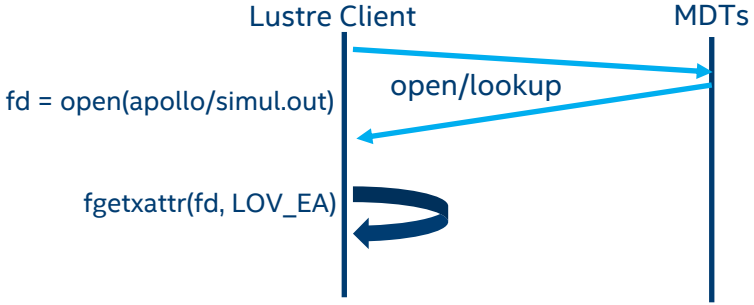
- Portable
- Performance Impact
  - Extra EA fetch on every lookup



- Can't prevent Lustre file/dir from being created under the special directory

Special LOV/LMV EA

- Not Portable
- Minimal Performance Impact
  - No extra RPC



- Prohibit regular file/dir creation

DAOS/LUSTRE INTEGRATION

- Extend LOV/LMV EAs
  - New layout type to point at external tier
  - Generic feature based on UUID
  - Can be integrated with any scale-out object stores
  - Opportunity to leverage layout swap functionality for cross-tier migration
  - Benefit of existing pre-fetch mechanism, DLM protection and of client/server multi-stage caches, for both LOV/LMV EAs
- Effort tracked in LU-11376
  - https://jira.whamcloud.com/browse/LU-11376
  - Merged to master and will included in v2.13

LU-11376 CHANGES DETAIL

- 2 patches developed
  - New foreign LOV format for files (https://review.whamcloud.com/33755)
  - New foreign LMV format for directories (https://review.whamcloud.com/34087)
  - Recently landed
- Same foreign format to be allowed for both on-disk LOV and LMV

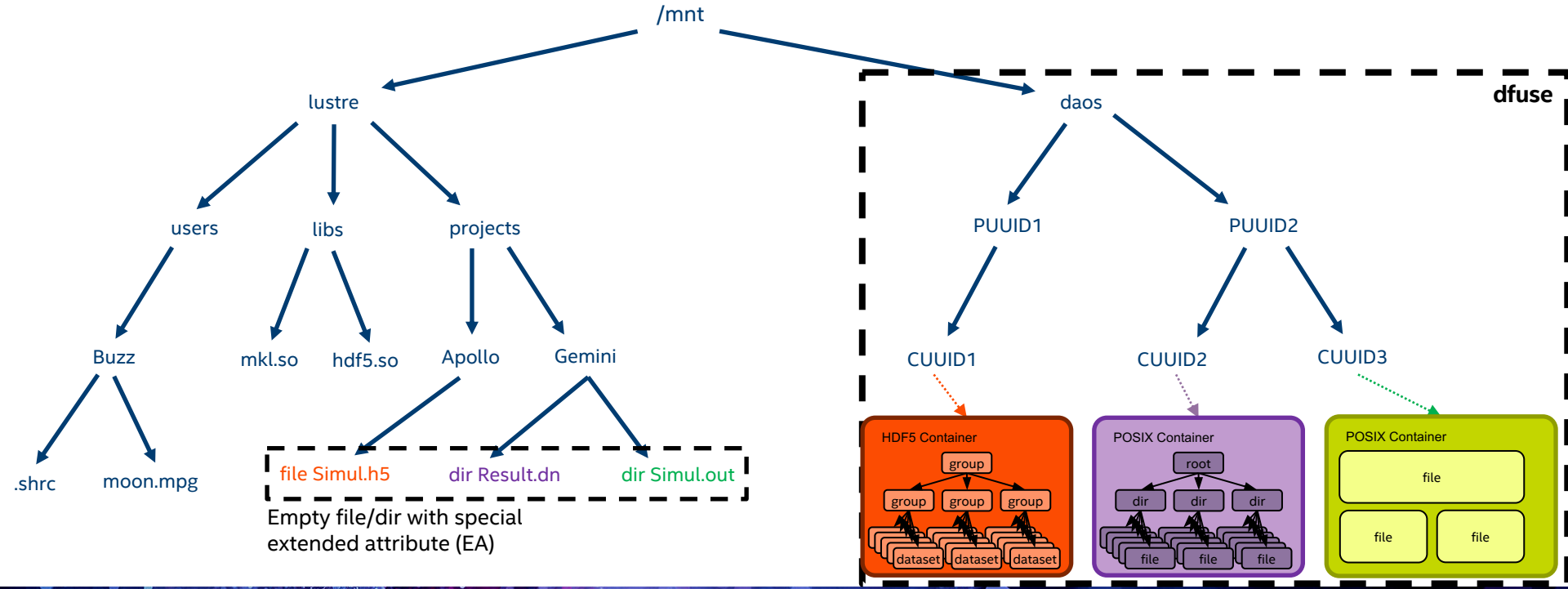
  {u32 new[LOV,LMV]magic, u32 length, u32 type, u32 flags, free string[length]}
- Both patches implement
  - Lustre API changes (added ll_ [set,get][dir]stripe() support of new format)
  - Lustre tools changes (new options in lfs [set,get][dir]stripe, lfs find)
  - lfsck compatibility changes

FOREIGN LOV/LMV FORMAT DETAIL

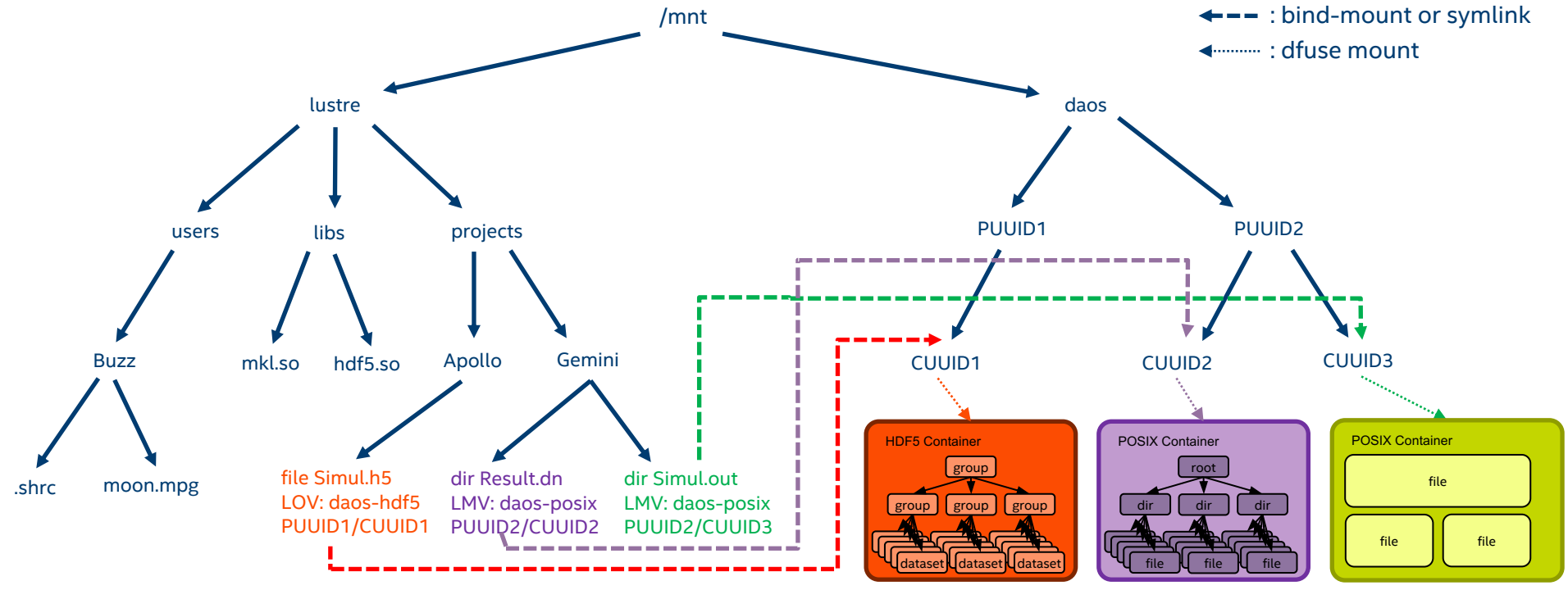
- New LOV/LMV magic (__u32) values :
  - 0x0BD70BD0 for LOV
  - 0x0CD50CD0 for LMV
- length (__u32) : length of free format string
- type (__u32) : optional, to identify a service/subsystem
- flags (__u32) : optional, type specific
  - type and flags added at request from Cray (see LU-11376 for more details)
- free format/length string

UNIFIED NAMESPACE LAYOUT

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



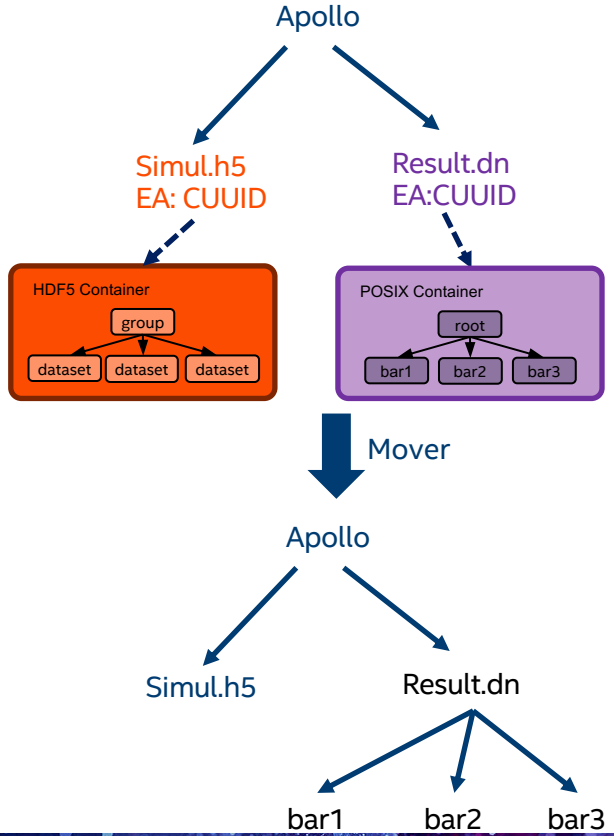
TRANSPARENT ACCESS OF DAOS STORAGE FROM LUSTRE



CONSISTENCY BETWEEN TIERS

- The DAOS container can store, in its attributes, the path of the special file that it is linked with in the Lustre namespace.
  - If the special file/dir is deleted, the container has some information to indicate the original path of the container.
- We can store the Lustre FID in the DAOS container attributes
  - If the link was broken between the DAOS container and the Lustre special file/dir, it can be recreated using the FID
  - Storing the path isn't enough because a rename in the lustre namespace doesn't change the path stored in the container (FID doesn't change in this case).

DATA MOVER



- Different use cases
  - POSIX container migration
  - Other middleware specific data migration (e.g. HDF5)
  - Cross-Pool Container Migration
- Develop an MPI application based on open source mpifileutils from LLNL.
  - Parallel movement of datasets between tiers.
- Provide a library and DAOS tool that allows integration with other data movement frameworks (e.g. Globus, DMF, etc.).

RESOURCES

Source code on GitHub

- https://github.com/daos-stack/daos

Community mailing list on Groups.io

- daos@daos.groups.io or https://daos.groups.io/g/daos

Wiki

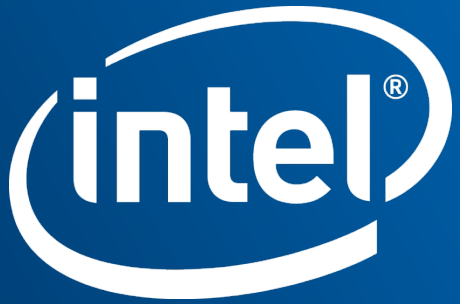
- http://daos.io or https://wiki.hpdd.intel.com

Bug tracker

- https://jira.hpdd.intel.com

Contact

- mohamad.chaarawi@intel.com


intel®