HPE ADVANCE DEVELOPMENT FOR DAOS

DAOS User Group

November 2021

AGENDA

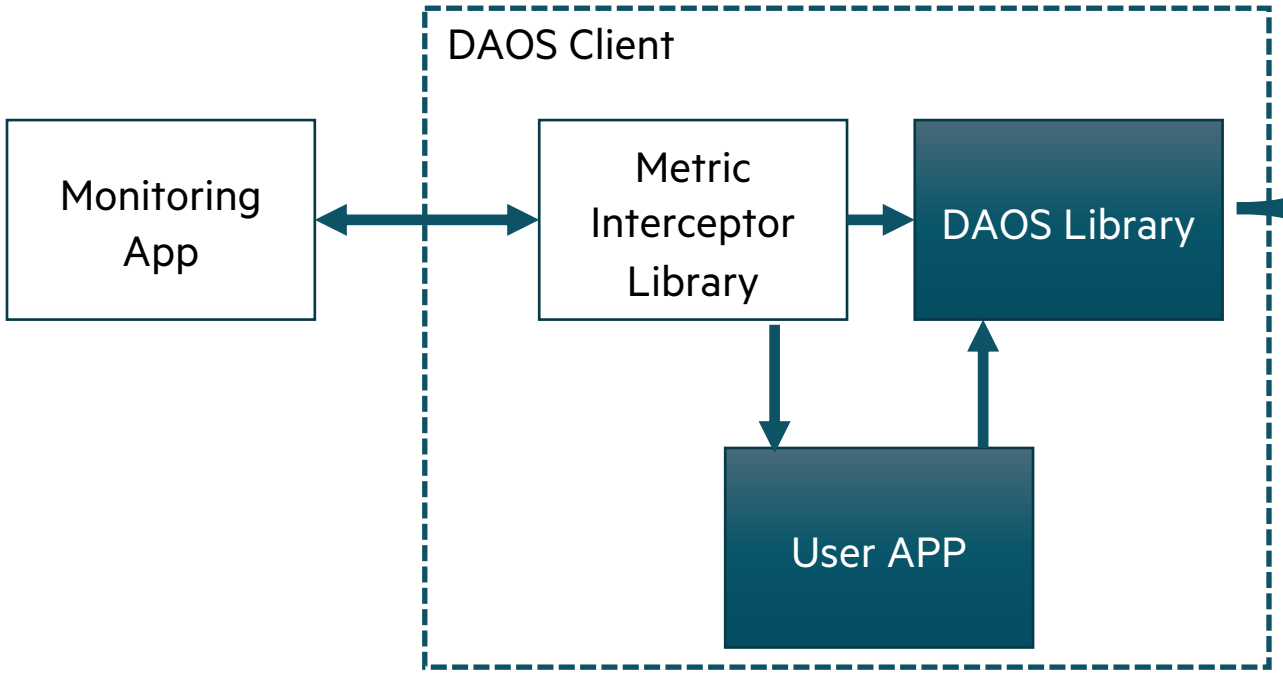
Client-side metrics feature recently submitted by HPE

Administration enablement with HPE's Performance Cluster Manager (HPCM)

HPE’s go-to-market activities

DAOS CLIENT-SIDE METRICS CONCEPTS

Potential Collection Approach



- **Counters**
  - **Various RPC calls**
  - **Successes, Failures, In-Flight**
- **Statistics**
  - **Fetches and Updates**
  - **Count, total size, avg size, deviation etc.**
- **Distributions (Histogram)**
  - **Statistics for Several Size Ranges**
  - **Statistics for Protection types**
- Code
  - https://github.com/daos-stack/daos/pull/6497
  - Expected to be released with DAOS 2.4
- Applications
  - Enable Metrics via New Library API calls
  - Allocate, Dump, Reset, Free Counters
  - Daos_test additions to validate

DAOS CLIENT-SIDE METRICS TEST UTILITY OUTPUT

```
********************  Dumping Pool RPC Counters **********************
Name                          Inflight         Success         Failure
pool connect                         0              14               5
pool disconnect                      0              14               0
pool attr(get/set)                   0               8               0
pool query                           0              26               0
******************  Dumping Container RPC Counters *******************
Name                          Inflight         Success         Failure
cont create                          0              40               4
cont destroy                         0              39               1
cont open                            0              37               2
cont close                           0              37               0
cont snapshot                        0               5               0
cont snaplist                        0               1               0
cont snapdestroy                     0               5               1
cont attr                            0               8               0
cont acl                             0               3               2
cont prop                            0               4               1
cont query                           0              11               9
cont oidalloc                        0               1               0
cont aggregate                       0               1               0
********************  Dumping Object RPC Counters ********************
Name                          Inflight         Success         Failure
obj update                           0             160               0
obj fetch                            0              84               0
obj enum dkey                        0               3               3
obj enum akey                        0               3               2
obj enum recx                        0              13               6
obj enum obj                         0               0               0
obj punch obj                        0               1               0
obj punch dkeys                      0              29               0
obj punch akeys                      0               7               0
obj query keys                       0               1               0
obj sync                             0               1               0
obj cpd                              0               0               0
***************  Dumping Object IO Stats ***************************
Name                  Count           Sum Size       Sum of Sqrs Size             Min             Max
update                  160           52402115         800682222004243               0        24494592
fetch                    84           24712051         199997103372031               1        12247296
```

Pool-Related RPCs

Container-Related RPCs

Object-Related RPCs

Object IO Statistics

```
***************  Dumping i/o Distribution by Size ******************
Name                          update cnt       fetch cnt
IO_0_1K                               65              34
IO_1K_2K                               1               1
IO_2K_4K                               1               1
IO_4K_8K                              22              34
IO_8K_16K                             41               2
IO_16K_32K                             7               3
IO_32K_64K                             4               1
IO_64K_128K                            1               1
IO_128K_256K                           1               1
IO_256K_512K                          11               1
IO_512K_1M                             1               1
IO_1M_2M                               1               1
IO_2M_4M                               1               1
IO_4M_INF                              3               2
**********  Dumping update call Distribution for RP ***************
Name                          update cnt            size
NO_RP                                  5            7050
RP2                                  139        52157970
RP3                                    1           26841
RP4                                    1           21208
RP6                                    1           31566
RP8                                    1           42472
RP12                                   0               0
RP16                                   0               0
RP24                                   0               0
RP32                                   0               0
RP48                                   0               0
RP64                                   0               0
RP128                                  0               0
RPU                                    0               0
**********  Dumping update call Distribution for EC ***************
Name                  fstripe/sng cnt        size     pstripe cnt            size
IO_EC2P1                               2       12352               1            8192
IO_EC2P2                               2       16480               1           12288
IO_EC4P1                               2       20544               1            8192
IO_EC4P2                               2       24672               1           12288
IO_EC8P1                               0           0               0               0
IO_EC8P2                               0           0               0               0
IO_EC16P1                              0           0               0               0
IO_EC16P2                              0           0               0               0
IO_ECU                                 0           0               0               0
```

Updates and Fetches For Varying Size Ranges

Updates Using Varied Replication Factors

Updates Using Varied Erasure Coding (With Partial vs Full Stripe)

DAOS CLIENT-SIDE METRICS API

- Counters
  - daos_metrics_alloc_cntrsbuf(daos_metrics_ucntrs_t **cntrs);
  - daos_metrics_get_cntrs(enum daos_metrics_cntr_grp mc_grp, daos_metrics_ucntrs_t *cntrs);
  - daos_metrics_free_cntrsbuf(daos_metrics_ucntrs_t *cntrs);
- Stats
  - daos_metrics_alloc_statsbuf(daos_metrics_ustats_t **stats);
  - daos_metrics_get_stats(enum daos_metrics_stats_grp ms_grp, daos_metrics_ustats_t *stats);
  - daos_metrics_free_statsbuf(daos_metrics_ustats_t *stats);
- Distribution
  - daos_metrics_alloc_distbuf(daos_metrics_udists_t **dist);
  - daos_metrics_get_dist(enum daos_metrics_dist_grp md_grp, daos_metrics_udists_t *dist);
  - daos_metrics_free_distbuf(daos_metrics_udists_t *dist);
- Misc
  - daos_metrics_reset();
  - daos_metrics_dump(FILE *fp);
  - daos_metrics_get_version(int *major, int *minor);

DAOS CLIENT-SIDE METRICS IMPLEMENTATION NOTES

- Counters
  - Global, and updated using HW atomics
  - Future Work
    - Replicate counters per thread basis to avoid HW atomics
    - Split failures into Retriable and Non-Retriable failures
    - Distinguish between User and libdaos initiated RPCs
- Stats
  - Accounting at thread-level via doubly linked list using Thread Local Storage
  - List protected by a lock tho field updates are not
  - Accumulation performed at thread exit
  - Future Work
    - Accounting of I/O part of compound RPC call
    - Explore a per-thread lock for data consistency on metrics get calls
    - Stats on i/o Latency
- Distribution
  - Predefined object classes represented only
  - Future Work
    - Accounting of I/O part of compound RPC calls
    - A better representation of full stripe vs partial stripe updates
    - Stats per distribution

DAOS ADMINISTRATION ENABLEMENT

- HPE Performance Cluster Manager (HPCM)
  - Server cluster management & monitoring via top-of-rack admin node
  - Can optionally manage compute nodes attached to DAOS as well
- DCM command set augments HPCM
  - Supports multiple logical DAOS systems / clusters within one physical cluster of HPE Proliant nodes
  - Programmatically sets up and tears down mini-clusters on subgroups of nodes
  - Operates / administers DAOS on each of the configured mini-clusters
  - Familiar to HPCM administrators using similar commands
- Cluster Setup Process
  - Compliant HW is pre-assembled onsite or in HPE Manufacturing with firmware / BIOS leveled / configured
  - Admin node's OS/HPCM is installed & added to customer admin network
  - OS distro to be deployed to DAOS servers is added to admin node's HPCM repository
  - BMC & server OS access MACs, and BMC login info are added
- Cluster deployment Process
  - Optionally configure a firewall/gateway from our private admin network thru the admin server to the customer network
  - Discover target nodes found in the config, and install a distro OS, verify the HW
  - Install the DCM package on the admin server
  - Create a DAOS repo on the admin server (may be from web or local DAOS repo mirror/copy)
  - Create and deploy DAOS server images
    - Clone the distro OS on the admin server
    - Install network drivers and DAOS RPMs into the DAOS server image
    - Deploy the image to all the running nodes
  - Use DCM commands to configure DAOS nodes for use
  - Later, DAOS upgrades can be deployed directly to running nodes without re-imaging

DEMAND FOR PERFORMANCE IS DRIVING A NEW STORAGE PARADIGM

Increasingly complex I/O patterns

- Resulting from the convergence of HPC, AI/ML and HPDA

Exascale is revealing POSIX limitations

- File locking, read/modify/write, metadata scaling …

PFS are limited by their design for HDD

- Spinning media data path designs limit effectiveness of solid state media

USE CASES FOR DAOS

Simulation & modeling

Small file I/O, large IOPS, low latency

Unaligned I/Os or shared file writes

Performance optimization via middleware integration

Artificial intelligence / Machine learning

Low latency, read-intensive I/Os

Machine learning training, streams processing

Support for AI frameworks

High performance data analytics

Volumes of small random read/write I/Os

Byte-granular access for unstructured and semi-structured data

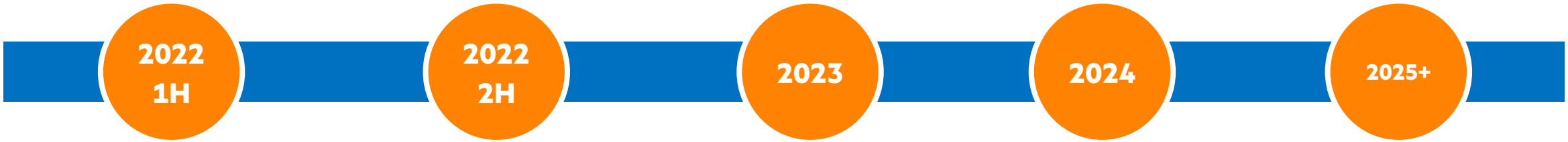
In-situ analysis

HPE TIMELINE FOR DAOS

Qualify
Test/Dev only
DAOS v2.0 to start
DL3xx Gen 10+
IB/Ethernet (Slingshot)
Supported by R&D

Integrate
Ready for production
DL3xx Gen 11
Full HPE Qualification
Supported by PointNext

Optimize
CXL support
Optimize for scale
Reduce HW cost
Reduce rack footprint



HPE Collaboration: **Market Development**

HPE Solution: **Biddable opportunities** → **Next gen exascale storage**

HPE MARKET DEVELOPMENT PROGRAMS FOR DAOS

Proof of Concept

- Customers who are curious about DAOS
- Desire a benchmarking experience with minimal commitment
- HPE provides lab environment and access
- Customer brings their data and their application
- 30 to 60 days duration

Early Adopter

- Customers who are ready to invest in DAOS
- Desire a test/dev environment for application development, performance tuning
- HPE vends the rack with servers, storage, network, cluster mgmt, and support
- Customer implements and maintains
- 1+ year duration

Both programs include collaboration with HPE R&D

- ✓ Register for participation via HPE sales and R&D
- ✓ Planning meetings with HPE R&D
- ✓ Implementation support
- ✓ Consultation and status updates with R&D
- ✓ Post-program debrief

EARLY ADOPTER EXAMPLE REFERENCE IMPLEMENTATION

- A Single-Rack Solution with up to:
  - Sixteen DL-360 Gen10 Plus; 128TB pmem, 4PB flash
  - Four 200Gb Switches (Mellanox or Slingshot)
  - ~1,400GBps/700GBps raw read/write throughput
  - ~150M/75M read/write operations per second
- Unbundled Repeatable Solution Delivery Method
  - Qualified hardware and software BOM
  - HPCM cluster management software
  - Lightweight installation / configuration scripting
  - Reference doc set: for field or factory integration
  - Customer system administration skills required
- Individual elements sold/supported separately

Up to 3 HPE Management Servers:
- DL-325 Gen10 single-socket

Up to 4 200GbE Switches:
- HPE Slingshot 1 (80 uplinks max)
- Mellanox QM8700 (72 uplinks max

Up to 32 HPE DAOS Server Max Cfg:
- DL-360 Gen10 Plus (Ice Lake)
- 8x Gen4 NVMe SSD 128TB ttl
- 16x Optane Memory 4TB ttl
- 200Gb NIC 2 each