Brookhaven National Laboratory

U.S. DEPARTMENT OF ENERGY

BNL ENDIT System: Efficient dCache Interface to HPSS

Zhenping (Jane) Liu

Scientific Data and Computing Center, Brookhaven National Laboratory

@BrookhavenLab

03/21/2024

The Challenges Leading to BNL ENDIT Development

- **Bottlenecks and Performance Issues:**
  - dCache staging limitations exposed during intense WLCG tape challenges.
  - Out-of-memory errors; non-functional servers during heavy restore requests of 120K or more.
  - Large amounts of duplicated restore requests submitted to HPSS Batch system due to PoolManager restore retry upon pool crash.
- **dCache's default driver interface with HPSS tape storage struggled with scalability.**
  - Scalability significantly limited due to the synchronous nature of this approach and the high resource demands resulting from periodic invocations of an external house-made HSM script for every file being staged.
  - Excessive concurrent PFTP connections from dCache pools caused HPSS PFTP gateway connection problems.

Motivation for ENDIT

- A scalable solution needed to efficiently handle increasing staging demands.
- BNL adapts NDGF's ENDIT system ideas, developing a customized version.
  - Specifically enhances HPSS tape storage interfacing for BNL's requirements.
  - Designed for robust performance in both staging from and migration to HPSS.
- Running in ATLAS dCache production for two years

Components of BNL ENDIT System

ENDIT Provider:

- Adapted from NDGF plugin with minor BNL-specific customizations (by Vincent Garonne)

ENDIT Daemons:

- **HPSSRetriever Daemon:**
  - Submits stage requests to the HPSS BATCH System.
  - Retrieves files from HPSS via PFTP after HPSS BATCH system processing.
- **HPSSArchiver Daemon:**
  - Responsible for flushing files from dCache tape write pools to HPSS.

Cron Jobs on HPSSBATCH:

- One job verifies new staging requests, ensuring only unique and non-duplicated requests are submitted to the BATCH system.
- Another job sends callbacks to the requesting pool(s) upon completion of file processing in the batch system.

dCache ENDIT Provider plugin

- Adapted from NDGF.
- Minor changes (additional metadata) to accommodate BNL's
- Mechanisms
  - Use predefined file actions within specified directories based on the status of requests.
    - File actions: creating, modifying, and deleting specific files in specific directories
  - WatchService: Watching file events triggered by those file actions
  - Java's and Google Guava's concurrent frameworks: handling file events asynchronously.

dCache ENDIT Provider plugin (Cont.)

- The ENDIT directory must reside on the same file system as the pool's data directory
- Several directories under the pool directory are recognized by the ENDIT provider:

./request: ENDIT provider sends stage/migration requests here

./in: ENDIT provider checks for completed staged files in this directory

./out: ENDIT provider places files to be written to tape here

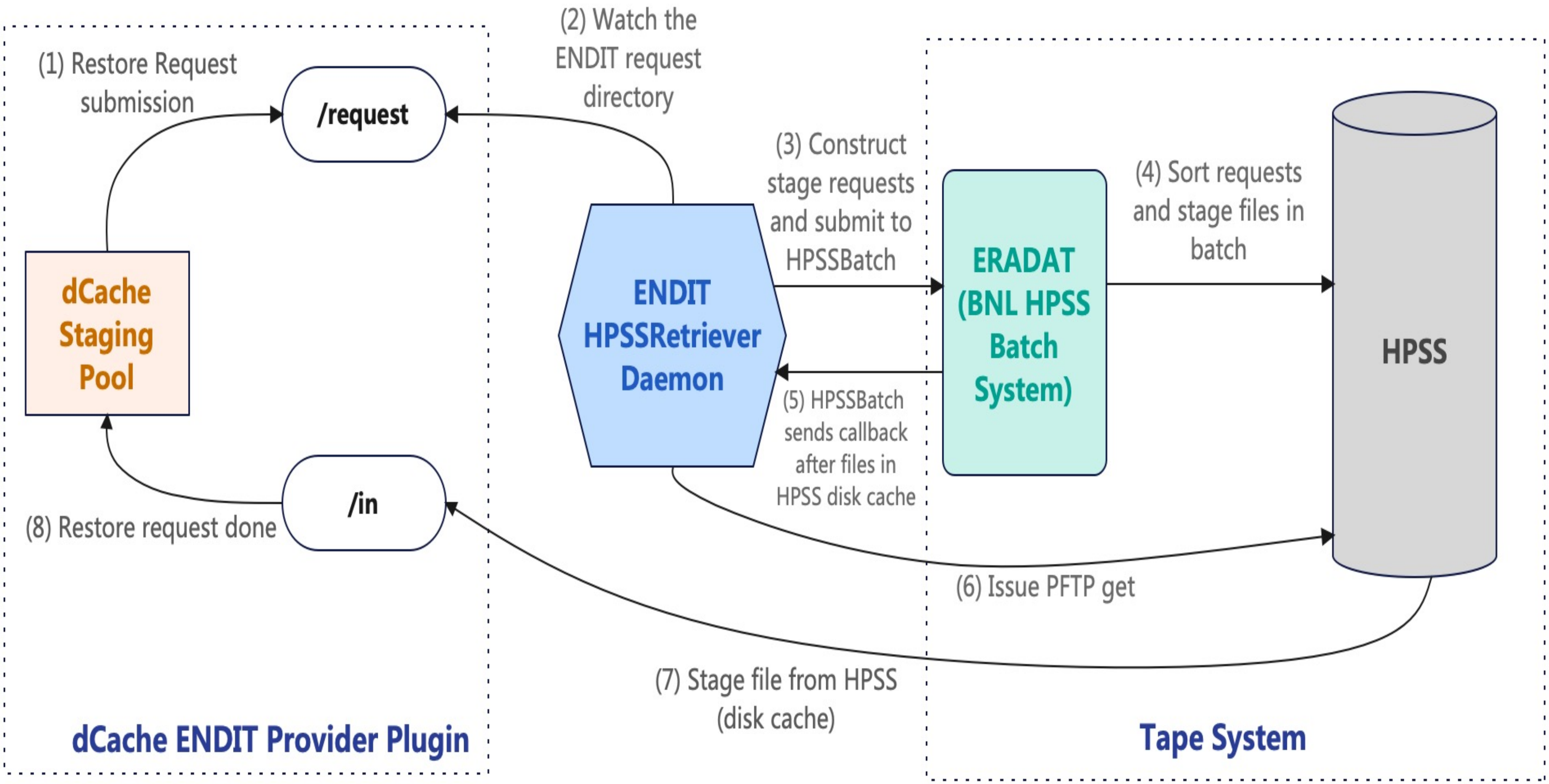
ENDIT HPSSRetriever daemon

- HPSSRetriever is a daemon to stage filles from HPSS to dCache pools
- New software developed by BNL
- **Workflow**

1. New stage requests are created under the ENDIT request directory by the ENDIT Provider.
2. The daemon monitors the ENDIT request directory and detects incoming new stage requests.
3. The daemon constructs stage requests and submits them to the ERADAT (HPSSBatch) request queue.
4. ERADAT (HPSSBatch) sorts requests and stages files in batches.
5. ERADAT (HPSSBatch) sends a callback after a file is in the HPSS disk cache.
6. The daemon checks the callback content. If it's good, the daemon invokes PFTP to retrieve the file from HPSS.
7. The file is staged from HPSS (disk cache) to the ENDIT ./in directory.
8. ENDIT Provider detects the data file, moves it from ./in to the pool data directory, and marks the end of the stage request

Brookhaven National Laboratory

Staging Workflow with ENDIT HPSSRetriever


(1) Restore Request submission
/request
(2) Watch the ENDIT request directory
(3) Construct stage requests and submit to HPSSBatch
(4) Sort requests and stage files in batch
dCache Staging Pool
ENDIT HPSSRetriever Daemon
ERADAT (BNL HPSS Batch System)
HPSS
(5) HPSSBatch sends callback after files in HPSS disk cache
/in
(8) Restore request done
(6) Issue PFTP get
(7) Stage file from HPSS (disk cache)
dCache ENDIT Provider Plugin
Tape System

Benefits of ENDIT HPSSRetriever

- Performance improvements on pool hosts

  - Eliminates polling on pool hosts, resulting in minimal load even with a high number of requests
  - Enables dCache to handle a large number of active staging requests simultaneously

- Provides flexible control over the maximum concurrent PFTP threads on each pool
- Prevents duplicated requests in HPSS Batch
- Reduces stress on PnfsManager and PoolManager due to non-polling nature

ENDIT HPSSArchiver Daemon

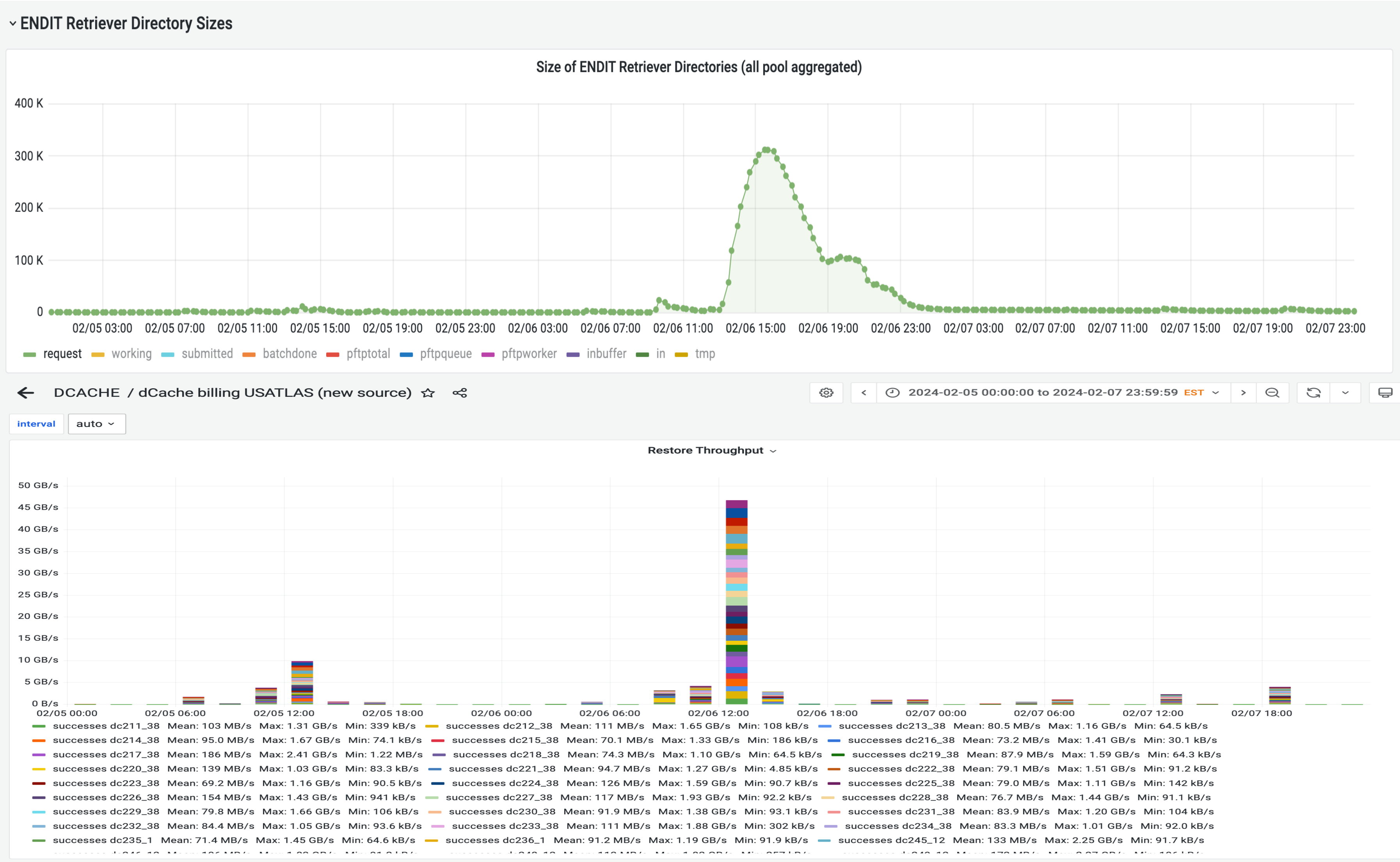
- HPSSArchiver is a daemon to flush dCache tape area files into HPSS.
- HPSSArchiver Daemon – Workflow

1. New flush requests are created under the ENDIT request directory by the ENDIT Provider.
2. The daemon monitors the ENDIT request directory and detects incoming new flush requests.
3. The daemon invokes PFTP to flush files to HPSS.
4. A file is flushed to HPSS successfully.
5. ENDIT Provider detects the completion of a flushing process and marks the end of the flush request.

Benefits of ENDIT HPSSArchiver

- Provides flexible control over the maximum concurrent PFTP threads on each pool
- A central place for a pool to control its migration requests. May add logic to do more (like hold requests for smart writing later ?).

Restore Performance



ENDIT Retriever Directory Sizes

Size of ENDIT Retriever Directories (all pool aggregated)

DCACHE / dCache billing USATLAS (new source)

2024-02-05 00:00:00 to 2024-02-07 23:59:59 EST

interval: auto

Restore Throughput

successes dc211_38 Mean: 103 MB/s Max: 1.31 GB/s Min: 339 kB/s
successes dc212_38 Mean: 111 MB/s Max: 1.65 GB/s Min: 108 kB/s
successes dc213_38 Mean: 80.5 MB/s Max: 1.16 GB/s Min: 64.5 kB/s
successes dc214_38 Mean: 95.0 MB/s Max: 1.67 GB/s Min: 74.1 kB/s
successes dc215_38 Mean: 70.1 MB/s Max: 1.33 GB/s Min: 186 kB/s
successes dc216_38 Mean: 73.2 MB/s Max: 1.41 GB/s Min: 30.1 kB/s
successes dc217_38 Mean: 186 MB/s Max: 2.41 GB/s Min: 1.22 MB/s
successes dc218_38 Mean: 74.3 MB/s Max: 1.10 GB/s Min: 64.5 kB/s
successes dc219_38 Mean: 87.9 MB/s Max: 1.59 GB/s Min: 64.3 kB/s
successes dc220_38 Mean: 139 MB/s Max: 1.03 GB/s Min: 83.3 kB/s
successes dc221_38 Mean: 94.7 MB/s Max: 1.27 GB/s Min: 4.85 kB/s
successes dc222_38 Mean: 79.1 MB/s Max: 1.51 GB/s Min: 91.2 kB/s
successes dc223_38 Mean: 69.2 MB/s Max: 1.16 GB/s Min: 90.5 kB/s
successes dc224_38 Mean: 126 MB/s Max: 1.59 GB/s Min: 90.7 kB/s
successes dc225_38 Mean: 79.0 MB/s Max: 1.11 GB/s Min: 142 kB/s
successes dc226_38 Mean: 154 MB/s Max: 1.43 GB/s Min: 941 kB/s
successes dc227_38 Mean: 117 MB/s Max: 1.93 GB/s Min: 92.2 kB/s
successes dc228_38 Mean: 76.7 MB/s Max: 1.44 GB/s Min: 91.1 kB/s
successes dc229_38 Mean: 79.8 MB/s Max: 1.66 GB/s Min: 106 kB/s
successes dc230_38 Mean: 91.9 MB/s Max: 1.38 GB/s Min: 93.1 kB/s
successes dc231_38 Mean: 83.9 MB/s Max: 1.20 GB/s Min: 104 kB/s
successes dc232_38 Mean: 84.4 MB/s Max: 1.05 GB/s Min: 93.6 kB/s
successes dc233_38 Mean: 111 MB/s Max: 1.88 GB/s Min: 302 kB/s
successes dc234_38 Mean: 83.3 MB/s Max: 1.01 GB/s Min: 92.0 kB/s
successes dc235_1 Mean: 71.4 MB/s Max: 1.45 GB/s Min: 64.6 kB/s
successes dc236_1 Mean: 91.2 MB/s Max: 1.19 GB/s Min: 91.9 kB/s
successes dc245_12 Mean: 133 MB/s Max: 2.25 GB/s Min: 91.7 kB/s

THANK YOU !

Brookhaven National Laboratory