Distributed **N**ame spac**E**

Jesse Hanley

Sarp Oral

Michael J. Brim

James Simmons

Dustin Leverman

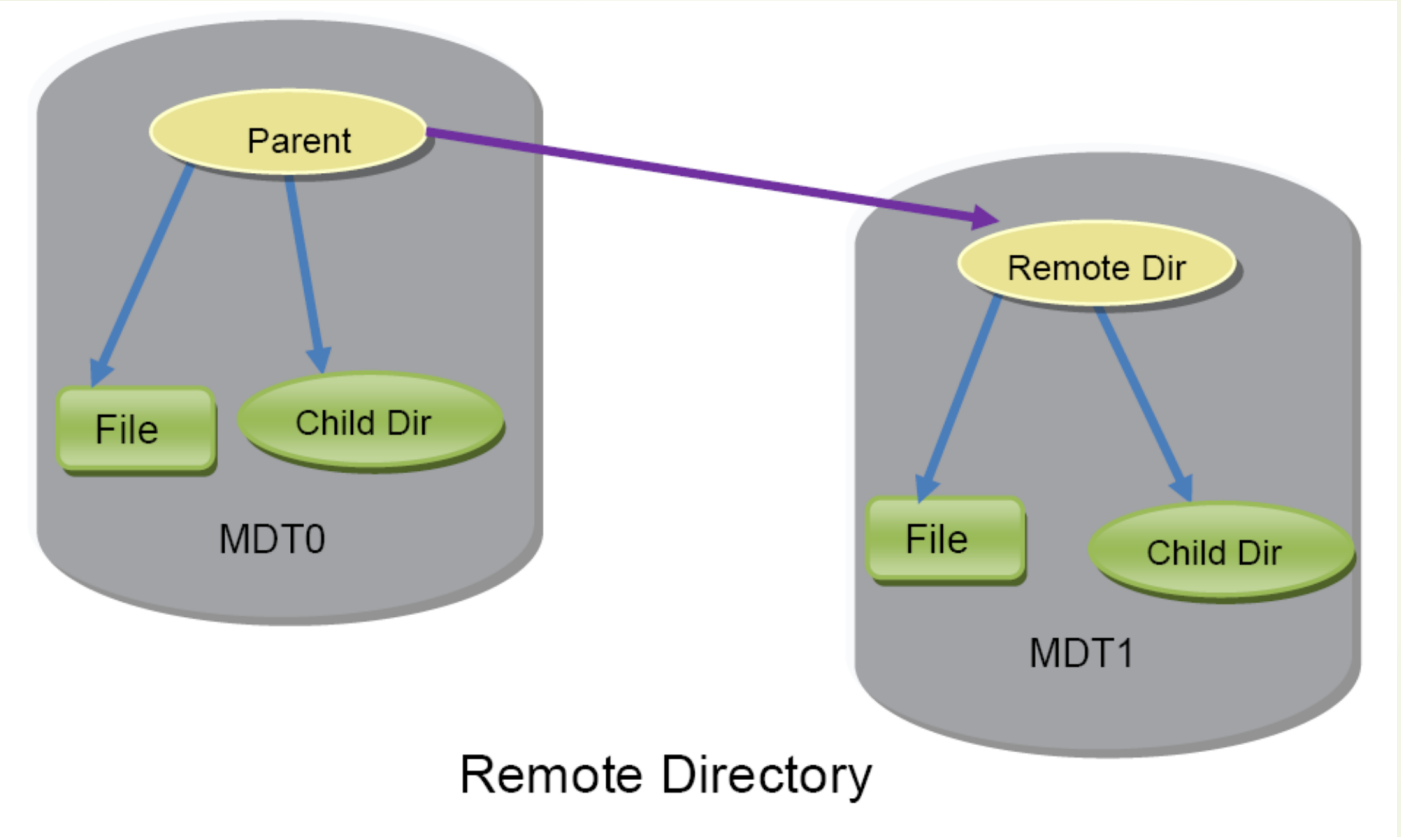
The need for scalable metadata

- Lustre metadata performance single pain point
  - Did not scale, hovered around 5,000 metadata operations per second
  - Suboptimal compared with the competitors
- OLCF is planning to move to Lustre 2.8 in production in 2016
  - First move clients to 2.8 then migrate the servers
  - Deployment will follow our standard testing procedure
    - Small scale testing
    - Medium scale testing on hardware similar to production
    - Large scale testing during dedicated test shot periods

Solution

- DNE is a development project funded by OpenSFS to improve metadata performance and scalability
- Development split into two parts
  - Phase 1: Remote directories on separate metadata servers and targets
    - Completed in Lustre 2.4 (2013)
  - Phase 2: Split directories over multiple targets, similar to spreading data over multiple OSTs
    - Completed in Lustre 2.8 (2016)

Phase 1 – Remote Directories

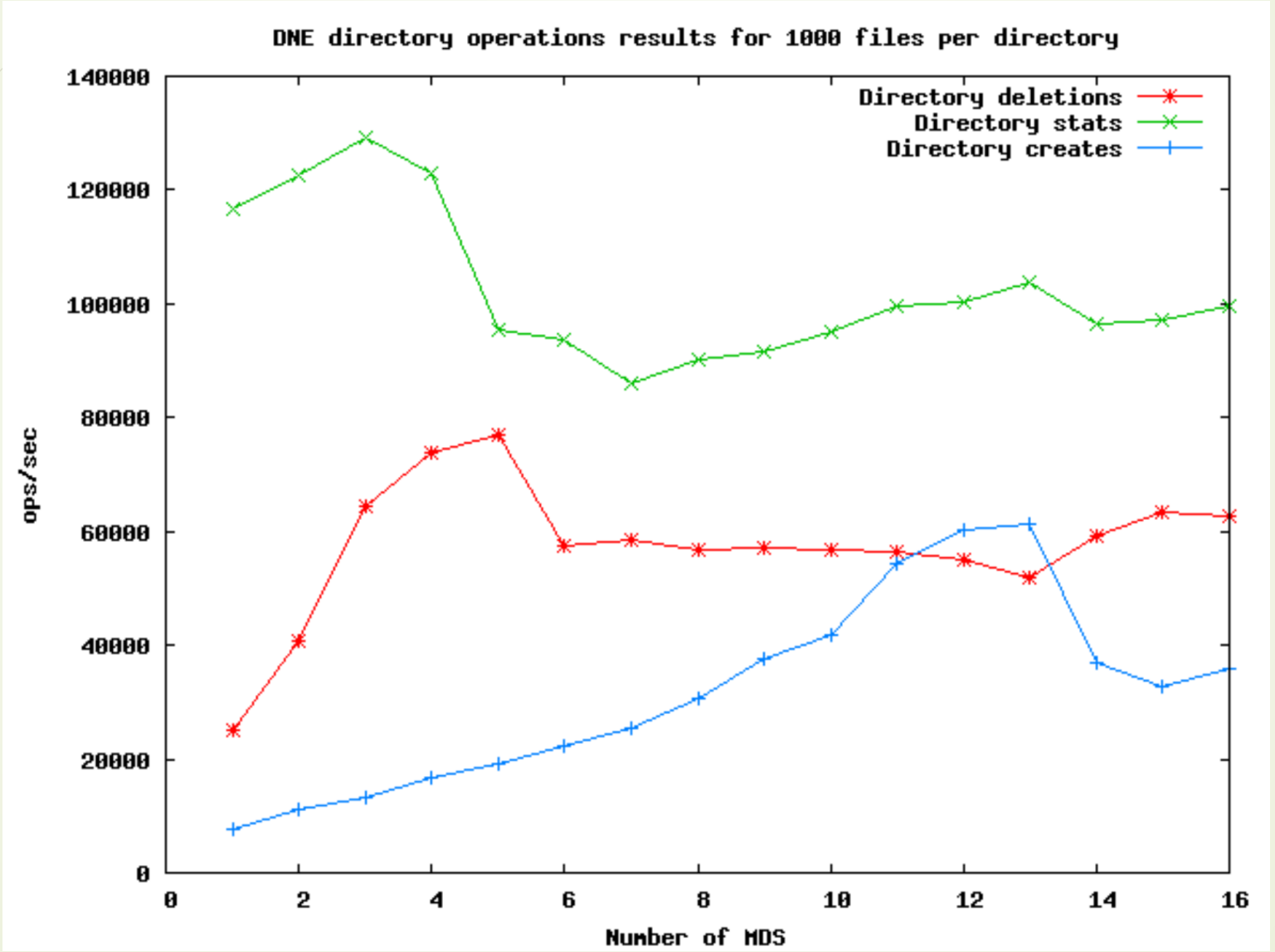


- Lustre subdirectories are distributed over multiple metadata targets (MDTs)
- By default:
  - Only admins can create children on remote MDTs
  - Only directories on `MDT0000` can contain directories that are not on the same MDT

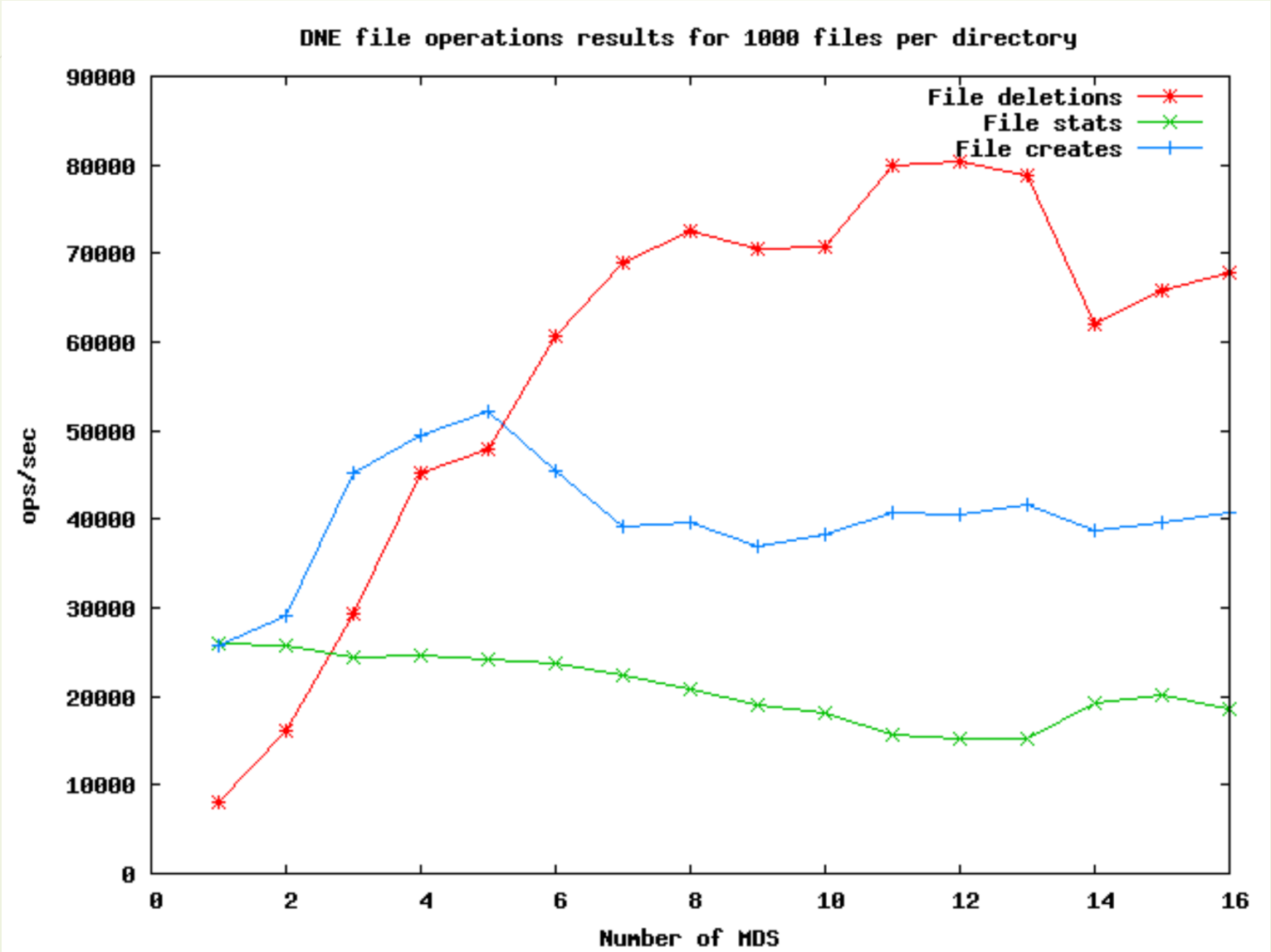
DNE Phase 1 – OLCF Evaluation Setup

- Lustre 2.5.3
- All tests were run on 1 MDT per MDS and number of MDS were scaled up
- MDT
  - TOSHIBA 600 GB 10k RPM SAS HDD
- MDS
  - Dual socket 8-core Intel Xeon 2.13 GHZ server with 94 GB DDR3 1066 RAM
- Clients
  - Arthur – a 22 node Cray XK7 running CLE 5.2.UP02 and Lustre 2.5.3
- IB FDR
- Mdtest
  - mdtest -I N -i 5 –d <lustre_path>/shared_Nk_$thread_count
  - Modified to support pre-created directories
- 56 OSTs on DDN SFA12KX using RAID 6 8+2 on NL SAS HITACHI 2 TB HDDs

DNE Phase 1 – Directory Scaling



DNE Phase 1 – File Scaling



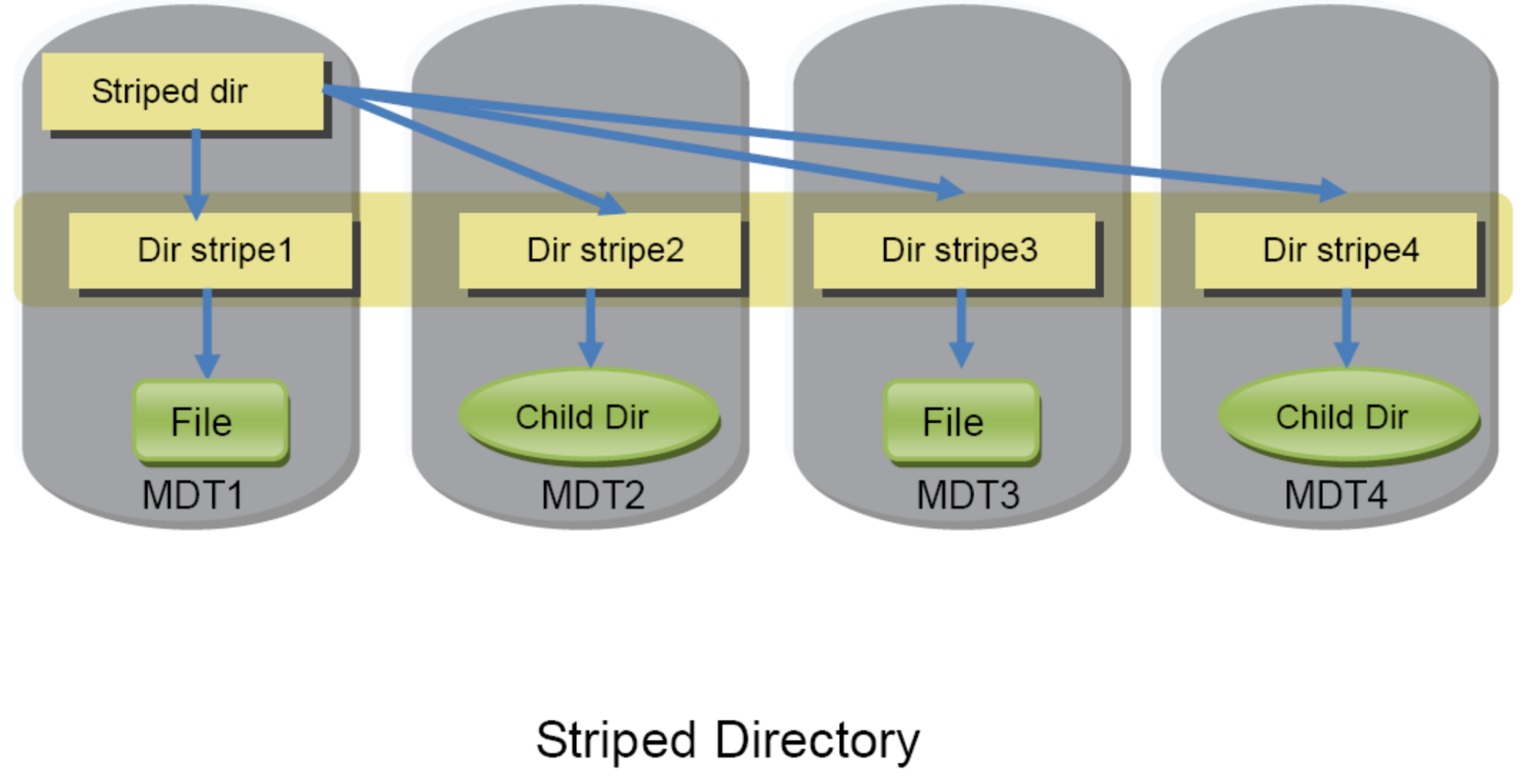
Analysis

- For a slight hit to directory stats, there's an increase close to double for directory creations and deletions.
  - These benefits appear with 4 MDTs, with diminishing returns thereafter
- File statistics follow a similar trend, with a penalty to stats and an increase to creations and deletions
  - File deletions scale strongly with the number of MDTs
  - File creations approach 40k operations per seconds
    - A potential bottleneck?

DNE Phase 2

- DNE P2 provides Lustre the capability to create striped directories in a similar fashion as bulk I/O striping across OSTs.
- Striping is determined on a hash calculated from the filename
- Code improvements from Phase 1

Phase 2 – Striped Directory



Striped Directory

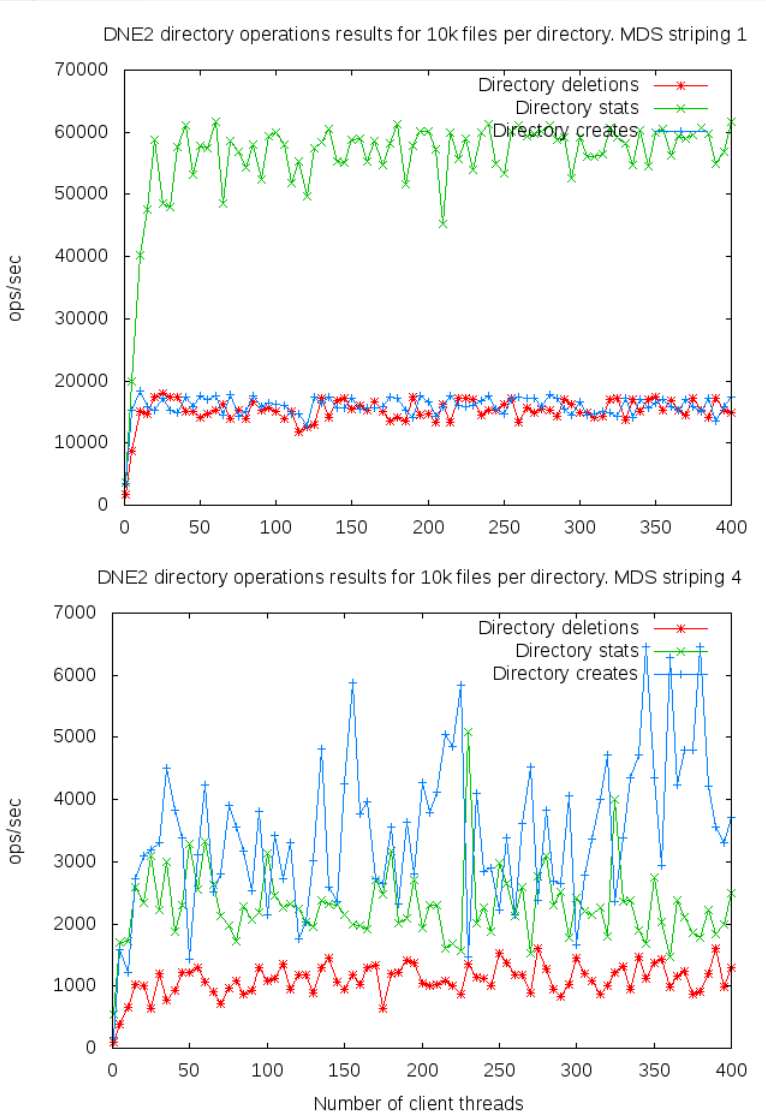
DNE Phase 2 – Use case analysis

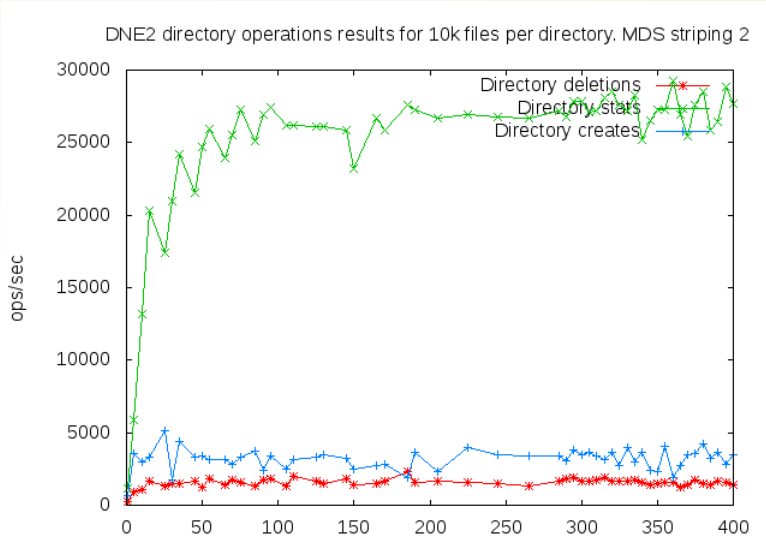
- A look into some of the downsides of scaling up metadata
- Analyzed performance scaling as directories are recursively striped
  - lfs setdirstripe -D
  - Similar to running a file-per-process IOR with files striped -1

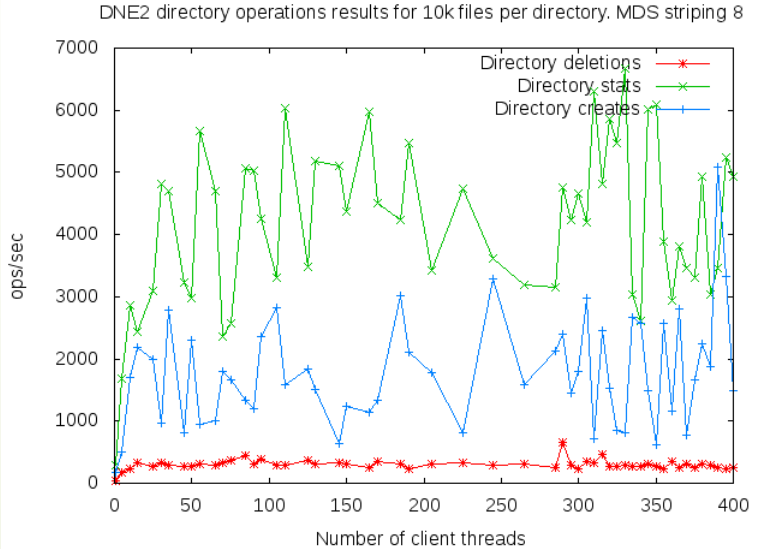
DNE Phase 2 – evaluation setup

- Lustre 2.8.0-rc2 servers
- All tests were run on 1 MDT per MDS and number of MDS were scaled up
- MDT
  - TOSHIBA 600 GB 10k RPM SAS HDD
- MDS
  - Dual socket 8-core Intel Xeon 2.13 GHZ server with 94 GB DDR3 1066 RAM
- Clients
  - Arthur – a 22 node Cray XK7 running CLE 5.2.UP02 and Lustre 2.7.90
- IB FDR
- Mdtest
  - mdtest -I N -i 5 –d <lustre_path>/dne2_1_mds_md_test/shared_Nk_$thread_count
  - Modified to support pre-created directories
- 56 OSTs on DDN SFA12KX using RAID 6 8+2 on NL SAS HITACHI 2 TB HDDs

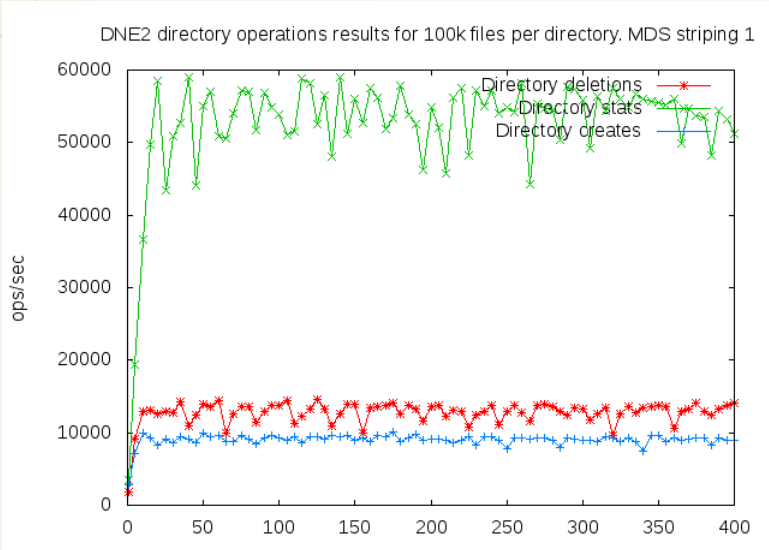
DNE Phase 2 - Directory Scaling – 10k files

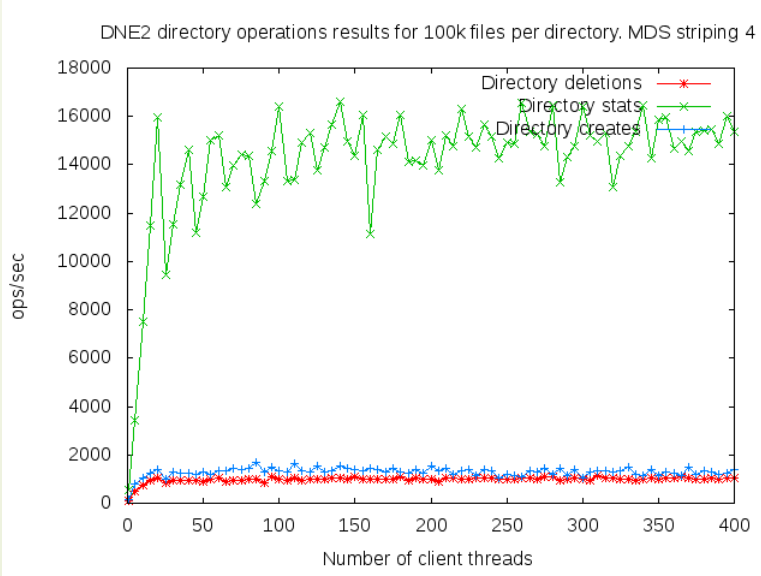


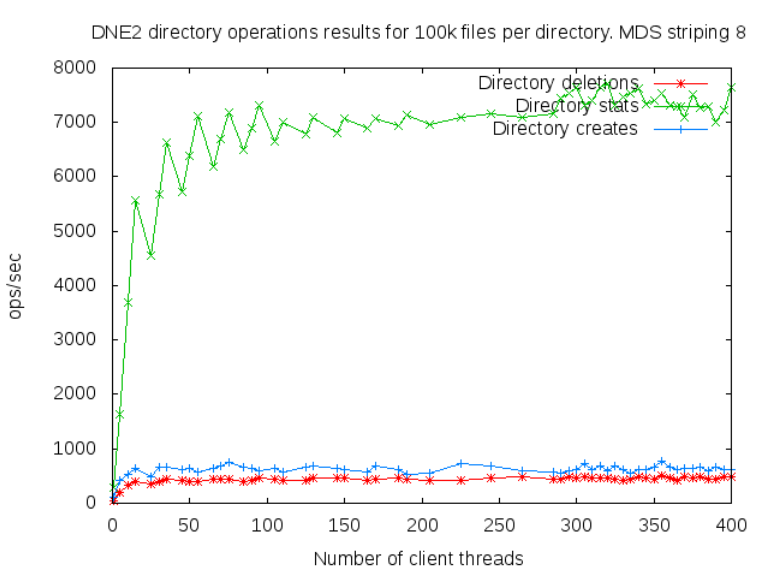




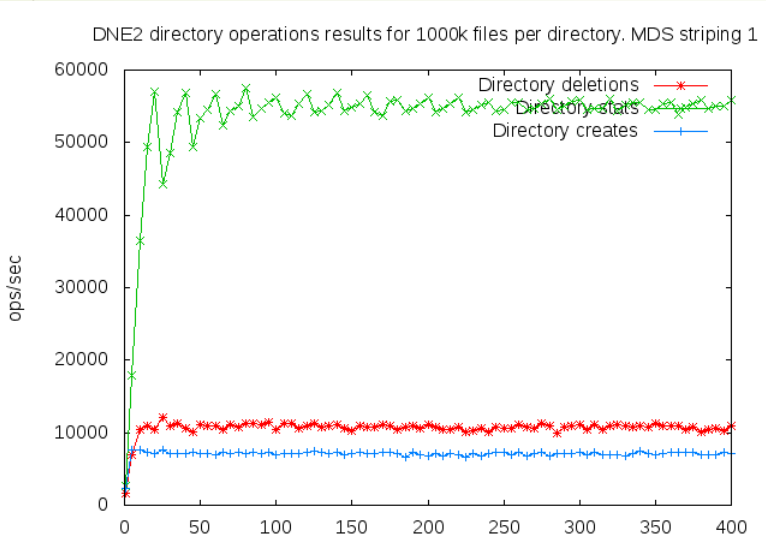
DNE Phase 2 - Directory Scaling – 100k files

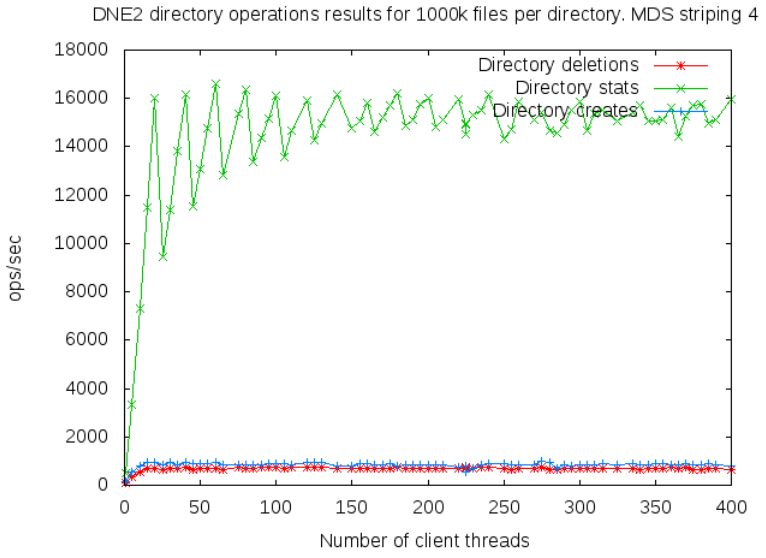






DNE Phase 2 - Directory Scaling – 1,000k files

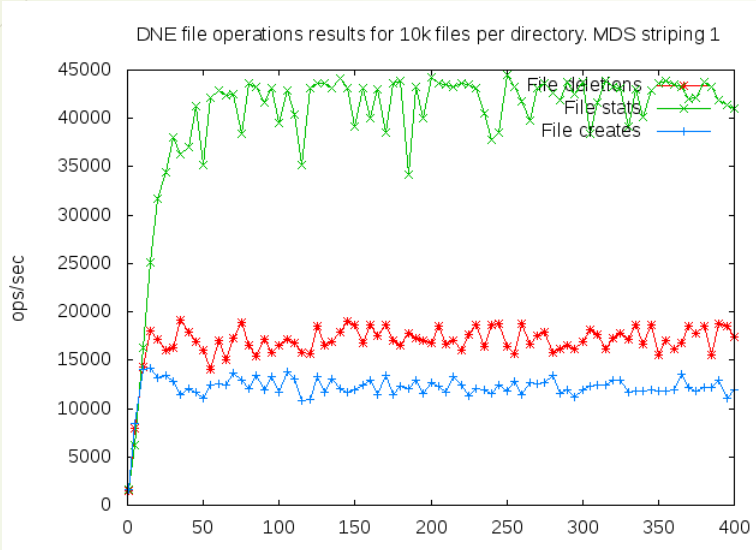


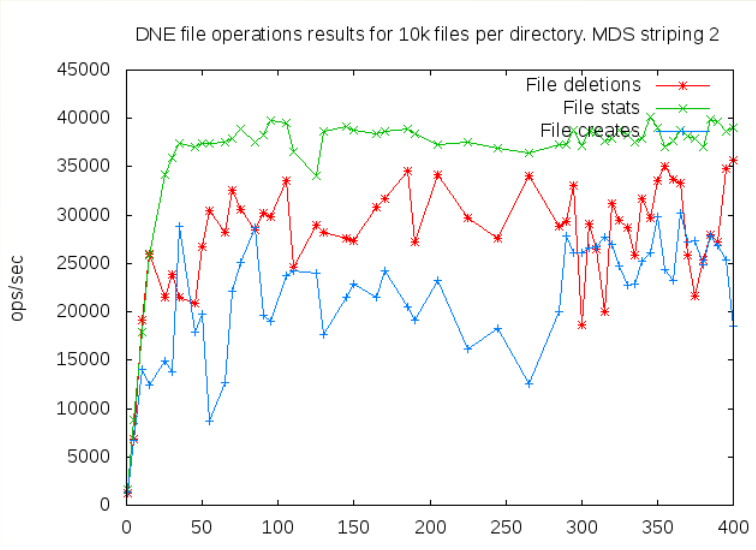


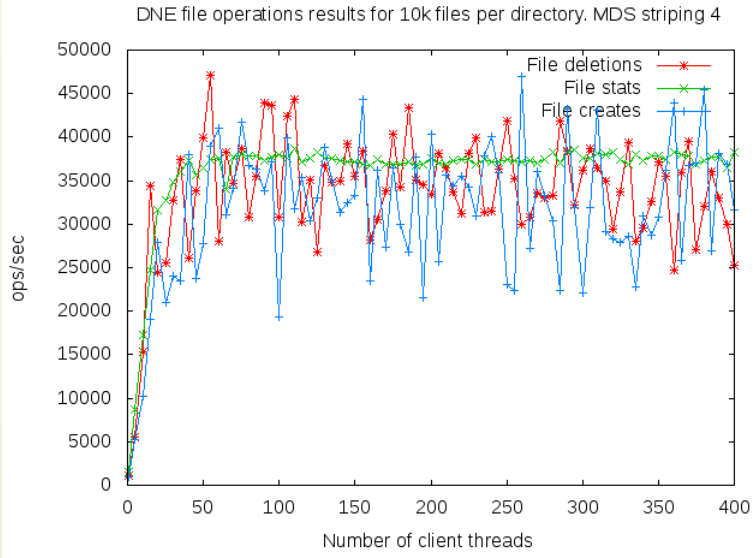
Analysis

- For directory operations scaling all directories on all MDS nodes is suboptimal
  - Appears to not scale well
  - Might be hitting an object creation bottleneck on OSTs
  - May also be hitting a client throughput or network limitation
- With a sufficient number of files per directory, directory metadata performance reaches an asymptotic limit

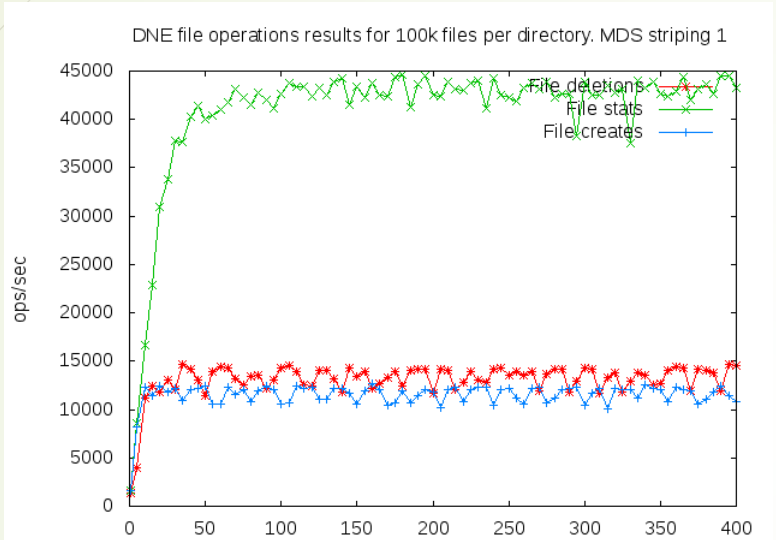
DNE Phase 2 - File Scaling – 10k files

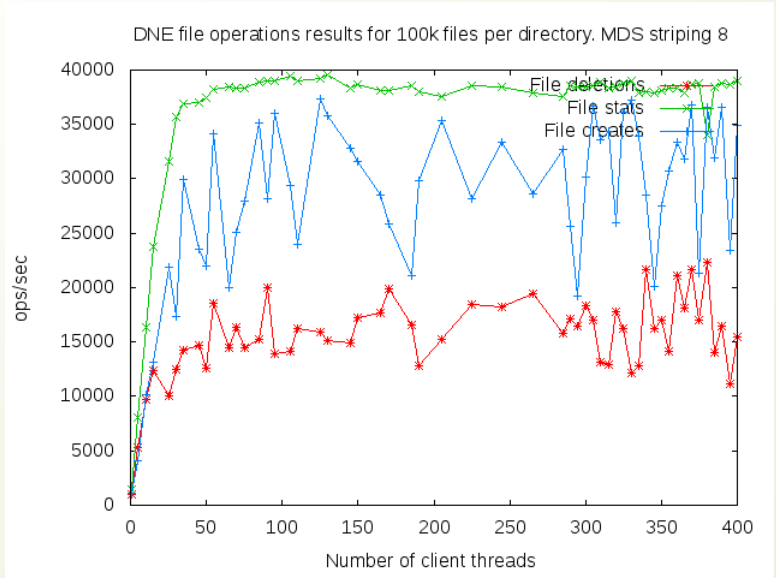




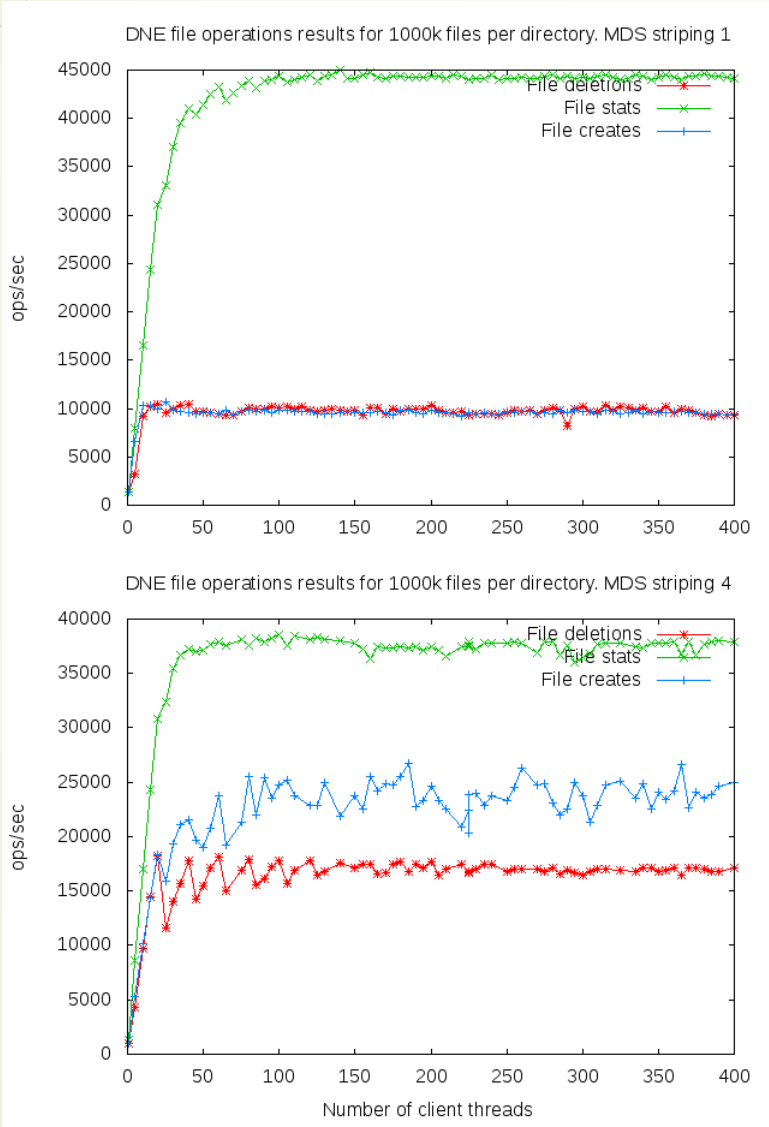


DNE Phase 2 - File Scaling – 100k files





DNE Phase 2 - File Scaling – 1,000k files



Analysis

- File stat operations per second are limited to ~40,000 ops/sec early on
  - Might be to limited number of clients in the test (ranged from 18 to 20 nodes)
- File creates and deletions benefit from scaling up
  - Nearly doubles, up to ~30,000 ops/sec
- Number files in a directory is effecting the scaling performance
  - More than 100,000 files in a single directory seems like suboptimal
- For 10,000 files in a directory, 4 MDS (1 MDT per each) seems like the sweet spot

What's next?

- "Hero" run on the same setup
  - Mdtest equivalent of an IOR
  - Single wide striped directory with single striped subdirectories
- Replicate OLCF's file/directory distribution for evaluation
- Repeat the tests with increased number of OSTs and clients

- What else you would like to see?

Thank You

Questions?

hanleyja@ornl.gov