New Lustre Features to Improve Lustre Metadata and Small-File Performance

John Fragalla
Bill Loewe
Torben Kling-Petersen

[jfragalla,bloewe,tpetersen]@cray.com

Agenda

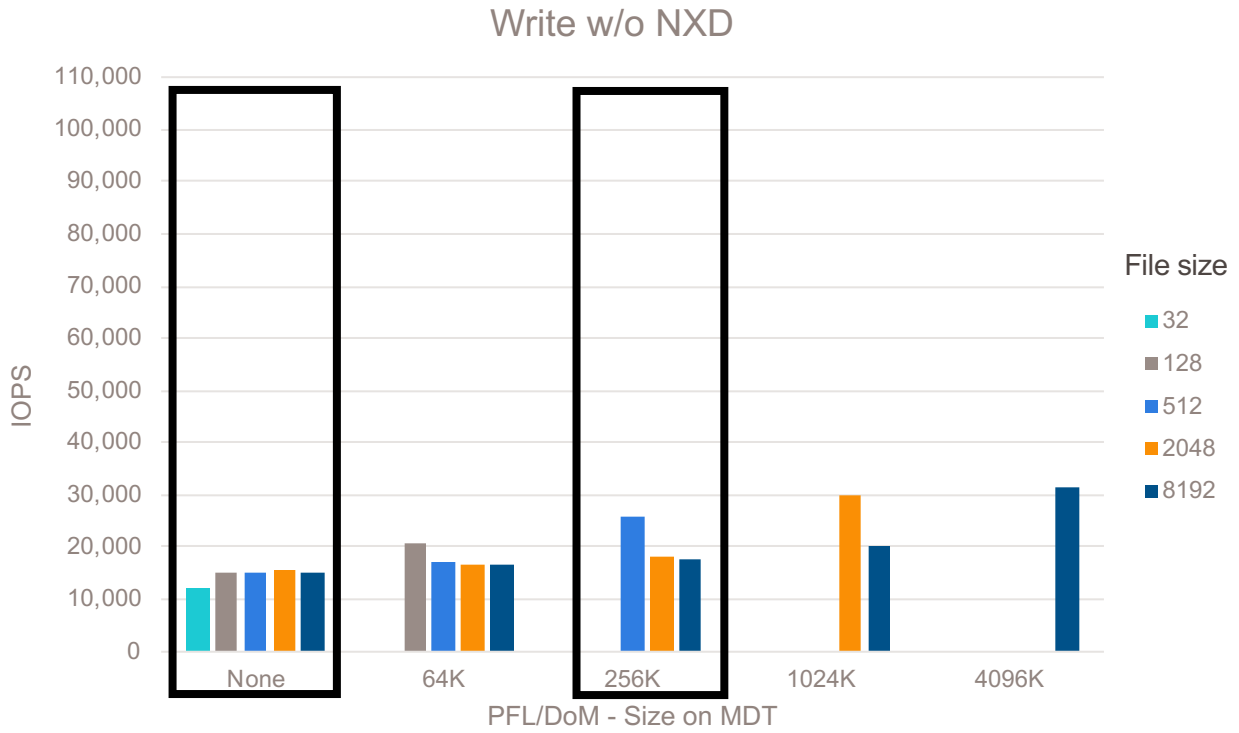
- New Lustre features and Cray NXD enabling flash to improve shared application performance
- Performance results to accelerate random 4KB I/O workloads
- Performance results isolating “noisy neighbor” applications
- Lustre metadata scaling
- Summary

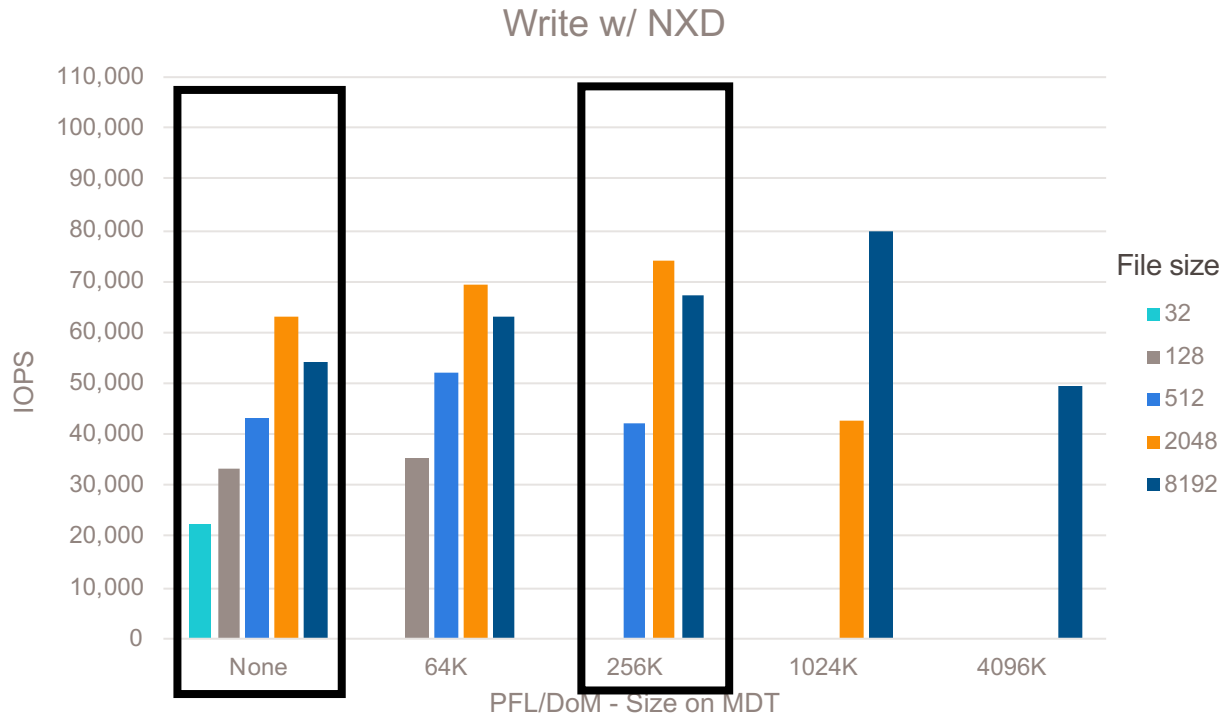
Transparent use of flash and HDD

- Recent features available in Lustre help automate I/O isolation and placement with transparent use of Flash and HDD Devices in a Single Namespace, to improve random small block I/O workloads, and improve shared I/O performance
  - Progressive File Layout (PFL) with Lustre storage pools
  - Data on Metadata (DoM)
  - Small block transparent cache filter driver - Cray NXD
  - Distributed NamespacE (DNE) 2 – clustered or sharded metadata

Improving random 4KB I/O performance

DoM + NXD write IOPS acceleration - random

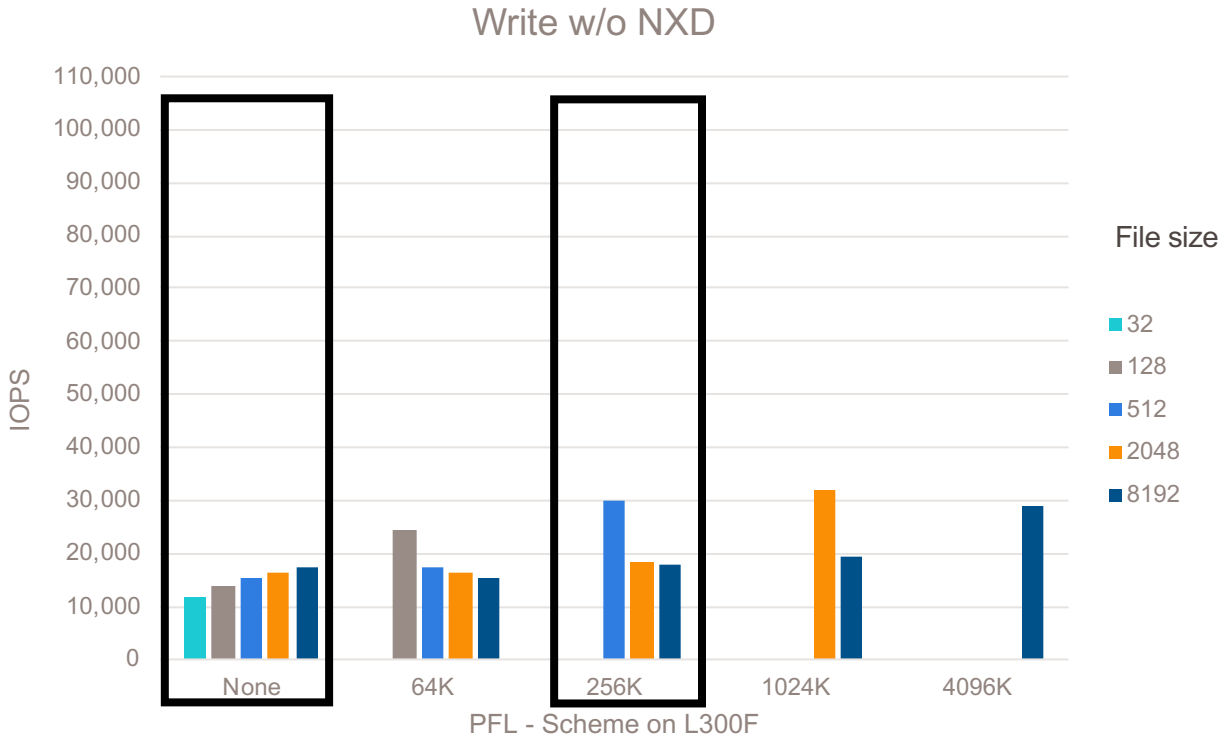


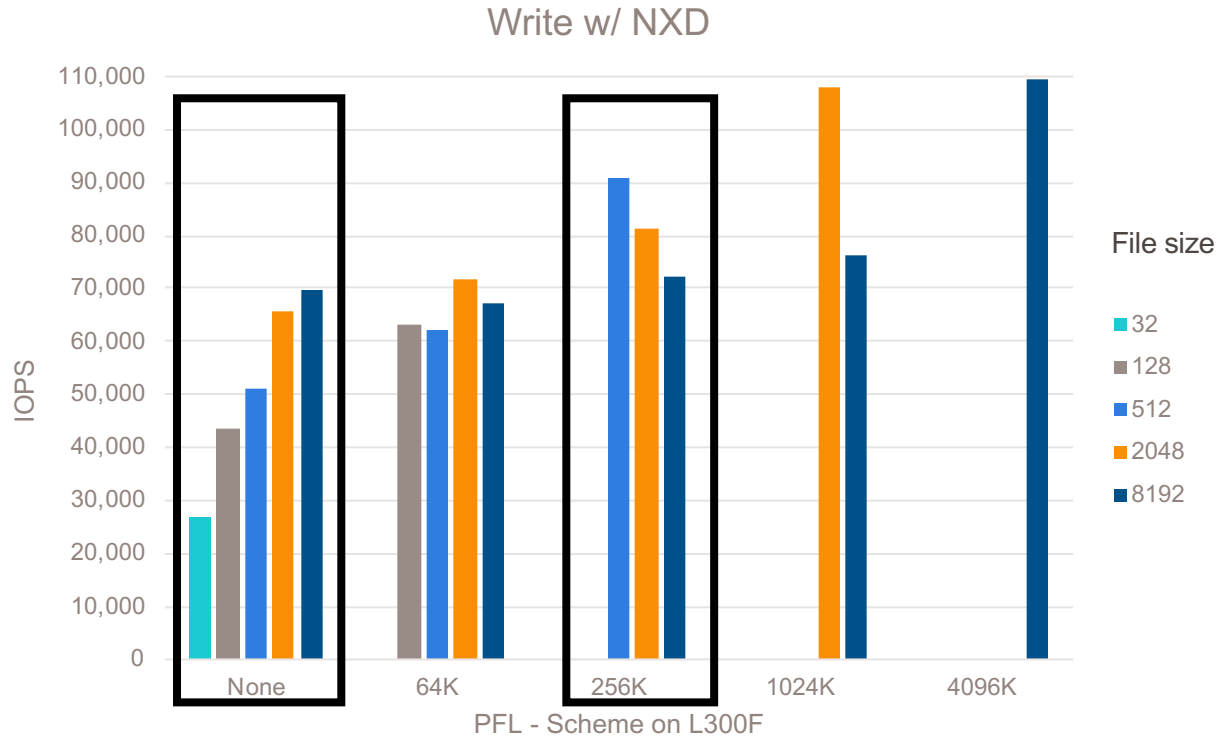


- Isolating the data showing spikes in write IOPS when DoM plus NXD are used in the benchmark
- On average, NXD accelerated Writes with combination of DoM and L300N with NXD by 3x to 4x speed up
- NXD with small File improvements using DoM with IOR Random Updates to a large File shows better performance than just using DoM

- IOR fandom, 4K block size, file size from 32K to 8192K

L300F + NXD write IOPS acceleration - random



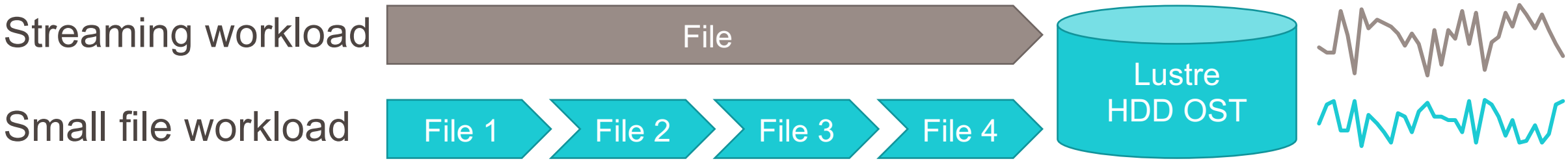


- Isolating the data showing spikes in write IOPS when L300F plus NXD are used in the benchmark
- On Average, NXD accelerated Writes with combination of L300F and L300N with NXD by 3.5x to 4.5x speed up
- NXD with small file improvements using L300F with IOR random updates to a large File shows better performance than just using L300F

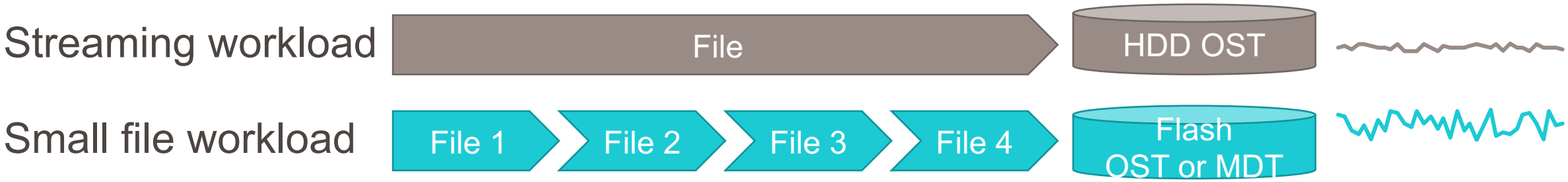
Solving ”noisy neighbor problem” with PFL

Lustre PFL “noisy neighbor” isolation

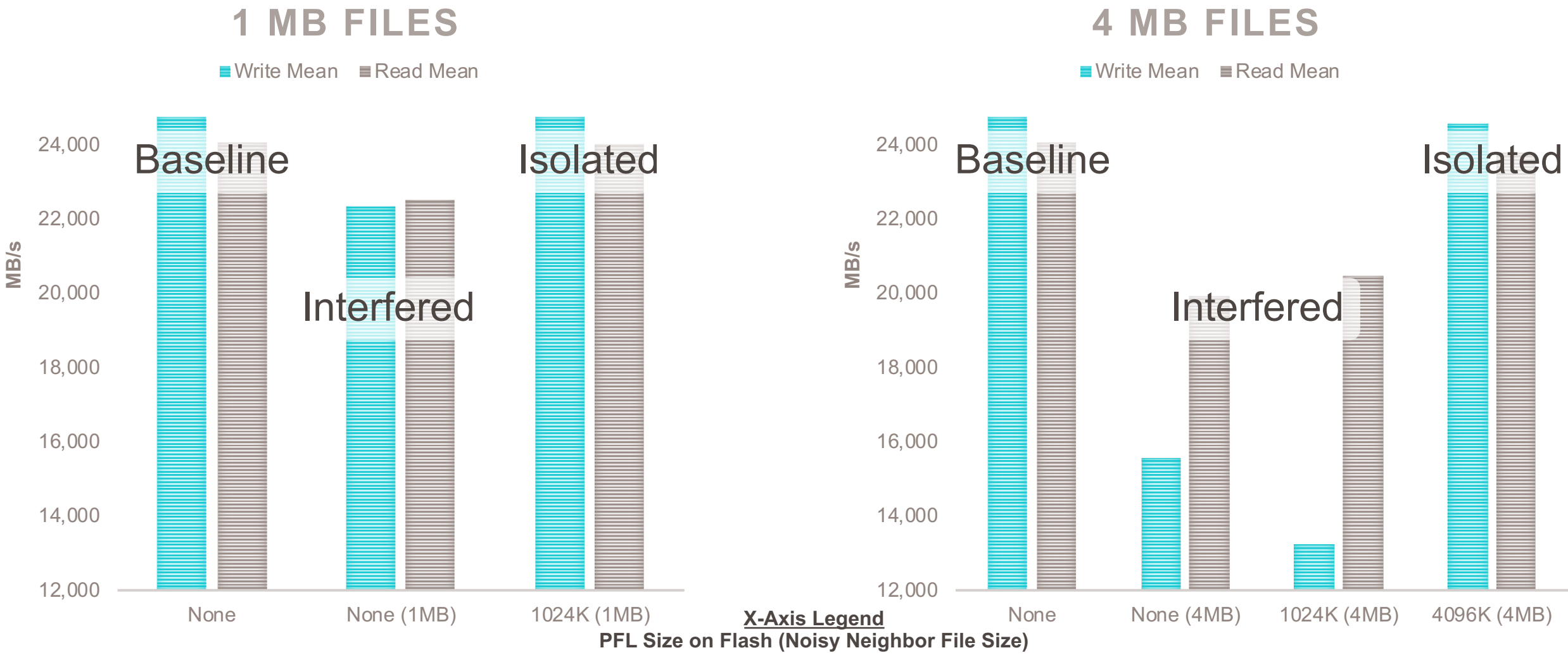
Two competing workloads on same resources



Two workloads separated using PFL



Lustre PFL “noisy neighbor” isolation
Flash tier (OST or DoM) -> HDD OST tier



PFL isolation for small I/O from streaming I/O improves performance

Metadata improvements

Metadata scaling performance – Lustre 2.11

		0KB files - Unique Directory			
DNE Striping	**Files/MDT**	**File Create/s**	**File Stat/s**	**File Read/s**	**File Unlink/s**
Sharded directory- 4x MDTs (No DoM)	1 048 576	167 611	753 885	602 834	346 796
Sharded directory - 4x MDTs (DoM 64K)	1 048 576	352 809	1 053 564	787 548	373 597

		0KB files - Shared Directory			
DNE Striping	**Files/MDT**	**File Create/s**	**File Stat/s**	**File Read/s**	**File Unlink/s**
1x MDT (No DoM)	1 048 576	76 578	181 320	152 441	80 390
Sharded directory - 4x MDTs (DoM 64K)	1 048 576	174 572	332 047	823 025	189 968

Conclusion/ Summary

Summary

- Lustre PFL preserves throughput on disk in a mixed workload that includes competing small file random
- Flash Targets improve IOPs compared to HDD
- Cray NXD plus Flash Targets improve IOPs for 4KB random I/O
- Sharded directory improved with DoM layout and better at automated optimization
- Sharded directory is preferred, and scales higher than single MDT for shared directory

THANK YOU

QUESTIONS?

[jfragalla,bloewe,tpetersen]@cray.com

cray.com

@cray_inc

linkedin.com/company/cray-inc-/