Leveraging Burst Buffer Coordination to Prevent I/O Interference

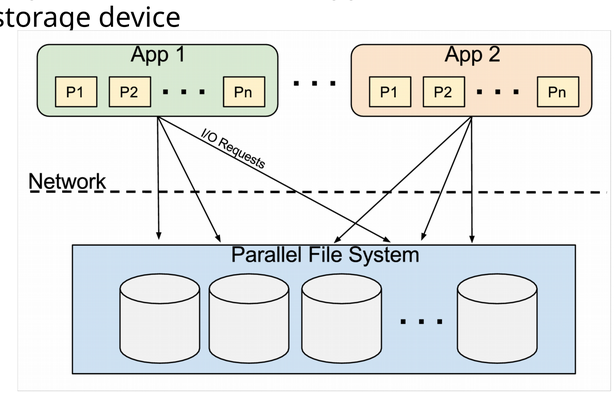
Anthony Kougkas[1,2] , Matthieu Dorier[2] , Rob Latham[2] , Rob Ross[2] , and Xian-He Sun[1]
[1]Illinois Institute of Technology, [2]Argonne National Laboratory

Abstract

Concurrent accesses to the shared storage resources in current HPC machines lead to severe performance degradation caused by I/O contention. We identify the key challenges to efficiently handling interleaved data accesses, and we propose a system-wide solution to optimize global performance.
We implemented and tested several **I/O scheduling** policies, including prioritizing specific applications by leveraging **burst buffers** to defer the conflicting accesses from another application and/or directing the requests to different storage servers inside the parallel file system infrastructure. We mitigate the negative effects of interference and optimize the performance **up to 2x** depending on the selected I/O policy.

I/O Interference

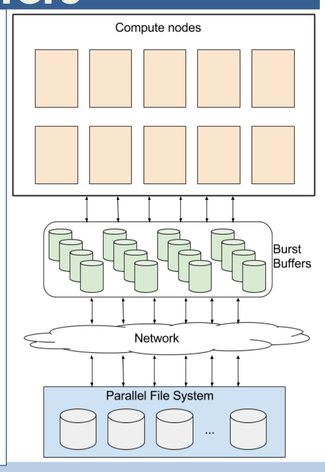
The phenomenon where multiple applications run concurrently and share the underlying storage system leading to severe degradation of the I/O bandwidth that each application experiences.
Major sources:
1) Network contention at the level of each storage server.
2) Poor scheduling decisions within the storage service.
3) Additional disk-head movements when interleaved requests from distinct applications reach the same storage device



Burst Buffers

- Burst buffers are an intermediate storage tier located between compute nodes and the underlying storage system.
- Main goal: to quickly absorb I/O requests from the computing elements and asynchronously issue them to the PFS, allowing the processing cores to return faster to computation.

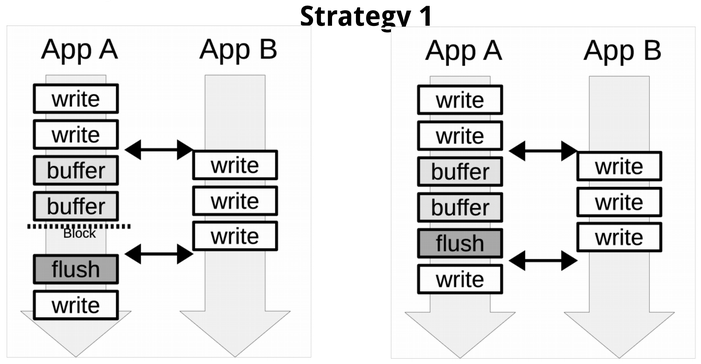
Utilize Burst Buffers as I/O traffic controllers



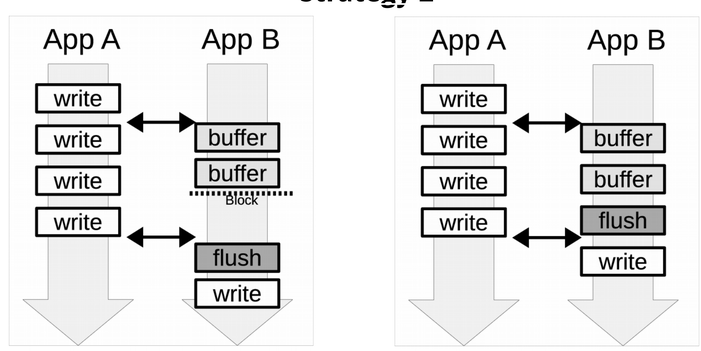
Approach

Coordinate data accesses to prevent applications to reach the underlying storage resources at the same time by imposing certain I/O policies implemented by the burst buffer layer.

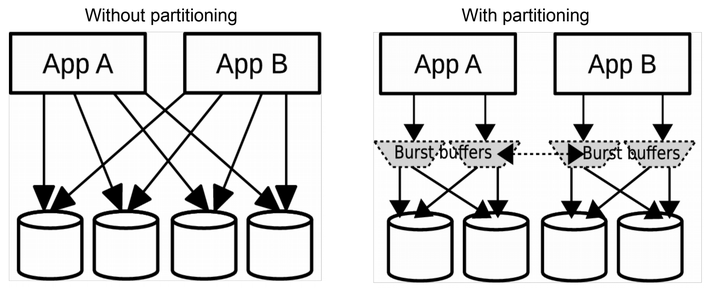
Strategy 1



Strategy 2

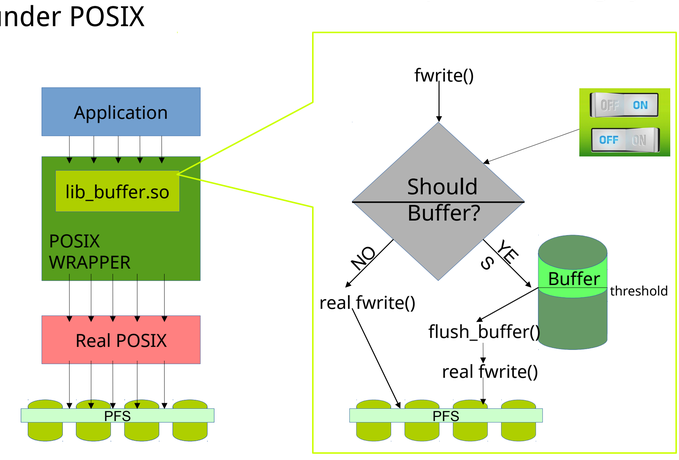


Strategy 3

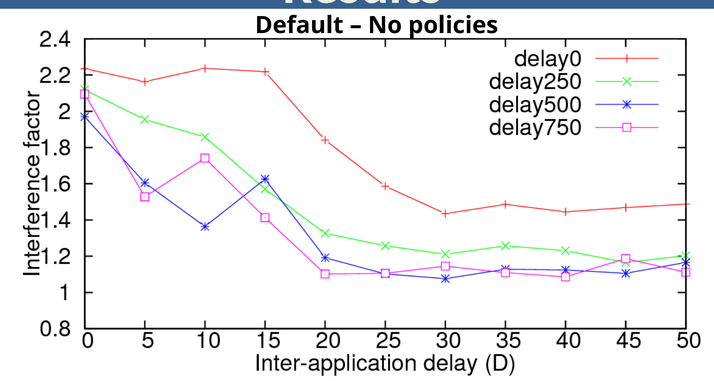


Implementation

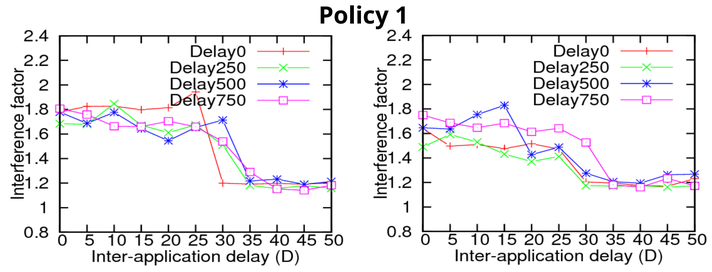
- Basic Buffered I/O (BBIO): a user-space buffering system under POSIX



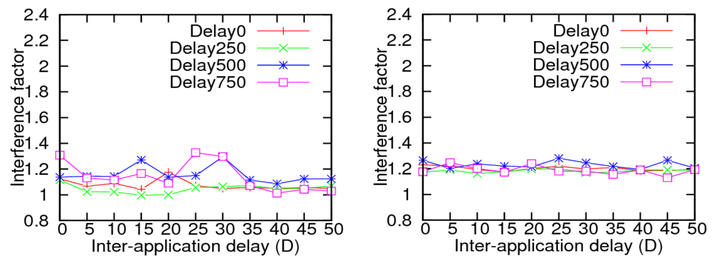
Results



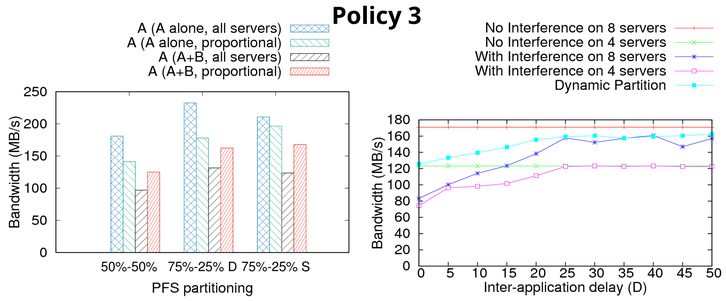
Policy 1



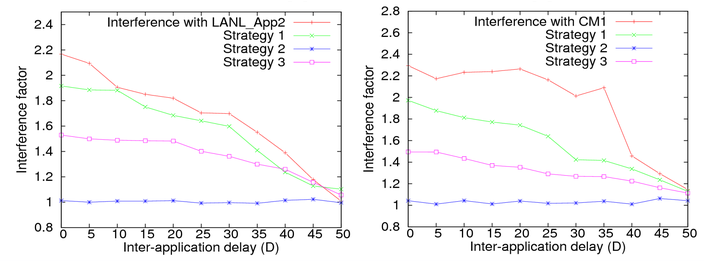
Policy 2



Policy 3



Real Applications – LANL_App 1



Conclusions

- We developed BBIO library which helps impose the proposed I/O policies to mitigate the performance degradation.
- Experimental results showed that we can achieve higher performance **up to 2x** depending on the selected policy.

Contact

Anthony Kougkas
Illinois Institute of Technology
Email: akougkas@hawk.iit.edu
Website: https://goo.gl/LkBz4N
Phone: 312-493-9389
Social media: https://www.linkedin.com/in/anthonykougkas