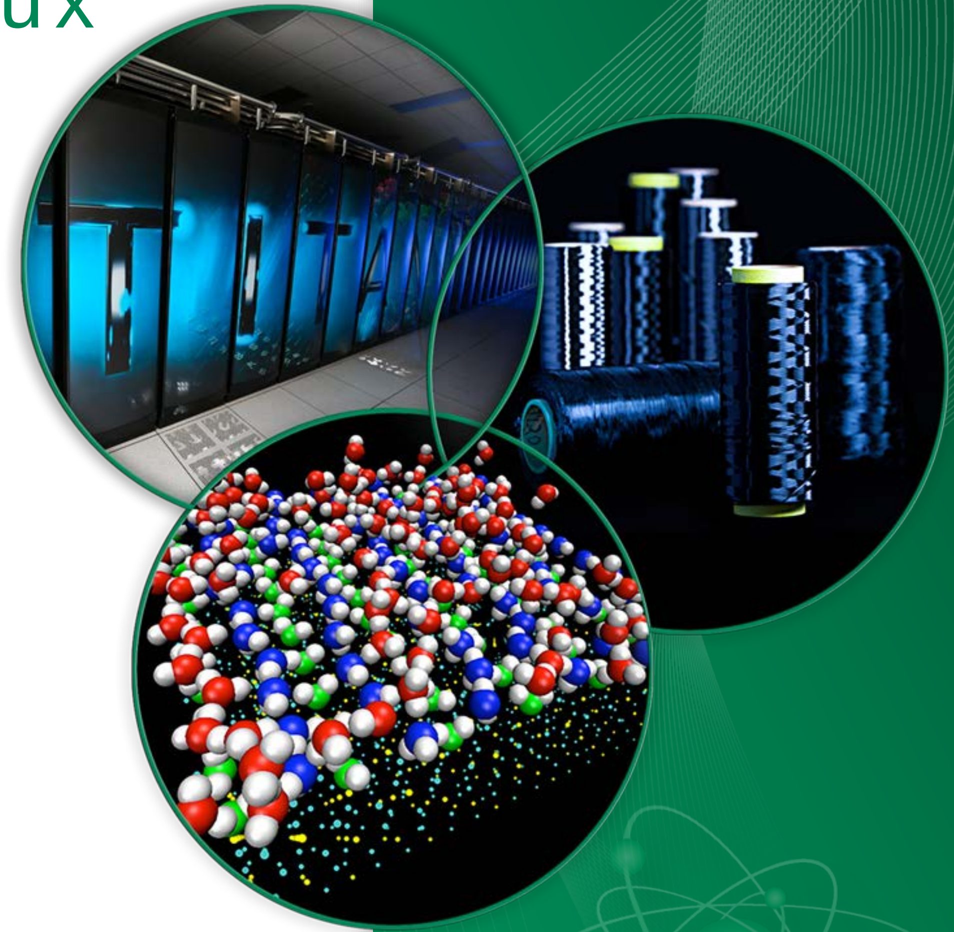
An Efficient Distributed Burst Buffer for Linux



Teng Wang

Weikuan Yu

Sarp Oral

Bradley W. Settlemyer*

Scott Atchley

LUG 2014

April 9, 2014

Checkpoint File Systems

- Largest factor driving the design of large-scale storage
  - Large fraction of the memory space of the entire application streaming to storage
  - Bursty I/O (write for 5 minutes, once an hour)
  - Almost entirely storage system write throughput limited
- Why can't scientific applications overlap computation and I/O?
  - Applications *should* overlap metadata operations
  - A time-step based simulation doesn't checkpoint after every time-step, just after some time-steps
  - Next program state depends on previous program state
  - Significant memory pressure
  - Coordination across multiple compute nodes makes these problems worse rather than better

Checkpoint Reality

- Checkpoints are wasted work if the machine never fails
  - Work done between last checkpoint and failure is also wasted
- Checkpoint as infrequently as possible
  - LCF MTTI – O(1 day)
  - Does not imply 1 checkpoint per day (4 hrs is rule of thumb)
- Next system
  - Desired MTTI – O(12-24 hours)
  - 90% job efficiency: 6 minutes to checkpoint, once an hour
  - May well checkpoint once per hour

Architecting Checkpoint Storage

	Widow (2008)	Atlas (2014)
LCF Clients	18,000	18,000
LCF Memory	300 TB	600 TB
IO Server count	192	288
IO Server Bandwidth	2.0 GB/s	4.2 GB/s
IS Server Capacity	56 TB	112 TB
System Capacity	10 PB	32 PB
System BW	240 GB/s	1 TB/s

Architecting Checkpoint Storage

	Widow (2008)	Atlas (2014)
LCF Clients	18,000	18,000
LCF Memory	300 TB	600 TB
IO Server count	192	288
IO Server Bandwidth	2.0 GB/s	4.2 GB/s
IS Server Capacity	56 TB	112 TB
System Capacity	10 PB	32 PB
System BW	240 GB/s	1 TB/s

IOS BW:Cap	0.036	0.038
System BW:Cap	0.024	0.031
IOS BW:LCF-M	0.006	0.007
Sys BW:LCF-M	0.8	1.7
All IOS BW:Sys BW	1.6	1.21

*ratios are in GB:TB or GB:GB, so unit-less, but scaled and not corrected for rounding, base-2, or base-10

Checkpointing Parameters

- Jaguar/Spider
  - 10.5 minutes to write 50% of RAM
  - 90% efficiency => checkpoint once each ~1.5 hours
  - 98% efficiency => checkpoint once each 12 hours
- Titan/Atlas
  - 5 minutes to write 50% of RAM
  - 90% eff. => checkpoint one each hour
  - 99% eff. => checkpoint once each 12 hours
- Next system
  - https://asc.llnl.gov/CORAL/
  - 10,000 – 50,000 nodes, at least 4PB of RAM
  - Desire 90% efficiency (6 minute checkpoint per hour)

Architecting Checkpoint Storage

	Widow	Atlas	Hypothetical
LCF Clients	18,000	18,000	*10,000 - 50,000*
LCF Memory	300 TB	600 TB	*4096 TB*
IO Server count	192	288	
IO Server Bandwidth	2.0 GB/s	4.2 GB/s	*8 GB/s*
IO Server Capacity	56 TB	112 TB	
System Capacity	10 PB	32 PB	
System BW	240 GB/s	1 TB/s	*5.5 TB/s*

IOS BW:Cap	0.036	0.038	*0.04*
System BW:Cap	0.024	0.031	
IOS BW:LCF-M	0.006	0.007	
Sys BW:LCF-M	0.8	1.7	
All IOS BW:Sys BW	1.6	1.21	*1.2*

ratios are in GB:TB or GB:GB, so unit-less, but scaled and not corrected for rounding, base-2, or base-10

Architecting Checkpoint Storage

	Widow	Atlas	Hypothetical
LCF Clients	18,000	18,000	*10,000 - 50,000*
LCF Memory	300 TB	600 TB	*4096 TB*
IO Server count	192	288	*825*
IO Server Bandwidth	2.0 GB/s	4.2 GB/s	*8 GB/s*
IO Server Capacity	56 TB	112 TB	*200 TB*
System Capacity	10 PB	32 PB	*165 PB*
System BW	240 GB/s	1 TB/s	*5.5 TB/s*
IOS BW:Cap	0.036	0.038	*0.04*
System BW:Cap	0.024	0.031	*0.033*
IOS BW:LCF-M	0.006	0.007	*0.002*
Sys BW:LCF-M	0.8	1.7	*1.34*
All IOS BW:Sys BW	1.6	1.21	*1.2*

ratios are in GB:TB or GB:GB, so unit-less, but scaled and not corrected for rounding, base-2, or base-10

Architecting Checkpoint Storage

	Widow	Atlas	Hypothetical
LCF Clients	18,000	18,000	*10,000 - 50,000*
LCF Memory	300 TB	600 TB	*4096 TB*
IO Server count	192	288	*688*
IO Server Bandwidth	2.0 GB/s	4.2 GB/s	*8 GB/s*
IO Server Capacity	56 TB	112 TB	*200 TB*
System Capacity	10 PB	32 PB	*138 PB*
System BW	240 GB/s	1 TB/s	*5.5 TB/s*

IOS BW:IOS Cap	0.036	0.038	*0.04*
System BW:Sys Cap	0.024	0.031	*0.039*
IOS BW:LCF-M	0.006	0.007	*0.002*
Sys BW:LCF-M	0.8	1.7	*1.34*
All IOS BW:Sys BW	1.6	1.21	*1.0*

ratios are in GB:TB or GB:GB, so unit-less, but scaled and not corrected for rounding, base-2, or base-10

Architecting Checkpoint Storage

	Widow	Atlas	Expected/CFS
LCF Clients	18,000	18,000	*10,000 - 50,000*
LCF Memory	300 TB	600 TB	*M TB (>4096)*
IO Server count	192	288	C
IO Server Bandwidth	2.0 GB/s	4.2 GB/s	
IO Server Capacity	56 TB	112 TB	*30 M / C*
System Capacity	10 PB	32 PB	*30 M*
System BW	240 GB/s	1 TB/s	*1.5 M / 3600 TB/s*
IOS BW:IOS Cap	0.036	0.038	
System BW:Sys Cap	0.024	0.031	
IOS BW:LCF-M	0.006	0.007	
Sys BW:LCF-M	0.8	1.7	
All IOS BW:Sys BW	1.6	1.21	

ratios are in GB:TB or GB:GB, so unit-less, but scaled and not corrected for rounding, base-2, or base-10

Architecting Checkpoint Storage

	Widow	Atlas	Expected/CFS
LCF Clients	18,000	18,000	*10,000 - 50,000*
LCF Memory	300 TB	600 TB	*M TB (>4096)*
IO Server count	192	288	*C*
IO Server Bandwidth	2.0 GB/s	4.2 GB/s	
IO Server Capacity	56 TB	112 TB	*30 M / C*
System Capacity	10 PB	32 PB	*30 M*
System BW	240 GB/s	1 TB/s	*1.5 M / 3600 TB/s*
IOS BW:IOS Cap	0.036	0.038	*0.04*
System BW:Sys Cap	0.024	0.031	
IOS BW:LCF-M	0.006	0.007	
Sys BW:LCF-M	0.8	1.7	
All IOS BW:Sys BW	1.6	1.21	*1.2*

ratios are in GB:TB or GB:GB, so unit-less, but scaled and not corrected for rounding, base-2, or base-10

Architecting Checkpoint Storage

	Widow	Atlas	Expected/CFS
LCF Clients	18,000	18,000	*10,000- 50,000*
LCF Memory	300 TB	600 TB	*4096 TB*
IO Server count	192	288	*226*
IO Server Bandwidth	2.0 GB/s	4.2 GB/s	*9 GB/s*
IO Server Capacity	56 TB	112 TB	*543 TB*
System Capacity	10 PB	32 PB	*123 PB*
System BW	240 GB/s	1 TB/s	*1.7 TB/s**

IOS BW:IOS Cap	0.036	0.038	*0.04*
System BW:Sys Cap	0.024	0.031	*0.013*
IOS BW:LCF-M	0.006	0.007	*0.002*
Sys BW:LCF-M	0.8	1.7	*0.42*
All IOS BW:Sys BW	1.6	1.21	*1.2*

ratios are in GB:TB or GB:GB, so unit-less, but scaled and not corrected for rounding, base-2, or base-10

*It may be only be required to store 1 of every 3 checkpoints, resulting in a smaller file system load

Architecting Checkpoint Storage

	Widow	Atlas	Expected/CFS	CFS Burst
LCF Clients	18,000	18,000	*10,000- 50,000*	
LCF Memory	300 TB	600 TB	*4096 TB*	
IO Server count	192	288	*226*	*100-200*
IO Server Bandwidth	2.0 GB/s	4.2 GB/s	*9 GB/s*	*29-57 GB/s*
IO Server Capacity	56 TB	112 TB	*543 TB*	*60-130 TB*
System Capacity	10 PB	32 PB	*123 PB*	*13 PB*
System BW	240 GB/s	1 TB/s	*1.7 TB/s*	*5.7 TB/s*

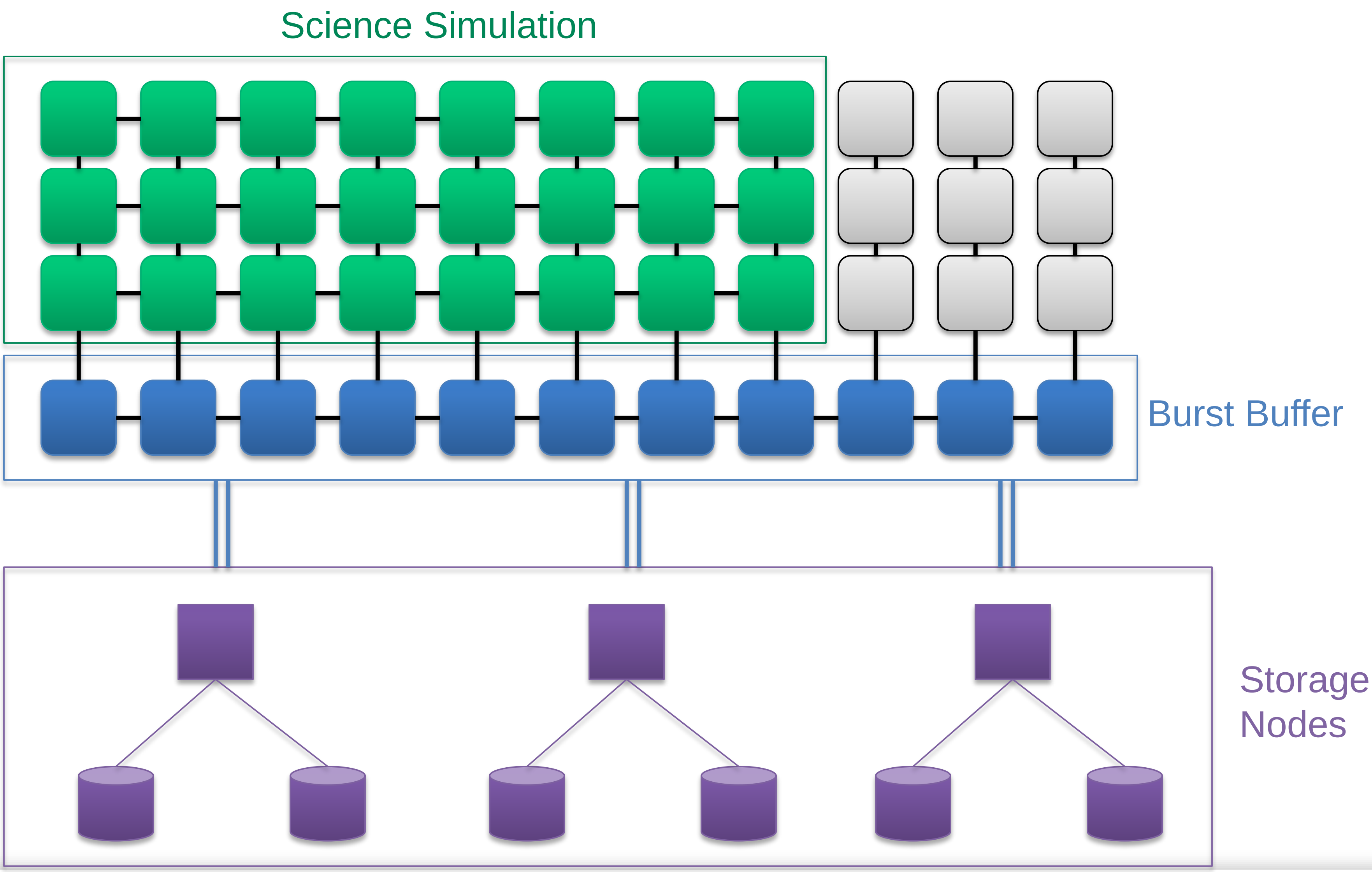
IOS BW:IOS Cap	0.036	0.038	*0.04*	*0.43 – 0.48*
System BW:Sys Cap	0.024	0.031	*0.013*	*0.43*
IOS BW:LCF-M	0.006	0.007	*0.002*	*0.007 – 0.013*
Sys BW:LCF-M	0.8	1.7	*0.42*	*1.1*
All IOS BW:Sys BW	1.6	1.21	*1.2*	*1.0*

ratios are in GB:TB or GB:GB, so unit-less, but scaled and not corrected for rounding, base-2, or base-10

Checkpoint Storage Design

- Burst buffer needs to provide roughly 10x bandwidth per byte of capacity (29 – 57 GB/s)
  - 2 – 4 DDR3 modules (likely 1 socket for DDR4)
  - 31 – 60 lanes PCIe 3.0
  - 16 – 30 lanes PCIe 4.0
- Burst buffer must have fast ingress
  - 5 times faster than today
  - Must be able to overlap ingress and egress?!
- Picture of the file system is unclear
  - Will it have some excess performance?
  - Switch to a different media (e.g. 5400 RPM SATA)?
  - Burst buffer just one of many consumers?

Burst Buffer Environment



Burst Buffer via Memcache

- Prototype with existing software
  - Memcache has a burst buffer-style semantic
    - Pools of storage servers
    - Socket-based communication
    - Stores Key-Value pairs in 2-level cache
    - Servers do not communicate
  - Simple
  - CAP friendly
- CAP theorem
  - Typical formulation: choose 2 of consistency, availability, and partition tolerance
  - More subtle: to provide a great deal of availability and partition tolerance, must sacrifice some consistency

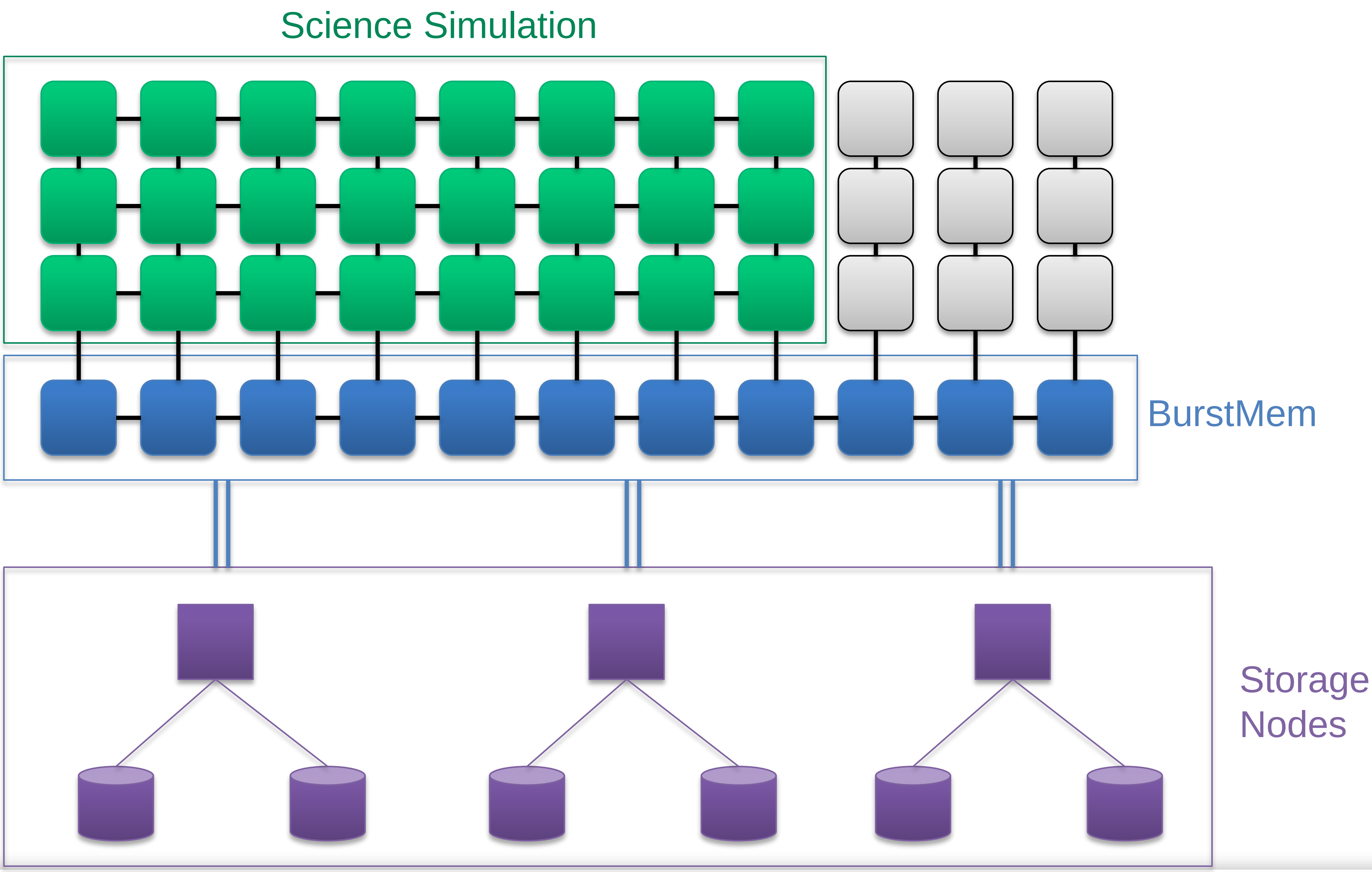
Memcache Overview

- Create a pool of storage servers
- Socket-based communication
- Stores Key-Value pairs in 2-level cache
  - Client chooses the server to send data
  - Server chooses the bucket to store value in
- Provides no guaranteed consistency (CAP!)
  - HPC uses middleware – probably easy to workaround
  - All clients need to know about all servers

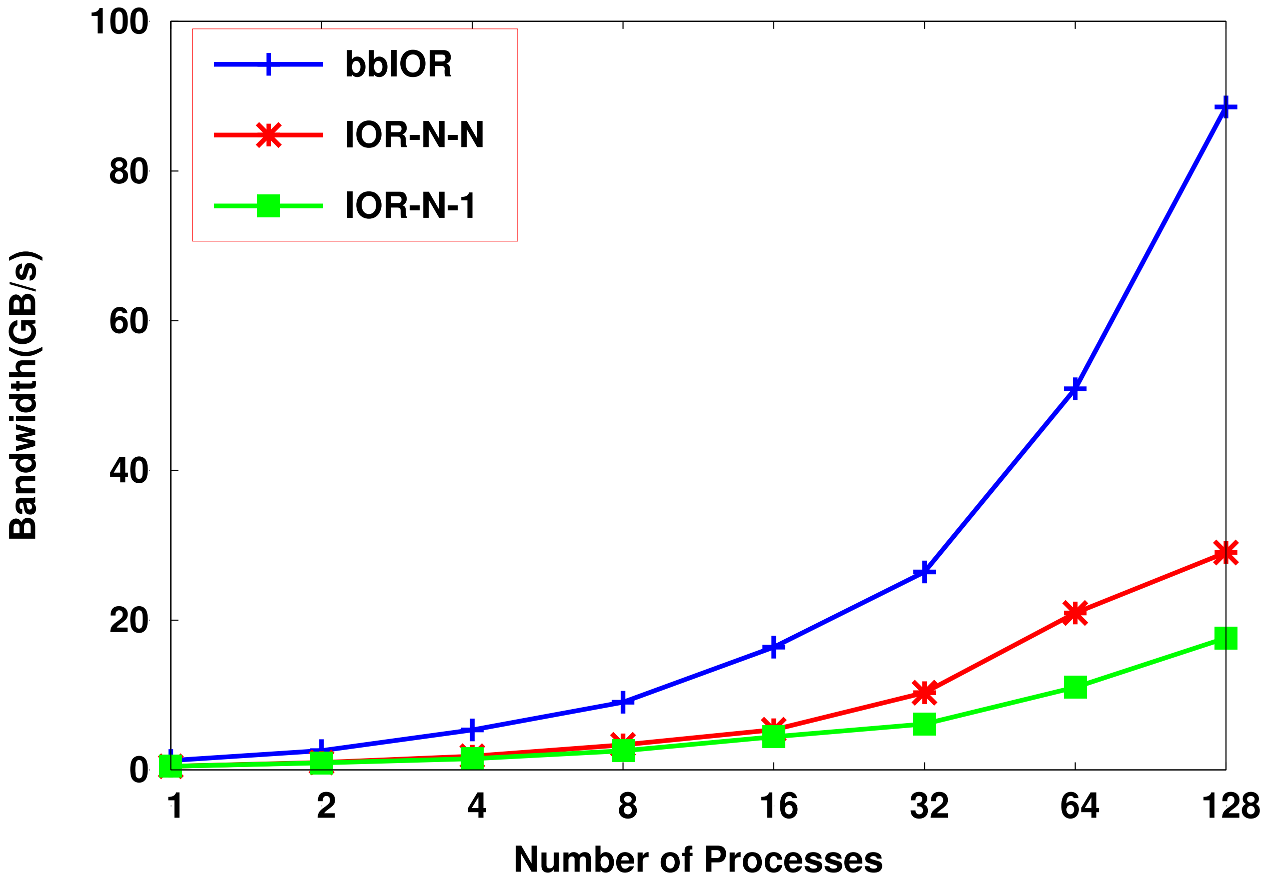
Memcache Modifications

- Port networking code to support multiple interconnects
  - CCI is developed at ORNL, so easy selection
- Add a checkpoint semantic
  - Modify the memcache key to annotate data
  - Ensure all data from a single checkpoint is tagged to a epoch
- Add scheme to efficiently flush data to FS
  - Leverage interconnect between burst buffer nodes
  - Explore multiple schemes
- Call it BurstMem

Burst Buffer Environment

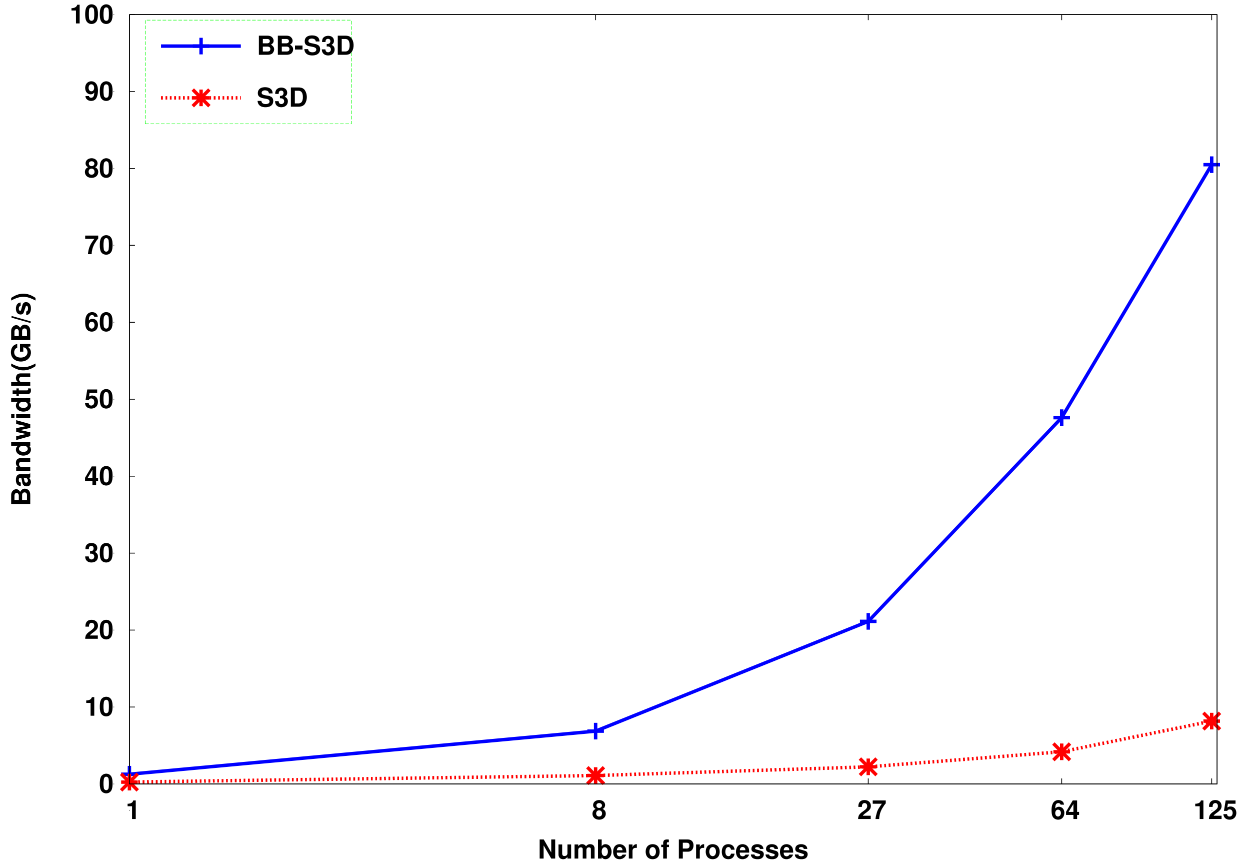


Ingress Results - IOR



Early results, experiments still in progress

Ingress Results – S3D I/O Kernel

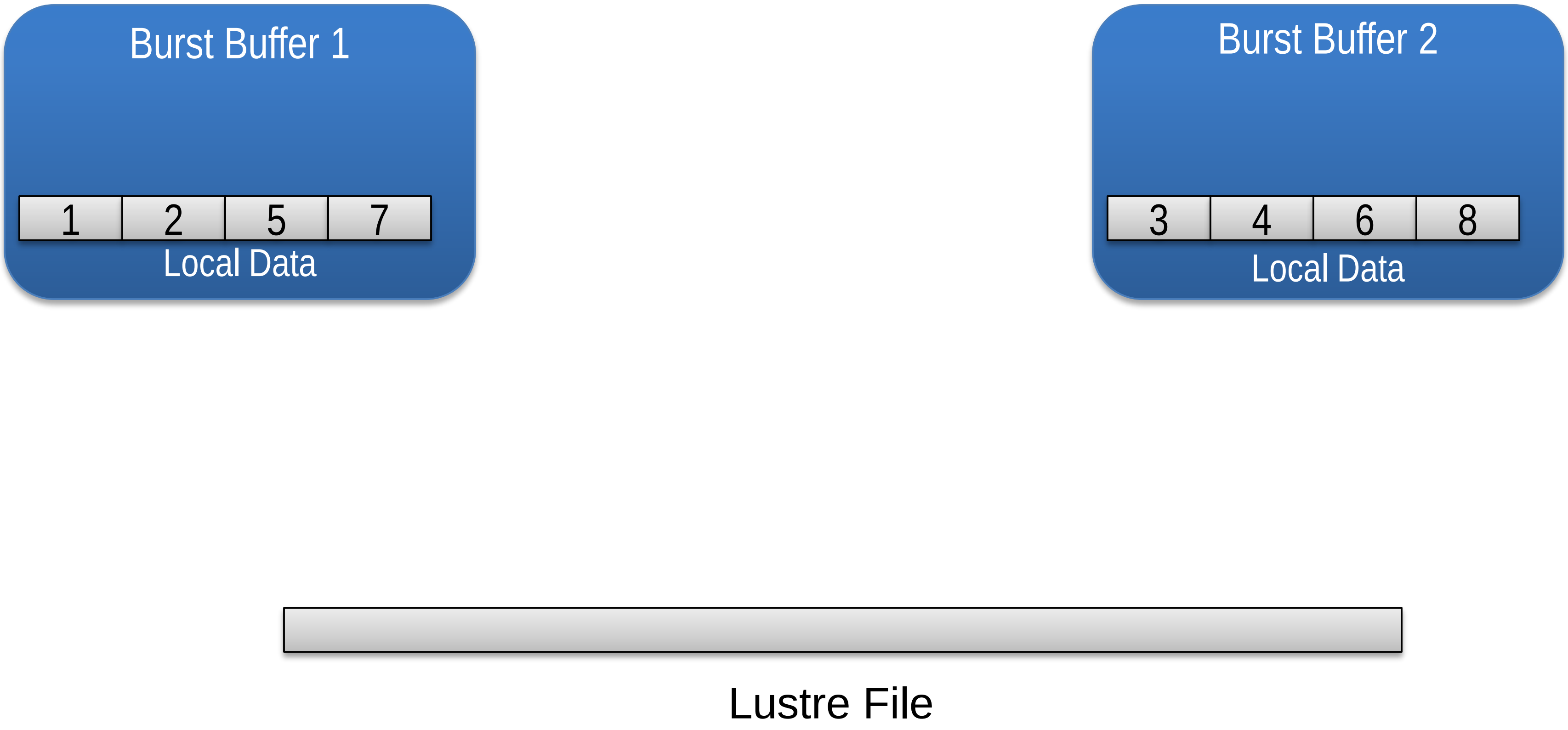


Early results, experiments still in progress

S3D is writing one file per process

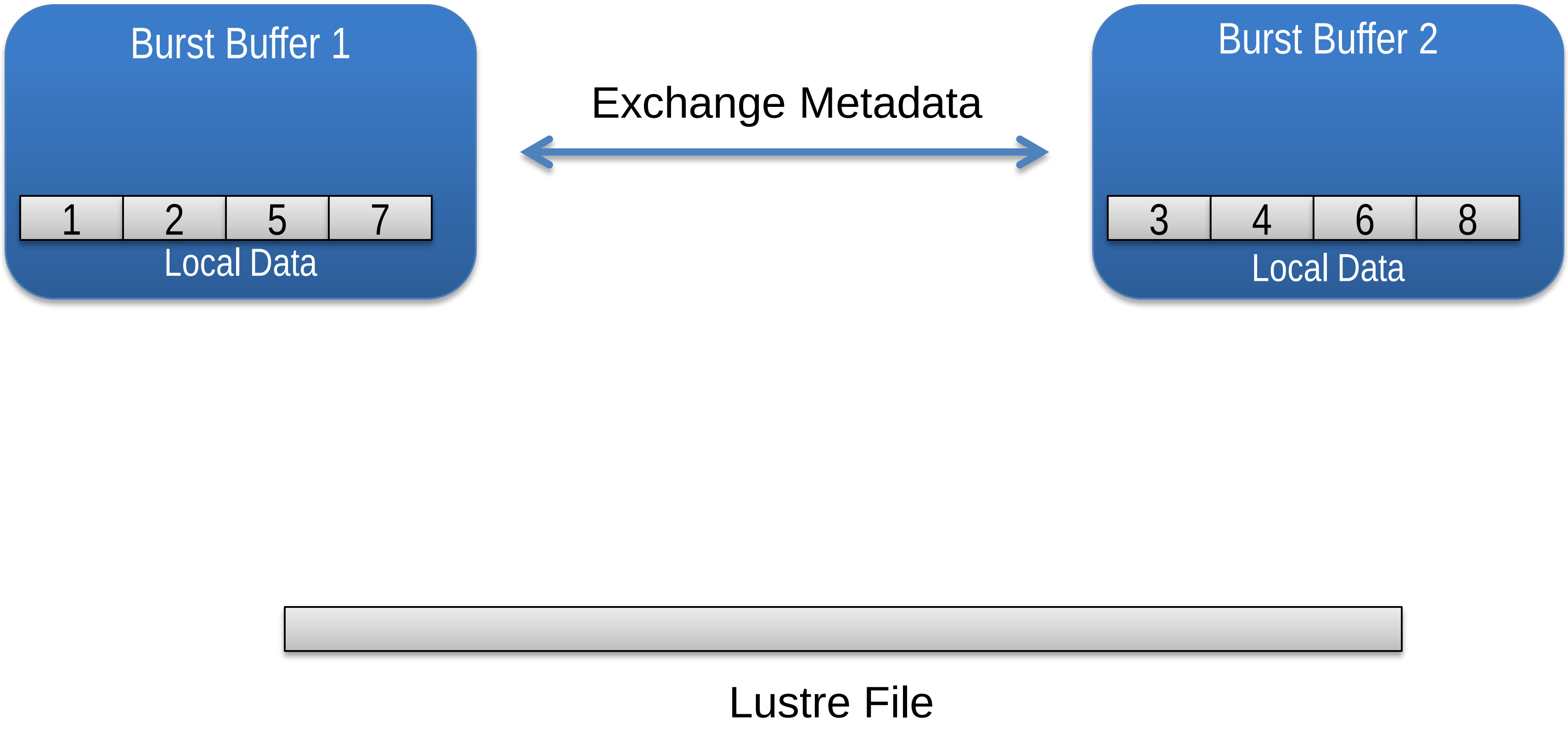
Egress Strategy

- Currently provide two-phase I/O
- New idea (I think): Limited Skew I/O



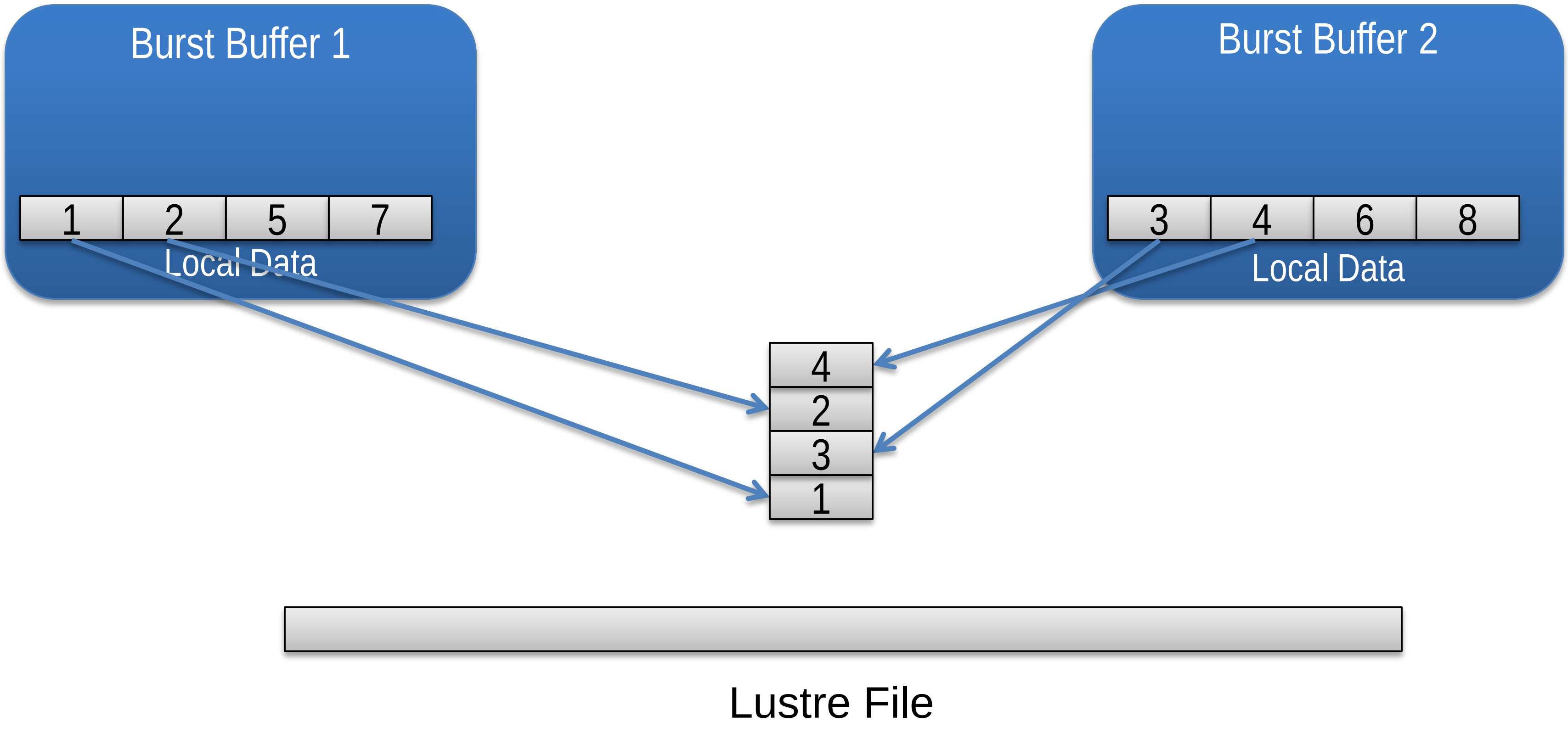
Egress Strategy

- Currently provide two-phase I/O
- New idea (I think): Limited Skew I/O



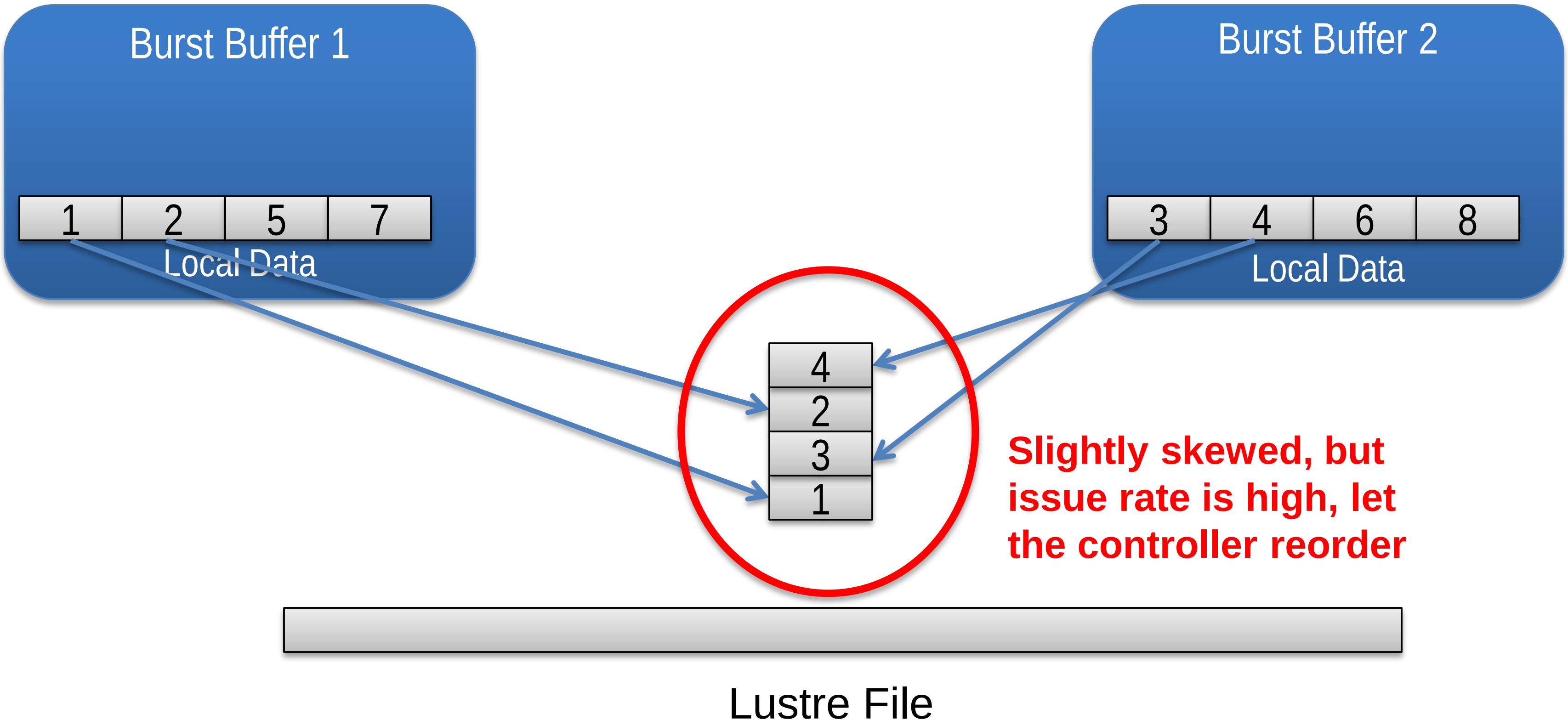
Egress Strategy

- Currently provide two-phase I/O
- New idea (I think): Limited Skew I/O



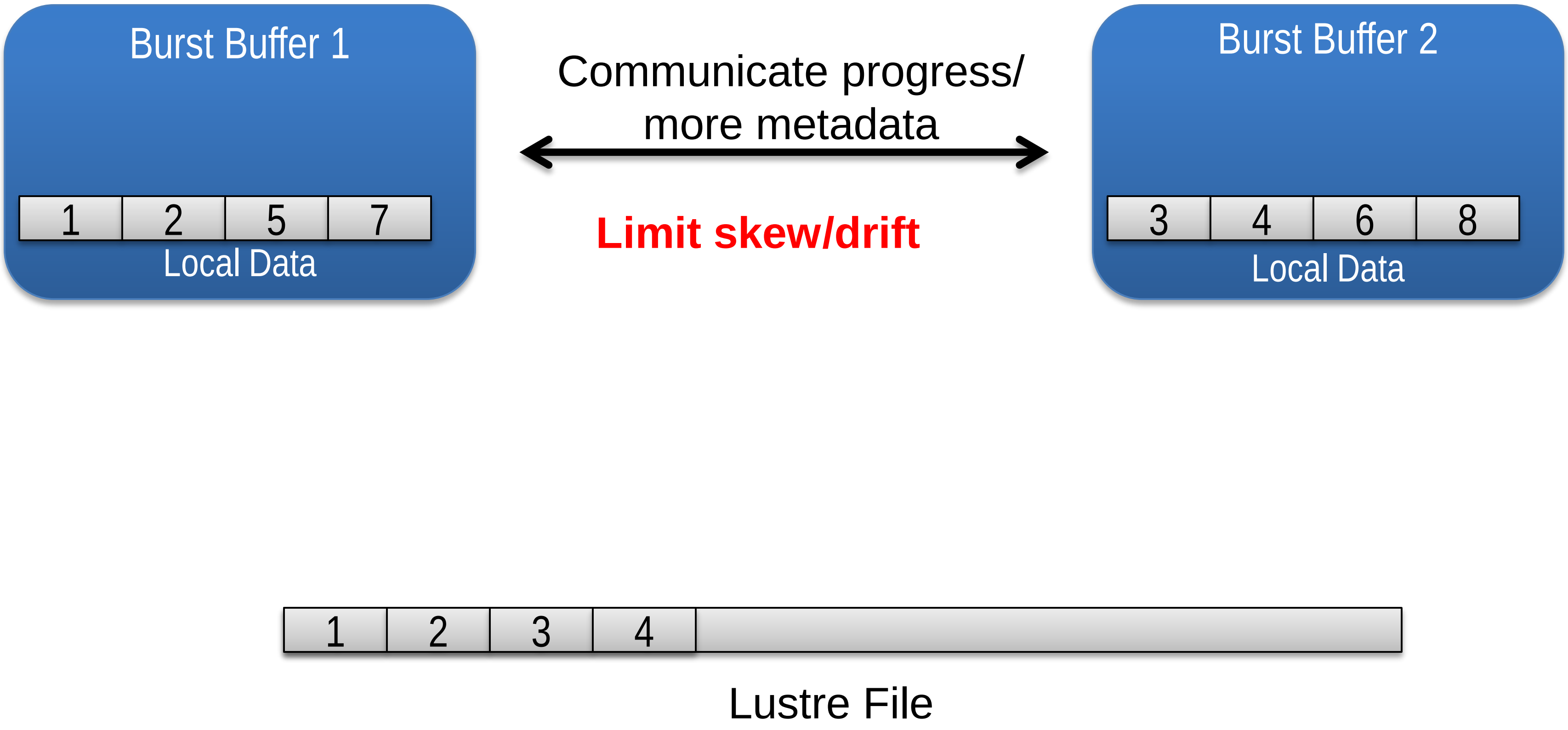
Egress Strategy

- Currently provide two-phase I/O
- New idea (I think): Limited Skew I/O



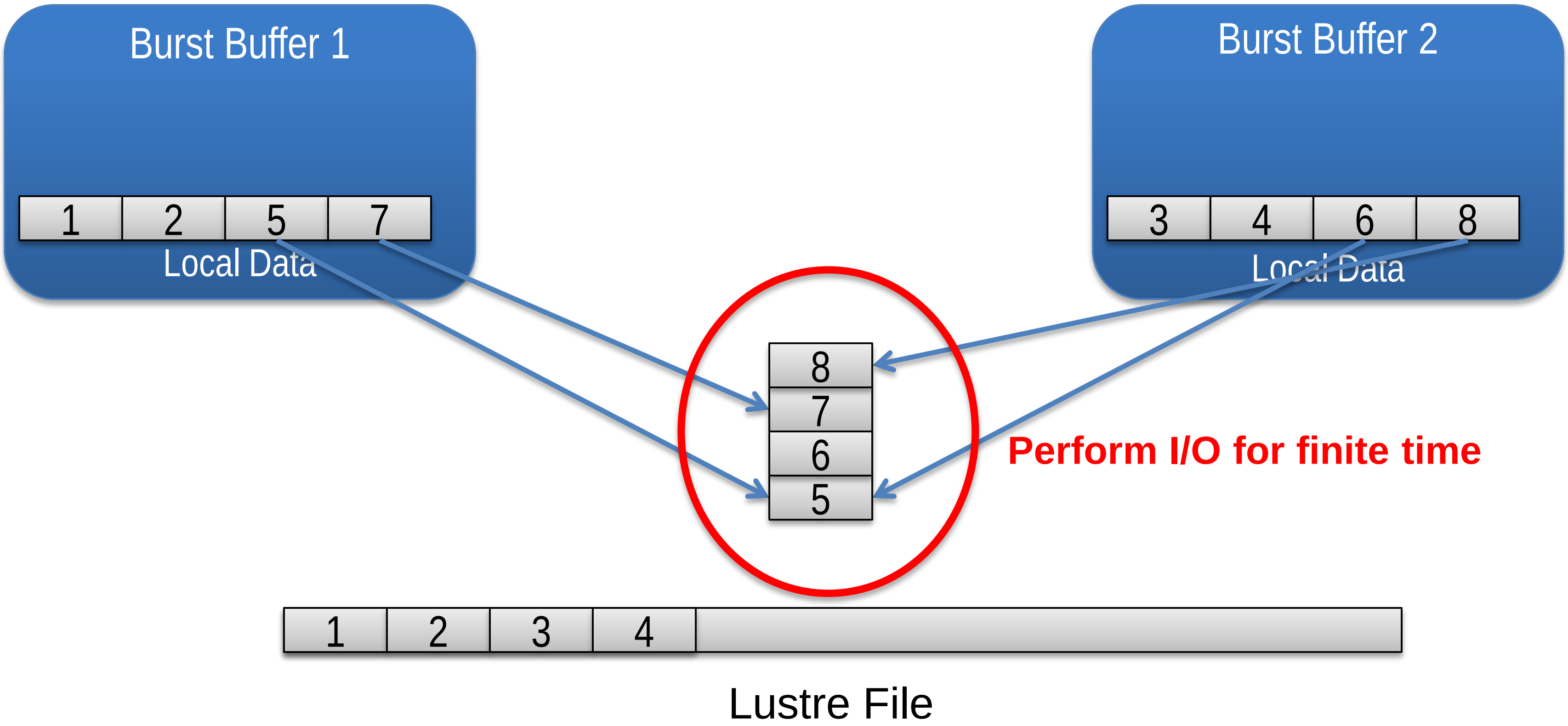
Egress Strategy

- Currently provide two-phase I/O
- New idea (I think): Limited Skew I/O



Egress Strategy

- Currently provide two-phase I/O
- New idea (I think): Limited Skew I/O



Current Concerns

- Consider a 10% performance loss due to some phenomena
  - Checkpoint time goes from 6 minutes to 6.5 minutes per hour
  - Flush time goes from 60 minutes to 66 minutes per hour
  - Egress unstable, falls further behind each hour
    - Prevent ingress
    - Rely on failures to recover time
- Flush data from the memory caches efficiently
  - Burst buffer local storage
  - File system storage
- Impacts of overlapping reads and writes
  - Log locking, GC, TRIM?

Acknowledgements

Research sponsored by the Laboratory Directed Research and Development Program of Oak Ridge National Laboratory, managed by UT-Battelle, LLC, for the U. S. Department of Energy.

This research used resources of the Oak Ridge Leadership Computing Facility at the Oak Ridge National Laboratory, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

Thanks!

Questions?