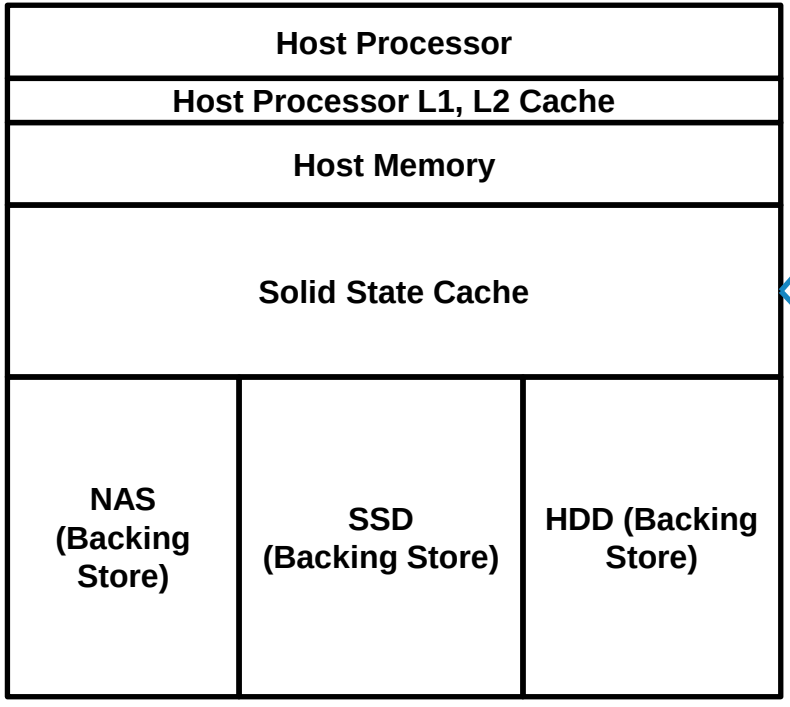
HEC: Improving Endurance of High Performance Flash-based Cache Devices

Jingpei Yang, Ned Plasson, Greg Gillis, Nisha Talagala, Swaminathan Sundararaman, Robert Wood

Patent Pending Technology

Why Flash Caching?

FUSION-IO



- High IOPS
- Low $/IOPS
- Local to server

$

- Storage services (snapshots, replication, etc)
- IOPS bottleneck
- Scaling IOPS is expensive

Primary Storage

- Provide improved application performance – high quality data closer to the application
- Read from cache; write to primary storage
- No need to reconfigure storage or applications
- Reduce storage costs

Flash Caches are Different

FUSION-IO

- SSCs are generally much smaller than their storage counterparts
- Workloads
  - Cache workloads are more write intensive – read misses become writes yielding Cache Layer Write Amplification (CLWA)
- Endurance of media
  - Limited endurance and getting more limited with new NAND node geometries
- Un-coordination at the Caching and Flash Translation Layer (FTL)
  - Can cause increased Flash Layer Write Amplification (FLWA)
- Techniques for reducing FLWA and CLWA must consider impact to cache hit rate

Example – TPC-E Polluted Workload

FUSION-IO

Writes due to cache misses

Writes due to garbage collection

Original Reads (GiB)	Original Writes (GiB)	Cache Size (GiB)	Cache Writes (GiB)	GC Writes (GiB)	Total Writes (GiB)	CLWA	FLWA	TCWA	Hit Rate (%)
331.9	36.8	80	322.13	1553.98	1876.11	8.75	5.82	50.93	14.03
		100	300.11	1459.13	1759.24	8.16	5.86	47.82	20.67
		120	275.83	1352.01	1627.84	7.5	5.9	44.25	27.98

GC – garbage collection
CLWA – Cache Layer Write Amplification (cache writes / original writes)
FLWA - Flash Layer Write Amplification (GC writes / cache writes)
TCWA – Total Combined Write Amplification (CLWA * FLWA)

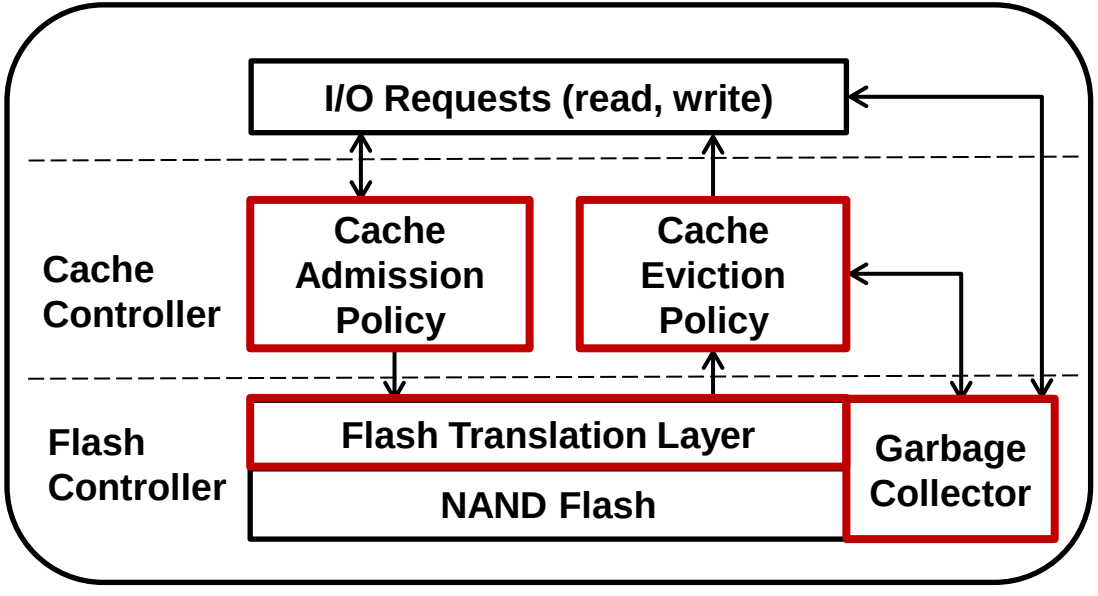
Conditions of simulation

1. No admission control
2. Tail-drop GC policy
3. Eviction performed at cache layer

TCWA of 40-50x and low hit rate demonstrates that opportunity exists for reducing writes and improving hit rate

Techniques for reducing CLWA and FLWA

- CLWA
  - How do admission policies affect device endurance?
  - Can cache eviction benefit from knowledge of the FTL?
- FLWA
  - Can garbage collection further reduce write amplification?
  - What if eviction was performed by the FTL?



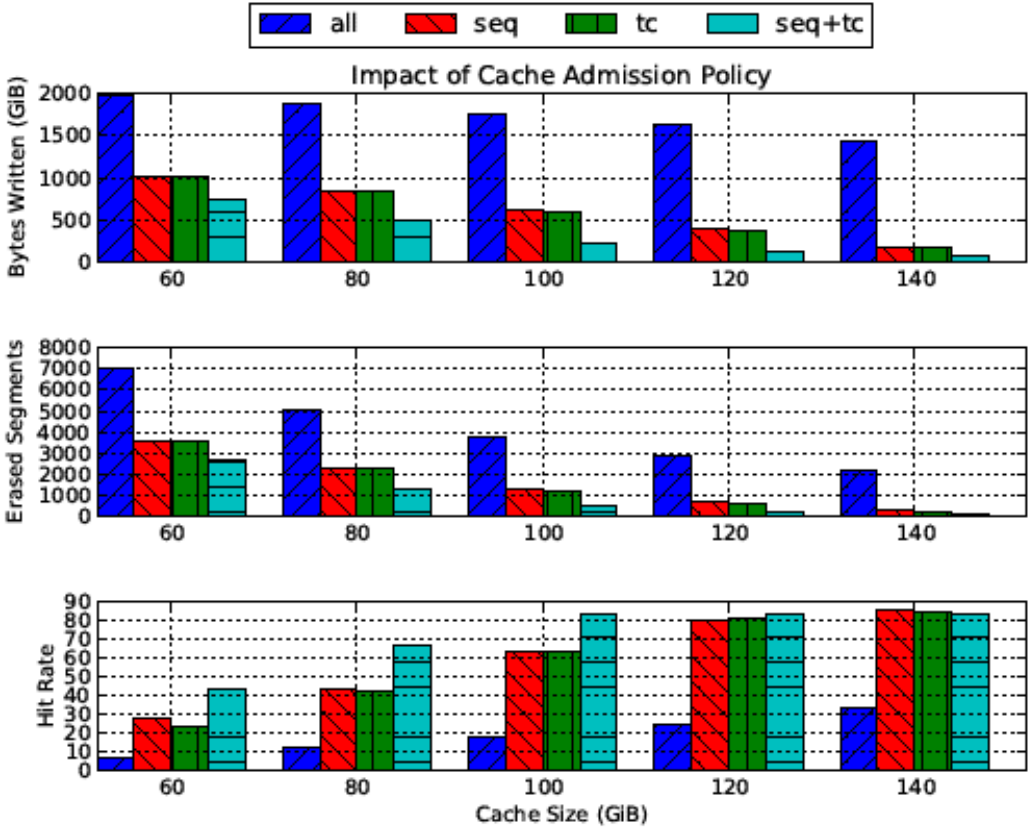
What is the combined impact of these techniques on hit rate and endurance?

Reducing CLWA – Cache Admission Policies

Admission Policy	Description
Admit All	All data is admitted to cache
Selective Sequential Rejection (SSEQR)	All non-sequential and short sequential sector ranges are admitted
Touch Count (TC)	Sectors or collections of sectors are admitted after they have been accessed N times
Sequential and Touch Count (SSEQR + TC)	Admission if data meets criteria for both SSEQR and TC

Admission policies can be used for improving the quality of data admitted to the cache thus improving hit rate, decreasing read misses and in turn writes to media

Reducing CLWA - Admission Policies



TPC-E Polluted Workload Trace

Selective admission policies

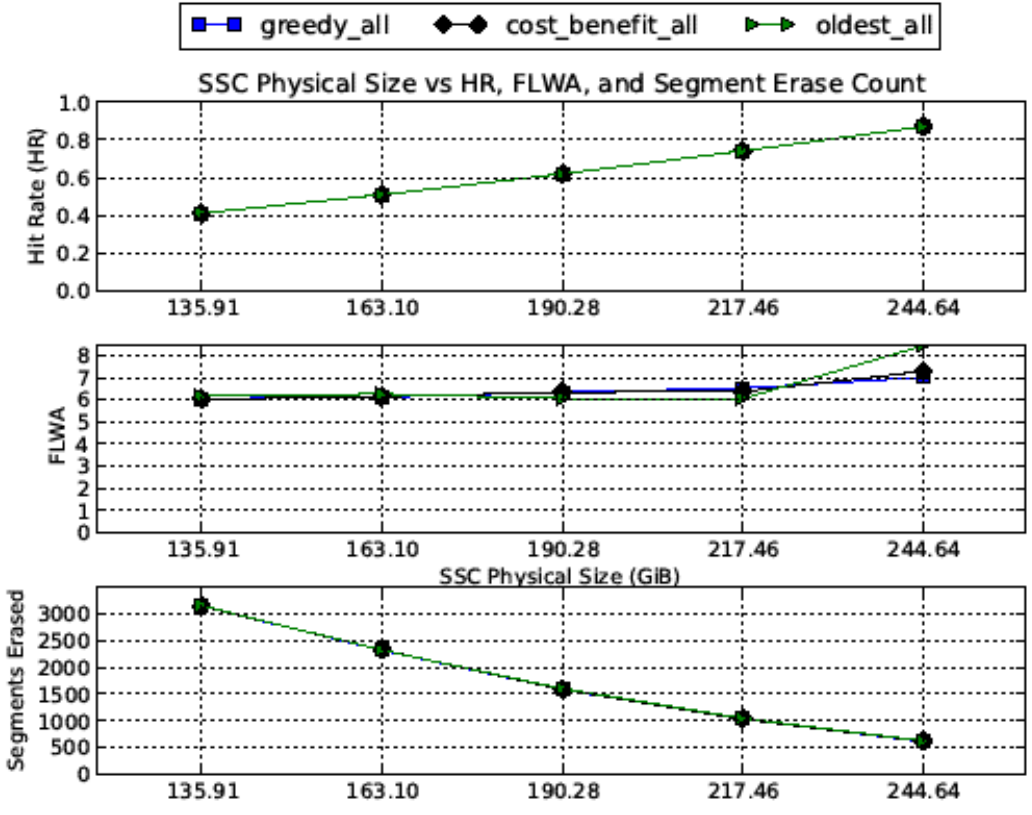
- Reduce bytes written
- Reduce segment erase counts
- Improve read hit rate

Reducing FLWA – FTL GC Policies

GC Policy	Description
Greedy	Victim segment is selected based on having the largest amount of “invalid” data
Cost Benefit	Segment selected based on “invalid” data percentage and age of data in the segment [Rosenblum LFS92]
Oldest (tail drop)	Segment selected is the oldest segment or tail of log

More selective GC policies can be used to reduce FLWA

Reducing FLWA – GC Policy



TPC-E Polluted Workload

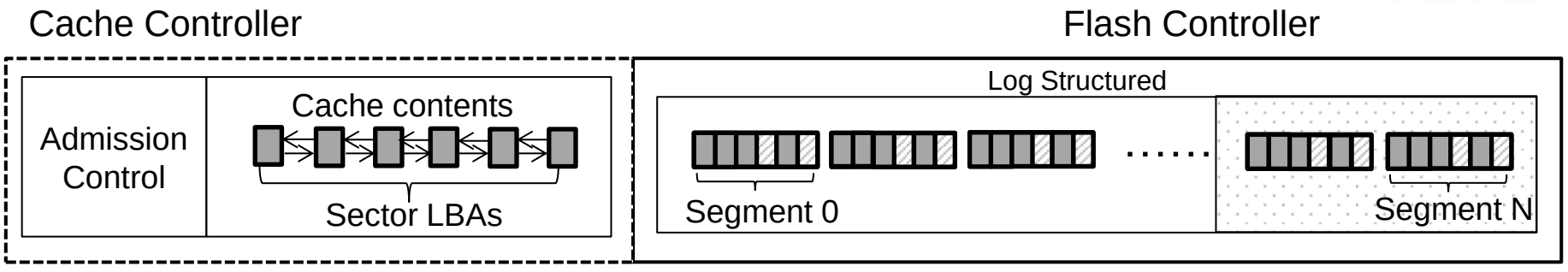
More selective GC policies

- Modest reductions in FLWA
- No noticeable change in hit rate and erased segments

Reducing FLWA – Cache Eviction Modes

FUSION-IO

Cache-based Eviction

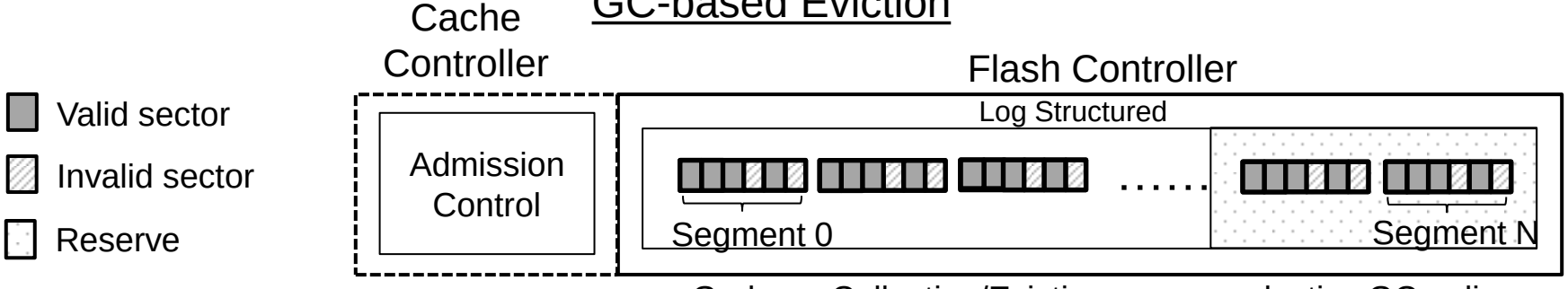


Eviction = LRW/LRU Replacement Policy
(TRIM to Flash Controller for invalidation)

TRIM →

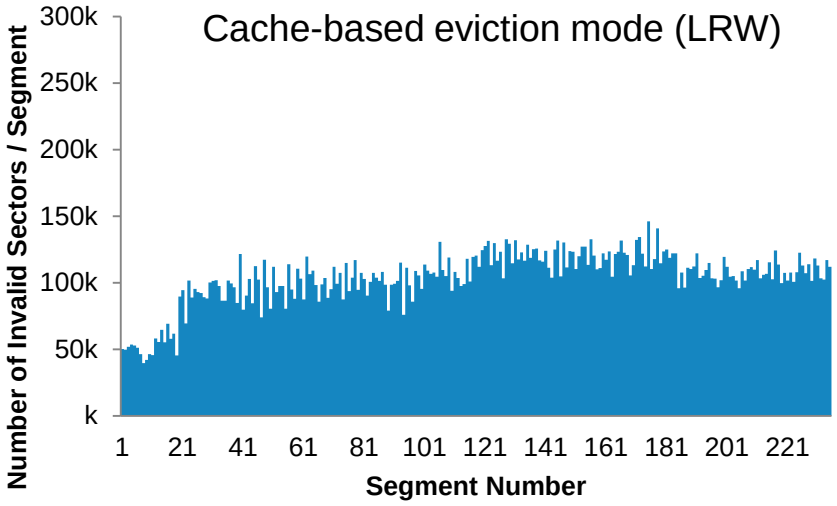
Garbage Collection = GC Policy (e.g. Tail Drop)

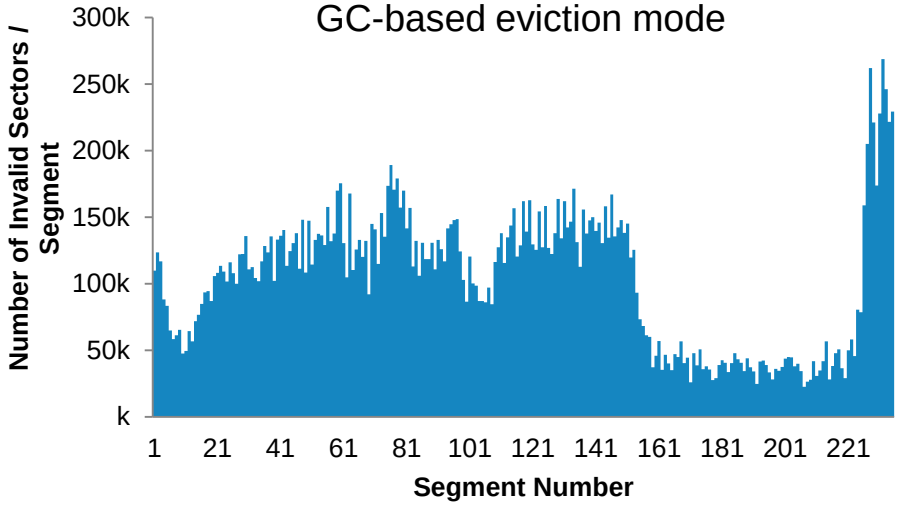
GC-based Eviction



Garbage Collection/Eviction = more selective GC policy

Reducing FLWA – Eviction Modes and Validity Distribution





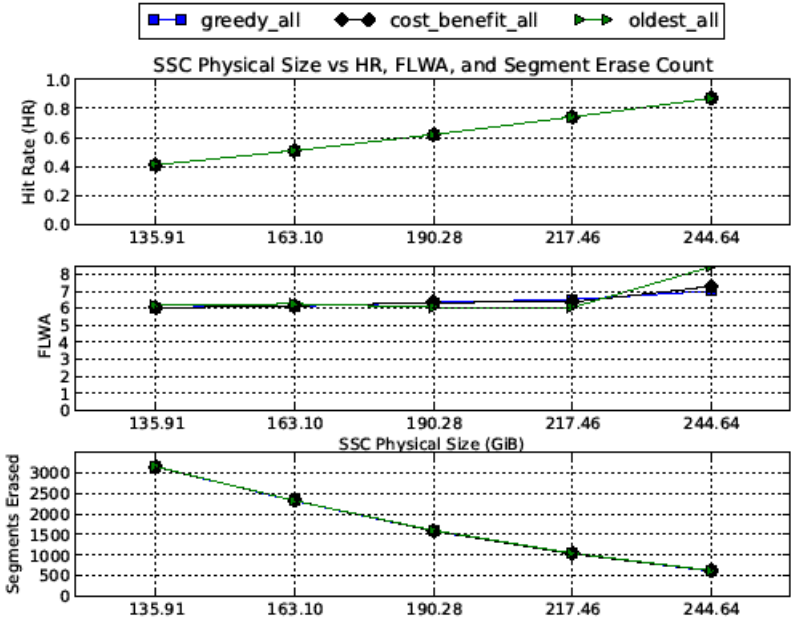
TPC-E Workload

Improved victim segment selection and improved GC effectiveness from non-uniform distribution

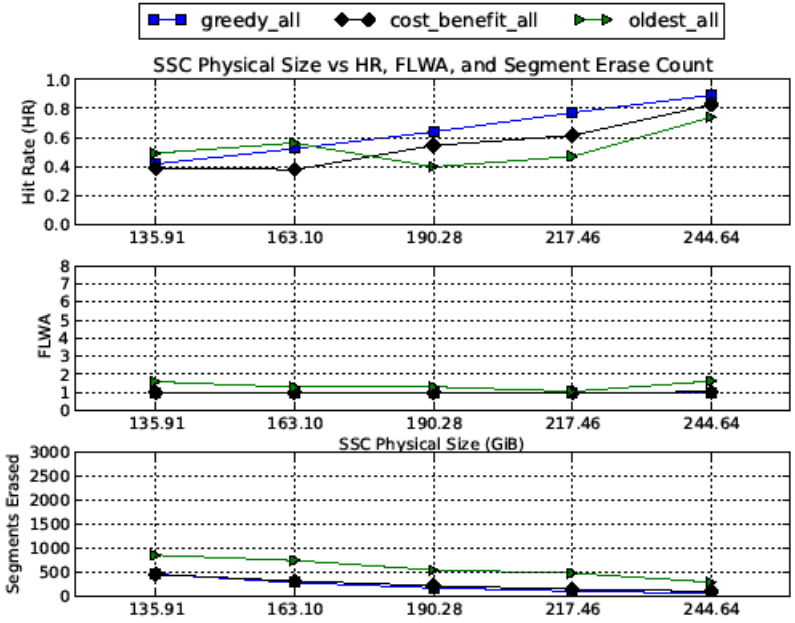
Reducing FLWA – GC Policy and Eviction Mode

FUSION-IO

Cache-based eviction mode (LRW)



GC-based eviction mode



TPC-E Polluted Workload

Significant reduction in FLWA and segments erased with GC-based eviction while maintaining hit rate

Collective Endurance Impact

FUSION-IO

Admission Policy = ADMIT_ALL
Eviction Mode = Cache-based
Eviction Policy = LRW

TPC-E Polluted Trace - Before

Original Reads (GiB)	Original Writes (GiB)	Cache Size (GiB)	Cache Writes (GiB)	GC Writes (GiB	Total Writes (GiB)	CLWA	FLWA	TCWA	Hit Rate (%)
331.9	36.8	80	322.13	1553.98	1876.11	8.75	5.82	50.93	14.03
		100	300.11	1459.13	1759.24	8.16	5.86	47.82	20.67
		120	275.83	1352.01	1627.84	7.5	5.9	44.25	27.98

Admission Policy = SSEQR+TC
Eviction Mode = GC-based
Eviction Policy = cost_benefit

TPC-E Polluted Trace - After

Original Writes (GiB)	Original Writes (GiB)	Cache Size (GiB)	Cache Writes (GiB)	GC Writes (GiB)	Total Writes (GiB)	CLWA	FLWA	TCWA	Hit Rate (%)
36.8	36.8	80	70.1	169.19	239.29	1.9	3.41	6.5	46.59
		100	37.65	169.05	206.7	1.02	5.49	5.62	59.78
		120	37.65	44.8	82.45	1.02	2.19	2.24	59.78

TCWA reduced while improving read cache hit rate using more selective admission, eviction, and GC policies

Conclusions

- CLWA and FLWA exists and their combined effects can increase original write workloads by 10x to 50x
- We have identified techniques that provide for reduction of CLWA up to 20x and FLWA up to 10x
- We have demonstrated that combinations of such techniques reduce TCWA up to 20x yielding significant improvements in endurance and use of PBW
- We show that such techniques maintain or improve hit rate with coordinated cache and flash controller
- Future work exists in understanding such effects in write-back caching and SSC configurations with multiple cache instances


THANK YOU
fusionio.com | REDEFINE WHAT'S POSSIBLE