SSD Caching:
Device-Mapper- and Hardware-based solutions compared

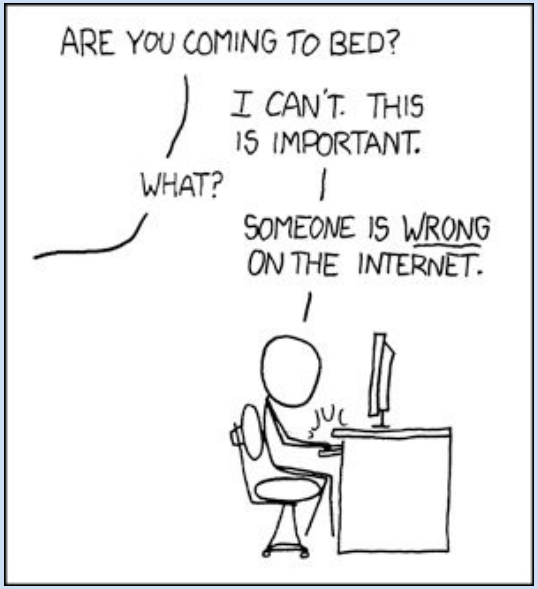
Werner Fischer & Georg Schönberger
Technology Specialists Thomas-Krenn.AG

18. LinuxTag / Berlin / Germany
24th May 2011

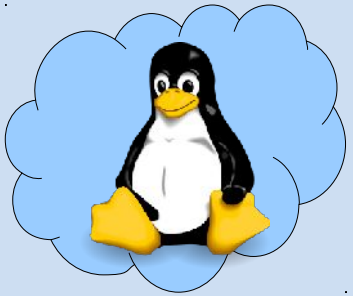
Agenda

- **Introduction**
- **Technologies compared**
- **A few details**
- **Performance results**
- **Test strategy**
- **Lessons learned**

Introduction



Source: xkcd.com



Do things the right way!

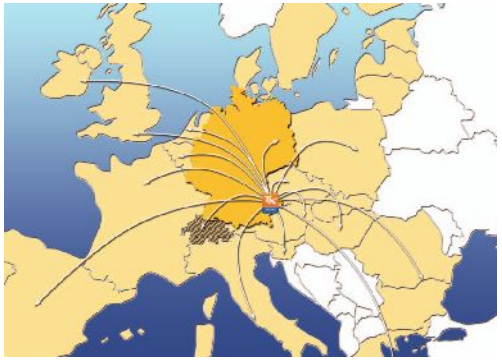
Introduction

who is Thomas-Krenn.AG®
The server experts

Server & accessories "Made in Germany"

based in Freyung, Bavaria

serving all over Europe



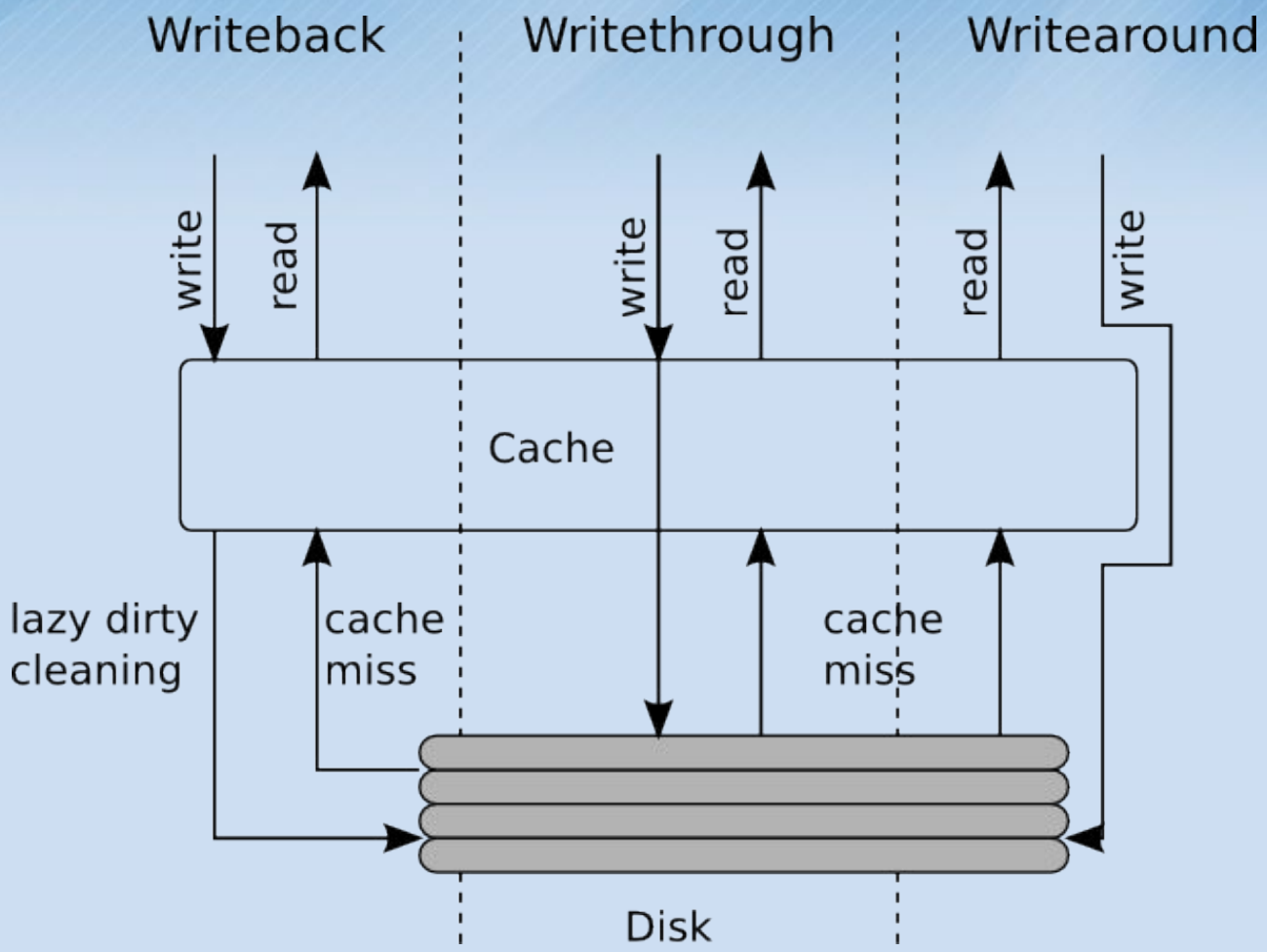
Should I use SSD caching?

Well, Yes and No...

```
if(dataSet == known &&
   ramAvailable != enough &&
   appAccess == analyzed &&
   perfTests == available)
  CheckForCacheTechnologies();
else
 NeedMoreInfo();
```

Establish some wording...

Caching Basics



Some main features compared...

SSD Caching – Compared 1 of 6

	WB	WT	WA/read-only
FlashCache	✓	✓	✓
CacheCade	✓ [1]	✓	✗
MaxCache	✓ ! [2]	✓	✗

[1] Including ForcedWB

! Attention: No redundant cache with multiple SSDs possible

[2] Including InstantWB

SSD Caching – Comparison 2 of 6

	Skip sequential I/O
FlashCache	☑ [1]
CacheCade	[2]
MaxCache	☑ [3]

- **Maybe your disk RAID is faster for sequential I/O**

[1]Configurable via sysctl (threshold)

[2]No further details known

[3]Not configurable, always skipped

SSD Caching – Comparison 3 of 6

	Lazy Cleaning	
FlashCache		1
CacheCade		2
MaxCache		3

- **WB: clean dirty pages from SSD to HDD**

[1]Based on threshold and idleness

- Dynamic via sysctl

[2]No further details known

[3]Background flush with dynamic rate

- FlushAndFetchRate
- dirtyPageThreshold

SSD Caching – Comparison 4 of 6

	Cache persistence
FlashCache	 ! [1]
CacheCade	 [2]
MaxCache	 [3]

[1]Only WB Cache is persistent

- https://groups.google.com/forum/#!topic/flashcache-dev/xRW64FOxgWw
- People are discussing to add persistence for WT

[1]On node crash, only Dirty Blocks are in cache

- https://groups.google.com/forum/#!topic/flashcache-dev/LqjKZcGL1eU

[2]Persistent for all cache modes

[3]Dirty shutdown: WT – discarded, WB – LV failed

SSD Caching – Comparison 5 of 6

Error strategy	WB	WT
FlashCache	 [1]	Currently - Error[2]
CacheCade	 [3]	
MaxCache	 [4]	[5]

[1]No switch from WB to WT if SSD redundant

[2]Patch to make WB/WT errors transparent

- https://groups.google.com/forum/#!topic/flashcache-dev/6XEhY87uv1g

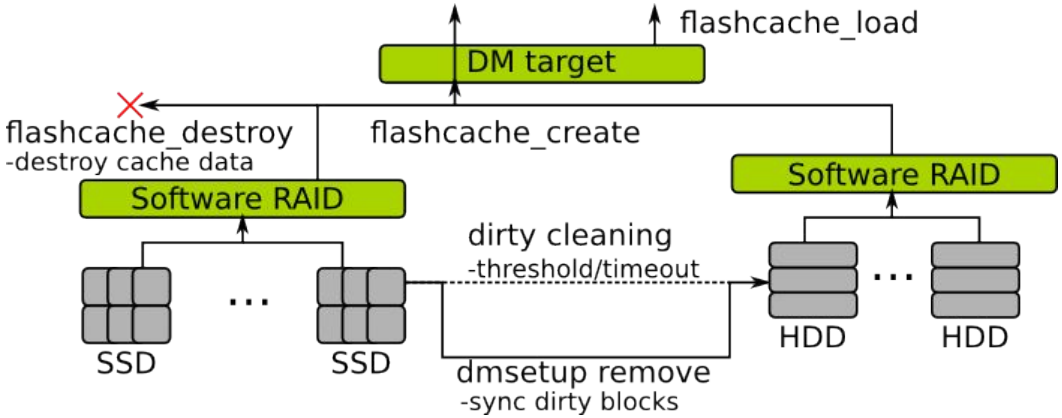
[3]Details on slide 26

[4]With write cache LV fails

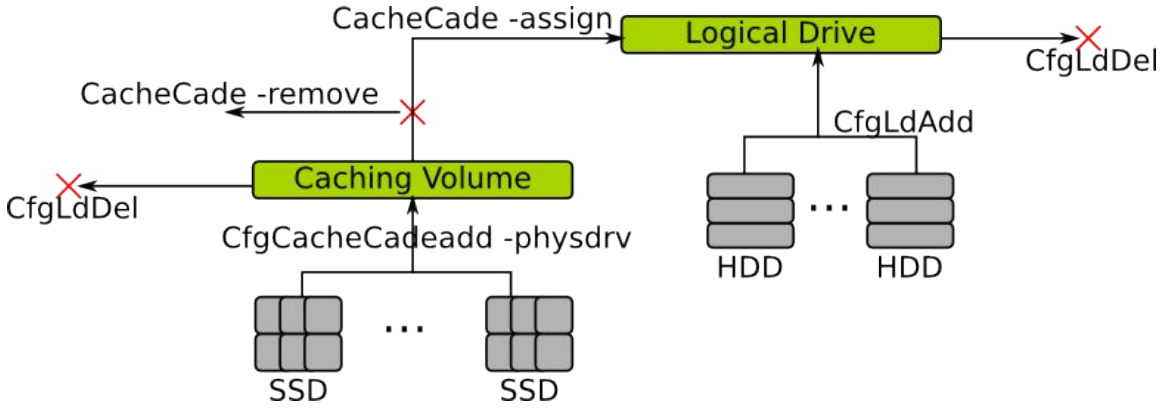
[5]No further information available

SSD Caching – Comparison 6 of 6

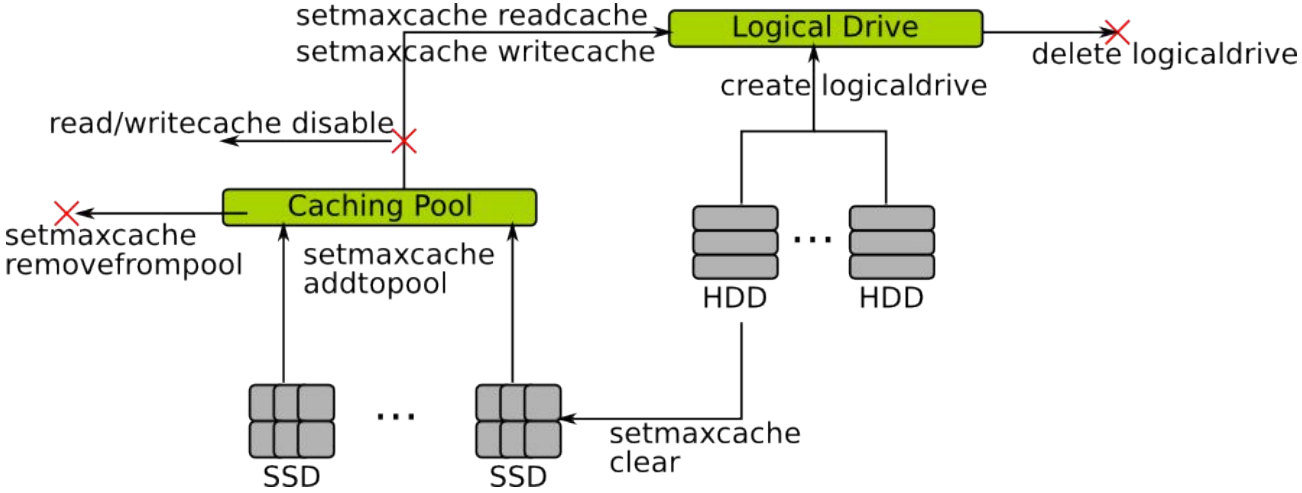
	Hot spot detection
FlashCache	
CacheCade	
MaxCache	



FlashCache



CacheCade



MaxCache

Take a closer look...

Not an official logo.
Source: deviantart.com

FlashCache

Not all about Facebook is bad...

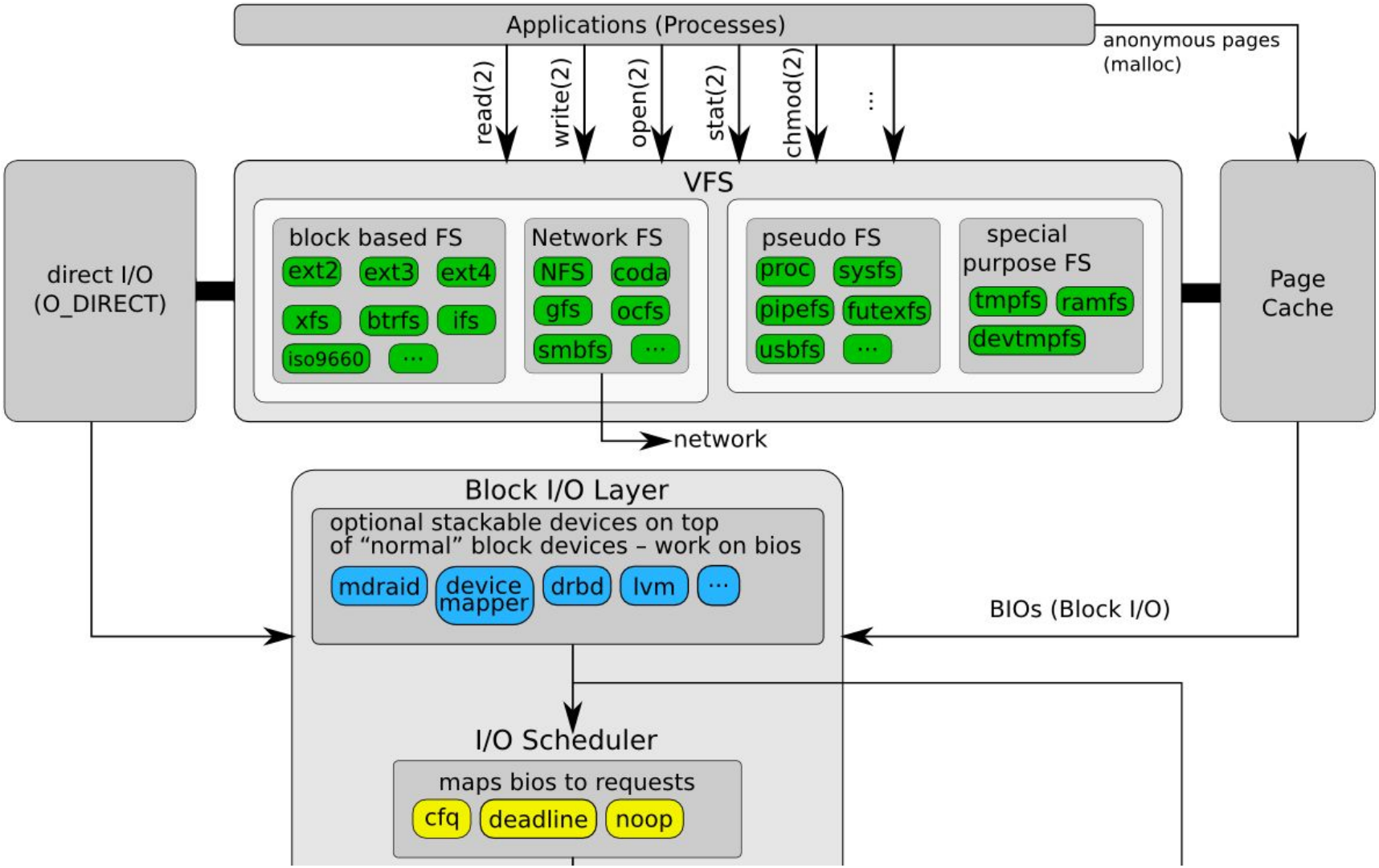
Visit https://github.com/facebook/flashcache/

FlashCache – Introduction

- **Open Source :D**
  - Get code from Github
- **Using Linux Device Mapper (cf. LVM)**
  - Visit our I/O stack diagram
    - http://www.thomas-krenn.com/en/oss/linux-io-stack-diagram.html
  - Based on dm-cache
- **Easy to deploy and use**
  - Compile
  - Load one kernel module

The Linux I/O Stack Diagram

version 0.1, 2012-03-06
outlines the Linux I/O stack as of Kernel version 3.3



FlashCache – Highlights

- **FIFO/LRU replacement policy – online changeable**
- **Define process blacklist/whitelist**
- **Cleaning of dirty blocks configurable**
  - Threshold
  - Idleness
- **Error injection flags via sysctl**
  - E.g. READCACHE_ERROR
- **Detailed Cache statistics**
  - dmsetup table

FlashCache – Statistics

```
fc-root: 0 3907029168 flashcache conf:
        ssd dev (/dev/sdd), disk dev (/dev/sdc) cache mode(WRITE_BACK)
        capacity(32638M), associativity(512), data block size(4K)
metadata block size(4096b)
        skip sequential thresh(0K)
        total blocks(8355328), cached blocks(1048668), cache
percent(12)
        dirty blocks(1048662), dirty percent(12)
        nr_queued(0)
Size Hist: 1024:1 4096:9765813
512:0 1024:1 1536:0 2048:0 2560:0 3072:0 3584:0 4096:9765813 4608:0
5120:0 5632:0 6144:0 6656:0 7168:0 7680:0 8192:0 8704:0 9216:0 9728:0
10240:0 10752:0 11264:0 11776:0 12288:0 12800:0 13312:0 13824:0
14336:0 14848:0 15360:0 15872:0 16384:0
reads=1048896 writes=8716916
read_hits=1048580 read_hit_percent=99 write_hits=9 write_hit_percent=0
dirty_write_hits=0 dirty_write_hit_percent=0 replacement=0
write_replacement=0 write_invalidates=80 read_invalidates=0
pending_enqueues=0 pending_inval=0 metadata_dirties=1048662
metadata_cleans=0 metadata_batch=1042405 metadata_ssd_writes=6257
cleanings=0 fallow_cleanings=0 no_room=0 front_merge=0 back_merge=0
disk_reads=316 disk_writes=7668261 ssd_reads=1048580
ssd_writes=1055014 uncached_reads=221 uncached_writes=7668261
uncached_IO_requeue=0 uncached_sequential_reads=0
uncached_sequential_writes=0 pid_adds=0 pid_dels=0 pid_drops=0
pid_expiry=0
```

FlashCache – Considerations

- **Make your cache redundant (WB)**
  - What if SSD fails?
  - Cf. https://groups.google.com/forum/#!topic/flashcache-dev/6XEhY87uv1g
- **With WB partial writes are possible**
  - There are ideas to fix this (cf. flashcache-doc.txt)
- **Only WB cache is persistent**
  - Across reboots and cache removals
  - No metadata on SSD for WT/WA
    - Read https://groups.google.com/forum/#!topic/flashcache-dev/xRW64FOxgWw

LSI MegaRAID CacheCade

CacheCade – Introduction

- **Hardware-based solution**
- **Caching algorithm in controller**
- **Redundant SSD caches possible**
  - Avoid data loss with WB

CacheCade – Error handling

	WB	WT	FWB
Red.	Change to WT	Read SSD	WB
Non. Red.	No VD access	Read HDD	No VD access

- **Except WB and non redundant SSD Cache**
  - Application doesn't notice SSD failure
- **Controller reports over Storage Manager**

CacheCade – Considerations

- **Our tests showed that...**
  - Assign → initialization
  - Fio laying out files
    - Not only on SSD (cf. FlashCache)
    - Background syncs
      - Test Read values not correct
  - Fio with 'create_only'
    - 100 MB/s read
    - Data not completely in cache
    - → read a 2nd time → 215MB/s

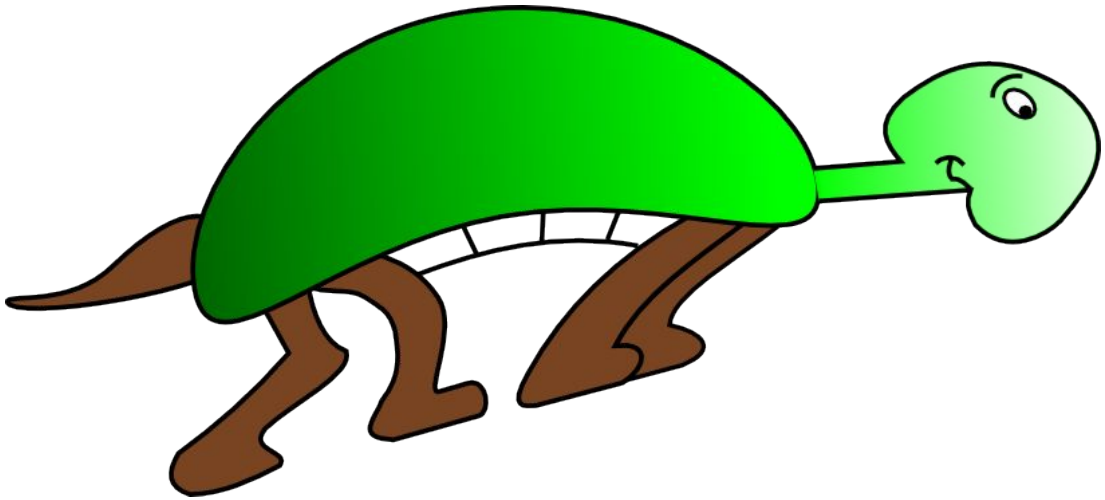
Adaptec MaxCache

MaxCache – Features

- **Manual**
  - http://download.adaptec.com/pdfs/user_guides/cli_v7_30_18837_users_guide.pdf
- **WriteCache**
  - Should not be used until now
  - Redundant cache cannot be created
- **Sequential I/O is not cached**
  - SAS drives may be faster
  - Performance stays at 110 MB/s with streaming I/O
- **Hot spot detection logic**


At the end its all down to performance...

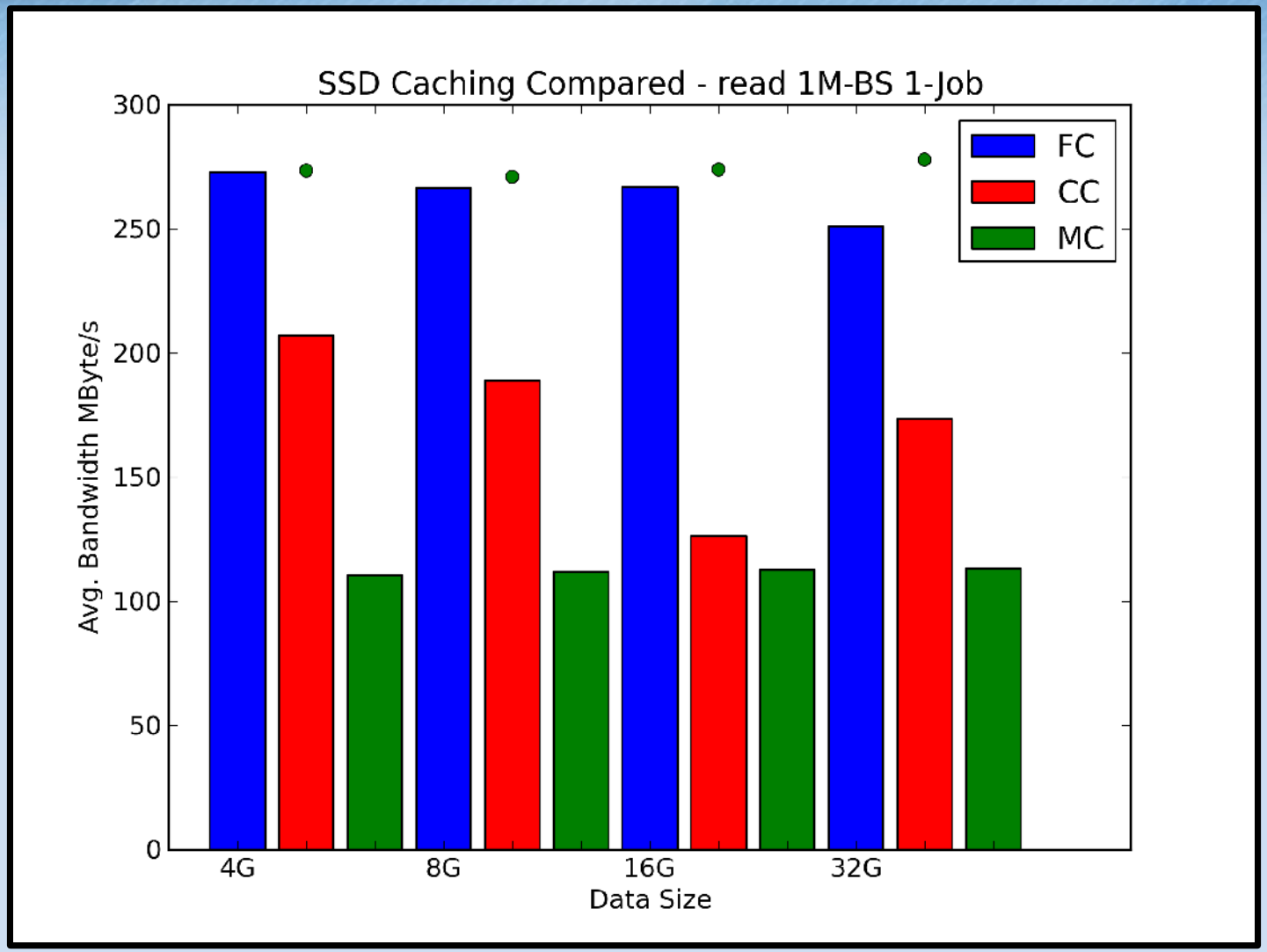
Prepare to get this...



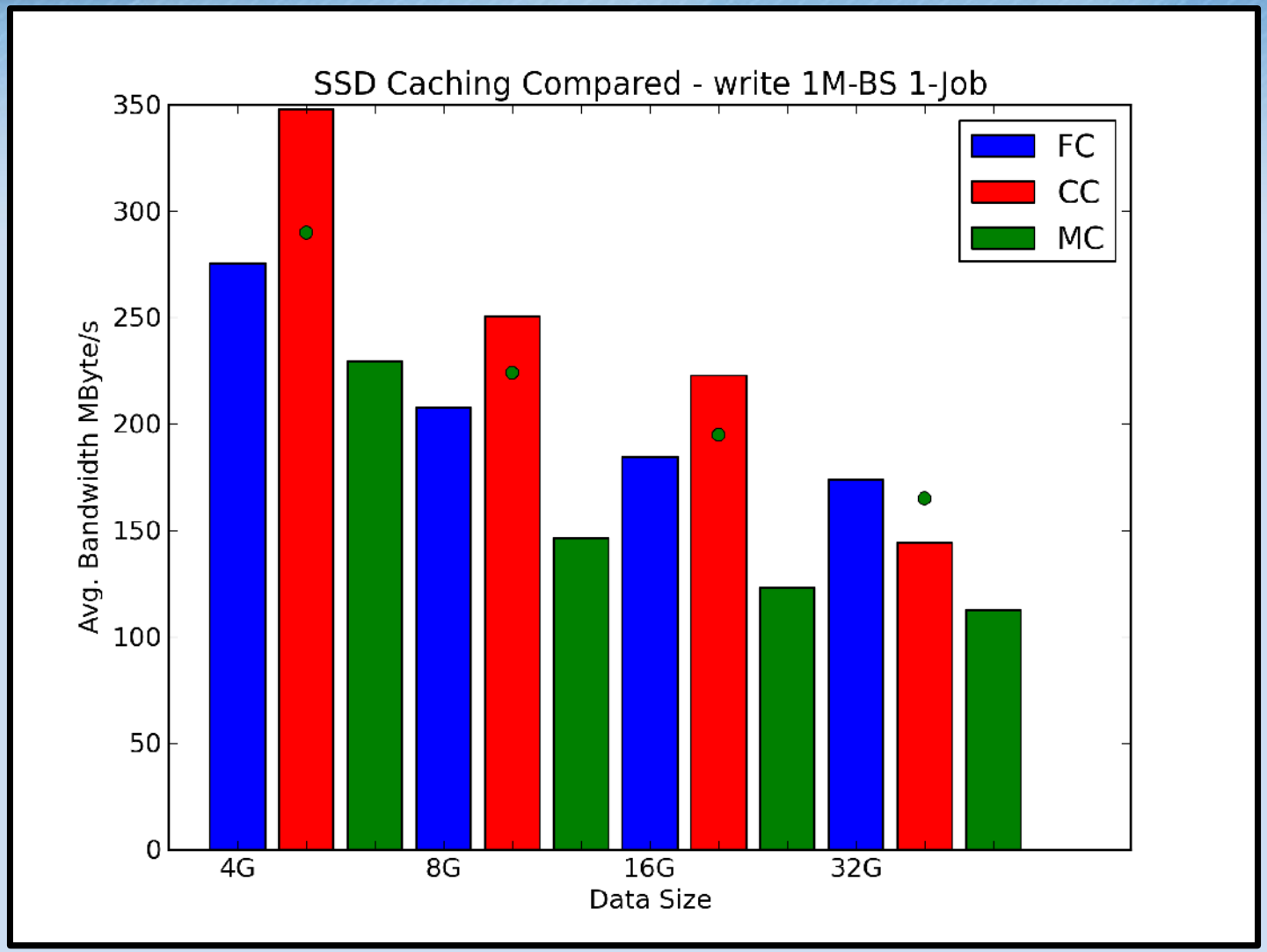
People might promise you...

Test your own specific application.
If you want with a little help from us :)

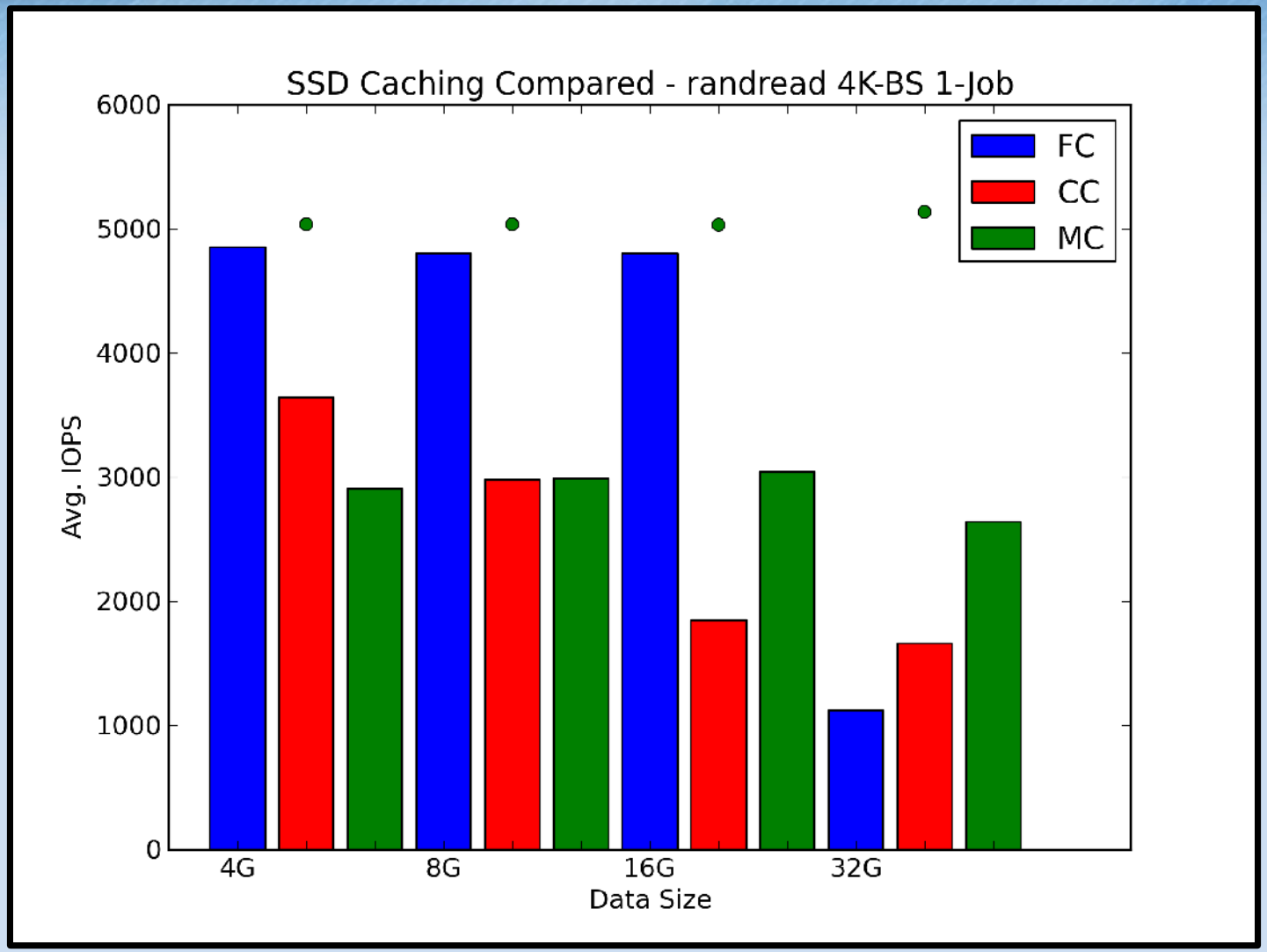
Comparison – Bandwidth read



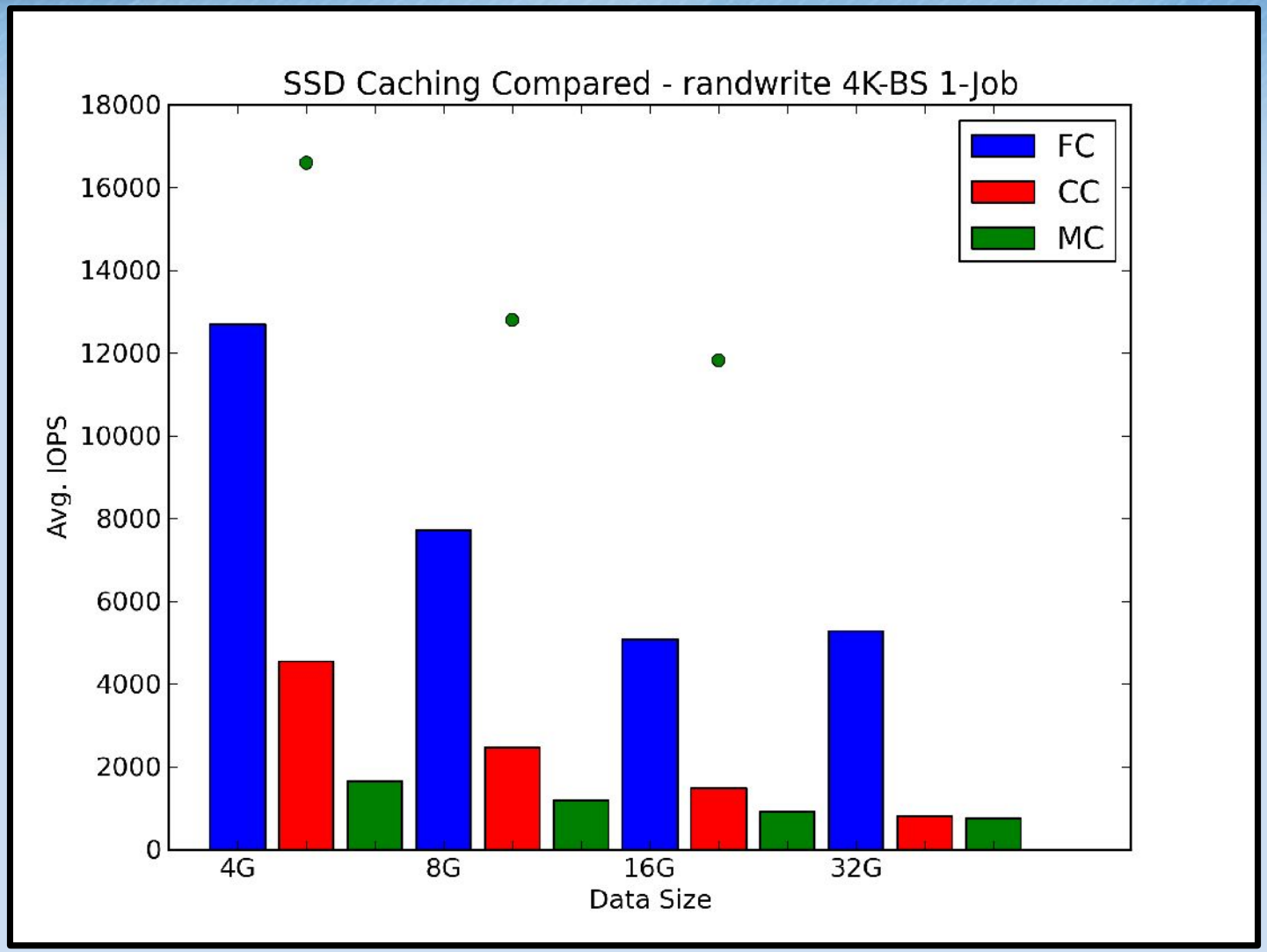
Comparison – Bandwidth write



Comparison – Bandwidth randread

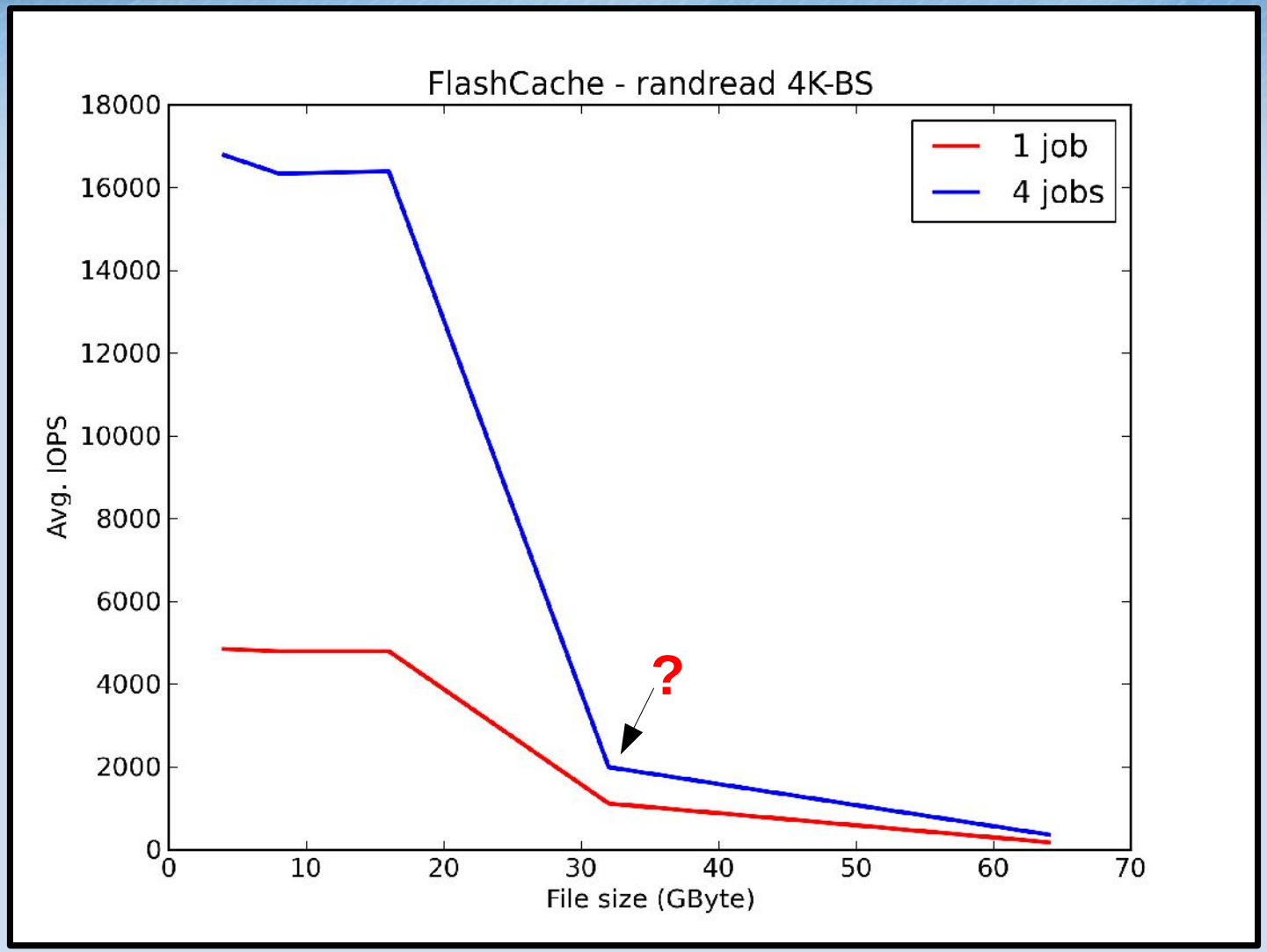


Comparison – Bandwidth randwrite

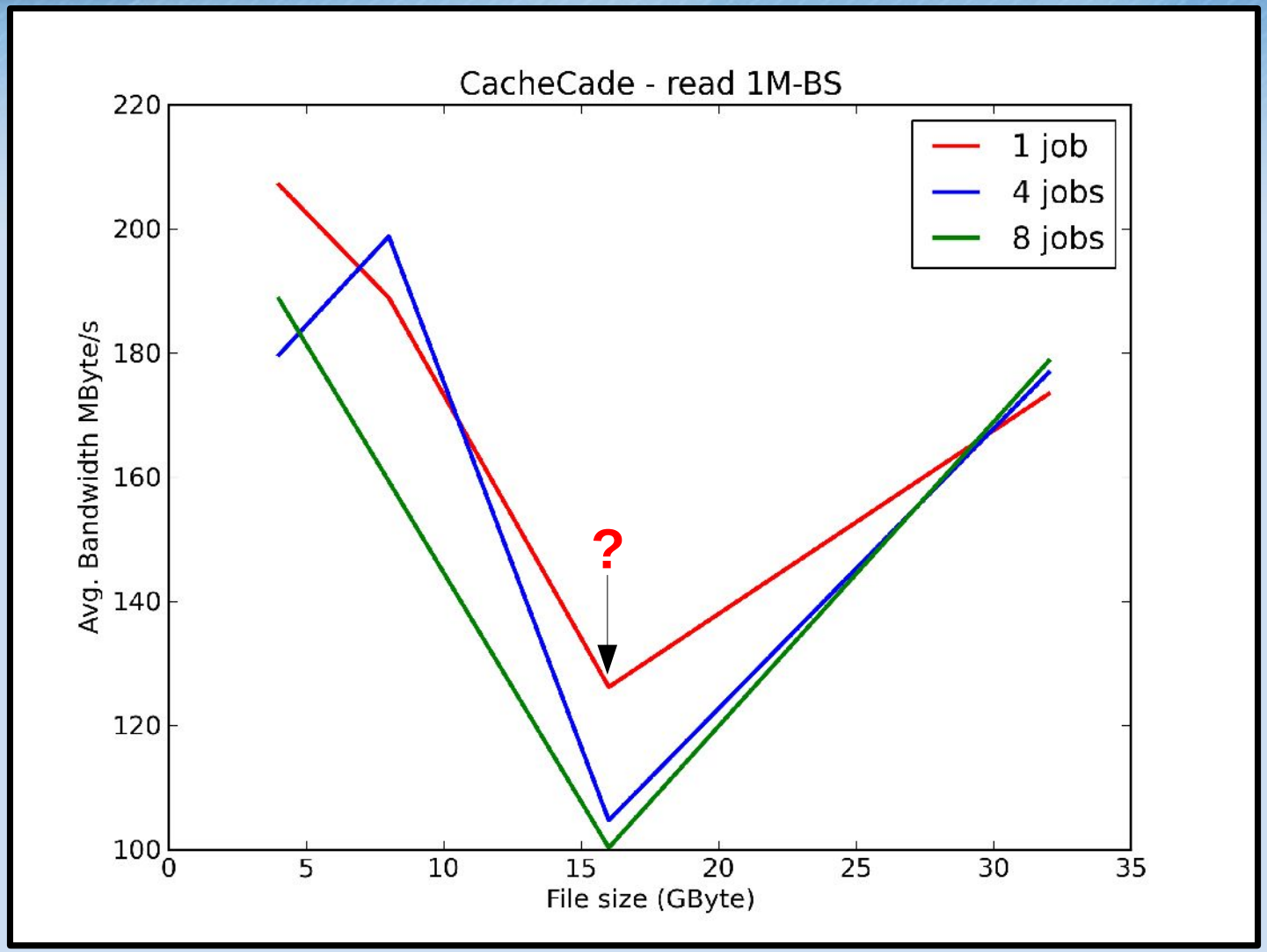


Still to analyze...

FlashCache

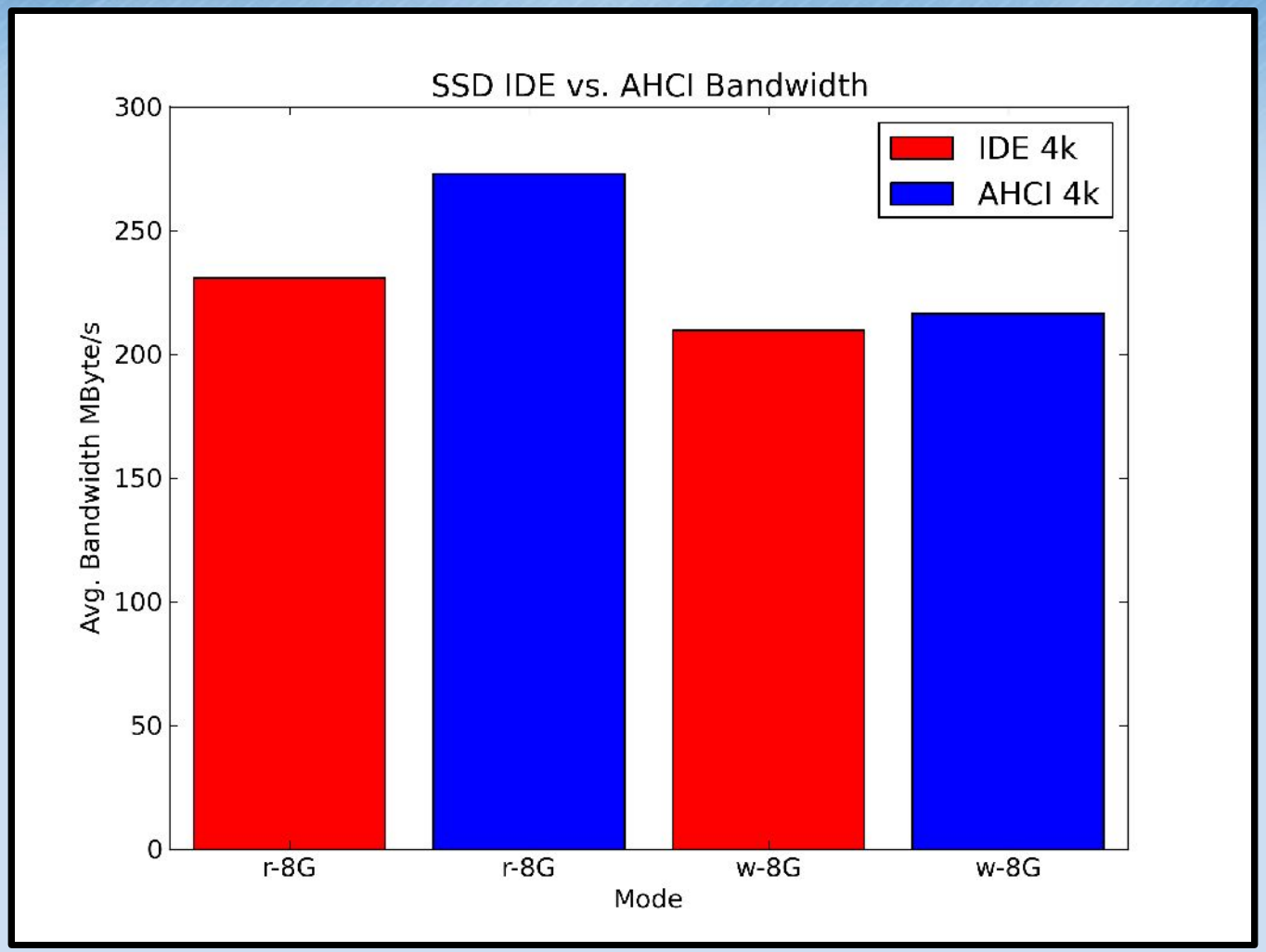

FlashCache - randread 4K-BS
1 job
4 jobs
Avg. IOPS
File size (GByte)
?
0
2000
4000
6000
8000
10000
12000
14000
16000
18000
0
10
20
30
40
50
60
70

CacheCade

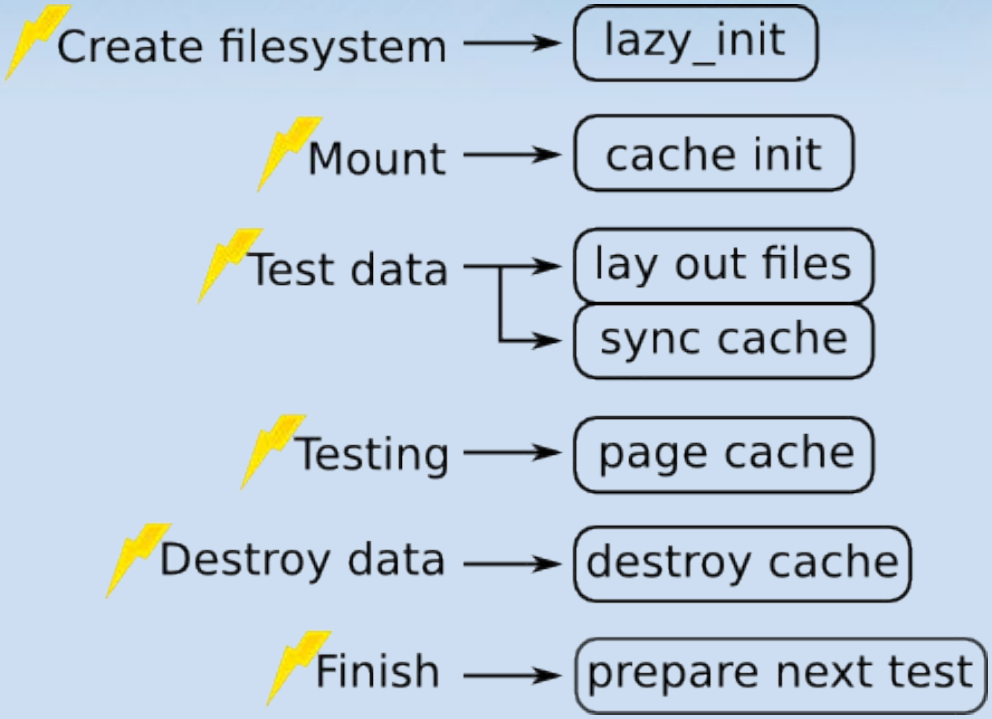

CacheCade - read 1M-BS
1 job
4 jobs
8 jobs
?
Avg. Bandwidth MByte/s
File size (GByte)
220
200
180
160
140
120
100
0
5
10
15
20
25
30
35

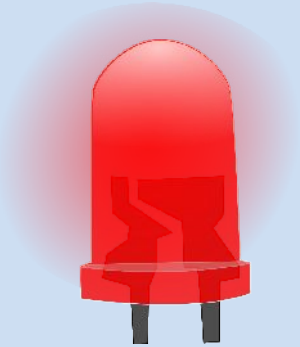
Summary – avoiding errors while testing...

SSDs – A well known error



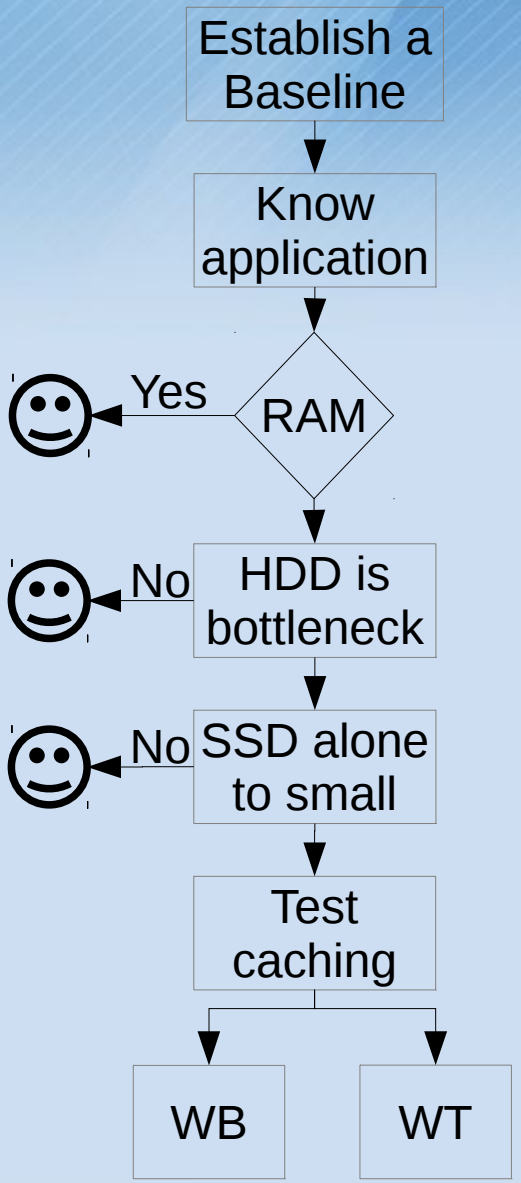
Testing caveats





Lessons Learned

- **Test your own application**
  - Get a comfortable testing tool
    - I prefer Fio (git://git.kernel.dk/fio.git)
  - Know how and what to test
- **Check your test results twice**
  - Visualize your results
  - Test numbers against the real world
- **For a first attempt**
  - Test with FlashCache



Questions?

Do not hesitate to ask us

(you can do it per mail also)

You want some test protocols?

You need a special chart?

Thanks for your time

Visit our booth -
Open Source Sponsorship
and Low Energy Server

LINUX TAG

Thomas-Krenn.AG®
The server experts

TK SERVER

References

- http://greenarrow.deviantart.com/art/The-Flash-Wally-West-35885287
- https://upload.wikimedia.org/wikipedia/commons/6/62/2006-07-26AbendhimmelRemstal19.jpg
- http://openclipart.org/detail/15238/tortue1-by-roym
- http://openclipart.org/detail/34657/architetto---struzzo-by-anonymous
- http://openclipart.org/detail/11118/check-sign-and-cross-sign-by-jetxee
- http://openclipart.org/detail/124837/led-lamp-by-eggib
- http://b--art.deviantart.com/art/Riddler-62371451

Backup Slides

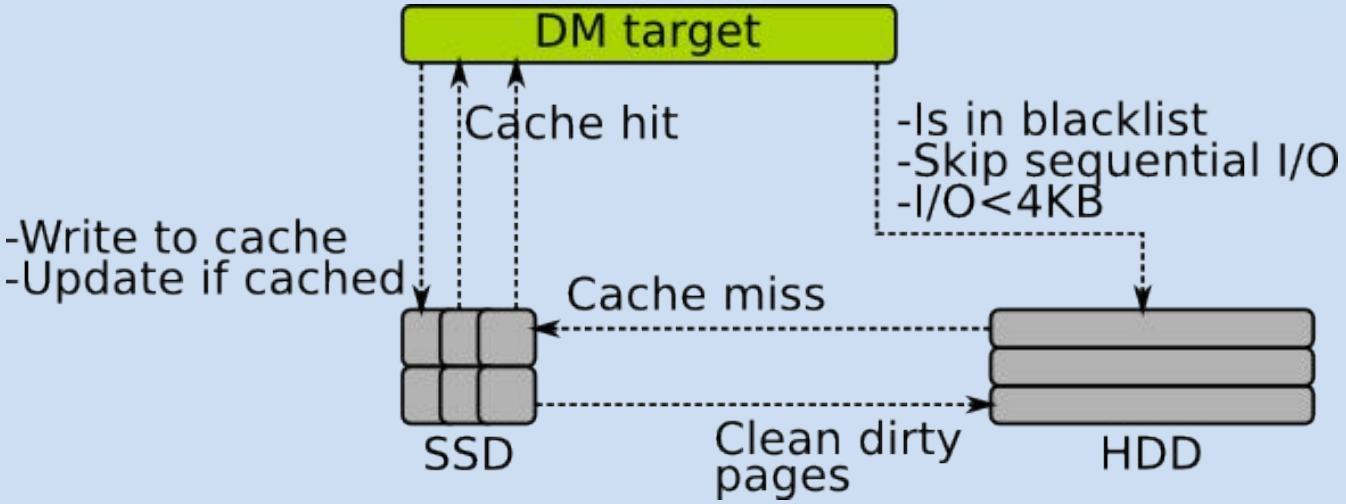
Test system

- **SSDs**
  - Intel Series 320 160GB
  - Via HPA reduced to 32GB
- **RAID Controller**
  - LSI MegaRAID SAS 9260-4i
  - Adaptec 6805Q
- **Software**
  - Fio 2.0.7
  - Ubuntu 12.04
    - Updates from Release Day

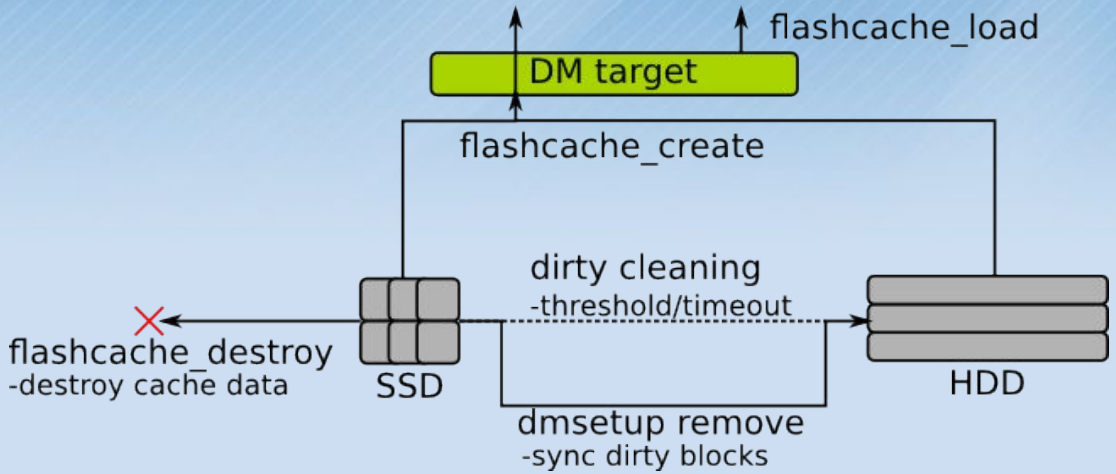
Flashcache – Test Script

- **flashcache_create**
  - Create a WB caching device
- **cache_all=0**
  - Don't cache ext4 initialization
- **mkfs.ext4 -q -E lazy_itable_init=0, lazy_journal_init=0 /dev/mapper/fc-root**
- **mount /dev/mapper/fc-root**
- **cache_all=1**
- **Call fio**
- **umount /dev/mapper/fc-root**
- **dmsetup remove**
- **flashcache_destroy /dev/sdd**

Flashcache – Workflow



Flashcache – Commands



- **Just a few examples**
  - flashcache_create -v -p back fc-root /dev/sdd /dev/sdc
  - sysctl -w dev.flashcache.sdd+sdc.cache_all=0
  - dmsetup table
  - flashcache_destroy /dev/sdd