
CERN

# Addressing the VM IO Bottleneck (or: The “lxplus problem”)

Arne Wiebalck

HEPiX Autumn Meeting
Lincoln, Nebraska, U.S.
Oct 15, 2014

# lxplus

- *P*ublic *L*ogin *U*ser *S*ervice for Linux
  - Interactive login cluster with software for s/w dev, batch submission, AFS, mail, …
- 185 servers with ~30-50 users each
  - 140 SLC6 (“lxplus”, virtual, 8-core/16GB)
  - ~~45 SLC5 (“lxplus5”, physical, stopped this week)~~
- “Reference” and gateway service

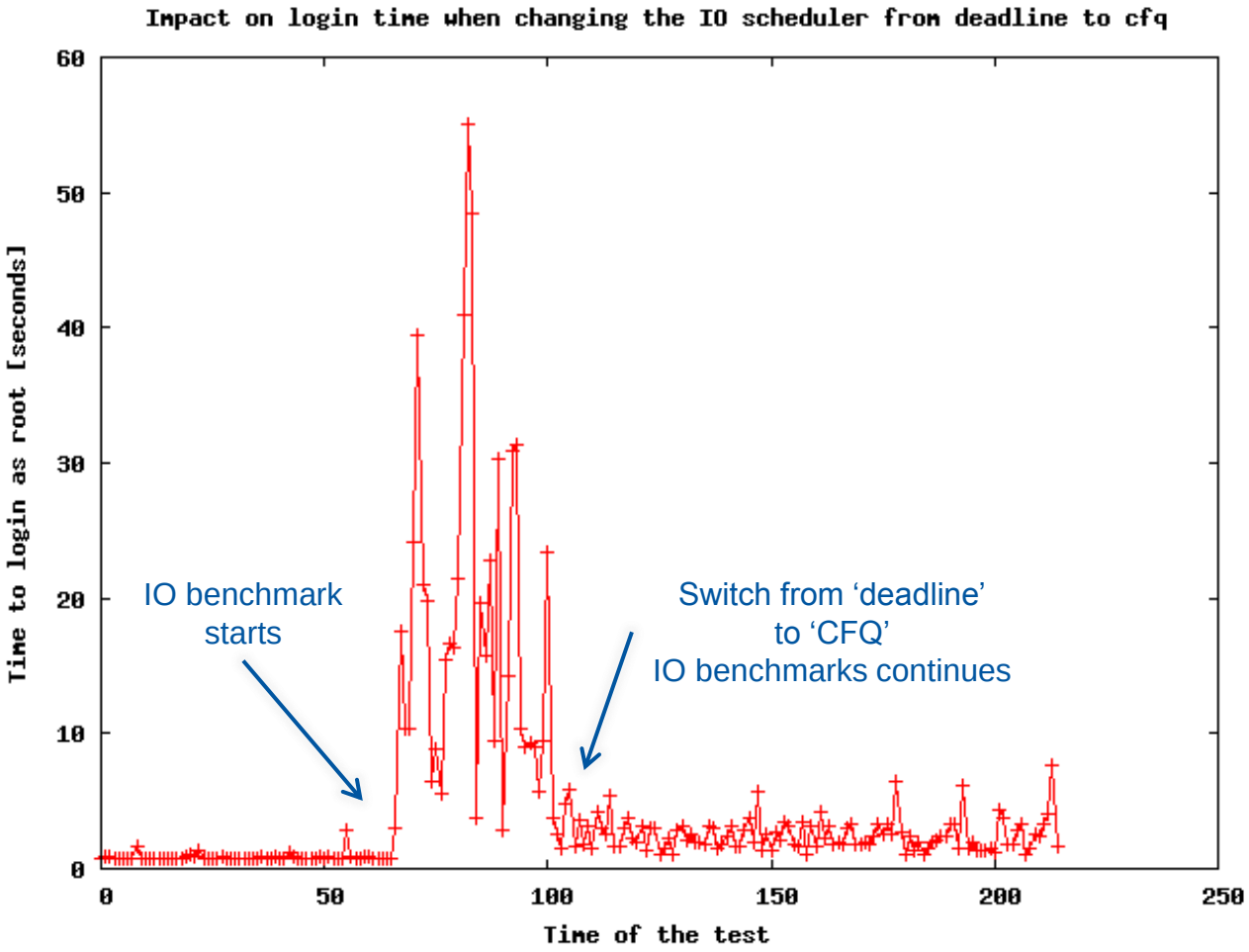
# The “lxplus problem”

- Erratic login times on SLC6 VMS
  - From 1 – 30 seconds before getting a prompt
- Virtual disk performance suspected
  - DB access of sssd (System Security Services Daemon)
  - ioping on the VM confirmed erratic IO performance
- No surprise: 140 servers on ~50 hypervisors
  - Up to 4 servers per hypervisor
  - 80 IOPS of HV’s disk shared by 4 virtual drives
  - 30 seconds still long → VM IO starvation?
    (updatedb and scrub caused a similar problem on the Ceph servers earlier)

# IO Scheduling

- Plus VMs used ‘deadline’ elevator
  - Set by ‘virtual-guest’ tuned profile, RH’s default for VMs
  - Not always ideal for interactive machines:
    ‘deadline’ prefers reads, can delay writes (default: 5 secs)
  - Made to allow reads under heavy load (webserver)
  - sssd makes DB updates during login
- IO scheduler on the VM changed to CFQ
  - Completely Fair Queuing
- Benchmark: login loop

# ‘deadline’ vs. ‘CFQ’

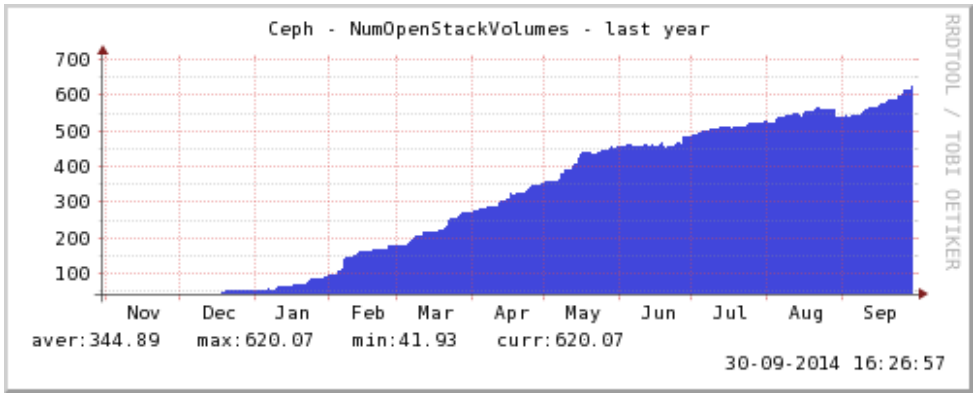


# Additional Measures

- Base load on hypervisor's disk 15-20%
- Reduce overall IO load
  - Configured sssd DB in tmpfs
  - Configured AFS cache in memory
  - Optional: Ceph volumes for /tmp
- However, in the end there are only 80 IOPS
  - Divided by number of VMs (plus hypervisor itself)
  - Users can still end up with pretty slow disk …

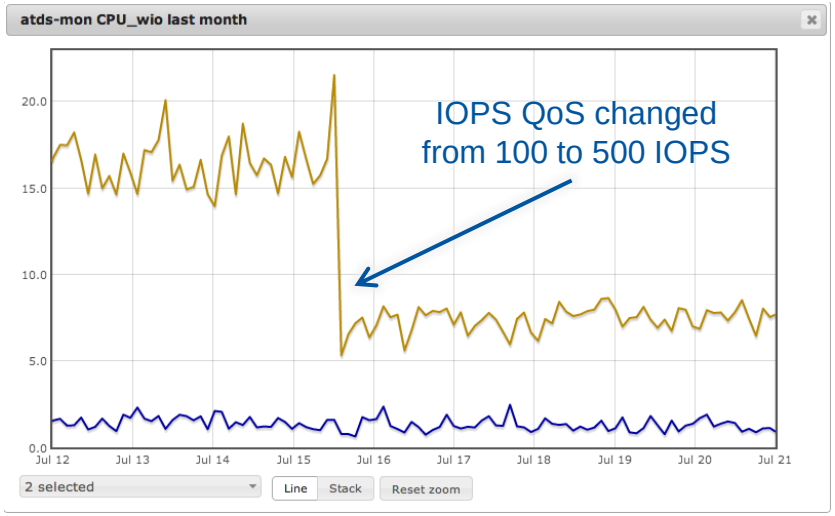
# IOPS++ Option 1: Cinder volumes

- Ceph volumes in production since Feb 2014




- QoS for IOPS and bandwidth
  - Default: 100 IOPS, 80 MB/s
  - Controllable at runtime!
  - IOPS are a limited resource (some 20-30k IOPS currently)
  - Will improve by a factor 4-5 once SSD write journals are in place

# Ceph volumes at work

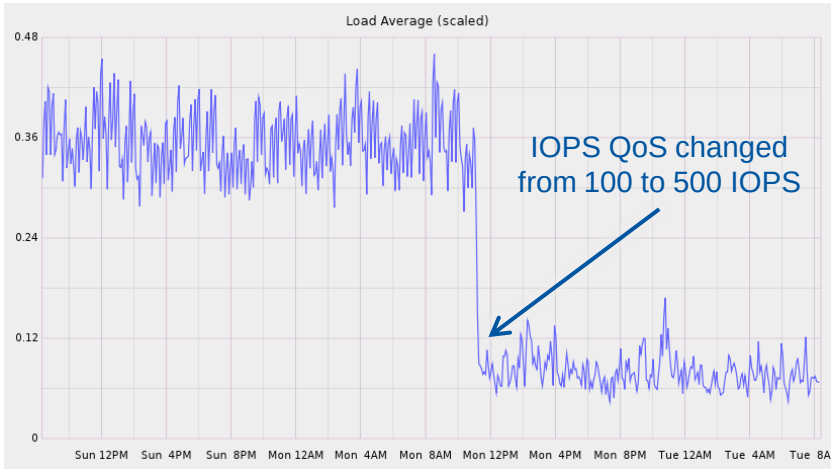



ATLAS TDAQ monitoring application

Y- Axis: CPU % spent in IOwait

Blue: CVI VM (h/w RAID-10 with cache)

Yellow: OpenStack VM

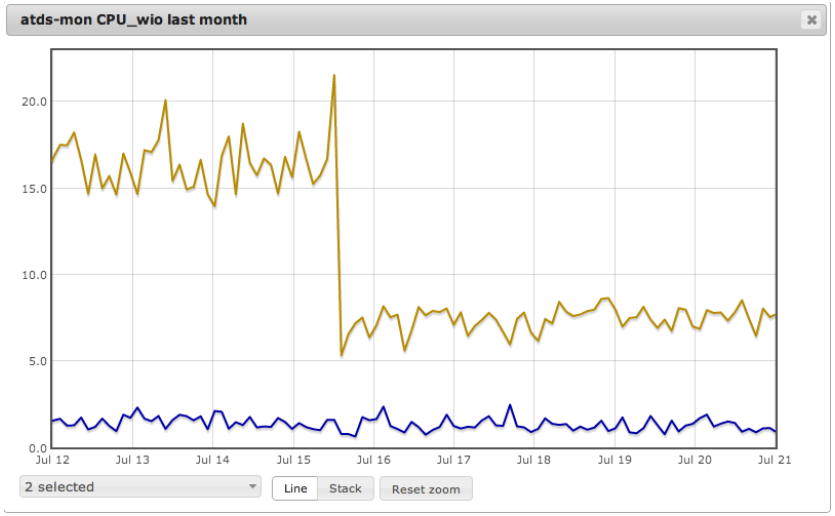



EGI Message Broker monitoring

Y- Axis: Scaled CPU load

(5 mins of load / #cores)

# IOPS vs. Latency




Gap between CVI and OpenStack.

Going to 1000 IOPS doesn't help.

If the application is not designed to keep enough requests in-flight, more IOPS do not reduce the time spent in IOwait.

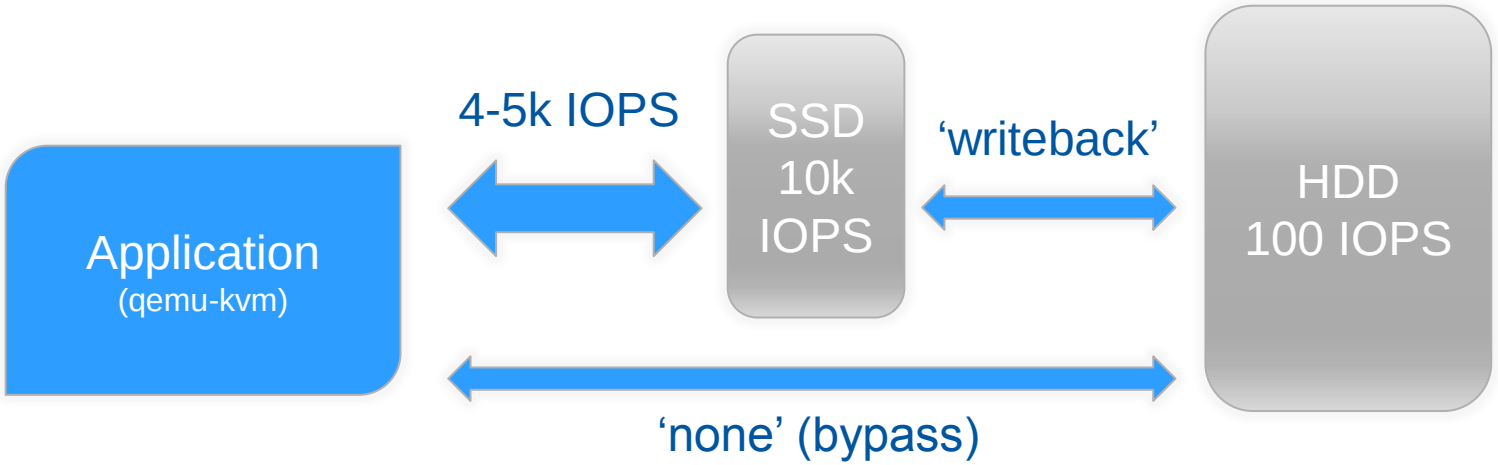
We need to reduce the latency as well …

# IOPS++ Option 2: SSDs

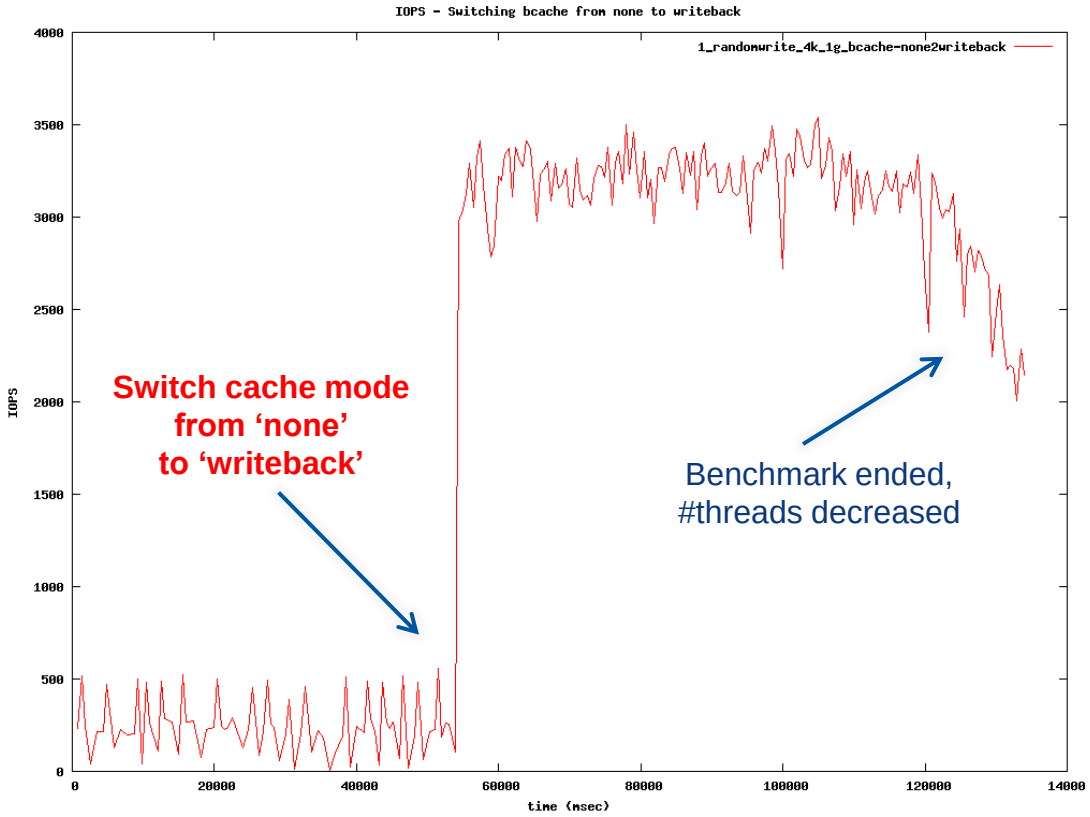
- SSDs as disks in hypervisors would solve all IOPS and latency issues
- But still (too expensive and) too small
- Compromise: SSD block level caching
  - flashcache (from Facebook, used at CERN for AFS before)
  - dm-cache (in-kernel since 3.9, rec. by RedHat, in CentOS7)
  - bcache (in kernel since 3.10)

# bcache

- Features
  - Change cache mode at run-time (think SSD replacements)
  - Strong error-handling capabilities
  - Easy to set up
  - Operational drawbacks
    - recent kernel (3.15+), SPOF in our current h/w setup



# bcache in action (1)



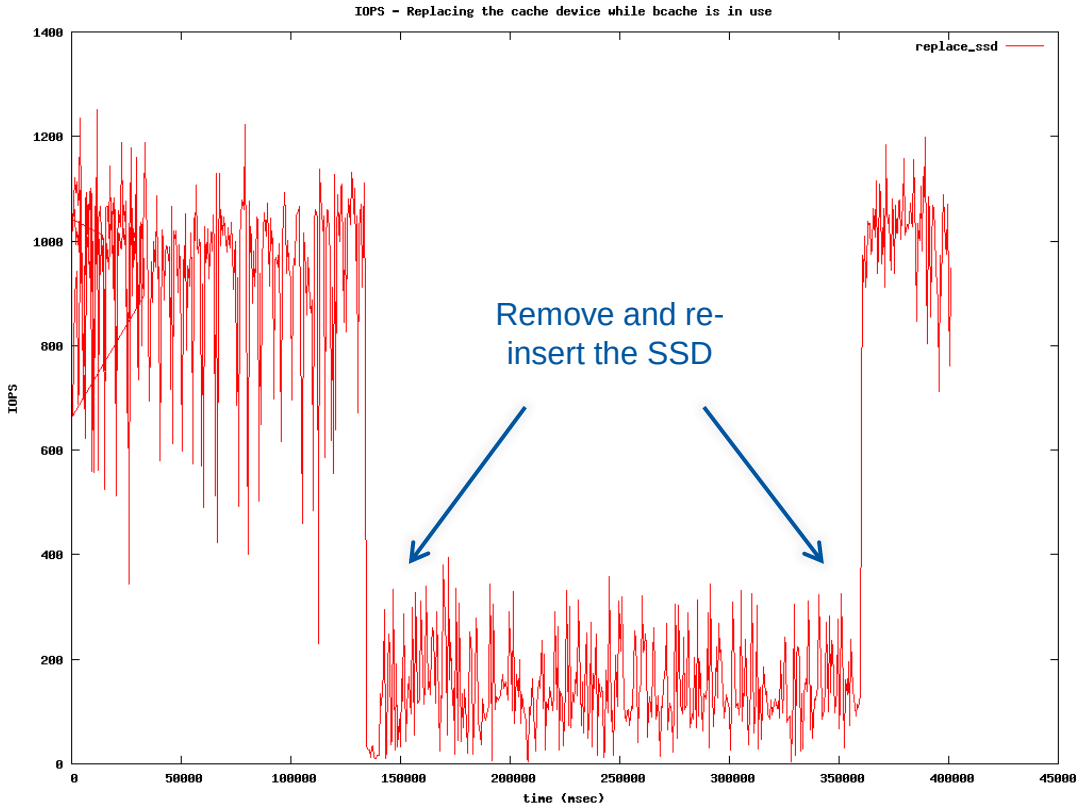

On a 4 VM hypervisor:

~20 IOPS/VM → ~1000 IOPS/VM

Benchmarking a caching system is non-trivial:

- SSD performance can vary over time
- SSD performance can vary between runs
- Data distribution important (c.f. Zipf)

# bcache in action (2)

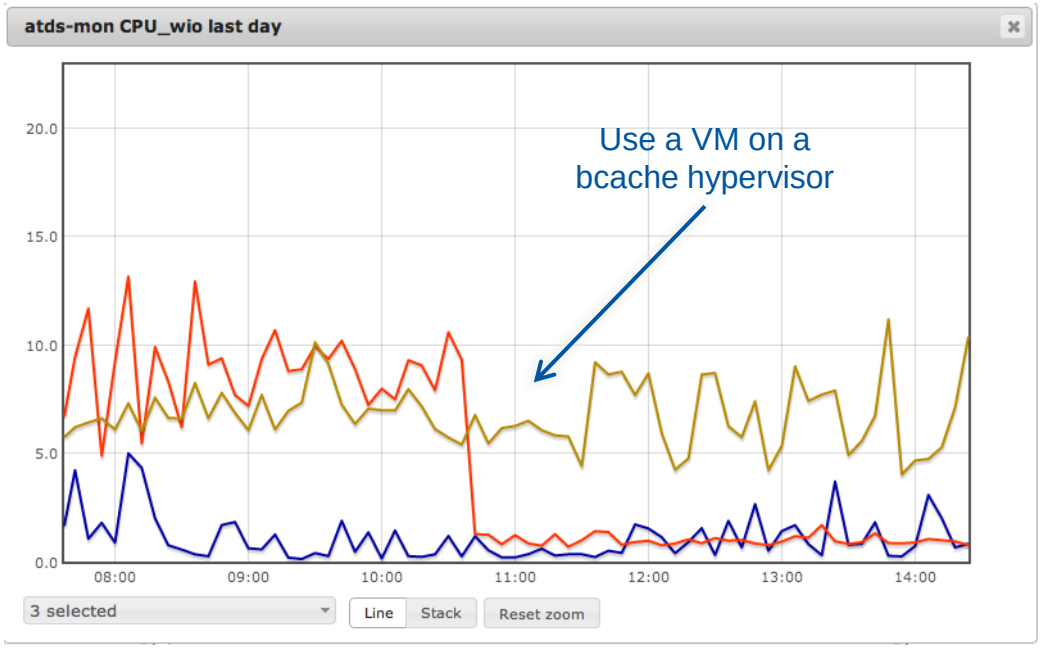



## Simulate SSD wear-out replacement procedure:

While fio is running:

- Switch cache mode to ‘none’
- Wait for dirty flag to become 0
- SCSI-remove the SSD
- SCSI-insert the SSD
- Add the SSD to the cache set

# bcache and Latency




SSD block level caching sufficient for IOPS and latency demands.

Will be very similar once the Ceph write journals will be on SSDs (planned for Q4/2014).

Blue: CVI VM (h/w RAID-10 w/ cache)

Yellow: OpenStack VM

Red: OpenStack on bcache HV

# Conclusions & Outlook

- Analyzing our workload and adapting our IO scheduler helped
  - Eliminated erratic login times
  - CERN Ceph servers (‘deadline’ → tuned ‘deadline’ → ‘CFQ’)
- SSD block level caching is one option to mitigate the VM IO bottleneck we face
  - bcache: lxplus VMs go from 20 to 1000 IOPS each (more with Intel DC S 3700!)
  - We will deploy bcache all lxplus hypervisors (and some additional HVs)
  - We’re looking into a similar option for Hyper-V
- Cinder volumes on Ceph are an alternative
  - Mind the difference between providing IOPS and latency
  - CERN Ceph servers will get SSD-based write journals
  - Not for everything: e.g. AFS caches, /tmp

# Questions?