

Automated virtualisation performance framework

Tim Bell

Sean Crosby (Univ. of Melbourne)

Jan van Eldik

Ulrich Schwickerath

Arne Wiebalck

Virtualisation performance

- Expected performance is hard to quantify on virtual machines, especially for end-users
 - They have very little idea of hypervisor hosting machine, let alone of performance of hypervisor
- Simplistic idea of performance would be to compare hypervisor performance to virtual machine
 - But what do you compare it to? Hypervisor can be different OS to virtual machine
 - Do you take into account hyperthreading? Turbo boost?
 - CERN takes it as being $\text{perf}(\text{VM})$ vs $\text{perf}(\text{original spec})$, where original spec is what the hardware team found the performance was when bought
- Original performance problems on CERN Cloud were not noticed by experiments or cloud team
 - It was actually first identified by Batch team

Monitor performance

- Experiments will notice changes in performance, or performance inconsistency more than absolute performance
 - Refer to Arne's talk
- Many things change on hypervisors
 - Weekly distro sync
 - Recent upgrade to CC7 from SLC6
 - QEMU/kvm just updated
 - Openstack upgrades
- Make sure these changes don't have adverse effect on performance
- We have regular QA tests for package updates
 - Extend it to test performance of VMs and hypervisors
 - Isolate specific hypervisors (and VMs on those) to just run benchmarks

Automated test framework

- Ability to schedule tests

- Rundeck



- Used internally for quota updates, notifications...
- Has login access to all nodes

Schedule to run repeatedly?

No Yes

Simple

Crontab

0 47 20 ? * * *

Ranges: 1-3. Lists: 1,4,6. Increments: 0/15 *every 15 units starting at 0*.

See: [Cron reference](#) for formatting help.

Automated test framework

- Store all benchmarks and graph trend over time
- Elasticsearch/Kibana
 - Used extensively by CERN
 - Monitoring team created plugin to send benchmark results to ES using Lemon
 - HS06 benchmark of most hardware when bought has been imported
 - Kibana has metrics of host (load, memory usage etc), so easy to cross reference metrics with benchmark
 - Can share dashboards with experiments to show the current performance of VMs



Automated test framework

- Need to profile multiple flavor types
 - 4, 8, 16, 32 core VMs
 - Need to ensure benchmark runs at same time on all VMs on hypervisor
- Apache ZooKeeper
 - Use ZooKeeper barriers to sync start and end of benchmark



Automated test framework

- Support multiple benchmarks
 - HS06
 - Every experiment has their own “benchmark in progress”
 - condor_mips
 - Fast LHCb

Automated test framework

- Python “pilot” script
 - Copied to hosts and run by Rundeck
 - Pass benchmark, tag and start time as argument
 - Joins ZooKeeper barrier, and waits for all participants to be ready
 - Runs benchmark, and captures result
 - Waits for all participants to finish
 - Cleans up temp files
 - Writes result to JSON file
 - Gets read by Lemon sensor and uploaded to ES

Graphs

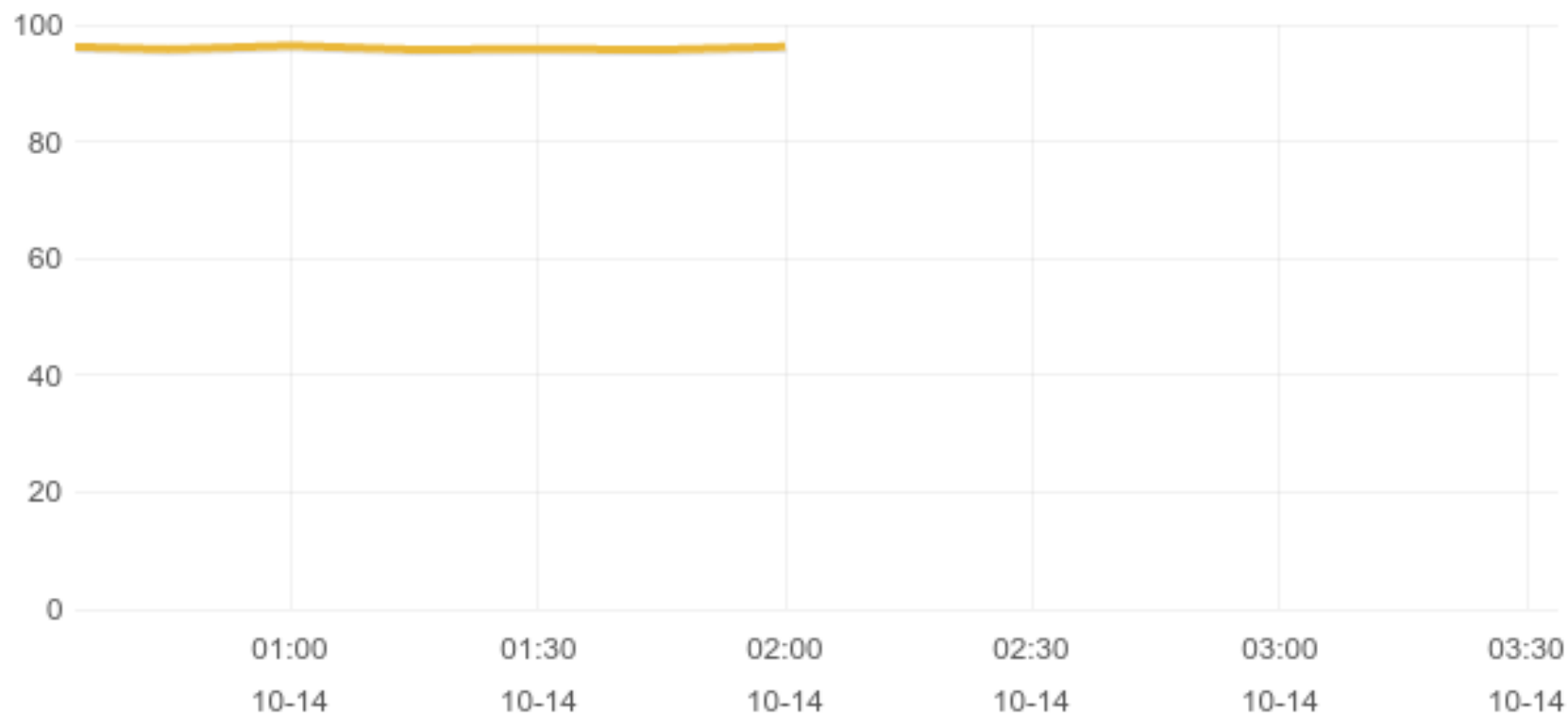


Graphs

16 CORE ON P05799597Q26316



[View](#) | [Zoom Out](#) | ● total(((@message.tag:sean_new_try).@message.hepspec) (24) per 15m | (24 hits)



Further work

- Isolate one (or more) of each HW type and start testing them
- Some of the possible causes of perf problems won't be picked up at the moment
 - QEMU/KVM version used by VM only changes when VM is recreated or hard rebooted
 - Same goes for shared libraries
- Integrate creation/destroy of VM for some hypervisors to pick up all changes

