Table of Contents

Table of Content

Subjects for Studies

HEP reference workloads in containers

- Dedicated page link

SPEC CPU 2017

- Compare HS06 and SPEC CPU 2017 scores
    - Correlation between SC17 and HS06
        - Very high correlation, measured on 7 different Intel CPU models
        - Not all scores are independent
        - Results reported here
    - Studies at the micro-code level (Trident)
        - ref

- Compare SPEC CPU 2017 with HEP jobs
    - Initial comparison based on grid jobs ref
    - Need to identify HEP reference workloads

Spectre, Meltdown,

- Evaluate performance effect of the patches
    - Several independent measurements performed, embracing WLCG workloads and HS06
    - All confirm that the performance degradation is within 1%-5%
    - ref
    - L1TF: effect within 2% (ref

HS06

- Shall HS06 be still run in 32-bit or in 64-bit (-m32 Vs -m64)
    - Discussion started in the mailing list. Motivations
        - New architectures can be only tested in 64-bit
        - The experiment applications are in 64-bit
        - Scattered studies have reported a ratio of 20% among -m32 and -m64. Is this ratio constant for all CPU models?
    - Results reported in here
    - Conclusions:
        - HS06 score would change of ~15% moving from 32 to 64 bits
        - Factor is different for different CPU models, but within 5%
        - **A change of the official procedure is not justified**

- HS06 variation with OS
    - Results reported in here
    - Conclusions: variations within few percent

- HS06 correlation with Experiment workloads
    - HS06 doesn't scale anymore (in new Intel CPU models) with simulation workloads.
    - Lack of "magic boost" seen for experiment applications.
        - Recent reports from LHCb and Alice
    - What's the situation for Reconstruction workloads?

        - What's the situation for Atlas and CMS workloads?
        - Status:
            - Alice and LHCb workloads not scaling anymore with HS06 (a.k.a. Haswell magic boost)
            - Independent studies still show agreement within 10% for Atlas and CMS workloads
            -

DB12

- DB12 boost in Haswell and Broadwell
    - Investigated by M. Guerri. Reason found to be due to the better branch prediction
    - pre-GDB
    - notebook

- DB12 variation with different OS and python versions
    - Is DB12 affected by different python or OS versions, on the same CPU model?
    - Studies here

- DB12 performance with SMT ON/OFF
    - Respect to HS06 DB12 doesn't seem to benefit from SMT enabled respect to the 20% seen in HS06
        - Studies on VMs benchmarking full HW node
            - Plot showing the Relative performance of DB12, WSN,KV and HS06 across different VM configurations. The performance of VM-83A is used as reference
            - Full presentation
        - Tables here

- DB12 Vs multi-core jobs performance
    - Is DB12 well correlated with the execution time of multi-core jobs, such as the ones running in ATLAS and CMS?

KV

- KV double peak effect
    - Investigated by M. Guerri. Is related to different performance of the first core of a dual-socket server respect to all the other cores.
    - pre-GDB
    - CERN internal note: Profiling CPU-bound workloads on Intel Haswell-EP platforms

- Reduce initialisation time for KV
    - the athena applications runs in ~2 mins to process 100 single muon events, but the initialization time (sw-mgr application) can take up to 3 additional minutes. Can initialization be reduced?
        - A slim implementation of the KV benchmark is available in Docker container
            - To run docker run -it --rm gitlab-registry.cern.ch/giordano/hep-workloads:atlas-kv-bmk-v17.8.0.9
            - gitlab repository
            - Further details described in this talk

- KV License
    - Atlas code is now in github with Open Source licence

Resources Available to Run Benchmarks

GridKa has reconfigured its compute farm to enable special benchmarking tasks:

- An open issue is the correlation of static benchmark results (like HS06, or DB12-at-boot) with applications, depending on the number of configured job slots. Therefore there are several flavors of worker nodes, for instance:
    - Intel Xeon E5-2630v4 (Broadwell, 10-core, Hyperthreading enabled):
        - 20 job slots (1.0 slots per physical core)
        - 32 job slots (1.6 slots per physical core)
        - 40 job slots (2.0 slots per physical core)
    - Intel Xeon E5-2630v3 (Haswell, 8-core, Hyperthreading enabled):
        - 24 job slots (1.5 slots per physical core)
        - 32 job slots (2.0 slots per physical core)
    - Intel Xeon E5-2665 (Sandy Bridge, 8-core, Hyperthreading enabled):
        - 16 job slots (1.0 slots per physical core)
        - 24 job slots (1.5 slots per physical core)
- The static benchmark scores are available to all batch jobs (submitted to either arc-1-kit.gridka.de, arc-2-kit.gridka.de, or arc-3-kit.gridka.de) using the machine job features (MJF):
    - $JOBFEATURES/hs06_job: HS06 score available to the job
    - $JOBFEATURES/db12_job: DB12 score available to the job
    - $JOBFEATURES/allocated_cpu: number of single-core job slots provided to the job
- Manfred Alef at KIT can provide static benchmark scores afterwards; please send a CVS (or Excel or ODF spreadsheet) file which contains at least the worker node hostnames and the individual performance (events/s) of the jobs

CERN

A number of resources can be made available for testing, based on bare metal servers or whole node VMs. Access, based on ssh public key, can be provided on demand.

* List of available resources (this list can change following the needs of Tier-0 resources)

Type	CPU model	OS	N cores	N machines
Bare-metal	Intel(R) Xeon(R) CPU E5-2650 v2 @ 2.60GHz (Ivy Bridge)	SLC6.8	32	2
VM	Intel Xeon E5-2630v3 (Haswell)	CC7 - x86_64	32	2
VM	Intel Xeon E5-2630v3 (Haswell)	SLC6 - x86_64	32	2
VM	Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20GHz (Broadwell)	SLC6 - x86_64	40	2

Other sites that would like to join

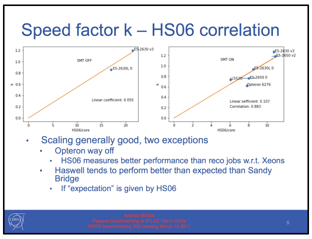
TBD: please describe the kind of resources available, the configuration and how it's possible to access them

Recipes to Run Experiment Workloads

Collect here the information about how to run experiment workloads. Possibly, provide instructions and setup (VM/containers , access from cvmfs) in order to allow execution by other members of the working group.

- ALICE
    - Contact person
    - Version of the experiment application (details about compiler flags)
    - Event Generation
    - Simulation
    - Digitization
    - Reconstruction
- ATLAS
    - Contact person
    - Version of the experiment application (details about compiler flags)
    - Event Generation
    - Simulation
    - Digitization
    - Reconstruction
- CMS
    - Contact person
    - Version of the experiment application (details about compiler flags)
    - Event Generation
    - Simulation
    - Digitization
    - Reconstruction
- LHCb
    - Contact person
    - Version of the experiment application (details about compiler flags)
    - Event Generation
    - Simulation
    - Digitization
    - Reconstruction

Passive Benchmark

- A method to compare server performance using the experiment job information
- Responsible: Andrea Sciaba (andrea.sciaba@cernNOSPAMPLEASE.ch)
- Description of the approach and results at pre-GDB and WG meeting
- Some results:
    - Speed factor k Vs HS06 correlation for ATLAS T0 jobs: 

- Data required to run the passive benchmark

Quantity	CMS variable	ATLAS Grid jobs variable	ATLAS T0 variable
CPU time	CpuTimeHr	cpuconsumptiontime	cpuTime
Number of events in job	KEvents	nevents	nevents
Job status	Status	jobstatus	n/a
Job type	TaskType	processingtype	n/a
Site name	Site	computingsite	n/a
Task	WMAgent_SubTaskName	jeditaskid	taskid
CPU model	n/a	cpuconsumptionunit	machine.model_name

Actions List

2017-03-10

- For the site representatives: to fill the information in this section
- For the experiment representatives: to fill the information in this section
- For Andrea Sciaba': to fill the information in this section

2017-04-19

- WLCG workshop: preparation of topics for discussion

-- ManfredAlef - 2016-06-03

This topic: HEPIX > CpuBenchmark
Topic revision: r32 - 2021-01-21 - DomenicoGiordano