Benchmarking Commercial Cloud Resources

D. Giordano
(CERN IT-SDC)

White Areas Lecture
16 December 2015

Acknowledgments

- Work done in collaboration with
  - SDC Cloud Team (Twiki)
  - Vladimir Petrov (Master Thesis student project)

- A similar talk was given at HEPiX Workshop, Fall 2015
  - This is a longer version
    - Including more details and plots
    - Recent findings in benchmarking commercial clouds

Performance Measurement

- *"Performance is a key criterion in the design, procurement, and use of computer systems […] to get the highest performance for a given cost."*
- *"The types of applications of computers are so numerous that it is not possible to have a standard measure of performance […] for all cases."*
- *"The first step in performance evaluation is to select the right measures of performance, the right measurement environments, and the right techniques."*
- *"The process of performance comparison for two or more systems by measurements is called benchmarking, and the workloads used in the measurements are called benchmarks."*
  - From "*Art of Computer Systems Performance Analysis Techniques For Experimental Design Measurements Simulation And Modeling*"
    - by **Raj Jain** , Wiley Computer Publishing, John Wiley & Sons, Inc
    - 1992 Computer Press Award Winner

'80-'90 CERN Unit

GG CERN-CN 92-13

su 9304

9

CERN - Computing and Networks Division
CN/92/13
December 1992

Benchmarking Computers for HEP

Eric McIntosh

CERN, Geneva, Switzerland

Abstract

This report summarises the results of the CERN benchmark tests carried out on a variety of Mainfames and Workstations during the last fifteen years. The tests are a suite of FORTRAN programs used to determine the CPU power of a computer system for running High Energy Physics applications. They are essentially scalar due to the well known difficulties in vectorising this type of application, but a matrix inversion in

convenience and continuity I decided to keep the CPU metric, but reduce the number of production codes. I was able to make GABI (now CRN5), JAN (CRN12), and FOWL (CRN3), reasonably easy to port and I added another modern event generator LUND (CRN4) to give a 50/50 distribution between event generation and reconstruction as that was the workload distribution at the time. I supplemented these codes with several "kernel" type applications to at least get a feel for compilation times (CRN4C), vectorisation

- https://cds.cern.ch/record/245028/files/CM-P00065729.pdf

Performance in Cloud Environments

- Performance measurement and monitoring are essential
- Additional benefits in a commercial cloud environment (private – public – hybrid clouds)
  - Deal with the intrinsic variability and inhomogeneity
  - Compare the presumed and perceived performance
  - Quickly identify performance issues

In The Next Slides

- Experience gained in procuring commercial cloud resources and connection with benchmarking
- Description of the Benchmark Suite implemented to collect and analyse performance metrics
- Some study results

- NB: Focus on CPU performance metric

Caveat

- Other benchmarking activities not covered in this talk
  - HEP-SPEC 06 benchmarking
    - The official CPU performance metric used by WLCG sites since 2009
      - It meets the HEP requirements:
        - » Percentage of floating point operations observed in batch jobs
    - Component of the procurement process for bare-metal servers
    - References: HEPiX Benchmark WG
  - Passive benchmarking
    - CPU power normalization based on multivariate fit
      - Use real workloads (experiments' jobs) as not-calibrated metrics
    - Christian Nieke (IT-DSS-DT)
    - References: Analytics WG meeting

Cloud Procurement & Benchmarking

CERN Cloud Procurement

- Started in 2011 within the Helix Nebula partnership among leading research organizations and European commercial cloud providers
  - Objective is to support the CERN's scientific computing programme
    - Integrate commercial cloud IaaS within the experiment frameworks
    - Improve the CERN procurement process for cloud IaaS
    - Evaluate cost and benefit of cloud IaaS

CERN Cloud Procurement Roadmap

- ☑ First Procurement, March '15
  - Target a single VO, run simulation jobs
- ☑ Second Procurement
  - Production activity currently running (started in November '15)
  - Target multi VOs, simulation jobs
- ❑ Third Procurement
  - Production activity to start during Spring '16
  - Target multi VOs, full chain processing
- ❑ EC co-funded joint Pre-Commercial Procurement (PCP) HNSciCloud project ('16-'18)
  - More details in R. Jones talk at Nov. GDB

Commoditize Cloud Resources

- Benchmarking needs during the procurement process
  - Define technical specs, adjudication criteria and remediation options
    - *Request a VM able to process at least N reference events/sec*
    - *Verify that resources are delivered according to the specified performance*
    - *Compare offers based on cost/event*
      - NB: Current CERN adjudication rules are based on cheapest compliant price and not best value for money

	cost ratio	cpu/evt [s]	cpu/evt ratio	cost/evt (ratio) ratio_(cost/h *s/evt)
CloudA	1.00	*0.94*	1.00	**1.00**
CloudB	3.04	*1.13*	1.20	**3.65**
CloudC	2.04	*3.36*	3.57	**7.31**

- Enable a "cloud" commodity exchange based on a measurable value
  - See Deutsche Boerse Cloud Exchange approach
    - *"Benchmarks are used to define a PU. The Compute Product and Memory Product are currently only available as Performance level regular"* (http://cloud.exchange/Product-Concept/DBCE-Product-Concept.pdf)

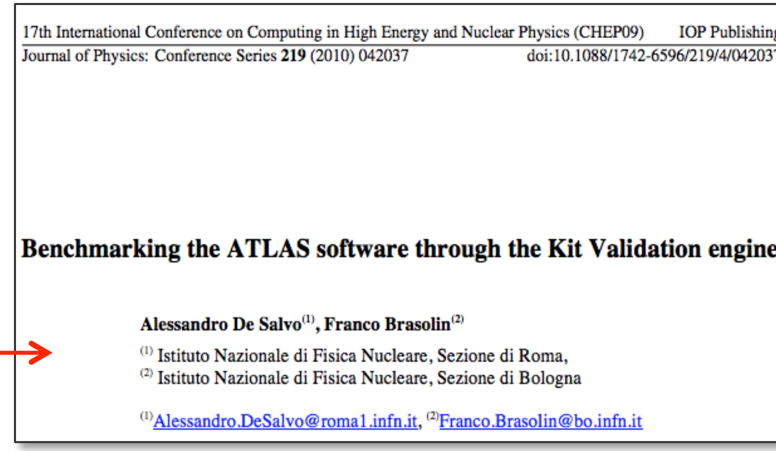
Benchmarking in the Procurement Phase

- Started ~1 year ago for the preparation of the first CERN cloud procurement (Production in March '15)

- Evaluated different alternatives based on the following requirements
  - Open source
    - Share it easily with cloud providers and let them run it
      - *Crucial in tender phase to allow proper choice of VM configuration*
  - Light weight installation, reasonably fast running time
    - Submission of bmks on many VM instances for fine-grained probing approach
      - *Possibly getting code from remote repository like cvmfs*
  - Reproducible
      - *If random generation is used, fix random seed in order to have always the same sequence of events*
  - Functional relationship with experiment workloads
    - To extrapolate expectations on job duration

ATLAS Kit Validation tool

- Considered ATLAS tools
  - Being the ATLAS MC production the targeted flagship use case for the March '15 production
  - Build on past experience



17th International Conference on Computing in High Energy and Nuclear Physics (CHEP09) IOP Publishing
Journal of Physics: Conference Series **219** (2010) 042037 doi:10.1088/1742-6596/219/4/042037

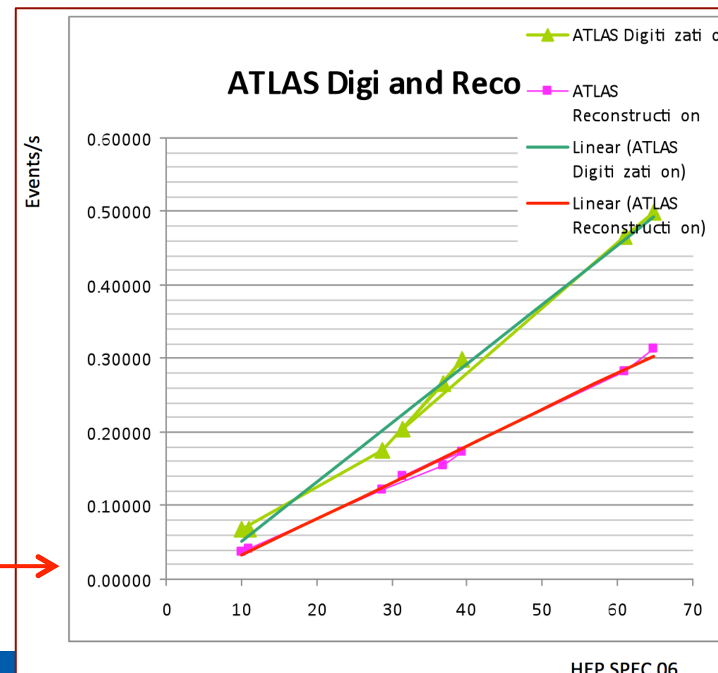
Benchmarking the ATLAS software through the Kit Validation engine

Alessandro De Salvo[(1)], **Franco Brasolin**[(2)]

[(1)] Istituto Nazionale di Fisica Nucleare, Sezione di Roma,
[(2)] Istituto Nazionale di Fisica Nucleare, Sezione di Bologna

[(1)]Alessandro.DeSalvo@roma1.infn.it, [(2)]Franco.Brasolin@bo.infn.it

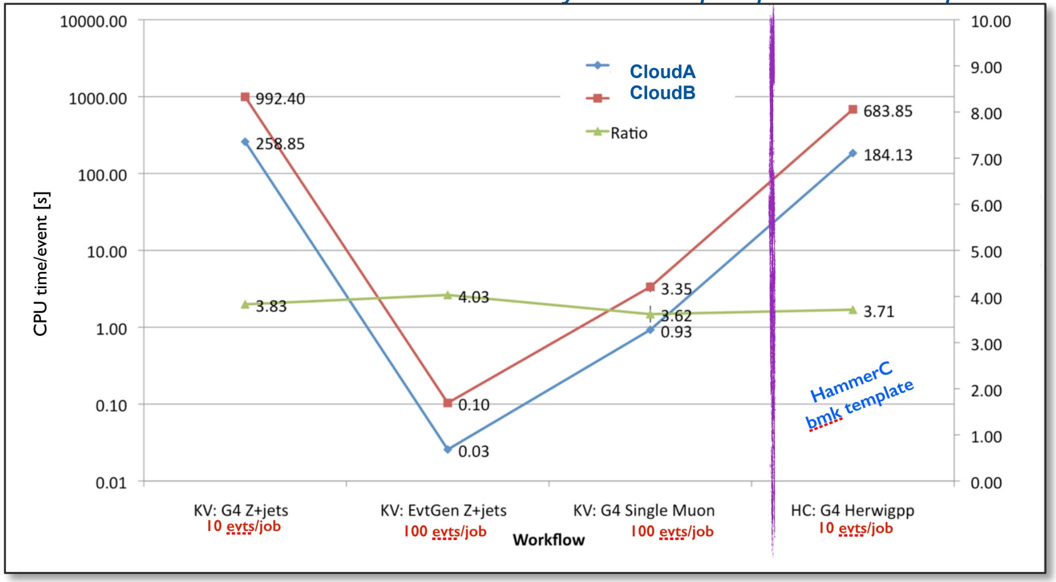
- ATLAS Kit Validation (KV)
  - Well known tool used by the ATLAS community
  - Framework essentially independent from the underlying tests
  - It's mainly bash and python based wrapper
    - ATLAS code accessed from CVMFS
  - Comparison with HEP-SPEC06 already studied in the past



KV Reference Workload

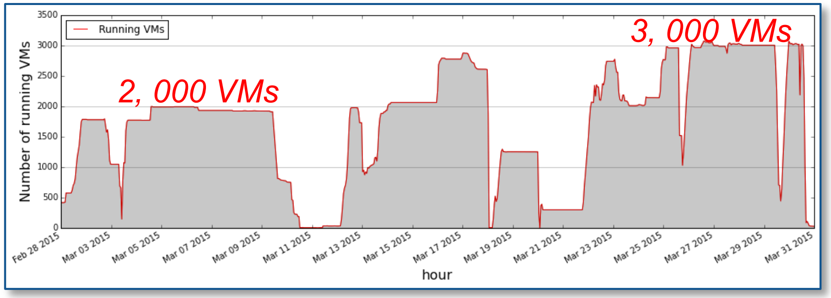
- Which workload to use for benchmarking?
  - CPU time/event is different for each workload
  - Measured that within ~10% the relative CPU/event performance doesn't depend on specific workloads
    - Confirmed also using a different approach: HammerCloud jobs

Study done in pre-procurement phase



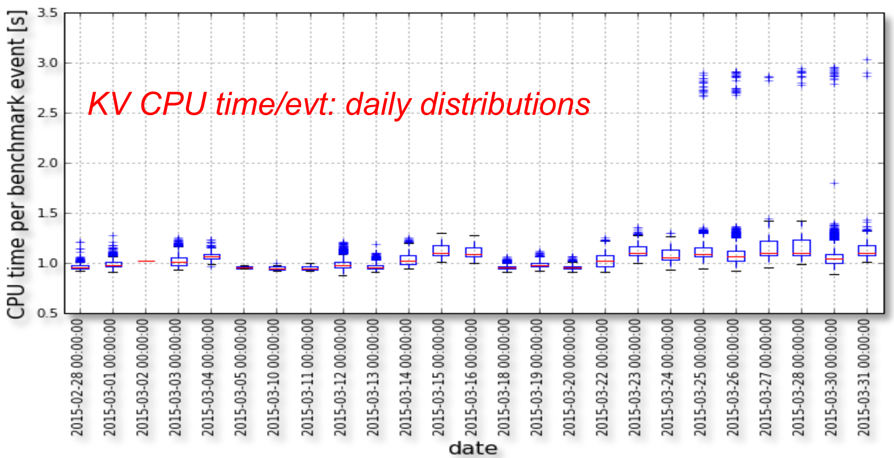
- Preferred workload: G4 single muon: faster running time O(few mins)
  - NB: the CPU time/event doesn't include the first event, to avoid bias due to the initialization process

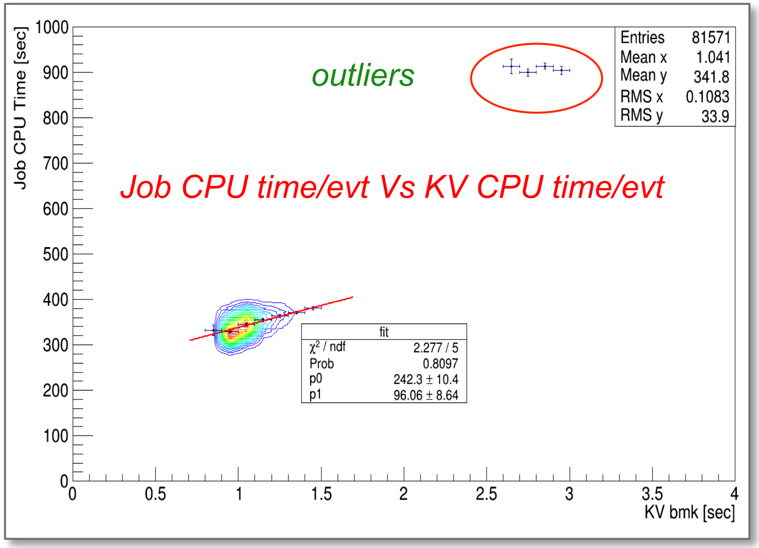
Benchmarking During March '15 Production



- Up to 3,000 concurrent running VMs
  - ~1.2 million CPU hours of processing
- Each provisioned VM has been benchmarked
  - ~30,000 VM benchmarks performed
  - KV benchmark: 100 Single Muon events simulated (~2 min to run)
- Results
  - CPU performance uniform within 15% spread
  - Benchmark profile consistent over time
  - Consistent job CPU performance and benchmark
    - Prompt identification of outliers

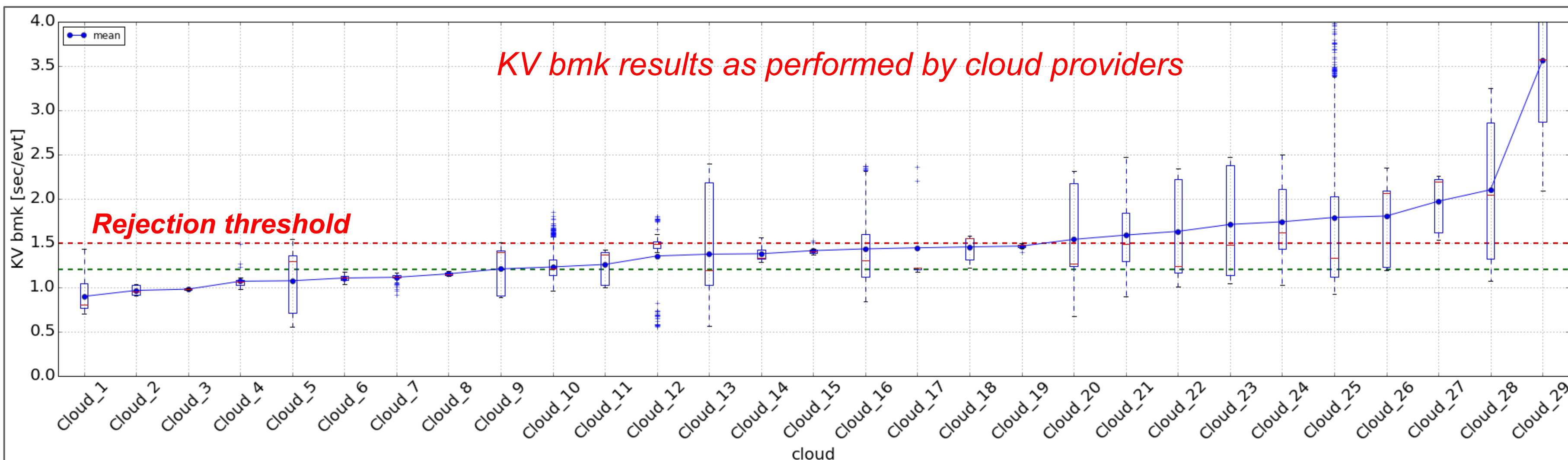
More details in GDB of June '15



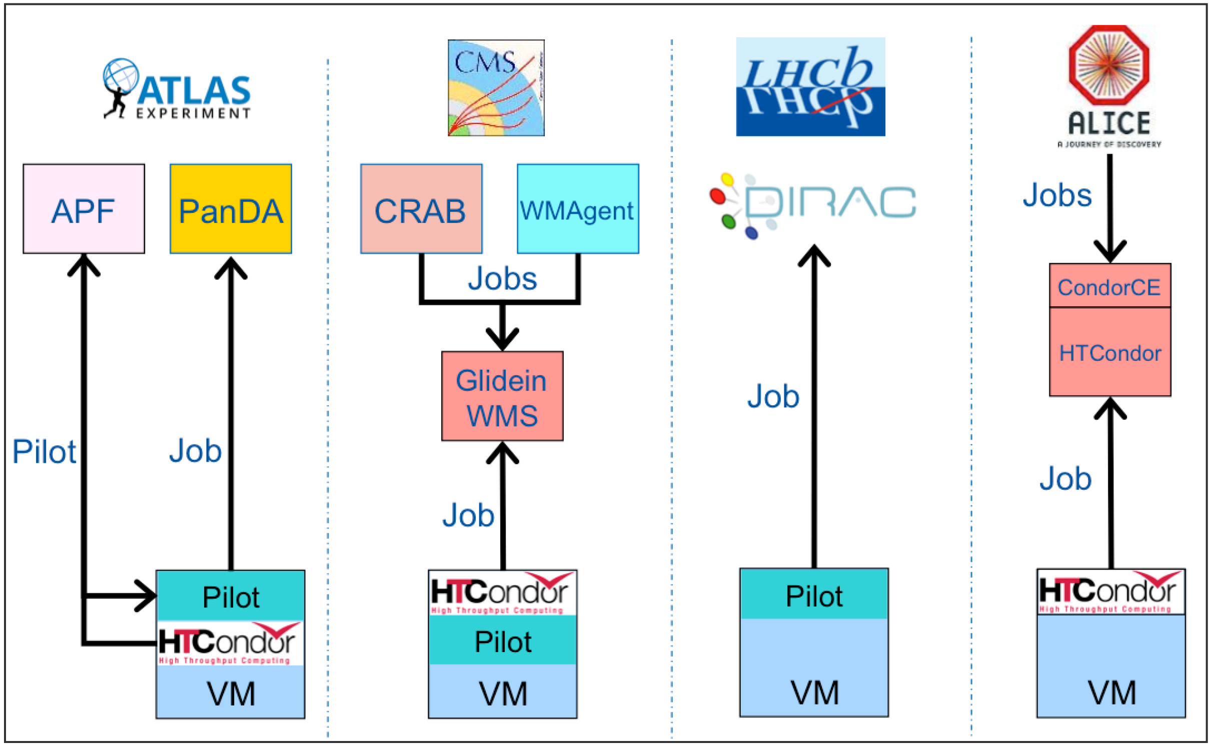


Benchmark in Current Procurement (Oct. '15)

- Building on the March '15 experience, CERN has launched a second larger procurement for commercial cloud resources
- In this case benchmarking is used to
  - Fix limits: min. desired (KV 1.2 s/evt) and tolerated performance (KV 1.5 s/evt)
    - Reminder: adjudication is on cheapest compliant bid
  - Define service credits for poor performance
    - All provisioned VMs are systematically profiled (shown in a dedicated slide)



Supporting Diverse WMS Approaches



- 4 single-core or 1 multi-core job per VM (4 vCPU)
- Different VM lifecycles
- Common benchmarking approach:
  - Profile each VM at the beginning of its lifetime and at each pilot cycle

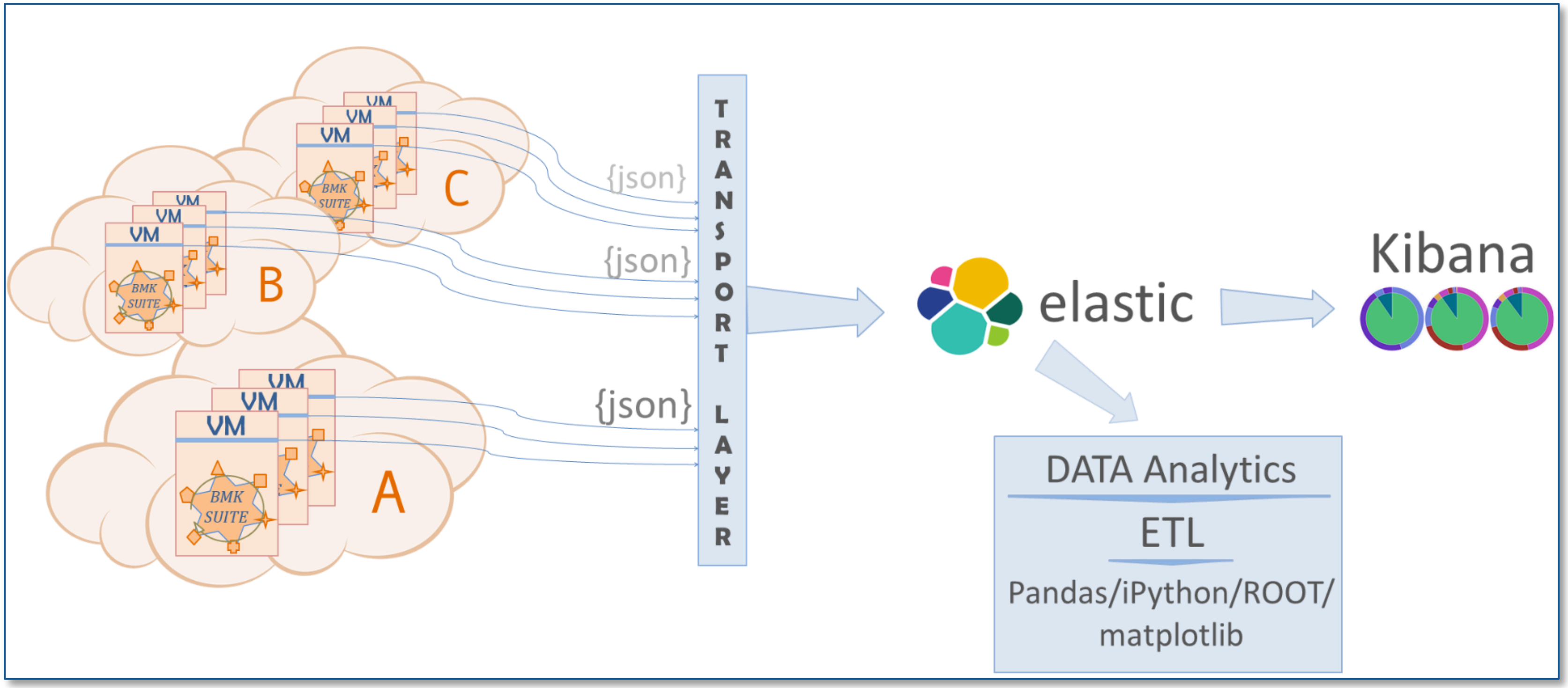
Cloud Benchmark Suite

Cloud Benchmark Suite

- Be able to run several benchmarks on the same cloud resource

- Strategy
  - Allow collection of a configurable number of benchmarks
    - *Compare the benchmark outcome under similar conditions*
  - Mimic the usage of cloud resources for experiment workloads
    - *Benchmark VMs of the same size used by VOs (1 vCPU, 4 vCPUs, etc)*
    - *Probe randomly assigned slots in a cloud cluster*
      - *Not knowing what the neighbor is doing*
  - Generalize the contextualization to run the benchmark suite in any cloud
  - Have a prompt feedback about executed benchmarks
    - *In production can suggest deletion and re-provisioning of underperforming VMs*
  - Ease data analysis and resource accounting

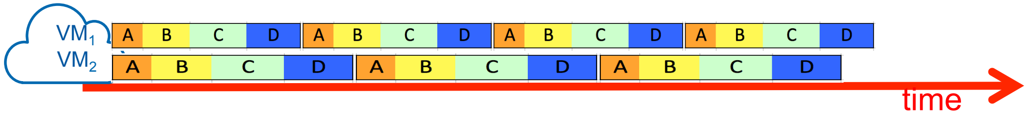
Benchmark Suite Architecture



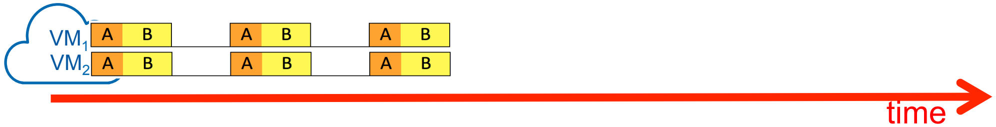
- A configurable sequence of benchmarks to run
- Results are collected in Elasticsearch cluster & monitored with Kibana
  - Metadata: VM UID, CPU architecture, OS, Cloud name, IP address, …
- Detailed analysis performed with Ipython analysis tools

Benchmarking Approach

- For each benchmark run in parallel as many threads as the number of vCPUs
- Two running modes
  - Sequential



  - Synchronized



6 Benchmarks Used (so far)

- HEP related
  - LHCb Fast Benchmark (fastBmk)
    - Original python code modified by A. Wiebalck to run python.multiprocessing
    - Very fast, gaussian random generator

```
t = random.normalvariate( 10, 1 )
```

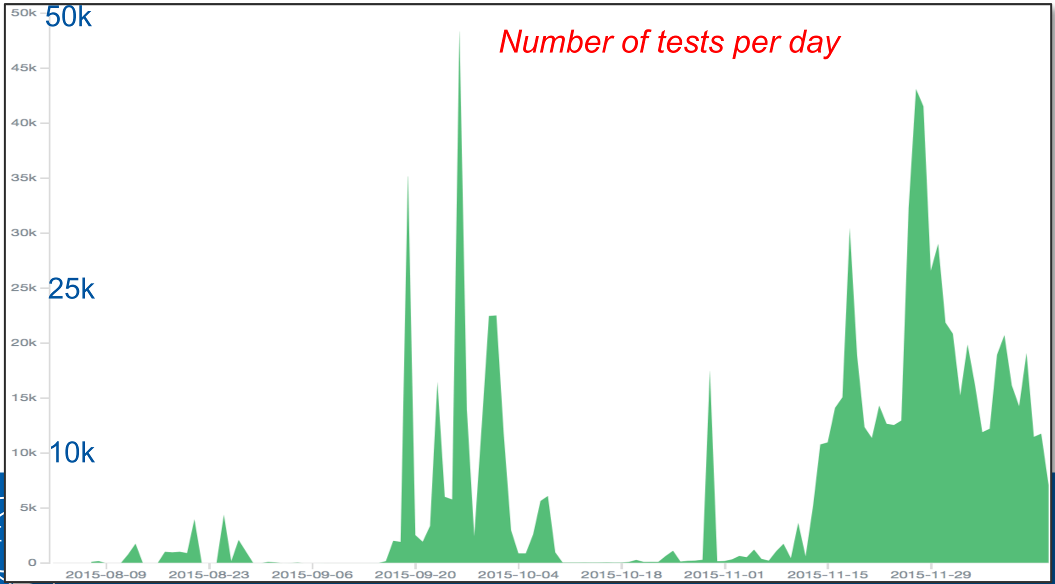
  - ATLAS KV
  - *NB: other benchmark candidates can be included*

- Open-source Phoronix benchmarks adopted by DBCE to "commoditize" resources

Category	Benchmark Tool (B_i)	Threshold (t_i)	Description
Application/CPU	7-Zip Compression	1600 MIPS	Number of MIPS of (virtual) CPU when compressing a file with 7zip
Application/CPU	LAME MP3 Audio Encoding	36 s	Time to convert a WAV into an MP3 file with LAME (CPU only)
Application/CPU	Linux Kernel Compiler3	1480 s	Time to compile a Linux kernel. (CPU only)
Application/CPU	x264 Video Encoding4	14.5 frames/s	Time to convert video from MPEG2 to MPEG4

Amount of Data Collected

- ~870k benchmark suites executed
- Various providers tested
  - Including CERN OpenStack
  - Different data centres (when possible)
    - Azure NorthEU, WestEU, CentralUS
- Different OS for the VM image
  - SLC6, CernVM, CentOS 6.*
- Different CPU models



 # tests per Cloud IaaS & OS

AZURE	Scientific Linux release 6.6 (Carbon)	242614
AZURE	CentOS release 6.7 (Final)	196
CERN-OPENSTACK*	Scientific Linux CERN SLC release 6.6 (Carbon)	454551
CERN-OPENSTACK*	Scientific Linux release 6.6 (Carbon)	1041
CERN-OPENSTACK*	Scientific Linux CERN SLC release 6.7 (Carbon)	4
CERN-OPENSTACK*	CentOS release 6.6 (Final)	2
ULTIMUM	CentOS release 6.6 (Final)	7756
ULTIMUM	Scientific Linux release 6.6 (Carbon)	6891
INNOVO	Scientific Linux release 6.6 (Carbon)	139119
INNOVO	CentOS release 6.5 (Final)	4921
CLOUDATA	Scientific Linux release 6.6 (Carbon)	6570
CLOUDATA	CentOS release 6.6 (Final)	138
DARZ	Scientific Linux release 6.6 (Carbon)	3692
DARZ	CentOS release 6.6 (Final)	2735
CLOUDHEAT	Scientific Linux release 6.6 (Carbon)	2202
CLOUDHEAT	CentOS release 6.6 (Final)	72

 # tests per Cloud IaaS & CPU model

AZURE	Intel(R) Xeon(R) CPU E5-2673 v3 @ 2.40GHz	131969
AZURE	Intel(R) Xeon(R) CPU E5-2660 0 @ 2.20GHz	72430
AZURE	AMD Opteron(tm) Processor 4171 HE	38409
AZURE	Intel Core i7 9xx (Nehalem Class Core i7)	2
CERN-OPENSTACK*	Intel(R) Xeon(R) CPU E5-2650 v2 @ 2.60GHz	455234
CERN-OPENSTACK*	Intel Core i7 9xx (Nehalem Class Core i7)	316
CERN-OPENSTACK*	Intel(R) Xeon(R) CPU E5-2630L 0 @ 2.00GHz	27
CERN-OPENSTACK*	Intel Xeon E312xx (Sandy Bridge)	16
CERN-OPENSTACK*	AMD Opteron 62xx class CPU	3
CERN-OPENSTACK*	Westmere E56xx/L56xx/X56xx (Nehalem-C)	2
ULTIMUM	Intel(R) Xeon(R) CPU E5-2650 0 @ 2.00GHz	14646
ULTIMUM	AMD Opteron 62xx class CPU	1
INNOVO	Intel Core i7 9xx (Nehalem Class Core i7)	130607
INNOVO	Intel Xeon E312xx (Sandy Bridge)	13433
CLOUDATA	Westmere E56xx/L56xx/X56xx (Nehalem-C)	6708
DARZ	Intel Xeon E312xx (Sandy Bridge)	6427
CLOUDHEAT	AMD Opteron 63xx class CPU	2274

Benchmark Results at a Glance

- Metric ~ [1/s]; each point ⇔10 min average; Colour ⇔ Cloud

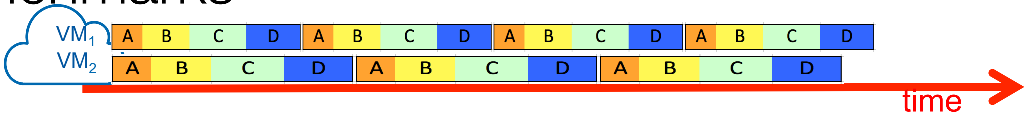
Benchmark Results at a Glance

- Metric ~ [s]; each point ⇔10 min average; Colour ⇔ Cloud

Case of Study: OpenStack at CERN

Work done in collaboration with J. Van Eldik

- Evaluate the effect of hypervisor load on the performance of single vCPU VMs
  - Extracted 5 nodes from pool of computing nodes
    - Intel(R) Xeon(R) CPU E5-2650 v2 @ 2.60GHz
  - Load phases: create a targeted number of VMs per hypervisor
    - 1 VM per HyperV
    - 16 VMs per HyperV
    - 30 VMs per HyperV
  - VM image: Scientific Linux CERN SLC release 6.6 (Carbon)
  - Run sequence of benchmarks



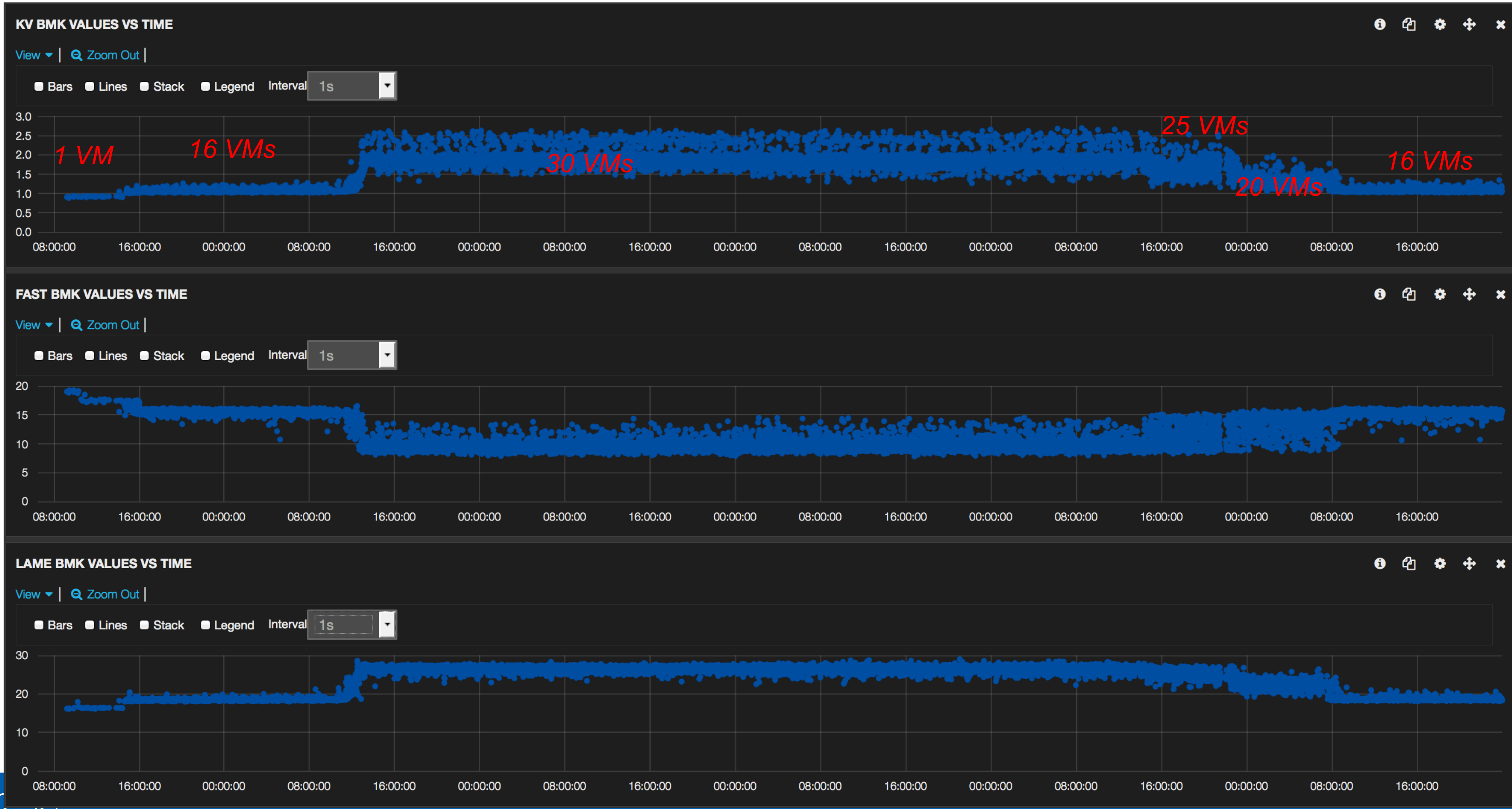
    - Used Phoronix open source benchmarks to produce load

Qualitative Look at Data

- Identifiable transition of CPU performance when load changes
  - Seen in all benchmark measurements. Performance recovers scaling down

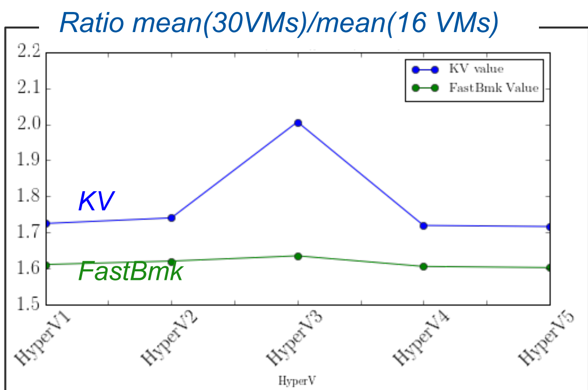
Qualitative Look at Data

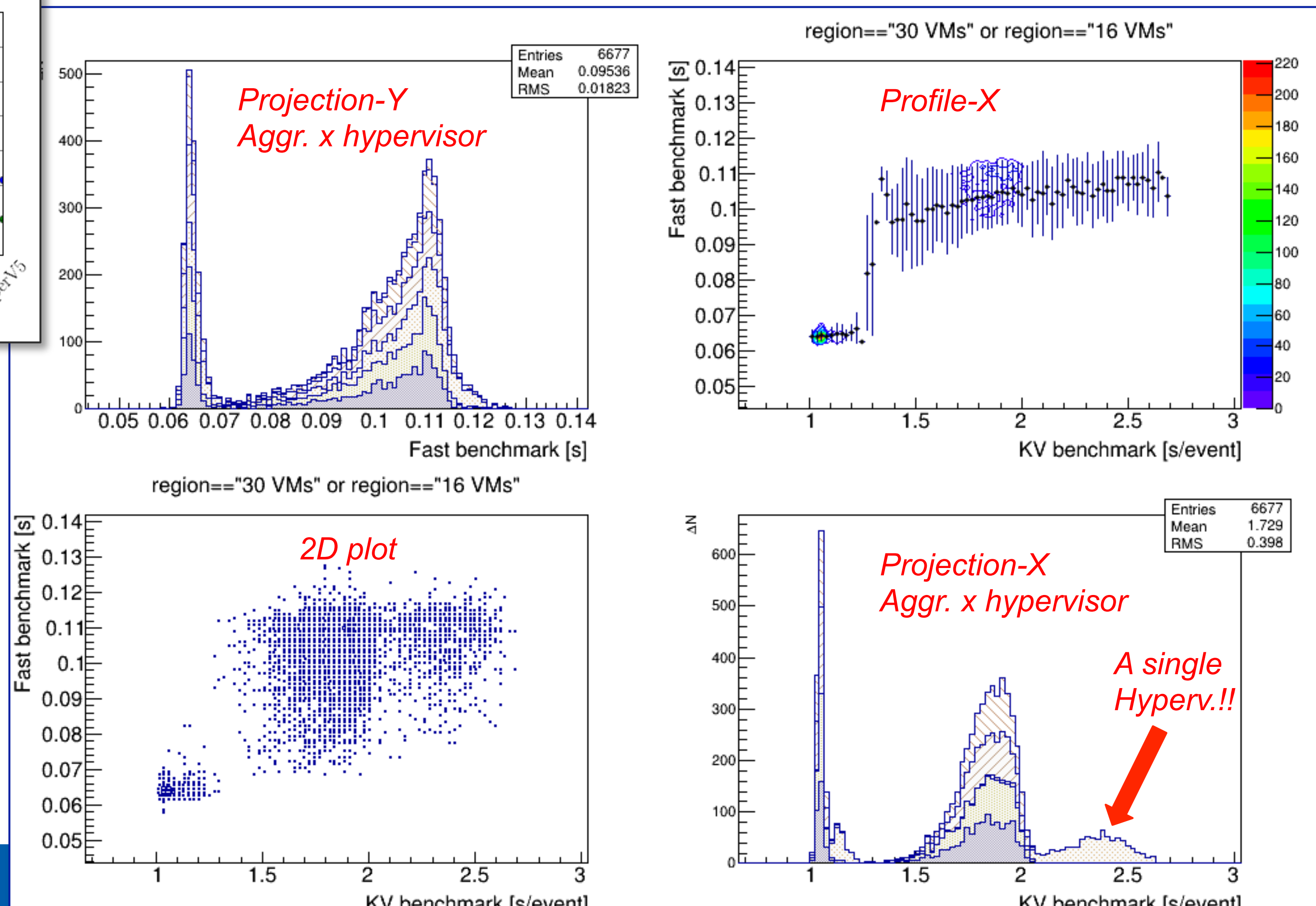
- Larger dispersion in KV and FastBmk values in the highest-load region

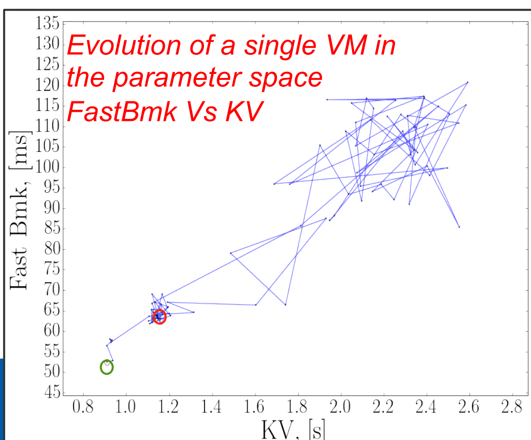


More Quantitative Analysis: FastBmk Vs KV

- Correlation study in the region 16 and 30 VMs
  - *NB: FastBmk metric transformed into value^{-1} [s]*
  - The average performance degradation differs per Hypervisor and Bmk used

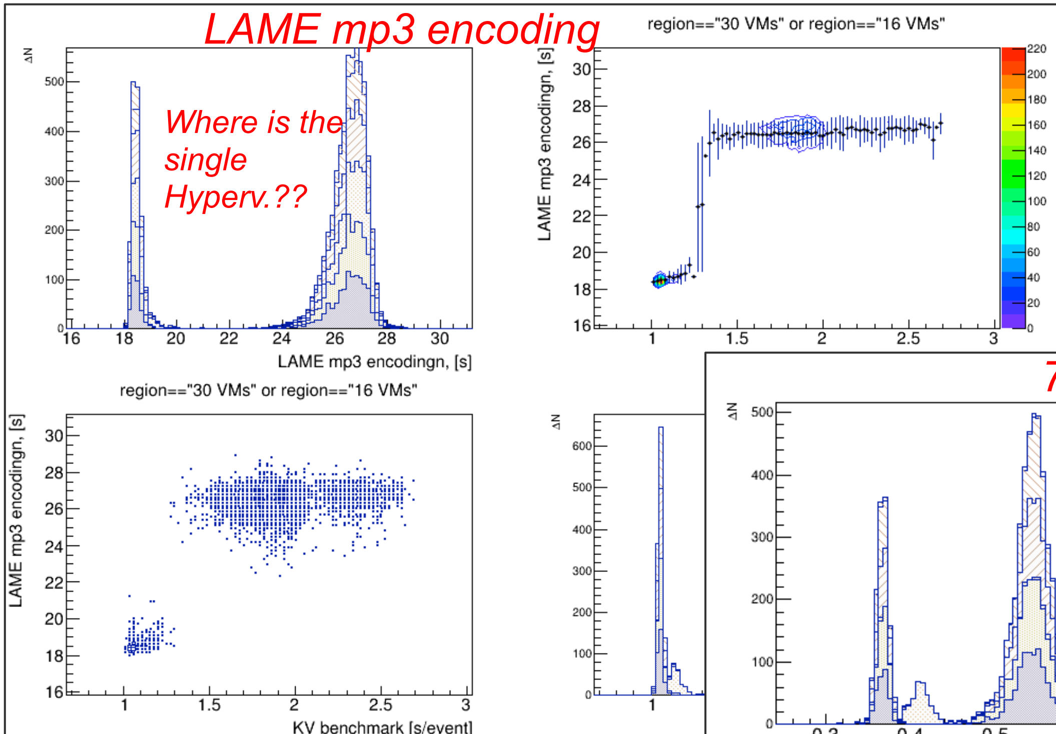


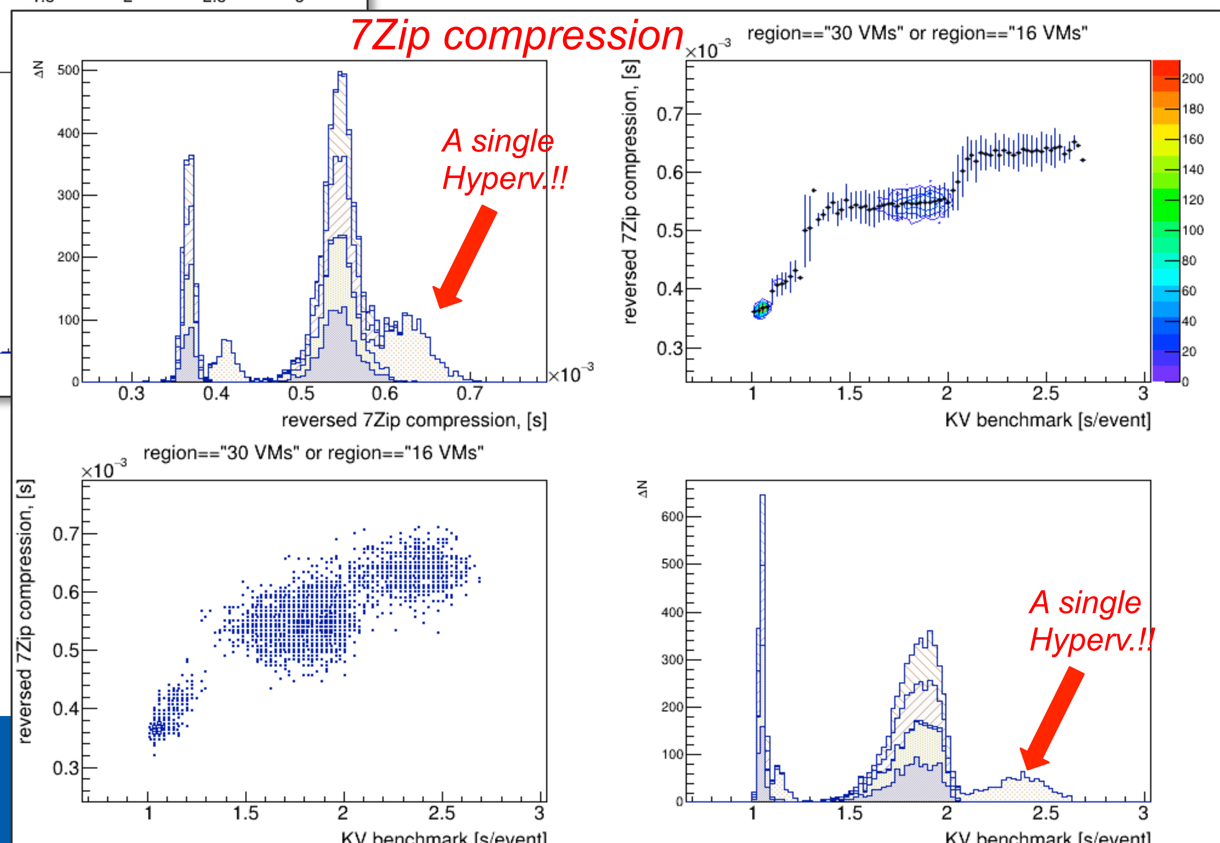




And the Other Benchmarks?

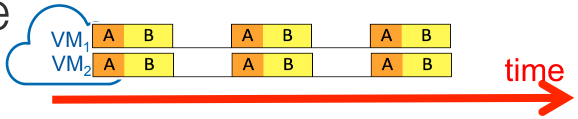
- Ability to discriminate different hypervisor performance depends on the specific test

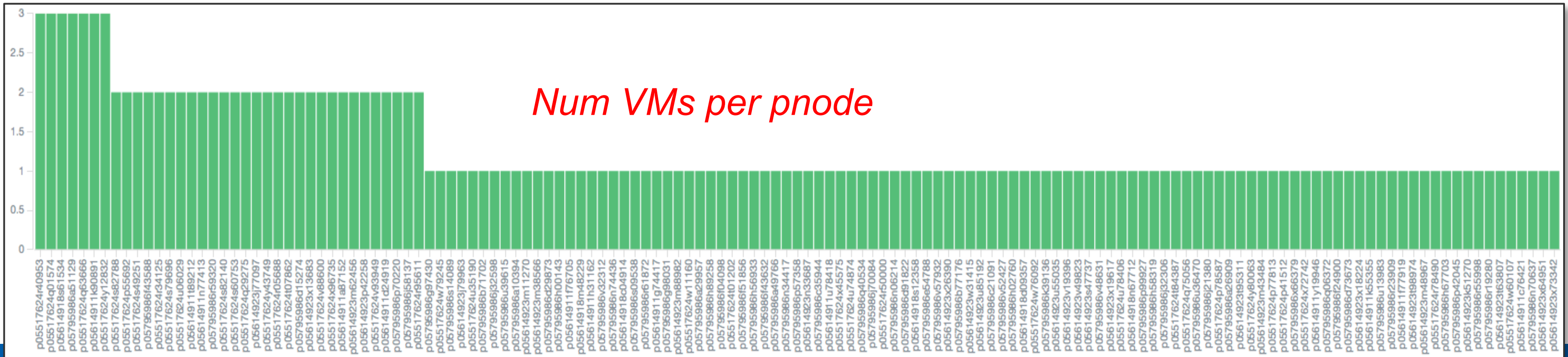




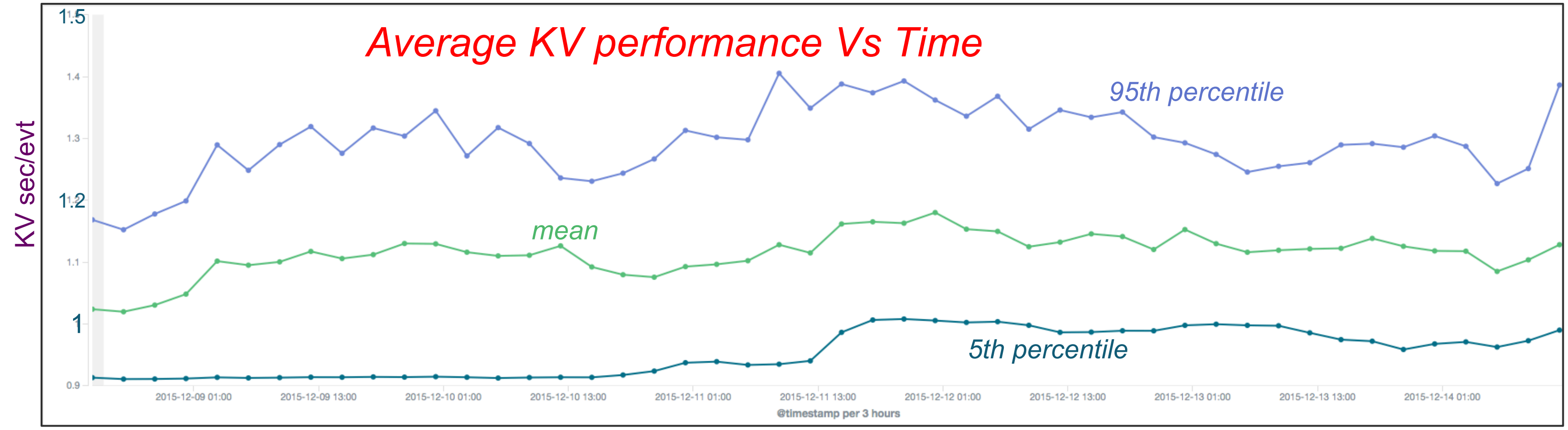
Probing the OpenStack Compute Environment

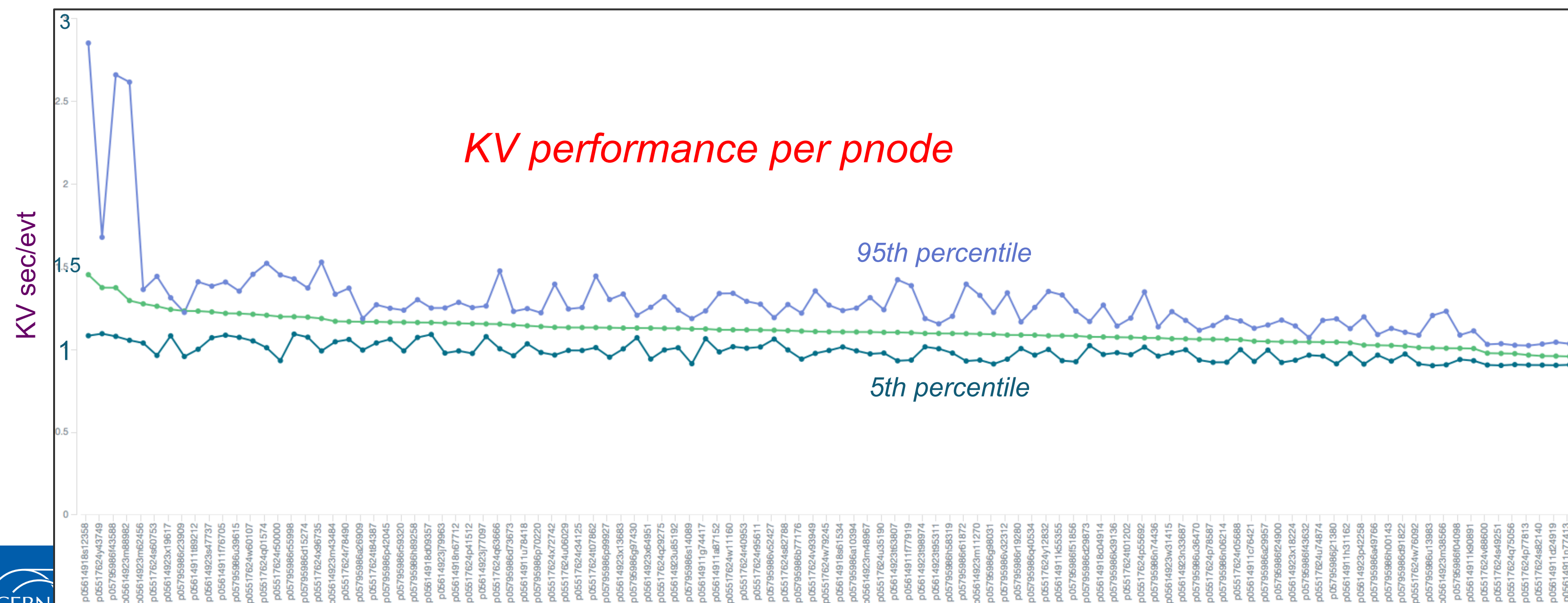
- Idea: probe performance of VMs in OpenStack Compute Environment
  - Where resources are assigned to the experiments for CERN cloud activities
  - Tenant with ~200 single-core VMs
    - Make sure VMs are provisioned in different Hypervisors
    - Run synchronized benchmarking suite





Preliminary Profiling Results

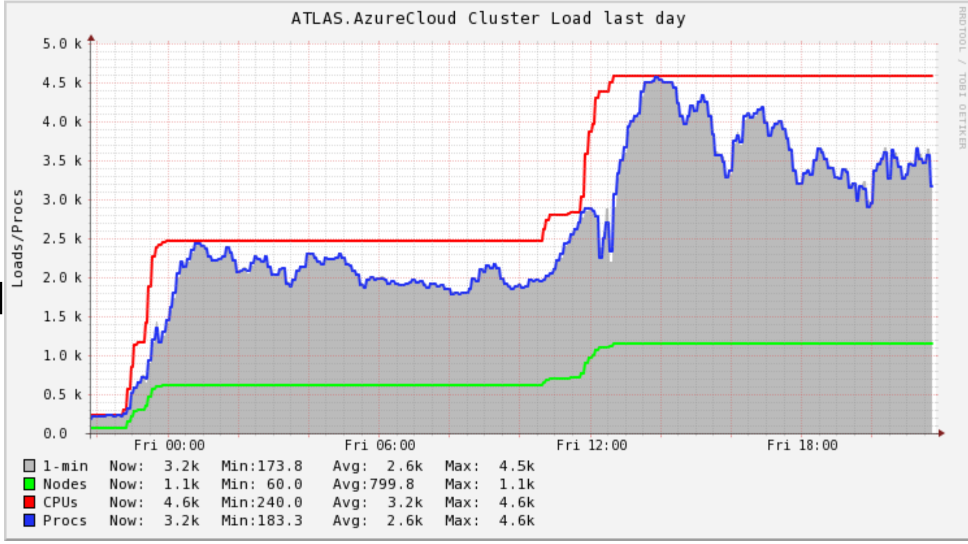




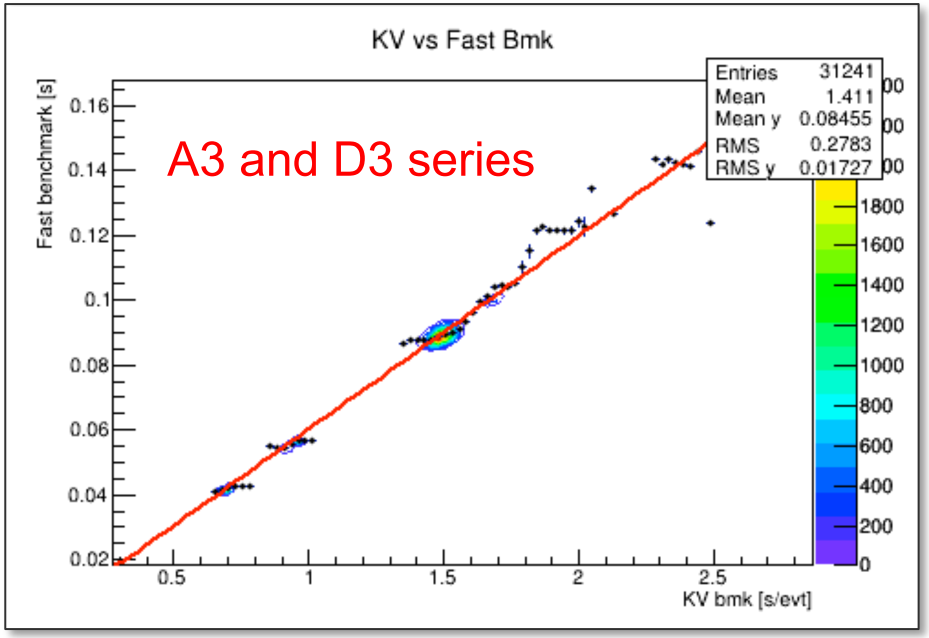
Microsoft Azure Evaluation

Objective

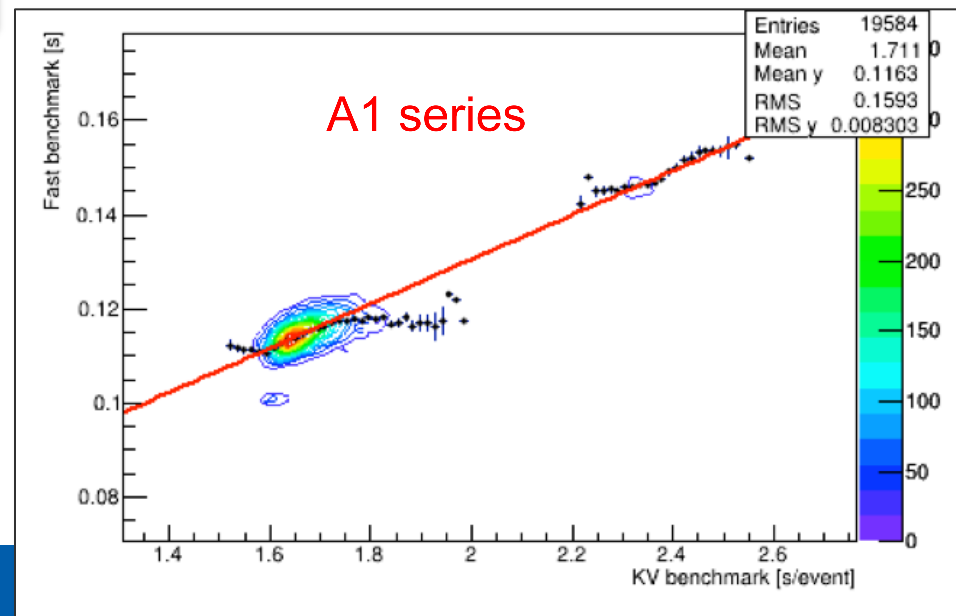
- Joint collaboration between CERN Openlab and Microsoft Azure
  - Evaluation of Azure platform and integration with existing WLCG tools
- Achievements
  - Evaluated two different provisioning models
    - Azure Service Manager (old), Azure Resource Manager (new)
  - Adopted CernVM image
  - Azure resources integrated in WMS of ATLAS, CMS, LHCb
  - Scale tests
    - Performed in 3 DCs (2 EU, 1 US)
    - Reached ~4800 vCPUs provisioned
  - Performance evaluation
    - Benchmarks and cost



KV Vs fastBmk Performance



- Good Linearity among two independent benchmarks: KV and fastBmk
- The measured effect does not depend on specific compiler flags



To Summarise

- In cloud environment the VM performance is highly variable
  - Changes with load on the IaaS
  - It is differently measured by different benchmark tests

- Synthetic, fast benchmarks running in each VM, iteratively along the VM lifetime, allow to spot performance changes
  - Can consolidate accounting of resources, supplying normalization factors
  - Choice of a valid benchmark to translate in job expectation is not trivial
    - Identify reliable benchmark for multi-core applications

What Next?

- Discussion about synthetic benchmarks is ongoing in WLCG and within experiments
  - Several good candidates available
  - Interesting GDB talks on HEP-SPEC06 scalability
- Correlation studies with job performance are ongoing
  - Still preliminary to be discussed here
- In addition to the measurement of CPU performance
  - Profile network and storage access
  - Network is a crucial component of a distributed system
    - Measure LAN and WAN performance
    - Monitoring experience already established in WLCG
      - Could be exported to cloud environments

Conclusions

- Benchmarking is an important aspect of the production process
  - Component of the procurement procedures
  - Monitor delivery on specifications
  - Support activities on performance improvement

- Cloud Benchmark Suite available
  - Configurable. Can include additional benchmarks
  - Tested in several IaaS. Data available through ES-Kibana

CERN

www.cern.ch