VM CPU Benchmarking the HEPiX Way

Manfred Alef, Ian Gable

FZK Karlsruhe
University of Victoria

May 28, 2009

Overview

- Motivation
- Benchmark Testbed
- Benchmarking Technique
- Results: Xen, KVM, Amazon EC2
- Conclusion
- Other Virtualization Activities at the University of Victoria

Motivation

- Measure Virtual Machines as they would be used on HEP Worker nodes
- There are plenty of VM benchmarks out there, but none of the them had been setup specifically for HEP
  - Multiple 100% CPU Loaded VMs running on a single box
  - Scientific Linux used
- Now possible to create your own Amazon like cloud infrastructure.
  - Open Nebula (seen at HEPiX), Nimbus (Globus project, UVic involved)
  - Does turning you cluster into a Cloud IaaS service cost you in lost CPU performance?
- Focus on CPU benchmarking because it's less dependent on specific setups.
- From working with the HEPiX CPU working group we were well positioned to get results quickly.

University of Victoria

GridKa

KIT Karlsruhe Institute of Technology

Benchmark Testbed

Selected to form a representative set of CPUs commonly used at HEP Sites:

CPU Model	Mem (GB)	n Cores	Mainboard	Year
AMD Opteron				
246 (2.0 GHz SC)	4	2	MSI-9145	2003
270 (2.0 GHz DC)	8	4	MSI-9145	2005
2376 (2.3 GHz, QC, Shanghai)	16	8	Supermicro H8DMU+	2009
Intel Xeon				
Intel Xeon 3.00 GHz, SC (Nocona)	2	2	HP ProLiant DL360 G4, HT off	2004
5160 (3.0 GHz, DC, Woodcrest)	6	4	Intel Server Board S5000VCL	2006
E5345 (2.33 GHz, QC, Clovertown)	16	8	Supermicro CSE-812L-520CB	2007
E5405 (2.0 GHz, QC, Harpertown)	16	8	Dell PowerEdge 2950	2007
L5420 (2.5 GHz, QC, Harpertown)	16	8	Supermicro X7DCT	2008

University of Victoria

GridKa Karlsruhe Institute of Technology

Selection of OS and Virtual Machine Monitors

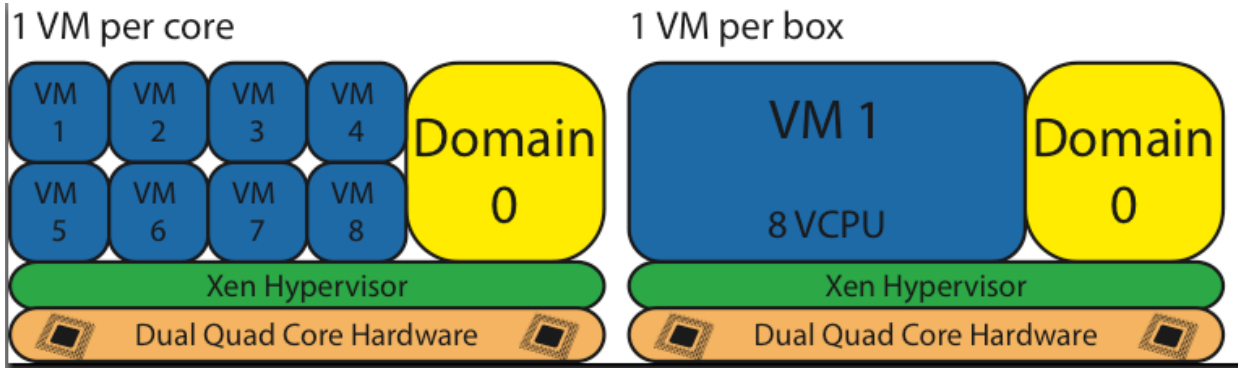
- Use HEP-SPEC06 as the benchmark. Obvious choice.
- Selected Scientific Linux 5.2 with the distribution provided Xen
  - Obvious choice for ease of use within the HEP community because of distribution integration
  - Had been running Xen in production mode at UVic for 3 years: lots of experience.
  - Test could be easily put together with ‘yum install kernel-xen’
- Limited KVM testing
  - Selected one box at UVic for further examination with KVM
  - Not a lot of experience with KVM
  - More work to get operational on SL (not a lot more)
  - Important to test because it appears to be RedHats new favorite

Benchmark Technique

- We wish to simulate the fully loaded HEP worker node used with virtualization
- Two scenarios we are most interested in:

1 VM per core: n VMs are booted where n is the number of cores. Each VM has 1 VCPU

1 VM per box: 1 VM is booted with VCPUs equal to the number of physical cores on the Box



University of Victoria

GridKa

KIT Karlsruhe Institute of Technology

VM Memory Allocation

VM Type	Hypervisor Version	Kernel Version	Disk Access
i386 x86_64	Xen 3.0.3	2.6.18-92.1.13.el5xen i686 2.6.18-92.1.13.el5xen x86_64	tap:aio

Hardware Type	VM Memory Allocation
All AMD and Quad Core Intels	$n \times 1900$ MB
Intel Nocona	$n \times 870$ MB
Intel Woodcrest	$n \times 870$ MB

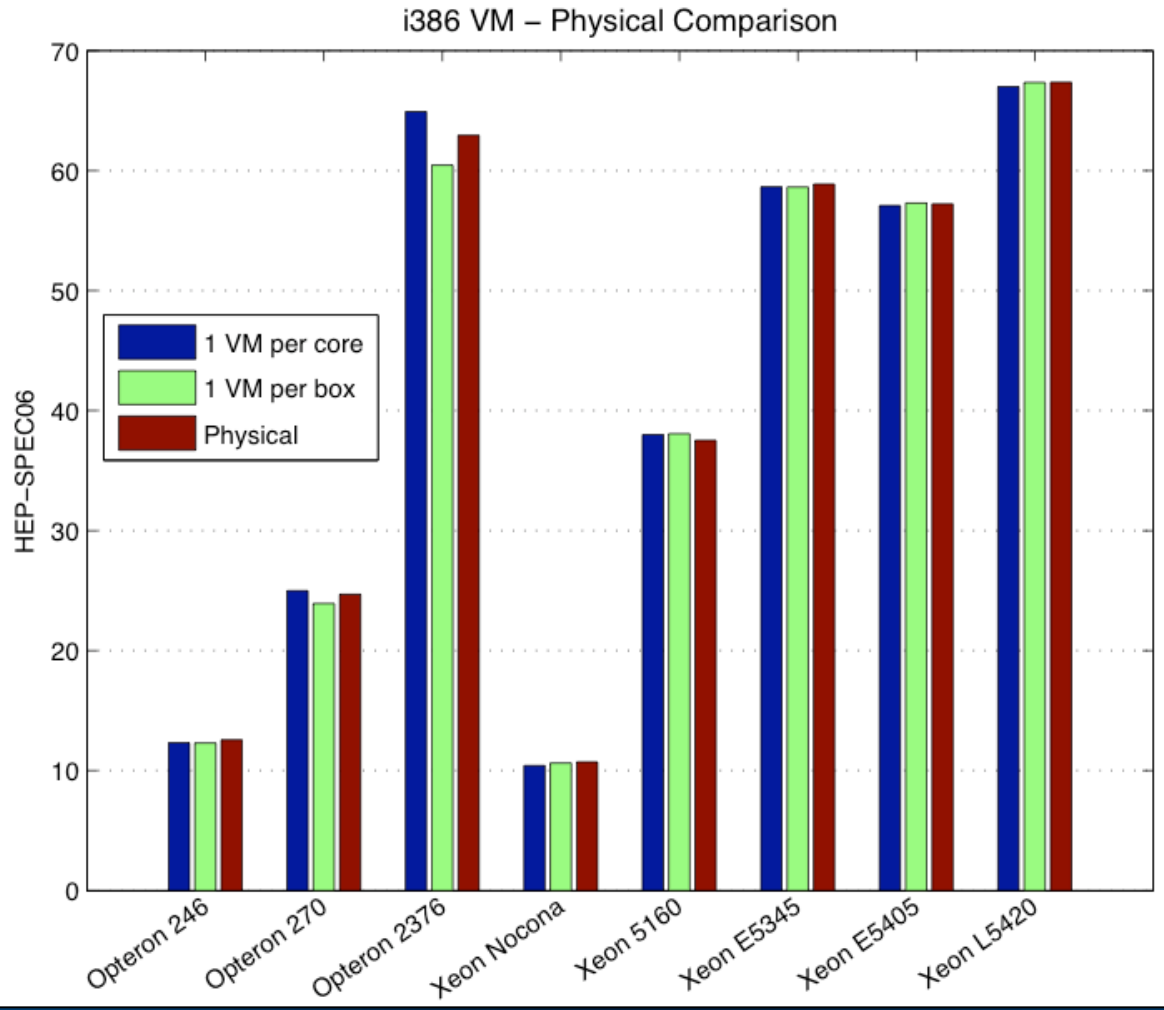
Where n is the number of cores

University of Victoria

GridKa

KIT Karlsruhe Institute of Technology

Xen Results i386 VMs

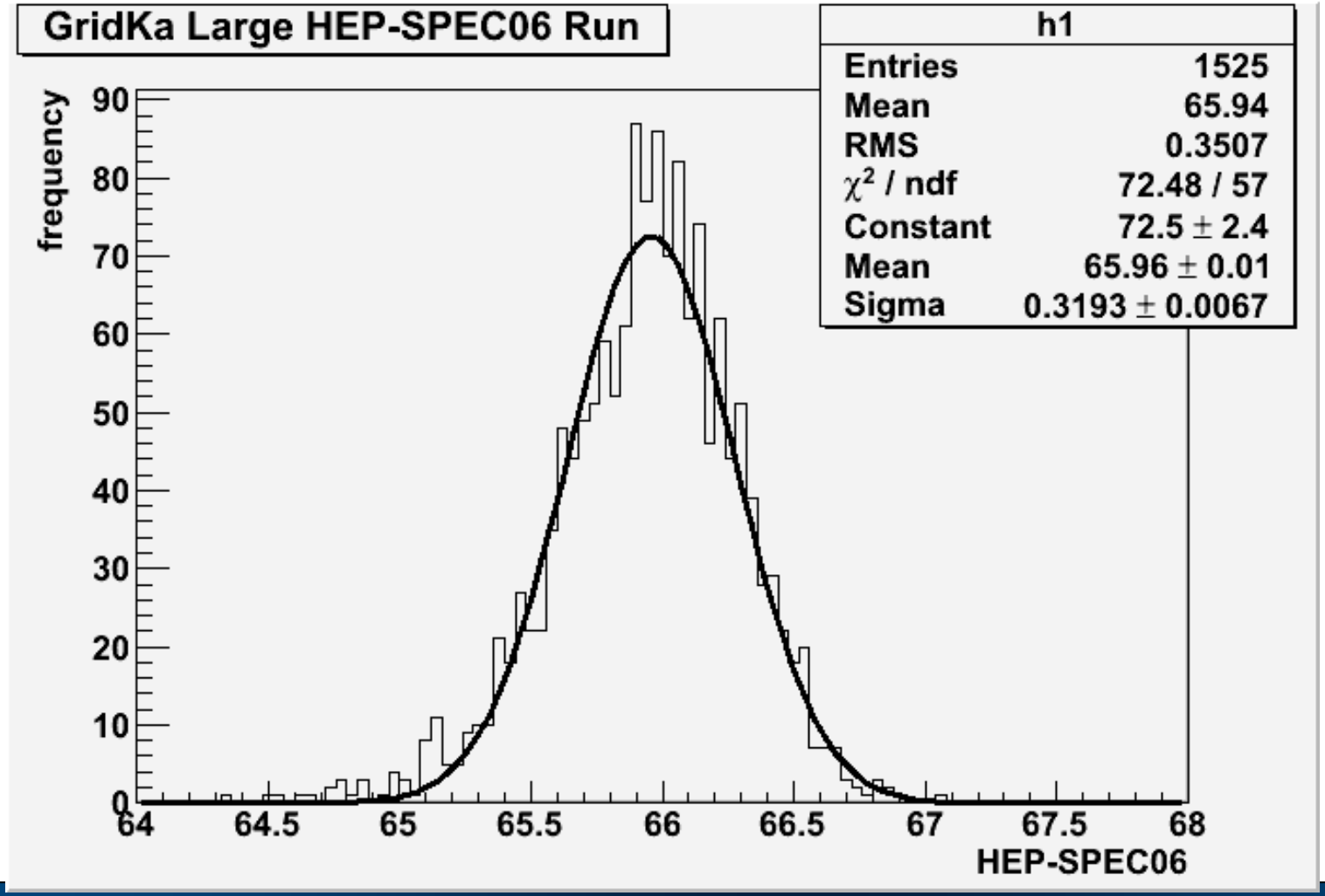


University of Victoria

GridKa

KIT Karlsruhe Institute of Technology

Distribution of HEP-SPEC06 Results

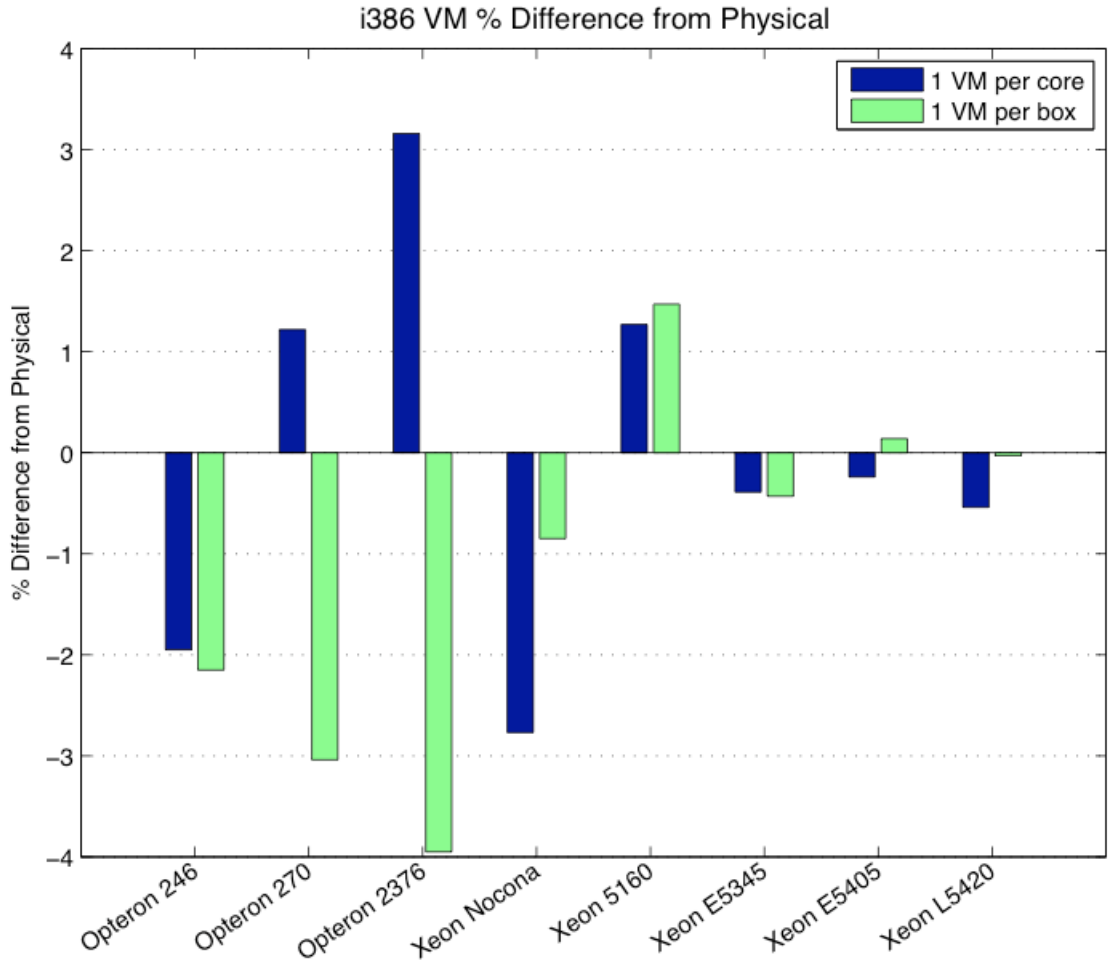


This machine type

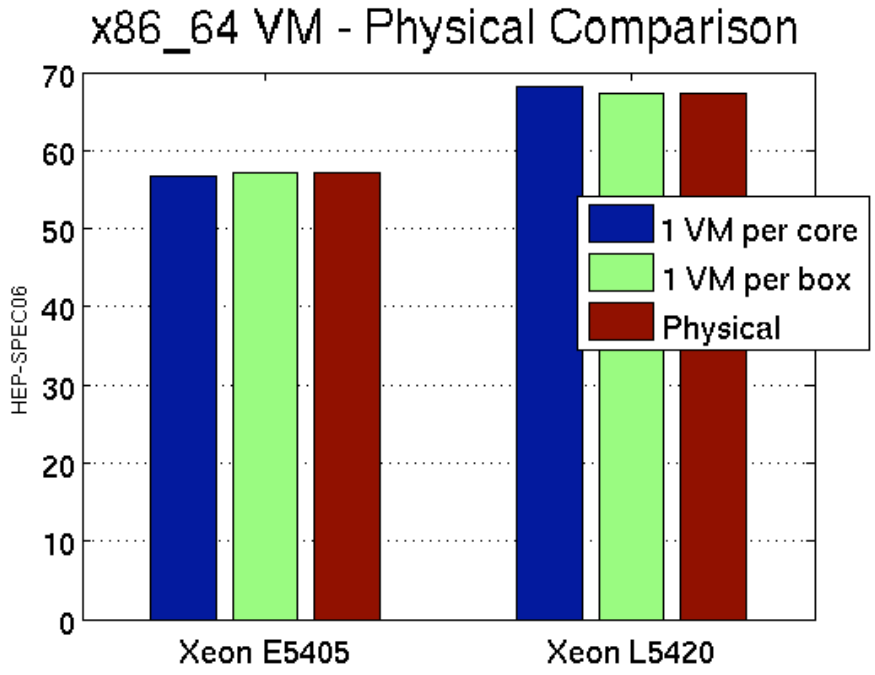
E5420:

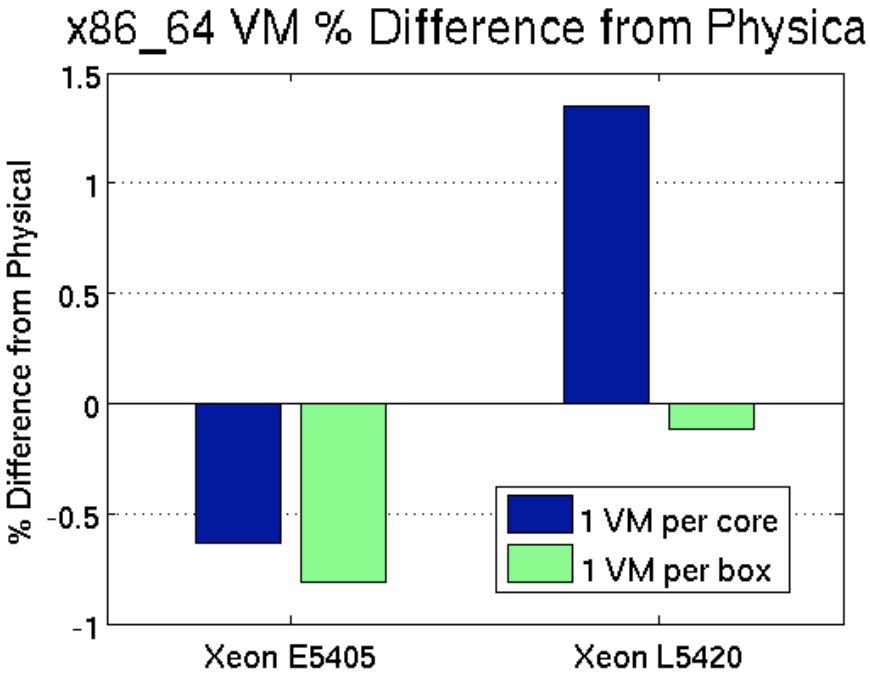
+/- 0.52 %

Xen i386 VM Relative performance



Xen Results x86_64 VMs



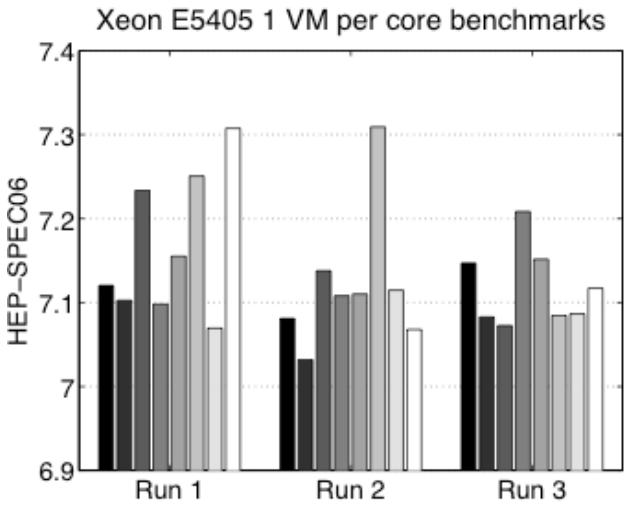


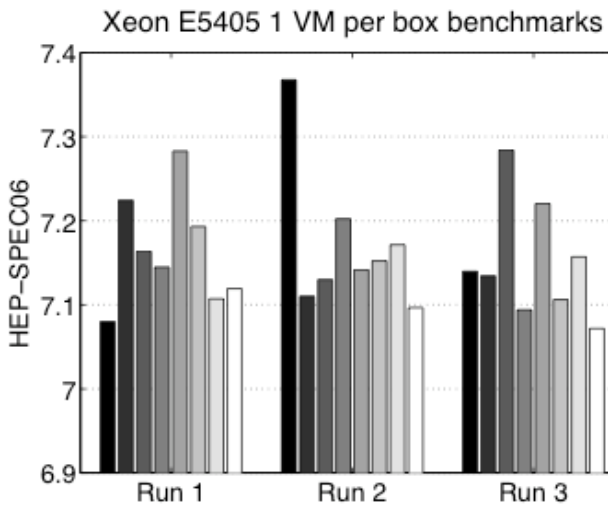
University of Victoria

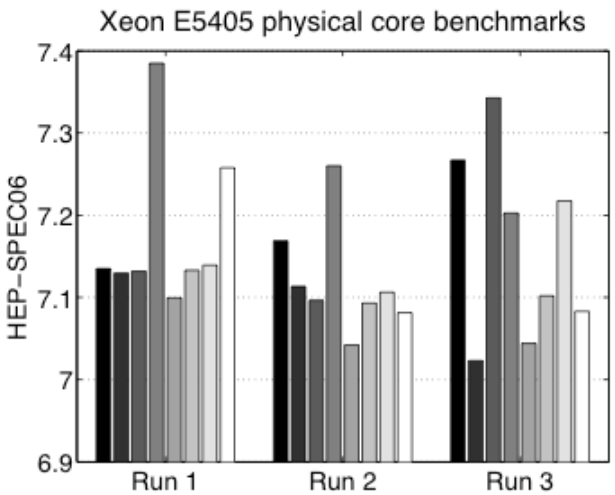
GridKa

KIT Karlsruhe Institute of Technology

Xen per core variability







Xen Results Wrap Up

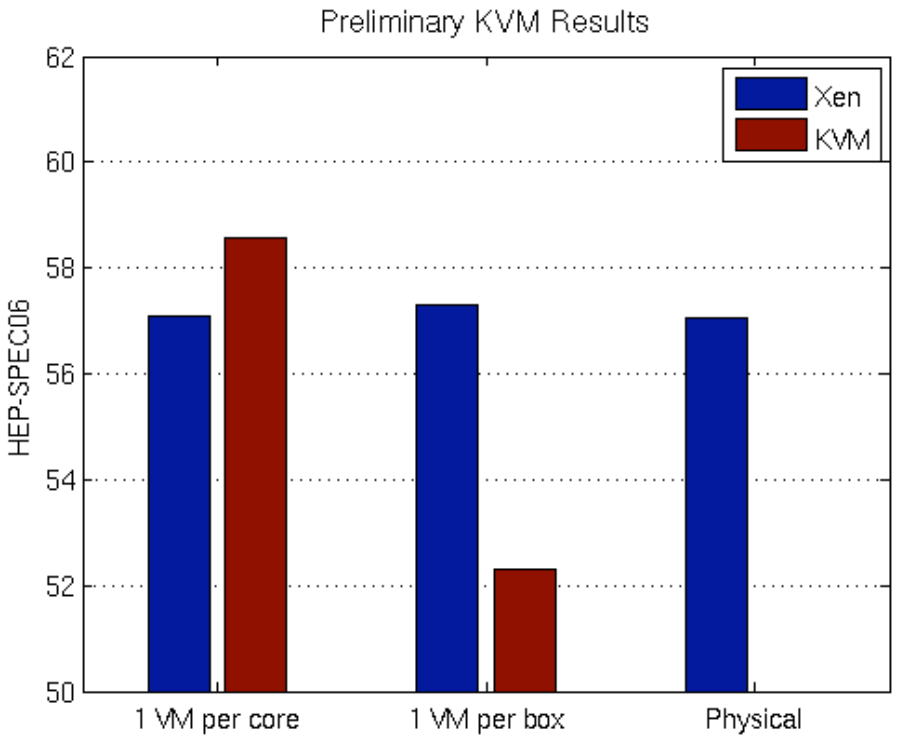
- Insignificant performance degradation in both 1 VM per core, and 1 VM per box cases.
- Some counter intuitive performance gains in the 1 VM per core case.
  - Could be associated with pinning of each VM to a core, hence fewer cache misses
  - Could be associated with the Xen Borrowed Virtual Time (BVT) scheduler vs the regular SMP kernel.
- Zero CPU Performance impediment to running 1 VM per job on a multiprocessor worker node.

A first look at KVM

- Selected Xeon E5405 (Harpertown) for further study
- Replaced host kernel with vanilla Kernel 2.6.29.2 x86_64
  - includes kvm modules as a part of mainline
  - SL 5.3 uses 2.6.18 (no kvm modules)
- Tested i386 KVM-84 SL 5.3 VM with virtio
- Exactly the same benchmark Technique.
- Results not strictly comparable with earlier Xen results.

Preliminary KVM Results

- KVM appears to out perform earlier Xen setup in the 1 VM per core case.
- KVM underperforms in the 1 VM per box case.
- Looking for an explanation



Amazon EC2 Benchmark

- Simply run HEP-SPEC06 on three different Amazon instance types

Small Instance ($0.10/ hour)
  - 1.7 GB Memory, 1 VCPU, 32-bit

Large Instance($0.40/hour)
  - 7.5 GB Memory, 2 VCPUs, 64-bit

Extra large Instance ($0.80/hour)
  - 15 GB Memory, 4 VCPUs, 64-bit

- All test done with SL5.3 Amazon Machine Images (AMI) produced at UVic using the 2.6.18 RHEL Xen kernels from Amazon.

Utility Computing Thruput measurement

$$\text{HEP-SPEC06} \cdot \frac{h}{\$}$$

Type	Average (HEP-SPEC06)	Cost ($/hour)	HEP-SPEC06*(h/$)
small instance	3.66	0.10	36.63
large instance	13.11	0.40	32.77
extra large instance	24.68	0.80	30.85

Based on 3 runs on each instance type.

To get same thruput as Xeon L5420: $1.84/h

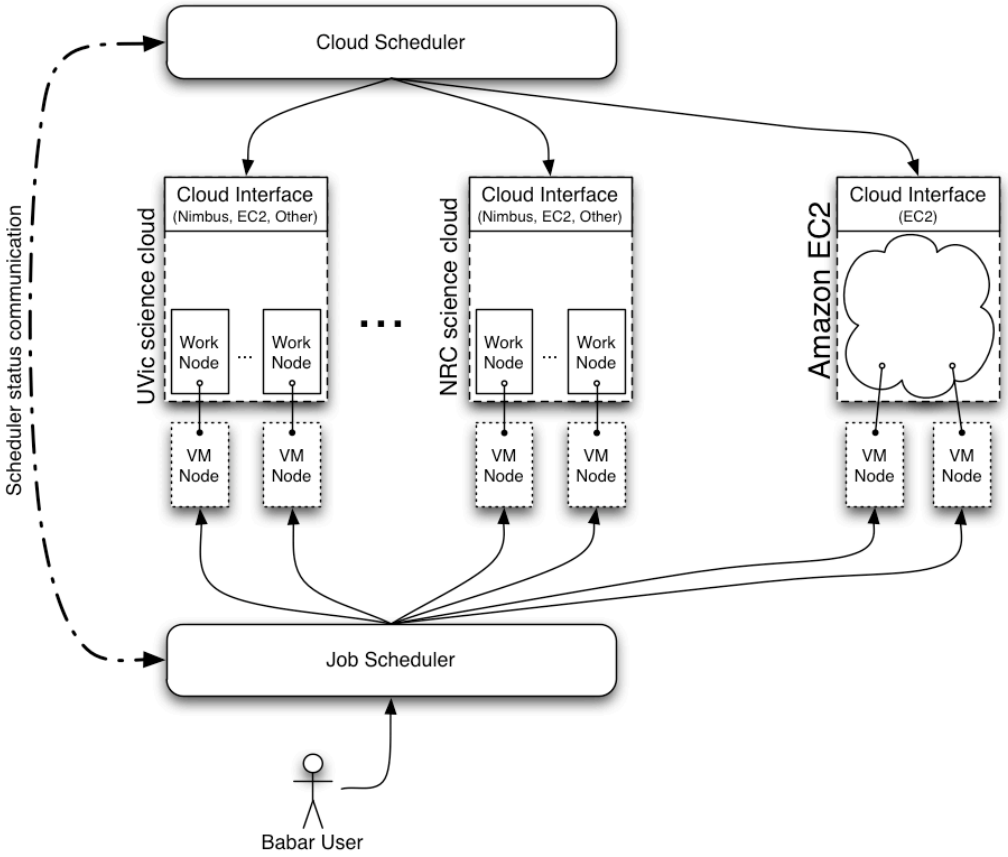
Conclusion

- In terms of CPU performance, it is indeed very feasible to run highly CPU loaded virtual machines on current generation multi-core CPUs
- Established this using the HEP-SPEC06 benchmark which has been proven to map to real HEP application performance.
- No benchmark suffered more then 5% decrease in performance, save for preliminary KVM results.
- Useful measure of utility computing cost effectiveness:

$$\text{HEP-SPEC06} \cdot \frac{h}{\$}$$

Other Virtualization Activities at UVic

- Developing BaBar Legacy Data Analysis system at UVic using a Nimbus Cloud deployed to a full blade chassis.
  - Nimbus is an Open Source IaaS software like Open Nebula (seen at HEPiX already)
- Google Summer of Code Student
  - develop code for for Nimbus project in Cloud Scheduling
  - Student gets $4500 stipend paid by Google
  - Ian to act as Google 'Project Mentor'.



University of Victoria

GridKa

KIT Karlsruhe Institute of Technology

Acknowledgements

- HEPiX CPU Benchmarking working group.
- Matt Vliet (UVic Undergraduate)