Evaluation of Virtualization based solutions

Predrag Buncic, CERN/PH-SFT

DISCLAIMER
The views expressed in this presentation are the views of the speaker and do not necessarily reflect the views of his employer ☺. The projections about the future of HEP computing are in particular very subjective and should be taken with a grain of salt. The intention is to look at the many CPU core future from a different perspective in order to stimulate discussion.

Why Virtualization in many core context?

- This Workshop is ultimately about improving performance…
- Performance and Virtualization do not usually appear in the same sentence
  - Unless we speak about performance penalty due to virtualization
- On the other hand, Cloud and Virtualization are often used interchangeably
- As it looks like we will have to live with Clouds much longer than we had to live with Grids we should ask ourselves:
  - What are the consequences for running our applications in virtualized environment?
  - How is virtualization technology evolving to reduce impact on performance?

How do you imagine your personal computer in 3-5 years?

Many Core,
GPU powered,
Liquid cooled,
Workstation?

Multi/Many Core,
(trans)portable
Laptop?

Multi Core,
Lightweight,
Netbook?

Ultra light,
Ultra portable
tablet?

Comfort and portability or performance?

- If you just look around you, the answer is obvious
  - People prefer more and more to use ultra portable devices
  - Good graphics capabilities and perhaps even considerable CPU power
  - Little storage left (after you upload all your digital media…)
- Good for typical daily work
  - e-mail, web
  - Software development, debugging…
- Not so good for any serious local data processing
  - SSD for the local storage will probably become the norm
    - Faster booting, energy saving
  - But larger disks will always be relatively expensive
- The OS will more an more depend on various Cloud services
  - Apps, storage, sync, backup…

How are our Computer Centers going to look like?

- Green is the new black
  - Energy saving is must
  - Server consolidation using virtualization is the most natural way to go
- Cloud is the new Grid
  - It is straight forward to incarnate the Grid environment on top of Cloud enabled infrastructure
  - Many components of the Grid middleware can be simplified or dropped without affecting the end user interfaces

- The clouds are here to stay
  - Amazon EC2, Microsoft Azure, iCloud (Apple), SmartCloud (IBM), Rackspace, RightScale, SalesForce.com, Google App Engine,
  - IaaA, SaaS, PaaS…
- It is likely that many of our computing centers will deploy Cloud like infrastructure
  - And that means virtualized worker nodes…

How our future computing environment might look like?

- As the files will be hosted in the Cloud and all large scale processing will happen in the same environment
- Typical server
  - Multi core: 2-4 x 6(12)-8(16) CPU cores(threads)
  - Lots (64-128 GB) of fast memory
  - PCIe 3.0
  - Optional MIC accelerator, 50+ cores
  - Basic building block in HPC environment
  - An ultimate platform for server consolidation in Cloud environment

- The high end physical nodes will most likely to be partitioned onto VMs with each VM given one or more CPU cores and perhaps a direct access to GPU (via OpenCL, CUDA) or MIC (MPI or OpenMP) in case if the application running in VM can effectively use it

The checklist for improving performance in many core environment

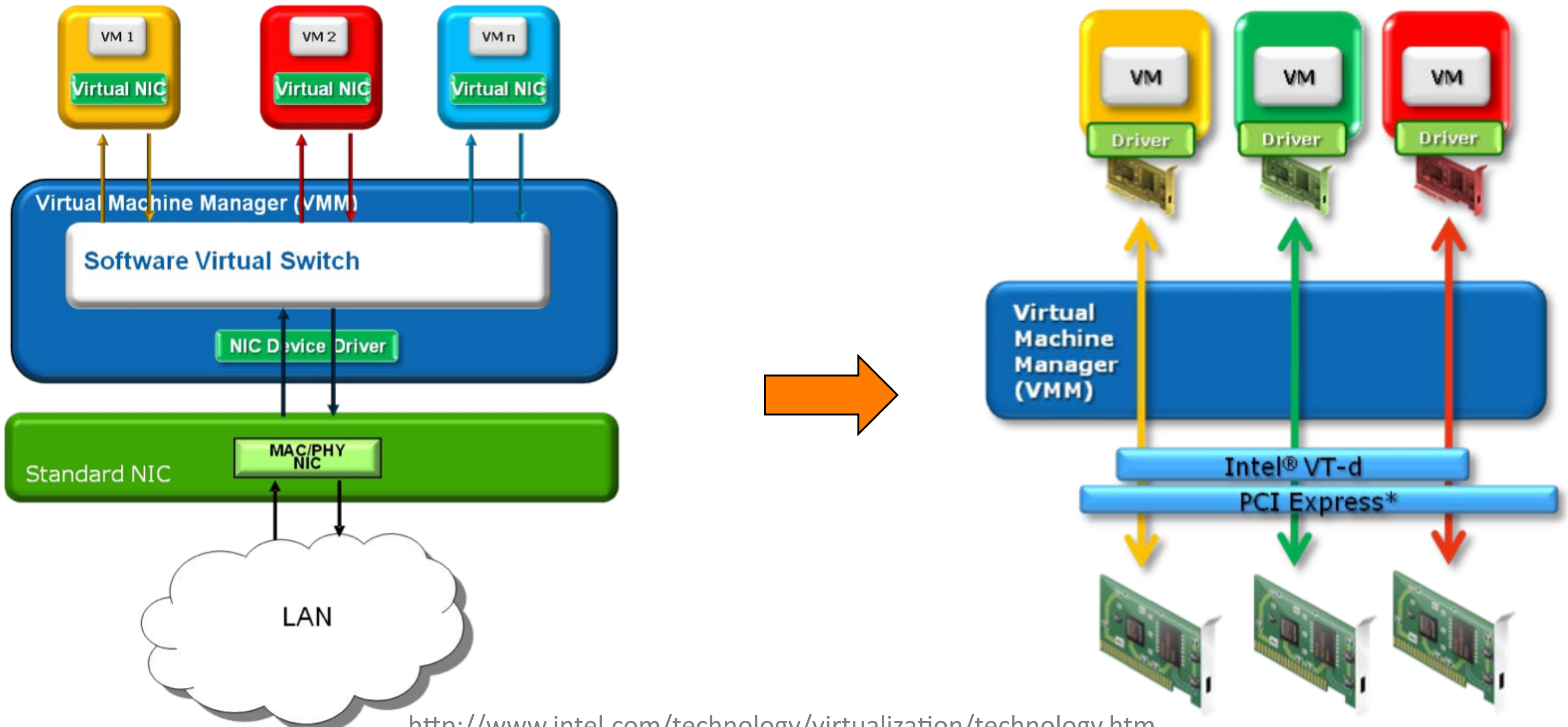
- First look at ways to improve the algorithms
- Work on better usage of single core capabilities
  - Exploit instruction and vector level parallelism
  - Identify and recode the critical parts in the code
  - Use a low level optimized library to vector operations
- Improve I/O and data handling
  - With so many cores, bringing enough data to CPU is going to be the biggest challenge
  - Multi level caching and buffering may be required
- Reduce memory utilization per core
  - Identify the pieces of information that can be shared between processes
  - Rework data structures so that sharing is possible
  - ✓ Share what can be shared (shared memory, COW, KSM)
- Parallelize execution of certain workflows
  - Ideally those that do not require or generate lots of I/O such as simulation

Evolution of virtualization technology

- If virtualization cannot help us, how can we be sure that it won't hurt us and is not going to be the bottleneck on many core boxes?

- Virtualization technologies that are now part of the latest generation of Intel/AMD CPUs
  - Processor (VT-x/AMD-V)
    - hardware virtualization support
    - Near native CPU performance
  - I/O MMU (VT-d/AMD-vi)
    - enables guest virtual machines to directly use peripheral devices, such as Ethernet, accelerated graphics cards, and hard-drive controllers
    - I/O performance was often seen as the weak spot for VMs
  - Network virtualization (VT-c)
    - Intel's "Virtualization Technology for Connectivity"
    - As VMs are most likely to use network for bulk of I/O operations, performance and flexibility of the network stack is probably the most important issue to solve

I/O Optimization (VT-d/AMD-vi)

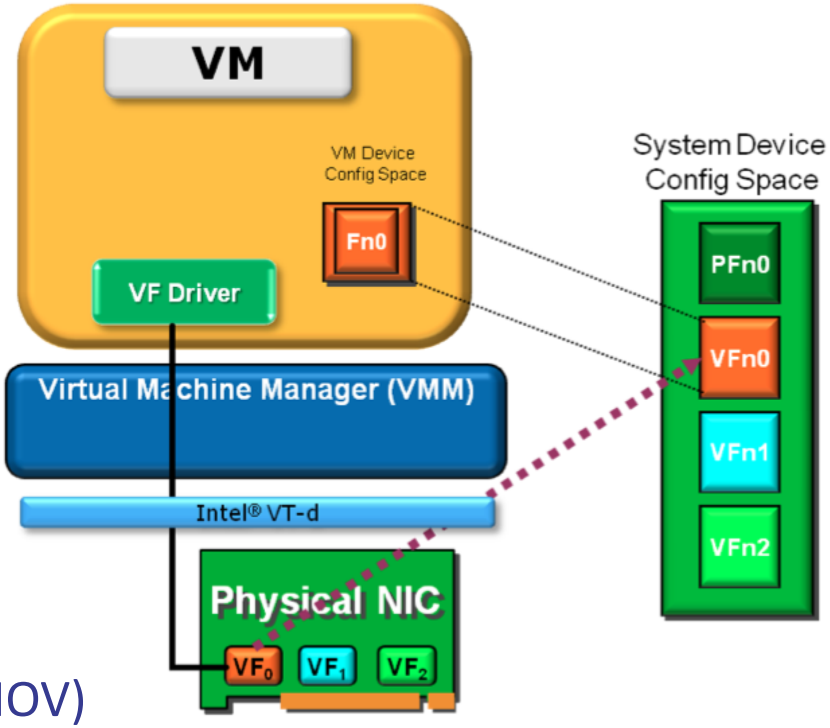


http://www.intel.com/technology/virtualization/technology.htm

- Each VM guest VM can directly use the peripheral devices (hard drive controller, NIC, GPU and perhaps even MIC)
- Drawbacks:
  - Only one guest can be assigned given physical resource
  - Once assigned a physical resource, VM cannot be transparently moved to another node

Intel's Virtualization Technology for Connectivity (VT-c)

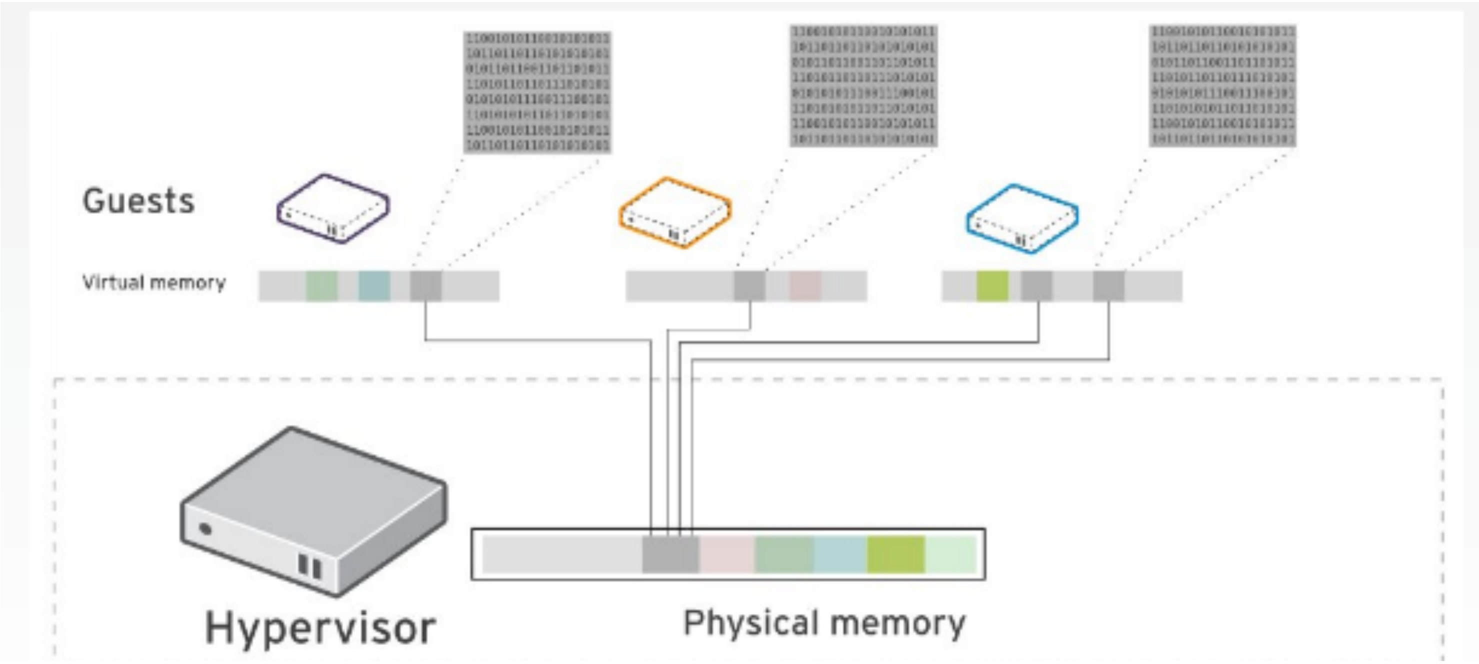
http://www.intel.com/technology/virtualization/technology.htm



- Virtual Machine Device Queues (VMDq)
- Improves traffic management within the server, helping to enable better I/O performance from large data flow while decreasing the processing burden on the software-based Virtual Machine Monitor (VMM)

- PCI-SIG Single Root I/O Virtualization (SR-IOV)
- Provides near native-performance by providing dedicated I/O to virtual machines, bypassing the software virtual switch in the hypervisor completely.
- It also improves data isolation among virtual machines, and provides flexibility and mobility by facilitating live virtual machine migration.

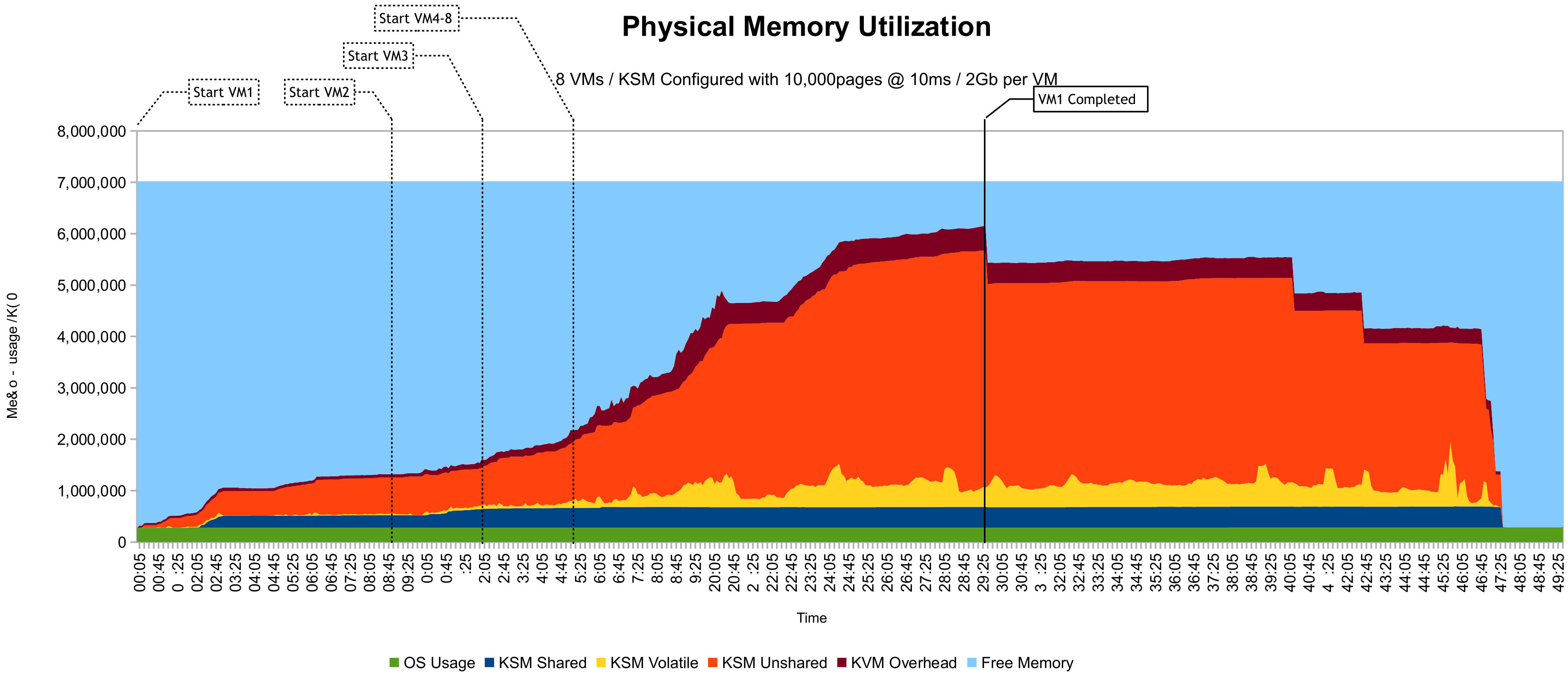
Sharing Memory with KVM/KSM



- This is where Virtualization can really help
- KVM/KSM (Kernel Samepage Merging) provides a mechanism to share the same memory pages between guests
  - If there is something to be shared, KSM it will be shared
- Equivalent mechanisms for memory sharing exists for other VMMs (VMware…)

An example: CMS simulation

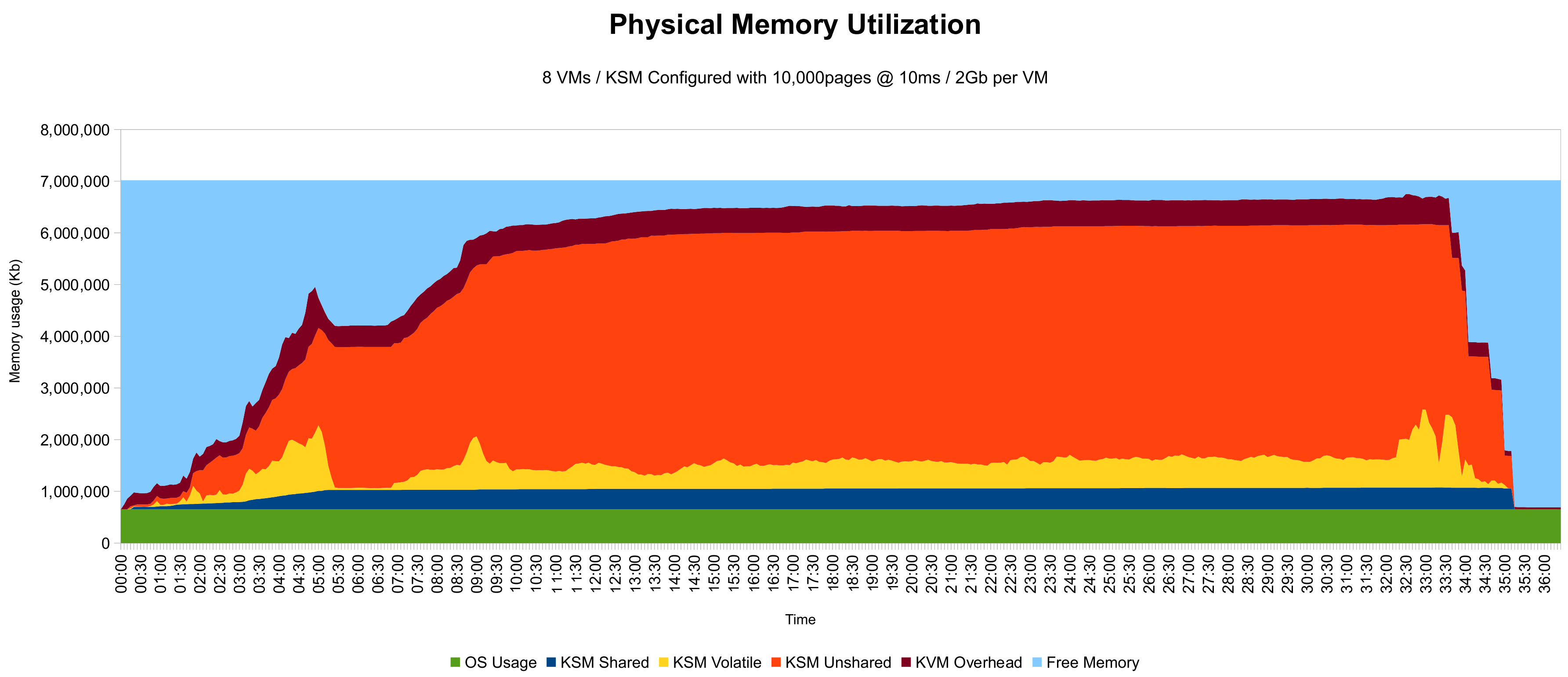
cmsDriver.py TTbar_Tauola_7TeV_cfi --conditions auto:startup -s GEN,SIM --datatier GEN-SIM -n 10 --relval 9000,50 --eventcontent RAWSIM --fileout file:step



- Running CMS simulation, 8 jobs in 8 VMs, 2GB/VM, 7GB host memory
  - Gradually increasing the number of running VMs
  - Memory sharing about 400 MB

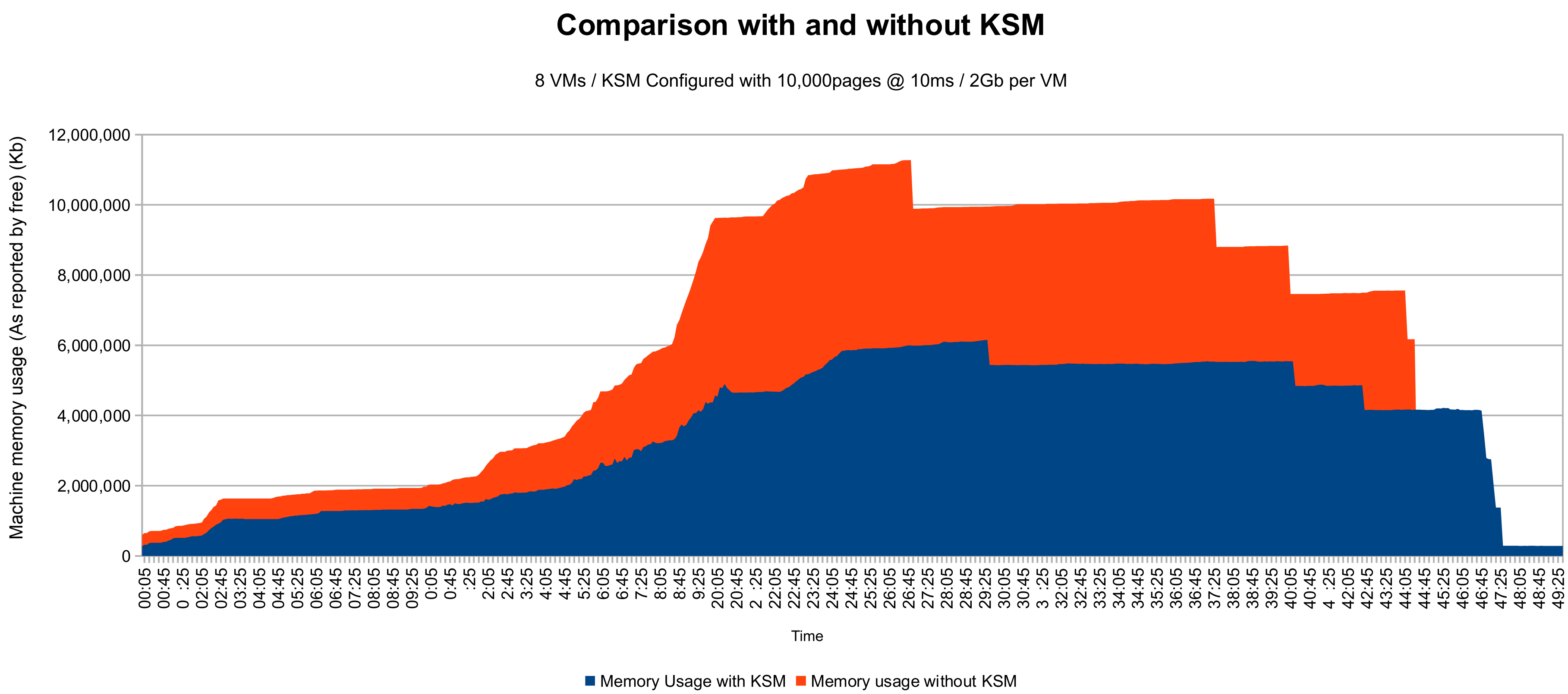
All VMs starting at the same time

cmsDriver.py TTbar_Tauola_7TeV_cfi --conditions auto:startup -s GEN,SIM --datatier GEN-SIM -n 10 --relval 9000,50 --eventcontent RAWSIM --fileout file:step



- The real question is what is actually being shared here:
  - Shared libraries?
  - Pages that are allocated not used?

Overall memory saving



- Still, with KSM, we can significantly overcommit the memory
  - Without KSM, all VMs would use up to 54% more memory
- However, there is no free lunch
  - Running with KSM adds about 6% to execution time

Next steps…

- Near term
  - Continue performance tests and establishing the baseline performance on the physical hardware
  - Develop a test suite to continuously monitor the performance of selected LHC applications in VM context
    - This could become a part of the CernVM test suite
    - Here we need some input and guidance form the experiments
  - Test multicore variants of experiment frameworks
    - Athena MP…
- Mid to long term
  - Depending on technology evolution and availability of the hardware
  - Investigate possible use of GPGPU/MIC from VM environment

Conclusions

if (cloudComputing == theFuture) {

- Virtualization is going to be the way of life
- Running multi and many core jobs on virtualized infrastructure is not necessarily contradiction in terms
  - The virtualization technology is evolving and addressing the key performance issues such as I/O
  - It is already possible to dedicate certain hardware devices such as GPGPU
  - It is very likely that we will soon be able to the same with MIC
- This gives complete flexibility to applications to choose their preferred environment and evolve with time
  - Sequential and parallel workflows can share the resources
- Pilot Job frameworks can be deployed directly in VMs reducing the load on batch scedulers
  - The resource owners remain in control of the physical hardware and VM scheduling
- With KSM (or equivalent approaches), substantial memory saving could be achieved to help solving one of the biggest problems facing the experiments

}