Linux-based virtualization for HPC clusters

Lucas Nussbaum, Fabienne Anhalt,
Olivier Mornard, Jean-Patrick Gelas

lucas.nussbaum@inria.fr

Laboratoire de l'Informatique et du Parallélisme
INRIA - UCB Lyon 1 - ENS Lyon - CNRS

Introduction

Virtualization :

- Subject of a lot of attention in the recent years
- Solves many problems in classic datacenters
- But limited adoption in High Performance Computing

Virtualization in HPC

Pros :

- Dynamic allocation of resources to job
  e.g automatically scale down a job from 100 to 10 machines
- Easier to share resources between different jobs
  Mix I/O-intensive with CPU-intensive jobs $\Rightarrow$ maximize usage
- Easy checkpointing of jobs
  Solves a long-term problem in HPC
- Deployment of job-specific work environment
  User no longer limited by administrator's choices

Virtualization in HPC

Cons :

- Overhead caused by the virtualization layer
  Poorly understood, rarely evaluated
- Non-exclusive access to hardware
  - Performance may vary
  - Harder to make use of HPC-specific devices
    High Performance networks (Infiniband, Myrinet)

Context of this work : HIPCAL Project

- Research project funded by ANR (= French NSF)
- Build virtualized infrastructures
  = virtual clusters spread over the Grid
- Combine system virtualization and network virtualization
  - multiple virtual nodes per physical node
  - multiple virtual overlay networks on a shared long distance communication infrastructure
    - Opportunity to allocate bandwidth per user (performance guarantees)
- Software : HIPerNET (still in development)

http ://hipcal.lri.fr/

Goal of this work

- Compare and evaluate two virtualization solutions
  - **Xen**
  - **KVM**
- In the context of High Performance Computing

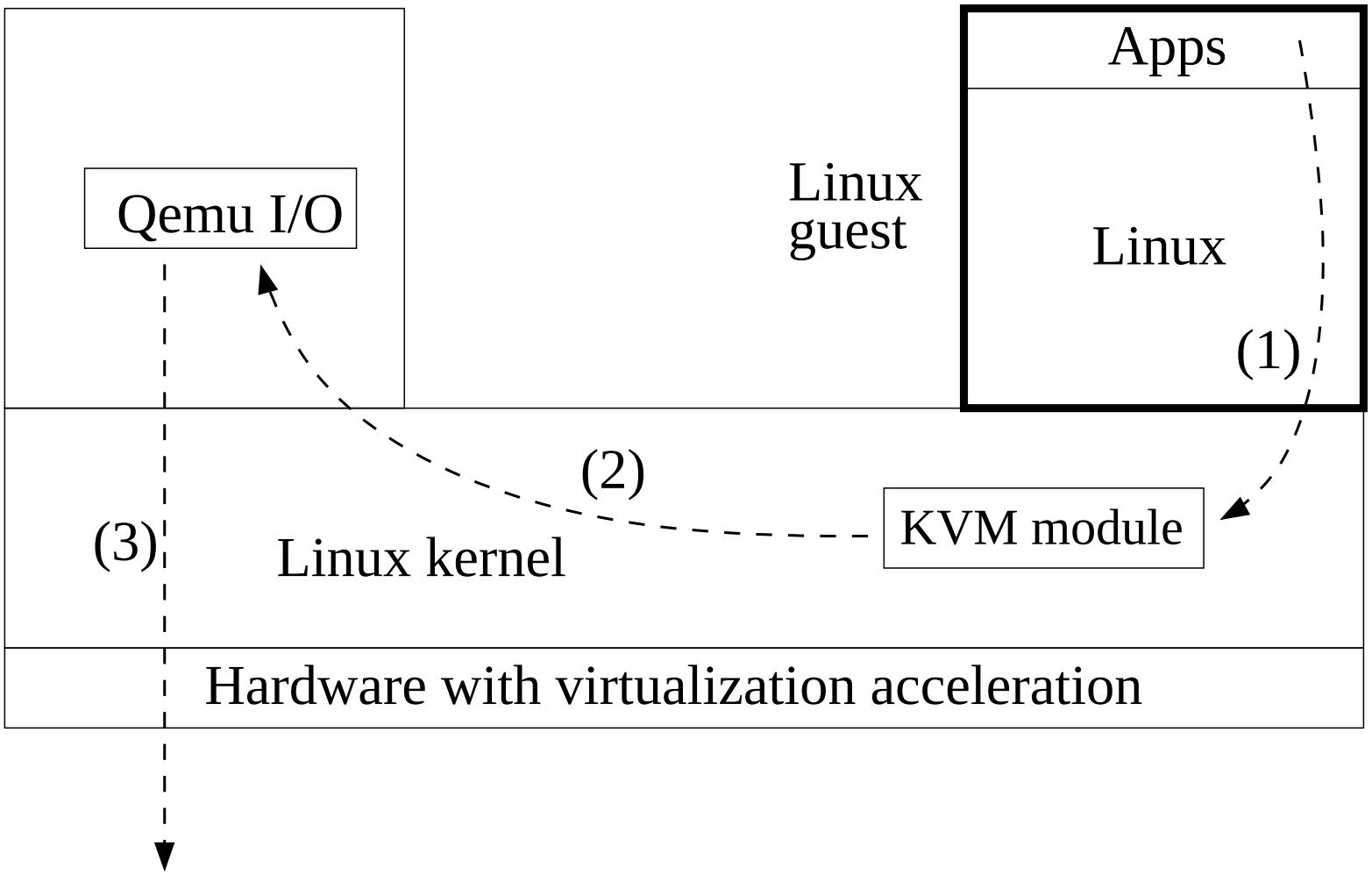
Xen

- Virtual Machine hypervisor
- First public release : 2003
- Usually uses Linux as dom0
- Developed outside Linux by XenSource (now Citrix)
  - DomU support in vanilla kernel
  - Dom0 merge under discussion
  - Distributions forward-port the Xen patch
- Several distributions have stopped supporting Xen recently, and focused on KVM

KVM

- Virtualization infrastructure integrated in Linux
  = *Linux as an hypervisor*
- Requires hardware virtualization support in CPU
- First release in 2007 (2.6.20)
- Developed by Qumranet (acquired by Red Hat)
- Relies on QEMU for I/O
  - Originally only QEMU emulated devices
  - Now paravirtualization with virtio (net, disk)

KVM : I/O path


Apps
Linux
guest
Linux
Qemu I/O
(1)
(2)
KVM module
(3)
Linux kernel
Hardware with virtualization acceleration

Evaluation : Benchmarks

Micro-benchmarks :

- CPU
- Disk
- Network

HPC Challenge benchmarks :

- Covers various aspects of High Performance Computing
- Used for the HPC Challenge @ SuperComputing
- Includes Linpack/HPL (used by Top500)
- Other benchmarks included : PTRANS, STREAM, LBB, ...

http://icl.cs.utk.edu/hpcc/

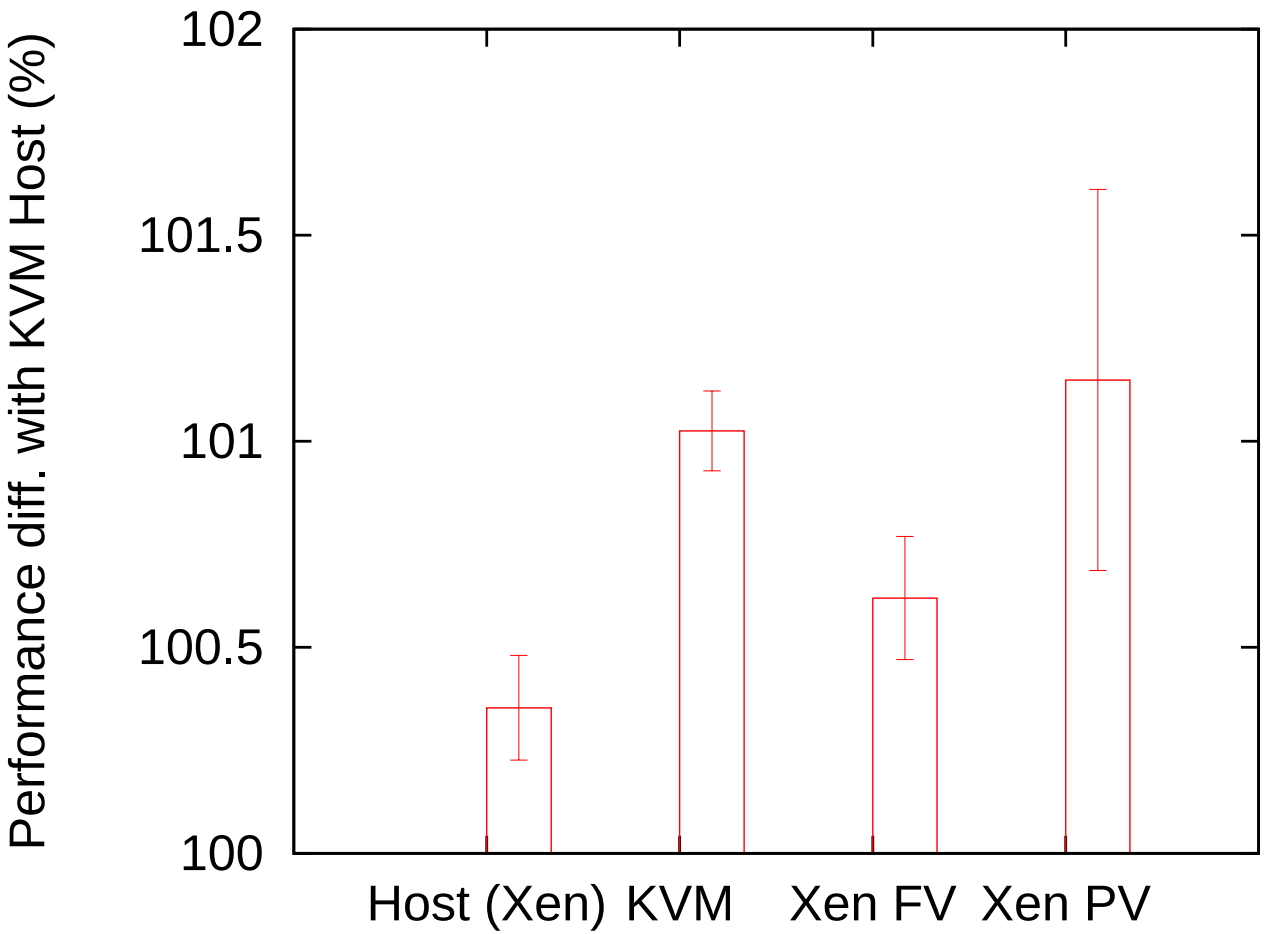
Evaluation : Setup

- 32-nodes cluster
  Dell PowerEdge 1950 (2 Dual-core Xen 5148 LV ; 8 GB RAM)
- Same userspace, based on Debian sid
- Different kernels (host and guest)
- 4 configurations evaluated :
  - Xen with full (hardware) virtualization (**Xen FV**)
  - Xen with paravirtualization (**Xen PV**)
  - Standard KVM, using Qemu devices (**KVM FV**)
  - KVM with virtio-based paravirtualization (**KVM PV**)
- Software versions :
  - Xen 3.3.1 + Linux 2.6.18 (from XenSource)
  - Linux 2.6.29 + KVM 84

CPU

- CPU-intensive synthetic benchmark
- Using 4 VCPU (mapped on 4 real CPU)
- Almost no memory used
- Compared with time on host system (2.6.29)

CPU

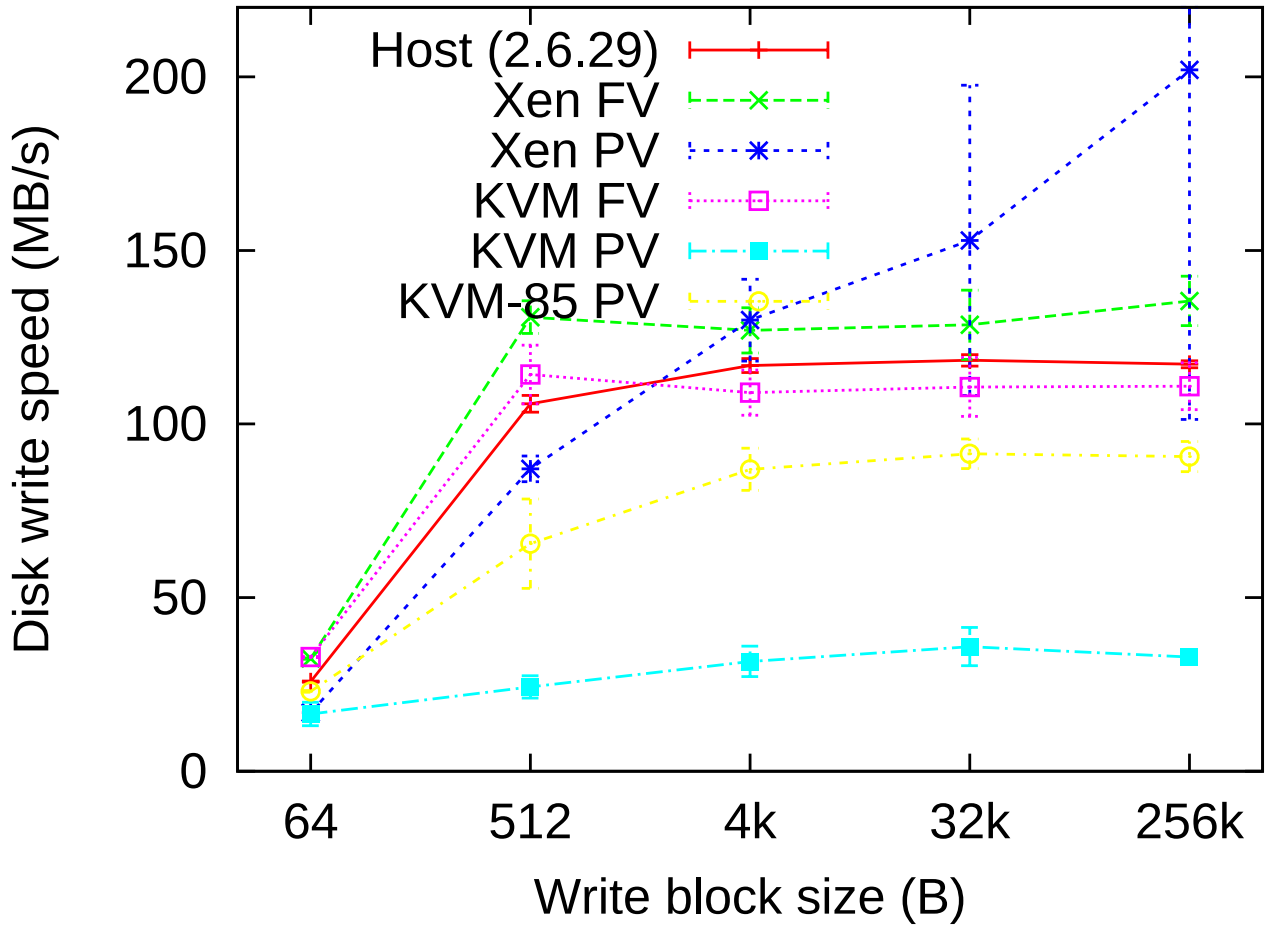


Similar results : almost no overhead

Disk

- Large file written using `dd`
- Varying block sizes
- Confirmed with `bonnie++`
- Host system using RAID-0 (expected 120 MB/s)

Disk

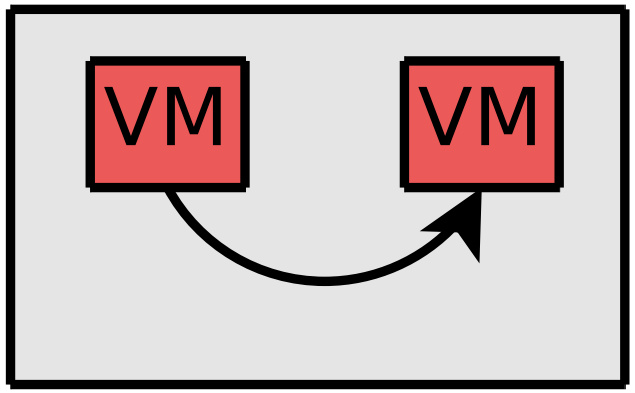


Xen acks writes before they are committed to disk ?
Bad KVM+virtio performance

Network

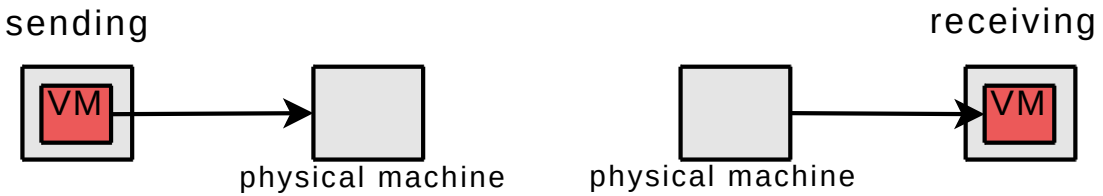
- Throughput measured using `iperf`
- KVM FV : e1000 NIC emulation
- Xen FV ; Realtek 8139 (no GbE NIC available)
- 3 configurations evaluated :
  - Communication between 2 VMs on the same machine
  - Communication between a VM and a remote system
  - Scalability (several VMs, several remote systems)

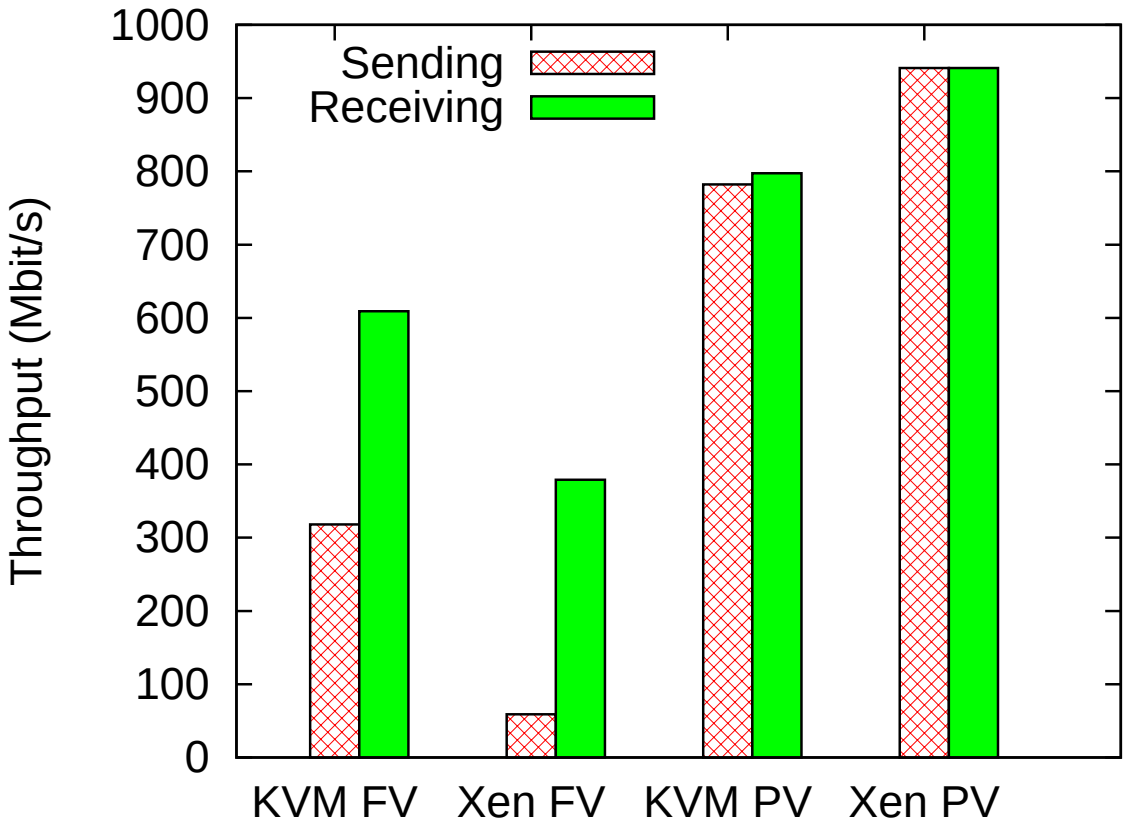
Network : inter-VM communications



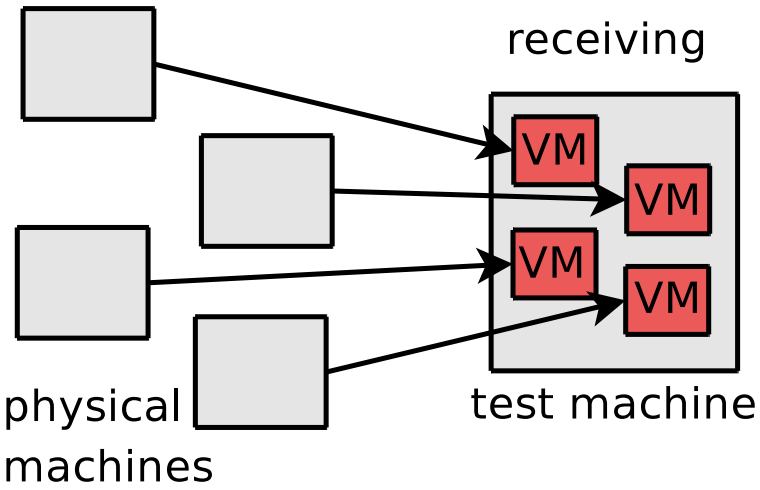
	FV	PV
KVM	648.3	813.2
Xen	96.05	4451

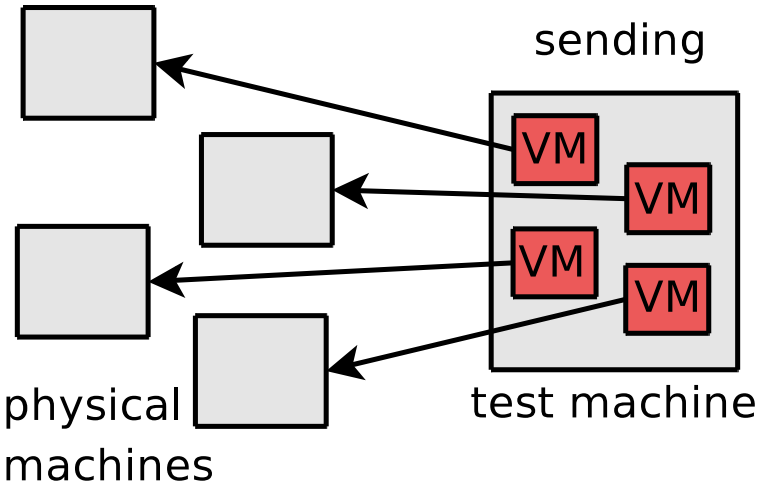
Network : sending/receiving to remote host



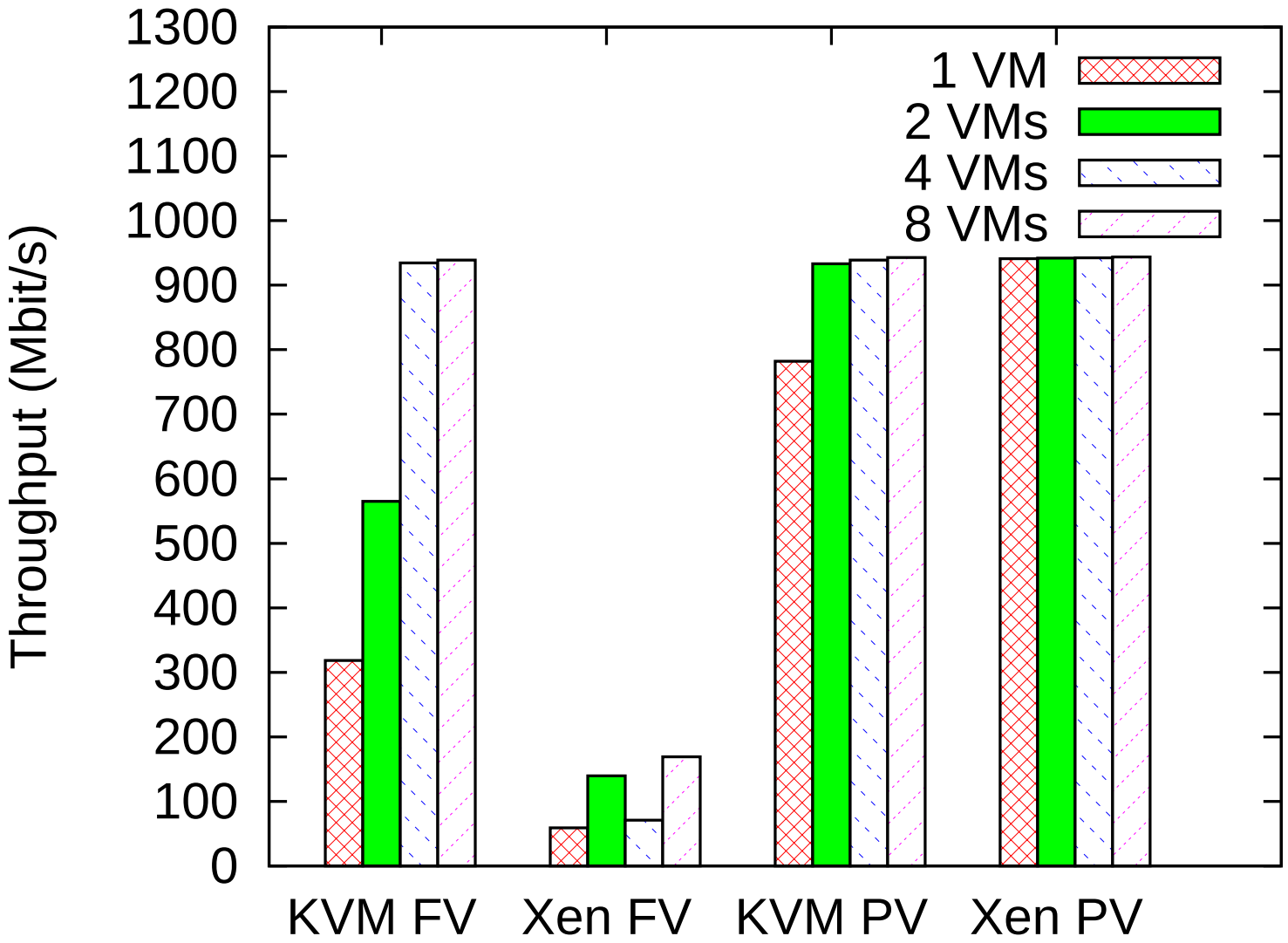


Network : several VM

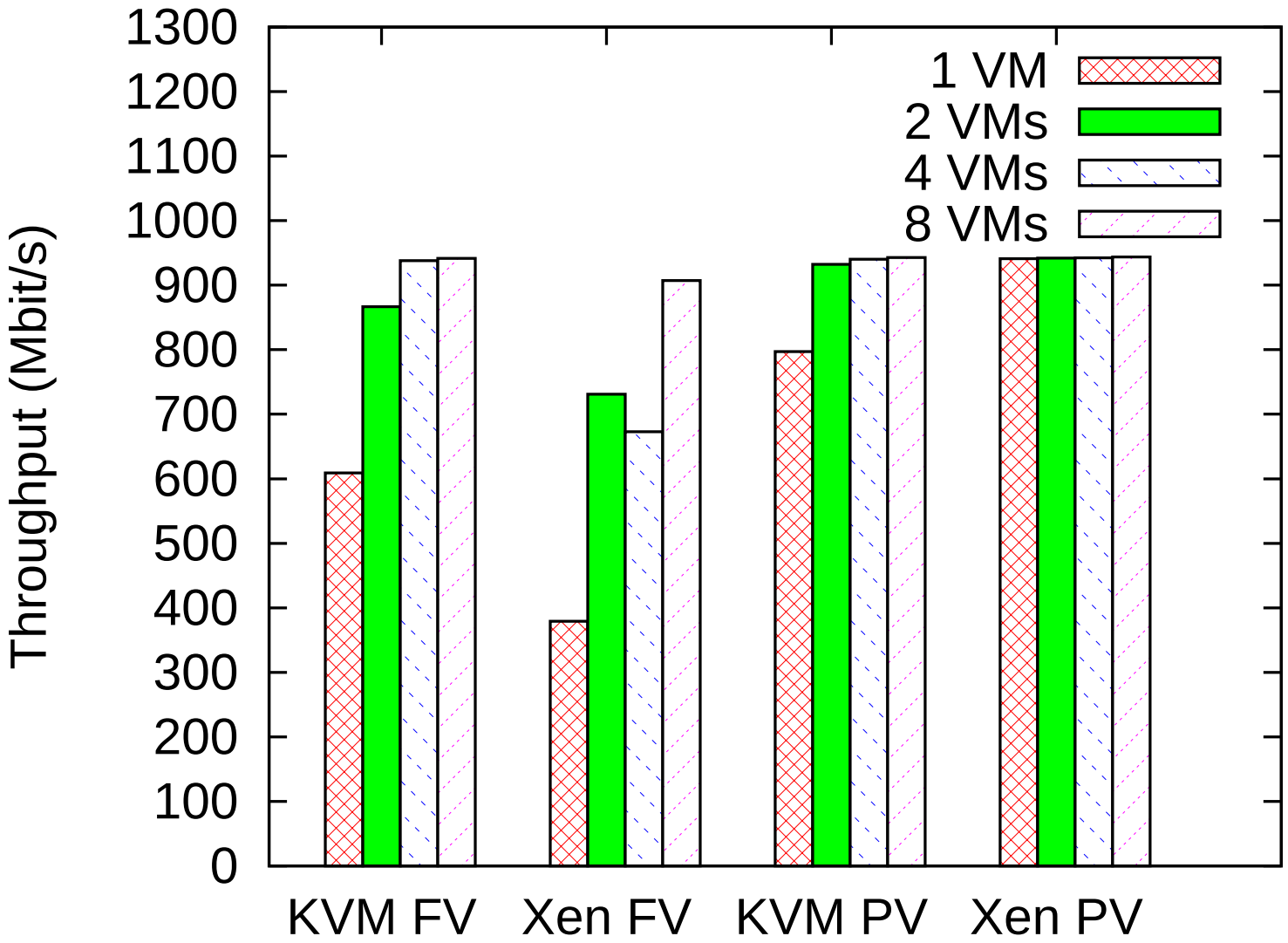




Network : several VM - sending



Network : several VM - receiving



Micro-benchmarks - summary

- CPU : similar results
- Disk : Xen better (but cheats ?), KVM+virtio has problems
- Network :
  - Xen PV very good
  - KVM+virtio OK
  - standard KVM a bit slower
  - Xen FV extremely slow

HPCC benchmarks

- 7 benchmarks used by the HPC Challenge
  - 3 used in this presentation
- Goal : evaluate all aspects of clusters
- 8 different configurations :
  - 32 host system - 2.6.29 + KVM
  - 32 dom0 - 2.6.18
  - 32 KVM VM with 4 CPU each, using virtio
  - 128 KVM VM with 1 CPU each, using virtio
  - 32 paravirtualized Xen VM, 4 CPU each
  - 128 paravirtualized Xen VM, 1 CPU each
  - 32 Xen VM with full virtualization, 4 CPU each
  - 128 Xen VM with full virtualization, 1 CPU each

HPCC benchmarks

Condensed Results - Base Runs Only - 216 Systems - Generated on Wed Jul 15 11:16:35 2009

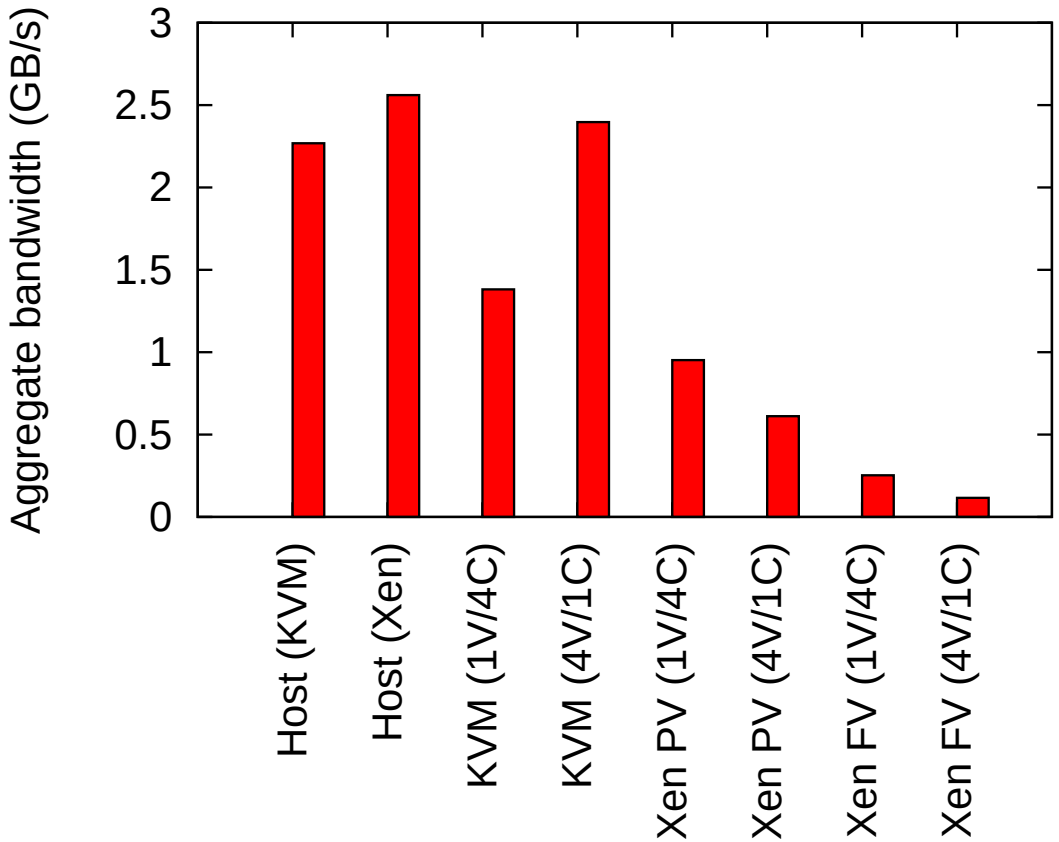
System Information System - Processor - Speed - Count - Threads - Processes					G-HPL	G-PTRANS	G-Random Access	G-FFTE	EP-STREAM Sys	EP-STREAM Triad	EP-
MA/PT/PS/PC/TH/PR/CM/CS/IC/IA/SD					TFlop/s	GB/s	Gup/s	GFlop/s	GB/s	GB/s	G
Atipa Conquest cluster AMD Opteron	1.4GHz	128	1	128	0.2526110	3.2471			208.525	1.6291	
cgna ma Intel Pentium 4	3.06GHz	14	1	28	0.0016225	0.0437	0.0017591	0.2725	13.775	0.4920	
Clustervision BV Beastie AMD Opteron	2.4GHz	32	1	32	0.1037640	0.8159	0.0002350	2.1470	106.951	3.3422	
ClusterVision/Dell/QLogic Darwin Intel Xeon 5160	3GHz	64	1	64	0.6327300	7.3008	0.2323370	14.3276	87.826	1.3723	
ClusterVision/Dell/QLogic Darwin Intel Xeon 5160	3GHz	128	1	128	1.2689700	15.8777	0.4199140	28.9749	176.827	1.3815	
ClusterVision/Dell/QLogic Darwin Intel Xeon 5160	3GHz	256	1	256	2.4601300	30.8088	0.7689970	54.3011	349.169	1.3639	
Cray Inc. Red Storm/XT3 AMD Opteron	2.4GHz	12960		125920	91.0350000	2356.9700	1.7401500	1554.0700	54840.499	2.1158	
Cray Inc. T3E Alpha 21164	0.6GHz	1024	1	1024	0.0481695	10.2765			529.242	0.5168	
Cray Inc. T3E Alpha 21164	0.675GHz	512	1	512	0.2231810	9.7741	0.0289464	15.4774	272.186	0.5316	
Cray Inc. X1 Cray MSP	0.8GHz	64	1	64	0.5215600	3.2288			959.334	14.9896	
Cray Inc. X1 Cray MSP	0.8GHz	60	1	60	0.5777790	30.4313			898.446	14.9741	
Cray Inc. X1 Cray MSP	0.8GHz	120	1	120	1.0609700	2.4603			1019.519	8.4960	
Cray Inc. X1 Cray MSP	0.8GHz	252	1	252	2.3847300	97.4076			3758.404	14.9143	
Cray Inc. X1 Cray MSP	0.8GHz	124	1	124	1.2054200	39.5252			1856.664	14.9731	

PTRANS benchmark

- Sends large messages between nodes
- Measures the total communication capacity

$\Rightarrow$ Result : aggregate bandwidth

PTRANS benchmark

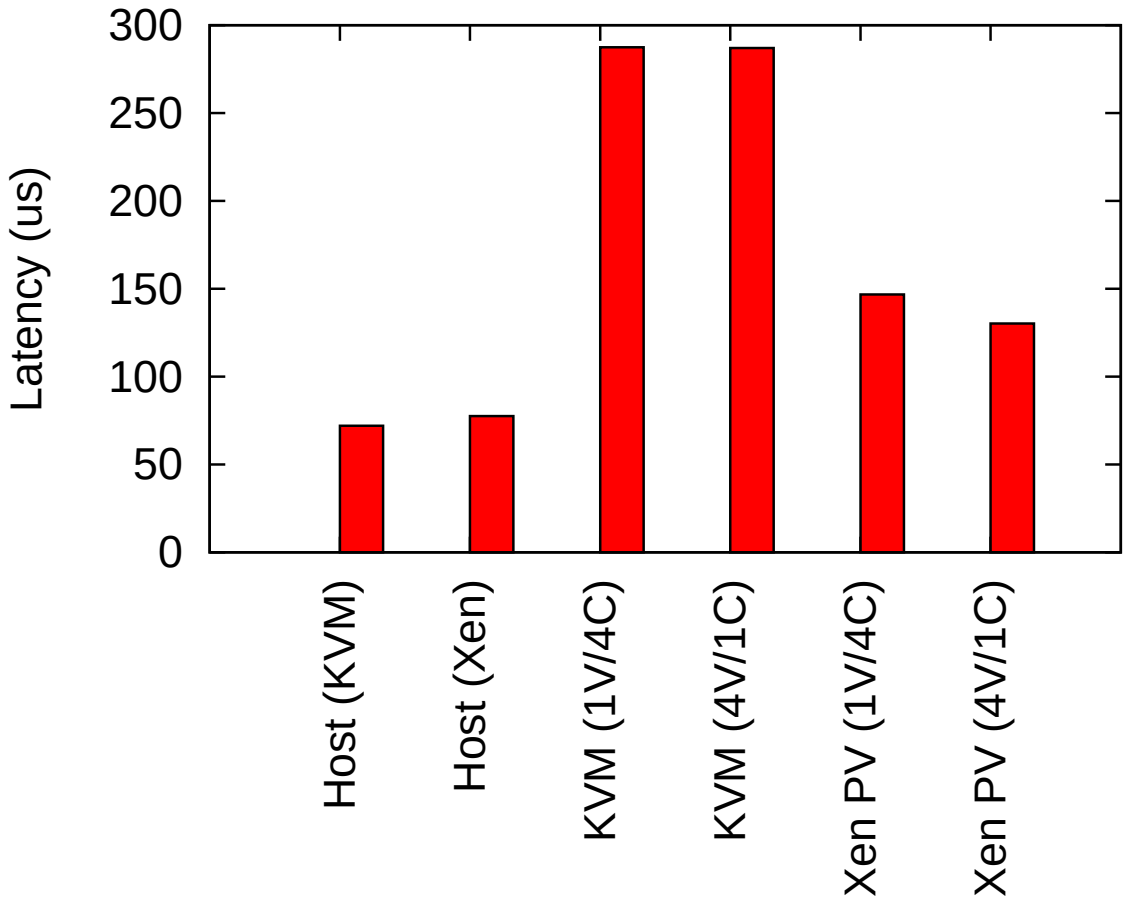


4 KVM VM per node $\Rightarrow$ better spread the I/O load ?
Poor Xen performance

Latency and Bandwidth benchmark

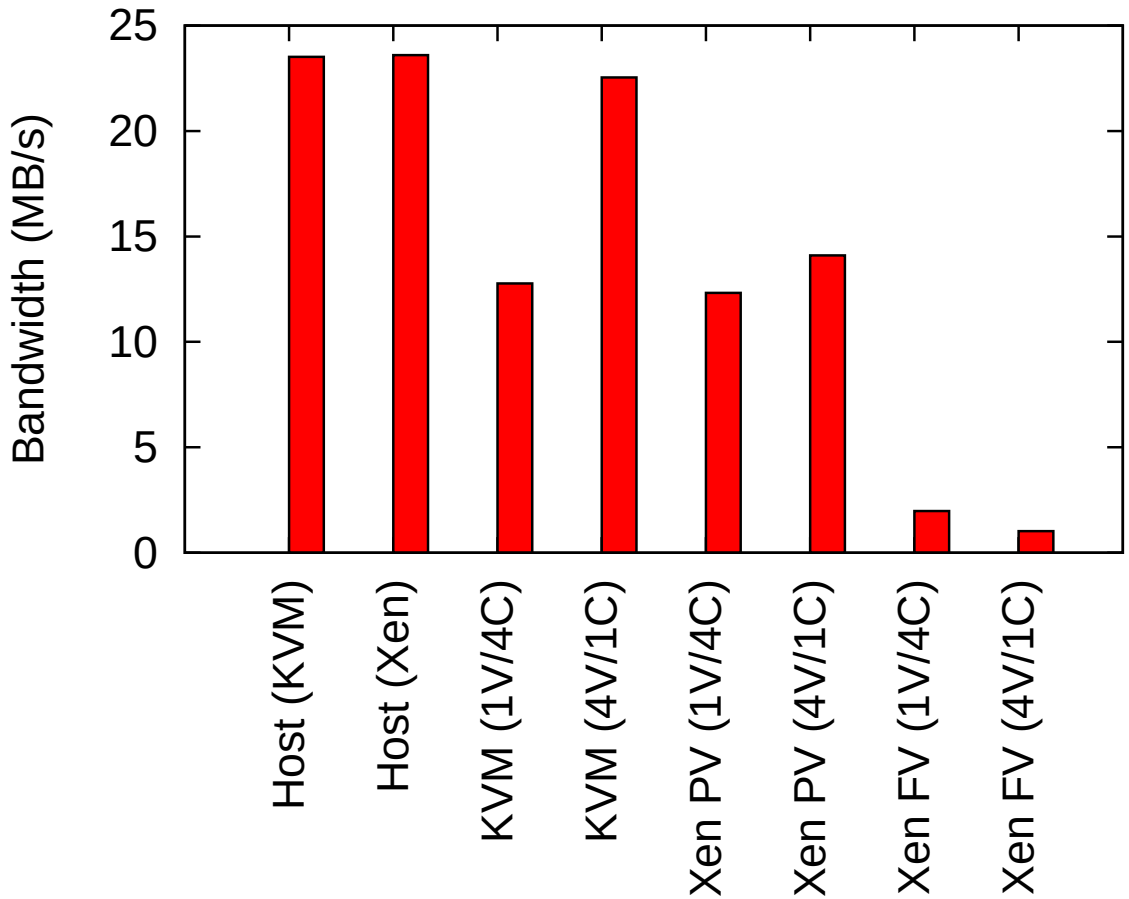
- Messages are being sent in a ring (ping-pong)
- Two sizes : 8 B and 2-MB
- Results : average node-to-node latency and bandwidth

Latency and Bandwidth benchmark



Latency : higher overhead with KVM

Latency and Bandwidth benchmark

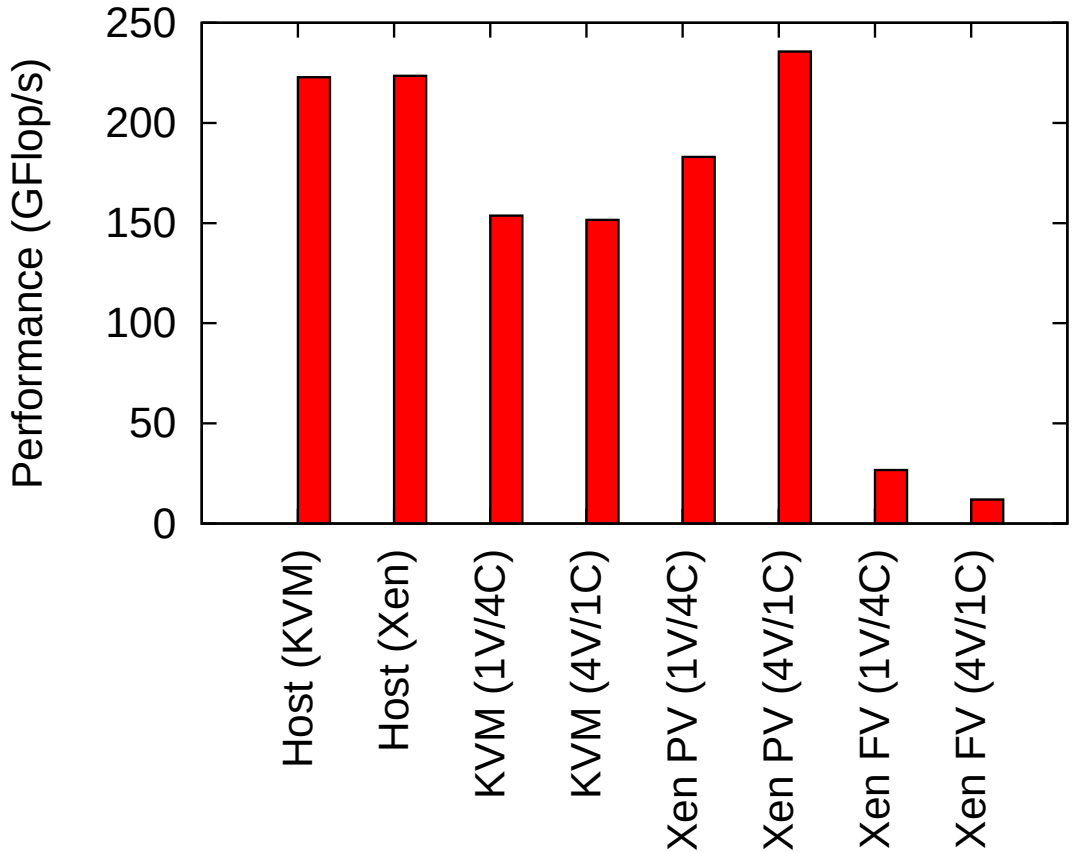


Bandwidth : better bandwidth with KVM (4 VM / node)

Linpack/HPL

- Combines computation and communication
  - Gives a global picture
- Used for Top500
- Sensitive to latency

Linpack/HPL



Host : 223 Gflop/s

KVM 1V/N : 154 (-31%)

KVM 4V/N : 152 (-32%)

Xen 1V/N : 235 (+5%)

Xen 4V/N : 183 (-18%)

Xen outperforms all other solutions
Even faster than host with 1 VM / node

Conclusion

- KVM and Xen : different performance characteristics
  - Xen+paravirtualization generally faster
- KVM : still has some rough edges
- Xen : poor support from distribution, harder to setup
- Future work :
  - Other aspects of virtualization : fairness, checkpoint/migration
  - PCI pass-through for high performance networks
  - Re-evaluate with new releases : Xen 3.4, KVM-88
  - Linux Containers : promising alternative ?

Linux-based virtualization for HPC clusters

Lucas Nussbaum, Fabienne Anhalt,
Olivier Mornard, Jean-Patrick Gelas

lucas.nussbaum@inria.fr

Laboratoire de l'Informatique et du Parallélisme
INRIA - UCB Lyon 1 - ENS Lyon - CNRS