# High Performance I/O in Virtualized Environments: A Networking Perspective

National Technical University of Athens
CSLab

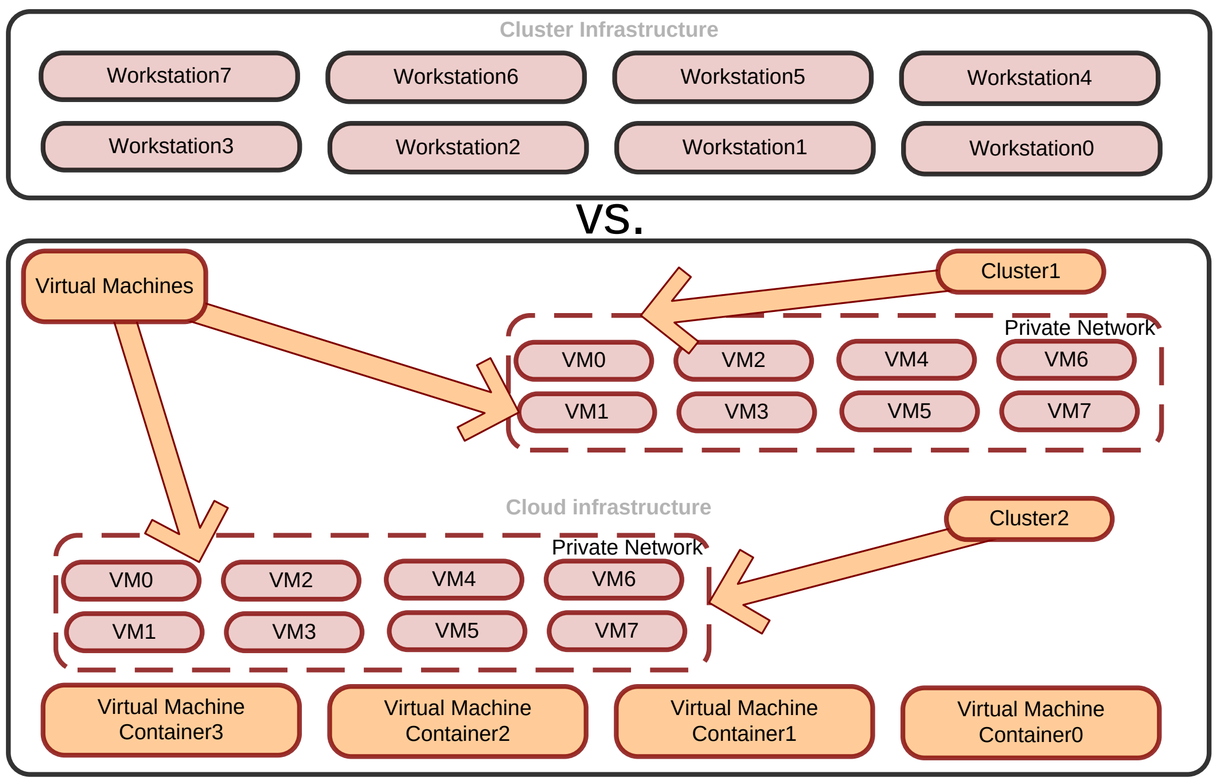
Anastassios Nanos, Nectarios Koziris
National Technical University of Athens
ananos,nkoziris@cslab.ece.ntua.gr
http://www.cslab.ece.ntua.gr/~ananos

## Motivation

- Server Consolidation
- Power, cooling, management
- Scientific Applications in Cloud Computing
- Stronger Infrastructures
- Cluster of VMs vs Clusters of Workstations

## Roadmap

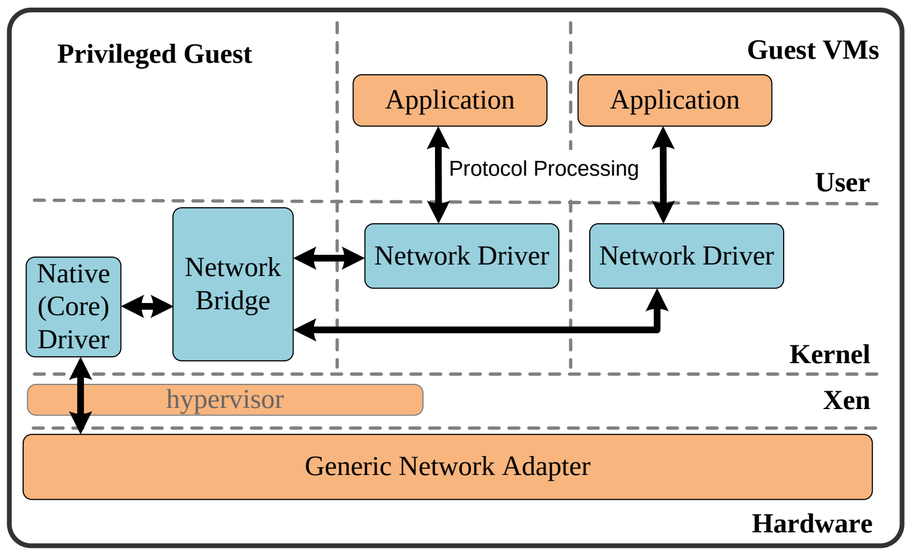
- Explore the implications of alternative datapaths
- Evaluate the performance of different setups
- Propose a framework for efficient I/O device sharing
- Evaluate the framework



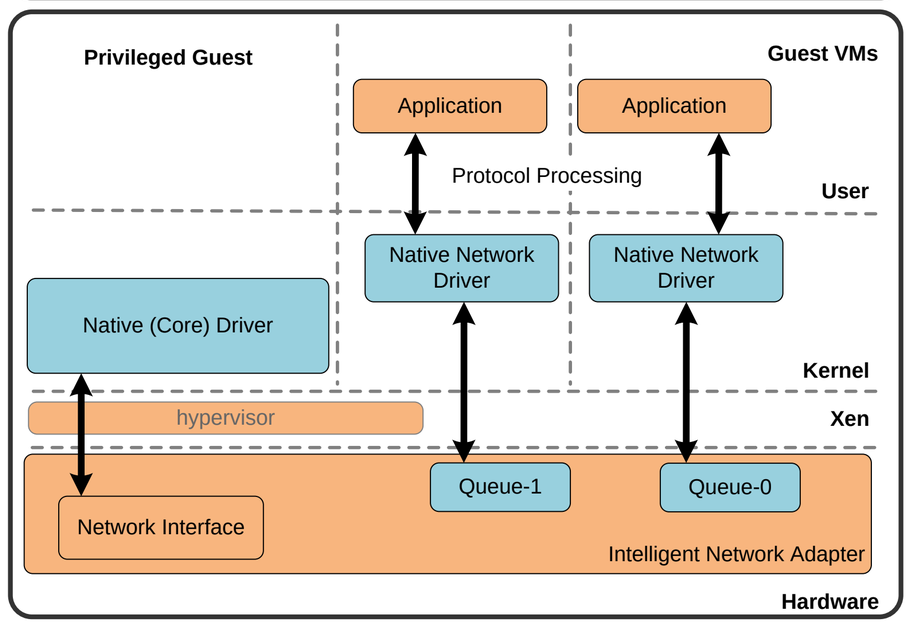

## Initial Setup

Our goal is to discover the *I/O bottlenecks* that occur in Virtualization environments when they host Scientific Applications. From the *networking perspective* we setup three different configurations for evaluating MPI applications in a Cluster of Xen VMs: (a) Bridged (b) using I/O Virtualization techniques and (c) using smart mappings (Xen's split driver model).

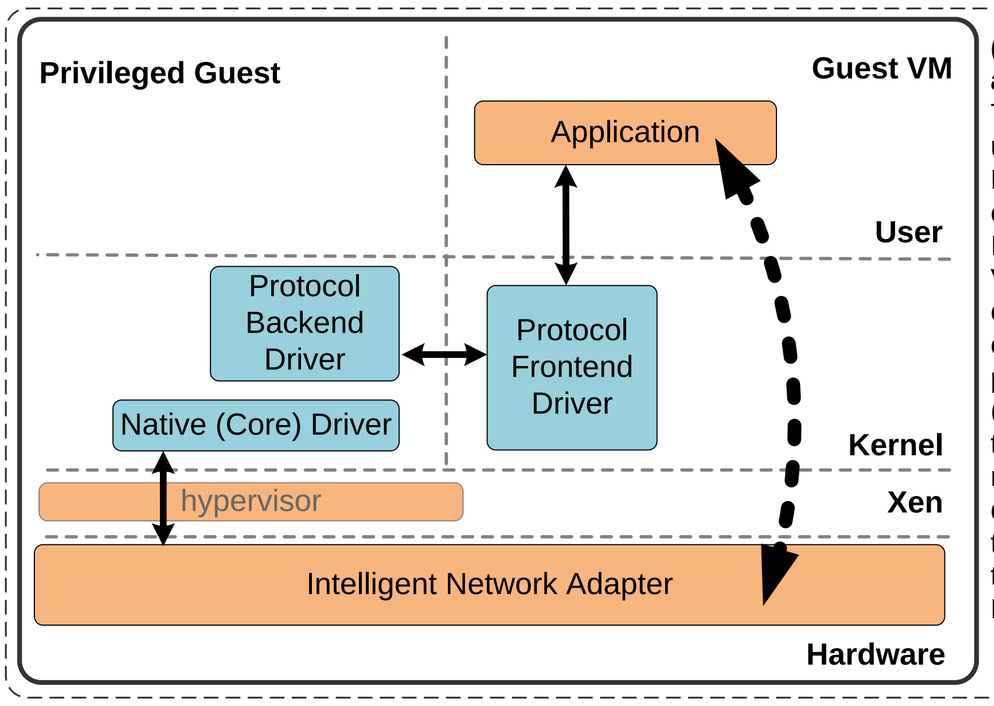
**(a) Bridged Case**
This is the default configuration provided by Xen. All guest VMs share a common bridge, setup by the privileged guest. Data flows from applications to the privileged guest via copying or page "flipping" and thus the software bridge becomes a bottleneck when it comes to High-performance I/O.




**(b) I/O Virtualization Case**
The Network Adapter exports Virtual Functions that are setup by the Privileged Guest. Each of these Functions provide Virtual Interfaces that are assigned to a guest VM and as a result, a direct data path is installed between applications and the network hardware.



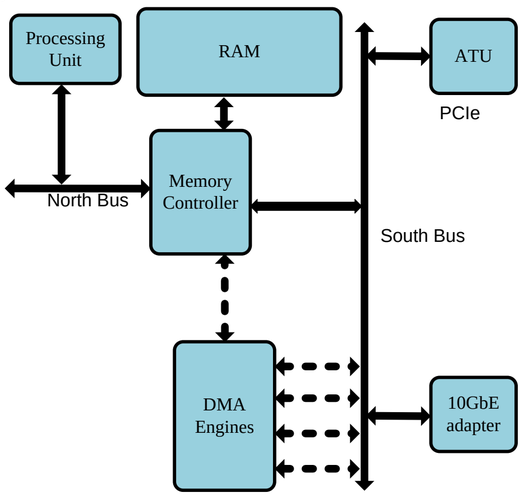

**(c) High-performance approach**
The Network Adapter, using hints from the Privileged Guest exports Virtual Instances to the guest VMs. By establishing communication channels with the privileged guest (endpoints), the VM transmits data to the network via a direct data path, installed by the Network Adapter, the hypervisor and the Privileged Guest.




## Intriguing Issues

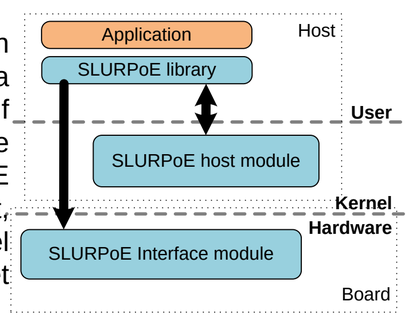
(a) **Application characterization**
- In which way are scientific applications I/O intensive ?

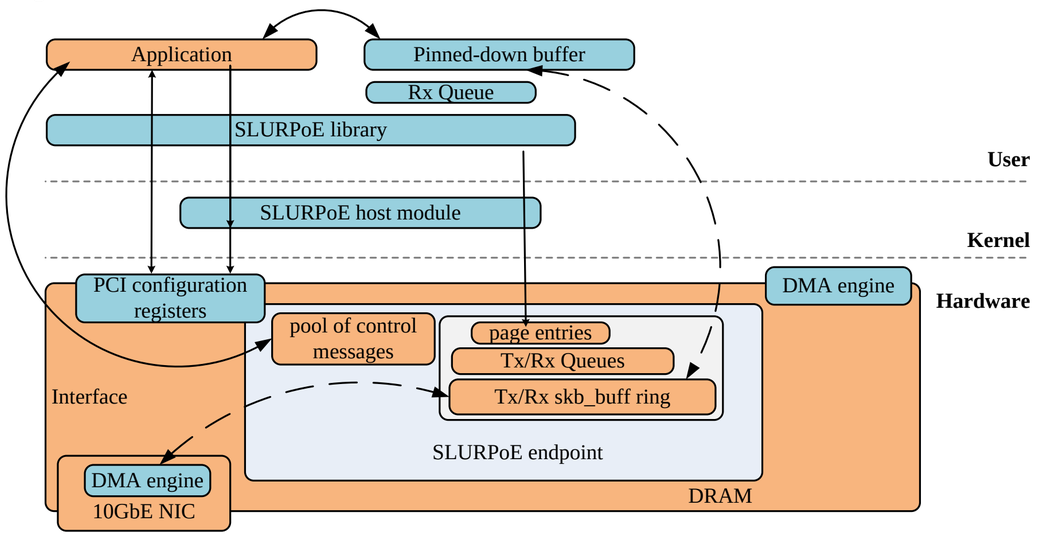
We are in the process of evaluating application-level benchmarks (as well as micro-benchmarks) in order to characterize the networking behavior of scientific applications. Placing MPI processes in a "network-aware" way can reduce the network load. On the other hand, we must take into account shared-memory mechanisms for intra-VM communication. Preliminary results show that I/O Virtualization mechanisms, such as SR-IOV, can reduce the Xen bridge overhead but several issues arise concerning interrupt handling, process-to-CPU assignment and VM migration.




(b) **Protocols**
- Do we need something more than a *message-passing protocol* and *Ethernet* ?
- Is the TCP/IP stack an overhead for HPC ? Or by using TOE (TCP Off-load Engines) we can alleviate its impact ?
- In the many-core era, do we need *intelligence* (CPU, memory, off-load engines) on the Network Adapter or we can just use one core to handle network traffic ?

In order to study the behavior of an Intelligent Network Adapter, we build a custom adapter, using off-the-shelf components such as an IOP XScale ARM processor and a generic 10GbE interface. To evaluate our interconnect, we design and implement a user-level RDMA protocol over Ethernet (SLURPoE).

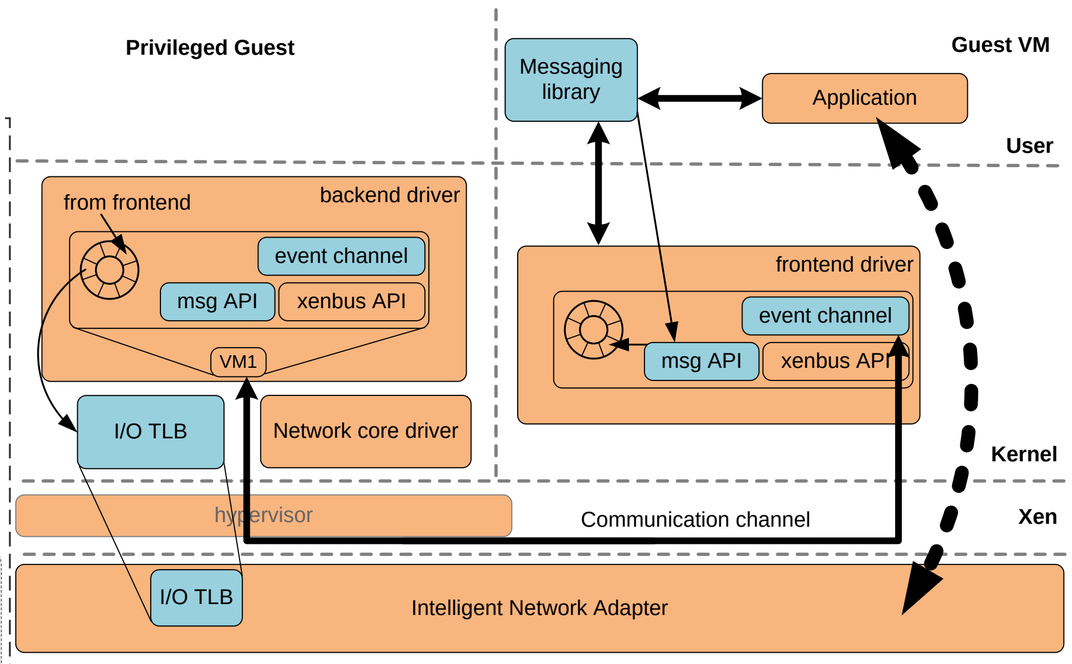



Our goal is to finetune the intelligence of a modern, high-performance interconnect in order to sustain line-rate bandwidth and low-latency communication in Virtualized environments. Preliminary results have already indicated the main bottlenecks of such an approach: *doorbell mechanisms* and *interrupt handling*.

We are in the process of evaluating the performance of MPI application-level benchmarks in a Cluster of Xen VMs using intelligent network adapters. We focus on:
(a) the implications of alternative data paths (direct or indirect) between applications and network hardware
(b) an optimized solution for running scientific applications that starve for network device access in Virtualized infrastructures.

To achieve this, we experiment with assigning network Virtual-Physical Functions to VMs. The objective of this work is to compare the applications' performance to a setup with Virtual Network Interfaces (multiplexing in hypervisor-level vs. firmware-level via IOV techniques). Thus, we can provide a feasible solution to I/O bottlenecks that arise due to intermediate software layers (hypervisor or privileged domains) by installing direct data paths and examining performance degradation issues (interrupt handling, doorbell mechanisms, overlapping techniques etc.).




**References**
- J.R. Santos, Y. Turner, G. Janakiraman, and I.A. Pratt: Bridging the gap between software and hardware techniques for I/O Virtualization. In Proceedings of USENIX 2008 Annual Technical Conference, Berkeley, CA, USA, USENIX Association (2008) 29–42
- A. Nanos and N. Koziris: MyriXen: Message Passing in Xen Virtual Machines over Myrinet and Ethernet", In Proceedings of the 4th Workshop on Virtualization in High-Performance Cloud computing, held in conjunction with Euro-par 2009, Delft, The Netherlands, 24-28 August, 2009
- L. Youseff, R. Wolski, B. Gorda, and C. Krintz: Evaluating the Performance Impact of Xen on MPI and Process Execution For HPC Systems. In Proceedings of the International Workshop on Virtualization Technologies in Distributed Computing (VTDC), 2006.