PDP 2012

Garching, Germany

Running User-provided Virtual Machines in Batch-oriented Computing Clusters

Vítor Oliveira, António Pina, André Rocha

{vspo,amp,arocha}@di.uminho.pt

Departamento de Informática,
Universidade do Minho,
Portugal

17th February 2012

Outline of the presentation

1. Introduction
2. The User Domains Platform
3. Performance evaluation
4. Conclusions

Problem statement

In HPC users have no control over the base software installed in the compute nodes, including the OS type, the kernel versions, the installed modules and services, etc.

But why is that a problem?

Because more control may be needed if you want to:

- Use software the administrator cannot install;
- Use software the administrator does not want to install;
- Use the software now, not later;
- Maintain consistent user setups after system updates.

Problem statement

User control over the software configuration may also be more practical on some occasions:

- To quickly deploy experimental configurations;
- To maintain more coherent environments for legacy applications;
- To better support software that is difficult to adapt to batch-oriented clusters.

Possible solutions

What to do when you cannot ask the administrator for the “keys to the cluster”?

Mess up some other infrastructure:

- Buy your own cluster;
- Rent virtual machines in a cloud;

- Build a user-mode infrastructure overlay and then run the user-provided and managed virtual machines safely on the computing resources allocated in a cluster!

Main drawbacks

What will you loose with this solution?

1. Performance
   - Virtual machines are slower than the real machines they run in;
   - An exclusively user-mode overlay imposes an even higher overhead.
2. Configuration effort
   - Virtual machines must be provided/updated by users;
   - The network access to the user virtual machines must be adapted to an user-mode overlay.

Our approach

User Domains:

- An entirely user-level platform with the ability to execute virtual machines inside batch jobs that is compatible, in the same nodes, with more typical applications.

It requires:

- a process level hypervisor that can run VMs in user-mode in the compute nodes and access a user-mode network overlay;
- a network overlay that safely interconnects the user VMs and links them to the exterior networks;
- a management layer with the ability to migrate VMs between the assigned job time slots.

The User Domains architecture

System Agents:

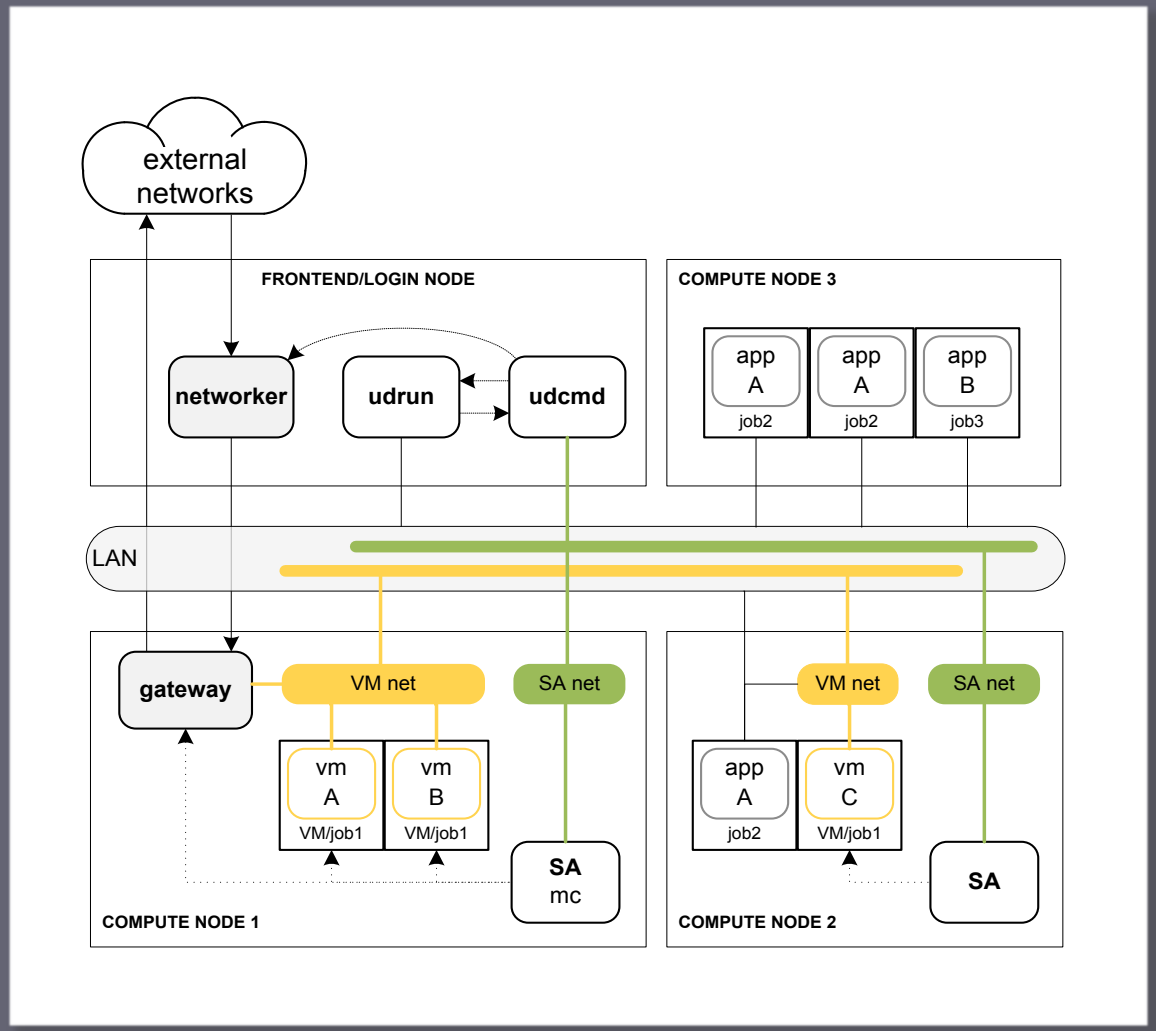
- User daemons in every node;
- Interconnected by a reliable network.

VM overlay network:

- L2 Ethernet with DHCP, DNS and NAT
- Port publishing on the network edge

User interface

- A startup app. (udrun)
- A system management application (udcmd)



The System Agent

SAs run in the nodes assigned to the user to:

- Manage the supporting software in the nodes, including spawning and monitoring the hypervisors and the VM overlay network daemons;
- handle remote requests such as migrating the VMs between nodes or suspending a VM to disk.

One SA is elected the main controller to:

- initiate, terminate and repair the VM network overlay;
- to execute the user-defined function to monitor the behavior of the system.

Communication between System Agents

A reliable group communication library is used to communicate between agents to:

- Manage membership;
- To elect main controllers coherently;
- To use the Virtual Synchrony model.

Failures are consistently converted to group membership changes consistently seen by all working nodes when nodes leave and enter the overlay;

Mainly to facilitate our future developments on this platform.

VM execution in batch-jobs

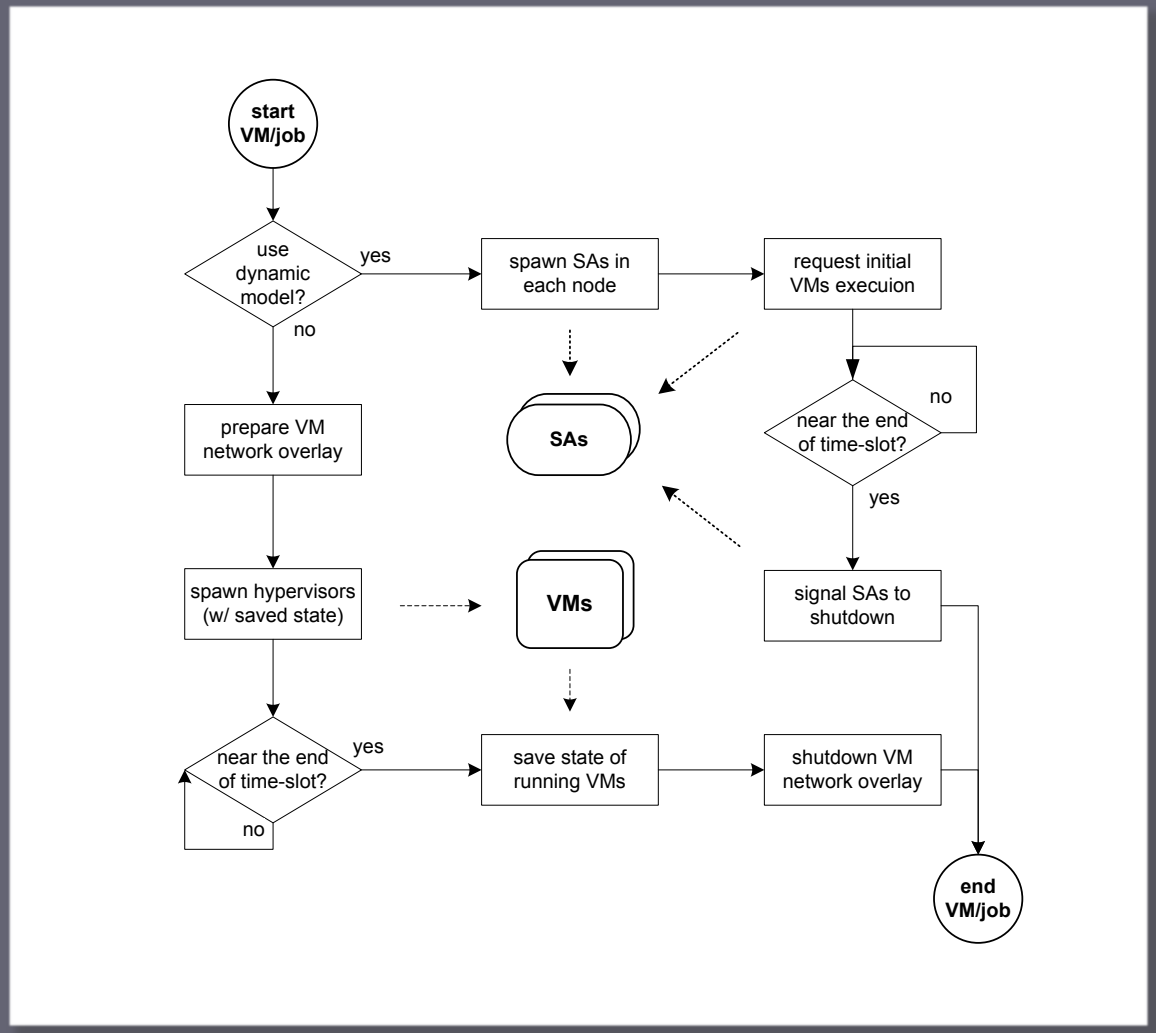
Two working modes:

1. Static

   the VMs run entirely using the resources of a single batch job;

2. Dynamic

   the VMs migrate between the allocated jobs node time-slots.



Implementation

User Domains implementation depends particularly on:

- The QEMU/KVM hypervisor;
- VDE software switch;
- the Spread toolkit.

Can use any batch system (only the job startup script needs to be adapted);

Overhead measuring

In order to give idea of the overhead, the performance of two sets of benchmarks were analyzed:

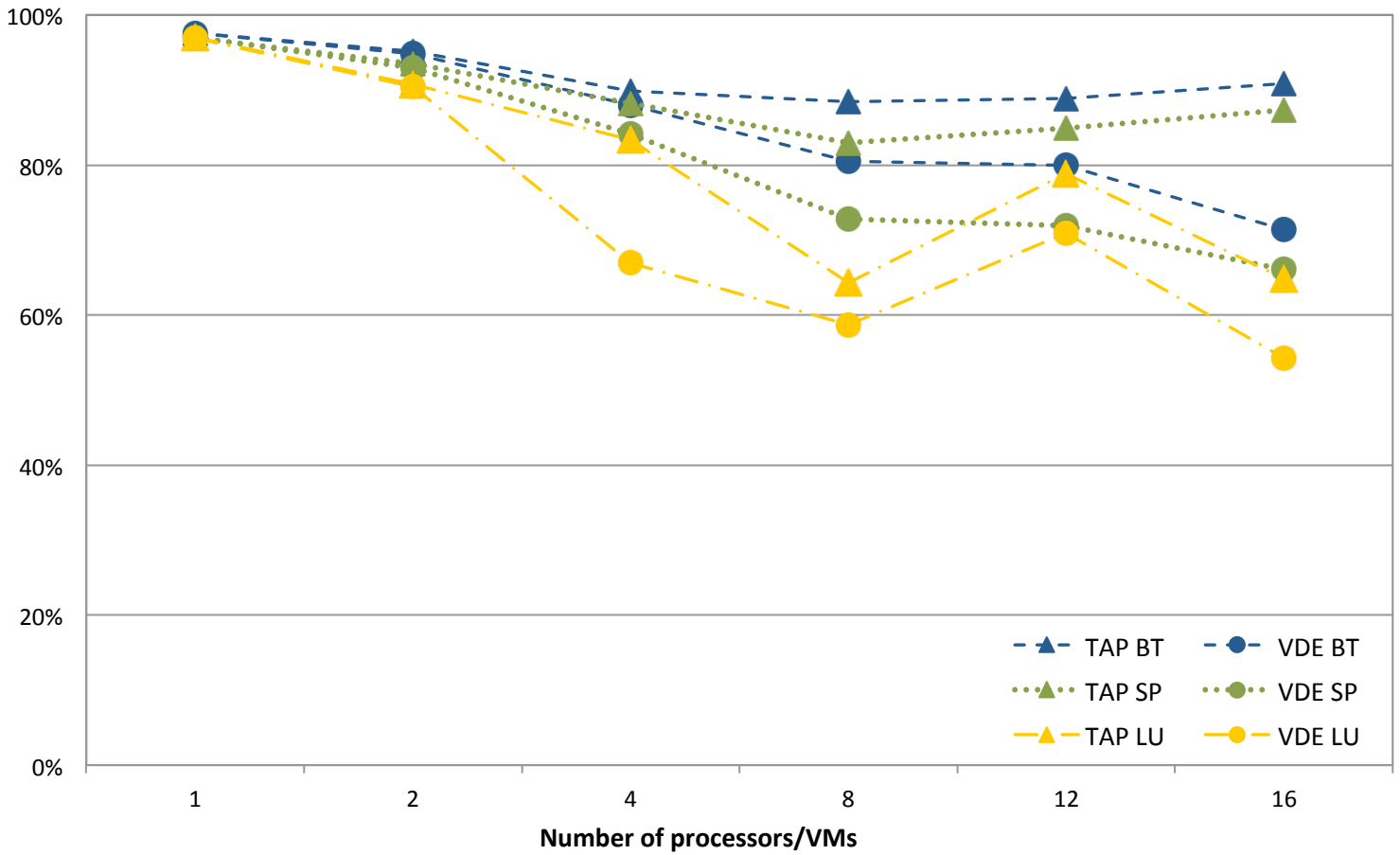
- NAS MZ to simulate computational fluid dynamics applications;
- NetPIPE to evaluate the raw bandwidth and latency of the Ethernet network overlay.

Native performance was compared to that of VMs on the same host and on different hosts.

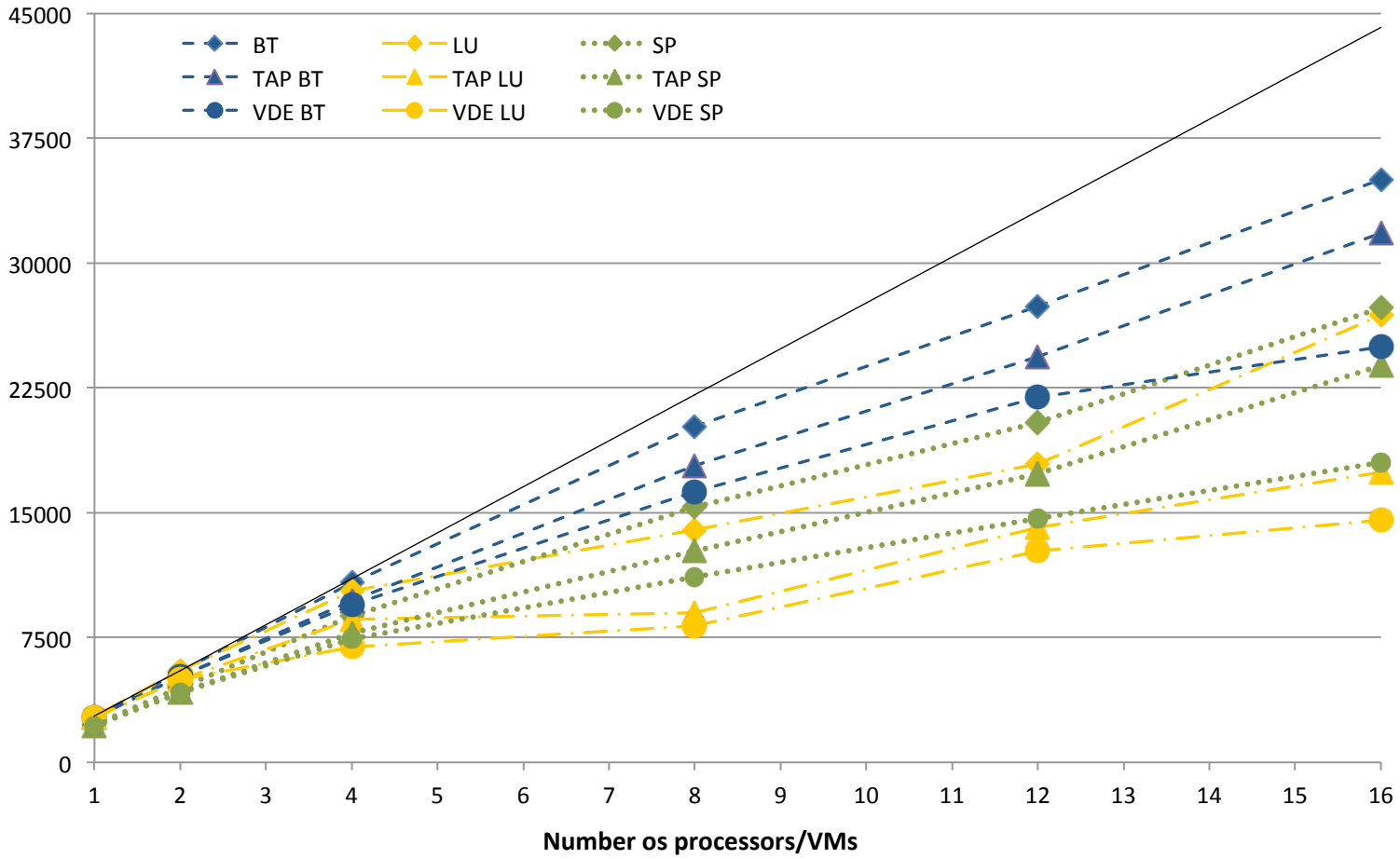
The overhead of the user-mode network overlay was compared to the system level TAP interface.

Note that no permission to use the /dev/kvm device leads to one order of magnitude slower VM performance.

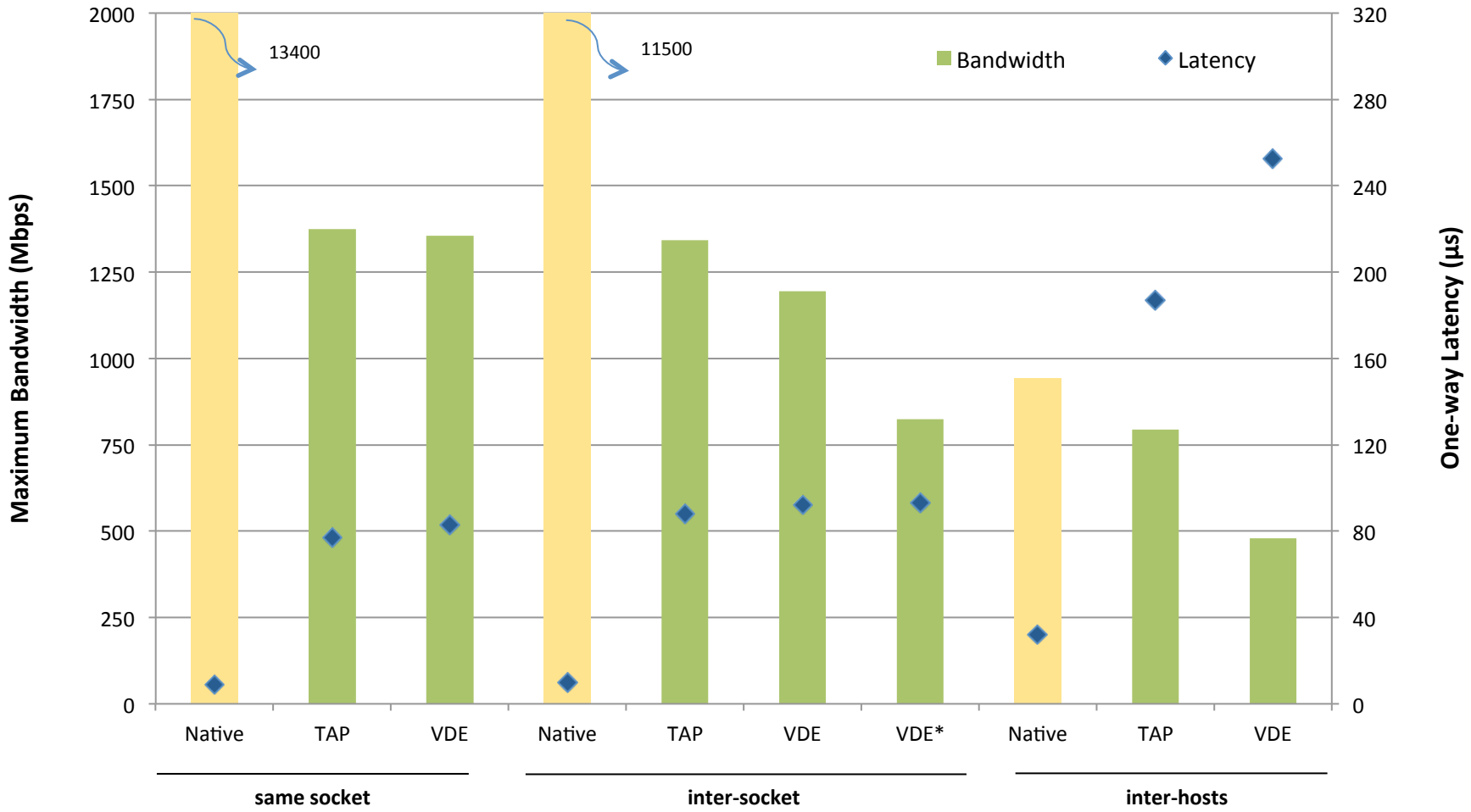
NAS-MZ Performance relative to native



NAS-MZ Efficiency



Raw bandwidth and latency



Conclusions

The system sacrifices performance to provide functionality and its scalability is limited by the overlay network performance;

However, applications where network bandwidth and latency is not critical can achieve near native performance;

User Domains is currently being extended to support running virtual machines in multiple independent clusters at the same time.

The End

Any questions?

Vítor Oliveira

vspo@di.uminho.pt