

A Project Proposal for CS6210 Advanced Operating Systems, Spring 2005

Vaddadi P Chandu (vchandu@cc.gatech.edu)

Exploring task, process, and virtual machine migration in support of single system image for HPC architectures and parallel programming models

1 Motivation

Virtualization presents the illusion of many smaller virtual machines, each running a separate operating system instance on the same machine. Isolation, security, low performance overhead, and, supporting heterogeneous applications, are the four most important requirements of such a virtualized environment [1]. Recently, XEN virtual machine monitor (an open source software) has become a popular resource to implement virtualization for managing multiple operating systems instances within one physical computing node. Current research has focused on migrating these instances between nodes during runtime. This capability is useful for numerous activities, such as fault management, load balancing, and low-level system maintenance. However, currently these efforts are not directed at supporting HPC architectures, SSI [2], or parallel programming models that inject dependencies into the migration (e.g., message passing with MPI, global memory accesses in UPC).

In this project, I wish to examine both the positive and negative aspects of virtualization in the context of high performance computing (HPC architectures, and parallel programming models). In this regard, the architectures of the virtualization monitor, the guest operating systems, the SSI framework, and the parallel programming models, need to be investigated in order to determine what modifications are necessary to be able to perform migration of the operating system instances that make up a SSI and a parallel application executing on that SSI. Next, the level of coordination among these software systems that can enable 1) easy management of applications (e.g., automatic load balancing of application processes), 2) easy management of the HPC architecture (e.g., automatic migration of OS instances away from physical resources that have been predicted to fail in the near future), and 3) reasonable performance need to be investigated.

2 Task List

2.1 Step 1:

Due Date: 8th Mar 2006

- 1) Install most recent Xen on target platforms.
- 2) Experiment and document the basic performance of OS functionality under loaded/unloaded Xen.

2.2 Step 2:

Due Date: 8th Apr 2006

- 1) Install MPI on Xen on target platforms.
- 2) Experiment and document basic performance of MPI functionality under loaded/unloaded Xen.

2.3 Step 3:

Due Date: 26th Apr 2006

- 1) Select a container that encapsulates an MPI task and study the migration behavior of Xen in the context of that container between hardware platforms.

- 2) Experiment and document the following w. r. t. the OS container:
 - a. How Xen handles MPI messages in transit?
 - b. Are there potential deadlock conditions?
 - c. What are the performance issues?

3 Acknowledgements

Support from Dr. Jeffrey Vetter of Oak Ridge National Laboratory is gracefully acknowledged.

4 References

[1] <http://www.cl.cam.ac.uk/Research/SRG/netos/xen/>

[2] <http://www.openssi.org/>