

Testing Virtual Machine Performance Running ATLAS Software

Yushu Yao
Paolo Calafiura
LBNL
April 15, 2008

Objectives

- We run bench tests and common ATLAS tasks on Virtual machines to test their performance and compare them with running them in LXPLUS. Hoping to give an impression to the audiences how well virtual machines work, and to provide some reference for the CERN VM developers.
- Common ATLAS tasks include:
 - Compiling Packages, using CMT, etc.
 - Running jobs (simulation, reconstruction, etc.)
- Virtual Machines include:
 - Windows: Vmware, Qemu
 - MacOS: Vmware, Qemu
- VMWARE is tested since it is the most popular one. FREE player for Linux and Windows, \$60 for Mac.
- QEMU is especially attractive because it is FREE for Linux, Windows and Mac.

Note: For Linux Hosts results and Parallel for Mac results, please refer to later talks of today.

Host Machine

- Tests are done on my laptop, since most users are likely to use virtual machines for development / analysis on their laptops or desktops.
- Hardware: 15" Mac Book Pro
 - CPU: 2.2 GHz Intel Core 2 Duo
 - Memory: 2 GB 677 MHz DDR2
 - HD: TOSHIBA MK2035GSS (200 G)
- Software: Running dual system via rEFIt.
 - Mac OS X - 10.5.2
 - VMWARE Fusion 1.1 / QEMU 0.9.0
 - Windows XP SP2
 - VMWARE Workstation/ QEMU 0.9.0

Virtual (Guest) Machine Setup

- The following cpu and memory allocation are used for the guest machine, to maximum the performance of the guest machine while ensure acceptable performance of the host machine (Let you read emails or surf the web, even running eclipse).
 - Single CPU
 - 1 G Memory
- Virtual machines use disk images as virtual disks. For portability, hard disk image type vmdk is used in this test.

Virtual Machine Setup (cont.)

- /dev/hda
 - Scientific Linux 3
 - Gcc 323
 - Kernel 2.4.21
- /dev/hdb
 - ATLAS Release 13.0.40 (downloaded by pacman)
- /dev/hdc
 - Working Disk

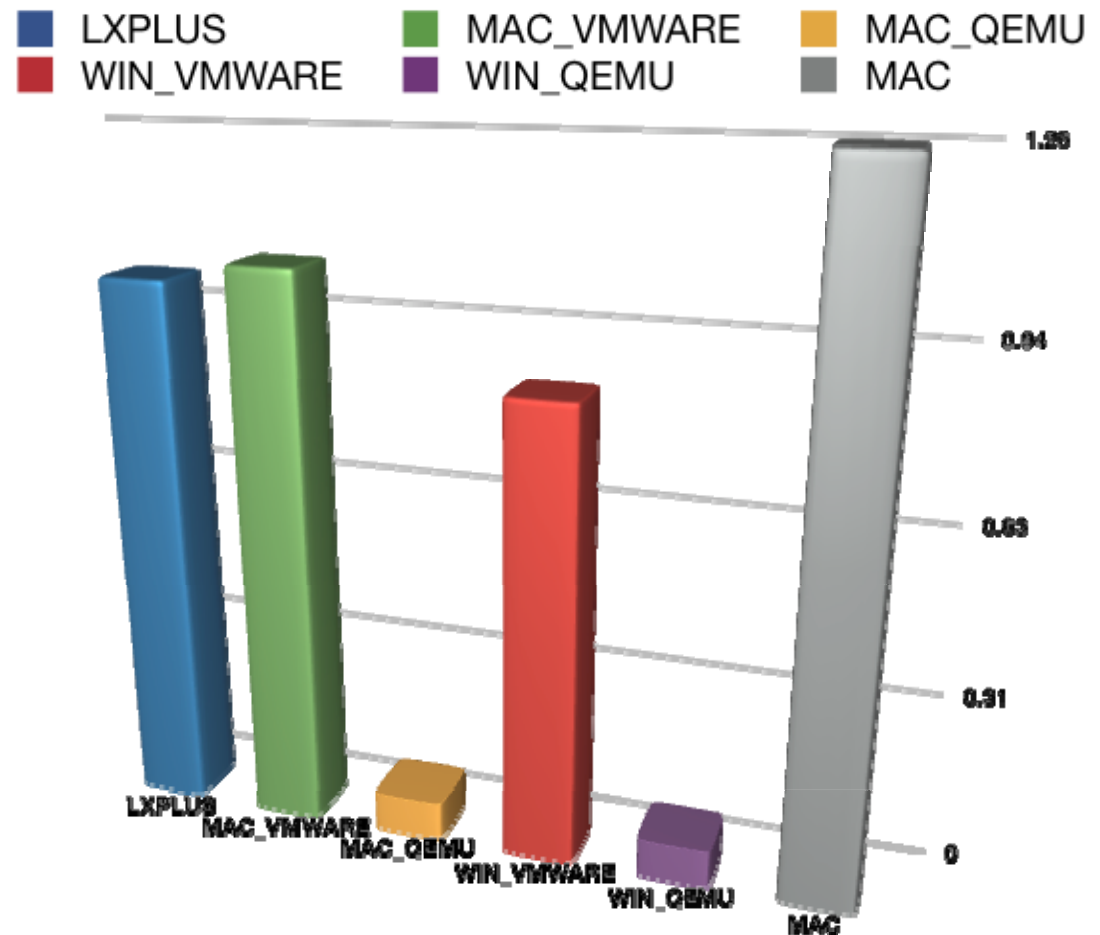
The Tests

- CPU Benchmark - nBench
 - Run CPU/Memory intensive jobs like sorting, FFT, etc.
- ATLAS specific tests
 - Compiling Test
 - Compile four ATLAS packages
 - ATLAS Jobs
 - A combination of event generation, simulation, digitalization, reconstruction and analysis.

For results go to next page->

CPU Performance (nBench)

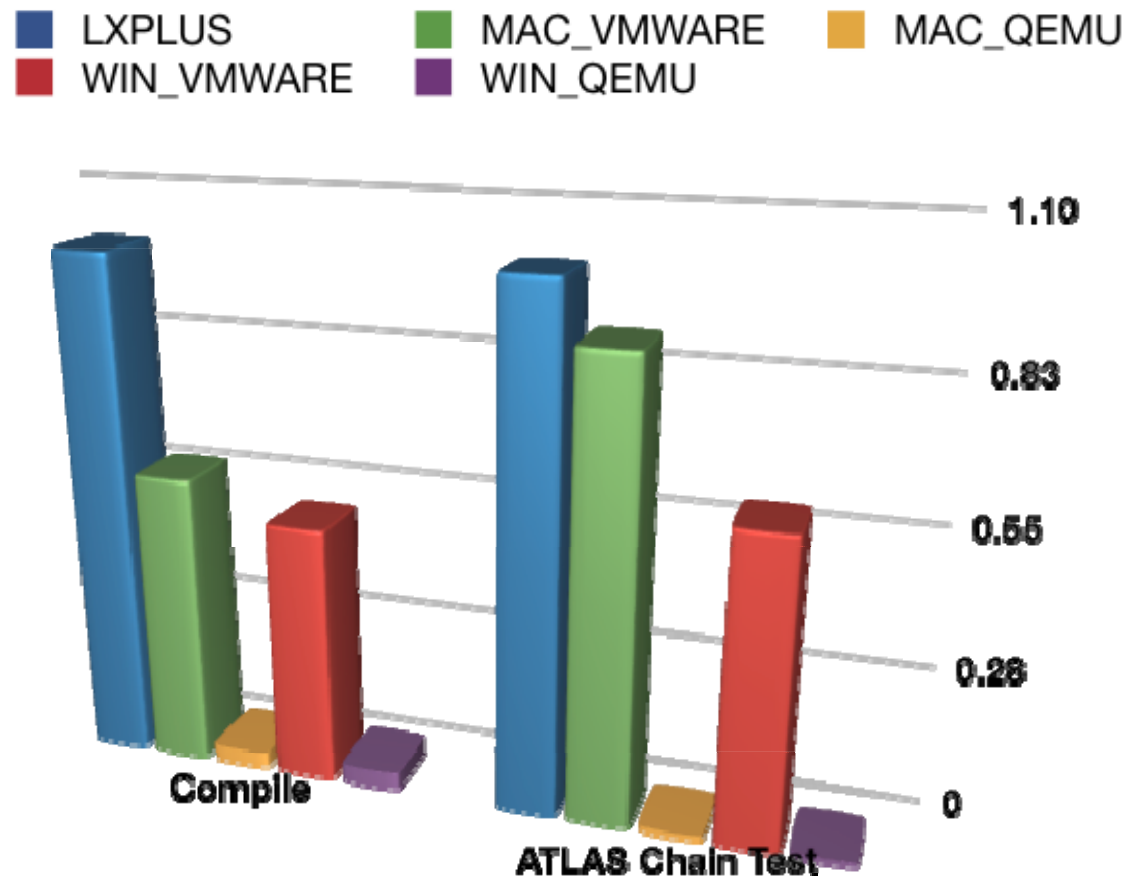
- 20-30% difference between Native Mac and VMWARE. Mac Vmware is slighter better than Windows Vmware.
- QEMU is very slow, unfortunately. (One hope: a kernel acceleration kqemu is available, but not for mac)
- Note that the LXPLUS run is testing a specific LXPLUS node (lxplus212), while all other runs are testing my MacBook Pro.



Results show the ratio = (Average Score Given by nBench in a VM) / (Average Score Given by LXPLUS)

ATLAS Job Performance

- Compiling is about half the speed of LXPLUS in VMWARE.
- The ATLAS full chain tests take 20% more time in Mac Vmware and 45% more time in Windows Vmware.
- Compiling in QEMU is 20 times slower. Chain tests crash in QEMU.



Results show the ratio = (the time to run the test on a VM) / (the time to run the test on LXPLUS)

Summary

- QEMU is slow. We need to figure out why QEMU is slow, and try to find solution for that. If no solution found, we will have to walk away from it.
- For development / analysis, performance of VMWARE is acceptable. Comparing to LXPLUS.

One Possible Way to Distribute ATLAS VM

- Is it better if the user can download the whole atlas software release (e.g. 13.0.40) as a virtual disk? (Instead of using pacman)
- User download the desired OS volumes and Software Release volumes.
- User choose which to use in a User Interface (GUI).
- The GUI plugs the proper volumes into the VM. Using either IDE (/dev/hda, ...) or SCSI (/dev/sda, ...)
- Optional: In case different initialization need to be done for different software releases, the GUI can pass these information to the VM through a “Setup Info” disk (could be virtual floppy, usb, network, etc)

