



A Performance Comparison of Clouds

Amazon EC2 and Ubuntu Enterprise Cloud

Jonathan S Ward

StACC (pronounced like 'stack') is a research collaboration launched in April 2009 focusing on research in the important new area of cloud computing. Unique in the UK, StACC aims to become an international centre of excellence for research and teaching in cloud computing and will provide advice and information to businesses interested in using cloud-based services.

More information available at: www.cs.st-andrews.ac.uk/stacc

A large, abstract graphic composed of overlapping, semi-transparent blue and grey polygons, creating a 3D effect. The text 'SICSA DemoFEST '09' is centered within this graphic.

SICSA DemoFEST '09

Executive Summary

Until recently organisations, seeking to deploy applications on a cloud, had no choice but to deploy applications on a public cloud. The public cloud service is largely dominated by the commercial Amazon Elastic Compute Cloud (EC2), however there are a number of software products designed to allow organisations to create their own private and hybrid clouds. Eucalyptus is the first open source product designed for such a purpose. Ubuntu Enterprise Cloud (UEC), a Linux distribution built around Eucalyptus, is purported as a complete solution for a private or hybrid cloud. This paper describes a series of tests performed in order to compare the performance of EC2 and UEC, and their suitability for meeting the challenges facing users of cloud computing.

EC2 and UEC are compared in terms of memory bandwidth, storage speed and real world application performance. We show that for most computational tasks Ubuntu Enterprise Cloud offers greater performance, despite being let down on application performance by the Kernel Virtual Machine. So too, we show that for cloud storage EC2 is the most advantageous solution, offering high end SAN performance without (as what UEC would require) the user having a dedicated storage infrastructure at their disposal.

1. Introduction

The increasing availability of virtualisation software both commercially and open source has led to the rise of numerous cloud computing services. Most infrastructure as a service software, both open source and commercial, is based around the open source Xen hypervisor. However there exists many other hypervisors around which cloud computing products are built, including VMWare ESX, the Linux Kernel Virtual Machine (KVM) and Microsoft's Hyper-V. Each of these products have substantially different designs, which all (according to the developers) offer near native performance. Amazon EC2 offers the most mature and most widely used infrastructure as a service technology, it is the benchmark for which all other cloud computing technologies much meet in order to be become viable. Eucalyptus (and in turn UEC) is steadily becoming the dominant open source cloud technology, however its viability when compared to established cloud technologies, namely EC2, is as yet unknown. In order to compare performance we tested an EC2 Virtual Machine (VM) and a UEC VM of identical capacity against different criteria: memory bandwidth, storage access speed and application performance. We show that despite several impositions UEC is as capable as EC2 and in numerous tests surpass it. It is the case that while EC2 is designed for running as many VMs per host as possible, UEC is designed for offering more computationally powerful VMs at the expense of running fewer VMs per host.

2. Amazon EC2

The Amazon Elastic Compute Cloud (EC2) is a commercial web service offering infrastructure as a service. EC2 allows customers to run virtual machine instances on top of Amazon's physical infrastructure. A customer runs a virtual machine with an OS and software of their choosing, for as long as necessary, being charged at a small hourly fee. With VM instances costing as little as

\$0.20/hour, EC2 is designed for cost effective on demand computation, as has been used extensively by businesses and education institutions, in place of using traditional on site servers or dedicated hosting. EC2 is custom designed by Amazon and is based around the Open Source Xen Virtual Machine Manager. Xen is somewhat unconventional. Unlike most other virtualisation offerings there is no host OS. Xen boots its own microkernel, on top of this runs the Xen hypervisor. Xen boots a special VM which is granted special privileges to configure the server, this VM is known as Dom0. All other VMs run as DomU and run without privileges to access the physical host. EC2 instances run as the latter type of Xen instance.

Amazon uses Xen to limit the CPU and RAM allocation to each VM instance. In order to run as many VMs as possible on each physical host, Xen allocates an undisclosed number of VMs per CPU. Amazon offers the following standard virtual machine specifications:

- Small Instance (Default): 1.7 GB of memory, 1 EC2 Compute Unit (1 virtual core with 1 EC2 Compute Unit), 160 GB of instance storage, 32-bit platform
- Large Instance: 7.5 GB of memory, 4 EC2 Compute Units (2 virtual cores with 2 EC2 Compute Units each), 850 GB of instance storage, 64-bit platform
- Extra Large Instance: 15 GB of memory, 8 EC2 Compute Units (4 virtual cores with 2 EC2 Compute Units each), 1690 GB of instance storage, 64-bit platform

The description of the processing capacity of an EC2 instance is somewhat ambiguous. Amazon describes the computational power of an EC2 instance in terms of EC2 Compute Units. A EC2 Compute Unit is an arbitrary value which has little relevance to the OS inside the virtual machine. The VM instance sees only the virtual CPU cores it has been allocated and has no concept of a Compute Unit. The Xen hypervisor on which VMs run is responsible for scheduling the instance's CPU time and it is the hypervisor which allocates CPU time based upon the allocated EC2 Compute Units. A single EC2 Compute Unit equates to around 1.0-1.2GHz of CPU capacity on a 2007 model AMD Opteron or Intel Xeon. Thus a small instance runs on a 1.0Ghz x86 processor. Amazon maintains that EC2 Compute Units are homogeneous across the cloud, giving a consistent compute capacity irrelevant of the physical host.

3. Ubuntu Enterprise Cloud

Ubuntu Enterprise Cloud (UEC) is an open source infrastructure, as a service software solution, that is based around Eucalyptus. Eucalyptus is an open source infrastructure for implementing a cloud on top of a cluster, which runs on the Ubuntu Linux distribution. As of version 9.04 of Ubuntu, Eucalyptus has been included in the universe repository and is supported by Canonical, Ubuntu's commercial sponsor. Unlike EC2, UEC is not a service, it allows for individuals or organisations to create their own private or hybrid cloud.

Ubuntu Enterprise Cloud and Eucalyptus make use of libvirt and the KVM hypervisor, a virtual machine manager which is part of the Linux kernel. Unlike EC2, the smallest computational resource that can be allocated to a VM is a single CPU core. The consequence of this is that each physical host can run considerably fewer VMs, however each VM receives 100% of the allocatable resource. Thus

with identical hardware, Eucalyptus instances have greater processor resource as they are bestowed 100% of the CPU's compute time. EC2 instances however will have less compute resource as at most one EC2 instance can use 70% (in the case of the largest allowed instance) of the CPU's available resource. Due to the open nature of UEC, the entire environment is customisable, unlike the closed EC2, the preset VM sizes can be modified to suit the needs of the administrator. Also unlike EC2 which uses Xen, UEC with KVM makes use of the Qemu processor emulator to virtualise devices. The implication of this is, that instances need not to run a modified kernel. Hence unlike EC2, Eucalyptus/KVM can run a guests OS without modification.

4. Testing Methodology

Both EC2 and UEC were tested using the m1.large instance type. This instance on both platforms entails the allocation of 2 CPU cores, 7.5GB RAM and 850 GB instance storage. Both platforms were tested using an Ubuntu 9.04 64-bit OS image created from the JeOS base package. Testing was performed with the open source phoronix test suite and data was collected using a series of in house scripts. Each test was conducted 10 times, the results given are the average of the 10 tests. The UEC test hosts are off the following specification:

- 2 Quad Core Intel Xeon E5420 2.50GHz
- 16GB ECC RAM
- 1TB Storage, Offered as instance storage to VMs
- 2 1000mbps network cards bonded and connected to a 100mbps switch

EC2 hosts use either of the following CPUs:

- Quad Core Intel Xeon E543 2.66Ghz
- AMD Dual Core Opteron 2218 HE 2.60Ghz

5. Memory Bandwidth

Memory bandwidth was tested using *bandwidth* an open source application for calculating memory and cache access speeds. *bandwidth* tests memory using both 64-bit and 128-bit (through the use of SSE2) operations. The memory bandwidth results are consistent across every virtual machine type for each cloud solution. While VM types have different CPU and memory allocations the speed at which memory can be accessed remains constant. Hence the memory bandwidth result is a combination of the physical hosts hardware and the hypervisor.

As the results show, there is a great discrepancy between the memory bandwidth of Intel and AMD EC2 hosts. When reading from memory and from cache, Intel hosts have on average 25% more bandwidth than AMD hosts. When writing to cache, Intel hosts offer around 7% more bandwidth. However when writing to memory AMD hosts are on average 37% faster than Intel hosts. On physical machines of similar specification to Amazon's EC2 hosts, the difference between Intel and AMD hosts was nowhere near as vast, with the largest difference between bandwidths being 18%. Unfortunately due to the proprietary nature of EC2, it is not possible to access the host in order to

deduce the cause of this discrepancy, however the Xen hypervisor would seem to be the most likely cause. Xen has supported AMDV (AMD's virtualisation extension) for a shorter period than Intel's VT-x and there are less contributors to the AMD Xen codebase. In the case of Ubuntu Enterprise Cloud and KVM the results are on average 25% greater than those of EC2. With KVM, memory bandwidth is 98% that of the physical host. It is impossible to say for certain, however EC2 VMs appear to achieve a much lower rate of performance, around 70% of the physical hosts bandwidth.

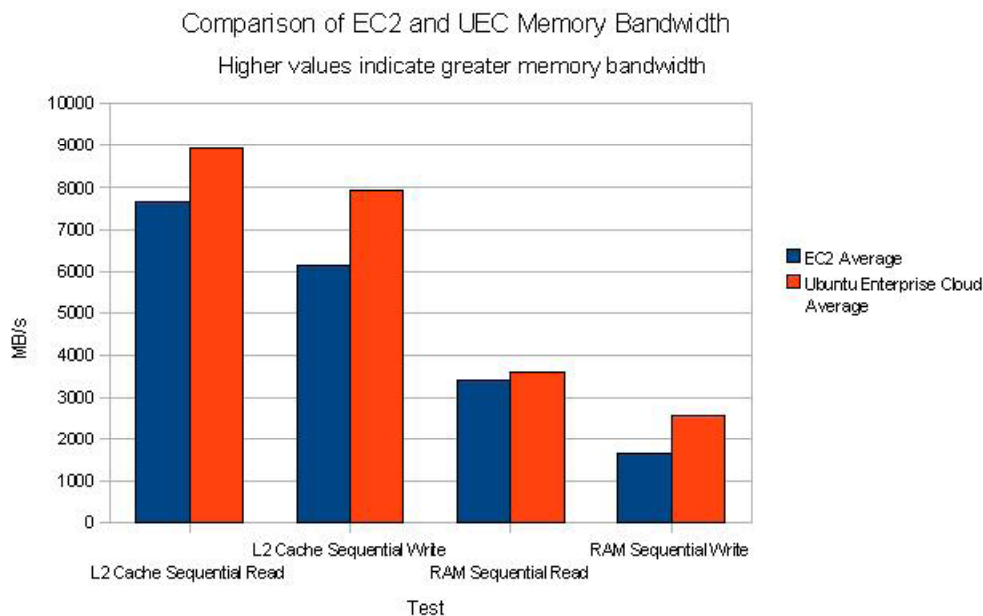


Figure 1: Comparison of Memory Bandwidth

6. Storage Access Speeds

Both UEC and Eucalyptus offer three forms of storage. Ephemeral Instance Storage, Block Storage and S3/Walrus Storage. Each storage is intended for a different purpose. Instance Storage is non transient, designed for instance specific data that does not need to be permanently stored. Block storage is designed for high speed access and storage of large scale data. The Amazon Simple Storage Service (S3) and the API compatible UEC equivalent, Walrus are designed for long term high availability cloud storage. Storage access speeds were tested using *hdparm*. *hdparm* performs synchronous and sequential reads/writes to test each storage medium in order to calculate an average access speed. These benchmarks are incredibly subjective, as (at least in the case of UEC) the results depend almost entirely on the physical storage environment.

S3/Walrus are the slowest storage medium. They are accessed as a web service and not through a dedicated storage protocol, as per block and instance storage. Furthermore S3/Walrus do not have any conventional file structure, inode structure or ACL, making accessing S3/Walrus as per a traditional file system difficult. Indeed S3/Walrus are not intended to be used in such a manner. Compared to S3, Walrus is slow. This is in part due to our test environment which uses a 100mbps at the centre, it is also in part due to Eucalyptus and its management of storage. Walrus is not a mature piece of software, and while feature complete, it is not optimised for fast file access. This is

quite the opposite of S3, which is both accessible over a high speed link and is optimised for fast access.

With EC2, instance storage is hosted on a virtualised network storage system, however instance storage on our UEC test cloud runs from a 1TB RAID array on the VMs physical host. The implication of this is, that while EC2 instances need to access instance storage over the network, UEC instances within our test cloud can access storage locally allowing for much greater speed. EC2 however opts for the network approach as it allows for greater and more scalable storage with greater redundancy at the expense of speed (on a storage medium that does not require it). Block storage is intended to be storage within a SAN or other high availability storage network, with EC2 this is most certainly the case. Our UEC test setup, which serves NFS shares over a 100 mbps link, is a far more economical option. As the results accurately show, our block storage is no match for the speeds offered by Amazon's SAN backed block storage.

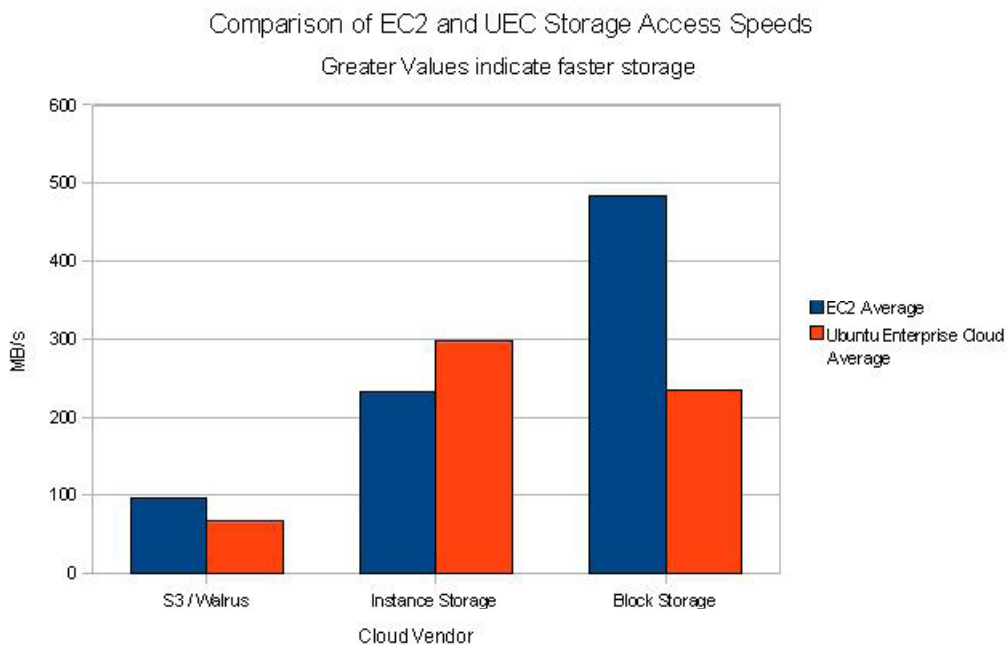


Figure 2: Comparison of Storage Speeds

7. Application Performance

Application performance is arguably the most important aspect of any server platform. Application performance was tested using *gcc*, *sqlite*, *gzip* and *gpg*. These applications test a broad spectrum of functions, computationally intensive and otherwise. The criteria for comparing application performance is speed of execution. Application performance depends on several factors: the application itself, the guest OS, the hypervisor and the physical hardware. In the case of EC2 and Xen the modified kernel, optimised for the virtual environment, should improve performance compared to KVM, which runs an unmodified kernel which offers no performance increase.

The UEC VM, which has 100% access to its two CPUs will be faster than EC2 when running computationally intensive applications. This is certainly the case with the *gcc* test where the

performance of UEC was on average 81% faster than EC2. GCC compilation performance was tested by compiling the Linux kernel source code with the maximum optimisation settings. While the CPUs used by EC2's physical hosts are marginally faster than those in our UEC hosts, the EC2 instances were only allocated 50% of the hosts CPU capacity. The difference in computationally intensive tasks given the difference in CPU allowances is considerable, however such a large margin of difference cannot be solely attributed to CPU allowance. This result is most likely a combination of CPU allowance and Xen performance, as the guest OS and the application are the same in both tests.

Sqlite performance is a different matter. 12500 SQL INSERTS into an sqlite database in instance storage took a mere 9.8 seconds on average on EC2, however with UEC the same test took 107.5 seconds. The speed of EC2 is consistent with machine of similar specification to the physical hosts. Indeed the performance of the UEC hosts was just milliseconds slower than that of the EC2 VM, performing the inserts at an average speed of 10.2 seconds. Thus the poor performance of UEC in this test can be attributed to the KVM hypervisor which would appear not to perform well when dealing with frequent small computation, where as EC2 excels in this test achieving near native performance.

gzip performance was tested by timing the compression of a 2GB file. *gzip* is a lightweight standard compression tool, it has very little memory and CPU overhead. The performance of *gzip* depends primarily on the speed at which RAM and cache can be accessed at. EC2 was 18.8 seconds slower than UEC with an average of 121.4 and 102.6 seconds respectively. The 15% difference can most likely be attributed to EC2s lesser CPU allocation and its lesser memory bandwidths as shown in previous tests.

gnupg is a Free Software implementation of the OpenPGP standard. *gnupg* performance was tested by timing the compression of a 2GB file. *gnupg* performance is dependant upon several factors: CPU speed (for the actual encryption), memory and cache speed (for writing the encrypted file and collecting entropy) and system throughput (for generating entropy from various elements of the system). Both EC2 and UEC performed similarly with a 13% difference in favour of UEC. EC2 on average took 44.9 seconds to complete the encryption whereas UEC took 39.1 seconds. This difference is once again attributable to EC2s lesser CPU allocated and memory bandwidth.

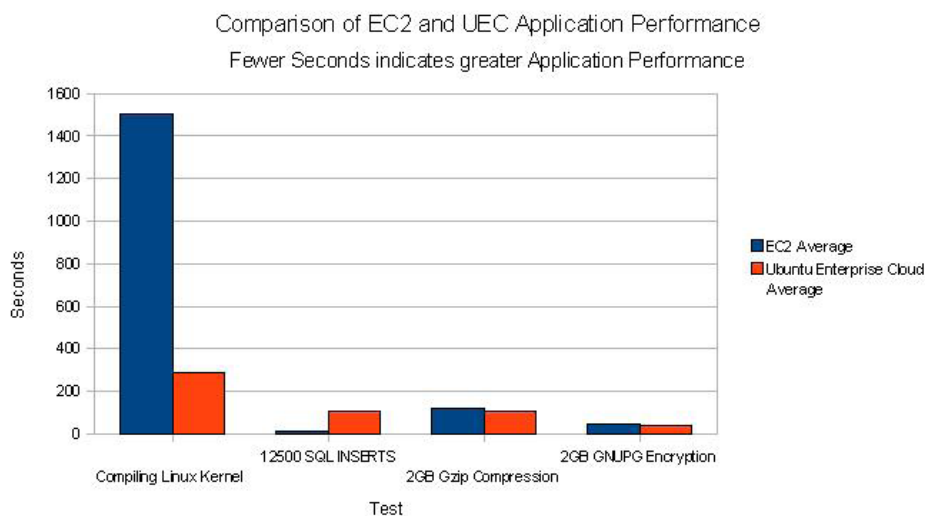


Figure 3: Comparison of Application Performance

8. Conclusion

As the testing shows, UEC is viable as platform for a private cloud, equalling, and in some cases surpassing EC2. EC2 is firmly rooted at the conservative end of the cloud computing spectrum, offering stability and afford-ability at the cost of performance. UEC oppositely has no such reservations and offers more computational resources to the user, at the expense of affordability. UEC offers near native memory performance and in most cases near native application performance. As application testing shows, KVM requires further development in order to offer this performance to all virtualised applications, especially database applications which do not perform well on UEC. Storage performance is where EC2 has the clear advantage. The storage performance of a cloud product is primarily dependant upon the physical storage infrastructure. As testing clearly demonstrated, our test environment could not match the enterprise level storage network used by Amazon. Users of UEC requiring the levels of performance offered by EC2 would require a comprehensive storage infrastructure, which is a substantial barrier for entry. Users seeking a superior computational environment are best suited with using UEC , whereas users seeking an acceptable computational environment supported by an expensive storage system are advised to use EC2. Continual rapid advances in cloud computing are leading to more complex and dependable systems for creating private clouds. UEC, the first complete open source offering is a definite contender, which with further development in database performance could surpass commercial alternatives within the near future.