



Hawk-i HPC CLOUD Benchmark Tool

Visakh Muraleedharan

August 6, 2012

MSc in High Performance Computing

The University of Edinburgh

Year of Presentation: 2012

Abstract

Scientific computing unravel the mysteries of science by constructing mathematical models and numerical algorithms. This required massive computational power and High-Performance Computing (HPC) solutions like cluster and grid have been the answer for massive computing to these needs for a long time. These HPC solutions are hard and expensive to setup, maintain and use. *Cloud computing* is a model of delivering the existing compute infrastructure where computation and storage can be dynamically provisioned on a pay as you go model. Using a vendor cloud service like Amazon Web Service (AWS) can significantly reduce the effort to access these on-demand high performance resources.

In this work we create a platform to study how useful Amazon EC2 cloud computing can be for scientific applications. We classify the applications based on the concept of computational *motif*[2]

Contents

0.1	Introduction	1
1	Background	2
1.1	Cloud computing	2
1.1.1	HPC in cloud	2
1.1.2	Cloud Computing Deployment models	3
1.1.3	Cloud computing service models	3
1.2	Amazon Web Services	5
1.2.1	Amazon EC2	5
1.2.2	Amazon Machine Instance	5
1.2.3	Types of Instances	6
1.3	Parallel Applications and 13 Dwarfs	6
1.3.1	N-Body Methods	6
1.3.2	Spectral Methods	6
2	Live Benchmark Tool	7
2.1	System Design	7
2.1.1	Architecture	7
2.1.2	Sequence diagram	7
2.1.3	Database Design	7
2.2	Cluster Computing Setup	7
2.2.1	Using Starcluster	7
2.2.2	Cluster Management package	7
2.2.3	Using Sun Grid Engine	7
2.2.4	Creating Dashboard	7
3	Benchmarking programs	8
3.0.5	Spectral Methods	8
3.0.6	N-body	8
3.1	Instance types Used	8
3.2	Sun Grid Engine Cluster	8
3.3	SGE Clustering in Amazon cloud	8
3.4	Web interface	8
3.4.1	Admin panel	8
3.4.2	User Dashboard	8

4	Results and Analysis	9
4.1	Serial	9
4.2	Parallel program	9
4.3	Stability of results	9
5	Conclusions	10

List of Tables

List of Figures

Acknowledgements

0.1 Introduction

In this dissertation, we are trying to prove that a live micro benchmark tool can be useful to cloud users to estimate the time and cost of execution before putting significant initial efforts and skills to port the applications to the cloud infrastructure. We also study the possibilities of classifying micro benchmarks based on 13 motifs[] or patterns. Our experiment uses two of these patterns, spectral methods and N-body simulation, to test the concept of live benchmark tool.

Chapter 1

Background

1.1 Cloud computing

In 1960 John McCarthy predicted that "computation may someday be organised as a public utility" [3]. Availability of high capacity networks and low cost hardware as well as the adoption of hardware virtualisation[] technology made this prediction come true. These technologies and delivery models called "cloud computing" enabled a large community of end-users to obtain computing and storage capacity on-demand.

There are several aspects in cloud computing which makes it attractive to users. The most important one is the illusion of infinite computing resources on demand which reduces the user's effort to plan ahead for resource provisioning. Another feature is that cloud computing allow users to start small in a test environment and grow bigger without any upfront commitments to service providers. Finally, from a pricing point of view, provision to pay only for the computing, storage or network resources that you use makes cloud a cost effective solution for users [1].

1.1.1 HPC in cloud

Cloud computing started in context of web applications and enterprise systems that have completely different requirements when compared to high performance computing (HPC) applications like scientific computing. Traditionally these applications rely on HPC centers with dedicated systems connected through high bandwidth interconnects and uses parallel file systems. Cloud provides the ability to manage or modify the software environment and get on-demand access to virtual resources which makes it attractive to HPC users. The cloud resources can be used to replace the existing systems or supplement them. One good example is the cloud storage provided by

Introduction Paragraph, comparison with Clusters Network Limitations, Availability, Stability of performance Benefits

1.1.2 Cloud Computing Deployment models

A cloud being a data center hardware and software can be deployed in several ways depending on several aspects like cost effectiveness, privacy and security. Four deployment models of cloud computing are:

Public Cloud When the cloud is available to the general public in a pay as you go manner, it is called a public cloud. Public cloud infrastructure is owned and maintained by vendors that can support very large infrastructure and hence they are also known as vendor clouds. Amazon Web Services[] which is studied in this project would fall in this category. Some of the other major companies offering cloud services are Rackspace[], Microsoft[] and Google[]. Typically, end-users registered for this service using a credit card and they are provided access to this cloud infrastructure via internet.

Private Cloud A cloud infrastructure operating for a single organisation is a private cloud. This can be managed by the organisation or by a third party. Typically a private cloud is hosted within the organisation network but they can also be hosted externally. To provide virtualisation to users cloud software stacks like OpenNebula[], Eucalyptus[] and Openstack[] are used. Private cloud provides most of the benefits of public clouds while avoiding the issues concerning security of data and performance of public clouds. Even though the organisation have to setup and manage the infrastructure, it can be less expensive than public clouds[?].

Community Cloud When a private cloud infrastructure is deployed by two or more organisations having similar requirements, it becomes a community cloud. The cost of operating the cloud is shared by the organisations in the community. In community cloud, cost need not directly translate to currency in a community cloud. Usage can be credits based or fixed usage based on the agreement between the organisations.

Hybrid Cloud To benefit from the advantages of multiple deployment models, the two or more cloud infrastructures can be bound together to form a hybrid cloud. It requires onsite resources accessible without internet that provides flexibility and remote server based infrastructure to provide scalability. Hybrid clouds can be very useful with the implementing concepts like cloud bursting[]. The software application running on an internal organisational cloud is dynamically ported to a public cloud to address the increase in resource demand. To enable portability of applications in hybrid model, the independent infrastructures should be technology compliant.

1.1.3 Cloud computing service models

Cloud computing services are normally categorized as Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS). high performance computing can significantly utilise each of these models. The difference between the service models is based on the abstraction level at which the service is provided to the end user. The end user have the freedom to customise the applications and environment

above the abstracted level. For example, in IaaS, a virtual hardware is provided to the end user and the user then decides on the operating system and applications to be installed on the system. We will discuss each of these service models with existing examples in the public clouds to understand their features.

Infrastructure as a Service

In Infrastructure as a Service service model, cloud provides virtual equipments including hardware, storage and networking components. The service provider is responsible for creating, running, and maintaining it. In public cloud, the client typically pays on a per-use basis for use of the equipment.

Today, the most widely used IaaS provider is Amazon Web Services. Amazon provides different types of Infrastructures for different pricing. Amazon Simple Storage Service (S3) and Elastic Block Storage (EBS) are methods of data storage. Being a key-based storage which is highly scalable, Amazon S3 is capable of handling data integrity and fault tolerance. Amazon EBS storage provides virtual block storage devices that can easily be associated with elastic compute instances to persistently store data. S3 charges for the amount of data transferred to and from the device and the number of metadata operations performed. Both EBS and S3 charges for data stored per month, but there is no charge for data transferred within an Availability Zone[1]. Data transferred between EC2 instances and S3 located in different Availability Zones in the same region will be charged Regional Data Transfer and data transferred between devices in different regions will be charged on both sides of transfer as Internet Data Transfer.

Eucalyptus, OpenNebula and OpenStack are open source software packages that can be used to create a private cloud Infrastructure service. Most of the services provided by AWS including machine image management, persistent storage etc are provided by these packages. They often provide interfaces similar to vendor clouds like Amazon EC2 which allow the use of same set of tools and methods.

Platform as a Service

Platform as a Service (PaaS) helps in building and delivering applications by providing a computing platform. It includes facilities for different phases of a project lifecycle including design development, deployment and testing. Windows Azure is a good example of a popular PaaS service. Azure provides an environment for development, hosting, and management of applications. Applications hosted in Azure platform are provided compute resources and storage on-demand. Azure has two virtual machine instance types: the webrole instance and the worker role instance. To store data and access it from these virtual machines Azure provides Blob storage Service.

Even though Azure platform is designed for web applications, its use for HPC applications is being explored [4].

Software as a Service

Software as a Service (SaaS) provides end users with direct access to an application with a specific function. Applications and related data are centrally hosted in the cloud and accessed via a web browser.

Examples in the public clouds include services like Google drive[] and Microsoft Sky-drive[]. Documents, images, spreadsheets and presentations can be created and imported using a web interface. Both of these services provide desktop application plugins to synchronise data created using the software suits like Microsoft Office. This gives users the convenience of accessing their application data from any workstation or mobile device.

1.2 Amazon Web Services

In 2006, Amazon.com[] started offering a collection of cloud computing services over the internet. This collection of web services is called Amazon Web Services. Today, Amazon Web Services provides an IT infrastructure platform for business in the cloud with data center locations in the U.S., Europe, Brazil, South America, Singapore, and Japan.

1.2.1 Amazon EC2

Amazon Elastic Compute Cloud (Amazon EC2) is an IaaS that provides resizable or elastic compute capacity in the cloud. It is designed to provide the freedom of choosing development environment and software stack to the developers. Amazon EC2's web service interface allows users to launch and configure capacity with minimal effort. The time required to launch a new instance is less than a few minutes allowing users to quickly increase or decrease capacity, as their computing requirements change.

1.2.2 Amazon Machine Instance

An Amazon Machine Image (AMI) is a pre-configured template operating system which is used to launch an EC2 instance. Users can create their custom AMI using the developer tools. A machine image is compressed and split into a set of 10MB chunks and stored in Amazon S3 storage. Even though each AMI has a pointer to a default kernel, user may choose a kernel from a list of compatible kernels maintained by Amazon and its partners like Redhat and Canonical.

1.2.3 Types of Instances

image

EC2 cluster compute

1.3 Parallel Applications and 13 Dwarfs

Explain the 13 dwarfs and under the two sections explain N-body and FFT

1.3.1 N-Body Methods

MD, About the dwarf and application types optimisation, n^2 and $n \log n$ Communication pattern!!!

1.3.2 Spectral Methods

In scientific computing, a class of techniques used to numerically solve differential equations involving the use of Fast Fourier Transform are called spectral methods. Fourier transform Jean Baptiste Joseph Fourier (1768-1830) first employed what we now call Fourier transforms whilst working on the theory of heat. Linear transform which takes temporal or spatial information and converts into information which lies in the frequency domain. Who would use Fourier Transforms? • Physics • Cosmology (P3M N-body solvers) • Fluid mechanics • Quantum physics • Signal and image processing • Antenna studies • Optics • Numerical analysis • Linear systems analysis • Boundary value problems • Large integer multiplication (Prime finding) • Statistics • Random process modelling • Probability theory. Discrete Fourier Transform The discrete Fourier transform of N complex points f_k is defined as Communication pattern!!!

Chapter 2

Live Benchmark Tool

2.1 System Design

2.1.1 Architecture

2.1.2 Sequence diagram

2.1.3 Database Design

2.2 Cluster Computing Setup

2.2.1 Using Starcluster

2.2.2 Cluster Management package

2.2.3 Using Sun Grid Engine

2.2.4 Creating Dashboard

Chapter 3

Benchmarking programs

3.0.5 Spectral Methods

3.0.6 N-body

3.1 Instance types Used

Refer to graph in Background Why these types?

3.2 Sun Grid Engine Cluster

Why sungrid Engine

3.3 SGE Clustering in Amazon cloud

benchmark to show how cluster instances are faster than normal high cpu instances

3.4 Web interface

Describe in detail

3.4.1 Admin panel

3.4.2 User Dashboard

Chapter 4

Results and Analysis

4.1 Serial

for each instance type time to result, execution time, increasing problem size N-body and FFT

4.2 Parallel program

execution time, increasing problem size, number of cores N-body and FFT

4.3 Stability of results

History of execution, snapshot from dashboard

Chapter 5

Conclusions

Bibliography

- [1] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, et al. A view of cloud computing. *Communications of the ACM*, 53(4):50–58, 2010.
- [2] K. Asanovic, R. Bodik, J. Demmel, T. Keaveny, K. Keutzer, J.D. Kubiawicz, E.A. Lee, N. Morgan, G. Nécua, D.A. Patterson, et al. The parallel computing laboratory at uc berkeley: A research agenda based on the berkeley view. *EECS Department, University of California, Berkeley, Tech. Rep*, 2008.
- [3] M.D. Dikaiakos, D. Katsaros, P. Mehra, G. Pallis, and A. Vakali. Cloud computing: Distributed internet computing for it and scientific research. *Internet Computing, IEEE*, 13(5):10–13, 2009.
- [4] J. Li, M. Humphrey, D. Agarwal, K. Jackson, C. van Ingen, and Y. Ryu. escience in the cloud: A modis satellite data reprojection and reduction pipeline in the windows azure platform. In *Parallel & Distributed Processing (IPDPS), 2010 IEEE International Symposium on*, pages 1–10. IEEE, 2010.