2009 Fifth International Joint Conference on INC, IMS and IDC

A Taxonomy and Survey of Cloud Computing Systems

Bhaskar Prasad Rimal, Eunmi Choi*
School of Business IT, Kookmin University
Jeongneung-Dong, Seongbuk-Gu, Seoul, 136-702, Korea
b.bprimal@gmail.com, emchoi@kookmin.ac.kr

Ian Lumb
Computing and Network Services, York University
4700 Keele Street, Toronto, Ontario, M3J 1P3, Canada
ian@yorku.ca

Abstract- The computational world is becoming very large and complex. Cloud Computing has emerged as a popular computing model to support processing large volumetric data using clusters of commodity computers. According to J.Dean and S. Ghemawat [1], Google currently processes over 20 terabytes of raw web data. It's some fascinating, large-scale processing of data that makes your head spin and appreciate the years of distributed computing fine-tuning applied to today's large problems. The evolution of cloud computing can handle such massive data as per on demand service. Nowadays the computational world is opting for pay-for-use models and Hype and discussion aside, there remains no concrete definition of cloud computing. In this paper, we first develop a comprehensive taxonomy for describing cloud computing architecture. Then we use this taxonomy to survey several existing cloud computing services developed by various projects world-wide such as Google, force.com, Amazon. We use the taxonomy and survey results not only to identify similarities and differences of the architectural approaches of cloud computing, but also to identify areas requiring further research.

Keyword- Cloud Computing, Distributed Computing, Taxonomy, Evolution, Massive Data, Large Scale Processors

I. INTRODUCTION

Is it actually buzz words or hype that's been created by Cloud Computing? But behind the hype, there is really something and cloud computing appears to be a highly disruptive technology, which is gaining momentum. It has inherited the legacy technology and adding new ideas. The concept of cloud computing addresses the next evolutionary step of distributed computing. The goal of this computing model is to make a better use of distributed resources, put them together in order to achieve higher throughput and be able to tackle large scale computation problems. Cloud Computing is not a completely new concept for the development and operation of web applications. It allows for the most cost-effective development of scalable web portals on highly available and fail-safe infrastructures.

In the cloud computing system we have to address different fundamentals like virtualization, scalability, interoperability, quality of service, fail over mechanism and the cloud delivery models (private, public, hybrid) within the context of the taxonomy. The taxonomy of Clouds includes the different participants involved in the cloud along with the attributes and technologies that are coupled to address their needs and the different types of services "*XaaS*" offerings where X is software, hardware, platform, infrastructure, data, business etc.

The taxonomy is more than defining the fundamentals that provides a framework for understanding current cloud computing offerings and suggests what's to come. Till now, there is no standard definition of cloud computing. Our main idea behind this taxonomy is to find out the technical strength, weakness and challenges in current cloud systems and we suggest what should be done in future to strengthen the systems. The emergence of cloud fabrics will enable new insights into challenging engineering, medical and social problems.

Our criteria for defining the taxonomy is based on the core ideas of distributed systems for massive data processing. The criteria focus on cloud architecture, virtualization management, services, fault tolerance and we analyze mechanisms like load balancing, interoperability and scalable data storage. The evolution of cloud computing can handle such massive data as per on demand service. Nowadays the computational world is opting for pay-for-use models and Hype and discussion aside, there remains no concrete definition of cloud computing. In this paper, we try to define taxonomy and survey of "Cloud Computing" based on recent advances from academia and industry as well as our experience. This paper also describes the comparative study of different cloud service provider and their systems.

This paper is organized as follows: Section 2 introduces the Background of study. Section 3 defines the taxonomy of cloud computing. Section 4 describes the comparative study of cloud computing service. Our findings are discussed in Section 5; finally Section 6 concludes the paper.

II. BACKGROUND

Everything is a service (XaaS) [17] like SaaS (Software as a Service), PaaS (Platform as a Service), HaaS (Hardware as a Service), DaaS ([Development, Database, Desktop] as a Service), IaaS (Infrastructure as a Service), BaaS (Business as a Service), FaaS (Framework as a Service), OaaS (Organization as a Service) etc. There are many cloud computing systems like Amazon EC2, Google App Engine (GAE), Microsoft Azure, IBM Blue Cloud, Nimbus, 3Tera etc. There is, however, no standard taxonomy, as everyone trys to define Cloud Computing and its services in their own ways. We feel that there is a need to address standardization issues. The users become newly conscious of the concerns, because their data, applications, and computing resources will no longer be under their control. So the users need such

*corresponding author

978-0-7695-3769-6/09 $26.00 © 2009 IEEE
DOI 10.1109/NCM.2009.218

tools for transparency and mechanisms to monitor the status of their data and control over it. Security and privacy issues are the most important concern. Cloud computing system must adopt the most sophisticated and up-to-date tools and procedures to provide better security and privacy but this issue is more challenging to provide more secure and trusty service for user. Interoperability is another concern for the cross platform, cross application and cross vendor support services etc.

We try to define the taxonomy focusing on the core technical elements like architecture, virtualization, interoperability issues, load balancing issues and storage mechanism etc. The following section covers the taxonomy in details.

III. TAXONOMY OF CLOUD COMPUTING

Several taxonomies of the cloud computing blueprint can be found, but most were created from the perspective of vendors that are part of the landscape and not from the perspective of enterprise IT, the consumers of cloud services and software. Our taxonomy helps academia, developer and researcher to understand the world of cloud computing. We list out the taxonomy and explain here in details.

A. Cloud Architecture

Fig.1 show the layered architecture of cloud computing. Cloud Architecture is the design of software applications that uses internet-accessible on-demand service. Cloud Architectures are underlying on infrastructure which is used only when it is needed that draw the necessary resources on-demand and perform a specific job, then relinquish the unneeded resources and often dispose them after the job is done.

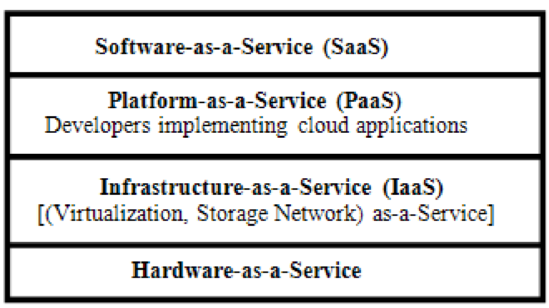

Figure 1. Cloud Layered Architecture

The services are accessible anywhere in the world, with the cloud appearing as a single point of access for all the computing needs of consumers. Cloud Architectures address the key difficulties surrounding large-scale data processing. Some examples of cloud architecture are processing Pipelines, Batch Processing Systems etc. Basically clouds mode can be defined by three types **1) Private Cloud:** Data and processes are managed within the organization without the restrictions of network bandwidth, security exposures and legal requirements that using public cloud services across open, public networks might entail. **2) Public Cloud:** It describes the cloud computing in the traditional mainstream sense, whereby resources are dynamically provisioned on a fine-grained, self-service basis over the Internet, via web applications/web services, from an off-site third-party provider who shares resources. **3) Hybrid Cloud:** The environment is consisting of multiple internal and/or external providers.

B. Virtualization Management

It is the technology that abstracts the coupling between the hardware and operating system. It refers to the abstraction of logical resources away from their underlying physical resources in order to improve agility, flexibility, reduce costs and thus enhance business value. Basically virtualizations in cloud are of different types such as server virtualization, storage virtualization and network virtualization. A common interpretation of server virtualization is the mapping of single physical resources to multiple logical representations or partitions. In a virtualized environment, computing environments can be dynamically created, expanded, shrunk or moved as demand varies. Virtualization [2] is therefore extremely well suited to a dynamic cloud infrastructure, because it provides important advantages in sharing, manageability and isolation.

C. Service

There are different categories of cloud services such as infrastructure, platform, application etc. These services are delivered and consumed in real-time over the Internet. We discuss these services in the broader view.

1) *Software -as- a- Service (SaaS):* Software as a Service is a multi-tenant platform. It uses common resources and a single instance of both the object code of an application as well as the underlying database to support multiple customers simultaneously. SaaS [3] [4] commonly referred to as the Application Service Provider (ASP) model, is heralded by many as the new wave in application software distribution. Examples of the key providers are SalesForce.com (SFDC), NetSuite, Oracle, IBM, and Microsoft etc.
2) *Platform -as- a -Service (PaaS):* It is the big idea to provide developers with a platform including all the systems and environments comprising the end-to-end life cycle of developing, testing, deploying and hosting of sophisticated web applications as a service delivered by a cloud based, a Platform-as-a-Service (PaaS). Key examples are GAE, Microsoft's Azure etc. Compared with conventional application development, this strategy can slash development time, offer hundreds of readily available tools and services, and quickly scale.
3) *Hardware -as- a- Service (HaaS):* According to Nicholas Carr [18], " *the idea of buying IT hardware - or even an entire data center - as a pay-as-you-go subscription service that scales up or down to meet your needs. But as a result of rapid advances in hardware virtualization, IT*

automation, and usage metering and pricing, I think the concept of hardware-as-a-service - let's call it HaaS - may at last be ready for prime time." This model is advantageous to the enterprise users, since they do not need to invest in building and managing data centers.

4) *Infrastructure- as -a -Service (IaaS):* IaaS is the delivery of computer infrastructure as a service. Aside from the higher flexibility, a key benefit of IaaS is the usage-based payment scheme. This allows customers to pay as they grow. Another important advantage is that by always using the latest technology. Customers can achieve a much faster service delivery and time to market. Key examples are GoGrid, Flexiscale, Layered Technologies, Joyent and Mosso/Rackspace etc.

D. Fault Tolerance

In case of failure, there will be a hot backup instance of the application which is ready to take over without disruption is called failover. Cloud computing outages extend into the more refined version of cloud service platforms. Some of the outages were quite lengthy. For example, Microsoft Azure had an outage that lasted 22 hours in March 13th -14th, 2008. Cloud reliance can cause significant problems if the control of downtime and outages is removed from your control. The Table 1 shows failover records from some of the cloud service provider system. These are the significant downtime incidents. Reliance on the cloud can cause real problems when time is money.

TABLE 1. OUTAGES IN DIFFERENT CLOUD SERVICES

Service and Outage	Duration	Date
Microsoft Azure: malfunction in Windows Azure [5]	22 hours	March 13-14, 2008
Gmail and Google Apps Engine [6]	2.5 hours	Feb 24,2009
Google search outage: programming error [7]	40 min	Jan 31, 2009
Gmail: site unavailable due to outage in contacts system [8]	1.5 hours	Aug 11,2008
Google AppEngine partial outage: programming error [9]	5 hours	June 17,2008
S3 outage: authentication service overload leading to unavailability [10]	2 hours	Feb 15,2008
S3 outage: Single bit error leading to gossip protocol blowup. [11]	6-8 hours	July 20,2008
FlexiScale: core network failure [12]	18 hours	Oct 31, 2008

Google has also had numerous difficulties with its Gmail and application services; these difficulties have generated significant interest in both traditional media and the blogosphere owing to deep-seated concerns regarding service reliability. The incidents we described here are just the tip of the iceberg. Every year, thousands of websites struggle with unexpected down-time, and hundreds of networks break or have other issues. So the major problem for cloud computing is how to minimize such kind of outage/failover to provide the reliable services.

E. Security

Usually security is the focal concern in terms of data, infrastructure and virtualization etc. Corporate information is not only a competitive asset, but it often contains information of customers, consumers and employees that, in the wrong hands, could create a civil liability and possibly criminal charges. Cloud computing can be made secure but securing cloud computing data is a contractual issue as well as a technical one.

F. Other Issues

The other issues are related to the mechanisms such as load balancing, interoperability and scalable storage etc. In this section we discuss these issues in details.

1) Load Balancing

Load balancing is often used to implement failover- the continuation of a service after the failure of one or more of its components. The components are monitored continually and when one becomes non-responsive, the load balancer is informed and no longer sends traffic to it. This is inherited feature from grid-based computing for cloud-based platforms. Energy conservation and resource consumption are not always a focal point when discussing cloud computing; however with proper load balancing in place resource consumption can be kept to a minimum. This is not only serves to keep costs low and enterprises "greener", it also puts less stress on the circuits of each individual box, making them potentially last longer. Load balancing also enables other important features such as scalability.

2) Interoperability

The issue of interoperability is needed to allow applications to be ported between clouds, or to use multiple cloud infrastructures before critical business applications are delivered from the cloud. Recently Cloud Computing Interoperability Forum (CCIF) [16] was formed to define an organization that would enable interoperable enterprise-class cloud computing platforms through application integration and stakeholder cooperation.

Most clouds are completely opaque to their users. Most of the time, users are fine with this until there is, e. g., an access issue. In such situations, frustration increases exponentially with time partly because of the opacity. Is a mechanism like network weather map required? In other words, some form of monitoring solution? Such a solution might also allow users, or indeed autonomous agents, to choose between different clouds - assuming high levels of interoperability, etc.

3) Scalable Data Storage

Cloud storage enables client to "throw" data into the cloud and without worrying about how it is stored or backing it up. The main issues related to cloud storage are reliability and security. Clients aren't likely to entrust their data to another company without a guarantee that they'll be able to access their information whenever they want and no one else will be able to get at it. Horizontal scalability is what cloud provides through load balancing and application delivery solutions. Vertical scalability is related to resources used, much like the old mainframe model. If an application doesn't vertically scale well, it's going to increase the costs to run in the cloud. Applications that fail to vertically scale well may end up costing more when deployed in the cloud because of the additional demand on compute resources required as demand increases.

IV. COMPARISION BETWEEN CLOUD SYSTEMS

Even though there has been some comparative research on cloud computing from academia to enterprise perspectives, there remains an absence of a comprehensive technical study. We study cloud computing systems in terms of various classifications such as, infrastructure technology and solutions, PaaS provider, and open source. In this section we discuss the technical comparative study of several projects which are currently devoted world-wide. Table 2 shows the comparative technical studies between different infrastructure technology and solution provider. Table 3 shows the comparative technical studies between different SaaS & PaaS service providers. Similarly Table 4 shows the comparative technical studies between different open source cloud-based services like Eucalyptus, Open Nebula, Nimbus, and Enomaly etc.

V. FINDINGS

Based on the proposed taxonomy, comprehensive technical studies and survey, we notice some of the findings from different cloud computing systems that may help in future for new development and improvement on existing systems. We also discuss here some complication, challenges and opportunities for future cloud computing.

►*Amazon EC2:* In the existing EC2 architecture, users are able to monitor and control their applications as an instance but not as a service. In order to achieve service manageability the following capabilities are required: Application-defined SLAs, such as workload capacity and concurrent computational tasks, dynamically provision additional services to handle additional workload and "Focal Server" approach. AWS is becoming popular as de facto standards; many cloud systems are using this API. EUCALYPTUS is an open-source implementation of the AWS APIs.

►*GigaSpaces:* In-Memory Data-Grid (IMDG) technique is used to manage state data in a database which bridges the bottleneck of scalability. It provides all the basic features of a high-end Data Grid as well as unique features, such as continuous query and seamless integration with external data sources and makes it extremely easy to deploy, modify, and ensure high-availability for applications running on Amazon EC2. GigaSpaces's Space-Based Architecture (SBA) approaches can meet the challenge of running low-latency transactional applications in a highly distributed environment such as Amazon EC2.

►*GAE:* It provides very useful ideas for people or companies to make web applications from scratch without needing to worry about infrastructure. It includes automatic scaling, load balancing. This feature alone will be worth it to a certain class of application developers. GAE has some clear advantages and it lowers the barriers to entry for startups and independent developers. The potential problem is lock-in that creates risks and more cost for long-term. The lock-in is caused by custom APIs such as BigTable, Python launcher, accounts and transparent scaling for both python scripts and database.

►*Sun Cloud:* The agile nature of Sun Cloud provides multiple hardware architectures to customize systems for workload, multi-tenancy and resource sharing amongst a large pool of users allowing centralized infrastructure with lower costs. Sun modular datacenter is flourishing and ten times faster to deploy than a conventional datacenter. Sun's open storage provides unique and business model which provides snapshot, replication and compression without additional cost for data services. Hybrid Cloud architecture is very important. One of the nice mechanisms of it is open storage model that is provided by Sun cloud which is a new and unique business model as well.

►*Open Source Cloud:* The role of open source cloud computing is to build some mechanism around digital identity management [14], and outlines some technological building blocks are needed for controllable trust and identity verification. Open Nebula and Nimbus are technically sound and popular. Enomaly cloud is focusing on the issue of interoperability which is essential for enterprise cloud system. Most of the open source clouds are providing IaaS.

►*Some common issues on all systems:* Virtualization is just a multi-tenancy strategy. Multi-tenant architecture a single instance of the hosted application is capable of servicing all customers (tenants). Not all clouds are using virtualization. Clouds like GAE and SFDC use completely different technologies to create multi-tenancy. From the developer point of view, multi-tenancy is not the main event. The goal of using a platform is to get the right product done as fast as possible. SFDC and GAE are using this technique to go as fast as possible because of the existing functional footprint. The biggest concern of current cloud computing system is auditing of the security controls and mechanism in terms of user level. Recently Amazon's work [13] towards Statement on Auditing Standards No. 70: Service Organizations, Type II (SAS70 type II) certification may be helpful for those concerned with widely varying levels of security competency. In the aggregate they are better than no certifications whatsoever. Some of the important security aspects of cloud-centric computing are secure cloud resource virtualization, security for cloud programming models, binary analysis of software for remote attestation and cloud protection, cloud-centric regulatory compliance issues and mechanisms plus foundations of cloud-centric threat models etc need to be considered for the future cloud work. Sun's open storage provides unique and business model which provides snapshot, replication and compression without additional cost for data services. RightScale provides an effective failover mechanism. Azure, Amazon and GAE have frequent outage problems as we described in Table 1.

TABLE 2. CLOUD COMPUTING INFRASTRUCTURE TECHNOLOGY AND SOLUTION PROVIDER

Feature	Amazon Web Service	GoGrid	Flexiscale	Mosso
Computing Architecture	-Elastic Compute Cloud (EC2) allows uploading XEN virtual machine images to the infrastructure and gives client APIs to instantiate and manages them. -Public Cloud	-Data centre architecture which is designed to deliver a guaranteed QoS level for the exported services. –Autonomically reconfiguring for infrastructure to cater for fluctuations in the demand	-Data centre architecture which is designed to deliver a guaranteed QoS level for the exported services. –Autonomically reconfiguring for infrastructure to cater for fluctuations in the demand -Allows multi-tier architectures through a high-speed internal GigE network - Public Cloud	-Merge the idea of cloud computing with the traditional managed/shared server environment that many web hosts provide - Doesn't provide root access to their severs -Servers are in intelligent clusters, which make us fairly efficient with infrastructure and power usage. - Public Cloud
Virtualization Management	-OS level running on a Xen hypervisor	-Xen hypervisor	-Xen hypervisor	-VMware ESX Server
Service	IaaS, Xen images	IaaS	IaaS	IaaS
Load balancing	-Service will allow users to balance incoming requests and traffic across multiple EC2 instances -Round-robin load balancing, HAProxy	-F5 load balancing, Load balancing algorithm used as Round Robin, Sticky Session , SSL Least Connect ,Source address	-Automatic Equalization of Server Load within Cluster	-Scales with traffic, inherits load-balancing,
Fault tolerance	-System should automatically alert, failover, and re-sync back to the "last known state" as if nothing had failed	-Instantly scalable and reliable file-level backup service	-Full self-service: Start/stop/delete, change memory/CPU/storage/IPs of Virtual Dedicated Servers	-Auto managing
Interoperability	-Support horizontal Interoperability e.g. interoperability among EC2, Eucalyptus etc	-Committed to furthering interoperability	-Releasing some code that will work for interoperability with other platforms to promote standardization	-Open Cloud Manifesto
Storage	-Simple Storage Service (S3) and SimpleDB. SimpleDB provides a semi-structured data store with querying capability.	-Storage is a two-step process i) Connecting each server to Private Network ii) transfer protocols (RSYNC, FTP, SAMBA, SCP) to transfer data to and from Cloud Storage	-Persistent storage based on a fully virtualized high-end SAN/NAS back-end	-Storage is based on Rackspace Mosso's Cloud Files which is reliable, scalable and affordable web-based storage for backing up and archiving all user static content.
Security	-Type II (SAS70 Type II) certification ,firewall, X.509 certificate, SSL-protected API, Access Control List	- Don't provide a guarantee of security	-Customer has own VLAN, Virtual Dedicated Servers with SLA and Tier 1 top-quality storage Backend	-Payment Card Industry Data Security Standard
Programming Framework	-Amazon Machine Image (AMI), Amazon Mapreduce framework	-Uses a REST-like Query interface & GoGrid API supports Java, Python , Ruby	-Flexiscale API support C, C#, C++, Java, PHP, Perl and Ruby	-Currently supports ASP, PHP

TABLE 3. CLOUD COMPUTING PASS AND SAAS PROVIDER

Feature	Google App Engine	GigaSpaces	Azure	RightScale	SunCloud	Force.com
Computing Architecture	-Google's geo-distributed architecture	-Space base architecture & used for mission-critical applications, where the need for extreme performance, reliability and scalability necessitates an alternative to traditional tier-based architectures	-An internet-scale cloud services platform hosted in Microsoft data centers, which provides an OS and a set of developer services that can be used individually or together	-Multi-server clusters -Gives Virtual Private Servers monitoring system is one of the advanced features of the RighScale Dashboard -Dynamic server configuration -Elastic IPs -single worker processing stage	-Solaris OS, and *Zettabyte File System (ZFS)* -clusters of servers and public IP addresses - hybrid cloud - Q-layer enabled for Data Warehouse and Enterprise Resource Planning - Open Dynamic Infrastructure Management Strategy	-Multitenant Architecture with metadata-driven development model -Allowing a single application to serve multiple customers
Virtualization Management	-Multitenant Architecture	-Application-level Virtualization	-Hypervisor (based on Hyper-V)	-Xen hypervisor	-Hypervisor (Sun xVM Server),OS (Solaris Containers), network (Crossbow), storage (COMSTAR, ZFS), and application (GlassFish & Java CAPS Technologies).	-Everyone starts with the same version of the application. The data are stored in a shared database, but each client has access only to its own information
Service	PaaS Runtime interpreter	PaaS	PaaS	PaaS	PaaS	Saas Confined to API
Load balancing	-Automatic scaling and load balancing	-Performed through the GigaSpaces high-performance communication protocol over the EC2 network infrastructure	-Built-in hardware load balancing	-High Availability Proxy load balancing in the cloud	-Hardware balancers may outperform software based balancers	-Load balancing among tenants
Fault tolerance	-Automatically pushed to a number of fault tolerant servers -App Engine Cron Service	-Each node in the AMI cluster automatically discovers the other nodes and the cluster becomes a fault-tolerant cluster	-Containers are used for load balancing and availability: If a failure occurs, SQL Data Services will automatically begin using another replica of the container	-Basic, intermediate & advance Failover Architectures as described in Best Practices for using Elastic IPs (EIP) and Availability Zones	- Resource based scheduling of service request & automatic failover when node fails	-Self management and self tuning
Interoperability	-Interoperability between platforms of different vendors and programming languages	-Heterogeneous environments, allowing for seamless interoperability between the different programming languages	-Interoperable platform can be used to build new applications to run from the cloud or enhance existing applications with cloud-based capabilities	-Interoperability and portability within cloud computing	-Open source philosophy & java principles -Interoperability for large-scale computing resources across multiple clouds -Open Cloud Manifesto	-Application-level integration between different clouds
Storage	-Proprietary database (BigTable distributed-storage)	-In memory data grid object relation mapping support -Support Amazon Security Groups,	-SQL Server Data Services (SSDS) - allows storing binary large objects (blobs) -Security	-Open storage model, MySQL backups are Elastic Block Store (EBS) are saved to S3	-MySQL's row-based replication - routine use of backup with replication	-Force.com database deals in terms of relationship fields

Security	-Google Secure Data Connector - SDC uses RSA/128-bit or higher AES CBC/SHA - SDC uses TLS-based server authentication	built-in SSH tunneling	token service (STS) creates Security Assertion Markup Language token according to rule	-Assign Multiple Security Groups to an Instance	-User-provisioning and meta directory solution -process and user rights management trusted extensions for mandatory Access Control - Control and role-based access management to back-line resources	-SysTrust SAS 70 Type II - Users and security, programmatic and platform security framework
Programming Framework	-MapReduce programming framework that support Python, Java as well as several java related standards such as the Java Servlet API, JDO and JPA	-Non intrusive programming model, single cluster model supports for Spring/Java .NET,C++	-Microsoft .NET	-Ruby Amazon's Simple Queue Service	-Solaris OS, Java, C, C++, FORTRAN, Open API-Public, RESTful, Java, Python, Ruby	-Apex language for database service and supports for .NET, C# Apache Axis (Java and C++)

TABLE 4. OPEN SOURCE BASED CLOUD COMPUTING SERVICES

Feature	Eucalyptus	OpenNebula	Nimbus	Enomaly
Computing Architecture	-Ability to configure multiple clusters, each with private internal network addresses, into a single Cloud. -Private Cloud	-Cluster into an IaaS cloud -Focused on the efficient, dynamic and scalable management of VMs within datacenters (private clouds) involving a large amount of virtual and physical servers -based on Haizea scheduling	- Science cloud -Client-side cloud-computing interface to Globus-enabled TeraPort cluster - Nimbus Context Broker that combines several deployed virtual machines into "turnkey" virtual clusters - Heterogeneous clusters of auto-configuring VMs with one command	-A clustered virtual server hosting platform; ElasticDrive, a distributed remote storage system; and GeoStratus, a private content delivery network. -Uses a clustered file-system called GlusterFS and is capable of scaling to several petabytes. It aggregates various storage bricks over Infiniband, RDMA or TCP/IP interconnects into one large parallel network file system -Private Cloud
Virtualization Management	-Xen hypervisor	-Xen, KVM and on-demand access to Amazon EC2	-Xen Virtualization	-Kernel-based virtual machine (KVM) supports Xen OpenVZ and Sun's VirtualBox
Service	IaaS	IaaS	IaaS	IaaS
Load balancing	-Simple load-balancing cloud controller	-Nginx Server configured as load balancer**,** used round-robin or weighted selection mechanism	-Launches self-configuring virtual clusters i.e. the context broker	-Uses user-mode load-balancing software with its own network stacks that runs over Linux and Solaris in the form of a virtual server -supports different load balancing methods, including round robin, random, hash, and least resource
Fault tolerance	-Separate cluster within the Eucalyptus cloud reduce the chance of correlated failure	-The daemon can be restarted and all the running VMs recovered -Persistent database backend to store host and VM information	-Checking worker nodes periodically and recovery	-Overflow, disaster and failover services
Interoperability	-Multiple cloud computing interfaces using the same "back end" infrastructure	-Interoperable between intra cloud services	-Standards: "rough consensus and working code"	-Cloud portability and interoperability to cross cloud vendors
Storage	-Walrus (the front end for the storage subsystem)	-Database, persistent storage for ONE data structures -SQLite3 backend is the core component of the	-GridFTP and SCP	-Multiple remote cloud storage services (S3, Nirvanix and CloudFS)

		OpenNebula internal data structures		
Security	-WS-security for authentication, Cloud controller generates the public/private key	-Firewall, Virtual Private Network Tunnel	-PKI credential required -Works with Grid proxies VOMS, Shibboleth (via GridShib), custom PDPs	-Clustered handling of security
Programming Framework	-Hibernate, Axis2 and Axis2c, Java	-Java, Ruby	Python, Java	-Ruby on rails, PHP, Python

problems as we described in Table 1. In the given set of hardware how do to decided whether to put a Grid like infrastructure or a cloud like infrastructure? What will be the determining factors for user to select grids vs. cloud?

These issues are still in discussion. In the current cloud systems, user can't find the status of their data may be someone is using these data for his/her own purpose. Generally users don't know data center politics. If enterprise vendors do not open this secret one day we can predict that there will be data war. Clients want to know 'what is Google's secret? What is Microsoft's secret? What is going inside Amazon?' Users want to monitor their data and control over it, so interoperability is one of the challenging issues for future cloud systems. Most of the current cloud computing systems are silent on these issues. There is no good way to translate Service Level Agreements (SLAs) in cloud allocation chain. It is important because customer consider SLAs as their first priority. Both service providers and clients of cloud computing need to assess the technology and business model that drives the adoption of cloud computing. There is a need for further progress on several issues such as standards, portability, mappings to Business Architecture, Security and privacy, multi-supplier and hybrid sourcing, management and governance plus Business analytics for cloud etc. If the service goes down for an hour, users are hit with the crises and users are handicapped. There are still many issues like this so cloud is still creating hype.

CONCLUSION

Cloud Computing is the promising paradigm for delivering IT services as computing utilities. Clouds are designed to provide services to external users; providers need to be compensated for sharing their resources and capabilities. "*Above the Clouds: A Berkeley View of Cloud Computing*" [15] also gives the future research direction for academia. There are many open issues regarding the cloud computing. This proposed taxonomy will provide researcher and developer the ideas on the current cloud systems, hype and challenges. This paper provides the information to evaluate and improve the existing and new cloud systems.

ACKNOWLEDGMENT

"This research was supported by the MKE (Ministry of Knowledge and Economy), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute for Information Technology Advancement)" (IITA-2009-C1090-0902-0026)

REFERENCES

[1] J.Dean and S.Ghemawat, "MapReduce: simplified data processing on large clusters", Communications of the ACM, January 2008/vol 51, No.1, Pages 107-113

[2] Sun Microsystems, "Virtualization for Dummies"

[3] Software & Information Industry Association, "Software as a Service: Strategic Backgrounder", February 2001

[4] V.Choudhary, Software as a Service: Implications for Investment in Software Development", Proceedings of the 40th Hawaii International Conference on System Sciences

[5] N.Kolakowski, "Microsoft's Cloud Azure Service Suffers Outage", http://www.eweekeurope.co.uk/news/microsoft-s-cloud-azure-service-suffers-outage-396

[6] A.Cruz, Gmail Site Reliability Manager, "Update in Gmail" http://googleblog.blogspot.com/2009/02/update-on-gmail.html

[7] M.Mayer, VP, Search Products & User Experience, "This site may harm your computer on every search results" http://googleblog.blogspot.com/2009/01/this-site-may-harm-your-computer-on.html

[8] T.Jackson, Gmail Product Manager. "We feel your pain, and we're sorry",http://gmailblog.blogspot.com/2008/08/we-feel-your-pain-and-were-sorry.html

[9] Pete, App Engine Team, "App Engine outage today", Available from: http://groups.google.com/group/google-appengine/browse_thread/thread/f7ce559b3b8b303b

[10] Allen Stern Update from Amazon Regarding Friday's S3 Downtime. http://www.centernetworks.com/amazon-s3-downtime-update

[11] AWS Service Health Dashboard .Amazon S3 Availability Event: July 20, 2008 [online]. July 2008. Available from: http: //status.aws.amazon.com/s3-20080720.html

[12] A.Tubanos, theWHIR.com, "FlexiScale Suffers 18-Hour Outage",http://www.thewhir.com/web-hosting-ews/103108_FlexiScale_Suffers_18_Hour_Outage

[13] Amazon Web Services (AWS),"Amazon Web Services: Overview of Security Processes, September 2008

[14] A.Cavoukian, "Privacy in the Clouds: Privacy and Digital Identity - Implications for the Internet", Information and privacy commissioner of Ontario, May 28, 2008

[15] M.Armbrust, A.Fox et al "Above the Clouds: A Berkeley View of Cloud Computing", EECS Department, University of California, Berkeley, Technical Report No. UCB/EECS-2009-28, February 10, 2009

[16] The Cloud Computing Interoperability Forum (CCIF). Available from: http://www.cloudforum.org/

[17] Gathering Clouds of XaaS! http://www.ibm.com/developer

[18] http://www.roughtype.com