CLOUD COMPUTING: A COST EFFECTIVE APPROACH TO ENTERPRISE WEB APPLICATION IMPLEMENTATION: A CASE FOR CLOUD ERP WEB MODEL

Christiana. C. Okezie
Electronic and Computer Engineering Department, Nnamdi Azikiwe University, NIGERIA.
christianaobioma@yahoo.com

Udeze Chidiebele. C
R & D Department, Electronics Development Institute, Awka, NIGERIA.
udezechidi@yahoo.com

Okafor Kennedy .C
R & D Department, Electronics Development Institute, Awka, NIGERIA.
arissyncline@yahoo.com

ABSTRACT

This paper presents cloud computing in a simplified context and develops an efficient framework for enterprise web application integration in organizational models using private cloud offering. From software engineering perspective, software development life cycle (SDLC) was used to derive a process model behavior of the proposed ERP system. A comparison was made among five cloud vendors based on standard metrics that justifies our selection. This work provides the architecture for creating Clouds with market-oriented resource allocation by leveraging on technologies such as Virtualization in Google App Engine Platform. This work proposes Java, Java Eclipse 3.6 with goggle plug-in Java SDK 1.5.5 and goggle web toolkit SDK 2.4.0 for its development and implementation while the ERP system integration is done on the Google App Engine cloud platform (platform as a service-PaaS) for high performance content delivery ERP system. The cost analysis presented in this work shows that cloud offering is a better alternative for enterprise web applications.

Keywords: Software Engineering, SDLC, Virtual Machines, ERP, Integration, VMs, SLA, HPC, PaaS.

INTRODUCTION

In these days of complex business intelligence, enterprise web applications now serves as core business model that manages full-fledged collaborative workflow-based applications, spanning multiple individuals and organizations. An Enterprise cloud computing is a controlled, internal place that offers the rapid and flexible provisioning of compute power, storage, software, and security services (Leena and Sushil, 2010). Cloud enables enterprises to unleash their potential for innovation through greater intelligence, creativity, flexibility and efficiency, all at reduced cost. Today, cloud computing gives businesses more control and flexibility over the technology they deploy and the way they deploy it. It helps organizations reduce costs and focus resources on gaining strategic advantage. While deployment strategies differ, it is critical that an organization's infrastructure is managed as a utility made up of secure, scalable and standards-based building blocks of integrated IT resources from storage to servers and network management tools (Leena and Sushil, 2010).

According to (D. Nurmi et al.,), Cloud computing systems provide a wide variety of interfaces and abstractions ranging from the ability to dynamically provision entire virtual machines (i.e., Infrastructure-as-a-Service systems such as Amazon EC2 and others [3, 4, 5, 6, 7]) to flexible access to hosted software services (i.e. Software-as-a-Service systems such as salesforce.com and others [8, 9, 10, 11]). All, however, share the opinion that delivered IT resources should be clearly defined, provide reasonably deterministic performance, and can be allocated and de-allocated on demand, hence need for cloud integration for our conceptual ERP web model.

We have focused our research on an implementation framework of the cloud computing system since this paper provides a solid foundation on the application-level development methodology.

In this work, we present Google App Engine (GAE): a scalable and high performance cloud-computing application framework that uses computational and storage infrastructure commonly available to enterprise web developers to provide a platform that is modular and open to experimental

ISSN-L: 2223-9553, ISSN: 2223-9944
Vol. 3, No. 1, July 2012

instrumentation. With GAE, we intend to implement an enterprise resource planning (ERP) web model in cloud infrastructure while providing a common framework for system integration. Here, we address several crucial cloud computing questions, including VM instance scheduling, VM, HPC and user data storage, cloud computing administrative interfaces, construction of virtual networks, definition and execution of service level agreements (cloud/user and cloud/cloud), and cloud computing user interfaces for web services. In this work, we will discuss each of these topics in more detail and provide a description of our own initial implementations of the ERP model within the GAE software framework by adopting a SDLC methodology.

CONTEXT AND REVIEW OF RELATED LITERATURE

The key considerations by enterprises to migrate towards Cloud computing was discussed in [2], while [12] outlined the Characteristics of Cloud computing which includes:

High Scalability

Cloud environments enable servicing of business requirements for larger audiences, through high scalability

Agility

The cloud works in the 'distributed mode' environment. It shares resources among users and tasks, while improving efficiency and agility (responsiveness).

High Availability and Reliability

Availability of servers is high and more reliable as the chances of infrastructure failure are minimal.

Multi-Sharing

With the cloud working in a distributed and shared mode, multiple users and applications can work more efficiently with cost reductions by sharing common infrastructure.

Services in Pay-Per-Use Mode

a. SLAs between the provider and the user must be defined when offering services in pay per use mode. This may be based on the complexity of services offered
b. Application Programming Interfaces (APIs) may be offered to the users so they can access services on the cloud by using these APIs

Support for All Service Oriented Applications

The authors in (Rajkumar Buyya et al, 2009) present cloud computing as the 5th utility after water, electricity, gas and telephony. The work argues that with cloud infrastructure, businesses and users are able to access applications from anywhere in the world on demand. Thus, the computing world is rapidly transforming towards developing software for millions to consume as a service, rather than to run on their individual computers.

At present, it is common to access content across the Internet independently without reference to the underlying hosting infrastructure just like the power grid supply. This infrastructure consists of data centers that are monitored and maintained around the clock by content providers. Cloud computing is an extension of this paradigm wherein the capabilities of business applications are exposed as sophisticated services that can be accessed over a network (Rajkumar Buyya et al, 2009). Cloud service providers are incentivized by the profits to be made by charging consumers for accessing these services. Consumers, such as enterprises, are attracted by the opportunity for reducing or eliminating costs associated with ``in-house" provision of these services. However, since cloud applications may be crucial to the core business operations of the consumers, it is essential that the consumers have guarantees from providers on service delivery. Typically, these are provided through Service Level Agreements (SLAs) brokered between the providers and consumers. Other computing paradigm in literature is shown in figure 1.

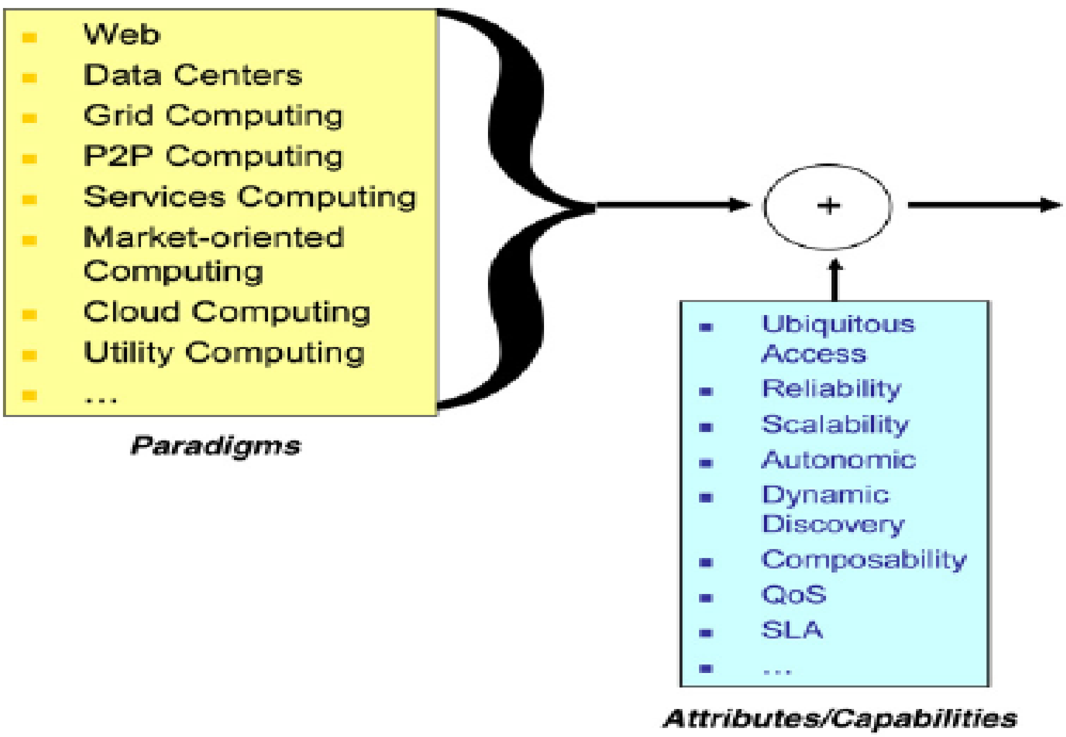

Figure1: Computing paradigms promising to deliver IT as services, [12]

Table 1: Differences between Convectional and Cloud computing

Conventional Computing	Cloud Computing
Applications i. Client-side apps ii. Client/server apps iii. Web interface to local server app iv. Data and process reside on PC or on local server	**End User cloud services** i. Rich internet application ii. Web 2.0 technologies iii. Software-as-a-service iv. Data and process resides at service provider
Developer tools and techniques i. Client-side development tool ii. Service-oriented architecture(SOA) iii. Composite application iv. Proprietary APIs, Such win 32	**App-components-as a- services** i. Internet-hosted software services that enable mashups ii. Web-hosted development tools iii. Community development tools for shared templates and code iv. Proprietary service provider APIs and schema
Middleware i. App server ii. File and object stores iii. Database iv. Integration server	**Software-platform – as-service** i. Hosted app platform ii. Hosted data, file, and object stores iii. Hosted database iv. Software-integration as-a-service
Physical infrastructure i. Server ii. Disks iii. Networks iv. System management	**Virtual-infrastructure as- a-service** i. Virtual server ii. Storage shares iii. Virtual LAN Configurations iv. Management-as-a service

Cloud computing is an online form of computing (Web 2.0 in fact) where users can access applications via a browser, while the application is installed and stored (as well as the Data) on a server ((Leena and Sushil, 2010). This is a whole new form of computing and is allowing thousands of

ISSN-L: 2223-9553, ISSN: 2223-9944
Vol. 3, No. 1, July 2012

users from all around the world to access something without having to download and install anything on their own computers. An example of this would be Google Docs. Though it may not sound safe, there are many benefits for cloud computing ((Leena and Sushil, 2010). In fact, many companies use it, including Amazon, Yahoo, Google, Zoho, Microsoft and Salesforce. Due to reason outline in [2], [12], [13], the Public sector and government IT organizations will eventually end up with cloud computing.

Cloud Computing Metrics & Vendor Comparison

With detailed study on existing literatures, this work defines cloud computing as a technology paradigm shift involving the provisioning of IT resources based on demand via the internet back bone. While there are many benefits of adopting the infrastructure, platforms & services offered by a Cloud Service provider, the applicability of these would depend on the nature & size of an enterprise [13]. With an ever-growing list of cloud computing service providers, the decision for enterprises on how far to leverage computing platforms and with whom is a complex decision to consider since each of the Cloud providers have their own set of pricing, billing, flexibility, support and other important parameters in their model of computing the service. The key considerations dealt with in [13] are from the perspective of Providers and vendors of Cloud Computing Services. In this work, a brief comparison is made between five cloud vendors as shown in Table 2.

From table 2, Amazon EC2 charges the user for the time when the instance of the cloud is alive, while Amazon S3 charges for any data transfer (both upload and download). Google App Engine platform allows a user to run Web applications written using the Java or Python programming language since it supports both run times. Google App Engine also supports Application Programming Interfaces (APIs) for the data store, Google Accounts, URL fetch, image manipulation, and email services. Google App Engine also provides a Web-based Administration Console for the user to easily manage his running Web applications. Currently, Google App Engine is free to use with up to 500MB of storage and about 5 million page views per month [14].

Microsoft Live Mesh [15] aims to provide a centralized location for a user to store applications and data that can be accessed across required devices (such as computers and mobile phones) from anywhere in the world. The user is able to access the uploaded applications and data through a Web-based Live Desktop or his own devices with Live Mesh software installed. Each user's Live Mesh is password-protected and authenticated via his Windows Live Login, while all file transfers are protected using Secure Socket Layers (SSL). Sun network.com (Sun Grid) [16] enables the user to run Solaris OS, Java, C, C++, and FORTRAN based applications. First, the user has to build and debug his applications and runtime scripts in a local development environment that is configured to be similar to that on the Sun Grid. Then, he needs to create a bundled zip archive (containing all the related scripts, libraries, executable binaries and input data) and upload it to SunGrid. Finally, he can execute and monitor the application using the Sun Grid Web portal or API. GRIDS Lab Aneka [17], which is being commercialized through Manjrasoft, is a .NET-based service-oriented platform for constructing enterprise Grids. It is designed to support multiple application models, persistence and security solutions, and communication protocols such that the preferred selection can be changed at any time without affecting an existing Aneka ecosystem. To create an enterprise Grid, the service provider only needs to start an instance of the configurable Aneka container hosting required services on each selected desktop computer. The purpose of the Aneka container is to initialize services and acts as a single point for interaction with the rest of the enterprise Grid. Aneka provides SLA support such that the user can specify QoS requirements such as deadline (maximum time period which the application needs to be completed in) and budget (maximum cost that the user is willing to pay for meeting the deadline). The user can access the Aneka Enterprise Grid remotely through the Gridbus broker. The Gridbus broker also enables the user to negotiate and agree upon the QoS requirements to be provided by the service provider [17], [18].

From our study, based on the combined benefits of runtime interface, virtualization, platform and user interface, this work proposes the Google App Engine as our platform for the ERP web model integration for a private cloud.

Table 2: Cloud Vendor Comparison

System Property	**Amazon Elastic Compute Cloud (EC2)**	**Google App Engine**	**Microsoft Live Mesh Windows Azure**	**Sun Network.com (Sun Grid**	**GRIDS Lab Aneka**
Type of Service	Infrastructure, Platform	Platform	Infrastructure, Platform	Infrastructure	Software Platform for Enterprise Clouds
Service Type	Compute, Storage (Amazon S3)	Web Application	Storage	Compute	Compute
Platforms Supported	**Operating systems** Red Hat Enterprise Linux Windows Server 2003/2008 Oracle Enterprise Linux Open Solaris Open SUSE Linux Ubuntu Linux Fedora Gentoo Linux Debian Software IBM DB2 IBM Informix Dynamic Server Microsoft SQL Server Standard 2005	**Runtime** Java Runtime Environment Python Runtime Environment **Features** Integration with Google Accounts URL Fetch Mail Memcache Image Manipulation Scheduled Tasks and Task Queues XMPP Blobstore (which supports objects upto 50MB in size	**Operating systems** Windows 7 Windows Server 2008 Windows Vista		**Operating systems** Windows server 2008 Windows server 2003 CentOS 5.1 CentOS 5.3 Redhat Linux 5.1 Redhat Linux 5.4
User Access Interface	Amazon EC2 Command-line Tools	Web-based Administration Console	Web-based Live Desktop and any devices with Live Mesh installed	Job submission scripts, Sun Grid Web portal	Workbench, Web-based Portal
Web APIs	Yes	Yes	Unknown	Yes	Yes
Programming Framework	Customizable Linux-based Amazon Machine Image (AMI)	Python Java	Not applicable	Solaris OS, Java, C, C++, FORTRAN	APIs supporting different programming models in C# and other .Net supported lang.

Virtualization in Cloud Domain

In the context of cloud computing, scalability is a key consideration, and the major technology that makes that possible is virtualization. In this paper, we define virtualization as the mirror imaging of one or more workstations/servers within a single physical computer utilizing the same system resources. Cloud computing servers use the same operating systems, enterprise and web applications

ISSN-L: 2223-9553, ISSN: 2223-9944
Vol. 3, No. 1, July 2012

as localized virtual machines and physical servers. Contemporarily, Modern data centers (cloud) are comprised of tens of thousands of servers, and perform the processing for many enterprise as well as Internet business applications, such as those used by financial institutions, Google, twitter, Facebook etc. However, the cost of deployment as well as maintaining the IT infrastructure in data center networks is very immense and hence calls for a better approach to cost reduction as well as service availability [19]. Figure 2a shows a traditional server concept lacking virtualization as discussed in [20], [21].

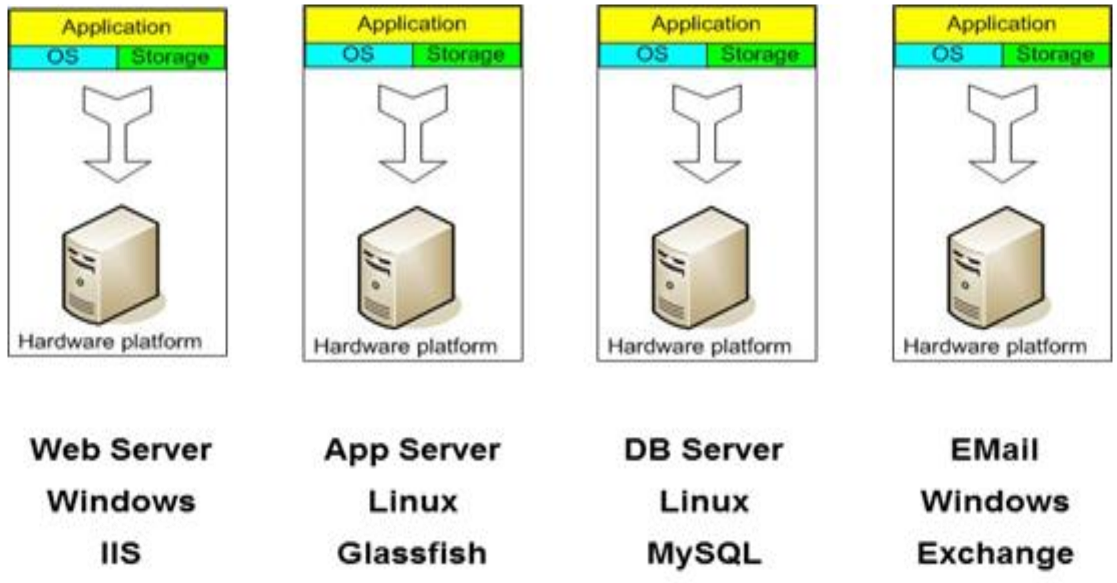

Figure 2a: A traditional server concept with no Virtualization only

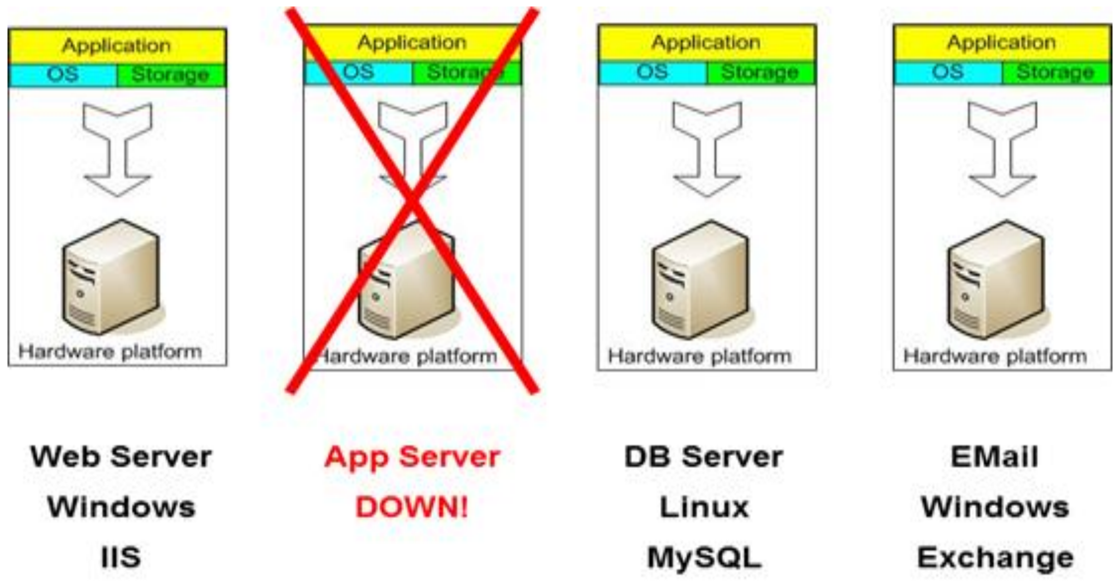

Figure 2b: A traditional server concept with no Virtualization and App Server Failure

The challenges with Figure 2a and 2b includes:

a. Expensive to acquire and maintain hardware (Servers)
b. Poor scalability
c. Difficult to replicate servers and services
d. Redundancy is difficult to implement
e. Vulnerable to hardware outages
f. In many cases, there is under-utilization of server processors

ISSN-L: 2223-9553, ISSN: 2223-9944
Vol. 3, No. 1, July 2012

These challenges results to huge capital expenditure (about 68% IT budget) for organizational models. Figure 3a and figure 3b shows Virtual Infrastructures in data center domains [20], [21].

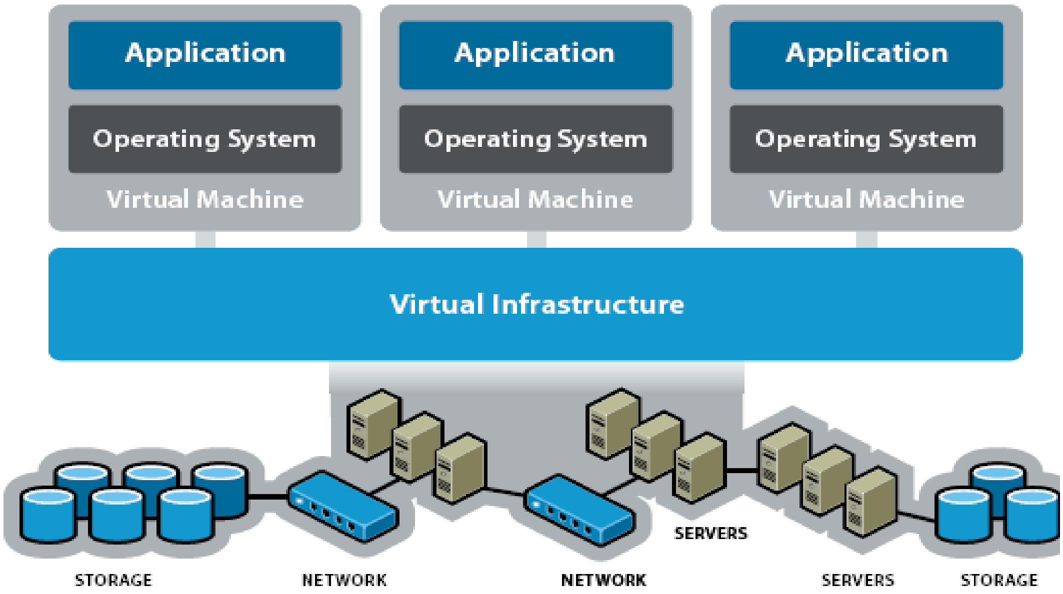

Figure 3a: A Model of VMware Virtual Infrastructure [20]

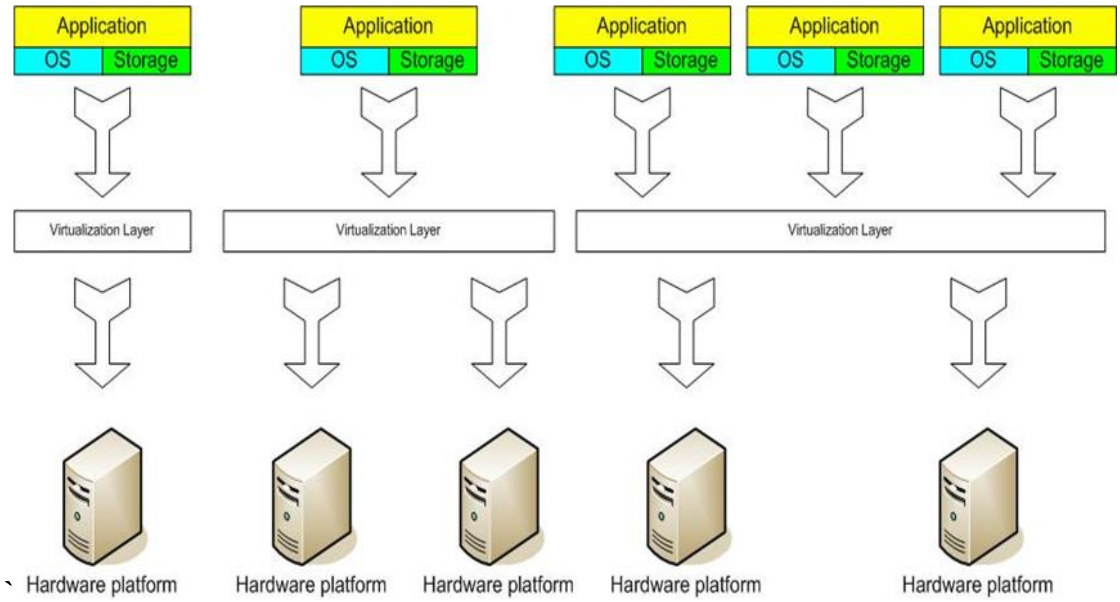

Figure 3b: Virtual Machine Monitor (VMM) layer between Guest OS and hardware [21]

Virtualization in data center domain offers the following benefits viz:

i. Efficient resource pooling
ii. Highly redundant infrastructure setup
iii. Highly available network
iv. Ease in the rapidity of deployable new servers
v. Easy to deploy server and hardware resources
vi. Auto-scaling (i.e.) reconfigurable while services are running
vii. Possibility in optimization of physical resources by doing more with less

Cloud computing leverages on these benefits of virtualization to provision highly dependable applications and infrastructures for organizational models.

A CONCEPTUAL WEB BASED ERP MODEL

The proposed ERP model (AppCloud.com) is a customer-facing web application which serves as an educational portal integrating many functionalities as shown in Figure 2. Students, Staff, partners, banks, payment gateway, etc use the web application to collaborate with each other using a rich web interface that can be viewed in a standard internet browser. The Web ERP model is self-automated and is projected to have 12 years implementation phases. The application generates a substantial amount of traffic to the site resulting in periodic spikes. Outside of the timeframes caused by these spikes, application experiences a fairly steady with predictable traffic load, which is characteristically high on weekdays and low on weekends.

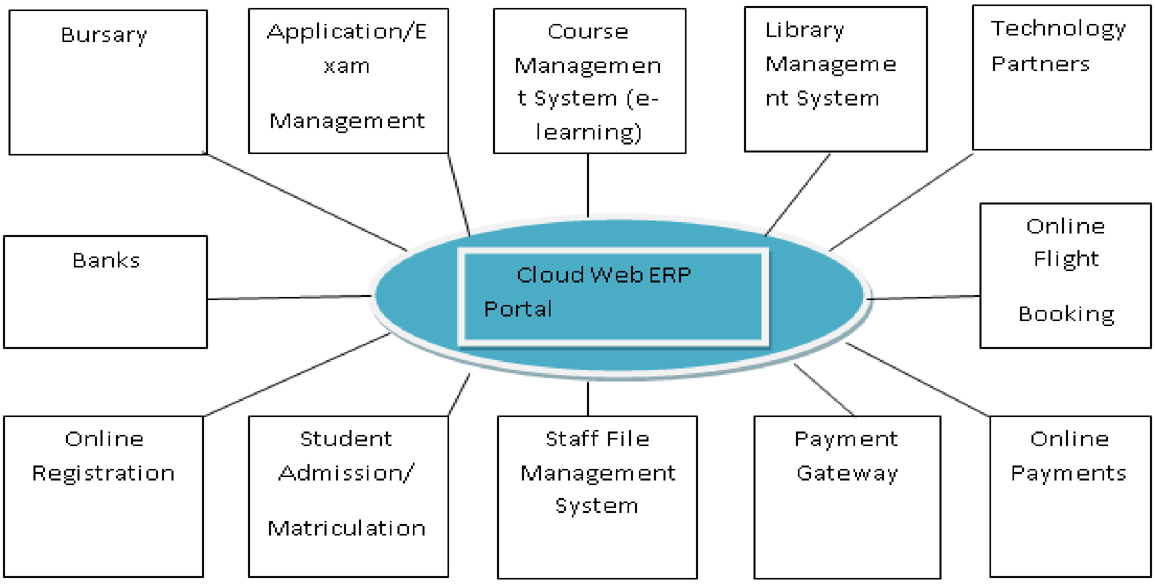

Figure 4: A Conceptual Model of the Proposed ERP Web Model

GOGGLE APP ENGINE FRAMEWORK (PAAS)

Google App Engine [22] allows a user to run Web applications written using the Java or Python programming languages. Other than supporting the Python standard library, Google App Engine also supports Application Programming Interfaces (APIs) for the data store, Google Accounts, URL fetch, image manipulation, and email services. Google App Engine also provides a Web-based Administration Console for the user to easily manage his running Web applications. Currently, Google App Engine is free to use with up to 500MB of storage and about 5 million page views per month. Again, this platform enables building and hosting of web applications on the same system that power Google application. The survey in [23] shows that GAE offers the fastest development, deployment, simple administration with no issues about hardware, patches, backups and effortless scalability. Hence, our private cloud in this work is Google App Engine. The proposed model is designed to serve millions of users concurrently just like search engines (e.g. Google), e-commerce sites (e.g. Amazon.com online shopping store), and social networking sites (e.g. Facebook). There are different types of cloud implementations; however, this work only adopts private cloud strategy for more administrative control and decentralized security.

Implementation Resources

Following the work done in [24], we propose the following in the prototype phase of the Web ERP model viz:

a. Java Eclipse (Eclipse Helios 3.6)- IDE support
b. Google Secure Data Connector
c. Private gadgets
d. Google Visualization API
e. Google Apps APIs
f. Google web toolkit

g. Google Plugins for Eclipse
h. Google App Engine Java SDK 1.5.5
i. Google Web Toolkit SDK 2.4.0
j. Programming Language Choice-Java.

Google App Engine runs Java applications using the Java 6 virtual machine (JVM). The JVM runs in a secured "sandbox" environment to isolate the ERP cloud application for service and security. The JVM can execute any Java bytecode that operates within the sandbox restrictions. The Administrative console provides the following details, viz: view access data and error logs, and analyze traffic browse the application's data store and manage indexes view the status of the application's scheduled tasks. The Supported Platform in GAE is Java Runtime Environment and Python Runtime Environment. Some of the features include [24]:

- Integration with Google Accounts
- URL Fetch
- Mail
- Memcache
- Image Manipulation
- Scheduled Tasks and Task Queues
- XMPP
- Blob store (which supports objects up to 50MB in size)

METHODOLOGY (SOFTWSRE DEVELOPMENT LIFE CYCLE (SDLC))

The SDLC starts with a planning phase, which identifies the business value of the proposed system model, conducts a feasibility analysis, and plans the project. The second phase is the analysis phase, which develops an analysis strategy, gathers information, and builds a set of analysis models. In the next phase, the design phase, the implementers develop the physical design, architecture design, interface design, data base and file specifications, and program design. In the final phase, implementation, the system is built, installed, and maintained.

In this work, the choice of our methodology was influenced by factors like: clarity of the requirements; familiarity with the base technology; system complexity; need for system reliability; time pressures; and need to see progress on the time schedule. For our proposed model in figure 4, the linear process model will be used in realizing the system since it offer better options for the implementation. Figure 5 and figure 6 shows Web ERP model decompositions.

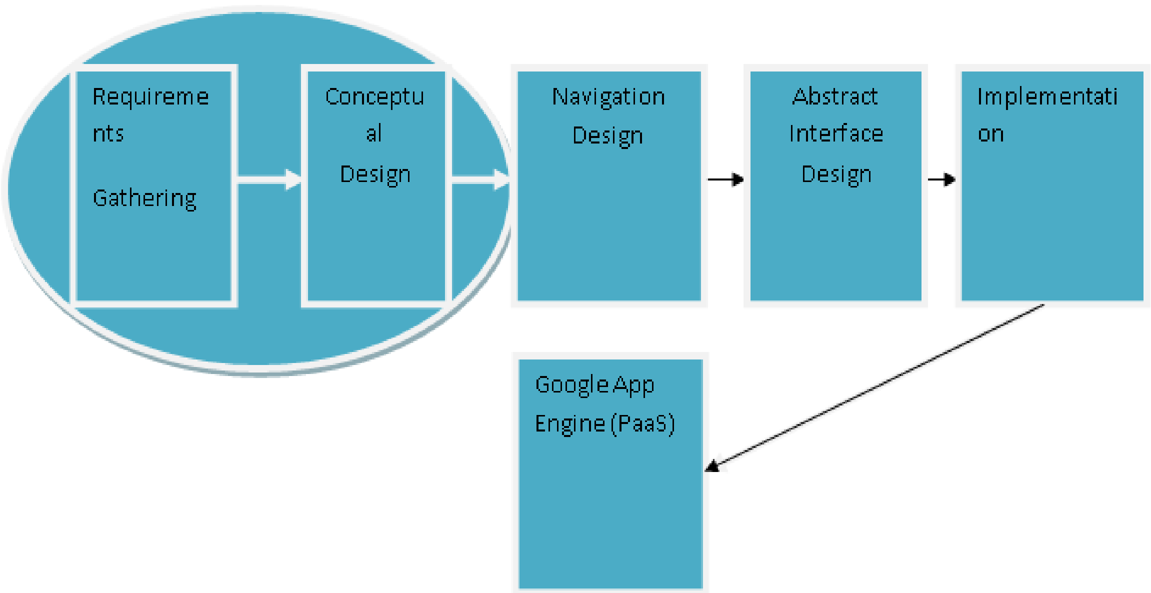

Figure 5: Phases in the Design and Implementation of Web ERP Model

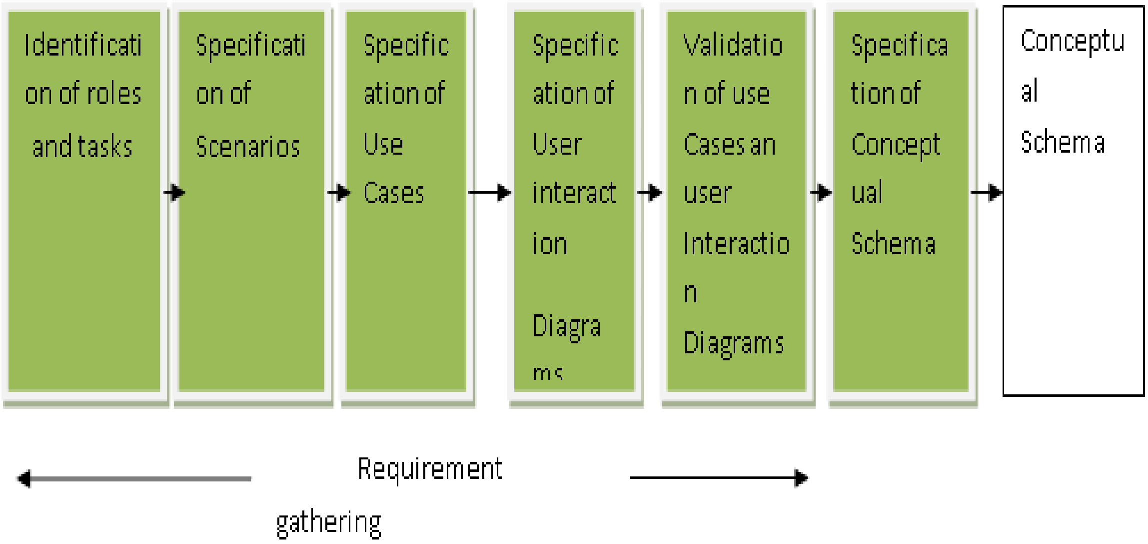

Figure 6: Phases of the requirements gathering and conceptual design [26]

Figure 6 shows the abstract decomposition of figure 5 with emphasis on requirement gathering and conceptual design. In first phase of this work, we gathered the system requirements from users and other stakeholders. Requirements gathering are divided into the following phases (Figure 6): identification of roles and tasks, specification of scenarios, specification of use cases, specification of user interaction diagrams, validation of use cases and user interaction diagrams. The conceptual design presents a single phase: specification of the conceptual schema. The implementation sequence in this work will be presented in the future work.

DISCUSSION (COST JUSTIFICATION OF THE WEB ERP)

We make a comparison of cost between a dedicated offering and Goggle Paas offering as shown in table 3

Table 2: Cost Comparison

Syskay Systems [25]	Google offering (GAE)
Premium Plan i. Domain Registration 100 GB ii. Web space 400 GB Monthly iii. Bandwidth Unlimited iv. MySQL Database v. Unlimited Email Accounts vi. Front Page Extensions #N30,000.00 Naira Annually #N60,000.00 Naira Biennially	**Service/Resource pricing** Pay- as you go pricing Mechanism Billing is based on the following parameters: -Outgoing Bandwidth -Incoming Bandwidth -CPU Time CPU -Stored Data -Recipients Emailed
Corporate Plan i. Domain Registration 200 GB ii. Web space 800 GB Monthly iii. Bandwidth Unlimited iv. MySQL Database v. Unlimited Email Accounts vi. Front Page Extensions vii. #N50,000.00 Naira Annually #N100,000.00 Naira Biennially	Flexible Pay as you go pricing structure

From the table 2, with Google app Engine cloud approach, flexible billing is feasible as one can enable billing, set a maximum daily budget, and allocate budget for each resource according to needs. This practically reduces capital expenditures improving the return on investment and reduces the operating expenditure by over 30%. Using the phased-driven approach, figure 7 is our projected cost saving projections for our model with GAE.

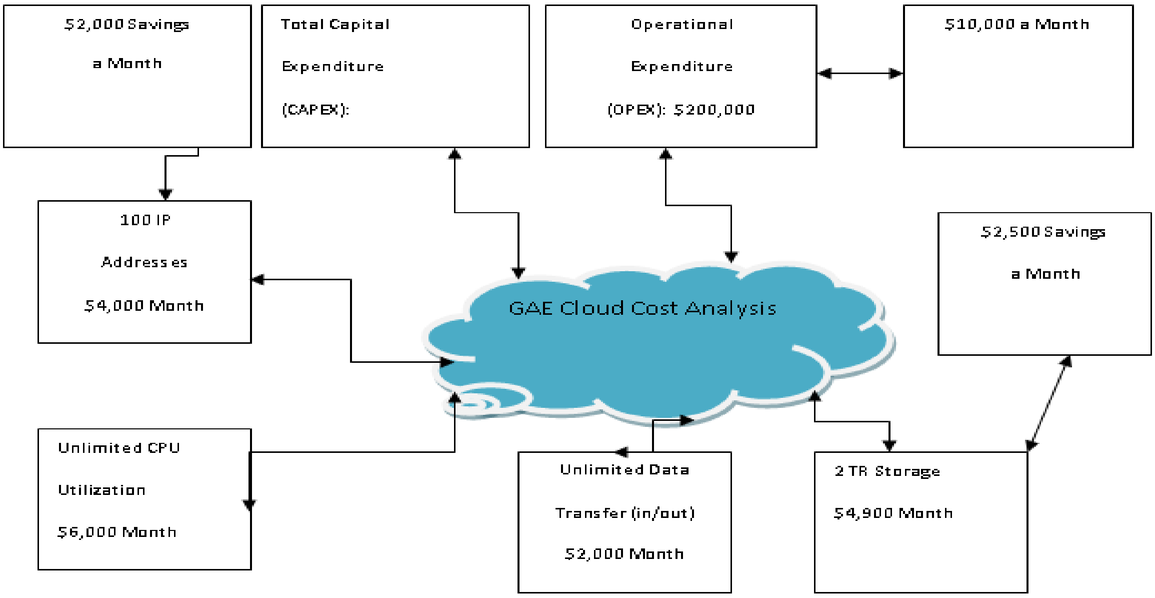

Figure 7: Abstract Cloud ROI Analysis Diagram

From figure 7, the resources considered for cost savings include CPU, bandwidth, storage, and Routing IPs. The GAE cloud cost analysis diagram shows return on investment (ROI) per month on the aforementioned resources. For a 2TB storage, 50% cost savings is realized, for a 100 IP addresses, 50% cost savings is acquired, etc. This is owing to the flexible pay-as –you go price model. In essence, the accumulated cost savings over a timeframe of 5 years can be diverted to other projects that have the potential to create a multiplier effect and economics of scale. This is a macroscopic view of cloud approach to our web ERP model in figure 4.

CONCLUSION AND FUTURE WORK

This work has discussed cloud computing in the context of a proposed cloud web model on Google App engine framework. Implementation tools have been presented with a linear SDLC process model. The work shows that cloud computing is very flexible and supports pay-as –you –go scheme. Whether implemented using web services, a collection of APIs or custom code, Cloud Computing is an ideal way to quickly prototype integrated applications that can be accessed through thin clients. They offer moderate performance and are low cost, scalable and rapidly deployable. Future work will discuss on the application security and interface integrations with various APIs.

REFERENCES

[1] Leena Jain and Sushil Bhardwaj, "Enterprise Cloud Computing: Key Considerations for Adoption" International Journal of Engineering and Information Technology Vol 2 , (2010). IJEIT 2010, 2(2), 113-117 ISSN 0976-0253 (Online).

[2] D. Nurmi et al., "The Eucalyptus Open-source Cloud-computing System"

[3] Amazon Web Services home page. http://aws.amazon.com/

[4] Enomalism elastic computing infrastructure. http:// www.enomaly.com

[5] K. Keahey, I. Foster, T. Freeman, and X. Zhang. Virtual workspaces: Achieving quality of service and quality of life in the grid. Sci. Program., 13(4):265–275, 2005.

[6] J. Chase, D. Irwin, L. Grit, J. Moore, and S. Sprenkle. Dynamic virtual clusters in a grid site manager.

High Performance Distributed Computing, 2003. Proceedings. 12th IEEE International Symposium on, pages 90–100, 2003.

[7] M. McNett, D. Gupta, A. Vahdat, and G. M. Voelker. Usher: An Extensible Framework for Managing Clusters of Virtual Machines. In Proceedings of the 21st Large Installation System Administration Conference (LISA), November 2007.

[8] Salesforce Customer Relationships Management (CRM) system. http://www.salesforce.com/

[9] D. Greschler and T. Mangan. Networking lessons in delivering ‘software as a service’: part i. Int. J. Netw. Manag., 12(5):317–321, 2002

[10] D. Greschler and T. Mangan. Networking lessons in delivering ’software as a service’: part ii. Int. J. Netw. Manag. 12(6):339–345, 2002.

[11] P. Laplante, J. Zhang, and J. Voas. What’s in a name? distinguishing between saas and soa. IT Professional, 10(3):46–50, May-June 2008.

[12] Rajkumar Buyya et al, “Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility, Future Generation Computer Systems, 25 (2009) 599-616, Journal homepage: http://www.elsevier.com/locate/fgcs

[13] Torry Harris, “ Cloud Computing Services – A comparison, 2008

[14] Google App Engine, http://appengine.google.com [18 July 2008]

[15] Microsoft Live Mesh, http://www.mesh.com [18 July 2008]

[16] Sun network.com (Sun Grid), http://www.network.com [18 July 2008]

[17] X. Chu, K. Nadiminti, C. Jin, S. Venugopal, and R. Buyya. Aneka: Next-Generation Enterprise Grid Platform for e-Science and e- Business Applications. In *Proceedings of the 3th IEEE International Conference on e-Science and Grid Computing (e-Science 2007),* Bangalore, India, Dec. 2007

[18] F. E. Gillett, “Future View: The New Tech Ecosystems Of Cloud, Cloud Services, And Cloud Computing”, August 28, 2008, Available : http://www.forrester.com

[19] Okafor Kennedy .C., Udeze Chidiebele. C, Prof. H. C. Inyiama, Dr. C.C Okezie, “ virtualization: a cost effective approach to infrastructure management in data center networks, (a case for elastic sky x server)”, (Unpublished).

[20] VMware- http://www.vmware.com

[21] Norman Wilde and Thomas Huber, “Virtualization and Cloud Computing” ppt, 2009

[22] Google App Engine, http://appengine.google.com [18 July 2008].

[23] Torry Harris, “ Cloud Computing Services – A comparison, 2008

[24] Okafor Kennedy .C, Michael Ndinechi, Udeze Chidiebele .C, “ Trusted Cashless Cloud: A Flexible Approach For The New Cashless Society and Its Security Recommendation For The Educational Sector In Nigeria” (Unpublished)

[25] www.syskay.com/contact.html

[26] Natacha Güell, Daniel Schwabe Patricia Vilain, “Modeling Interactions and Navigation in Web Applications”, (Unpublished).