

Architecture & Design of Affordable and Highly Available Enterprise Cloud Service

Rabi Prasad Padhy*, Manas Ranjan Patra**

*Senior Software Engg, Oracle Corp., Bangalore, Karnataka, India

** PG Department of Computer Science, Berhampur University, Odisha, India

Article Info

Article history:

Received Oct 10th, 2012

Revised Dec 12th, 2012

Accepted Dec 30th, 2012

Keyword:

Enterprise Cloud Service

High Availability Cloud

Google PaaS Cloud

Amazon Web Services

Cloud Migration

ABSTRACT

Cloud computing is a way to increase capacity or add capabilities on the fly without investing in additional infrastructure, licensing additional software or training new personnel. Enterprises must embrace this bleeding edge real-time business model in order to compete in the current marketplace. Deployment of enterprise applications in public cloud can reduce investment on IT infrastructure both in terms of hardware and software, whenever new services are to be provisioned. Furthermore, cloud services are quite attractive to business because of its dynamic scalability, privacy, performance and ability to handle heterogeneous environments. Instead of spending a lot of time in figuring out server setup and working on routers, it is judicious to subscribe cloud based 24/7 support and affordable services. It's an extremely important consideration to subscribe to at least two cloud vendors for smooth running of applications and ensure application availability by switching from one provider to another with minimal management effort. In this article we have proposed an architecture and design of a cost affordable online enterprise cloud application that subscribes to two public cloud vendors with respect to cost and availability parameters. We also present the reasons for the current trend of enterprises moving towards 24/7 online business cloud services.

Copyright © 2013 Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

Rabi Prasad Padhy,
Senior Software Engineer,
Oracle Corporation, Bangalore
Karnataka, India.
Email: rabi.padhy@gmail.com

1. INTRODUCTION

Enterprises implementing e-commerce applications need to invest on resources both hardware and software. As the business expands in scale, the required resources will surge up beyond the capacity of enterprises. In the existing conditions, cloud computing models allow enterprises with less investment to e-commerce business activities. Currently, the combination of e-commerce and cloud computing research focuses on the technical level, therefore, to start cloud-based e-commerce application model research will have a high practical value. The cloud computing as a new service model which is a collection of resources like software, platform, infrastructure offered as a service which can access using web applications. Typical cloud services include applications that cater to managing human resource, finance, customer operations, production, sales and marketing, and even affiliated business operations such as legal compliance and risks. They would also include the various services related to IT infrastructure with hardware, software and network systems as well as middleware that could offer integration, messaging, connectivity, R&D and other information tuning or performance services. Companies like Amazon, Microsoft, Google, salesforce.com

Journal homepage: <http://iaesjournal.com/online/index.php/IJ-CLOSER>

have been spending hundreds of thousands of dollars in the research and development of cloud services. It provides unique opportunities to companies to reduce costs, outsource non-core functions and scale costs to match demand. A cloud provider provides various cloud-based services like server, storage and content delivery. A content provider utilizes these two services to store, share, and distribute her content to multiple subscribers. The content provider and subscribers can access content via a cloud-based application service, which reads and manages the content stored in the storage service via cloud storage APIs. The application service is an application deployed in the cloud by the content provider or a third party. The content provider can use multiple cloud-based services from different cloud service providers to host her application service, content storage service, and content delivery service. For example, Netflix uses Amazon EC2 and S3 for content processing and storage, and uses multiple content delivery services such as Limelight, Level 3 etc. In a public cloud certain number of copies of the same data is created, regularly updated and stored at various geographical locations to ensure that a mishap in one of the geographical location doesn't affect the continuous data access process any of the client's applications might be doing. Traditionally, availability has been the primary measure of the success of IT service delivery and is defined through service level targets that measure the percentage of uptime (e. g. 99.99 percent availability). There are two fundamental assumptions behind using availability as the measure of success. First, that any service outage will be significant enough in length that the consumer will be aware of it and second, that there will be a significant negative impact to the business every time there is an outage. It is also a reasonable assumption that the longer it takes to restore the service, greater the impact on the business. The Cloud Service Providers manage high cloud applications, security, and infrastructure, as well as deploy, configure, maintain, and update the operation of the software applications on a cloud infrastructure. At the SaaS service level, the Cloud Consumers have limited administrative rights. They may be able to manage certain configuration settings of applications, but do not have control on the cloud infrastructure (network, servers, operating systems, storage & applications). Creating a dynamic IT infrastructure that guarantees the performance and availability of critical applications both at the LAN and across the WAN to support remote locations and yet is changeable on-the-fly has been the holy grail that IT operations people have longed for and computer architects have targeted for over the years.

2. SCENARIO DESCRIPTION

Online shopping is becoming increasingly popular due to various factors like Convenience, Better Prices, Variety, Gifts, Comparison of Prices, No Crowds, Compulsive Shopping, Discreet Purchases etc in upcoming times. As the e-commerce websites/portals of more and more businesses come into their own as significant revenue drivers, major retailers are now recognizing their online stores as mission-critical businesses. So Enterprises and their IT counterparts are starting to look for more-sophisticated applications that can best meet their current and future needs. The traditional application hosting and maintenance is required much storage capacity, scalability as well as high availability. Also once the application was developed, hosted then adding new features like launched Giftag, send wish cards, share information with friends through email, Facebook, Twitter, festival offers, change the look of home page frequently etc and scaling it for heavy use was onerous. Investing heavily in IT infrastructure when developing the new service. Buying and managing hardware would be costly and would take time away from building and regular maintenance of the platform. The main challenges lies in deploy multiple versions of application code in parallel and test new features before they go live in a traditional enterprise application. Instead of spending time managing servers, service systems and routers, it's always better to adopt a cloud-based Infrastructure for the same application resolving lot of burdens including human resources.

2.1 Problems with traditional infrastructure

Life Cycle Management: cost-effective management of structured & unstructured data throughout and testing to archiving and retirement replication facility needs to be provided and it's cumbersome. As the complexity and sophistication of the software development task has grown it needs to use increasing numbers of tools.

Licensing: Software licensing is required to renew product validity & support time to time where lot of human resource, downtime, cost required otherwise the vendor won't be fix the product code bugs time to time. It may leads to huge revenue loss in case the apps go down any time.

Modifiability: Traditional infrastructure example as they are not inherently scalable the provisioning cost and achieve high availability for moving from 1000 users to 10000 users could consume lots of IT resources.

Scalability: To achieve scalability, a customer needs to depend on multiple vendors for IT resources and chase with support to fulfill this which is the reality due to collaboration with multiple vendors simultaneous while service is down.

High Availability (HA): Extra infrastructure & cost required to scale the traditional online application. Also there is no guarantee of 100% data recovery.

3. ARCHITECTURE OF PROPOSED MODEL

In this scenario study, we choose two cloud providers as our experiment subjects. They are Amazon cloud (AWS) and Google PaaS Cloud i.e. Google App Engine. We choose these two cloud providers because they are popular and have been continuously ranked as the top 5 cloud providers. Amazon EC2 and S3 are well-known Infrastructure-as-a-Service (IaaS) cloud providers. They can provision VMs with different hardware configurations based on the user requests. In our scenario we consider two parameters i.e. Cost and High Availability for Service 'A' which is subscribed from two different Public Cloud reputed Vendors with multiple cloud Data Center Locations.

Lets Subscribe Service 'A' from Google PaaS Cloud (Google App Engine) with available configuration (always running) based on Location L1. Also the same Service 'A' preferred configuration (Standby) on Amazon Cloud (AWS) based on Location L2. The subscription based on Post Paid and prepaid basis as the Service always running on Google PaaS Cloud, so let's assume it's a post paid subscription service. Also as the Service A is preferably configured on Amazon Cloud (AWS), so let's assume it's a pre-paid subscription. In this study, we compare the service performance, availability, cost based on different factors. These factors include data center location and the number of users logged in at the same time. Clouds organize data centers as networks of virtualized IaaS (computing servers, databases, and networks) and PaaS (load balancers and auto scalers) so that providers are able to access and deploy applications (SaaS) from anywhere in the world on demand at competitive costs driven by QoS requirements. The Cloud computing system, P, is a set of Cloud infrastructures owned and maintained by 3rd party IaaS/PaaS providers such as Amazon AWS, Google PaaS Cloud (Google App Engine). More formally, $P = (c_1, c_2, \dots, c_n)$, where c_1, c_2, \dots, c_n are data centers that are part of P. An application deployment is composed of m VM instances $\{v_1, v_2, \dots, v_m\}$, where m is either fixed or varies with time based on the current workload and performance requirements. Application instances are examples of SaaS software.

We consider a online shopping application scenario with cost & high availability parameters in the Public Cloud Provider environment. A Large Enterprise application should be available at anytime for testing and should provide the fastest response time regardless of the number of users served. Therefore, a Cloud service system should be Highly Available. Users may not know the under laying infrastructure to efficiently perform their jobs. Thus, a standby compatible service provider should be in place. The overall architecture that effectively handles our scenario is illustrated in Figure 1. If in case the service A running on Google PaaS Cloud fails to fulfill the requirement then switchover to Amazon cloud (AWS).

In our scenario resource provisioning process and management on cloud infrastructures consists of three key steps.

Virtual Machine provisioning: It involves instantiation of one or more Virtual Machines (VMs) that match the specific hardware characteristics and software requirements of an application. For example Amazon EC2 supports 11 types of VMs, each one with different options of processors, memory, and I/O performance;

Resource Provisioning: It is the mapping and scheduling of VMs onto physical Cloud servers within a cloud. Currently, most IaaS providers do not provide any control over resource provisioning to application providers. In other words, mapping of VMs to physical servers is completely hidden from application providers; and

Application Provisioning: It is the deployment of our online shopping applications within VMs and mapping of end-user's requests to application instances.

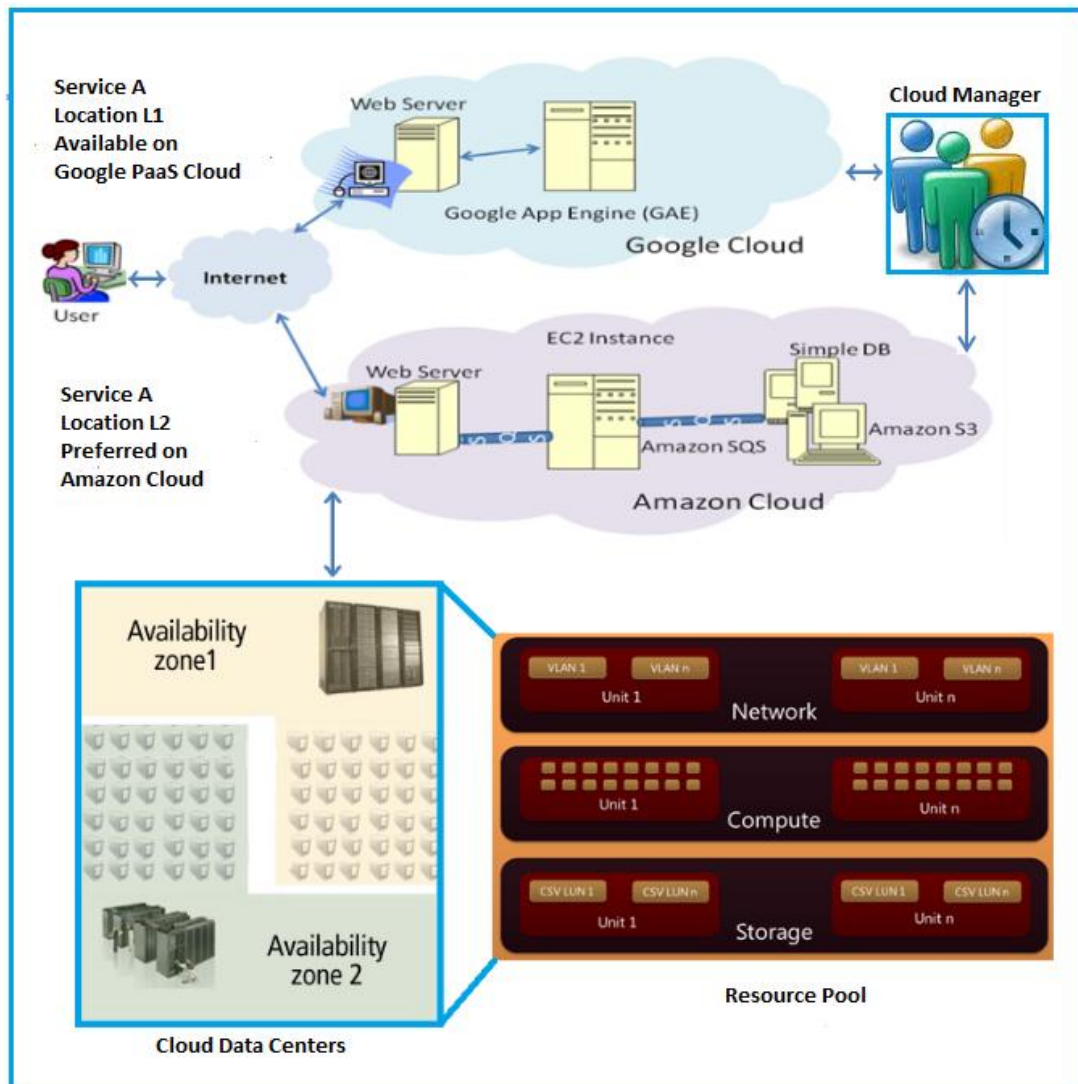


Figure 1. Effects of selecting different switching under dynamic condition

3.2 Components of the Proposed Model

Cloud API: Application Program Interface used by the tenant or customers to interact with the application. The most common API is RESTful HTTP or HTTPS.

Cloud Domain Name System (DNS): It is responsible for storing the IP and domain name information for IPv4 and IPv6 and also system metadata, such as the information gathered by the cloud controller and unique machine identification information. For our architecture, DNS was chosen as the storage location for metadata because it is a standards-based system that already exists in most networks, it is available to all cloud components, and it alleviates the requirement of adding additional database-type storage to the infrastructure. Any DNS solution that is managed by IT and has the ability to add additional zones and zone files can be used. A key element of the DNS is a worldwide collection of DNS servers. A DNS server is any computer registered to join the Domain Name System. A DNS server runs special-purpose networking software, features a public IP address, and contains a database of network names and addresses for other Internet hosts.

Cloud Controller: The cloud controller is the front-end system responsible for gathering and aggregating the preliminary data required to start a provisioning process. Initially, this information is provided by an administrator as part of the creation process and is specific to each type of workflow used for provisioning.

For example, the cloud controller gathers information that includes VM location, class of application (web server, database server, etc.) and minimum resource requirements.

Cloud Load balancer: Load balancing mechanisms are applied to webpage browsing or database transactions. The load balancer assumes the IP address of the web application, so all communication with the web application hits the load balancer first. The load balancer is connected to one or more identical web servers in the back-end. Depending on the user session and the load on each web server, the load balancer forwards packets to different web servers for processing. The hardware-based load balancer is designed to handle high-level of load, so it can easily scale. Google App Engine implicitly over a hardware-based load balancer.

Cloud Resource Pooling: A Resource Pool is a collection of shared resources composed of compute, storage, and network that create the fabric that hosts virtualized workloads. Subsets of these resources are allocated to the customers as needed and conversely, returned to the pool when they are not needed. Ideally, the Resource Pool should be as homogenized and standardized as possible.

Cloud Manager: It consists of multiple sub-components that provide different capabilities covering the various aspects of the end-to-end fulfillment and management of cloud services. This includes separate Customer and Service Provider portals, a component to handle all customer facing processes such as browsing catalog, account management, customer order handling, shopping cart, monitoring/managing cloud services, customer update emails, etc., and components to manage the provisioning, activation, resource allocation, change management database (CMDB), IP address/domain name management, and the management of IP servers such as DNS/DHCP. The system also provides a Software Development Kit (SDK) for designing, developing, and deploying custom interfaces, custom business processes that modify the core business processes, reports, email templates, etc.

Availability zones: Cloud data centers are located in four different geographic regions. Each region contains a set of resources to be used like instances, AMIs, security groups etc. Availability Zones are allows users to assign instances to locations that are very fault-tolerant from one another yet that have very high bandwidth between each other. Availability zones are equals to a datacenter. If power goes out in one datacenter and the generators fail to start then it doesn't affect the other datacenter. Or if there's a fire, one datacenter may burn out or be otherwise incapacitated, but others are unaffected.

Cloud Data Storage: Cloud storage involves storing data on multiple virtual servers sharing and retrieval on any IP enabled device. Unlike the traditional hosting types where if the server hardware fails for some reason, the application or websites tend to go unresponsive causing a downtime, but in cloud, despite one of the servers within the cluster going down, the application/website wouldn't face any downtime, still we can access our data from anywhere provided we have access to the Internet.

Cloud Switching Fabric: Fabric Management can be thought of as an orchestration engine, which is responsible for managing the life cycle of a consumer's workload (one or more VMs which collectively deliver a service). It responds to service requests (e.g. to provision a new VM or set of VMs), Systems Management events (e.g. moving/restarting VMs as a result of a warning or failure), and Service Management policies (e.g. adding another VM to a consumer workload in response to load). To ensure resiliency we must be able to automatically detect if a hardware component is operating at a diminished capacity or has failed. This requires an understanding of all of the hardware components that work together to deliver a service, and the interrelationships between these components. Fabric Management provides this understanding of interrelationships to determine which services are impacted by a component failure. This enables the Fabric Management system to determine if an automated response action is needed to prevent an outage, or to quickly restore a failed service onto another host within the fabric.

3.3 Connectivity to Google PaaS Cloud

Google App Engine made it really easy to build and deploy online application without worrying about things like hardware, redundancy, load balancing, data backups, generic patches, kernel upgrades and Security. It also enables businesses to build and host web apps on the same systems that power Google applications. It offers fast development and deployment, effortless scalability, simple administration and add new features quickly and continually refine the service. User just needs to upload application, and it's ready to serve. In our proposed model i.e online shopping application we subscribed Google App Engine for Java

runtime environment even it supports several programming languages. With App Engine there are no set-up costs and no recurring fees. The resources for our application uses, such as storage and bandwidth, are measured by the gigabyte, and billed at competitive rates. We control the maximum amounts of resources our app can consume, so it always stays within our budget. App Engine costs nothing to get started. Google App Engine makes it easy to build an application that runs reliably, even under heavy load and with large amounts of data. Our app accesses most App Engine services using Java standard APIs. For the App Engine data store, the Java SDK includes implementations of the Java Data Objects (JDO) and Java Persistence API (JPA) interfaces. Your app can use the Java Mail API to send email messages with the App Engine Mail service. The java.net HTTP APIs accesses the App Engine URL fetch service. App Engine includes various features like persistent storage with queries, sorting and transactions, automatic scaling and load balancing, APIs for authenticating users and sending email using Google Accounts, Task queues for performing work outside of the scope of a web request, Scheduled tasks for triggering events at specified times and regular intervals, Dynamic web serving, with full support for common web technologies.

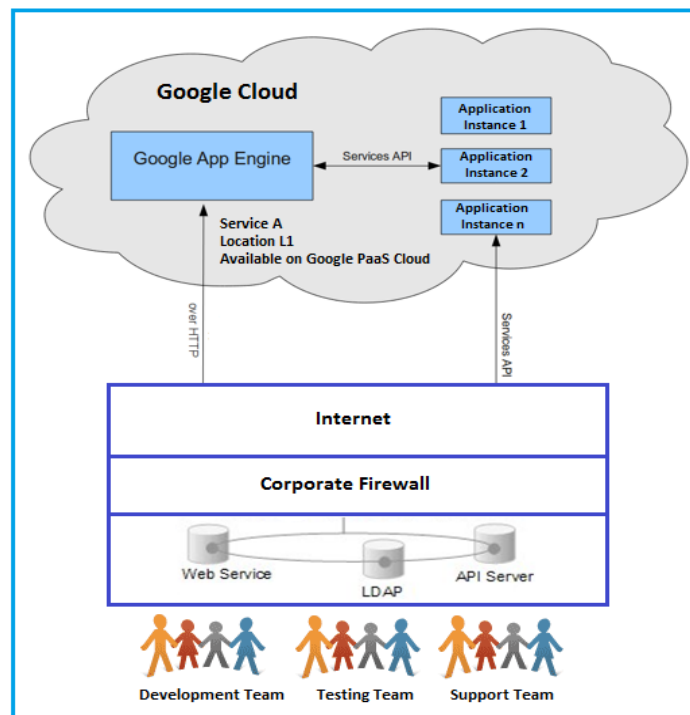


Figure 2: Connectivity & Workflow for Google PaaS Cloud

3.4 Connectivity to Amazon Cloud

Amazon Cloud (Amazon Web Services) provides a set of basic services to use computing infrastructure (CPU cycles, storage, and network), install a platform (e.g., a Tomcat web/app server and MySQL database server), and run an application on that platform. Amazon, therefore, provides a great basis to implement different architectural variants. The basic Amazon services used in the experiments reported in this paper are EC2 (for CPU cycles), EBS (for a storage service that can be mounted like a disk), and S3 (for storage that can be used as a key-value store). EC2 currently provides resources in three different regions: 'US – N. Virginia', 'US – N. California' and 'EU – Ireland', while the lower costs are currently for resources located in 'US – N. Virginia'. Each region has a number of availability zones which ensures that machines are not within the same failure zone. We report the different options available to date to access EC2 resources and their cost. Note that all options are IP-based and are layer-3 solutions at best.

Dynamically assigned IP addresses: EC2 instances (servers) are dynamically assigned IP addresses when they are created. A public IP address is available for remote connections and a private IP address is created for internal communications (inside EC2). All network traffic between EC2 instances (inside the same availability zone of the same region) is free. Traffic between resources in different regions is considered Internet traffic.

Static IP addresses: Amazon offers static IPv4 addresses that can be assigned to instances at an additional cost. A customer is limited to 5 such addresses by default but more can be obtained on demand. An instance first starts with a generic public and private IP addresses and then can be remapped to a static IP address (called Elastic IP).

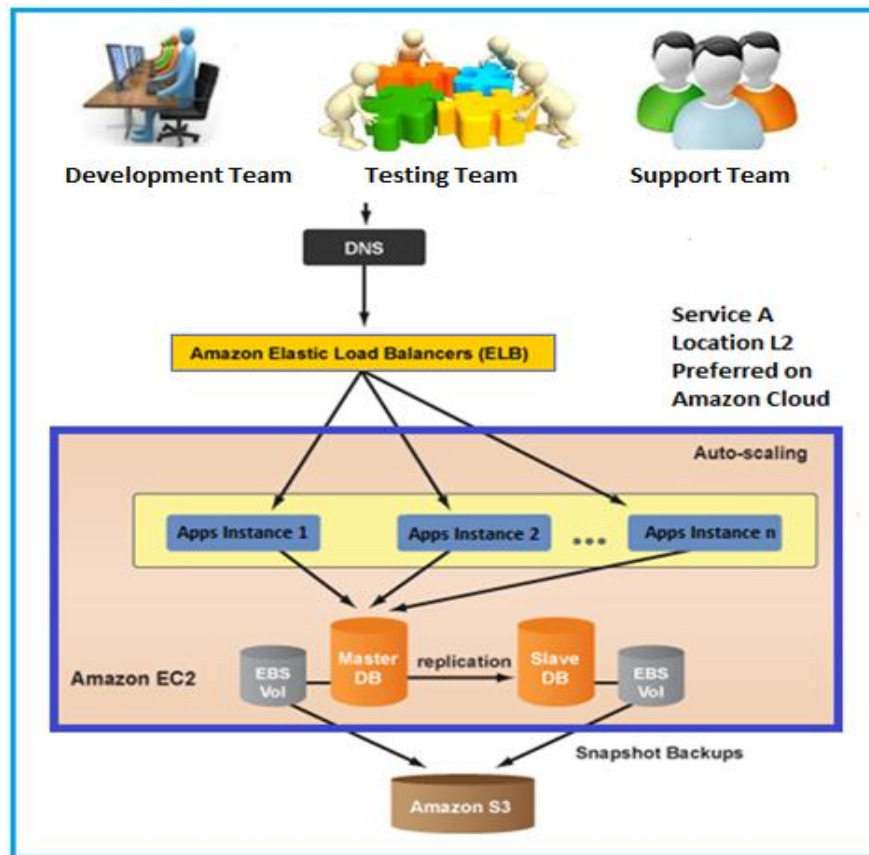


Figure 3: Connectivity & workflow for Amazon Cloud (AWS)

3.5 Case 1: Cost Affordability

Cost is generally one of the primary reasons for moving a business application from traditional data center to the cloud. In a traditional data center, it may not be possible to determine what percentage of a shared resource, such as infrastructure, is consumed by a particular service. This makes benchmarking services against the market an impossible task. By defining the cost of infrastructure through service classification and consumption modeling, a more accurate picture of the true cost of utilizing shared resources can be gained. This allows the business to make fair comparisons of internal services to market offerings and enables informed investment decisions. Cost transparency also incents service owners to think about service retirement. In a traditional data center, services may fall out of use but often there is no consideration on how to retire an unused service. The cost of ongoing support and maintenance for an under-utilized service may be hidden in the cost model of the data center. Monthly consumption costs for each service can be provided to the business, inciting service owners to retire unused services and reduce their cost.

In traditional test, development and hosting systems, companies would set up an environment for each project. Different systems may have different functions, performances, or stabilities and thus software and hardware configurations will vary accordingly. However, in a cloud platform, all the servers, memories and networks needed are pooling-managed and through the technology of virtualization, each test or development project is provided with a logical hardware platform. The virtual hardware platforms of multiple projects can share the same set of hardware resources, thus through integrating the different project, the hardware

investment will be greatly reduced. The new generation of intelligent cloud platforms needs the support of intelligent IT infrastructures. By establishing intelligent platforms through cloud computing, a new IT resource supply mode can be formed. Under this mode, the test development and hosting center can automatically manage and dynamically distribute, deploy, configure, reconfigure and recycle IT resources based on the requirements of different projects; besides, it can also install software and application systems automatically. When projects are over, the test development center can recycle the resources automatically, thereby making the best use of the computing capabilities.

3.6 Case 2: High Availability

Technically there are several levels where high availability can be achieved. These levels include application level, data center level, infrastructure level and geographic location level. One of the very basic goals of high availability is to avoid single point of failures as much as possible to achieve operational continuity, redundancy and fail-over capability. At the infrastructure level the basic configuration might look like the one in the Figure 1. This configuration has two public cloud service providers with two different locations. The consumer accesses the cloud via the Internet. At each location both active and passive cloud resource provisioned to provide high availability. Even If one public cloud service goes bad, the second cloud service provider supports service continuity and hence reducing the downtime. This is applicable for both the public cloud providers as shown in Figure 1.

There are two main factors that affect availability. First is reliability which is measured by Mean-Time-Between-Failures (MTBF). This measures the time between service outages. Second is resiliency which is measured by Mean-Time-to-Restore-Service (MTRS). MTRS measures the total elapsed time from the start of a service outage to the time the service is restored. The fact that human intervention is normally required to detect and respond to incidents limits how much MTRS can be reduced. Therefore organizations have traditionally focused on MTBF to achieve availability targets. Achieving higher availability through greater reliability requires increased investment in redundant hardware and an exponential increase in the cost of implementing and maintaining this hardware. In a traditional data center, the MTRS may average well over an hour while a cloud data center can recover from failures in a matter of seconds. Combined with the automation of detection and response to failure and warn states within the infrastructure, this can reduce the MTRS (from the perspective of IaaS) dramatically. Thus a significant increase in resiliency makes the reliability factor much less important.

4. WORKFLOW OF THE PROPOSED MODEL

The work flow of our proposed model shown in figure 4 explained in the below steps:

Admin Authentication: The Cloud Admin request a service to the Cloud manager from the service provider end who gives the specification in the given format < Operating system, Memory, Hard Disk, Processor, Software > with admin access. The cloud manager always checks resource approval sheet before grant the service with the help of resource monitoring and billing teams.

Cloud Administrator: Once a client submits the specification, the Cloud Admin creates a template for virtual machine. Then a virtual network is created. Then Cloud Admin issues the create virtual machine command.

User Authentication: The application teams / users should authenticate them self to access the online shopping application environment through some process by Cloud Administrator.

Access Service: Once the service is re-configure with security access as per the requirement then the cloud admin provide the user authentication to access the service.

Service Request: The application teams (Dev / Test / Support) request for use of the service for application customization and different purposes.

Remote Access: The management team of the cloud consumer end always keep on monitoring the quality of the service using cloud enable mobile devices with remote access privileges.

Online Shopping Application Portal: First the user has to register to the online shopping portal. Once an account is created, then user can login to the portal and can browse carts and place the order.

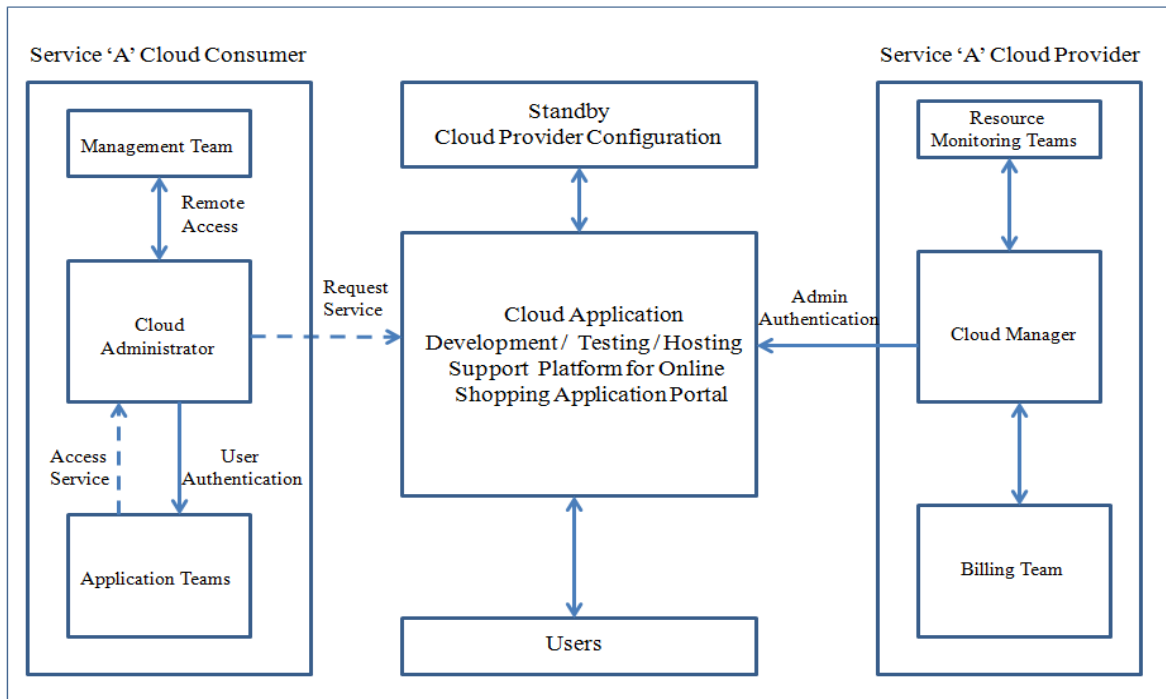


Figure 4. Block Diagram of the Our Proposed Model

5. LOAD BALANCING MANAGEMENT OF THE PROPOSED MODEL

- The user/client/admin connects to the Internet and requests a service.
- DNS routes the user to a specific IP address which is connected to the cloud data center.
- The user is connected to the closest & local Time Zone specific cloud data center.
- The cloud data center accepts the connection based on customer specified policy, decides which of the customer’s datacenters to send the user to based on performance, geography, etc.
- The user is directed to the customer’s datacenter containing the desired application.
- Service is delivered to the user.

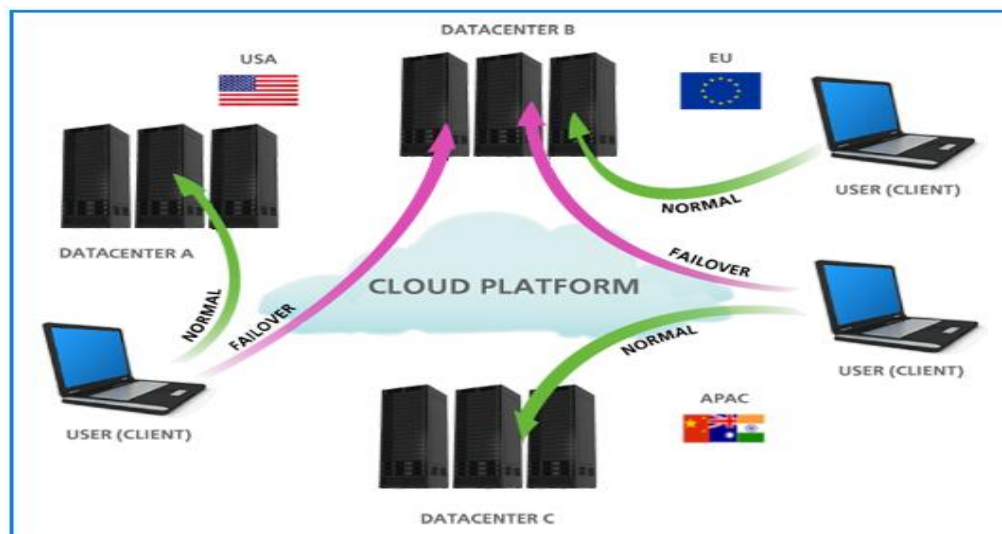


Figure 5. Overall View of Load Balancing of Proposed Model

6. SWITCH-OVER / SWITCH-BACK AND FAILOVER BETWEEN CLOUD PROVIDERS

The Infrastructure layer which support the cloud service 'A' require a similar configuration while switching to Public cloud 2 as per below parameters.

Network : routing, address assignment, name services
 System : OS, authentication services
 Virtualization : Virtual/physical mapping, storage mapping
 Services : load balancing of a web service

There are several cases where we need to switch over from one service providers to another. The following are some of those cases.

Maintenance: There are various business requirements where we should do the maintenance of the application as well as infrastructure, so there is always pre-downtime window will open and well informed in advance both the cloud provider and consumers.

Service Connectivity Testing: Security Testing is one of the important test in a cloud environment to check the integrity of the service.

Backup & Restoration of Data: This test also happended for storage compatibility for the subscribed cloud vendors time to time for high availability and interportability of the data.

Follow the Sun cloud service: In a 24/7 application cloud provider will options to change the service center location frequently for quality of the service support. This requires a major change in VM units showing in figure 5.

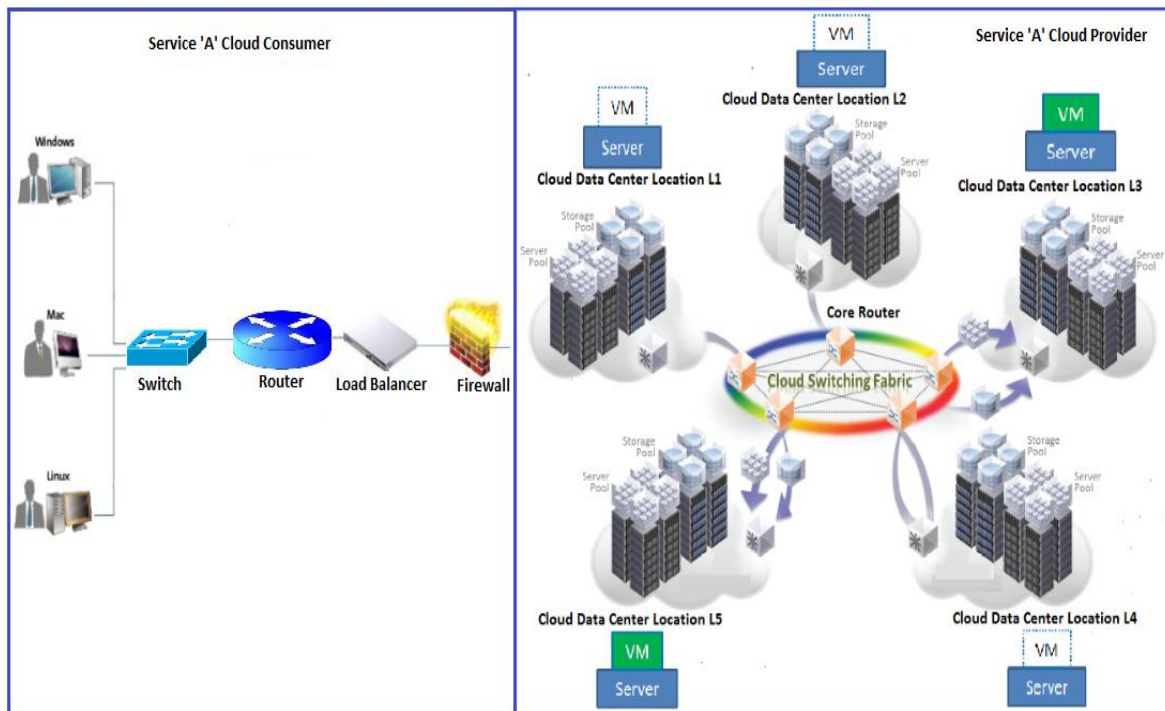


Figure 6: High Level view of follow the sun support cloud infrastructure architecture

Geographical failover is a redundant system that is made to survive disruptions that can involve whole regions. Geographical failover eliminates certain weaknesses inherent to natural disasters, man-made disasters, mistakes, and law enforcement orders. Failover is a mechanism that ensures service continuity

whereby single-point failures are eliminated. It is a mechanism because failover has to be minutiously designed, implemented, and deployed. There are several types of failover mechanisms but the most resilient configuration spans geographical locations like failover from a primary VM Unit to a secondary VM Unit and failover from a primary VM Unit to a secondary VM Unit or to tertiary VM Unit to handle cascading failures.

Traditionally, failover is implemented at the network level of the infrastructure to prevent network disruptions. But in reality, there are several elements of an infrastructure that could experience disruption. To avoid disruption, the first cloud instance acts as the Primary Network in one geographical location, and during normal usage it serves all requests. In the event of a failure the Second Network or Failover Cloud Instance becomes active automatically in a different geographical location and serves all requests. The data from the Primary cloud instance to the Failover cloud instance is automatically synchronized on a daily basis. Both the Primary and Failover instances are maintained by identical software's, and all users specific data is synchronized automatically.

6.1 Quality of Service (QoS) & Service Level Agreement (SLA)

During switch over & switch back /failover operation between cloud providers, its very importenet to measue the quality of the service. To maintain a good quality of service (QoS), cloud administrators should provision adequate resources to cope with workload demand fluctuations. Unfortunately, overprovisioning implies extra cost and decreases the business profit. On the other hand, under-provisioning degrades the QoS, irritates the clients, and consequently retreats the EBusiness durability. QoS is determined in the form of SLA (Service Level Agreement), which is determined either by minimum throughput or maximizes response time. This can differ from system to system. QoS parameters are necessary to fill the services request in order to negotiate the cloud resource, describe the Cloud offer, match the compliant services and build the best available solution define the SLA and Monitor the service levels. Some of the most importement SLA parameters include defining the Outage, how does a consumer prove an outage to get credit, how does the credit get applied etc.

7. DESIGN OF THE PROPOSED MODEL

We used UML sequence diagrams for our proposed Model.

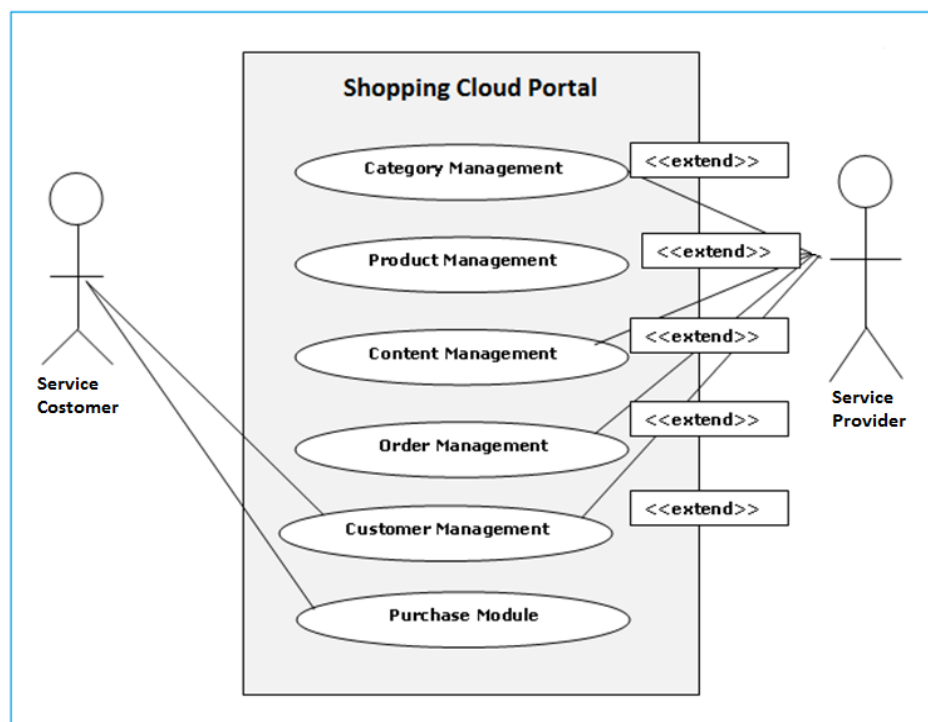


Figure 7: Use Case Diagram for Online Shopping Portal Cloud Service

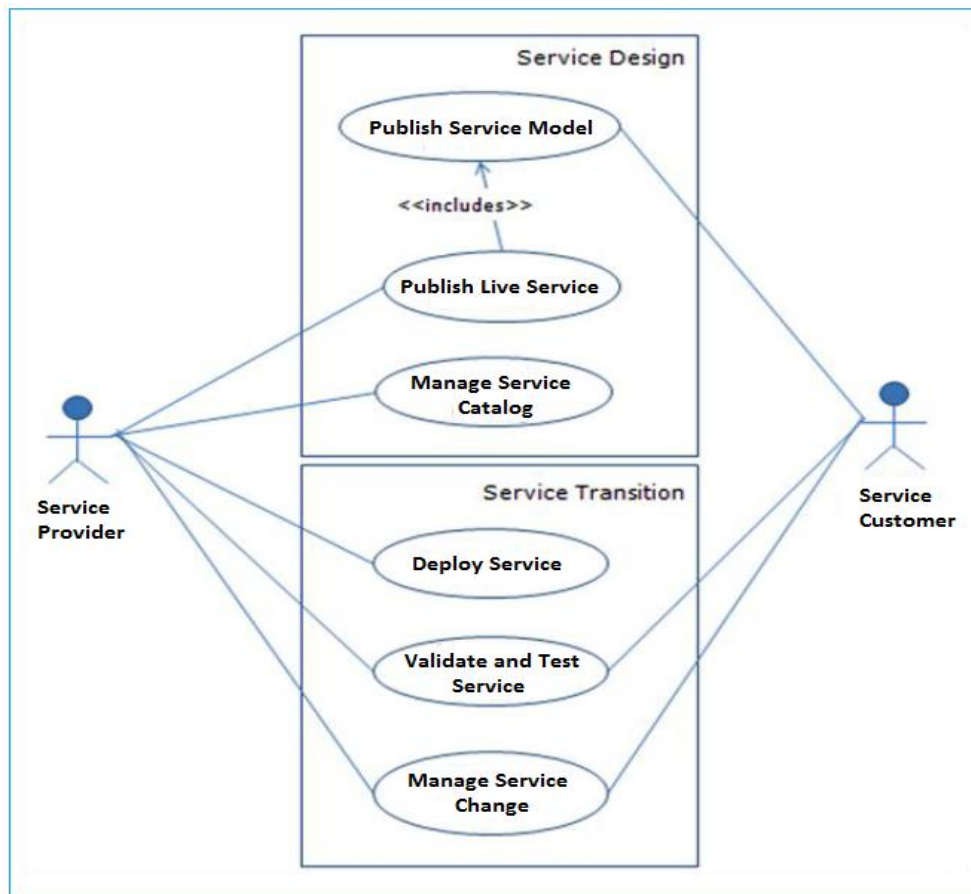


Figure 8: Use Case Diagram of the Online Shopping Deployment Cloud Service

Customer accesses the cloud online shopping application via Internet.

Use Cases Description

Login: The customer can login to the cloud online shopping application by enter his user name and password. The system will verify that the login name matches the login password. If they do not match, error message will be indicated to the customer.

Add to Shopping Cart: When the customer finds the products he wants, he adds them to the shopping carts. The application will store and keep track the information of the products that have been added into shopping cart.

View Shopping Cart: The customer can request to view the contents of the shopping cart. The application will return the contents of the shopping cart to the customer; the unit price and total price will be shown as well.

Update Customer Info: The customer can request to update their customer info. Customer information includes the relative information of customer such as username, password, address, etc. The application will display the current customer info to the customer. The customer updates the customer information and the application will store the updated customer info in the cloud database. After one successful purchase, the payment information is also stored in customer information.

Register: If the customer is a new user, he can request to register with the application. The application displays a registration page and asks the customer to choose a login name (any email id) and password. The

customer is also required to enter their name and address. Shipping information and credit card information are optional entries at this point.

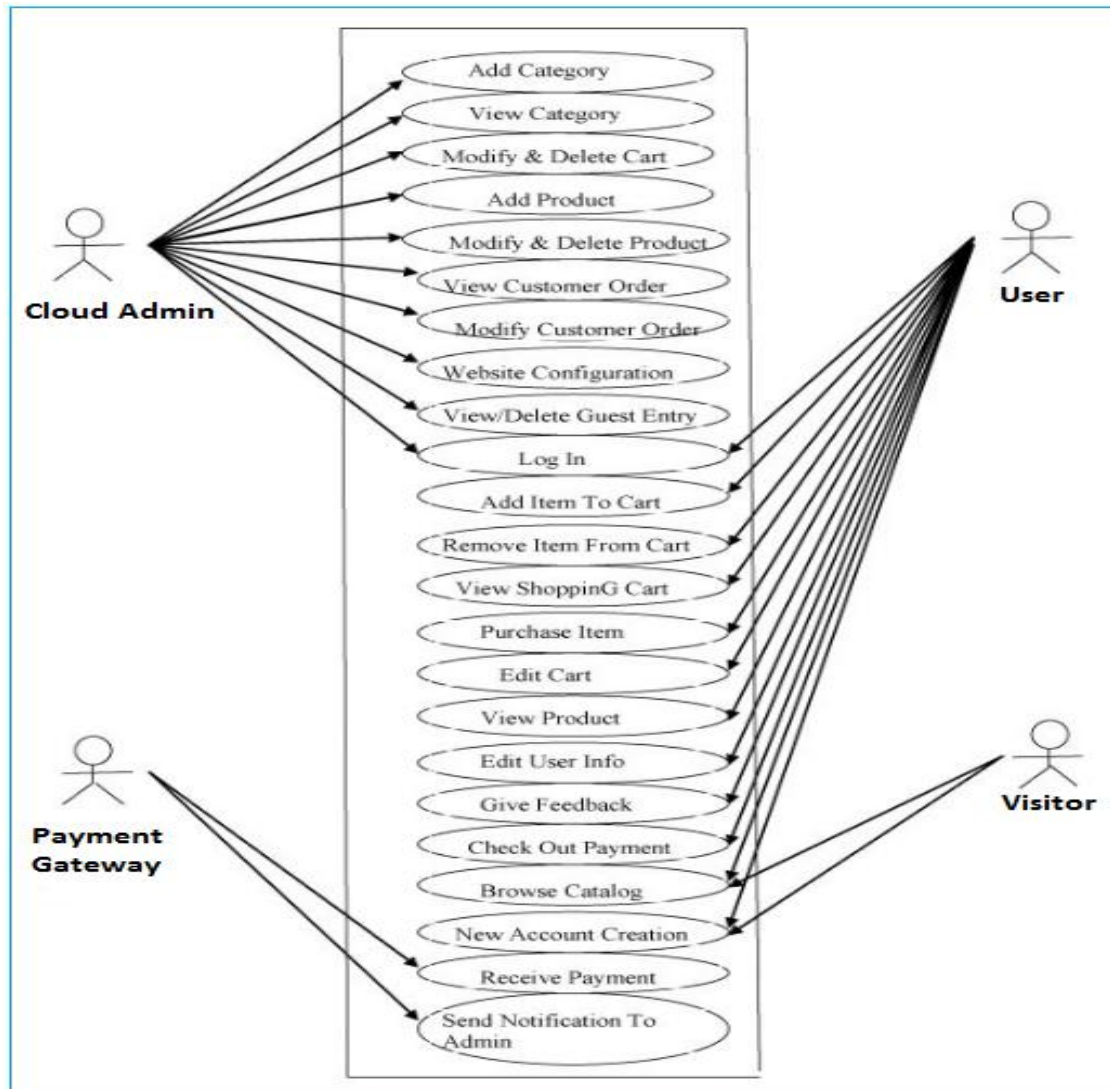


Figure 9: Use Case Diagram for our Proposed Model

Enter Shipping & Credit Card Information: When the customer requests to checkout and he does not have credit card information stored at this point (application can not find his payment information), the application will prompt credit card information page. The customer will be given a choice on whether he wants the item shipped to his stored address or to an alternative address. The input payment information will be save into the order form.

Verify Credit Card: When the customer checks out, the credit verification company validates the customer's credit card when given the customers name, credit card number, and expiration date, and then returns the validation result to checkout department (sales person). If the response shows that the credit card is invalid, the customer will be asked to re-input his payment information.

Update Product Information: The sales clerk requests to update products information. This includes the products price, description, brand, title, or number. The system will save the updated product information in the cloud database.

Update Inventory: User requests to update inventory. The application will update the product information in the cloud database.

Ship Products: After getting the order request, the customers sales staff ships the order products to their respective clients within three to five business days.

Checkout: When the user finishes shopping, she/he requests to checkout. If the payment information of this user already exists, the application prompts the customer to review or input a new one. The application then forwards the credit card information to credit Verification Company. If the credit card is invalid, the customer is given the option to use another credit card or just cancel the order. If the credit card is valid, the order form will be processed by the application and checkout is complete.

Browse catalog: The customer requests to view the products in a product catalog. The application will display the product information of the selected category.

Perform Products Search: The customer enters product search parameters and requests a product search. The application will search through the products category in its database and return the matches to the customer. If there are no matches, the system will display a fail message.

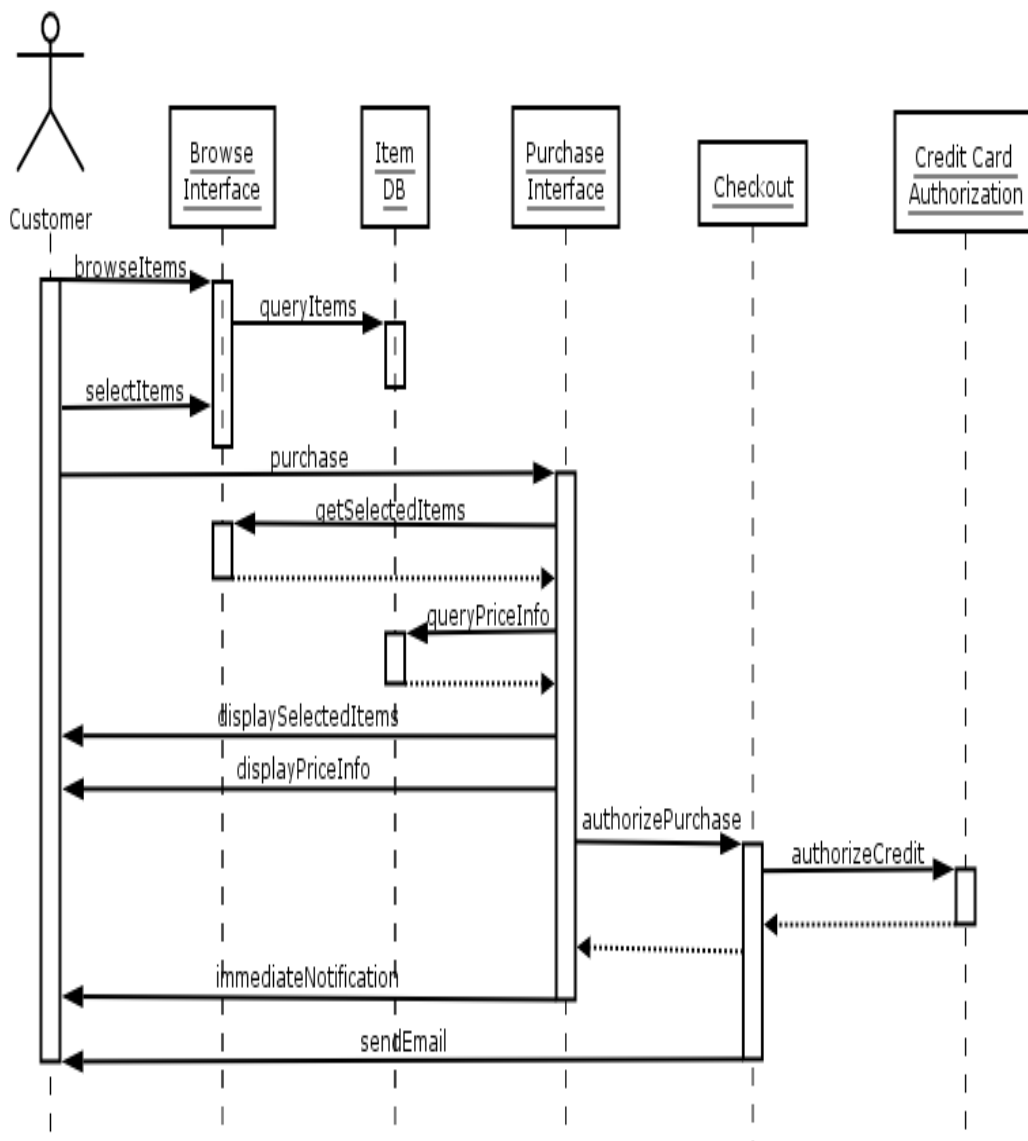


Figure 10: Sequence diagram for online shopping cloud service

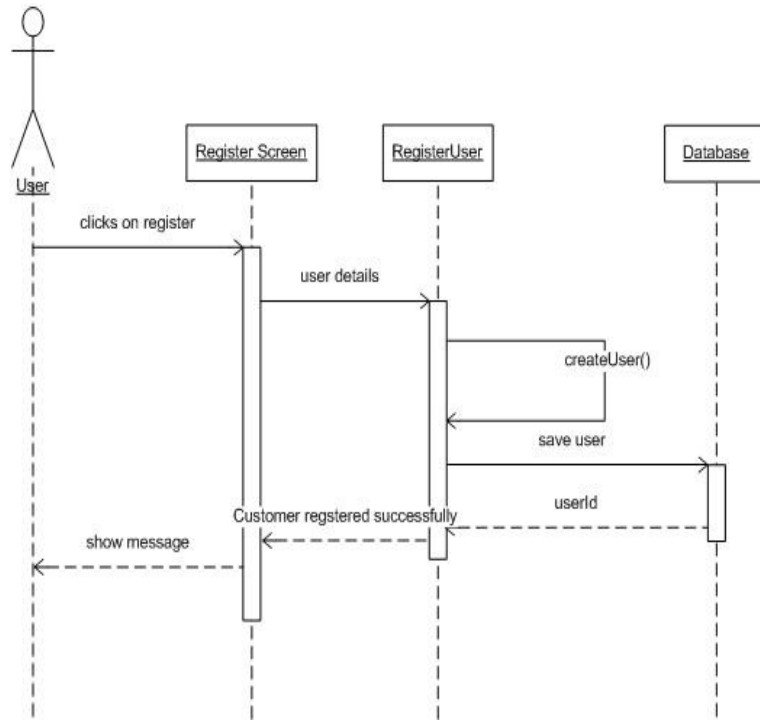


Figure 11: Sequence Diagram for Register a user

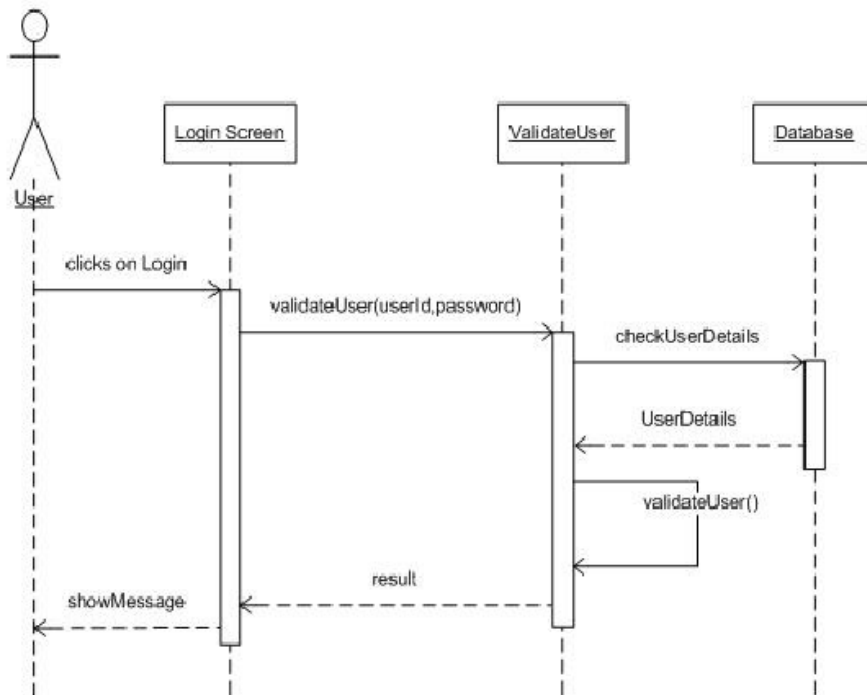


Figure 12: Sequence Diagram for login a user

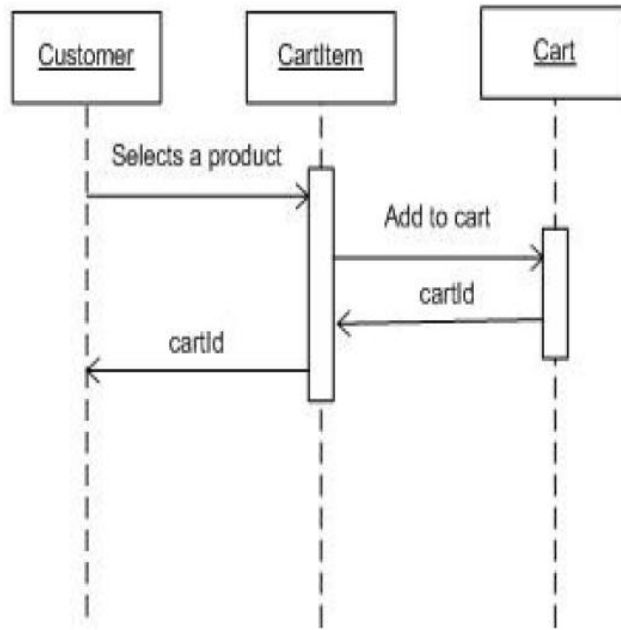


Figure 13: Sequence Diagram for add a cart

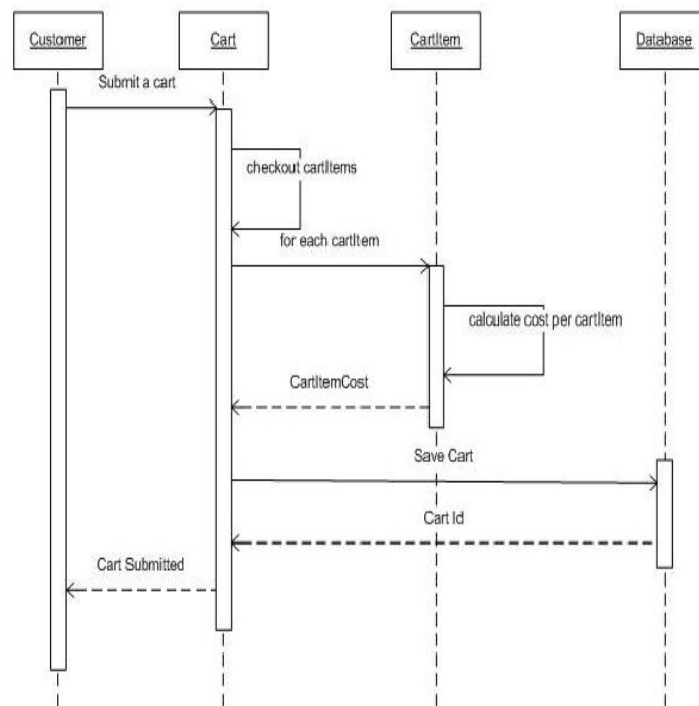


Figure 14: Sequence Diagram for submit a cart

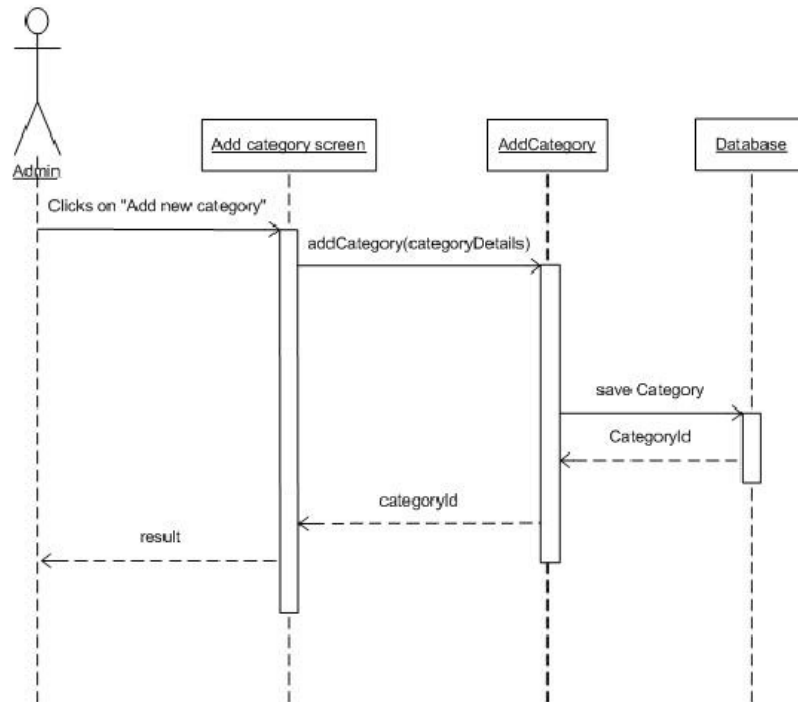


Figure 15: Sequence Diagram for add a new category

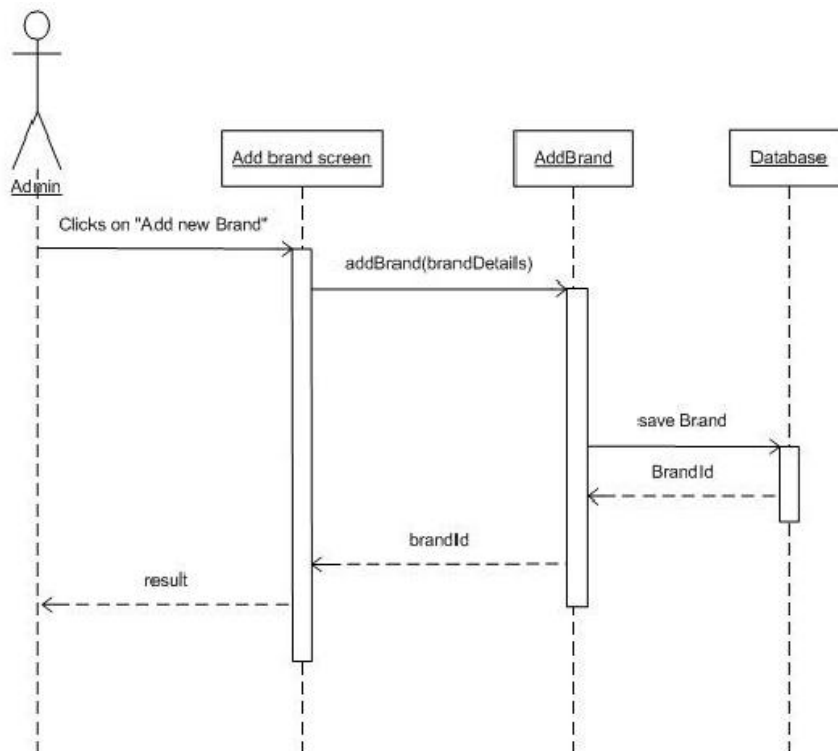


Figure 16: Sequence Diagram for add a new brand

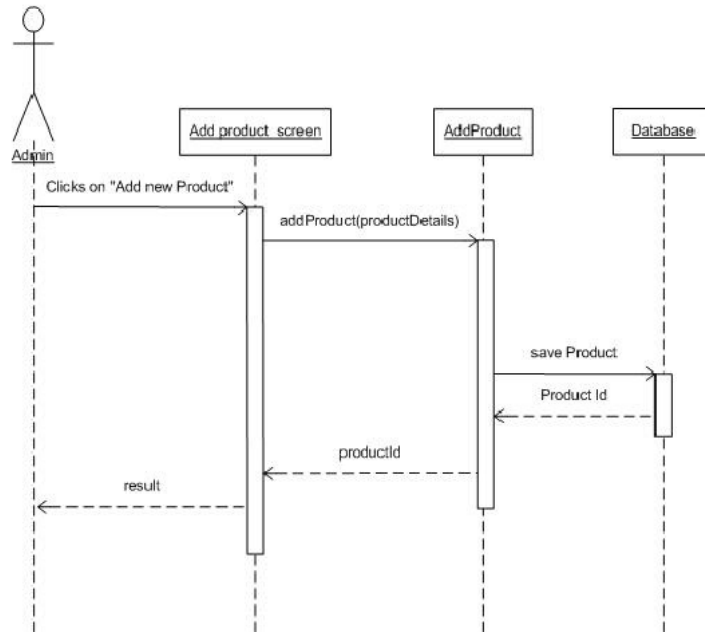


Figure 17: Sequence Diagram for add a new product

8. ADVANTAGES & DISADVANTAGES OF THE PROPOSED MODEL

8.1 Advantages

Significant IT Cost Reduction: As compare to the cost of implementing the traditional online shopping application in cloud is very very low, upfront capital expenditures eliminated, dramatically reduced IT admin burden and no extra capital expenditure as well.

Increased Utilization of IT Resources: Applications hosted in the Cloud can use Service Interfaces to access application-specific data. This data is stored “in the Cloud” and can be accessed by any component and any instance of the hosted application. The Data Services ensure reliable access, concurrency, backup. So our model helps to optimum use of cloud infrastructure by sharing resources across different application support, development, testing teams and business units.

High Flexibility: On-demand computing across technologies, business solutions and large ecosystems of providers; reduced new solution implementation times. Clients are capable to customize the application accordingly to their own business rules. So these applications could instantly scales to match your new requirements.

Access Anywhere Any IP Enable Devices: The online shopping application will access from a single computer or network, also from different computers, networks with enough security.

Easy Scalability & Highly Availability: Add & subtract capacity, switch over & switch back as per the requirement and Data backup & Recovery testing.

Easy to Implement & Maintain: No need to purchase hardware, software licenses or administration work. our model aims to minimize in-house IT infrastructure and support effort.

Service quality: Reliable services, large storage and computing capacity, and 24/7/365 service support and up-time.

Software License & Version: Always the latest software and patch updates are automatic in cloud computing

Integration with services that run outside the Cloud: Applications need to get access to resources other than those provided by the Cloud Services. These could be external services available on the internet – like a

payment, search or mapping service accessed via Web Service etc interfaces. It could also be accessing data from other applications that we run on-premise.

8.2 Disadvantages

Performance: One of the biggest challenges presented by the dynamic nature of the Cloud is troubleshooting performance issues. There are currently no good approaches to quickly identify the root cause of application performance issues in the Cloud. Also only this is possible overcommitted clouds.

Security: All Public cloud services run on virtual machines and a hypervisor allows multiple operating systems to share a single hardware host. If a hypervisor is flawed, it will negatively impact all the instance resources. This can significantly impact the availability of the cloud service.

9. CONCLUSION

There is tremendous potential for consumers to exploit the cloud computing technology. Availability of cloud based systems will be critical for its long-term success. It is not limited to availability of the applications being used, they also need to be delivered to the consumer in a consistent way without delays. Some of the basic properties such as scalability and flexibility are still important factors. Analyzing and proactively mitigating the risks and identifying single points of failure involved in the cloud can go a long way in achieving high availability of the cloud. In future, we plan to extend our work by introducing factors like platform security, technical standards, regulatory mechanisms and providing support towards the use of hybrid clouds.

REFERENCES

- [1] M. A. Morsy, J. Grundy, and I. Miller, "An Analysis of The Cloud Computing Security Problem," in Proc. APSEC, 2010 Cloud Workshop.
- [2] A.S. Ibrahim, J. Hamlyn-Harris, and J. Gurundy, "Emerging Security Challenges of Cloud Virtual Infrastructure," in Proc. APSEC, 2010 Cloud Workshop.
- [3] T. Ristenpart, E. Tromer, H. Shacham, and S. Savage, "Hey, You, Get Off of My Cloud: Exploring Information Leakage in Third-Party Compute Clouds", Proc. 16th ACM Conf. Computer and Communication Security (CCS 09), ACM Press, 2009, pp. 119-212
- [4] W. Dawoud, I. Takouna, and C. Meinel, "Infrastructure as a service security: Challenges and solutions," The 7th International Conference on Informatics and Systems (INFOS), pp.1-8, March. 2010.
- [5] H. Tsai, M.Siebenhaar, A. Miede, Y. Huang, and R. Steinmetz, "Threat as a Service? Virtualization's Impact on Cloud Security", IT Professional, vol. 14, no. 1, pp.32-37.
- [6] Liang Zhong; Tianyu Wo; Jianxin Li; Bo Li; "A Virtualization-based SaaS Enabling Architecture for Cloud Computing" Autonomic and Autonomous Systems (ICAS), 2010 Sixth International Conference
- [7] Ohlman, B.; Eriksson, A.; Rembarz, R.; "What Networking of Information Can Do for Cloud Computing" Enabling Technologies: Infrastructures for Collaborative Enterprises, 2009. WETICE '09. 18th IEEE International Workshops
- [8] Pandey, S.; Barker, A.; Gupta, K.K.; Buyya, R.; "Minimizing Execution Costs when using Globally Distributed Cloud Services " Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference
- [9] Xinhui Li; Ying Li; Tiancheng Liu; Jie Qiu; Fengchun Wang; "The Method and Tool of Cost Analysis for Cloud Computing" Cloud Computing, 2009. CLOUD '09. IEEE International Conference
- [10] Guohui Wang; Ng, T.S.E.; "The Impact of Virtualization on Network Performance of Amazon EC2 Data Center" INFOCOM, 2010 Proceedings IEEE
- [11] Wei-Tek Tsai; Xin Sun; Balasooriya, J.; "Service-Oriented Cloud Computing Architecture " Information Technology: New Generations (ITNG), 2010 Seventh International Conference
- [12] Zhang, Xiujuan; Dong, Guoqing; "A new architecture of online trading platform based on Cloud computing" Wearable Computing Systems (APWCS), 2010 Asia-Pacific Conference
- [13] Buyya R, Yeo CS, Venugopal S, Broberg J, Brandic I. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Generation Computer Systems 2009; 25(6):599–616.
- [14] ASHRAE Technical Committee 99. Datacom equipment power trends and cooling applications 2005.
- [15] Belady C. In the data center, power and cooling costs more than the it equipment it supports 2007. URL <http://www.electronics-cooling.com/articles/2007/feb/a3/>.
- [16] Barroso LA, Holzle U. The case for energy-proportional computing. Computer 2007; 40(12):33–37.
- [17] Fan X, Weber WD, Barroso LA. Power provisioning for a warehouse-sized computer. Proceedings of the 34th Annual International Symposium on Computer Architecture (ISCA 2007), ACM New York, NY, USA, 2007; 13–23.
- [18] de Assunao MD, Gelas JP, Lefevre L, Orgerie AC. The Green Grid'5000: Instrumenting and using a Grid

- with energy sensors. Proceedings of the 5th International Workshop on Distributed Cooperative Laboratories: Instrumenting the Grid (INGRID 2010), Poznan, Poland, 2010.
- [19] Ranganathan P, Leech P, Irwin D, Chase J. Ensemble-level power management for dense blade servers. Proceedings of the 33rd International Symposium on Computer Architecture (ISCA 2006), Boston, MA, USA, 2006; 66–77.
- [20] The green grid consortium 2011. URL <http://www.thegreengrid.org>.
- [21] Xen and the art of virtualization. Proceedings of the 19th ACM Symposium on Operating Systems Principles (SOSP 2003), Bolton Landing, NY, USA.
- [22] Clark C, Fraser K, Hand S, Hansen JG, Jul E, Limpach C, Pratt I, Warfield A. Live migration of virtual machines. Proceedings of the 2nd Symposium on Networked Systems Design and Implementation (NSDI 2005), USENIX, Boston, MA, USA, 2005.
- [23] Ben-David S, Borodin A, Karp R, Tardos G, Wigderson A. On the power of randomization in on-line algorithms. *Algorithmica* 1994; 11(1):214.
- [24] Nathuji R, Schwan K. Virtualpower: Coordinated power management in virtualized enterprise systems. *ACM SIGOPS Operating Systems Review* 2007; 41(6):265–278.
- [25] Kusic D, Kephart JO, Hanson JE, Kandasamy N, Jiang G. Power and performance management of virtualized computing environments via lookahead control. *Cluster Computing* 2009; 12(1):1–15.
- [26] Srikantiah S, Kansal A, Zhao F. Energy aware consolidation for cloud computing. *Cluster Computing* 2009; 12:1–15. Cardoso M, Korupolu M, Singh A. Shares and utilities based power consolidation in virtualized server environments. Proceedings of the 11th IFIP/IEEE Integrated Network Management (IM 2009), Long Island, NY, USA, 2009.
- [27] Verma A, Ahuja P, Neogi A. pMapper: Power and migration cost aware application placement in virtualized systems. Proceedings of the 9th ACM/IFIP/USENIX International Conference on Middleware (Middleware 2008), Springer, Leuven, Belgium, 2008; 243–264.
- [28] G. Kaefer, “Cloud Computing Architecture”, Corporate Research and Technologies, Munich, Germany, Siemens AG 2010, Corporate Technology, (2010).
- [29] A. Gopalakrishnan, “Cloud Computing Identity Management”, SETLabs Briefings, vol. 7, no. 7, (2009).
- [30] VeriSign, “Digital ID, A Brief Overview”, A VeriSign White Paper, 2004 VeriSign, <http://www.verisign.com/static/005326.pdf>
- [31] John Harauz et al., 2009 “Data security in the world of cloud computing”, IEEE Computer and Reliability Societies, pp. 61- 64
- [32] Balachandra Reddy Kandukrui et al, 2009, “Cloud Security Issues”, IEEE International Conference on Services Computing, pp. 51-520.
- [33] Weili Huang et al., 2009, “New Network Security Based on Cloud Computing”. 2nd IEEE International Workshops on Educational Technology and computer science, pp. 604 – 609
- [34] Zhang wei et al., 2010, “Two factor Worm Propagation Model in Cloud Security Network“, 2nd IEEE International Workshops on Network security, wireless communications and trust computing, pp. 385-388
- [35] Sameera Abdulrahman Almulla et al., 2010, “Cloud Computing Security Management “, 2nd IEEE International Workshops on Network security, wireless communications and trust computing, pp. 385-388
- [36] Gansan Zhao et al., 2010, “Deployment Models: Towards Eliminating Security Concerns from Cloud Computing “, IEEE International Conference., pp. 189 – 195.
- [37] S.Subashini et al., 2011, “A survey on security issues in service delivery models of cloud computing “, Journal of Network and Computer Applications 34 pp. 1-11.
- [38] http://en.wikipedia.org/wiki/Cloud_computing_security referred on (27.01.2012)
- [39] B. Schmidt-Wesche, Bleizeffer, T., Calcaterra, J., Nair, D Rendahl, R. and Sohn, P. Cloud User Roles: Establishing Standards for Describing Core Tasks of Cloud Creators, Providers, and Consumers, 2011 IEEE International Conference on Cloud Computing, 2011, pp. 764-765.
- [40] Microsoft White Paper. 2009. Microsoft Forefront Unified Access Gateway and Direct Access, Microsoft, 1-14.
- [41] S.D. Farnham and Churchill, E.F. Faceted Identity, Faceted Lives: Social and Technical Issues with Being Yourself Online, CSCW 2011, March 19–23, 2011, Hangzhou, China. 2011.
- [42] J.M. DiMicco, and Millen, D.R. Identity Management: Multiple Presentations of Self in Facebook. GROUP’2007, Sanibel Island, Florida. 2007.
- [43] W. Stuerzlinger, On- and Off-Line User Interfaces for Collaborative Cloud Services, International Conference of Human-computer Interaction CHI 2011, Vancouver, Canada. ACM. 2011.
- [44] Väänänen-Vainio-Mattila, K., Kaasinen, E. and Roto., V. 2011. User Experience in the Cloud: Towards a Research Agenda, International Conference of Human-computer Interaction CHI2011, Vancouver, Canada. ACM.
- [45] J. Elson, and Howell, J. Refactoring human roles solves systems problems, Conference on Hot Topics in Cloud Computing, Microsoft Research. 2009.
- [46] D. Thibeau, Enabling Citizen Involvement through Open Identity Technologies, OpenID White Paper, 2009, pp. 1-10.
- [47] A. Narayanan, and Shmatikov, V. Robust De-anonymization of Large Sparse Datasets, Proceedings of IEEE Symposium on Security and Privacy, IEEE Computer Society. 2008.
- [48] Y. Low, Gonzalez, J. Kyrola, A. Bickson, D. and Guestrin, C., GraphLab: A Distributed Framework for Machine Learning in the Cloud. arXiv:1107.0922v1, 2011, pp. 1-14.
- [49] V. Kundra, Federal Cloud Computing Strategy. Chief Information Officers Council Report, USA, 2011. pp. 1-43.

- [50] Cisco White Paper. Cisco Intelligent Automation for Cloud, Cisco Data Sheet, 2011, pp. 1-9.
- [51] Wayne Jansen, Timothy Grance, "Guidelines on Security and Privacy in Public Cloud Computing", National Institute of Standards and Technology Gaithersburg, January 2011.
- [52] Jeep Ruiter, Martijn Warnier, "Privacy Regulations for Cloud Computing", Faculty of Sciences, VU University Amsterdam.
- [53] Dancho Danchev, "Building and Implementing a successful Information Security Policy" windowsecurity.com-Windows Security Resources for IT admins.
- [54] Anthony T. Velte, Toby J. Velte, Robert Elsenpeter, Cloud Computing A Practical Approach, TATA McGRAW-HILL Edition 2010.
- [55] Martin Randles, David Lamb, A. Taleb-Bendiab, A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing, 2010 IEEE 24th International Conference on Advanced Information Networking and Applications Workshops.
- [56] Richard N. Katz, "Looking at Clouds from All Sides Now", EDUCAUSE Review, vol. 45, no. 3 (May/June 2010): 32-45.
- [57] Mladen A. Vouk, Cloud Computing Issues, Research and Implementations, Proceedings of the ITI 2008 30th Int. Conf. on Information Technology Interfaces, 2008, June 23-26.
- [58] Ali M. Alakeel, A Guide to Dynamic Load Balancing in Distributed Computer Systems, IJCSNS International Journal of Computer Science and Network Security, VOL.10 No.6, June 2010.
- [59] Sandeep Sharma, Sarabjit Singh, Meenakshi Sharma "Performance Analysis of Load Balancing Algorithms", World Academy of Science, Engineering and Technology, 38, 2008 pp. 269- 272.
- [60] Luqun Li, "An Optimistic Differentiated Service Job Scheduling System for Cloud Computing Service Users and Providers", Third International Conference on Multimedia and Ubiquitous Engineering, IEEE Explore, 4-6, June 2009, pp. 295-299.
- [61] D. Yuan, Y. Yang, X. Liu, J. Chen, A cost-effective strategy for intermediate data storage in scientific cloud workflows, in: 24th IEEE International Parallel & Distributed Processing Symposium, IPDPS'10, Atlanta, Georgia, USA, 2010, pp. 1-12.
- [62] D. Kondo, B. Javadi, P. Malecot, F. Cappello, D.P. Anderson, Cost-benefit analysis of cloud computing versus desktop grids, in: IEEE International Symposium on Parallel & Distributed Processing, IPDPS'09, Rome, Italy, 2009, pp. 1-12.
- [63] K. Keahey, R. Figueiredo, J. Fortes, T. Freeman, M. Tsugawa, Science clouds: early experiences in cloud computing for scientific applications, in: First Workshop on Cloud Computing and its Applications, CCA'08, Chicago, Illinois, USA, 2008, pp. 1-6.
- [64] Buyya, R., Yeo, C. S., & Venugopal, S. (2009). Market-oriented cloud computing: Vision, hype, and reality of delivering IT services as computing utilities. In The 10th IEEE international conference on high performance computing and communications (pp. 5-13).
- [65] Lin, G., Fu, D., Zhu, J., & Dasmalchi, G. (2009). Cloud computing: IT as a service. IT Professional, 11(2), 10-13.

BIOGRAPHY OF AUTHORS



Rabi Prasad Padhy is currently working as a Senior Software Engineer - Oracle India Private Ltd. Bangalore, Karnataka, India. He has achieved his MCA degree from Berhampur University. He carries more than 8 years of extensive IT Experience with MNC's like EDS, Dell, IBM and Oracle. His area of interests includes IT Infrastructure Optimization, Virtualization, Enterprise Grid Computing, Cloud Computing and Cloud Databases. He has published several research papers in national and international journals. He is a certified professional for Oracle & Microsoft Database, Oracle Apps, Linux & Solaris System Admin and also ITIL Certified.



Dr. Manas Ranjan Patra holds a Ph.D. Degree in Computer Science from the Central University of Hyderabad, India. Currently he is an Associate Professor in the Post Graduate Department of Computer Science, Berhampur University, India. He has about 24 years of experience in teaching and research in different areas of Computer Science. He had visiting assignment to International Institute for Software Technology, Macao as a United Nations Fellow and for sometime worked as assistant professor in the Institute for Development and Research in Banking Technology, Hyderabad. He has about 90 publications to his credit. His research interests include Service Oriented Computing, Software Engineering, Applications of Data mining and E-Governance. He has presented papers, chaired technical sessions and served in the technical committees of many International conferences.