Clouds for Different Services

Sanjeev Narayan Bal
Dept.of Comp.Sc.
TACT, Bhubaneswar

Abstract

The Cloud has become a new vehicle for delivering resources such as computing and storage to customers on demand. Rather than being a new technology in itself, the cloud is a new business model wrapped around new technologies such as server visualization that take advantage of economies of scale and multi-tenancy to reduce the cost of using information technology resources. This paper discusses the business drivers in the Cloud delivery mechanism, cloud models and cloud architectures, what the requirements are in this space, and how standard interfaces, coordinated between different organizations can meet the emerging needs for interoperability and portability of data between clouds. Cloud is a style of computing where scalable and elastic IT-related capabilities are provided as a service to external customers using internet technologies.

Keywords :Cloud Computing, Cloud Data Management Interface, Open Cloud Computing Interface, Amazon SQS.

1. Introduction

What is 'cloud computing'--Many people are confused as to exactly what cloud computing is, especially as the term can be used to mean almost anything. Roughly, it describes highly scalable computing resources provided as an external service via the internet on a pay-as-you-go basis. The cloud is simply a metaphor for the internet, based on the symbol used to represent the worldwide network in computer network diagrams.

Economically, the main appeal of cloud computing is that customers only use what they need, and only pay for what they actually use. Resources are available to be accessed from the cloud at any time, and from any location via the internet. There's no need to worry about how things are being maintained behind the scenes. you simply purchase the IT service you require as you would any other utility. Because of this, cloud computing has also been called utility computing, or 'IT on demand'. This new, web-based generation of computing utilizes remote servers housed in highly secure data centres for data storage and management, so organizations no longer need to purchase and look after their IT solutions in-house.

2. Understanding Public and Private Clouds

Enterprises can choose to deploy applications on Public, Private or Hybrid clouds. Cloud Integrators can play a vital part in determining the right cloud path for each organization.

2.1 Public Cloud

Public cloud (also referred to as 'external' cloud) describes the conventional meaning of cloud computing: scalable, dynamically provisioned, often virtualized resources available over the Internet from an off-site third-party provider, which divides up resources and bills its customers on a 'utility' basis. An example is Think Grid, a company that provides a multi-tenant architecture for supplying services such as Hosted Desktops, Software as a Service and Platform as a Service. Other popular cloud vendors include Salesforce.com, Amazon EC2 and Flexi scale.

2.2 Private Cloud

Private cloud (also referred to as 'corporate' or 'internal' cloud) is a term used to denote a proprietary computing architecture providing hosted services on private networks. This type of cloud computing is generally used by large companies, and allows their corporate network and data centre administrators to effectively become in-house 'service providers' catering to 'customers' within the corporation. However, it negates many of the benefits of cloud computing, as organizations still need to purchase, set up and manage their own clouds.

2.3 Hybrid Cloud

It has been suggested that a hybrid cloud environment combining resources from both internal and external providers will become the most popular choice for enterprises. For example, a company could choose to use a public cloud service for general computing, but store its business-critical data within its own data centre. This may be because larger organizations are likely to have already invested heavily in the infrastructure required to provide resources in-house or they may be concerned about the security of public clouds .

IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 4, No 1, July 2012
ISSN (Online): 1694-0814
www.IJCSI.org

3. Cloud Computing Models

Let us consider some of the basics of cloud computing models with the aspects such as:

- Realities and risks of the model
- Components in the model
- Characteristics and Usage of the model

Cloud Providers offer services that can be grouped into three categories.

Software as a Service (SaaS)-

In this model, a complete application is offered to the customer, as a service on demand. A single instance of the service runs on the cloud & multiple end users are serviced. On the customers' side, there is no need for upfront investment in servers or software licenses, while for the provider, the costs are lowered, since only a single application needs to be hosted & maintained. Today SaaS is offered by companies such as Google,Salesforce, Microsoft, Zoho, etc.

Platform as a Service (Paas)-

Here, a layer of software, or development environment is encapsulated & offered as a service, upon which other higher levels of service can be built. The customer has the freedom to build his own applications, which run on the provider's infrastructure. To meet manageability and scalability requirements of the applications.

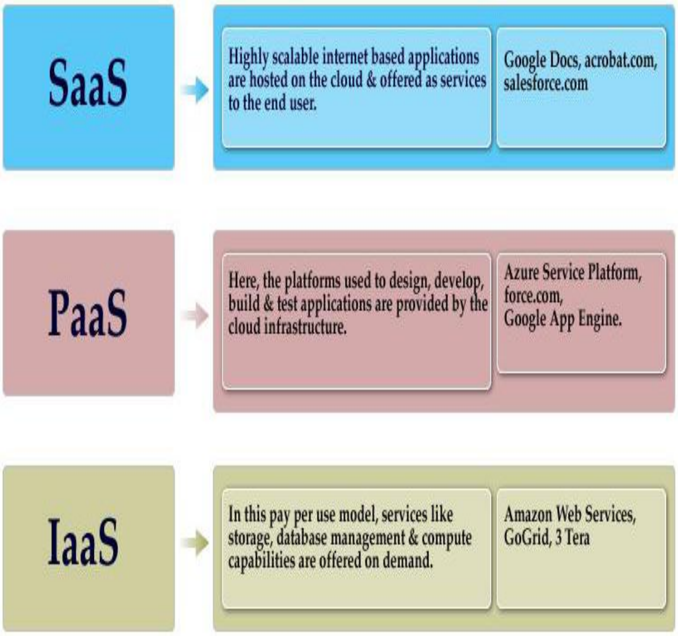

Figure 1: Cloud models

PaaS providers offer a predefined combination of OS and application servers, such as LAMP platform (Linux. Apache. MySql and PHP), restricted J2EE, Ruby etc. Google's App Engine, Force.com,etc are some of the popular PaaS examples.

Infrastructure as a Service (laas)-

IaaS provides basic storage and computing capabilities as standardized services over the network. Servers, storage systems, networking equipment, data centre space etc. are pooled and made available to handle workloads. The customer would typically deploy his own software on the infrastructure. Some common examples are Amazon,GoGrid, 3Tera, etc.

Standardizing Cloud Computing Interfaces-

Having a programmable interface to the IaaS infrastructure means that you can write client software that uses this interface to manage your use of the Cloud. Many cloud providers have licensed their proprietary APIs freely allowing anyone to implement a similar cloud infrastructure. Despite the accessibility of open APIs, cloud community members have been slow to uniformly adopt any proprietary interface controlled by a single company. The Open Source community has attempted responses, but this has done little to stem the tide of API proliferation. In fact, Open Source projects have increased the tally of interfaces to navigate in a torrent of proprietary APIs.What is needed instead is a vendor neutral, standard API for cloud computing that all vendors can implement with minimal risk and assured stability. This will allow customers to move their application stacks from one cloud vendor to another, avoiding lock-in and reducing costs.

The Open Grid Forum has created a working group to standardize such an interface. The Open Cloud Computing Interface (OCCI) is a free, open, community consensus driven API, targeting cloud infrastructure services. The API shields IT data centers and cloud partners from the disparities existing between the lineup of proprietary and open cloud APIs.

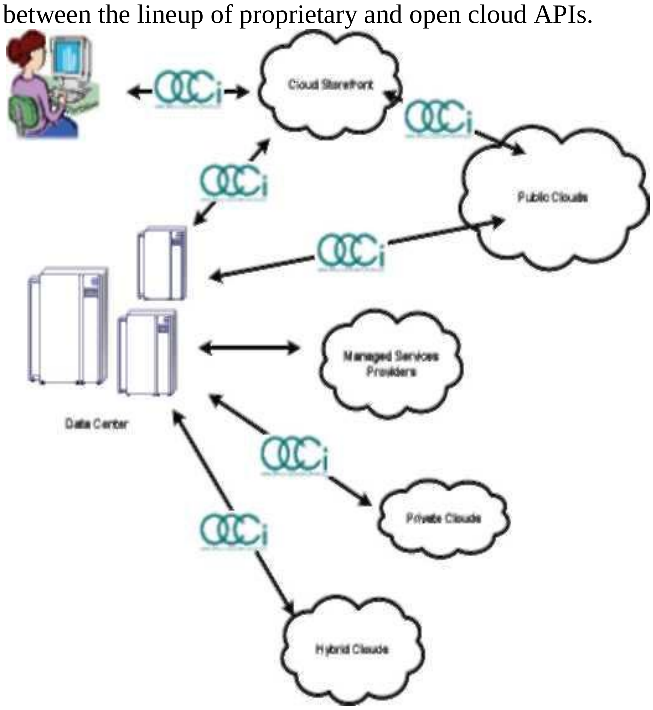

Figure 2: OCCI API

The Storage Networking Industry Association has created a technical work group to address the need for a cloud storage standard. The new Cloud Data Management Interface (CDMI) is meant to enable interoperable cloud storage and data management. In CDMI, the underlying storage space exposed by the above interfaces is abstracted using the notion of a container. A container is not only a useful abstraction for storage space, but also serves as a grouping of the data stored in it, and a point of control for applying data services in the aggregate.

Using CDMI and OCCI for a Cloud Computing Infrastructure

CDMI Containers are accessible not only via CDMI as a data path, but other protocols as well. This is especially useful for using CDMI as the storage interface for a cloud computing environment as shown in Figure 4 below.

The exported CDMI containers can be used by the Virtual Machines in the Cloud Computing environment as virtual disks on each guest as shown. With the internal knowledge of the network and the Virtual Machine, the cloud infrastructure management application can attach exported CDMI containers to the Virtual Machines.

How it works-

The cloud computing infrastructure management shown above supports both OCCI and CDMI interfaces. To achieve interoperable, CDMI provides a type of export that contains information obtained via the OCCI interface. In addition, OCCI provides a type of storage that corresponds to exported CDMI containers.

OCCI and CDMI can achieve interoperability initiating storage export configurations from either OCCI or CDMI interfaces as starting points

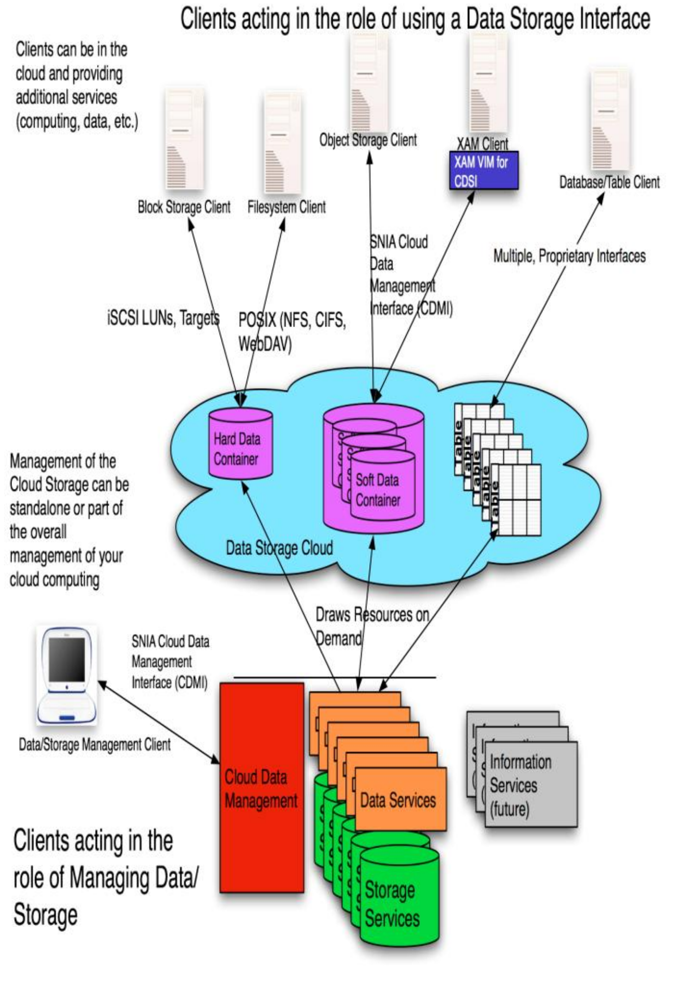

Figure 3: The Cloud Storage Reference Model

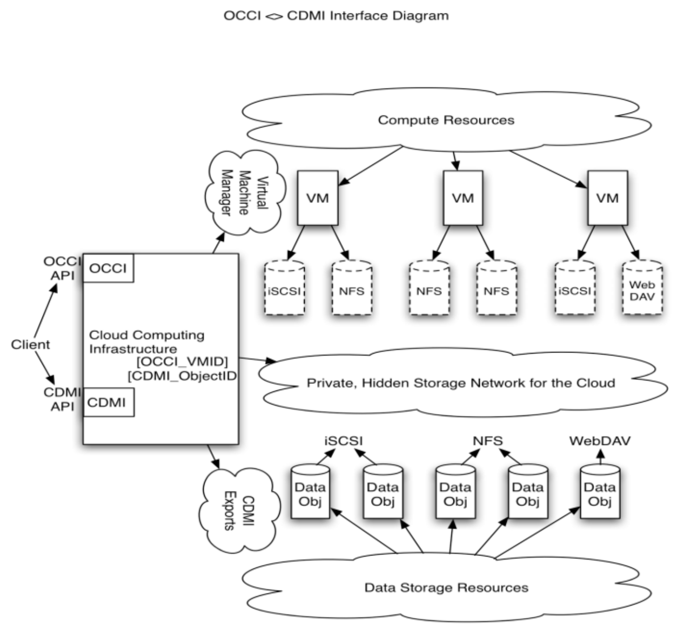

Figure 4: CDMI and OCCI in an integrated cloud computing environment

4. Enterprise Private Clouds

In order to overcome these challenges, organizations are looking at enterprise private cloud offerings. Enterprise private cloud solutions help organizations leverage the existing IT environment and create a cloud computing platform in the private internal network. This model overcomes several challenges faced in public cloud adoption. Enterprise private clouds are seen as a natural progression of initiatives like visualization already taken up by several organizations. Enterprise private cloud solutions add capabilities like self-service, automation and charge back over the virtualized infrastructure. Figure 5 provides the recommended logical architecture for an enterprise private cloud.

Self Service

The private cloud solution should have a self service portal that enables users request infrastructure and platforms as a service. It should contain a service catalog that lists the categories and the services available, the associated SLAs and costs. The service portal should enable reserving as well as requesting the services on demand.

Automation

The private cloud solution should have certain traits -

■ A provisioning engine that automates the provisioning of the infrastructure

■ Workflow driven with built-in approval mechanisms enabling governance

■ Enable user management and integration with enterprise authentication and authorization mechanisms

■ Enable enforcing enterprise policies on resource allocation through a rules engine

■ Enable capturing the common deployment patterns using templates. Self-service and automation helps reduce the time-to-market so that users can request for infrastructure as a service and can get it provisioned on demand.

Management and Monitoring--

The private cloud solution should also have an integrated monitoring and management platform that should have the following components —

Monitoring and Management: Track various metrics at the software arid infrastructure level

Metering & Chargeback: Track the usage of the various services and allow to charge back mechanisms to be plugged in SLA Management: Enable, define and monitor SLAs for the services

Patch Management: Enable patches to be rolled out to the various software components used

Reports: Generate reports on usage, SLA adherence, etc.

hidden t Management: Generate alerts when there are issues and provide ticketing mechanism to track arid resolve incidents.

Visualization

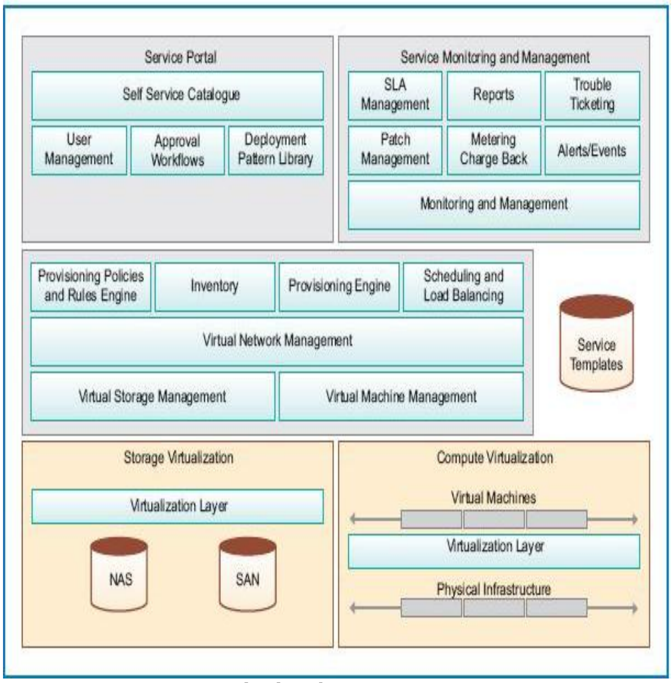

Figure 5:Enterprise private cloud Architecture

The private cloud solution should have visualization layer that virtualizes the key infrastructure components including compute, storage and network

Cloud Architectures are designs of software applications that use Internet-accessible on-demand services. Applications built on Cloud Architectures are such that the underlying computing infrastructure is used only when it is needed (for example to process a user request), draw the necessary resources on-demand (like compute servers or storage), perform a specific job, then relinquish the unneeded resources and often dispose themselves after the job is done. While in operation the application scales up or down elastically based on resource needs.

Examples of Cloud Architectures-

There are plenty of examples of applications that could utilize the power of Cloud Architectures. These range from back-office bulk processing systems to web applications. Some are listed below:

Processing Pipelines

♦ Document processing pipelines – convert hundreds of thousands of documents from Microsoft Word to PDF, OCR millions of pages/images into raw searchable text

♦ Image processing pipelines - create thumbnails or low resolution variants of an image, resize millions of images

♦ Video transcoding pipelines - transcode AVI to MPEG movies

♦ Indexing - create an index of web crawl data

♦ Data mining - perform search over millions of records Batch Processing Systems

♦ Back-office applications (in financial, insurance or retail sectors)

♦ Log analysis - analyze and generate daily/weekly reports

♦ Nightly builds - perform nightly automated builds of source code repository every night in parallel

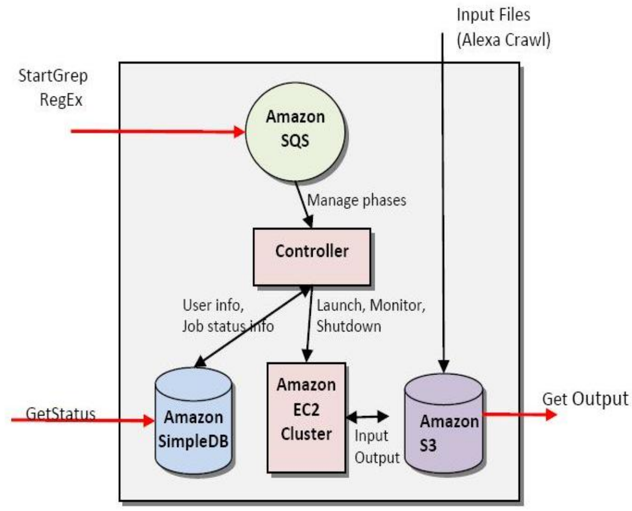

Figure 6: GrepTheWeb Architecture

♦ Automated Unit Testing and Deployment Testing - Test and deploy and perform automated unit testing (functional, load, quality) on different deployment configurations every night

Websites

♦ Websites that "sleep" at night and auto-scale during the clay

♦ Instant Websites - websites for conferences or events (Super Bowl, sports tournaments)

♦ Promotion websites

♦ "Seasonal Websites" - websites that only run during the tax season or the holiday season ("Black Friday" or Christmas)

Cloud Architecture Example: GrepTheWeb The Alexa Web Search web service allows developers to build customized search engines against the massive data that Alexa crawls every night. One of the features of their web service allows users to query the Alexa search index and get Million Search Results (MSR) back as output. Developers can run queries that return up to 10 million results. The resulting set, which represents a small subset of all the documents on the web, can then be processed further using a regular expression language.

This allows developers to filter their search results using criteria that are not indexed by Alexa thereby giving the developer power to do more sophisticated searches. Developers can run regular expressions against the actual documents, even when there are millions of them, to search for patterns and retrieve the subset of documents that attached that regular expression.
This application is currently in production at Amazon.com and is code-named GrepTheWeb.
Amazon S3 for retrieving input datasets and for storing the output dataset, Amazon SQS for durably buffering requests acting as a "glue" between controllers Amazon Simple for storing intermediate status, log, and for user data about tasks, Amazon EC2 for running a large distributed processing Hadoop cluster on-demand,Hadoop for distributed processing, automatic parallelization, and job scheduling.

5. Questions For The Cloud

The following is a non-definitive list of questions which should be asked of any cloud service provider. This list is meant as a starting point and should not be considered a complete list of questions. Ask that the potential cloud service provider provide answers in writing and where applicable in contract.

(i) UNDERLYING HARDWARE & SOFTWARE PLATFORMS
Describe the physical and logical architecture of the cloud environment.
What are the hardware & software components?
Describe the facility that the technology hosts the technology.
(ii) ABILITY TO INTERMIX PHYSICAL AND VIRTUAL/ CLOUD PLATFORMS
Can clients colocate the system they don't want in the cloud?
What are the cloud provider's limitations on power and space?
What are the interconnect costs for bandwidth between the colocation and the cloud?
(iii) PERFORMANCE METRIC CRITERIA
What performance metrics are monitored by the cloud provider? Does the client have access to view those metrics?
What happens when the client or other clients in the environment need more CPU or Disk 10? How are the individual environments protected from one another for these variable resources? is the SLA written to address performance issues as well as availability issues?
(iv) ENTERPRISE GRADE SERVICE & SUPPORT
Describe the technical talent that is available to support issues.
What support is provided if the client feels an issue may be cloud related?
Is the SLA written to address performance issues of the support organization?
(v) STANDARD APIS & ACCESS INTERFACES
Does the cloud provider make available APIs to manage the cloud environment?
Are those APIs standard and can any other providers use them?
(vi) AUDITS
What will the cloud provider provide in support of audits that the environment is subject to? What are the audits that the provider has completed in the last year? What was the outcome of those audits and are those records available for review?

References

[1] James Staten, Deliver Cloud Benefits inside Your Walls, Forrester, April 2009.Available at http://www.forrester.com /Research/Document/Excerpt/0,7211,54035,00.html
[2]John Foley, 20 Cloud Computing Startups You Should Know, Information Week, September 2008. Available at http://www.informationweek.com/news/software/hosted/show Article.jhtml?articleID=210602537
[3] Private Cloud Computing for Enterprises: Meet the Demands of High Utilization and Rapid Change. Available at http://www.cisco.com/en/US/solutions/collateral/ns340/ns517/ ns224/ns836/ns976/white_paper_c11-543729.html
[4] Daniel Nurmi, Rich Wolski, Chris Grzegorczyk Graziano Obertelli, Sunil Soman, Lamia Youseff and Dmitrii Zagorodnov, The Eucalyptus Open source Cloud-computing System University of California , Santa Barbara. Available at http://open.eucalyptus.com/documents/nurmi_ Et_al-eucalyptus_open_source_cloud_computing_system- cca_2008.pdf.

Author
Sanjeev Narayan Bal
MBA, MCA MTech(CSE)
Assistant Professor (CSE)
Trident Academy of Creative Technology
Life Member - ISTE