Self-Services and Broad Network Access in Cloud Computing Stack

. Dr U Ravi Babu

[1]Professor, Department of CSE, Narsimha Reddy Engg. Colleg, HYD, TS, India
uppu.ravibabu@gmail.com

ABSTRACT

As with other significant developments in technology, many vendors have seized the term "Cloud" and are using it for products that sit outside of the common definition. In order to truly understand how the Cloud can be of value to an organization, it is first important to understand what the Cloud really is and its different components. Since the Cloud is a broad collection of services, organizations can choose where, when, and how they use Cloud Computing. In this report we will explain the different types of Cloud Computing services commonly referred to as Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) and give some examples and case studies to illustrate how they all work. Cloud Computing is a broad term that describes a broad range of services. Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

Keywords: On-demand self-service, Broad network access, Resource pooling, Measured Service**.**

I. INTRODUCTION

Cloud Computing is often described as a stack, as a response to the broad range of services built on top of one another under the moniker "Cloud". The generally accepted definition of Cloud Computing comes from the National Institute of Standards and Technology (NIST) [1]. NIST also offers up several characteristics that it sees as essential for a service to be considered "Cloud". These characteristics include;

Cloud Computing is often described as a stack, as a response to the broad range of services built on top of one another under the moniker "Cloud". The generally accepted definition of Cloud Computing comes from the National Institute of Standards and Technology (NIST), essentially says that; Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.

• On-demand self-service. The ability for an end user to sign up and receive services without the long delays that have characterized traditional IT.

• Broad network access. Ability to access the service via standard platforms (desktop, laptop, mobile etc).

• Resource pooling. Resources are pooled across multiple customers [3].

• Rapid elasticity. Capability can scale to cope with demand peaks [4].

• Measured Service. Billing is metered and delivered as a utility service [5].

More than a semantic argument around categorization, we believe that in order to maximize the benefits that Cloud Computing brings, a solution needs to demonstrate these particular characteristics. This is especially true since in recent years there has been a move by traditional software vendors to market solutions as "Cloud Computing" which are generally accepted to not fall within the definition of true Cloud Computing, a practice known as

"cloud-washing." The diagram below depicts the Cloud Computing stack – it shows three distinct categories within Cloud Computing: Software as a Service, Platform as a

Service and Infrastructure as a Service.

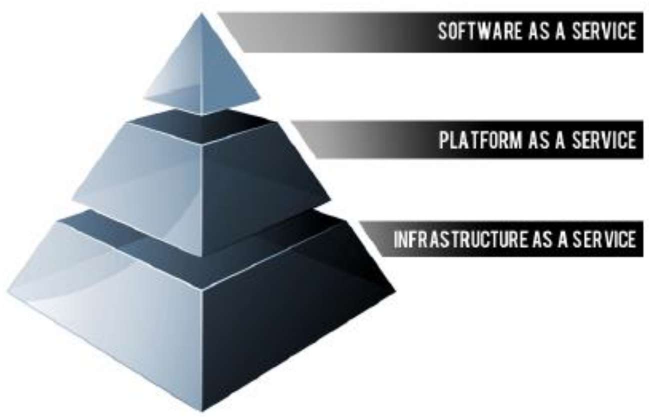


Fig-1: cloud computing stack diagram

All three categories in detail however a very simplified way of differentiating these flavors of Cloud Computing is as follows;

• SaaS applications are designed for end-users, delivered over the web

• PaaS is the set of tools and services designed to make coding and deploying those applications quick and efficient

• IaaS is the hardware and software that powers it all – servers, storage, networks, operating systems.

2. Cloud Computing Stack (SaaS, PaaS, IaaS)

Where SaaS Makes Sense

• "Vanilla" offerings where the solution is largely undifferentiated. A good example of a vanilla offering would include email where many times competitors use the same software precisely because this fundamental technology is a requirement for doing business, but does not itself confer an competitive advantage

• Applications where there is significant interplay between the organization and the outside world. For example, email newsletter campaign software

• Applications that have a significant need for web or mobile access. An example would be mobile sales management software

• Software that is only to be used for a short term need. An example would be collaboration software for a specific project

• Software where demand spikes significantly, for example tax or billing software used once a month.

Where PaaS Makes Sense

• Applications where extremely fast processing of real time data is required

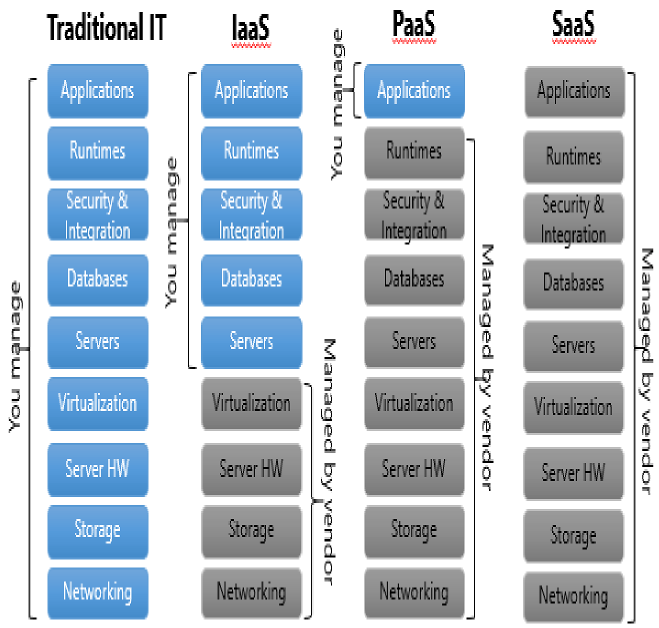
• Applications where legislation or other regulation does not permit data being hosted externally

• Applications where an existing on-premise solution fulfills all of the organization's needs

Software as a Service may be the best known aspect of Cloud Computing, but developers and organizations all around the world are leveraging Platform as a Service, which mixes the simplicity of SaaS with the power of IaaS, to great effect.

Fig* 2: cloud service models

•Applications that have a significant need for web or mobile access. An example would be mobile sales management software



•Software that is only to be used for a short term need. An example would be collaboration software for a specific project

•Software where demand spikes significantly, for example tax or billing software used once a month.

Where IaaS Makes Sense

IaaS makes sense in a number of situations and these are closely related to the benefits that Cloud Computing bring. Situations that are particularly suitable for Cloud infrastructure include;

• Where demand is very volatile – any time there are significant spikes and troughs in terms of demand on the infrastructure

• For new organizations without the capital to invest in hardware

• Where the organization is growing rapidly and scaling hardware would be problematic

• Where there is pressure on the organization to limit capital expenditure and to move to operating expenditure

• For specific line of business, trial or temporary infrastructural needs.

3. Open Stack Compute

Cloud Computing is a term that doesn't describe a single thing – rather it is a general term that sits over a variety of services from Infrastructure as a Service at the base, through Platform as a Service as a development tool and through to Software as a Service replacing on-premise applications.

For organizations looking to move to Cloud Computing, it is important to understand the different aspects of Cloud Computing and to assess their own situation and decide which types of solutions are appropriate for their unique needs.

The OpenStack cloud operating system enables enterprises and service providers to offer on-demand computing resources, by provisioning and managing large networks of virtual machines. Compute resources are accessible via APIs for developers building cloud applications and via web interfaces for administrators and users. The compute architecture is designed to scale horizontally on standard hardware, enabling the cloud economics companies have come to expect.

Flexible Architecture: OpenStack is architected to provide flexibility as you design your cloud, with no proprietary hardware or software requirements and the ability to integrate with legacy systems and third party technologies. It is designed to manage and automate pools of compute resources and can work with widely available virtualization technologies, as well as bare metal and high-performance computing (HPC) configurations[4].

Popular Use Cases: Service providers offering an IaaS compute platform or services higher up the stack.

- **IT departments** acting as cloud service providers for business units and project teams
- **Processing** big data with tools like Hadoop
- **Scaling** compute up and down to meet demand for web resources and applications
- **High-performance** computing (HPC) environments processing diverse and intensive workloads.

Open Stack Storage: Many organizations now have a variety of storage needs with varying performance and price requirements. Open Stack has support for both Object Storage and Block Storage, with many deployment options for each depending on the use case. Object Storage is ideal for cost effective, scale-out storage. It provides a fully distributed, API-accessible storage platform that can be integrated directly into applications or used for backup, archiving and data retention. Block Storage allows block devices to be exposed and connected to compute instances for expanded storage, better performance and integration with enterprise storage platforms, such as NetApp, Nexenta and SolidFire[3].

Object Storage Capabilities:

OpenStack provides redundant, scalable object storage using clusters of standardized servers capable of storing petabytes of data

Object Storage is not a traditional file system, but rather a distributed storage system for static data such as virtual machine images, photo storage, email storage, backups and archives. Having no central "brain" or master point of control provides greater scalability, redundancy and durability.

Objects and files are written to multiple disk drives spread throughout servers in the data center, with the OpenStack software responsible for ensuring data replication and integrity across the cluster.

Storage clusters scale horizontally simply by adding new servers. Should a server or hard drive fail, OpenStack replicates its content from other active nodes to new locations in the cluster. Because OpenStack uses software logic to ensure data replication and distribution across different devices, inexpensive commodity hard drives and servers can be used in lieu of more expensive equipment.

Block Storage Capabilities:

OpenStack provides persistent block level storage devices for use with OpenStack compute instances.The block storage system manages the creation, attaching and detaching of the block devices to servers. Block storage volumes are fully integrated into OpenStack Compute and the Dashboard allowing for cloud users to manage their own storage needs.In addition to using simple Linux server storage, it has unified storage support for numerous storage platforms including Ceph, NetApp, Nexenta, SolidFire, and Zadara.

.Block storage is appropriate for performance sensitive scenarios such as database storage, expandable file systems, or providing a server with access to raw block level storage[2].

.Snapshot management provides powerful functionality for backing up data stored on block storage volumes. Snapshots can be restored or used to create a new block storage volume.

4. STRUCTIURE AND NETWORKING

Today's datacenter networks contain more devices than ever before— servers, network equipment, storage systems and security appliances — —many of which are further divided into virtual machines and virtual networks. The number of IP addresses, routing configurations and security rules can quickly grow into the millions. Traditional network management techniques fall short of providing a truly scalable, automated

Open Stack Networking is a pluggable, scalable and API-driven system for managing networks and IP addresses. Like other aspects of the cloud operating system, it can be used by administrators and users to increase the value of existing datacenter assets. Open Stack Networking ensures the network will not be the bottleneck or limiting factor in a cloud deployment

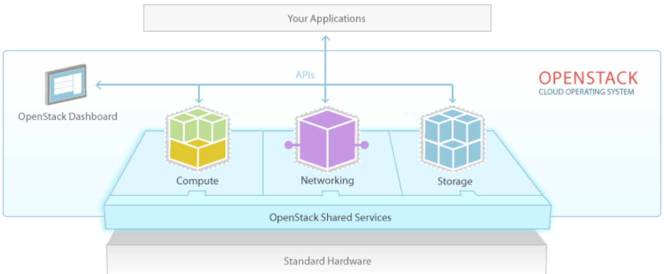


Fig-3 Open Stack Model

Networking Capabilities: OpenStack provides flexible networking models to suit the needs of different applications or user groups. Standard models include flat networks or VLANs for separation of servers and traffic.

OpenStack Networking manages IP addresses, allowing for dedicated static IPs or DHCP. Floating IPs allow traffic to be dynamically rerouted to any of your compute resources, which allows you to redirect traffic during maintenance or in the case of failure. Users can create their own networks, control traffic and connect servers and devices to one or more networks.

The pluggable backend architecture lets users take advantage of commodity gear or advanced networking services from supported vendors.

Administrators can take advantage of software-defined networking (SDN) technology like OpenFlow to allow for high levels of multi-tenancy and massive scale.

OpenStack Networking has an extension framework allowing additional network services, such as intrusion detection systems (IDS), load balancing, firewalls and virtual private networks (VPN) to be deployed and managed.

OpenStack Shared Services :

OpenStack has several shared services that span the three pillars of compute, storage and networking, making it easier to implement and operate your cloud. These services — including identity, image management and a web interface— — integrate the OpenStack components with each other as well as external systems to provide a unified experience for users as they interact with different cloud resources.

Identity Service: OpenStack Identity provides a central directory of users mapped to the OpenStack services they can access. It acts as a common authentication system across the cloud operating system and can integrate with existing backend directory services like LDAP. It supports multiple forms of authentication including standard username and password credentials, token-based systems and AWS-style logins.

Database Service: Designed to run entirely on OpenStack, the service has the goal of allowing users to quickly and easily utilize the features of a relational database without the burden of handling complex administrative tasks. Cloud users and database administrators can provision and manage multiple database instances as needed. Initially, the service will focus on providing resource isolati on at high performance while automating complex administrative tasks including deployment, configuration, patching, backups, restores, and monitoring.

V. CONCLUSION AND FUTURE WORK

The Open Cloud Computing Interface comprises a set of open community-lead specifications delivered through the Open Grid Forum. OCCI is a Protocol and API for all kinds of Management tasks. OCCI was originally initiated to create a remote management API for IaaS model based Services, allowing for the development of interoperable tools for common tasks including deployment, autonomic scaling and monitoring. It has since evolved into a flexible API with a strong focus on integration, portability, interoperability and innovation while still offering a high degree of extensibility. The current release of the Open Cloud Computing Interface is suitable to serve many other models in addition to IaaS, including e.g. PaaS and SaaS.

mining, cloud computing, Computer Networks.

REFERENCES

[1] M Armbrust et al., "Above the clouds: A berkeley view of cloud computing," University of California, Berkeley, Tech. Rep., Feb. 2009.

[2] S. Das, D. Agrawal, and A. El Abbadi, "Elastras: an elastic transactional data store in the cloud," in USENIX HotCloud, 2009.

[3] D. J. Abadi, "Data management in the cloud: Limitations and opportu-nities," IEEE Data Engineering Bulletin, Mar. 2009.

[4] A. J. Lee and M. Winslett, "Safety and consistency in policy-based authorization systems," in ACM CCS, 2006.

[5] M. ITU-T Technology Watch Report. (2009). Distributed Commuting: Utilities, Grid & Clouds. Retrieved from www.itu.int/dms_pub/itut/oth/23/01/T23010000090001PDFE.pdf0.

[6] Scalability and fast provisioning – for IT at web scale - http://broadcast.rackspace.com/hosting knowledge/ whitepapers /Revolution_Not_Evolution-Whitepaper.pdf

[7] From Water-wheel to Utility Power – An analogy for the Cloud- http://broadcast.rackspace.com /hosting_knowledge/ whitepapers/Revolution_Not_Evolution- Whitepaper.pdf

[8] http://www.qrimp.com/blog/blog.The-Difference-between-IaaS-and-PaaS.html

[9] Big data: The next frontier for innovation, competition, and productivity. James Manyika, Michael Chui, Brad Brown, Jacques Bughin, Richard Dobbs, Charles Roxburgh, 2011.

