

Cloud Infrastructure Pattern

Keiko Hashizume

Florida Atlantic University, Boca Raton, FL, USA, ahashizu@fau.edu

Eduardo B. Fernandez

Florida Atlantic University, Boca Raton, FL, USA, ed@cse.fau.edu

Maria M. Larrondo-Petrie

Florida Atlantic University, Boca Raton, FL, USA, petrie@fau.edu

ABSTRACT

Cloud computing is a new paradigm that improves the utilization of resources and decreases the power consumption of hardware. Cloud computing allows users to have access to resources, software, and information using any device that has access to the Internet. The users consume these resources and pay only for the resources they use. We developed a Cloud Infrastructure Pattern which describes the infrastructure to allow the sharing of distributed virtualized computational resources such as servers, storage, and network.

Keywords: infrastructure, cloud computing, pattern, cloud service

1. INTRODUCTION

Cloud computing is a new paradigm that improves the utilization of resources and decreases the power consumption of hardware. Cloud computing allows users to have access to resources, software, and information using any device that has access to the Internet. The users consume these resources and pay only for the resources they use.

A cloud model provides three types of services [Mat09]: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS). IaaS provides processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. PaaS offers platform layer resources, including operating system support and software development frameworks to build, deploy and deliver applications into the cloud. SaaS provides end-user applications that are running on a cloud infrastructure. The applications are accessible from various client devices through a thin client interface such as a web browser (e.g., web-based email). In this paper, we develop a Cloud Infrastructure pattern which is described in the following section.

2. CLOUD INFRASTRUCTURE PATTERN

2.1 INTENT

The Cloud Infrastructure Pattern describes the infrastructure to allow the sharing of distributed virtualized computational resources such as servers, storage, and network.

2.2 CONTEXT

Distributed systems where we want to improve the utilization of resources and provide convenient access to all users.

2.3 PROBLEM

Some organizations do not have the resources to invest in infrastructure, middleware, or applications needed to run their businesses. Also, they may not be able to handle higher demands, or they cannot afford to maintain and store unused resources. How can they have access to computational resources?

This problem is affected by the following forces:

- *Transparency* - The underlying architecture should be transparent to its users. Users should be able to use the provider's services without understanding its infrastructure.
- *Flexibility* - Different infrastructure configurations, operating systems, and software can be demanded by users.
- *Elasticity* - Users should be able to expand or reduce resources in order to meet the different needs of their applications.
- *Pay-per-use* - Users should only pay for the resources they consume.
- *On-demand-service* – Services should be provided on demand.
- *Manageability* - In order to manage a large amount of service requests, the cloud must be easy to deploy and manage.
- *Accessibility* - Users should access resources from anywhere at anytime.
- *Shared resources* - Many users should be able to share resources in order to increase the amount of resource utilization and thus reduce costs.
- *Isolation* - Different user execution instances should be isolated from each other.
- *Shared Non-functional requirements (NFR)* – Sharing of the costs for NFRs is necessary to allow providers to offer a higher level of NFRs.

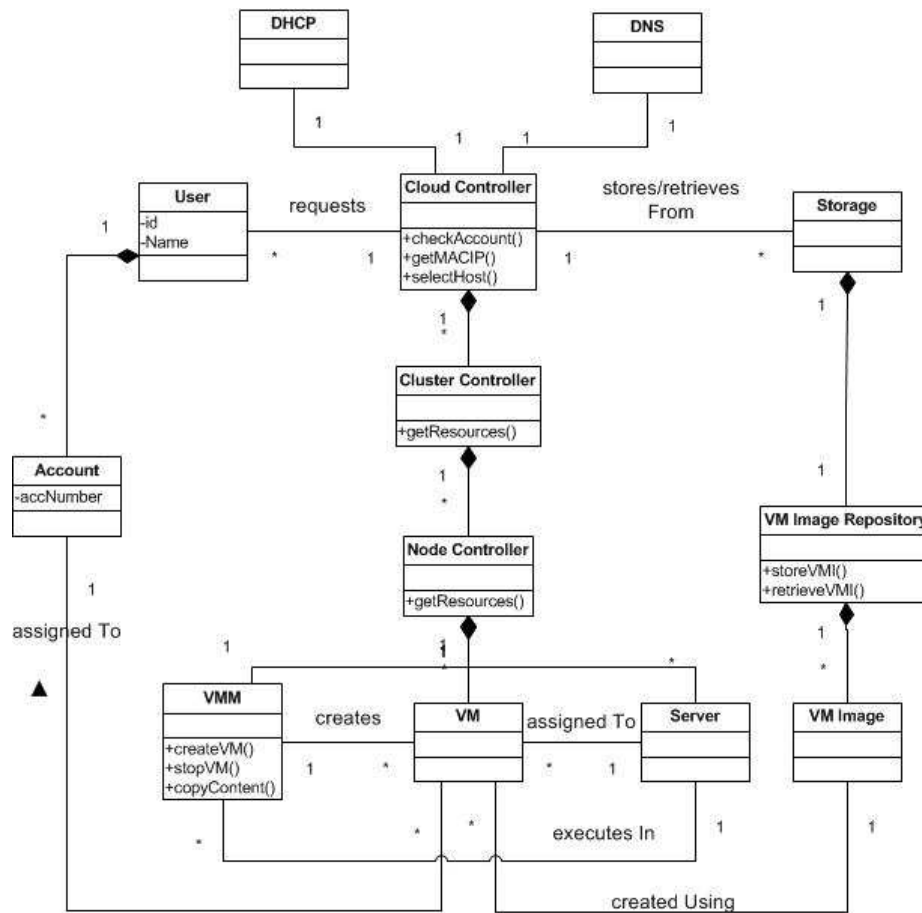
2.4 SOLUTION

The solution to this problem is a structure that is composed of many servers, storage, and a network, which can be shared by multiple users and accessible through the Internet. These resources are provided to the users as a form of service called Infrastructure-as-a-Service (IaaS). IaaS is based on virtualization technology which creates unified resources that can be shared by different applications. This foundation layer – IaaS – can be used as a reference for non-functional requirements.

Structure

Figure 1 shows a class diagram for a cloud infrastructure. The Cloud Controller is the main component which processes requests from Users (customers and administrators). A User can have one or more Accounts. The Cloud Controller coordinates a collection of services such as VM scheduling, authentication, VM monitoring and management. When a Cloud Controller receives a request from the user to create a VM, it requests its corresponding Cluster Controllers to provide a list their free resources. With this information, the Cloud Controller can choose which cluster will host the requested virtual machine. A Cluster Controller is composed of a collection of Node Controllers, which consist of a pool of Servers that host Virtual Machine (VM) instances. The Cluster Controller handles the state information of its Node Controllers, and schedules incoming requests to run instances. A Node Controller controls the execution, monitoring, and termination of the VMs through a Virtual Machine Monitor (VMM) which is the one responsible to run VM instances. The Cloud Controller retrieves and stores user data and Virtual Machine Images (VMI). The Virtual Machine Image Repository contains a collection of Virtual Machine Images that are used to instantiate a VM. The Dynamic Host Configuration Protocol (DHCP) server assigns a MAC/IP (Media Access Control/Internet Protocol) pair address for each VM through the Cloud Controller, and requests the Domain Name System (DNS) server to translate domain names into IP addresses in order to locate cloud resources.

Figure 1: Class Diagram for a Cloud Infrastructure



Dynamics

Some use cases are the following [Nis]:

- Open/close an account (actor: user)
- Copy data objects into/out a cloud (actor: administrator)
- Erase data objects in a cloud (actor: administrator)
- Store/Remove VM images (actor: administrator, user)
- Create a VM (actor: user)
- Migrate a VM (actor: administrator, user)

We show two UCs below:

UC1: Create a Virtual Machine (Figure 2)

Summary: Create of a Virtual Machine for a user, assign to it requested resources and assign it to a server

Actor: User

Precondition: The user has a valid account

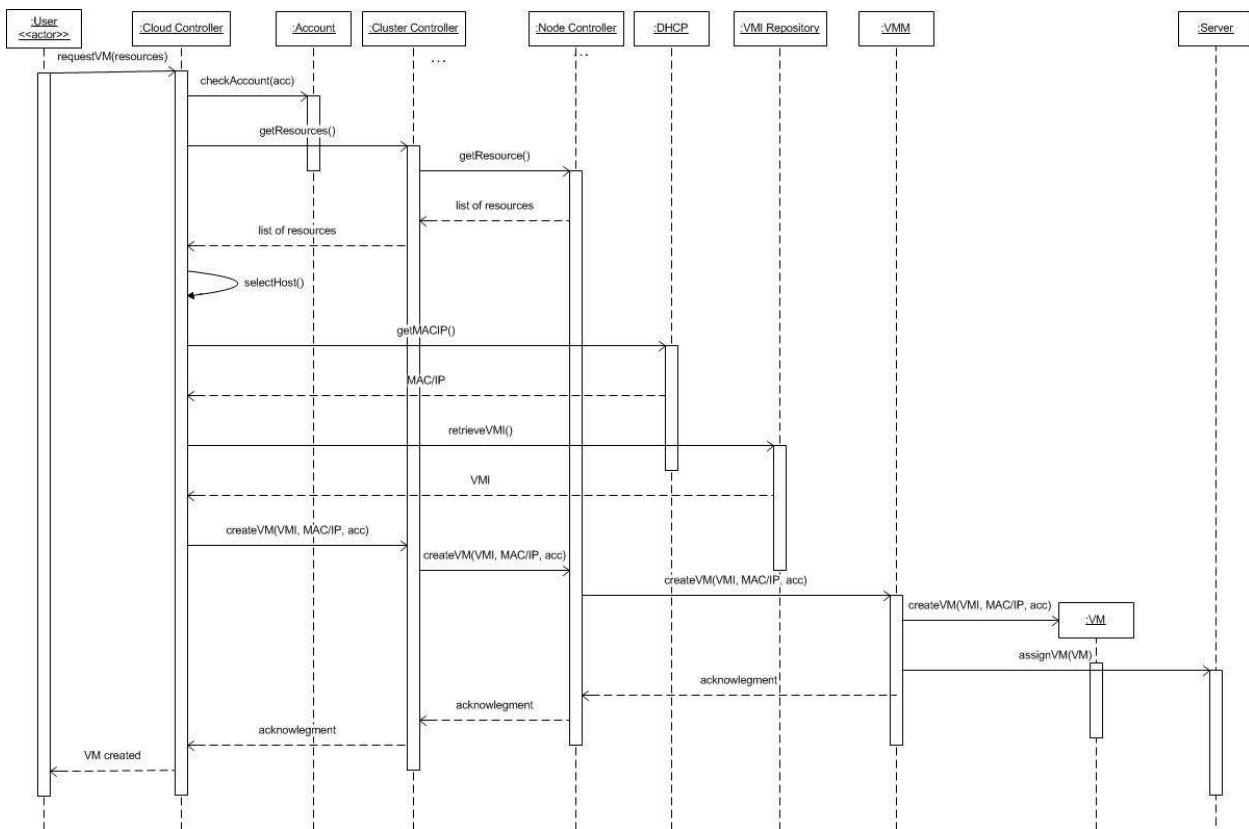
Description:

- a) A user requests a VM with some computational resources to the Cloud Controller.

- b) The Cloud Controller verifies whether the requester has a valid account.
- c) The Cloud Controller requests the available resources to the Cluster Controllers closer to the location of the user. In turn, the Cluster Controller queries its Node Controllers about their available resources. In the sequence diagram, there is only one Cluster Controller and one Node Controller in order to maintain the diagram simple, but there can be more Clusters and Node Controllers.
- d) The Node Controller sends the list of its available resources to the Cluster Controller, and the Cluster Controller sends it back to the Cloud Controller.
- e) The Cloud Controller chooses the first Cluster Controller that can support the computational resources requirements.
- f) The Cloud Controller requests a MAC/IP pair address from the DHCP server for the new VM.
- g) The Cloud Controller retrieves a virtual machine image from the repository (VMI)
- h) The Cloud Controller sends a request to the Cluster Controller to instantiate a VM.
- i) The Cluster Controller forwards the request to the Node Controller which forwards it to the VMM.
- j) The VMM creates a VM with the requested resources.
- k) The VMM assigns the VM to one of the servers.

Postcondition: A virtual machine is created and assigned to an account and a server

Figure 2: Sequence Diagram for Use Case Create a Virtual Machine



UC2: Migrate a Virtual Machine (Figure 3)

The administrator can migrate a VM to a specific Node Controller that can be located in the same or in a different Cluster Controller. Even the administrator can migrate a VM to a specific location or to the first node that has the available resources. For this use case, we assume that the administrator will move a VM to the first available Node Controller within the same Cluster Controller.

Summary: A VM is migrated from one Node Controller to another one

Actor: Administrator

Precondition: a VM resides in some Node Controller (Compute Node Source)

Description:

- a) The Administrator requests to the Cloud Controller to migrate a VM. However, the migration process can be automatic due to load balancing for example.
- b) The Cloud Controller requests to the Cluster Controller a list of its available resources, and the Cluster Controller requests its corresponding Node Controllers about the list of their free resources. The sequence diagram only shows one Cluster Controller and one Node Controller in order to simplify the process.
- c) The Cloud Controller chooses the first Node that has the necessary resources to host the VM.
- d) The Cloud Controller sends a request to the Cluster Controller to start the migration of the VM.
- e) The Cluster Controller requests the Node Controller Source to stop the VM. The Node Controller Source forwards this request to the VMM Source.
- f) The VMM Source stops the VM, copies the content of the VM, and sends the content back to the Cluster Controller through the Node Controller.
- g) The Cluster Controller requests to the Node Controller Destination to create a VM.
- h) The Node Controller Destination requests the VMM Destination to create a VM.
- i) The VMM Destination creates a VM.

Postcondition: The VM has migrated to another host

Figure 3: Sequence Diagram for Use Case Migrate a Virtual Machine



2.5 IMPLEMENTATION

In Eucalyptus [Euc], the cloud setup consists of five components. The two higher level components are: the Cloud Controller and Walrus. The Cloud Controller is a Java program that offers EC2-compatible SOAP and web interfaces. Walrus is a data storage where users can store and access virtual machine images and their data. Walrus can be accessed through S3-compatible SOAP and REST interfaces.

Cluster Controller and Node Controllers are deployed as web services, and communications between them takes place over SOAP with WS-Security [Has09].

Typically, the two higher level components can be deployed in a single physical machine. However, the Cloud Controller and Cluster Controller can also be run on a separate physical machine in order to improve the overall performance of the system.

A typical configuration includes [Ubu]:

- 1 Cloud Controller (CPU-1GHz, Memory-512MB, Disk-5400rpm IDE, Disk space-40GB)
- 1 Walrus Controller (CPU-1GHz, Memory-512MB, Disk-5400rpm IDE, Disk space-40GB)
- 1 Cluster Controller + Storage Controller (CPU-1GHz, Memory-512MB, Disk-5400rpm IDE, Disk space-40GB)
- Nodes (VT extensions, Memory-1GB, Disk-5400rpm IDE, Disk space-40GB)

2.6 KNOWN USES

- Eucalyptus [Euc] is an open source framework used for hybrid and private cloud computing.
- OpenNebula [Ope] is an open source toolkit to build clouds.
- Nimbus [Nim] is an open source set of tools that offers IaaS capabilities to the scientific community.
- Amazon's EC2 [Ama] provides compute capacity through web services.
- HP Cloud Services [Hp] is a public cloud solution that provides scalable virtual servers on demand.
- IBM SmartCloud Foundation [Ibm] offers servers, storage and virtualization components in order to build private, public and hybrid clouds.

2.7 CONSEQUENCES

The Cloud Infrastructure Pattern provides the following benefits:

- *Transparency* – Cloud users are usually not aware where their virtual machines are running or where their data is stored. However, in some cases users can request a general location zone for virtual machines or data.
- *Flexibility* - Cloud users can request different types of computational and storage resources. For instance, Amazon's EC2 [Ama] provides a variety of instance types and operating systems.
- *Elasticity* - Resources provided to users can be scaled up or down depending on their needs. Multiple virtual machines can be initiated and stopped in order to handle increased or decreased workloads.
- *Pay-per-Use* – Cloud users can save on hardware investment because they do not need to purchase more servers. They just need to pay for the services that they use. Cloud services are usually charged using a fee-for-service billing model [Cent10]. For instance, users pay for the storage, bandwidth or computing resources they consume per month.
- *On-demand-services* – IaaS providers deliver computational resources, storage and network as services with one click.
- *Manageability* - Users place their requests to the cloud administrator who allocates, migrates, and monitors VMs.
- *Accessibility* - Cloud services are delivered using user-centric interfaces via the Internet from anywhere and anytime.
- *Shared resources* - Virtualization enables to share a pool of resources such as processing capacity, storage, and networks. Thus, higher utilization rate can be reached [Amr].

- *Isolation* – A VMM provides strong isolation between different virtual machines, whose guest operating systems are then protected from one another [Kar08].
- *Shared Non-functional requirements (NFRs)* – Some IaaS providers offer security features such as authentication and authorization to customers that can be added as part of the service. Sharing allows the provider to offer a higher degree of NFRs at a reasonable cost.

The Cloud Infrastructure Pattern has the following liabilities:

- Cloud computing is dependent on network connections. While using cloud services, users must be connected to the Internet, although a limited amount of work can be done offline.
- The Cloud may bring security risks associated to privacy and confidentiality areas since the users do not have the control of the underlying infrastructure.
- The isolation between VMs may not be so strong [Has12a].
- Virtualization increases some performance overhead.

2.8 RELATED PATTERNS

- The Virtual Machine Operating System [Fer05] describes the VMM and its created VMs from the point of view of an OS architecture.
- The Grid Architectural Pattern [Cam06] allows the sharing of distributed and heterogeneous computational resources such as CPU, memory, and disk storage for a grid environment.
- Misuse patterns [Has12a] describe possible attacks to cloud infrastructures.
- The Platform-as-a-Service (PaaS) pattern [Has12b] describes development platforms that provide virtual environments for developing applications in the cloud.

REFERENCES

- [Amr] D. Amrhein, “Forget Defining Cloud Computing,” <http://soa.sys-con.com/node/1018801>
- [Ama] Amazon Web Services LLC, “Amazon Elastic Compute Cloud (Amazon EC2),” <http://aws.amazon.com/ec2/>
- [Cam06] R. Camargo, A. Goldchleger, M. Carneiro, and F. Kon, “The Grid Architectural Pattern: Leveraging Distributed Processing Capabilities,” *Procs. of the International Conference on Pattern Languages of Program Design 5*, 2006, 337-356
- [Cent10] Centre for the Protection of National Infrastructure, “Information Security Briefing 01/2010 Cloud Computing,” March 2010
- [Euc] Eucalyptus Systems, Inc, <http://www.eucalyptus.com/>
- [Fer05] E.B.Fernandez and T. Sorgente, “A pattern language for secure operating system architectures”, *Procs. of the 5th Latin American Conference on Pattern Languages of Programs*, Brazil, August 16-19, 2005.
- [Hp] HP Cloud Service, <http://hpcloud.com/>
- [Ibm] IBM, <http://www.ibm.com/cloud-computing/us/en/>
- [Has12a] K. Hashizume, E.B. Fernandez, and N. Yoshioka, “Three Misuse Patterns for Cloud Computing”. In *Security Engineering for Cloud Computing: Approaches and Tool*, IGI Global: 2012.
- [Has12b] K. Hashizume, E.B. Fernandez, and M.M. Larrondo-Petrie, “Cloud Service Model Patterns”. Sent to the *19th Conference on Pattern Languages of Programs PLoP 2012*.
- [Kar08] P. Karger and D. Safford, “I/O for Virtual Machine Monitors: Security and Performance Issues,” *IEEE Security & Privacy*, Sep./Oct. 2008, 16-23
- [Mat09] T. Mather, S. Kumaraswamy, and S. Latif, *Cloud Security and Privacy. 2009*: O’Reilly Media, Inc.
- [Nim] Nimbus, <http://www.nimbusproject.org/about/>
- [Nis] The National Institute of Standards and Technology (NIST), “Cloud Computing Use Cases”, <http://www.nist.gov/itl/cloud/use-cases.cfm>
- [Ope] OpenNebula Project Leads, <http://opennebula.org/>
- [Ubu] Ubuntu, “UEC Package Install Separate”, <https://help.ubuntu.com/community/UEC/PackageInstallSeparate#Overview>

Authorization and Disclaimer

Authors authorize LACCEI to publish the paper in the conference proceedings. Neither LACCEI nor the editors are responsible either for the content or for the implications of what is expressed in the paper.