Q1 Installation and configuration of Hadoop/Euceliptus etc.

Problem Statement

By first setting up a small Eucalyptus Cloud on a few local servers the thesis can answerwhich problems and obstacles there are when preparing the open-source infrastructure. The main priority is setting up a cloud that can deliver virtual instances capable of running Hadoop MapReduce on them to supply a base to perform the analysis of the framework.

Simplifying launching Hadoop MapReduce clusters inside the Eucalyptus Cloud is ofsecond priority after setting up the infrastructure and testing the feasibility of MapReduceon virtual machines. This can include scripts, stand-alone programs or utilities beyond Eucalyptus and/or Hadoop.

Goals

The goals of this thesis is to do a software study and analysis of the performance and Usability of Hadoop MapReduce running on top of virtual machines inside an Eucalyptus Cloud infrastructure. It will study means to setup, launch, maintain and remove virtual instances that can together form a MapReduce cluster.

Related Work

Apache Whirr is a collection of scripts that has sprung out as a project of its own. The purpose of Whirr is to simplify controlling virtual nodes inside a cloud like Amazon Web Services. Whirr controls everything from launching, removing and maintaining instancesthat Hadoop then can utilize in a cluster.

Another similar controller program is Puppet [14] from Puppet Labs. This program fully controls instances and clusters inside an EC2-compatible (AWS or Eucalyptus for example) cloud. It uses a program outside the cloud infrastructure that can control whether to launch, edit or remove instances. Puppet also controls the Hadoop MapReduce clusterinside the virtual cluster. Mathias Gug, an Ubuntu Developer, has tested how to deploy avirtual cluster inside an Ubuntu Enterprise Cloud using Puppet. The results can be foundon his blog [13].Hadoop's commercial and enterprise o spring, Cloudera [6], has released a distribu-tion called CDH. The current version, version 3, contains a virtual

machine with HadoopMapReduce cgured along with Apache Whirr instructions. This is to simplify launching
and conguring Hadoop MapReduce clusters inside a cloud. These releases also contains extra packages for enterprise clusters, such as Pig, Hive, Sqoop and HBase. CDH also uses Apache Whirr to simplify AWS deployment.

Software study – Eucalyptus

Eucalyptus is a free open-source cloud management system that is using the same APIas the AWS are using. This enables tools that originally where developed for Amazon tobe used with Eucalyptus, but with the added benet of Eucalyptus being free and open-source. It provides the same functionality in terms of IaaS deployment and can be used as3.3. Software study - Eucalyptus 11 a private, hybrid or even a public cloud system with enough hardware.

The different parts of Eucalyptus

Walrus

Walrus is the name of the storage container system, similar to Amazon S3. It stores data in buckets and have the same API to read and write data in a redundant system.Eucalyptus stores a way to limit access and size of the storage buckets through the same means as S3, by enforcing user credentials and size limits. Walrus is written in Java, and is accessible through the same means as S3 (SOAP, REST or Web Browser).

Cloud Controller

The Cloud Controller (CLC) is the Eucalyptus implementation of the Elastic Compute Cloud (EC2) that Amazon provides. The CLC is responsible for starting, stopping and controlling instances in the system, as this is providing the computational power (CPU & RAM) to the user. The CLC is indirectly contacting the hypervisors through Cluster Controllers (CC) and Node Controllers (NC). The CLC is written in Java.

Storage Controller

This is the equivalent to the EBS found in Amazon. The Storage Controller (SC) is responsible for providing fast dynamic storage devices with low latency and variable

storage size. It resides outside the virtual CLC-instances, but can communicate with them as external devices in a similar fashion of the EBS system. The SC is written in Java.

Hadoop MapReduce & HDFS

HDFS

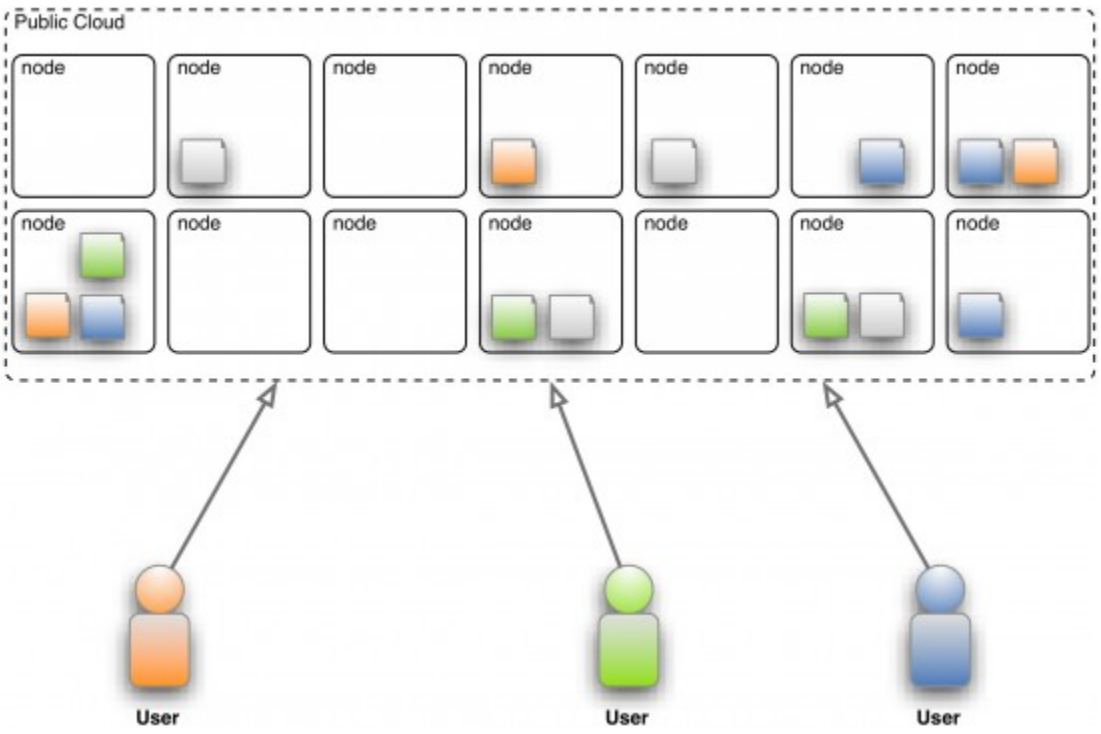
The Hadoop Distributed File System, HDFS, is a specialized lesystem to store large amounts of data across a distributed system of computers with very high through-put and multiple replication on a cluster. It provides reliability between the different physical machines to support a base for very fast computations on a large dataset. MapReduce MapReduce is a programming idiom for analyzing and process extremely large datasets in a fast, scalable and distributed way. Originally conceived by Google as a way of handling the enormous amount of data produced by their search bots .lesystems and I/O. This is more of a library that has all thefeatures that HDFS and MapReduce uses to handle the distributed computation. Ithas the the code for persistent data structures and Java RPC that HDFS needs tostore clustered data .

Q 2Service deployment & Usage over cloud.

Cloud services can be deployed in different ways, depending on the organizational structure and the provisioning location. Four deployment models are usually distinguished, namely public, private, community and hybrid cloud service usage.

Public Cloud

The deployment of a public cloud computing system is characterized on the one hand by the public availability of the cloud service offering and on the other hand by the public network that is used to communicate with the cloud service. The cloud services and cloud resources are procured from very large resource pools that are shared by all end users. These IT factories, which tend to be specifically built for running cloud computing systems, provision the resources precisely according to required quantities. By optimizing operation, support, and maintenance, the cloud provider can achieve significant economies of scale, leading to low prices for cloud resources.



Some of the best-known examples of public cloud systems are Amazon Web Services (AWS) containing the Elastic Compute Cloud (EC2) and the Simple Storage Service (S3) which form an IaaS cloud offering and the Google App Engine with provides a PaaS to its customers. The customer relationship management (CRM) solution Salesforce.com is the best-known example in the area of SaaS cloud offerings.

Private Cloud

Private cloud computing systems emulate public cloud service offerings within an organization's boundaries to make services accessible for one designated organization. Private cloud computing systems make use of virtualization solutions and focus on consolidating distributed IT services often within data centers belonging to the company. The chief advantage of these systems is that the enterprise retains full control over corporate data, security guidelines, and system performance. In contrast, private cloud

offerings are usually not as large-scale as public cloud offerings resulting in worse economies of scale.

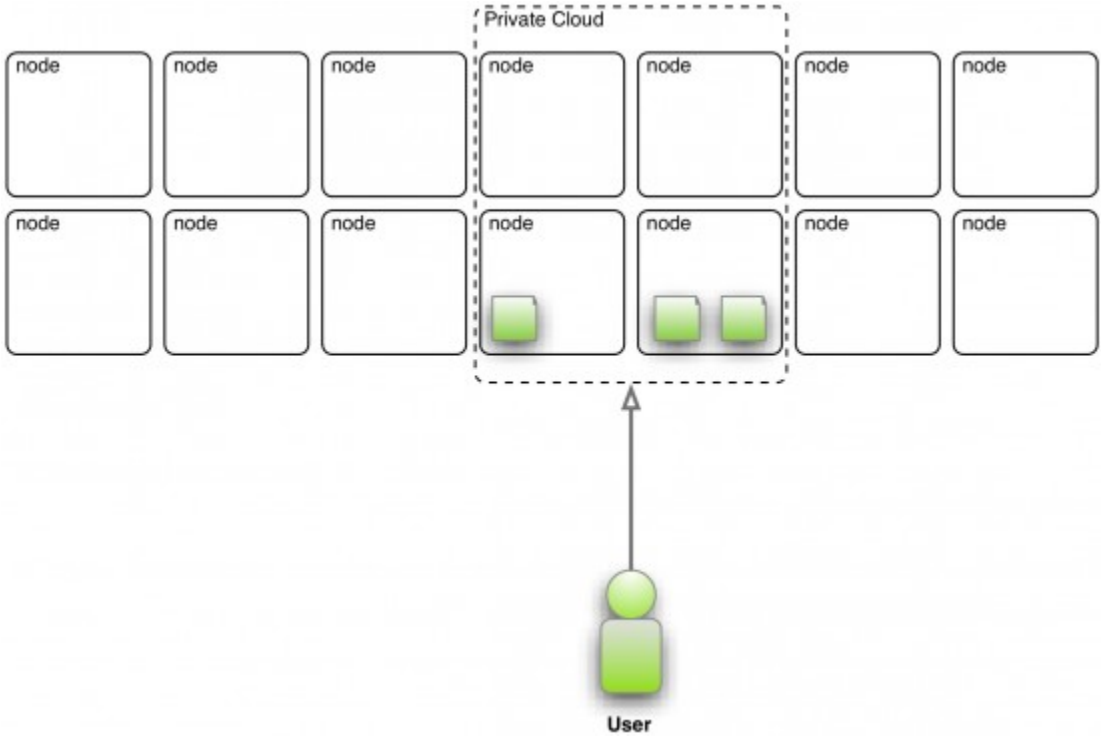


Figure 1: A user accessing a private cloud

Community Cloud

In a community cloud, organizations with similar requirements share a cloud infrastructure. It may be understood as a generalization of a private cloud, a private cloud being an infrastructure which is only accessible by one certain organization.

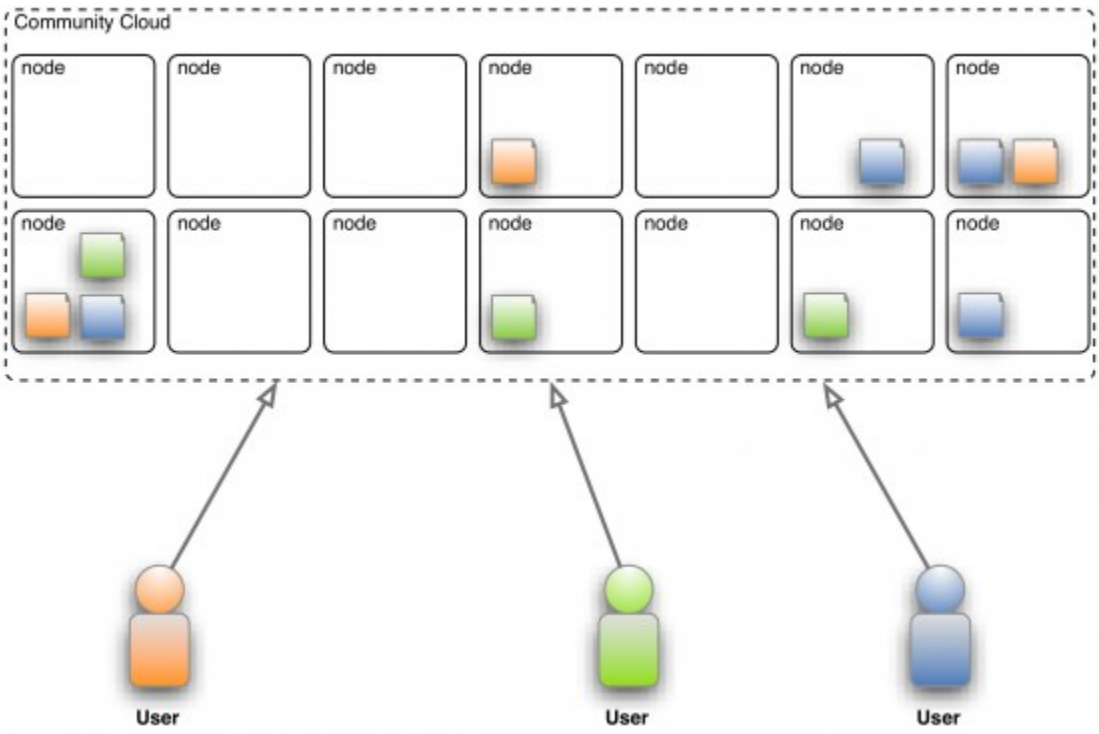


Figure 3: Three users accessing a community cloud

Hybrid Cloud

A hybrid cloud service deployment model implements the required processes by combining the cloud services of different cloud computing systems, e.g. private and public cloud services. The hybrid model is also suitable for enterprises in which the transition to full outsourcing has already been completed, for instance, to combine community cloud services with public cloud services.

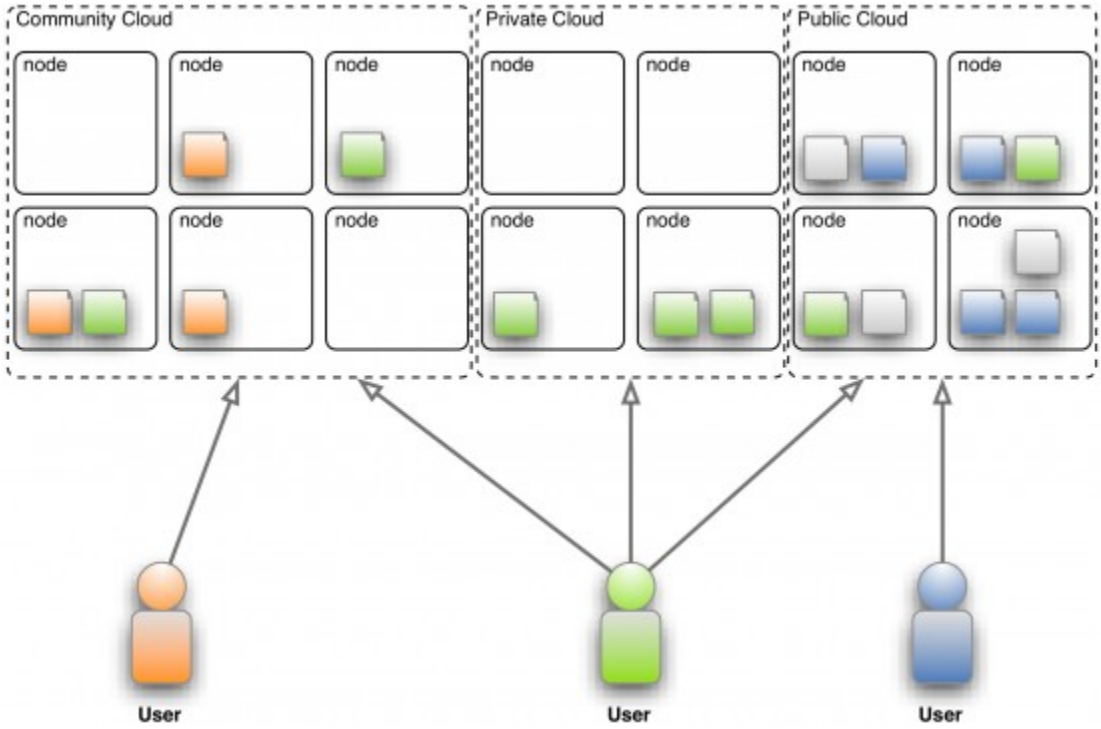


Figure 4: Hybrid cloud usage

Q 3 Management of cloud resources.

Managing resources at large scale while providing performance isolation and efficient use of underlying hardware is a key challenge for any cloud management software. Most virtual machine (VM) resource management systems like VMware DRS clusters, Microsoft PRO and Eucalyptus, do not currently scale to the number of hosts and VMs needed by cloud offerings to support the elasticity required to handle peak demand. In addition to scale, other problems a cloud-level resource management layer needs to solve include heterogeneity of systems, compatibility constraints between virtual machines and underlying hardware, islands of resources created due to storage and network connectivity and limited scale of storage resources.

Managing your cloud infrastructure can be a lot of work. You need to integrate with an architecture defined by the cloud provider, using its specific primitives for working with cloud components. This requires tying into the cloud APIs for configuring IP addresses, subnets and firewalls, as well as data service functions for your storage. Because control of these functions is based on the cloud provider's infrastructure and services, you also have to modify your internal processes and control systems to integrate with the cloud infrastructure management.

Even managing your operating systems as part of a cloud deployment presents challenges. Many cloud services provide "base servers" or templates that contain a simple distribution or OS, which are then used to build up your specific server/OS/application. This approach works well when the provider has the exact base server you want to start from, and you have a process in place to build from a running server. The challenge is that when you build up a server based on a gold image, it may: a) not match the base cloud OS version, b) be built from a non-running or base OS versus a fully-running OS (as required by most clouds), and c) use internal resources (boot servers, internal repositories, etc.) that are not available in the cloud. From a maintenance perspective, many organizations use central controls for updates

(like WSUS for windows), and these services depend on access to data center networks and services. Since public clouds are running external to your data center, these services either won't work, or need to be altered to run the hybrid environment.
Finally, the cloud creates additional complexity for managing applications. You almost always need to modify applications to accommodate cloud differences (virtual environment, networks and storage), which means that the applications in the cloud diverge from the "original" or base applications in your data center. You may also use third-party tools to help with integration into the cloud (such as VPN software, integration scripts, encryption software, etc.), which then need to be maintained. Each of these software elements has it its own lifecycle and update management, most of which apply to every image deployed into the cloud.

The management problems introduced by including the cloud in your infrastructure all have their source in the same issue – the cloud is something separate and different from your data center. This separation becomes clear when you consider the integration and management issues that span everything from provisioning to reengineering your applications to changes in lifecycle management. At CloudSwitch we're streamlining and automating cloud management to eliminate most of these issues, and bridge the separation between the cloud and your data center.

4 Using existing cloud characteristics & Service models

Cloud computing is a *colloquial* expression used to describe a variety of different types of computing concepts that involve a large number of computers connected through a real-time communication network . Cloud computing is a jargon termwithout a commonly accepted unequivocal scientific or technical definition. In science, cloud computing is a synonym for distributed computing over a network and means the ability to run a program on many connected computers at the same time. The phrase is also, more commonly used to refer to network-based services which appear to be provided by real server hardware, which in fact are served up by virtual hardware, simulated by software running on one or more real machines. Such virtual servers do not physically exist and can therefore be moved around and scaled up (or down) on the fly without affecting the end user - arguably, rather like a cloud.

The popularity of the term can be attributed to its use in marketing to sell hosted services in the sense of application service provisioning that run client server software on a remote location.

Key characteristics of Cloud Computing

On-demand self-service

The first Cloud Computing characteristic is defined by NIST as:

A consumer can unilaterally provision computing capabilities, such as server time and network storage, as needed automatically without requiring human interaction with each service's provider. What this means is that a consumer controls the provisioning process and does not need to interact with anyone or submit a request for approval in order to obtain computing resources. More directly, this means developers (consumers does not refer to the ultimate end-users of an application, but rather to someone associated with creating or operating the application that runs on the Cloud infrastructure) can obtain resources without going to the IT operations and infrastructure group. The developer 'selfserves' and directly chooses resources, which are automatically delivered to him or her.

TWO :Broad network access

The second Cloud Computing characteristic is defined by NIST as:

Capabilities are available over the network and accessed through standard mechanisms that promote use by heterogeneous thin or thick client platforms (e.g. mobile phones, tablets, laptops and workstations).This characteristic speaks directly to the methods by which Cloud Computing applications interact with Cloud Computing infrastructures and applications, and indirectly to what we may expect as to likely demand profiles

Three: Resource pooling

The third Cloud Computing characteristic identified is defined by NIST as:

The provider's computing resources are pooled to serve multiple consumers using a multi-tenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand. There is a sense of location independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g. country, state or data centre). Examples of resources include storage, processing, memory and network bandwidth.

Rapid elasticity

The fourth Cloud Computing characteristic is defined by NIST as:

Capabilities can be elastically provisioned and released, in some cases automatically, to scale rapidly outward and inward commensurate with demand. To the consumer, the capabilities available for provisioning often appear to be unlimited and can be appropriated in any quantity at any time.

Measured service

Cloud systems automatically control and optimise resource use by leveraging a metering capability at some level of abstraction appropriate to the type of service (e.g. storage, processing, bandwidth and active user accounts). Resource usage can be monitored, controlled and reported, providing transparency for both the provider and consumer of the utilised service.

This NIST Cloud Computing characteristics is by far the most controversial, as it represents a huge change to how most IT organisations charge for their services, as well as an enormous challenge to implement, should the IT organisation attempt to fully comply with the characteristic. Moreover, as this characteristic focuses on money and finance, traditionally the most compelling of topics to humans, it garners immediate attention from everyone.

Q5 Cloud Security Management.

Cloud computing security (sometimes referred to simply as "cloud security") is an evolving sub-domain of computer security, network security, and, more broadly, information security. It refers to a broad set of policies, technologies, and controls deployed to protect data, applications, and the associated infrastructure of cloud computing.

Security issues associated with the cloud

Organizations use the Cloud in a variety of different service models (SaaS, PaaS, IaaS) and deployment models (Private, Public, Hybrid). There are a number of security issues/concerns associated with cloud computing but these issues fall into two broad categories: Security issues faced by cloud providers (organizations providing software-, platform-, or infrastructure-as-a-service via the cloud) and security issues faced by their customers. In most cases, the provider must ensure that their infrastructure is secure and that their clients' data and applications are protected while the customer must ensure that the provider has taken the proper security measures to protect their information.

Cloud Security Controls

Cloud security architecture is effective only if the correct defensive implementations are in place. An efficient cloud security architecture should recognize the issues that will arise with security management. The security management addresses these issues with security controls. These controls are put in place to safeguard any weaknesses in the system and reduce the effect of an attack. While there are many types of controls behind a cloud security architecture, they can usually be found in one of the following categories:

Deterrent Controls

These controls are set in place to prevent any purposeful attack on a cloud system. Much like a warning sign on a fence or a property, these controls do not reduce the actual vulnerability of a system.

Preventative Controls

These controls upgrade the strength of the system by managing the vulnerabilities. The preventative control will safeguard vulnerabilities of the system. If an attack were to occur, the preventative controls are in place to cover the attack and reduce the damage and violation to the system's security.

Corrective Controls

Corrective controls are used to reduce the effect of an attack. Unlike the preventative controls, the corrective controls take action as an attack is occurring.

Detective Controls

Detective controls are used to detect any attacks that may be occurring to the system. In the event of an attack, the detective control will signal the preventative or corrective controls to address the issue.

Security and privacy

Identity management

Every enterprise will have its own identity management system to control access to information and computing resources. Cloud providers either integrate the customer’s identity management system into their own

Physical and personnel security infrastructure,using federation or SSOtechnology, or provide an identity management solution of theirown.

Providers ensure that physical machines are adequately secure and that access to these machines as well as all relevant customer data is not only restricted but that access is documented.

Availability

Cloud providers assure customers that they will have regular and predictable access to their data and applications.

Application security

Cloud providers ensure that applications available as a service via the cloud are secure by implementing testing and acceptance procedures for outsourced or packaged application code. It also requires application security measures be in place in the production environment.

Privacy Finally, providers ensure that all critical data (credit card numbers, for example) are masked and that only authorized users have access to data in its entirety. Moreover, digital identities and credentials must be protected as should any data that the provider collects or produces about customer activity in the cloud.

Legal issues In addition, providers and customers must consider legal issues, such as Contracts and E-Discovery, and the related laws, which may vary by country.

Compliance

Numerous regulations pertain to the storage and use of data, including Payment Card Industry Data Security Standard (PCI DSS), theHealth Insurance Portability and Accountability Act (HIPAA), the Sarbanes-Oxley Act, among others. Many of these regulations require regular reporting and audit trails. Cloud providers must enable their customers to comply appropriately with these regulations.

Business continuity and data recovery

Cloud providers have business continuity and data recovery plans in place to ensure that service can be maintained in case of a disaster or an emergency and that any data loss will be recovered. These plans are shared with and reviewed by their customers.

Logs and audit trails

In addition to producing logs and audit trails, cloud providers work with their customers to ensure that these logs and audit trails are properly secured, maintained for as long as the customer requires, and are accessible for the purposes of forensic investigation (e.g.,eDiscovery).

Unique compliance requirements

In addition to the requirements to which customers are subject, the data centers maintained by cloud providers may also be subject to compliance requirements. Using a cloud service provider (CSP) can lead to additional security concerns around data jurisdiction since customer or tenant data may not remain on the same system, or in the same data center or even within the same provider's cloud.

Legal and contractual issues

Aside from the security and compliance issues enumerated above, cloud providers and their customers will negotiate terms around liability (stipulating how incidents involving data loss or compromise will be resolved, for example), intellectual property, and end-of-service (when data and applications are ultimately returned to the customer).

Public records

Legal issues may also include records-keeping requirements in the public sector, where many agencies are required by law to

retain and make available electronic records in a specific fashion. This may be determined by legislation, or law may require agencies to conform to the rules and practices set by a records-keeping agency. Public agencies using cloud computing and storage must take these concerns into the account.

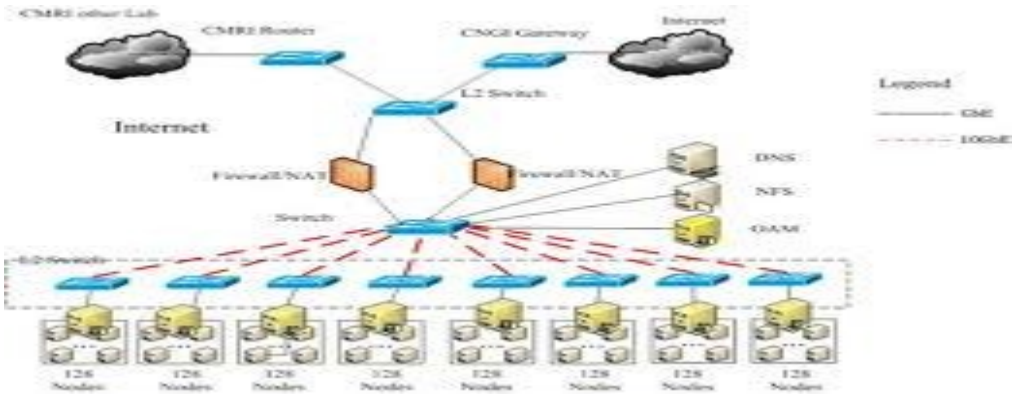
Q6 Performance evaluation of services over cloud .

Existing commercial cloud computing infrastructures such as Google App Engine Microsoft Azure and Amazon Web Services (AWS) [have proved the impact that cloudcomputing can have on current and future computations. These systems offer different levels of computation, storage and networking models to cloud users for deploying large scale applications. Cloud users can use these facilities over the Internet from anywhere in the world. For example, Amazon's AWS offers a number of high-level managed services. Amazon's Elastic Compute Cloud (EC2) presents a virtual computing environment to cloud users. Amazon also rents out different storage services, for example Simple Storage Service (S3).

Open Cirrus testbed

The cluster also has an additional 43network-attached storage nodes having 288TB of space. Each of these compute and storagenodes has a separate 1Gb/s and 10Gb/s link respectively to different switches.The testbed is partitioned into two separate logical clusters. The first cluster is namedAltocumulus and is used for running MapReduce [4] jobs using Hadoop .It consistsof 64 compute nodes running Hadoop Distributed File System (HDFS) on approximately96TB of space. The second cluster, named Cumulonimbus, consists of the remaining 64compute nodes. This cluster is dedicated for systems research and provides the capability to run experiments on dedicated hardware.

The storage facility allocated to Cumulonimbus is accessed through Network File System (NFS).



Limitations

The Illinois CCT is still not matured enough compared to other commercial clouds. It does not support any virtualization. Thus in some respects, it still does not support elastic computing facilities like Amazon EC2. Although the network-attached storage facilities available in this testbed mimic the Amazon S3 service, currently this storage service is not directly accessible from any external computer. This storage facility can be accessed only via a compute node acting as a gateway to the storage nodes. Considering these limitations, the Illinois CCT may not represent a commercial cloud. Still, we believe that some of our experimental results can be used to detect the root cause of performance variations and degradations

Experiment with the Storage Facility of the Illinois CCT

Cloud computing facilities are largely used for data-intensive applications, for example log processing or storing data at the cloud. User-perceived data transfer throughput is an important metric for the effectiveness of these applications. We, therefore, have

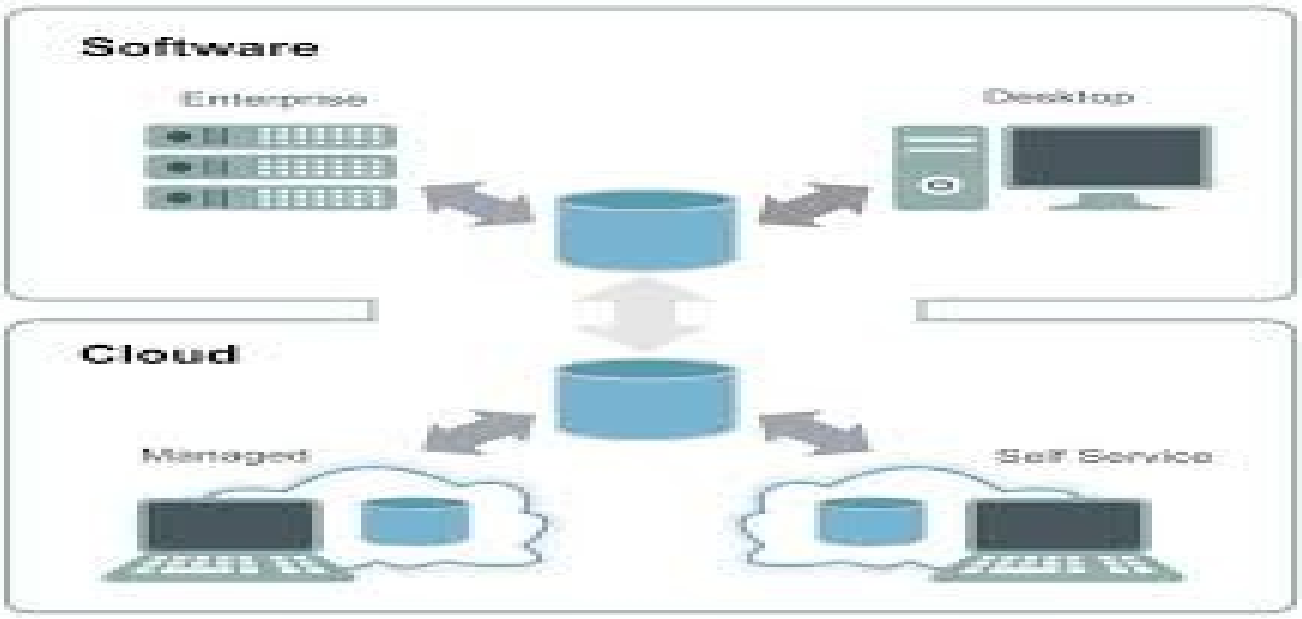
implemented a throughput measurement program to measure the storage and network performance of the.

Experiment with Distributed Cloud Facilities

Currently, lack of standardization in cloud computing APIs prevents cloud users from deploying their services at multiple cloud facilities. Such methods can be quite advantageous for cloud users as it ensures fault-tolerance of the offered services and improves availability. This also allows cloud users to move their data and/or tasks to a cloud that gives better performance. In [9], Greenberg et al. emphasized on the use of multiple cloud data centersand geo-distributed computational facilities.When a cloud user has access to multiple cloud facilities, it can selectively decide where torun the computation and where to put the associated data. As shown in Figure 4, if a cloud user decides to use the computational facility of Cloud 1 but experiences low throughput while communicating with this cloud, it may choose to send the data to Cloud 2 so that data transfer during computation can take place using the high performance path between Cloud 1 and Cloud 2. Users can also select which cloud to use based on their geographical locations. But before that, cloud users need to be aware of the data transfer throughput that can be achieved between two clouds. Currently, we only have access to the Illinois

CCT. As we do not have access to other cloud facilities, we employ an alternative strategy to perform the aforementioned experiments. We use Emulab to emulate a cloud computing facility. Instead of accessing the data from local Emulab machines, we measurethe data transfer throughput while accessing the networked storage of Emulab.

However, Emulab is not built to provide cloud computing facility, rather is more suited for other systems and networking research. Thus, results obtained from these experiments may not necessarily portray the actual performance characteristics of cloud-to-cloud communication. Still, we are able to have some insight on data transfer characteristics between two infrastructures hosting a cluster of machines shared by multiple users. In addition to measuring the data



Q 7Service models of cloud computing.

Service delivery in Cloud Computing comprises three different service models, namely Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS). The three service models or layer are completed by an end user layer that encapsulates the end user perspective on cloud services. The model is shown in figure 1.

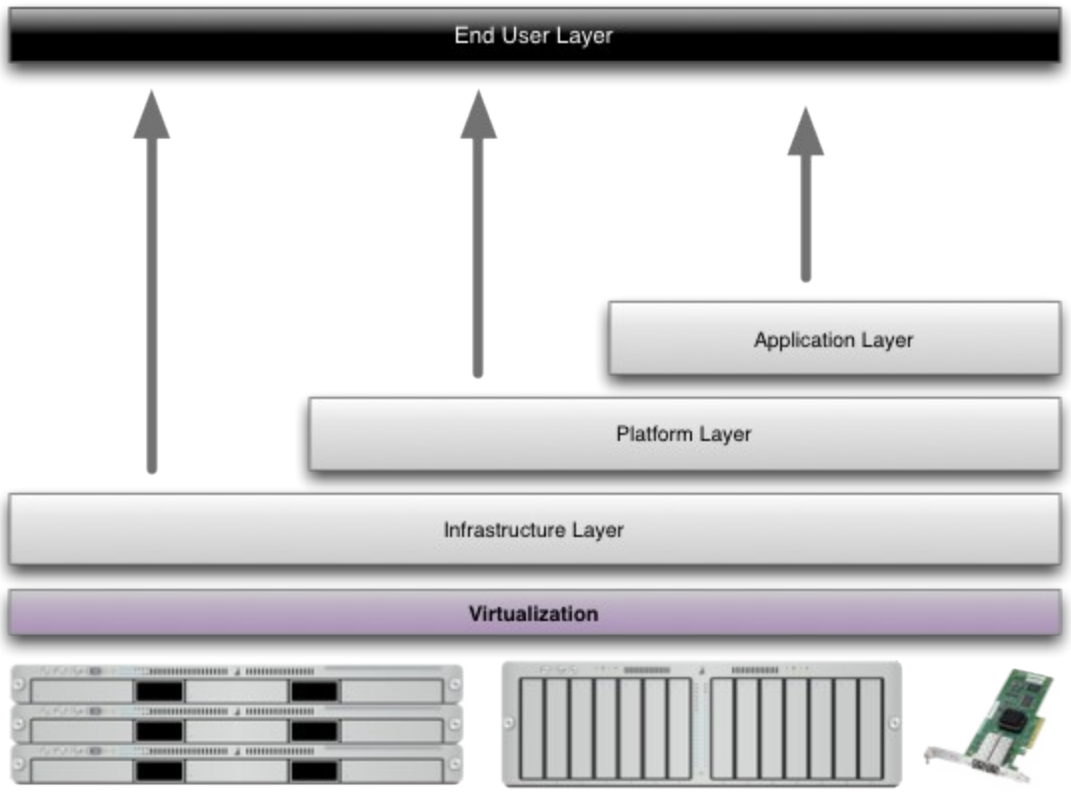


Figure 1: Service models and end user layer

If a cloud user accesses services on the infrastructure layer, for instance, she can run her own applications on the resources of a cloud infrastructure and remain responsible for the support, maintenance, and security of these applications herself. If she accesses a service on the application layer, these tasks are normally taken care of by the cloud service provider.

SaaS

Software-as-a-Service provides complete applications to a cloud's end user. It is mainly accessed through a web portal and service oriented architectures based on web service technologies. Credit card or bank account details must be provided to enable the fees for the use of the services to be billed.

The services on the application layer can be seen as an extension of the ASP (application service provider) model, in which an application is run, maintained, and supported by a service vendor. The main differences between the services on the application layer and the classic ASP model are the encapsulation of the application as a service, the dynamic procurement, and billing by units of consumption (pay as you go). However, both models pursue the goal of focusing on core competencies by outsourcing applications.

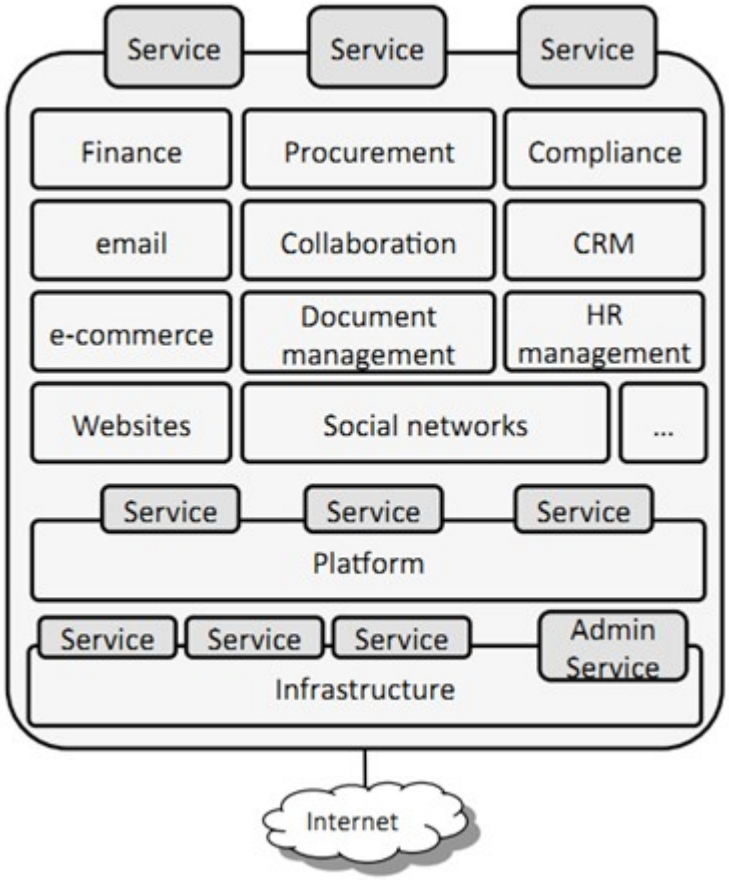


Figure 2: Software-as-a-Service (SaaS) Stack

PaaS

PaaS comprises the environment for developing and provisioning cloud applications. The principal users of this layer are developers seeking to develop and run a cloud application for a particular platform. They are supported by the platform operators with an open or proprietary language, a set of essential basic services to facilitate communication, monitoring, or service billing, and various other components, for instance to facilitate startup or ensure an application's scalability and/or elasticity (see figure 3). Distributing the application to the underlying infrastructure is normally the responsibility of the cloud platform operator. The services offered on a cloud platform tend to represent a compromise between complexity and flexibility that allows applications to be implemented quickly and loaded in the cloud without much configuration. Restrictions regarding the programming languages supported, the programming model, the ability to access resources, and persistency are possible downsides.

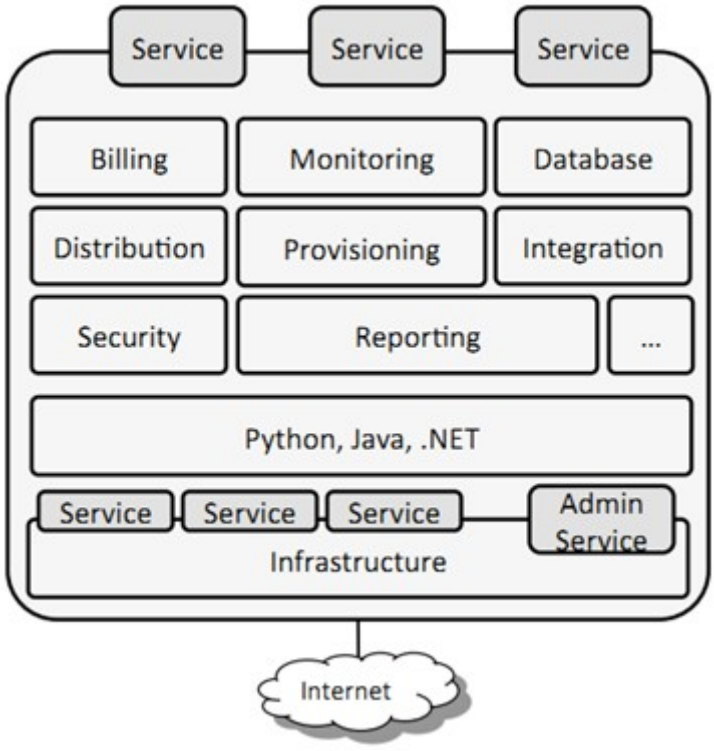


Figure 3: Platform-as-a-Service (PaaS) Stack

IaaS

The services on the infrastructure layer are used to access essential IT resources that are combined under the heading Infrastructure-as-a-Service (IaaS). These essential IT resources include services linked to computing resources, data storage resources, and the communications channel. They enable existing applications to be provisioned on cloud resources and new services implemented on the higher layers. Physical resources are abstracted by virtualization, which means they can then be shared by several operating systems and end user environments on the virtual resources – ideally, without any mutual interference. These virtualized resources usually comprise CPU and RAM, data storage resources (elastic block store and databases), and network resources as displayed in figure 4.

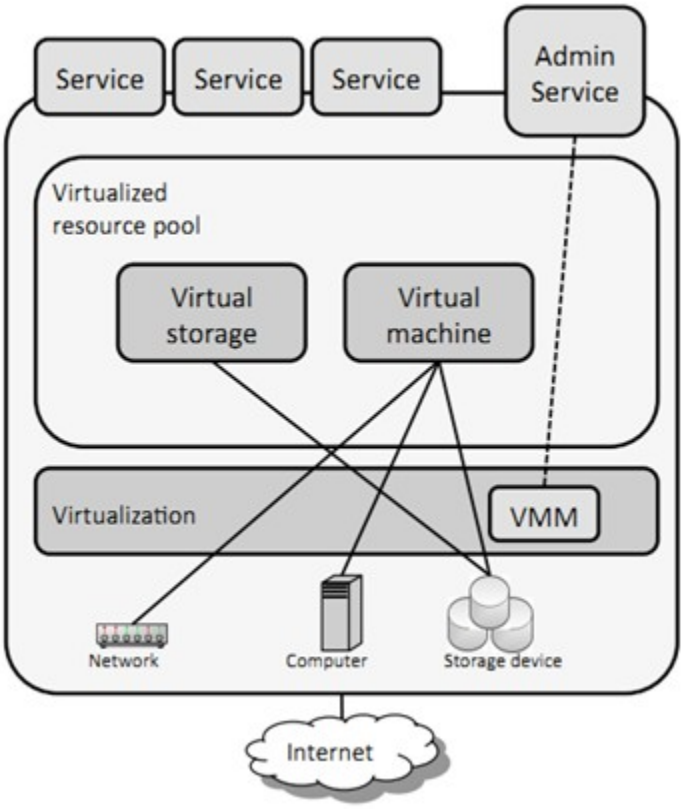


Figure 4: Infrastructure-as-a-Service (IaaS) Stack