ANNAMALAI UNIVERSITY

FACULTY OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

B.E.(CSE) Eighth Semester

Course Name : CLOUD COMPUTING

Staff: Dr.KT. Meena Abarna

UNIT – I

INTRODUCTION TO CLOUD COMPUTING

CLOUD COMPUTING

Cloud computing is the on-demand availability of computer system resources, especially data storage and computing power, without direct active management by the user. The term is generally used to describe data centers available to many users over the Internet. Large clouds, predominant today, often have functions distributed over multiple locations from central servers. If the connection to the user is relatively close, it may be designated an edge server.

Clouds may be limited to a single organization (enterprise clouds), or be available to many organizations (public cloud).

Cloud computing relies on sharing of resources to achieve coherence and economies of scale.

Advocates of public and hybrid clouds note that cloud computing allows companies to avoid or minimize up-front IT infrastructure costs. Proponents also claim that cloud computing allows enterprises to get their applications up and running faster, with improved manageability and less maintenance, and that it enables IT teams to more rapidly adjust resources to meet fluctuating and unpredictable demand, providing the **burst computing** capability: high computing power at certain periods of peak demand.

Cloud providers typically use a "pay-as-you-go" model, which can lead to unexpected operating expenses if administrators are not familiarized with cloud-pricing models.

The availability of high-capacity networks, low-cost computers and storage devices as well as the widespread adoption of hardware,virtualization, service-oriented,architecture and autonomic and utility computing has led to growth in cloud computing.By 2019, Linux was the most widely used operating system, including in Microsoft's offerings and is thus described as dominant. The Cloud Service Provider (CSP) will screen, keep up and gather data about the firewalls, intrusion identification or/and counteractive action frameworks and information stream inside the network.

Types of Cloud

Cloud computing is an Internet-based computing in which shared the pool of resources are available over a broad network access, these resources can e provisioned or released with minimum management efforts and service provider interaction.

There are four types of cloud:

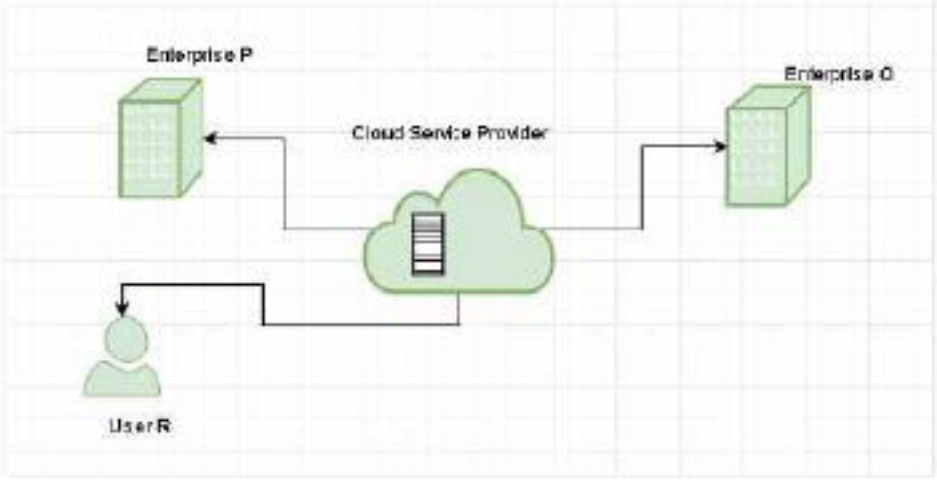
1. Public cloud
2. Private cloud
3. Hybrid cloud
4. Community cloud

Public cloud:

Public cloud are managed by third parties which provide cloud services over the internet to public, these services are available as pay-as-you-go billing mode.

They offer solutions for minimizing IT infrastructure costs and act as a good option for handling peak loads on the local infrastructure. They are a goto option for small enterprises, which are able to start their businesses without large upfront investments by completely relying on public infrastructure for their IT needs.

A fundamental characteristic of public clouds is **multitenancy**. A public cloud is meant to serve multiple users, not a single customer. A user requires a virtual computing environment that is separated, and most likely isolated, from other users.



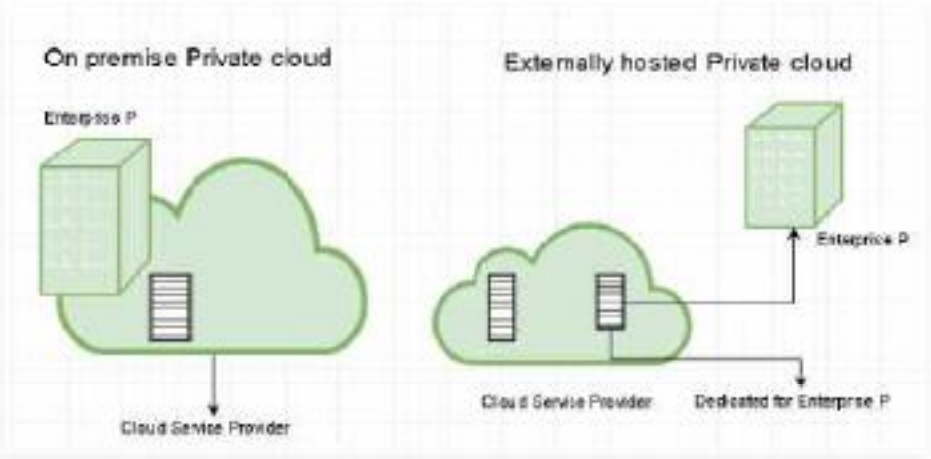
Private cloud :

Private clouds are distributed systems that work on a private infrastructure and providing the users with dynamic provisioning of computing resources. Instead of a pay-as-you-go model as in public clouds, there could be other schemes in that take into

account the usage of the cloud and proportionally billing the different departments or sections of an enterprise.

The advantages of using a private cloud are:

1. **Customer information protection:** In private cloud security concerns are less since customer data and other sensitive information does not flow out of a private infrastructure.
2. **Infrastructure ensuring SLAs:** Private cloud provides specific operations such as appropriate clustering, data replication, system monitoring and maintenance, and disaster recovery, and other uptime services.
3. **Compliance with standard procedures and operations:** Specific procedures have to be put in place when deploying and executing applications according to third-party compliance standards.This is not possible in case of public cloud.

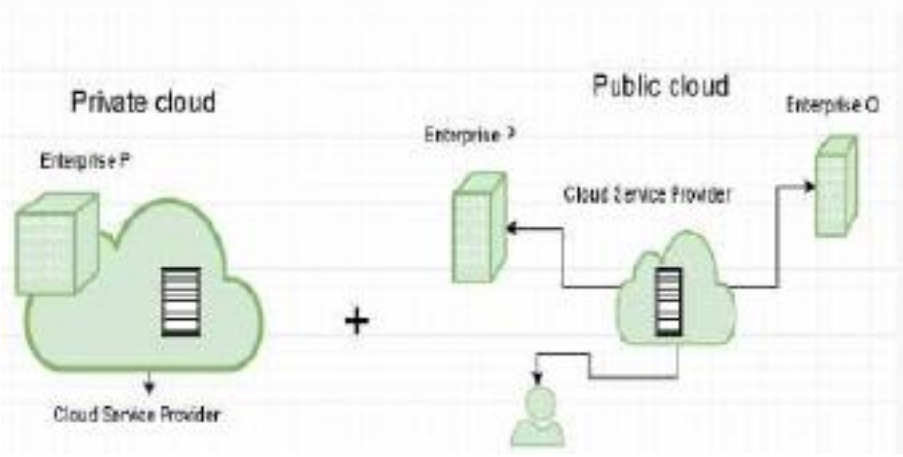


Hybrid cloud:

Hybrid cloud is a heterogeneous distributed system resulted by combining facilities of public cloud and private cloud. For this reason they are also called **heterogeneous clouds.**

A major drawback of private deployments is the inability to scale on demand and to efficiently address peak loads. Here public

clouds are needed. Hence, a hybrid cloud takes advantages of



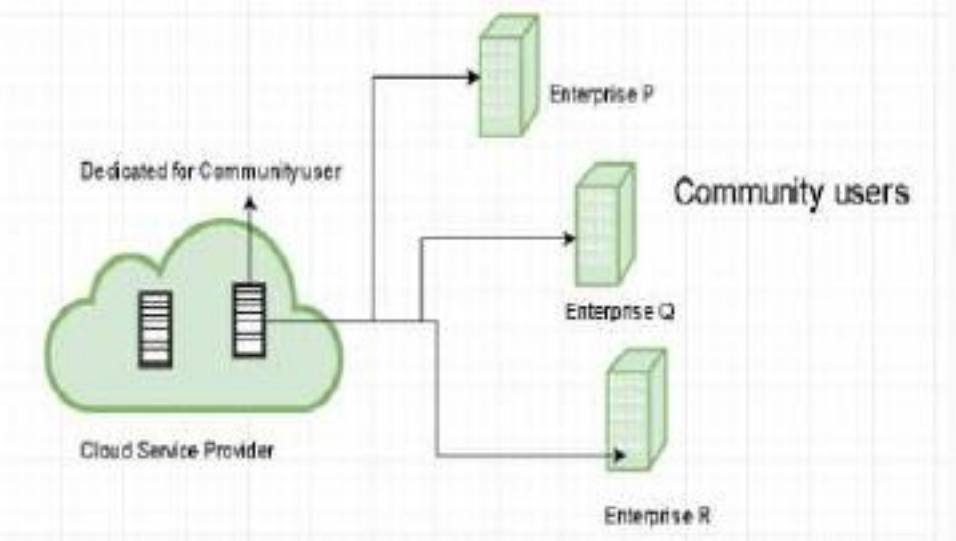
both public and private cloud.

Community cloud:

Community clouds are distributed systems created by integrating the services of different clouds to address the specific needs of an industry, a community, or a business sector.
In community cloud, the infrastructure is shared between organization which have shared concerns or tasks. The cloud may be managed by an organization or a third party.

Sectors that use community clouds are:

1. **Media industry:** Media companies are looking for quick, simple, low-cost way for increasing efficiency of content generation.Most media productions involve an extended ecosystem of partners. In particular, the creation of digital content is the outcome of a collaborative process that includes movement of large data, massive compute-intensive rendering tasks, and complex workflow executions.
2. **Healthcare industry:** In healthcare industry community clouds are used to share information and knowledge on the global level with sensitive data in the private infrastructure.
3. **Energy and core industry:** In these sectors, the community cloud is used to cluster set of solution which collectively addresses management, deployment, and orchestration of services and operations.
4. **Scientific research:** In this organization with common interests of science share large distributed infrastructure for scientific computing.



CLOUD COMPUTING HISTORY

Cloud computing is the on-demand availability of computer system resources, especially data storage and computing power, without direct active management by the user. The term is generally used to describe data centers available to many users over the Internet. Large clouds, predominant today, often have functions distributed over multiple locations from central servers. If the connection to the user is relatively close, it may be designated an edge server.

Clouds may be limited to a single organization (enterprise clouds), or be available to many organizations (public cloud).

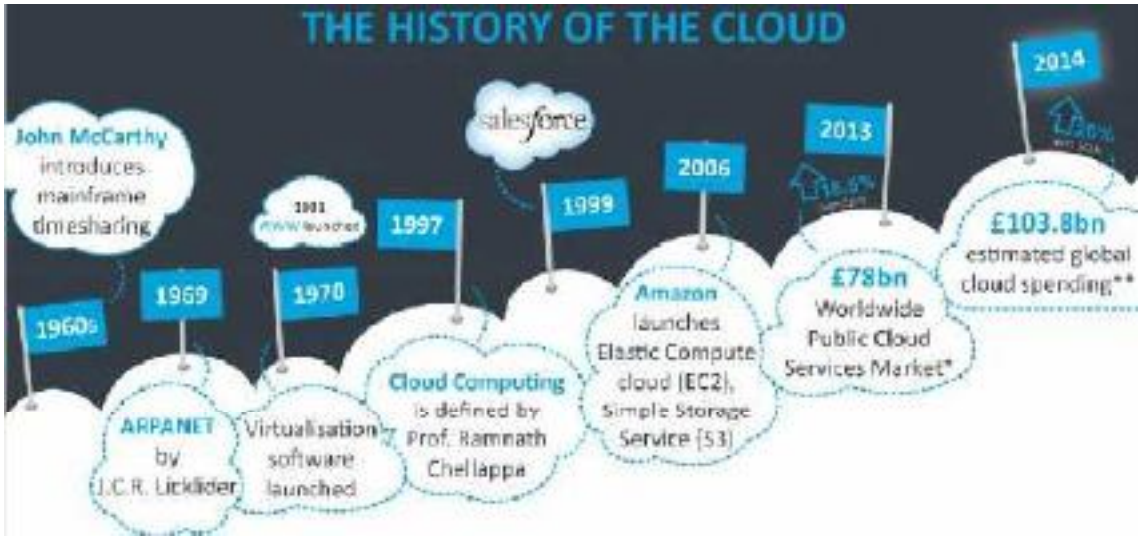
Cloud computing relies on sharing of resources to achieve coherence and economies of scale.

Advocates of public and hybrid clouds note that cloud computing allows companies to avoid or minimize up-front IT infrastructure costs. Proponents also claim that cloud computing allows enterprises to get their applications up and running faster, with improved manageability and less maintenance, and that it enables IT teams to more rapidly adjust resources to meet fluctuating and unpredictable demand, providing the burst computing capability: high computing power at certain periods of peak demand.

Cloud providers typically use a "pay-as-you-go" model, which can lead to unexpected operating expenses if administrators are not familiarized with cloud-pricing models.

The availability of high-capacity networks, low-cost computers and storage devices as well as the widespread adoption of hardware virtualization, service-

oriented architecture and autonomic and utility computing has led to growth in cloud computing. By 2019, Linux was the most widely used operating system, including in Microsoft's offerings and is thus described as dominant. The Cloud Service Provider (CSP) will screen, keep up and gather data about the firewalls, intrusion identification or/and counteractive action frameworks and information stream inside the network.



HISTORY

Cloud computing was popularized with Amazon.com releasing its Elastic Compute Cloud product in 2006.

References to the phrase "cloud computing" appeared as early as 1996, with the first known mention in a Compaq internal document.

The cloud symbol was used to represent networks of computing equipment in the original ARPANET by as early as 1977, and the CSNET by 1981—both predecessors to the Internet itself. The word cloud was used as a metaphor for the Internet and a standardized cloud-like shape was used to denote a network on telephony schematics. With this simplification, the implication is that the specifics of how the endpoints of a network are connected are not relevant to understanding the diagram.

The term cloud was used to refer to platforms for distributed computing as early as 1993, when Apple spin-off General Magic and AT&T used it in describing their (paired) Telescript and PersonaLink technologies.

EARLY HISTORY

During the 1960s, the initial concepts of time-sharing became popularized via RJE (Remote Job Entry); this terminology was mostly associated with large vendors such as IBM and DEC. **Full-time-sharing solutions** were available by the early 1970s on such platforms as Multics (on GE hardware), Cambridge CTSS, and the earliest UNIX ports (on DEC hardware). Yet, the "data center" model where users submitted jobs to operators to run on IBM's mainframes was overwhelmingly predominant.

In the 1990s, telecommunications companies, who previously offered primarily dedicated point-to-point data circuits, began offering virtual private network (VPN) services with comparable quality of service, but at a lower cost. By switching traffic as they saw fit to balance server use, they could use overall network bandwidth more effectively. They began to use the cloud symbol to denote the demarcation point between what the provider was responsible for and what users were responsible for.

Cloud computing extended this boundary to cover all servers as well as the network infrastructure. As computers became more diffused, scientists and technologists explored ways to make large-scale computing power available to more users through time-sharing. They experimented with algorithms to optimize the infrastructure, platform, and applications to prioritize CPUs and increase efficiency for end users.

The use of the cloud metaphor for virtualized services dates at least to General Magic in 1994, where it was used to describe the universe of "places" that mobile agents in the Telescript environment could go. As described by Andy Hertzfeld:

"The beauty of Telescript," says Andy, "is that now, instead of just having a device to program, we now have the entire Cloud out there, where a single program can go and travel to many different sources of information and create a sort of a virtual service."

2000s

In August 2006, Amazon created subsidiary Amazon Web Services and introduced its Elastic Compute Cloud (EC2).

In April 2008, Google released the beta version of Google App Engine.

In early 2008, NASA's OpenNebula, enhanced in the RESERVOIR European Commission-funded project, became the first open-source software for deploying private and hybrid clouds, and for the federation of clouds.

By mid-2008, Gartner saw an opportunity for cloud computing "to shape the relationship among consumers of IT services, those who use IT services and those who sell them" and observed that "organizations are switching from company-owned hardware and software assets to per-use service-based models" so that the "projected shift to computing ... will result in dramatic growth in IT products in some areas and significant reductions in other areas."
In 2008, the U.S. National Science Foundation began the Cluster Exploratory program to fund academic research using Google-IBM cluster technology to analyze massive amounts of data.

2010s

In February 2010, Microsoft released Microsoft Azure, which was announced in October 2008.

In July 2010, Rackspace Hosting and NASA jointly launched an open-source cloud-software initiative known as OpenStack. The OpenStack project intended to help organizations offering cloud-computing services running on standard hardware. The early code came from NASA's Nebula platform as well as from Rackspace's Cloud Files platform. As an open-source offering and along with other open-source solutions such as CloudStack, Ganeti, and OpenNebula, it has attracted attention by several key communities. Several studies aim at comparing these open source offerings based on a set of criteria.

On March 1, 2011, IBM announced the IBM SmartCloud framework to support Smarter Planet. Among the various components of the Smarter Computing foundation, cloud computing is a critical part. On June 7, 2012, Oracle announced the Oracle Cloud. This cloud offering is poised to be the first to provide users with access to an integrated set of IT solutions, including the Applications (SaaS), Platform (PaaS), and Infrastructure (IaaS) layers.

In May 2012, Google Compute Engine was released in preview, before being rolled out into General Availability in December 2013.

In 2019, it was revealed that Linux is most used on Microsoft Azure.

MAJOR MILESTONES

- 1999 : Salesforce.com
  - Established the ability to use a simple website on the Internet to deliver enterprise-level applications
- 2002 : Amazon Web Services
  - Featured several cloud-based retail services that included data storage and computation
- 2006 : Amazon's Elastic Compute Cloud (EC2) – the first commercial cloud
  - Enabled small companies to rent computers that would host and run their own applications
- 2006 : Google launches Google Docs
  - End users were directly able to use cloud computing for document sharing purposes.
- 2007 : Dropbox
  - MIT student created this file hosting service that offers file storage and synchronization
- 2009 :
  - Google Apps – example of browser-based enterprise applications
  - Windows Azure – Microsoft's cloud computing platform

EUCALYPTUS

Eucalyptus is a paid and open-source computer software for building Amazon Web Services (AWS)-compatible private and hybrid cloud computing environments, originally developed by the company Eucalyptus Systems. Eucalyptus is an acronym for Elastic Utility Computing Architecture for Linking Your Programs To Useful Systems.

Eucalyptus enables pooling compute, storage, and network resources that can be dynamically scaled up or down as application workloads change. Mårten Mickos was the CEO of Eucalyptus. In September 2014, Eucalyptus was acquired by Hewlett-Packard and then maintained by DXC Technology. After DXC stopped developing the product in late 2017, AppScale Systems forked the code and started supporting Eucalyptus customers.

HISTORY

The software development had its roots in the Virtual Grid Application Development Software project, at Rice University and other institutions from 2003 to 2008. Rich Wolski led a group at the University of California, Santa Barbara (UCSB), and became the chief technical officer at the company headquartered in Goleta, California before returning to teach at UCSB.

Eucalyptus software was included in the Ubuntu 9.04 distribution in 2009. The company was formed in 2009 with $5.5 million in funding by Benchmark Capital to commercialize the software.

The co-founders of Eucalyptus were Rich Wolski (CTO), Dan Nurmi, Neil Soman, Dmitrii Zagorodnov, Chris Grzegorczyk, Graziano Obertelli and Woody Rollins (CEO). Eucalyptus Systems announced a formal agreement with Amazon Web Services in March 2012.

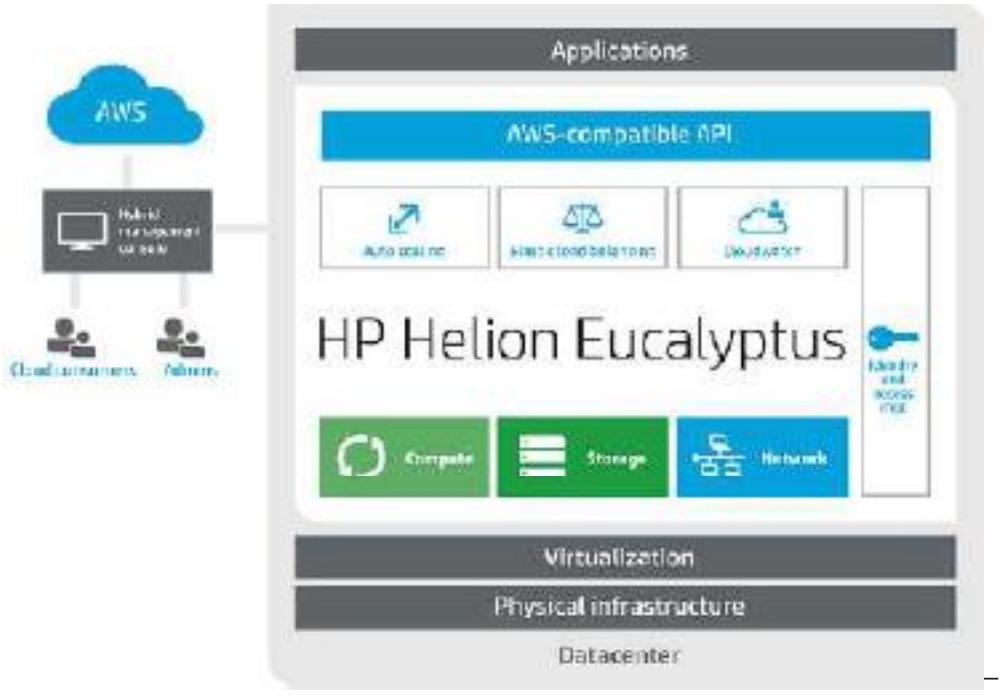
SOFTWARE ARCHITECTURE

Eucalyptus commands can manage either Amazon or Eucalyptus instances. Users can also move instances between a Eucalyptus private cloud and the Amazon Elastic Compute Cloud to create a hybrid cloud. Hardware virtualization isolates applications from computer hardware details.

Eucalyptus uses the terminology:

- ***Images*** – An image is a fixed collection of software modules, system software, application software, and configuration information that is started from a known baseline (immutable/fixed). When bundled and uploaded to the Eucalyptus cloud, this becomes a *Eucalyptus machine image (EMI)*.

- ***Instances*** – When an image is put to use, it is called an instance. The configuration is executed at runtime, and the Cloud Controller decides where the image will run, and storage and networking is attached to meet resource needs.
- ***IP addressing*** – Eucalyptus instances can have public and private IP addresses. An IP address is assigned to an instance when the instance is created from an image. For instances that require a persistent IP address, such as a web-server, Eucalyptus supplies elastic IP addresses. These are pre-allocated by the Eucalyptus cloud and can be reassigned to a running instance.
- ***Security*** – TCP/IP security groups share a common set of firewall rules. This is a mechanism to firewall off an instance using IP address and port block/allow functionality. Instances are isolated at TCP/IP layer 2. If this were not present, a user could manipulate the networking of instances and gain access to neighboring instances violating the basic cloud tenet of instance isolation and separation.
- ***Networking*** – There are three networking modes. In Managed Mode, Eucalyptus manages a local network of instances, including security groups and IP addresses. In System Mode, Eucalyptus assigns a MAC address and attaches the instance's network interface to the physical network through the Node Controller's bridge. System Mode does not offer elastic IP addresses, security groups, or VM isolation. In Static Mode, Eucalyptus assigns IP addresses to instances. Static Mode does not offer elastic IPs, security groups, or VM isolation.



- A user of Eucalyptus is assigned an identity, and identities can be grouped together for access control.

EUCALYPTUS COMPONENTS

- The *Cloud Controller (CLC)* is a Java program that offers EC2-compatible interfaces, as well as a web interface to the outside world. In addition to handling incoming requests, the CLC acts as the administrative interface for cloud management and performs high-level resource scheduling and system accounting. The CLC accepts user API requests from command-line interfaces like euca2ools or GUI-based tools like the Eucalyptus User Console and manages the underlying compute, storage, and network resources. Only one CLC can exist per cloud and it handles authentication, accounting, reporting, and quota management.
- *Walrus*, also written in Java, is the Eucalyptus equivalent to AWS Simple Storage Service (S3). Walrus offers persistent storage to all of the virtual machines in the Eucalyptus cloud and can be used as a simple HTTP put/get storage as a service solution. There are no data type restrictions for Walrus, and it can contain images (i.e., the building blocks used to launch virtual machines), volume snapshots (i.e., point-in-time copies), and application data. Only one Walrus can exist per cloud.
- The *Cluster Controller (CC)* is written in C and acts as the front end for a cluster within a Eucalyptus cloud and communicates with the Storage Controller and Node Controller. It manages instance (i.e., virtual machines) execution and Service Level Agreements (SLAs) per cluster.
- The *Storage Controller (SC)* is written in Java and is the Eucalyptus equivalent to AWS EBS. It communicates with the Cluster Controller and Node Controller and manages Eucalyptus block volumes and snapshots to the instances within its specific cluster. If an instance requires writing persistent data to memory outside of the cluster, it would need to write to Walrus, which is available to any instance in any cluster.
- The *VMware Broker* is an optional component that provides an AWS-compatible interface for VMware environments and physically runs on the Cluster Controller. The VMware Broker overlays existing ESX/ESXi hosts and transforms Eucalyptus Machine Images (EMIs) to VMware virtual disks. The VMware Broker mediates interactions between the Cluster Controller and VMware and can connect directly to either ESX/ESXi hosts or to vCenter Server.
- The *Node Controller (NC)* is written in C and hosts the virtual machine instances and manages the virtual network endpoints. It downloads and caches images from Walrus as well as creates and caches instances. While there is no theoretical limit to the number of Node Controllers per cluster, performance limits do exist.

FUNCTIONALITY

The Eucalyptus User Console provides an interface for users to self-service provision and configure compute, network, and storage resources. Development and test teams can manage virtual instances using built-in key management and encryption capabilities. Access to virtual instances is available using familiar SSH and RDP mechanisms. Virtual instances with application configuration can be stopped and restarted using encrypted boot from EBS capability.

IaaS service components Cloud Controller, Cluster Controller, Walrus, Storage Controller, and VMware Broker are configurable as redundant systems that are resilient to multiple types of failures. Management state of the cloud machine is preserved and reverted to normal operating conditions in the event of a hardware or software failure.

Eucalyptus can run multiple versions of Windows and Linux virtual machine images. Users can build a library of Eucalyptus Machine Images (EMIs) with application metadata that are decoupled from infrastructure details to allow them to run on Eucalyptus clouds. Amazon Machine Images are also compatible with Eucalyptus clouds. VMware Images and vApps can be converted to run on Eucalyptus clouds and AWS public clouds.

Eucalyptus user identity management can be integrated with existing Microsoft Active Directory or LDAP systems to have fine-grained role based access control over cloud resources.

Eucalyptus supports storage area network devices to take advantage of storage arrays to improve performance and reliability. Eucalyptus Machine Images can be backed by EBS-like persistent storage volumes, improving the performance of image launch time and enabling fully persistent virtual machine instances. Eucalyptus also supports direct-attached storage.

Eucalyptus 3.3 offers new features for AWS compatibility. These include resource tagging, which allows application developers and cloud administrators to assign customizable metadata tags to resources such as firewalls, load balancers, Web servers, and individual workloads to better identify them. Eucalyptus 3.3 also supports an expanded set of instance types to more closely.

Eucalyptus 3.4, released on October 24, 2013, added new features including improved image management and migration tools, capabilities for warm upgrades, a hybrid cloud user console to manage both Eucalyptus and AWS resources, Identity and Access Management (IAM) roles, and improved High Availability (HA) capabilities.[17]

Faststart demonstration configurations that allow you to set up your own private cloud quickly with as few steps as possible are available.

NIMBUS

Nimbus is a toolkit that, once installed on a cluster, provides an infrastructure as a service cloud to its client via WSRF-based or Amazon EC2 WSDL web service APIs. Nimbus is free and open-source software, subject to the requirements of the Apache License, version 2.

Nimbus supports both the hypervisors Xen and KVM and virtual machine schedulers Portable Batch System and Oracle Grid Engine. It allows deployment of self-configured virtual clusters via contextualization.[1] It is configurable with respect to scheduling, networking leases, and usage accounting.

Nimbus is a powerful toolkit focused on converting a computer cluster into an Infrastructure-as-a-Service (IaaS) cloud for scientific communities. Essentially, it allows a deployment and configuration of virtual machines (VMs) on remote resources to create an environment suitable for the users' requirements. Being written in Python and Java, it is totally free and open-source software, released under the Apache License.

Nimbus consists of two basic products:

- Nimbus Infrastructure is an open source EC2/S3-compatible IaaS solution with features that benefit scientific community interests, like support for auto-configuring clusters, proxy credentials, batch schedulers, best-effort allocations, etc.
- Nimbus Platform is an integrated set of tools for a multi-cloud environment that automates and simplifies the work with infrastructure clouds (deployment, scaling, and management of cloud resources) for scientific users.

This toolkit is compatible with Amazon's Network Protocols via EC2 based clients, S3 REST API clients, as well as SOAP API and REST API that have been implemented in Nimbus. Also it provides support for X509 credentials, fast propagation, multiple protocols, and compartmentalized dependencies. Nimbus features flexible user, group and workspaces management, request authentication and authorization, and per-client usage tracking.

NIMBUS KEEPS DEVELOPERS, PROVIDERS AND USERS SATISFIED

To open all power and versatility of IaaS to scientific users Nimbus project developers targeted the main three goals and their open source implementations:

- ***Give capabilities to providers of resources for private or community IaaS clouds development***. The Nimbus Workspace Service enables lease of computational resources by deploying virtual machines on those resources. Cumulus is an open source implementation of the S3 REST API that was built for scalable quota-based storage cloud implementation and multiple storage cloud configuration.
- ***Give capabilities to users for IaaS clouds application***. Among Nimbus scaling tools (users can automatically scale across multiple distributed providers) the Nimbus Context Broker is especially robust. It coordinates large virtual cluster launches automatically and repeatedly using a common configuration and security context across resources.
- ***Give capabilities to developers for extension, experimentation and customization of IaaS***. For instance, the Workspace Service can support several virtualization implementations (either Xen or KVM), resource management options (including schedulers such as Portable Batch System), interfaces (including compatibility with Amazon EC2), and other options.

Different combinations of these tools assist users in rapid development of custom community-specific solutions. For example, Nimbus enables users to build multiple virtual machines and deploy them throughout the cloud, so that they will co-operate and supplement each other.

User can connect a virtual machine to resources on a cloud regardless its owner/provider. Or, with the Nimbus cloud client, user can provision customized compute nodes (a workspace) and manage it using a leasing model based on the EC2 service.

Such flexibility and on-demand computing power is particularly essential for specific computational jobs and data-intensive research.

Nimbus is a set of robust tools providing Storage Cloud Service and Infrastructure-as-a-Service capabilities to the scientific community.

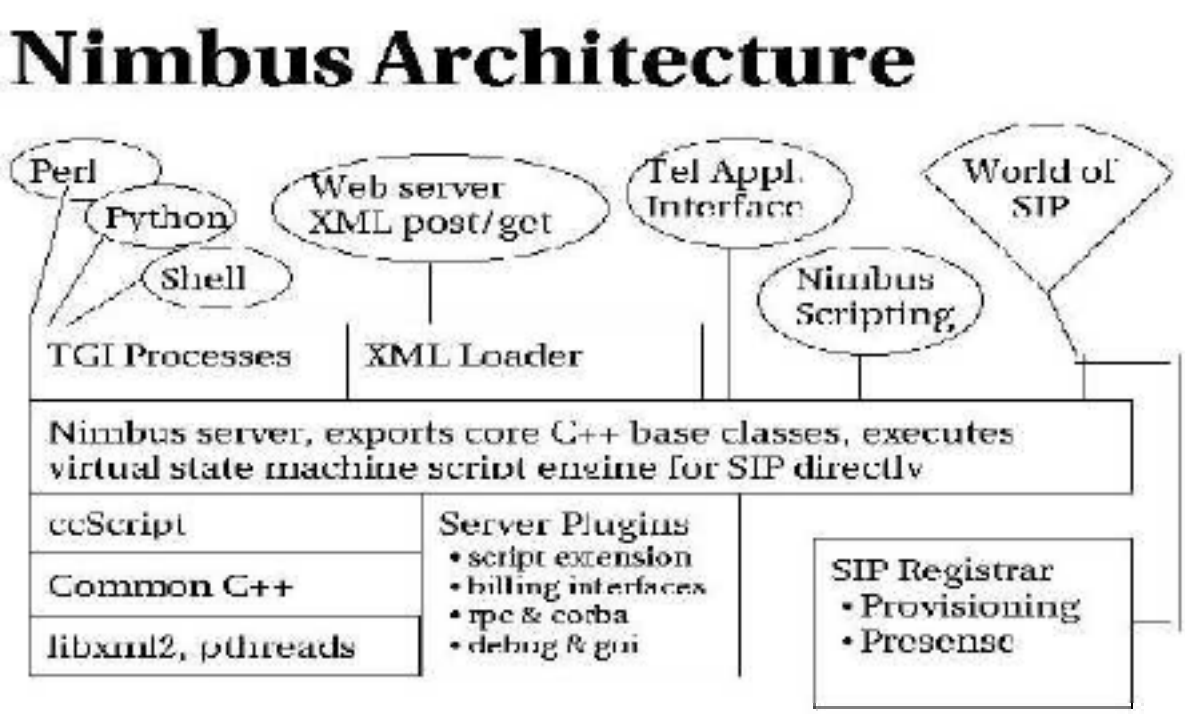
It is highly configurable, including scheduling, networking leases, usage accounting, remote deployment and lifecycle management of VMs.

Moreover, Nimbus allows deployment of self-configured virtual clusters via contextualization.

Nimbus is an open-source toolkit to convert a computer cluster into an Infrastructure-as-a-Service cloud to provide compute cycles for scientific communities. It allows a client to lease remote resources by deploying virtual machines (VMs) on those resources and configuring them to represent an environment desired by the user.

Nimbus is comprised of two products:

- **Nimbus Infrastructure** is an open source EC2/S3-compatible Infrastructure-as-a-Service implementation specifically targeting features of interest to the scientific community such as support for proxy credentials, batch schedulers, best-effort allocations and others.
- **Nimbus Platform** is an integrated set of tools, operating in a multi-cloud environment, that deliver the power and versatility of infrastructure clouds to scientific users. Nimbus Platform allows you to reliably deploy, scale, and manage cloud resources.
  The Nimbus cloud client allows the user to provision customized compute nodes, called a workspace, and maintain full control over it using a leasing model based on the Amazon's Elastic Compute Cloud (EC2) service.
  The Nimbus cloud-computing infrastructure allows scientists working on data-intensive research to create and use such virtual machines with a cloud provider.
  Nimbus also allows users to create multiple virtual machines to complete specific computational jobs that can be deployed throughout the cloud and still work in tandem with each other. This flexibility allows a user to configure a virtual machine and then connect it to resources on a cloud, regardless of who is providing the cloud.



NEBULA

OpenNebula is a cloud computing platform for managing heterogeneous distributed data center infrastructures. The OpenNebula platform manages a data center's virtual infrastructure to build private, public and hybrid implementations of infrastructure as a service.

The two primary uses of the OpenNebula platform are data center virtualization solutions and cloud infrastructure solutions. The platform is also capable of offering the cloud infrastructure necessary to operate a cloud on top of existing infrastructure management solutions.

OpenNebula is free and open-source software, subject to the requirements of the Apache License version 2.

HISTORY

The OpenNebula Project was started as a research venture in 2005 by Ignacio M. Llorente and Ruben S. Montero. The first public release of the software occurred in 2008. The goals of the research were to create efficient solutions for managing virtual machines on distributed infrastructures. It

was also important that these solutions had the ability to scale at high levels. Open-source development and an active community of developers have since helped mature the project. As the project matured it began to become more and more adopted and in March 2010 the primary writers of the project founded C12G Labs, now known as OpenNebula Systems, which provides value-added professional services to enterprises adopting or utilizing OpenNebula.

Description

OpenNebula orchestrates storage, network, virtualization, monitoring, and security technologies to deploy multi-tier services (e.g. compute clusters) as virtual machines on distributed infrastructures, combining both data center resources and remote cloud resources, according to allocation policies. According to the European Commission's 2010 report "... only few cloud dedicated research projects in the widest sense have been initiated most prominent amongst them probably OpenNebula ...".

The toolkit includes features for integration, management, scalability, security and accounting. It also claims standardization, interoperability and portability, providing cloud users and administrators with a choice of several cloud interfaces (Amazon EC2 Query, OGF Open Cloud Computing Interface and vCloud) and hypervisors (KVM, LXD and VMware vCenter), and can accommodate multiple hardware and software combinations in a data center.

OpenNebula is sponsored by OpenNebula Systems (formerly C12G).

OpenNebula is widely used by a variety of industries, including cloudproviders, telecommunication, information technology services, government, banking, gaming, media, hosting, supercomputing, research laboratories, and international research projects. The OpenNebula Project is also used by some other cloud solutions as a cloud engine.OpenNebula has grown significantly since going public and now has many notable users from a variety of industries. Notable users from the telecommunications and internet industry include Akamai, Blackberry, Fuze, Telefónica, and INdigital.

Users in the information technology industry include CA Technologies, Hewlett Packard Enterprise, Hitachi Vantara, Informatica, CentOS, Netways, Ippon Technologies, Terradue 2.0, Unisys, MAV Technologies, Liberologico, Etnetera, EDS Systems, Inovex, Bosstek, Datera, Saldab, Hash Include, Blackpoint, Deloitte, Sharx dc, Server Storage Solutions[buzzword], and NTS. Government solutions[buzzword] utilizing the OpenNebula Project include the National Central Library of Florence, bDigital, Deutsch E-Post, RedIRIS, GRNET, InstitutoGeograficoNacional, CSIC, Gobex, ASAC Communications, KNAW, Junta De Andalucia, Flanders Environmental Agency, red.es, CENATIC, Milieuinfo, SIGMA, and Computaex. Notable users in the financial sector include TransUnion, Produpan, Axcess Financial, Farm Credit Services of America, and Nasdaq Dubai.

Media and gaming users include BBC, Unity, R.U.R., Crytek, iSpot.tv, and Nordeus. Hosting providers include ON VPS, NBSP, Orion VM, CITEC, LibreIT, Quobis, Virtion, OnGrid, Altus, DMEx, LMD, HostColor, Handy Networks, BIT, Good Hosting, Avalon, nooswps, Opulent Cloud, PTisp, Ungleich.ch, TAS France, TeleData, CipherSpace, Nuxit, Cyon, Tentacle Networks, Virtiso BV, METANET, e-tugra, lunacloud, todoencloud, Echelon, Knight Point Systems, 2 Twelve Solutions, and flexyz. SaaS and enterprise users include Scytl, LeadMesh, OptimalPath, RJMetrics, Carismatel, Sigma, GLOBALRAP, Runtastic, MOZ, Rentalia, Vibes, Yuterra, Best Buy, Roke, Intuit, Securitas Direct, trivago, and Booking.com.

Science and academia implementations include FAS Research Computing at Harvard University, FermiLab, NIKHEF, LAL CNRS, DESY, INFN, IPB Halle, CSIRO, fccn, AIST, KISTI, KIT, ASTI, FatecLins, MIMOS, SZTAKI, Ciemat, SurfSARA, ESA, NASA, ScanEX, NCHC, CESGA, CRS4, PDC, CSUC, Tokyo Institute of Technology, CSC, HPCI, Cerit-SC, LRZ, PIC, Telecom SUD Paris, Universidade Federal de Ceara, InstitutoSuperiore Mario Barella, Academia Sinica, UNACHI, UCM, UniversiteCatholique de Louvain, Universite de Strasbourg, ECMWF, EWE Tel, INAFTNG, TeideHPC, Cujae, and Kent State University. Cloud products using OpenNebula include ClassCat, HexaGrid, NodeWeaver, Impetus, and ZeroNines.

Internal architecture

Basic components

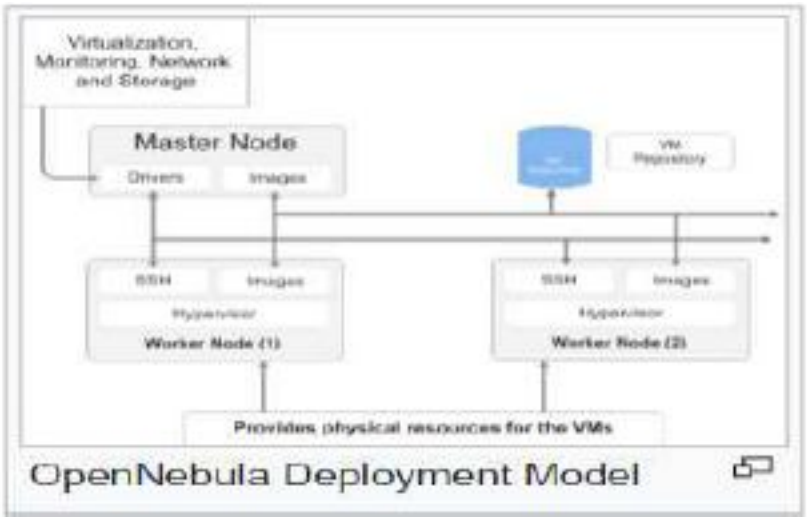
- **Host**: Physical machine running a supported hypervisor.
- **Cluster**: Pool of hosts that share datastores and virtual networks.
- **Template**: Virtual Machine definition.
- **Image**: Virtual Machine disk image.
- **Virtual Machine**: Instantiated Template. A Virtual Machine represents one life-cycle, and several Virtual Machines can be created from a single Template.
- **Virtual Network**: A group of IP leases that VMs can use to automatically obtain IP addresses. It allows the creation of Virtual Networks by mapping over the physical ones. They will be available to the VMs through the corresponding bridges on hosts. Virtual network can be defined in three different parts:

1. Underlying of physical network infrastructure.
2. The logical address space available (IPv4, IPv6, dual stack).
3. Context attributes (e.g. net mask, DNS, gateway). OpenNebula also comes with a Virtual Router appliance to provide networking services like DHCP, DNS etc.

Components and Deployment Model

The OpenNebula Project's deployment model resembles classic cluster architecture which utilizes

- **A front-end (master node)**
- **Hypervisor enabled hosts (worker nodes)**
- **Datastores**
- **A physical network**



Front-end machine

The master node, sometimes referred to as the front-end machine, executes all the OpenNebula services. This is the actual machine where OpenNebula is installed. OpenNebula services on the front-end machine include the management daemon (oned), scheduler (sched), the web interface server (Sunstone server), and other advanced components. These services are responsible for queuing, scheduling, and submitting jobs to other machines in the cluster. The master node also provides the mechanisms to manage the entire system. This includes adding virtual machines, monitoring the status of virtual machines, hosting the repository, and transferring virtual machines when necessary. Much of this is possible due to a monitoring subsystem which gathers information such as host status, performance, and capacity use. The system is highly scalable and is only limited by the performance of the actual server.

Hypervisor enabled-hosts

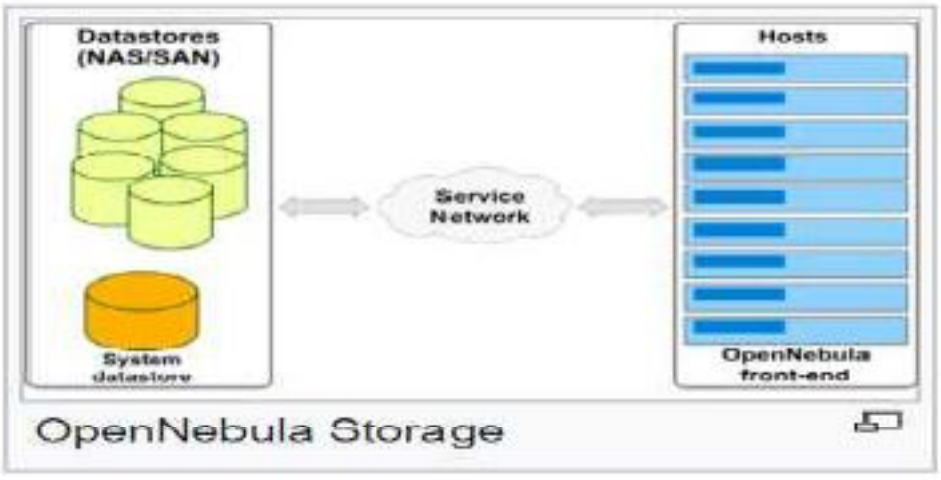
The worker nodes, or hypervisor enabled-hosts, provide the actual computing resources needed for processing all jobs submitted by the master node. OpenNebula hypervisor enabled-hosts use a virtualization hypervisor such as Vmware, Xen, or KVM. The KVM hypervisor is natively supported and used by default. Virtualization hosts are the physical machines that run the virtual machines and various platforms can be used with OpenNebula. A Virtualization Subsystem interacts with these hosts to take the actions needed by the master node.

Storage

The datastores simply hold the base images of the Virtual Machines. The datastores must be accessible to the front-end; this can be accomplished by using one of a variety of available technologies such as NAS, SAN, or direct attached storage.

Three different datastore classes are included with OpenNebula, including system datastores, image datastores, and file datastores. System datastores hold the images used for running the virtual machines. The images can be

complete copies of an original image, deltas, or symbolic links depending on the storage technology used. The image datastores are used to store the disk image repository. Images from the image datastores are moved to or from the system datastore when virtual machines are deployed or manipulated. The file datastore is used for regular files and is often used for kernels, ram disks, or context files.

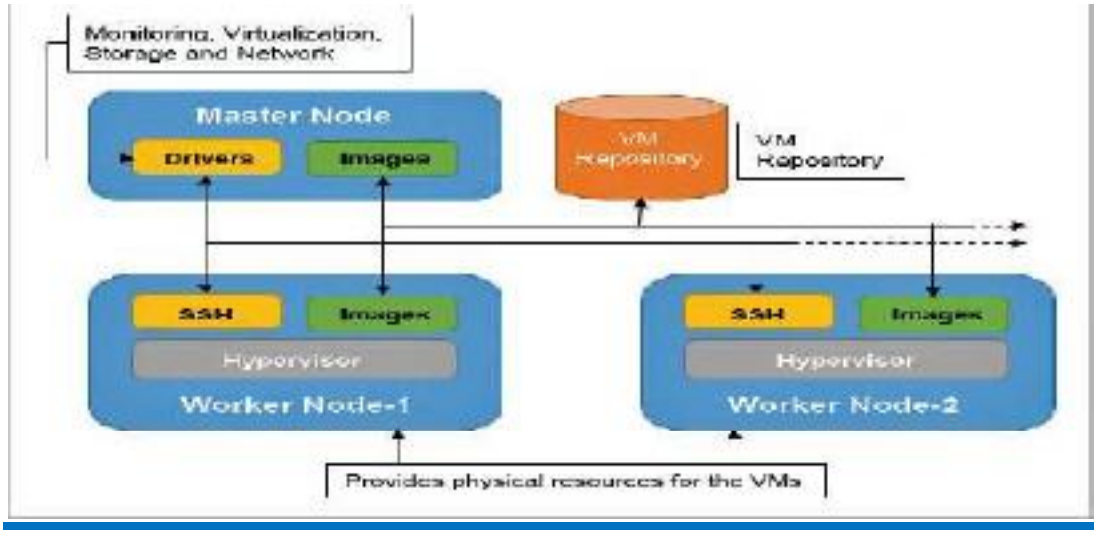


Physical networks

Physical networks are required to support the interconnection of storage servers and virtual machines in remote locations. It is also essential that the front-end machine can connect to all the worker nodes or hosts. At the very least two physical networks are required as OpenNebula requires a service network and an instance network.

The front-end machine uses the service network to access hosts, manage and monitor hypervisors, and to move image files. The instance network allows the virtual machines to connect across different hosts.

The network subsystem of OpenNebula is easily customizable to allow easy adaptation to existing data centers.



CLOUDSIM

CloudSim is a simulation toolkit that supports the modeling and simulation of the core functionality of cloud, like job/task queue, processing of events, creation of cloud entities(datacenter, datacenter brokers, etc), communication between different entities, implementation of broker policies, etc. This toolkit allows to:

- Test application services in a repeatable and controllable environment.
- Tune the system bottlenecks before deploying apps in an actual cloud.
- Experiment with different workload mix and resource performance scenarios on simulated infrastructure for developing and testing adaptive application provisioning techniques

Core **features of CloudSim** are:

- The Support of modeling and simulation of large scale computing environment as federated cloud data centers, virtualized server hosts, with customizable policies for provisioning host resources to virtual machines and energy-aware computational resources
- It is a self-contained platform for modeling cloud's service brokers, provisioning, and allocation policies.
- It supports the simulation of network connections among simulated system elements.
- Support for simulation of federated cloud environment, that inter-networks resources from both private and public domains.

- Availability of a virtualization engine that aids in the creation and management of multiple independent and co-hosted virtual services on a data center node.
- Flexibility to switch between space shared and time shared allocation of processing cores to virtualized services.

CloudSim is a framework for modeling and simulation of cloud computing infrastructures and services. Originally built primarily at the Cloud Computing and Distributed Systems (CLOUDS) Laboratory the University of Melbourne, Australia, CloudSim has become one of the most popular open source cloud simulators in the research and academia. CloudSim is completely written in Java.

FEATURES OF CLOUDSIM SIMULATION TOOLKIT

1. Support for modeling and simulation of large-scale Cloud computing environments, including data centers, on a single physical computing node(could be a desktop, laptop, or server machine).
2. A self-contained platform for modeling Clouds, service brokers, provisioning, and allocation policies.
3. Facilitates the simulation of network connections across the simulated system elements.
4. Facility for simulation of federated Cloud environment that inter-networks resources from both private and public domains, a feature critical for research studies related to Cloudbursts and automatic application scaling.
5. Availability of a virtualization engine that facilitates the creation and management of multiple, independent, and co-hosted virtualized services on a data center node.
6. Flexibility to switch between space-shared and time-shared allocation of processing cores to virtualized services.

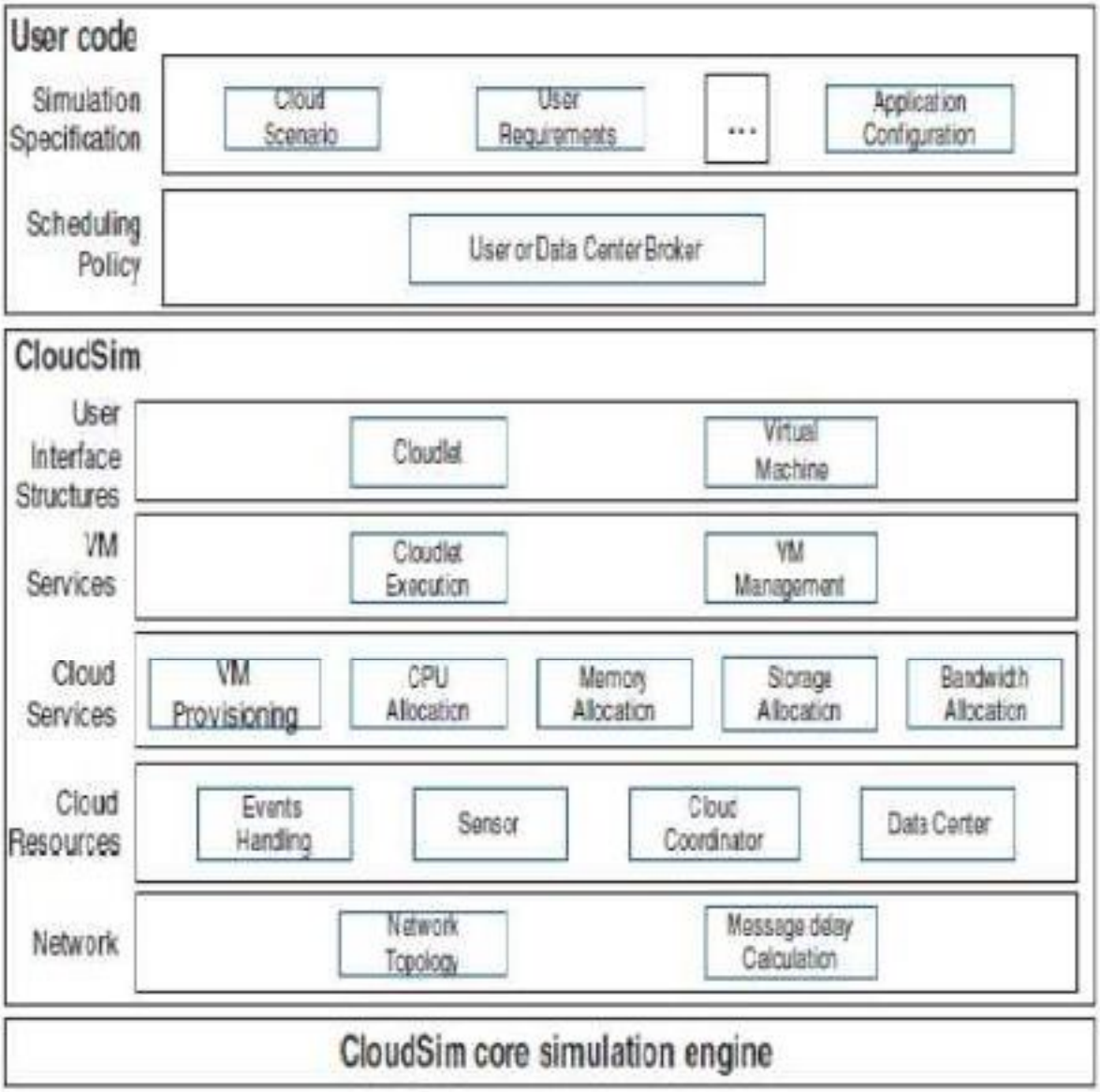
All these features would help in accelerating the development, testing and deployment of potential resource/application provisioning policies/algorithms for Cloud Computing based systems.

CLOUDSIM ARCHITECTURE

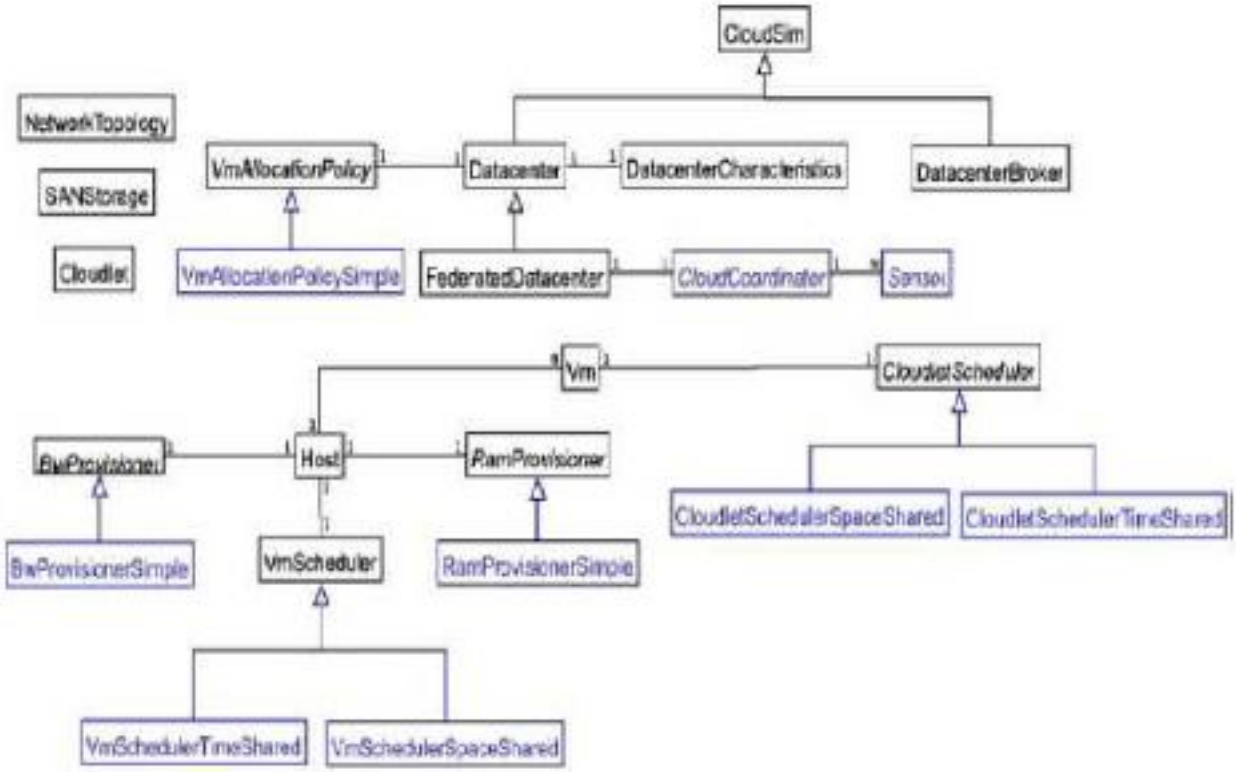
The **CloudSim Core simulation engine** provides support for modeling and simulation of virtualized Cloud-based data center environments including queuing and processing of events, creation of cloud system entities (like data center, host, virtual machines, brokers, services, etc.) communication between components and management of the simulation clock.

The **CloudSim layer** provides dedicated management interfaces for Virtual Machines, memory, storage, and bandwidth. Also, it manages the other fundamental issues, such as provisioning of hosts to Virtual Machines, managing application execution, and monitoring dynamic system state(e.g. Network topology, sensors, storage characteristics, etc), etc.

The **User Code layer** is a custom layer where the user writes their own code to redefine the characteristics of the stimulating environment as per their new research findings.



DESIGN AND IMPLEMENTATION OF CLOUDSIM



1. **Cloudlet**: This class model[define specific attributes such as length of instruction, input/output filesize, no of processor required, etc] the Cloud-based application services [program based tasks] such as content delivery, social networking, etc. CloudSim implements the complexity of an application in terms of its computational requirements. As a developer, we know that every individual executable application/service workload has a pre-defined instruction length and requires certain network data flow [both pre and post fetches] overhead that it needs to undertake during its life cycle. this class allows modeling all the above-said requirements
2. **CloudletScheduler**: This is responsible for the implementation of different policies that determine the share of processing power among Cloudlets in a VM. ***There are two types of provisioning policies offered: space-shared (using CloudetSchedulerSpaceShared class) and time-shared (using CloudletSchedulerTimeShared class).***
3. **Datacenter**: This class model the core infrastructure-level services [i.e. hardware] that are offered by Cloud providers [Amazon, Azure, and App Engine]. ***It encapsulates a set of hosts(resembling server machine model) instances that***

can either be homogeneous or heterogeneous concerning their hardware configurations (memory, cores, capacity, and storage). Also, every Datacenter component takes care of generalized application provisioning that enforces a set of policies for the allocation of bandwidth, memory, and storage devices to hosts and its related VMs.

4. **DatacenterBroker or Cloud Broker**: This class model a broker, which is responsible for mediating negotiations between SaaS and Cloud providers and such negotiations are driven by QoS requirements. The broker class acts on behalf of applications. Its prime role is to query the CIS to discover suitable resources/services and undertakes negotiations for the allocation of resources/services that can fulfill the application's QoS needs. This class must be extended for evaluating and testing custom brokering policies.
5. **DatacenterCharacteristics**: This class contains configuration information of data center resources like the available host list, the fine-grained cost for each resource type, etc.
6. **Host**: This class model a physical resource such as a computer or storage server. It encapsulates important information such as the amount of memory and storage, a list and type of processing cores (if it is a multi-core machine), an allocation of policy for provisioning the compute, memory and bandwidth to the VMs.
7. **NetworkTopology**: This class contains the information for inducing network behavior (latencies) in the simulation. It stores the topology information, which is generated using the BRITE topology generator.
8. **RamProvisioner**: This is an abstract class that represents the provisioning policy for allocating primary memory (RAM) to Virtual Machines. The execution and deployment of VM on a host are feasible only if the RamProvisioner component approves that the host has the required amount of free memory. The RamProvisionerSimple does not enforce any limitation on the amount of memory that a VM may request. The RAM resource request will be rejected if it is beyond the available capacity.
9. **BwProvisioner**: The main role of this component is to undertake the allocation of network bandwidths to a set of competing VMs that are deployed across the data center. Cloud system developers and researchers can extend this class with their policies (priority, QoS) to reflect the needs of their applications.

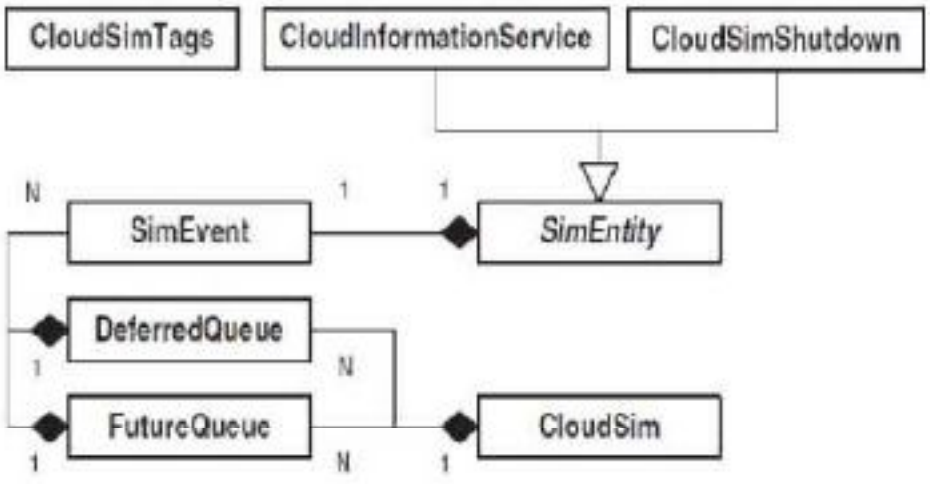
10. **Vm**: This class model a Virtual Machine (VM), which is managed and hosted by a Cloud host component. Every VM component has access to a component that stores the following characteristics related to a VM (i.e.) accessible memory, processor, storage size, and the VM's internal provisioning policy that is extended from an abstract class called the CloudletScheduler.
11. **VmAllocationPolicy**: This is an abstract class that represents a provisioning policy to be utilized by VM Monitor for mapping VMs to hosts. The primary role is to select the best fit host in a data center that meets the memory, storage, and availability requirement for VM deployment mapping.
12. **VmScheduler**: This is an abstract class implemented by a Host component that models the allocation policies (space-shared, time-shared) defining the rules for processor cores allocations to VMs. To accommodate application-specific processor sharing policies, the class functionality can be extended to define a new set of provisioning rules.
13. **CloudSim**: This is the prime class, with the role of managing entity event queues and controlling the sequential execution of simulation events. Every event that is generated by the CloudSim entity at run-time is stored in the queue called future events. These events are sorted by their time parameters and are enqueued into the future queue. Next, the events that are scheduled at each step of the simulation are removed from the future events queue and transferred to the deferred event queue. Following this, an event processing method is invoked for each entity, which chooses events from the deferred event queue and performs appropriate actions. Such an organization allows flexible management of simulation and provides the following powerful capabilities:
    a. Deactivation (hold/pause) of entities.
    b. Context switching of entities between different states (e.g. waiting to active). Pause and resume the process of simulation.
    c. Creation of new entities at run-time.
    d. Aborting and restarting simulation at run-time.
14. **FutureQueue**: This class implements the future event queue accessed by CloudSim and ***acts as a ready queue to the simulation engine.***
15. **DeferredQueue**: This class implements the deferred event queue used by CloudSim and hold such events which are failed or paused. ***It acts as a wait queue of the***

simulation engine, where preempted resource requests are kept.

16. **CloudInformationService**: A CIS is an entity that provides resource registration, indexing, and discovering capabilities. CIS supports two basic primitives:
    a. **publish()**, on the start of simulation it allows entities to register themselves with CIS
    b. **search()**, allows Brokers in discovering resources status and endpoint addresses of other entities. This entity also acts as a notification service to the other entities about the end of the simulation.
17. **SimEntity**: This is an abstract class, which represents a simulation entity [such as DataCenter, DatacenterBroker, etc]] that is able to send messages to other entities and processes received messages as well as fire and handle events. ***SimEntity class provides the ability to schedule new events and send messages to other entities, where network delay is calculated according to the BRITE model. Once created, entities automatically register with CIS.*** All entities must extend this class and override its three core methods:
    a. **startEntity**(), which define actions for entity initialization.
    b. **processEvent**(), which defines actions processing of each called event(s) with respect to the entity.
    c. **shutdownEntity**(),which define actions for entity destruction.
18. **CloudSimTags**. This class contains various static event/command tags that indicate the type of action that needs to be undertaken by CloudSim entities when they receive or send events.
19. **SimEvent**: This entity represents a simulation event that is passed between two or more entities. SimEvent stores the following information about an event:
    1. type,
    2. init time,
    3. time at which the event should occur,
    4. finish time,
    5. time at which the event should be delivered to its destination entity,
    6. IDs of the source and destination entities,
    7. the tag of the event, and
    8. Data that have to be passed to the destination entity.

CloudSimShutdown: This is an entity class that waits for the termination of all end-user submitted cloudlets(tasks) and

broker entities events, and once detected, then signals the end of simulation to CIS.



FEATURES OF CLOUDSIM:

Overview of CloudSim functionalities:

- support for modeling and simulation of **large scale Cloud computing data centers**
- support for modeling and simulation of virtualized server hosts, with customizable policies for provisioning host resources to virtual machines
- support for modeling and simulation of **application containers**
- support for modeling and simulation of **energy-aware computational resources**
- support for modeling and simulation of data center network topologies and message-passing applications
- support for modeling and simulation of **federated clouds**
- support for dynamic insertion of simulation elements, stop and resume of simulation
- support for **user-defined policies for allocation of hosts to virtual machines** and policies for allocation of host resources to virtual machines

H/W

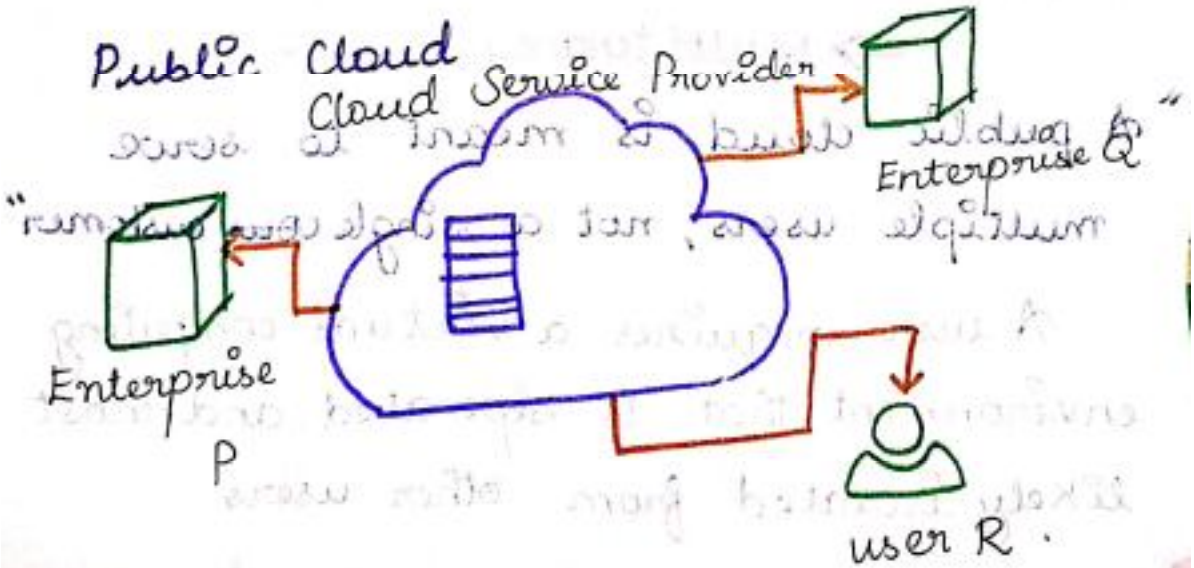
Types Of Cloud

Cloud Computing:

Cloud Computing is an Internet based computing in which shared pool of resources are available over a broad network access, these resources can be provisioned or released with minimum management efforts and service provider interaction.

Four Types of Cloud

1. Public Cloud
2. Private cloud
3. Hybrid Cloud
4. Community cloud.



Public cloud are managed by third parties.

These third parties provide cloud services over the internet to public. These services are available as pay-as-you-go billing mode.

They offer solution for minimizing IT infrastructure.

It acts as a good option for handling peak loads on the local infrastructure.

They are a goto option for SMALL ENTERPRISES which are able to start their business without large upfront investments as they completely rely on the public infrastructure for their IT needs.

Fundamental Characteristics

⇒ Multi tenancy

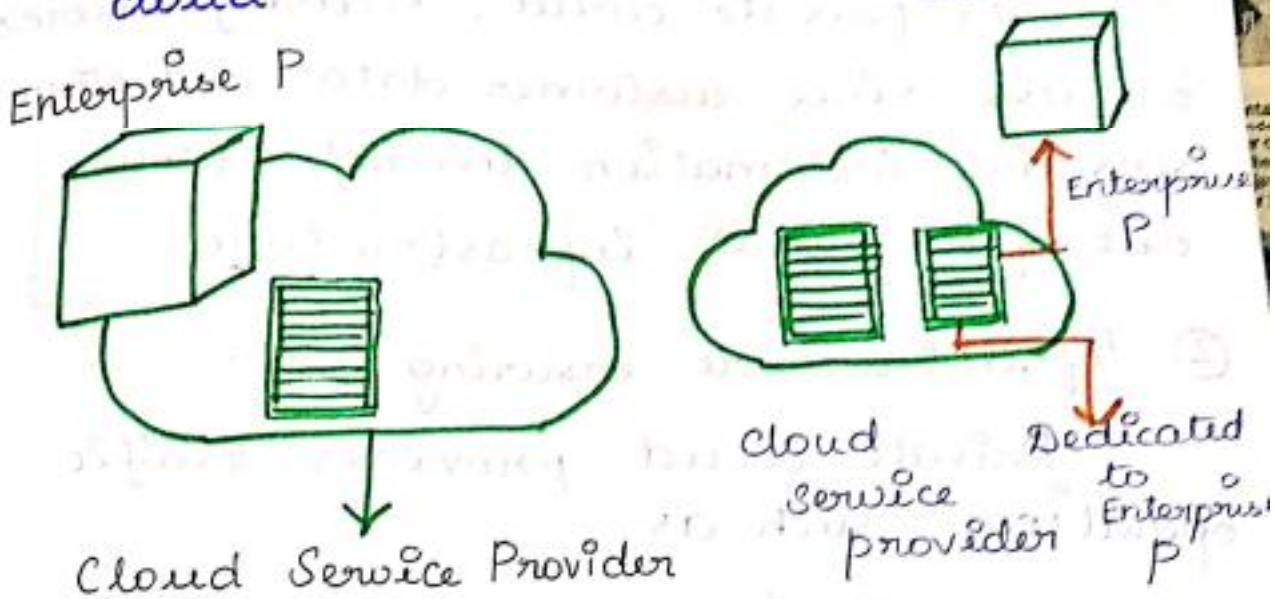
"A public cloud is meant to serve multiple users, not a single customer"

A user requires a virtual computing environment that is seperated, and most likely isolated, from other users.

Private cloud

On-Premise Private cloud

Externally hosted Private cloud



Private clouds are distributed systems that work on private infrastructure and providing user with dynamic provisioning of computing resources.

Instead of pay-as-you-go model in public clouds, there could be other schemes in that taken into account the usage of the cloud and proportionally billing the different departments or sections of an enterprise.

Advantages:

* customer information protection
* Infrastructure ensuring SLA's.
* Compliance with standard

procedures and operations.

① Customer information protection:

In private cloud, security concerns are less since customer data and other sensitive information does not flow out of a private infrastructure.

② Infrastructure ensuring SLA's:

Private cloud provides specific operations such as

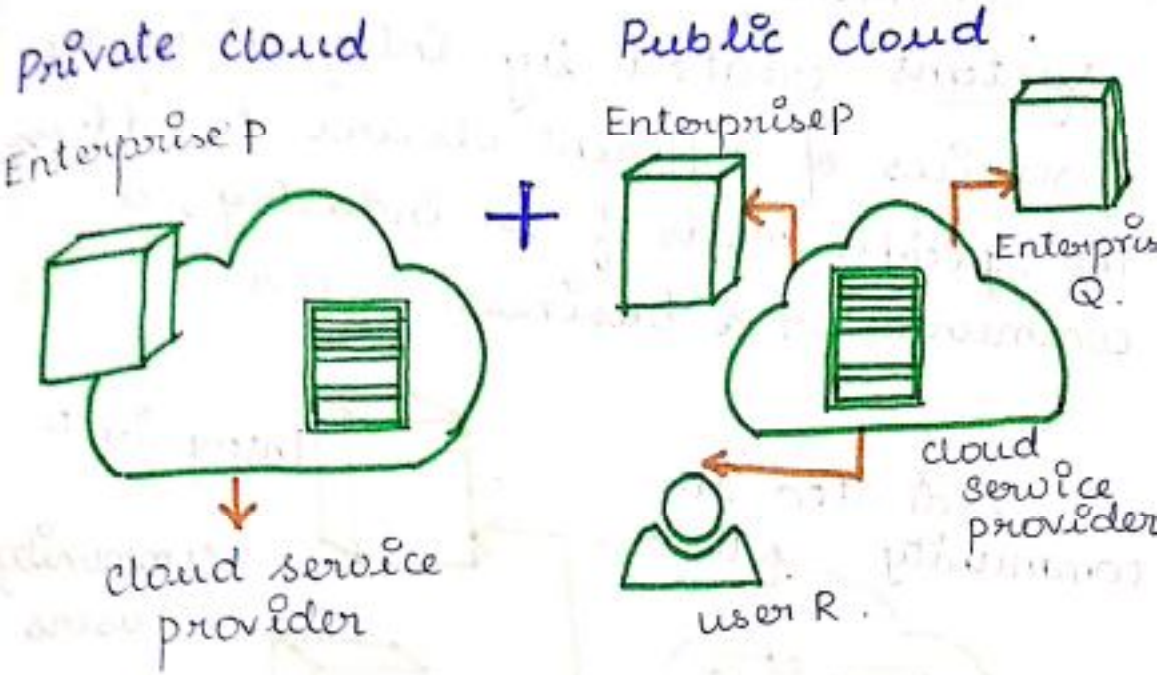
Data replication	Disaster recovery
Appropriate clustering	Maintanence
System monitoring	Other uptime services.

③ Compliance with standard procedures and operations:

Specific procedures have to be put in place when executing and deploying applications to third party compliance standards.

This is not possible in case of public cloud.

Hybrid Cloud

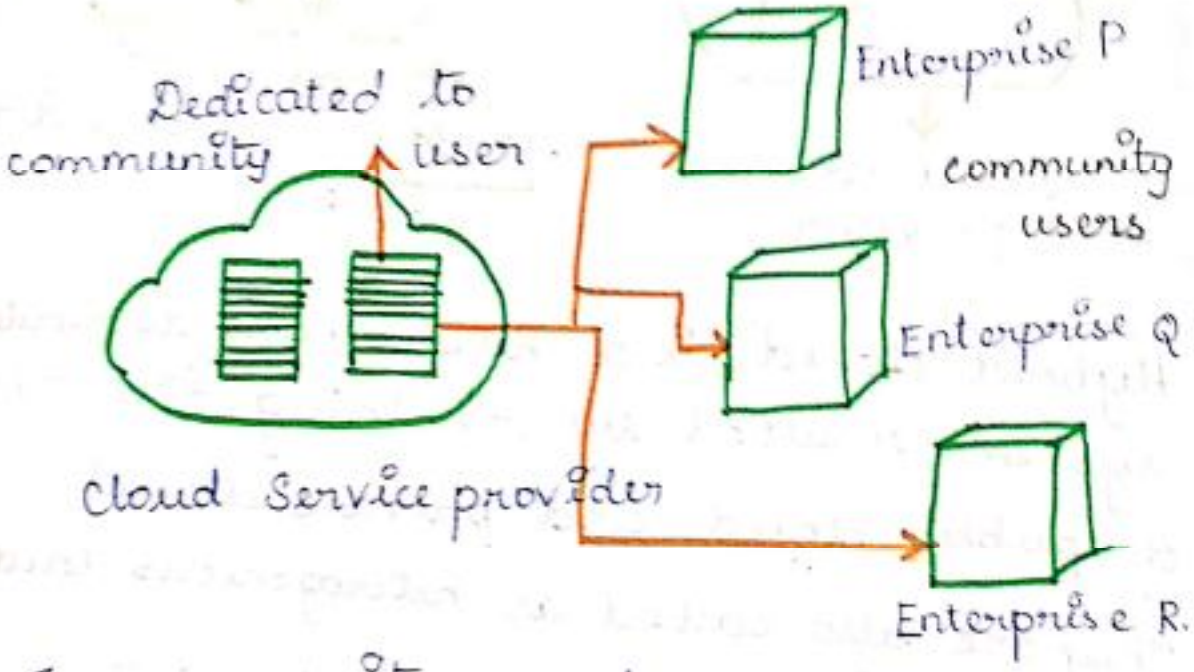


Hybrid cloud is a heterogenous distribute system resulted by combining facilities of public cloud and private cloud. They are also called as heterogeneous cloud. Hybrid cloud takes advantages of both public and private cloud.

A major drawback of private deployments is the inability to scale on demand and to efficiently address peak loads. Here public clouds are needed.

Community cloud

Community clouds are distributed systems created by integrating the services of different clouds to address the specific needs of an industry, a community on a business sector.



In community cloud, the infrastructure is shared between organisation which have shared concerns on tasks.
The cloud may be managed by an organisation or a third party.

Media Industry:

Media companies are looking for quick, simple, low-cost way for increasing efficiency of content generation

Most media productions involve an extended ecosystem of partners.

Healthcare Industry:

In healthcare industry, community clouds are used to share information and knowledge on the global level with sensitive data in private infrastructure.

Energy and Core Industry:

In this sector, community cloud is used to cluster a set of solution which collectively addresses

* Management
* Deployment
* Co-ordination of services and operations.

Scientific Research:

In this organisation with common interests of science share large distributed infrastructure for scientific computing.

UNIT – II

CLOUD SERVICES

SAAS(Software as a Service)

The traditional model of software distribution, in which software is purchased for and installed on personal computers, is sometimes referred to as Software-as-a-Product. Software-as-a-Service is a software distribution model in which applications are hosted by a vendor or service provider and made available to customers over a network, typically the Internet.

SaaS is becoming an increasingly prevalent delivery model as underlying technologies that support web services and service-oriented architecture (SOA) mature and new developmental approaches become popular. SaaS is also often associated with a pay-as-you-go subscription licensing model. Meanwhile, broadband service has become increasingly available to support user access from more areas around the world.

The huge strides made by Internet Service Providers (ISPs) to increase bandwidth, and the constant introduction of ever more powerful microprocessors coupled with inexpensive data storage devices, is providing a huge platform for designing, deploying, and using software across all areas of business and personal computing.

SaaS applications also must be able to interact with other data and other applications in an equally wide variety of environments and platforms. SaaS is closely related to other service delivery models we have described. IDC identifies two slightly different delivery models for SaaS. The hosted application management model is similar toan Application Service Provider (ASP) model. Here, an ASP hosts commercially available software for customers and delivers it over the Internet.

The other model is a software on demand model where the provider gives customers network-based access to a single copy of an application created specifically for SaaS distribution. IDC predicted that SaaS would make up 30% of the software market by 2007 and would be worth $10.7 billion by the end of 2009.

SaaS is most often implemented to provide business software functionality to enterprise customers at a low cost while allowing those customers to obtain the same benefits of commercially licensed, internally operated software without the associated complexity of installation, management, support, licensing, and high initial cost.

Most customers have little interest in the how or why of software implementation, deployment, etc., but all have a need to use software in their work. Many types of software are well suited to the SaaS model (e.g., accounting, customer relationship management, email software, human resources, IT security, IT service management, video conferencing, web analytics, web content management).

The distinction between SaaS and earlier applications delivered over the Internet is that SaaS solutions were developed specifically to work within a web browser. The architecture of SaaS-based applications is specifically designed to support many concurrent users (multitenancy) at once. This is a big difference from the traditional client/server or application service provider (ASP)based solutions that cater to a contained audience. SaaS providers, on the other hand, leverage enormous economies of scale in the deployment, management, support, and maintenance of their offerings.

SaaS Implementation Issues

Many types of software components and applications frameworks may be employed in the development of SaaS applications. Using new technology found in these modern components and application frameworks can drastically reduce the time to market and cost of converting a traditional on-premises product into a SaaS solution.

According to Microsoft, SaaS architectures can be classified into one of four maturity levels whose key attributes are ease of configuration, multitenant efficiency, and scalability. Each level is distinguished from the previous one by the addition of one of these three attributes. The levels described by Microsoft are as follows.

- **SaaS Architectural Maturity Level 1—Ad-Hoc/Custom**. The first level of maturity is actually no maturity at all. Each customer has a unique, customized version of the hosted application. The application runs its own instance on the host's servers. Migrating a traditional non-networked or client-server application to this level of SaaS maturity typically requires the least development effort and reduces operating costs by consolidating server hardware and administration.
- **SaaS Architectural Maturity Level 2—Configurability.**The second level of SaaS maturity provides greater program flexibility through configuration metadata. At this level, many customers can use separate instances of the same application. This allows a vendor to meet the varying needs of each customer by using detailed configuration options. It also allows the vendor to ease the maintenance burden by being able to update a common code base.
- **SaaS Architectural Maturity Level 3—Multitenant Efficiency**. The third maturity level adds multitenancy to the second level. This results in a single program instance that has the capability to serve all of the vendor's customers. This approach enables more efficient use of server resources without any apparent difference to the end user, but ultimately this level is limited in its ability to scale massively.
- **SaaS Architectural Maturity Level 4—Scalable.** At the fourth SaaS maturity level, scalability is added by using a multitiered architecture. This architecture is capable of supporting a load-balanced farm of

identical application instances running on a variable number of servers, sometimes in the hundreds or even thousands. System capacity can be dynamically increased or decreased to match load demand by adding or removing servers, with no need for further alteration of application software architecture

Characteristics of SaaS

Deploying applications in a service-oriented architecture is a more complex problem than is usually encountered in traditional models of software deployment. As a result, SaaS applications are generally priced based on the number of users that can have access to the service. There are often additional fees for the use of help desk services, extra bandwidth, and storage. SaaS revenue streams to the vendor are usually lower initially than traditional software license fees. However, the trade-off for lower license fees is a monthly recurring revenue stream, which is viewed by most corporate CFOs as a more predictable gauge of how the business is faring quarter to quarter. These monthly recurring charges are viewed much like maintenance fees for licensed software.

The key characteristics of SaaS software are the following:

- Network-based management and access to commercially available software from central locations rather than at each customer's site, enabling customers to access applications remotely via the Internet.
- Application delivery from a one-to-many model (single-instance, multitenant architecture), as opposed to a traditional one-to-one model.
- Centralized enhancement and patch updating that obviates any need for downloading and installing by a user. SaaS is often used in conjunction with a larger network of communications and collaboration software, sometimes as a plug-in to a PaaS architecture.

Benefits of the SaaS Model

Application deployment cycles inside companies can take years, consume massive resources, and yield unsatisfactory results. Although the initial decision to relinquish control is a difficult one, it is one that can lead to improved efficiency, lower risk, and a generous return on investmen

An increasing number of companies want to use the SaaS model for corporate applications such as customer relationship management and those that fall under the Sarbanes-Oxley Act compliance umbrella (e.g., financial recording and human resources).

The SaaS model helps enterprises ensure that all locations are using the correct application version and, therefore, that the format of the data being recorded and conveyed is consistent, compatible, and accurate.

By placing the responsibility for an application onto the doorstep of a SaaS provider, enterprises can reduce administration and management burdens they would otherwise have for their own corporate applications.
SaaS also helps to increase the availability of applications to global locations. SaaS also ensures that all application transactions are logged for compliance purposes. The benefits of SaaS to the customer are very clear:

- Streamlined administration
- Automated update and patch management services
- Data compatibility across the enterprise (all users have the same version of software)
- Facilitated, enterprise-wide collaboration
- Global accessibility

PAAS(Platform as a Service)

Cloud computing has evolved to include platforms for building and running custom web-based applications, a concept known as Platform-as-aService. PaaS is an outgrowth of the SaaS application delivery model.
The PaaS model makes all of the facilities required to support the complete life cycle of building and delivering web applications and services entirely available from the Internet, all with no software downloads or installation for developers, IT managers, or end users.
Unlike the IaaS model, where developers may create a specific operating system instance with homegrown applications running, PaaS developers are concerned only with webbased development and generally do not care what operating system is used.
PaaS services allow users to focus on innovation rather than complex infrastructure. Organizations can redirect a significant portion of their budgets to creating applications that provide real business value instead of worrying about all the infrastructure issues in a roll-your-own delivery model.
The PaaS model is thus driving a new era of mass innovation. Now, developers around the world can access unlimited computing power. Anyone with an Internet connection can build powerful applications and easily deploy them to users globally.

The Traditional On-Premises Model

The traditional approach of building and running on-premises applications has always been complex, expensive, and risky. Building your own solution has never offered any guarantee of success. Each application was designed to meet specific business requirements. Each solution required a specific set of hardware, an operating system, a database, often a middleware package, email and web servers, etc.
Once the hardware and software environment was created, a team of developers had to navigate complex programming development platforms to build their

applications. Additionally, a team of network, database, and system management experts was needed to keep everything up and running. Inevitably, a business requirement would force the developers to make a change to the application. The changed application then required new test cycles before being distributed.

Large companies often needed specialized facilities to house their data centers. Enormous amounts of electricity also were needed to power the servers as well as to keep the systems cool. Finally, all of this required use of fail-over sites to mirror the data center so that information could be replicated in case of a disaster. Old days, old ways—now, let's fly into the silver lining of todays cloud.

The New Cloud Model

PaaS offers a faster, more cost-effective model for application development and delivery. PaaS provides all the infrastructure needed to run applications over the Internet. Such is the case with companies such as Amazon.com, eBay, Google, iTunes, and YouTube. The new cloud model has made it possible to deliver such new capabilities to new markets via the web browsers.

PaaS is based on a metering or subscription model, so users pay only for what they use. PaaS offerings include workflow facilities for application design, application development, testing, deployment, and hosting, as well as application services such as virtual offices, team collaboration, database integration, security, scalability, storage, persistence, state management, dashboard instrumentation, etc.

Key Characteristics of PaaS

1. Chief characteristics of PaaS include services to develop, test, deploy, host, and manage applications to support the application development life cycle.
2. Web-based user interface creation tools typically provide some level of support to simplify the creation of user interfaces, based either on commonstandards such as HTML and JavaScript or on other, proprietary technologies.
3. Supporting a multitenant architecture helps to remove developer concerns regarding the use of the application by many concurrent users.
4. PaaS providers often include services for concurrency management, scalability, fail-over and security.
5. Another characteristic is the integration with web services and databases.
6. Support for Simple Object Access Protocol (SOAP) and other interfaces allows PaaS offerings to create combinations of web services (called mashups) as well as having the ability to access databases and reuse services maintained inside private networks.
7. The ability to form and share code with ad-hoc, predefined, or distributed teams greatly enhances the productivity of PaaS offerings. Integrated

PaaS offerings provide an opportunity for developers to have much greater insight into the inner workings of their applications and the behavior of their users by implementing dashboard-like tools to view the inner workings based on measurements such as performance, number of concurrent accesses, etc. Some PaaS offerings leverage this instrumentation to enable pay-per-use billing models.

Infrastructure-as-a-Service (IaaS)

According to the online reference Wikipedia, Infrastructure-as-a-Service (IaaS) is the delivery of computer infrastructure (typically a platform virtualization environment) as a service.IaaS leverages significant technology, services, and data center investments to deliver IT as a service to customers. Unlike traditional outsourcing, which requires extensive due diligence, negotiations ad infinitum, and complex, lengthy contract vehicles, IaaS is centered around a model of service delivery that provisions a predefined, standardized infrastructure specifically optimized for the customer's applications. Simplified statements of work and à la carte service-level choices make it easy to tailor a solution to a customer's specific application requirements. IaaS providers manage the transition and hosting of selected applications on their infrastructure. Customers maintain ownership andmanagement of their application(s) while off-loading hosting operations and infrastructure management to the IaaS provider. Provider-owned implementations typically include the following layered components:

1. Computer hardware (typically set up as a grid for massive horizontal scalability)
2. Computer network (including routers, firewalls, load balancing, etc.)
3. Internet connectivity (often on OC 192 backbones4)
4. Platform virtualization environment for running client-specified virtual machines
5. Service-level agreements
6. Utility computing billing

Rather than purchasing data center space, servers, software, network equipment, etc., IaaS customers essentially rent those resources as a fully outsourced service. Usually, the service is billed on a monthly basis, just like a utility company bills customers. The customer is charged only for resources consumed. The chief benefits of using this type of outsourced service include:

1. Ready access to a preconfigured environment that is generally ITIL-based (The Information Technology Infrastructure Library [ITIL] is a customized framework of best practices designed to promote quality computing services in the IT sector.)
2. Use of the latest technology for infrastructure equipment
3. Secured, "sand-boxed" (protected and insulated) computing platforms that are usually security monitored for breaches
4. Reduced risk by having off-site resources maintained by third parties

5. Ability to manage service-demand peaks and valleys
6. Lower costs that allow expensing service costs instead of making capital investments
7. Reduced time, cost, and complexity in adding new features or capabilities

Modern On-Demand Computing

On-demand computing is an increasingly popular enterprise model in which computing resources are made available to the user as needed. Computing resources that are maintained on a user's site are becoming fewer and fewer, while those made available by a service provider are on the rise.

The on-demand model evolved to overcome the challenge of being able to meet fluctuating resource demands efficiently. Because demand for computing resources can vary drastically from one time to another, maintaining sufficient resources to meet peak requirements can be costly.

Overengineering a solution can be just as adverse as a situation where the enterprise cuts costs by maintaining only minimal computing resources, resulting in insufficient resources to meet peak load requirements.

Concepts such as clustered computing, grid computing, utility computing, etc., may all seem very similar to the concept of on-demand computing, but they can be better understood if one thinks of them as building blocksthat evolved over time and with techno-evolution to achieve the modern cloud computing model we think of and use today

One example we will examine is Amazon's Elastic Compute Cloud (Amazon EC2). This is a web service that provides resizable computing capacity in the cloud. It is designed to make web-scale computing easier for developers and offers many advantages to customers:

1. It's web service interface allows customers to obtain and configure capacity with minimal effort.
2. It provides users with complete control of their (leased) computing resources and lets them run on a proven computing environment.
3. It reduces the time required to obtain and boot new server instances to minutes, allowing customers to quickly scale capacity as their computing demands dictate.
4. It changes the economics of computing by allowing clients to pay only for capacity they actually use.
5. It provides developers the tools needed to build failure-resilient applications and isolate themselves from common failure scenarios.

Amazon's Elastic Cloud

Amazon EC2 presents a true virtual computing environment, allowing clients to use a web-based interface to obtain and manage services needed to launch one or more instances of a variety of operating systems (OSs). Clients can load the OS environments with their customized applications. They can manage their network's access permissions and run as many or as few systems as needed.

In order to use Amazon EC2, clients first need to create an Amazon Machine Image (AMI). This image contains the applications, libraries, data, and associated configuration settings used in the virtual computing environment.

Amazon EC2 offers the use of preconfigured images built with templates to get up and running immediately. Once users have defined and configured

their AMI, they use the Amazon EC2 tools provided for storing the AMI by uploading the AMI into Amazon S3.

Amazon S3 is a repository that provides safe, reliable, and fast access to a client AMI. Before clients can use the AMI, they must use the Amazon EC2 web service to configure security and network access.

Using Amazon EC2 to Run Instances

During configuration, users choose which instance type(s) and operating system they want to use. Available instance types come in two distinct categories, Standard or High-CPU instances. Most applications are best suited for Standard instances, which come in small, large, and extra-large instance platforms. High-CPU instances have proportionally more CPU resources than random-access memory (RAM) and are well suited for compute-intensive applications. With the High-CPU instances, there are medium and extra large platforms to choose from.

After determining which instance to use, clients can start, terminate, and monitor as many instances of their AMI as needed by using web service Application Programming Interfaces (APIs) or a wide variety of other management tools that are provided with the service.

Users are able to choose whether they want to run in multiple locations, use static IP endpoints, or attach persistent block storage to any of their instances, and they pay only for resources actually consumed. They can also choose from a library of globally available AMIs that provide useful instances. For example, if all that is needed is a basic Linux server, clients can choose one of the standard Linux distribution AMIs.

Monitoring-as-a-Service (MaaS)

Monitoring-as-a-Service (MaaS) is the outsourced provisioning of security, primarily on business platforms that leverage the Internet to conduct business.MaaS has become increasingly popular over the last decade. Since the advent of cloud computing, its popularity has, grown even more. Security monitoring involves protecting an enterprise or government client from cyber threats.

A security team plays a crucial role in securing and maintaining the confidentiality, integrity, and availability of IT assets. However, time and resource constraints limit security operations and their effectiveness for most companies. This requires constant vigilance over the security infrastructure and critical information assets.

Many industry regulations require organizations to monitor their security environment, server logs, and other information assets to ensure the integrity of these systems. However, conducting effective security monitoring can be a daunting task because it requires advanced technology, skilled security experts, and scalable processes—none of which come cheap.

MaaS security monitoring services offer real-time, 24/7 monitoring and nearly immediate incident response across a security infrastructure—they help to protect critical information assets of their customers. Prior to the advent of electronic security systems, security monitoring and response were heavily

dependent on human resources and human capabilities, which also limited the accuracy and effectiveness of monitoring efforts.

Over the past two decades, the adoption of information technology into facility security systems, and their ability to be connected to security operations centers (SOCs) via corporate networks, has significantly changed that picture. This means two important things: (1) The total cost of ownership (TCO) for traditional SOCs is much higher than for a modern-technology SOC; and (2) achieving lower security operations costs and higher security effectiveness means that modern SOC architecture must use security and IT technology to address security risks.

Protection Against Internal and External Threats SOC-based security monitoring services can improve the effectiveness of a customer security infrastructure by actively analyzing logs and alerts from infrastructure devices around the clock and in real time.

Monitoring teams correlate information from various security devices to provide security analysts with the data they need to eliminate false positives and respond to true threats against the enterprise. Having consistent access to the skills needed to maintain the level of service an organization requires for enterprise-level monitoring is a huge issue.

The information security team can assess system performance on a periodically recurring basis and provide recommendations for improvements as needed. Typical services provided by many MaaS vendors are described below.

Early Detection

An early detection service detects and reports new security vulnerabilities shortly after they appear. Generally, the threats are correlated with thirdparty sources, and an alert or report is issued to customers. This report is usually sent by email to the person designated by the company. Security vulnerability reports, aside from containing a detailed description of the vulnerability and the platforms affected, also include information on the impact the exploitation of this vulnerability would have on the systems or applications previously selected by the company receiving the report. Most often, the report also indicates specific actions to be taken to minimize the effect of the vulnerability, if that is known.

Platform, Control, and Services Monitoring

Platform, control, and services monitoring is often implemented as a dashboard interface10 and makes it possible to know the operational status of the platform being monitored at any time. It is accessible from a web interface, making remote access possible. Each operational element that is monitored usually provides an operational status indicator, always taking into accountthe critical impact of each element. This service aids in determining which elements may be operating at or near capacity or beyond the limits of established parameters. By detecting and identifying such problems, preventive measures can be taken to prevent loss of service.

Intelligent Log Centralization and Analysis

Intelligent log centralization and analysis is a monitoring solution based mainly on the correlation and matching of log entries. Such analysis helps to establish a baseline of operational performance and provides an index of security threat. Alarms can be raised in the event an incident moves the established baseline

parameters beyond a stipulated threshold. These types of sophisticated tools are used by a team of security experts who are responsible for incident response once such a threshold has been crossed and the threat has generated an alarm or warning picked up by security analysts monitoring the systems.

Vulnerabilities Detection and Management

Vulnerabilities detection and management enables automated verification and management of the security level of information systems. The service periodically performs a series of automated tests for the purpose of identifying system weaknesses that may be exposed over the Internet, including the possibility of unauthorized access to administrative services, the existence of services that have not been updated, the detection of vulnerabilities such as phishing, etc. The service performs periodic follow-up of tasks performed by security professionals managing information systems security and provides reports that can be used to implement a plan for continuous improvement of the system's security level.

Continuous System Patching/Upgrade and Fortification

Security posture is enhanced with continuous system patching and upgrading of systems and application software. New patches, updates, and service packs for the equipment's operating system are necessary to maintain adequate security levels and support new versions of installed products. Keeping abreast of all the changes to all the software and hardware requires a committed effort to stay informed and to communicate gaps in security that can appear in installed systems and applications.

Intervention, Forensics, and Help Desk Services

Quick intervention when a threat is detected is crucial to mitigating the effects of a threat. This requires security engineers with ample knowledge in the various technologies and with the ability to support applications as well as infrastructures on a 24/7 basis. MaaS platforms routinely provide this service to their customers. When a detected threat is analyzed, it often requires forensic analysis to determine what it is, how much effort it will take to fix the problem, and what effects are likely to be seen. When problems are encountered, the first thing customers tend to do is pick up the phone. Help desk services provide assistance on questions or issues about the operation of running systems. This service includes assistance in writing failure reports, managing operating problems, etc.

Delivering Business Value

Some consider balancing the overall economic impact of any build-versusbuy decision as a more significant measure than simply calculating a return on investment (ROI). The key cost categories that are most often associated with MaaS are (1) service fees for security event monitoring for all firewalls and intrusion detection devices, servers, and routers; (2) internal account maintenance and administration costs; and (3) preplanning and development costs.

Based on the total cost of ownership, whenever a customer evaluates the option of an in-house security information monitoring team and infrastructure compared to outsourcing to a service provider, it does not take long to realize that establishing and maintaining an in-house capability is not as attractive as outsourcing the service to a provider with an existing infrastructure. Having an

in-house security operations center forces a company to deal with issues such as staff attrition, scheduling, around the clock operations, etc.
Losses incurred from external and internal incidents are extremely significant, as evidenced by a regular stream of high-profile cases in the news. The generally accepted method of valuing the risk of losses from external and internal incidents is to look at the amount of a potential loss, assume a frequency of loss, and estimate a probability for incurring the loss. Although this method is not perfect, it provides a means for tracking information security metrics. Risk is used as a filter to capture uncertainty about varying cost and benefit estimates.
If a risk-adjusted ROI demonstrates a compelling business case, it raises confidence that the investment is likely to succeedbecause the risks that threaten the project have been considered and quantified. Flexibility represents an investment in additional capacity or agility today that can be turned into future business benefits at some additional cost. This provides an organization with the ability to engage in future initiatives, but not the obligation to do so. The value of flexibility is unique to each organization, and willingness to measure its value varies from company to company.
Real-Time Log Monitoring Enables Compliance
Security monitoring services can also help customers comply with industry regulations by automating the collection and reporting of specific events of interest, such as log-in failures. Regulations and industry guidelines often require log monitoring of critical servers to ensure the integrity of confidential data. MaaS providers' security monitoring services automate this timeconsuming process.

Communication-as-a-Service (CaaS)

CaaS is an outsourced enterprise communications solution. Providers of this type of cloud-based solution (known as CaaS vendors) are responsible for the management of hardware and software required for delivering Voice over IP (VoIP) services, Instant Messaging (IM), and video conferencing capabilities to their customers. This model began its evolutionary process from within the telecommunications (Telco) industry, not unlike how the SaaS model arose from the software delivery services sector. CaaS vendors are responsible for all of the hardware and software management consumed by their user base. CaaS vendors typically offer guaranteed quality of service (QoS) under a service-level agreement (SLA).

A CaaS model allows a CaaS provider's business customers to selectively deploy communications features and services throughout their company on a pay-as-you-go basis for service(s) used. CaaS is designed on a utility-like pricing model that provides users with comprehensive, flexible, and (usually) simple-to-understand service plans. According to Gartner,1 the CaaS market is expected to total $2.3 billion in 2011, representing a compound annual growth rate of more than 105% for the period.

CaaS service offerings are often bundled and may include integrated access to traditional voice (or VoIP) and data, advanced unified communications functionality such as video calling, web collaboration, chat, realtime presence and unified messaging, a handset, local and long-distance voice services, voice mail, advanced calling features (such as caller ID, threeway and conference calling, etc.) and advanced PBX functionality. A CaaS solution includes redundant switching, network, POP and circuit diversity, customer premises equipment redundancy, and WAN fail-over that specifically addresses the needs of their customers. All VoIP transport components are located in geographically diverse, secure data centers for high availability and survivability.

CaaS offers flexibility and scalability that small and medium-sized business might not otherwise be able to afford. CaaS service providers are usually prepared to handle peak loads for their customers by providing servicescapable of allowing more capacity, devices, modes or area coverage as their customer demand necessitates. Network capacity and feature sets can be changed dynamically, so functionality keeps pace with consumer demand and provider-owned resources are not wasted. From the service provider customer's perspective, there is very little to virtually no risk of the service becoming obsolete, since the provider's responsibility is to perform periodic upgrades or replacements of hardware and software to keep the platform technologically current.

CaaS requires little to no management oversight from customers. It eliminates the business customer's need for any capital investment in infrastructure, and it eliminates expense for ongoing maintenance and operations overhead for infrastructure. With a CaaS solution, customers are able to leverage enterprise-class communication services without having to build a premises-based solution of their own. This allows those customers to reallocate budget and personnel resources to where their business can best use them.

Advantages of CaaS

From the handset found on each employee's desk to the PC-based software client on employee laptops, to the VoIP private backbone, and all modes in between, every component in a CaaS solution is managed 24/7 by the CaaS vendor. As we said previously, the expense of managing a carrier-grade data center is shared across the vendor's customer base, making it more economical for businesses to implement CaaS than to build their own VoIP network. Let's look as some of the advantages of a hosted approach for CaaS.

Hosted and Managed Solutions

Remote management of infrastructure services provided by third parties once seemed an unacceptable situation to most companies. However, over the past decade, with enhanced technology, networking, and software, the attitude has changed. This is, in part, due to cost savings achieved in using those services. However, unlike the "one-off" services offered by specialist providers, CaaS

delivers a complete communications solution that is entirely managed by a single vendor. Along with features such as VoIP and unified communications, the integration of core PBX features with advanced functionality is managed by one vendor, who is responsible for all of the integration and delivery of services to users.

Fully Integrated, Enterprise-Class Unified Communications

With CaaS, the vendor provides voice and data access and manages LAN/WAN, security, routers, email, voice mail, and data storage. By managing the LAN/WAN, the vendor can guarantee consistent quality of service from a user's desktop across the network and back. Advanced unified communications features that are most often a part of a standard CaaS deployment include

- Chat
- Multimedia conferencing
- Microsoft Outlook integration
- Real-time presence
- "Soft" phones (software-based telephones)
- Video calling
- Unified messaging and mobility

Providers are constantly offering new enhancements (in both performance and features) to their CaaS services. The development process and subsequent introduction of new features in applications is much faster, easier, and more economical than ever before. This is, in large part, because the service provider is doing work that benefits many end users across the provider's scalable platform infrastructure. Because many end users of the provider's service ultimately share this cost (which, from their perspective, is miniscule compared to shouldering the burden alone), services can be offered to individual customers at a cost that is attractive to them.

No Capital Expenses Needed

When business outsource their unified communications needs to a CaaS service provider, the provider supplies a complete solution that fits the company's exact needs. Customers pay a fee (usually billed monthly) for what they use. Customers are not required to purchase equipment, so there is no capital outlay. Bundled in these types of services are ongoing maintenance and upgrade costs, which are incurred by the service provider. The use of CaaS services allows companies the ability to collaborate across any workspace. Advanced collaboration tools are now used to create high-quality, secure, adaptive work spaces throughout any organization. This allows a company's workers, partners, vendors, and customers to communicate and collaborate more effectively. Better communication allows organizations to adapt quickly to market changes and to build competitive advantage. CaaS can also accelerate decision making within an organization. Innovative unified communications capabilities (such as presence, instant messaging, and rich media services) help ensure that information quickly reaches whoever needs it.

Flexible Capacity and Feature Set

When customers outsource communications services to a CaaS provider, they pay for the features they need when they need them. The service provider can distribute the cost services and delivery across a large customer base. As previously stated, this makes the use of shared feature functionality more economical for customers to implement. Economies of scale allow service providers enough flexibility that they are not tied to a single vendor investment. They are able to leverage best-of-breed providers such as Avaya, Cisco, Juniper, Microsoft, Nortel and ShoreTel more economically than any independent enterprise.

No Risk of Obsolescence

Rapid technology advances, predicted long ago and known as Moore's law,2 have brought about product obsolescence in increasingly shorter periods of time. Moore's law describes a trend he recognized that has held true since the beginning of the use of integrated circuits (ICs) in computing hardware. Since the invention of the integrated circuit in 1958, the number of transistors that can be placed inexpensively on an integrated circuit has increased exponentially, doubling approximately every two years.

Unlike IC components, the average life cycles for PBXs and key communications equipment and systems range anywhere from five to 10 years. With the constant introduction of newer models for all sorts of technology (PCs, cell phones, video software and hardware, etc.), these types of products now face much shorter life cycles, sometimes as short as a single year. CaaS vendors must absorb this burden for the user by continuously upgrading the equipment in their offerings to meet changing demands in the marketplace.

No Facilities and Engineering Costs Incurred

- CaaS providers host all of the equipment needed to provide their services to their customers, virtually eliminating the need for customers to maintain data center space and facilities. There is no extra expense for the constant power consumption that such a facility would demand. Customers receive the benefit of multiple carrier-grade data centers with full redundancy—and it's all included in the monthly payment.

Guaranteed Business Continuity

- If a catastrophic event occurred at your business's physical location, would your company disaster recovery plan allow your business to continue operating without a break? If your business experienced a serious or extended communications outage, how long could your company survive? For most businesses, the answer is "not long." Distributing risk by using geographically dispersed data centers has become the norm today. It mitigates risk and allows companies in a location hit by a catastrophic event to recover as soon as possible.
- This process is implemented by CaaS providers because most companies don't even contemplate voice continuity if catastrophe strikes. Unlike data continuity, eliminating single points of failure for a voice network is

usually cost-prohibitive because of the large scale and management complexity of the project.

- With a CaaS solution, multiple levels of redundancy are built into the system, with no single point of failure.

CaaS is an outsourced enterprise communications solution. Providers of this type of cloud-based solution (known as CaaS vendors) are responsible for the management of hardware and software required for delivering Voice over IP (VoIP) services, Instant Messaging (IM), and video conferencing capabilities to their customers. This model began its evolutionary process from within the telecommunications (Telco) industry, not unlike how the SaaS model arose from the software delivery services sector. CaaS vendors are responsible for all of the hardware and software management consumed by their user base. CaaS vendors typically offer guaranteed quality of service (QoS) under a service-level agreement (SLA).

A CaaS model allows a CaaS provider's business customers to selectively deploy communications features and services throughout their company on a pay-as-you-go basis for service(s) used. CaaS is designed on a utility-like pricing model that provides users with comprehensive, flexible, and (usually) simple-to-understand service plans. According to Gartner,[1] the CaaS market is expected to total $2.3 billion in 2011, representing a compound annual growth rate of more than 105% for the period.

CaaS service offerings are often bundled and may include integrated access to traditional voice (or VoIP) and data, advanced unified communications functionality such as video calling, web collaboration, chat, real-time presence and unified messaging, a handset, local and long-distance voice services, voice mail, advanced calling features (such as caller ID, three-way and conference calling, etc.) and advanced PBX functionality. A CaaS solution includes redundant switching, network, POP and circuit diversity, customer premises equipment redundancy, and WAN fail-over that specifically addresses the needs of their customers. All VoIP transport components are located in geographically diverse, secure data centers for high availability and survivability.

CaaS offers flexibility and scalability that small and medium-sized business might not otherwise be able to afford. CaaS service providers are usually prepared to handle peak loads for their customers by providing services capable of allowing more capacity, devices, modes or area coverage as their customer demand necessitates. Network capacity and feature sets can be changed dynamically, so functionality keeps pace with consumer demand and provider-owned resources are not wasted. From the service provider

customer's perspective, there is very little to virtually no risk of the service becoming obsolete, since the provider's responsibility is to perform periodic upgrades or replacements of hardware and software to keep the platform technologically current.

CaaS requires little to no management oversight from customers. It eliminates the business customer's need for any capital investment in infrastructure, and it eliminates expense for ongoing maintenance and operations overhead for infrastructure. With a CaaS solution, customers are able to leverage enterprise-class communication services without having to build a premises-based solution of their own. This allows those customers to reallocate budget and personnel resources to where their business can best use them.

ADVANTAGES OF CAAS

Flexible Capacity and Feature Set

When customers outsource communications services to a CaaS provider, they pay for the features they need when they need them. The service provider can distribute the cost services and delivery across a large customer base. As previously stated, this makes the use of shared feature functionality more economical for customers to implement. Economies of scale allow service providers enough flexibility that they are not tied to a single vendor investment. They are able to leverage best-of-breed providers such as Avaya, Cisco, Juniper, Microsoft, Nortel and ShoreTel more economically than any independent enterprise.

Hosted and Managed Solutions

Remote management of infrastructure services provided by third parties once seemed an unacceptable situation to most companies. However, over the past decade, with enhanced technology, networking, and software, the attitude has changed. This is, in part, due to cost savings achieved in using those services. However, unlike the "one-off" services offered by specialist providers, CaaS delivers a complete communications solution that is entirely managed by a single vendor. Along with features such as VoIP and unified communications, the integration of core PBX features with advanced functionality is managed by one vendor, who is responsible for all of the integration and delivery of services to users.

No Risk of Obsolescence

Rapid technology advances, predicted long ago and known as Moore's law,[2] have brought about product obsolescence in increasingly shorter periods of time. Moore's law describes a trend he recognized that has held true since the beginning of the use of integrated circuits (ICs) in computing hardware. Since the invention of the integrated circuit in 1958, the number of transistors that can be placed inexpensively on an integrated circuit has increased exponentially, doubling approximately every two years.

Unlike IC components, the average life cycles for PBXs and key communications equipment and systems range anywhere from five to 10 years. With the constant introduction of newer models for all sorts of technology (PCs, cell phones, video software and hardware, etc.), these types of products now face much shorter life cycles, sometimes as short as a single year. CaaS vendors must absorb this burden for the user by continuously upgrading the equipment in their offerings to meet changing demands in the marketplace.

Guaranteed Business Continuity

If a catastrophic event occurred at your business's physical location, would your company disaster recovery plan allow your business to continue operating without a break? If your business experienced a serious or extended communications outage, how long could your company survive? For most businesses, the answer is "not long." Distributing risk by using geographically dispersed data centers has become the norm today. It mitigates risk and allows companies in a location hit by a catastrophic event to recover as soon as possible. This process is implemented by CaaS providers because most companies don't even contemplate voice continuity if catastrophe strikes. Unlike data continuity, eliminating single points of failure for a voice network is usually cost-prohibitive because of the large scale and management complexity of the project. With a CaaS solution, multiple levels of redundancy are built into the system, with no single point of failure.

No Facilities and Engineering Costs Incurred

CaaS providers host all of the equipment needed to provide their services to their customers, virtually eliminating the need for customers to maintain data center space and facilities. There is no extra expense for the constant power consumption that such a facility would demand. Customers receive the benefit of multiple carrier-grade data centers with full redundancy—and it's all included in the monthly payment.

Types of Cloud models :

The basic functions of the cloud models can be summarized in the phrases Host, Build and Consume.

NBC

Each model offers different level of flexibility and control over the product that your business is buying

There are four types of cloud models. They are :

① SaaS (Software as a Service)

adv

Saas model allows your business to quickly access cloud based web applications without committing to new infrastructure installation.

The applications run on vendor's cloud, which they control and maintain.

The applications are available for use with a paid licensed subscription

or for free with limited access.

Saas does not require any installations or downloads in your existing infrastructure, which eliminat the need to install, maintain and updat applications on each of your computers

Advantages of Saas :

① AFFORDABLE :

On-premise hardware is not required for this model, which keeps the cost associated low.

Small scale businesses might find this cloud platform appealing.

② ACCESSIBLE EVERYWHERE :

Cloud-based applications are ~~everyw~~ accessible in all the places where there is internet access.

companies that require frequent collabration find Saas platforms useful as their employees can easily access the programs that they need.

③ READY TO USE :

With Saas the programs you need are already full developed and ready to use

The Set up time for Saas programs is greatly decreased

Disadvantages of Saas :

① LACK OF CONTROL

With Saas, vendor has control over the programs that your company is using.

If you don't feel comfortable releasing control of your critical busines applications to another party, perhaps Saas is not the best option for your business.

② SLOWER SPEEDS :

Relying upon internet access to function, Saas applications tend to be slower than client/server applications.

These programs are typically quick but instantaneous.

③ VARIABLE FUNCTIONS AND FEATURES :

Saas cloud based applications have less functionality and features

than client/server counterparts in many cases

This disadvantage can be void when your business only needs features offered in the Saas version to function

② PaaS (Platform as a Service)

With this ~~platform~~ model, a third party vendor provides your business with a platform upon which your business can develop and run applications.

Because the vendor is hosting cloud infrastructure which supports the platform, PaaS eliminates your need to install in-house hardware or software.

Your business would not manage or control the underlying cloud infrastructure but you would maintain control over deployed applications (unlike SaaS).

Advantages :

① Rapid Time to Market :

PaaS simplifies application management by eliminating the need to maintain and control the underlying

infrastructure.

As a result, applications can be developed and deployed faster.

② Cost-effective deployment:

③ Scalability.

Disadvantages:

① Vendor Lock-in

② Security and compliance (Security reason)

③ Lack of compatability.

IaaS (Infrastructure as a Service)

IaaS, is the most flexible of the cloud models. allows your business to have complete scalable control over the management and customization of your infrastructure.

In the IaaS model, the cloud provider hosts your infrastructure components that would traditionally be present in an on-site data center

(such as servers, storage and networking hardware).

Your business, however, would maintain control over operating systems, storage, deployed applications and possibly limited control of selected networking components.

Advantages of IaaS:

1. Eliminates Capital Expenses
2. Supports Flexibility.
3. Simple Deployment.

Disadvantages of IaaS:

1. Insight.
2. Variability of Resilience → adapt it to the consequences of power failure.
3. Costly.

AZW.

Google cloud platform.

Azure.

Google Cloud Platform (GCP)

Google is one among the leading cloud providers that offer secure storage of users data.

Cloud Services:

* cloud platform
* app engine
* cloud print
* cloud connect
* cloud storage.

Features:

⇒ scalable ⇒ reliable ⇒ secure.
⇒ support ⇒ flexibility

* Google uses the cloud services internally for its end-user products like Google Search and Youtube.
* It also provides cloud services for computing, data analytics and machine learning.
* It can be written in Java, C++, Python, Go, Ruby.
* GCP provides
  Infrastructure as a Service
  Platform as a Service

② Serverless computing environments

⇒ As per 2019, Google Cloud Platform is available in 20 regions and 61 zones.

TIMELINE

April 2008	Google App Engine announced in preview
May 2010	⇒ Google Cloud Storage launched. ⇒ Google BigQuery and Prediction API announced in preview.
October 2011	⇒ Google Cloud SQL is announced in preview.
June 2012	⇒ Google Compute Engine is launched in preview.
May 2013	⇒ Google Compute Engine is released to general audience.
August 2013	⇒ Cloud storage begins automatically encrypting each storage objects data. Each encryption key is itself encrypted with a regularly rotated set of master keys
February 2014	⇒ Google Cloud SQL becomes Generally available.

May 2014	Stackdriver is acquired by Google.
June 204.	⇒ Kubernetes is announced as an open source container manager. ⇒ Cloud Dataflow is announced in preview.
October 2014	⇒ Google acquires Firebase.
March 2015	⇒ Google Cloud Pub/Sub becomes available in Beta.
April 2015	⇒ Google cloud DNS becomes generally available.
July 2015	⇒ Google releases V1 of Kubernetes
August 2015	⇒ Google Cloud Dataflow, Google Cloud Pub/Sub, Google Kubernetes Engine becomes available in to general.
Februarry 2015	⇒ Google Cloud Functions becomes available in Alpha
September 2016	⇒ Stackdriver becomes generally available ⇒ Apigee, a provider of API is acquired by google.
May 2017	⇒ Google cloud IoT Core is launched in Beta.
Feb 2018	⇒ Google Cloud IoT Core becomes generally available.
May 2018	⇒ Google Cloud Memorystore becomes available in Beta.

(b) April 2019	Google Cloud Run Beta Release
November 2019	Google Cloud Run General availability release.

In short.

April 2008 * Google App Engine
* made available to 10,000 developers on first come first served basis
* only a preview mode.

Nov 2011 * Google pulled Google App Engine out of preview mode and dubbed into official mode.

From 2008 - 2018 * Added many services (around 90) under Google Platform umbrella.

Cloud Storage:

Google Cloud storage is RESTful online file storage web service for storing and accessing one's data on Googles infrastructure.

Representational state transfer (REST) is an architectural style consisting of a coordinated set of architectural set of constraints applied to components, connectors and data elements within a distributed system.

ADVANTAGES:

* advanced security
* advanced sharing capabilities
* safe and secure
* scalability and performance
* Data are protected through redundant storage at multiple physical locations.

Few tools for Google Cloud Storage:

GOOGLE DEVELOPERS CONSOLE ⇒ web application where one can perform simple storage management tasks on Google cloud storage system.

gsutil ⇒ a Python application that lets the user access Google Cloud Storage from command line.

ex: Cloud SQL, Cloud BigTable, Cloud Spanner, Cloud Datastore, Cloud Memorystore

Google Cloud Connect:

Google cloud Connect is a feature provided by Google cloud for integrating cloud and API for Microsoft office.

After installing plug-in for MS office suite of programs, one can save files to cloud ⇒ master copy.

* assigns a unique URL for sharing.
* Any changes made in the document will show up for everyone else viewing it.
* When multiple changes are made, user can choose which set of changes to be kept
* Metadata (information about other information) is inserted when the file is uploaded.

* As the documents sync to the master file, Google cloud connect sends the updated data out to all downloaded copies using metadata.

Backend ⇒ similar to Google File System

⇓ relies on Google Docs Infrastructure

Google Cloud Print:

Google Cloud Print is a service that extends the printer's function to any device that can connect to the Internet.

Requirements: (for user) * a free Google profile * an app, a program or a website that incorporates Google cloud print feature.

* cloud ready printer or printer connected to a computer logged on the Internet.

Printers which are not cloud ready

* computer acts as liaison.
* Google cloud Print is an extension built into a Google chrome Browser along with a piece of code called connector.

Cloud ready Printer

* no need for dedicated computer.
* has to be registered with Google cloud print to take advantage of its capabilities.

DISADVANTAGE:

Not every app or site will have Google print built into it, which limits its functionality.

Google builds service into its own product but many people rely on service from multiple sources and find that it does not have wide enough adoption.

Google Cloud Platform:

⇒ enables developers to build, test and deploy applications on Google's highly scalable and reliable infrastructure.

Google cloud Platform includes

* virtual machines
* block storage
* NoSQL data storage
* Big data analytics

⇒ provides a range of storage services.
* easy maintanence
* quick user data access.

⇒ offers fully managed platform as well as flexible virtual machines.

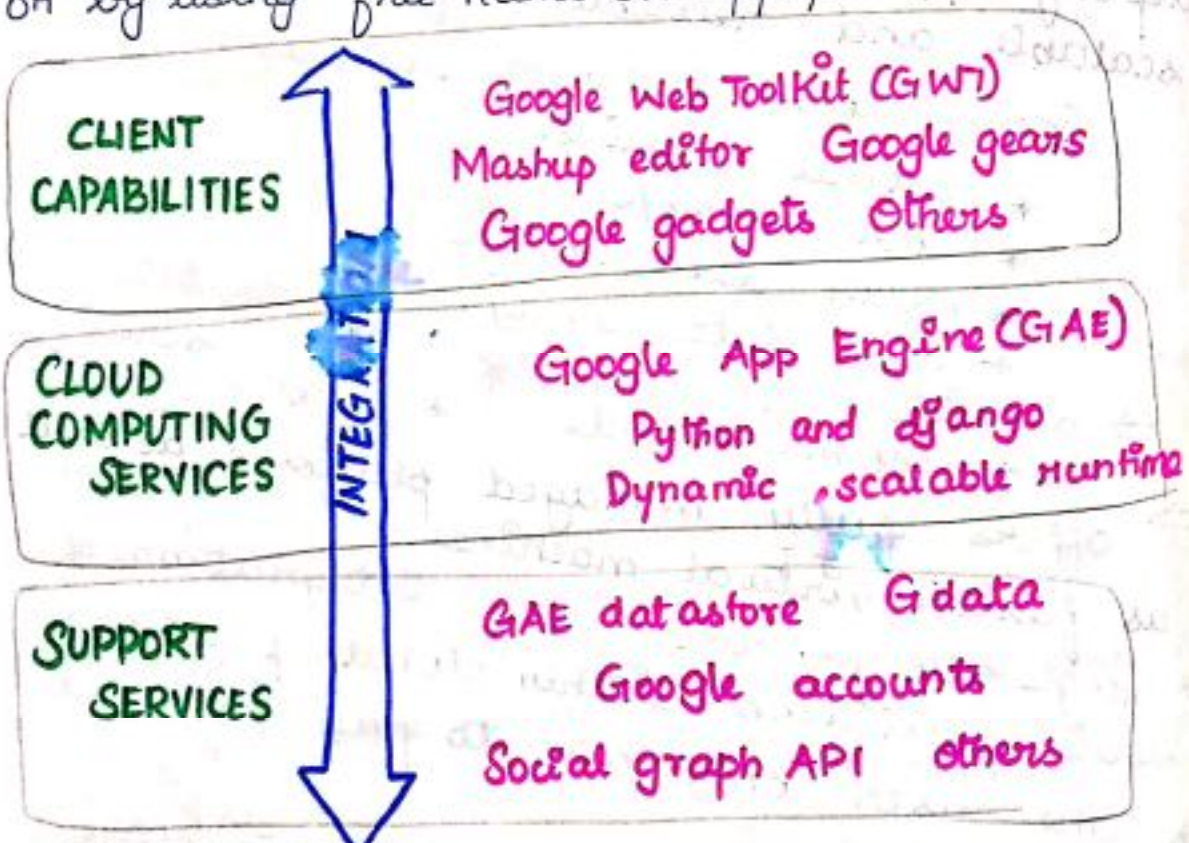
* Google also provides easy integration of users application within cloud platform.
* The user only has to pay for what he or she uses.
* Cloud platform is designed to scale like Google's own products even when there is a huge traffic spike.
* Applications hosted on platform can automatically scale up to handle the most demanding workloads.

Google App Engine:

(10) Google App Engine lets the user run web application on Google's infrastructure.

Applications:
- easy to build
- easy to maintain
- easy to scale as traffic ~~grow~~ needs and storage needs grow.

App can be served from user's domain name (http://www.example.com/) using Google Apps or by using free name on appspot.com



Languages Compatible: JAVA Python.

No setup cost and No recurring cost.

Microsoft

* Cloud Computing provides a new way of looking at IT at Microsoft called Microsoft IT (MSIT).

*Cloud computing is preferred and default environment for new and migrated applications at Microsoft.

MSIT has captured best practices and document -ed them for other Microsoft customers who wish to migrate their organisations to cloud computing.

It supports many different programming languages.

It provides * Infrastructure as a Service,
* Platform as a Service and
* Software as a Service

operating system Linux, Microsoft Windows.

Microsoft lists over 600 Azure Services.

Tools available:

Migration Assessment Tool (MAT

Microsoft Assessment and Planning Toolkit (MAP)

Sharepoint.

Types of Services

Timeline :

October 2008	Windows Azure Platform was announced
March 2009	SQL Azure Relational Database (announced)
Febraury 2010	Windows Azure Platform commercially available.
June 2010	Windaos Azure update, SQL Azure update.
October 2010	Windows Azure Connect, IT Pro Experience.
June 2012	Virtual Machines for Windows, Linux
April 2014	Windows Azure renamed to Microsoft Azure.
July 2014	Azure Machine Learning public review
December 2015	Azure ARM Portal released.

March 2016	Azure Service Fabric (Generally available)
July 2016	Azure Service Fabric Mesh (Public Preview)
September 2018	Microsoft Azure IoT Central (Generally Available.
April 2019	Azure Front Door Service (now available.

Azure is generally available in 54 regions.

Windows Azure:

Windows Azure Cloud Services allow developers to easily deploy and manage application services.

It allows management of role instances and operating systems to the Windows Azure platform.

Windows Azure Pack for Windows Server is a collection of Windows Azure technologies available to Microsoft customers at no additional cost for Installation.

TOOLS:

Migration Assessment Tool:

It encapsulates all the information to be aware of before attempting the application migration to Windows Azure.

Response for a simple binary questions, tool generates a report that outlines the amount of development effort to involved to migrate the application.

Windows Azure Pricing Calculator:

It analyses an application's potential public cloud requirements against the cost of the application's existing Infrastructure.

This tool can help to current compare current operational costs with operating costs would be on Windows Azure and SQL Azure.

Microsoft Assessment and Planning Toolkit

MAP is an agentless, automated, multi product planning and assessment tool for cloud migration.

MAP provides detailed readiness assessment reports, executive proposals, and hardware and software information.

It also provides recommendations to help organisations accelerate the application migration process for both private and public cloud planning assessments.

Sharepoint:

Microsoft's own online collaboration tool ⇒ Sharepoint

MS Sharepoint is a web application platform that comprises a multipurpose set of web technologies.

Sharepoint has a MS Office like Interface and it is closely integrated with the Office suite

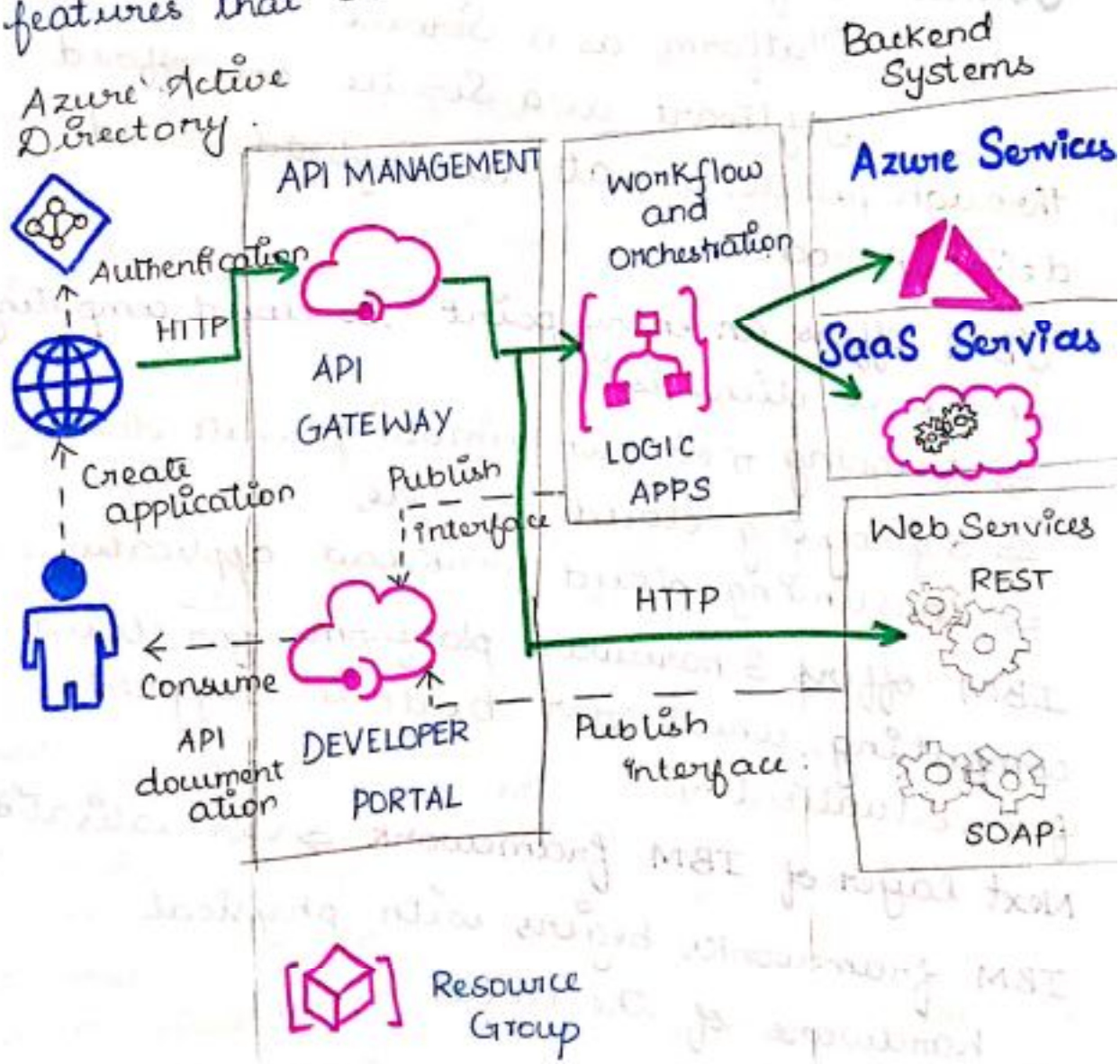
Web tools are designed to be usable by non technical users.

Sharepoint can be used to provide

* Intrand portals
* collaboration
* websites
* business Intelligence
* social networks
* extranet
* enterprise search
* document and file management.

Windows Azure Technologies allows you to offer a rich, self service, multitenant cloud, consistent with public experience.

Unlike Google Cloud Connect, Mirasoft Sharepoint is not a free tool. But it has additional features that cannot be matched by Google.



IBM

International Business Machines

It is one among the players in the field of cloud computing offering various cloud services to the consumers.

All offerings are designed for business use (by global IT company IBM), marketed under the name IBM Smart cloud.

Services: Infrastructure as a Service.

Platform as a Service

Software as a Service are offered through public, private and hybrid cloud delivery models.

IBM offers an entry point to cloud computing whether a client is

⇒ designing their own virtual private cloud.

⇒ deploying cloud service.

⇒ consuming cloud workload applications.

IBM offers 3 hardware platforms for cloud computing, which offer built in support for virtualisation.

Next layer of IBM framework ⇒ Virtualisation

IBM frameworks begins with physical hardware of the cloud.

Management layer of IBM cloud framework includes IBM Tivoli middleware. (17)
Management Tools provide capabilities to
regulate images with automated provisioning and deprovisioning
monitor app operations
meter usage while tracking costs.
allocating billing.
Last layer provides workload tools.
Workloads are servics of codes that can be executed to meet spefic business needs.

Cloud Models:

* offers a spectrum of cloud delivery options ranging from solely private cloud to solely public cloud and variations in between them.
* gives option to build a customised cloud solution out of a combination of public and private cloud.

(i) All data and process behind their own firewall can choose private solution.
(ii) Pay-as-you-go allows to run lower profile applications on a secure public cloud model.
(iii) Hybrid clouds some process are managed by IBM, other procss are kept on private cloud or VPN or Virtual Local Area Network.

Cloud Computing is the best choice for mobile Software. (18)

IBM offers 5 different cloud provision models:

1) Private cloud, owned and operated by customer.
2) Private cloud owned by customer and operated by IBM (or another provider).
3) Private cloud owned ~~by~~ and operator by IBM (or another provider.
4) Virtual private cloud services based on multitenant support for individual enterprises.
5) Public Cloud Servias. based on provision of functions to individuals.

Majority of cloud users choose a hybrid cloud model, with
some workloads served by internal systems
some from commercial cloud providers
some from public cloud service providers.

IBM specialises in secure private cloud offerings. For building strictly private clouds, IBM offers * IBM Workload Deployer and

* Cloud burst as

ready to deploy, cloud in a box-style solutions.

For customers who prefer to perform their own integration of private clouds, IBM offers a choice of hardware and software building blocks, alongwith recommendations, leading way to deployment.

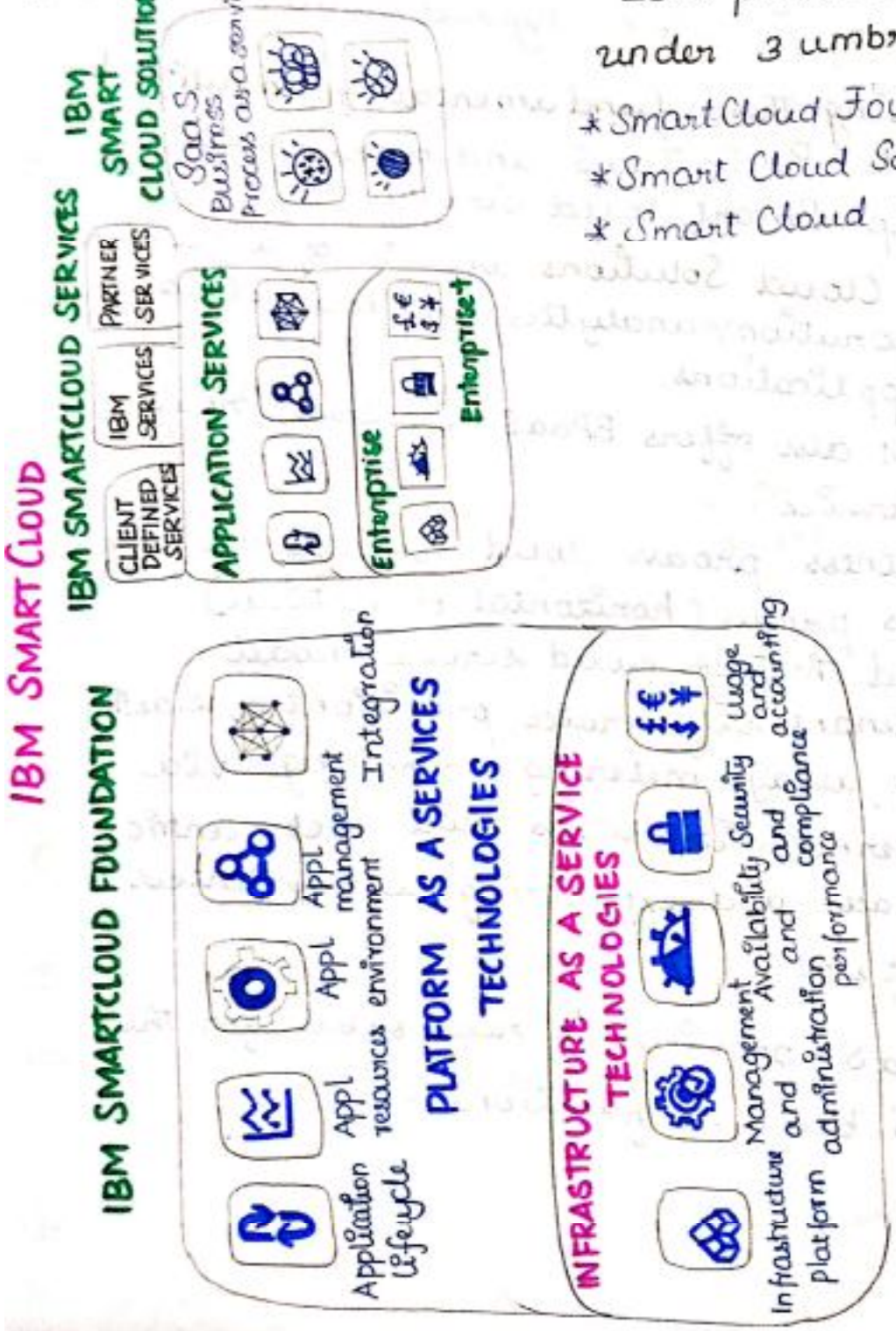
IBM Smart Cloud:

IBM SmartCloud is a branded ecosystem of cloud computing products and solutions from IBM.

It includes IaaS, PaaS, SaaS offered through public, private and hybrid cloud delivery models.

IBM places 3 offerings under 3 umbrellas:

* Smart Cloud Foundation
* Smart Cloud Solutions
* Smart Cloud Services.



Smart Cloud Foundation consists of

Infrastructure
hardware
provisioning
management
Integration
security. that serves as the under pinnings of private or hybrid cloud.

Build using these fundamental foundational components, PaaS, IaaS and backup services make up Smart Cloud Services.

Smart Cloud Solutions consists of a number of collaboration, analytics and marketing SaaS applications.

* IBM also offers BPaaS - Business Proass as a Service.

Business proass cloud services are any business proass (horizontal or vertical) delivered through cloud service model (multitenant, self-service provisioning, elastic scaling, usage metering or pricing) via the Internet with access via web-centric interfaces and exploiting web-oriented architecture.

BPaaS provider is responsible for the related business functions.

Timeline :

January June 2009	⇒ IBM Smart Business Services ⇒ IBM Cloudburst
October 2009	⇒ IBM Smart Business Storage cloud.
November 2009	⇒ IBM Smart Analytics Cloud
July 2010	⇒ IBM Smart Business Desktop Cloud.
October 2010	⇒ IBM Cloud Burst V2.1
November 2010	⇒ IBM federal community cloud for government
April 2011	Launch of IBM Smartcloud and its Enterprise.
June 2011	⇒ IBM Smart Cloud Managed Backup and IBM Smart Cloud Virtualised Server Recovery.
October 2011	⇒ IBM SmartCloud Application Services. ⇒ IBM Smart Cloud Foundation
October 2014	⇒ IBM Cloud Manager with Openstack.
July 2016	⇒ IBM Cloud Power VC Manager.
March 2018	⇒ cloudant migrated to IBM Cloud.

SALESFORCE

Salesforce.com is a cloud computing and social enterprise Saas provider

vendor → uses technical aspect.

Salesforce → cloud vendors

↳ Customer Relationship Management.

Manages the sales, business. standard application

Key platform →
- Chatterwork.com,
- Salesforce1 platform
- Sales cloud
- Serve cloud

Key products
Key factors of CRM.

Salesforce communities.

Exact type target marketing cloud

Pardot

~~Sales~~

GENERAL CLOUD COMPUTING.

- Software support and Development
- e-business; business.
- IaaS
- PaaS
- SaaS
- Through Internet ✓

Business process → flowchart analysis (Users)

Stack Holders are classified as

* end users.
* cloud service providers (CSP's)
* cloud Tool providers (CTP's)
* cloud application vendors (CAV's)

(i) consumer of cloud application providers (services)

→ end users (Amazon, google)

(ii) Cloud service providers (provides cloud capability)

↑ (open source → information storage) net.

↳ Business

(iii) Managability of tools (Software protection dismantle).

(iv) vendor of cloud services (Amazon).

(i) Amazon cloud service.
(ii) Box.
(iii) Dropbox.
(iv) Egnyte
(v) Gsuite
(vi) Icbud
(vii) idrive
(viii) Microsoft one Drive
(ix) open drive
(x) Sugarsync.

It manage the sales with the help of the key products.

CLOUD BUILDING BLOCKS

IaaS
PaaS
SaaS.
} write how it is supporting to this Salesforce

Sales-forces.com

SaaS. → major role (software development)

→ proper delivery of the application over internet as a service

→ easily access. Internet through PC or computer.

Objectives followed in SaaS for Salesforce:

1) Campaign. (marketing tactics).

2) Leads (future perspective)
B to C (Business to ~~customer~~ Consumer)

3) Account. (unique entity)
(database to store data about million products)

4) Contacts. (contact detail of customer) → / employer/vendors

5) opportunity (financial transaction happens only at a particular time)
umbrella sweater

6) products

7) pricebook (catalog of standard price). ↓ content page.

8) quota
quotation → proposal price of services and products (fixing margin).

Service cloud consists of ~~CASES, SOLUTIONS, IDEAS, Q & A.~~ (reviews)

1. CASES (customer's description and complaints)
2. SOLUTIONS (solutions to customer issues) warranty
3. IDEAS (online suggestion, forms)
4. Q & A (questions - answers) → converted as case. (feedback good bad)
5. REPORTS (summarised representation of cloud).
6. DASHBOARDS (graphical representation of data)
7. DOCUMENTS (it is very ☺)
8. TASKS (scenario of the project).

PaaS in Salesforce.com

acts as a platform.

functions are based on Infrastructure (wide)

wide Infrastructure across the platform vendors.

Advanced features.

multitenant architecture database relation.

free for software developer

application oriented.

reassemble

Based on developers Paas is developed

Open standards
SOAP
REST
} Technologies

Apex ⇒ programming.
↳ programmed

virtualising computing resource from force. using JAVA
VM ware.

IaaS :

Resources sharing → main objective

↳ Benefit of the customer, vendor.

⇒ virtually resource sharing.

→ user → guest machine → uses virtual machine

→ provider → virtual machine monitor.

ex: virtualisation (⇒ for creation and run using virtual machines).

UNIT – III

COLLABORATING USING CLOUD SERVICES

CUSTOMER RELATIONS MANAGEMENT

Customer relationship management (**CRM**) is an approach to managing a company's interaction with current and potential customers. It uses data analysis about customers' history with a company to improve business relationships with customers, specifically focusing on customer retention and ultimately driving sales growth.

One important aspect of the CRM approach is the systems of CRM that compile data from a range of different communication channels, including a company's website, telephone, email, live chat, marketing materials and more recently, social media. Through the CRM approach and the systems used to facilitate it, businesses learn more about their target audiences and how to best cater to their needs.

Types

Strategic

Strategic CRM is concentrated upon the development of a customer-centric business culture.

Operational

The primary goal of customer relationship management systems is to integrate and automate sales, marketing, and customer support. Therefore, these systems typically have a dashboard that gives an overall view of the three functions on a single customer view, a single page for each customer that a company may have. The dashboard may provide client information, past sales, previous marketing efforts, and more, summarizing all of the relationships between the customer and the firm. Operational CRM is made up of 3 main components: sales force automation, marketing automation, and service automation.

- Sales force automation works with all stages in the sales cycle, from initially entering contact information to converting a prospective client into an actual client.It implements sales promotion analysis, automates the tracking of a client's account history for repeated sales or future sales and coordinates sales, marketing, call centers, and retail outlets. It prevents duplicate efforts between a salesperson and a customer and also automatically tracks all contacts and follow-ups between both parties.
- Marketing automation focuses on easing the overall marketing process to make it more effective and efficient. CRM tools with marketing automation capabilities can automate repeated tasks, for example, sending out automated marketing emails at certain times to customers, or posting marketing

information on social media. The goal with marketing automation is to turn a sales lead into a full customer. CRM systems today also work on customer engagement through social media.
- Service automation is the part of the CRM system that focuses on direct customer service technology. Through service automation, customers are supported through multiple channels such as phone, email, knowledge bases, ticketing portals, FAQs, and more.

Analytical

The role of analytical CRM systems is to analyze customer data collected through multiple sources and present it so that business managers can make more informed decisions. Analytical CRM systems use techniques such as data mining, correlation, and pattern recognition to analyze the customer data. These analytics help improve customer service by finding small problems which can be solved, perhaps by marketing to different parts of a consumer audience differently.For example, through the analysis of a customer base's buying behavior, a company might see that this customer base has not been buying a lot of products recently. After scanning through this data, the company might think to market to this subset of consumers differently, in order to best communicate how this company's products might benefit this group specifically

Collaborative

The third primary aim of CRM systems is to incorporate external stakeholders such as suppliers, vendors, and distributors, and share customer information across groups/departments and organisations. For example, feedback can be collected from technical support calls, which could help provide direction for marketing products and services to that particular customer in the future.

Customer Data Platform

A customer data platform (CDP) is a computer system used by marketing departments that assembles data about individual people from various sources into one database, with which other software systems can interact.As of February 2017 there were about twenty companies selling such systems and revenue for them was around US$300 million.

COMPONENTS

The main components of CRM are building and managing customer relationships through marketing, observing relationships as they mature through distinct phases, managing these relationships at each stage and recognizing that the distribution of value of a relationship to the firm is not homogeneous.

When building and managing customer relationships through marketing, firms might benefit from using a variety of tools to help organizational design, incentive schemes, customer structures, and more to optimize the reach of its marketing campaigns. Through the acknowledgement of the distinct phases of CRM, businesses will be able to benefit from seeing the interaction of multiple relationships as connected transactions.

The final factor of CRM highlights the importance of CRM through accounting for the profitability of customer relationships. Through studying the particular spending habits of customers, a firm may be able to dedicate different resources and amounts of attention to different types of consumers.

Relational Intelligence, or awareness of the variety of relationships a customer can have with a firm, is an important component to the main phases of CRM.

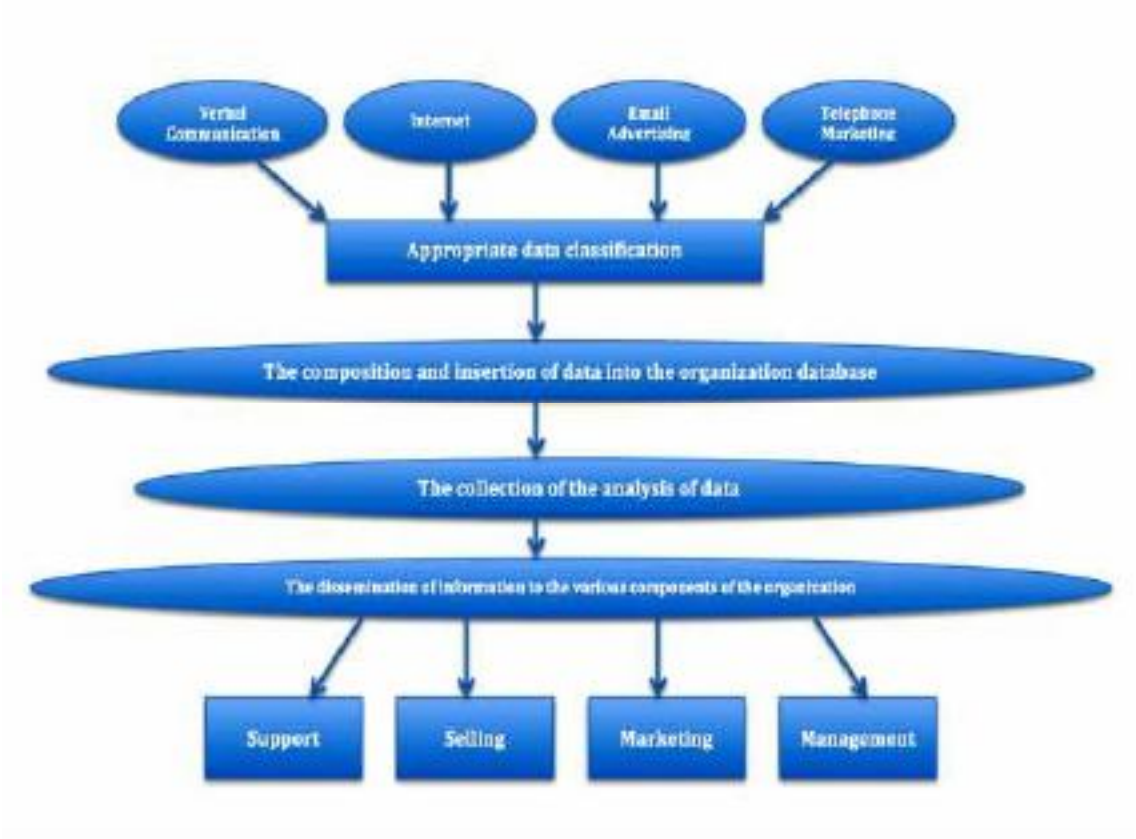
Companies may be good at capturing demographic data, such as gender, age, income, and education, and connecting them with purchasing information to categorize customers into profitability tiers, but this is only a firm's mechanical view of customer relationships.

This therefore is a sign that firms believe that customers are still resources that can be used for up-sell or cross-sell opportunities, rather than humans looking for interesting and personalized interactions.

CRM systems include:

- Data warehouse technology, used to aggregate transaction information, to merge the information with CRM products, and to provide key performance indicators.
- Opportunity management which helps the company to manage unpredictable growth and demand, and implement a good forecasting model to integrate sales history with sales projections.
- CRM systems that track and measure marketing campaigns over multiple networks, tracking customer analysis by customer clicks and sales.
- Some CRM software is available as a software as a service (SaaS), delivered via the internet and accessed via a web browser instead of being installed on a local computer. Businesses using the software do not purchase it, but typically pay a recurring subscription fee to the software vendor.
- For small businesses a CRM system may consist of a contact manager system that integrates emails, documents, jobs, faxes, and scheduling for individual accounts. CRM systems available for specific markets (legal, finance) frequently focus on event management and relationship tracking as opposed to financial return on investment (ROI).
- CRM systems for eCommerce, focused on marketing automation tasks, like: cart rescue, re-engage users with email, personalization.
- Customer-centric relationship management (CCRM) is a nascent sub-discipline that focuses on customer preferences instead of customer leverage. CCRM aims to add value by engaging customers in individual, interactive relationships.

- Systems for non-profit and membership-based organizations help track constituents, fundraising, sponsors' demographics, membership levels, membership directories, volunteering and communication with individuals.
- CRM not only indicates to technology and strategy but also indicates to an integrated approach which includes employees knowledge, organizational culture to em-brass the CRM philosophy.



BENEFITS OF COLLOBORATION

1. IMPROVED ORGANIZATION

With documents kept in a central, cloud-accessible location, employees can work on a document without having to send an updated version (not to mention trying to keep track of the latest version) to all the necessary team members.

2. HIGHER PARTICIPATION LEVELS

Allowing access to projects can lead to higher levels of employee participation. With cloud collaboration, all team members have an equal opportunity to provide input, and it can be done from wherever they are, at any time.

3. IMPROVED ACCESS TO LARGE FILES

Most email servers cannot handle documents larger than a few MB. When dealing with large audio or video files that email servers can't accommodate, cloud computing solutions have the answer. Because you can provide access to the cloud, where the large files are stored, there is no need to send files. Through the cloud, there is no delay in receipt or distribution dilemmas.

4. REAL-TIME UPDATES

Teams can work on projects without having to be in the same room, or even country. Edits and updates appear in real time and can be accessed by everyone. Any confusion over which version is the latest is eliminated with cloud collaboration.

5. BETTER BRAINSTORMING

The cloud can become a brainstorming forum, allowing ideas to be shared and productive conversations to take place. The cloud is an ideal medium to facilitate better communication between staff and project managers, various team members and other collaborators

6. REDUCED INVESTMENT

With cloud-based collaboration you only pay for the services you use. Unlike the olden time used when companies used to deploy local systems, cloud-based collaboration does that for you thus cutting a big percentage of expense. The advantage on this is that if your employee number increases, you only need to pay for extra users and if it reduces you can reduce a cut the cost.

Another evident advantage is when you are required to take over another project from another team and you gain funding, you can assimilate new users into your cloud platform.

7. SCALABILITY

In traditional based systems, if you need to increase the functions and capacities of your system, you will have a lot of expense. This ranges from hardware purchase, licensing, configurations and hiring consultants. With cloud-based platform you will only need to scale your price and the support will all be done for you.

8. HIGHER LEVELS OF PARTICIPATION

All employees want to be equal at something. Cloud collaboration benefits allow all employees to have an input. Every employee will have access to certain projects of which they can add their view anywhere and at any time. With ideas that cannot be directly shared from the executives to the employees, employees can have a chance to reach out to the whole institution without having to fear or pass through too many protocols that can be discouraging.

9. LARGE FILES ARE EASY TO ACCESS

While working with emails and email servers. Larger files more than a few megabytes cannot be handled. With cloud based collaboration platform; audios, videos and large files are easy to share. Because there is storage in a cloud server, there is no need to share or send your files. If you want anyone to get your file they just go directly to the intended cloud storage and get the file without delays.

10. UPDATES IN REAL TIME

Time zones and region are no longer a problem with cloud based collaboration. Teams no longer have to be physically present as they can be anywhere in the world. Updated edits and changes are real time for everyone. There is no earlier or later version. The present document is the latest so teams can work from there.

11. IMPROVED BRAINSTORM

When sharing ideas, project managers and other team members rarely have ease in communication. Cloud-based collaboration is a solution to that as team members can conduct brainstorming sessions to share ideas between themselves and even with project managers and can come up with better ways of doing project or find newer projects that may be profitable.

CRM MANAGEMENT

- Cloud CRM (or CRM cloud) means any customer relationship management (CRM) technology where the CRM software, CRM tools and the organization's customer data resides in the cloud and is delivered to end-users via the Internet (see "cloud computing").

- Cloud CRM typically offers access to the application via Web-based tools (or Web browser) logins where the CRM system administrator has previously defined access levels across the organization.

- Employees can log in to the CRM system, simultaneously, from any Internet-enabled computer or device. Often, cloud CRM provide users with mobile apps to make it easier to use the CRM on smartphones and tablets.

BENEFITS OF CRM MANAGEMENT

- One main benefit of CRM software delivered in the cloud is scalability. A cloud-based system is designed to be flexible with expanding capacity so a business can scale up (or down) their CRM depending on current business needs.
- Typically costs of the CRM, which is often based on the number of users and storage requirements also scales up and down as you requirements change. In most cases scaling up is as simple as contacting your cloud CRM vendor and requesting changes to your implementation.
- Cloud CRM is often a good choice for small businesses who lack the in-house IT expertise to deploy, manage and upgrade an on-premises CRM application.
- With Cloud CRM the vendor is responsible for managing the software, providing updates across the system and taking care of technical glitches, bugs and other issues as they arise.
- Other benefits of CRM in the cloud include integration with commonly used office applications and email systems, integration with social data (social CRM) and automatic data backups.

Customer relationship management (CRM) is a technology for managing all your company's relationships and interactions with customers and potential customers. The goal is simple: Improve business relationships. A CRM system helps companies stay connected to customers, streamline processes, and improve profitability.

CRM system, a tool that helps with contact management, sales management, productivity, and more.

A CRM solution helps you focus on your organization's relationships with individual people — including customers, service users, colleagues, or suppliers — throughout your lifecycle with them, including finding new customers, winning their business, and providing support and additional services throughout the relationship.

A CRM system gives everyone — from sales, customer service, business development, recruiting, marketing, or any other line of business — a better way to manage the external interactions and relationships that drive success.

A CRM tool lets you store customer and prospect contact information, identify sales opportunities, record service issues, and manage marketing campaigns, all in one central location — and make information about every customer interaction available to anyone at your company who might need it.

CRM can help companies of all sizes drive business growth, and it can be especially beneficial to a small business, where teams often need to find ways to do more with less.

CRM FOR BUSINESS

Gartner predicts that by 2021, CRM will be the single largest revenue area of spending in enterprise software. If your business is going to last, you know that

you need a strategy for the future. You have targets for sales, business objectives, and profitability.

Marketers can use a CRM solution to better understand the pipeline of sales or prospects coming in, making forecasting simpler and more accurate. You'll have clear visibility of every opportunity or lead, showing you a clear path from inquiries to sales.

Some of the biggest gains in productivity can come from moving beyond CRM as a sales and marketing tool, and embedding it in your business – from HR to customer services and supply-chain management.

Though CRM systems have traditionally been used as sales and marketing tools, customer service teams are seeing great benefits in using them.

A CRM platform lets you manage the inquiry across channels without losing track, and gives sales, service, and marketing a single view of the customer.

EMAIL BASED CLOUD COMMUNICATION

- It was reported that nearly 90% of the companies surveyed worldwide were still using on-premises email (or some other legacy option).
- For most businesses, email is one of the prime component services provided by IT.
- Although traditionally an on-premises email environment or a third-party hosted email solution, there has been an upsurge in organizations moving to Cloud-based email over the last few years.

BENEFITS OF EMAIL BASED CLOUD COMMUNICATION

Cost Savings

Cloud-based email is a subscription service model, which provides a significant cost savings for implementing and maintaining the environment. By switching to Cloud-based email, you are getting rid of your large capital outlay and converting your email to a monthly or annual operational expense.

Flexible Scalability

The scalability of Cloud-based email allows for an increase in future email capacity without having to do any major changes to the environment itself. With the increased scalability that Cloud-based email offers, your organization can experience the simplicity of being able to adjust your user license counts, storage capacity, and compute capacity either up or down, depending on your specific needs.

Improvement of Uptime

The third benefit of Cloud-based email is an improvement of uptime, which allows for more optimal email usage. Because many Cloud-based email providers use multiple redundant sites, and because your data will be stored in the Cloud, your organization will experience better uptime and disaster recovery response times than those organizations that still rely on their on-premise solutions.

Simplified Administration

Cloud-based email simplifies administration of your email system. By switching to a Cloud-based email system, you won't need to worry about version control issues or maintenance issues that may come up. As with all web-based applications, Cloud-based email simplifies your administrative needs.

Improved Security

When you use Cloud-based email system, your corporate emails will reside in an off-site, highly protected location, making it more secure than if it were on-premises. A Cloud-based email provider, like Microsoft, can devote way more resources to securing their facilities than most other organizations out there. Organizations that switch to Cloud-based email solutions, therefore, are taking a step in the right direction of further securing their data.

Remote Access

By switching to Cloud-based email, you can give your remote workers access to their email from wherever they are, making them more productive. A major trend that's happening right now in many industries is a push for a more mobile/remote workforce. Cloud-based email can be accessed from anywhere, at any time. All you need is an Internet connection. The convenient remote access to their emails is appreciated by many remote workers.

PROS OF CLOUD BASED EMAIL

- **No need to hire dedicated IT staff** to manage internal email servers, which results in a huge **cost savings** for your organization.
- Cloud-based email is **completely scalable** to your current situation. Need to hire more employees? Get additional licenses at the click of a mouse. Need to downsize your workforce? A few clicks of your mouse disable those user accounts so you're no longer charged for them.
- Cloud-based email is **always up-to-date** with the latest security patches and features.
- **No need to purchase software**, unless you're looking for a front-end redundancy. Email lives in the Cloud; if one of your systems goes down, it's still backed up there.
- **Integrating mobile devices** into the workplace is easier.
- A Service-oriented model creates **better service relationships** than typical "sell and forget" software.

Cons OF THE CLOUD BASED EMAIL

- Ongoing costs per user can add up over time.
- The Cloud provider is responsible for not only securing your service, but also providing you with visibility into how they secure it – if you don't go with a trustworthy Cloud provider, you may end up not getting access to the information you're entitled to.
- Cloud-based email solutions mean that you lose a certain degree of physical control over your email (and the fact that it's not stored on-premises), so it's important to pick a service provider you can trust.

EVENT MANAGEMENT IN CLOUD COMPUTING

Event management is not an easy job. Even when you are trying to manage a small event, you need to organize dozens of things and accomplish different sets of tasks.

Using a well-designed cloud-based event management software will ensure that you can take care of all the tasks with minimal fuss. Apart from increasing information accessibility and simplifying communication, it will also help you in many other ways.

1) Location Independence

If you use an event management software that runs on the cloud, you don't have to depend on a computer or any other device. You can access the software from any device, as long as it is connected to the Internet. Thus, a cloud-based event management software makes it easy for you to interact with employees, suppliers and clients.

2) Online Registrations and Online Payment

Registering and taking payment is an integral part of many events. This is one area where cloud technology can be a game changer. At a physical venue, you can use devices joined with the software to register users and complete all the processes that accompany registration – the data is automatically synced with the software. Automatic online registration also becomes easy when you are using an event management software as you can use it to take, track and manage online payments. It can also help you store the data regarding users and send notifications and reminders.

3) Increased Efficiency

A traditional event management software can help you manage just a few aspects of the event from the venue. But with a cloud-based event manager, you can leverage the features of the mobile app development while on the go. You can track events right on the spot, plan your event online while discussing ideas with the client or the supplier, have a single online diary where all event related notes taken by your staff are stored, export your data to different programs and devices, and do a dozen other important tasks anytime, anywhere. And not just you – any employee from your team can take the benefit of the cloud-based event management solution.

4) Reduced Costs

Not only do you get more efficiency when you use a cloud-based event software, but you also save money. Traditionally, cloud computing is known for its ability to cut down cost of investment in software, maintenance and electricity bills. When you use a cloud-based software, you can invest that money in business rather than waste it on infrastructure.

While event management is, at times, a labor-intensive job, the more efficiently you can manage your resources, the more profits you can make. Cloud-based event management tools can let you shift your procurement process online and it can also let you automate

several aspects of your work. This leads to reduction in the time and effort, which helps save money.

5) Management of Common Tasks

Apart from providing all the advantages pertaining to cloud, an event management software will also simplify all the common tasks associated with event management. Such a software will provide specific tools for simple events, multi-day events, conferences, exhibitions and festivals. It can act as a ticketing system, help you manage hotel and restaurant bookings, chart out travel plans, manage speakers and sponsors, and enable users to accomplish most tasks online. This will turn most of the tedious and labor-intensive tasks into easy, automated processes.

6) Wrapping up

CALENDAR IN CLOUD COMPUTING

Calendar software can be quite useful. Busy executives might refer to a calendar program as they navigate from one business meeting to another. Popular socialites use them as they book parties and other events. And the perpetually disorganized depend on calendar software to avoid missing important appointments. There are several calendar programs on the market. One such application that's growing in popularity is the **Google Calendar**.

Created by the multi-billion dollar corporation Google, the Google Calendar application allows users to create personal or public calendars after signing up for a Google account. The accounts are free, and Google stores the calendars within its **cloud computing** system. That means that the company stores the application and user information on its own servers. The user doesn't have to download special software to access the calendar -- all access is through a Web browser.

With traditional desktop software, users store information to their own computers' hard drives or other storage devices. That means if they want to access their information, they always have to use the same computer. Since the information in Google Calendars exists on the Web, users can view and make changes to calendars from any computer connected to the Internet

Storing calendars on the Internet also means that it's easier to share information with other users. In turn, scheduling events and creating invitations becomes easier. Want to know if a fellow Google Calendar owner is free this weekend? If the owner opts to share his or her calendar with you, you'll be able to see if there

are any scheduled events that might interfere with your plans. If there are conflicts, you may be able to adjust your plans.
Like all tools, Google Calendar is only helpful when people use it. It doesn't magically organize your life, but it can make the task much easier for you.

WORD PROCESSING IN CLOUD COMPUTING

Google offers a variety of useful free tools merchants can use in place of other more costly software. One of these is Google Docs, a free word processing tool with plenty of room for file storage and many unique tools unmatched by other word processors. The most important of these is the collaborative tools where users can easily share documents with other users.

Sharing Features

Google Docs's main dashboard is a list of all documents, along with sharing information. Docs makes it easy for users to share any sort of document with other users, regardless of whether or not they have a Google account. Shared documents can be locked, as well, so others can only access them as a read-only document. Or, if a project has multiple writers, users can choose to allow others to work on the document with them. The "owner" of the document is the one with all the sharing power, but fortunately Google Docs allows users to change the owner so there is little interruption to work flow. Collaborators also have the ability to chat with each other in real time, using a feature similar to GChat, without leaving the document.

Screen capture showing all documents on Google Docs.

One thing Docs could add to improve its sharing functions would be to save multiple copies of documents so users could return to previous versions. For example, if one user makes a mistake while editing the document and saves it, another would have to go back in and change it. Other online collaborative word processors, such as Basecamp by 37Signals, save multiple versions of documents so there is no work lost. Users can return to older versions in case of errors, something unavailable in Google Docs.

Word Processor

Docs's most basic feature is modeled after word processors like Microsoft Word. With a Google account, users can create a blank document that is viewable in print format. Here you can see the basic format, using text from the Emancipation Proclamation as an example.

Users have a more limited set of fonts than with other word processors—only Arial, Times New Roman, Cominc Sans, Courier, Garamond, Georgia, Tahoma, Trebuchet and Verdana. Font nerds like me will be dismayed at the omission of Helvetica, but it's a comprehensive enough list for most purposes.

One unique aspect of Google Docs is the translation tool, which allows users to translate an entire document into a wide variety of languages. This could be useful for merchants who could copy a body of text, then use the machine translator to get a rough translation. Then the merchant could share the document with a human translator who could edit the document and re-save it. Because the document is stored online, there is no need for tedious copying, pasting and attaching files to emails. It also frees up hard drive space.

Translation feature on Google Docs.

Another unique feature is the expanded ability to insert other elements than text into the document. As with Word, users can insert images, links, comments, headers, footers and a table of contents. But they can also add some more extended elements, such as mathematical equations and drawings. Here's an example of a hasty drawing.

Insert options on Google Docs.

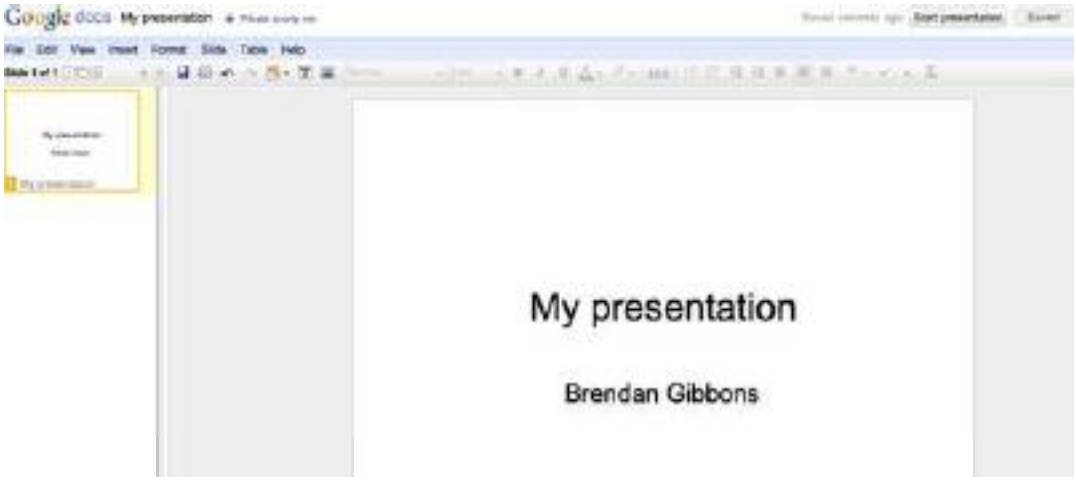
Drawings on other processors are usually limited to predefined templates, and it can be very difficult to add mathematical symbols into normal documents. Usually to do either of these, I have to use another program and copy-and-paste it into the document or upload a file. Unfortunately, this often tweaks the font and paragraph formatting.

Google Docs's auto-save feature has also saved me numerous times. Before I became more diligent with saving my documents on other word processors, I lost a lot of my work due to crashes and accidental window exiting. Google Docs saves your work every minute or so, making it easy to hold onto.

Other Templates

With Docs, there are many more options than just creating a text document. Users can choose from an ever-expanding variety of templates, from presentations to spreadsheets to forms.

The presentation feature is basically a slimmed-down version of Microsoft PowerPoint, allowing users to create a slideshow. They can share these presentations with other viewers or collaborators and publish them online under a unique URL, so there is no need to attach them in emails. This solves the problem of people using different versions of presentation software and not being able a document because it is not saved as a compatible file type.



Example of PowerPoint on Google Docs.

Merchants can use many other Docs templates to create spreadsheets for accounting, order management and other purposes. While this would require much manual input and probably would not be useful for large ecommerce companies, it could be a good tool to manage personal finances or small inventories. These spreadsheets have comparable features to Microsoft Excel, but because they are all stored online, there is no chance of losing important information.

Google has also approved hundreds of templates, both from its own developers and other Docs users. These range from business cards to birthday cards, letters, resumes and much more. After perusing dozens of these templates and seeing the hundreds that are available, it is safe to say that Docs offers templates for most any type of document you might want to create.

Broad, Not Deep

Google Docs is a vast free resource for word processing, spreadsheets, presentations and much more. It does all of these things well, though it is more comprehensive than it is extensive. Like most Google services, it is free, collaborative and constantly being updated. However, it seems the only thing Docs does better than many other software available is ordinary word processing.

SPREADSHEET IN CLOUD COMPUTING

Cloud Computing has caused a paradigm shift in the world of computing. Several use case scenarios have been floating around the programming world in relation to this. Applications such as Spreadsheets have the capability to use the Cloud framework to create complex web based applications. In our effort to do the same, we have proposed a Spreadsheet on the cloud as the framework for building new web applications, which will be useful in various scenarios, specifically a School administration system and governance scenarios, such as Health and Administration. This paper is a manifestation of this work, and contains some use cases and architectures which can be used to realize these scenarios in the most efficient manner.

As enterprises and organizations across the world gear up to embrace the technology, it is up to the creators of software to localize Cloud Computing and customize it in a manner suitable to the situation in which it is being deployed. It has also become imperative to import the existing systems into the corresponding cloud versions and utilize them to expand our presence in the cloud.

One such essential platform is that of the Spreadsheet. In the business world, the enterprises and education system, the Spreadsheet has become an indispensable tool of productivity and organization. The Spreadsheet is not just an effective information system, but it is also a strong and reliable framework for building applications. While the movement of Spreadsheet towards the cloud has started taking place in various forms such as Google Docs and Editgrid, what remains to be done is using the Spreadsheet framework over the Cloud to create innovative services which utilize the mathematical and programming capabilities of the activity and at the same time leverage upon the collaborated environment of the cloud. Not only this, these services must also have the capability to be customized for the typical use-cases. Through this paper, we examine various such scenarios.

Cloud Computing has caused a paradigm shift in the world of computing. Several use case scenarios have been floating around the programming world in relation to this. Applications such as Spreadsheets have the capability to use the Cloud framework to create complex web based applications. In our effort to do the same, we have proposed a Spreadsheet on the cloud as the framework for building new web applications, which will be useful in various scenarios, specifically a School administration system and governance scenarios, such as Health and Administration. This paper is a manifestation of this work, and contains some use cases and architectures which can be used to realize these scenarios in the most efficient manner.

The main idea of the Spreadsheet activity is to include features that would enable children to make easy use of the typical features of Spreadsheet activities

such as organization, graphing and simple calculations in their respective languages. The main features of this spreadsheet activity are:

Tabulation

Organization

· Graphing and Calculation

· Localization in different languages

· Multi-user editing over the mesh network

· Ability to read and edit single sheet Excel 1997-2003 (.xls), Lotus (.wk4) and other popular spreadsheet files

· Optimization in saving of sheet data.

· Collaboration over the Cloud

· Chat integration

TAKING SPREADSHEET TO THE CLOUD

In order to take our Spreadsheet Activity to the cloud, we devised an architecture which enables the application to reside on the school server, common to all the XO laptops. All the systems integrated to the cloud access the application on the browser, while the server handles operations such as saving, etc. Though SocialCalc was ready to be used by individual browsers on their XOs, additional infrastructure was needed in order to support collaboration. Changes were introduced in the Python as well as JavaScript parts with XOCOM acting as the base, to create the infrastructure.

To put into effect this use-case, we first accomplished the same on an established Cloud server, that is, the Google App Engine. The Python code was used for the server side scripting, while the JavaScript code running on the Browser acted as the main activity.This application was named SocialCalcNet.

FEATURES OF SPREADSHEET ON THE CLOUD

Login - User login is required to keep record of the sheets that a user creates. SocialCalcNet provides the user an option of logging in either using his

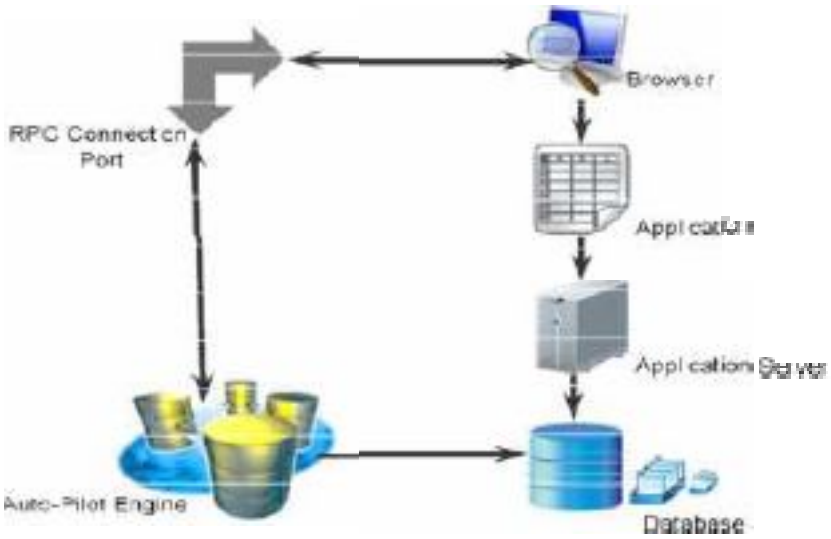
Googleaccount or creating his own account with the application. Whenever a user logs in, a new session is created which identifies each user.

Account Details - This displays the account details of the user and also provides the options for further actions – new sheet, load a previous sheet, edit account details, etc.

Edit Account - Users can edit their accounts created with the application. They can change username or password. However, Google account details cannot be changed.

Socialcalc Sheet - Users can create new sheets and work on them. A tab by the name Options was added to the application. This tab provides options to the user such as saving the current sheet,opening a previous sheet or logging out of the current account.

Once the basic collaboration over the cloud through the Spreadsheet was achieved, we devised architecture for using this Spreadsheet as a chassis for building several other Activities which require the support of a Spreadsheet. The diagram below shows this architecture in brief:



SOCIAL NETWORKS IN CLOUD COMPUTING

With the increasingly ubiquitous nature of Social networks and Cloud computing, users are starting to explore new ways to interact with, and exploit these developing paradigms. Social networks are used to reflect real world relationships that allow users to share information and form connections between one another, essentially creating dynamic Virtual Organizations.

We propose leveraging the pre-established trust formed through friend relationships within a Social network to form a dynamic "Social Cloud", enabling friends to share resources within the context of a Social network. We believe that combining trust rela- tionships with suitable incentive mechanisms (through financial payments or bartering) could provide much more sustainable resource sharing mechanisms.

This paper outlines our vision of, and experiences with, creating a Social Storage Cloud, looking specifically at possible market mechanisms that could be used to create a dynamic Cloud infrastructure in a Social network environment.

Social networking has become an everyday part of many peoples' lives as evidenced by the huge user communities. Some communities even exceed the population of large coun- tries, for example Facebook has over 400 million active users.

Social networks provide a platform to facilitate communication and sharing between users, therefore modelling real world relationships. Social networking has also extended beyond communication between friends, for instance, there are a multitude of integrated applications and some organizations even utilize a user's Facebook credentials for authentication rather than requiring their own credentials (for example the Calgary Airport authority in Canada uses Facebook Connect to grant access to their WiFi network).

The structure of a Social Network is essentially a dynamic virtual organization with inherent trust relationships between friends. We propose using this trust as a foundation for resource (information, hardware, services) sharing in a Social Cloud.

Cloud environments typically provide low level ab- stractions of computation or storage. Computation and Storage Clouds are complementary and act as building blocks from which high level service Clouds and mash-ups can be created.

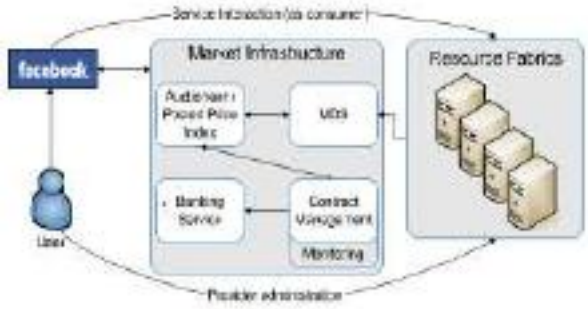
Storage Clouds are often used to extend the capabilities of storage-limited devices such as phones and desktops, and provide transparent access to data from anywhere. There are a large number of commercial Cloud providers such as Amazon EC2/S3, Google App Engine, Microsoft Azure and also many smaller scale open Clouds like Nimbus and Eucalyptus .These Clouds provide access to scalable vir- tualized resources (computation, storage, applications) through pre-dominantly posted price mechanisms.

A Social Cloud, therefore, is a scalable computing model in which virtualized resources contributed by users are dynamically provisioned amongst a group of friends. Compensation for use is optional as users may wish to share resources without payment, and rather utilize a reciprocal credit (or barter) based model.

The Social Cloud architecture presented is designed as a Facebook application, to make use of this widely used platform development environment and API.

In a Social Cloud, services can be mapped to particular users through Facebook identification, allowing for the definition of unique policies regarding the interactions between users. For example, a user could limit trading with close friends only, users in the same country/network/group, all friends, or even friends of friends.

A specialized banking component manages the transfer of credits between users while also storing information relating to current reservations. A high level architecture of a Social Cloud



FACEBOOK APPLICATIONS

Facebook exposes an application API through a REST- like interface which includes methods to get a range of data including friends, events, groups, application users, profile information, and photos.

Facebook Markup Language (FBML) includes a subset of HTML with proprietary extensions that enables the creation of applications that integrate completely with the Facebook look and feel. Facebook JavaScript (FBJS) is Facebook's version of JavaScript – rather than sandboxing JavaScript, FBJS is parsed when a page is loaded to create a virtual application scope.

Facebook applications are hosted independently and are not hosted within the Facebook environment. A Facebook canvas URL is created for user access, this URL maps to a user defined callback URL which is hosted remotely. The process of rendering an application page.

When a page is requested by the user through the Facebook Canvas URL (http://apps.facebook.com/socialcloud/) the Facebook server forwards the request to the defined callback URL. The application creates a page based on the request and returns it to Facebook. At this point the page is parsed and Face- book specific content is added according to the FBML page instructions.

The final page is then returned to the user. This routing structure presents an important design consideration in a Social Cloud context as access to the Cloud services would be expensive if routed through both the Facebook server and the callback application server in order to get data from the actual Cloud services.

To reduce the effect FBJS can be used to request data asynchronously from the specified service in a transparent manner without routing through the application servers.

GROUPWARE

Collaborative software was originally designated as *groupware* and this term can be traced as far back as the late 1980s, when Richman and Slovak (1987) wrote: "Like an electronic sinew that binds teams together, the new *groupware* aims to place the computer squarely in the middle of communications among managers, technicians, and anyone else who interacts in groups, revolutionizing the way they work."

Even further back, in 1978 Peter and Trudy Johnson-Lenz coined the term groupware; their initial 1978 definition of groupware was, "intentional group processes plus software to support them." Later in their article they went on to explain groupware as "computer-mediated culture... an embodiment of social organization in hyperspace." Groupware integrates co-evolving human and tool systems, yet is simply a single system.

In the early 1990s the first commercial groupware products were delivered, and big companies such as Boeing and IBM started using electronic meeting systems for key internal projects. Lotus Notes appeared as a major example of

that product category, allowing remote group collaboration when the internet was still in its infancy.

Kirkpatrick and Losee (1992) wrote then: "If GROUPWARE really makes a difference in productivity long term, the very definition of an office may change. You will be able to work efficiently as a member of a group wherever you have your computer. As computers become smaller and more powerful, that will mean anywhere." In 1999, Achacoso created and introduced the first wireless groupware.

Design and implementation issues

The complexity of groupware development is still an issue. One reason for this is the socio-technical dimension of groupware. Groupware designers do not only have to address technical issues (as in traditional software development) but also consider the organizational aspects and the social group processes that should be supported with the groupware application. Some examples for issues in groupware development are:

- Persistence is needed in some sessions. Chat and voice communications are routinely non-persistent and evaporate at the end of the session. Virtual room and online file cabinets can persist for years. The designer of the collaborative space needs to consider the information duration needs and implement accordingly.
- Authentication has always been a problem with groupware. When connections are made point-to-point, or when log-in registration is enforced, it's clear who is engaged in the session. However, audio and unmoderated sessions carry the risk of unannounced 'lurkers' who observe but do not announce themselves or contribute.
- Until recently, bandwidth issues at fixed location limited full use of the tools. These are exacerbated with mobile devices.
- Multiple input and output streams bring concurrency issues into the groupware applications.
- Motivational issues are important, especially in settings where no pre-defined group process was in place.
- Closely related to the motivation aspect is the question of reciprocity. Ellis and others have shown that the distribution of efforts and benefits has to be carefully balanced in order to ensure that all required group members really participate.
- Real-time communication via groupware can lead to a lot of noise, over-communication and information overload.

One approach for addressing these issues is the use of design patterns for groupware design. The patterns identify recurring groupware design issues and discuss design choices in a way that all stakeholders can participate in the groupware development process.

Groupware and levels of collaboration

Groupware can be divided into three categories depending on the level of collaboration:

1. **Communication** can be thought of as unstructured interchange of information. A phone call or an IM Chat discussion are examples of this.
2. **Conferencing** (or collaboration level, as it is called in the academic papers that discuss these levels) refers to interactive work toward a shared goal. Brainstorming or voting are examples of this.
3. **Co-ordination** refers to complex interdependent work toward a shared goal. A good metaphor for understanding this is to think about a sports team; everyone has to contribute the right play at the right time as well as adjust their play to the unfolding situation - but everyone is doing something different - in order for the team to win. That is complex interdependent work toward a shared goal: collaborative management.

Collaborative management (coordination) tools

Collaborative management tools facilitate and manage group activities. Examples include:

- Electronic calendars (also called time management software) — schedule events and automatically notify and remind group members
- Project management systems — schedule, track, and chart the steps in a project as it is being completed
- Online proofing — share, review, approve, and reject web proofs, artwork, photos, or videos between designers, customers, and clients
- Workflow systems — collaborative management of tasks and documents within a knowledge-based business process
- Knowledge management systems — collect, organize, manage, and share various forms of information
- Enterprise bookmarking — collaborative bookmarking engine to tag, organize, share, and search enterprise data
- Prediction markets — let a group of people predict together the outcome of future events
- Extranet systems (sometimes also known as 'project extranets') — collect, organize, manage and share information associated with the delivery of a project (e.g.: the construction of a building)
- Intranet systems — quickly share company information to members within a company via Internet (e.g.: marketing and product info)
- Social software systems — organize social relations of groups
- Online spreadsheets — collaborate and share structured data and information
- Client portals — interact and share with your clients in a private online environment

Collaborative software and human interaction

The design intent of collaborative software (groupware) is to transform the way documents and rich media are shared in order to enable more effective team collaboration.

Collaboration, with respect to information technology, seems to have several definitions. Some are defensible but others are so broad they lose any meaningful application. Understanding the differences in human interactions is necessary to ensure the appropriate technologies are employed to meet interaction needs.

There are three primary ways in which humans interact: conversations, transactions, and collaborations.

Conversational interaction is an exchange of information between two or more participants where the primary purpose of the interaction is discovery or relationship building. There is no central entity around which the interaction revolves but is a free exchange of information with no defined constraints, generally focused on personal experiences.[28] Communication technology such as telephones, instant messaging, and e-mail are generally sufficient for conversational interactions.

Transactional interaction involves the exchange of transaction entities where a major function of the transaction entity is to alter the relationship between participants.

In *collaborative interactions* the main function of the participants' relationship is to alter a collaboration entity (i.e., the converse of transactional). When teams collaborate on projects it is called Collaborative project management.

Collaborative software or **groupware** is application software designed to help people working on a common task to attain their goals. One of the earliest definitions of groupware is "intentional group processes plus software to support them".

As regards available interaction, collaborative software may be divided into: real-time collaborative editing platforms that allow multiple users to engage in live, simultaneous and reversible editing of a single file (usually a document), and version control (also known as revision control and source control) platforms, which allow separate users to make parallel edits to a file, while preserving every saved edit by every user as multiple files (that are variants of the original file).

Collaborative software is a broad concept that overlaps considerably with computer-supported cooperative work (CSCW). According to Carstensen and Schmidt (1999) groupware is part of CSCW. The authors claim that CSCW, and thereby groupware, addresses "how collaborative activities and their coordination can be supported by means of computer systems."

The use of collaborative software in the work space creates a collaborative working environment (CWE).

Finally, collaborative software relates to the notion of collaborative work systems, which are conceived as any form of human organization that emerges any time that collaboration takes place, whether it is formal or informal, intentional or unintentional.

Whereas the groupware or collaborative software pertains to the technological elements of computer-supported cooperative work, collaborative work systems become a useful analytical tool to understand the behavioral and organizational variables that are associated to the broader concept of CSCW

What is **Groupware**?

- Software *specifically* designed
  - to support group working
  - with cooperative requirements in mind
- NOT just tools for communication
- Groupware can be classified by
  - *when* and *where* the participants are working
  - the *function* it performs for cooperative work
- Specific and difficult problems with groupware implementation

Types of architecture

centralised – single copy of application and data
- client-server – simplest case
  - N.B. opposite of X windows client/server
- master-slave special case of client-server
  - N.B. server merged with one client

replicated – copy on each workstation
- also called peer-peer
- + local feedback
- race conditions

Often 'half way' architectures:
- local copy of application + central database
- local cache of data for feedback
- some hidden locking

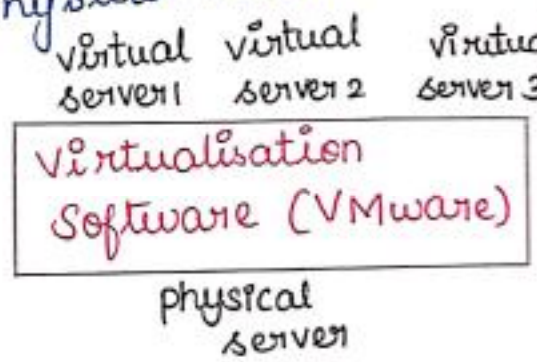
UNIT – IV

VIRTUALISATION FOR CLOUD

Compare VMware and Cloud Computing

VMware	Cloud Computing
1) SCALABILITY	
Virtual machines configuration limits its scalability	Cloud can be extended as much as you want
2) QUICK SETUP	
It is very simple to setup virtual environ-ment	Setting up cloud is very tidious task.
3) FLEXIBILITY	
Proper authentication is required before accessing the virtual machines	It is very flexible for user access. User can access its cloud from any locating with internet (depending upon permission)
4) DEDICATED HARDWARE	
Dedicated hardware required for multiple virtual machines	Multiple hardware creats cloud computing
5) INTEGRATION	
Virtualisation integration allows expansion of new machines within the same infrastructure.	Cloud integration allows future expansion of users, application etc.

VMware	Cloud Computing
6) Dependency	
Multiple OS can be installed on single server or computer	Multiple user can access the network using same link.
7) ACCESSIBILITY	
Proper permissions are required for accessing from outside the network	It can be accessed from all over the world (Internet based cloud)
8) DISASTER RECOVERY	
Single machine failure can bring damage to multiple virtual machines	It does not dependent on one machine.
9) Definition	
Virtualisation is a software that virtualises your hardware into multiple machines.	Cloud Computing is the combination of multiple hardware devices.
10) USER LOGIN	
In Virtualization, a user gets dedicated hardware.	In cloud Computing, multiple hardware devices provide one login environment for the user

VMware	Cloud Computing
Cloud VM is a software that creates seperated multiple images of the hardware and software on the same machine. This makes possible to install multiple os, multiple software and multiple applications on the same physical machine. 	Cloud Computing is a model for enabling convenient on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction 
~~Various applications, storages and infrastructure servers are running over~~ VM software has divided the physical servers into multiple machines and all the virtual machines reside with the same physical server itself	Various applications, storages and infrastructure servers are running over the cloud and accessible for all type devices such as mobile phones. Cloud is useful for external user access.

VMware	Cloud Computing
Server virtualisation is the top reason behind its success.	It is accessible to all the users without any restrictions.
Multiple applications can be installed on a single physical machine despite OS dependency.	Least the possibility of access failure due to non-dependency on a single machine.

High Level Language Virtual Machine.

* The collection of binary classes or modules that make up a program specify specify its operation in terms of HLL VM architecture.
* It is up to the HLL VM implementation to carry out the specified operations.
* There are many HLL VM implementation But the best known high level language implementation is described by JAVA VM.
* A CLI implementation is similar in many respects.
* A We describe the major implementation components and then look at ways of improving performance in HLL VM. implementations.

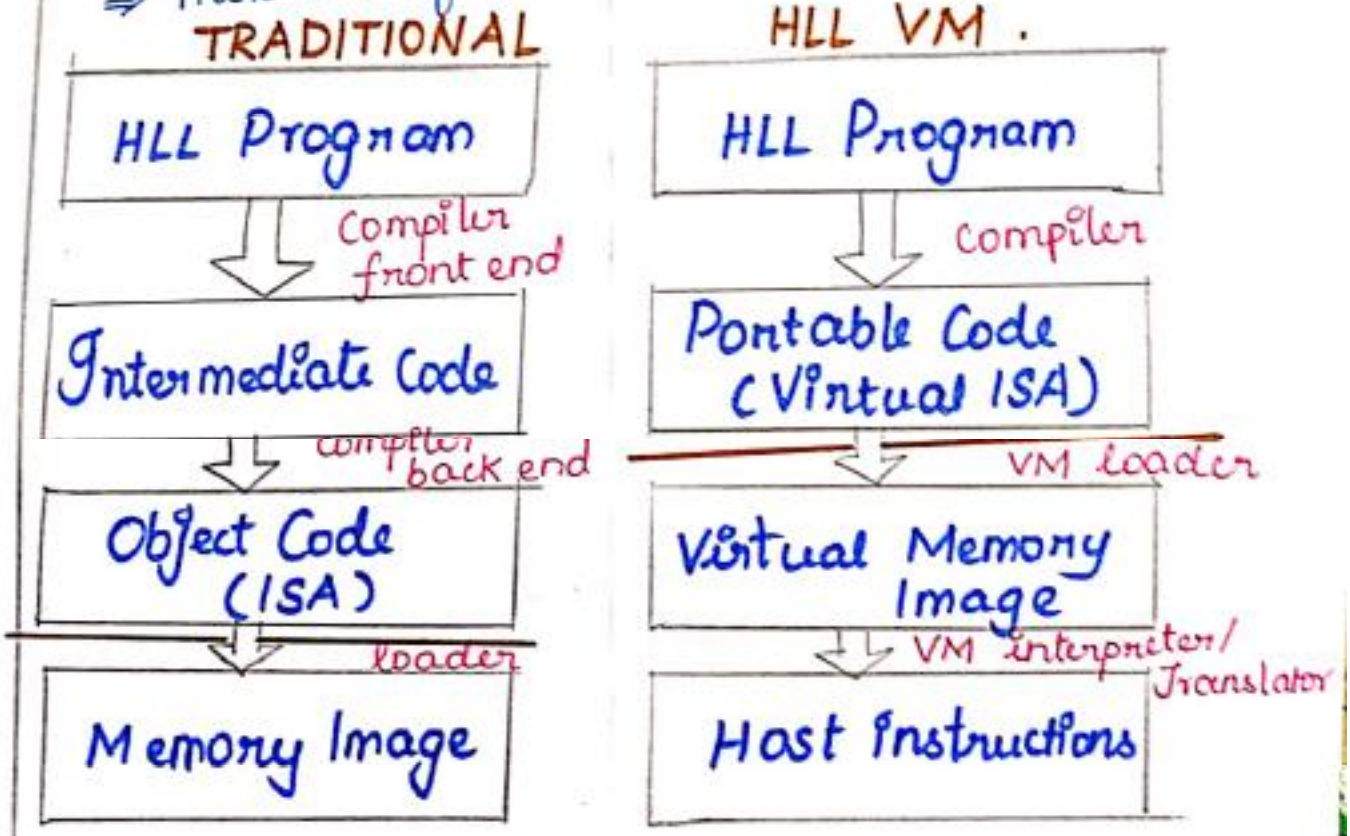
HLL VM is similar to Process VM

but ISA defined for user-mode programs only.

* ISA not designed for real hardware
  → Only to be executed on virtual processor.
  → Referred to as virtual ISA or V-ISA.
* System Interface is a set of standardised API's.

HLL VMs from language/compiler perspective:

GOAL: complete platform independence for applications. Virtual instruction set + libraries
⇒ Instead of ISA and OS Interface.

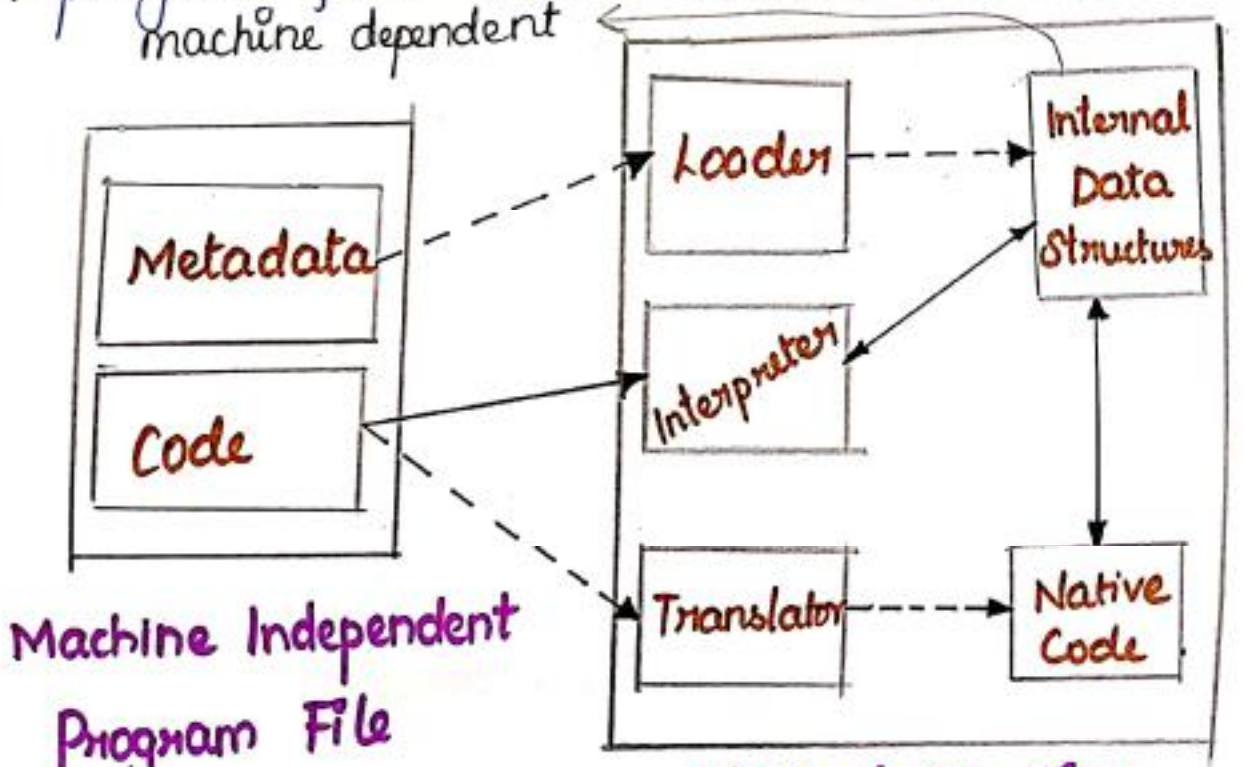


MODERN HLL VMs

* Superficially similar to Pcode scheme
  - Stack oriented ISA
  - Standard libraries
* Network computing Environment
  - → Untrusted software (This is Internet, after all)
  - → Robustness (Generally a good idea)
    - ⇒ object oriented programming
  - → Bandwidth is a consideration
  - → Good performance must be maintained
* Two major examples
  - → JAVA VM
  - → Microsoft Common Language Infrastructure (CLI)

MODERN HLL VMs

* Compiler forms program files (eg: class files)
  std. format
* program files contain both code and metadat
  machine dependent



Machine Independent Program File

Virtual Machine Implementation

Terminology:

* JAVA virtual machine Architecture ⇒ CLI
* JAVA virtual Machine Implementation ⇒ CLR (Common Language Runtime)
* JAVA bytecodes ⇒ Microsoft Intermediate Language (MSIL), CIL, IL.
* JAVA platform ⇒ .NET framework.
  → ISA + Libraries; a higher level ABI

Characteristics of HLL VMs

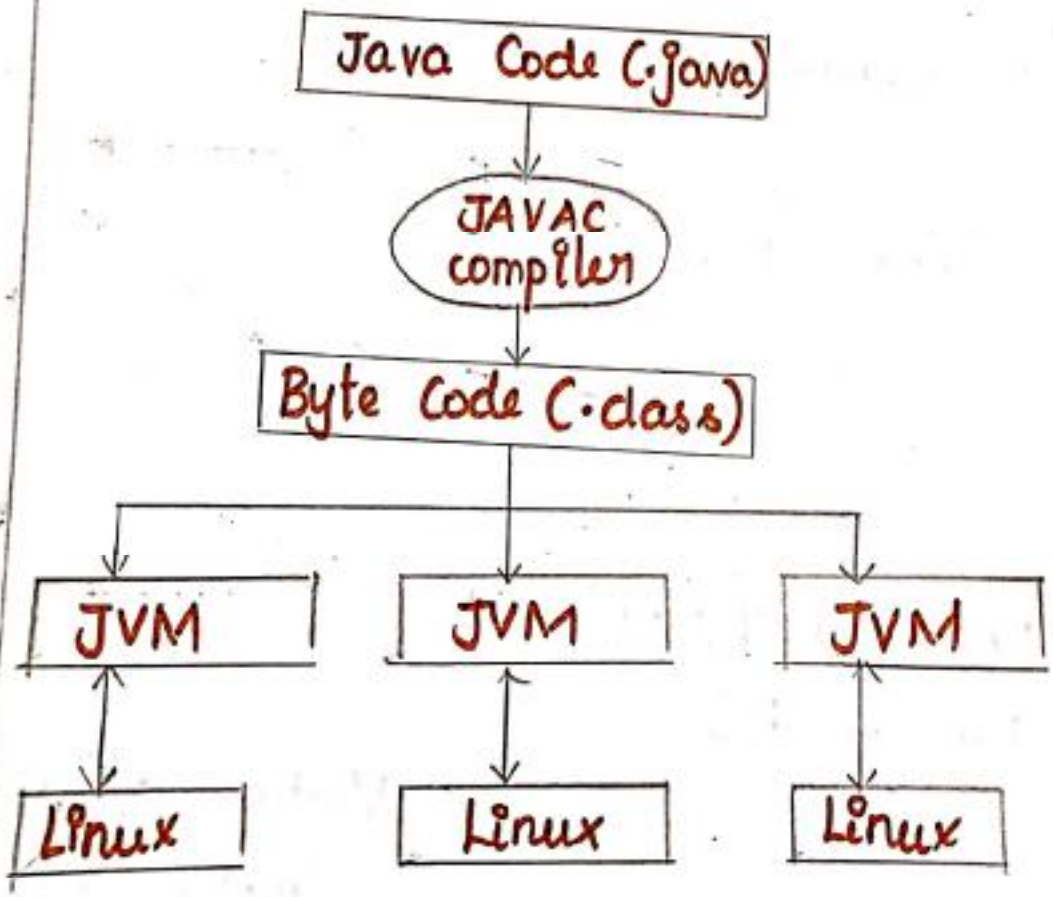
Security

Robustness

Networking

Performance.

JAVA VIRTUAL MACHINE (JVM)



JVM Bytecode Emulation :

* Interpretation
  * simple, fast startup
* Just-in-Time (JIT) Compilation
  * simple static optimizations
  * compile each ~~no~~ method
* Hot-Spot Compilation .
  * Find frequently executed code
  * Apply more aggressive optimizations on that code
  * Typically phased with interpretations on JIT.
* Dynamic Compilation
  * Based on Hotspot compilation
  * Use runtime information to optimize .

JVM is hence :

* an abstract entity that gives meaning to class files
  * Has many concrete implications
    → hardware
    → interpreter
    → JIT compiler .
  * Persistence .
    → an instance is created when application starts and terminates when application ends.

DIFFERENT TYPES OF VIRTUALISATION

(i) Server Virtualisation

(ii) Client/ Desktop / Application Virtualisation

(iii) Network virtualisation

(iv) Storage Virtualisation

(v) Service/ Application Infrastructure Virtualisation

(vi) Hardware Virtualisation.

(vii) Operating System Virtualisation.

Server Virtualisation :

(i) Server Virtualisation is the most active segment of the virtualisation industry featuring established companies such as VMware, Microsoft, Citrix.

(ii) With server virtualisation one physical machine is divided into many virtual servers.

(iii) The core is the concept of hypervisor (virtual monitor.

Application / Desktop Virtualization :

* Application virtualisation is an umbrella term that describes software technologies that improve manageability and compatability of legacy applications by encapsulating applications from underlying OS on which they are executed.
* They are classified as Local and Hosted.

Network Virtualisation :

* In computing, network virtualisation is the process of combining hardware and software network resources and network functionality into a single, software based administrative entity, a virtual network.
* Network virtualisation involves platform virtualisation, often combined with resource virtualisation.

Storage Virtualisation :

* Storage Virtualisation refers to the process of abstracting logical storage from physical storage.

Service/Application Infrastructure Virtualisation

* Application Infrastructure virtualisation (sometimes referred to as application fabrics) unbundle an application from a physical OS and hardware.
* Application developers can then write to a virtualization layer.

Hardware & Virtualization:

* Hardware Virtualisation used in server platform as it is flexible to use Virtual Machine. rather than physical machines.
* Virtual machine software is installed in the hardware system and this is called as hardware virtualisation.

Operating System Virtualisation:

* In this, the virtual machine software installs in the operating system of the host rather than directly on the hardware system.
* The most important use of operating system virtualization is for testing the application on different platforms or operating systems.

Characteristics of Virtualisation:

* In virtualisation, a single physical infrastructure can be used to run multiple operating systems (OSs) and applications.
* Virtualisation is a technology that enables the single physical infrastructure to function as a multiple logical infrastructure or resources.
* Virtualisation is the process of abstracting the physical resources to the pool of virtual resources that can be given to any virtual machines (VMs).

(i) Isolation:

* It provides fault and security isolation at the hardware level.
* It preserves performance with advanced resource control.

(ii) Hardware Independence:

* It can provide provision or migrate any virtual machine to any physical server.

(iii) Partitioning :

* It can run multiple operating systems on one physical machine.
* It divides system resources between virtual machines.

(iv) Encapsulation :

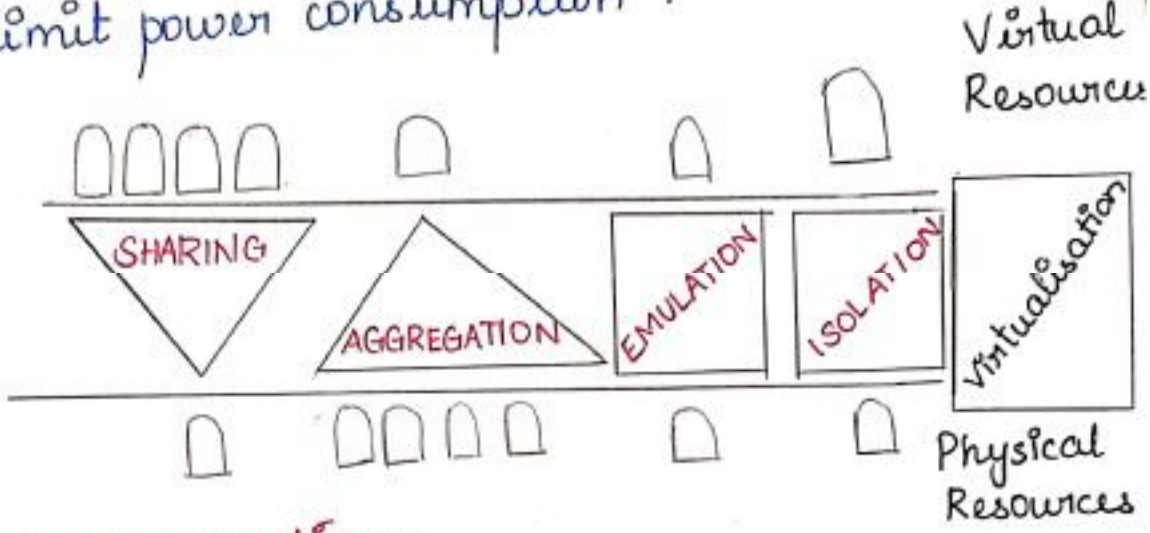
* It can save the entire state of a virtual machine to files.
* It can move and copy virtual machines as easily as moving and copying files.

(v) Increased Security :

* The ability to control the execution of a guest programs in a completely transparent manners opens new possibilities for delivering a secure, controlled execution environment.
* All the operations of the guest programs are generally performed against the virtual machine, which then translates and applies them to the host programs.
* A virtual machine manager can control and filter the activity of the guest programs thus preventing some harmful operations from being performed.
* Increased security is a requirement when dealing with untrusted code.

(vi) Sharing :

* Virtualisation allows the creation of a seperate computing environments within the same host. This basic feature is used to reduce the number of active servers and limit power consumption.



(vii) Aggregation :

* Virtualisation also allows aggregation
* A group of seperate hosts can be tied together and represented to guests as a single virtual host.
* This functionality is implemented with cluster management software, which harnesses the physical resources of a homogeneous group of machines and represents them as a single resource.

(viii) Emulation :

* Guests programs are executed within an environment that is controlled by the virtualisation layer, which ultimately is a program.
* Also a completely different environment with respect to the host can be emulated, thus allowing the execution of guest programs requiring specific characteristics that are not present in the physical host.

(ix) Isolation :

* Virtualisation allows providing guests - whether they are operating systems, applications, or other entities - with a completely seperate environment, in which they are executed.
* The guest program performs its activity by interacting with an abstraction layer, which provides access to the underlying resources.
* The virtual machine can filter the activity of the guest and prevent harmful operations against the host.

x) Performance Tuning

* The important capability enabled by virtualisation is performance tuning.
* This feature is a reality at present, given the considerable advances in hardware and software supporting virtualisation.
* It becomes easier to control the performance of the guest by finely tuning the properties of the resources exposed through virtual environment.
* This capability provides a means to effectively implement a quality-of-service (QoS) infrastructure

xi) Portability:

* The concept of portability applies in different ways according to specific type of virtualisation considered.

1. In the case of a hardware virtualisation solution, the guest is packed into a virtual image that, in most cases, can be safely moved and executed on type of different virtual machines.
2. In case of programming-level virtualisation, as implemented by JVM or .NET runtime, the binary code representing application components (jars or assemblies) can run without any recompilation on any implementation of the corresponding virtual machine.

SIMPLE MESSAGE TRANSFER PROTOCOL (SMTP)

STANDARDS FOR MESSAGING

* A message is a unit of information that is moved from one place to another.
* According to the Internet Engineering Task Force (IETF), the standards message

- Simple Message Transfer Protocol (SMTP)
- Post Office Protocol (POP)
- Internet Messaging Access Protocol (IMAP)
- Syndication (Atom, Atom Publishing Protocol, Rss)
- Web Services: REST (Representational State Transfer)
- Web Services: SOAP (Simple Object Access Protocol)
- Communications (HTTP, SIMPLE, xMPP)

SIMPLE MESSAGE TRANSFER PROTOCOL

* Simple Message Transfer Protocol is arguably the most important protocol in use today for basic messaging.
* Before SMTP was created, email messages were sent using File Transfer Protocol (FTP).
* A sender would compose a message and transmit it to the recipient as if it were a file.
* While this process worked, it had its shortcomings.
* The FTP protocol was designed to transmit files, not messages, so it did not provide any means for recipients to identify the sender or for the sender to designate an intended recipient.
* If a message showed up on an FTP server, it was up to administrator to open or print it (and sometimes even deliver it) before anyone even know who it was supposed to be receiving it.

* SMTP was designed so that sender and recipient information could be transmitted with the message.
* SMTP was intially defined in 1973 by IETF RFC 561.
* It has evolved over the years and has been modified by RFCs 680, 724 and 733.
* The current RFCs applying to SMTP are RFC 821 and RFC 822.
* SMTP is a two-way protocol that usually operates using TCP (Transmission Control Protocol) port 25.
* Though many people don't realize it, SMTP can be used to both send and receive messages.
* Typically, though, workstations use POP (Post Office Protocol) rather than SMTP to receive messages.
* SMTP is usually used for either sending a message from a work station to a mail server for for communications between mail servers.

SMTP

* Most of the internet systems use SMTP as a method to transfer mail from one user to another.
* SMTP is a push protocol and is used to send the mail whereas POP or IMAP are used to retrive those mails at the receiver's side.

SMTP fundamentals:

* SMTP is an application layer protocol.
* The client who wants to send mail opens a TCP connection to the SMTP server and then sends mail across the connection.
* The SMTP server is always on listening mode. As soon as it listens for a TCP connection from any client, the SMTP process initialises a connection on that port.
* After successfully establishing TCP connection the client process sends the mail instantly.

SMTP protocol:

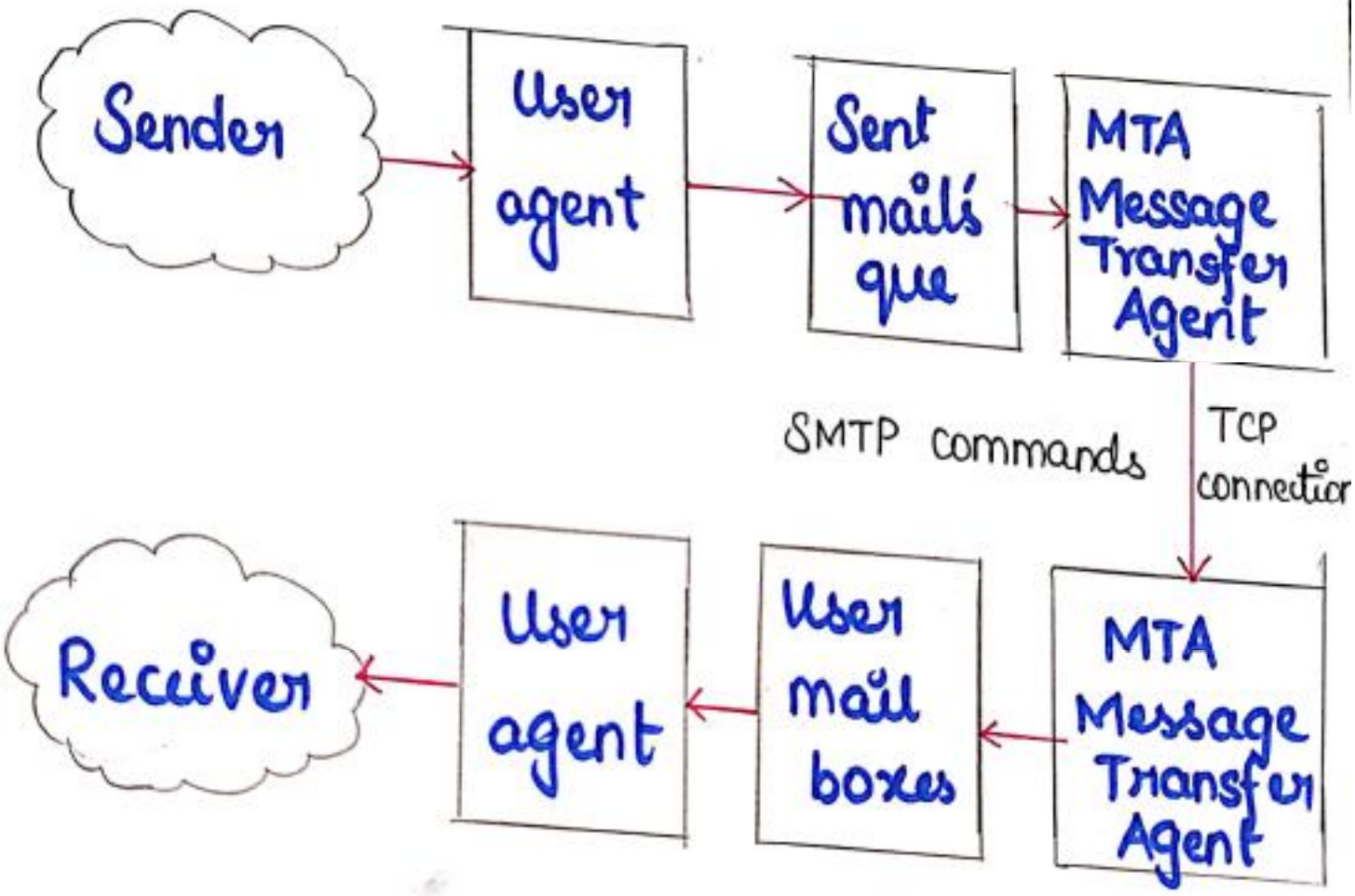
1) End-to-End
2) Store and forward.

* The end-to-end model is used to communicate between different organization whereas the store and forward method are used within an organisation.
* A SMTP client who wants to send the mail will contact the destination's host SMTP directly in order to the send the mail destination.
* The SMTP server will keep the mail to itself until it is successfully copied to the receiver's SMTP.

Model of SMTP:

* In the SMTP model, user deals with the user agent for ex: Microsoft Outlook, Netscape, Mozilla, etc.
* In order to exchange the mail using TCP, MTA is used.
* The users sending the mail do not have to deal with the MTA it is the responsibility of the system admin to set up the local MTA.
* The MTA maintains a small queue of mail so that it can schedule repeat delivery of mail in case the receiver is not available.

* The MTA delivers the mail to the mailboxes and the information can later be downloaded by the user agents.



UNIT – V

SECURITY,STANDARDS AND APPLICATIONS

STANDARDS FOR APPLICATION DEVELOPERS

The purpose of application development standards is to ensure uniform, consistent, high-quality software solutions. Programming standards are important to programmers for a variety of reasons. Some researchers have stated that, as a general rule, 80% of the lifetime cost of a piece of software goes to maintenance. Furthermore, hardly any software is maintained by the original author for its complete life cycle. Programming standards help to improve the readability of the software, allowing developers to understand new code more quickly and thoroughly. If you ship source code as a product, it is important to ensure that it is as well packaged and meets industry standards comparable to the products you compete with. For the standards to work, everyone developing solutions must

conform to them. In the following sections, we discuss application standards that are commonly used across the Internet in browsers, for transferring data, sending messages, and securing data.

Browsers (Ajax)

Ajax, or its predecessor AJAX (Asynchronous JavaScript and XML), is a group of interrelated web development techniques used to create interactive web applications or rich Internet applications. Using Ajax, web applications can retrieve data from the server asynchronously, without interfering with the display and behavior of the browser page currently being displayed to the user. The use of Ajax has led to an increase in interactive animation on web pages. Despite its name, JavaScript and XML are not actually *required* for Ajax. Moreover, requests do not even need to be asynchronous. The original acronym AJAX has changed to the name Ajax to reflect the fact that these specific technologies are no longer required.

In many cases, related pages that coexist on a web site share much common content. Using traditional methods, such content must be reloaded every time a request is made. Using Ajax, a web application can request only the content that needs to be updated. This greatly reduces networking bandwidth usage and page load times. Using asynchronous requests allows a client browser to appear more interactive and to respond to input more quickly. Sections of pages can be reloaded individually. Users generally perceive the application to be faster and more responsive. Ajax can reduce connections to the server, since scripts and style sheets need only be requested once.

An Ajax framework helps developers create web applications that use Ajax. The framework helps them to build dynamic web pages on the client side. Data is sent to or from the server using requests, usually written in JavaScript. On the server, some processing may be required to handle these requests, for example, when finding and storing data. This is accomplished more easily with the use of a framework dedicated to process Ajax requests. One such framework, ICEfaces, is an open source Java product maintained by http://icefaces.org.

ICEfaces Ajax Application Framework

ICEfaces is an integrated Ajax application framework that enables Java EE application developers to easily create and deploy thin-client rich Internet applications in pure Java. ICEfaces is a fully featured product that enterprise

developers can use to develop new or existing Java EE applications at no cost. ICEfaces is the most successful enterprise Ajax framework available under open source. The ICEfaces developer community is extremely vibrant, already exceeding 32,000 developers in 36 countries. To run ICEfaces applications, users need to download and install the following products:

Java 2 Platform, Standard Edition
Ant
Tomcat
ICEfaces
Web browser (if you don't already have one installed)

ICEfaces leverages the entire standards-based Java EE set of tools and environments. Rich enterprise application features are developed in pure Java in a thin-client model. No Applets or proprietary browser plug-ins are required. ICEfaces applications are JavaServer Faces (JSF) applications, so Java EE application development skills apply directly and Java developers don't have to do any JavaScript-related development.

Because ICEfaces is a pure Java enterprise solution, developers can continue to work the way they normally do. They are able to leverage their existing Java integrated development environments (IDEs) and test tools for development. ICEfaces supports an array of Java Application Servers, IDEs, third-party components, and JavaScript effect libraries. ICEfaces pioneered a technique called Ajax Push. This technique enables server/application-initiated content rendering to be sent to the browser. Also, ICEfaces is the one of the most secure Ajax solutions available. Compatible with SSL (Secure Sockets Layer) protocol, it prevents cross-site scripting, malicious code injection, and unauthorized data mining. ICEfaces does not expose application logic or user data, and it is effective in preventing fake form submits and SQL (Structured Query Language) injection attacks. ICEfaces also supports third-party application server Asynchronous Request Processing (ARP) APIs provided by Sun Glassfish (Grizzly), Jetty, Apache Tomcat, and others.

Data (XML, JSON)

Extensible Markup Language (XML) is a specification for creating custom markup languages. It is classified as an extensible language because it allows

the user to define markup elements. Its purpose is to enable sharing of structured data. XML is often used to describe structured data and to serialize objects. Various XML-based protocols exist to represent data structures for data interchange purposes. Using XML is arguably more complex than using JSON (described below), which represents data structures in simple text formatted specifically for data interchange in an uncompressed form. Both XML and JSON lack mechanisms for representing large binary data types such as images.

XML, in combination with other standards, makes it possible to define the content of a document separately from its formatting. The benefit here is the ability to reuse that content in other applications or for other presentation environments. Most important, XML provides a basic syntax that can be used to share information among different kinds of computers, different applications, and different organizations without needing to be converted from one to another.

An XML document has two correctness levels, *well formed* and *valid*. A well-formed document conforms to the XML syntax rules. A document that is not well formed is not in XML format, and a conforming parser will not process it. A valid document is well formed and additionally conforms to semantic rules which can be user-defined or exist in an XML schema. An XML schema is a description of a type of XML document, typically expressed in terms of constraints on the structure and content of documents of that type, above and beyond the basic constraints imposed by XML itself. A number of standard and proprietary XML schema languages have emerged for the purpose of formally expressing such schemas, and some of these languages are themselves XML-based.

XML documents must conform to a variety of rules and naming conventions. By carefully choosing the names of XML elements, it is possible to convey the meaning of the data in the markup itself. This increases human readability while retaining the syntactic structure needed for parsing. However, this can lead to verbosity, which complicates authoring and increases file size. When creating XML, the designers decided that by leaving the names, allowable hierarchy, and meanings of the elements and attributes open and definable by a customized schema, XML could provide a syntactic foundation for the creation of purpose-specific, XML-based markup languages. The general syntax of such languages is very rigid. Documents must adhere to the general rules of XML, ensuring that all XML-aware software can at least read and understand the arrangement of information within

them. The schema merely supplements the syntax rules with a predefined set of constraints.

Before the advent of generalized data description languages such as XML, software designers had to define special file formats or small languages to share data between programs. This required writing detailed specifications and special-purpose parsers and writers. XML's regular structure and strict parsing rules allow software designers to leave the task of parsing to standard tools, since XML provides a general, data model-oriented framework for the development of application-specific languages. This allows software designers to concentrate on the development of rules for their data at relatively high levels of abstraction.

JavaScript Object Notation (JSON)

JSON is a lightweight computer data interchange format. It is a text-based, human-readable format for representing simple data structures and associative arrays (called objects). The JSON format is specified in Internet Engineering Task Force Request for Comment (RFC) 4627. The JSON format is often used for transmitting structured data over a network connection in a process called serialization. Its main application is in Ajax web application programming, where it serves as an alternative to the XML format. JSON is based on a subset of the JavaScript programming language. It is considered to be a language-independent data format. Code for parsing and generating JSON data is readily available for a large variety of programming languages. The json.org website provides a comprehensive listing of existing JSON bindings, organized by language.

Even though JSON was intended as a data serialization format, its design as a subset of the JavaScript language poses security concerns. The use of a JavaScript interpreter to dynamically execute JSON text as JavaScript can expose a program to bad or even malicious script. JSON is also subject to cross-site request forgery attacks. This can allow JSON-encoded data to be evaluated in the context of a malicious page, possibly divulging passwords or other sensitive data. This is only a problem if the server depends on the browser's Same Origin Policy to block the delivery of the data in the case of an improper request. When the server determines the propriety of the request, there is no problem because it will only output data if the request is valid. Cookies are not adequate for determining whether a request is authorized and valid. The use of cookies is subject to cross-site request forgery and should be avoided with JSON. As you can see, JSON

was built for simple tasks and can be useful, but there is some risk involved in using it—especially given the alternative solutions available today.

Solution Stacks (LAMP and LAPP) LAMP

LAMP is a popular open source solution commonly used to run dynamic web sites and servers. The acronym derives from the fact that it includes **L**inux, **A**pache, **M**ySQL, and **P**HP (or Perl or Python) and is considered by many to be the platform of choice for development and deployment of high-performance web applications which require a solid and reliable foundation. The combination of these technologies is used primarily to define a web server infrastructure or for creating a programming environment for developing software. While the creators of these open source products did not intend for them all to work with each other, the LAMP combination has become popular because of its open source nature, low cost, and the wide distribution of its components (most of which come bundled with nearly all of the current Linux distributions). When used in combination, they represent a solution stack of technologies that support application servers.

Linux, Apache, PostgreSQL, and PHP(or Perl or Python)

The LAPP stack is an open source web platform that can be used to run dynamic web sites and servers. It is considered by many to be a more powerful alternative to the more popular LAMP stack. These advanced and mature components provide a rock-solid foundation for the development and deployment of high-performance web applications. LAPP offers SSL, PHP, Python, and Perl support for Apache2 and PostgreSQL. There is an administration front-end for PostgreSQL as well as web-based administration modules for configuring Apache2 and PHP. PostgreSQL password encryption is enabled by default. The PostgreSQL user is trusted when connecting over local Unix sockets. Many consider the LAPP stack a more secure out-of-the-box solution than the LAMP stack. The choice of which stack to use is made by developers based on the purpose of their application and the risks they may have to contend with when users begin working with the product.

STANDARDS FOR MESSAGING

You probably think you know what a messaging standard is. Unfortunately, the term *messaging* means different things to different people. So does the word *standard*. People may assume you are talking about networking when you begin discussing messaging standards. The term *messaging*, however, covers a lot of ground, and not all of it is specific to networking. For our purposes here, a *message* is a unit of information that is moved from one place to another. The term *standard* also is not always clearly defined. Different entities have differing interpretations of what a standard is, and we know there are open international standards, *de facto* standards, and proprietary standards. A true standard is usually characterized by certain traits, such as being managed by an international standards body or an industry consortium, and the standard is created jointly by a community of interested parties. The Internet Engineering Task Force (IETF) is perhaps the most open standards body on the planet, because it is open to everyone. Participants can contribute, and their work is available online for free. In the following sections, we discuss the most common messaging standards used in the cloud—some of which have been used so much so that they are considered *de facto* standards.

Simple Message Transfer Protocol (SMTP)

Simple Message Transfer Protocol is arguably the most important protocol in use today for basic messaging. Before SMTP was created, email messages were sent using File Transfer Protocol (FTP). A sender would compose a message and transmit it to the recipient as if it were a file. While this process worked, it had its shortcomings. The FTP protocol was designed to transmit files, not messages, so it did not provide any means for recipients to identify the sender or for the sender to designate an intended recipient. If a message showed up on an FTP server, it was up to the administrator to open or print it (and sometimes even deliver it) before anyone even knew who it was supposed to be receiving it.

SMTP was designed so that sender and recipient information could be transmitted with the message. The design process didn't happen overnight, though. SMTP was initially defined in 1973 by IETF RFC 561. It has evolved over the years and has been modified by RFCs 680, 724 and 733. The current RFCs applying to SMTP are RFC 821 and RFC 822. SMTP is a two-way protocol that usually operates using TCP (Transmission Control Protocol) port 25. Though many people don't realize it, SMTP can be used

to both send and receive messages. Typically, though, workstations use POP (Post Office Protocol) rather than SMTP to receive messages. SMTP is usually used for either sending a message from a workstation to a mail server or for communications between mail servers.

Post Office Protocol (POP)

SMTP can be used both to send and receive messages, but using SMTP for this purpose is often impractical or impossible because a client must have a constant connection to the host to receive SMTP messages. The Post Office Protocol (POP) was introduced to circumvent this situation. POP is a lightweight protocol whose single purpose is to download messages from a server. This allows a server to store messages until a client connects and requests them. Once the client connects, POP servers begin to download the messages and subsequently delete them from the server (a default setting) in order to make room for more messages. Users respond to a message that was downloaded using SMTP. The POP protocol is defined by RFC 1939 and usually functions on TCP port 110.

Internet Messaging Access Protocol (IMAP)

Once mail messages are downloaded with POP, they are automatically deleted from the server when the download process has finished. Thus POP users have to save their messages locally, which can present backup challenges when it is important to store or save messages. Many businesses have compulsory compliance guidelines that require saving messages. It also becomes a problem if users move from computer to computer or use mobile networking, since their messages do not automatically move where they go. To get around these problems, a standard called Internet Messaging Access Protocol was created. IMAP allows messages to be kept on the server but viewed and manipulated (usually via a browser) as though they were stored locally. IMAP is a part of the RFC 2060 specification, and functions over TCP port 143.

Syndication (Atom, Atom Publishing Protocol, and RSS)

Content syndication provides citizens convenient access to new content and headlines from government via RSS (Really Simple Syndication) and other online syndication standards. Governments are providing access to more and more information online. Sharing headlines

headlines and content through syndication standards such as RSS (the little orange [XML] button, ATOM, and others) essentially allows a government to control a small window of content across web sites that choose to display the government's headlines. Headlines may also be aggregated and displayed through "newsreaders" by citizens through standalone applications or as part of their personal web page.

Portals can automatically aggregate and combine headlines and/or lengthier content from across multiple agency web sites. This allows the value of distributed effort to be shared, which is more sustainable. Press releases may be aggregated automatically from different systems, as long as they all are required to offer an RSS feed with content tagged with similar metadata. Broader use of government information online, particularly time-sensitive democratic information, justifies the effort of production and the accountability of those tasked to make it available.

Benefits: Ability to scan headlines from many sources, all in one place, through a newsreader. Time-saving awareness of new content from government, if the RSS feed or feeds are designed properly. Ability to monitor new content from across the council, as well as display feeds on their own web site. Awareness of new content position councilors as guides to government for citizens. Ability to aggregate new content or headlines from across multiple office locations and agencies. This allows a display of "joined-up" government despite structural realities. Journalists and other locally focused web sites will be among the primary feed users.

Limitations: Dissemination via syndication is a new concept to governments just getting used to the idea of remote online public access to information. Governments need to accept that while they control the content of the feed, the actual display of the headlines and content will vary. Popular RSS feeds can use significant amounts of bandwidth. Details on how often or when a feed is usually updated should be offered to those grabbing the code behind the orange [XML] button, so they "ping" it once a day instead of every hour. Automated syndication requires use of a content management system. Most viable content management systems have integrated RSS functions, but the sophistication, ease of use, and documentation of these tools vary. There are three variants of RSS, as well as the emerging ATOM standard. It

is recommended that a site pick the standard most applicable to their content rather than confuse users with different feeds providing the same content.

RSS

RSS is a family of web feed formats used to publish frequently updated works—such as blog entries, news headlines, audio, and video—in a standardized format. An RSS document includes full or summarized text, plus metadata such as publishing dates and authorship. Web feeds benefit publishers by letting them syndicate content automatically. They benefit readers who want to subscribe to timely updates from favored web sites or to aggregate feeds from many sites into one place. RSS feeds can be read using software called a reader that can be web-based, desktop-based, a mobile device, or any computerized Internet-connected device. A standardized XML file format allows the information to be published once and viewed by many different programs. The user subscribes to a feed by entering the feed's URI (often referred to informally as a URL, although technically, those two terms are not exactly synonymous) into the reader or by clicking an RSS icon in a browser that initiates the subscription process. The RSS reader checks the user's subscribed feeds regularly for new work, downloads any updates that it finds, and provides a user interface to monitor and read the feeds.

Atom and Atom Publishing Protocol (APP)

The name Atom applies to a pair of related standards. The Atom Syndication Format is an XML language used for web feeds, while the Atom Publishing Protocol (AtomPub or APP) is a simple HTTP-based protocol (HTTP is described later in this chapter) for creating and updating web resources, sometimes known as web feeds. Web feeds allow software programs to check for updates published on a web site. To provide a web feed, a site owner may use specialized software (such as a content management system) that publishes a list (or "feed") of recent articles or content in a standardized, machine-readable format. The feed can then be downloaded by web sites that syndicate content from the feed, or by feed reader programs that allow Internet users to subscribe to feeds and view their content. A feed contains entries, which may be headlines, full-text articles, excerpts, summaries, and/or links to content on a web site, along with various metadata.

The Atom format was developed as an alternative to RSS. Ben Trott, an advocate of the new format that became Atom, believed that RSS had

limitations and flaws—such as lack of ongoing innovation and its necessity to remain backward compatible—and that there were advantages to a fresh design. Proponents of the new format formed the IETF Atom Publishing Format and Protocol Workgroup. The Atom syndication format was published as an IETF "proposed standard" in RFC 4287, and the Atom Publishing Protocol was published as RFC 5023.

Web feeds are used by the weblog community to share the latest entries' headlines or their full text, and even attached multimedia files. These providers allow other web sites to incorporate the weblog's "syndicated" headline or headline-and-short-summary feeds under various usage agreements. Atom and other web syndication formats are now used for many purposes, including journalism, marketing, "bug" reports, or any other activity involving periodic updates or publications. Atom also provides a standardized way to export an entire blog, or parts of it, for backup or for importing into other blogging systems.

A program known as a feed reader or aggregator can check web pages on behalf of a user and display any updated articles that it finds. It is common to find web feeds on major web sites, as well as on many smaller ones. Some web sites let people choose between RSS- or Atom-formatted web feeds; others offer only RSS or only Atom. In particular, many blog and wiki sites offer their web feeds in the Atom format.

Client-side readers and aggregators may be designed as standalone programs or as extensions to existing programs such as web browsers. Browsers are moving toward integrated feed reader functions. Such programs are available for various operating systems. Web-based feed readers and news aggregators require no software installation and make the user's feeds available on any computer with web access. Some aggregators syndicate web feeds into new feeds, e.g., taking all football-related items from several sports feeds and providing a new football feed. There are several search engines which provide search functionality over content published via these web feeds.

Web Services (REST)

REpresentational State Transfer (REST) is a style of software architecture for distributed hypermedia systems such as the World Wide Web. As such, it is not strictly a method for building "web services." The terms "representational state transfer" and "REST" were introduced in 2000 in the doctoral

dissertation of Roy Fielding,[3] one of the principal authors of the Hypertext Transfer Protocol (HTTP) specification.

REST refers to a collection of network architecture principles which outline how resources are defined and addressed. The term is often used in a looser sense to describe any simple interface which transmits domain-specific data over HTTP without an additional messaging layer such as SOAP or session tracking via HTTP cookies. These two meanings can conflict as well as overlap. It is possible to design a software system in accordance with Fielding's REST architectural style without using HTTP and without interacting with the World Wide Web.[4] It is also possible to design simple XML+HTTP interfaces which do not conform to REST principles, but instead follow a model of remote procedure call. Systems which follow Fielding's REST principles are often referred to as "RESTful."

Proponents of REST argue that the web's scalability and growth are a direct result of a few key design principles. Application state and functionality are abstracted into resources. Every resource is uniquely addressable using a universal syntax for use in hypermedia links, and all resources share a uniform interface for the transfer of state between client and resource. This transfer state consists of a constrained set of well-defined operations and a constrained set of content types, optionally supporting code on demand. State transfer uses a protocol which is client-server based, stateless and cacheable, and layered. Fielding describes REST's effect on scalability thus:

> REST's client-server separation of concerns simplifies component implementation, reduces the complexity of connector semantics, improves the effectiveness of performance tuning, and increases the scalability of pure server components. Layered system constraints allow intermediaries—proxies, gateways, and firewalls—to be introduced at various points in the communication without changing the interfaces between components, thus allowing them to assist in communication translation or improve performance via large-scale, shared caching. REST enables intermediate processing by constraining messages to be self-descriptive: interaction is stateless between requests, standard methods and media types are used to indicate

3. Roy T. Fielding, "Architectural Styles and the Design of Network-Based Software Architectures," dissertation, University of California, Irvine, 2000, http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm.
4. Ibid.

> semantics and exchange information, and responses explicitly indicate cacheability.[5]

An important concept in REST is the existence of resources, each of which is referenced with a global identifier (e.g., a URI in HTTP). In order to manipulate these resources, components of the network (user agents and origin servers) communicate via a standardized interface (e.g., HTTP) and exchange representations of these resources (the actual documents conveying the information). For example, a resource which is a circle may accept and return a representation which specifies a center point and radius, formatted in SVG (Scalable Vector Graphics), but may also accept and return a representation which specifies any three distinct points along the curve as a comma-separated list.

Any number of connectors (clients, servers, caches, tunnels, etc.) can mediate the request, but each does so without "seeing past" its own request (referred to as "layering," another constraint of REST and a common principle in many other parts of information and networking architecture). Thus an application can interact with a resource by knowing two things: the identifier of the resource, and the action required—it does not need to know whether there are caches, proxies, gateways, firewalls, tunnels, or anything else between it and the server actually holding the information. The application does, however, need to understand the format of the information (representation) returned, which is typically an HTML, XML, or JSON document of some kind, although it may be an image, plain text, or any other content.

REST provides improved response time and reduced server load due to its support for the caching of representations. REST improves server scalability by reducing the need to maintain session state. This means that different servers can be used to handle different requests in a session. REST requires less client-side software to be written than other approaches, because a single browser can access any application and any resource. REST depends less on vendor software and mechanisms which layer additional messaging frameworks on top of HTTP. It provides equivalent functionality when compared to alternative approaches to communication, and it does not require a separate resource discovery mechanism, because of the use of hyperlinks in representations. REST

5. Ibid.

also provides better long-term compatibility because of the capability of document types such as HTML to evolve without breaking backwards or forwards compatibility and the ability of resources to add support for new content types as they are defined without dropping or reducing support for older content types.

One benefit that should be obvious with regard to web-based applications is that a RESTful implementation allows a user to bookmark specific "queries" (or requests) and allows those to be conveyed to others across email, instant messages, or to be injected into wikis, etc. Thus this "representation" of a path or entry point into an application state becomes highly portable. A RESTFul web service is a simple web service implemented using HTTP and the principles of REST. Such a web service can be thought of as a collection of resources comprising three aspects:

1. The URI for the web service
2. The MIME type of the data supported by the web service (often JSON, XML, or YAML, but can be anything)
3. The set of operations supported by the web service using HTTP methods, including but not limited to POST, GET, PUT, and DELETE

Members of the collection are addressed by ID using URIs of the form <baseURI>/<ID>. The ID can be any unique identifier. For example, a RESTFul web service representing a collection of cars for sale might have the URI:

```
http://example.com/resources/cars
```

If the service uses the car registration number as the ID, then a particular car might be present in the collection as

```
http://example.com/resources/cars/yxz123
```

SOAP

SOAP, originally defined as Simple Object Access Protocol, is a protocol specification for exchanging structured information in the implementation

of Web Services in computer networks. It relies on XML as its message format and usually relies on other application-layer protocols, most notably Remote Procedure Call (RPC) and HTTP for message negotiation and transmission. SOAP can form the foundation layer of a web services protocol stack, providing a basic messaging framework on which web services can be built.

As a simple example of how SOAP procedures can be used, a SOAP message can be sent to a web service-enabled web site—for example, a house price database—with the parameters needed for a search. The site returns an XML-formatted document with the resulting data (prices, location, features, etc). Because the data is returned in a standardized machine-parseable format, it may be integrated directly into a third-party site.

The SOAP architecture consists of several layers of specifications for message format, message exchange patterns (MEPs), underlying transport protocol bindings, message processing models, and protocol extensibility. SOAP is the successor of XML-RPC. SOAP makes use of an Internet application-layer protocol as a transport protocol. Critics have argued that this is an abuse of such protocols, as it is not their intended purpose and therefore not a role they fulfill well. Proponents of SOAP have drawn analogies to successful uses of protocols at various levels for tunneling other protocols.

Both SMTP and HTTP are valid application-layer protocols used as transport for SOAP, but HTTP has gained wider acceptance because it works well with today's Internet infrastructure; specifically, HTTP works well with network firewalls. SOAP may also be used over HTTPS (which is the same protocol as HTTP at the application level, but uses an encrypted transport protocol underneath) with either simple or mutual authentication; this is the advocated WS-I method to provide web service security as stated in the WS-I Basic Profile 1.1. This is a major advantage over other distributed protocols such as GIOP/IIOP or DCOM, which are normally filtered by firewalls. XML was chosen as the standard message format because of its widespread use by major corporations and open source development efforts. Additionally, a wide variety of freely available tools significantly eases the transition to a SOAP-based implementation.

Advantages of using SOAP over HTTP are that SOAP allows for easier communication through proxies and firewalls than previous remote execution technology. SOAP is versatile enough to allow for the use of different transport protocols. The standard stacks use HTTP as a transport protocol,

but other protocols are also usable (e.g., SMTP). SOAP is platform-independent, language-independent, and it is simple and extensible.

Because of the verbose XML format, SOAP can be considerably slower than competing middleware technologies such as CORBA (Common Object Request Broker Architecture). This may not be an issue when only small messages are sent. To improve performance for the special case of XML with embedded binary objects, Message Transmission Optimization Mechanism was introduced. When relying on HTTP as a transport protocol and not using WS-Addressing or an ESB, the roles of the interacting parties are fixed. Only one party (the client) can use the services of the other. Developers must use polling instead of notification in these common cases.

Most uses of HTTP as a transport protocol are made in ignorance of how the operation is accomplished. As a result, there is no way to know whether the method used is appropriate to the operation. The REST architecture has become a web service alternative that makes appropriate use of HTTP's defined methods.

Communications (HTTP, SIMPLE, and XMPP)

HTTP is a request/response communications standard based on a client/server model. A client is the end user, the server is the web site. The client making a HTTP request via a web browser or other tool sends the request to the server. The responding server is called the origin server. HTTP is not constrained to use TCP/IP and its supporting layers, although this is its most popular application on the Internet. SIMPLE, the Session Initiation Protocol for Instant Messaging and Presence Leveraging Extensions, is an instant messaging (IM) and presence protocol suite based on Session Initiation Protocol, and it is managed by the IETF. Like XMPP, SIMPLE is an open standard. Extensible Messaging and Presence Protocol (XMPP) is also an open, XML-based protocol originally aimed at near-real-time, extensible instant messaging and presence information (e.g., buddy lists) but now expanded into the broader realm of message-oriented middleware. All of these protocols are discussed in detail in the following paragraphs.

Hypertext Transfer Protocol (HTTP)

HTTP is an application-level protocol for distributed, collaborative, hypermedia information systems. Its use for retrieving linked resources led to the establishment of the World Wide Web. HTTP development was

coordinated by the World Wide Web Consortium and the Internet Engineering Task Force, culminating in the publication of a series of Requests for Comments, most notably RFC 2616 (June 1999), which defines HTTP/1.1, the version of HTTP in common use today.

HTTP is a request/response standard between a client and a server. A client is the end-user, the server is the web site. The client making a HTTP request—using a web browser, spider, or other end-user tool—is referred to as the user agent. The responding server—which stores or creates resources such as HTML files and images—is called the origin server. In between the user agent and origin server may be several intermediaries, such as proxies, gateways, and tunnels. HTTP is not constrained to using TCP/IP and its supporting layers, although this is its most popular application on the Internet. In fact, HTTP can be implemented on top of any other protocol; all it requires is reliable transport, so any protocol, on the Internet or any other network, that provides reliable transport can be used.

Typically, an HTTP client initiates a request. It establishes a TCP connection to a particular port on a host (port 80 by default). An HTTP server listening on that port waits for the client to send a request message. Upon receiving the request, the server sends back a status line such as "HTTP/1.1 200 OK" and a message of its own, the body of which is perhaps the requested resource, an error message, or some other information. Resources to be accessed by HTTP are identified using Uniform Resource Identifiers (URIs or, more specifically, Uniform Resource Locators, URLs) using the http: or https URI schemes.

SIMPLE

Session Initiation Protocol for Instant Messaging and Presence Leveraging Extensions (SIMPLE) is an instant messaging (IM) and presence protocol suite based on the Session Initiation Protocol (SIP). Like XMPP, SIMPLE is an open standard. SIMPLE makes use of SIP for registering for presence information and receiving notifications when presence-related events occur. It is also used for sending short messages and managing a session of real-time messages between two or more participants. Implementations of the SIMPLE-based protocols can be found in SIP softphones and also hardphones.[6] The SIMPLE presence specifications can be broken up into core

6. In computing, a softphone is a software program for making telephone calls over the Internet using a general-purpose computer; a hardphone is a conventional telephone set.

protocol methods, presence information, and the handling of privacy, policy. and provisioning.

The core protocol methods provide SIP extensions for subscriptions, notifications, and publications. The methods used, **subscribe** and **notify,** are defined in RFC 3265. **Subscribe** allows a user to subscribe to an event on a server. **Notify** is the method used whenever the event arises and the server responds back to the subscriber. Another standard, RFC 3856, defines precisely how to use these methods to establish and maintain presence. Presence documents contain information encoded using XML. These documents are transported in the bodies of SIP messages.[7] Privacy, policy, and provisioning information is needed by user agents to determine who may subscribe to presence information. A framework for authorization policies controlling access to application-specific data is defined in RFC 4745 and RFC 5025. SIP defines two modes of instant messaging, the Page mode and the Session mode. Page mode makes use of the SIP method MESSAGE, as defined in RFC 3428. This mode establishes no sessions, while the Session mode based on the Message Session Relay Protocol (RFC 4975, RFC 4976) defines text-based protocol for exchanging arbitrarily sized content of any time between users.

XMPP

Extensible Messaging and Presence Protocol (XMPP) is an XML-based protocol used for near-real-time, extensible instant messaging and presence information. XMPP remains the core protocol of the Jabber Instant Messaging and Presence technology. Jabber provides a carrier-grade, best-in-class presence and messaging platform. According to a press release following its acquisition by Cisco Systems in November 2008, "Jabber's technology leverages open standards to provide a highly scalable architecture that supports the aggregation of presence information across different devices, users and applications. The technology also enables collaboration across many different presence systems such as Microsoft Office Communications Server, IBM Sametime, AOL AIM, Google and Yahoo!"

Built to be extensible, the XMPP protocol has grown to support features such as voice-over-IP and file transfer signaling. Unlike other instant messaging protocols, XMPP is an open standard. Like email, anyone who has a domain name and an Internet connection can run the Jabber server

7. RFC 3863 and RFC 4479 describe this procedure.

and chat with others. The Jabber project is open source software, available from Google at http://code.google.com/p/jabber-net.

XMPP-based software is deployed on thousands of servers across the Internet. The Internet Engineering Task Force has formalized XMPP as an approved instant messaging and presence technology under the name XMPP, and the XMPP specifications have been published as RFC 3920 and RFC 3921. Custom functionality can be built on top of XMPP, and common extensions are managed by the XMPP Software Foundation.

XMPP servers can be isolated from the public Jabber network, and robust security (via SASL and TLS) is built into the core XMPP specifications. Because the client uses HTTP, most firewalls allow users to fetch and post messages without hindrance. Thus, if the TCP port used by XMPP is blocked, a server can listen on the normal HTTP port and the traffic should pass without problems. Some web sites allow users to sign in to Jabber via their browser. Furthermore, there are open public servers, such as www.jabber80.com, which listen on standard http (port 80) and https (port 443) ports and allow connections from behind most firewalls.

STANDARDS FOR SECURITY

Security standards define the processes, procedures, and practices necessary for implementing a security program. These standards also apply to cloud-related IT activities and include specific steps that should be taken to ensure a secure environment is maintained that provides privacy and security of confidential information in a cloud environment. Security standards are based on a set of key principles intended to protect this type of trusted environment. Messaging standards, especially for security in the cloud, must also include nearly all the same considerations as any other IT security endeavor. The following protocols, while not exclusively specific to cloud security, merit coverage here. In the next few sections, we explain what they are and how they are used in the cloud environment.

7.6.1 Security (SAML OAuth, OpenID, SSL/TLS)

A basic philosophy of security is to have layers of defense, a concept known as *defense in depth*. This means having overlapping systems designed to provide security even if one system fails. An example is a firewall working in conjunction with an intrusion-detection system (IDS). Defense in depth provides security because there is no single point of failure and no single-entry vector at which an attack can occur. For this reason, a choice between

implementing network security in the middle part of a network (i.e., in the cloud) or at the endpoints is a false dichotomy.[8]

No single security system is a solution by itself, so it is far better to secure all systems. This type of layered security is precisely what we are seeing develop in cloud computing. Traditionally, security was implemented at the endpoints, where the user controlled access. An organization had no choice except to put firewalls, IDSs, and antivirus software inside its own network. Today, with the advent of managed security services offered by cloud providers, additional security can be provided inside the cloud.

Security Assertion Markup Language (SAML)

SAML is an XML-based standard for communicating authentication, authorization, and attribute information among online partners. It allows businesses to securely send assertions between partner organizations regarding the identity and entitlements of a principal. The Organization for the Advancement of Structured Information Standards (OASIS) Security Services Technical Committee is in charge of defining, enhancing, and maintaining the SAML specifications.[9] SAML is built on a number of existing standards, namely, SOAP, HTTP, and XML. SAML relies on HTTP as its communications protocol and specifies the use of SOAP (currently, version 1.1). Most SAML transactions are expressed in a standardized form of XML. SAML assertions and protocols are specified using XML schema. Both SAML 1.1 and SAML 2.0 use digital signatures (based on the XML Signature standard) for authentication and message integrity. XML encryption is supported in SAML 2.0, though SAML 1.1 does not have encryption capabilities. SAML defines XML-based assertions and protocols, bindings, and profiles. The term SAML Core refers to the general syntax and semantics of SAML assertions as well as the protocol used to request and transmit those assertions from one system entity to another. SAML protocol refers to what is transmitted, not how it is transmitted. A SAML binding determines how SAML requests and responses map to standard messaging protocols. An important (synchronous) binding is the SAML SOAP binding.

SAML standardizes queries for, and responses that contain, user authentication, entitlements, and attribute information in an XML format.

8. Bruce Schnier, http://www.schneier.com/blog/archives/2006/02/security_in_the.html, 15 Feb 2006, retrieved 21 Feb 2009.
9. The reader is encouraged to consult http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=security.

This format can then be used to request security information about a principal from a SAML authority. A SAML authority, sometimes called the asserting party, is a platform or application that can relay security information. The relying party (or assertion consumer or requesting party) is a partner site that receives the security information. The exchanged information deals with a subject's authentication status, access authorization, and attribute information. A subject is an entity in a particular domain. A person identified by an email address is a subject, as might be a printer.

SAML assertions are usually transferred from identity providers to service providers. Assertions contain statements that service providers use to make access control decisions. Three types of statements are provided by SAML: authentication statements, attribute statements, and authorization decision statements. SAML assertions contain a packet of security information in this form:

```
<saml:Assertion A...>
     <Authentication>
     ...
     </Authentication>
     <Attributes>
     ...
     </Attributes>
     <Authorization>
     ...
     </Authorization>
</saml:Assertion A>
```

The assertion shown above is interpreted as follows:

```
Assertion A, issued at time T by issuer I, regarding subject
S, provided conditions C are valid.
```

Authentication statements assert to a service provider that the principal did indeed authenticate with an identity provider at a particular time using a particular method of authentication. Other information about the authenticated principal (called the authentication context) may be disclosed in an authentication statement. An attribute statement asserts that a subject is associated with certain attributes. An attribute is simply a name–value pair. Relying parties use attributes to make access control decisions. An

authorization decision statement asserts that a subject is permitted to perform action A on resource R given evidence E. The expressiveness of authorization decision statements in SAML is intentionally limited.

A SAML protocol describes how certain SAML elements (including assertions) are packaged within SAML request and response elements. It provides processing rules that SAML entities must adhere to when using these elements. Generally, a SAML protocol is a simple request–response protocol. The most important type of SAML protocol request is a query. A service provider makes a query directly to an identity provider over a secure back channel. For this reason, query messages are typically bound to SOAP. Corresponding to the three types of statements, there are three types of SAML queries: the authentication query, the attribute query, and the authorization decision query. Of these, the attribute query is perhaps most important. The result of an attribute query is a SAML response containing an assertion, which itself contains an attribute statement.

Open Authentication (OAuth)

OAuth is an open protocol, initiated by Blaine Cook and Chris Messina, to allow secure API authorization in a simple, standardized method for various types of web applications. Cook and Messina had concluded that there were no open standards for API access delegation. The OAuth discussion group was created in April 2007, for the small group of implementers to write the draft proposal for an open protocol. DeWitt Clinton of Google learned of the OAuth project and expressed interest in supporting the effort. In July 2007 the team drafted an initial specification, and it was released in October of the same year.

OAuth is a method for publishing and interacting with protected data. For developers, OAuth provides users access to their data while protecting account credentials. OAuth allows users to grant access to their information, which is shared by the service provider and consumers without sharing all of their identity. The Core designation is used to stress that this is the baseline, and other extensions and protocols can build on it.

By design, OAuth Core 1.0 does not provide many desired features (e.g., automated discovery of endpoints, language support, support for XML-RPC and SOAP, standard definition of resource access, OpenID integration, signing algorithms, etc.). This intentional lack of feature support is viewed by the authors as a significant benefit. The Core deals with fundamental aspects of the protocol, namely, to establish a mechanism for

exchanging a user name and password for a token with defined rights and to provide tools to protect the token. It is important to understand that security and privacy are not guaranteed by the protocol. In fact, OAuth by itself *provides no privacy at all* and depends on other protocols such as SSL to accomplish that. OAuth can be implemented in a secure manner, however. In fact, the specification includes substantial security considerations that must be taken into account when working with sensitive data. With Oauth, sites use tokens coupled with shared secrets to access resources. Secrets, just like passwords, must be protected.

OpenID

OpenID is an open, decentralized standard for user authentication and access control that allows users to log onto many services using the same digital identity. It is a single-sign-on (SSO) method of access control. As such, it replaces the common log-in process (i.e., a log-in name and a password) by allowing users to log in once and gain access to resources across participating systems.

The original OpenID authentication protocol was developed in May 2005 by Brad Fitzpatrick, creator of the popular community web site LiveJournal. In late June 2005, discussions began between OpenID developers and other developers from an enterprise software company named NetMesh. These discussions led to further collaboration on interoperability between OpenID and NetMesh's similar Light-Weight Identity (LID) protocol. The direct result of the collaboration was the Yadis discovery protocol, which was announced on October 24, 2005.

The Yadis specification provides a general-purpose identifier for a person and any other entity, which can be used with a variety of services. It provides a syntax for a resource description document identifying services available using that identifier and an interpretation of the elements of that document. Yadis discovery protocol is used for obtaining a resource description document, given that identifier. Together these enable coexistence and interoperability of a rich variety of services using a single identifier. The identifier uses a standard syntax and a well-established namespace and requires no additional namespace administration infrastructure.

An OpenID is in the form of a unique URL and is authenticated by the entity hosting the OpenID URL. The OpenID protocol does not rely on a central authority to authenticate a user's identity. Neither the OpenID protocol nor any web sites requiring identification can mandate that a specific

type of authentication be used; nonstandard forms of authentication such as smart cards, biometrics, or ordinary passwords are allowed. A typical scenario for using OpenID might be something like this: A user visits a web site that displays an OpenID log-in form somewhere on the page. Unlike a typical log-in form, which has fields for user name and password, the OpenID log-in form has only one field for the OpenID identifier (which is an OpenID URL). This form is connected to an implementation of an OpenID client library. A user will have previously registered an OpenID identifier with an OpenID identity provider. The user types this OpenID identifier into the OpenID log-in form.

The relying party then requests the web page located at that URL and reads an HTML link tag to discover the identity provider service URL. With OpenID 2.0, the client discovers the identity provider service URL by requesting the XRDS document (also called the Yadis document) with the content type **application/xrds+xml,** which may be available at the target URL but is always available for a target XRI. There are two modes by which the relying party can communicate with the identity provider: **checkid_immediate** and **checkid_setup.** In **checkid_immediate,** the relying party requests that the provider not interact with the user. All communication is relayed through the user's browser without explicitly notifying the user. In **checkid_setup,** the user communicates with the provider server directly using the same web browser as is used to access the relying party site. The second option is more popular on the web.

To start a session, the relying party and the identity provider establish a shared secret—referenced by an associate handle—which the relying party then stores. Using **checkid_setup,** the relying party redirects the user's web browser to the identity provider so that the user can authenticate with the provider. The method of authentication varies, but typically, an OpenID identity provider prompts the user for a password, then asks whether the user trusts the relying party web site to receive his or her credentials and identity details. If the user declines the identity provider's request to trust the relying party web site, the browser is redirected to the relying party with a message indicating that authentication was rejected. The site in turn refuses to authenticate the user. If the user accepts the identity provider's request to trust the relying party web site, the browser is redirected to the designated return page on the relying party web site along with the user's credentials. That relying party must then confirm that the credentials really came from the identity provider. If they had previously established a shared

secret, the relying party can validate the shared secret received with the credentials against the one previously stored. In this case, the relying party is considered to be stateful, because it stores the shared secret between sessions (a process sometimes referred to as persistence). In comparison, a stateless relying party must make background requests using the **check_authentication** method to be sure that the data came from the identity provider.

After the OpenID identifier has been verified, OpenID authentication is considered successful and the user is considered logged in to the relying party web site. The web site typically then stores the OpenID identifier in the user's session. OpenID does not provide its own authentication methods, but if an identity provider uses strong authentication, OpenID can be used for secure transactions.

SSL/TLS

Transport Layer Security (TLS) and its predecessor, Secure Sockets Layer (SSL), are cryptographically secure protocols designed to provide security and data integrity for communications over TCP/IP. TLS and SSL encrypt the segments of network connections at the transport layer. Several versions of the protocols are in general use in web browsers, email, instant messaging, and voice-over-IP. TLS is an IETF standard protocol which was last updated in RFC 5246.

The TLS protocol allows client/server applications to communicate across a network in a way specifically designed to prevent eavesdropping, tampering, and message forgery. TLS provides endpoint authentication and data confidentiality by using cryptography. TLS authentication is one-way—the server is authenticated, because the client already knows the server's identity. In this case, the client remains unauthenticated. At the browser level, this means that the browser has validated the server's certificate—more specifically, it has checked the digital signatures of the server certificate's issuing chain of Certification Authorities (CAs).

Validation does not identify the server to the end user. For true identification, the end user must verify the identification information contained in the server's certificate (and, indeed, its whole issuing CA chain). This is the only way for the end user to know the "identity" of the server, and this is the only way identity can be securely established, verifying that the URL, name, or address that is being used is specified in the server's certificate. Malicious web sites cannot use the valid certificate of another web site because they

have no means to encrypt the transmission in a way that it can be decrypted with the valid certificate. Since only a trusted CA can embed a URL in the certificate, this ensures that checking the apparent URL with the URL specified in the certificate is an acceptable way of identifying the site.

TLS also supports a more secure bilateral connection mode whereby both ends of the connection can be assured that they are communicating with whom they believe they are connected. This is known as mutual (assured) authentication. Mutual authentication requires the TLS client-side to also maintain a certificate. TLS involves three basic phases:

1. Peer negotiation for algorithm support
2. Key exchange and authentication
3. Symmetric cipher encryption and message authentication

During the first phase, the client and server negotiate cipher suites, which determine which ciphers are used; makes a decision on the key exchange and authentication algorithms to be used; and determines the message authentication codes. The key exchange and authentication algorithms are typically public key algorithms. The message authentication codes are made up from cryptographic hash functions. Once these decisions are made, data transfer may begin.

ZIMBRA

* It is a messaging server with an innovative browser based email and calendar application.
* Alternative to Microsoft Outlook email.

Zimbra Goal:

* To provide a better messaging experience for the administrators and end users.
* To become the leader in the next generation messaging and colloboration.

Special Features:

* E-mail, address book, calenders
* Contacts, documents.
* Yahoo Maps.
* Phone @ Skype
* RSS Feeds (Really Simple Syndication (RSS) is an XML based format for content distribution).

Google Maps:

Zimbra automatically recognizes patterns that conform to addresses then makes a web services call to the Yahoo Maps.

Phone and Skype :

* You can program Zimbra to communicate with third party info systems.
* Recognizes your proprietary data such as shipping number, invoices and purchase orders.
* To call the sender directly just clicks on the phone number in your e-mail.
* Zimbra uses the installed Skype applicat -ion to initiate a domestic or international cal

RSS Feeds :

* Allows you to subscribe to RSS feeds and sends them to your inbox as e-mail type documents.
* Because they are e-mails, sharing informa -tion with friends or colleagues is easy !
* When a new e-mail is sent each time the weblog / feed is updated.

Calendar @ Inbox :

* Zimbra connects calender with e-mail. You do not have to switch between your calendar and Inbox. You can "data hover" in your e-mail to see your schedule.
* Zimbra will show you the date and any appointments as your calender along with their status.

* Share schedule information with all colleagues.
* Allows administrators to schedule meetings when there are a minimal number of scheduling conflicts.

ZIMBRA FOR EDUCATION:

* Rich, interactive web interface for complete functionality from campus, home or any other PC location.
* Collaborative document authoring for efficient work on group project and assignments.
* Comprehensive and flexible sharing model, enabling sharings of address books, calendars, documents, or other content by class list, study group, research team, faculty department, etc.

ZIMBRA IN EDUCATION:

Mykel Bates @ Shawn Owens stand beside their newly configured Zimbra box.

* In 2007-2008 school year a 11th grade student Mykel Bates decided to assist Shawn due to his lack of experience.

* After joining the Zimbra online community the two students worked with professionals in the field and successfully configured the lab's new Zimbra network.

* Educators everywhere are encouraged to have students engage in this type of activity

* Zimbra community is more than helpful. and this project motivates students to engage in other high level Internet Technology projects.

STRENGTHS AND ADVANTAGES:

* Allows for more rapid and efficient communication amongst teachers, administrators and parents (SKype, Yahoo Maps, POs).

* Allows easy sharing of RSS feed/ blog information.

* Makes administrationsions and integration less time consuming Through calendar and scheduling applications.

* Encourages collaboration and communicat -ion.

Definition:

Zimbra Collaboration formerly known as Zimbra Collaboration Suite (ZCS) before 2019, is a collaborative software suite that includes an email server and a web client.

Developer : Synacor
Initial release : July 26, 2005
Stable release : July 22, 2019
Platform : Linux

History:

* Zimbra was initially developed by LiquidSys which changed their name to Zimbra in 2005.
* Zimbra Collaboration Suite was first released in 2005.
* Company was purchased by Yahoo! in 2007 which later sold it to VMware in 2010.
* In 2013, VMware sold it to Telligent Systems which changed its name as Zimbra, Inc
* It was then acquired by Synacor in 2015.

Zimbra Collaborative Suite

The software consist of both client and server components and at one time also offered a desktop email client called Zimbra desktop.

TWO VERSIONS OF ZIMBRA

(i) Open Source Version

(ii) Closed Source Version with closed source components such as proprietary Messaging Application Programming Interface connector to Outlook for calender and contact synchronisation. It is a commercially supported version

* The now discontinued Zimbra Desktop was a full-featured free desktop email connect.

* Zimbra Web Client is a full featured collaboration suite that supports email and group calendars.

* At one time it featured document-sharing using Ajax web interface that enabled tool tips, drag and drop items and

and right click menus in the UI.

* Today it has document sharing, chat, videoconferencing, advanced searching capabilities, date relations, online document authoring, "Zimlet" mashups and a full administration UI.
* It is written using Zimbra Ajax Toolkit.
* The Zimbra Server uses several open source projects.
* It exposes a SOAP application programming interface to all its functionality.
* It is also an IMAP and POP3 server.
* The server runs on many Linux distributions
* It can also be run on a Windows Server, using virtual machine and container technology.
* It supports CalDAV, CardDAV and SMTP for messaging, LDAP for directory services and Microsoft Active Directory (AD)
* Zimbra uses Postfix for its MTA functionality.
* It uses technology from ClamAV, DSPAM and SpamAssassin for anti-malware features and S/MIME for email signing encryption.

OS X Server support was dropped with version ZCS 7.0.

* Zimbra can synchronise mail, contacts and calendars items with open source mail clients such as Mozilla Thunderbird and Evolution. and also with proprietary clients such as Microsoft Outlook and Apple Mail either through proprietary connectors or using ActiveSync protocol both available exclusively in the commercially supported version.

* Zimbra also provides native two-way sync to many mobile devices.

Security challenges in Cloud

i) General Challenges
ii) Data center Security challenges
iii) Virtualisation Security Challenges
iv) Network security challenges
v) Platform related security challenges

i) General Challenges:
(i) Interoperability issues
(ii) Hidden cost.
(iii) Unexpected behaviour.
(iv) Threshold policy.
(v) Lack of transparency
(vi) Insecure Interfaces and API's
(vii) Compliance Complexity.
(viii) Vendor Lock in.
(ix) Lack of visibility and control.
(x) Shared technology vulnerability.
(xi) Complexity of risk assessment.
(xii) Implications for consumer piracy.

(ii) Data center security challenges:

* Data management at rest
* Data protection in motion
* Breach notification.
* Lack of application awareness
* Lack of performance and availability.
* Management complexity
* Congested storage network.

(iii) Virtualisation security challenges:

* Traffic management.
* Storage concerns

(iv) Network security:

* Topology dependent complexity
* Policy enforcement complexities.
* Flexible deployment of appliances.
* Application performance.
* Multilayer network complexity.

(v) Platform related security issues:

* Data segregation (SaaS)
* Identity management and access control
* Network and Hypervisor security (IaaS)
* Identity Management (Id M) and sign on process (SaaS).

XEN ARCHITECTURE

Xen

Xen is a unique open source technology[33] invented by a team led by Ian Pratt at the University of Cambridge. Xen was originally developed by the Systems Research Group at the University of Cambridge Computer Laboratory as part of the XenoServers project, funded by the UK-EPSRC. XenoServers aimed to provide a public infrastructure for global distributed computing. Xen plays a key part in that, allowing one to efficiently partition a single machine to enable multiple independent clients to run their operating systems and applications in an environment. This environment provides protection, resource isolation, and accounting. The project web page contains further information as well as pointers to papers and technical reports.[34]

Using Xen server virtualization, the Xen Hypervisor is installed directly on the host hardware and exists as a thin layer between the hardware and the operating system. This abstraction layer allows the host device to run one or more virtual servers. It isolates hardware from the operating system and its applications. Xen is licensed under the GNU General Public License (GPL2) and is available at no charge in both source and object format. According to the official web site, "Xen is, and always will be, open sourced, uniting the industry and the Xen ecosystem to speed the adoption of virtualization in the enterprise."

The Xen Hypervisor supports a wide range of guest operating systems including Windows, Linux, Solaris, and various versions of the BSD operating systems. The Xen Hypervisor has an exceptionally lean footprint. The Xen Hypervisor offers a smaller code base, greater security, and up to 10

times less overhead than alternative virtualization approaches. That means that it has extremely low overhead and near-native performance for guests. Xen reuses existing device drivers (both closed and open source) from Linux, making device management easy. Xen is robust to device driver failure and protects both guests and the Hypervisor from faulty or malicious drivers.

Virtual device monitors (which are also known as hypervisors) are often used on mainframes and large servers seen in data center architectures. Increasingly, they are being used by Internet service providers (ISPs) to provide virtual dedicated servers to their customers. Xen support for virtual-machine live migration from one host to another allows workload balancing and avoids system downtime. Some of the main advantages of Xen server virtualization are

- Consolidation and increased utilization
- The ability to rapidly provision and start a virtual machine
- Better ability to dynamically respond to faults by rebooting a virtual machine or moving a virtual machine to a different hardware platform
- The ability to securely separate virtual operating systems on the same platform
- The ability to support legacy software as well as new operating system instances on the same computer

For operating system development tasks, virtualization has a significant additional benefit—running the new system as a guest avoids any need to reboot the computer whenever a bug is encountered. This protected or insulated environment is known as a "sandbox," and such sandboxed guest systems are useful in computer security research and development. In order to study the effects of malware, viruses, and worms without compromising the host system, developers often prefer to use a sandbox. Hardware appliance vendors increasingly have begun to ship

their products preconfigured with several guest systems. This allows them to deliver complex solutions that are able to execute various software applications running on different operating systems.

Xen touts a para-virtualization technology that is widely acknowledged as the fastest and most secure virtualization software in the industry. Para-virtualization takes full advantage of the latest Intel and AMD hardware virtualization advancements and has fundamentally altered the way virtualization technology is built. Virtual servers and the Hypervisor cooperate to achieve very high performance for I/O, CPU, and memory virtualization.

According to the Xen User Manual,[35] the Xen system has multiple layers, the lowest and most privileged of which is Xen itself. Xen can host multiple guest operating systems. Each operating system is run within a secure virtual machine environment known as a domain. In order to make effective use of the available physical CPUs, such domains are scheduled by Xen. Each guest operating system is responsible for managing its own applications. This management includes scheduling each application within the time allotted by Xen to the virtual machine. The primary domain, domain 0, is created automatically when the system boots, and it has special management privileges. Domain 0 builds other domains and manages their virtual devices. Domain 0 also performs administrative tasks such as suspending, resuming, and migrating other virtual machines. Within domain 0, a process called *xend* is responsible for managing virtual machines and providing access to their consoles.

Unit 5

SECURITY , STANDARDS AND APPLICATIONS
SECURITY IN CLOUDS

Cloud security challenges:

Challenge 1: DDoS attacks

As more and more businesses and operations move to the cloud, cloud providers are becoming a bigger target for malicious attacks. Distributed denial of service (DDoS) attacks are more common than ever before. Verisign reported IT services, cloud and SaaS was the most frequently targeted industry during the first quarter of 2015.

A DDoS attack is designed to overwhelm website servers so it can no longer respond to legitimate user requests. If a DDoS attack is successful, it renders a website useless for hours, or even days. This can result in a loss of revenue, customer trust and brand authority.

Complementing cloud services with DDoS protection is no longer just good idea for the enterprise; it's a necessity. Websites and web-based applications are core components of 21st century business and require state-of-the-art security.

Challenge 2: Data breaches

Known data breaches in the U.S. hit a record-high of 738 in 2014, according to the Identity Theft Research Center, and hacking was [by far] the number one cause. That's an incredible statistic and only emphasizes the growing challenge to secure sensitive data.

Traditionally, IT professionals have had great control over the network infrastructure and physical hardware (firewalls, etc.) securing proprietary data. In the cloud (in private, public and

hybrid scenarios), some of those controls are relinquished to a trusted partner. Choosing the right vendor, with a strong record of security, is vital to overcoming this challenge.

Challenge 3: Data loss

When business critical information is moved into the cloud, it's understandable to be concerned with its security. Losing data from the cloud, either though accidental deletion, malicious tampering (i.e. DDoS) or an act of nature brings down a cloud service provider, could be disastrous for an enterprise business. Often a DDoS attack is only a diversion for a greater threat, such as an attempt to steal or delete data.

To face this challenge, it's imperative to ensure there is a disaster recovery process in place, as well as an integrated system to mitigate malicious attacks. In addition, protecting every network layer, including the application layer (layer 7), should be built-in to a cloud security solution.

Challenge 4: Insecure access points

One of the great benefits of the cloud is it can be accessed from anywhere and from any device. But, what if the interfaces and APIs users interact with aren't secure? Hackers can find these types of vulnerabilities and exploit them.

A behavioral web application firewall examines HTTP requests to a website to ensure it is legitimate traffic. This always-on device helps protect web applications from security breaches.

Challenge 5: Notifications and alerts

Awareness and proper communication of security threats is a cornerstone of network security and the same goes for cloud security. Alerting the appropriate website or application managers as soon as a threat is identified should be part of a thorough

security plan. Speedy mitigation of a threat relies on clear and prompt communication so steps can be taken by the proper entities and impact of the threat minimized.

Final Thoughts

Cloud security challenges are not insurmountable. With the right partners, technology and forethought, enterprises can leverage the benefits of cloud technology.

CDNetworks' cloud security solution integrates web performance with the latest in cloud security technology. With 160 points of presence, websites and web applications are accelerated on a global scale and, with our cloud security, our clients' cloud-based assets are protected with 24/7 end to end security, including DDoS mitigation at the network and application levels.

CLOUD USAGE AND BENEFITS

According to a study by RightScale, in which they surveyed 1000 IT professionals, 96% of respondents use cloud (link).

Both public and private cloud adoption have increased in the last year. The number of respondents now adopting public cloud is 92 percent, up from 89 percent in 2017, while the number of respondents now adopting private cloud is 75 percent, up from 72 percent in 2017. As a result, the overall portion of respondents using at least one public or private cloud is now 96 percent.

The rate of cloud adoption will continue to rise in 2018 and beyond. As more and more CIOs remain troubled by rising IT infrastructure costs and problems of delivering availability, security and performance, adopting cloud infrastructure become a natural step to resolving these problems. With cloud adoption growing among enterprises, forward-thinking CIOs are looking for ways to improve their Internet performance and security, while reducing costs.

Some of the major business advantages of cloud computing:

1. Cost reduction - while some companies are concerned about the cost of migration to the cloud, once you're on the cloud, cost savings are obvious. Easy access to company's data saves time and money, and cloud infrastructure is cheaper than on-premises hardware you have tu buy and upgrade with time.

2. Flexiblity - the cloud offers businesses more flexibility overall versus hosting on a local server. And, if you need extra bandwidth, a cloud-based service can meet that demand instantly, rather than undergoing a complex [and expensive] update to your IT infrastructure. This improved freedom and flexibility can make a significant difference to the overall efficiency of your organization.

3. Mobility - the cloud allows mobile access to enterprise data via smartphones which means that workers with busy schedules, or those living long way from corporate office, can use this feature to instantly keep up-to-date with clients and coworkers.

4. Quality control - by using the cloud, all of the company's data is stored in single format and in one location. With everyone accessing the same information, you can maintain consistency in data, avoid human error, and have a clear record of any revisions or updates.

5. Scalability - Scalability and elasticity via dynamic ["on-demand"] provisioning of resources on a fine-grained, self-service basis in near real-time, without users having to engineer for peak loads. This gives the ability to scale up when the usage need increases or down if resources are not being used.

But what about security?

Cloud security is a debated topic, with some claiming that the cloud is not secure. Some companies are concerned with the fact that if their data is stored on someone else's servers [cloud provider], and that data is accesible from anywhere, how can they be sure that their data is safe from unwanted use by cyber criminals.

Most of these concerns are unwarranted. RapidScale claims that 94 percent of businesses saw an improvement in security after switching to the cloud, and 91 percent said the cloud makes it easier to meet government compliance requirements. The key to this amped-up security is the encryption of data being transmitted over networks and stored in databases.

By using encryption, information is less accessible by hackers or anyone not authorized to view your data. As an added security measure, with most cloud-based services, different security settings can be set based on the user.

Cloud security is often as good as or better than other traditional systems, in part because service providers are able to devote resources to solving security issues that many customers cannot afford to tackle or which they lack the technical skills to address.

Still, cloud security faces some challenges and threats that need to be addressed to make sure that all data is safely stored.

CLOUD SECURITY CHALLENGES

Here are the major security challenges that companies using cloud infrastructure have to prepare for.

Data breaches

A data breach might be the primary objective of a targeted attack or simply the result of human error, application vulnerabilities, or poor security practices. It might involve any kind of information that was not intended for public release, including personal health information, financial information, personally identifiable information, trade secrets, and intellectual property. An organization's cloud-based data may have value to different parties for different reasons.

Access management

Since cloud enables acess to company's data from anywhere, companies need to make sure that not everyone has access to that data. This is done through various policies and guardrails that ensure only legitimate users have access to vital information, and bad actors are left out.

Data encryption

Implementing a cloud computing strategy means placing critical data in the hands of a third party, so ensuring the data remains secure both at rest (data residing on storage media) as well as when in transit is of paramount importance. Data needs to be encrypted at all times, with clearly defined roles when it comes to who will be managing the encryption keys. In most cases, the only way to truly ensure confidentiality of encrypted data that resides on a cloud provider's storage servers is for the client to own and manage the data encryption keys.

Denial of service (DoS/DDoS attacks)

Distributed denial-of-service attack (DDoS), like any denial-of-service attack (DoS), has as its final goal to stop the functioning of the targeted site so that no one can access it. The services of the

targeted host connected to the internet are then stopped temporarily, or even indefinitely.

Advanced persistent threats (APTs)

APTs are a parasitical form of cyber attack that infiltrates systems to establish a foothold in the IT infrastructure of target companies, from which they steal data. APTs pursue their goals stealthily over extended periods of time, often adapting to the security measures intended to defend against them.

Software as a service security:

Software-as-a-service (SaaS) is an on-demand, cloud-based software delivery model that enables organizations to subscribe to the applications they need without hosting them in house. SaaS is one of several categories of cloud subscription services, including platform-as-a-service and infrastructure-as-a-service. SaaS has become increasingly popular because it saves organizations from needing to purchase servers and other infrastructure or maintain an in-house support staff. Instead, a SaaS provider hosts and provides SaaS security and maintenance to their software. Some well-known SaaS applications include Microsoft Office 365, Salesforce.com, Cisco Webex, Box, and Adobe Creative Cloud. Most enterprise software vendors also offer cloud versions of their applications, such as Oracle Financials Cloud.

Benefits of software-as-a-service

According to Market Research Future, the global SaaS market is expected to grow 21% annually for the next few years, reaching $117 billion by the end of 2022. This growth in the popularity of software-as-a-service is due to:

On-demand and scalable resources. Organizations can purchase additional storage, end-user licenses, and features for their applications on an as-needed basis.

Fast implementation. Organizations can subscribe almost instantly to a SaaS application and provision employees, unlike on-premises applications that require more time.

Easy upgrades and maintenance. The SaaS provider handles patches and updates, often without the customer being aware of it.

No infrastructure or staff costs. Organizations avoid paying for in-house hardware and software licenses with perpetual ownership. They also do not need on-site IT staff to maintain and support the application. This enables even small organizations to use enterprise-level applications that would be costly for them to implement.

SaaS security

SaaS providers handle much of the security for a cloud application. The SaaS provider is responsible for securing the platform, network, applications, operating system, and physical infrastructure. However, providers are not responsible for securing customer data or user access to it. Some providers offer a bare minimum of security, while others offer a wide range of SaaS security options.

By 2022, Gartner projects that 95% of cloud security failures will be the customer's fault. To avoid security breaches, customers can implement improved security practices and technologies. Below are SaaS security practices that organizations can adopt to protect data in their SaaS applications.

- **Detect rogue services and compromised accounts.** The average organization uses 1,935 unique cloud services. Unfortunately, the IT departments believe they use only 30 cloud services, according to the 2019 McAfee Cloud Adoption and Risk Report. Moreover, nearly 9% of those cloud services were rated as high-risk services. Organizations can use tools, such as cloud access security brokers (CASB) to audit their networks for unauthorized cloud services and compromised accounts.

- **Apply identity and access management (IAM).** A role-based identity and access management solution can ensure that end users do not gain access to more resources than they require for their jobs. IAM solutions use processes and user access policies to determine what files and applications a particular user can access. An organization can apply role-based permissions to data so that end users will see only the data they're authorized to view.
- **Encrypt cloud data.** Data encryption protects both data at rest (in storage) and data in transit between the end user and the cloud or between cloud applications. Government regulations usually require encryption of sensitive data. Sensitive data includes financial information, healthcare data, and personally identifiable information (PII). While a SaaS vendor may provide some type of encryption, an organization can enhance data security by applying its own encryption, such as by implementing a cloud access security broker (CASB).
- **Enforce data loss prevention (DLP).** DLP software monitors for sensitive data within SaaS applications or outgoing transmissions of sensitive data and blocks the transmission. DLP software detects and prevents sensitive data from being downloaded to personal devices and blocks malware or hackers from attempting to access and download data.
- **Monitor collaborative sharing of data.** Collaboration controls can detect granular permissions on files that are shared with other users, including users outside the organization who access the file through a web link. Employees may inadvertently or intentionally share confidential documents through email, team spaces, and cloud storage sites such as Dropbox.
- **Check provider's security.** The Cloud Adoption and Risk Report surveyed respondents on their trust of cloud providers' security. It found that nearly 70% of them trust their providers to secure their data. However, only 8% of cloud services actually meet the data security requirements defined in the CloudTrust Program. Only 1 in 10 providers encrypt data at rest, and just 18% support multifactor authentication. Clearly, not all of that customer trust is

deserved. An audit of a SaaS provider can include checks on its compliance with data security and privacy regulations, data encryption policies, employee security practices, cybersecurity protection, and data segregation policies.

SaaS security solutions

Several types of security solutions can help organizations improve SaaS security. The solutions can be implemented separately or together as part of a CASB.

- **Data loss prevention (DLP)**] safeguards intellectual property and protects sensitive data in cloud applications, as well as at endpoints such as laptops. Organizations can define data access policies that DLP enforces.
- **Compliance solutions** provide controls and reporting capabilities to ensure compliance with government and industry regulations.
- **Advanced malware prevention** includes technologies such as behavioral analytics and real-time threat intelligence that can help detect and block zero-day attacks and malicious files that may be spread through cloud email and file sharing applications.
- **Cloud access security brokers (CASBs)** protect enterprise data and users across all cloud services, including SaaS, PaaS, and IaaS. According to Gartner's Magic Quadrant for Cloud Access Security Brokers, CASBs detect threats and provide IT departments with greater visibility into data usage and user behavior for cloud services, end users, and devices. CASBs also act immediately to remediate security threats by eliminating security misconfigurations and correcting high-risk user activities applications. CASBs provide a variety of security services, including:
    - Monitoring for unauthorized cloud services
    - Enforcing data security policies including encryption
    - Collecting details about users who access data in cloud services from any device or location
    - Restricting access to cloud services based on the user, device, and application

- Providing compliance reporting

CASB solutions, which are typically SaaS applications, may provide additional capabilities. These may include:

- File encryption
- Pre-built policy templates to guide IT staff through the process of policy creation
- User entity behavior analytics (UEBA) backed by machine learning
- In-application coaching to help end users learn improved security practices
- Security configuration audits to suggest changes to security settings based on best practices

IT departments can learn to protect their cloud applications and data by following cloud security best practices and implementing effective SaaS security solutions. Cloud security solutions from McAfee enable organizations to accelerate their business growth by giving them visibility and control over their applications, devices, and data. Learn more about McAfee cloud security technology.

COMMON STANDARDS

The open cloud consortium:

The Open Cloud Consortium (OCC) has only been around since 2008. That's not very long, but they've done some good work. One of the things they've done, which I think is quite interesting, is establish a testbed that keeps growing every year.
The first phase of the testbed consisted of getting it into operation. It consisted of 240 cores in four U.S. data centers. Those data centers are located at University of Illinois at Chicago, StarLight in Chicago, Calit2 in LaJolla, and Johns Hopkins University in Baltimore. All the racks were connected to a wide area 10 gb/s network. Before the end of its first year, the testbed was upgraded to 480 cores.

In its second year of operation, the OCC conducted phase 2 of operations. In this phase, the number of racks was increased to 9

and the number of nodes to over 250. The number of cores went to over 1,000.

Phase 3 began last year and is currently underway. The goal is to increase some of the 10G network connections to 100G.

The Open Cloud Consortium testbed is an experiment with a lot of potential. Future cloud operations could hinge on what takes place in this testbed. Policy and best practices could be implemented based on their testing.

The Open Cloud Consortium is a newly formed group of universities that is both trying to improve the performance of storage and computing clouds spread across geographically disparate data centers and promote open frameworks that will let clouds operated by different entities work seamlessly together.

Everyone's talking about building a cloud these days. But if the IT world is filled with computing clouds, will each one be treated like a separate island or will open standards allow all to interoperate with each other?

That's one of the questions being examined by the Open Cloud Consortium (OCC), a newly formed group of universities that is both trying to improve the performance of storage and computing clouds spread across geographically disparate data centers and promote open frameworks that will let clouds operated by different entities work seamlessly together.

Cloud is certainly one of the most used buzzwords in IT today, and marketing hype from vendors can at times obscure the real technical issues being addressed by researchers such as those in the Open Cloud Consortium.

"There's so much noise in the space that it's hard to have technical discussions sometimes," says Robert Grossman, chairman of the Open Cloud Consortium and director of the Laboratory for Advanced Computing (LAC) and the National Center for Data Mining (NCDM) at the University of Illinois at Chicago.

Say you're running an application with one cloud provider, such as Amazon's EC2 service, and want to switch to another one. "Our goal

would be that you would not have to rewrite that application if you shifted the provider of cloud services," Grossman says.

The Distributed management Task force:

The Distributed Management Task Force (DMTF) is an industry organization involved in the development, adoption, and interoperability of management standards and initiatives for enterprise and Internet environments. The aim is the exchange of management information in a platform-independent and technology-neutral way, streamlining integration and reducing costs by enabling end-to-end multi-vendor interoperability in management systems.

Within the DMTF, the DMTF Utility Computing Working Group operates in close collaboration with other organizations, like the Global Grid Forum (GGF) and the Organization for the Advancement of Structured Information Standards (OASIS) Web Services Distributed Management (WSDM) Technical Committee, to develop standards related to utility computing. The result of this collaboration is to unify the industry on a set of highly functional and extensible management interfaces.

DMTF	
Abbreviation	DMTF
Formation	1992
Type	Standards Development Organization
Purpose	Developing management standards and promoting interoperability for enterprise

	and Internet environments
Membership	Broadcom Inc., Cisco, Dell Technologies, Hewlett Packard Enterprise, Hitachi, Ltd., HP Inc., Intel Corporation, Lenovo, NetApp, and Software AG.
Website	www.dmtf.org

History

Founded in 1992 as the Desktop Management Task Force, the organization's first standard was the now-legacy Desktop Management Interface (DMI). As the organization evolved to address distributed management through additional standards, such as the Common Information Model (CIM), it changed its name to the Distributed Management Task Force in 1999 , but is now known as, DMTF.

The DMTF continues to address converged, hybrid IT and the Software Defined Data Center (SDDC) with its latest specifications, such as the Redfish standard, SMBIOS and PMCI standards.

DMTF Standards

DMTF standards include:

CADF - Cloud Auditing Data Federation

CIMI - Cloud Infrastructure Management Interface

CIM - Common Information Model

DASH - Desktop and Mobile Architecture for System Hardware

MCTP - Management Component Transport Protocol Including NVMe-MI™, I2C/SMBus and PCIe® Bindings

NCSI - Network Controller Sideband Interface

OVF - Open Virtualization Format

PLDM - Platform Level Data Model Including Firmware Update, Redfish Device Enablement (RDE)

Redfish – Including Protocols, Schema, Host Interface, Profiles

SMASH - Systems Management Architecture for Server Hardware]]

SMBIOS - System Management BIOS

Standards for Application developers:

The reasons to adopt standards in cloud computing closely match the same logic that made the universal usability of the Internet a reality: The more accessible data is, the more interoperable software and platforms are, the more standardized the operating protocols are, the easier it will be to use and the more people with use it -- and the cheaper it will be to implement, operate, and maintain. Systems and software designers see this logic in action when they create a cloud platform and don't have to worry about figuring out how to make it work with a dozen or so network protocols. Cloud application developers feel the power of standards when they build an application using a framework that guarantees almost 100 percent success in such areas as data access, resource allocation, debugging, failover mechanisms, user interface reconfiguration, and error, data, and exception handling... not to mention the shouts of joy when a developer realizes that a favored toolkit can integrate into a favored development platform, sometimes with only the push of a button.
For cloud developers and designers, standards are a powerful addition to their toolbox that allows them to spend more time creating fascinating new apps engineered with elegant code, and less time working out compatibility issues.
IBM and cloud open standards
IBM is deeply involved in the movement to develop cloud standards, advocating open cloud architecture and emphasizing the importance of building standards that support systems interoperability. To move toward a more standards-based, open

cloud ecosystem, IBM has initiated several bold changes over the past few years:

- It based its open cloud services on the open source OpenStack cloud operating system.
- Many of IBM's cloud management products now incorporate OpenStack, including IBM's Smart Cloud Orchestrator and IBM Cloud Manager with OpenStack.
- The company implemented the Bluemix PaaS platform, based on Cloud Foundry, to help developers quickly build web and mobile applications while integrating multiple languages, frameworks, and services as needed.
- IBM integrated Docker containers to use with Bluemix; this simplifies the development and administration of distributed apps since you can build an app with any language and toolchain, then ship it safely for use on virtually any device. And it can scale to thousands of nodes.
- The company has dedicated a number of programmers to open source projects each year.
- Software-defined networking, providing an abstraction layer API for managing the network, has become a star strategy for IBM technologies; so has open source deployment automation with the tools of your choice (like Puppet or Juju or Chef). OAuth has become a key security technology for the integration of REST APIs into the enterprise.
- To prove that global standards are vital to cloud computing, IBM supports and actively participates in multiple cloud standards organizations, including the OpenStack Foundation, CSCC, OASIS, and W3C, to name just a few.

Standards that help development

Let's catch up on cloud standards that designers and developers can use in 2015 to help make software design simpler, cheaper, and faster.

Cloud Standards Customer Council (CSCC)

CSCC is an end-user advocacy group that seeks to "accelerate cloud's successful adoption" as a means to strengthen 21st century enterprises. It is not really a standards organization but a facilitator; it works with existing standards groups to ensure that client requirements are addressed as standards evolve. This group understands that the transition from a traditional IT environment to

a cloud-based environment can require significant changes, so it attempts to guarantee that this transition won't cost end-users the choice and flexibility they enjoy with their current IT environments. Another role of the CSCC is to advocate for the establishment of open, transparent standards for cloud computing; the council believes that the agility and economic efficiencies cloud offers are only possible if the performance, security, and interoperability issues that arise during the transition to the cloud are answered in an open, transparent way.

For designers and developers assisting a client in a move to the cloud, the CSCC case studies, best practices, and roadmaps are an excellent resource.

Distributed Management Task Force (DMTF)

DMTF is an association of industry IT companies and professionals collaborating on and promoting enterprise systems management and interoperability standards with a goal of providing "common management infrastructure components for instrumentation, control, and communication in a platform-independent and technology-neutral way."

The DMTF sports several areas of focus.

Open Virtualization Format (OVF)

The OVF standard, adopted as ISO 17203 by the International Organization for Standardization (ISO), creates uniform formatting for virtual systems-based software. OVF is platform independent, flexible, and open, and can be used by anyone who needs a standardized package for creating a virtual software solution that requires interoperability and portability. OVF simplifies management standards using the Common Information Model (CIM) to standardize management information formats; this reduces design and development overhead by allowing for quicker and more cost-effective implementation of new software solutions.

The payoff for developers: uniform formatting for virtual systems software.

Open Cloud Standards Incubator working group

The Open Cloud Standards Incubator working group's goal is to facilitate management interoperability between in-enterprise private clouds and public and hybrid clouds. The components — cloud

resource management protocols, packaging formats, and security mechanisms—address the increasing need for open, consistent cloud management architecture standards.
Developers get insight into linking internal enterprise private clouds to external clouds of all shapes.

Cloud Management Working Group (CMWG)
CMWG uses the Cloud Infrastructure Management Interface (CIMI) to visually represent the total lifecycle of a cloud service so that you can enhance the implementation and management of that service and make sure it is meeting service requirements. This group can explain how to model the characteristics of an operation, allowing variation of your implementation to be tested prior to final development; it does this with CIM, which creates data classes with well-defined associations and characteristics, as well as a conceptual framework for organizing these components. CIM uses discrete layers: core model, common model, and extension representations.
A programmer or designer can use CIM to create a management model. For developers, this tool lets you test cloud services (and meet requirements) before you finish the project.

Cloud Auditing Data Federation Working Group (CADF)
CADF works to standardize "audit events across all cloud and service providers" with the goal of resolving significant issues in cloud computing due to inconsistencies or incompatibilities. It seeks to ensure consumers of cloud computing systems that the security policies required on their applications are properly managed and enforced. The CADF Working Group develops the DMTF's CADF standard, a model programmers, managers, and users can employ to self-audit application security. An audit event model will eventually support the ability to submit and retrieve audit event data through reports.
For designers and developers, the standards set forth by this group help with event auditing across cloud systems.

For developers and designers tasked with creating and enhancing cloud enterprise systems management (and who isn't in any

project?), these working groups, formats, and components are like a mini-toolbox.

European Telecommunications Standards Institute (ETSI)

ETSI is an organization that produces internationally-applicable standards in information and communications technology to improve systems interoperability, efficiencies, and economies through shared knowledge and expertise.

ETSI Technical Committee Cloud

ETSI Technical Committee Cloud examines issues arising from the convergence of IT and telecommunications. With cloud computing requiring connectivity to extend beyond the local network, cloud network scalability has become dependent on the ability of the telecom industry to handle rapid increases in data transfer; it also works on issues related to interoperability and security.

For developers, this is an excellent resource for cloud mobile standards covering scalability, data transfer, and security.

Cloud Standards Coordination (CSC)

The CSC initiative is responsible for developing a detailed set of standards required to support European Commission policy objectives that address security, interoperability, data portability, and reversibility.

Global Inter-Cloud Technology Forum (GICTF)

GICTF is an organization promoting the standardization of network protocols and interfaces in an effort to create a more reliable cloud services network that solves the problems of security, data quality, system responsiveness, and reliability. This group looks at the cloud ecosphere from the perspective of leased IT and cloud architecture; its operating approach is to assume that the rapidly growing and leased nature of cloud systems may contribute to non-compatible and unstable cloud networks.

GICTF helps developers solve cloud networking and interface problems with security, data quality, system responsiveness, and reliability.

International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC)

ISO is a well-known, 70-year old, independent, non-governmental membership organization made up of 163 member countries. It is the world's largest developer of voluntary international technology standards. The IEC is more than 100 years old and is the leading force behind international standards for all technologies involving the electrical, electronic, and related fields.

Together, these two groups have built JTC 1, a development environment through which international standards for business and consumer applications are created. Working through JTC 1, technology experts build core infrastructure technologies and integrate complex and varied existing technologies. Sub Committee 38 of JTC 1 is concerned with distributed application platforms and services (that is, web services, service-oriented architectures, and cloud). Standards to come out of this group in recent years include Cloud Computing Service Level Agreements (CCSLA), Cloud Computing Interoperability and Portability (CCIP), Cloud Computing Data and its Flow (CCDF), and Cloud Data Management Interface (CDMI).

Developers and designers should consider this a primary fount of cloud standards.

International Telecommunications Union (ITU)

ITU is a specialized agency of the United Nations focusing on developing technical standards to ensure network interoperability and working to improve access in underserved communities. ITU Study Group 13 focuses on next-generation networks (NGN), including mobile technologies and cloud computing, especially as they relate to the ongoing international change from circuit to packet-based networks. Study Group 13 is also interested in developing technologies with reduced energy consumption.

A sub-group of Study Group 13 is the Joint Coordination Activity on Cloud Computing (JCA-Cloud). This group coordinates cloud computing standardization work within the ITU and with other organizations.

If your task is to build low-energy mobile cloud networks or transition to a packet-based network, this is where you start..

National Institute of Standards and Technology (NIST)

NIST is part of the U.S. Department of Commerce and it works to advance measurement science, standards, and technologies.

NIST defines cloud computing for government and industry. It outlines cloud as follows:

- Five essential characteristics:
  - On-demand self service
  - Broad network access
  - Resource pooling
  - Rapid elasticity
  - Measured service
- Three service models:
  - Cloud Software as a Service (SaaS)
  - Cloud Platform as a Service (PaaS)
  - Cloud Infrastructure as a Service (IaaS)
- Four deployment models:
  - Private cloud
  - Community cloud
  - Public cloud
  - Hybrid cloud

NIST promotes systems standardization for security, interoperability, and streamlined connectivity. It does so through its Standards Acceleration to Jumpstart Adoption of Cloud Computing (SAJACC), a program that drives the creation and adoption of cloud computing standards by providing key use cases that show how specific applications can be successfully supported on the cloud.

If you plan to develop cloud technologies for U.S government consumption, NIST will provide your basic guide.

Open Grid Forum (OGF)/Open Cloud Computing Interface (OCCI)

OGF is an international group of IT professionals working through open forums and events to push for the rapid development and deployment of advanced applied distributed computing environments such as cloud, grid, and allied storage and network methods. OGF focuses on scalable enterprise solutions as well as supporting applications for research and science.

The OCCI specification, a "RESTful protocol and API for all kinds of management tasks," is available through OGF. OCCI includes various general-purpose implementations and tools focused on

integration, portability, interoperability, and autonomic scaling and monitoring.
For developers and designers looking to implement cloud technologies that deal with scalable enterprise computing or scientific or research applications (as well as storage and network methods), this group has some good resources.

Object Management Group (OMG)
OMG is an international technology standards consortium that originally targeted standardizing distributed object-oriented systems, but now focuses on modeling programs, systems, and business processes and creating model-based standards. It provides only specifications, not implementations (but any group attempting to get a spec accepted by OMG must provide a working implementation within one year after acceptance). OMG is part of the CSCC (the first group in this list).
OMG's Unified Modeling Language™ (UML) is the basis for modeling application structure, data structure, business process, and architecture. When used with UML's Meta Object Facility (MOF™) and Model-Driven Architecture®, the entire development process is unified which helps reduce cloud portability, interoperability, and reuse issues.
Some of OMG's recent hot topics have included software-defined networking and the industrial version of the Internet of Things.
As a developer or designer, turn to OMG when you want to see what's hot in cloud standards.

Open Cloud Consortium (OCC)
OCC is an organization of universities, companies, and government labs and agencies that supports medical, health care, scientific, and environmental research by managing and operating cloud computing infrastructure. The OCC also develops benchmarks and standards to improve cloud computing, including the MalStone Benchmark which is designed to measure the performance of cloud computing middleware when mining data in data-intensive settings.
Try OCC when you're looking for a real-world testbed for cloud performance benchmarks.

Organization for the Advancement of Structured Information Standards (OASIS)

OASIS is a consortium that represents members in more than 65 countries and promotes multiple cloud protocols and standards:

- OASIS Cloud Application Management for Platforms (CAMP) for cloud interoperability
- OASIS Identity in the Cloud (IDCLOUD) for identity management security challenges
- OASIS Symptoms Automation Framework (SAF) is a catalog-based XML knowledge framework designed to make it easy to use knowledge across domains
- OASIS Topology and Orchestration Specification for Cloud Application (TOSCA) focuses on enhancing the portability of cloud applications and services
- OASIS Cloud Authorization (CloudAuthZ) enables contextual attributes and entitlements sets to be delivered to policy enforcement points in real time (cloud policy management)
- OASIS Public Administration Cloud Requirements (PACR) is a set of public-administration-specific attributes and operational requirements that are necessary in cloud computing services

OASIS has an extensive list of technology committees, so you can probably find one that is dealing with any cloud issue you'd like to resolve.

Storage Networking Industry Association (SNIA)/Cloud Data Management Interface (CDMI)

SNIA is a global group focusing on developing standards and technologies for managing information and storage. Its CDMI is a functional interface that applications can use to manage data elements in the cloud. Management and administrative personnel can also use the interface to manage data, security access, and storage availability.

Look to SNIA for cloud data and storage standards and interfaces to manage them.

Open Group

The Open Group's mission is to enhance business success through IT. It trumpets standards as a way to reduce costs and achieve the

primary goal. Its Cloud Computing Group works to educate its members (and others) on how enterprises of all sizes can take advantage of the cost, scalability, and agility benefits of a cloud supported by standards.
If you need to make a business case to support standards, start with The Open Group.

Association for Retail Technology Standards (ARTS)
As a division of the National Retail Federation, ARTS seeks to reduce the cost of technology through the implementation of standards; as a tool to meet that goal, it has developed the ARTS Data Model, now a standard in the retail industry. Software developers can use the ARTS Data Model as a base for their applications, allowing them to focus more resources on the development of unique user interfaces.
The ARTS Data Model can be a useful application development platform if you're building a retail application.

TM Forum
TM Forum is a global trade association that works to promote the concept of IT as a Service through its Cloud Forum. The group provides research, benchmarks, and roadmaps for the industry technology, as well as best-practice guidebooks, software standards-based interfaces, and training, conferences, and publications.
If you need to take a look at some general best practices, strategies, or operating models for the implementation of cloud systems, TM Forum may have some case studies for you.

STANDARDS FOR MESSAGING:

` Messaging standards Enable simple, unambiguous translation of instructions across different platforms (both synchronous and asynchronous), security of message routing, integrity and encryption. Historically, enterprises used tight coupling of applications within defined infrastructure environments for any business process integration, but they are now progressively looking to exploit the cost saving, agility and best of breed application benefits of moving to cloud based solutions.

Unfortunately achieving cost saving and agility can sometimes be at the expense of process integrity. As a simple example, moving CRM to the cloud may appear very cost effective, but in making the move other business processes integration (including any devices which are accessing associated data), may become disjointed or broken. Of course the integration links can be rebuilt, but this impacts some of the major advantages of a cloud delivery model – flexibility and agility. This then triggers a question for an enterprise: How can disparate cloud applications and data sources (see Figure 1) be made to work together in a seamless, robust and in standardized way

At a fundamental level, Cloud Orchestration acts as tool for connecting heterogeneous clouds and the Internet of Things (IoT), whereas Cloud Messaging acts as a communication platform and enabler of interactions across the orchestrated environment. The principle of a Cloud Messaging Platform is basically that it enables a layer of abstraction between the disparate component parts of a cloud delivery eco-system, whilst still allowing the different elements to interact in a seemingly integrated way via a definable set of messaging protocols. Principal Architecture - Cloud Messaging When end-user organizations push their onpremise applications into cloud environments, the need for an abstraction of messaging capabilities (rather than applicationspecific messaging) becomes particularly pronounced. The traditional messaging approach is poorly suited where vendor and language-specific messaging constrains the applications to use proprietary protocols. Cloud Messaging allows greater flexibility in using the technical environments and the language API's of choice, with the necessary messaging abstracted via the Cloud Messaging Platform. It also allows synchronous or asynchronous communications across networks with greater technical simplicity and efficiency. The Cloud Messaging platform offers a shared cloud-based message queuing framework (Cloud Message QueuingCMQ), enabling messaging between various entities that wish to communicate with each other seamlessly and reliably using standard vendor neutral protocols (like AMQP – Advanced Message Queuing Protocol). App2 App 1 App 3 App 4 App 5 End point End point Publish Publish Subscribe Publish Subscribe Queue 1 Queue 2 Queue 3 Cloud Messaging In

principle, an entity is anything that can participate in a given enterprise business process. With cloud message queuing, the subscriber to a service does not need to understand the protocol used by the service provider or vice versa but can focus on requesting the required business functionality. A Cloud Messaging platform from a logical point of view can be considered as shared queue space in a cloud which enables interoperability between various clients or entities as shown in Figure 2. Viewing this figure from left to right, we see clients [static or mobile] publishing requests or messages to process engines in the cloud, these in turn generate entries to the relevant process queues to be subsequently consumed by the registered client subscribers.

Cloud Security Standards:

As customers transition their applications and data to use cloud computing, it is critically important that the level of security provided in the cloud environment be equal to or better than the security provided by their non-cloud IT environment. Failure to ensure appropriate security protection could ultimately result in higher costs and potential loss of business, thus eliminating any of the potential benefits of cloud computing. This paper focuses primarily on information security requirements for public cloud deployment since this model introduces the most challenging information security concerns for cloud service customers. The CSCC Security for Cloud Computing: 10 Steps to Ensure Success white paper [1] prescribes a series of ten steps that cloud service customers should take to evaluate and manage the security of their cloud environment with the goal of mitigating risk and delivering an appropriate level of support. The following steps are discussed in detail:

1. Ensure effective governance, risk and compliance processes exist
2. Audit operational and business processes

3. Manage people, roles and identities

4. Ensure proper protection of data and information

5. Enforce privacy policies

6. Assess the security provisions for cloud applications

7. Ensure cloud networks and connections are secure

8. Evaluate security controls on physical infrastructure and facilities

9. Manage security terms in the cloud service agreement

10. Understand the security requirements of the exit process

This white paper uses the same list of ten steps as a straightforward way to complement and extend the original whitepaper. For each step, the corresponding subsection highlights the security standards and certifications that are currently available as well as the cloud specific security standards that are currently being developed. Recommendations on which standards and certifications should be required of prospective cloud service providers are highlighted for each step.

Step 1: Ensure effective governance, risk and compliance processes exist Effective governance is essential to guiding management processes and decision making to deliver IT services in accordance with the needs of the organization. Standards to support the governance of IT have existed for a number of years and they are in common use around the world. These governance standards are not specific to cloud computing, but they are sufficiently general so that they can be applied to the governance of cloud computing. General governance standards include: • ISO/IEC 38500 – IT Governance [2] The ISO [International Organization of Standardization] 38500 standard provides a framework for the governance of IT within an organization, offering guiding principles for the senior management

of the organization for the effective, efficient and acceptable use of IT. It is not specific to cloud computing, but it can be used by both cloud service providers and cloud service customers. • COBIT [3] COBIT (Control Objectives for Information and Related Technology) was created by the ISACA organization and provides a framework for IT governance and IT management. It is positioned as a high level framework that sits between business goals and processes and the IT goals and processes. COBIT can be used in conjunction with more detailed standards such as ISO/IEC 20000 and ISO/IEC 27000. • ITIL [4] ITIL (Information Technology Infrastructure Library) is a set of practices for IT service management, which can be applied to the management of cloud services. Information security management is covered, but it is typical to address this area using the ISO/IEC 27002 standard (see below). • ISO/IEC 20000 [5] ISO/IEC 20000 is a series of well-established and internationally recognized standards for IT service management. It is not specific to cloud computing and cloud services, but a new standard, ISO/IEC 20000-7, is being developed to address the application of ISO/IEC 20000 to cloud computing. In addition, the ISO/IEC 20000-11 specification, under development, will describe the relationship of ISO/IEC 20000 to other frameworks and in particular to ITIL. • SSAE 16 [6] SSAE (Statement on Standards for Attestation Engagement) 16 is an audit standard which applies to service organizations including cloud service providers. SSAE 16 audits come in three forms: SOC (Service Organization Controls) 1; SOC 2; and SOC 3. SOC 1 is focused on financial reporting controls, while SOC 2 emphasizes Trust Services Principles to assess the effectiveness of technical and operational security controls. SOC 3 is similar to SOC 2 but reports on whether the organization has achieved Trust Services Principles compliance (yes or no) rather than a detailed analysis of capability. Additionally, the SOC 3 report can be freely distributed. • National Institute of Standards and Technology (NIST) Cybersecurity Framework (CSF) [7] CSF is a cross industry reference framework geared at overlaying federal security assessment and authorization (SA&A) security controls into the private industry (specifically critical infrastructure environments). This is emerging as a standard governance framework for cloud computing in private industry. • Cloud Security Alliance (CSA) Cloud Controls Matrix [8] The CSA conducts

cloud security research, professional education, and provider certification to promote secure delivery and use of cloud computing services. The CSA has published a Cloud Controls Matrix that provides insight into the key security control considerations when assessing cloud provider services. This document is helpful in establishing effective cloud security governance. In addition to the general standards and frameworks listed above, there are others that operate at country or regional levels or that apply to specific industries or to specific types of data. If your business operates in the relevant countries or in the relevant industry sector, these may apply. Some examples are listed below, but there are others and it is necessary to understand which may apply to your use of cloud services: Health Care • HIPAA [9] The Health Insurance Portability and Accountability Act (HIPAA) is a regulation that requires U.S. health care providers to maintain the confidentiality and security of protected health information (PHI). Payment Card • PCI-DSS [10] The Payment Card Industry Data Security Standard (PCI-DSS) is an industry mandate that defines the minimum security controls needed to protect customer cardholder data throughout its lifecycle

End user access to cloud computing:

End-user computing (EUC) is a term that refers to the technologies that IT professionals use to deploy, manage and secure the devices, applications and data that workers require to perform their jobs. The major components of EUC are physical desktop computing, virtual desktop computing and mobile computing, each of which involves several different technologies.

Types of EUC

End-user computing encompasses a wide variety of user-facing resources, including:

- desktop and notebook computers;
- desktop operating systems and applications;
- smartphones, tablets, wearables and other mobile devices;
- mobile, web and cloud applications; and
- virtual desktops and applications.

EUC also covers the technologies that IT professionals use to provide access to these resources, such as:

- Windows management and security tools;
- enterprise mobility management software, which includes mobile device management and mobile application management;
- desktop and application virtualization platforms and management tools; and
- enterprise file sync-and-share services.

End-user computing services

Traditionally, IT managed the different components of end-user computing separately. As the consumerization of IT and the bring your own device (BYOD) trend gained steam, however, more organizations realized the need to provide access to corporate applications and data across multiple device types.

In an attempt to simplify this process, vendors began offering products and services designed to work across multiple areas of EUC. Examples of these products and services include:

- tools that provide monitoring and management of both physical and virtual desktops and applications;
- app refactoring, which uses virtualization to create mobile-friendly versions of Windows and web apps;
- unified endpoint management (UEM), which allows IT to apply and enforce mobile device management policy on Windows 10 PCs; and
- workspace suites, which aim to provide centralized consoles where end users can access all of their required applications and data, and IT can securely manage that access.

Advantages and disadvantages

End-user computing benefits organizations by securely enabling a mobile, distributed workforce. Its major disadvantage, despite all the work that has gone into unified management, is its complexity.Most organizations have not yet migrated to Windows 10, which means they can't take advantage of UEM, and they have to use separate products to manage PCs and mobile devices. Workspace suites do not yet provide complete integration between

all of the disparate products they bundle together. And trying to run an application on an operating system or device that it wasn't built for can result in problems around compatibility and user experience.

Mobile Internet devices and cloud:

A Mobile Internet Device (MID) is a small multimedia-enabled mobile device that provides wireless Internet access. MIDs facilitate real-time and two-way communication by filling the multimedia gap between mobile phones and tablets.

A MID is larger than a handheld device, like a smartphone, but smaller than an ultra-mobile PC (UMPC). MID technology focuses on providing entertainment, information and location-based services to individual consumers, rather than enterprises.

Techopedia explains *Mobile Internet Device (MID)*

A MID has several positive advantages over smaller and larger devices. It provides a larger display than a typical mobile phone with preloaded Internet functionality, which facilitates Web browsing. The compact MID design allows users to easily carry a MID in a backpack or purse. Also, MID devices are significantly lighter than standard laptops.

MIDs provide efficient and wireless connectivity. Features based on Intel's 2007 prototype are as follows:

- Display screen: 4.5 to 6 inches
- Boot time: Faster than UMPC
- Manufacturer's suggested retail price (MSRP): Lower than UMPC
- Random access memory (RAM): 256 or 512 Mb
- Pixel resolution: 800x480 or 1024x600
- Easy interface
- Wide local area network (WLAN) or Wi-Fi technology

In 2007, Intel introduced its first generation MID (code-named McCaslin) with a 90 nm Intel A100/A110 processor that ran at 600-800 MHz. In 2011, Intel will release its fourth generation (4G) processor (code-named Medfield), which will contain a 32 nm Intel Atom processor (speed unknown).

Intel MIDs use the Moblin (now known as MeeGo) model, which is an open source Linux OS with the latest dual core processors. Key features are a built-in Global Positioning System (GPS) and long battery life.

Mobile Cloud Computing (MCC) is the combination of cloud computing and mobile computing to bring rich computational resources to mobile users, network operators, as well as cloud computing providers.[1][2][3] The ultimate goal of MCC is to enable execution of rich mobile applications on a plethora of mobile devices, with a rich user experience.[4] MCC provides business opportunities for mobile network operators as well as cloud providers.[5][6] More comprehensively, MCC can be defined as "a rich mobile computing technology that leverages unified elastic resources of varied clouds and network technologies toward unrestricted functionality, storage, and mobility to serve a multitude of mobile devices anywhere, anytime through the channel of Ethernet or Internet regardless of heterogeneous environments and platforms based on the pay-as-you-use principle.

Open research issues [edit]

Although significant research and development in MCC is available in the literature, efforts in the following domains is still lacking:[5][7]

- **Architectural issues:** A reference architecture for heterogeneous MCC environment is a crucial requirement for unleashing the power of mobile computing towards unrestricted ubiquitous computing.
- **Energy-efficient transmission:** MCC requires frequent transmissions between cloud platform and mobile devices, due to the stochastic nature of wireless networks, the transmission protocol should be carefully designed.[11][12]
- **Context-awareness issues:** Context-aware and socially-aware computing are inseparable traits of contemporary handheld

computers. To achieve the vision of mobile computing among heterogeneous converged networks and computing devices, designing resource-efficient environment-aware applications is an essential need.

- **Live VM migration issues:** Executing resource-intensive mobile application via Virtual Machine (VM) migration-based application offloading involves encapsulation of application in VM instance and migrating it to the cloud, which is a challenging task due to additional overhead of deploying and managing VM on mobile devices.
- **Mobile communication congestion issues:** Mobile data traffic is tremendously hiking by ever increasing mobile user demands for exploiting cloud resources which impact on mobile network operators and demand future efforts to enable smooth communication between mobile and cloud endpoints.
- **Trust, security, and privacy issues:** Trust is an essential factor for the success of the burgeoning MCC paradigm. It is because the data along with code/component/application/complete VM is offloaded to the cloud for execution. Moreover, just like software and mobile application piracy, the MCC application development models are also affected by the piracy issue.[10] Pirax[10] is known to be the first specialized framework for controlling application piracy in MCC requirements

TASK MANAGEMENT

Task management is an activity in which an individual or team leader tracks a task throughout its life cycle and makes decisions based on the progress. Task management is done using software tools that help effectively organize and manage tasks by using functions such as task creation, planning and assignment, tracking and reporting.

The reports generated assist the management in analyzing the overall efficiency of an individual, department or organization.

Task management tools are used to track personal, group or shared tasks. The tools may be free or premium software applications, and run in either standalone, LAN-based or Web-based mode. The size and functions of the tools depend on the requirements of the task and on whether they are used for an individual, small-sized or medium-sized business or for a corporate task management's activity. Typical features include the following:

- Task and subtask creation, assignment and reassignment, prioritization, task sharing, etc.
- Notification and report generation
- Calendar
- Security and access control
- Mobile capability, integration with other systems and chat systems
- Sorting

The team leader is responsible for creating, assigning, prioritizing and monitoring a task to ensure that it is completed on time. When managing a task assigned to a group, some tools provide a real-time view and easy access to all related content and discussions. Administrative features allow administrators to change priorities, reassign tasks, add more time or people to handle the tasks and approve tasks when finished.

With a centralized task management point, it is possible to track and identify a team based on what it is doing, determine the time a task is taking and to determine the team's efficiancy. Most tools allow users to visually manage a task and to see the history of completed, pending, overdue and ongoing tasks. The reports generated by the tools may contain details such as the start date, deadline, overdue date, task budget, main tasks, subtasks and time allocation.

Task management is therefore an important process that allows supervisors to monitor the time employees spend on a task, the ongoing and completed tasks, and an employee's workload and performance. This information can be used to balance workloads, forecast bottlenecks and guard against delays and missed deadlines.

BENEFITS OF CLOUD COMPUTING FOR EVENT MANAGEMENT PROFESSIONALS

The truth is, successful event professionals are very quick, flexible and excellent at coping with stress. But the secret to their success also lies in the technology they use: Cloud computing gives event professionals the flexibility they need.

1. Permanent Connection and Accessibility

The daily work routine of many people starts by logging in to their computers and opening their e-mails, spreadsheets, or event software. Whenever something goes wrong, there's IT staff that may help.

The event industry, however, has a different rhythm. Event managers usually do not sit in front of their PC's from 9 to 5. Being on-site, spending time with clients or travelling – these are only a few things that form a huge part of event professionals' jobs. As a consequence, permanent accessibility to client and event data is essential to them in order to deliver smooth and successful events.

Imagine being connected to your important data at any time. Whether you want to look up contact details, phone numbers or find out about clients' event histories. All you need is an internet connection and your laptop or mobile device. You simply log in to your event software via your browser wherever you are. Sound like a dream? It is possible thanks to cloud computing.

2. Accuracy Thanks to Real-Time Data

Forget that "daily-4pm-function-sheet" that will be out-of-date soon. If your data is constantly updated in one place and accessible through the internet, there is no need to print out event data. And that gives you so much more flexibility with client wishes and short notice changes.

Real-time data comes with another major benefit: It is reliable. Every item that you order until the very end of the event is captured and therefore invoiced.

3. No IT and Hardware Worries

Really, you don't need to be a tech expert to work with the cloud. You can put your mind at rest and let others take care of IT. There are savvy people at the "end of the line" making sure you can work efficiently and assuring data security at any time.

Not having to maintain an internal IT infrastructure is a big opportunity to save money. We often hear about time and resource constraints when speaking to our clients. It is very difficult for event organisations to find the time to maintain an internal technical department.

If the thought of simply opening your browser, logging in and immediately being connected to your client and event data makes you happy? Then you should consider moving to the cloud and choosing the easy and efficient way of managing events.

PROJECT MANAGEMENT

Coordination and **collaboration** are the two essential components of handling a project. Coordination is within a location for traditional projects, and across locations for distributed projects. There is a need for Collaborative Project Management Architectures (CPMAs) in order to build systems that can overcome the challenges faced by traditional project management.

Traditional Project Management Scenarios

When team members or companies carry out project management (PM), there are many potential mistakes or pitfalls to which they can easily fall prey. Instead of highlighting them all, let's focus on a few common overarching themes identified in the literature. Combining together all of these themes account for the reason why many major projects either fail or are significantly less efficient and effective than they could be.

Over-emphasizing of PM as a Project Reporting Mechanism

Traditional project management often employs a simple passive reporting mechanism instead of a dynamic teamwork coordinating approach. In many companies, the project management methodology is assumed as a corporate reporting tool rather than an efficient system that the various parts of the company can use to help themselves. In this type of situation, information flow is less among project contributors.

Ineffective and Inefficient Communication

In traditional PM, communication may be ineffective due to many reasons −

- Misunderstandings due to inexplicit or poor communication.
- Members having a poor grasp regarding the problem.
- Different interpretations by different team members.

Communication is also inefficient or not up to the mark because of various reasons like −

- Untimely communication.
- Failure to update latest notification to every team member who needs to know.
- Poor communication skills and capabilities are mostly cited as the main reason for project failure.

Managing Project Inputs and Outputs but not Process

Another serious problem in traditional project management is that employees manage deliverables and resources, but they don't manage the process.

- Team leaders create PERT and plan the project within a timeline, they manage time, budget, equipment, human resources, and the product; but fail to manage work process.
- One reason for the failure of software projects is the lack of real-time improvement measurement systems to identify potential risks in the initial stages, before they become serious threats to the progress of the product.
- If employees only handle project inputs and outputs, the process remains a black box and project members are unaware of the fact that something has gone wrong until it is too late to correct the issue without causing large amounts of rework and increase complexity.

This results in making PM a reactive process, rather than a proactive one.

Reactive Management

Reactive management defines a passive PM strategy in which project managers conduct incomplete planning with a hope that everything will be fine in the end.

- Reactive project managers react to what has happened and they seldom plan for the future. They do not review their own or others' previous experiences to gain insight from lessons learned over time.
- In reactive management, employees spend a significant amount of project time on reworking deliverables and rectifying errors.
- Another common issue in reactive situations is almost all the rework must be done manually, including searching for work that is influenced by changes in other parts of the project.

Reactive Project Management is often accompanied by lack of systematic procedure for storing project information which leads to compounding the problems of poor planning and the need for rework.

Lack of an Electronic Project Repository

Lack of an electronic repository is a company-wide problem as well as a project–specific issue. A paper-based repository has several limitations like –

- Retrieval delays
- Lost documents
- Incomplete files and storage problems
- Error proneness due to data extraction, interpretation, and repackaging.
- Difficulty in coordination and failure under given time constraints.

Lack of an electronic project repository leads to inadequate project documentation.

- Project members are usually more concerned with accomplishment of current project rather than capturing and archiving information that can be useful at a later time.
- Most of the project related information is not stored at all, like the project processes, contexts, rationales, or artifacts. Even if they are stored, they may not be structured, organized and indexed in a way that enables project members to easily access, search, and retrieve the information.

Collaborative Project Management as a Solution

We assume that various challenges faced in traditional PM can be addressed by using collaborative PM tools and processes. A collaborative PM tool deals with explicit representation of project information and timely sharing of the adequate information.

Let's have a look at how a collaborative PM environment can overcome the limitations that plague traditional PM.

Considering PM as a Project Analysis Mechanism

When team members consider PM as a project reporting tool, they care about the outputs of the PM rather than the analysis process which gives those outputs.

- When people consider PM as a project reporting tool, extra project-related information that is usually not formally captured, will effectively be lost when memory fades.
- On the other hand, when employees treat PM as a project analysis tool instead of considering it as merely a reporting tool, the product will be the task information, decision rationale, and other related artifacts.

Effective and Efficient Communication

Explicit representation of project information is important for effective and efficient communication, especially in distributed situations.

- Effective communication also describes clear specification and unanimous agreement of significant project information such as key concepts, ideas, project process, team member duties, and responsibilities.
- All these are documented and saved for future reference by the team members.
- In addition to support for explicit representation of project information, a collaborative PM tool needs to support, manage and handle automatic notification of task status changes, and allow members to discuss and give feedback on one another's work.

Explicit representation, however, is an important step towards effective communication.

Managing Project Process as well as Inputs and Outputs

Managing the project process is the most crucial part of PM. One way to get an idea about the process is though a project lifecycle. The project lifecycle is broadly categorized into four major steps –

- **Step 1** – Understanding the project (problem definition and specification) – planning the project.
- **Step 2** – Executing.
- **Step 3** – Tracking and controlling the project.
- **Step 4** – Closing the project.

Here the team members manage the inputs and outputs, but not the process, they overemphasize step 1, 2, and 4 at the cost of step 3.

The nature of project processes is dynamic and changes significantly from the original project plans and expectations as the project improves further. An ongoing process always leads to some changes in project inputs and outputs and these changes, in turn, lead to further changes in the project process.

A collaborative PM tool allows team members to update, and review one another's work progress, collect project measures like resources spent on the task, and access the current work of others within a time bound.

Proactive Project Management

Proactive project management refers to future-oriented planning, risk management, and change management in the current ongoing project. Proactive management requires project team members to conduct precise, specified, clear, and detailed planning at the beginning of the project cycle, identifying potential risks, and making plans to mitigate those risks.

A project manager, who conducts proactive management, examines task interdependencies and makes their decisions based on precise "hard" data rather than wishful thinking.

- Proactive management is followed by learning.
- Proactive management of the PM process requires an Enterprise's project memory, from which members can learn during an ongoing project and refer back for future projects.

One way to implement an effective business organizational project memory is with the help of an electronic project repository.

Employing an Electronic Project Repository

With the growing advancement of information technology, files in digital format are easier to store, access, retrieve, edit, and route. The paper-based repository is replaced with an electronic project repository. The goal of an electronic project repository is to control, handle, and share project information efficiently and effectively.

- Effective information management improves the overall project performance within budget, reducing data entry and reentry costs, eliminating duplication, information loss, reducing product development time, fostering progress in process quality, standardizing work processes, improving management's ability to efficiently retrieve accurate information, and increasing management control.
- An electronic project repository can be connected via middleware with other information systems in the organization and provide a smooth information flow.

Database-as-a-service (DBaaS)

With a database as a service model, application owners do not have to install and maintain the database themselves. Instead, the database service provider takes responsibility for installing and maintaining the database, and application owners are charged according to their usage of the service. This is a type of Saas - Software as a Service.

Architecture and common characteristics

- Most database services offer web-based consoles, which the end user can use to provision and configure database instances.
- Database services consist of a database-manager component, which controls the underlying database instances using a service API. The service API is exposed to the end user, and permits users to perform maintenance and scaling operations on their database instances.
- Underlying software-stack stack typically includes the operating system, the database and third-party software used to manage the database. The service provider is responsible for installing, patching and updating the underlying software stack and ensuring the overall health and performance of the database.
- Scalability features differ between vendors – some offer auto-scaling, others enable the user to scale up using an API, but do not scale automatically.
- There is typically a commitment for a certain level of high availability (e.g. 99.9% or 99.99%). This is achieved by replicating data and failing instances over to other database instances..

Data model

The design and development of typical systems utilize data management and relational databases as their key building blocks. Advanced queries expressed in SQL work well with the strict relationships that are imposed on information by relational databases. However, relational database technology was not initially designed or developed for use over distributed systems. This issue has been addressed with the addition of clustering enhancements to the relational databases, although some basic

tasks require complex and expensive protocols, such as with data synchronization.[1]

Modern relational databases have shown poor performance on data-intensive systems, therefore, the idea of NoSQL has been utilized within database management systems for cloud based systems.[2] Within NoSQL implemented storage, there are no requirements for fixed table schemas, and the use of join operations is avoided. "The NoSQL databases have proven to provide efficient horizontal scalability, good performance, and ease of assembly into cloud applications."[3] Data models relying on simplified relay algorithms have also been employed in data-intensive cloud mapping applications unique to virtual frameworks.[4]

It is also important to differentiate between cloud databases which are relational as opposed to non-relational or NoSQL.[5]

SQL databases

are one type of database which can run in the cloud, either in a virtual machine or as a service, depending on the vendor. While SQL databases are easily vertically scalable, horizontal scalability poses a challenge, that cloud database services based on SQL have started to address.[6][*need quotation to verify*]

NoSQL databases

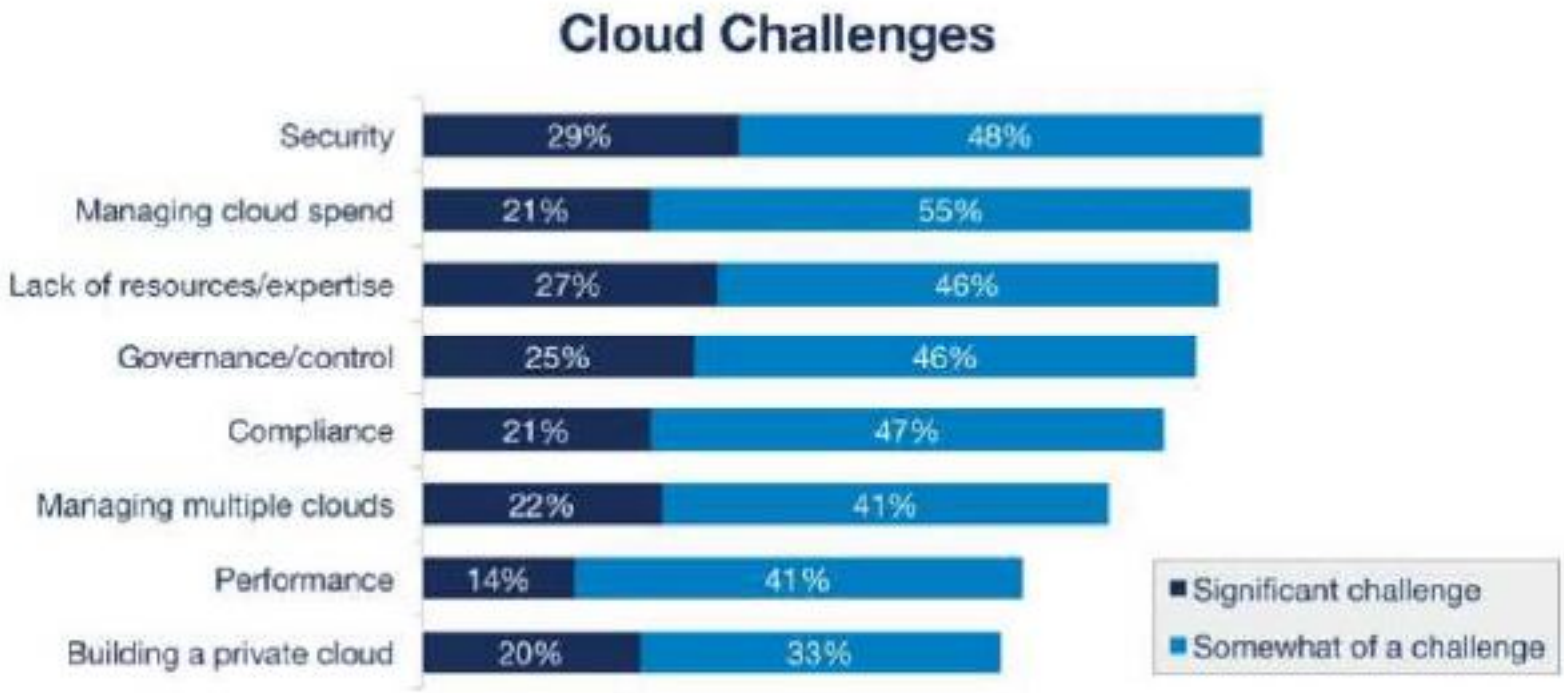
are another type of database which can run in the cloud. NoSQL databases are built to service heavy read/write loads and can scale up and down easily,[7] and therefore they are more natively suited to running in the cloud.: However, most contemporary applications are built around an SQL data model, so working with NoSQL databases often requires a complete rewrite of application code.[8]

Some SQL databases have developed NoSQL capabilities including JSON, binary JSON (e.g. BSON or similar variants), and key-value store data types.

A multi-model database with relational and non-relational capabilities provides a standard SQL interface to users and applications and thus facilitates the usage of such databases for contemporary applications built around an SQL data model. Native multi-model databases support multiple data models with one core and a unified query language to access all data models.

CHALLENGES AND RISKS IN CLOUD COMPUTING

In January 2018, RightScale conducted its annual State of the Cloud Survey on the latest cloud trends. They questioned 997 technical professionals across a broad cross-section of organizations about their adoption of cloud infrastructure. Their findings were insightful, especially in regards to current cloud computing challenges. To answer the main question of what are the challenges for cloud computing, below we have expanded upon some of their findings and provided additional cloud computing problems that businesses may need to address.



1. Security issues

Security risks of cloud computing have become the top concern in 2018 as 77% of respondents stated in the referred survey. For the longest time, the lack of resources/expertise was the number one voiced cloud challenge. In 2018 however, security inched ahead.

We already mentioned the hot debate around data security in our business intelligence trends 2019 article, and security has indeed been a primary, and valid, concern from the start of cloud computing technology: you are unable to see the exact location where your data is stored or being processed. This increases the cloud computing risks that can arise during the implementation or

management of the cloud. Headlines highlighting data breaches, compromised credentials, and broken authentication, hacked interfaces and APIs, account hijacking haven't helped alleviate concerns. All of this makes trusting sensitive and proprietary data to a third party hard to stomach for some and, indeed, highlighting the challenges of cloud computing. Luckily as cloud providers and users, mature security capabilities are constantly improving. To ensure your organization's privacy and security is intact, verify the SaaS provider has secure user identity management, authentication, and access control mechanisms in place. Also, check which database privacy and security laws they are subject to.

While you are auditing a provider's security and privacy laws, make sure to also confirm the third biggest issue is taken care of: compliance. Your organization needs to be able to comply with regulations and standards, no matter where your data is stored. Speaking of storage, also ensure the provider has strict data recovery policies in place.

The security risks of cloud computing have become a reality for every organization, be it small or large. That's why it is important to implement a secure BI cloud tool that can leverage proper security measures.

2. Cost management and containment

The next part of our cloud computing risks list involves costs. For the most part cloud computing can save businesses money. In the cloud, an organization can easily ramp up its processing capabilities without making large investments in new hardware. Businesses can instead access extra processing through pay-as-you-go models from public cloud providers. However, the on-demand and scalable nature of cloud computing services make it sometimes difficult to define and predict quantities and costs.

Luckily there are several ways to keep cloud costs in check, for example, optimizing costs by conducting better financial analytics and reporting, automating policies for governance, or keeping the management reporting practice on course, so that these issues in cloud computing could be decreased.

3. Lack of resources/expertise

One of the cloud challenges companies and enterprises are facing today is lack of resources and/or expertise. Organizations are increasingly placing more workloads in the cloud while cloud technologies continue to rapidly advance. Due to these factors, organizations are having a tough time keeping up with the tools. Also, the need for expertise continues to grow. These challenges can be minimized through additional training of IT and development staff. A strong CIO championing cloud adoption also helps. As Cloud Engineer Drew Firment puts it:

"The success of cloud adoption and migrations comes down to your people—and the investments you make in a talent transformation program. Until you focus on the #1 bottleneck to the flow of cloud adoption, improvements made anywhere else are an illusion."

SME (small and medium-sized) organizations may find adding cloud specialists to their IT teams to be prohibitively costly. Luckily, many common tasks performed by these specialists can be automated. To this end companies are turning to DevOps tools, like Chef and Puppet, to perform tasks like monitoring usage patterns of resources and automated backups at predefined time periods. These tools also help optimize the cloud for cost, governance, and security.

4. Governance/Control

There are many challenges facing cloud computing and governance/control is in place number 4. Proper IT governance should ensure IT assets are implemented and used according to agreed-upon policies and procedures; ensure that these assets are properly controlled and maintained, and ensure that these assets are supporting your organization's strategy and business goals.

In today's cloud-based world, IT does not always have full control over the provisioning, de-provisioning, and operations of infrastructure. This has increased the difficulty for IT to provide the governance, compliance, risks and data quality management required. To mitigate the various risks and uncertainties in transitioning to the cloud, IT must adapt its traditional IT governance and control processes to include the cloud. To this effect, the role of central IT teams in the cloud has been evolving over the last few years. Along with business units,

central IT is increasingly playing a role in selecting, brokering, and governing cloud services. On top of this third-party cloud computing/management providers are progressively providing governance support and best practices.

5. Compliance

One of the risks of cloud computing is facing today is compliance. That is an issue for anyone using backup services or cloud storage. Every time a company moves data from the internal storage to a cloud, it is faced with being compliant with industry regulations and laws. For example, healthcare organizations in the USA have to comply with HIPAA (Health Insurance Portability and Accountability Act of 1996), public retail companies have to comply with SOX (Sarbanes-Oxley Act of 2002) and PCI DSS (Payment Card Industry Data Security Standard).

Depending on the industry and requirements, every organization must ensure these standards are respected and carried out.

This is one of the many challenges facing cloud computing, and although the procedure can take a certain amount of time, the data must be properly stored.

Cloud customers need to look for vendors that can provide compliance and check if they are regulated by the standards they need. Some vendors offer certified compliance, but in some cases, additional input is needed on both sides to ensure proper compliance regulations.

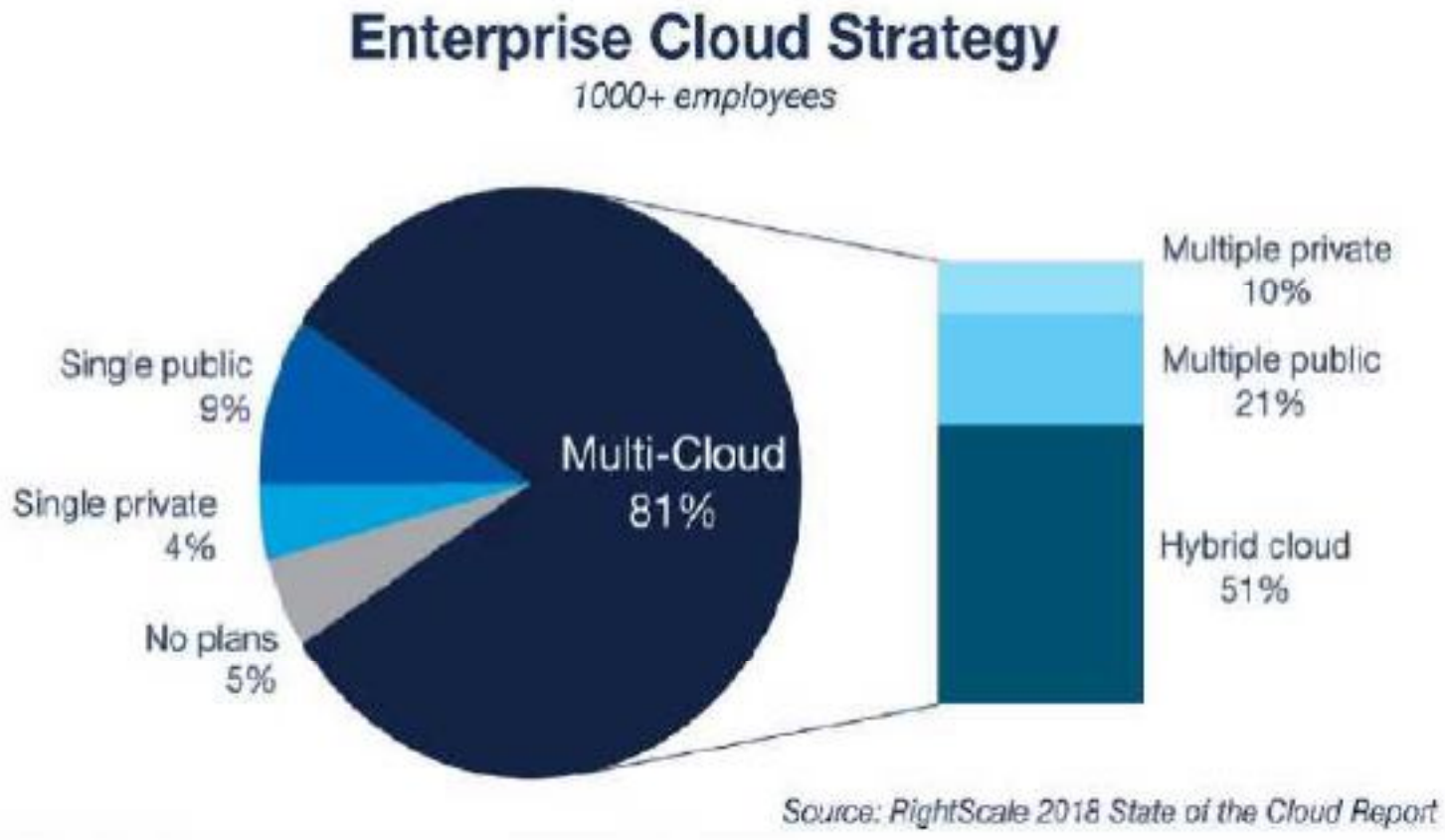
6. Managing multiple clouds

Challenges facing cloud computing haven't just been concentrated in one, single cloud.

The state of multi-cloud has grown exponentially in recent years. Companies are shifting or combining public and private clouds and, as mentioned earlier, tech giants like Alibaba and Amazon are leading the way.

In the referred survey, 81 percent of enterprises have a multi-cloud strategy. Enterprises with a hybrid strategy (combining public and private clouds) fell from 58 percent in 2017 to 51 percent in 2018,

while organizations with a strategy of multiple public clouds or multiple private clouds grew slightly.



While organizations leverage an average of almost 5 clouds, it is evident that the use of the cloud will continue to grow. That's why it is important to answer the main questions organizations are facing today: what are the challenges for cloud computing and how to overcome them?

7. Performance

When a business moves to the cloud it becomes dependent on the service providers. The next prominent challenges of moving to cloud computing expand on this partnership. Nevertheless, this partnership often provides businesses with innovative technologies they wouldn't otherwise be able to access. On the other hand, the performance of the organization's BI and other cloud-based systems is also tied to the performance of the cloud provider when it falters. When your provider is down, you are also down.

This isn't uncommon, over the past couple of years all the big cloud players have experienced outages. Make sure your provider has the right processes in place and that they will alert you if there is ever an issue.

For the data-driven decision making process, real-time data for organizations is imperative. Being able to access data that is stored on the cloud in real-time is one of the imperative solutions an organization has to consider while selecting the right partner.

With an inherent lack of control that comes with cloud computing, companies may run into real-time monitoring issues. Make sure your SaaS provider has real-time monitoring policies in place to help mitigate these issues.

8. Building a private cloud

Although building a private cloud isn't a top priority for many organizations, for those who are likely to implement such a solution, it quickly becomes one of the main challenges facing cloud computing – private solutions should be carefully addressed.

Creating an internal or private cloud will cause a significant benefit: having all the data in-house. But IT managers and departments will need to face building and gluing it all together by themselves, which can cause one of the challenges of moving to cloud computing extremely difficult.

It is important to keep in mind also the steps that are needed to ensure the smooth operation of the cloud:

- Automating as many manual tasks as possible (which would require an inventory management system)
- Orchestration of tasks which has to ensure that each of them is executed in the right order.

As this article stated: *the cloud software layer has to grab an IP address, set up a virtual local area network (VLAN), put the server in the load balancing queue, put the server in the firewall rule set for the IP address, load the correct version of RHEL, patch the server software when needed and place the server into the nightly backup queue.*

That being said, it is obvious that developing a private cloud is no easy task, but nevertheless, some organizations still manage and plan to do so in the next years.

9. Segmented usage and adoption

Most organizations did not have a robust cloud adoption strategy in place when they started to move to the cloud. Instead, ad-hoc strategies sprouted, fueled by several components. One of them was the speed of cloud adoption. Another one was the staggered expiration of data center contracts/equipment, which led to intermittent cloud migration. Finally, there also were individual development teams using the public cloud for specific applications or projects. These bootstrap environments have fostered full integration and maturation issues including:

- Isolated cloud projects lacking shared standards
- Ad hoc security configurations
- Lack of cross-team shared resources and learnings

In fact, a recent survey by IDC of 6,159 executives found that just 3% of respondents define their cloud strategies as "optimized". Luckily, centralized IT, strong governance and control policies, and some heavy lifting can get usage, adoption, and cloud computing strategies inline.

Nearly half of the decision makers believe that their IT workforce is not completely prepared to address the cloud computing industry challenges and managing their cloud resources over the next 5 years. Since businesses are adopting the cloud strategy more often than ever, it is eminent that the workforce should keep up and carefully address the potential issues.

10. Migration

One of the main cloud computing industry challenges in recent years concentrates on migration. This is a process of moving an application to a cloud. An although moving a new application is a straightforward process, when it comes to moving an existing application to a cloud environment, many cloud challenges arise.

A recent survey conducted by Velostrata showed that over 95% of companies are currently migrating their applications to the cloud,

and over half of them find it more difficult than expected – projects are over budget and deadline.

What are the challenges faced during storing data in the cloud? Most commonly cited were:

- Extensive troubleshooting
- Security challenges
- Slow data migrations
- Migration agents
- Cutover complexity
- Application downtime

In another survey, although not that recent, but a picturesque perception of the migration to the cloud; IT professionals stated they would rather "get a root canal, dig a ditch, or do their own taxes" than address challenges in cloud computing regarding the deployment process