Platform as a Service (PaaS)

- **PaaS" stands for "Platform as a Service,"** which is a cloud computing model where a third-party provider delivers a complete platform for developing, deploying, and managing applications, allowing users to focus on building their applications without managing the **underlying infrastructure like servers, operating systems, and databases**; essentially, it provides a ready-to-use environment for application development in the cloud.

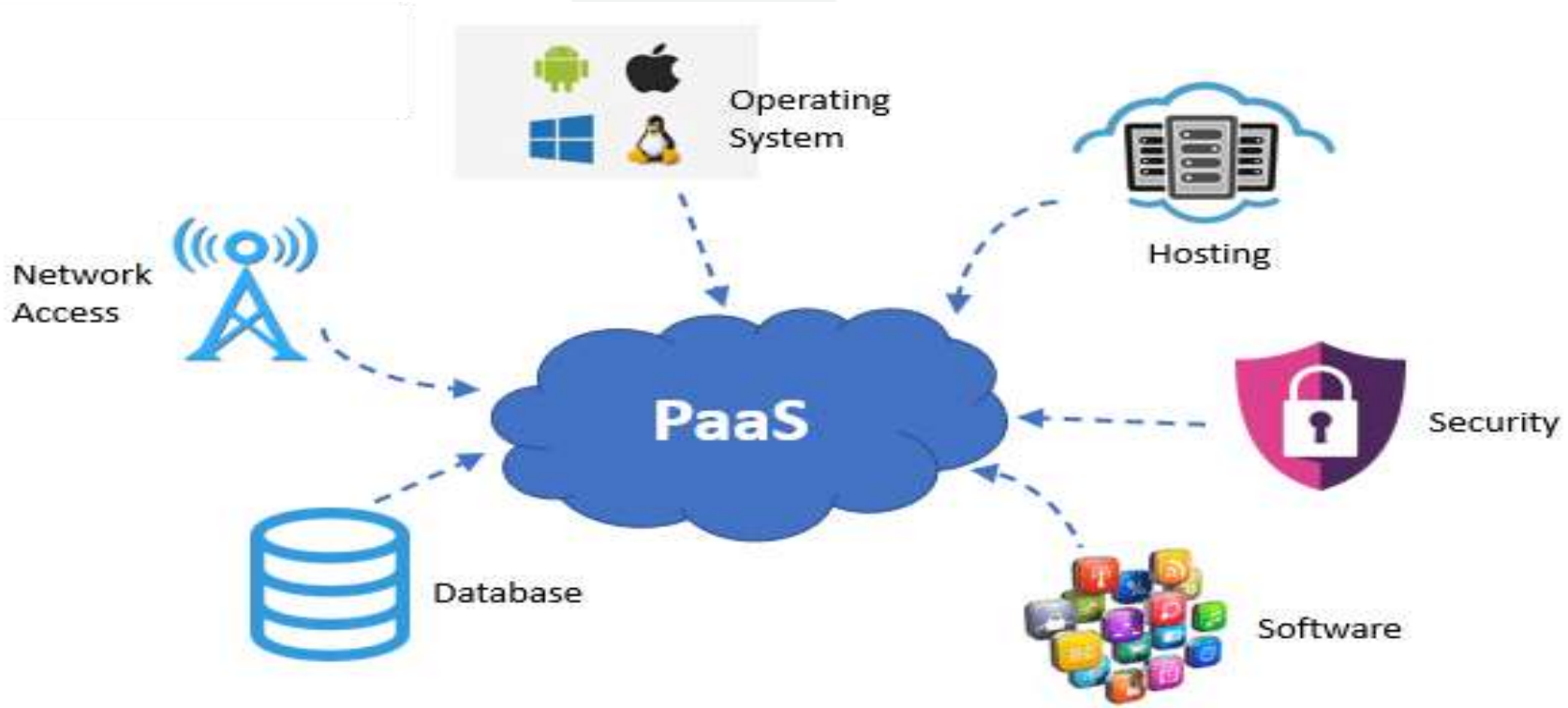
- Platform as a service (PaaS) is a complete development and deployment environment in the cloud, with resources that enable you to deliver everything from simple cloud-based apps to sophisticated, cloud-enabled enterprise applications.

- The Platform as a Service model describes a software environment in which a developer can create customized solutions within the context of the development tools that the platform provides.

- Platforms can be based on specific types of development languages, application frameworks, or other constructs. A PaaS offering provides the tools and development environment to deploy applications on another vendor's application.

- PaaS tool is a fully integrated development environment; that is, all the tools and services are part of the PaaS service. To be useful as a cloud computing offering, PaaS systems must offer a way to create user interfaces, and thus support standards such as HTLM, JavaScript, or other rich media technologies.

Platform as a Service (PaaS)



Platform as a Service (PaaS)

- In a PaaS model, customers may interact with the software to enter and retrieve data, perform actions, get results, and to the degree that the vendor allows it, customize the platform involved.
- The customer takes no responsibility for maintaining the hardware, the software, or the development of the applications and is responsible only for his interaction with the platform.
- The vendor is responsible for all the operational aspects of the service, for maintenance, and for managing the product(s) lifecycle.
- The one example that is most quoted as a PaaS offering is Google's App Engine platform. Developers program against the App Engine using Google's published APIs.
- The tools for working within the development framework, as well as the structure of the file system and data stores, are defined by Google.
- Another example of a PaaS offering is Force.com, Salesforce.com's developer platform for its SaaS offerings.

Platform as a Service (PaaS)

- Force.com is an example of an add-on development environment. A developer might write an application in a programming language like Python using the Google API.
- The vendor of the PaaS solution is in most cases the developer, who is offering a complete solution to the customer.
- Google itself also serves as a PaaS vendor within this system, because it offers many of its Web service applications to customers as part of this service model.
- You can think of **Google Maps, Google Earth, Gmail, and the myriad** of other PaaS offerings as conforming to the PaaS service model, although these applications themselves are offered to customers.
- The difficulty with PaaS is that it **locks the developer** (and the customer) into a solution that is dependent upon the platform vendor.
- An application written in **Python against Google's API using the Google App** Engine is likely to work only in that environment. There is considerable vendor lock-in associated with a PaaS solution.

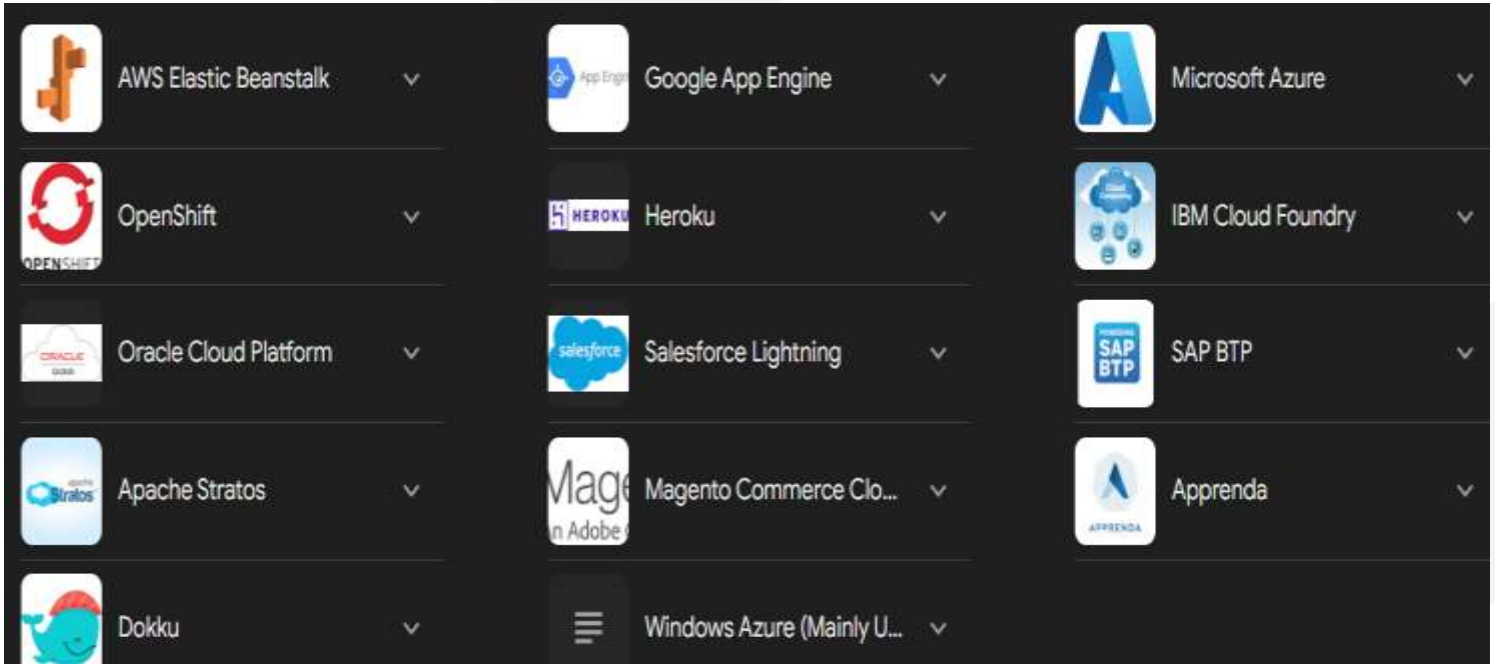
Platform as a Service (PaaS)-Benefits

- **Reduce coding time**: With PaaS development tools, teams can reduce the amount of time spent coding new applications with built-in pre-coded components like workflow, security, search, etc.
- **Increased development capabilities:** PaaS offerings provide development teams with sophisticated tools and added capabilities, without having to bring on new team members to get the job done.
- **Support dispersed or remote teams:** Cloud-native computing environments allow **remote teams to collaborate** and communicate in real-time from different locations.
- **Ability to develop for multiple platforms:** Certain PaaS service providers provide development options for different platforms like computers, mobile, and web browsers.
- **Web application lifecycle support:** With PaaS solutions, development teams have access to all the tools they need to effectively build, test within a virtual machine, deploy, and update apps within an integrated environment.
- **Cost-effective development:** Without having to start from scratch, application development teams that use PaaS can focus on building apps that provide an exceptional user experience. This results in cost savings when it comes to equipment and worker productivity.

Platform as a Service (PaaS)-Benefits

- PaaS is leading the way for a new era in business innovation. Cloud computing is the only way forward for businesses looking to grow in an era of digital transformation.
- Companies big and small should consider transitioning to cloud-based systems for increased flexibility, productivity, and business continuity.
- It offers a cost-effective solution to the development challenges posed by complex infrastructures, allowing businesses to focus their workforce while benefiting from secure data storage, sophisticated tools, and streamlined operations.

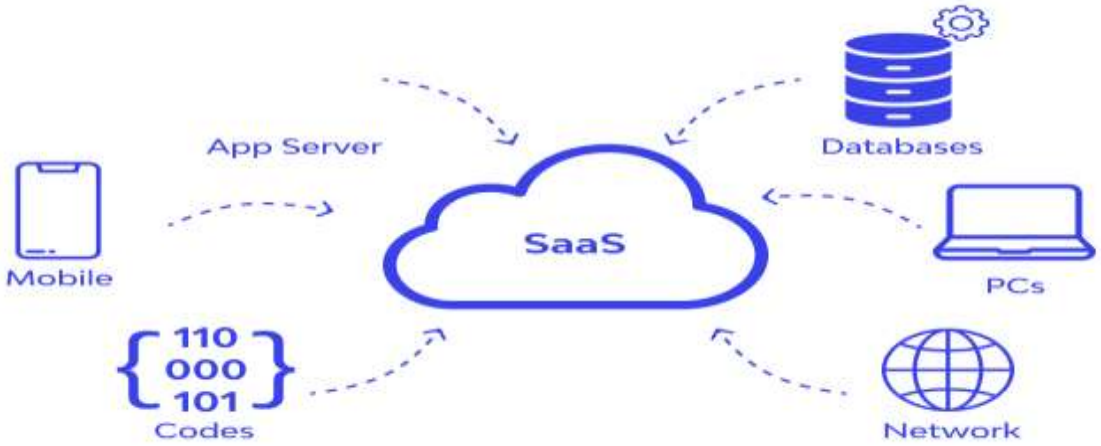
Platform as a Service (PaaS)-Examples



Software as a Service (SaaS)

- SaaS, or Software as a Service, is a cloud-based model for delivering software applications to users over the internet. SaaS is a popular and commonly used cloud computing model.

- Software as a service (SaaS) allows users to connect to and use cloud-based apps over the Internet**. Common examples are email, calendaring, and office tools (such as Microsoft Office 365).** SaaS provides a complete software solution that you purchase on a pay-as-you-go basis from a cloud service provider.



Software as a Service (SaaS)

- The most complete cloud computing service model is one in which the computing hardware and software, as well as the solution itself, are provided by a vendor as a complete service offering.
- It is referred to as the Software as a Service (SaaS) model. SaaS provides the complete infrastructure, software, and solution stack as the service offering
- Software as a Service (SaaS) may be succinctly described as software that is deployed on a hosted service and can be accessed globally over the Internet, most often in a browser.
- With the exception of the user interaction with the software, all other aspects of the service are abstracted away.
- Every computer user is familiar with SaaS systems, which are either replacements or substitutes for locally installed software. Examples of SaaS software for end-users are Google Gmail and Calendar, QuickBooks online, Zoho Office Suite, and others that are equally well known.
- SaaS applications come in all shapes and sizes, and include custom software such as billing and invoicing systems, Customer Relationship Management (CRM) applications, Help Desk applications, Human Resource (HR) solutions, as well as myriad online versions of familiar applications.

Software as a Service (SaaS)-Benefits

- Many people believe that SaaS software is not customizable, and in many SaaS applications this is indeed the case.
- For user-centric applications such as an office suite, that is mostly true; those suites allow you to set only options or preferences.
- However, many other SaaS solutions expose Application Programming Interfaces (API) to developers to allow them to create custom composite applications.
- These APIs may alter the security model used, the data schema, workflow characteristics, and other fundamental features of the service's expression as experienced by the user.
- Examples of an SaaS platform with an exposed API are Salesforce.com and Quicken.com. So SaaS does not necessarily mean that the software is static or monolithic.

Software as a Service (SaaS)

Software as a Service (SaaS)-Benefits

- The software update is not required.
- It does not create expenses like licensing and purchasing the software.
- Server cost is avoided.
- You pay as much as you use the software with no need of purchasing it.
- Since **SaaS** is web-based, you are able to access your data anywhere at any time.
- Needs such as technical maintenance and IT staff disappear.
- You do not need to allocate a budget for security.
- Physical servers are backed up at regular intervals as a precaution against any malfunction.
- The cloud server service is configured in the Cluster infrastructure where multiple computers are running as copies of each other, it continues to run on one of the other redundant servers without interruption when a problem is encountered.

Software as a Service (SaaS)-Characteristics

- The software is available over the Internet globally through a browser on demand.
- The typical license is subscription-based or usage-based and is billed on a recurring basis.
- The software and the service are monitored and maintained by the vendor, regardless of where all the different software components are running.
- There may be executable client-side code, but the user isn't responsible for maintainin that code or its interaction with the service.
- Reduced distribution and maintenance costs and minimal end-user system costs generally make SaaS applications cheaper to use than their shrink-wrapped versions.
- Such applications feature automated upgrades, updates, and patch management and much faster rollout of changes.
- SaaS applications often have a much lower barrier to entry than their locally installed competitors, a known recurring cost, and they scale on demand (a property of cloud computing in general).
- All users have the same version of the software so each user's software is compatible with another's.
- SaaS supports multiple users and provides a shared data model through a single-instance, multi-tenancy model.
- The alternative of software virtualization of individual instances also exists, but is less common.

Software as a Service (SaaS)

Shrink-Wrapped versus SaaS Licensing

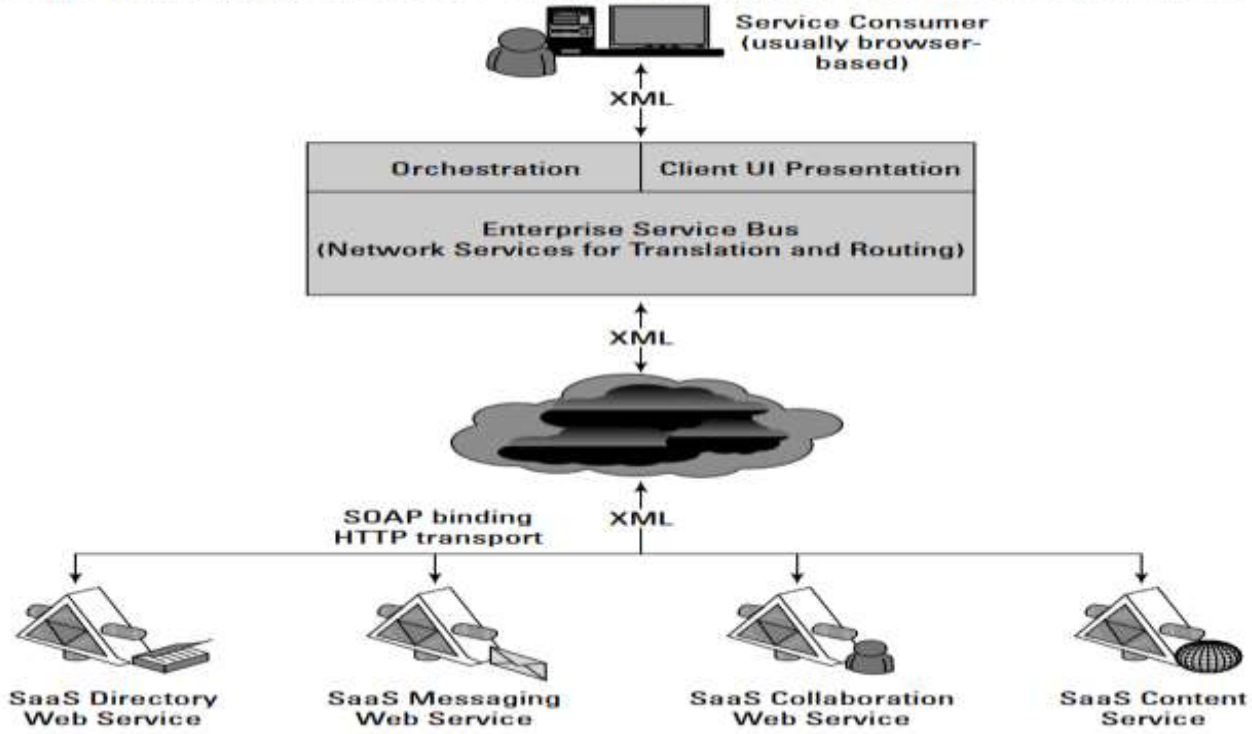
	Shrink-Wrapped Software	Hybrid Model	SaaS
Licensing	Owned	Subscription (flat fee)	Metered subscription
Location	Locally installed	Available through an application	Cloud based
Management	Local IT staff	Application Service Provider (ASP)	Cloud vendor through a Service Level Agreement (SLA)

Open SaaS and SOA

- A considerable amount of SaaS software is based on open source software. When open source software is used in a SaaS, you may hear it referred to as Open SaaS.
- The advantages of using open source software are that systems are much cheaper to deploy because you don't have to purchase the operating system or software, there is less vendor lock-in, and applications are more portable.
- The popularity of open source software, from Linux to APACHE, MySQL, and Perl (the LAMP platform) on the Internet, and the number of people who are trained in open source software make Open SaaS an attractive proposition.
- The impact of Open SaaS will likely translate into better profitability for the companies that deploy open source software in the cloud, resulting in lower development costs and more robust solutions.

Open SaaS and SOA

A modern implement of SaaS using an Enterprise Service Bus and architected with SOA components



Open SaaS and SOA

- The componentized nature of SaaS solutions enables many of these solutions to support a feature called mashups.
- A mashup is an application that can display a Web page that shows data and supports features from two or more sources.
- Annotating a map such as Google maps is an example of a mashup. Mashups are considered one of the premier examples of Web 2.0, and that is technology's ability to support social network systems.
- A mashup requires three separate components: An interactive user interface, which is usually created with HTML/XHTML, Ajax, JavaScript, or CSS.
- Web services that can be accessed using an API, and whose data can be bound and transported by Web service protocols such as SOAP, REST, XML/HTTP, XML/RPC, and JSON/RPC.
- Data transfer in the form of XML, KML (Keyhole Markup Language), JSON (JavaScript Object Notation), or the like.

Open SaaS and SOA

- Mashups are an incredibly useful hybrid Web application, one that SaaS is a great enabler for. The Open Mashup Alliance (OMA; see http://www.openmashup.org/) is a non-profit industry group dedicated to supporting technologies that implement enterprise mashups.

- This group supports the developing standard, the Enterprise Mashup Markup Language (EMML), which is a Domain Specific Language (DSL). This group predicts that the use of mashups will grow by a fac- tor of 10 within just a few years.

- Gartner Group predicts that approximately 25 percent of all software sold by 2011 will use the SAAS model, offered either by vendors or an intermediary party, sometimes referred to as an aggregator.

- An aggregator bundles SaaS applications from different vendors and presents them as part of a unified platform or solution.

Salesforce.com and CRM SaaS

- The best-known example of Software as a Service (SaaS) is the Customer Relationship Management software offered by Salesforce.com whose solution offers sales, service, support, marketing, content, analytical analysis, and even collaboration through a platform called Chatter.
- Salesforce.com was founded in 1999 by a group of Oracle executives and early adopters of many of the technologies that are becoming cloud computing staples.
- Salesforce.com extended its SaaS offering to allow developers to create add-on applications, essentially turning the SaaS service into a Platform as a Service (PaaS) offering called the Force.com
- Platform. Applications built on Force.com are in the form of the Java variant called Apex using an XML syntax for creating user interfaces in HTML, Ajax, and Flex.
- Nearly a thousand applications now exist for this platform from hundreds of vendors.

Salesforce.com and CRM SaaS

Salesforce.com is the largest SaaS provider of CRM software and a pioneer in this type of cloud computing software. This is the company's home page.

Assessing the Role of Open Standards

- Open standards determine the format, storage, and exchange of data and enable different organizations and systems to communicate seamlessly.
- Open Standards" are standards made available to the general public and are developed (or approved) and maintained via a collaborative and consensus driven process. "Open Standards" facilitate interoperability and data exchange among different products or services and are intended for widespread adoption.
- Open standards allow others to make compatible products, so they're not locked into just one company's software or hardware. They promote compatibility, interoperability, and innovation.
- **Open standards can help with interoperability**, which is often a requirement for cloud environments.
- Interoperability can be viewed in two dimensions:
- **Horizontal**: Between two different applications
- **Vertical**: Within the components of a single application.

Assessing the Role of Open Standards

- Cloud computing's promise of scalability completely changes the manner in which services an applications are deployed. Without standards, the industry creates proprietary systems with vendor lock-in, sometimes referred to as **"stovepipe" clouds**.

- Clients do not want to be locked into any single system, there is a strong industry push to create standards-based clouds.

The cloud computing industry is working with these **architectural standards:**

- Platform virtualization of resources
- Service-oriented architecture
- Web-application frameworks
- Deployment of open-source software
- Standardized Web services
- Autonomic systems
- Grid computing
- These standards help to enable different business models that cloud computing vendors can support, most notably Software as a Service (SaaS), Web 2.0 applications, and utility computing.
- These businesses require open standards so that data is both portable and universally accessible.

Assessing the Role of Open Standards

- The race to create the first generation of open cloud platform technologies that will compete with proprietary technologies offered by companies such as Microsoft (Azure Platform) and VMware (vSphere) is already underway.

- Rackspace.com, one of the large IaaS cloud service providers, announced in July 2010 that it is initiating an open-source project called OpenStack that will begin with the code used to run its Cloud Files and Cloud Servers technologies.

- NASA has also donated some of the Nebula Cloud Platform technology that it developed. The software developed will be released under the Apache 2.0 license. Founding members of this project include AMD, Citrix, Dell, Intel, NTT Data, and several other cloud service providers.

- openStack.org's home page (http://www.openstack.org/).

Assessing the Role of Open Standards

- Eucalyptus (http://open.eucalyptus.com/) is a Linux-based software platform for creating cloud computing IaaS systems based on computer clusters.
- The project has an interface that can connect to Amazon's compute and storage cloud systems (EC2 and S3), and it maintains a private cloud as a sandbox for developers to work in. Eucalyptus works with a number of technologies for system virtualization, including VMware, Xen, and KVM. Eucalyptus is an acronym taken from the expression "Elastic Utility Computing Architecture for Linking Your Programs to Useful Systems."
- Most of the major Linux vendors support this project, which is based on the original work of Rich Wolski at the University of California at Santa Barbara.
- The company Eucalyptus Systems was formed in 2009 to support the commercialization of the Eucalyptus Cloud Computing Platform.
- OpenStack and Eucalyptus are by no means unique; several other projects are underway to create open-source cloud platforms. There also are numerous research projects in the area.
- The IEEE Technical Committee on Services Computing (http://tab.computer.org/tcsc/) sponsors a conference in this area called CLOUD and has some working groups and publications in this area.

Assessing the Role of Open Standards

- The first two deliverables of the project are a distributed object store based on Rackspace Cloud Files and a scalable machine provisioning technology based on NASA Nebula and Rackspace Cloud Servers.
- OpenStack Compute software will automatically create large groups of virtual private servers on industry-standard systems.
- OpenStack Storage is the software that will create redundant object-based storage using clusters of commodity servers and storage systems.

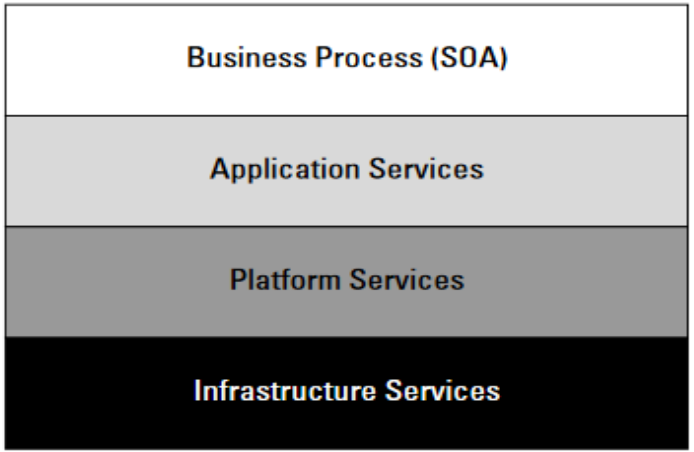
OpenStack.org is an industry group seeking to create open cloud standards based on Rackspace.com and NASA technologies.

Infrastructure as a Service (IaaS)

- Cloud computing into four layers that form a cloud computing ecosystem.
- The Application layer forms the basis for Software as a Service (SaaS),
- The Platform layer forms the basis for Platform as a Service (PaaS) model.
- Infrastructure as a Service (IaaS) creates what may be determined to be a utility computing model, something that you can tap into and draw from as you need it without significant limits on the scalability of your deployment.
- You pay only for what you need when you need it. IaaS may be seen to be an incredibly disruptive technology, one that can help turn a small business into a large business nearly overnight.
- This is a most exciting prospect; one that is fuelling a number of IaaS start-ups during one of the most difficult recessions of recent memory.

Infrastructure as a Service (IaaS)

The cloud computing ecosystem



Infrastructure as a Service (IaaS)

- **Infrastructure as a Service (IaaS)** is a cloud computing service model in which hardware is virtualized in the cloud. In this particular model, the service vendor owns the equipment: servers, storage, network infrastructure, and so forth.
- The developer creates virtual hardware on which to develop applications and services. Essentially, an IaaS vendor has created a hardware utility service where the user provisions virtual resources as required.
- The developer interacts with the IaaS model to create virtual private servers, virtual private storage, virtual private networks, and so on, and then populates these virtual systems with the applications and services it needs to complete its solution.
- In IaaS, the virtualized resources are mapped to real systems. When the client interacts with an IaaS service and requests resources from the virtual systems, those requests are redirected to the real servers that do the actual work.

Infrastructure as a Service (IaaS)

- The fundamental unit of virtualized client in an IaaS deployment is called a **workload.** A workload simulates the ability of a certain type of real or physical server to do an amount of work.
- The work done can be measured by the number of Transactions Per Minute (TPM) or a similar metric against a certain type of system. A workload has certain other attributes such as Disk I/Os measured in Input/Output Per Second IOPS, the amount of RAM consumed under load in MB, network throughput and latency, and so forth.
- In a hosted application environment, a client's application runs on a dedicated server inside a server rack or perhaps as a standalone server in a room full of servers.
- In cloud computing, a provisioned server called an instance is reserved by a customer, and the necessary amount of computing resources needed to achieve that type of physical server is allocated to the client's needs.
- Figure shows how three virtual private server instances are partitioned in an IaaS stack. The three workloads require three different sizes of computers**: small, medium, and large.**
- A client would reserve a machine equivalent required to run each of these workloads. The IaaS infrastructure runs these server instances in the data center that the service offers, drawing from a pool of virtualized machines, RAID storage, and network interface capacity.

Infrastructure as a Service (IaaS)

- A client would reserve a machine equivalent required to run each of these workloads. The IaaS infrastructure runs these server instances in the data center that the service offers, drawing from a pool of virtualized machines, "**redundant array of inexpensive disks**" or "**redundant array of independent disks (**RAID) storage, and network interface capacity.
- These three layers are expressions of physical systems that are partitioned as logical units. **Logical Unit Number(LUNs)** the cloud interconnect layer, and the virtual application software layer are logical constructs.

- LUNs are logical storage containers, the cloud interconnect layer is a virtual network layer that is assigned IP addresses from the IaaS network pool, and the virtual application software layer contains software that runs on the **physical VM instance(s)** that have been partitioned from physical assets on the IaaS' private cloud.

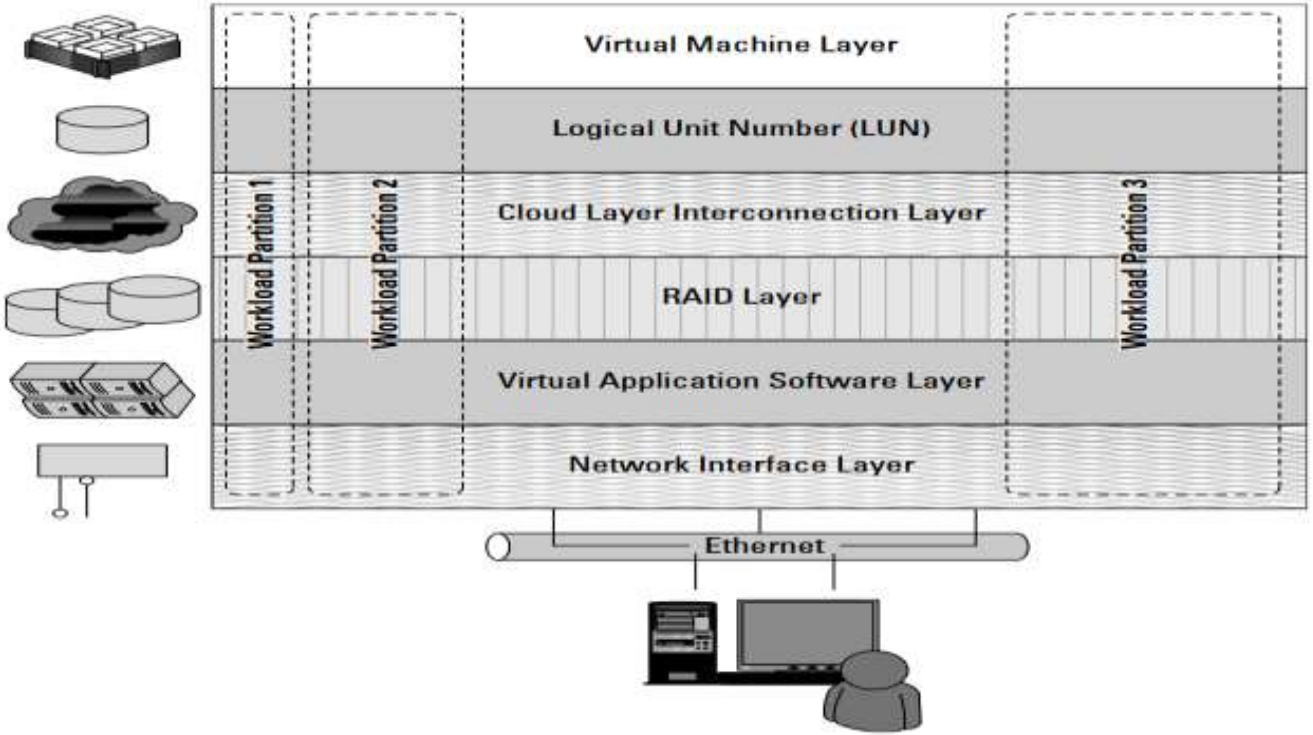
- From an architectural standpoint, the client in an IaaS infrastructure is assigned its own private network. The Amazon **Elastic Computer Cloud (EC2),** behaves as if each server is its own separate network—unless you create your own Virtual Private Cloud (an EC2 add-on feature), which provides a workaround to this problem.

- When you scale your EC2 deployment, you are adding additional networks to your infrastructure, which makes it easy to logically scale an EC2 deployment, but imposes additional network overhead because traffic must be routed between logical networks.

Infrastructure as a Service (IaaS)

-

A virtual private server partition in an IaaS cloud



Infrastructure as a Service (IaaS)

- Amazon Web Service's routing limits broadcast and multicast traffic because Layer-2 (Data Link) networking is not supported. Rackspace **Cloud (http://www.rackspacecloud.com/)** follows the AWS IP assignment model.

- Other IaaS infrastructures such as the one Cloudscaling.com (http://www.cloudscaling.com) offers or a traditional VMWare cloud-assigned networks on a per-user basis, which allows for Level 2 networking options.

- The most prominent Level 2 protocols that you might use are tunneling options, because they enable VLANs.

- Consider a transactional eCommerce system, for which a typical stack contains the following components:

- Web server
- Application server
- File server
- Database
- Transaction engine

- This eCommerce system has several different workloads that are operating: queries against the database, processing of business logic, and serving up clients' Web pages.

Infrastructure as a Service (IaaS)

- The classic example of an IaaS service model is **Amazon.com's Amazon Web Services (AWS).** AWS has several data centers in which servers run on top of a virtualization platform (Xen) and may be partitioned into logical compute units of various sizes.

- Developers can then apply system images containing different operating systems and applications or create their own system images.

- Storage may be partitions, databases may be created, and a range of services such a messaging and notification can be called upon to make distributed application work correctly.

Cross-Ref

- Amazon Web Services offers a classic Service Oriented Architecture (SOA) approach to IaaS.

Pods, aggregation, and silos

- **Workloads support** a certain number of users, at which point you exceed the load that the instance sizing allows. When you reach the limit of the largest virtual machine instance possible, you must make a copy or clone of the instance to support additional users.
- A group of users within a particular instance is called a **Pod**. Pods are managed by a Cloud Control System (CCS). In AWS, the CCS is the AWS Management Console.

Infrastructure as a Service (IaaS)

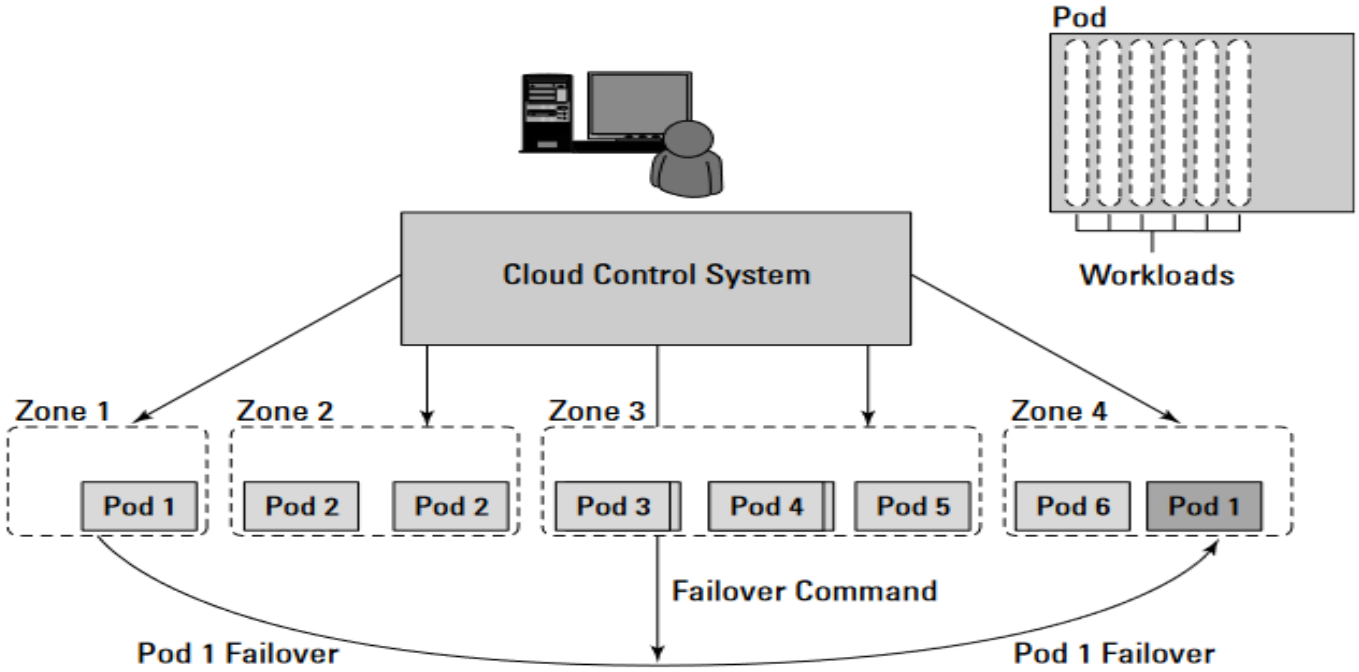
- Sizing limitations for pods need to be accounted for if you are building a large cloud-based application. Pods are aggregated into pools within an IaaS region or site called an availability zone.
- In very large cloud computing networks, when systems fail, they fail on a pod-by-pod basis, and often a zone-by-zone basis.
- For AWS' IaaS infrastructure, the availability zones are organized around the company's data centers in Northern California, Northern Virginia, Ireland, and Singapore.
- A failover system between zones gives IaaS private clouds a very high degree of availability.
- Figure shows how pods are aggregated and virtualized in IaaS across zones.
- When a cloud computing infrastructure isolates user clouds from each other so the management system is incapable of interoperating with other private clouds, it creates an information silo, or simply a silo.
- Most often, the term silo is applied to PaaS offerings such as Force.com or QuickBase, but silos often are an expression of the manner in which a cloud computing infrastructure is architected.

Infrastructure as a Service (IaaS)

- **Silos** are the cloud computing equivalent of compute islands: They are processing domains that are sealed off from the outside.
- When you create a private virtual network within an IaaS framework, the chances are high that you are creating a Silo.
- **Silos impose restrictions on interoperability** that runs counter to the open nature of build-componentized service-oriented applications.
- However, that is not always a bad thing. A silo can be its own ecosystem; it can be protected and secured in ways that an open system can't be. Silos just aren't as flexible as open systems and are subject to vendor lock-in.

Infrastructure as a Service (IaaS)

Pods, aggregation, and failover in IaaS



Cloud -Types

- Cloud computing has two distinct sets of models: **NSIT, Cloud Cube Model**

Deployment models:

- This refers to the location and management of the cloud's infrastructure.

Service models:

- This consists of the particular types of services that you can access on a cloud computing platform.

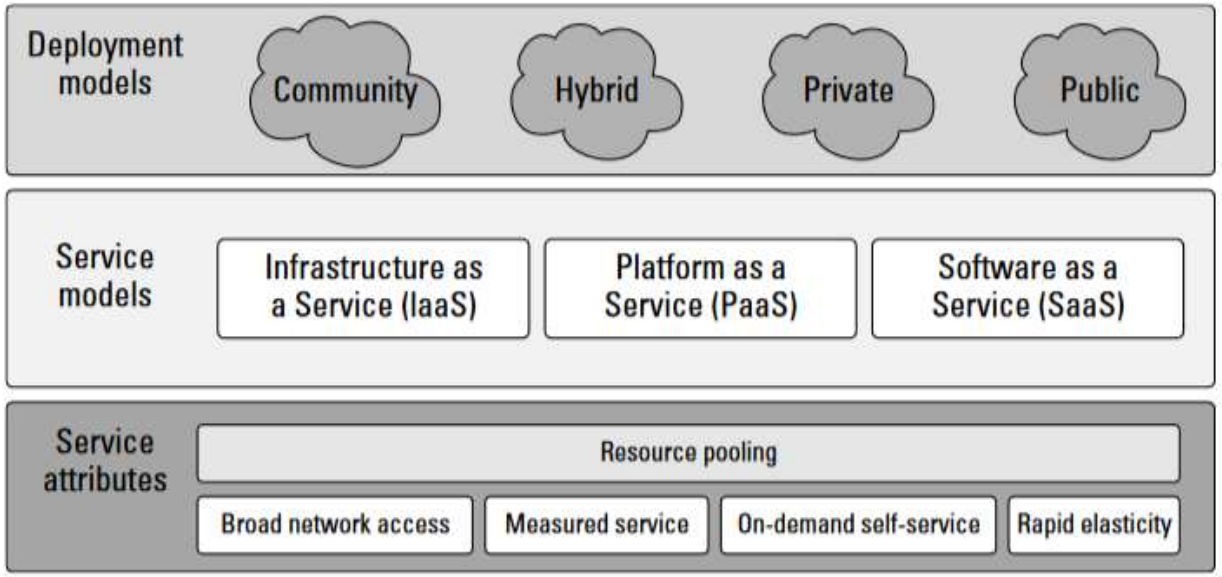
NIST model

- The United States government is a major consumer of computer services and, therefore, one of the major users of cloud computing networks.
- The **U.S. National Institute of Standards and Technology (NIST)** has a set of working definitions **(http://csrc.nist.gov/groups/SNS/cloud-computing/cloud-def-v15.doc)** that separate cloud computing into service models and deployment models.

- The NIST model originally did not require a cloud to use virtualization to pool resources, nor did it absolutely require that a cloud support multi-tenancy in the earliest definitions of cloud computing.
- Multi-tenancy is the sharing of resources among two or more clients. The latest version of the NIST definition does require that cloud computing networks use virtualization and support multi-tenancy.

Cloud –NSIT Model

- The NIST cloud model doesn't address a number of intermediary services such as transaction or service brokers, provisioning, integration, and interoperability services that form the basis for many cloud computing discussions.



Cloud Cube Model

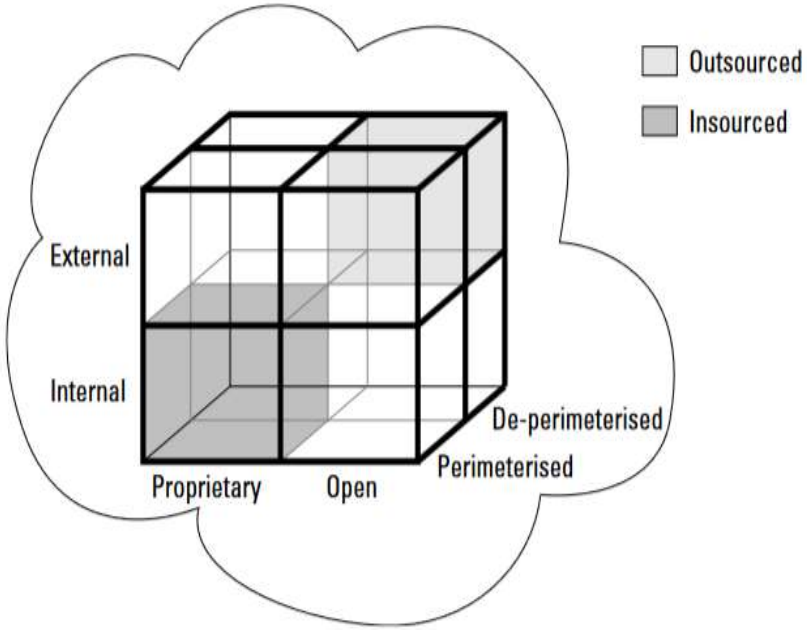
- The Open Group maintains an association called the Jericho Forum (https://www.opengroup.org/jericho/index.htm) whose main focus is how to protect cloud networks.
- The group has an interesting model that attempts to categorize a cloud network based on four dimensional factors.
- Paper-**"Cloud Cube Model:** Selecting Cloud Formations for Secure Collaboration" **(http://www.opengroup.org/jericho/cloud_cube_model_v1.0.pdf),** the type of cloud networks you use dramatically changes the notion of where the boundary between the client's network and the cloud begins and ends.

The four dimensions of the **Cloud Cube Model** are:

- **Physical location of the data:** Internal (I) / External (E) determines your organization's boundaries.
- **Ownership:** Proprietary (P) / Open (O) is a measure of not only the technology ownership, but of interoperability, ease of data transfer, and degree of vendor application lock-in.
- **Security boundary**: Perimeterised (Per) / De-perimiterised (D-p) is a measure of whether the operation is inside or outside the security boundary or network firewall.
- **Sourcing:** Insourced or Outsourced means whether the service is provided by the customer or the service provider.

Cloud Cube Model

The Jericho Forum's Cloud Cube Model



- The fourth dimension corresponds to two different states in the eight possible cloud forms
  Per (IP, IO, EP, EO)
  and D- p (IP, IO, EP, EO).
- The sourcing dimension addresses the deliverer of the service.
- What the Cloud Cube Model is meant to show is that the traditional notion of a network boundary being the network's firewall no longer applies in cloud computing.

Deployment models

A deployment model defines the purpose of the cloud and the nature of how the cloud is located.
The NIST definition for the **four deployment models:**

Public cloud: The public cloud infrastructure is available for public use alternatively for a large industry group and is owned by an organization selling cloud services.

Private cloud: The private cloud infrastructure is operated for the exclusive use of an organization. The cloud may be managed by that organization or a third party. Private clouds may be either on- or off-premises.

Hybrid cloud:

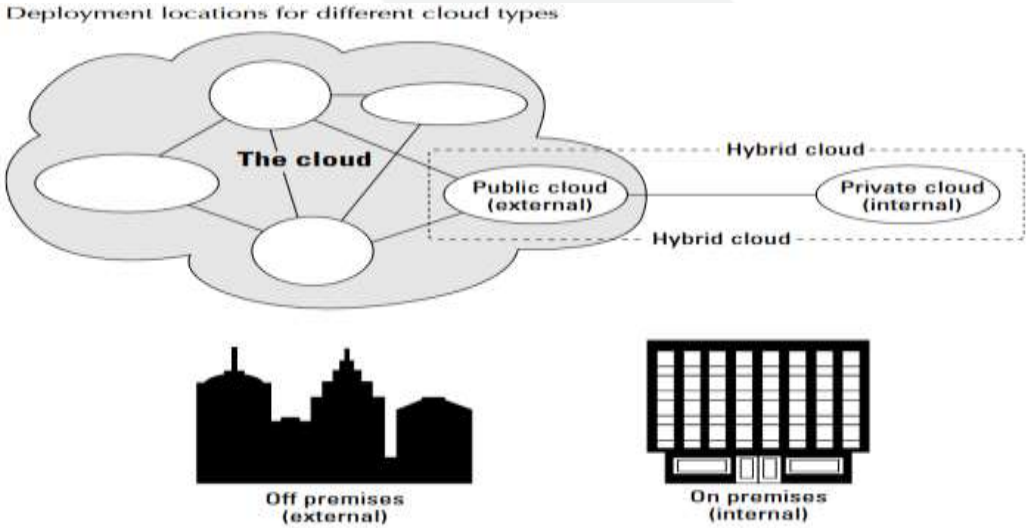
- A hybrid cloud combines multiple clouds (private, community of public) where those clouds retain their unique identities, but are bound together as a unit.

- Hybrid cloud may offer standardized or proprietary access to data and applications, as well as application portability.

Deployment models

Community Cloud:

- It is one where the cloud has been organized to serve a common function or purpose.
- It may be for one organization or for several organizations, but they share common concerns such as their mission, policies, security, regulatory compliance needs, and so on.
- A community cloud may be managed by the constituent organization(s) or by a third party.



Deployment models

Community Cloud:

- The United States Government, under the auspices of the General Services Administrator (GSA), launched a cloud computing portal called Apps.gov.
- The purpose of providing cloud services to federal agencies. Described under the "U.S. Federal Cloud Computing initiative **(http://www.scribd.com/doc/17914883/US-Federal-Cloud-Computing-Initiative-RFQ-GSA).**
- **T**he goal of the initiative is to make large portions of the federal govern- ment's apparatus available under a cloud computing model.
- This is a good example of a community cloud deployment, with the government being the community.

Deployment models

- Apps.gov is the U.S. government's cloud computing system for its various agencies.

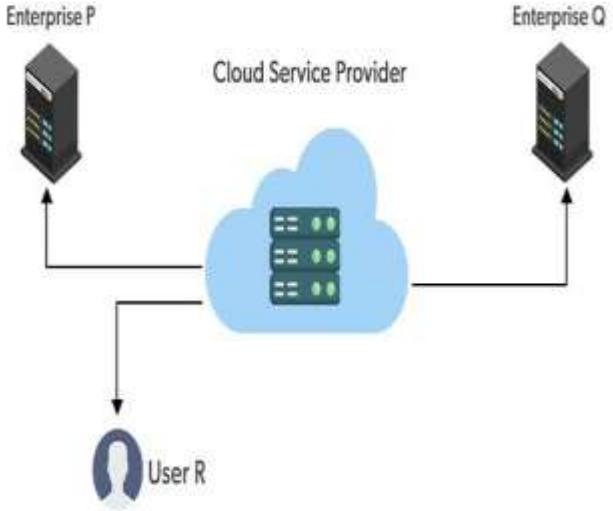
Service Models

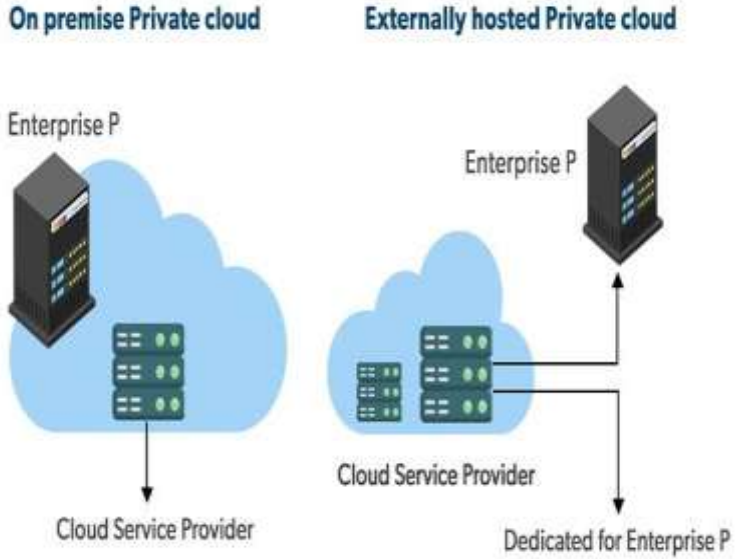
- Cloud computing has developed, different vendors offer clouds that have different services associated with them.
- There are many different service models described in the literature, all of which take the following form:
- XaaS, or "<Something> as a Service"
- Three service types have been universally accepted:

Infrastructure as a Service (IaaS**):**

- IaaS provides virtual machines, virtual storage, virtual infrastructure, and other hardware assets as resources that clients can provision.
- The IaaS service provider manages all the infrastructure, while the client is responsible for all other aspects of the deployment. This can include the operating system, applications, and user interactions with the system.

Cloud Deployment Models





Cloud Deployment Models

Factors	Public Cloud	Private Cloud	Hybrid Cloud
Resources	Resources are shared among multiple customers	Resources are shared with a single organization	It is a combination of public and private clouds. based on the requirement.
Tenancy	Data of multiple organizations is stored in the public cloud	Data of a single organization is stored in a clouds the public cloud	Data is stored in the public cloud, and provide security in the public cloud.
Pay Model	Pay what you used	Have a variety of pricing models	It can include a mix of public cloud pay-as-you-go pricing, and private cloud fixed pricing. It has other pricing models such as consumption-based, subscription-based, etc.
Operated by	Third-party service provider	Specific organization	Can be a combination of both
Scalability and Flexibility	It has more scalability and flexibility,	It has predictability and consistency	It has scalability and flexibility by allowing organizations to use a combination of public and private cloud services.
Expensive	less expensive	More expensive	Can be more expensive, but it can also be less expensive , depending on the specific needs and requirements of the organization.
Availability	The general public (over the internet)	Restricted to a specific organization	Can be a combination of both.

Service Models

Platform as a Service (Paas):

PaaS provides virtual machines, operating systems, applications, services, development frameworks, transactions, and control structures.

The client can deploy its applications on the cloud infrastructure or use applications that were programmed using languages and tools that are supported by the PaaS service provider.

The service provider manages the cloud infrastructure, the operating systems, and the enabling software. The client is responsible for installing and managing the application that it is deploying.

Software as a Service (SaaS)

It is a complete operating environment with applications, management, and the user interface.

- In the SaaS model, the application is provided to the client through a thin client interface (a browser, usually), and the customer's responsibility begins and ends with entering and managing its data and user interaction.
- Everything from the application down to the infrastructure is the vendor's responsibility.

Service Models

- The three different service models taken together have come to be known as the SPI model of cloud computing. Many other service models have been mentioned: StaaS, Storage as a Service;
- IdaaS, Identity as a Service; CmaaS, Compliance as a Service; and so forth. However, the SPI services encompass all the other possibilities.
- It is useful to think of cloud computing's service models in terms of a hardware/software stack. One such representation called the Cloud Reference Model.
- At the bottom of the stack is the hardware or infrastructure that comprises the network. As you move upward in the stack, each service model inherits the capabilities of the service model beneath it.
- IaaS has the least levels of integrated functionality and the lowest levels of integration, and SaaS has the most examples of IaaS service providers include: Amazon Elastic Compute Cloud (EC2), Eucalyptus, GoGrid, FlexiScale, Linode, Rack Space Cloud, Terremark.

Service Models

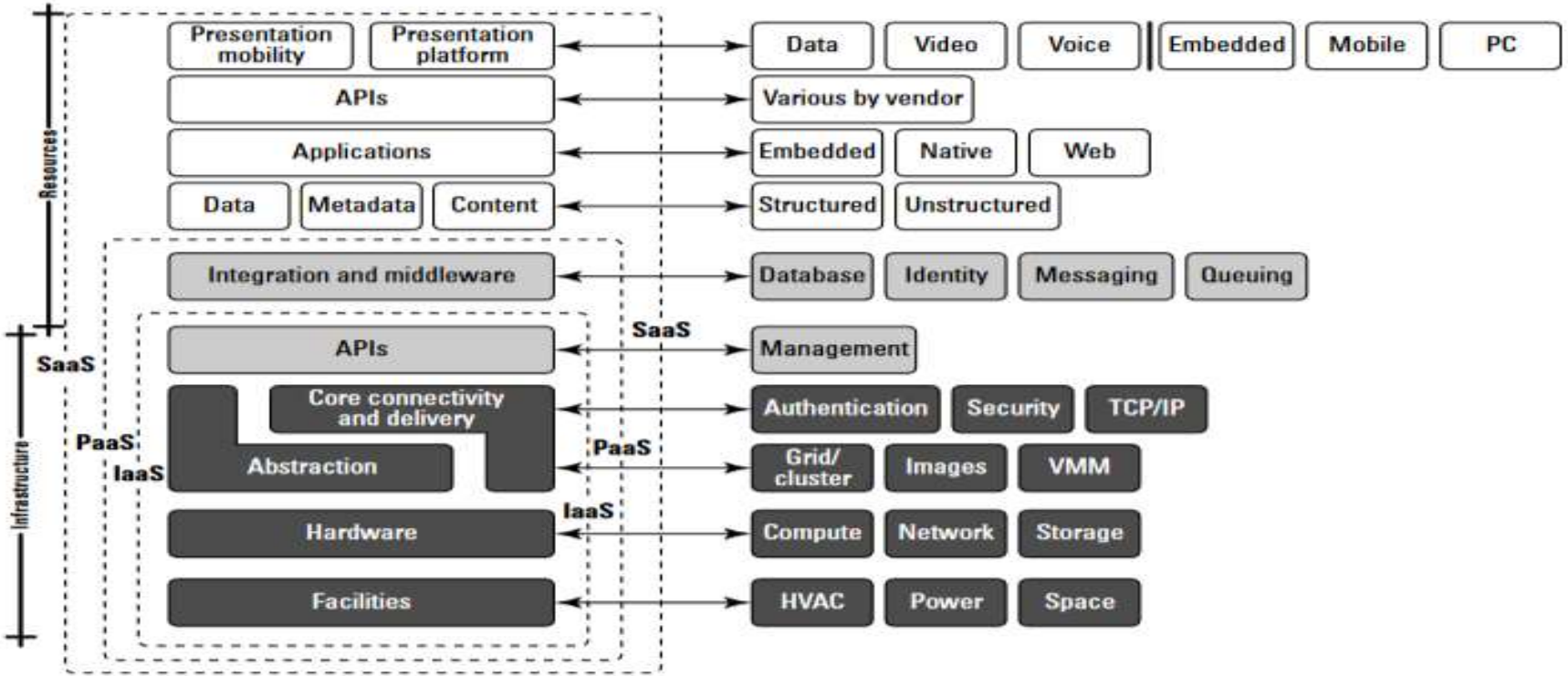
- These vendors offer direct access to hardware resources. On Amazon EC2, considered the classic IaaS example, a client would provision a computer in the form of a virtual machine image, provision storage, and then go on to install the operating system and applications onto that virtual system.

- Amazon has a number of operating systems and some enterprise applications that they offer on a rental basis to customers in the form of a number of canned images, but customers are free to install whatever software they want to run.

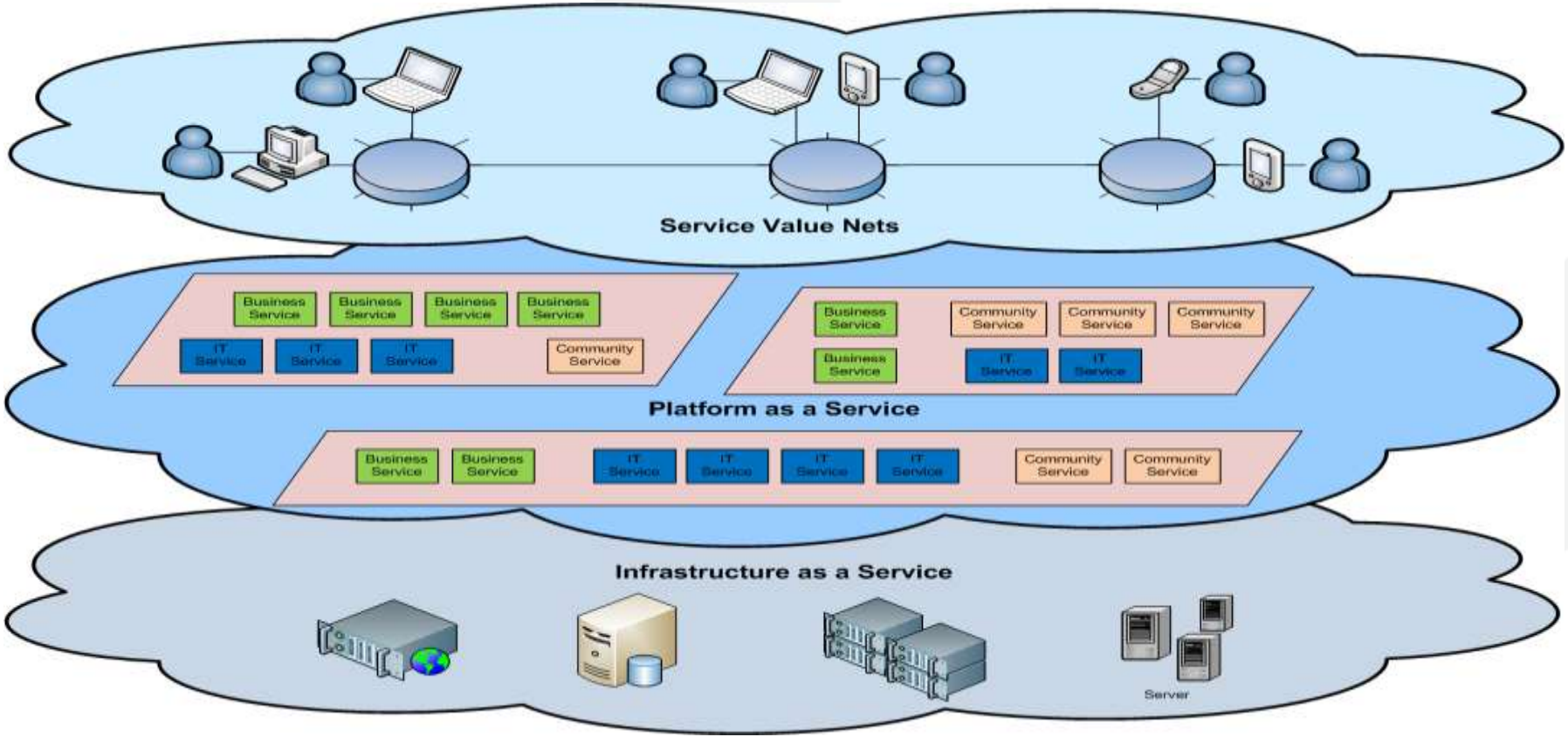
- Amazon's responsibilities as expressed in its Service Level Agreement, which is published on Amazon's Web site, contractually obligates Amazon to provide a level of performance commensurate with the type of resource chosen, as well as a certain level of reliability as measured by the system's uptime.

Service Models

The Cloud Reference Model



Cloud Service Models



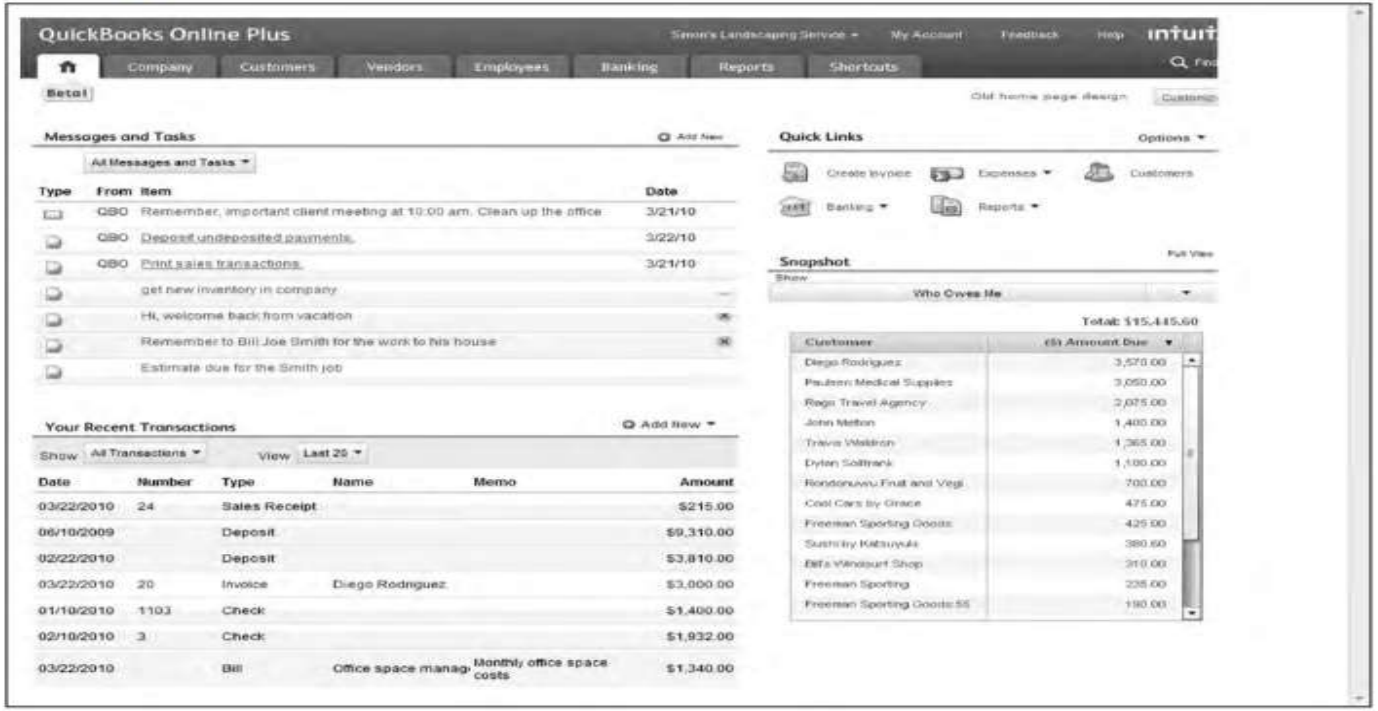
Service Models

- A PaaS service adds integration features, middleware, and other orchestration and choreography services to the IaaS model.
- Examples of PaaS services are:Force.com, GoGrid CloudCenter, Google AppEngine, Windows Azure Platform.
- When a cloud computing vendor offers software running in the cloud with use of the application on a pay-as-you-go model, it is referred to as SaaS.
- With SaaS, the customer uses the application as needed and is not responsible for the installation of the application, its maintenance, or its upkeep.
- A good example of an SaaS offering is an online accounting package, with the online versions of Quicken and Quickbooks a prime example.

Service Models

- Figure shows a home page for QuickBooks Online plus on the Intuit.com Web site.

A home page for a Quickbooks customer on the Intuit.com Web site is an example of an SaaS service.



Service Models

- A client using an SaaS service might as is the case for Quickbooks online log into the service from his browser, create an account, and enter data into the system.

- Intuit.com has a service agreement that not only covers the performance of the hardware and software, but extends to protecting the data that they store for clients, and other fundamental characteristics.

- Other good examples of SaaS cloud service providers are: GoogleApps, Oracle On Demand SalesForce.com, SQL Azur.

- These service model classifications start to get confusing rather quickly when you have a cloud service provider that starts out offering services in one area and then develops services that are classified as another type.
- For example, SalesForce.com started out as a Customer Relationship Management SaaS platform that allowed clients to add their own applications.
- Over time SalesForce.com opened an API called the Force API that allowed developers to create applications based on the SalesForce.com technologies. Force.com is thus their PaaS service.

Service Models

- As another example, take the PaaS offering that is the Windows Azure Platform.
- Windows Azure Platform allows .NET developers to stage their applications on top of Microsoft's infrastructure so that any application built with the .NET Framework can live locally, in Microsoft's cloud network or some combination thereof.
- As Microsoft adds enterprise applications to its cloud service portfolio, as it has in the case of SQL Azure (and many other enterprise applications to come), these offerings fall under the rubric of being an SaaS service model.

Service Models-Differences

Basis Of	IAAS	PAAS	SAAS
Stands for	Infrastructure as a service.	Platform as a service.	Software as a service.
Uses	IAAS is used by network architects.	PAAS is used by developers.	SAAS is used by the end user.
Access	IAAS gives access to the resources like virtual machines and virtual storage.	PAAS gives access to run time environment to deployment and development tools for application.	SAAS gives access to the end user.
Model	It is a service model that provides virtualized computing resources over the internet.	It is a cloud computing model that delivers tools that are used for the development of applications.	It is a service model in cloud computing that hosts software to make it available to clients.
Technical understanding.	It requires technical knowledge.	Some knowledge is required for the basic setup.	There is no requirement about technicalities company handles everything.
Popularity	It is popular among developers and researchers.	It is popular among developers who focus on the development of apps and scripts.	It is popular among consumers and companies, such as file sharing, email, and networking.
Percentage rise	It has around a 12% increment.	It has around 32% increment.	It has about a 27 % rise in the cloud computing model.
Usage	Used by the skilled developer to develop unique applications.	Used by mid-level developers to build applications.	Used among the users of entertainment.
Cloud services.	Amazon Web Services, sun, vCloud Express.	Facebook, and Google search engine.	MS Office web, Facebook and Google Apps.
Enterprise services.	AWS virtual private cloud.	Microsoft Azure.	IBM cloud analysis.
Outsourced cloud services.	Salesforce	Force.com, Gigaspaces.	AWS, Terremark
User Controls	Operating System, Runtime, Middleware, and Application data	Data of the application	Nothing
Others	It is highly scalable and flexible.	It is highly scalable to suit the different businesses according to resources.	It is highly scalable to suit the small, mid and enterprise level business

Cloud Computing

- **Cloud Computing** is model for enabling network users on **demand access to share pool** of configurable computing resources that can be **rapidly provisioned** and release to the clients without **direct service provider interaction.**

- **Cloud Computing** is a general term used to describe a **new class of network based computing** that takes place over the Internet- Using different resources such as servers, storage, databases, software and applications are provided as a service to users on-demand. Users can access these resources an a pay-as-you-utilized them.

- A collection / group of integrated and networked hardware, software and Internet infrastructure- Platform. Using the Internet for communication and transport provides hardware, software and networking services to clients.

- These platforms hide the complexity and details of the underlying infrastructure from users and applications by providing very simple graphical interface or Applications Programming Interface.

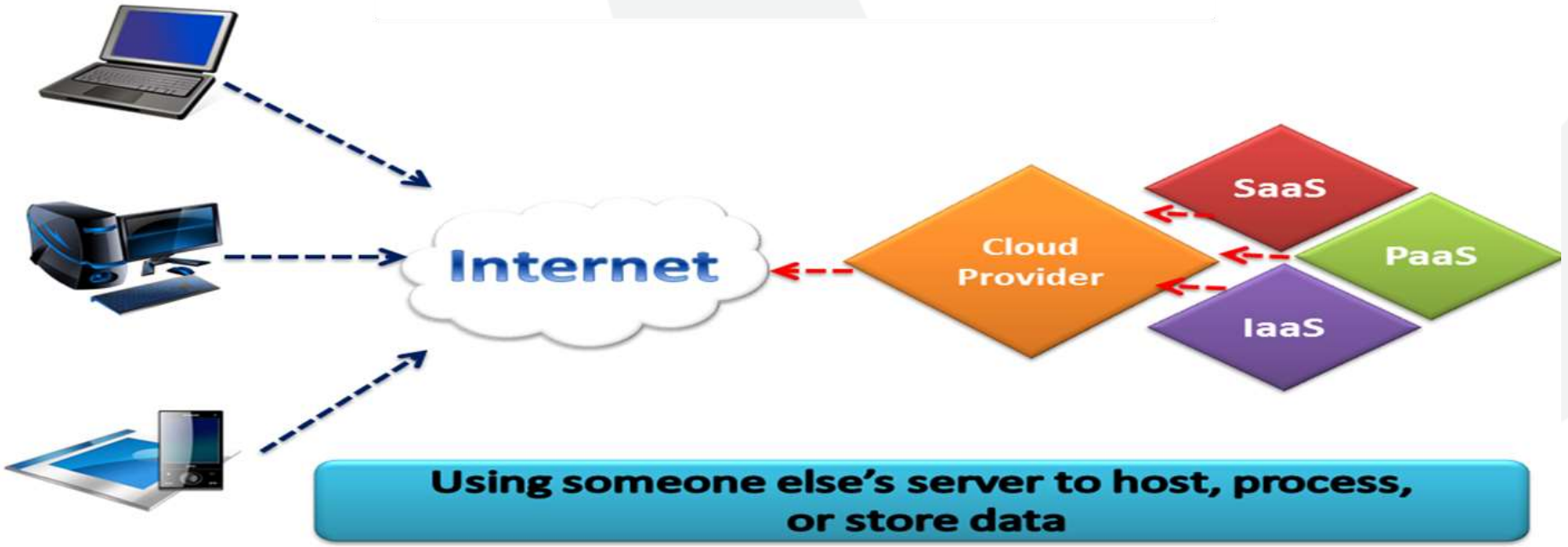
Characteristics :

- On-demand self-service, Scalability, Flexibility
- Broad network access, Reliability
- Resource pooling, Security, Collaboration
- Rapid elasticity, Automation, Real-time analytics
- Measured service, Compliance and Governance
- Multi-tenancy, High performance Computing

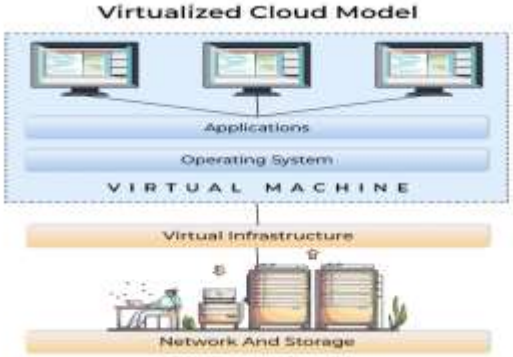
Applications :

- Data Storage and Management
- Web and Mobile applications
- Big data analytics
- Internet of Things
- Artificial Intelligence and Machine learning
- Online Education and learning platforms
- Social media and collaboration tools
- Gaming and virtual reality
- Cyber security and threat intelligence
- CRM, SCM, FM and Accounting, Health care and Medical Research

Cloud Computing



Cloud -Essentials



Abstraction:

- Cloud computing abstracts the details of system implementation from users and developers.
- Applications run on physical systems that aren't specified, data is stored in locations that are unknown, administration of systems is outsourced to others, and access by users is ubiquitous.

Virtualization:

- Cloud computing virtualizes systems by pooling and sharing resources.
- Systems and storage can be provisioned as needed from a centralized infrastructure, costs are assessed on a metered basis, multi-tenancy is enabled, and resources are scalable with agility.

- To help clarify how cloud computing has changed the nature of commercial system deployment, consider these three examples:

Google:

- Google has built a worldwide network of datacenters to service its search engine. In doing so Google has captured a substantial portion of the world's advertising revenue.
- That revenue has enabled Google to offer free software to users based on that infrastructure and has changed the market for user-facing software.

Azure Platform:

- Microsoft is creating the Azure Platform. It enables .NET Framework applications to run over the Internet as an alternate platform for Microsoft developer software running on desktops.

Amazon Web Services:

- One of the most successful cloud-based businesses is Amazon Web Services, which is an Infrastructure as a Service offering that lets you rent virtual computers on Amazon's own infrastructure.

Cloud Computing

- **Cloud computing** is the result of evolution and adoption of existing technologies and paradigms: **Autonomic computing, Client–Server model, Grid computing, Mainframe computer, Utility computing, Peer-to-peer and Virtualization**.

- **Goal of Cloud Computing:** To allow users to take **maximum benefits** from all of these technologies, without the need for deep knowledge about or expertise with each one of them.

- To **reduced costs**, and help the users **focus on their core business** instead of being restricted by IT obstacles.
- **Service Models : Saas, Paas, Iaas, Caas, Naas.**

Deployment models:

- **Private cloud** : Enterprise owned or lease
- **Public cloud** : Sold to the public, mega-scale infrastructure
- **Hybrid cloud** : Composition of two or more clouds
- **Community cloud** : Shared infrastructure for specific community

SaaS: (Software as a Service)

- Google Apps, Microsoft Office 365, Google, AWS, ORACLE, SAP, Zoom

PaaS: (Platform as a Service)

- Google App Engine, Salesforce- Heroku, Openshift-RedHat, AWS Elastic Beanstalk, Amazon Web Services(AWS)

IaaS:(Infrastructure as a Service)

- Amazon EC2, Google computer engine, HP Cloud, Oracle infrastructure, Rackspace Open Cloud, Relia Cloud, IBM Cloud, Vmware cloud an AWS, Microsoft Azure, Digital Ocean

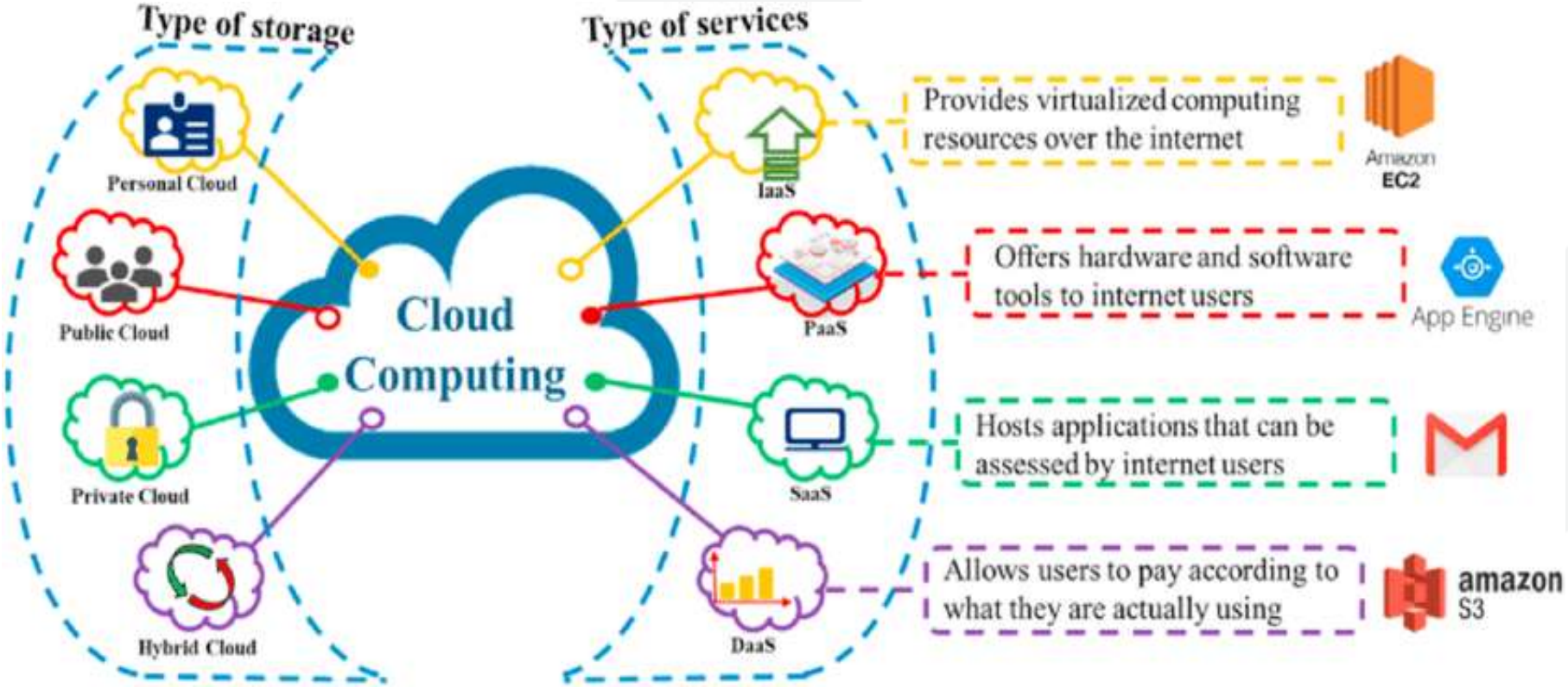
Caas: (Container as a Service)

- Google Kubernetes, Docker Swarm, AWS Elastic Container Service(ECS), Google cloud Container, Azure Container Instances (ACI)

NaaS : (Network as a Service)- SDN, NFV, Cloud Networking, WAN Optimization, Security as a service(SECaaS)- Use cases

- AWS network services, Google, Microsoft Azure, CISCO Cloud, VM ware NSX cloud.

Cloud Computing



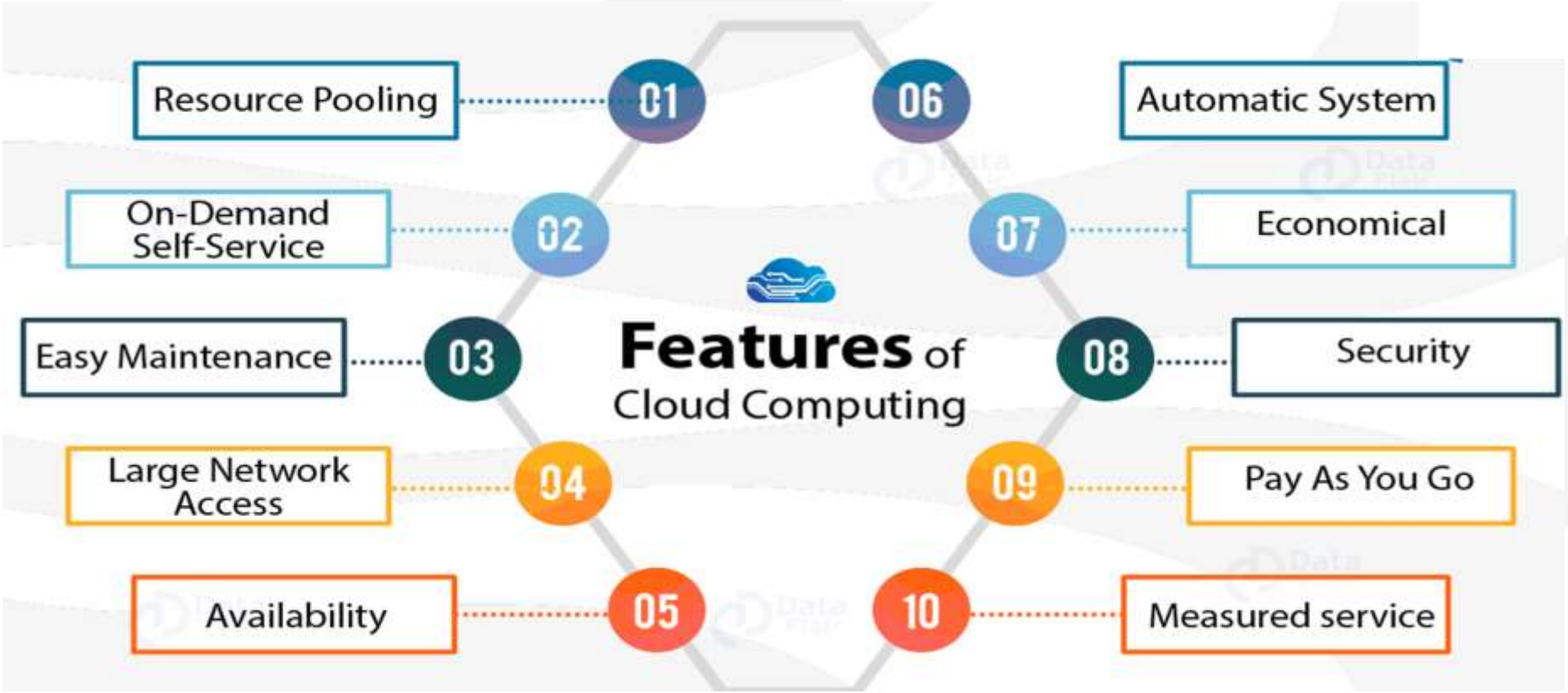
Cloud Computing-Benefits

- Environmental Friendly
- Software Integration
- Cost Proficient
- More secure
- Greater Flexibility
- Infinite storage
- Rapid Development
- Backup and Recovery
- Document Control
- Fewer maintaince issues

Cloud Computing-Limitations

- Doesn’t work well in low speed connections
- Outsourcing data storage increases potential for attacker
- Supplier stability
- Accessibility
- Security of stored data
- Enable piracy and Copyright infringement
- Requires constant Internet connections
- Unreliable services due to due to inadequate ICT or Power Infrastructure
- Risk of Vendor Locking

Cloud Computing-Features



Cloud Computing-Challenges

Percentage	Cloud Usage / Issues
97%	Many organizations uses cloud services- Public, Private, Hybrid
65%	Cloud strategy aspects
10%	Anticipate decrease in cloud investment
53%	Unauthorized access
44%	Hijacking of accounts
49%	Insecure Interfaces or APIs
33%	External Sharing of data

Identity as a Service (IDaaS)

- Identity as a Service, or IDaaS companies supply cloud-based authentication or identity management to enterprises who subscribe.
- IDaaS companies supply cloud-based authentication or identity management to enterprises who subscribe. The X-as-a-service model in information technology is easy to understand.
- The X-as-a-service model in information technology is easy to understand. It means some feature is being delivered or served to a company through a remote connection from a third-party provider, as opposed to a feature being managed on site and by in-house personnel alone.
- Think of local email, such as Microsoft Outlook or Thunderbird, operating primarily on one's own computer versus cloud email, such as Gmail, being provided to users as a service through web connections. Identity, security, and other features can similarly be provided as a service.
- The goal of an Identity Service is to ensure users are who they claim to be, and to give them the right kinds of access to software applications, files, or other resources at the right times. If the infrastructure to make this happen is built on site, then the company has to figure out what to do every time a problem comes up.

Identity as a Service (IDaaS)

- If Bring Your Own Device (BYOD) employees are changing to different types of phones, for example, the local identity provisioning has to adapt immediately.
- It is much simpler to implement a centralized cloud-based system created by identity experts who have already solved such problems for hundreds of organizations.
- Services that provide digital identity management as a service have been part of internetworked systems from Day One. Like so many concepts in cloud computing, IDentity as a Service is a FLAVor (Four Letter Acronym) of the month, applied to services that already exist.
- The Domain Name Service can run on a private network, but is at the heart of the Internet as a service that provides identity authorization and lookup. The name servers that run the various Internet domains (.COM, .ORG, .EDU, .MIL, .TV, .RU, and so on) are IDaaS servers.
- DNS establishes the identity of a domain as belonging to a set of assigned addresses, associated with an owner and that owner's information, and so forth. If the identification is the assigned IP number, the other properties are its metadata.

Identity as a Service (IDaaS)

- An identity is a set of characteristics or traits that make something recognizable or known. In computer network systems, it is one's digital identity that most concerns us.
- A digital identity is those attributes and metadata of an object along with a set of relationships with other objects that makes an object identifiable.
- Not all objects are unique, but by definition a digital identity must be unique, if only trivially so, through the assignment of a unique identification attribute. An identity must therefore have a context in which it exists.
- This description of an identity as an object with attributes and relationships is one that programmer's would recognize.
- Databases store information and relationships in tables, rows, and columns, and the identity of information stored in this way conforms to the notion of an entity and a relationship or alternatively under the notion of an object role model (ORM)and database architects are always wrestling with the best way of reducing their data set to a basic set of identities.
- You can extend this notion to the idea of an identity having a profile and profiling services such as Facebook as being an extension of the notion of Identity as a Service in cloud computing.

Identity as a Service (IDaaS)

An identity can belong to a person and may include the following:

- Things you are: Biological characteristics such as age, race, gender, appearance, and so forth.
- Things you know: Biography, personal data such as social security numbers, PINs, where you went to school, and so on
- Things you have: A pattern of blood vessels in your eye, your fingerprints, a bank account you can access, a security key you were given, objects and possessions, and more
- Things you relate to: Your family and friends, a software license, beliefs and values, activities and endeavors, personal selections and choices, habits and practices, an iGoogle account, and more.

Identity as a Service (IDaaS)

- To establish your identity on a network, you might be asked to provide a name and password, which is called a single-factor authentication method.
- More secure authentication requires the use of at least two-factor authentication; for example, not only name and password (things you know) but also a transient token number provided by a hardware key (something you have).
- To get to multifactor authentication, you might have a system that examines a biometric factor such as a fingerprint or retinal blood vessel pattern—both of which are essentially unique things you are.
- Multifactor authentication requires the outside use of a network security or trust service, and it is in the deployment of trust services that our first and most common IDaaS applications are employed in the cloud.
- Of course, many things have digital identities. User and machine accounts, devices, and other objects establish their identities in a number of ways.
- For user and machine accounts, identities are created and stored in domain security databases that are the basis for any network domain, in directory services, and in data stores in federated systems.
- Network interfaces are identified uniquely by Media Access Control (MAC) addresses, which alternatively are referred to as Ethernet Hardware Addresses (EHAs). It is the assignment of a network identity to a specific MAC address that allows systems to be found on networks.

Identity as a Service (IDaaS)

- The manner in which Microsoft validates your installation of Windows and Office is called Windows Product Activation and creates an identification index or profile of your system, which is instructive.

- **During activation, the following unique data items are retrieved:**

- A 25-character software product key and product ID
- The uniquely assigned Global Unique Identifier or GUID
- PC manufacturer
- CPU type and serial number
- BIOS checksum
- Network adapter and its MAC address
- Display adapter
- SCSCI and IDE adapters
- RAM amount
- Hard drive and volume serial number
- Optical drive
- Region and language settings and user locale

Identity as a Service (IDaaS)

- From this information, a code is calculated, checked, and entered into the registration database.
- Each of these uniquely identified hardware attributes is assigned a weighting factor such that an overall sum may be calculated. If you change enough factors NIC and CPU, display adapter, RAM amount, and hard drive—you trigger a request for a reactivation based on system changes.
- This activation profile is also required when you register for the Windows Genuine Advantage program. Windows Product Activation and Windows Genuine Advantage are cloud computing applications, albeit proprietary ones.
- Whether people consider these applications to be services is a point of contention.

Networked identity service classes

- To validate Web sites, transactions, transaction participants, clients, and network services various forms of identity services—have been deployed on networks.
- Ticket or token providing services, certificate servers, and other trust mechanisms all provide identity services that can be pushed out of private networks and into the cloud.
- Identity protection is one of the more expensive and complex areas of network computing. If you think about it, requests for information on identity by personnel such as HR, managers, and others.
- By systems and resources for access requests; as identification for network traffic; and the myriad other requirements mean that a significant percentage of all network traffic is supporting an identification service.
- Literally hundreds of messages on a network every minute are checking identity, and every Ethernet packet contains header fields that are used to identify the information it contains.
- As systems become even more specialized, it has become increasingly difficult to find the security experts needed to run an ID service. Identity as a Service or the related hosted (managed) identity services may be the most valuable and cost effective distributed service types you can subscribe to.

Networked identity service classes

- Identity as a Service (IDaaS) may include any of the following:

- Authentication services (identity verification)
- Directory services
- Federated identity
- Identity governance
- Identity and profile management
- Policies, roles, and enforcement
- Provisioning (external policy administration)
- Registration
- Risk and event monitoring, including audits
- Single sign-on services (pass-through authentication)

- The sharing of any or all of these attributes over a network may be the subject of different government regulations and in many cases must be protected so that only justifiable parties may have access to the minimal amount that may be disclosed.

- This level of access defines what may be called an identity relationship

Identity system codes of conduct

- Certain codes of conduct must be observed legally, and if not legally at the moment, then certainly on a moral basis. Cloud computing services that don't observe these codes do so at their peril. In working with IDaaS software, evaluate IDaaS applications on the following basis:
- **User control for consent:** Users control their identity and must consent to the use of their information.
- **Minimal Disclosure:** The minimal amount of information should be disclosed for an Intended use.

- **Justifiable access:** Only parties who have a justified use of the information contained in a digital identity and have a trusted identity relationship with the owner of the information may be given access to that information.

- **Directional Exposure:** An ID system must support bidirectional identification for a public entity so that it is discoverable and a unidirectional identifier for private entities, thus protecting the private ID.

- **Interoperability:** A cloud computing ID system must interoperate with other identity services from other identity providers.

- **Unambiguous human identification:** An IDaaS application must provide an unambiguous mechanism for allowing a human to interact with a system while protecting that use against an identity attack.

- **Consistency of Service:** An IDaaS service must be simple to use, consistent across all its uses, and able to operate in different contexts using different technologies.

IDaaS interoperability

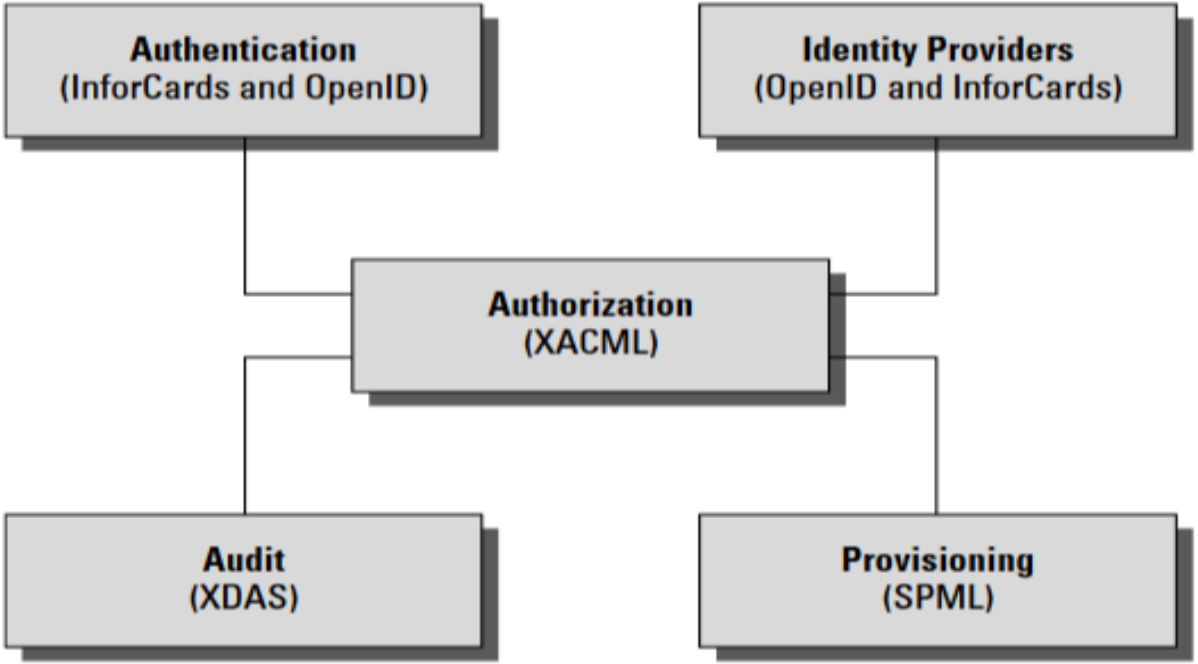
- Identity as a Service provides an easy mechanism for integrating identity services into individual applications with minimal development effort, by allowing the identification logic and storage of an identity's attributes to be maintained externally.

- Cloud computing IDaaS applications must rely on a set of developing industry standards to provide interoperability. The following are among the more important of these services:

- **User centric authentication** (usually in the form of information cards): The OpenID and CardSpace specifications support this type of data object.
- **The XACML Policy Language**: This is a general-purpose authorization policy language that allows a distributed ID system to write and enforce custom policy expressions.
- **XACML can work with SAML**; when SAML presents a request for ID authorization, XACML checks the ID request against its policies and either allows or denies the request.
- **The SPML Provisioning Language:** This is an XML request/response language that is used to integrate and interoperate service provisioning requests. SPML is a standard of OASIS's Provision Services Technical Committee (PSTC) that conforms to the SOA architecture.
- **The XDAS Audit System:** The Distributed Audit Service provides accountability for users accessing a system, and the detection of security policy violations when attempts are made to access the system by unauthorized users or by users accessing the system in an unauthorized way.

IDaaS interoperability

Open standards that support an IDaaS infrastructure for cloud computing



IDaaS interoperability

- The Identity Governance Framework (IGF) is a standards initiative of the Liberty Alliance (http://www.projectliberty.org/) that is concerned with the exchange and control of identity information using standards such as WS-Trust, ID-WSF, SAML, and LDAP directory services.

- The Liberty Alliance was established by an industry group in 2001 with the purpose of promoting open identity interchanges through policy standards that applications can use to enforce privacy as well as to allow privacy auditing.

- In 2009, this group released its Client Attribute Requirements Markup Language (CARML) and a set of IGF Privacy Constraints that forms the basis of the open source project called Aristotle (http://www.openliberty.org/wiki/index.php/ ProjectAris), which has as its goal the creation of an API for identity interchange.

User authentication

- OpenID is a developing industry standard for authenticating “end users” by storing their digital identity in a common format. When an identity is created in an OpenID system, that information is stored in the system of any OpenID service provider and translated into a unique identifier.

- Identifiers take the form of a Uniform Resource Locator (URL) or as an Extensible Resource Identifier (XRI) that is authenticated by that OpenID service provider. Any software application that complies with the standard accepts an OpenID that is authenticated by a trusted provider.

- A very impressive group of cloud computing vendors serve as identity providers (or OpenID providers), including AOL, Facebook, Google, IBM, Microsoft, MySpace, Orange, PayPal, VeriSign, LiveJournal, Ustream, Yahoo!, and others.

- The OpenID standard applies to the unique identity of the URL; it is up to the service provider to store the information and specify the forms of authentication required to successfully log onto the system.

- Thus an OpenID authorization can include not only passwords, but smart cards, hardware keys, tokens, and biometrics as well. OpenID is supported by the OpenID Foundation (http://openid.net/foundation/), a not-for-profit organization that promotes the technology.

User authentication

- These are samples of trusted providers and their URL formats:

- Blogger: <username>.blogger.com or <blogid>.blogspot.com
- MySpace: myspace.com/<username>
- Google: https://www.google.com/accounts/o8/id
- Google Profile: google.com/profiles/<username>
- Microsoft: accountservices.passport.net/
- MyOpenID: <username>.myopenid.com
- Orange: openid.orange.fr/username or simply orange.fr/
- Verisign: <username>.pip.verisinglabs.com
- WordPress: <username>.wordpress.com
- Yahoo!: openid.yahoo.com
- After you have logged onto a trusted provider, that logon may provide you access to other Web sites that support OpenID.

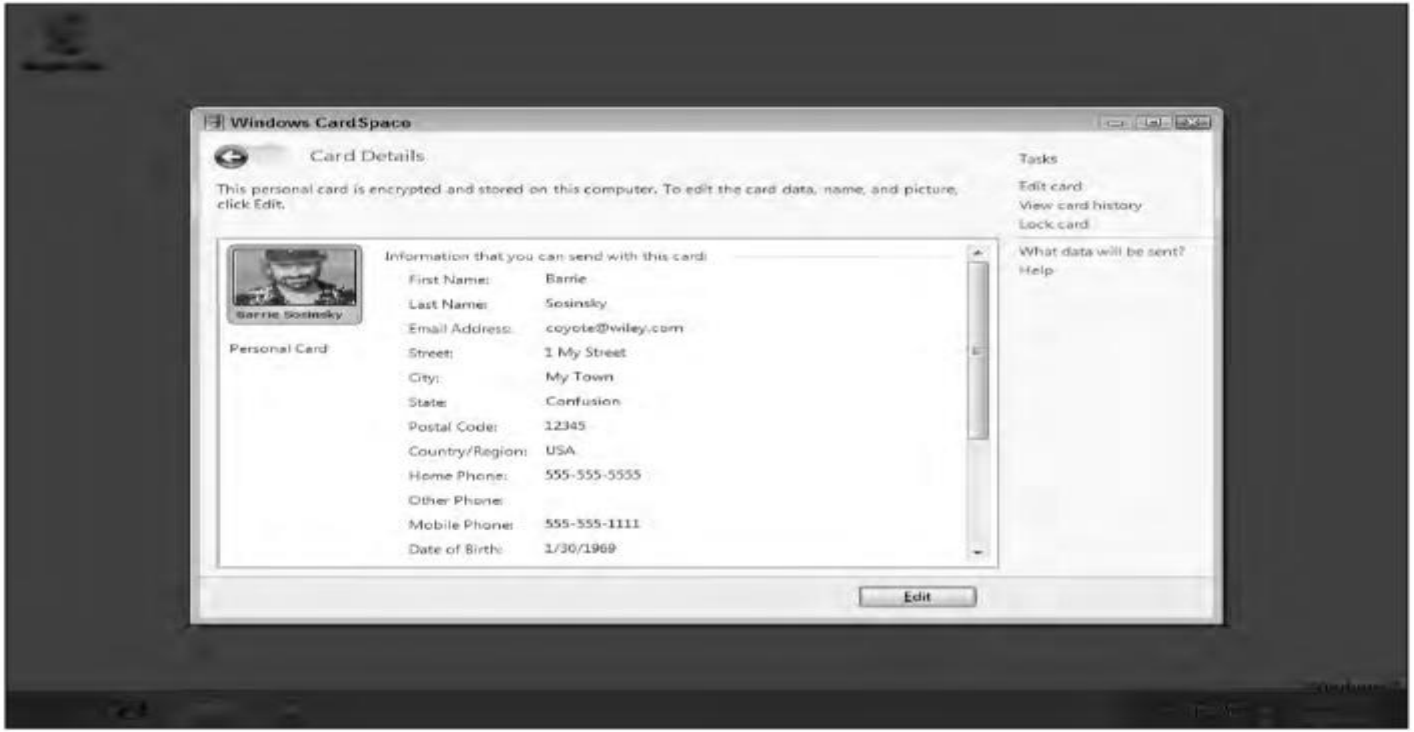
- When you request access to a site through your browser (or another application that is referred to as a user-agent), that site serves as the “relying party” and requests of the server or server-agent that it verify the end-user’s identifier.

User authentication

- CardSpace is a Microsoft software client that is part of the company's Identity Metasystem and built into the Web Services Protocol Stack.
- This stack is built on the OASIS standards (WS-Trust, WS-Security, WS-SecurityPolicy, and WS-MetadataExchange), so any application that conforms with the OASIS WS- standards can interoperate with CardSpace. CardSpace was introduced with .NET Frameworks 3.0 and can be installed on Windows XP, Server 2003, and later.
- It is installed by default on Windows Vista and Windows 7.
- CardSpace offers another way of authenticating users in the cloud. An Information Card may be requested with an HTML <OBJECT> tag, and the trusted Identity Provider then creates an encrypted and digitally signed token using the Security Token Service (STS) that is part of a WS-Trust request/ reply mechanism.
- CardSpace may be seen as an alternative mechanism to the use of OpenID and SAML and is used to sign into those services as well as Windows Live ID accounts.

User authentication

This is a private CardSpace Identification Card. Managed Identification Cards that store similar information are stored on a network service and can be shared to the cloud.



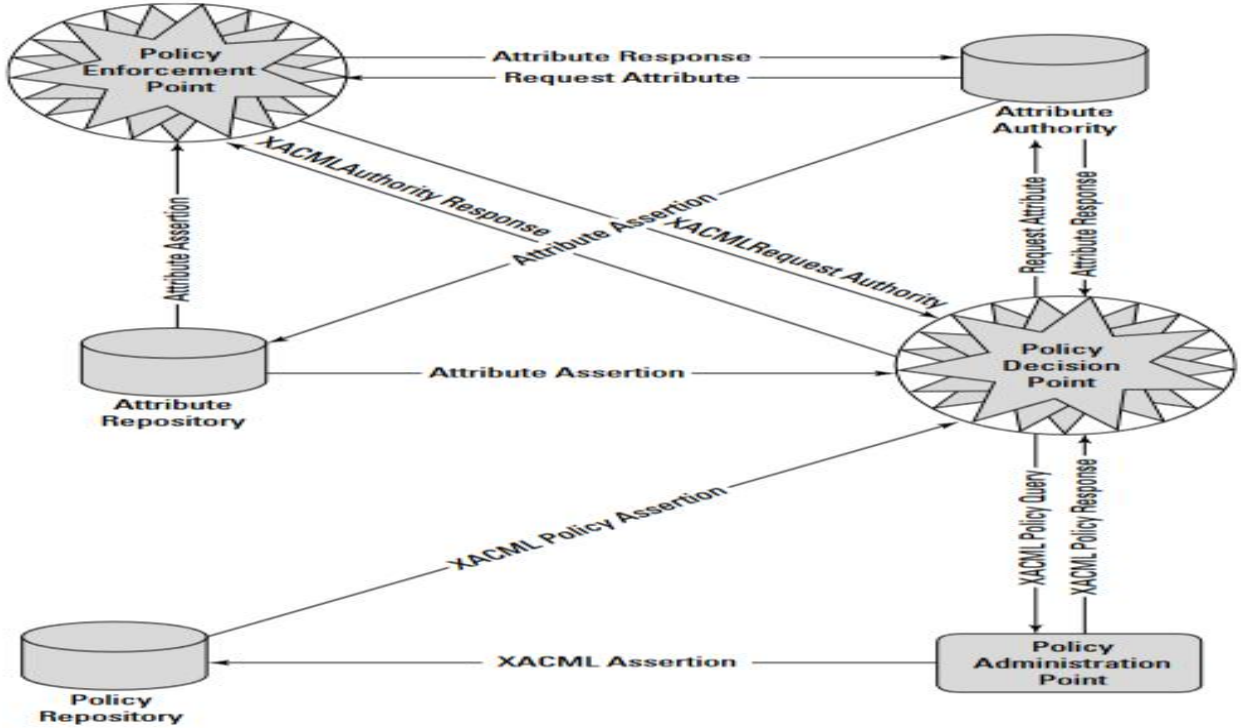
User authentication

The Higgins Open Source Identity Framework uses an i-Card metaphor and interoperable identity service APIs to create a vendor-neutral cloud-based authentication service.

Authorization Markup Languages

- Information requests and replies in cloud computing are nearly always in the form of XML replies or requests. XML files are text files and are self-describing.
- That is, XML files contain a schema that describes the data it contains or contains a point to another text file with its schema.
- A variety of specialized XML files are in the identity framework, the ones of note being XACML and SAML, shown in Figure.
- The eXtensible Access Control Markup Language (XACML) is an OASIS standard (see http://xml.coverpages.org/xacml.html) for a set of policy statements written in XML that support an authentication process.
- A policy in XACML describes a subject element that requests an action from a resource. These three elements operate within an environment that also can be described in terms of an Action element.
- Subject and Action elements (which are terms of art in XACML) are elements that can have one or more attributes. Resources (which are services, system components, or data) have a single attribute, which is usually its URL.

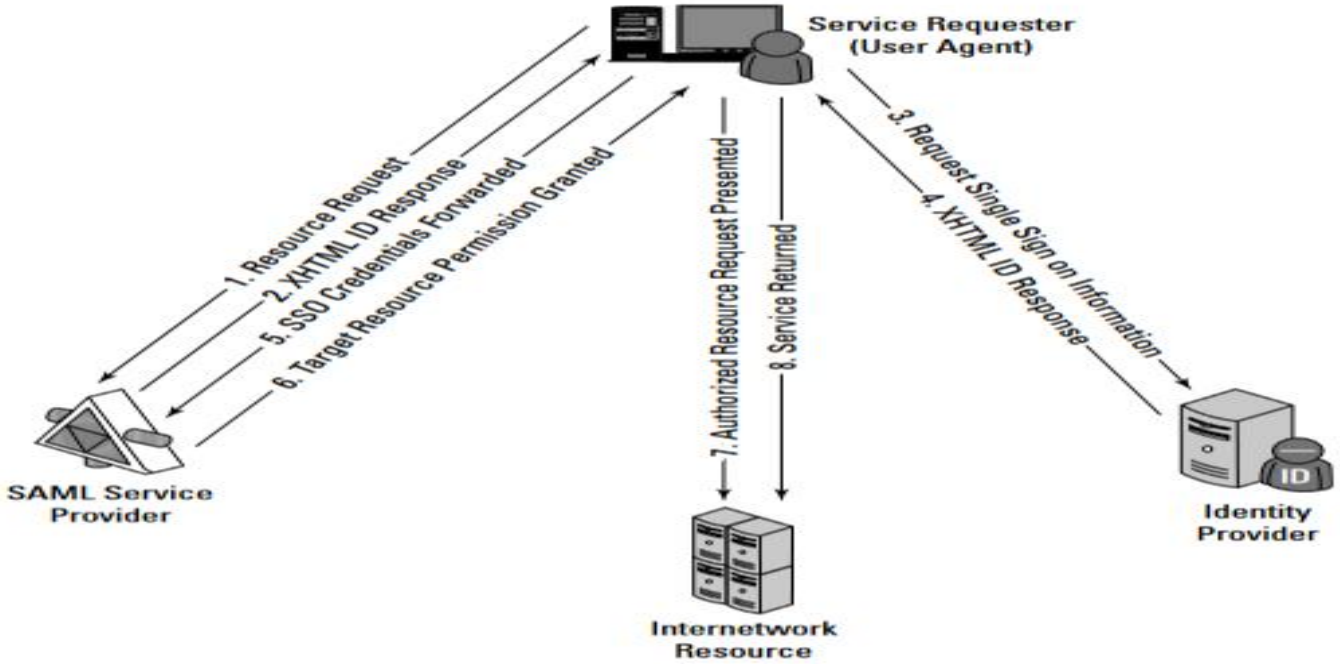
SAML integrates with XACML to implement a policy engine in a Service Oriented Architecture to support identity services authorization.



Authorization Markup Languages

SAML provides a mechanism by which a service requester can use a Single Sign On logon to access Web services securely.

SAML Single Sign On Request/Response Mechanism



Authorization Markup Languages

- A SAML assertion is a security statement in the SAML file that makes a claim regarding authenticaion, attributes, or authorization.

The statement is of this form:

- Assertion X created at Time T by User U about Subject S is true when Conditions Care TRUE.

- It is up to the identity provider to parse this statement and determine its validity. The SAML protocol request is often referred to as a query; the three different supported query types are an authentication query, an attribute query, and an authorization decision query.

- SAML requests use a SOAP binding; that is, the SAML request or response is embedded in a SOAP wrapper within an HTTP message.

- SAML is used to provide a mechanism for a Web Browser Single Sign On (SSO). In this instance, a Web browser is the user agent, which requests access to a resource that is authorized by a SAML service provider.

Authorization Markup Languages

- The service provider takes a request from a user for access to the resource and sends an authentication request to the SAML identity provider directly from the initiating user agent (Web browser). Figure shows the SAML Single Sign On Request/Response mechanism.

- The Service Provisioning Markup Language (SPML) is another of the OASIS open standards developed to provide for service provisioning. Provisioning is the process by which a resource is prepared for use, reserved, accessed, used, and then released when the transaction is completed. A classic example of provisioning a resource is the reservation and use of a phone line or a Virtual Private Network.

- A provisioning system has three types of components: A Requesting Authority (RA) is the client, the Provisioning Service Point (PSP) is the cloud component that receives the request and returns a response to the RA, and a Provisioning Service Targets (PST) is the software application upon which the provisioning action is performed.

- The SPML provisioning system (which can be thought of as an architectural layer) means that identity information need only be entered into these three components once. SPML is used to prepare Web services and applications for use, signal that the resource is available for use and waiting for instructions, and signal when the use or transaction has been completed.

- With SPML, a system can provide automated user and system access, enforce access rights, and make cloud computing services available across network systems. Without a provisioning system, a cloud computing system can be very inefficient and potentially unreliable.

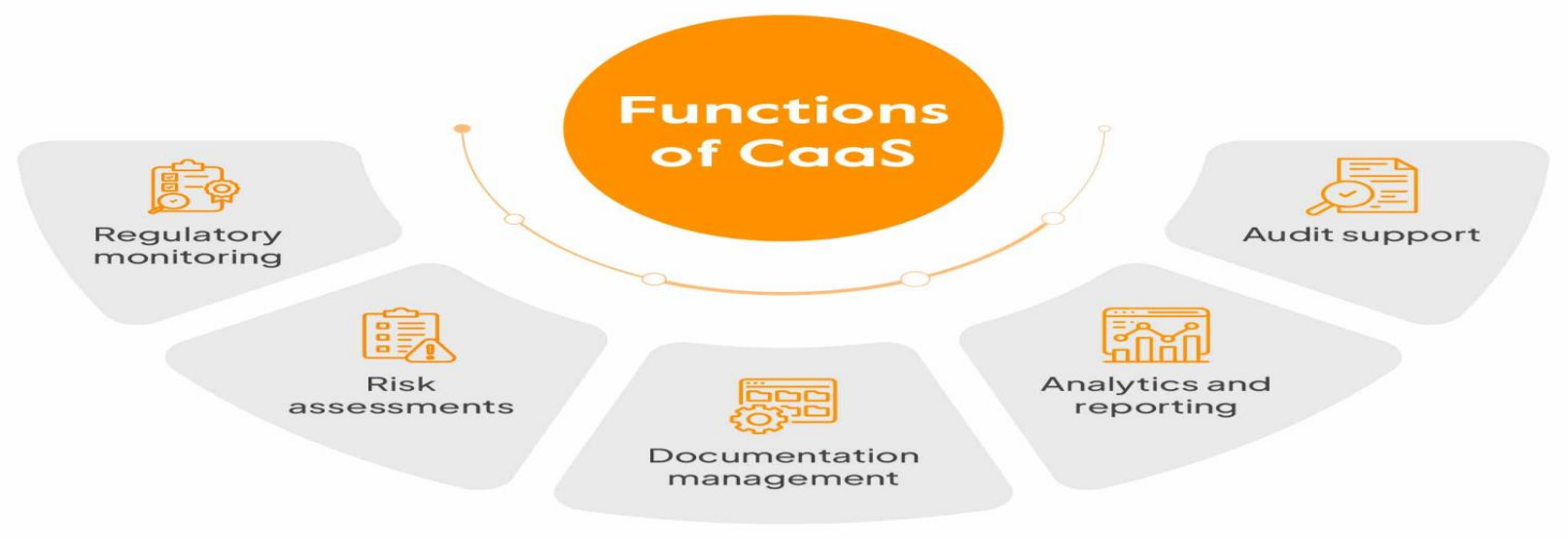
Identity as a Service (IDaaS)-Benefits

- Advantage of IDaaS is savings.
- Provisioning identity on site, with software such as Active Directory Domain Services, can be full of costs.
- Your team has to keep up servers; purchase, upgrade, and install software; back up data regularly; pay hosting fees; monitor the additional turf on premises for network security; set up VPNs; and much more.
- With IDaaS, costs drop to the subscription fee and the administration work.
- Besides savings, ROI for IDaaS includes improved cybersecurity and saved time with faster logins and fewer password resets.
- Whether a user is signing in from open WiFi at an airport or from a desk in the office, the process is seamless and secure.
- The improved security can keep companies from facing a hack or breach that might topple their business.

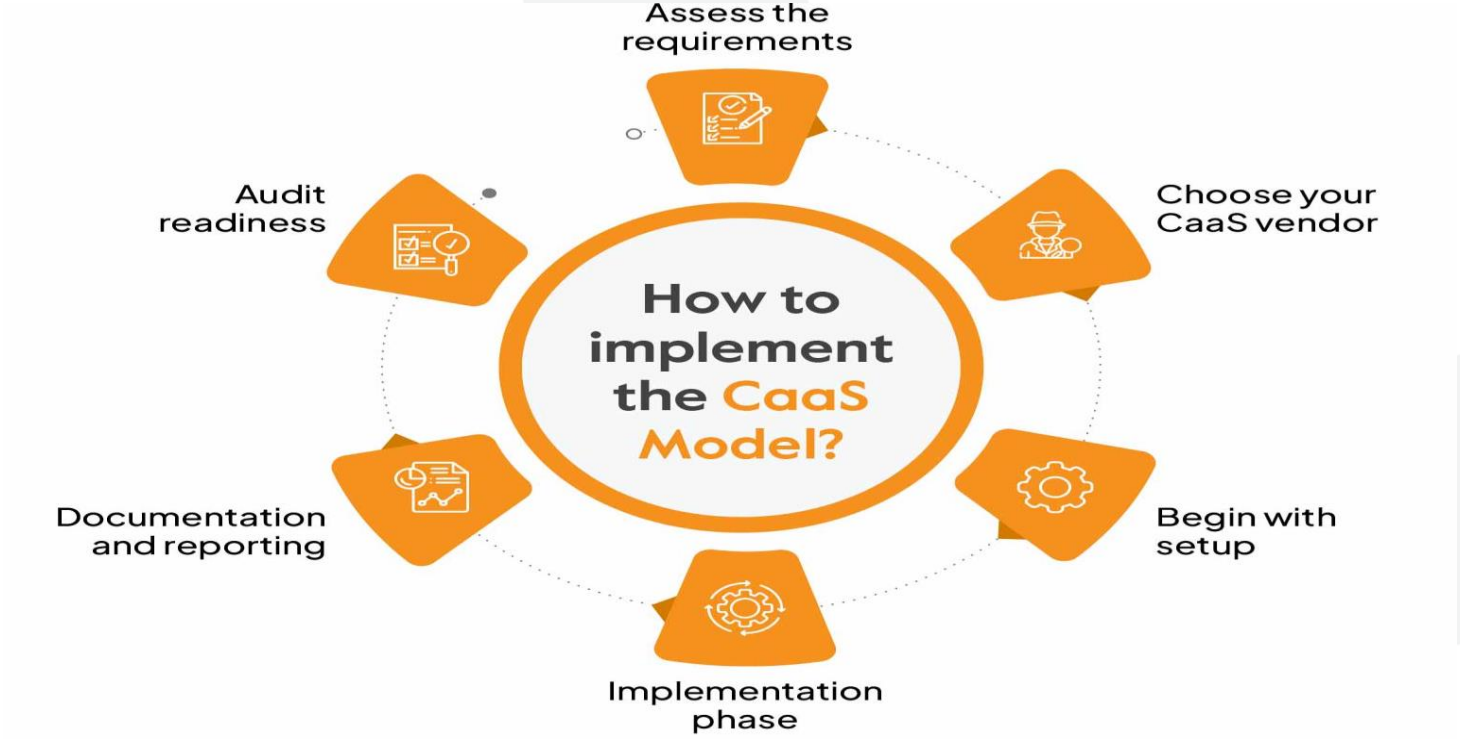
Compliance as a Service (CaaS)

- Compliance as a Service (CaaS) is a cloud-based model that helps organizations meet regulatory compliance requirements by outsourcing compliance management to a third party.

- Compliance implementation, management and maintenance services to regulated companies in various industries, often such as healthcare, financial and government.



Compliance as a Service (CaaS)



Compliance as a Service (CaaS)

- Cloud computing by its very nature spans different jurisdictions. The laws of the country of a request's origin may not match the laws of the country where the request is processed, and it's possible that neither location's laws match the laws of the country where the service is provided.
- Compliance is much more than simply providing an anonymous service token to an identity so they can obtain access to a resource. Compliance is a complex issue that requires considerable expertise.
- While Compliance as a Service (CaaS) appears in discussions, few examples of this kind of service exist as a general product for a cloud computing architecture.
- A Compliance as a Service application would need to serve as a trusted third party, because this is a man-in-the-middle type of service. CaaS may need to be architected as its own layer of a SOA architecture in order to be trusted.
- A CaaS would need to be able to manage cloud relationships, understand security policies and procedures, know how to handle information and administer privacy, be aware of geography, provide an incidence response, archive, and allow for the system to be queried, all to a level that can be captured in a Service Level Agreement.
- CaaS has the potential to be a great value-added service

Compliance as a Service (CaaS)

- In order to implement CaaS, some companies are organizing what might be referred to as “verticalclouds,” clouds that specialize in a vertical market.

Examples of vertical clouds that advertise CaaS capabilities include the following:

- **athenahealth** (http://www.athenahealth.com/) for the medical industry
- **bankserv** (http://www.bankserv.com/) for the banking industry
- **ClearPoint** PCI Compliance-as-a-Service for merchant transactions under the Payment Card Industry Data Security Standard
- **FedCloud** (http://www.fedcloud.com/) for government
- **Rackserve** PCI Compliant Cloud (http://www.rackspace.com/; another PCI CaaS service)

- CaaS system built inside a private cloud where the data is under the control of a single entity, thus ensuring that the data is under that entity’s secure control and that transactions can be audited. Indeed, most of the cloud computing compliance systems to date have been built using private clouds.
- CaaS could be an incredibly valuable service. A well-implemented CaaS service could measure the risks involved in servicing compliance and ensure or indemnify customers against that risk.
- CaaS could be brought to bear as a mechanism to guarantee that an e-mail conformed to certain standards, something that could be a new electronic service of a network of national postal system and something that could help bring an end to the threat of spam.