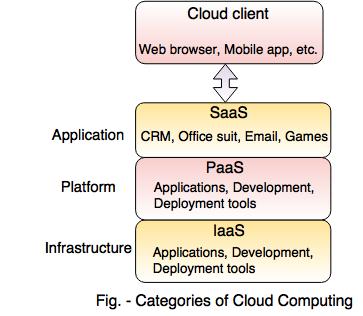
**CLOUD SERVICE MODELS**

**The service models are categorized into three basic models:**  
  
1) Software-as-a-Service (SaaS)  
2) Platform-as-a-Service (PaaS)  
3) Infrastructure-as-a-Service (IaaS)  
  


1) Software-as-a-Service (SaaS)

* SaaS is known as **'On-Demand Software'.**
* It is a software distribution model. In this model, the applications are hosted by a cloud service provider and publicized to the customers over internet.
* In SaaS, associated data and software are hosted centrally on the cloud server.
* User can access SaaS by using a thin client through a web browser.
* CRM, Office Suite, Email, games, etc. are the software applications which are provided as a service through Internet.
* The companies like Google, Microsoft provide their applications as a service to the end users.

**Advantages of SaaS**

* SaaS is easy to buy because the pricing of SaaS is based on monthly or annual fee and it allows the organizations to access business functionalities at a small cost, which is less than licensed applications.
* SaaS needed less hardware, because the software is hosted remotely, hence organizations do not need to invest in additional hardware.
* Less maintenance cost is required for SaaS and do not require special software or hardware versions.

**Disadvantages of SaaS**

* SaaS applications are totally dependent on Internet connection. They are not usable without Internet connection.
* It is difficult to switch amongst the SaaS vendors.

2) Platform-as-a-Service (PaaS)

* PaaS is a programming platform for developers. This platform is generated for the programmers to create, test, run and manage the applications.
* A developer can easily write the application and deploy it directly into PaaS layer.
* PaaS gives the runtime environment for application development and deployment tools.
* Google Apps Engine(GAE), Windows Azure, SalesForce.com are the examples of PaaS.

**Advantages of PaaS**

* PaaS is easier to develop. Developer can concentrate on the development and innovation without worrying about the infrastructure.
* In PaaS, developer only requires a PC and an Internet connection to start building applications.

**Disadvantages of PaaS**

* One developer can write the applications as per the platform provided by PaaS vendor hence the moving the application to another PaaS vendor is a problem.

3) Infrastructure-as-a-Service (IaaS)

* IaaS is a way to deliver a cloud computing infrastructure like server, storage, network and operating system.
* The customers can access these resources over cloud computing platform i.e Internet as an on-demand service.
* In IaaS, you buy complete resources rather than purchasing server, software, datacenter space or network equipment.
* IaaS was earlier called as Hardware as a Service(HaaS). It is a Cloud computing platform based model.
* HaaS differs from IaaS in the way that users have the bare hardware on which they can deploy their own infrastructure using most appropriate software.

**Advantages of IaaS**

* In IaaS, user can dynamically choose a CPU, memory storage configuration according to need.
* Users can easily access the vast computing power available on IaaS Cloud platform.

**Disadvantages of IaaS**

* IaaS cloud computing platform model is dependent on availability of Internet and virtualization services.

# Cloud Computing Infrastructure as a Service (IaaS)

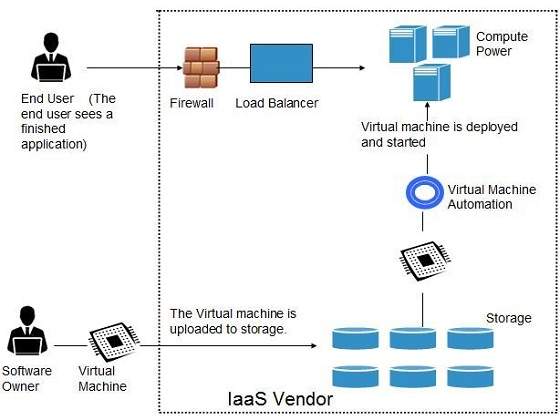
**Infrastructure-as-a-Service** provides access to fundamental resources such as physical machines, virtual machines, virtual storage, etc. Apart from these resources, the IaaS also offers:

* Virtual machine disk storage
* Virtual local area network (VLANs)
* Load balancers
* IP addresses
* Software bundles

All of the above resources are made available to end user via **server virtualization.** Moreover, these resources are accessed by the customers as if they own them.

In this service the Cloud Provider provides the customer with virtual machines and other resources as a service, they abstract the user from the physical machine, location, data partitioning etc. If the user wants a Linux machine, he gets a linux machine, he will not worry about the physical machine or the networking of the system on which the OS is installed, simple.

* For Example AWS(Amazon Web Services) is IaaS, like [AWS EC2](https://www.edureka.co/blog/ec2-aws-tutorial-elastic-compute-cloud/).



Characteristics

Here are the characteristics of IaaS service model:

* Virtual machines with pre-installed software.
* Virtual machines with pre-installed operating systems such as Windows, Linux, and Solaris.
* On-demand availability of resources.
* Allows to store copies of particular data at different locations.
* The computing resources can be easily scaled up and down.

Benefits

**IaaS** allows the cloud provider to freely locate the infrastructure over the Internet in a cost-effective manner. Some of the key benefits of IaaS are listed below:

* Full control of the computing resources through administrative access to VMs.
* Flexible and efficient renting of computer hardware.
* Portability, interoperability with legacy applications.

Full control over computing resources through administrative access to VMs

**IaaS** allows the customer to access computing resources through administrative access to virtual machines in the following manner:

* Customer issues administrative command to cloud provider to run the virtual machine or to save data on cloud server.
* Customer issues administrative command to virtual machines they owned to start web server or to install new applications.

Flexible and efficient renting of computer hardware

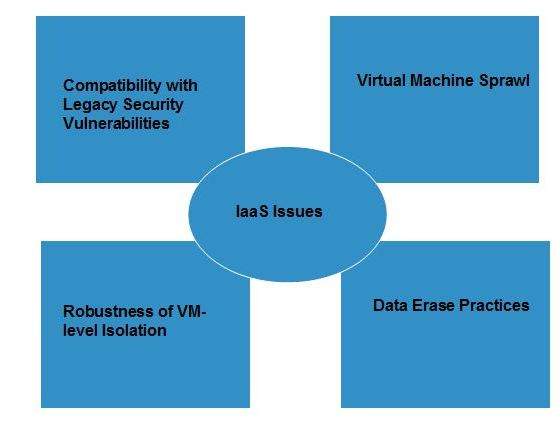
IaaS resources such as virtual machines, storage devices, bandwidth, IP addresses, monitoring services, firewalls, etc. are made available to the customers on rent. The payment is based upon the amount of time the customer retains a resource. Also with administrative access to virtual machines, the customer can run any software, even a custom operating system.

Portability, interoperability with legacy applications

It is possible to maintain legacy between applications and workloads between IaaS clouds. For example, network applications such as web server or e-mail server that normally runs on customer-owned server hardware can also run from VMs in IaaS cloud.

Issues

IaaS shares issues with PaaS and SaaS, such as Network dependence and browser based risks. It also has some specific issues, which are mentioned in the following diagram:



Compatibility with legacy security vulnerabilities

Because IaaS offers the customer to run legacy software in provider's infrastructure, it exposes customers to all of the security vulnerabilities of such legacy software.

Virtual Machine sprawl

The VM can become out-of-date with respect to security updates because IaaS allows the customer to operate the virtual machines in running, suspended and off state. However, the provider can automatically update such VMs, but this mechanism is hard and complex.

Robustness of VM-level isolation

IaaS offers an isolated environment to individual customers through hypervisor. Hypervisor is a software layer that includes hardware support for virtualization to split a physical computer into multiple virtual machines.

Data erase practices

The customer uses virtual machines that in turn use the common disk resources provided by the cloud provider. When the customer releases the resource, the cloud provider must ensure that next customer to rent the resource does not observe data residue from previous customer.

# Cloud Computing Software as a Service (SaaS)

**Software-as–a-Service (SaaS)** model allows to provide software application as a service to the end users. It refers to a software that is deployed on a host service and is accessible via Internet. There are several SaaS applications listed below:

* Billing and invoicing system
* Customer Relationship Management (CRM) applications
* Help desk applications
* Human Resource (HR) solutions

In this service the Cloud Provider leases applications or softwares which are owned by them to its client. The client can access these softwares on any device which is connected to the Internet using tools such as a web browser, an app etc.

For Example: salesforce.com provides the CRM(Customer Relation Manager) on a cloud infrastructure to its client and charges them for it, but the software is owned by the salesforce company only.

Some of the SaaS applications are not customizable such as **Microsoft Office Suite.** But SaaS provides us **Application Programming Interface (API),** which allows the developer to develop a customized application.

## Characteristics

Here are the characteristics of SaaS service model:

* SaaS makes the software available over the Internet.
* The software applications are maintained by the vendor.
* The license to the software may be subscription based or usage based. And it is billed on recurring basis.
* SaaS applications are cost-effective since they do not require any maintenance at end user side.
* They are available on demand.
* They can be scaled up or down on demand.
* They are automatically upgraded and updated.
* SaaS offers shared data model. Therefore, multiple users can share single instance of infrastructure. It is not required to hard code the functionality for individual users.
* All users run the same version of the software.

## Benefits

Using SaaS has proved to be beneficial in terms of scalability, efficiency and performance. Some of the benefits are listed below:

* Modest software tools
* Efficient use of software licenses
* Centralized management and data
* Platform responsibilities managed by provider
* Multitenant solutions

### Modest software tools

The SaaS application deployment requires a little or no client side software installation, which results in the following benefits:

* No requirement for complex software packages at client side
* Little or no risk of configuration at client side
* Low distribution cost

### Efficient use of software licenses

The customer can have single license for multiple computers running at different locations which reduces the licensing cost. Also, there is no requirement for license servers because the software runs in the provider's infrastructure.

### Centralized management and data

The cloud provider stores data centrally. However, the cloud providers may store data in a decentralized manner for the sake of redundancy and reliability.

### Platform responsibilities managed by providers

All platform responsibilities such as backups, system maintenance, security, hardware refresh, power management, etc. are performed by the cloud provider. The customer does not need to bother about them.

### Multitenant solutions

Multitenant solutions allow multiple users to share single instance of different resources in virtual isolation. Customers can customize their application without affecting the core functionality.

## Issues

There are several issues associated with SaaS, some of them are listed below:

* Browser based risks
* Network dependence
* Lack of portability between SaaS clouds

### Browser based risks

If the customer visits malicious website and browser becomes infected, the subsequent access to SaaS application might compromise the customer's data.

To avoid such risks, the customer can use multiple browsers and dedicate a specific browser to access SaaS applications or can use virtual desktop while accessing the SaaS applications.

### Network dependence

The SaaS application can be delivered only when network is continuously available. Also network should be reliable but the network reliability cannot be guaranteed either by cloud provider or by the customer.

### Lack of portability between SaaS clouds

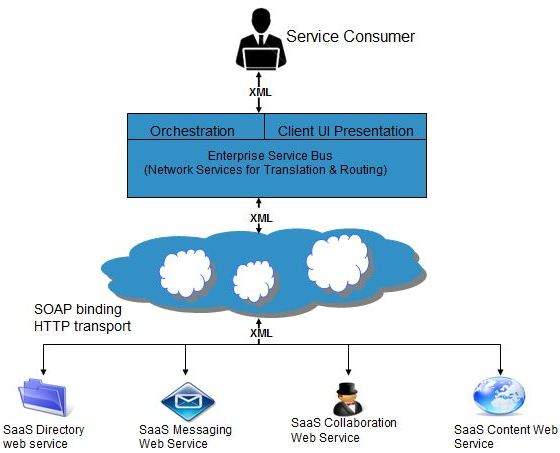
Transferring workloads from one SaaS cloud to another is not so easy because work flow, business logics, user interfaces, support scripts can be provider specific.

## Open SaaS and SOA

**Open SaaS** uses those SaaS applications, which are developed using open source programming language. These SaaS applications can run on any open source operating system and database. Open SaaS has several benefits listed below:

* No License Required
* Low Deployment Cost
* Less Vendor Lock-in
* More portable applications
* More Robust Solution

The following diagram shows the SaaS implementation based on SOA:



# 3.Cloud Computing Platform as a Service (PaaS)

**Platform-as-a-Service** offers the runtime environment for applications. It also offers development and deployment tools required to develop applications. PaaS has a feature of **point-and-click** tools that enables non-developers to create web applications.

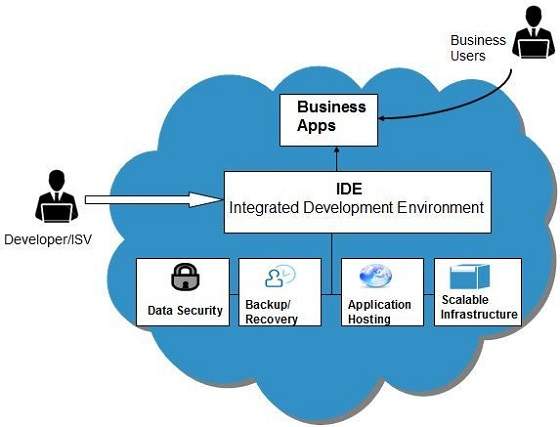
In this service the Cloud Provider gives the ability to the customer to deploy customer created application using programming languages, tools etc that are provided by the Cloud Provider. The customer cannot control the underlying architecture including operating  systems, storage, servers etc.

For Example: This service would make sense to you only if you are a developer, since this service provides you a platform for developing applications, like Google App Engine.

**App Engine of Google** and **Force.com** are examples of PaaS offering vendors. Developer may log on to these websites and use the **built-in API** to create web-based applications.

But the disadvantage of using PaaS is that, the developer **locks-in** with a particular vendor. For example, an application written in Python against API of Google, and using App Engine of Google is likely to work only in that environment.

The following diagram shows how PaaS offers an API and development tools to the developers and how it helps the end user to access business applications.



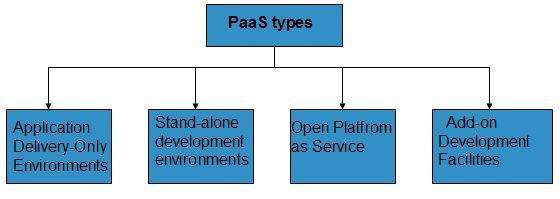
## Characteristics

Here are the characteristics of PaaS service model:

* PaaS offers **browser based development environment.** It allows the developer to create database and edit the application code either via Application Programming Interface or point-and-click tools.
* PaaS provides **built-in security, scalability,** and **web service interfaces.**
* PaaS provides built-in tools for defining **workflow, approval processes,** and business rules.
* It is easy to integrate PaaS with other applications on the same platform.
* PaaS also provides web services interfaces that allow us to connect the applications outside the platform.

## PaaS Types

Based on the functions, PaaS can be classified into four types as shown in the following diagram:



### Stand-alone development environments

The **stand-alone PaaS** works as an independent entity for a specific function. It does not include licensing or technical dependencies on specific SaaS applications.

### Application delivery-only environments

The **application delivery PaaS** includes **on-demand scaling** and **application security.**

### Open platform as a service

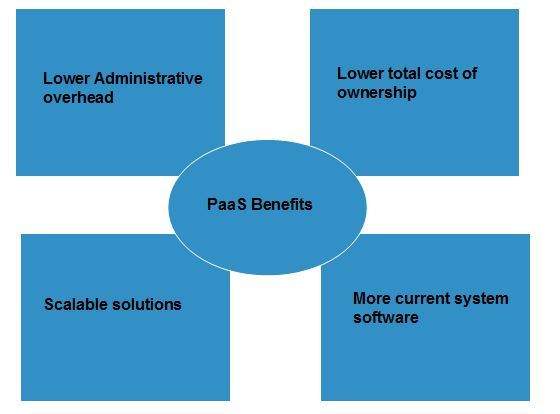
**Open PaaS** offers an **open source software** that helps a PaaS provider to run applications.

### Add-on development facilities

### The add-on PaaS allows to customize the existing SaaS platform

## Benefits

Following are the benefits of PaaS model:



### Lower administrative overhead

Customer need not bother about the administration because it is the responsibility of cloud provider.

### Lower total cost of ownership

Customer need not purchase expensive hardware, servers, power, and data storage.

### Scalable solutions

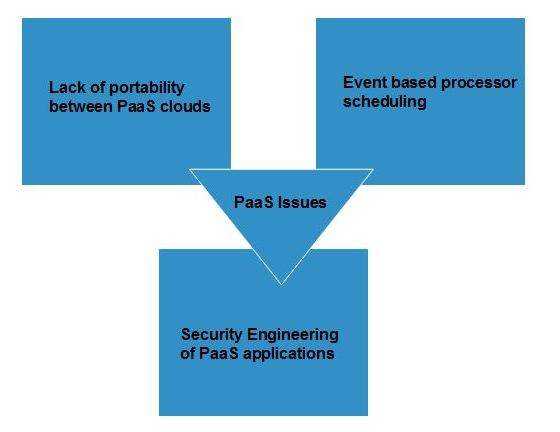
It is very easy to scale the resources up or down automatically, based on their demand.

### More current system software

It is the responsibility of the cloud provider to maintain software versions and patch installations.

## Issues

Like **SaaS, PaaS** also places significant burdens on customer's browsers to maintain reliable and secure connections to the provider’s systems. Therefore, PaaS shares many of the issues of SaaS. However, there are some specific issues associated with PaaS as shown in the following diagram:



### Lack of portability between PaaS clouds

Although standard languages are used, yet the implementations of platform services may vary. For example, file, queue, or hash table interfaces of one platform may differ from another, making it difficult to transfer the workloads from one platform to another.

### Event based processor scheduling

The PaaS applications are event-oriented which poses resource constraints on applications, i.e., they have to answer a request in a given interval of time.

### Security engineering of PaaS applications

Since PaaS applications are dependent on network, they must explicitly use cryptography and manage security exposures.

.

The diagram below, summarizes the differences b/w IaaS, PaaS and SaaS

