



SNS COLLEGE OF ENGINEERING

Kurumbapalayam (Po), Coimbatore – 641 107

An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with ‘A’ Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING(IoT and
Cybersecurity Including BCT)**

COURSE NAME : Cloud Service Management /19OE219

IV YEAR / VII SEMESTER

Unit II-

Topic : Understanding Value of Cloud Services



- The services a cloud provides to business can be divided into three categories according to the [National Institute of Standards and Technology \(NIST\) cloud model](#):
- Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS). While all clouds are built on real, physical infrastructure of network, storage and computer systems,
- IaaS provides a conveniently flexible way to configure and control a (usually) virtualized version of that data center class infrastructure
- **Infrastructure-as-a-Service(IaaS)**
- . The IaaS space is clearly exemplified by tools like the OpenStack middleware or Amazon Web Services 'as a service' platform. IaaS takes the classic systems infrastructure that an IT department might have managed in the past and enables it in an on demand version of that exact service and capability.
- **IaaS Business uses:**
- IaaS used in cases where there is a need to have the ability to rapidly scale up (or down) resources available to an application or even entire system or even rapidly deploy and remove an entire solution.





- As long as the expertise to manage the system is available in house- or can be hired as a service – IaaS systems can be used as allow your application to add additional servers to handle load – automatically, or to create a test environment that replicates the exact production environment, and remove it after the testing is done.
- Worst case, an IaaS can provide you with DR (disaster recovery) abilities as rebuilding an entire infrastructure is now automatable and can bring you from down to up in a matter of minutes or seconds.

IaaS Pros:

The main benefit to users is the ability to nearly instantly get access to resources: another computer, more storage, operating system and configurations of your choice in an on-demand fashion against a shared pool of resources that enables rapid elastic use of the infrastructure.

IaaS Cons:

On the down side users are now also responsible for ensuring that security and on-going maintenance of their newly deployed infrastructure, something that they may be woefully underprepared to manage.

Users are also responsible for releasing unused resources, a process that can be simple if the infrastructure configuration and deployment is automated, but much like the light in the living room, it's easier to turn it on than it is to remember to turn it off...



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

- The platform concept takes the cloud on-demand and elastic resources models and applies them to the services used to build an application, letting the developer focus on the application rather than the rest of the underlying infrastructure and services components.
- So rather than starting a compute server, and configuring the application deployment stack (like the Apache web server), the developer just deploys their web application code onto a web platform that provides an apache based environment.
- One could acquire a database-as-a-service as a part of a platform offering, and the application developer would just manage the tables and data within the database rather than the database application and supporting compute and storage services.
- Tools like Heroku or OpenShift provide one key variant of the Platform-as-a-Service model, highlighting a Model-View-Controller application development paradigm platform.
- Content Management Tools like WordPress, Drupal, and Django provide another PaaS model that uses content as the medium, and the platform aspects is the support of presenting content (text, video, audio) without the content creator having to focus on how to create the rest of the infrastructure and services needed to deliver it.





PaaS Business uses:

- If your business is building or running a software based solution and the application fits into the model of the PaaS systems service offering, then PaaS is an attractive option.
- Focused on ease of software or content development and management, businesses large and small benefit from being released from the concerns of managing the infrastructure their apps are running on and can focus on the job of improving the software itself.

PaaS Pros:

- With PaaS, your application developers can focus on building code and not worry about configuring and managing the underlying infrastructure.
- It is possible to develop, test, deploy and update your applications on a PaaS system without needing to have an infrastructure expert on staff.

PaaS Cons:

- If your developers need access to the underlying infrastructure, this may not be an option with many of the PaaS systems.





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

To complete our review of the basic XaaS offerings (i.e. X-as-a-Service) , we are left with Software-as-a-Service.

- The key value of software as a service is the elastic on-demand delivery and access to a specific application or software service directly to the business or end user.
- A classic example would be Google Mail, Expensify or Salesforce.com providing software tools (email, expense management, and CRM respectively) for individual users or entire business operations in that same on-demand/self-service and elastic model.
- They're available against an elastic pool of resources (what a resource means in this context really dependent on the software in question) so I can scale up or scale down my service consumption against these systems.

SaaS Business uses:

- More and more businesses are moving toward this model as both merchants or end users of software services.
- With SaaS, the IT issues of patches and updates is gone, it is typically possible to pay for what you need rather than buying and maintaining fixed numbers of licenses that can go up in quantity, but rarely go down.
- In the case of software that had a pricey unit cost, a SaaS offering may have a much more reasonable price tag, and the potential to manage service costs by disabling or returning licenses is a welcome shift in the IT consumption models that are considered normal for purchased software licenses.



SaaS Pros:

- As a consumer of a SaaS offering, you can now just use the software.
- It is possible to provide a software service such as an online catalog of software services and only use the subset of “resources” needed out of the total service provider pool of resources. Scale is not a concern for the user, and neither should maintenance/updates or new feature additions.
- If your Salesforce customer order list substantially increases due to a major new contract, the infrastructure behind your software will scale to meet the demand, but as the end user, you shouldn't worry about it, and in fact, worrying won't help, as the end user has no control over the underlying services that make up the end-user view of the system.

SaaS Cons:

- Users of SaaS offers have little control over the underlying infrastructure and components, and the same level of control over how they are deployed or maintained.
- For example, it is not possible to control or even know how much compute processing power is supporting a particular instantiation of a SaaS tool used by any individual tenant.
- If you are used to buying a piece of software and owning it for eternity, you are now effectively renting access and access is gone if you do not continue renting, along with access to your data unless you migrate away from the service before relinquishing your previously consumed resources.