

Cloud Service Models

Dr. Mukti Padhya
Assistant Professor, NFSU

Offerings of This Session : Unit II

- PAAS-Working Principle, Example
- SAAS-Working Principle, Example
- IAAS- Working Principle, Example
- SLA
- Billing & Accounting

Platform as a Service | PaaS

PaaS, or Platform-as-a-Service, provides a complete, flexible and cost-effective cloud platform for developing, running and managing applications.

Platform as a Service | PaaS

- Platform as a Service (PaaS) provides a runtime environment. It allows programmers to easily create, test, run, and deploy web applications
- In PaaS, back end scalability is managed by the cloud service provider, so end- users do not need to worry about managing the infrastructure.
- PaaS includes infrastructure (servers, storage, and networking) and platform (middleware, development tools, database management systems, business intelligence, and more) to support the web application life cycle.
- Example: Google App Engine, Force.com, Joyent, Azure.
- PaaS providers provide the Programming languages, Application frameworks, Databases, and Other tools



Platform as a Service | PaaS

- 1. Programming languages
 - PaaS providers provide various programming languages for the developers to develop the applications. Some popular programming languages provided by PaaS providers are Java, PHP, Ruby, Perl, and Go.
- 2. Application frameworks
 - PaaS providers provide application frameworks to easily understand the application development. Some popular application frameworks provided by PaaS providers are Node.js, Drupal, Joomla, WordPress, Spring, Play, Rack, and Zend.
- 3. Databases
 - PaaS providers provide various databases such as ClearDB, PostgreSQL, MongoDB, and Redis to communicate with the applications.
- 4. Other tools
 - PaaS providers provide various other tools that are required to develop, test, and deploy the applications.

Platform as a Service | PaaS

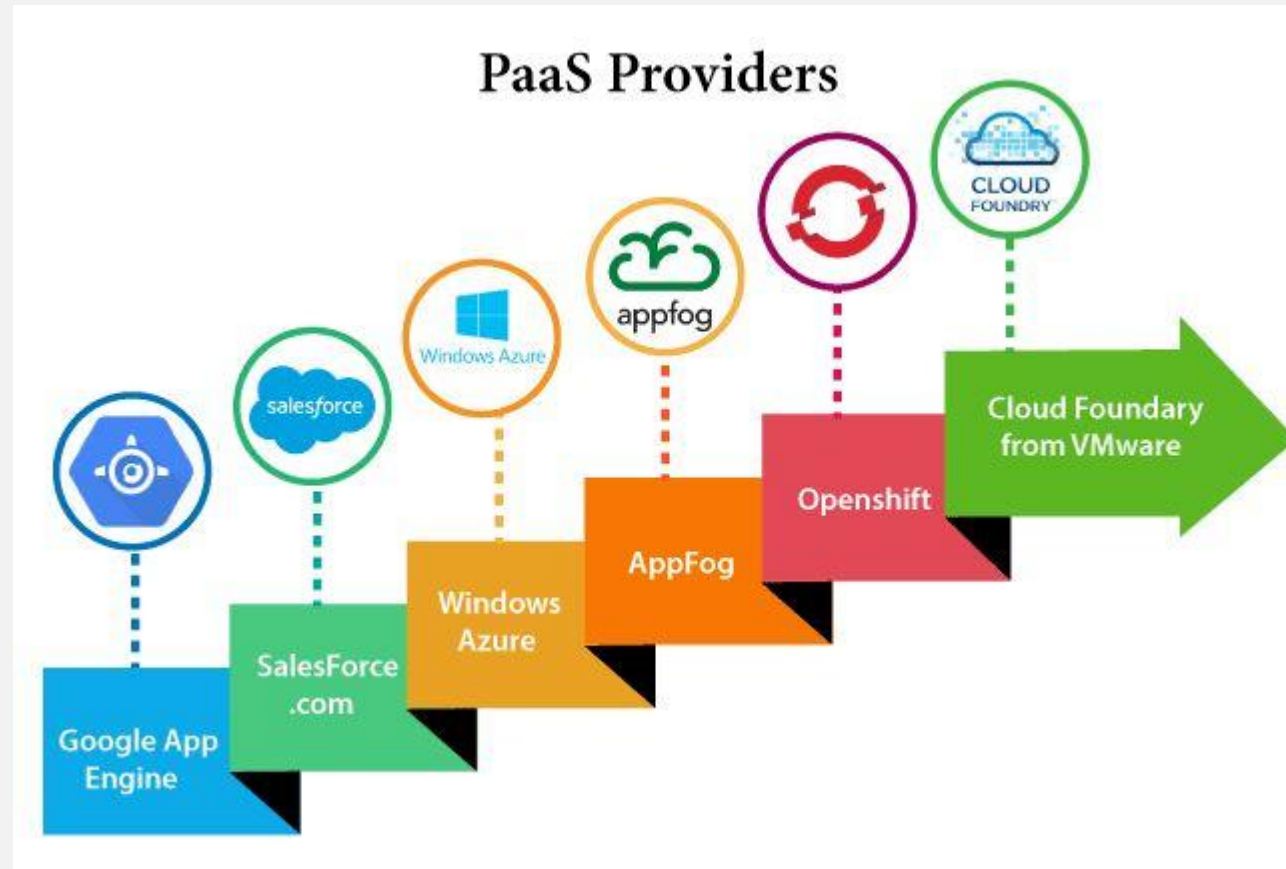
- **Advantages:**

- **Reduced Costs:** Maintaining a physical development infrastructure can be a big commitment. From the IT staff needed to maintain applications to the costly server space, a traditional environment is expensive. PaaS alleviates most of these costs, offering a more affordable alternative.
- **Improved Time to Market:** PaaS platforms often offer more options than a standard development environment can. When this includes automated tools and easier testing, it's possible to release products to market on an accelerated timeline.
- **Scalability:** If your company is growing, it's likely your development needs will change. It can be hard to properly update a physical environment, but PaaS solutions can grow with you, accommodating the evolving state of your business.
- **Regular Updates:** Software updates can be a burden, particularly when they need to be rolled out among individual machines. Most PaaS models offer regular and automatic updates, ensuring everyone is always using the most current tools for development.
- **Flexibility:** Rather than being tethered to a single machine or network, PaaS makes it possible to develop on the go, even when away from the office. PaaS virtual machines are available anywhere, at any time, meaning that developers can do more than ever before.

Platform as a Service | PaaS

- **Disadvantages:**
 - **PaaS Provider Dependence:** When using third-party architecture, as most PaaS solutions require, there's a certain reliance on a PaaS vendor that can be challenging for those who like holding the reins. If something changes or service isn't reliable, there's little that can be done. Further, seeking help for issues means reaching out to an off-site help desk rather than turning to an in-house IT professional.
 - **Compatibility Challenges:** For those who rely on other tools outside of the PaaS environment, compatibility can be an important. Before choosing a PaaS model, confirm that your selection complements the other resources you currently use or plan to use in the future.
 - **Lack of Customization:** While plenty of PaaS products allow for ways to make an environment specific to a company's own needs, there are limitations. Not all companies require extensive functionality that may be unique to in-house projects, but some do. Before moving forward with a PaaS product, companies are encouraged to evaluate their current platform and any peculiarities that could affect the transition process.
 - **Security Risks:** When managing security in house, it's possible to manage security in any manner desired. However, using a third-party means trusting that their tools are adequate to support your needs. A PaaS provider with poor security practices can put any project at risk.

Platform as a Service | PaaS



Platform as a Service | PaaS

Providers	Services
Google App Engine (GAE)	App Identity, URL Fetch, Cloud storage client library, Logservice
Salesforce.com	Faster implementation, Rapid scalability, CRM Services, Sales cloud, Mobile connectivity, Chatter.
Windows Azure	Compute, security, IoT, Data Storage.
AppFog	Justcloud.com, SkyDrive, GoogleDocs
Openshift	RedHat, Microsoft Azure.
Cloud Foundry from VMware	Data, Messaging, and other services.

Platform as a Service | PaaS

- **How PaaS works**
- In general, PaaS solutions have three main parts:
 - Cloud infrastructure including virtual machines (VMs), operating system software, storage, networking, firewalls
 - Software for building, deploying and managing applications
 - A graphic user interface, or GUI, where development or DevOps teams can do all their work throughout the entire application lifecycle
- Because PaaS delivers all standard development tools through the GUI online interface, developers can log in from anywhere to collaborate on projects, test new applications, or roll out completed products. Applications are designed and developed right in the PaaS using middleware. With streamlined workflows, multiple development and operations teams can work on the same project simultaneously.
- PaaS providers manage the bulk of your cloud computing services, such as servers, runtime and virtualization. As a PaaS customer, your company maintains management of applications and data.

PaaS Types

- **AIPaaS (PaaS for Artificial Intelligence)**
- **iPaaS (integration platform as a service)**
- **cPaaS (communications platform as a service)**
- **mPaaS (mobile platform as a service)**

PaaS Types

- **AIPaaS (PaaS for Artificial Intelligence)** lets development teams build [artificial intelligence \(AI\)](#) applications without the often prohibitive expense of purchasing, managing and maintaining the significant computing power, storage capabilities and networking capacity these applications require. AIPaaS typically includes pre-trained [machine learning](#) and [deep learning](#) models developers can use as-is or customize, and ready-made APIs for integrating specific AI capabilities, such as [speech recognition](#) or speech-to-text conversion, into existing or new applications.
- **iPaaS (integration platform as a service)** is a cloud-hosted solution for integrating applications. iPaaS provides organizations a standardized way to connect data, processes, and services across public cloud, private cloud and on-premises environments without having to purchase, install and manage their own backend integration hardware, middleware and software. (Note that PaaS solutions often include some degree of integration capability—[API management](#), for example—but iPaaS is more comprehensive.)
- **cPaaS (communications platform as a service)** is a PaaS that lets developers easily add voice (inbound and outbound calls), video (including teleconferencing) and messaging (text and social media) capabilities to applications, without investing in specialized communications hardware and software.
- **mPaaS (mobile platform as a service)** is a PaaS that simplifies application development for mobile devices. mPaaS typically provides low-code (even simple drag-and-drop) methods for accessing device-specific features including the phone's camera, microphone, motion sensor and geolocation (or GPS) capabilities.

Use cases for PaaS

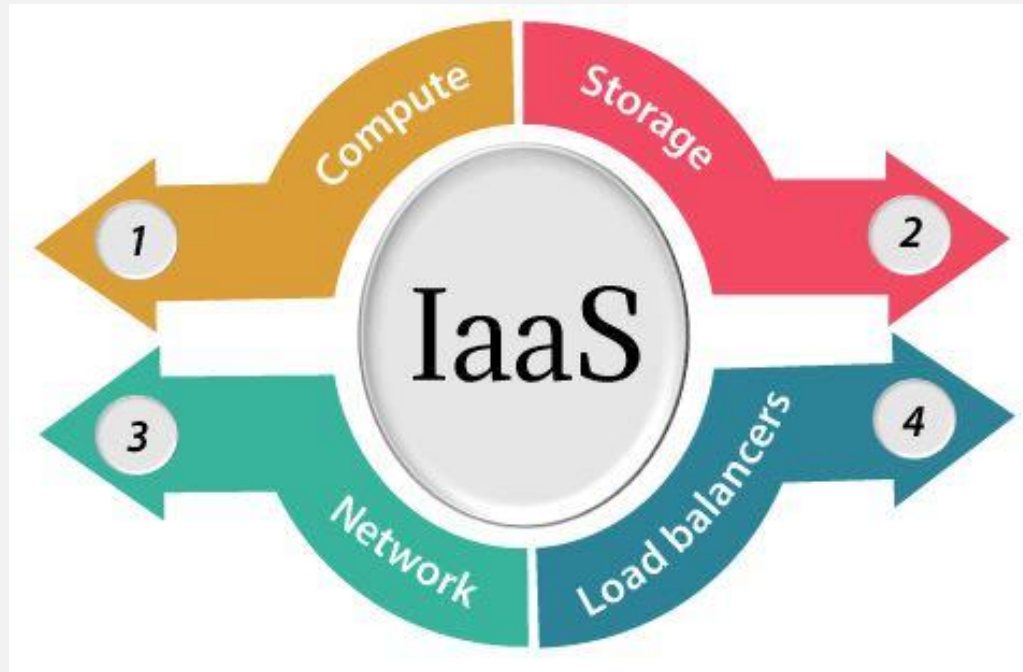
- **API development and management:** Because of its built-in frameworks, PaaS makes it much simpler for teams to develop, run, manage and secure **APIs** for sharing data and functionality between applications.
- **Internet of Things (IoT):** Out of the box, PaaS can support a range of programming languages (Java, Python, Swift, etc.), tools and application environments used for IoT application development and real-time processing of data generated by IoT devices.
- **Agile development and DevOps:** PaaS can provide fully-configured environments for automating the software application lifecycle including integration, delivery, security, testing and deployment.
- **Cloud migration and cloud-native development:** With its ready-to-use tools and integration capabilities, PaaS can simplify migration of existing applications to the cloud—particularly via *replatforming* or *refactoring*.

Infrastructure as a Service | IaaS

- **Hardware as a Service (HaaS)** - it provides resources that belong to virtualized hardware termed as computing infrastructure that offers virtual servers storage, networking connections and IP address.
- provides visualized computing features to the clients through the internet.
- Customers access these resources on the Internet using a pay-as-per use model.
- With the help of the IaaS cloud computing platform layer, clients can dynamically scale the configuration to meet changing requirements and are billed only for the services actually used.
- IaaS cloud computing platform layer eliminates the need for every organization to maintain the IT infrastructure.

Infrastructure as a Service | IaaS

- IaaS provider provides the following services -
 1. **Compute:** Computing as a Service includes virtual central processing units and virtual main memory for the Vms that is provisioned to the end- users.
 2. **Storage:** IaaS provider provides back-end storage for storing files.
 3. **Network:** Network as a Service (NaaS) provides networking components such as routers, switches, and bridges for the Vms.
 4. **Load balancers:** It provides load balancing capability at the infrastructure layer.



Advantage of IaaS

Reduce expenses	Businesses that have switched to IaaS don't have to buy, manage, and maintain their infrastructure, and they pay only for what they use—even over five-year or longer depreciation periods.
Improve business continuity	Cloud infrastructure typically provides a higher degree of uptime and more disaster recovery options than on-premises deployments because it has redundancy built in at every layer, offers multiple fault domains, and geographically distributed locations, and is run at massive scale by operations experts.
Accelerate innovation	IaaS makes it fast, easy, and affordable to test new products and ideas. Instead of having to develop detailed forecasts and invest in new infrastructure, businesses can ramp up their cloud infrastructure in minutes, then scale up or down as needed.
Take advantage of the latest technologies	Many cloud providers package and deploy new hardware and software—including artificial intelligence and machine learning frameworks—long before businesses can implement them on-premises.
Speed provisioning	Even virtualized on-premises infrastructures suffer from long provisioning times of weeks or even months. With IaaS, entire application environments can be provisioned in minutes.
Focus on core business	IaaS frees IT departments from spending as much as half of their resources on managing and maintaining on-premises hardware and software. With IaaS, organizations can also enable DevOps and other teams to access the infrastructure themselves, so they can run and test without delay.
Scale faster	Businesses need more resources during peak workloads—such as during monthly reporting periods. With IaaS, the infrastructure can scale in minutes, so reports can be run quickly and staff can focus on more business-critical activities.

Who uses IaaS solutions?

Test and development	With IaaS, DevOps teams can set up and take down test and development environments quickly and at low cost, so they can get new applications to market faster.
Traditional applications	IaaS supports both cloud native applications and traditional enterprise applications, including ERP and business analytics applications.
Website hosting and apps	Many businesses run their websites on IaaS to optimize costs. IaaS also supports web and mobile apps, which can be quickly deployed and scaled.
Storage, backup, and recovery	Storing and backing up data on-premises, as well as planning for and recovering from disasters, requires a great deal of time and expertise. Moving their infrastructure to the cloud helps businesses reduce costs and frees them up to focus on other tasks.
High performance computing	With its pay-as-you-go model, IaaS makes high performance computing (HPC) and other data-intensive, project-oriented tasks more affordable.

Examples of IaaS

- A suitable example is Amazon Web Services (AWS EC2), where it provides scalability to host cloud-based applications. Moreover EC2 users need not have physical servers, AWS provides a virtual environment to work on. In this, the cost is minimized and the users only pay for the services they booked.
- In the Case of the E-commerce platform, it depends upon the user's interest in hosting the applications either on cloud or on-premise. Here also the users pay for the services actually used (i.e. hosting plan for the server).
- A virtualized Data Center is established to provide cloud hosting options, integrating the cloud operations. The Data Center contains several Virtual servers that meet the user demand as per their business requirements.
- Another cloud computing service provider known as Digital Ocean, founded in the year 2011, which provides IaaS (Infrastructure as a Service) for open source developers. Mostly Digital Ocean provides droplets, where a developer can resize the droplets after creating them. Developers can scale and grow their business through the Digital Ocean more efficiently.
- IaaS can manage big data to handle large workloads and integrate with BI tools.
- GCE (Google Compute Engine) is an IaaS component that runs search engine, Gmail and other services.

IaaS Providers



IaaS Providers

IaaS Vendor	IaaS Solution	Details
Amazon Services	Web Elastic, Elastic Compute Cloud (EC2) MapReduce, Route 53, Virtual Private Cloud, etc.	The cloud computing platform pioneer, Amazon offers auto scaling, cloud monitoring, and load balancing features as part of its portfolio.
Netmagic Solutions	Netmagic IaaS Cloud	Netmagic runs from data centers in Mumbai, Chennai, and Bangalore, and a virtual data center in the United States. Plans are underway to extend services to West Asia.
Rackspace	Cloud servers, cloud files, cloud sites, etc.	The cloud computing platform vendor focuses primarily on enterprise-level hosting services.
Reliance Communications	Reliance Internet Data Center	RIDC supports both traditional hosting and cloud services, with data centers in Mumbai, Bangalore, Hyderabad, and Chennai. The cloud services offered by RIDC include IaaS and SaaS.
Sify Technologies	Sify IaaS	Sify's cloud computing platform is powered by HP's converged infrastructure. The vendor offers all three types of cloud services: IaaS, PaaS, and SaaS.
Tata Communications	InstaCompute	InstaCompute is Tata Communications' IaaS offering. InstaCompute data centers are located in Hyderabad and Singapore, with operations in both countries.

BMaaS vs. IaaS

- A bare metal server is a computer server that hosts one tenant, used by a single customer and not shared between consumers—a single tenant server.
- IaaS is a cloud computing model in which storage, compute, and networking resources are available on demand over the internet through an IaaS provider.
- BMaaS can be considered a subset of IaaS that refers specifically to the practice of providing access to dedicated servers free of any installed operating systems or virtualization infrastructure—bare-metal. The customer may be provided access to the physical server hardware either remotely or on-premises.
- Compared to the traditional cloud model, where multiple users (multi-tenancy) may reside on the same physical server within a cloud vendor's data center, the bare metal server only has one customer on the server (a single tenant server).
- Access is usually provisioned on-demand via a single contract or service level agreement (SLA).

BMaaS vs. IaaS

- The primary advantage of BMaaS is that you get direct access to system hardware. This means applications that require access to system performance counters to run in a cloud environment may do so.
- Additionally BMaaS gives you all the benefits of a traditional dedicated server:
- Dedicated resources—you don't have to compete for resources with other tenants hosted on the same server
- High performance and security—eliminates the possibility you'll be sharing a server with a malicious tenant.
- Control over your server's configuration and software installations.
- On-demand provisioning of storage, compute, and networking resources

Pricing

- IaaS is typically priced on a consumption basis, meaning users are only charged for what they use.
- Subscriptions and reserved instances: Many providers offer discounts off the sticker price for clients willing to commit to longer contract terms, typically around one to three years.
- Monthly billing: Monthly billing models are most common in the BMaaS market, where physical infrastructure typically implies steady state workloads without spiky characteristics.
- By the hour/second: The most common granularity for traditional cloud infrastructure, end users are charged only for what they use.
- Transient/spot: Some providers will offer up unused capacity at a discount via transient/spot instances, but those instances can be reclaimed if the capacity is needed.

SaaS (Software as a Service)

- Software as a service (SaaS) is a way of delivering applications remotely over the internet instead of locally on machines (known as “on-premise” software). SaaS applications are also known as:
 - Web-based software
 - On-demand software
 - Hosted software
- SaaS, is application software hosted on the cloud and used over an internet connection via a web browser, mobile app or thin client. The SaaS provider is responsible for operating, managing and maintaining the software and the infrastructure on which it runs. The customer simply creates an account, pays a fee, and gets to work.
- Traditionally, IT departments installed business software on employees’ computers and had to manually update and maintain it. Users had only so much storage on their hard drives, and companies had to buy individual licenses for everyone. That made it difficult to collaborate, to work from a communal database, and to update data as needed.
- With SaaS, applications live on software providers’ servers. That frees users from licensing and hardware limitations, as well as maintenance responsibilities.

How does software as a service work?

- With software as a service, you don't install and maintain software. Instead, you log in using the internet. The provider manages application security, access, storage, and performance. This empowers your IT department to spend more time on strategic work instead of software and hardware management.
- The SaaS software vendor can host the application on its own cloud infrastructure or with a cloud service provider (such as Amazon Web Services (AWS), Google Cloud, IBM Cloud or Microsoft Azure).
- SaaS applications are accessible to any customer with an internet connection and an internet-connected end-user device (e.g. a computer, mobile phone or tablet). SaaS applications typically run in any web browser; on mobile devices.
- A few SaaS applications, such as Adobe Acrobat, may offer or require a dedicated thin client that users download and install on their computers.
- Industry analyst Gartner forecasts worldwide SaaS software revenues to exceed USD 145 billion by the end of 2022 ; another analyst, International Data Corporation (IDC), projects the worldwide market for SaaS will grow to USD 302.1 billion by 2025 .

How does software as a service work?

- SaaS applications exploit multi-tenant architecture, in which a single instance of the application serves every customer. For security and data privacy, each customers' application data, user data, system data and custom configurations are segregated from those of other customers.
- SaaS applications require little to no management and zero maintenance from the customer.
- The SaaS vendor is responsible for
 - Provisioning, managing and maintaining all the servers, networking equipment, storage hardware and operating software required to run the application
 - Applying feature fixes and security patches as needed
 - Providing load balancing, redundant infrastructure, data backup, cloud security and disaster recovery services to prevent outages and meet the performance, availability and data protection standards specified in the service level agreement (SLA).

SaaS Pricing

- SaaS applications are available in a wide variety of subscription pricing schemes—all of them very different from traditional software licensing, which provided access to the software and related services for a specified period of for a fixed price (or menu of fixed prices).
- **Free or ‘freemium’ pricing**
 - Most SaaS vendors offer a free version of the software that customers can try for a short period of time, usually a week or a month.
 - Some SaaS companies offer ‘freemium’ versions of their applications: versions that are always free, with a limited but useful subset of the full products features
- **Monthly or annual tiered subscription pricing**
 - For small and mid-sized businesses, SaaS products typically offer pricing ‘tiers’ based on feature set or usage level. For example, a SaaS email marketing solution may offer a ‘standard’ tier for up to 5,000 contacts for USD 250 per month, a ‘pro’ tier for up to 20,000 contacts for USD 500 per month, and an ‘business’ tier for 100,000 contacts plus landing page design and hosting for USD 600 per month. For very large companies SaaS vendors my offer an ‘enterprise’ tier that allows a more granular selection of features, priority support, special SLA terms, and even negotiable pricing.

Advantages of SaaS

- **SaaS provides faster—even instant—adoption and time-to-benefit.** Customers can purchase and start using SaaS applications immediately, sometimes in minutes, for a minimal upfront cost (essentially the first month's subscription cost). Compare to traditional software, which could require purchasing and provisioning servers, installing software on every end-user device, and budgeting for and purchasing a full license for every user.
- **SaaS provides access to new features and versions as soon as they're available.** SaaS providers often upgrade features and add functionality several times a week, without customers even noticing; they can even upgrade the interface and user experience without disrupting the customers' work. Compare to traditional on-premises software, for which periodic upgrades are often so costly and disruptive that customers might wait months for the functionality in a new version (if they don't choose to skip some upgrades altogether).
- **SaaS enables cost-effective, on-demand scalability.** Customers can scale SaaS applications up and down as needed, by simply upgrading or downgrading tiers or purchasing more capacity. Compare to traditional software, which requires customers to purchase additional capacity in anticipation of usage spikes—capacity that sits idle and wasted until needed.
- **SaaS offers predictable costs and dramatically lower overhead.** With SaaS there's no need to budget for infrastructure on which to run the software, for periodic software upgrades and the infrastructure to support *them*, and—perhaps most important—for in-house IT staff to install, upgrade and maintain the software. Almost all of the expense goes directly to use of the software.

Everything as a Service (XaaS)

- Everything as a Service (XaaS) means anything can now be a service with the help of cloud computing and remote accessing.
- Where cloud computing technologies provide different kinds of services over the web networks. In Everything as a Service, various tools and technologies, and services are provided to users as a service.
- Before XaaS and cloud services, companies have to buy licensed products and install them, had to all securities on their site and provide infrastructure for business purposes.
- With XaaS, business is simplified as they have to pay for what they need. This Everything as a Service is also known as Anything as a Service.

Examples of XaaS

1. Software as a Service (SaaS)
2. Platform as a Service (PaaS)
3. Disaster Recovery as a Service (DRaaS)
4. Infrastructure as a service (IaaS)
5. Communication as a Service (CaaS)
6. Network as a Service (NaaS)
Database as a Service (DBaaS)
7. Desktop as a Service (DaaS) etc.

Examples of XaaS

- Hardware as a Service (HaaS) –
Managed Service Providers (MSP) provide and install some hardware on the customer's site on demand. The customer uses the hardware according to service level agreements. This model is very similar to IaaS as computing resources present at MSP's site are provided to users substituted for physical hardware.
- Communication as a Service (CaaS) –
This model comprises solutions for different communication like IM, VoIP, and video conferencing applications which are hosted in the provider's cloud. Such a method is cost-effective and reduces time expenses.
- Desktop as a Service (DaaS) –
DaaS provider mainly manages storing, security, and backing up user data for desktop apps. And a client can also work on PCs using third-party servers.

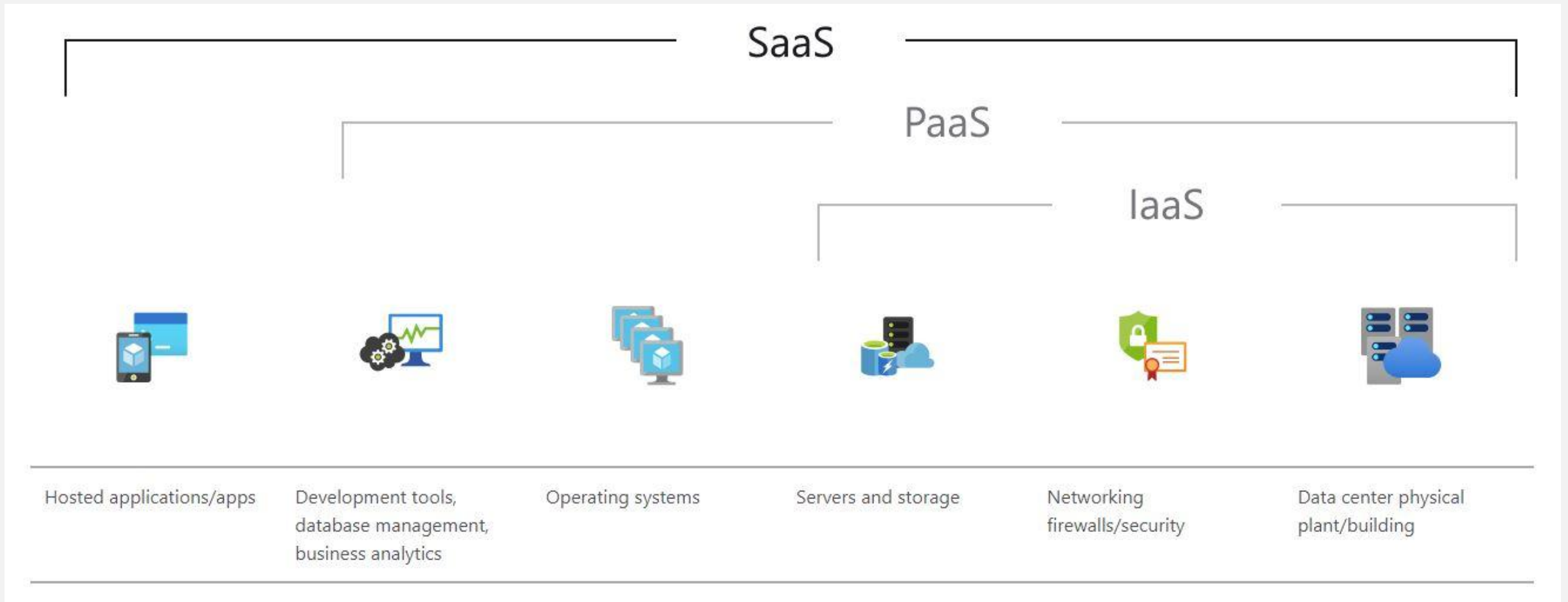
Examples of XaaS

- Security as a Service (SECaaS) –
In this method, the provider integrates security services with the company's infrastructure through the internet which includes anti-virus software, authentication, encryption, etc.
- Healthcare as a Service (HaaS) –
The healthcare industry has opted for the model HaaS service through electronic medical records (EMR). IoT and other technologies have enhanced medical services like online consultations, health monitoring 24/7, medical service at the doorstep e.g. lab sample collection from home, etc.
- Transport as a Service (TaaS) –
Nowadays, there are numerous apps that help in mobility and transport in modern society. The model is both convenient and ecological friendly e.g. Uber taxi services is planning to test flying taxis and self-driving planes in the future.

Advantages of XaaS

- **Cost Saving –**
When an organization uses XaaS then it helps in cost-cutting and simplifies IT deployments.
- **Scalability –**
XaaS can easily handle the growing amount of work by providing the required resources/service.
- **Accessibility –**
It helps in easy accessing and improving accessibility as long as the internet connection is there.
- **Faster Implementation –**
It provides faster implementation time to various activities of the organization.
- **Quick Modification –**
It provides updates for modification as well as undergoes quick updating by providing quality services.
- **Better Security –**
It contains improved security controls and is configured to the exact requirements of the business.
- **Boost innovation –**
While XaaS is used it Streamlines the operations and frees up resources for innovation.
- **Flexibility –**
XaaS provides flexibility by using cloud services and multiple advanced approaches.

Summary



service-level agreement (SLA)

A service-level agreement (SLA) is a contract between a service provider and its customers that documents what services the provider will furnish and defines the service standards the provider is obligated to meet.

A service-level commitment (SLC) is a broader and more generalized form of an SLA. The two differ because an SLA is bidirectional and involves two teams. In contrast, an SLC is a single-directional obligation that establishes what a team can guarantee its customers at any given time.

SLA

A service-level agreement (SLA) defines the level of service you expect from a vendor, laying out the metrics by which service is measured, as well as remedies or penalties should agreed-on service levels not be achieved. It is a critical component of any technology vendor contract.

SLAs are a critical component of any [outsourcing](#) and technology vendor contract. Beyond listing expectations of service type and quality, an SLA provides remedies when requirements aren't met.

A telecom company's SLA, for example, may promise network availability of 99.999 percent (that works out to about five and a quarter minutes of downtime per year), and allow the customer to reduce their payment by a given percentage if that is not achieved, usually on a sliding scale based on the magnitude of the breach.

Benefits of SLA



Benefits of SLA

They clearly state metrics, responsibilities and expectations so that, in the event of issues with the service, neither party can plead ignorance. It ensures both sides have the same understanding of requirements.

The SLA protects both parties in the agreement.

Ideally, SLAs should be aligned to the technology or business objectives of the engagement. Misalignment can have a negative impact on deal pricing, quality of service delivery, and customer experience.

Types of SLA

Customer SLAs

The most common (or at least the most well-known) type of service level agreement, the customer SLA is a contract between a service provider and an external customer. This is sometimes also called an external service agreement.

A customer service-level agreement includes:

- exact details of the service expected by the customer;
- provisions of the service availability;
- standards for each level of service;
- each party's responsibilities;
- escalation procedures; and
- terms for cancellation.

Types of SLA

Internal SLAs

Internal SLAs are designed to establish and adhere to service standards within a specific company or organization. These may function between teams or departments and help ensure that different groups that depend on each other within the business are meeting vital goals.

Multilevel SLAs

For occasions when there is more than one service provider or more than one end user, a multilevel SLA can divide the contract into multiple levels. These can be applied to internal or external customers and may also cater to different price ranges for varying levels of service within a single product.

Key Components of an SLA

Agreement overview. This first section sets forth the basics of the agreement, including the parties involved, the start date and a general introduction of the services provided.

Description of services. The SLA needs detailed descriptions of every service offered, under all possible circumstances, with the turnaround times included. Service definitions should include how the services are delivered, whether maintenance service is offered, what the hours of operation are, where dependencies exist, an outline of the processes and a list of all technology and applications used.

Exclusions. Specific services that are not offered should also be clearly defined to avoid confusion and eliminate room for assumptions from other parties.

Service performance. Performance measurement metrics and performance levels are defined. The client and service provider should agree on a list of

Key Components of an SLA

Redressing. Compensation or payment should be defined if a provider cannot properly fulfill their SLA.

Stakeholders. Clearly defines the parties involved in the agreement and establishes their responsibilities.

Security. All security measures that will be taken by the service provider are defined. Typically, this includes the drafting and consensus on antipoaching, IT security and nondisclosure agreements.

Risk management and disaster recovery. Risk management processes and a [disaster recovery plan](#) are established and clearly communicated.

Service tracking and reporting. This section defines the reporting structure, tracking intervals and [stakeholders](#) involved in the agreement.

Key Components of an SLA

Periodic review and change processes. The SLA and all established key performance indicators ([KPIs](#)) should be regularly reviewed. This process is defined as well as the appropriate process for making changes.

Termination process. The SLA should define the circumstances under which the agreement can be terminated or will expire. The notice period from either side should also be established.

Signatures. Finally, all stakeholders and authorized participants from both parties must sign the document to show their approval of every detail and process.

Billing & Accounting

In cloud accounting both the accounting system and business data is stored in the cloud means on internet. It is a cloud based approach of performing accounting activities.

Advantages of Cloud Accounting :

- Allows multiple users to use at a time.
- No need of any software installation.
- Provides automatic update and backup.
- Allows accessing data from multiple devices.

Disadvantages of Cloud Accounting :

- Requires internet connection to work.
- Cost is typically higher.
- Doing work may be slow if your connection is slow.
- Requires regular fee in advance or later.

Billing & Accounting

- Each available cloud model has its own spin on how resource allocation is determined, and that spin is different from traditional IT business models in terms of affordability and the expense model in use.
- For example, the number of message queue GET and PUT operations per request can provide a cost structure for each customer that can in turn be accumulated for a total cost per transaction and ultimately per customer per month (similar to a mobile phone bill).

Billing & Accounting

Account Activity		Welcome Jason Meiers Sign Out Account Number			
View Previous Statement					
This Month's Activity as of July 8, 2011					
The billing cycle for this report is July 1 - July 31, 2011. The AWS service usage charges on this page currently show activity for all accounts through approximately 07/08/2011					
Summary	Activity by Account				
You can download a detailed activity report in Comma Separated Value (CSV) format.		Download Report			
Expand All Services Collapse All Services		Printer Friendly Version			
			Totals		
Amazon Elastic Compute Cloud					
US East (Northern Virginia) Region					
Amazon EC2 running Linux/UNIX Reserved Instances					
\$0.03 per Small Instance (m1.small) instance-hour (or partial hour)		108 Hrs	5.64		
Amazon EC2 running Linux/UNIX					
\$0.085 per Small Instance (m1.small) instance-hour (or partial hour)		3 Hrs	0.26		
Amazon EC2 EBS					
\$0.10 per GB-month of provisioned storage		2,978 GB-Mo	0.30		
\$0.10 per 1 million I/O requests		\$19,963 IOs	0.03		
\$0.01 per 10,000 gets (when loading a snapshot)		14,336 Requests	0.01		
Download Usage Report			6.26		
Amazon Simple Storage Service					
Download Usage Report			0.05		
AWS Data Transfer (excluding Amazon CloudFront)					
			0.01		
Bill Summary					
Usage charges and monthly recurring fees during this billing cycle† (More Info)			\$6.32		
One-time fees during this billing cycle (More Info)			\$0.00		
Taxes Estimated Taxes			\$0.00		
Total new charges this billing cycle			\$6.32		
No payments received to date.					
Current estimated unpaid balance to be charged for this billing cycle			\$6.32		

Billing & Accounting

Infrastructure as a service and billing and metering services

The primary concepts of IaaS include:

- Servers per hour serving an on-demand model
- Reserved servers for better planning
- Higher and lower compute resource units based on application performance
- Volume-based metering on the number of instances consumed
- Prepaid and reserved infrastructure resources
- Clustered server resources

Billing & Accounting

Platform as a service and billing and metering services

The primary concepts for PaaS metering and billing include:

- Incoming and outgoing network bandwidth
- CPU time per hour
- Stored data
- High availability
- Monthly service charge

Billing & Accounting

SaaS and billing and metering services

The primary concepts for SaaS billing and metering include:

- Monthly subscription fees
- Per-user monthly fees