Cloud service models
with examples

Cloud service models

Main:

- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)
- Software as a Service (SaaS)

Others:

- Function as a Service (FaaS) – a.k.a. *Serverless*
- Bare Metal as a Service (BMaaS)
- Database as a Service (DBaaS)
- ...
- Anything as a Service (XaaS)

Service models comparison

Traditional IT	IaaS	PaaS	SaaS
Applications	Applications	Applications	Applications
Data	Data	Data	Data
Runtime	Runtime	Runtime	Runtime
Middleware	Middleware	Middleware	Middleware
O/S	O/S	O/S	O/S
Virtualization	Virtualization	Virtualization	Virtualization
Servers	Servers	Servers	Servers
Storage	Storage	Storage	Storage
Networking	Networking	Networking	Networking

IBM Cloud

You Manage   Vendor Managed

IaaS – Infrastructure as a Service

*The capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where **the consumer is able to deploy and run arbitrary software**, which can include operating systems and applications.*

*The consumer does not manage or control the underlying cloud infrastructure but has **control over operating systems, storage, and deployed applications**; and possibly limited control of select networking components (e.g., host firewalls).*

Users buy Virtual Machines (= virtualized hardware) that they configure and manage

Users are like **IT administrators** (system administrators)

source: NIST Definition of Cloud Computing (2011)

Common IaaS business scenarios

Typical things businesses do with IaaS include:

- **Test and development.** Teams can quickly set up and dismantle test and development environments, bringing new applications to market faster. IaaS makes it quick and economical to scale up dev-test environments up and down.

- **Website hosting.** Running websites using IaaS can be less expensive than traditional web hosting.

- **Storage, backup, and recovery.** Organizations avoid the capital outlay for storage and complexity of storage management, which typically requires a skilled staff to manage data and meet legal and compliance requirements. IaaS is useful for handling unpredictable demand and steadily growing storage needs. It can also simplify planning and management of backup and recovery systems.

- **Web apps.** IaaS provides all the infrastructure to support web apps, including storage, web and application servers, and networking resources. Organizations can quickly deploy web apps on IaaS and easily scale infrastructure up and down when demand for the apps is unpredictable.

- **High-performance computing.** High-performance computing (HPC) on supercomputers, computer grids, or computer clusters helps solve complex problems involving millions of variables or calculations. Examples include earthquake and protein folding simulations, climate and weather predictions, financial modeling, and evaluating product designs.

- **Big data analysis.** Big data is a popular term for massive data sets that contain potentially valuable patterns, trends, and associations. Mining data sets to locate or tease out these hidden patterns requires a huge amount of processing power, which IaaS economically provides.

source: https://azure.microsoft.com/en-us/overview/what-is-iaas/

Advantages of IaaS

- **Eliminates capital expense and reduces ongoing cost.** IaaS sidesteps the upfront expense of setting up and managing an onsite datacenter, making it an economical option for start-ups and businesses testing new ideas.

- **Improves business continuity and disaster recovery.** Achieving high availability, business continuity, and disaster recovery is expensive, since it requires a significant amount of technology and staff. But with the right service level agreement (SLA) in place, IaaS can reduce this cost and access applications and data as usual during a disaster or outage.

- **Innovate rapidly.** As soon as you've decided to launch a new product or initiative, the necessary computing infrastructure can be ready in minutes or hours, rather than the days or weeks—and sometimes months—it could take to set up internally.

- **Respond quicker to shifting business conditions.** IaaS enables you to quickly scale up resources to accommodate spikes in demand for your application— during the holidays, for example—then scale resources back down again when activity decreases to save money.

- **Focus on your core business.** IaaS frees up your team to focus on your organization's core business rather than on IT infrastructure.

- **Increase stability, reliability, and supportability.** With IaaS there's no need to maintain and upgrade software and hardware or troubleshoot equipment problems. With the appropriate agreement in place, the service provider assures that your infrastructure is reliable and meets SLAs.

- **Better security.** With the appropriate service agreement, a cloud service provider can provide security for your applications and data that may be better than what you can attain in-house.

- **Gets new apps to users faster.** Because you don't need to first set up the infrastructure before you can develop and deliver apps, you can get them to users faster with IaaS.

source: https://azure.microsoft.com/en-us/overview/what-is-iaas/

Example: Amazon EC2

	vCPU	ECU	Memory (GiB)	Instance Storage (GB)	Linux/UNIX Usage
Region: US East (Ohio)					
General Purpose – Current Generation					
a1.medium	1	N/A	2 GiB	EBS Only	$0.0255 per Hour
a1.large	2	N/A	4 GiB	EBS Only	$0.051 per Hour
a1.xlarge	4	N/A	8 GiB	EBS Only	$0.102 per Hour
a1.2xlarge	8	N/A	16 GiB	EBS Only	$0.204 per Hour
a1.4xlarge	16	N/A	32 GiB	EBS Only	$0.408 per Hour
a1.metal	16	N/A	32 GiB	EBS Only	$0.408 per Hour
t4g.nano	2	N/A	0.5 GiB	EBS Only	$0.0042 per Hour
t4g.micro	2	N/A	1 GiB	EBS Only	$0.0084 per Hour

Tabs: Linux | RHEL | SLES | Windows | Windows with SQL Standard | Windows with SQL Web | Windows with SQL Enterprise | Linux with SQL Standard | Linux with SQL Web | Linux with SQL Enterprise

https://aws.amazon.com/ec2/pricing/on-demand/

Example: IBM Cloud Virtual Server Instances (VSI)

https://cloud.ibm.com/vpc-ext/provision/vs

PaaS – Platform as a Service

*The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired **applications created using programming languages, libraries, services, and tools supported by the provider**.*

*The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but **has control over the deployed applications** and possibly configuration settings for the application-hosting environment.*

- Users buy already configured development environments (e.g. containers) and computing resources
- Users are **software developers**

source: NIST Definition of Cloud Computing (2011)

Common PaaS business scenarios

Organizations typically use PaaS for these scenarios:

- **Development framework.** PaaS provides a framework that developers can build upon to develop or customize cloud-based applications. Similar to the way you create an Excel macro, PaaS lets developers create applications using built-in software components. Cloud features such as scalability, high-availability, and multi-tenant capability are included, reducing the amount of coding that developers must do.

- **Analytics or business intelligence.** Tools provided as a service with PaaS allow organizations to analyze and mine their data, finding insights and patterns and predicting outcomes to improve forecasting, product design decisions, investment returns, and other business decisions.

- **Additional services.** PaaS providers may offer other services that enhance applications, such as workflow, directory, security, and scheduling.

source: https://azure.microsoft.com/en-us/overview/what-is-paas/

Advantages of PaaS

By delivering infrastructure as a service, PaaS offers the same advantages as IaaS. But its additional features—middleware, development tools, and other business tools—give you more advantages:

- **Cut coding time.** PaaS development tools can cut the time it takes to code new apps with pre-coded application components built into the platform, such as workflow, directory services, security features, search, and so on.

- **Add development capabilities without adding staff.** Platform as a Service components can give your development team new capabilities without your needing to add staff having the required skills.

- **Develop for multiple platforms—including mobile—more easily.** Some service providers give you development options for multiple platforms, such as computers, mobile devices, and browsers making cross-platform apps quicker and easier to develop.

- **Use sophisticated tools affordably.** A pay-as-you-go model makes it possible for individuals or organizations to use sophisticated development software and business intelligence and analytics tools that they could not afford to purchase outright.

- **Support geographically distributed development teams.** Because the development environment is accessed over the Internet, development teams can work together on projects even when team members are in remote locations.

- **Efficiently manage the application lifecycle.** PaaS provides all of the capabilities that you need to support the complete web application lifecycle: building, testing, deploying, managing, and updating within the same integrated environment.

source: https://azure.microsoft.com/en-us/overview/what-is-paas/

Example: Heroku

The Heroku network runs the customer's apps in virtual containers which execute on a reliable runtime environment. Heroku calls these containers "Dynos." These Dynos can run code written in Node, Ruby, PHP, Go, Scala, Python, Java, or Clojure. Heroku also provides custom buildpacks with which the developer can deploy apps in any other language. Heroku lets the developer scale the app instantly just by either increasing the number of dynos or by changing the type of dyno the app runs in.
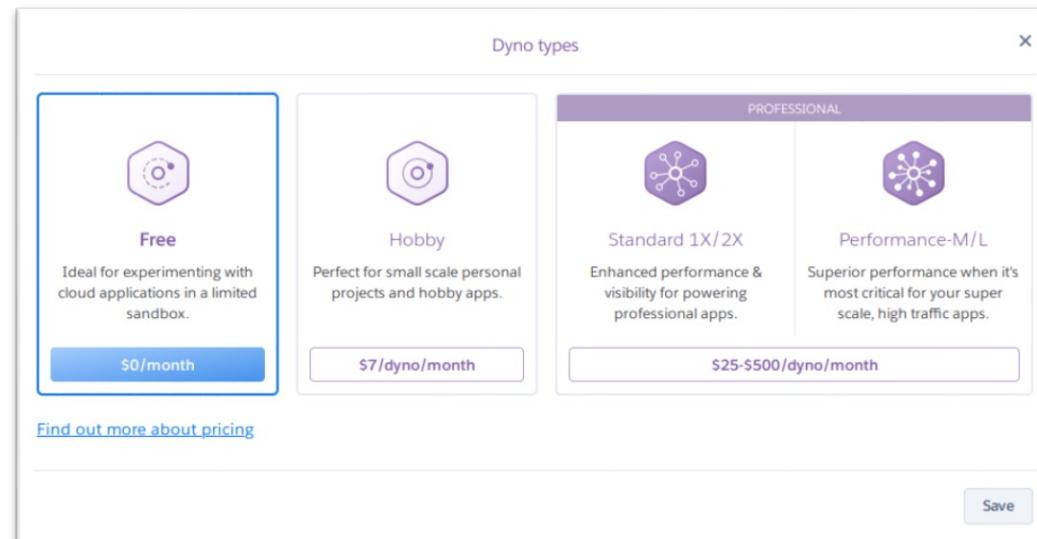
https://www.heroku.com/pricing

Example: IBM Cloud Code Engine

IBM Cloud Code Engine is a fully managed, serverless platform that runs containerized workloads (such as web apps, micro-services, event-driven functions, batch jobs, ...).

Code Engine can build container images directly from the source code. All these workloads are all hosted within the same Kubernetes infrastructure.

The Code Engine experience is designed so that developers can focus on writing code and not on the infrastructure that is needed to host it.

Features

A single runtime environment. Any workload. Scale on demand.

Pick the type of workload that best fits your needs without learning multiple serverless products.

Fully managed infrastructure	**Autoscaling down to zero**	**Simplified app security**	**No Kubernetes training needed**
Open source frameworks for container orchestration, service mesh and more — all managed and optimized for you.	Automatically scale your workloads up and down – even down to zero – and pay only for what you use.	Secure networking, pod security policies, resource quotas and limit ranges.	Cluster complexity is invisible to you. Focus on your code and not on configuring and securing clusters or VMs.

https://www.ibm.com/cloud/code-engine

SaaS – Software as a Service

*The capability provided to the consumer is to **use the provider's applications** running on a cloud infrastructure. The applications are accessible from various client devices through either a thin client interface, such as a web browser (e.g., web-based email), or a program interface.*

The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings.

- Users buy subscriptions to use the software
- Users are **end users** of the software

source: NIST Definition of Cloud Computing (2011)

Common SaaS business scenarios

If you've used a web-based email service such as Outlook, Hotmail, or Yahoo! Mail, then you've already used a form of SaaS. With these services, you log into your account over the Internet, often from a web browser. The email software is located on the service provider's network, and your messages are stored there as well. You can access your email and stored messages from a web browser on any computer or Internet-connected device.

The previous examples are free services for personal use. For organizational use, you can rent productivity apps, such as email, collaboration, and calendaring; and sophisticated business applications such as customer relationship management (CRM), enterprise resource planning (ERP), and document management. You pay for the use of these apps by subscription or according to the level of use.

source: https://azure.microsoft.com/en-us/overview/what-is-saas/

Advantages of SaaS

- **Gain access to sophisticated applications.** To provide SaaS apps to users, you don't need to purchase, install, update, or maintain any hardware, middleware, or software. SaaS makes even sophisticated enterprise applications, such as ERP and CRM, affordable for organizations that lack the resources to buy, deploy, and manage the required infrastructure and software themselves.

- **Pay only for what you use.** You also save money because the SaaS service automatically scales up and down according to the level of usage.

- **Use free client software.** Users can run most SaaS apps directly from their web browser without needing to download and install any software, although some apps require plugins. This means that you don't need to purchase and install special software for your users.

- **Mobilize your workforce easily.** SaaS makes it easy to "mobilize" your workforce because users can access SaaS apps and data from any Internet-connected computer or mobile device. You don't need to worry about developing apps to run on different types of computers and devices because the service provider has already done so. In addition, you don't need to bring special expertise onboard to manage the security issues inherent in mobile computing. A carefully chosen service provider will ensure the security of your data, regardless of the type of device consuming it.

- **Access app data from anywhere.** With data stored in the cloud, users can access their information from any Internet-connected computer or mobile device. And when app data is stored in the cloud, no data is lost if a user's computer or device fails.

source: https://azure.microsoft.com/en-us/overview/what-is-saas/

Example: Microsoft 365

https://www.microsoft.com/en-us/microsoft-365/business/compare-all-microsoft-365-business-products

FaaS – Function as a Service

The capability provided to the consumer is to run on the cloud infrastructure consumer-created functions that are triggered by some events without the complexity of building and maintaining the infrastructure typically associated with developing and launching an app. Building an application following this model is one way of achieving a "serverless" architecture, and is typically used when building microservices applications.

- Users pay depending on the times the function is called and the amount of the resources (e.g. RAM) it uses

- Users are software developers

BMaaS – Bare Metal as a Service

Same as IaaS but without the virtualization layer.

Bare-metal servers have a single 'tenant'. They are not shared between users. Bare-metal cloud servers do not run a hypervisor, are not virtualized -- **but can still be delivered via a cloud-like service model**.

- Users buy computing time of an entire physical machine.
- Users are like IT administrators.

DBaaS – DataBase as a Service

The capability provided to the consumer is to setup, operate and scale databases using a common set of abstractions (primitives), without having to either know nor care about the exact implementations of those abstractions for the specific database.

Similar to SaaS or to PaaS, depending on whether a Database is considered a software or a platform.

- Users buy storage space (e.g. $0.10 per GB-month) and I/O rate (e.g. $0.20 per 1 million requests) and data transfer rate (e.g. $0.09 per GB)
- Users are software developers

Questions, comments?