

Aim

To study and use any SaaS (software as a service)

Case Study

Definition

Software as a service (SaaS) is a software distribution model in which a cloud provider hosts applications and makes them available to end users over the internet. In this model, an independent software vendor (ISV) may contract a third-party cloud provider to host the application. Or, with larger companies, such as Microsoft, the cloud provider might also be the software vendor.

SaaS is one of three main categories of cloud computing, alongside infrastructure as a service (**IaaS**) and platform as a service (**PaaS**). A range of IT professionals, business users and personal users use SaaS applications. Products range from personal entertainment, such as Netflix, to advanced IT tools. Unlike IaaS and PaaS, SaaS products are frequently marketed to both B2B and B2C users.

How does SaaS work?

SaaS works through the cloud delivery model. A software provider will either host the application and related data using its own servers, databases, networking, and computing resources, or it may be an ISV that contracts a cloud provider to host the application in the provider's data centre. The application will be accessible to any device with a network connection. SaaS applications are typically accessed via web browsers.

As a result, companies using SaaS applications are not tasked with the setup and maintenance of the software. Users simply pay a subscription fee to gain access to the software, which is a ready-made solution.

SaaS is closely related to the application service provider (**ASP**) and on-demand computing software delivery models where the provider hosts the customer's software and delivers it to approved end users over the internet.

In the software-on-demand SaaS model, the provider gives customers network-based access to a single copy of an application that the provider created specifically for SaaS distribution. The application's source code is the same for all customers, and when new features or functionalities are released, they are rolled out to all customers. Depending on the service-level agreement (**SLA**), the customer's data for each model may be stored locally, in the cloud or both locally and in the cloud.

Organizations can integrate SaaS applications with other software using application programming interfaces (**APIs**). For example, a business can write its own software tools and use the SaaS provider's APIs to integrate those tools with the SaaS offering.

SaaS architecture

SaaS applications and services typically use a **multi-tenant** approach, which means a single instance of the SaaS application will be running on the host servers, and that single instance will serve each

subscribing customer or cloud tenant. The application will run on a single version and configuration across all customers, or tenants. Though different subscribing customers will run on the same cloud instance with a common infrastructure and platform, the data from different customers will still be segregated.

The typical multi-tenant architecture of SaaS applications means the cloud service provider can manage maintenance, updates, and bug fixes faster, easier and more efficiently. Rather than having to implement changes in multiple instances, engineers can make necessary changes for all customers by maintaining the one, shared instance.

Furthermore, multi-tenancy allows a greater pool of resources to be available to a larger group of people, without compromising important cloud functions such as security, speed and privacy.

SaaS advantages

SaaS removes the need for organizations to install and run applications on their own computers or in their own data centres. This eliminates the expense of hardware acquisition, provisioning and maintenance, as well as software licensing, installation and support. Other benefits of the SaaS model include:

Flexible payments.

Rather than purchasing software to install, or additional hardware to support it, customers subscribe to a SaaS offering. Transitioning costs to a recurring operating expense allows many businesses to exercise better and more predictable budgeting. Users can also terminate SaaS offerings at any time to stop those recurring costs.

Scalable usage.

Cloud services like SaaS offer high Vertical scalability, which gives customers the option to access more or fewer services or features on demand.

Automatic updates.

Rather than purchasing new software, customers can rely on a SaaS provider to automatically perform updates and patch management. This further reduces the burden on in-house IT staff.

Accessibility and persistence.

Since SaaS vendors deliver applications over the internet, users can access them from any internet-enabled device and location.

Customization.

SaaS applications are often customizable and can be integrated with other business applications, especially across applications from a common software provider.

SaaS disadvantages

SaaS also poses some potential risks and challenges, as businesses must rely on outside vendors to provide the software, keep that software up and running, track and report accurate billing and facilitate a secure environment for the business's data.

Issues beyond customer control.

Issues can arise when providers experience service disruptions, impose unwanted changes to service offerings or experience a security breach -- all of which can have a profound effect on the customers' ability to use the SaaS offering. To proactively mitigate these issues, customers should understand their SaaS provider's SLA and make sure it is enforced.

Customers lose control over versioning.

If the provider adopts a new version of an application, it will roll out to all its customers, regardless of whether the customer wants the newer version. This may require the organization to provide extra time and resources for training.

Difficulty switching vendors.

As with using any cloud service provider, switching vendors can be difficult. To switch vendors, customers must migrate very large amounts of data. Furthermore, some vendors use proprietary technologies and data types, which can further complicate customer data transfer between different cloud providers. Vendor lock-in is when a customer cannot easily transition between service providers due to these conditions.

Security.

Cloud security is often cited as a significant challenge for SaaS applications.

SaaS security and privacy

The cybersecurity risks associated with software as a service are different from those associated with traditional software. With traditional software, the software vendor is responsible for eliminating code-based vulnerabilities, while the user is responsible for running the software on a secure infrastructure and network. As a result, security is more the responsibility of the independent software vendor and third-party cloud provider.

Despite the rapid adoption of cloud-based models for fully serviced software products, organizations still have certain reservations about SaaS products when it comes to security and privacy. These concerns include:

- Encryption and key management.
- Identity and access management (**IAM**).
- Security monitoring.
- Incident response.
- Poor integration into broader, company-specific security environments.
- Fulfilment of data residency requirements.
- Data privacy.
- Cost of investing in third-party tools to offset the SaaS security risk.
- Lack of communication with technical and security experts during the sales process.

SaaS vendors and examples

The SaaS market includes a variety of software vendors and products. Industry players include small, single-product vendors all the way up to cloud giants such as AWS and Google.

SaaS products are also diverse, ranging from video streaming services to IT business analytics tools. There are SaaS applications for fundamental business applications such as email, sales management, customer relationship management (**CRM**), financial management, human resource management (**HRM**), billing and collaboration. Enterprise SaaS products for specific industries, such as insurance or medical, are known as vertical SaaS products.

SaaS products may be primarily marketed to B2B, B2C markets or both. Examples of popular SaaS products include:

- Salesforce
- Google Workspace apps

- Microsoft 365
- HubSpot
- Trello
- Netflix
- Zoom
- Zendesk
- DocuSign
- Slack
- Adobe Creative Cloud
- Shopify
- Mailchimp

SaaS pricing

Generally, using a SaaS product is more cost-effective than a traditional software license for enterprise software, as setup and installation onto hardware are not necessary. SaaS providers typically use one of many subscription-based pricing models for customers.

Free, or ad based.

A service may be free for users, with the SaaS provider generating revenue through selling advertisement space. In this model, there is typically an option to upgrade to a paid tier that doesn't include intrusive ads.

Flat rate.

Customers are granted access to the software's full suite of features for a fixed monthly or annual subscription fee.

Per user.

Pricing is determined by how many people will be using the service for each subscription. There is a fixed price for every user.

Per user tiers.

Pricing tiers are based on a range of how many active users can exist on a single subscription.

Storage tiers.

Customers may have free access to a service but will be required to pay for storage if they wish to continue using the product after they pass the free limit.

Pay-as-you-go, or usage-based.

The more customers use the service, the more they are billed and vice versa.

Per active user. This incorporates aspects of the "per-user" and "pay-as-you-go" strategies. Subscribers are billed per user, but only if the user has been actively using the service beyond a defined threshold.

Feature-based tiers.

Price tiers are determined by the amount of features the subscriber seeks. In this model, reduced versions of the software with limited features are available for a lower price than the maximum functionality tier. Additional feature tiers in between the minimum and maximum functionality tiers may also exist.

Freemium.

The service will be generally free to use with an entry-level tier. However, there will typically be functional restrictions in place that are designed to upsell customers to a paid tier.

Conclusion

Hence, we were able to study SaaS (Software As A Service).