Title	Introduction to Software as a Service
Document ID	A000000003 [+++]
Short description	A high-level guide to various aspects of SaaS.
Status	WIP
Created	29-SEPT-2020
Modified	15-MAR-2021
Current Version	1.9
Audience type/s	Internal
Audience level/s	Beginner
Document type	Technical guide
Project ID/s	PD000000003
JIRA ID/s	JD000000369
Author user ID/s	MM-091824
Author name/s	Mark Mehmet
Additional information where applicable	Related documents Revision history Periodic review history Sign-off history

[+++] This table is an example of an optional, automatically-generated document information page for inclusion in internal documents only. The information shown is sourced from a documentation repository, created and maintained as part of a documentation development and review lifecycle. The hyperlinks in this table are for illustration only and do not link to any internal or external resources.

Introduction to Software as a Service

A high-level guide to various aspects of Software as a Service (SaaS)

Date

Topics

Diagrams

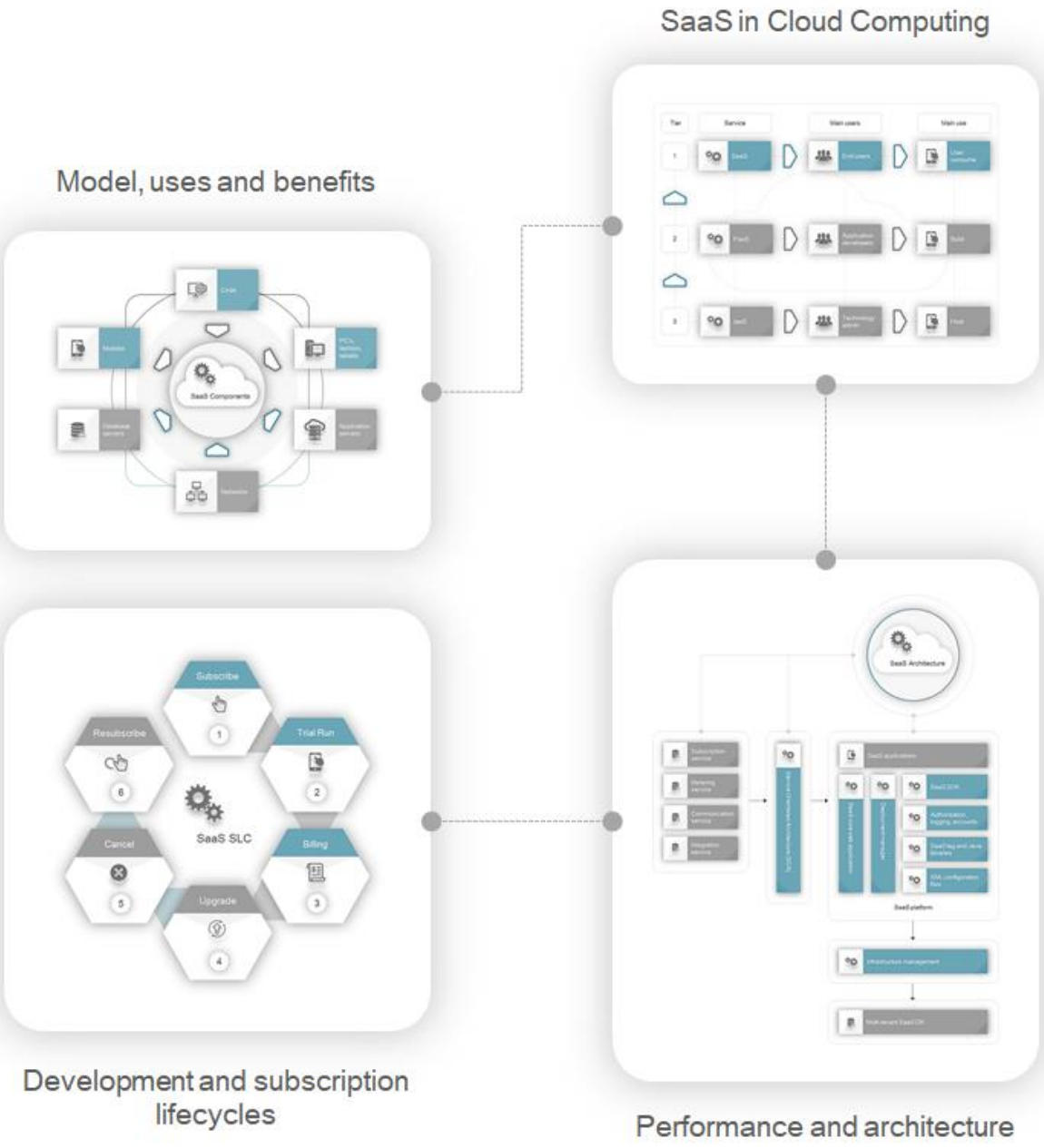
Tables

Synopsis

This article introduces some of the technical and functional aspects of Software as a Service (SaaS), covering the following topics.

- Characteristics, service provisions, uses, and benefits.
- SaaS in the Cloud Computing services framework.
- SaaS performance considerations and architectures
- Development and subscription life-cycles
- Some current SaaS market and growth trends.

SaaS-diagram-1. Introduction to SaaS document overview



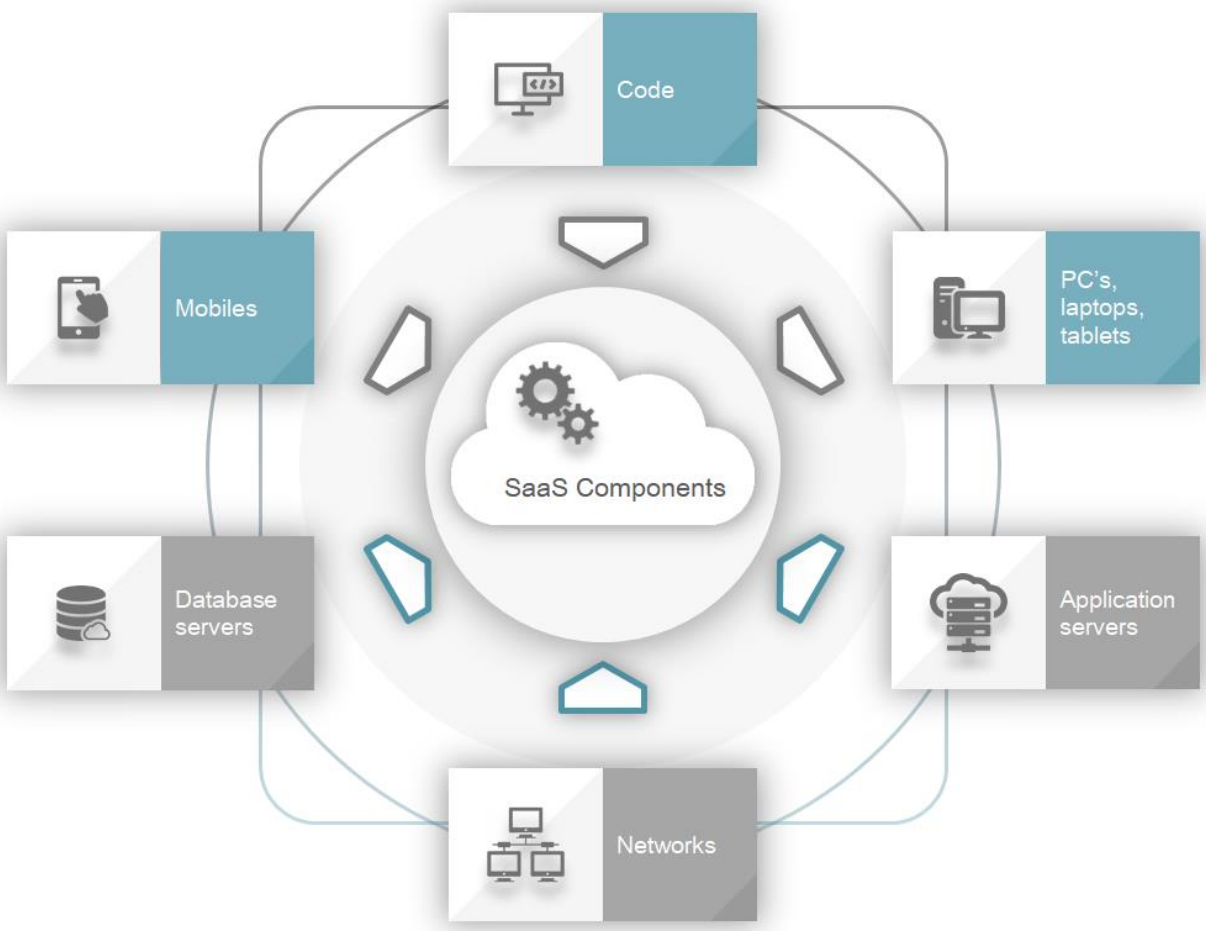
Introduction to Software as a Service

Software as a Service (SaaS) is an internet-based software licencing and distribution model. Software providers centrally host a combination of application servers, databases and code for creating applications and services. Users access these applications and services via internet-enabled devices. Examples of devices include PC's, tablets and mobile phones, typically using a thin client, such as a web browser. SaaS-diagram-2. Components of a simplified SaaS conceptual model illustrates these hardware and software components.

Some of the key characteristics of SaaS are as follows.

- Software updates are applied automatically, without customer intervention.
- Software and services are subscribed to, usually on a monthly or yearly basis.
- Other than the customer's connected devices, no additional hardware is required.
- Customers access software and services from any location, at any time.
- SaaS is also known as *hosted software* or *on-demand software*.

SaaS-diagram-2. Components of a simplified SaaS conceptual model



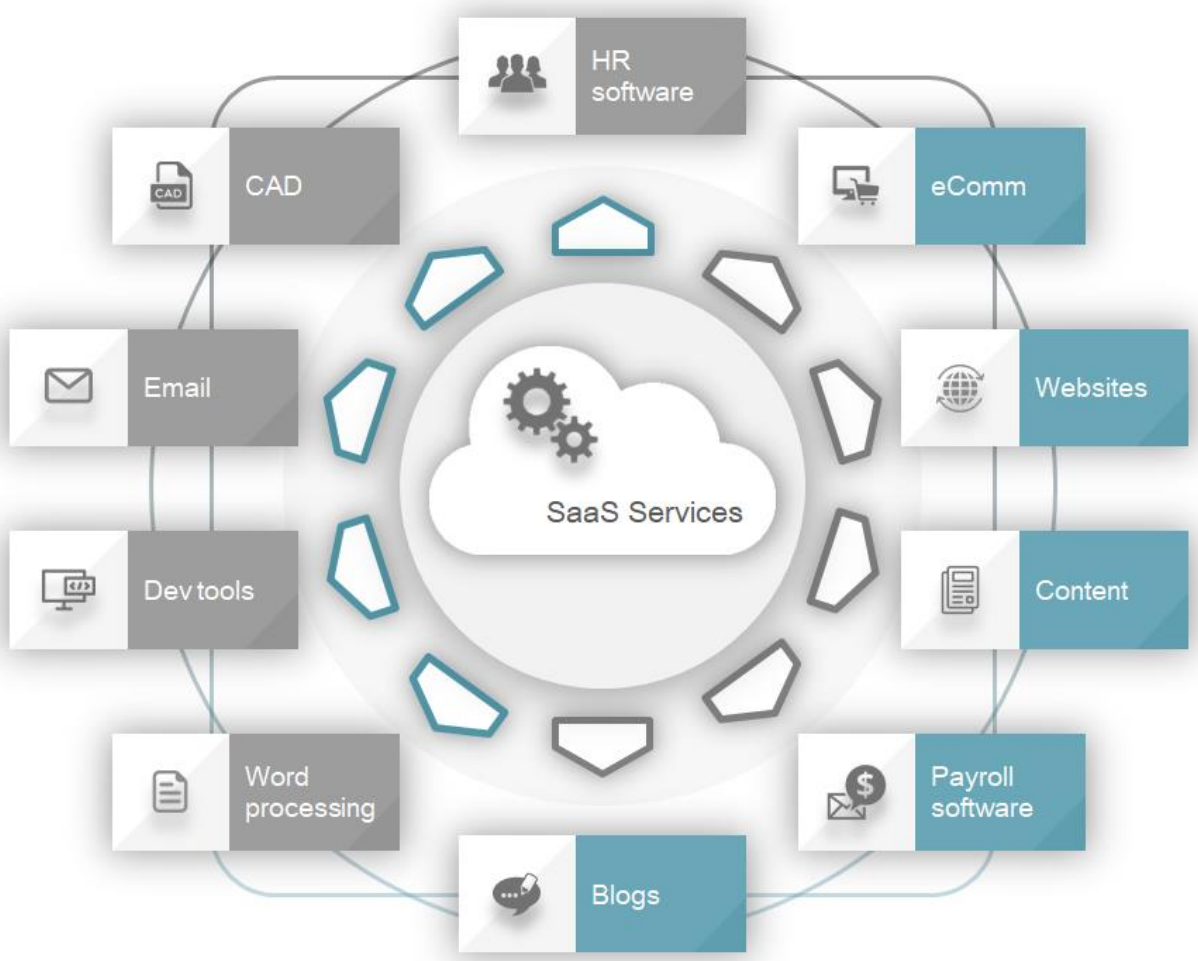
Typical uses of Software as a Service

Many software companies use SaaS for tactical and strategic application delivery. SaaS-diagram-3. A selection of typical SaaS Services illustrates some of the applications and service provisions of SaaS platforms.

Examples of some additional software provisions are listed below.

- Database Management Systems (DBMS)
- Customer Relationship Managements (CRM)
- Management Information Systems (MIS)
- Enterprise Resource Planning (ERP)
- Geographic Information Systems (GIS)

SaaS-diagram-3. A selection of typical SaaS Services



Benefits of using Software as a Service

SaaS has numerous benefits, especially over traditional software use, where organisations develop or purchase software to install and manage on proprietary infrastructure. SaaS-table-1. A summary of the benefits of using SaaS lists some of the cost and growth benefits.

SaaS-table-1. A summary of the benefits of using SaaS

Category	Benefit	Benefit overview
Cost	**No hardware costs**	The SaaS provider is responsible for hardware and software operation.
	No initial set-up costs	Applications are ready to be used once the client subscribes.
	Payment flexibility	Clients can pay only when they require the software.
	Reduced staff costs	SaaS providers have the necessary development and support expertise.
Growth	**Usage scalability**	Additional services or storage facilities utilised on-demand.
	Automatic updates	Updates are deployed automatically by the SaaS provider and are often free.
	Cross compatibility	SaaS products and services accessed from any internet-enabled device.
	Global access	SaaS products and services accessed from any location, at any time
	Customisation	Applications customised to client needs, such as changing the Service Level Agreements (SLA.)

Organisations using Software as a Service

Numerous organisations use SaaS. Some of the larger subscribers are listed below

- Amazon Web Services (AWS)
- Google Apps (G Suite)
- Microsoft Office 365
- Oracle Cloud

Software as a Service in cloud computing

The rise of Cloud Computing has increased the popularity of Software as a Service (SaaS.) SaaS is a subset of the Cloud Computing model or framework. The term Cloud generally describes data centres available to many users over the Internet. SaaS is a tier of the multi-tiered Could Computing Service model.

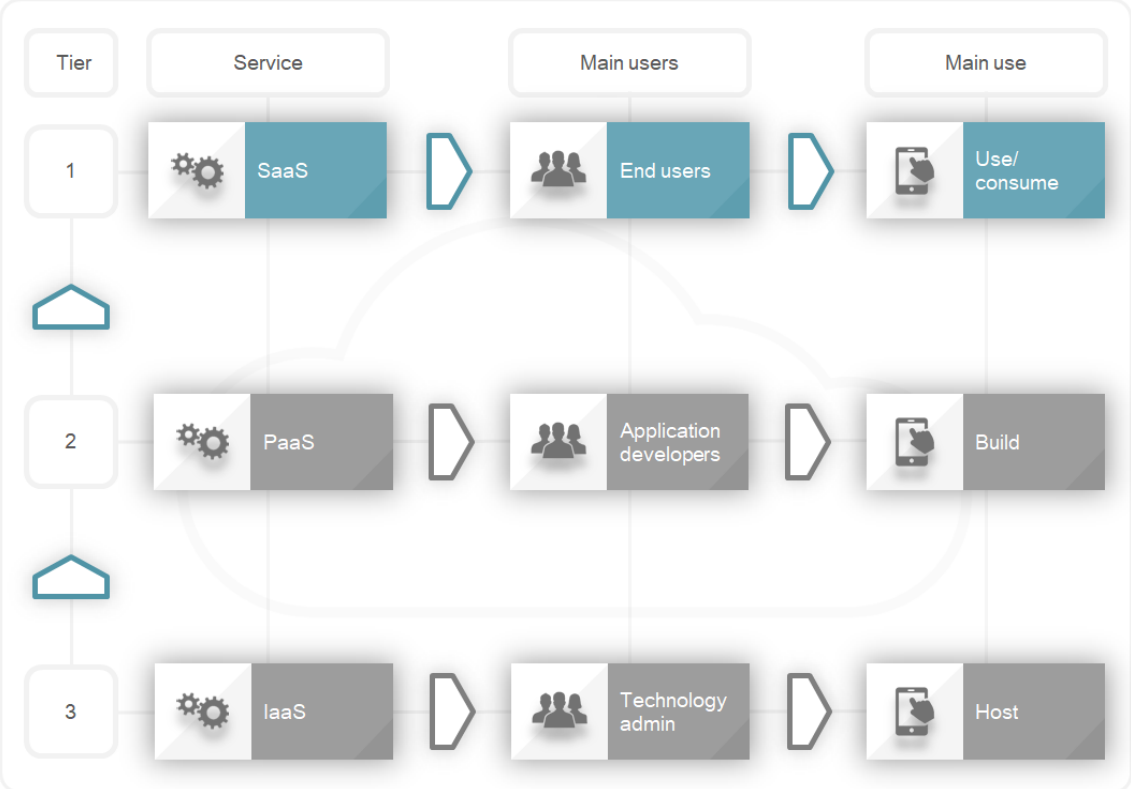
Cloud computing service categories

Together with Infrastructure as a Service (IaaS) and Platform as a Service (PaaS), SaaS forms the three main components or tiers of the Cloud Computing Service Model. SaaS-diagram-4. SaaS in the cloud computing service model illustrates the interoperability and interactions between these tiers. SaaS-table-2. Cloud computing service tiers lists the main functions of these tiers.

Integration protocols in cloud computing

SaaS applications are hosted in the cloud and significantly abstracted from application users. However, light-weight integration protocols, such as REST and SOAP, enable internal client applications to integrate with cloud-based SaaS applications.

SaaS-diagram-4. SaaS in the cloud computing service model



SaaS-table-2. Cloud computing service tiers

Tier	Service	Main users/usage	Main functions	Examples
1	SaaS	**End users** Application subscription and usage	The provider hosts applications and makes them available to end-users.	• Amazon Web Services (AWS) • Cisco WebEx • Google Apps (G Suite) • Microsoft Office 365 • Oracle Cloud
2	PaaS	**Developers** Application development, build, deployment	The provider hosts application development tools and makes them available to developers.	• AWS Elastic Beanstalk • Heroku • Magento Commerce Cloud • OpenShift
3	IaaS	**IT administration** Infrastructure hosting, maintenance, support.	The provider hosts servers, storage, and other computing resources, and makes them available to clients.	• AWS Elastic Compute Cloud • Google Compute Engine • Rackspace

The performance of Software as a Service systems

The number of Software as a Service (SaaS) providers and subscribers continues to grow. Therefore, providers must ensure the continuous availability and acceptable performance of their applications and the underlying hardware. This growth and demand necessitate the standardisation and monitoring of key performance indicators (KPIs.) Examples of KPIs include resource utilisation, volume, throughput, and responsiveness.

Service Level Agreements and Quality of Service

Several options are available for measuring the performance of SaaS systems. SaaS-diagram-5. A simplified SaaS performance sourcing architecture presents an architecture for gathering performance data. This approach highlights the significance of maintaining and monitoring Service Level Agreement (SLA) and Quality of Service (QoS) as key performance indicators in measuring system performance and, more importantly, client satisfaction.

Additional service logistics

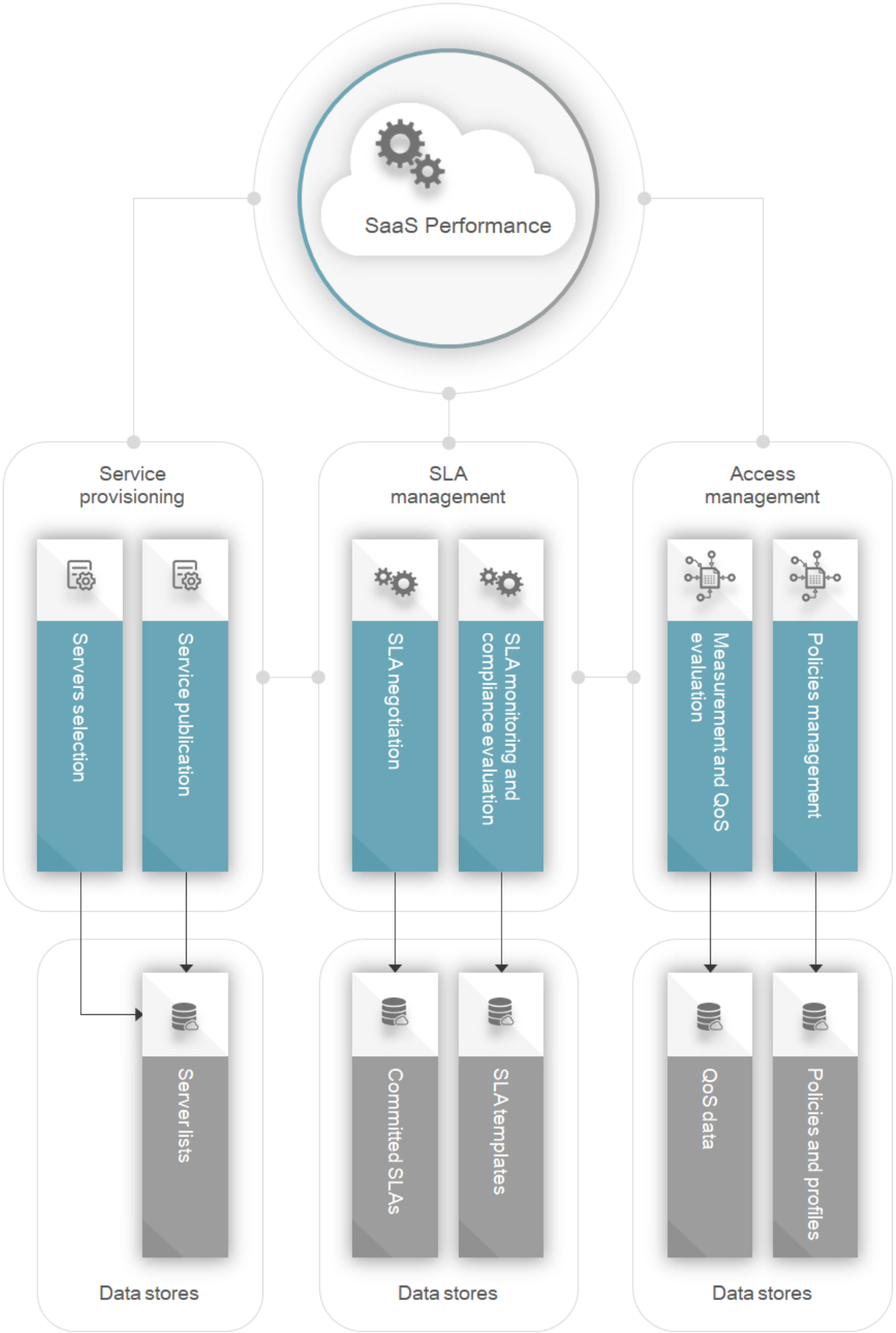
In addition to system performance, SaaS providers must demonstrate performance efficiency in several critical logistical areas. These include the following.

- The response and resolution times to incidents and client support enquiries.
- The responsibilities of the provider in situations where SLA contraventions occur.
- The implementation and maintenance of access security and user profiles.
- The integration and customisation of SaaS offerings with on-site applications where required.

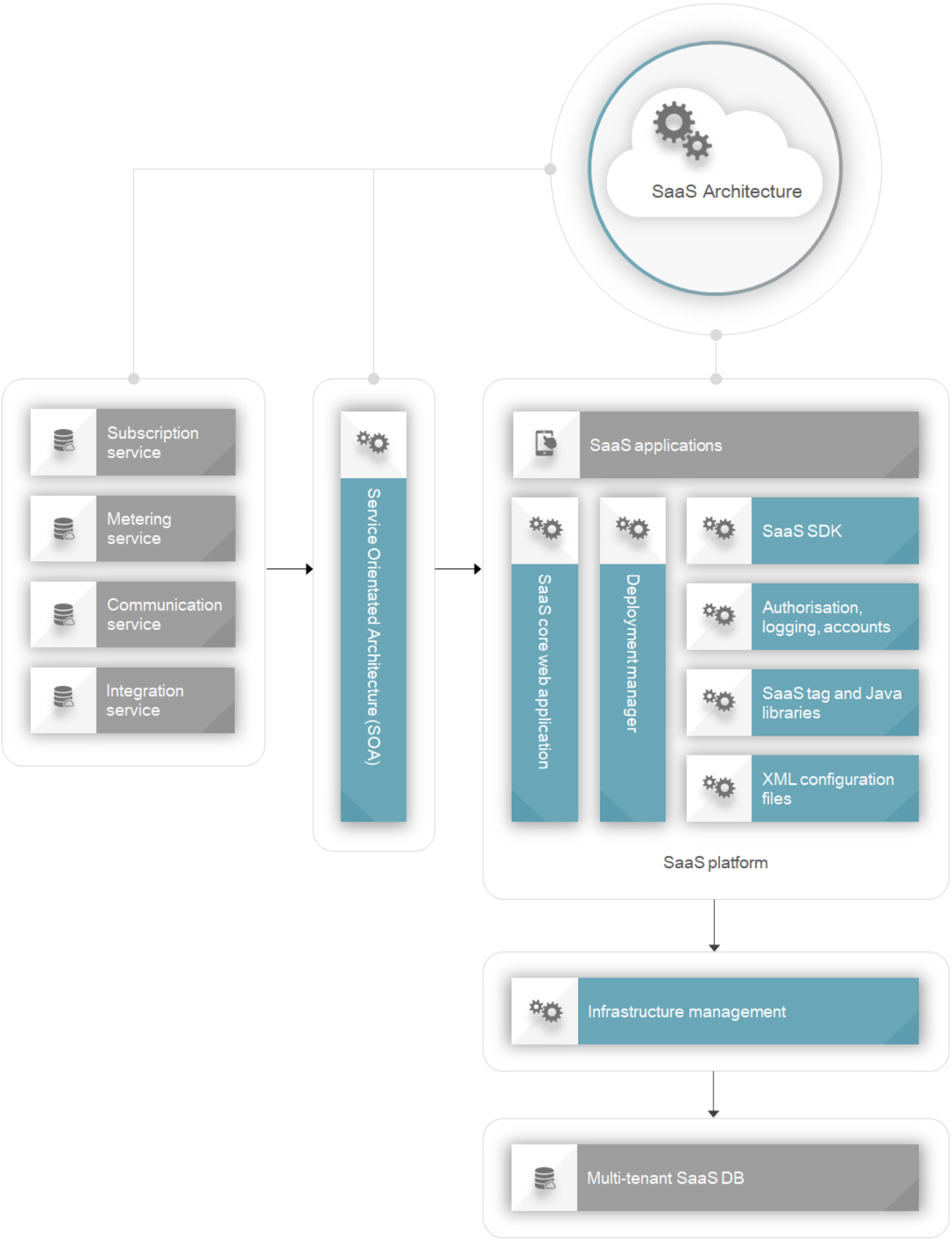
SaaS-table-3. Software as a Service platform technologies

	Component	Technologies
1	Language platform	J2EE
2	Web container	Apache Tomcat
3	Web framework	Apache Struts
4	Web services	Apache Axis
5	Dependency injection	Spring
6	Components 4 and 5 integration	WSO2
7	Multi-tenancy layer	Java SQL and Java Annotations
8	Persistence layer	Hibernate and Java Persistence API (JPA)
9	Database management	MySQL

SaaS-diagram-5. A simplified SaaS performance sourcing architecture



SaaS-diagram-6. A simplified SaaS platform architecture



The Software as a Service Development Life-Cycle

The Software as a Service (SaaS) Development Life-Cycle (DLC) is similar to the traditional Software Development process, with three further stages; Evaluate, Subscribe and Operate.

The SaaS DLC mainly applies to enterprises seeking to implement SaaS as part of their strategic technology vision. The activities performed during these stages are critical success factors for SaaS deployments, as briefly addressed in The Performance of SaaS Systems.

SaaS-diagram-7. The SaaS Development Life-Cycle illustrates the stages. Stages 3 and 5 are iterative. SaaS-table-4. Stages of the SaaS Software Development Life-Cycle provides more information on each stage.

SaaS-diagram-7. The SaaS Development Life-Cycle

SaaS-table-4. Stages of the SaaS Software Development Life-Cycle

	Stage	Stage considerations
1	**Envision**	Analyse SaaS platforms used by similar firms, and review relevant use-cases, to implement the organisation's required strategic technological vision.
2	**Evaluate**	Evaluate potential platforms and solutions, taking into account several factors such as capacity, reliability, scalability and security.
3	**Plan**	Plan for the subscription to the chosen platform, consisting of the traditional waterfall or agile project management methodologies. The subscription plan usually included all divisions expected to be impacted by the platform.
4	**Subscribe**	Subscribe to the chosen SaaS platform and extend production support and operations processes to cover the functionality. This stage includes Target Operating Models, Roles and Responsibilities Matrices, and Service Level Agreements.
5	**Develop**	Develop the platform, including customisation and integration with external systems where required. Development may involve the use of APIs and integration protocols.
6	**Operate**	Operate and utilise the SaaS platform within the Operating and Service Level Agreements established in the Subscribe stage.

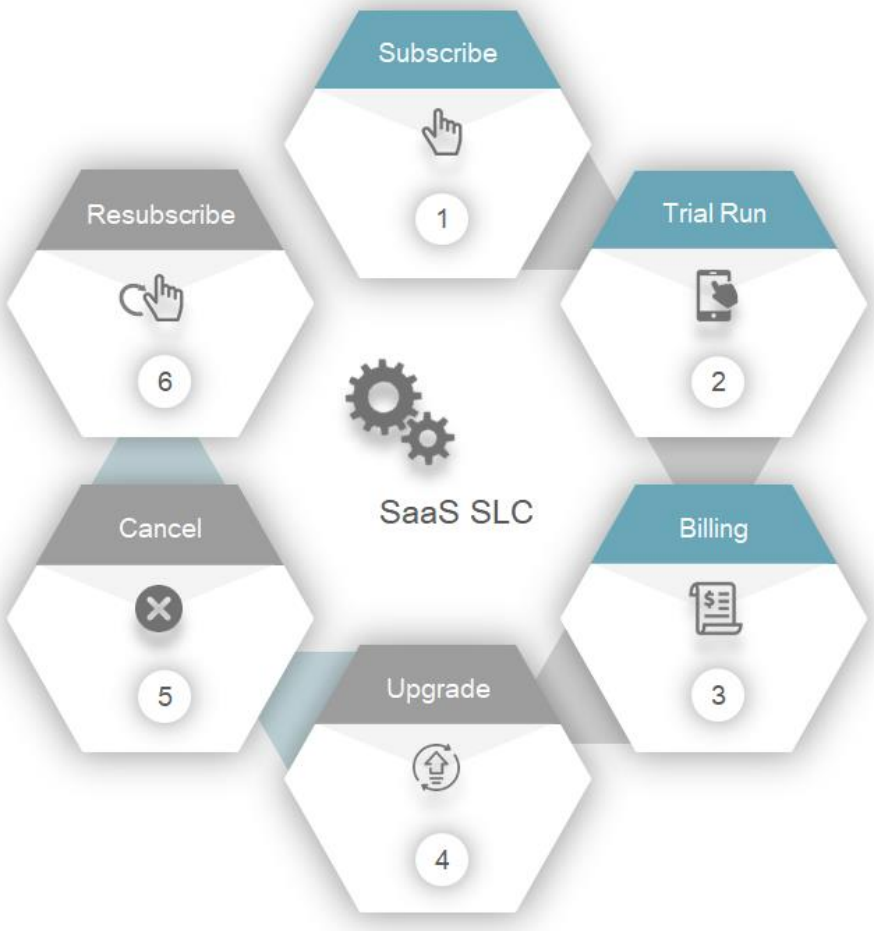
The Software as a Service Subscription Life-Cycle

The Software as a Service (SaaS) Subscription Life-Cycle (SLC) is the automation of several subscription processes such as creating new client accounts, offering free trials, and handling cancellations. SaaS-diagram-8. A simple SaaS Subscription Life-Cycle illustrates the main stages of a simple SaaS SLC. SaaS-table-5. Stages of a simple SaaS Subscription Life-Cycle provides more information on these stages.

Simple vs complex subscription models

The main stages of the SaaS DLC, described in The SaaS Development Life-Cycle, generally apply to enterprise-sized endeavours. In contrast, the SaaS SLC discussed in this section mainly applies to single-user or client subscriptions, subscribing to a single or a small number of applications or services. The subscription process, in this simplified context, can be standardised and fully automated. Enterprise-level subscriptions are inherently more complicated, where subscription agreements are likely to be negotiated and tailored between provider and subscriber.

SaaS-diagram-8. A simple SaaS Subscription Life-Cycle



SaaS-table-5. Stages of a simple SaaS Subscription Life-Cycle

	Stage	Stage Factors
1	**Subscribe**	The client signs-up for a subscription. A process is triggered to create a user account and initiate a free trial period (if offered.)
2	**Trial Run**	If offered by the provider, the client uses the application and services free-of-charge during the trial period.
3	**Billing**	On completion of the trial period, the client is charged (monthly or yearly) according to the terms set out by the provider.
4	**Upgrade**	If a subscription upgrade is available, the client may decide to upgrade. Upgrading usually results in an increased premium.
5	**Cancel**	The client cancels a subscription. A cancellation request is triggered to suspend services at the end of the billing period.
6	**Re-subscribe**	The client re-subscribes, triggering a process to reactivate the client's account and subscription.

Conclusion

Cloud computing continues to transform the information technology landscape. Businesses continue to invest heavily in Software as a Service (SaaS) to gain a competitive advantage by achieving increased profitability and agility, and reduced operational costs. Recent market and growth statistics, a selection outlined in SaaS-table-6. SaaS market and growth statistics demonstrate the growing popularity of SaaS.

With the growing demand for SaaS and the increasing heterogeneity and sophistication of SaaS applications, the performance of SaaS, at both system and logistical levels, becomes increasingly critical to the success of both the provider and subscriber. These factors ensure continued investments in all levels of the Cloud Computing Services framework.

SaaS-table-6. SaaS market and growth statistics

Category	Statistics (quotation)	Source
Market	In 2018, SaaS vendors spent $63.1 billion on R&D, which equals to 20% of all US-based enterprise R&D	OpenView Partners
	To date, the best SaaS IPO year was 2018, with 17 new public vendors collecting a total of $5.1 billion	Battery
	By revenue, Microsoft is the largest worldwide provider of SaaS services, with an 18% market share	Synergy Research Group
Growth	For 2019 to 2023, the global SaaS market is predicted to be worth $60.36 billion	Technavio
	In 2018, the global SaaS workload grew to 206 million and is predicted to reach 380 million by 2021	Statista/Cisco
	The corporate mobile SaaS market is predicted to reach $7.4 billion by 2021	Strategy Analytics