

# Whitepaper on SaaS Application



Sanket Khandare

Lead Technical Architect



## Introduction

Information technology (software) setup in business corporation is consists of many operations, they have to create datacenter with high end data server, then have to setup backup and maintenance activities, have to install applications on every computer. Along with this, they also have to do periodic maintenance and troubleshooting. This is not only long time-consuming process but also leads to high cost.

Large business corporations are becoming flustered with this long deployment and maintenance cycle, and they also are looking for smart solution which will reduce setup and operational cost. So they are moving from data centric applications which require high system and infrastructure management towards integrated business platforms and services.

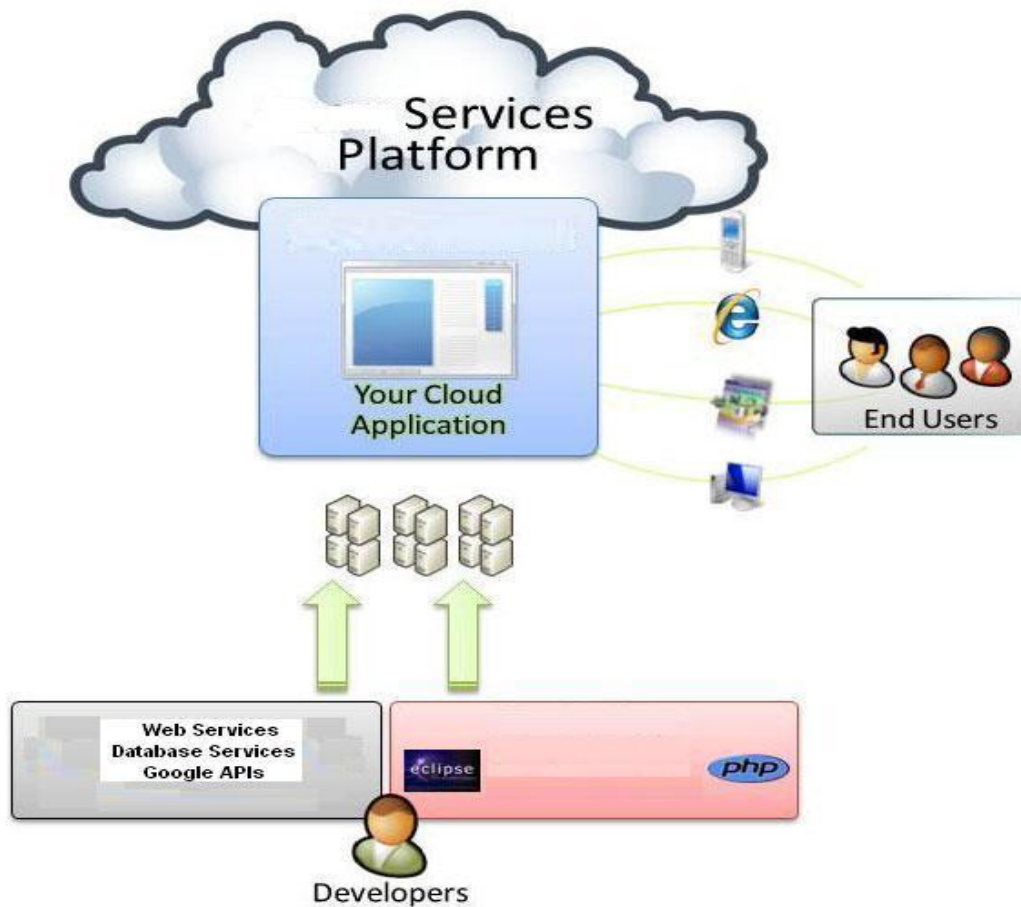
To resolve all such problems, many software development organizations are come up with solution which is popularly named as SaaS (Software as a service). From last decade, SaaS is becoming fastest growing sector of Information Technology. It provides easy low cost solutions to corporations and also gives freedom from long setup and maintenance phase.

In this whitepaper we will have a broad look at SaaS and other related technologies.

## Defining SaaS

You will find many definitions' of SaaS I will define SaaS in very simple statement as ***“SaaS is an application which is hosted on single logical place, and provides service to its customer across the globe with the help of internet”.***

Following fig. shows the logical structure of SaaS



So there will be single instance of application which will be shared by many clients. Activities are managed from central locations rather than at each customer's site, enabling customers to access applications remotely via the Web.

## Advantages of SaaS

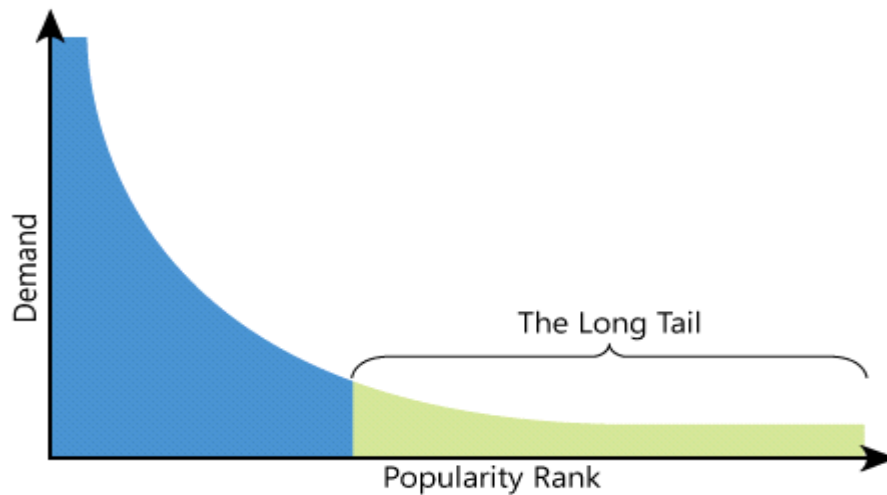
Following are the some advantages of SaaS as a Customer point of view:

1. **Low cost and lower total cost of ownership (TCO):** Generally when customer wants to have a software business solution (IT enabled business solution), there are many factors for which cost have to consider. Like data center cost, most of cost is associated with having proper datacenter with periodic backup facilities. Other costs are setup cost, software licensing cost, application environment cost, training cost, setup cost, and maintenance & troubleshooting cost, support cost etc. When Customer use SaaS as a business solution, then they don't have to place all this infrastructure and ultimately there is reduction in cost.
2. **Zero Infrastructure - Reduced Overheads:** Customer not has to consider setup and maintenance so there is less overhead associated with SaaS model.
3. **On demand service:** Customer can rent the software they why they want to buy it? Customer can think for SaaS products when they will need it. And when there is no need they can think for discontinuing the same.
4. **Much lower risk - faster implementations and outsourced expertise dramatically lower risk**
5. **A more powerful and secure IT infrastructure**
6. **Cost-effective Infinite Scalability:** Customer can Scale the no of user licenses as per there requirements.
7. **Easy to implement:** As there is no infrastructure setup is associated with SaaS, it is easy to implement
8. **Freedom of Choice:** As there are many SaaS provides, customer can select best from that.

9. Centralized feature updating, which obviates the need for downloadable patches and upgrades
10. Per licence cost has to give

Advantages from SaaS Provider's point of view:

1. Now a day everyone is using computer, and as SaaS don't require any special framework installed on machine, provider can target for big market.
2. Single time designed specialized software can be used by several customers, provider can get better revenue
3. Provider can focus on large no of small customers as show in following graph



4. There are good supporting technologies like Grid computing, Cloud computing, Virtualization, Multi-tenancy, SOA etc. so which makes development easy
5. SaaS Product is generally build as Scalable, configurable so can provide services to large no of customers

## Licensing patterns in SaaS

There are varieties of licensing patterns are followed by SaaS providers and Customers, some of them are listed as follows:

1. **Subscription-Based Model:** In this model monthly payment is calculated, customer has to subscribe the given services
2. **Data Volume Based Model:** In this model payment is charged as per volume of data saved
3. **User Based Model:** In this type, payment is charged as per number of users using the system
4. **Fixed Fee Model:** In this model, onetime payment is taken for specific period. User can have any users and any volume of data.
5. **Custom component based Model:** Usually for basic service, fixed cost is associated. Any additional component is selected they user have to pay for that component.
6. **Hybrid Model:** many providers use combinations of one or two models define above.

## Common business areas of SaaS applications

As many of the times SaaS is generalized application which is made for long list of customers, so it is very necessary that any buddy should develop SaaS product which is standard business product, for most popular business activities and have standard processes. Following are some business areas where lots of development organizations have focused for their SaaS product.



### 1. CRM (Customer Relationship Management)

It is a process or methodology used to learn more about customers' needs and behaviors in order to develop stronger relationships with them. There are many technological components to CRM, but thinking about CRM in primarily technological terms is a mistake

### 2. ERP (Enterprise Business Planning)

A business management system that integrates all facets of the business, including planning, manufacturing, sales, and marketing

3. HRM (Human Resource Management)

Human Resource Management (HRM) is the function within an organization that focuses on recruitment of, management of, and providing direction for the people who work in the organization. Human Resource Management can also be performed by line managers.

4. SCM (Supply chain management)

SCM is the oversight of materials, information, and finances as they move in a process from supplier to manufacturer to wholesaler to retailer to consumer

5. CMS (Content Management System)

This is for mainly for managing business documents

6. Finance and Accounting

SaaS services are available to manage day to day finance and accounting operations

7. PLM (Product lifecycle management)

PLM is the process of managing the entire lifecycle of a product from its conception, through design and manufacture, to service and disposal

8. SRM (Supplier relationship management)

SRM is to streamline and make more effective the processes between an enterprise and its suppliers just as customer relationship management (CRM)

## Other related technologies, an Introduction

When we discuss about SaaS (Software as a Service) there are lots of other technologies or terms come in mind. There are many technologies which are directly or indirectly associated with SaaS. We will discuss some of these important technologies.

### ASP – Application Service Provider

Traditionally SaaS is considered as ASP but there is thin-line difference between SaaS and ASP.

1. ASP is One-to-One, means each ASP application is developed for single customer; unlike that SaaS is One-to-Many, means one application can be share by many customers.
2. SaaS is generally low cost solution than ASP, in which entire application cost has to bear by single customer.
3. SaaS model provides easy implementation, customization and Integration.

### Cloud computing

Cloud computing is the term in internet based computing. In cloud computing number of computers are connected via network and they used to deliver services. When applications ask for particular service then task is distributed over various computers formally known as cloud for processing.

Other important thing related to Cloud computing is that it uses for *“On Demand Services”* that’s why it is linked to SaaS.

## Grid computing

Grid computing is a form of distributed computing whereby a task is divided between multiple computers which are connected over network. Tasks can range from data storage to complex calculations and can be spread over large geographical distances. These computers join together to create a virtual supercomputer. Computers networked together can work on the same problems.

Cloud Computing is border concept of Grid Computing, Grid Computing can be part of Cloud Computing.

## Thin Client

This is also called as server based computing, the entire application is run at the server, most of the application processing is at server side which includes business logic as well as database handling. Clients are called as thin client as they generally don't do any business logic processing; they just have a GUI (user interface) to show data and reports.

Categories of thin clients include Windows-Based Terminals (WBT, which comprise the largest segment), X-Terminals, and Network Computers (NC)

## Virtualization

According to Wikipedia, "Virtualization is a broad term that refers to the abstraction of computer resources. One useful definition is 'a technique for hiding the physical characteristics of computing resources from the way in which other systems, applications, or end users interact with those resources. This includes making a single physical resource (such as a server, an operating system, an application, or storage device) appear to function as multiple logical resources; or it can include making multiple physical resources (such as storage devices or servers) appear as a single logical resource.'"

Actually Virtualization is the creation of a virtual (rather than actual) version of something, such as an operating system, a server, a storage device or network resources.

Network virtualization is a method of combining the available resources in a network by splitting up the available bandwidth into channels.

Storage virtualization is the pooling of physical storage from multiple network storage devices it appears as a single storage device to end user/application.

Server virtualization is the masking of server resources (including the number and identity of individual physical servers, processors, and operating systems) from server users. So user does not have to worry about complex structure of servers.

One of the principal benefits of virtualization is that it can increase the system's capacity without additional programming.

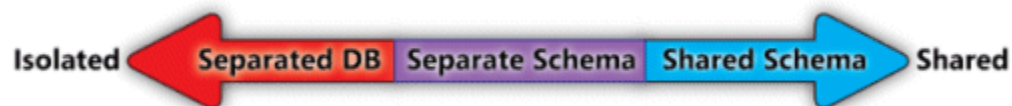
## Multitenancy

As define in Wikipedia “Multitenancy refers to the architectural principle, where a single instance of the software runs on a software-as-a-service (SaaS) vendor's servers, serving multiple client organizations (tenants)”.

Following are some factors which we have to consider while considering multitenancy as a solution

1. Cost: Generally multitenancy consider as cost effective solution, it helps to reduce operational cost. But when we consider the cost of redesigning already seated multitenant data architecture, then that cost is high.
2. Complexity: multitenant data architecture is generally complex architecture. Because of the additional customization complexity and the need to maintain per-tenant metadata, multitenant applications require a larger development effort than applications designed using a more isolated approach

Now we will look at Approaches to Managing Multi-Tenant Data. There are three basic approaches based on distinction between shared data and isolated data.



1. Separated DB: Computing resources and application code are generally shared between all the tenants on a server, but each tenant has its own set of data that remains logically isolated from data that belongs to all other tenants
2. Shared Database separate Schema: Tenants are in the same database, with each tenant having its own set of tables that are grouped into a schema created specifically for the tenant
3. Shared Database shared Schema: A third approach involves using the same database and the same set of tables to host multiple tenants' data. A given table can include records from

multiple tenants stored in any order and there is one identifier is maintained which distinguishes records for each tenant.