Software as a Service (SaaS)

Software as a Service (SaaS) is a model of software that can be rented or leased from a software vendor that provides maintenance, daily technical operation, and support for the software. SaaS is a model of software delivery rather than a market segment; it assumes the software is delivered over a secure Internet connection. Software can be accessed from a browser by any market segment, including home consumers and small, medium, and large businesses. The SaaS model brings lower risk in the implementation cycle and better knowledge transfer from integrators to users of systems. When the implementation partner leaves, the implementing hosting vendor is still there managing the solution. So the knowledge transfer happens automatically, for no additional cost, no impact on schedule, and, of course, lowering risk. With conventional implementations the opportunity exists for disconnects that could hamper the knowledge-transfer process to the customer's support staff."[13]

BENEFITS OF THE SAAS MODEL The traditional rationale for outsourcing of IT systems is that by applying economies of scale to the operation of applications, a service provider can offer better, cheaper, more reliable applications than companies can themselves. The use of SaaS-based applications has grown dramatically, as reported by many of the analyst firms that cover the sector. But, it's only in recent years that SaaS has truly flourished. The advent of PC's and high-speed Internet has provided an opportunity to the way we work and made this rapid acceptance possible. Some benefits of the SaaS model are as follows:

- ***Universal access.*** Most information workers have access to a computer and are familiar with conventions from mouse usage to Web interfaces. As a result, the learning curve for new Web applications is lower, requiring less hand-holding by internal IT staff.
- ***Ubiquitous computing.*** In the past, corporate mainframes were jealously guarded as strategic advantages. The applications were later viewed as strategic. Today, people know it's the business processes and the data themselves (i.e., customer records, workflows, and pricing information) that matter. Computing and application licenses are cost centers. As such, they're suitable for cost reduction and outsourcing. The adoption of SaaS could also drive applications to become a commodity.[14]
- ***Standardized applications.*** With some notable, industry-specific exceptions, most people spend most of their time using standardized applications. An expense reporting page, an applicant screening tool, a spreadsheet, or an e-mail system are all sufficiently ubiquitous and well understood that most users can switch from one system to another easily. This is evident from the number of Web-based calendar, spreadsheet, and e-mail systems that have emerged in recent years.
- ***Parameterized applications.*** In older applications, the only way to change a workflow was to modify the code. In more recent applications, however, and particularly Web-based applications, significantly new applications can be created from parameters and macros. These allow organizations to create many different kinds of business logic atop a common application platform. Many SaaS providers allow a wide range of customization within a basic set of functions.

---

[13] Traudt, E., and Konary, A. (June 2005). *2005 Software as a Service Taxonomy and Research Guide* 7. IDC.
[14] www.saasblogs.com/2006/09/26/scale-as-a-commodity-2/ SaaSBlogs: Scale as a Commodity (accessed February 2011).

- ***Global market.*** A company that made software for human resource management at boutique hotels might once have had a hard time finding enough of a market to sell its applications. A hosted application, however, can instantly reach the entire market, making specialization within a vertical both possible and preferable. This in turn means that SaaS providers can often deliver products that meet their markets' needs more closely than traditional "shrink-wrap" vendors.
- ***Reliability of Web.*** Despite sporadic outages and slowdowns, most people are willing to use the public Internet, the hypertext transfer protocol, and the TCP/IP stack to deliver business functions to end users.
- ***Transparent security and trust.*** With the broad adoption of SSL and HTTPS protocols, organizations have a way of reaching their applications without the complexity and burden of end-user configurations or VPNs.

LIMITATIONS WITH THE SAAS MODEL SaaS is conceptually similar to the original mainframe computing model that had a centralized control, minimal user privacy, and limited flexibility allowed to the individual user. Much of the explosive success of the PC after its introduction in the late 1970s and early 1980s was due to the power it gave to individual users. This empowerment will erode once users feel that with SaaS they lose their privacy and control. Another mitigating factor is the need for disconnected use. Many users (e.g., traveling salespeople) with expensive wireless connections need access to data in offline mode. Although some vendors provide offline modes that synchronize data, solutions are not optimal and not all vendors provide such functionality.

Although there is no large investment for software license at the onset of the project, the ongoing costs of SaaS are categorized as monthly expenses and do not depreciate over time as would a capital investment of perpetual software licenses. Such vendors can easily mislead customers into thinking that with SaaS there is no cost to configure the software or customize integrations because it's all delivered "out of the box." Smart ERP teams will see through this myth and realize that in order for any ERP solution to be successful, there needs to be significant investment in resources (and possibly third-party technology) to configure and support the solution, perform change management, and facilitate business process redesign so that ERP efficiencies can be realized. This cannot take place without thorough understanding of the requirements of the business, the SaaS configuration capabilities, and the difference between the two. It is quite possible that over a three-or five-year period, traditional ERP architecture might even be cheaper than a SaaS solution.

TYPES OF SAAS PROVIDERS There are two types of SaaS providers. The first has often been referred to as an application service provider (ASP) where a customer purchases and brings to a hosting company a copy of software, or the hosting company offers widely available software for use by customers (e.g., hosting Microsoft Office and making that available across the Web to customers who pay a fee per month for access to the software). **The second type of SaaS provider offers what is often called software on demand (SOD),** where a company offers to customers' software specifically built for one-to-many hosting. This means that one copy of the software is installed for use by many companies who access the software from the Internet.

In the first type of provider, a licensing fee and a monthly fee are separate and are paid to the maker of the software and to the software host like an ISP. With the second type of hosting there is no division between licensing and hosting fees, and there is traditionally little or no customization of software for customers. With mature SaaS providers (e.g., Salesforce.com) on-demand solutions can be highly customized.