



A SaaS Primer

Plex Online
White Paper

At a Glance

- Software as a Service (SaaS) solutions can be a strategic advantage to businesses, letting companies avoid costly hardware, software licenses and complex version upgrades.
- Most major software vendors are touting some version of SaaS, leading to confusion about the solution model and its advantages.
- This paper defines the term, and describes the short-term and long-term benefits of the Software as a Solution model and its variants.



Cloud ERP for Manufacturers

www.plex.com | 855-534-8012

SaaS Defined

Software as a Service, or SaaS, is an application delivery model in which the user accesses software over the Internet, from anywhere, at any time. This is why the solution model is also called “on-demand” by some providers.

The physical location and ownership/maintenance responsibility of the system that actually serves the software is outside the responsibility and concern of the end users.

SaaS is available to manufacturers via subscription-based pricing, rather than the “perpetual license” that usually accompanies client/server software.

In true SaaS applications, all users run off a robust infrastructure and a single instance of the software.

There are obvious financial and resource usage advantages to the SaaS model, as compared to the traditional on-premises software model, in which a manufacturer purchases a costly license.

Total Costs: In terms of initial cost, with the subscription pricing model, the up-front investment in software licenses, operating licenses, databases, servers, backup equipment, etc., are eliminated. The service provider bears all of these costs, which are passed along in the subscription pricing.

Operations: Because the service provider is operating the system, performance of low-value but mission-critical IT operations are eliminated or reduced. These low-value operations include nightly backups, tuning databases, defragging storage systems, applying security patches, and more.

These advantages only begin to scratch the surface of the SaaS model. To understand the long-term strategic advantages of the SaaS model, we must first understand a few key points about how SaaS services are developed and delivered.



These advantages only begin to scratch the surface of the SaaS model. To understand the long-term strategic advantages of the SaaS model, we must first understand a few key points about how SaaS services are developed and delivered.

Multi-Tenancy and Agile Development

True SaaS applications must be delivered from a “multi-tenant” system, which means that there is a single instance of the software running, and multiple companies use this system as if it were dedicated to their own use.

Non-technical evaluators sometimes worry that their data may somehow “be mixed up with another company’s data”; however, modern database structures and security technology easily prevent this from happening.

More important, only multi-tenant systems can be combined with new software development models to deliver the long-term advantages of the SaaS model.

The most advanced SaaS providers do away with the traditional “waterfall” development schedule (version 1.0, 1.1, 2.0, etc.) in favor of an agile development, or “rapid application development,” methodology.

Using these new software development techniques has two important distinctions for the users: they are never faced with an upgrade (i.e., they are always using the most current version of the software); and they get direct access to the development team to ask for new features and customizations whenever they are needed.

“In true SaaS, all users run off a robust infrastructure and a single instance of the software.”

The Benefit of Rapid Application Development

Even more important, these new features and customizations get automatically folded into the base application, so they are no longer “customizations.” This combines the operational/financial advantages of SaaS with an entirely new set of advantages, most of them focused on the end users. Advantages include:

	CLIENT/SERVER	SaaS
PRICING MODEL	Perpetual Software License (Purchase)	Monthly subscription fee
HARDWARE	Customer Purchase	Included
OS LICENSES	Customer Purchase	Included
SOFTWARE LICENSES	Customer Purchase	Included
USER ACCESS	Client installed on every PC	Web browser
MAINTENANCE FEES	18–21% of software cost	Included
IT OPERATIONS	Customer must perform	Included
UPGRADE COSTS	One-time purchase	Included

- All users are on the same software, and it's always the most current version.
- Since custom feature requests are folded into the (one) base application, they are never obsoleted with the next upgrade, and instead they are carried forward as standard features.
- Typically these new features are surrounded by configuration settings that turn off the new feature by default; it's only when user asks for this new feature that it is enabled for a new company.
- Because users are talking directly to the developers, it's easy to ask for new features or tweaks to interfaces.

The end result is that the underlying feature-set of a multi-tenant SaaS model gets better every day, and each new deployment of the software dramatically increases the features/functions of the software. This is in direct contrast to most enterprise software models, where companies get frozen into an obsolete release because of the heavy customization required to implement it in the first place.

A Closer Look

There are variations in how a SaaS application can be accessed.

In one model, the developer simply “tweaks” the client/server software, allowing the client software to be installed locally and accessed through the server over an Internet connection.

In many of these applications, a Web interface is added that allows a portion of the functionality to be accessed via a Web browser.

These applications are typically bandwidth-hungry, as they were originally designed for LANs instead of WANs.

A second model is for developers to enhance the functionality of a Web browser with plug-in or add-in software, providing an enhanced feature-set over what is available in a standard browser.

While these applications use dramatically less bandwidth than the first case described above, there are security issues with downloads as plug-ins, as well as portability issues (e.g., accessing from a home PC).

In the “zero footprint” model, a SaaS application requires only a browser, with no plug-ins or downloads required. As browser functionality is improved, the potential functionality of these applications continues to increase, including features such as drag-and-drop.

The Dangers of SaaS “Pretenders”

As stated above, virtually every software company in the market today is offering some type of service and calling it SaaS.

However, a closer look at these offerings demonstrates that they do not offer the same advantages of a multi-tenant SaaS solution, and in many cases have some of the same disadvantages of the legacy on-premises model.



The most common type of service that is mistakenly sold as SaaS is actually a hosted application. Application hosting is a widely available service that is useful in many circumstances. In a hosted application, a data center company provides a dedicated server and related hardware, network connectivity, rackspace, power and related services, and sometimes offers management services from the hardware to the OS and up through the application.

In contrast to the multi-tenant SaaS solution model, this is a dedicated set of hardware that is being hosted and managed by a third party. The same IT operations are required as with an on-premises model (backups, applying patches, etc.), but these are now performed by a third party in an outsourced fashion rather than by an internal IT department.

Application hosting may be advantageous for a company with limited IT resources, but hosting does not offer any of the long-term advantages of a multi-tenant SaaS solution. Because three parties are involved (customer, software vendor, hosting

company) the costs are often higher than running a system internally— though often with guaranteed uptime and high quality IT services.

Moreover, because each hosted application is an island unto itself, the model suffers the same challenges surrounding upgrades and customizations as the on-premises model.

“SaaS does away with complex version upgrades in favor of rapid application development methodology”

A True SaaS ERP Solution

Based on the definitions outlined above, there is a Web-based Software as a Service application that goes beyond the traditional categories of enterprise resource planning (ERP), manufacturing execution systems (MES), and supply chain management (SCM) to offer manufacturers everything they need to operate their businesses efficiently and profitably.

Plex Online is a true multi-tenant application that provides the flexibility to manage a single department, an entire plant, or a global enterprise.

Manufacturers benefit from a comprehensive software solution via the Software as a Service delivery model, speeding technology deployments, lowering costs, and delivering long-term value.

About Plex Online

Plex Online, built on a Software as a Service (SaaS) Cloud model, offers more than 400 functions, providing manufacturers instant access to vital information and management features using a simple Web browser. The on-demand solution includes product lifecycle management (PLM) functions such as program and change management; enterprise resource planning (ERP) functions such as accounting and finance programs; customer relationship management (CRM) functions such as order entry and tracking; manufacturing execution systems (MES) functions such as production scheduling and machine integration; and supply chain management (SCM) functions such as supplier quality and traceability. For more information, see www.plex.com.



Cloud ERP for Manufacturers

www.plex.com | 855-534-8012