



# The Value of Software-as-a-Service

April, 2006

## INTRODUCTION

The manner in which organizations purchase, implement and maintain software solutions has changed dramatically over the years. You may have heard of the most recent approach, known as Software-as-a-Service (SaaS). To understand the entire value of the SaaS model, let's first review the previous ways in which you have acquired and implemented software.

## SOFTWARE DELIVERY MODELS

In the classic software delivery model, you either developed your own applications in-house or purchased software under a perpetual use license agreement from a vendor. With a license agreement, you also typically paid an ongoing maintenance fee to cover bug fixes and minor enhancements. Under this scenario, it was your responsibility to host, implement, maintain and support the software. Most likely, you needed to engage the services of an implementation partner that was knowledgeable in enterprise deployments of the software. Your relationship with your implementation partner would likely continue when it came time to upgrade the software to the next release. The classic software delivery model required you to invest not only in the software, but also in hardware, training, consulting services, support and long-term maintenance.

With the widespread adoption of the Internet, software vendors began to look at a new approach to delivering their software solutions. You could license software and run it in what has become known as the Application Service Provider (ASP) model. The ASP model eliminated the need for you to host, implement, maintain and support the software. The applications were hosted at a remote data center and delivered over the Internet, so you didn't need to make hardware investments. You essentially outsourced the implementation, maintenance and support of the software. Your ASP charged for these services in addition to billing a monthly hosting fee. Of course, you also still paid upfront for the software itself.

Some software providers have recognized that there is still a better way to deliver software from a data center, and they are now delivering their applications in the latest model, Software-as-a-Service (SaaS). In the SaaS approach, software vendors design their solution in a whole new way – the applications are only meant to be delivered via the Internet using a common web browser. Instead of outsourcing your individual applications, you access a shared application that is configured with your data and preferences and pay only a recurring fee. This new approach meets your individual requirements while lowering the cost of maintenance and reducing the time required to upgrade to new versions, service packs and patches.

The remainder of this paper will further examine how SaaS works, why it is an attractive option for many enterprise applications and how it might be of value in your organization, particularly for talent acquisition activities such as recruiting and vendor/contingent labor management. These functions both have a number of stakeholders outside of your organization, such as candidates, staffing agencies, background check vendors, assessment vendors, job boards and contingent workers. Providing a consistent, seamless and high level of service to these groups is a critical part of managing the workforce acquisition lifecycle.

## HOW DOES SAAS WORK?

First, remember how the ASP model works. In this model, each customer has their own version of the application running in the ASP's hosting center (also known as single-tenancy). Most likely during implementation, the application is heavily customized to replicate the business

processes and reports needed by the users. This takes a great deal of time and requires significant involvement from users to ensure the software matches the way they work. These customizations to the software are an additional cost on top of the maintenance agreements. Organizations soon find that the total-cost-of-ownership (TCO) with this model is very high. Software upgrades to new versions, service packs and patches are disruptive to the business, require a great deal of effort due to the customization, and are very costly. This often causes companies to remain on an old version of critical software long after improved capabilities and greater functionality have been released. Ultimately, the ASP client is unable to respond quickly to changes in the marketplace or their business.

The SaaS model is delivered via the Internet using multi-tenant architecture, which means there is a single instance of the software serving all clients. Consequently, SaaS customers share common parts of the application, while receiving complete security for their data and retaining exclusive access to their business process configurations.

The ability to configure the application is an important distinction between SaaS and ASP delivery. SaaS solutions are designed to be highly configurable. Each client has the capability to configure the system to meet their individual needs without time-consuming and expensive custom coding. In addition, multi-tenancy allows all clients to continually utilize the latest versions of the software, seamlessly upgrade at their own convenience and eliminate the need for the ongoing costs of support and maintenance of custom software.

SaaS also delivers the concept of self-service, where clients can accomplish many of their own configurations through an administrative function. Finally, SaaS has a predictable cost, with a modular approach to fees. Fees are paid quarterly or annually for a fixed period of time – anywhere from one to five years. Clients pay for what they receive and may opt to choose higher levels beyond the basic services. Clients always know what to expect with a regular subscription payment that includes maintenance, upgrades and support.

## WHAT ARE THE BENEFITS OF SAAS?

The benefits of the SaaS application delivery model are as follows:

### Low Financial Risk

An enterprise application project using an ASP can be a costly proposition. Although the ASP has the technology infrastructure to run the application, you must devote significant resources to the implementation project. Implementation fees can easily more than double the base price for the software license. Further modifications and upgrades will incur an additional cost.

Once the implementation is complete, there are additional costs for support and maintenance – often an annual fee. Upgrading the product to a new version is costly and disruptive depending on the degree of customization. You should also count on a considerable training investment for end-users.

A SaaS implementation is different. There is no up-front IT infrastructure or technology development investment. Deployments are done in a few weeks rather than months, greatly decreasing the cost of ownership. Many future changes can be done easily by the SaaS provider or by a certified user with an administrative tool included in the application.

A SaaS product is also less financially risky because the recurring costs are predictable. There are no hidden costs for upgrades, maintenance or support – these are all included in the subscription. This results in an overall lower total cost of ownership, especially as high first-year costs associated with a licensed product are avoided.

## Easy Deployment

In addition to carrying a higher total cost of ownership, implementing an enterprise-wide software application on an ASP can be time-consuming and resource-intensive. Today's competitive organizations simply can't afford to spare internal resources for a huge project. Applications that are run in an ASP environment are designed using a single-tenant architecture – one application instance per customer. This is essentially the same as if it were being run in-house, except it is being hosted by a third-party. An implementation consists of significant customization that generally requires developers to rewrite core code to support the customer's needs.

In contrast, SaaS applications are fundamentally different. They are designed and built on a multi-tenant architecture with the expectation that they will be solely delivered via the Internet. A SaaS implementation consists of configuring the application to your business needs. Configuration involves changing settings, choosing options and adjusting process flows without altering the core code. A highly configurable product can offer you a great deal of flexibility without the time and expense of software redesign and coding. Configuration allows clients to leverage best practices in business and recruiting, create the best candidate experience, leverage the company brand, and create local cultural experiences for multi-national organizations.

Further, SaaS products are typically designed with an emphasis on self-service for future configurations. As your business requirements change trained system administrators can make the majority of changes to the system, eliminating the need for future implementation consulting.

## Comprehensive Support

Unlike traditional ASP, the SaaS model includes comprehensive support in the base monthly fee. This also includes product maintenance and upgrades, so there are no hidden costs or additional fees. The SaaS vendor has complete ownership of the technology environment, so you do not incur any capital expenses for the service.

Many SaaS providers also provide different levels of support and services, each with a different impact on the monthly fee depending on the package. This way, you can choose the best level of service for your needs, without paying for more or receiving less than you need. The vendor manages the application release process, which is simplified by retaining all configurations from release to release.

## Relief for IT Staff

If your organization is like most, your Information Technology (IT) staff is already overburdened. SaaS does not place any additional burden on internal technology teams – because there is no internal technology to support. All you do is access your SaaS application using your browser. All the tasks that IT usually does to support an internal application are performed by the SaaS provider, or by a certified system administrator. Your SaaS provider ensures the application is constantly up and running, trains your users and provides user support.

## High Security

You need assurance that your system is secure from both physical and virtual standpoints. SaaS vendors house their applications in world-class data centers, with state-of-the-art power, ventilation and security features.

Careful encryption and the latest in security protocols ensure the safety and integrity of your data. The system architecture must be designed with maximum security, reliability, scalability and performance in mind. You should expect a SaaS application to include

- Hardware load balancing and storage array network (SAN) support for high performance
- Internal firewalls and vulnerability testing to ensure data security
- Multiple redundant servers and network monitoring tools to ensure reliability
- Data backups and disaster recovery plans to protect vital information

A high level of security is particularly important when dealing with personal data, such as in a recruiting/talent acquisition function. And scalability can be critical when it's difficult to predict how many users will be accessing the system. Companies that receive recognition as one of the best places to work can see their career site traffic spike by over 2000 percent. Favorable financial reports, acquisitions, job fairs and advertising campaigns can all significantly increase the number of candidates that access a career site. The ability to scale the application both horizontally and vertically is critical to ensuring that all candidates have a good experience.

## How Can SaaS Power Your Talent Acquisition Initiatives?

The human resources sector is proving to be fertile ground for SaaS vendors, as enterprises have become accustomed over the last 10-20 years to outsourcing various HR functions, such as payroll. A natural extension is to purchase HR software through a subscription model.

Talent acquisition activities, such as those involved in attracting, selecting and retaining permanent and contingent workers, are particularly well-suited to the SaaS environment due to the confidential nature of recruiting and the need to store large volumes of information.

Peopleclick successfully designed its recruitment management and vendor/contingent labor management software from the ground-up for SaaS delivery. Using Peopleclick RMS™ and Peopleclick VMS™, clients can accomplish the full range of activities needed to efficiently engage permanent and contingent workers with the confidence that their data is secure.

Our commitment to and investment in SaaS ensures our clients receive the maximum benefits with the lowest possible total cost of ownership and faster return on their technology investment. We are confident that SaaS is the best possible way to deliver our award-winning RMS and VMS solutions, now and well into the future.

## Comparison of Software Delivery Models

	Traditional - Licensed Software Delivery Model	ASP - Application Service Provider Model	SaaS - Software-as-a-Service Delivery Model
<b>Purchase</b>	<ul style="list-style-type: none"> <li>▪ Purchase software license or develop in-house</li> <li>▪ Purchase hardware</li> </ul>	<ul style="list-style-type: none"> <li>▪ Purchase software license</li> <li>▪ No hardware investment</li> </ul>	<ul style="list-style-type: none"> <li>▪ No software purchase</li> <li>▪ No hardware investment</li> </ul>
<b>Implementation and hosting</b>	<ul style="list-style-type: none"> <li>▪ Unique, in-house implementation by customer's IT department</li> <li>▪ Hosted in-house</li> <li>▪ Heavy customization</li> </ul>	<ul style="list-style-type: none"> <li>▪ Single tenant - implementation by vendor</li> <li>▪ Hosted by vendor</li> <li>▪ Very high degree of customization</li> </ul>	<ul style="list-style-type: none"> <li>▪ Multi-tenant - implementation by vendor</li> <li>▪ Hosted by vendor</li> <li>▪ Configuration - NO customization of core code</li> </ul>
<b>Maintenance and support</b>	<ul style="list-style-type: none"> <li>▪ Maintenance and support by customer's IT department</li> </ul>	<ul style="list-style-type: none"> <li>▪ Maintenance and support by vendor</li> <li>▪ Monthly hosting and support fee</li> </ul>	<ul style="list-style-type: none"> <li>▪ Maintenance and support by vendor</li> <li>▪ Consulting and value-added services</li> <li>▪ All-inclusive monthly service fee</li> </ul>
<b>Upgrades and change requests</b>	<ul style="list-style-type: none"> <li>▪ Product roadmap is owned by customer's IT department</li> <li>▪ Change requests handled internally</li> <li>▪ Upgrades not supported</li> <li>▪ New release requires new license purchase and re-implementation</li> </ul>	<ul style="list-style-type: none"> <li>▪ Product roadmap relative to specific implementation is owned by vendor</li> <li>▪ Change requests by vendor</li> <li>▪ Upgrades require migration or re-implementation</li> <li>▪ Upward compatibility risk: customizations may be lost or not supported in future releases</li> </ul>	<ul style="list-style-type: none"> <li>▪ Single product roadmap across all implementations is owned by vendor</li> <li>▪ Change requests by vendor</li> <li>▪ Automatic, free upgrades allow clients to benefit from all product enhancements</li> </ul>
<b>Conclusion</b>	<ul style="list-style-type: none"> <li>▪ Considerable upfront investment in software, hardware, implementation</li> <li>▪ Application owned and supported by IT department makes delivery model vulnerable</li> <li>▪ Unsupported upgrades and heavy customizations result in high total cost of ownership, unpredictable cost model and significant risk of sunken costs</li> </ul>	<ul style="list-style-type: none"> <li>▪ Considerable upfront investment in software and implementation</li> <li>▪ Unsupported upgrades, high customization, and single tenancy result in high total cost of ownership, unpredictable cost model and high probability of considerable hidden costs</li> <li>▪ Upward compatibility issues related to customizations and bug fixes result in high risk of service disruption</li> </ul>	<ul style="list-style-type: none"> <li>▪ All-inclusive service fee guarantees transparent and predictable cost model</li> <li>▪ Multi-tenancy and configuration strategy guarantee maximum benefits from product upgrades and enhancements without risk of service disruptions</li> <li>▪ Implementation timelines and expenses reduced due to configuration strategy</li> <li>▪ Value-added services and best practice-based consulting maximize benefits for entire client base</li> </ul>

© Copyright 2006, Peopleclick, Inc. Peopleclick is a registered trademark, and Peopleclick RMS and Peopleclick VMS are trademarks of Peopleclick, Inc. All other trademarks are property of their respective owners.