Software as a Service (SaaS)

ASSOCIATED SOFTWARE CONSULTANTS

PODCAST

Podcast by Tim Liston, ASC president, and John Liston, director of loan origination products, respectively for Associated Software Consultants, Inc. a leading developer and marketer of mortgage lending software solutions. Tim and John provide their insight on issues impacting the mortgage technology industry in a series of podcasts. Below are some additional notes relating to the podcast.

SaaS, Sorting It Out

There's been a lot of talk about Software as a Service (SaaS) in the mortgage trade publications lately. A lot of what has been written doesn't make a lot of sense to us, so we wrote up some of our own questions and answers to help sort out fact from fiction.

What is SaaS?

We look at SaaS as the combination of two things. One, SaaS implies application hosting by a third-party provider, which in turn implies Internet-based access by end-users and centralized management of the application. And two, SaaS promises to eliminate large up-front license fees in favor of ongoing subscription fees. It is as simple as that. None of this is particularly new – horizontal applications like CRM have been available as SaaS applications for several years. However, SaaS as a deployment option for a front-end mortgage or consumer origination system is relatively new because such applications are industry-specific.

What's important to understand is that most applications can be priced and deployed as SaaS applications, including obsolete “legacy” applications. What that means is that it is critical that mortgage lenders “look under the hood” to make sure that the loan origination system (LOS) or other system it is considering uses a modern programming language and tools. Systems that are obsolete can be redeployed as SaaS applications, and some of the vendors of these applications do exactly that to apply a modern veneer to an otherwise ancient system that is difficult to support. Legacy LOS vendors, and there are many, are really good at talking the talk. But walking the walk, well…

Certainly one of the myths about SaaS is that systems priced and deployed this way are technically up-to-date. That often is not the case at all.

Is a SaaS application easier to deploy than an in-house application?

Perhaps the most important myth is that SaaS systems can be deployed more quickly than their in-house counterparts. Sure, SaaS deployed applications can be installed on a server and readied for use more quickly than say LAN-based client-server systems. But what takes the most time in any application, and especially in a front-end LOS, is the customization to a user's unique requirements, and the training of the staff who will use it. How a system is deployed has little correlation with the training and customization effort. And frankly, true Web applications like a Web-based LOS, can be more difficult to customize than a traditional GUI desktop application. By “Web applications” we mean software applications that execute primarily HTML/CSS/JavaScript/etc. in a browser and are hosted by a Web server such as Apache or IIS.

When Web applications are deployed more quickly, it is most often because there are few options available to tailor the system. Plus, most computer users already know how to interact with an application via a browser so UI training is not an issue.

Must a SaaS application be a Web application?

Another myth is that only Web applications can be deployed as SaaS applications. These days, with “desktop virtualization” becoming commoditized, GUI desktop applications that have been traditionally deployed as client-server systems on an in-house LAN or WAN can now be centrally hosted and

managed, and used on any remote PC with no application software. So, in fact, these days it is possible to enjoy both the robustness and full functionality of a client-server system and the ease of deployment of a Web application. Thanks to desktop virtualization you get the best of both worlds.

What are some advantages to SaaS?

- Mobility. SaaS applications can be used from any PC with Internet access
- Does not require in-sourced IT infrastructure and staff
- Can use public Internet to deliver functionality
- Can leverage open-source stack (managed service providers are often more proficient with open-source software than in-house staff)
- Low up-front cost, and as a result less risk of large monetary losses associated with failed projects

What are some disadvantages to SaaS?

- Long term software cost is probably greater
- Vendor Lock-in: Lock-in is a possible issue with any application, and no less so with SaaS applications. Think about the costs of switching vendors later on.
- Data control: Your valuable data is stored by your hosting provider, who may not share your security principles, or who may make data integration complicated.

You seem to have a bias against what you call "Web applications" systems. Why is that?

For a number of reasons, Web applications are just not as functional or easy-to-use as are desktop-based systems. For example, heads-down applications benefit from keyboard shortcuts like Alt-this and Ctrl-that, which Web applications simply do not support. There are lots of useful features like that missing in Web applications. Web applications are also relatively sluggish, are not well-designed for data entry and do not validate data very well. Web applications are also harder to develop and test.

Now that desktop-based systems can be centrally hosted and managed, and deployed much like Web applications, there really is no reason to tolerate the shortcomings of Web applications.

What SaaS Myths are Perpetuated by Legacy LOS Vendors?

As we mentioned before, vendors of legacy LOS systems invoke trends and use buzzwords to make it sound like their systems are technically up-to-date. For example, legacy vendors tend to throw around a broad definition of Service-Oriented Architecture (SOA). "True" SOA uses technologies such as XML, SOAP, and WSDL and has yet to be widely adopted by some mortgage system vendors. Most data interchange is not accomplished using these SOA technologies. Legacy vendors also pay lip service to open source software, but SaaS and open source are two different things.

Question Comments on this or any of our podcasts? We would like to hear from you. Tim Liston can be reached at listont@asconline.com, and John can be reached at listonj@asconline.com.