LEARNING TECHNOLOGIES IN THE CLOUD – ARE YOU ASKING THE RIGHT QUESTIONS?

What are the pitfalls of evaluating SaaS? Ray Ruff gives a brief analysis of the key technical aspects leading to a successful implementation.

Almost all L&D teams involved in selecting and implementing learning technologies come across the challenge of evaluating the Software as a Service (SaaS) offerings now available from all major vendors. SaaS is at the heart of many business strategies today, including learning. As companies look to differentiate themselves in this field, there has been a tendency to obfuscate the true meaning of this phrase. So, let's start at the beginning. Wikipedia's definition is as follow: ***Software as a Service**, sometimes referred to as 'on-demand software', is a software delivery model in which software and associated data are centrally hosted on the cloud. SaaS is typically accessed by users using a thin client via a web browser.*

Looking specifically into the L&D industry, it is the case that most new deployments of Learning or Talent Management Systems in the past two years have actually been SaaS as opposed to the traditional model of on-premise implementations. SaaS offers compelling advantages to learning organisations in terms of predictable total cost of ownership, easier administration, rapid deployment and global accessibility. However, there are two areas in which there is still some degree of controversy: What the key considerations in evaluating a SaaS offering should be; and how secure are SaaS deployments, given that data is 'in the cloud.'

WHAT REALLY MATTERS

Many of the advantages of SaaS relate to well-designed, robust delivery architecture efficiently and cost-effectively managed. As you would expect, the specifics of SaaS architectures vary considerably from one vendor to another, so any buyer needs to carefully evaluate their options based on their specific needs.

Here are some key points to consider:

Automatic admin

Most SaaS providers manage the systems environment, security and upgrades at no additional cost to their clients. This, along with the speed of deployment, is one of the most attractive aspects of SaaS.

One Code Base?

In an optimal SaaS model, all clients of the SaaS provider share the same code base. Some vendors do not allow customisations. Some allow the addition of customisations or integrations to the standard code base as configured components. This ensures

that each of their clients can seamlessly upgrade to the next release without any additional services or costs required.

Rapid innovation

SaaS providers that have all their clients on the same code base can always make the latest and greatest feature set available to all their clients in a clean and efficient manner. New application releases incorporate new features suggested by all their clients, leveraging the community at large for best-practice improvements that benefit all users.

Control: or lack thereof

In many cases companies need to control the timing of environment changes, especially if much has been invested in the current release user documentation; and time is needed to prepare the user community for a change, especially if you have sensitive program roll-out timetables or regulatory approval requirements that make upgrades a disruptive issue.

This may be a challenge as some widespread multi-tenant[1] SaaS solutions tend to be weak in this area. However, single instances offer greater flexibility and control as each individual company dictates the timing of an upgrade.

Requirements adaptability

Customisation is a dirty word in modern software thinking. Every vendor wants to make the environment as flexible and configuration-driven as possible, because this is a win-win for the client as well as the vendor. Certainly client-definable features such as custom attributes for modules and users, variable enrolment policies, auto-assignment rules, custom ad-hoc reports etc., are part of the standard functionality expected of any modern enterprise software application and do not require customisation.

But in the real world there are some things that require special attention. How you handle these special requirements is critical. For example, integrating with back-end systems (e.g. HRMSs or ERPs) requiring some internal handling changes, and special regulatory or legal requirements not necessarily handled by native software workflows. Many 'pure' SaaS implementations struggle with this value proposition. However, for some vendors it is possible to incorporate these special cases into configurable components that are a part of the standard code base to maintain single code base cost advantages.

Every vendor wants to make the environment as flexible and configuration-driven as possible, because this is a win-win for the client as well as the vendor.

SECURITY – THE ELEPHANT IN THE ROOM

For many organisations a big concern about SaaS is the lack of 'control' over their data once it is in the cloud.

Security is complex and constantly evolving, so much effort in this area relates to establishing best security practices, preventing specific (known) types of vulnerabilities, and essentially minimising overall risk levels to cater for the unknown. While there are well-established standards and best practices to manage this problem, they require foresight and effort to carry out properly. Unfortunately, anyone can say they are secure, but the real question is can you prove it?

Some key considerations for a secure SaaS environment are:

Security management standards

Perhaps the most respected and well established standard for data centre security practices is ISO/IEC 27001, an information management system established by the International Organisation for Standardisation in 2005. This system requires that management design a comprehensive framework of security controls and practices that address information security vulnerabilities and risks. Most importantly, it is a standard that requires an external audit and certification by an outside organisation and it is very much a security-focused audit.

Some SaaS providers rely on SAS70 Type II, which is not a security certification but primarily a statement by an auditor that stated financial controls are fairly implemented (although nothing is specified as to what the security controls actually should be). The weaknesses of SAS70 (which has been found wanting as a standard) are spurring additional efforts in the security space, such as NIST CyberSecurity standards.

Location, location, location

For many reasons multinational corporations must navigate a minefield of country-specific legal guidelines regarding data privacy controls that include HR and training data. For European operations it is essential that data in the cloud be contained somewhere in the EU to ensure that the company stays in compliance with EU data privacy requirements. Likewise, US companies often prefer the legal protection of the US for their managed data. So, in reality, where the data is in the cloud actually matters for many clients.

Infrastructure architecture

Providing the minimum security is often all that you get from a SaaS vendor – often this means they have a firewall, require eight-character passwords, and keep their fingers crossed that nothing bad happens. However, a serious security infrastructure can go much further and establish procedures such as intrusion detection, VPN connectivity, regular penetration testing, audits of all server activity, back-up encryptions, and many more.

Client data compartmentalisation

In the most secure of cases, each client should have their data in their own database, with application and content elements also compartmentalised. Many multi-tenant SaaS implementations mix these information sources. This can be done in a managed manner, but relies on limiting client functionality (for example, the ability to connect external third-party report tools to the database) and expert programming, which requires a lot of trust.

Make it secure before it's in production

As if the above arguments were not enough (and with security there is never enough), some vendors go through regular source code scanning for security vulnerabilities in addition to all the automated unit and system tests. It is impossible to guarantee any system or environment 100% secure. However, it is inexcusable not to do everything that can be done in a practical, common sense manner beforehand to prevent a problem from occurring needlessly. These preventive measures are another powerful weapon in managing risk and reducing vulnerabilities.

As you can see, SaaS offers great benefits. But buyers need to evaluate solution providers on the premise that good software should adapt a much as possible to the way an organisation works, not the other way around.

[1] *Multitenant applications rely on a software architecture where a single instance of the software runs on a server, serving multiple client organisations (tenants).*

Ray Ruff is Chief Scientist and Co-founder at NetDimensions
www.NetDimensions.com