Your ERP Terminology Guide

CINDY JUTRAS, Principal, Mint Jutras

By CINDY JUTRAS, Principal, Mint Jutras [1]

With all the hype circulating about the cloud, it is no wonder that there is ample confusion over how enterprise resource planning (ERP) is bought and paid for. Even for seasoned enterprise application buyers, terminology can be confusing. SaaS. Cloud. On demand. Hosted. On premise. Perpetual license. Term license. Subscriptions. Multi-tenant. Multi-instance.

Do you know how all of these are different and where they have similarities? Let’s get down to basics. For those familiar with some of this, bear with me. For those who need a primer … here goes.

Let’s first talk about how you pay for ERP, or any enterprise-level software for that matter. Enterprise application software is typically not bought and sold; it is instead licensed for use. It may be licensed to be used by a company on a particular computer or by other criteria, such as the number of users. This is similar to consumer software. Buying it once doesn’t mean you can duplicate it and share it with all your friends, or even sometimes with all your own computers. For enterprise-application software, how you pay for that license and the term of the license can vary dramatically.

A software license can be perpetual. That means you pay for it once and can use the enterprise application forever. Maybe. This used to be the case, but more and more often today, a perpetual license agreement might have a stipulation that you have the right to use that software only for as long as you continue to pay maintenance to the software vendor that provides the product. So be careful to read all the fine print in the contract.

If all that you are familiar with is consumer software, there may be no equivalent. If I buy a product like Microsoft Office or Adobe Acrobat, I can continue to use the software on my computer as long as I would like. But if Microsoft or Adobe comes out with a newer version, they don't just give it to me. I might get a break on the price if I buy the newer product, but I still have to buy it.

A maintenance agreement, which is a recurring cost, typically provides both technical support and certain innovations. Some of those innovations will be included in my maintenance fee and others I may still have to purchase. Maintenance is typically priced as a percentage of the software license. The going rate at list price today is around 22 percent for ERP.

But perpetual licenses are not the only type offered. Instead, your license might be for a specific period of time. This is generally referred to as a "term" license. At the end of the term, you must either renew the license or discontinue use of the software. In fact, the application might have the equivalent of a kill switch in it that will disable it and prevent you from continuing to use it at the end of the term. This requires some license management code to be embedded in the solution and is not always done, particularly in older software. If it is not, and you don't renew, you might find some software auditors on your doorstep.

Subscription-based pricing usually represents a form of a term license. And this is one source of confusion. You are able to use the software as long as you keep your subscription current. But this is where some confusion starts to creep in. Often people equate subscription to SaaS. But SaaS is an acronym for software as a service. In this case, the customer doesn't purchase a license, but instead pays for the software as a service. So generally, most applications that are delivered as SaaS are paid for through a subscription, but not all subscriptions are delivered as SaaS. A subscription-based license may be offered as a way to reduce up-front costs and therefore knock down barriers of entry.

When software is installed at the company's site, it is generally referred to as on premise. But some companies prefer not to invest in their own computers, or may choose to outsource the care and feeding of the application to a third party. But this doesn't necessarily mean the software is delivered as a service. Often, the software is just delivered to a different destination and is licensed just as it would be if it were running on premise.

This is generally referred to as a hosted environment, and services may also be purchased to perform that care and feeding, in addition to the hosting fees. In a hosted environment, the company that licenses the software may, in fact, be responsible for running the application itself, and just outsource the technical infrastructure and maintenance. Or it may decide to also outsource much of the work involved in running the application. This is generally referred to as application-managed services (AMS).

If the software is not located at your physical site, but you can access it virtually, this operating environment is generally referred to as a cloud. A cloud can be

private — nobody can access it except your company. This, in fact, is likely to be the case in a hosted environment as well.

Or a cloud can be public — you access it through the Internet, and you may indeed be sharing the software with other companies, even though your data is secured from anyone else seeing it. This is more likely to be a SaaS environment. In an SaaS or on-demand model (and I use the two terms interchangeably), the software itself is neither licensed, nor owned by the company using it.

The software is delivered as a service and is typically paid for through a subscription for the service provided. Cloud terminology is often intermingled with SaaS, but reference to the cloud simply refers to the operating environment, and not how the software is bought or paid for.

Now SaaS purists will insist that, in order to be true SaaS, the solution must also be multi-tenant. This means that there is only one instance of the software itself. The data belonging to each subscriber to the software as a service is segregated and secured. But everyone runs a common set of code and configuration settings tailored and personalized to their business processes. But, in fact, the software can be delivered as a service in a single-tenant or multi-instance, rather than a multi-tenant environment.

In reality, whether the solution is delivered multi-tenant or multi-instance matters far more to the vendor than to the end user. It is the solution provider that benefits most directly from being able to offer a multi-tenant solution because this allows them to scale delivery with less cost. Obviously delivering bug fixes and product innovation to a single instance of software supporting many different customers is far easier and more efficient for the vendor.

But, in fact, some vendors choose to not deliver their SaaS solutions as multi-tenant for one of two reasons : Either their solution is not architected to support this, or because they feel they can deliver a more customized solution through multiple instances. In fact, some companies purchasing ERP solutions prefer not to run in a multi-tenant environment for the same (latter) reason. While it is not impossible to deliver customized solutions through a multi-tenant SaaS solution (in fact, Plex Online does this), it adds a level of complexity for the solution provider that few have mastered and most are not willing to absorb.

To the non-technical ERP users, the most important aspect is that they are able to connect to the application and its data from any computer with a browser. This may be accomplished with any of these deployment and license options. If, in fact, this is possible, often times the end user does not know, care or need to know which of these deployment options are actually being used to deliver the application, and they are even less likely to care how it is paid for.

But for those responsible for the purchase and deployment decisions, it is of paramount importance to understand all these potentially confusing options.

What’s your take? Still confused? Please feel free to contact me at

cindy@mintjutras.com *[2] or leave a comment below! For more information, please visit* www.mintjutras.com *[1].*

Source URL (retrieved on *08/20/2014 - 9:09am*):
http://www.chem.info/blogs/2012/05/your-erp-terminology-guide?qt-recent_content=0

Links:
[1] http://www.mintjutras.com/
[2] mailto:cindy@mintjutras.com