

**Q&A**  
**WEBINAR: BUYING EVERYTHING AS A SERVICE**  
**SPONSORED BY ISM INDIRECT/SERVICES GROUP**  
*June 18, 2015*

---

**1) Can you describe what IP rights/protections can be added to software-related contracts for customizations that were jointly developed by the contracting company and the software provider; particularly if those customizations would provide the contracting company with a competitive edge?**

**A:** (Mark Trowbridge) This question was answered orally in the webinar, but to summarize...

- Often a technology provider will “partner” with a leading edge customer to Beta additional functionality. If the provider can later offer that new functionality to their other clientele, they often will do the enhancement project at little or no cost to the Beta client. In the world of SaaS utilizing a true SOA (Service Oriented Architecture), the functional improvements must be perceived as beneficial to most/all of the provider’s other clients to make the project worthwhile.
- A true SaaS offering will be less-probable for a non-compete arrangement than would have been older style License/Install/Maintain offerings. Pure SaaS is an architecture where one application is run on behalf of all of the provider’s clients...thus non-compete restrictions work contrary to the provider profit model.

**2) Software vendors will not use cost plus models – what are the popular pricing models utilized by software vendors?**

**A:** (Mark Trowbridge) This question was already answered orally in the webinar...but to summarize and add a few additional concepts...“Cost-plus” pricing models are very difficult to format in an arena of pure services which pass through CAPx costs to the users. Thus, a fixed pricing model, or discount % off of a list price, in a subscription mode are becoming normative in the Cloud world. Important cost factors to target in any SaaS deal include:

- Pricing for integration of SaaS application into other buyer systems.
- Pricing for customization/tailoring.
- Pricing for training (user, administrator, and technical).
- Price of subscription blocks (each with a specific number of users...i.e. 100, 50, 25, 10, and 5)
- Subscriptions for users should be “contemporaneous” (i.e. so that you only pay for a block of users who would be simultaneously logged in to the application...NOT paying a subscription fee for every user who might ever log in)
- Decreased subscription prices (tiered) for additional users.
- Ability to reduce number of users following initial subscription term.
- Pre-negotiated pricing for additional data storage.
- Etc.

**3) Some companies struggle with procurement of SaaS since they are unable to capitalize the expenses and classified as O&M. Can you share your insight with this challenge?**

**A:** (Mark Trowbridge) This question was already answered orally in the webinar, but to summarize...Yes, accounting became more complicated in moving from traditional License/Install/Maintain where the first two cost elements could be capitalized and depreciated as a fixed asset. With the movement to SaaS, IaaS, and PaaS, the annual subscription fees are typically ‘expensed’ during the year of expenditure. Notwithstanding the foregoing though, there is little accounting difference between having a depreciated portion of an upfront CAPx cost hitting the books in a fiscal year OR having an annual subscription fee hitting the books in the same time period. The main differential will be the term of the depreciation for SW License/Install versus the service cost allocation used by the provider in a SaaS model...both are spreading certain costs over a defined time period – - but the length of the time period will likely vary under each methodology, thus altering the annual impact. As Pierre suggested in our webinar discussion of this question, some companies get around the issue by approaching SaaS as a lease...although bundling multiple annual term expenditures together structured as a lease, and then paying interest on it doesn’t make much sense in the big picture. Still, how many ‘interesting’ deals are struck for accounting reasons...☺?

**4) What is an appropriate range of ROI of utilizing cloud-based services?**

**A:** (Pierre Mitchell) It is impossible to provide a prescriptive model that can really be used off the shelf. Every company will have a different situation based on so many factors: what type of service (e.g., SaaS/PaaS/IaaS), complexity, integration, risk profile, IT competencies, match to supply market solutions, and on and on. There is a lot of open web content though if you want to start building a model. The IT industry analysts often put this type of information out. For example, here are a few SaaS ROI models from Forrester....

[http://www.crosscountry-consulting.com/literature\\_217944/Workday - ROI of SaaS](http://www.crosscountry-consulting.com/literature_217944/Workday_-_ROI_of_SaaS)

<http://www.truecloud.com/c.830372/site/ForresterROI.pdf>

[http://blogs.technet.com/cfs-file.ashx/\\_key/telligent-evolution-components-attachments/01-8044-00-00-03-48-19-74/Analysis-of-SaaS-and-On-Premise-ICT-solutions-for-SMEs-in-Turkey.pdf](http://blogs.technet.com/cfs-file.ashx/_key/telligent-evolution-components-attachments/01-8044-00-00-03-48-19-74/Analysis-of-SaaS-and-On-Premise-ICT-solutions-for-SMEs-in-Turkey.pdf)

Here is a good article that highlights some of the SaaS cost elements from a providers standpoint...

<http://cloudstrategies.biz/imperative-reduce-cost-saas-service/>

**5) How would a company handle the scenario of two SaaS providers sharing data on the company's behalf in order to integrate certain processes?**

**A:** (Mark Trowbridge) Many SaaS solutions are "integrated" with others for combined functionality. That's not unusual and can be handled cleanly. A well-written agreement with each firm is essential in this case. But the danger in any multi-party provider situation is the tendency for them to ascribe blame and risk to the "other guy". "Whose fault was this? - It is their fault. No, it's your fault". This is especially true when the two solution suites overlap with each other (i.e. they compete in the marketplace for the SaaS functional modules the customer has elected to use from the other provider). It becomes more messy if you have data going back and forth between solutions rather than a clearly-defined linear pathway across solutions. Another method is for the customer to contract with one provider as the "prime" and have that firm then subcontract with the other firm and thus hold sole responsibility for the combined performance of the combined services.

**A:** (Pierre Mitchell). Mark's comment on assigning primary responsibility is a good one – in addition to having good individual agreements. Also, each providers system that extracts/accepts data from an another is in essence acting on your behalf as if it was a human user accessing the data. Still, each provider will likely have their own requirements that they'll present to you about this in terms of other provider's accessing their systems. Often, they might have to build a specific API that they'll feel comfortable using as the only method to support the scenario you're looking for.

**6) Software vendors tend to be bought out by larger companies – how should a procurement professional handle? Is the only concern/option to assign the original agreement to the new company?**

**A:** (Mark Trowbridge) This is a very misunderstood issue. Many procurement folks unknowingly set themselves up to allow the whole deal to be renegotiated by the salesperson for new merged SaaS provider...when they don't need to. As a matter of law (check with your own attorneys of course), when one company buys another they automatically (i) benefit from the acquired companies assets such as customer revenues; and (ii) accept the acquired companies liabilities. This is a concept called 'successor in interest' and happens automatically. No assignment is needed because you still have a contract with the new merged legal entity.

But other exposures do happen when mergers occur, whether on our side or the providers. There's some really good language every company should have in their key SaaS, IaaS, and PaaS agreements to put yourself in a controlling position when any merger occurs...shoot me a note at [MTrowbridge@StrategicProcurementSolutions.com](mailto:MTrowbridge@StrategicProcurementSolutions.com) if you'd like an example.

Finally, a very real danger in the rapid M&A environment in today's technology world is the dumbing down of functionality. As billionaire Warren Buffet once said, "First come the innovators, then come the imitators, and then come the idiots." All of us have seen "best of breed" solutions acquired by larger providers and then sunsetted; or only a portion of their excellence morphed into the new company's standard offering (grrrrr). From a contractual point of view, you may not be able to prevent a merger from occurring BUT you can insert language stating that any updates/upgrades/replacements to the SaaS tool will contain "equal or better functionality".

**A:** (Pierre Mitchell). Mark covered this well. The biggest risk to you is that the innovation tends to slow and service degrades as your acquired provider gets absorbed into the larger player. Also, depending on your view of the acquiring company, when the deal is first announced, you have a window to pursue actions in your best interest (e.g., renegotiate for longer duration if you expect a price increase; or conversely, begin to look for alternatives and for ways out of the contract if necessary... such as service degradation.). You can also negotiate a 'poison pill' clause that helps eliminate this problem. Vendors shouldn't have a problem with them if they're truly "in the market for the long haul", right?

**7) With open systems and SaaS added on from multiple suppliers, what should procurement professionals keep their eyes on?**

**A:** (Pierre Mitchell). Not sure I understand the question, but use of open standards is always a big plus in evaluating providers, and similarly, choosing technology built on 'industry standards' is desirable, but challenging given how the mega vendors want to make their technology stacks 'standard' (often an oxymoron). Finally, open source based providers can be useful to consider in certain infrastructure areas (e.g., Linux in OS; MySQL in database; etc.), and it can bring some good competitive leverage in many situations when large incumbent providers on older technology stacks are in the mix.