



Print Article



Close Window

From: www.cio.com

Are Cloud Computing Pay Walls Coming?

– David Taber, CIO

July 06, 2011

When it comes to innovation at low cost, it's hard to beat cloud services and applications. As a user of these apps, there's basically no infrastructure cost beyond your ISP fees. On the supply side of apps and services, cloud deployments result in huge cost efficiencies, particularly if the supplier has designed around a multi-tenant architecture. Even if they've just used simple VM deployments, vendors can wring a whole lot more value out of a server rack by driving up utilization rates.

There really are some free lunches in the cloud, but they're not going to last forever. Some very useful free cloud services have gone dark recently, and it seems that even some academic projects aren't getting grants renewed. In the commercial world, some cloud applications have put up a pay wall, now that their user bases have grown and the VCs have grown impatient. The bottom line is that you can't bank on free, and you need a transition strategy when economic reality sets into the cloud. This applies equally to applications, plug-in products/extensions, and services deployed in the cloud.

The Private Cloud

If you're already following a private cloud strategy, you're almost certainly paying for all the key assets already. This gives you essentially complete control of your destiny.

But you might want to double-check for freebies that you take for granted. You may be using a service — even something as simple as a mashup — that has been given away as a teaser. If you can find an open-source version of that freebie, double-down on the private cloud strategy by bringing the service in-house and hosting it in your own cloud. You'll have to bear some incremental deployment, administration, and maintenance costs, but you'll have gotten rid of a dependency.

If the freebie service isn't open-sourced, talk to the provider of that cloud service. See if they are willing to provide a service contract to you with an SLA that applies for as long as you need the service. Beware the tempting upsell: those bright glittery features that the vendor will be advertising for future releases may be the equivalent of bloatware. More important, their "improved" versions are unlikely to be bug-for-bug compatible with what you're using now. Every new feature brings with it more compatibility testing and temptations to rework your side of the application. This usually means integration changes that look trivial but seldom turn out to be. Better to [keep it simple](#).

The Public Cloud

If you're following a public-cloud strategy, the situation is a bit different: your entire objective is to avoid having to take ownership of (or responsibility for) most (if not all) of the services you're using.

Public cloud freeware apps usually start with limited functionality in the totally free version. Within a few quarters, it's typical to see some sort of advertising or sponsorship model added to the UI. (I have yet to see an advertising model added to a SOAP or JSON API, but somebody will devise one soon enough.)

Usually, the pay wall is erected a few months later, in the form of an unrestricted version that enables additional features or higher usage levels of the app or service. In most cases, the freeware is still available and isn't explicitly changed. (However, I have recently seen examples where the freeware is literally turned off or put into read-only mode, rendering it essentially useless.) In order to justify the price for the "full up" version, vendors typically add enough new features that the original user experience and APIs are obsoleted. At the very least, you'll need to do some testing.

What if Free is the Most Expensive Strategy?

Free is great for demos and proofs of concept, but that doesn't mean you want to put free items into production systems. If the cloud freebie is from a consortium, government agency, open source vendor, or university with solid funding, you're probably not running a big risk. But if the freebie is coming from a small startup, there's no way of telling (let alone assuring) how long it will be around in its current form. So if you leverage these Cloud freebies, make the integration as light-weight and modular as possible, and put a budgetary place-holder in the out-years in case you need to replace the application or service with something else or your own code.

If the service or app you're looking at using is available commercially, the lowest long-run costs will come from paying for that version. Most of the fees are nominal in comparison with the DIY (or should I say re-DIY) project that can result from depending on freeware versions.

And Then There's Support

If paid-for cloud services can have seriously lame support (and the answer is "yes"), it's irrational to expect better support from a freebie. I'm not going to mention names, but you're definitely going to want to research your cloud vendor's reputation for support before you commit in a big way. Go to their discussion groups, user forums, and other online sources to find out what their service levels really are. If your Cloud vendor also has on-premises versions of their software, make sure to separate the reputation for their high-cost enterprise support from the one that's relevant to you in cloud-space. You'll find that some old-line vendors that really ought to know better are way under-investing in support...which means you'll have to invest more in self-support.

The cloud doesn't change this part of software economics.

David Taber is the author of the new Prentice Hall book, "[Salesforce.com Secrets of Success](#)" and is the CEO of [SalesLogistix](#), a certified Salesforce.com consultancy focused on business process improvement through use of CRM systems. SalesLogistix clients are in North America, Europe, Israel, and India, and David has over 25 years experience in high tech, including 10 years at the VP level or above.

Follow everything from CIO.com on Twitter [@CIOonline](#).

© 2010 CXO Media Inc.