



## Introduction

A large educational institution was in the market for a new Learning Management System (LMS). They hired a consultant who spent several months putting together the procurement process which resulted in a public Request for Proposal (RFP) 150 pages long. The management team had heard great things about Moodle, and knew that most of their colleagues in the sector had selected Moodle. Surprised that “Moodle didn’t respond” to their RFP, they selected a proprietary LMS that no-one else in the country was using. Nor did any Moodle services supplier respond. What isn’t surprising is that they have had cost control, support, and software interoperability issues ever since. Where did they go wrong?



Government entities and the corporate market are now embracing open source software like never before, primarily for its twin attributes of cost-effectiveness and flexibility.

### The software market has changed

Government entities and the corporate market are now embracing open source software like never before, primarily for its twin attributes of cost-effectiveness and flexibility. IT departments are now aware of the undeniable benefits, and routinely use open source applications within large, mission-critical systems. In parallel an innovative marketplace has emerged, delivering a range of services such as implementation, maintenance and customisation. Forward thinking governments have developed policy positions to encourage departments to harness the benefits of an increasing pool of mature enterprise-level open source software.

For example, in the UK’s report ‘Open Source, Open Standards and Re-Use: Government Action Plan’ policy statements include:

*“Procurement decisions will be made on the basis of the best value for money solution for the business requirement, taking account of total lifetime cost of ownership of the solution, including exit and transition costs”*

*“Where there is no significant overall cost difference between open and non-open source products, open source will be selected on the basis of its additional inherent flexibility.”*



The uptake of open source software is undoubtedly growing, but there remain significant barriers to adoption caused by procurement practices which favour (often unintentionally) the typical proprietary software business model.

The purpose of this white paper is to explore the reasons why, and recommend how, prospective customers can act to ensure they receive all the potential benefits from looking at both proprietary and commercial open source options.

## Comparing the revenues versus services

When you purchase proprietary software you pay twice. Once for the most common element, a license granting the right to use the software. And a second time for compulsory support services. These are usually structured on a per user basis, and renewed annually. Because the code is closed, software licensing can be viewed as a form of rent for the intellectual property encapsulated within the software. In the open source model, there is usually no corresponding mechanism to receive rent for intellectual property.

For enterprise software there is usually a support agreement which is often mandatory to validate the license. The support agreement will include service levels, and some important business assurances that you will be supported if and when problems are encountered.

In stark contrast, there are no licence fees to pay for open source software and subscription agreements are not mandatory. Subscription agreements are most often compared to the proprietary support agreement. An open source subscription offers code support and maintenance, security updates, error correction, and patch updates. Where it differs from a proprietary support agreement is that it includes new features and new version releases. In the proprietary model new features are only available if license fees are paid. In either case it's worth purchasing support as this delivers transparent business value.

Simon Phipps argues that open source subscriptions offer much more than simple proprietary support agreements. Phipps makes the point that open source subscriptions should be more expensive than their proprietary equivalents on the basis that they deliver the functionality as well as the extra value of the inherent flexibility and freedoms that open source offers.<sup>1</sup> It's compelling logic, although the reality is that open source subscriptions tend to be much cheaper than their proprietary equivalents and this may be due to the comparatively wider mix of for-payment, and not-for-payment inputs in many open source projects. As an example, Totara LMS is a distribution of Moodle, and therefore the sunk costs of developing Moodle are not reflected in a Totara subscription, only the specific costs associated with developing and supporting the Totara extensions.

.....  
*The uptake of open source software is undoubtedly growing, but there remain significant barriers to adoption*  
.....

1 Open Source Procurement: Subscriptions. Simon Phipps, Opensource.com, March 2011  
<http://opensource.com/life/11/3/open-source-procurement-subscriptions>

## The challenges with procurement

The models seem straight-forward enough and the benefits of open source clear, so why are there any problems?

Firstly RFPs are generally inherently weighted towards a proprietary software outcome. These are often huge, complex documents with responses explicitly at the vendor's expense. Proprietary software vendors will put together bid teams and spend the requisite weeks of effort to respond because over time, when they win a bid, they will more than recoup the cost of the tender response through licensing fees. This is one of the contributing factors to the opaque relationship license fees have with the true value of the intellectual property being "rented" through licenses.

There is no corresponding inbuilt mechanism for the open source vendor to recoup the cost of sale when responding to a proprietary-styled RFP. In most instances, the investment in pre-sales can't be readily recouped so it is simply not logical to respond.

There are several other typical pre-sales activities that cause similar problems for the open source vendor. Free trials or proof-of-concept installations, often with extensive pre-sales consulting thrown in, naturally incur a cost. The proprietary vendor takes the approach of recouping through license fees. In a sense, the large pre-sales investment that still ends in unsuccessful sales is simply borne by customers paying high license fees typical of enterprise proprietary software. In addition the vendor can uninstall or disable the proprietary software if a trial doesn't translate into a sale. Not so with open source software, the would-be customer already has the code and has chosen not to purchase the support contract. The free trial carrot doesn't have the same proprietary stick accompanying it.

A key benefit of open source is the absence of vendor lock-in. This is as it should be from the client's perspective, but from the vendor's standpoint there is risk when investing in pre-sales that another entity may reap the reward of that effort. Furthermore, a customer may decide to take the basic subscription but none of the value added services. This presents another disincentive for extensive and unrewarded efforts in pre-sales. With open source, a vendor makes money from selling services around the code, not for selling the code itself.

Who is missing out and who stands the most to gain or lose if this procurement model was reframed to achieve a balanced evaluation of all options? Under the heavy proprietary styled RFP process, or other pre-sales activities like free trials or proof of concept implementations, both the customer and the open source vendor are losing out. By excluding the open source options the customer is likely to pay much more for their solution and more importantly, be locked in for years to come - losing the software freedoms available to them from the open source model.

If the procurement process was re-framed to establish a level playing field then the client stands to benefit from more pressure being brought to bear on the largesse of proprietary licensing fees.



.....  
*The models seem straight-forward enough and the benefits of open source clear, so why are there any problems?*  
.....

## Recommendations for re-framing the procurement process

The following recommendations are not inter-related but neither are they mutually exclusive.

1. First and foremost, balance your investment in the RFP process with real research into the options in the market. This may include using an initial Request for Information (RFI) that allows for responses to be in the vendor's formats e.g. information packs, brochures and product sheets. You then have the option to prioritise the vendors and select respondents for a closed RFP.
2. Split the procurement into two parallel processes. One RFP designed for proprietary solutions and one for open source solutions. Write the second RFP overtly for an open source solution. On top of the attributes of the product, criteria for selection will differ in some areas such as which open source license is being used, availability of commercial support, reputation of suppliers and their product development model.
3. Include a two-stage procurement component with a paid-for discovery stage of one-two weeks on your short-listed open source candidates. In other words engage the potential vendor commercially to help answer all of the same questions you may be expecting from the proprietary vendors as part of their pre-sales. If you don't want to consider paying for any pre-sales activity, undertake the research internally or engage a suitably qualified third party consultant who is up-to-speed on the potential of open source solutions and their benefits. It is worth noting that hiring a third party denies the purchaser potentially invaluable insights into the skills and culture of their potential suppliers.

.....  
*By excluding the open source options the customer is likely to pay much more for their solution and more importantly, be locked in for years to come*  
.....

Do not expect the open source vendor to operate in the same way as a proprietary software vendor. The software development models are poles apart as are the business models.

4. Place a score or qualitative value on issues such as vendor lock-in, exit costs, speed of the development and release process, number of contributing coders/companies in relation to your risk exposure, ability to keep the code and support of interoperability standards. These are valuable attributes often over-looked by the purchaser.

## Conclusion

Procurement models need to reflect the changes now occurring in the software marketplace. If the prospective purchaser is unable to adapt the procurement decision making process, this will be reflected in the limited range of market solutions presented.

By levelling the playing field with some understanding and appreciation of the different business models on offer, buyers of enterprise software stand to gain enormously from more choice, lower cost of ownership, freedom from vendor lock-in, control over their system's direction and flexibility to extend or customise their solution. In summary, this is known as software freedom, and the demand-side has as important a part to play as the supply-side.



.....  
*Procurement  
models need  
to reflect the  
changes now  
occurring in  
the software  
marketplace*  
.....