

## Going Open Source Software in IT – Opportunities and Challenges

Dave Thomas, Bedarra Research Labs

### GOING OSS IN IT

In this article we look at the opportunities and challenges of using Open Source Software (OSS) in IT. Since there has been a lot written on the benefits, we focus specifically on the challenges which face large IT organizations, most of which lack the latent technical talents of the early adopter community.

While initially limited to the Linux operating system and associated tools, open source solutions now include enterprise frameworks such as J2EE, as well as complete applications. IT adoption is driven by the enthusiasm of developers and increasingly by vendors seeking to challenge the closed source dominance of MS in particular and also, to some extent, IBM and Sun. The mainstream support of IBM, HP and Sun for open source Linux based solutions has made open source a safe choice for IT executives.

### THE BENEFITS OF GOING OPEN SOURCE

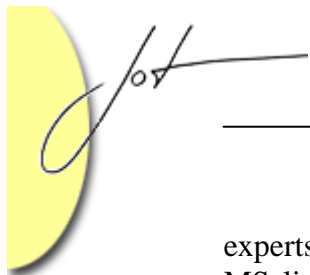
The benefits of open source have been widely published. Open Source Software promises significant benefits for IT development and deployment. The major benefit of using OSS is increased independence from a smaller and smaller number of closed source suppliers. This independence is seen to provide reduced expense, access to source code to reduce lock in, and access to more suppliers.

OSS system and application software is usually available at little or no cost from the Internet, although serious IT buyers will almost always acquire software through a trusted platform vendor or systems integrator. In most cases there are several suppliers that provide similar offerings with associated support.

The availability of source code allows IT to assume responsibility for critical fixes, especially those associated with stability or security<sup>1</sup>. These fixes can be provided using

---

<sup>1</sup> The benefits of open source and a community of experts is clearly a major benefit for independent identification and repair of defects. The challenge for large IT organizations is to carefully select and apply



experts who are independent from the developer of the software. Recent concerns such as MS licensing and security have made some larger companies focus on OSS solutions such as Linux/Apache. The final benefit is the potential that smaller, trusted and highly competent suppliers can be used to provide multi-sourced development and support.

One beneficial side effect of OSS is the increased adoption of the best OSS agile development practices such as: test first unit testing, acceptance testing, and meritocracies for design and release approvals; refactoring, continuous integration and integrated documentation.

## USABILITY AND CLIENT APPLICATIONS

In general, open source software is similar to most UNIX software and has the usability of UNIX application applications. This makes it well suited to server-based operation, but the client side GUI applications lack the polish, desktop integration and usability of commercial PC client products. Further, X based GUIs are somewhat notorious for their overhead and usability.

There are major efforts by vendors such as Novell and IBM to change this, but few serious office users would welcome a switch to Gnome and Star Office at least at this time. Indeed the Mono project seems determined to achieve compatibility. At this point Linux remains a long way from the usability of the Mac or even Windows.

Some organizations have chosen a hybrid approach using web based client applications on current MS Windows platforms to talk to shared server based apps and opting to defer upgrades to newer versions of MS products for office functions.

It is more likely that MS will increasingly integrate its office applications with Linux servers as has been announced recently by Sun and MS<sup>2</sup>. Further, the increased openness of MS Office with XML likely will result in most businesses migrating to this version over the coming year.

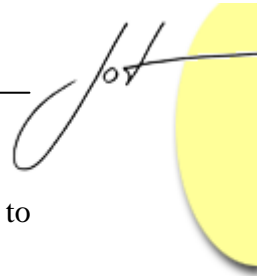
## THE ROLE OF GOVERNMENT PROCUREMENT

Increasingly governments, especially outside America, are mandating open source. Open source is chosen by governments and IT managers for essentially the same reasons: to reduce the expenses relative to closed source solutions; to obtain source code so that they

---

the appropriate security patches for each of the various OS components. In some cases the difficult of this process is increased rather than reduced.

<sup>2</sup> Given MS has a major investment in BSD, it would not be surprising to see MS Office running natively on Linux or BSD in the future should it make commercial sense for MS to do so. Indeed MS Office is already available for Mac OS X which is BSD based.



---

may have the ability to maintain the software independent of the original supplier: and to encourage the increased opportunities<sup>3</sup> and innovation<sup>4</sup> for local companies.

In the case of developing economies such as Asia, open source provides the opportunity to leverage their inexpensive highly educated labor pool with open IP which enables them to quickly develop services and products which are strong alternatives to established IT vendors especially for their rapidly growing local markets. Recently Asian governments have opted to develop home grown independent software and hardware IP for wireless, security and media compression. These efforts have the benefit of a large body of OSS code to seed their efforts. The global availability of expertise also reduces the economics of commercial open source product development in developed countries substantially.

## THE CHALLENGES OF GOING OSS

### Selection

Source code propagates bad practices as fast as or faster than good practice so as one would expect there are a large number of unfinished or very poorly coded OSS projects, making project selection a critical factor.

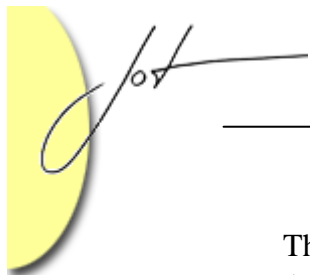
Online code repositories such as [www.freshmeat.net](http://www.freshmeat.net) and [www.newsforge.net](http://www.newsforge.net) provide comprehensive catalogs and notification of new software releases. Ratings include its OSS license and some rough measure of the vitality of the software. However, such voluntary catalogs and reviews are just a starting point in the selection process and often contain information provided by the vendors or enthusiastic early adopters. More recently reviews of OSS software are now appearing in mainstream sources along side those of their commercial counter parts.

While sources like [www.freshmeat.net](http://www.freshmeat.net) are often initially appealing due to their comprehensive nature, there is still a huge amount of effort required to sort through the information and determine which software is actually valuable.

---

3 Some contend that for many software sectors open source is a destructive commoditization that in the end favors increased consolidation into the hands of major “distributors”. Further, coupled with globalization and outsourcing, such government copyright initiatives may move software development into the same category as music. This is of course not what OSS zealots or government hoped for but it seems to be what economic reality will dictate - witness the current state of the music industry.

4 While this is the great promise of open source there is as yet little evidence of truly innovative applications. The lack of progress in an end user interface which addresses the problems that people complain about with Windows really flies in the face of innovation claims. One danger to the software industry is that open source becomes a way to condone the mere cloning successful of commercial products that will just accelerate increased software protection through copyright (limit reverse engineering and clear reasonable use) and patents (obvious and natural features) further limiting innovation.



The fact that software is OSS doesn't make it intrinsically better, free, reliable or easy to maintain. These characteristics can only be determined through careful evaluation or more likely through obtaining proper references from users in the same or similar applications. (In reality, trial and error, local testing etc. are the only ways to be sure) In many cases, the selection of the supplier of the OSS support and upgrades will be as important as the supplier of the initial product.

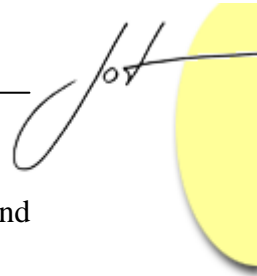
## TOTAL COST OF OWNERSHIP – WHO YOU GONNA CALL?

Open source is inexpensive to procure but the grim reality is that the major cost in IT organization is on going upgrades and support. The latter costs often far exceed the purchase price. Integrating complex applications which may require different versions of components is a challenging and difficult problem which demands expensive system integration. While Linux OS distributions have been greatly enhanced, there are still major challenges integrating associated middleware, applications and tools especially across Linux OS competitors who each have their own stack and install and update platforms. Clearly the closed source vendors such as MS have their own challenges with respect to release support, but OSS vendors will need to do better to ease the cost of ownership for large IT shops especially those using OSS components from different suppliers.

## OSS Licenses – So Many to Choose From

Software licensing has become increasingly challenging as vendors seek to find business models to maintain and increase revenues. This is best seen as a transition from software as a packaged product to software as a service or subscription. OSS licensing varies from public domain software which can be freely modified and redistributed even commercially to that which allows viewing of course code, but no modification or redistribution. Often OSS suppliers will offer different licenses to meet the needs of different customers ranging from personal user, IT or commercial software provider. It is important to understand the various OSS license options and to have a clear policy articulated for your organization. It is also important to note that just because a license is OSS foundation [www.oss.org](http://www.oss.org) approved doesn't imply the software may be freely used, modified, embedded or redistributed.

The most common and infamous OSS licenses are the GPL and LGPL licenses from the Free Software Foundation, which are designed to ensure that software developed under GPL is always available to the community at large. In particular GPL requires that all code enhancements to GPL licensed code be available for redistribution to others. This is a condition many commercial organizations are unwilling or simply unable to comply with. Indeed some organizations forbid the use of use of GPL code due to the *virial* nature of the license. Products such as Ghostscript and MySQL are provided free under GPL,



---

and may be otherwise licensed on commercial terms. Some licenses such as CPL and Eclipse for example cover software patents<sup>5</sup> contained in the software others do not.

Many closed source vendors have responded to competitive pressure from the OSS market by providing various forms of “community” licenses. However, virtually all community licenses have strong limitations of what can be done with derivative works, associated support and vest many rights with the vendors. They do provide access to source code which is now demanded by governments and large IT firms for security concerns.

OSS licenses are a particular challenge for companies that build products which embed OSS elements since this saves them time and money, but unless carefully managed may require painstaking legal review of all components and their licenses to ensure that the company has the rights to distribute their product, especially in binary form. Any organization which plans to redistribute software should undertake a careful review of its practices. In the worst case this requires all downloads to be vetted by the organization and/or comprehensive prerelease audits of the code base. .

The world is subject to a creeping, and in some cases accidental, expansion of © copyright as governments attempt to deal with products of the mind. This is especially true for creative works such as music, art and literature and may well spread to some or all areas of software. One very constructive response to this challenge is a new set of (cc) licenses from the <http://creativecommons.org/>. These licenses allow creators to easily and clearly communicate the intended use of their work.

One of the benefits of OSS is the frequent release of patches, new versions etc. in response to customer or developer enhancements or repairs. Unfortunately most IT organizations have differing rates of change to OSS software and the ability of the adopting organization to accommodate change is much slower than they are typically available. So an adopting organization has to make sure that it can either accommodate rapid change or that it can accommodate the existing software with its existing bugs, features, etc, and/or stage the changes appropriately.

## OSS CODE MANAGEMENT

### Source Code Can Harmful - A Brief History Lesson

The ideal use of source code is to understand where a component is failing or lacking functionality, so that one can work with the supplier or work around it. Modifying code owned by someone else makes it yours to maintain! Many years ago, it was common for IT to have full access to the source code for the operating system and many of the applications. IT organizations and their vendors developed enhancements in the form of

---

<sup>5</sup> It will be interesting to see the impact of the European software patent legislation should they reject software patents.

source and object code *patches*. While IBM was blamed for the Y2K problem on mainframes, the reality was that many organizations were incapable of upgrading to the current IBM releases that fixed the problems due to these patches! Even though mainframe releases come out infrequently, many customers were 3-6 versions behind due to their own inability to move their changes forward.

There is a big lesson here for open source proponents that modify the software to suit their needs without a very disciplined configuration management and test process. You are implicitly digging your own legacy grave!

### **OSS Demands Best Practices in Configuration Management**

The combination of large amounts of C/C++ code, Java libraries and a plethora of different utilities and tools makes it imperative that IT have best practices for OSS code management.

There are substantial challenges ahead for those who seek to modify and enhance open source frameworks from different suppliers. Open Source projects vary widely with respect to their practices for documenting, testing, release to release compatibility, and migration. Current frameworks each have their own component models, support libraries etc. Hence integration and support of different frameworks is challenging at best.

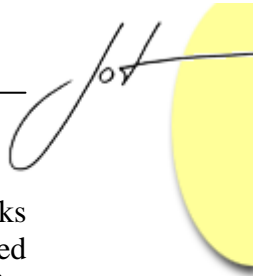
Most OS projects use RCS like CM tooling such as CVS, while adequate for development they were not designed to build field upgradeable products. The patch tools again are effective but CM and patching a large number of IT servers or desktops requires a lot of arcane Linux technical expertise. In particular recompiling from source or rebuilding the system from binaries isn't a scalable solution. Modern Linux distributions provide excellent upgrade and package management tools though however they require the same source/binary tree as the vendor.

In order to embrace open source the organization must become proficient with the various operating system and framework development and configuration tools. A wise organization will invest in fine-grained comparison and versioning tools to carefully track changes and compare new versions of frameworks to determine the impact of upgrading to a future release.

Organizations that develop software for other parties will need to be particularly stringent with respect to their use of OSS code. They will need to have each developer authenticate the authorship of their code and its release under the appropriate license.

## **SUMMARY**

The widely articulated benefits of OSS provide clear options for IT organization to achieve flexibility and cost reduction. However, like all good things in the real world there is no free lunch so one must put in the place the appropriate internal and supplier peopleware and tooling to obtain the substantial benefits of open source software.



---

Clearly the industry is maturing and with consolidation and standardization the risks will continue to be reduced, but so will some of the opportunities and benefits associated with diversity of suppliers. For most IT organizations it seems ideally suited for infrastructure and tool areas supported through traditional support agreements with either traditional or new OSS firm.

### About the author



**Dave Thomas** is CEO of Bedarra Corp., Adjunct Professor at Carleton University, Canada and University of Queensland, Australia, founding Director of AgileAlliance.com, and founder of Object Technology International. Bedarra works with research labs and commercial partners to transition innovations into products and practices.