ANARCHY IN A SUIT AND TIE: OPEN SOURCE SOFTWARE IN THE ENTERPRISE

Leanne Powers, College of Business Administration, California State Polytechnic University, Pomona,
*3801 W. Temple Ave., Pomona, CA 91768, 909-869-4216, lpowers@csupomona.ed*u
Rhonda Rhodes, College of Business Administration, California State Polytechnic University, Pomona,
*3801 W. Temple Ave., Pomona, CA 91768, 909-869-2454, rrhodes@csupomona.ed*u

INTRODUCTION

Although often associated with anarchic fringe groups in computing, the software phenomenon known as “open source” may pose intriguing opportunities for use in commercial enterprises. Although its development and distribution model seems to defy the conventions of market economics, open source software has nonetheless been gaining worldwide popularity, both in public and in for-profit organizations. This paper surveys the relevant literature and discusses the use of open source software as a means of competitive advantage in commercial firms. Specifically, this research addresses the following questions:

1 Are there potential financial advantages to using open source software in business environments?
2 Can the use of open source software provide operational efficiencies and/or security advantages superior to those from available proprietary software?
3 What enterprises might benefit from using open source software?
4 How can companies make use of available open source tools?
5 What special challenges and risks are associated with open source software use, and how can managers improve the probability of successful implementation?

This research is important because it suggests potential solutions to profound concerns related to information technology infrastructure. Licensing expenditures in businesses have been on a trend of increase, from an average of 9% of total IT budgets in 1998 to about 20% in 2004 (Gomolski, 2004). Security of enterprise and customer data have also become a concern of grave importance; incidents of reported system compromise increased exponentially over the same period, from less than four thousand in 1998 to well in the hundreds of thousands when CERT stopped publishing these statistics in 2004.[1] Armed with a wider range of alternatives for addressing these challenges, managers can better evaluate which infrastructure solutions are the best fit for their organizations' needs.

OVERVIEW AND DEVELOPMENT OF OPEN SOURCE SOFTWARE

In its simplest terms, “open source” refers to a model of software development and distribution that expressly places the software and its source code in the public domain. First conceived of by Richard Stallman as a means of keeping software that had been developed in open and collaborative (typically academic) projects from being converted to proprietary use, open source software is distributed under a specific set of terms known as the Generic Public License, or GPL, that prohibits developers from asserting copyright claims on modified versions. Open source systems and applications are developed, tested, and improved by a worldwide community of developers numbering in the hundreds of thousands

[1] Carnegie Mellon Software Engineering Institute CERT Coordination Center. http://www.cert.org/stats/cert_stats.html

and spanning stakeholder groups from academia, to enterprise, to computer hobbyists (Bonaccorsi and Rossi, 2003). Quality control is addressed by the inherent characteristics of the open source community, to which software is released early and frequently for review. Error reporting systems such as Red Hat's Bugzilla[2] provide a structured means of peer review, helping development teams find and fix errors far more rapidly than the developers of proprietary software. This unique development paradigm has contributed to a reputation for excellent quality, stability, and security among open source software systems and applications (Paulson, 2004).

CURRENTLY-AVAILABLE OPEN SOURCE SOFTWARE

Beyond operating systems, there exists an open source equivalent for nearly every commonly-used proprietary computing application. Figure 1 outlines an abbreviated summary of popular applications and their open source equivalents:

Function	*Proprietary application*	*Open source equivalent*
Web browsing	MS Internet Explorer	Mozilla Firefox
Word processing	MS Word	OpenOffice.org Writer
Spreadsheet	MS Excel	OpenOffice.org Calc
Presentation Aids	MS Powerpoint	OpenOffice.org Impress
E-mail	MS Outlook	Kmail, MozillaMail
Photo editing	Adobe Photoshop	The Gimp
Desktop Window Management	MS Windows	Gnome, KDE, X
Personal Organization	MS Outlook	KOrganizer
Mail Server	MS Exchange	Sendmail
Web Server	MS Windows NT	Apache
Database Server	MS SQL Server	MySQL, PostgreSQL, Firebird
ERP/CRM	Oracle, Peoplesoft	Compiere

Fig. 1 – Open source application equivalents Source: Authors

These open source tools have been successfully deployed in computing environments ranging from educational institutions (Harrisonburg City Public Schools, Oregon State University, University of Illinois), government entities (National Weather Service, numerous municipal governments), and corporations (Goodyear, HealthFirst, Inc., Ernie Ball Guitars). Enterprise managers who switch to open source computing platforms typically cite cost advantages, reliability, and vendor independence as factors influencing their decision to choose an open source software solution (Bansal, 2004).

Although published case studies typically concern themselves with open-source as an overall enterprise computing platform, open source software is also commonly used in specific computing applications. Perhaps the most common is in web server architecture; nearly 70% of Internet web servers are powered by Apache server software.[3] Another growing trend in computing involves the use of open-source operating software, such as Linux, as the basic infrastructure in enterprise computing environments, which is then integrated with a proprietary ERP or CRM system. An example of this integration is Oracle's support for the Linux operating platform in its line of ERP applications. Other examples of

[2] Red Hat, Inc. is a leading developer/distributor firm for enterprise Linux. Red Hat's Bugzilla system can be accessed at https://bugzilla.redhat.com/bugzilla/

[3] Source: Netcraft.com Site Survey March, 2005.

proprietary software that has been ported to open source operating systems include Corel, Lotus, and Novell exteNd.

POTENTIAL BENEFITS FROM THE USE OF OPEN SOURCE SOFTWARE

Benefits attributed to enterprise use of open source software typically focus on cost, modularity, security, and vendor independence. Martin Fink (2002) outlines several of the most oft-cited benefits to the Linux operating system, many of which can be applied to other open source software applications:

- *Cost* – That open source software can be freely copied, modified, and distributed makes it a powerful contender for multi-user computing environments where the license fee structure of proprietary software becomes cost-prohibitive. The up-front cost savings associated with end-user desktops may make open source particularly attractive to start-up enterprises (Wall, 2001), while reduction or elimination of recurrent licensing/maintenance fees, especially for businesses who are out of compliance or in need of upgrades, improves the value proposition for larger enterprises.
- *Wide availability of development and implementation resources* – The cost advantages of Linux and other open source infrastructure has led many universities to pursue aggressive implementation strategies. One by-product of this phenomenon is a wealth of human capital, well-trained in development and implementation of these tools. Additionally, the global nature of open source development provides ample outsourcing opportunities.
- *Options for support* – With the entry of major IT firms such as Novell and IBM into the open source arena, strategic vendor support for open source tools is becoming more and more widely available. The open source model also allows companies to develop self-support strategies that would be impossible without access to the software's source code and development libraries (Simonyi, 2000).
- *Control and vendor independence* – The Generic Public License used in open source software allows an enterprise to choose—and change—software and support vendors based strictly on the value the vendor provides.

In addition to the factors cited by Fink, four other benefits appear regularly in the literature:

- *Security* – In general, open source tools have earned a better performance record on measures related to security (CERT, 2005). This is the likely result of a combination of factors; first, the open source community's peer review process is cited as a contributor to the fast detection and repair of errors that may contribute to security vulnerabilities (Paulson, 2004). Additionally, the overwhelming dominance of Windows-based tools in both enterprise and consumer computing environments appears to result in the vast majority of malware (viruses, worms, spyware) being developed to exploit vulnerabilities on Windows machines.
- *Stability* – Open source platforms, and Linux in particular, have excellent reputations for stability (Buck, 2005). While Microsoft has made great improvements in Windows 2000 and Windows XP, a stability edge in Linux may still make it a better candidate for server applications.
- *Reduced upgrade costs for both software and hardware* – Open source platforms enjoy long life-cycles, and upgrades are rarely driven by compatibility issues out of the enterprise's control. Open source operating systems are also ideally-suited for use on older hardware that cannot run the current closed-source operating systems.
- *Reduced costs for custom application development* – Freely available open source modules and libraries may help speed the pace of custom application development by allowing developers to incorporate existing code into custom applications.

POTENTIAL CHALLENGES POSED BY OPEN SOURCE SOFTWARE

With every novel trend in enterprise computing comes potential challenges associated with its use. Some of the most commonly-voiced concerns with open source are:

- *Limited support for open-source platforms among top-tier hardware manufacturers* **–** Desktop adoption of open source operating systems can be frustrated by limited options to purchase systems with the OS pre-installed. In many cases, the configuration of a system with Linux as the primary operating system actually costs more (Buck, 2005). Finally, although hardware compatibility with available open source drivers is excellent, it is NOT universal.
- *Difficulties in installing and using open source tools* **–** while trained support for Linux and other open source software is widely available, installing and learning open source applications can be intimidating to the novice. Small businesses hoping for cost advantages from open source can find these "soft" costs to be substantial in operating system migrations when experienced technical support is not used in deployment (Yeoh, 2004).
- *Application availability* – While Linux and FreeBSD are maturing as desktop platforms, the availability of specialized applications still lags well behind that of Windows (Buck, 2005).
- *Compatibility difficulties* – Although many open source applications offer functionality that rivals their closed-source equivalents, cross-platform compatibility often remains problematic, particularly in enterprises that attempt implementation at the departmental level. Additionally, vendor lock-in resulting from the use of legacy systems or combinations of proprietary applications may further frustrate adoption of open source tools.
- *End-user resistance* – Transition of desktop end-users to a new computing environment can be inherently problematic unless great care is taken to ensure that users are properly trained and understand the business benefits enabled by a switch from more familiar software.
- *Failure to fulfill cost management objectives* – Although open source products can usually be obtained for little or no initial expense, it is important to note that the notion of "free" software doesn't mean there is no associated cost of ownership. Training, for example, can quickly drive up the cost of open source solutions.

In light of these challenges, a realistic assessment of any open source deployment requires managers to develop clear business objectives to be enabled by the use of these tools, a careful analysis of all the associated costs, and a detailed implementation plan.

EVALUATING AND IMPLEMENTING OPEN SOURCE ALTERNATIVES

If potential business benefits to the use of open source have been identified, the task shifts to determining whether—and which—open source products may benefit your organization. Business issues to be considered in an evaluation of open source tools include:

- *Current and evolving IT architectures* – As Microsoft is ending support for Windows NT servers and for Exchange Server 5.5, many organizations are faced with the prospect of critical server upgrades. Migration of NT servers to Windows Server 2003 also requires a conversion of the directory architecture from the domain-like structure used on NT to Active Directory. However, migration to Linux allows a choice between preservation of the domain architecture, or different options altogether, such as OpenLDAP directory (Novell, 2005).
- *Application dependencies* – If business units are dependent on specialized Windows-based applications that lack open source equivalents, platform migration is usually much more problematic.

- *Interoperability* – As more and more business applications are being ported to open source platforms, interoperability concerns are becoming less troublesome for firms wishing to make the switch. Nevertheless, file and application compatibility is a critical factor in open source evaluation.
- *End-user acceptance* – The impact of IT infrastructure changes on end users may constitute one of the largest sources of "soft" costs and business risk. Thus, end-user acceptance must be a primary decision factor in any open source evaluation.

For purposes of this assessment, three different deployment scenarios will be considered, each of which can be evaluated independently. These scenarios and their associated risks are shown in Figure 2:

1. Deployment of open source software on back-end servers, including web, mail, database, application, print, and file storage servers
2. Deployment of selected open source applications, such as web browsers or productivity tools, on existing desktop platforms
3. Deployment of open source operating systems on end-user desktop systems

	Relative potential for	**Relative potential for**	**Business risk:**	**Business risk:**
Deployment Scenario	**cost improvements**	**security improvements**	**Technical**	**End user acceptance**
Implementation of open source software and operating systems on back-end servers	Moderate to high	High	Moderate to high	Low
Implementation of selected open source desktop applications over existing operating systems	Moderate	Moderate to high	Low	Low to moderate
Implementation of open source operating systems on end-user desktops	Moderate to high	High	Moderate	Moderate to high

Fig. 2 – Deployment scenarios and relative business risks Source: Authors

DEPLOYING OPEN SOURCE SOLUTIONS

Once a decision has been made to implement open source in the enterprise, a deployment plan must be determined. The specifics of that plan depend on the deployment scenario chosen, the existing IT infrastructure, and the nature of the open source tools to be included. However, a general overview of the process will help managers understand the logistics of acquiring, installing, and migration to open source:

Acquiring open source software

The vast majority of open source software can be obtained for little (or no) cost, either by downloading a copy from the Internet or purchasing it on a disk. The most common open source operating platform, Linux, is bundled into "distributions" that include the operating kernel, common hardware drivers, one or two desktop environments, and a suite of applications that can be configured based on the system's intended use. According to Distrowatch.com, there are nearly 350 active distributions of Linux and eight distributions of BSD (an open source operating platform similar to Unix and Linux) currently in use. Although all Linux distributions are based on the Linux kernel, they vary widely in their inclusion of application packages and configuration tools, making the installation and management process different for each. One of the most popular, Red Hat Enterprise, is a commercial product oriented toward enterprise applications and supported by a wide range of services from the vendor. Other distinguishing features of certain distributions include differences in their resource (storage and memory) use, development communities, general user-friendliness, and overall stability.

Implementation concerns

Implementing open source in server applications can vary widely in cost and difficulty, based primarily on the intended use and the IT architecture being migrated from. While most databases can be migrated to Linux or BSD with reasonable ease, some legacy applications will require extensive and potentially risky data conversion. E-mail servers—often an ideal candidate for migration due to the excellent security of most open source products—can be particularly troublesome if an organization intends to migrate from a mail server with tight application integration, such as Lotus Notes. Training of IT staff for server or desktop deployments depends heavily on the prior IT infrastructure; companies migrating from Unix-based systems usually do so with little difficulty, while those who switch from Windows may need to invest in extensive staff retraining. The simplest implementation scenario—using open source applications over existing desktop operating systems—may nonetheless reap impressive rewards in cost and security improvements. While reliable statistics related to application security are difficult to obtain and interpret, most of the literature surveyed in this project cited Microsoft's Internet Explorer web browser, Outlook e-mail client, and Office productivity software as exploited far more often than their open source counterparts. Even adjusting for the relative popularity of each platform, individual Windows systems have been identified as having a much higher probability of being exploited by malware such as viruses, internet worms, and spyware. While many attribute the prevalence of Windows infections to the substantially higher payoff of writing malware for Windows as opposed to less-popular systems, the flip side of this relationship is that users of the less-popular applications enjoy relative freedom from compromise as a result (Locke, 2004). Alternatives to Internet Explorer and Outlook, such as the Mozilla Firefox browser and Thunderbird e-mail client, are readily available and free to end users. OpenOffice.org, an alternative to Microsoft's Office suite, is both free and may also be used on Windows systems.

CONCLUSION AND RECOMMENDATIONS

Despite the assertions from parties on both sides of the open source debate, the use of open source software carries a strong value proposition for many—though not all—enterprises who wish to improve return on their IT investments. When appropriately evaluated and implemented, open source solutions can reduce IT costs, improve security, and leverage the talents of the open source community to the organization's advantage. However, the success of any open source initiative depends on the careful and accurate analysis of a wide range of business, technical, and human considerations.

Organizations wishing to employ open source software as a means of achieving business objectives should take a methodical approach to the evaluation and implementation of these tools. The following general guidelines should be part of any open source IT initiative:

1. Begin the process by determining which business objectives might be enabled by the use of open source solutions and how progress will be measured. Implementing vast changes to IT architecture without setting clear goals may well turn the project into a pointless and frustrating exercise.
2. Carefully evaluate the suitability of open source for this enterprise, based on a combination of factors that include:

- appropriateness to the business need
- acceptable value for cost, particularly with respect to “soft” costs
- migration risks in the intended user environment (server, desktop, etc)
- end-user acceptance

3. Develop a clear and reasonable implementation plan
4. Treat the alignment of the open source initiative with business strategies and a high degree of organizational support to be project requirements in the IT implementation.

REFERENCES

[1] Bansall, P (2004). Change for the better. *The Banker*. July 2, 2004, p.14.

[2] Bonaccorsi, A., and Rossi, C. (2003) Why open source software can succeed. *Research Policy.* Amsterdam Jul 2003 32: 7, p. 1243-1258.

[3] Buck, W. M. (2005). Corporate desktop Linux: The hard truth. *O'Reilly OSDir,* February 4, 2005. Retrieved March 17, 2005 from http://osdir.com/Article3992.phtml.

[4] Fink, M. (2003). *The business and economics of Linux and open source.* Indianapolis: Prentice Hall.

[5] Gomolski, B. (2004). *Budget 2005: Win some, lose some.* ComputerWorld, September 20, 2004.

[6] Lineweaver, R. (2003). *Cost savings of open source software in the server room: An informal case study in K-12 education.* Retrieved March 2, 2005 from http://staff.harrisonburg.k12.va.us/~rlineweaver/

[7] Locke, John (2004). *Open source solutions for small business problems.* Hingham, MA: Charles River Media.

[8] Lyman, J. (2004). Open source on offense in ERP and business applications market. NewsForge. June 3, 2004. Retrieved March 1, 2005 from http://software.newsforge.com/article.pl?sid=04/05/22/193215.

[9] Newman, N (1999). *The origins and future of open source software: A NetAction white paper.* Retrieved March 5, 2005 from http://www.netaction.org/opensrc/future/

[10] Open Source Initiative (2005). J*obs for hackers: Yes, you CAN eat open source.* Retrieved March 3, 2005, from http://www.opensource.org/advocacy/jobs.php.

[11] Open Source Development Labs, Inc. (2005). OSDL Higher education OSS projects. Retrieved March 2, 2005 from http://groups.osdl.org/forums/higher_ed_projects/

[12] Paulson, J. (2004). An empirical study of open-source and closed-source software products. *IEEE Transactions on Software Engineering.* 30:4, p. 246-256.

[13] Raymond, E. (2001). *The Cathedral and the Bazaar: Musings on Linux and open source by an accidental revolutionary.* Cambridge: O'Reilly and Associates.

[14] Simonyi, M (2000). Linux: Alternative directions in computing. *Information Systems Management,* 17:2, Spring, 2000.

[15] Stafford, J (2004) Migration in-depth: Why Health First is dumping MS Office. SearchEnterpriseLinux.com, May 17, 2004. Retrieved March 1, 2005 from http://searchenterpriselinux.techtarget.com/qna/0,289202,sid39_gci964811,00.html

[16] Stallman, R. (1999). The GNU operating system and the free software movement. *Open sources: Voices from the open source revolution.* Cambridge: O'Reilly and Associates.

[17] Walker, T. (2004). The future of open source in government. gOSapps, LLC. Retreived March 2, 2004 from http://oss-institute.org/newspdf/ walker_oss_white_paper_2292004.pdf.

[18] Wall, D. (2001). Using open-source for a profitable startup. *Computer.* December, 2001.

[19] Yeoh, O. (2004). Internet time: The pros and cons of open source. *The Edge Malaysia,* August 16, 2004.