Perceptions of Open Source Versus Commercial Software: Is Higher Education Still on the Fence?

Shahron Williams van Rooij
Datatel, Inc.

Abstract

This exploratory study investigated the perceptions of technology and academic decision-makers about open source benefits and risks versus commercial software applications. The study also explored reactions to a concept for outsourcing campus-wide deployment and maintenance of open source. Data collected from telephone interviews were analyzed, emergent themes identified, and a model of differentiators of open source versus commercial software was created, which was then used to evaluate reactions to the outsourcing concept. Interviews revealed perceived barriers to open source adoption and the extent to which the outsourcing concept could alleviate risks. Recommendations for overcoming adoption barriers are offered and future research opportunities identified to ensure that open source software applications are both technically efficient and supportive of engaged learning. (Keywords: open source, adoption, outsourcing, benefits vs. risks, software perceptions.)

INTRODUCTION

As institutions of higher education try to reconcile tight funding with the rising costs of technology, some institutions are turning to open source—software delivered with its computer program source code—for campus-wide applications such as course management systems and administrative systems. With access to source code, developers can modify the software to meet the needs of the institution, save the license fees charged by commercial vendors, and provide the institution with the flexibility to build learning environments that are both pedagogically sound and technically efficient (Pavlicek, 2000; Weber, 2004; Williams, 2002). The rollout of Sakai, an open source platform for teaching, learning, and research collaboration (Sakai, n.d.), the endorsement of the Kuali open source financial management system by the National Association of College and University Business Officers (Kuali, n.d.), and the growing number of institutions worldwide adopting the Moodle open source course management system, have all contributed to the higher education "buzz" surrounding open source.

Theoretical Framework: A Cross-Disciplinary Perspective

The chief technology officer or chief information officer (CTO/CIO) is responsible for the information technology infrastructure of the institution and is the primary decision-maker when deciding how and when to acquire the hardware and software that will be used across all departments and divisions of the institution (Green, 2004). This includes the enterprise-wide administrative software applications that automate the business functions of the institution and course management systems (CMS), which offer faculty and students

Web tools to complement or replace classroom seat time. Typically, a CMS will include curriculum design tools and templates, automated testing and scoring tools, student tracking, calendars, and a variety of tools (discussion forums, chat, whiteboard, etc.) for faculty-to-student and student-to-student communication (Gibbons, 2005). The CIO also makes the decision to use open source versus proprietary vendor products.

Recently, the chief academic officer (CAO) has become an equal partner with the CIO with respect to the selection of enterprise-wide software applications for teaching and learning (Green, 2004). As the administrator responsible for the institution's instruction and research affairs, the CAO assesses the extent to which a particular technology meets or does not meet pedagogical needs. Consequently, any exploration of open source and the learning environment requires drawing upon two bodies of literature—software engineering and education—to determine the theoretical framework in which such an exploration would take place.

The Software Engineering Literature

Open source advocates point to an extensive body of research in the field of software engineering that explores the benefits and risks of open source in the context of (a) social movement theory and appeals to the common good (Coleman, 2004; Franck & Jungwirth, 2003; Kelty, 2004; O'Mahoney, 2002; Perens, 1999), (b) a new paradigm in software development methodology, where developers participate without monetary compensation (Evans, 2002; Gacek, Lawrie, & Arief, n.d.; Raymond, 2001; Scacchi, 2001; Stewart & Gosain, 2004; Von Krogh, 2003), and (c) security and risk management (Raymond, 2001; Stallman, 1999; Weber, 2004). Other conceptual frameworks in the literature include the Diffusion of Innovations theory first developed by Rogers (1995), applied to the adoption of technology by Moore (1991, 2005), then applied to the adoption of open source software, with technical skills as a critical barrier to adoption (Attewell, 1992; Evans, 2002; James, n.d.; Linden & Fenn, 2003). Organizational know-how and ability to respond to innovation has also been the basis for framing open source adoption (Au & Kaufmann, 2003; Overby, Bharadwaj, & Bharadwah, n.d.), while public policy think tanks have published monographs about U.S. government policy toward open source (Hahn, 2002; PITAC, 2000).

Higher education technologists and financial administrators —particularly those in the doctoral/research institutions—are vocal advocates of the efficiencies of open source. For example, in 2004, the National Association of College and University Business Officers (NACUBO) conducted a survey of its members about their perceptions of open source software. Nearly half (46%) see open source as a viable alternative to commercial software systems, with the primary reasons being "open standards and interoperability with other application systems" (61%); "freedom to modify the code" (58%); "software designed by and for the industry" (58%), and; "lower cost of ownership" (55%). Although the study's author (Hignite, 2004) acknowledges that these results are based on the responses of less than 5% (n=257) of the NACUBO membership,

the buzz surrounding the survey results were enough to spur the organization to apply for a grant to explore the feasibility of an open source financial system.

In the July 2006 issue of the trade publication *Campus Technology*, the CIO of a private masters institution with a higher education publication (HEP) enrollment of 3,500 purports to have saved 20% in annual maintenance costs by switching from vendor products to open source software (Villano, 2006). What is not stated is what type of open source software (infrastructure level vs. application level) the institution uses or the number and skill set of the institution's programmers. In a recent survey of 195 CIOs conducted by the IMS Global Learning Consortium (Abel, 2006), more than half (57%) of the respondents purport to have seen cost-of-ownership improvements since replacing vendor products with open source. However, the applications adopted include a mix of desktop (e.g., MyOffice), user interface (e.g., uPortal), and teaching/learning applications (e.g., Moodle), all with adoption rates of $\leq 24\%$ of the total survey sample.

The software engineering literature is unanimous in deeming open source software to be for technologists; the overarching assumption is that the technologist IS the end user. However, this is not the case for open source applications intended to support teaching and learning. Moreover, the software engineering literature offers no insights on the extent to which open source enables the incorporation of sound pedagogy into the construction of the learning environment.

The Education Literature

International education has been quick to research and adopt open source software for the development of instructional content and delivery systems. Researchers in the UK note that education already has a long-established legacy of shared responsibility for projects, peer review and distributed development, and cite projects aimed at open source development of educational materials with a centralized repository of freely-available, searchable quality controlled materials (Carmichael & Honour, 2000; Hirst, 2001). There are also studies that provide European examples of open source course management systems that incorporate sound pedagogy (Leinonen, T., Hakkarainen, K., Appelt, W., Dean, P., Gomez-Skarmetav, A., Ligorio, B., et al., 2001; Dunlap & Wilson, 2002; Jasinski, n.d.).

In the U.S., the collaborative development and sharing of instructional content is not new. The TLT Group (TLT, n.d.) has been offering online assessment tools, Web page templates and tutorials to institutions on a low-cost subscription basis since 1992. MERLOT is one of the longest-running Web sites providing free materials from faculty for faculty in a variety of disciplines (MERLOT, n.d.). MIT's Open Course Ware Web site is a more recent example of free access to instructional materials. The materials are targeted to educators and learners worldwide, with the biggest market outside the United States and the primary use being the enhancement of personal knowledge (Carson, 2002). It is not yet clear to what extent these materials have been adopted by U.S. learners, institutions, and faculty.

Incorporating technology into instruction has meant re-examining the nature of the environment in which students learn, including the setting or "space" in which learning is fostered and supported (Wilson, 1996). Perkins (1992) parses a learning environment into five components: (a) information banks that include information repositories of all types, ranging from textbooks to digital media to faculty; (b) symbol pads or surfaces for manipulating symbols and languages and that include notebooks, word processors, database programs, and the like; (c) construction kits or packaged collections of content components for assembly and manipulation. Examples would include Legos or digital authoring tools; (d) phenomenaria or areas for presenting, observing, and manipulating phenomena, such as the well-known SimCity; and (e) task managers, the elements of the learning environment that set tasks, provide guidance, feedback, and changes in direction. This component has traditionally been the faculty member, but also includes electronic task managers and intelligent tutor computer-based programs.

The interplay of these components is part of the constructivist perspective on learning. Constructivism holds that learners build their own personal interpretation of the world based on their own experiences. Because there are many ways of structuring the world and its entities—i.e., there are multiple perspectives—learners make their own meaning rather than having it imposed from some external, independent reality (Duffy & Jonassen, 1992). Thus, the pedagogical goals of a learning environment grounded in constructivism must provide these multiple perspectives and enable learners to make their own meaning by providing multiple modes of representation (Honebein, 1996).

Technology's potential lies in the extent to which it enables the creation of a learning environment grounded in constructivism (Domine, 2006). There is evidence that commercially developed course management systems can accommodate a wide variety of learning activities and perspectives. Studies of individual courses using WebCT or Blackboard have found that those technologies augmented the learning experience by enabling collaboration, the development of a strong sense of community, and the inclusion of constructivist strategies of collaborative learning into the instructional environment (DeNeui & Dodge, 2006; Gill, 2006; Iyer, 2003). In a recent review of the literature on the use of course management systems in higher education, Papastergiou (2006) states that although students and faculty have positive attitudes toward CMS-based learning and that faculty can apply participatory, constructivist approaches to learning in CMS environments, increased faculty workload, limited assessment capabilities, the inability to support subjects that involve hands-on tasks, and the need for more sophisticated collaborative facilities beyond the commonly-offered discussion boards are all weaknesses in commercial course management systems. However, the education literature has not yet systematically addressed the extent to which open source enables sound online pedagogy.

A New Approach

Figure 1 illustrates the themes from the software engineering literature and the education literature. What is clear is the gap between what the technologists

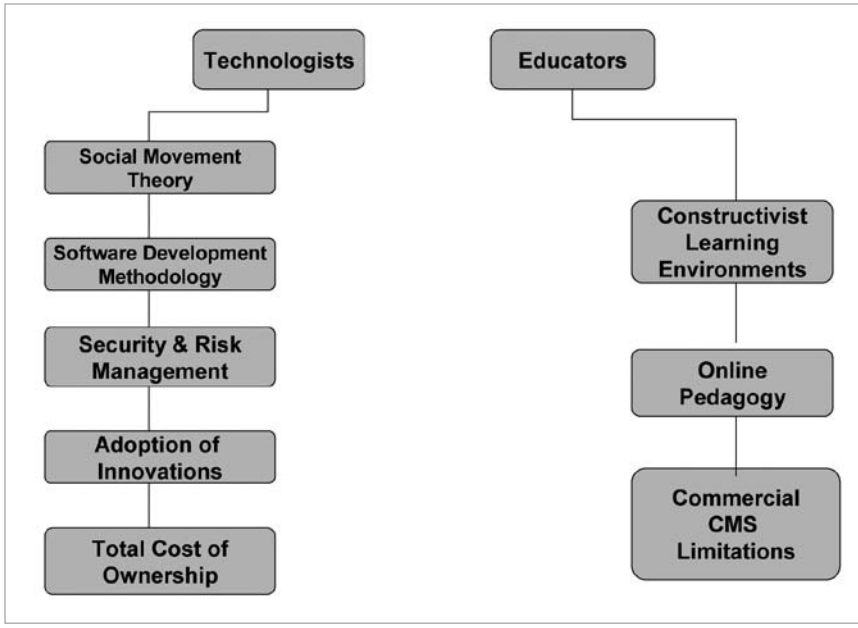

Figure 1. Open source theoretical framework

need/want and what academics need and want for teaching and for scholarly work. Consequently, there is a clear need for systematic analysis of the benefits and risks of open source software from the perspective of the CIO and from the perspective of the CAO. The more we understand the similarities and differences in perceptions of these two decision-makers, the more complete the picture of the impact of open source on both teaching and learning and technical efficiencies. Moreover, there is a need to better understand the extent to which commercial organizations can assist institutions interested in adopting open source, so that alternative cost of ownership models can be developed and institutions can make informed decisions about where to use their resources.

RESEARCH QUESTIONS

Two research questions guided this study:

- What are the characteristics that CIOs and CAOs believe differentiate open source software from commercially developed software?
- When presented with a concept for outsourcing the integration and maintenance of campus-wide open source academic systems, how interested would CIOs and CAOs be in purchasing this service and at what price?

METHOD

Participants and Setting

A pool of 45 individuals representing a variety of Carnegie classifications and institution sizes had volunteered to participate in this exploratory study

Table 1: Distribution of In-depth Interviews by Carnegie Classification

Carnegie Classification	CAOs	CIOs
Doctoral/Research	0	3
Masters	4	4
Baccalaureate	2	0
Associates	3	3
Specialty	1	0
Total Number of Completed Interviews:	**10**	**10**

by filling in their particulars in a call-for-volunteers question on a survey of open source adoption practices this author conducted in the winter of 2006. Of this pool of 45, 10 CIOs and 10 CAOs were interviewed, for a total of 20 completed in-depth interviews. Participants were selected based on Carnegie classification and institution size, to get as broad a representation as possible. Table 1 shows the distribution of interviews by Carnegie classification.

Procedure

Each participant was contacted by e-mail and offered a choice of dates and times for the interview. Once the participant stated his/her availability for an interview, an e-mail confirmation containing an informed consent form was e-mailed to the respondent, with a request to reply to the confirmation to indicate consent. The interviews took place via phone during business hours. The tape-recorded interviews averaged 20 to 30 minutes in length.

Validity and Reliability

The validity threats associated with qualitative research (Maxwell, 1996), along with the measures used in this study to address those threats are as follows:

- Valid description or inaccurate/incomplete data. This threat was addressed by the audio recording and verbatim transcription of the interviews.
- Researcher bias or the imposition of the researcher's own framework or meaning on those of the participants. This was addressed during the interview by corroborating what the researcher thought was said with the participant.
- Theoretical validity threatened by not collecting/considering discrepant data/explanations/understandings. Documentation of each step of the analysis using the NVivo software program automatically developed a log of code constructions and proposition testing. Documentation is a method of improving the immediate analysis task being carried out, enhancing the sophistication of later analyses, and strengthening confidence in the final conclusions (Miles & Huberman, 1994).

This study sought to flesh out general constructs, patterns, and themes to obtain insights into the mindset of the participants and better understand some of the ongoing conversations around open source adoption in higher education.

As such, the outcomes are not representative of the entire population of higher education institutions.

Data Analysis

After the data were collected, the audio tapes transcribed and uploaded into the NVivo software program, a preliminary exploratory analysis (Creswell, 2002) was conducted. This analysis involved reading through all of the transcripts, jotting down ideas and obtaining a general sense of the data. Using the software's visual coding capabilities, words, phrases, and activities that seemed to be similar were grouped into categories or themes. Emergent themes were identified and given tentative names or labels that served as the foundation for preliminary analyses. The software enabled the construction of an audit trail (Hoepfl, 1997) or schema that maps the themes with their speakers and context, so that multiple perspectives on a given theme could be captured and identified. Further analysis enabled themes to be layered and interconnected, to flesh out major and minor themes, and obtain broader as well as deeper levels of abstraction (Creswell, 2002).

RESULTS

Differentiators of Open Source Versus Commercial Software: Context

As a first step to understanding how CAOs and CIOs compare and contrast open source versus commercial software, each interview participant was asked about his or her familiarity with and usage of industry terminology, specifically, the terms "open source," "collaborative development software," and "community source software." All 20 participants were most familiar with the term "open source," volunteering that "open source" meant use by anyone free of license fees, with access to the source code via the Internet, and the ability to share additions/modifications with others.

Participants were then asked about publications, conference events or other resources used to become knowledgeable about open source. Common to both the CAO and CIO participants was the reliance on presentations and publications from the EDUCAUSE organization, with 10 of 20 participants mentioning EDUCAUSE as their primary knowledge source. Beyond EDUCAUSE, both CAO and CIO participants relied on individuals with technical knowledge, including IT faculty, either at their own institutions or at other institutions, as well as local and/or regional higher education consortia and professional associations. CAO participants in particular read articles in *The Chronicle of Higher Education* and *University Business* dealing with open source in higher education. On the whole, however, CAO participants looked to the technologists to keep them informed about the value of open source for enhancing teaching and learning. In explaining the reliance on technologists, a CAO participant at a community college with a HEP enrollment of 2,600 stated succinctly: "I talk to IT. We're all too busy to be doing each other's jobs."

CIO Perceptions

Only three of the 10 CIO participants have already made a conscious decision not to adopt open source. All three of these non-adopters are from private,

masters institutions with HEP enrollments in the 3,000-6,500 range. Two of the non-adopters mentioned the skill set of their technical staff as the reason for non-adoption, while all three non-adopters cited overall satisfaction with their commercial product as a reason for non-adoption.

IT staff skill set remained the key factor when discussing the benefits and risks of open source versus commercial products. All three of the non-adopters voluntarily mentioned the freedom from vendor license fees as the attraction of open source. However, all three also stated that the need to hire programmers to supplement their IT staff would raise the total cost of ownership (TCO) of open source to levels that would probably exceed what they are currently paying to their commercial vendors. To illustrate his point, one non-adopter disputed the idea that open source was a savings, stating:

> You can tell me that open source is free or you can get access to it at nominal fees, but what does it take to maintain it in your environment, to do releases, to do support, to enhance it if you opted to enhance? When I've listened to some of the conference calls and I hear the resources that are being invested by some of the universities to operationalize some of the tools, I don't have any vision that open source comes and it's a freebie. (CIO, masters institution, HEP 4,826)

Institutional resource constraints and skepticism about the true cost of ownership of open source software were not unique to these three non-adopters. The seven other CIO participants also mentioned maintenance and support as a risk associated with open source software. However, these participants remain open to adopting open source software and are currently at various stages ranging from assessment to pilot testing to deployment of specific software applications. What differentiates the undecided participants from the non-adopters are the direct experiences of these participants with vendor price increases and mergers, creating fear, uncertainty, and doubt about the financial and human resource costs of staying with vendors. Typical of undecided CIO comments about the fear, uncertainty and doubt about the financial and human resource costs associated with vendor products versus open source are these comments:

> WebCT is the platform that we're on right now for our learning management or course management system. They've been bought by Blackboard and it seems like that there's a collapsing of the off-the-shelf course software applications out there, so we want to have all options open so we don't just have to pay what Blackboard asks us to pay each and every year. (CIO, masters institution, HEP 2,600)

Lack of support, security, and questionable longevity were the common risk themes. However, CIO participants appeared to view risk management as part of their job. CIO participants also point to functionality gaps in the vendor software, suggesting that CIO's are indeed working collaboratively with their CAOs when evaluating open source alternatives to a commercial CMS. Typical CIO comments about commercial software functionality gaps are these:

> I think budget and instructional flexibility are the two big ones. Over the years our usage of Blackboard has increased by our faculty and students. We went from a basic license to an enterprise license and that escalated our cost by about six times in one fell swoop, and we anticipate that that's gonna get worse since Blackboard and WebCT merged. There are so very few players in the commercial learning management systems space that they really do have the capacity to charge whatever they please. From a faculty standpoint, they find that their ability to develop customized components is greatly enhanced for those people who are interested, so that if you do have a faculty member that has some good programming skill and has an interest in investing their time in doing things that are unique, and then, of course, sharing with the community at large, that has a very attractive feature that most commercial applications don't offer in the same way. (CIO, Associates institution, HEP 15,655)

CAO Perceptions

When asked about the business issues driving their institution's consideration of open source, the CAO participants voluntarily mentioned the financial and human resource costs associated with commercial products. However, the CAOs appear to be looking at costs more from a value perspective—i.e., the functionality received versus the dollars spent—rather than at the total cost of ownership perspective of their CIO counterparts. Typical CAO comments are as follows:

> Probably the main thing that drives that—there are really two. One is the cost management. The problems that we're having with Blackboard/WebCT right now across the state are very rapidly escalating problems. We don't see open source as a cost savings. It just means that the costs that we put in can be directed to things that we need to see done with the applications. You know, right now our costs for Blackboard/WebCT go up and we're not seeing a whole lot of attention to what we perceive as needs at my local campus or across the university system. (CAO, masters institution, HEP 5,027)

> It's primarily the licensing dollars. And there are some functional reasons that I think we're interested in looking at some different kinds of software because as we go more and more into the online environment, we're bumping up against restrictions of Blackboard, and we're not happy about that. The biggest restriction we have is we want to do videos and we want do conferencing type things. You run into restrictions with Blackboard in those regards. (CAO, masters institution, HEP 4,200)

Perceptions of the pedagogical fit of open source versus commercial systems are well stated by this CAO of a baccalaureate institution with a HEP enrollment of 1,591:

> The benefit for small schools is the ability to customize basically to get what you need and not what you don't. So what's happening now, and whether or not it'll be successful is another question, is that you get places like Mellon and other funding agencies interested in the health of higher education, and also in staying on top of technology in a cost efficient way, that are promoting ways for segments of higher education, notably liberal arts colleges, to work together to get what they need and enhance teaching, and make outcomes demonstrable internally and externally. And you're not really going to do that with Blackboard.

Figure 2 provides a graphic illustration of the themes differentiating open source software from commercial software and the relationships among those themes from the CIO and CAO perspectives. For CIOs, vendor experiences reinforce negative perceptions of the cost of vendor software, which in turn, contributes to a negative perception of the value of commercial software as defined by the software's fit with desired needs/functions. CAOs perceive open source as the counterpoint to that functionality gap, driven by IT faculty knowledge, the availability of funding for projects focused on pedagogy and the opportunity to collaborate with peers. The net benefit is flexibility, with control over one's own destiny as the end benefit. The potential risks posed by security, the lack of support, and quality control contribute to perceptions of questionable project longevity. The threshold that enables perceived benefits to outweigh perceived risks is the knowledge base of the institution's own internal IT staff.

Reactions to an Onshore Outsourcing Concept

In order to flesh out any existing biases toward using external talent, participants were asked if they were using any outside companies or consultants to help with the evaluation or deployment of open source software applications. Although only one participant was currently using a consultant company (to deploy the Sakai learning management platform), six of the 20 participants had experience with consultants helping to deploy their administrative software systems, with mixed results. In all cases, satisfaction with the consultant's work was countered strongly by cost overruns. With reference to using consultants specifically for open source software, 19 of the 20 participants felt that they were not far enough along in the evaluation process to bring in an outside consultant. Instead, they preferred to rely on the knowledge of peers knowledgeable about the subject. This is consistent with comments earlier in the interview when participants mentioned the higher education conferences, presentations and networking contacts used to become knowledgeable about open source.

During the second half of the interview, participants received an e-mail message describing a concept for a package of consulting and training services aimed at assisting institutions in deploying open source software for teaching and learning (see Table 2, p. 444) and were asked if they had seen or heard anything like it. Although no one had seen or heard of a set of services exactly

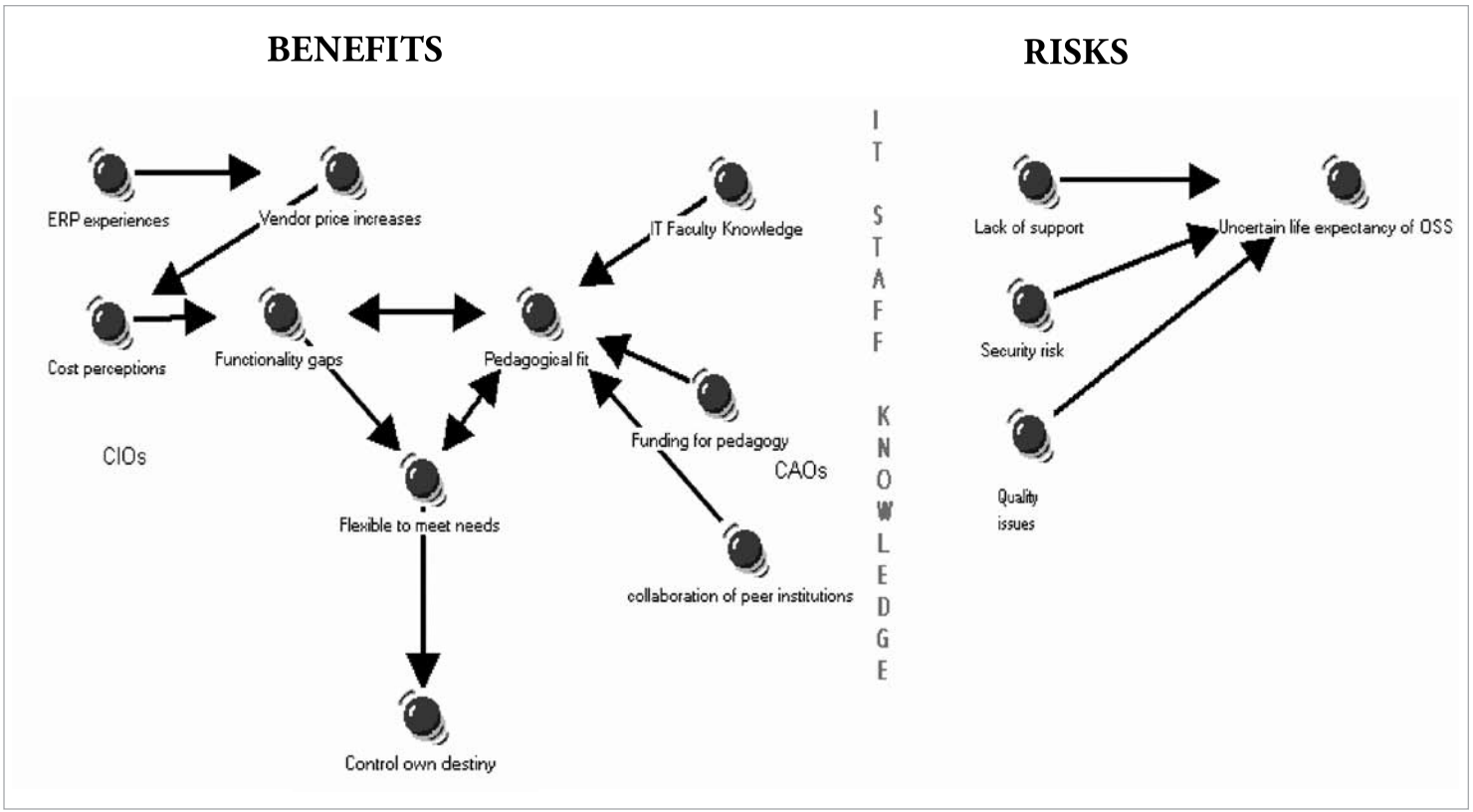

Figure 2. Model of differentiators of open source software versus commercial vendor software.

Table 2: Open Source Services Concept

Concept Statement

Company X's open source management services will provide academic deans, department chairs, and IT staff with a comprehensive suite of services for the development, deployment, and maintenance of open source software applications supporting teaching and learning. The services include:

- Academic and administrative system needs assessment
- Current resource allocation and deployment
- Migration from and/or integration with legacy and vendor systems
- Faculty and IT Help Desk planning
- Education and training for faculty and IT support staff

Company X's open source management services will be differentiated from other professional services on the market today by incorporating both the pedagogical needs of the academic units of the institution with technology goals and objectives of the institution.

The services will be offered for a fixed, predetermined period at a fixed, predetermined cost.

like the ones described in the concept, one CAO participant and one CIO participant stated that they were familiar with similar services, primarily for deploying administrative systems.

When asked whether or not they would be interested in a service like the one described in the concept, 15 of the 20 participants stated that they would be interested. However, the interest was cautious at best. Interest was largely bounded by the option of picking and choosing the services they needed rather than taking all of the services as a package. This "Chinese menu" approach would also help keep down costs, a point of concern when hiring any consulting firm.

The one service desired by all 15 of the interested participants (CAOs and CIOs) was the migration from and/or integration with legacy and vendor systems, primarily the institution's administrative software system. Services desired specifically by the interested CAOs were the academic needs assessment and faculty education and training. The five participants (4 CAOs and 1 CIO) who were not interested in the concept were disinterested for a variety of unrelated reasons.

Regardless of whether or not they were interested in the services described in the concept, most of the participants had a fairly realistic idea of the length of time that would be required for the services. For the CAOs, who viewed faculty education/training, academic needs assessment and systems integration/migration as the most interesting components of the concept package (regardless of whether or not they would actually contract for those services), the estimated length of the service engagement was 12 months or longer. CIOs based their estimates on the time it took to implement their administrative systems, with data migration and systems integration alone taking 6–9 months

Table 3: Reactions to an Onshore Outsourcing Concept Among CAO (N=10) and CIO (N=10) Interview Participants

Reactions	CAOs	CIOs
Interested in Concept	**6**	**9**
Preferred services	Academic and administrative needs assessment	
		Current resource allocation and deployment
	Migration from and/or integration with legacy and vendor systems	**Migration from and/or integration with legacy and vendor systems**
	Education and training for faculty	Education and training for IT support staff
Not interested in concept	**4**	**1**
Reasons for disinterest	Technology decision/CIOs domain	
	Need to first educate senior leadership	
	Philosophically opposed to outsourcing	
		No single organization can do it all
Expected duration of services engagement	≥ 12 months	≥ 12 months, of which 6-9 months is data migration
Expected investment in services	• One-time • $10-$30K (integration) • $100-$500K (full suite)	• Annual • $20K per service

to complete. Only one participant—the CAO of a small (HEP = 289) specialty institution—had no idea how long it would take to implement any of the services described in the concept.

Estimating the cost of the services described in the concept proved to be a bit more challenging for the participants, particularly for the CAOs. CAOs tended to draw on the implementation costs paid to course management/learning management vendors like Blackboard/WebCT as the frame of reference and placed the data migration/systems integration component of the concept in the $10,000–$30,000 range. CAO estimates for the full suite of services described in the concept ranged from $100,000 to $500,000. CIO participants used their administrative software implementation costs and/or previous experiences with external consultants as the frame of reference for estimating the costs of the

services described in the concept. Most were fairly consistent in estimating the costs in terms of an annual investment rather than a one-off charge and most expected to pay $20,000 a year for each of the services described in the concept. Table 3 (p. 445) provides a summary of participant reactions to the concept, including similarities and differences between the CAOs and the CIOs.

Finally, interview participants were given the opportunity to share any additional thoughts and comments about open source software. All of the participants acknowledged that the jury is still out—and probably will be for some time—on the promise of open source in terms of controlling one's own destiny. How far they are willing to test the open source waters has depended thus far on how they view their internal capabilities, both from a technical and a financial perspective.

Summary

For both CAO and CIO participants, commercial software implementation experiences and vendor price perceptions have reinforced negative perceptions about the cost of vendor software which, in turn, has contributed to a negative perception of the value of commercial software in terms of its fit with desired needs and functions. Reinforced by IT faculty knowledge, the availability of funding for projects focused on pedagogy, and the opportunity to collaborate with peers, open source is perceived as the solution to the functionality gap, the end benefit of which is control over one's own destiny. These plusses outweigh the potential deltas posed by security, the lack of support and questionable longevity if (and only if) the knowledge base of the institution's internal IT staff is deemed to be able to meet these challenges. Finally, the outsourcing concept generated a good deal of interest mixed with a healthy dose of skepticism grounded in previous experiences with external consultants and implementation cost overruns. The concept achieved greater acceptance if available as a menu of consulting services rather than as a fixed package. Further, enthusiasm was slightly less among the CAO participants, some of whom perceived the decision about open source consulting services as being the domain of their CIO counterparts. Nevertheless, most of the interview participants were intrigued with the ideas behind the concept.

RECOMMENDATIONS

The results of this study call for a deeper, more comprehensive look at the value of open source for enhancing teaching and learning, as well as for achieving technology efficiencies. One avenue is to maximize the CAO-CIO partnership. Study findings show that CAO participants rely on individuals with technical knowledge—either at their own institutions or at other institutions—as well as presentations and publications from EDUCAUSE, to learn about open source software applications and receive recommendations about the value of open source for enhancing teaching and learning. CAO reliance on technology counterparts is based in the firm belief that it is the CIOs job to inform and recommend. What is not clear is the point in time at which CIOs should begin liaising with their academic counterparts. The case

could be made that CIO evaluation of open source software applications is not yet far enough along to merit bringing their CAO partners to the table. This line of thinking recalls the discussion in the software engineering literature (Behlendorf, 1999; Evans, 2002; Gacek et al., n.d.; Glass, 2003) about the gap between the technologist, who is the end user of infrastructure-level software such as computer operating systems, and the non-technologist, who is the end user of application-level software such as course management systems. In their evaluation of open source software in higher education, Courant and Griffiths (2006) affirm the need for mutual understanding between users and developers about how the software is to be used and what is important for it to accomplish.

If, at the end of the day, it is the technology professional who determines what instructional tools will (not) be supported, how can the institution ensure that technology remains in the service of pedagogy, and not the other way around? To ensure that open source teaching and learning applications support the institution's pedagogical needs, CIOs need to liaise with their academic counterparts long before the deployment phase. Constructivist-based use case scenarios that describe how students interact with faculty, with course materials, with external resources, and with their fellow students, need to be laid out early in the evaluation process. CAOs should have input into the evaluation process timeline, so that the academic calendar and key institutional dates (e.g., registration, final exams, etc.) are taken into account when soliciting CAO, faculty, and student feedback on software pilot tests. If, as CIOs contend, the CAO and the CIO are partners in the process of making decisions around the purchase and deployment of enterprise-wide applications for teaching and learning (Green, 2004), then CIOs need to evaluate open source software applications that impact teaching and learning holistically and include their academic counterparts, rather than making the go/no go decision solely on technical merits first, then looking at building functionality after an adoption decision is made.

Another avenue concerns the cost of open source software. Although the contribution of open source to the student's learning experience is the primary driver for CAO consideration of open source, CAOs voluntarily discussed costs when asked about the business issues underlying consideration of open source software. However, CAO participants were discussing cost in the context of functionality received. To better understand the true cost-value proposition of open source requires institutions that have already fully deployed open source software applications to document and publicize the level of effort spent on open source deployment and maintenance at their institution. From a technology perspective, the number of IT staff supporting each application, the skill set of that staff, salaries and number of hours spent on open source versus basic institution operations (hardware maintenance, etc.), as well on faculty and student training and support, would all have to be reported. This would enable institutions to get a more complete picture of the budgetary impact of open source and whether the savings in software license fees is a true savings. As Villano's (2006) article notes, with only the top 300 U.S. institutions possessing the human resources necessary to implement software and possessing the skills

necessary to manipulate source code, institutions that have deployed open source software applications have had to hire specialists or increase the training budget for existing programmers to build and maintain open source software applications. The result is no savings at all and could mean increased costs.

From a teaching and learning perspective, the level of faculty and/or third party instructional design resource effort versus student learning outcomes would need to be monitored and documented. Students own perceptions of their learning gains and experiences would also have to be tracked and reported. The bottom line: The absence of hard numbers will perpetuate what Green (2004) terms "affirmative ambivalence" towards open source; namely, agreement with the underlying concept, but reluctance to actually adopt and to replace vendor product with open source solutions.

A third avenue concerns the feasibility of using commercial providers to assist in the deployment and maintenance of campus-wide open source software applications, particularly in terms of offering services that support sound pedagogy. Although the study participants liked the idea of being able to pick and choose the type of support services their institution needs to adopt open source software solutions, there is still a gap between the CIO desire to control costs given previous experiences with commercial vendor price increases, implementation cost overruns, and less than stellar experiences with some external consultants on the one hand, and on the other hand, CAO uncertainty as to whether the academic services that they deemed interesting are part of the CIOs decision domain or whether such services should be outsourced at all. This suggests that the availability of open source support services may not provide the ticket to entry that some open source advocates believe. Commercial companies need a financial incentive to provide support for open source software solutions, and that would require a critical mass of institutions contracting for those services (Courant & Griffiths, 2006). For commercial service providers, there is unlikely to be a rush to provide open source support services —technical or pedagogical—until it is clear what price institutions are willing to pay for control over their own destiny or to trade one master for another, as it were.

OPPORTUNITIES FOR FUTURE RESEARCH

One way to address the limited scope of this study would be would be to extend the research to a longitudinal study of open source software awareness, adoption, attitudes, and impact on commercial software usage. Incremental steps in overcoming the adoption barriers identified in this research need to be tracked and the predictors of successful open source implementation and deployment identified, so that institutions can map their own resources, polices, procedures and budget constraints with the "knowns" associated with successful adoption. Future research must also address opportunities for enhancing the CIO–CAO partnership, so that some mechanism(s) for greater and earlier collaboration evolve, ensuring that the open source software applications are both technically efficient and supportive of engaged learning.

Research needs to be conducted in evaluating the long-term effect of the higher education-specific collaborative software development communities like Sakai. The extent to which the volunteer-developers address the quality and functionality issues of the broad base of community members, not just the issues of their home institutions or the community founding members, needs to be studied. Finally, there is an urgent need for research into the documentation and dissemination of proven cost-value models. With resource limitations a given, it is important for institutions to know what the true value of open source solutions will be, so that they can make an informed decision as to where they will invest their resources.

Contributor

Shahron Williams van Rooij is director of product marketing at Datatel, Inc. in Fairfax, Virginia, where she facilitates the engineering of new e-learning technology solutions. She earned a PhD in instructional technology at George Mason University and has five years of college classroom teaching plus a strong background in e-learning. As a Certified Distance Education Professional (C.D.E.P.) credentialed by Texas A&M University, she is a frequent speaker on the development of Web-based education and has published in journals such as the *WebNet Journal* and the *Journal of Interactive Learning Research. (Address:* Shahron Williams van Rooij, Datatel, Inc., 5116 Woodfield Drive, Centreville, VA 20120; 703.502.6745; Fax: 703.968.4573; swvr@cox.net.)

References

Abel, R. (2006). *The state of open source software in higher education: Time for a reality check*. Retrieved September 25, 2006, from the IMS Global Learning Consortium Web site: http://www.imsglobal.org/articles/presentations.cfm.

Attewell, P. (1992). Technology diffusion and organizational learning: The case of business computing. *Organization Science, 3*(1), 1–19.

Au, Y. A., & Kauffman, R. J. (2003, January 6). *Information technology investment and adoption: A rational expectations perspective.* Paper presented at the 36th Hawaii International Conference on System Sciences. Retrieved February 7, 2004, from http://www.hicss.hawaii.edu/HICSSpapers/OSSC105.pdf

Behlendorf, B. (1999). Open source as a business strategy. In C. DiBona, S. Ockman, & M. Stone (Eds.), *Open sources: Voices from the open source revolution* (pp. 149–170). Sebastopol, CA: O'Reilly & Associates.

Carmichael, P., & Honour, L. (2002). Open source as appropriate technology for global education. *International Journal of Educational Development, 22*, 47–53.

Carson, S. (2002). *MIT OpenCourseWare: A new model for open sharing.* Paper presented at the MERLOT International Conference, Costa Mesa, CA. Retrieved August 23, 2004, from http://www.merlot.org.

Coleman, G. (2004). The political agnosticism of free and open source software and the inadvertent politics of contrast. *Anthropological Quarterly, 77*(3), 507–519.

Courant, P. N. & Griffiths, R. J. (2006). *Software and collaboration in higher education: A study of open source software.* Retrieved September 26, 2006, from the EDUCAUSE web site: http://www.educause.edu/LibraryDetailPage/666?ID=CSD4633.

Creswell, J. W. (2002). *Educational research: Planning, conducting and evaluating quantitative and qualitative research.* Upper Saddle River, NJ: Pearson Education, Inc.

DeNeui, D. & Dodge, T. (2006). Asynchronous learning networks and student outcomes: The utility of online learning components in hybrid courses. *Journal of Instructional Psychology*, *33*(4), 256–259.

Domine, V. (2006). Online pedagogy: Beyond digital "chalk and talk." A*cademic Exchange Quarterly*, *10*(1), 48–51.

Duffy, T. M., & Jonassen, D. H. (1992). Constructivism: New implications for instructional technology. In T. M. Duffy (Ed.), *Constructivism and the technology of instruction: A conversation* (pp. 1–16). Hillsdale, NJ: Lawrence Erlbaum Associates

Dunlap, J. C., Young, D. L., & Wilson, B. (2002, June). *Xtreme learning control: Examples of the open source movement's impact on our educational practice in a university setting.* Paper presented at the meeting of the ED-MEDIA 2002 World Conference on Educational Multimedia, Hypermedia & Telecommunications, Denver, CO. Available: http://www.editlib.org/index.cfm

Evans, D. S. (2002). Politics and programming: Government preferences for promoting open source Software. In R. W. Hahn (Ed.), *Government policy toward open source software* (pp. 34–49). Washington, D.C.: Brookings Institution Press.

Franck, E., & Jungwirth, C. (2003). Open source software: New roles for the market economy. *Journal of Management and Governance, 7*(4), 401–421.

Gacek, C., Lawrie, T., & Arief, B. (n.d.). *The many meanings of open source.* Retrieved January 30, 2005, from University of Newcastle Web site: http://www.cs.newcastle.ac.uk/research/pubs/trs/papers/737.pdf

Gibbons, S. (2005). Course-management systems. *Library Technology Reports*, *41*(3). Retrieved August 17, 2006, from the American Library Association Web site: https://publications.techsource.ala.org/products/archive.pl?article=2556.

Gill, T. (2006). The memory grid: A glass box view of data representation. *Journal of Information Systems Education, 17*(2), 119–129.

Glass, R. L. (2003). A sociopolitical look at open source. *Communications of the ACM, 46*(11), 21–23.

Green, K. C. (2004). *Campus computing 2004: The 15th national survey of computing and information technology in American higher education.* Encino, CA: The Campus Computing Project.

Hahn, R. (2002). Government policy toward open source: An overview. In R. Hahn (Ed.), *Government policy toward open source* (pp. 1–11). Washington, D.C.: Brookings Institution Press.

Hignite, K. (2004, August). *An open mind on open source.* Retrieved December 1, 2004, from National Association of College and University Business Officers (NACUBO) Web site: http://www.nacubo.org.

Hirst, T. (2001). *The open source teaching project (OSTP): Research note.* United Kingdom: Open University. (ERIC Document Reproduction Service No. 474 093).

Hoepfl, M. (1997). Choosing qualitative research: A primer for technology education researchers. *Journal of Technology Education*, 9(1), Retrieved June 30, 2006, from the Virginia Tech Web site: http://scholar.lib.vt.edu/ejournals/JTE/v9n1/hoepfl.htm.

Honebein, P. C. (1996). Seven goals for the design of constructivist learning environments. In B. G. Wilson (Ed.), *Constructivist learning environments: Case studies in instructional design* (pp. 11–24). Englewood Cliffs, NJ: Educational Technology Publications.

Iyer, H. (2003). Web-based instructional technology in an information science classroom. *Journal of Education for Library and Information Science*, *44*(3), 296–315.

James, C. A. (n.d.). *The care and feeding of FOSS (or, the lifecycle of software technology).* Retrieved February 24, 2005, from http://www.moonviewscientific.com/essays/software_lifecycle.htm

Jasinski, M. (n.d.). EDUCHAOS-disruptive technologies. Message posted to electronic mailing list, archived at http://community.flexiblelearning.net.au/GlobalPerspectives/content/article_5333.htm.

Kelty, C. M. (2004). Culture's open source. *Anthropological Quarterly, 77*(3), 499–506.

Kuali project: A financial system of, by, and for higher education (n.d.). Retrieved June 4, 2004, from Kuali Project Web site: http://www.kuali.org/faq/

Leinonen, T., Hakkarainen, K., Appelt, W., Dean, P., Gomez-Skarmetav, A., Ligorio, B., et al. (2001, June 25). *ITCOLE project: Designing innovative technology for collaborative learning and knowledge building.* Paper presented at the ED-MEDIA 2001 World Conference on Educational Multimedia, Hypermedia & Telecommunications, Tampere, Finland. Retrieved August 4, 2004, from Association for the Advancement of Computing in Education Web site: http://www.editlib.org/index.cfm?fuseaction=Reader.SearchResults&search_query=AND%28Leinonen%29& SEARCH_STRING=Leinonen&CFID=37088808&CFTOKEN=60784793

Linden, A., & Fenn, J. (2003, May 30). *Understanding Gartner's hype cycles.* Retrieved February 3, 2005, from Gartner Group Web site: http://www.gartner.com.

Maxwell, J. (1996). *Qualitative research design: An interactive approach*. Thousand Oaks, CA: Sage Publications.

The MERLOT community. (n.d.). Retrieved December 1, 2005, from the MERLOT Web site: http://www.merlot.org.

Miles, M. & Huberman, A. (1994). *Qualitative data analysis* (2nd ed.). Thousand Oaks, CA: Sage Publications.

Moore, G. A. (1991). *Crossing the chasm.* New York, NY: HarperBusiness.

Moore, G. A. (2005, April 5). *Dealing with Darwin: The role of open source in computing.* Paper presented at the open source Business Conference (OSBC), San Francisco, CA. Retrieved April 29, 2005, from http://www.osbc2004.com/live/13/events

O'Mahoney, S. C. (2002). *The emergence of a new commercial actor: Community managed software projects.* Unpublished manuscript, Stanford University. http://opensource.mit.edu/papers/omahony.pdf

Overby, E. M., Bharadwaj, A. S., & Bharadwaj, S. G. (n.d.). *An investigation of firm-level open source software adoption: Theoretical and practical implications.* Retrieved May 9, 2005, from Emory University Web site: http://userwww.service.emory.edu/~eoverby/files/overby_open_source_adoption_study.pdf

Papastergiou, M. (2006). Course management systems as tools for the creation of online learning environments: Evaluation from a social constructivist perspective and implications for their design. *International Journal on E-Learning*, *5*(4), 593–622.

Pavlicek, R. C. (2000). *Embracing insanity: Open source software development.* Indianapolis, IN: SAMS.

Perens, B. (1999). The open source definition. In C. DiBona, S. Ockman, & M. Stone (Eds.), *Open sources: Voices from the open source revolution* (pp. 171–188). Sebastopol, CA: O'Reilly & Associates, Inc.

Perkins, D. N. (1992). Technology meets constructivism: Do they make a marriage? In T. M. Duffy & D. H. Jonassen (Eds.), *Constructivism and the technology of instruction: A conversation* (pp. 45–56). Hillsdale, NJ: Lawrence Erlbaum Associates.

President's Information Technology Advisory Committee (PITAC). (2000, October). *Developing open source software to advance high end computing: Report to the President.* Arlington, VA: National Coordination Office for Information Technology Research and Development. (ERIC Document Reproduction Service No. ED 462 967).

Raymond, E. S. (2001). *The cathedral and the bazaar: Musings on Linux and open source by an accidental revolutionary.* Cambridge, MA: O'Reilly.

Rogers, E. (1995). *Diffusion of innovations* (4th ed.) New York, NY: The Free Press. (Original work published 1962)

Sakai (n.d.). Retrieved December 6, 2004, from The Sakai Project Web site: http://www.sakaiproject.org

Scacchi, W. (2001, December). *Understanding the requirements for developing open source software systems.* Retrieved November 7, 2004, from Free/Open source Research Community Web site: http://opensource.mit.edu/papers/Scacchi.pdf

Stallman, R. (1999). *The GNU project.* Retrieved July 16, 2004, from The GNU Web site: http://www.gnu.org/gnu/thegnuproject.html

Stewart, K. J., & Gosain, S. (2004, August 23). *The impact of ideology on effectiveness in open source software development teams.* Retrieved September 20, 2004, from Free/Open Source Research Community Web site: http://opensource.mit.edu/papers/stewartgosain.pdf

The TLT Group history and data (n.d.). Retrieved December 1, 2005, from The TLT Group Web site: http://www.tltgroup.org/about/history.html

Villano, M. (2006). Open source vision. *Campus Technology*, *19*(11), 26–36.

Von Krogh, G. (2003). Open source software development. *MIT Sloan Management Review, 44*(3), 14–18.

Weber, S. (2004). *The success of open source.* Cambridge, MA: Harvard University Press.

Williams, S. (2002). *Free as in freedom: Richard Stallman's crusade for free software.* Sebastopol, CA: O'Reilly & Associates, Inc.

Wilson, B. G. (1996). What is a constructivist learning environment? In B. G. Wilson (Ed.), *Constructivist learning environments: Case studies in instructional design* (pp. 3–10). Englewood Cliffs, NJ: Educational Technology Publications.