

This work is licensed under the Creative Commons Attribution-ShareAlike License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-sa/2.0/> or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.



the r-smart group

Open source - opens learning

Why open source makes sense for education

Summer 2004

the r-smart group
4343 E. Camelback Rd.
Suite 210
Phoenix, AZ 85018

☎ 602.840.7300
☎ 602.840.7500

Chris Coppola
chris.coppola@rsmart.com

Ed Neelley
ed.neelley@rsmart.com

contents

Introduction-----	2
Open source plain and simple-----	2
Comparison to proprietary software -----	3
The open source movement in higher education ---	5
How could open source make life better?-----	11
Ok, it makes sense... what should I do?-----	13
References -----	14

www.rsmart.com

Introduction

Because of the rise in popularity and consideration of open source applications in all markets from education to government to business, it is critical for all decision makers to understand what open source applications are and what the implications are for their organization.

This is especially true in the education market where budget pressures make the right decision an imperative. This white paper will offer a simple, yet thorough definition of open source in the context of education, describe the new market models, and dispel the myths about open source.

Open source plain and simple

Open source software (OSS) refers to software programs that are distributed with the source code—hence open source. The open source license allows users the freedom to run the program for any purpose, to study and modify the program, and to freely redistribute copies of the original or modified program.

Open source is a way of building software. Open source software is often constructed using the same time-tested methodologies and practices used to construct proprietary software. However there are a number of interesting differences that result from the difference in rights conferred by the license:

- The software evolves more rapidly and organically. Many individuals and teams working in parallel produce advances beyond the capacity of a single team. The improvements are managed into releases by a core team much like the proprietary alternative.
- Users' needs are rapidly met as the OSS model harnesses their collective expertise and contribution. Improvements to the software are often contributed by users who have fixed a problem or added new features.
- New versions are released very often and rely on the community of users and developers to test it, resulting in superior quality software tested on more platforms, and in more environments than most commercial software.
- The development "team" is often largely volunteers, distributed, many in numbers, and diverse. Often, paid members of the development team will manage the project and organize the work of the volunteers.
- Security is enhanced because the code is exposed to the world. In proprietary packages, shortcuts and poor programming are more easily overlooked because only a select few see the code. Like peer reviewed papers, open source code is subject to much greater scrutiny and thus produces a higher caliber result.

Open source is a way of doing business. Open source is often thought of as free software. In fact, one of the benefits is that anyone can use, modify, and redistribute the software without a license fee. While the code is free, the benefits have much more to do with the freedoms than the cost.

On the surface it may appear that *free software* is in opposition to the 'business' of software and, while the open source model challenges the *traditional* models, open source represents great opportunity for customers and businesses alike. Without the restrictions of a

proprietary software license, any institution or company with the skills can use the software to solve their own needs or offer products and services of value to others. Businesses based on open source software typically add value by:

- Offering services such as implementation, training, and support;
- Packaging and integrating open source software to make its installation and use easier for a wider market;
- Creating complementary, add-on, or enhanced software for sale.

Open source affords institutions, as consumers of software, unparalleled control over the use of software to meet business demands and control over where they deploy their budget dollars. Taking into consideration its core competencies and available resources, an institution may use its own people and resources, hire external resources, or partner to share costs.

The principles of open source software—freedom to run the program for any purpose, to study and modify the program, and to freely redistribute copies and modifications—are simple, yet have powerful and far reaching implications.

Comparison to proprietary software

One of the great misconceptions about open source is that it is an alternative to commercial software. Rather, open source software is an alternative to proprietary software, but often acquired through or supported by a company—in other words commercial.

Fundamentally, the difference between open source and proprietary software has to do with control. Where the open source license imparts freedoms to use, modify, and redistribute the software, the proprietary license restricts use, modification, distribution, and more. Vendors of proprietary software restrict access to and use of the source code because the source is knowledge—and knowledge is power.

The open source license corrects the balance of power between producers and consumers of software. Access to the source code is about freedom and choice. After all, most of us aren't software engineers—we can't start adding features and fixing bugs just because we have access. Yet access to the source code is important to all of us because access is control. Red Hat CEO Bob Young asks the question, “Would you buy a car with the hood welded shut?” and after everyone answers “No.” He follows up with “What do you know about modern internal-combustion engines?” The ability to open the hood, like access to the source code, gives the consumer control. The consumer chooses to have the car serviced at the dealer, if he's not satisfied with the work or the price, he can choose to take the car somewhere else.ⁱ

Historically, the business model for software has not given consumers the kind of choices that let market forces work to keep costs in check and innovation thriving. The model for software has been a for-fee license to use software, bundled with for-fee services. The open source model un-bundles the software and services, and creates distinct markets for each.ⁱⁱ Consumers have greater choice and market forces drive competition and control price and quality.

Open source clearly offers advantages for the consumer, but it also represents a significant competitive advantage to companies that adopt compatible strategies.ⁱⁱⁱ IBM, HP, Sun

Microsystems, Apple Computer, Red Hat, Novell, and many other companies attribute significant revenue from strategies built on commercialization of open source software.

Mainstream press coverage of open source has been accelerating all over the world, but the press coverage in higher education has been nothing short of explosive in recent months. Vendors of proprietary software have been carefully crafting their powerful and influential messages about open source. Some attempt to categorize open source as something applicable to research and experimentation. Others cast open source as comparable to open standards. Still others warn about considering open source software for mission critical applications. Each of these messages uses generalizations that distort the truth. Let's examine some of the important attributes and how open source and proprietary approaches compare:

<i>Attribute</i>	<i>Open Source</i>	<i>Proprietary</i>
License	Licensed to provide freedom to use for any purpose, modify, and redistribute.	Licensed to restrict use to "acceptable uses", protect against modification and redistribution.
Control	Balanced. Consumers and providers of commercial offerings have equal access.	Vendor is in control. Often large up-front investments in the software, training, and other implementation costs create a lock-in situation that strips the consumer of control.
Innovation	Rapid and diverse. Leverages a very large community of users and developers working in parallel.	Limited to vendor investments. Typically caters to the features sought by the largest audience. Low levels of innovation.
Longevity / Risk of abandonment	Software will always be available as long as it serves a useful purpose. The larger the adoption the safer the investment. No single point of failure.	Dependent on the success of the vendor. Single point of failure.
Security and reliability	As the great tradition of peer review produces high quality academic and scientific works, open source projects produce software that is generally more secure and reliable. ^{iv} Like peer reviewed papers, open source software is subject to greater scrutiny and leverages a larger collective intelligence than proprietary software.	
Total cost of ownership (TCO)		
Acquisition cost	Low or no initial license cost.	License costs on the rise in education.
Implementation and support costs	OS community support and competition for commercial support keeps costs low. Choice allows consumer to bear costs internally which may further decrease cost.	Costs determined by and coupled to vendor. Lack of choice. Alternative service organizations often lead to higher costs.
Costs of scale	No additional license cost with additional users, servers, etc.	License costs generally rise with increased use.

Platform support	Broader platform and server support increases leveraged investments in existing institutional infrastructure.	Platform support limited to what the vendor offers. May not fit well with institutional infrastructure.
------------------	---	---

As you can see from the comparison, open source promises some impressive advantages over proprietary software. While open source has a rich history in education, a recent convergence of powerful drivers and enablers are precipitating a strong movement toward open source in higher education.

The open source movement in higher education

“Colleges are in search of a new business model for developing and licensing software, and open-source-code projects are leading the way.”

*Richard N. Katz
Vice President of Educause,
the higher education technology consortium*

Open source is changing the nature of the software business in many settings, but the movement is particularly impressive in education.

While open source has nearly a 20 year history of success producing leading software,^{iv} it has mostly been in the *infrastructure* domain—the protocol implementations, servers, development frameworks, and utilities that enable the email programs, browsers, and other applications we all depend on. Open source has not been a significant model for producing *application* software—until recently.

In the past few years colleges and universities have begun to produce enterprise open source applications like course management systems and electronic portfolios that compete directly with their proprietary counterparts. These e-learning applications are leading a movement in higher education from proprietary software toward open source.

A snapshot of the current proprietary software environment in education reveals some compelling drivers...

- Tight budgets have focused attention on software acquisition costs^v, and total cost of ownership
- Growing resentment of vendor power, particularly in the wake of price increases and licensing changes that many institutions felt powerless to reject^v
- Lack of innovation. Learning technology has not lived up to its potential to improve learning

... and some enablers of a more effective model:

- Collaboration technology has made large-scale collaborative work across institutional, geographic, and cultural boundaries more effective
- Software design patterns, development technologies, and standards have evolved in a way that facilitates modular, interoperable software components

Open source - opens learning: why open source makes sense for education.

- Proven business models and education focused companies that embrace open source
- Strong cultural appeal of open source in academia

We observe many of these drivers and enablers beyond the context of higher education. Perhaps the cultural appeal in academia represents the ‘tipping point’ for a powerful movement unique to colleges and universities.

At the heart of the cultural fit between academia and open source are common philosophy and values. Creating and sharing knowledge for public good is a key part of the mission of colleges and universities, and a core part of the philosophy driving open source software.


Colleges and universities are also home to some of the best software engineers in the world. Yet most institutions don’t have a plentiful supply of available talent to meet evolving needs—individually. The open source model provides colleges and universities a way to leverage cultural values of collaboration and sharing to gather resources and work together for the common good. Unlike contexts outside of education, working together doesn’t blur institutional brand or threaten a school’s income. As a result, higher education not only boasts some of the best software engineers, but also has a plentiful supply of willing talent—a critical success factor for open source.


A common challenge to developing software is understanding the needs of the users and turning that understanding into a suitable software design. While there are many different approaches and software methodologies, the open source model clearly has an advantage since the individuals most involved in teaching and learning are deeply involved in the development of the software for teaching and learning.

Another critical success factor for open source is the emergence of education-focused companies that embrace open source as the preferred model for business. Although colleges and universities are well suited to develop world class enterprise software, supporting that software for other institutions isn’t an activity that helps fulfill their mission. A new ecology has emerged where commercial organizations, colleges, and universities play complimentary and collaborative roles.



The Andrew W. Mellon foundation, the William and Flora Hewlett Foundation, and others have recognized the need and the potential and have made considerable financial resources available to seed open source projects and help create the new ecology.

While there are literally thousands of active open source projects all over the world addressing most application domains, there are a number of notable projects particularly interesting for institutions of higher education. The following table illustrates a sample of the momentum and potential.


<i>Project</i>	<i>Description</i>
 www.uportal.org	uPortal is a free, sharable portal under development by institutions of higher-education. This group sees an institutional portal as an abridged and customized version of the institutional Web presence... a "pocket-sized" version of the campus Web. Portal technology adds "customization" and "community" to the

<p>Partially funded by the Mellon foundation</p>	<p>campus Web presence. Customization allows each user to define a unique and personal view of the campus Web. Community tools, such as chat, forums, survey, and so on, build relationships among campus constituencies.</p> <p>uPortal is a product of the JA-SIG collaboration which is governed by a Steering Committee of between 4 and 7 members. Currently, the Steering Committee includes Princeton University, the University of British Columbia, Boston College, the University of Delaware, and Sun Microsystems.</p>
<p>Shibboleth shibboleth.internet2.edu</p>	<p>Shibboleth, a project of Internet2/MACE, is developing architectures, policy structures, practical technologies, and an open source implementation to support inter-institutional sharing of web resources subject to access controls. In addition, Shibboleth will develop a policy framework that will allow inter-operation within the higher education community.</p> <p>The Shibboleth system provides a standards-based link between existing campus authentication systems and resource providers of all kinds. For example, when a student requests access to a protected video clip, her home organization (origin site) requests her to authenticate (if she has not done so already) and then passes on the information that she is enrolled in Biology 562 to the site housing the video. The provider (target site) uses the fact that she is enrolled in this course to determine her eligibility to access the video.</p>
<p>Globus www.globus.org</p>	<p>The Globus Alliance is developing fundamental technologies needed to build computational grids. Grids are persistent environments that enable software applications to integrate instruments, displays, computational and information resources that are managed by diverse organizations in widespread locations.</p> <p>The Globus Toolkit is an open source software toolkit used for building grids. It is being developed by the Globus Alliance and many others all over the world. A growing number of projects and companies are using the Globus Toolkit to unlock the potential of grids for their cause.</p> <p>The core Globus alliance team are: Argonne National Laboratory, University of Chicago, University of Southern California Information Sciences Institute, University of Edinburgh, Swedish Royal Institute of Technology, and the High Performance Computing Laboratory Northern Illinois University.</p>
 <p>www.theospi.org</p> <p>Partially funded by the Mellon foundation</p>	<p>The Open Source Portfolio Initiative (OSPI) is a collaborative, open-source, software development project based on the University of Minnesota Enterprise System's electronic portfolio software. The University of Minnesota (U of MN), University of Delaware, and the r-smart group, founded this collaborative to open the evolution of the U of MN ePortfolio to diverse input, rapid development, and widespread use.</p>

 <p>www.dspace.org</p> <p>Partially funded by the Mellon foundation</p>	<p>DSpace is a groundbreaking digital library system to capture, store, index, preserve, and redistribute the intellectual output of a university's research faculty in digital formats.</p> <p>Developed jointly by MIT Libraries and Hewlett-Packard (HP), DSpace is now freely available to research institutions worldwide as an open source system that can be customized and extended.</p>
 <p>www.sakaiproject.org</p> <p>Partially funded by the Mellon foundation</p> <p>Partially funded by the Hewlett foundation</p>	<p>The Sakai Project is a community source project founded by The University of Michigan, Indiana University, MIT, Stanford, the uPortal Consortium, and the Open Knowledge Initiative (OKI) with the support of the Andrew W. Mellon Foundation and the Hewlett Foundation. The project is producing open source Collaboration and Learning Environment (CLE) software.</p>
<p>Chandler</p>  <p>www.osafoundation.org</p> <p>Partially funded by the Mellon foundation</p>	<p>The Open Source Applications Foundation (OSAF) is developing a Personal Information Manager (PIM) intended for use in everyday information and communication tasks, such as composing and reading email, managing an appointment calendar and keeping a contact list. Because of the ease with which Chandler users can share information with others, Chandler might be called the first Interpersonal Information Manager. (The term PIM was first used in conjunction with the product Lotus Agenda in the 1980's. Chandler is the spiritual descendant of Agenda (and has a common designer in Mitch Kapor.)</p> <p>Chandler is intended to be an open source personal information manager for email, calendars, contacts, tasks, and general information management, as well as a platform for developing information management applications. It is currently under development and will run on Windows, Mac, and Linux-based PC's.</p>
 <p>FEDORA</p> <p>www.fedora.info</p> <p>Partially funded by the Mellon foundation</p>	<p>Jointly developed by the University of Virginia and Cornell University, the system completely implements the Fedora architecture that was originally conceived of at Cornell. Fedora is a general-purpose digital object repository system that can be used in whole or part to support a variety of use cases including: institutional repositories, digital libraries, content management, digital asset management, scholarly publishing, and digital preservation.</p>

 <p>LionShare</p> <p>lionshare.its.psu.edu</p> <p>Partially funded by the Mellon foundation</p>	<p>The LionShare P2P project is an innovative effort to facilitate legitimate file-sharing among individuals and educational institutions around the world. By using Peer to Peer (P2P) technology and incorporating features such as authentication, directory servers, and owner controlled sharing of files, LionShare promises secure file-sharing capabilities for the easy exchange of image collections, video archives, large data collections, and other types of academic information. In addition to authenticated file-sharing capabilities, the developing LionShare technology will also provide users with resources for organizing, storing, and retrieving digital files.</p> <p>The LionShare project began as an experimental software development project at Penn State University to assist faculty with digital file management. The project has now grown to be a collaborative effort between Penn State University, Massachusetts Institute of Technology Open Knowledge Initiative, researchers at Simon Fraser University, and the Internet2 P2P Working Group. A generous grant from the Andrew W. Mellon Foundation will be used to fund the first two years of the project.</p>
<p>PKI</p> <p>www.dartmouth.edu/~pkilab</p> <p>Partially funded by the Mellon foundation</p>	<p>Dartmouth College with support from the Mellon Foundation is working to develop use of Public Key Infrastructure in Academic Computing. There are many potential applications for PKI in an academic setting. There are also many obstacles to be overcome in order to provide PKI based services broadly in the standard network computing environment. This project proposes to address these issues with a dual approach. A deployment track, lead by Computing Services, is working to simplify the use of current X.509 PKI technology and to develop usable applications of it that address current security concerns. A design track, lead by Computer Science is examining limitations of the technology and working to develop solutions to the problems uncovered. The design track plans to implement a more secure infrastructure by incorporating secure co-processing technology in the services. Alternate formulations of PKI technology are also being examined.</p> <p>The project aims to develop inter-institutional applications and has been working with a group of Internet2 institutions, along with the University of Wisconsin and the University of California - San Diego. The Dartmouth PKI Lab web site is used to disseminate general information about PKI systems and share the results of our development work.</p>
 <p>Haystack</p> <p>haystack.lcs.mit.edu</p>	<p>Haystack is a tool designed by MIT to let every individual manage all of their information in the way that makes the most sense to them. By removing the arbitrary barriers created by applications only handling certain information "types", and recording only a fixed set of relationships defined by the developer, we aim to let users define whichever arrangements of, connections between, and views of information they find most effective. Such personalization of information management will</p>

	<p>dramatically improve each individual's ability to find what they need when they need it.</p>
 <p>www.lamsinternational.com</p>	<p>LAMS is a revolutionary new tool for designing, managing and delivering online collaborative learning activities.</p> <p>It provides teachers with a highly intuitive visual authoring environment for creating sequences of learning activities. These activities can include a range of individual tasks, small group work and whole class activities based on both content and collaboration.</p> <p>Whether you are a long time user of current e-learning tools, or new to the area, LAMS will release you from administration and propel you and your students to a new level of interaction and satisfaction that has the potential to dramatically increase student learning.</p> <p>LAMS International and LAMS Foundation are both organisations managed by Macquarie University in Sydney Australia (www.mq.edu.au). The LAMS Foundation (www.lamsfoundation.org) is designed to do applied research into the impact and development of LAMS visit the site for a brief overview.</p>
 <p>www.moodle.org</p>	<p>Moodle is a course management system (CMS) - a software package designed to help educators create quality online courses. Such e-learning systems are sometimes called Learning Management Systems (LMS) or Virtual Learning Environments (VLE). One of the main advantages of Moodle over other systems is a strong grounding in social constructionist pedagogy.</p> <p>Moodle is Open Source software, which means you are free to download it, use it, modify it and even distribute it (under the terms of the GNU General Public License). Moodle runs without modification on Unix, Linux, Windows, Mac OS X, Netware and any other system that supports PHP, including most webhost providers. Data is stored in a single database: MySQL and PostgreSQL are best supported, but it can also be used with Oracle, Access, Interbase, ODBC and others.</p> <p>Moodle is available in 40 languages, including: Arabic, Catalan, Chinese (simplified and traditional), Czech, Danish, Dutch, English (UK and US versions), Finnish, French (France and Canada versions), German, Greek, Hungarian, Indonesian, Italian, Japanese, Norwegian, Polish, Portuguese (Portugal and Brazil), Romanian, Russian, Slovak, Spanish (Spain, Mexico, Argentina and Caribbean versions), Swedish, Thai and Turkish.</p>
<p>Visual Understanding Environment (VUE)</p> <p>vue.tccs.tufts.edu</p> <p>Partially funded by the Mellon foundation</p>	<p>The Visual Understanding Environment (VUE) project at Tufts' Academic Technology department provides faculty and students with flexible tools to successfully integrate digital resources into their teaching and learning. VUE provides a visual environment for structuring, presenting, and sharing digital information and an OKI-compliant</p>

	<p>software bridge for connecting to FEDORA-based digital repositories. Using VUE's concept mapping interface, faculty and students design customized semantic networks of digital resources drawing from digital libraries, local files and the Web. The resulting content maps can then be viewed and exchanged online. This project is supported by the Andrew W. Mellon Foundation.</p>
 <p>Partially funded by the Mellon foundation</p>	<p>The Open Knowledge Initiative™ started at MIT in 2001 with funding from the Andrew W. Mellon Foundation. It is a collaboration among leading universities and specification and standards organizations to support innovative learning technology in higher education.</p> <p>The result of this collaboration is an open and extensible architecture that specifies how the components of an educational software environment communicate with each other and with other enterprise systems. O.K.I. provides a modular development platform for building both traditional and innovative applications while leveraging existing and future infrastructure technologies.</p> <p>O.K.I. is designed for broad adoption in the higher education domain. It provides a stable, scalable base that supports the flexibility needed by higher education and commercial developers of educational software.</p>

How could open source make life better?

“...if an institution acquires a commercial, proprietary LMS, and then finds that the system is deficient in some way, they often must wait until the vendor decides it is financially viable to develop the enhancement—an event that may never occur.”^{vi}

*Ira Fuchs
Vice President for Research in Information Technology
Andrew W. Mellon Foundation*

Imagine...

... the Freedom to choose...

- Institutions are discovering the freedom to choose where to spend their technology dollars. Open source software is not free, but allows different kinds of budget decisions than proprietary software. With an open source application, license fees and maintenance fees are generally low or non-existent. The institution can now

invest in the user experience through training, customization, and increasing the level of adoption. In addition, budgeting for system costs becomes more predictable.

- Institutions are free to select who to work with because open source de-couples the software from the services. With proprietary software, there is often a single source for both software and services. Open source provides the freedom for institutions to select software and service provider separately, each based upon its respective merits.

... control over your destiny...

- Institutions can develop additional functionality at their own pace rather than begging a proprietary vendor to include a feature—and then waiting another development cycle or more to get it.
- Open source applications are most often architected for the highest level of flexibility through community development. This level of flexibility allows developers to create an application that works the way instructors teach and students learn. Users now can interact in the most effective way for teaching and learning, allowing the tool to change rather than having to change their way of conducting their instruction and learning.
- Open source now allows users to choose from a wide range of best of breed enhancements to align their best practices with the behavior of the software.

... connection to a global community ...

- The open source community is vested in mutual success. Collaborative peer support is common from individuals with a stake in the outcome.
- Because of the nature of the open source community, applications are subjected to intense peer evaluation for everything from functionality to usability. Problems and bugs are identified and addressed quickly.
- In the open source community, technology Darwinism is at work. Well-designed applications (as well as enhancements) tend to survive the rigorous testing and scrutiny of the community. Weak applications are weeded out very quickly.

Imagine...

... Peace of Mind and confidence ...

- With open source, there's no single point of failure. Open source software is not dependent on the viability of one organization; the risk is distributed among all that use and profit from it.
- Open source projects can't be bought and sold. They are not subject to the acquisitions and changing business models that make software investments risky and unpredictable.
- Since companies can't rely on ownership of the software to keep customers, they must compete on competence and quality of service. Market forces provide checks and balances.

... Innovation in teaching and learning technology...

- The spirit of open source is formed around diversity of input, recombination of ideas, creativity, and collaboration. These are essential ingredients for innovation and clear advantages to the open source philosophy.
- The open source community evolves projects quickly and organically. Open source projects are living systems that promote rapid development and constant improvement of concepts.

Ok, it makes sense... what should I do?

The open source story makes a compelling argument for adoption and personal action. In today's education environment, leaders must be innovative to not only elevate the stature of their institution, but to simply survive. And open source innovation represents the leading edge of a revolution that will literally change the landscape of technology in education for years to come. And the change is upon us. The question is: Do you want to benefit now or later?

The first step in getting involved is to make open source part of your vocabulary. Build up your understanding of open source and the landscape in education, and as you do:

- Discuss the potential and the challenges of open source with your colleagues.
- Find ways to contribute to an open source project. There is much more to open source than writing code and each of us has talents that can contribute to the movement. You might be a developer, or a designer, or simply a user of the software. You might be a leader on campus that can encourage your colleagues to consider open source alternatives.
- Initiate a pilot program on your campus.

Second, you will find that after you get involved you will want to be an active part of the community—a community of like minded individuals with a stake in the outcome. Developers and users support each other across institutional, political, and geographic boundaries. Users not only use the software, they contribute to the design.

"If we don't change the direction we're going, we're likely to end up where we are headed."

Chinese proverb

The demands on higher education require a fundamental change in direction—and technology can facilitate that change. But the present technology for teaching and learning hasn't lived up to its potential. Open source will pave a new road—changing not only the destination, but the journey, which is the real reward.

Open source - opens learning: why open source makes sense for education.

References

Eric Raymond (2000). The Cathedral and the Bazaar. <http://www.catb.org/~esr/writings/cathedral-bazaar/cathedral-bazaar/>

Steven Weber (April 2004). The Success of Open Source. Harvard University Press. <http://www.hup.harvard.edu/catalog/WEBSUC.html>

Open Source and Free Software-Concepts Controversies, and Solutions, http://epresence.tv/website_archived.aspx?dir=May~9-11,~2004:~Open~Source~and~Free~Software:~Concepts,~Controversies~and~Solutions

Bradley C. Wheeler (2004). Open Source 2007: How Did This Happen? EDUCAUSE Review, vol. 39, no. 4 (July/August 2004): 12–27. <http://www.educause.edu/pub/er/erm04/erm0440.asp>

Josh Lerner, Jean Tirole, National Bureau of Economic Research (March 2000). The simple Economics of Open Source. <http://www.people.hbs.edu/jlerner/simple.pdf>

Florence Olsen (August 2003). Sharing the Code. The Chronicle of Higher Education. <http://chronicle.com/free/v49/i47/47a03101.htm>

Trask, Gallagher, Harden, Smith (January 2002). Understanding Open Source: Investigating an Alternative Software Development Model for the Postsecondary Market. http://www.eduventures.com/research/industry_research_resources/opensource.cfm

ⁱ Bob Young, Red Hat CEO / Chairman

ⁱⁱ Bradley C. Wheeler (2004). The Inevitable Unbundling of Software and Support. Syllabus Magazine. <http://www.syllabus.com/article.asp?id=9026>

ⁱⁱⁱ John Koenig (2004). Seven Open Source Business Strategies for Competitive Advantage. IT Manager's Journal. <http://management.itmanagersjournal.com/management/04/05/10/2052216.shtml?tid=85>

^{iv} David A. Wheeler (May 2003). Why Open Source Software / Free Software (OSS/FS)? Look at the Numbers! www.dwheeler.com/oss_fs_why.html

^v R. Yanosky, M. Harris, M. Zastrocky (December 2003). Higher-Education E-Learning Meets Open Source

^{vi} Ira Fuchs (July 2004). Learning Management Systems: Are We There Yet? <http://www.syllabus.com/article.asp?id=9675>