

## Proprietary software versus Open Source Software for Education

N. Pankaja<sup>1</sup>, Mukund Raj P K<sup>2</sup>

<sup>1</sup>EMRC (Electronic Media Research Center, India)

<sup>2</sup>(MLRCC/ University of Mysore, India)

**Abstract:** - The Internet has brought learning "online" and offers many advantages. It is convenient, available at any time of the day, and can be accessed nearly anywhere in the world. Recently, Cloud computing is all the rage. E-Learning offers tremendous potential to increase the availability and convenience of education. Today, online content is varied and can include: text on a website, digital audio, digital video, animated images, and virtual reality environments. This content can be created in a variety of ways by utilizing a variety of authoring tools and softwares. These days, we observe a movement in higher education leading from proprietary software to open source, for e-learning applications (1) In fact, open source software (OSS) development can provide the necessary flexibility to combine languages, scripts, learning objects and lesson plans, effectively, without the cost and rigidity of proprietary software. In recent years, numerous open access LMS software's have emerged as viable alternatives to costly proprietary and commercial products. Open source software's of Content Management Systems (CMS) and Learning Management System (LMS) are gaining popularity. We weigh the pro's and cons of utilizing the OSS and proprietary software in this paper.

**Keywords:** - Educational software, Open Source Software (OSS), proprietary software, e-learning, Content and Learning Management System

### I. INTRODUCTION

The Internet has brought learning "online" and offers many advantages. It is convenient, available at any time of the day, and can be accessed nearly anywhere in the world. E-Learning offers tremendous potential to increase the availability and convenience of education. When e-Learning is made accessible and Section 508 compliant, it enables employees with disabilities to receive equivalent access to training materials used by their peers. As with many types of products and technologies, including those used in e-Learning, people with disabilities may inadvertently be excluded if accessibility is not considered and incorporated into products and technologies. However, accessibility is not only of concern to those with disabilities. The potential for e-Learning expands when developers embrace the widest possible range of individual learning styles, preferences, and abilities. Today, online content is varied and can include: text on a website, digital audio, digital video, animated images, and virtual reality environments. This content can be created in a variety of ways by utilizing a variety of authoring tools and softwares. e-learning means a combination of Content Management System (CMS) and Learning management system (LMS). We shall henceforth call the combination of CMS and LMS a total solution. There are only two choices of a total solution for an educational system – either opt proprietary software or an open source software system. Hence any institution will have to evaluate their choices. The total solution plays an ever increasing and prominent role in the teaching and learning process, paving a new road changing the existing ways of teaching and learning, from a traditional in class way to totally synchronous or asynchronous distant one. While CMS is the front end for the total solution, LMS is the back-end. These days, we observe a movement in higher education leading from proprietary software to open source, for e-learning applications (2). In fact, open source software development can provide the necessary flexibility to combine languages, scripts, learning objects and lesson plans, effectively, without the cost and rigidity of proprietary packages (3) The purpose of this paper is to consider the challenges posed by educational systems into choosing either a proprietary system or an open source systems and the criteria that is to be adopted to choose from either.

## II. SOFTWARE'S AND ITS MODELS

Software (SW) can be shortly defined as the executable code that controls computer behaviour and operations. The term is used, however, to describe a wide range of programming languages, applications, procedures and all related documentation resources. SW also refers to a full cycle of processes from basic architecture to development, packaging and distributing. It is responsible for controlling, integrating, and managing the individual hardware components of a computer system so that other software and the users of the system see it as a functional unit without having to be concerned with the low-level details of the computational system. There are three models of software's for developing any application in computer. They are Freeware, Shareware, Trialware, Proprietary, Public Domain Software and Open Source.

### 2.1 Freeware

It is provided to be used without any monetary charges. However, severe restrictions of its use, modification and redistribution are still imposed and Source code is not provided. It can be passed on to anyone free of cost.

### 2.2 Open-source software (OSS)

It is a computer software with its source code made available and licensed with an open-source license in which the copyright holder provides the rights to study, change and distribute the software for free to anyone and for any purpose. Open-source software is very often developed in a public, collaborative manner. Open-source software is the most prominent example of open-source development and often compared to (technically defined) user-generated content or (legally defined) open-content movements

### 2.3 Proprietary Software

The term proprietary is derived from the Latin word *proprietas* meaning property. Proprietary Software is computer software licensed under the exclusive legal rights of the copyright holder. Proprietary software is developed by a person or firm who has rights of using existing or developing new tools to create new software. A proprietary software developer sells or provides his creation under some concrete conditions which should be followed in order to avoid any legal issues. In general, these concrete conditions involves usage using software with a purchased license, within the permitted boundaries, no modification allowed, no further re-distributions and no reverse engineering to applied. The main identity of proprietary software is that its source code is always kept secret from outside world. Thus, the internal structure of proprietary software is not exposed. The restrictions on proprietary software are generally imposed through a document called EULA (End-user license agreements) to which users are supposed to agree before using the software. It works just like a contract of usage conditions between the user and vendor. If a user is found indulged in activities leading to the breaking of copyright conditions, the selling authority has the right to impose legal actions against the misusing personal.

### 2.4 Public Domain Software

The copyright holder donates it to the public. Thus, it is no longer copyrighted and restricted and thus moves towards Open source.

### 2.5 Shareware

It is generally that version of software which is limited in terms of functionality. Users can download it from internet, uses it with restrictions and then decide to pay for its full version. These can be shared among other people.

### 2.6 Trialware

It is generally that version of software which is limited in terms of duration of use. User gets the full version of software which can be limited to the usage number of days or usage number of times. After paying and getting registered, the user will get unrestricted access to usage and updates of that software.

## III. PROPRIETARY AND OPEN SOURCE SOFTWARE

Assuming that an educational institute requires developing, implementing and deploying an e-learning solution We will now focus on the OSS and Proprietary software for the purpose of this paper Before we proceed to evaluate OSS and Proprietary software, let us first understand both of them.

### 3.1 Definition of Proprietary Software:

Software that is distributed under commercial license agreements is usually for fee. The main difference between the proprietary software license and the open source license is that the recipient does not normally receive the right to copy, modify, or redistribute the software without fees or royalty obligations.

Something proprietary is something exclusively owned by someone, often with connotations that it is exclusive and cannot be used by other parties without negotiations. It may specifically mean that something is covered by one or more patents, as in proprietary technology. Proprietary software means that some individual or company holds the exclusive copyrights on a piece of software, at the same time denying other people the access to the software’s source code and the right to copy, modify and study the software.

**3.2 Definition of Open Source Software:**

Software, whose source code is published and made available to the public, enabling anyone to copy, modify and redistribute the source code without paying royalties or fees. Open source code evolves through community cooperation. These communities are composed of individual programmers, users as well as very large companies. Some examples of open source initiatives are GNU/Linux, Eclipse, Apache, Mozilla etc

**TABLE 1 – Proprietary versus Open source software**

Details	Proprietary Software	Open Source Software
Cost	Varies from a few thousand to a few hundred thousand dollars, depending on the complexity of the system required. This cost is made up of a base fee for software, integration and services and annual licensing/support fees. This cost may be prohibitive for some; however what the user is paying for is a more customized product from a trusted brand that includes higher levels of security and functionality, continuous innovation, a greater ability to scale, on-going training and support and a lower requirement for technical skills.	OSS comes at a low cost because of Open source software. We don't need to get an expensive software or hardware to run the system. Organizations can use this system as long as they like, without thinking of paying any set up, activation, and monthly subscription charges
Service and Support	Service is probably the greatest advantage of using proprietary software. Proprietary software providers offer ongoing support to users, a key selling point for users without technical expertise. If the user manual or guide is not enough, or if a user experiences a problem with the software, there is an immediate point of call to turn to for assistance. There is a certain reduction in the risk undertaken with proprietary software because users are working with companies that are viable, and people with intimate knowledge of the products and services being used should any questions arise. Because service is one of the main reasons users choose proprietary over open source software, many proprietary software providers compete on service, increasing the bargaining power of buyers and thereby increasing customer service levels among providers.	While Service is one of the key issues regarding open source software. Open source software relies on its online community network to deliver learning support via forums and blogs. While there are massive, loyal and engaged online communities that users are turning to. This requires some basic knowledge and skill set from the user to understand feedback from online community and resolve them. Sometimes, the trouble shooting is faster than those of proprietary software. But users who query, should know what their actual problem is, else, they will not receive the right feedback.
Innovation	Proprietary software providers do not allow users to view or alter the source code. While this may be viewed as a disadvantage to some, it ensures the security and reliability of the software. Furthermore, many proprietary software providers customize software for specific users to provide more flexibility while investing in R&D in order to regularly offer new products and upgrades. Proprietary software providers have online user communities that create value by sharing ideas, strategies and best practices through feedback mechanisms such as forums and surveys, which also foster innovation and allow the product to adapt with changing needs. This innovation comes fully tested, and is available to all users of the software. It does not require investment in R&D or the technical understanding of source code, and assistance with implementation is generally part of the package. Because vendors must ensure their software does not become redundant, users also benefit from the type of targeted innovation undertaken- continuous investment in R&D rather than “innovation for innovation’s sake”, business focused rather than technology focused.	OSS enables innovation by providing users with the freedom and flexibility to adapt the software to suit, without restriction. However, innovation may or may not be passed on to all users of the software. It is a user’s prerogative whether they wish to share their innovation with any online communities, and users must be actively participating in these communities to become aware of such innovations. It has been debated whether customized changes to the original source code limit the future support and growth of the software, as these can potentially result in a limited ability to apply future updates, fixes or modules aimed at improving the software, leaving the user with a version that may have irresolvable issues. It is relevant to note that open source software providers in the initial stages generally struggle to attract large scale R&D funding.

Security	There is always a debate on security. Many proprietary software based developed from proprietary operating systems are perceived relatively less secure to those from OSS. But the total solution from proprietary software is viewed as secure because it is developed in a controlled environment by a concentrated team with a common direction. Moreover, the source code may be viewed and edited by the team alone, and is heavily audited, eliminating the risk of back door Trojans and reducing the risk of any bugs or issues with the software.	Open source operating software is perceived as the most secure OS. Linux being a classic example. But total solutions developed from OSS are perceived less secure since the source code is already available freely to all, hence there may be chances for sabotage. Big players using OSS have robust security policies, hence security in big organizations using OSS is not an issue. Earlier, open source software was not always peer reviewed or validated for use. But currently, with the growing importance of OSS, they are reviewed and corrected. Examples of Drupal, wordpress, Joomla etc are there to see.
Usability	Proprietary software generally employs expert usability testing, and as the software is normally aimed at a more targeted audience, and therefore more tailored, usability is generally ranked quite high. In addition, detailed user manuals and guides are provided. This enables faster training and provides an immediate reference, allowing users to move along the learning curve more quickly. Supporting services include seminars, targeted training courses and extensive support to help maximise use of the software. It is also important to note that while many people see proprietary software as “closed”, today’s proprietary software offers a vast array of mechanisms for enhancement by third party systems and developers	Earlier OSS had been highly criticized for its lack of usability, as generally, the technology is not reviewed by usability experts and does not cater to the vast majority of computer users. Nowadays, OSS is tested by the community for usability and good solutions are being developed globally. Though, open source software does not legally require documentation such as user manuals or guides, hindering the creation of such tools, they are now well documented.
Standards	Describes software interfaces, protocols and electronic formats that are developed by and controlled by a given company and have not been made freely available for adoption by the industry. Some proprietary software uses proprietary standards, i.e. non-public interfaces or electronic formats. When an interface, a protocol or an electronic format is non-public, the owner of the proprietary interface controls it, including when and how the interface changes, who can adopt it, and how it is to be adopted (resulting in user lock-in).	“Open standards” software interfaces, protocols, or electronic formats that are openly documented and have been accepted in the industry through either formal or de facto processes, and which are freely available for adoption by the industry. The open source community has been a leader in promoting and adopting open standards. Some of the success of open source software is due to the availability of worldwide standards for exchanging information, standards that have been implemented in browsers, email systems, file sharing applications, and many other tools. Without open standards it would be impossible to interact and exchange information on the Internet.
Availability	These are available through their respected companies that own the rights to the packages. Sometimes, trial versions are available for free download and testing	These are freely available over the net. Some OSS are also developed into a limited proprietary software with 24X7 support from online community and the developer as well
Transparency	PS does not provide an open look to the internal structure. Only user interfaces are provided to work with it. User cannot know the internal processing and other details.	The source code of OSS is freely available along with the product. Any person can read, modify, build and distribute a modified version of original product. Thus, it gives a transparent look at the core structure of the product.
Reliability	PS is developed by specialized teams at vendor’s end only. Only finished products are provided at outlets. Since there is no un-authenticated modification, the outcome is always reliable.	Since OSS are available on a large number of unverified websites and even most of these distributions may be modified by any technologically sound user, all the distributions are reliable in terms of security, robustness, performance. The reason is that is a user adds/modifies some component; it may work good individually, but, may clash with other components and ultimately degrade the product.

- Some Classic examples proprietary software include Microsoft (windows, asp.net, Internet explorer etc), Apple (Macintosh), Google (Gmail, Chrome etc)
- Some classic examples of OSS include Mozilla Firefox, Linux, Android etc

It is evident that for every successful and established open source software, we have an equivalent and sometimes more popular OSS.

#### IV. POINTS TO PONDER FOR OSS

This leads us to think, why OSS is gaining so much popularity. Why android based smart are phones more popular, why are Joomla, Drupal, Moodle equally popular to proprietary software like Sharepoint, Blackboard etc We now go back to the points the institution has to ponder while making a decision about procurement of a total solution. Some critical criteria are:

1. Budget
2. Stability
3. Flexibility
4. Scalability
5. Usability
6. Life of the solution/ software
7. Security
8. Knowledge base and Self reliance
9. Intellectual Property Rights (IPR)
10. Adaptability to various cross platforms and hardware like mobile, internet, tablet pcs, etc

We have discussed and compared most of these in Table 1.

#### V. QUICK RECON OF OSS

As a quick recon, we will peruse few points here:

- 6.1 Budget: Return on Investment (ROI) is critical to all educational institutions. It is becoming more evident that open source software is providing the ROI. It has been noticed that there is a tendency for software companies to over-specify IT solutions. This is an area where OSS precedes, proprietary software vendors.
- 6.2 Knowledge Base: Implementing an OSS solution requires knowledge base at the User (institution). Without technical know-how – it is difficult to manage OSS. With proprietary software, all help is just a call or email away
- 6.3 Intellectual Property Rights (IPR): OSS provides an excellent opportunity for the user to claim IPR of the final product, whereas with proprietary software, this is impossible. All said and done, only a Non Disclosure Agreement will provide a security to end user with regards to its practices and final solution. This is an area of growing concern and one of the main factors influencing use of OSS
- 6.4 Adaptability: The online community is providing many add-ons and plug-ins that enhance the final solution of an OSS, whereas with proprietary solution, this is limited. A glowing example would be the applications for android and joomla etc wrt other proprietary software

#### VI. WHY USE AN OPEN SOURCE SERVICE

Exactly like the personal computer and the Internet, open source software recently got the attention of the press as a totally new thing that ‘suddenly appeared’. Many people believe that the impact of open source software in the information technology industry and in society in general will be huge, and without precedent in its nature, to the point that the current rules by which the software industry behaves will completely change.

Open source Software’s: In recent years, numerous open access LMS software’s have emerged as viable alternatives to costly proprietary and commercial products. Open source software’s of LMS are gaining popularity. Some of the OS softwares are Moodle, Drupal, Wordpress, Blackboard, Joomla etc.

#### VII. RECENT SUCCESSES OF OSS

Recently OSS has become more of a mainstream product with many companies adopting it in their offices. OSS market share has also grown rapidly making many companies sit up and take notice of the OSS phenomenon.

- IBM recently announced that the company would devote almost \$1 billion dollars to support Linux. (Burke, 2000)
- Forrester Research estimates that more than 55% of the world's 2,500 biggest firms use open source software, with almost a quarter using the software in production systems.
- Sun released Star Office, an office suite similar to Microsoft Office, under the GPL license.

To be sure, free/open source software still faces challenges. Both Red Hat and VA Linux, two of the most prominent corporate supporters of Linux, still lose money.

## VIII. OPEN SOURCE ADVANTAGES

### 9.1 Free Redistribution

The license does not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale.

### 9.2 Source Code

The OSS includes source code, and allows distribution in source code as well as compiled form.

### 9.3 Derived Works

Most license allow modifications and derived works, and allow them to be distributed under the same terms as the license of the original software.

### 9.4 No Discrimination against Fields of Endeavour

The license does not restrict anyone from making use of the program in a specific field of endeavour. For example, it may not restrict the program from being used in a business, or from being used for genetic research.

### 9.5 Is not specific to a Product

The rights attached to the program does not depend on the program's being part of a particular software distribution. If the program is extracted from that distribution and used or distributed within the terms of the program's license, all parties to whom the program is redistributed have the same rights as those that are granted in conjunction with the original software distribution.

### 9.6 Not Restrict Other Software

The license does not place restrictions on other software that is distributed along with the licensed software. For example, the license does not insist that all other programs distributed on the same medium must be open-source software.

### 9.7 Technology-Neutral

Most of OSS is technology neutral.

## IX. POPULARITY OF OSS

The basic idea behind the open source movement more people look at the code on the internet, more people read it, more people adapt it, more people redistribute it, more people fix bugs and slowly you have this huge community of people which cannot be employed by a single corporation, all working for the same project and churning out new software at speeds which never could be achieved by conventional software development team. Cost is another factor. OSS is available for download freely. As compared to General idea of open source software, the initial cost of OSS is zero which is very attractive for start-ups and small businesses. Other advantages include reliability and a large number of software engineers working round the clock at no cost to the user. The contributors to the project could be distributed at different corners of the world making support and fixes available round the clock. The other reason for using OSS could be purely based on principles that open source is an ethical business model, all the source code is in the open and none of it is concealed and is good for the client or the user of the software to modify to fit them to their needs rather than depend and wait for the seller of the product to make changes.

## X. GENERAL IDEA OF OPEN SOURCE SOFTWARE

When we talk, in English, about 'free software', there is a dangerous ambiguity, due to 'free' meaning both 'freedom' and 'gratis'. Therefore, in this document, we will use mainly the term 'open source' when referring to users freedom of use, redistribution, etc., and 'gratis software' when referring to zero acquisition cost. The use of the Spanish and French word 'libre', by the way, has been adopted in many environments to refer to open source software, but will not be used here for the sake of uniformity.

## XI. RECENT SUCCESSES

Recently OSS has become more of a mainstream product with many companies adopting it in their offices. OSS market share has also grown rapidly making many companies sit up and take notice of the OSS phenomenon.

- IBM recently announced that the company would devote almost \$1 billion dollars to support Linux. (Burke, 2000);
- Forrester Research estimates that more than 55% of the world's 2,500 biggest firms use open source software, with almost a quarter using the software in production systems. (Connor, 2000);
- Sun released Star Office, an office suite similar to Microsoft Office, under the GPL license; To be sure, free/open source software still faces challenges. Both Red Hat and VA Linux, two of the most prominent corporate supporters of Linux, still lose money. Microsoft has various code sharing licensing methods under “Shared Source Initiative” program which details around 22 different licenses tailored around different industries and products.

## XII. WHOM OPEN SOURCE WILL HELP

Open source is not something you adopt because it is cool and everybody else seems to do it. If your business is doing absolutely fine in the sense that there are no existing issues that cannot be solved and no future opportunities that cannot be grasped within the traditional software development model, then the open source is probably not for those people.

### 12.1 Resource Constraint

Businesses usually do not have unlimited resources devoted to their endeavours. It is the responsibility of the management to optimize the allocation of limited resources to maximize profit. Constraints of resources surface in various forms. It prevents a great idea from turning into reality; it hampers the quality of the product due to the lack of proper quality assurance; it forces companies to continue sustained engineering for years to come instead of targeting new markets and developing new products. Open source strategy allows you to enlist an army of outside developers and testers to work for you “for free”.

### 12.2 Strategic competition

When well managed, open source projects can change the landscape of the competition in a way the traditional software. First, there is the “free” factor. By offering a piece of software for free, the usage increases. Then there is the “peer” factor, which further increases the popularity of the free software. When something is popular, wonderful things happen. What you end up with is a product that packs increased functionalities, meets the customer requirements more precisely, is more reliable and integrates well with the rest of the world. In the end, your product competes better among peers, which in turn improves the standing of your company and motivates your employees to produce even better products. If your business faces similar challenges and these benefits are of interest to you, there are a few aspects of open source software that you need to evaluate in order to come to the conclusion whether open source is the right solution.

## XIII. CONCLUSION

In recent years, numerous open access software's have emerged as viable alternatives to costly proprietary and commercial products. Open source software's are gaining popularity. PS and OSS have advantages & disadvantages depending upon the application features they provide. PS have a long lasting stand in the computing era. Open source is fast gaining acceptability and is the future of e-learning. Many world class institutes have adopted Open source. We conclude with the following excerpt from The Guardian, 28 March 2013 (4). “There are clear cost savings available to adopting open source solutions. The University of London Computer Centre hosts Moodle for 2 million students across 150 UK higher and further education providers, many of whom have migrated from a closed source solution....”

## REFERENCES

- [1]. Coppola C, Neelley E (2004).
- [2]. Coppola C, Neelley E (2004).
- [3]. Williams, Roy (2003). 2nd European Conference on E-Learning Glasgow Caledonian University, Glasgow, 6–7 November 2003. Academic Conferences Limited.)
- [4]. Scott Wilson, Open source in higher education: how far have we come? (the guardian, 28 March 2013)