FREE AND OPEN SOURCE SOFTWARE IN EDUCATION

INTRODUCTION AND ABSTRACT

On 21st November 2006, John Pugh, MP raised an Early Day Motion (No. 179 "Software in Education"), to raise concerns about the behaviour of Becta and the Department for Education and Skills who are effectively forcing schools to purchase expensive proprietary software tools and thereby preventing them from benefiting from free and open source software alternatives.

This document aims to summarise some of the benefits referred to in the EDM. An Annex describing some of the most important free and open source software packages available is also provided, including further information about the software tools mentioned in the text.

Free and open source software is provided with a licence that allows software developers the freedom to examine and modify the source code of the program and then distribute the result of their work. This feature means that users of the program can generally obtain the software for little or no charge and are then free to copy it as many times as they like to use on their own or their organisation's computers. Much of the software available in this way is of a very high quality and can bring many benefits to schools and colleges.

Table of Contents

1 THE NATURE OF FREE AND OPEN SOURCE SOFTWARE

1.1 Definition

The free and open source software methodology provides programs in which the source code is available to the general public for use and/or modification. Open source code is typically created as a collaborative effort in which programmers improve upon the code and share the changes within the community.

The licence for the software defines what the user may do with the code and is built on the legal definition of Copyright, which is recognised Worldwide. There are many licences, (see http://www.opensource.org/licenses/), but all have a common thread; the licence provides more rights to the user than basic Copyright law would allow, not less.

Some of the most common Licenses are the GNU General Public License (GPL) (see http://www.gnu.org/copyleft/gpl.html) and the Berkeley Software Distribution (BSD) (see http://www.freebsd.org/copyright/license.html). The additional rights given by the licences are often known as the 'Freedoms', because they provide the user with the freedom to view and modify the source code and therefore learn from the experience.

The presence of the 'Freedoms' in free and open source software, leads to the use of the term Free/Libre and Open Source Software (FLOSS), to show that the code is free as in freedom, but not necessarily free as in cost.

1.2 Explanation

This ability to learn when using free and open source software is often likened to the open nature of science, where papers written by one scientist are peer reviewed by other scientists all over the world. In this way science grows by 'Standing on the Shoulders of Giants' (paraphrased from Sir Isaac Newton's letter to Robert Hooke in 1676, (see http://en.wikipedia.org/wiki/Standing_on_the_shoulders_of_giants). In the context of this discussion it is interesting to note that Sir Isaac borrowed the phrase from earlier writers. Free and open source software grows because programmers from all over the world contribute to the code, improving its functionality, reliability and security.

This methodology promotes:

- Learning and sharing in an inclusive environment.
- Best of breed developers.
- Innovation.
- The creation and use of Open Standards.

Although community based, the environment works in much the same way as free and open markets do. As a result, prices fall, benefiting consumers.

In contrast, proprietary software is kept closed with only the vendor's programmers able to see it. Although this allows the vendor to make more money, it denies to the code base the benefits of having many eyes to improve it. Many who oppose free and open source software, claim that opening the code denies them their livelihood. However, there are many commercial organisations who make a great deal of money from free and open source software by providing consultancy, training and support. The standard business model for open source companies is to separate the software from the rest of the package and sell everything except the software, (eg installation support, documentation, defect resolution, education, warranty, indemnity and ongoing support).

Additionally, a company whose business model depends on selling copies of programs to sustain its revenue is forced into a never ending cycle of updates to encourage the users to upgrade. Very often these upgrades include hundreds of new features which have little value to the average user (this is known as 'feature-bloat'). However, the user still has to upgrade, because the new software comes with new file formats which the old versions cannot read. (This topic is covered in more detail under the heading of Open Standards below). The result of this is 'vendor lock-in', which is precisely what the supplier is trying to achieve; in extreme cases the result is a virtual monopoly, where consumers have little or no choice and prices are artificially high. In any case, selling software is a poor way to do business as a large proportion of the sales needs to be spent on ongoing support. This usually means insufficient money is allocated to support and users get a poor service.

Free and open source software does not depend on continuously selling new versions or vendor lock-in, because revenue is generated by selling consultancy, training and support. This requirement is naturally ongoing and so there is no benefit from developing software that is incompatible with older versions, so all versions interoperate seamlessly.

2 PRODUCERS OF OPEN SOURCE SOFTWARE

Free and open source software is produced by a rich and diverse group of programmers, who generally fall into the following categories.

2.1 Enthusiasts

The programmers who develop free and open source software come from many walks of life and it is usually the enthusiasts who start the ball rolling. Detractors of free and open source software like to portray the programmers as 'amateurs' who beaver away in their bedrooms into the small hours, producing crude and unreliable software. As with any community based movement, there are some people who fit that description, but the community is a meritocracy and so contributions which do not make the grade are not accepted into projects of any size. Linus Torvalds, (see http://en.wikipedia.org/wiki/Linus_Torvalds), the Finnish programmer who initiated the Linux Project in 1991, was a bedroom programmer but his idea now powers much of the internet. He is still the lead maintainer of the Linux kernel (the heart of the operating system) but he is now paid for his efforts by Open Source Development Labs, (see http://www.osdl.org/), a consortium of commercial and non-commercial organisations who benefit from Linux.

2.2 Companies/Corporations

Once a project gains traction, it often attracts the attention of commercial organisations. Examples of large companies that have recognised the value of free and open source software include Sun, Red Hat and IBM. Apart from using the software within their own organisations, or selling support services to other organisations, these companies contribute code back to the community, generally by paying programmers among their employees to work on free and open source projects.

Not all free and open source software is originally written by enthusiasts. Two of the most well known open source software packages, the office suite OpenOffice.org and the web browser Firefox were originally closed source, which were opened by the companies that owned the code base in order to take advantage of the benefits provided by the open source community.

Note that only about 5% of software programmers are employed developing software that is sold. The rest write and support software that is only used inside organisations or is integral to a hardware product. Adoption of free and open source software will not mean current programmers will be left with no work.

Although the companies mentioned above have a high profile and names which are well known, there are thousands of smaller companies throughout the world who provide support to free and open source software; these companies include IT Consultancies, Web developers, ISPs and many more. Within the group of IT Consultancies there is a sub-group that specialise in educational tools. In the UK, companies such as The Cutter Project (see http://www.cutterproject.co.uk) and Sirius Corporation (see http://www.siriusit.co.uk/), help schools set up their IT infrastructure and make the best use of the tools available from free and open source software. An umbrella organisation has been set up, (the Open Schools Alliance, see http://www.openschoolsalliance.org/), to provide support and to campaign for free and open source solutions and the providers.

2.3 Users

Everyone who uses free and open source software is in the Users class, but clearly, not everyone has the necessary skills or ability to write software. The community still benefits from thousands of such users however, who contribute bug reports, documentation, artwork and advocacy, without being able to program a single line of code.

Many of those users come from within the educational community and have contributed to such projects as the Edubuntu Linux distribution, aimed at educational establishments, and the online learning tool Moodle.

3 BENEFITS

All users of free and open source software, whether they be in the education sector, in commercial organisations or at home can benefit from the cost savings and improved quality brought about by the development methodology. One major advantage is that the security issues that beset much of the software produced by proprietary vendors, such as viruses and spyware, are rare. In addition, users in the education sector enjoy a number of other benefits.

3.1 The Student

In the context of education, the main users are the children. Children come in all shapes and sizes and their abilities vary just as much, but many of the brighter ones are born with an urge to learn. Free and open source software can provide the tools to feed that urge.

A good example of this are the kids who are attracted to computer programming. A generation or so ago, there were few colleges or organisations training software engineers and as a result many programmers were self taught, especially in the UK, by having access to low-cost platforms such as the Sinclair series of computers, or the BBC Micro. The reason that these platforms promoted this learning process was that they all came with simple programming capability, generally the language BASIC. It was natural for young people to satisfy their curiosity by 'playing with' these tools. In the 1980s, the UK produced thousands of world class programmers who had learned their basic skills on these machines and graduated to professional development.

Modern PCs are relatively low cost devices, but the software to run on them can cost far more than the hardware platform. The lack of bundled programming tools means that young budding programmers find it much more difficult to get into programming, but free and open source software provides access to a huge range of tools at no or low cost, including programming tools.

The benefits of free and open source software are not limited to aspiring programmers. There are now many alternatives to everyday tools on the desktop and the low cost of these tools (they are often free), mean that any student can obtain and run them on relatively cheap hardware, either on hardware at school or at home, (see below). Apart from OpenOffice.org and Firefox, some other examples of modern tools are the image painting program The Gimp and the desktop publishing tool Scribus. Even students who don't use free and open source software on their computers may benefit from it by using tools such as Google and Wikipedia, which are themselves free and open source.

3.2 The School or College

The first reaction is that the School or College gains financially, because free and open source software is 'free'. This is true up to a point, because there are no longer licence fees to pay, but it is not that simple. The licence generally permits providers of free and open source software to make a charge for it or, (more usually), charge for providing Training and Support. Even if the School's IT department downloads all of the software from the Internet, they may need a different skill set to administer and use it. This fact is often used by detractors of free and open source software to 'prove' that it is no cheaper than its proprietary equivalent, but this is disingenuous because users and administrators of proprietary software frequently have to retrain to use the new versions released to generate more revenue.

In reality, there are many resources available to IT departments to learn how to administer and use free and open source software. These include some of the commercial organisations mentioned in Section 2.2 above, but there are also free resources on the Internet that help staff progress up the learning curve.

Once established, the school or college can then calculate its ongoing costs on Training, Support and its staff costs, without having to pay costly licensing fees which are often so complicated that they are impossible to keep track of without yet another costly database. The software that they can then make available to Teachers and Students is then generally more reliable, more secure and more diverse than any of the proprietary alternatives and with lower ongoing costs. Less obvious is that the nature of free and open source software keeps IT technicians interested and is an important factor in retaining staff.

Money saved by reduced software costs can then be spent on other products and services, frequently from the local community and certainly providing greater benefits to the school and its students than would be obtained by paying licence fees to huge multinationals.

3.3 The Family

The family is often forgotten when IT decisions are made in schools. Many families quite naturally wish to give their children the best possible advantages in school and are prepared to make computers available to them so that they can study and complete homework assignments without being reliant on school facilities. However, this can also be divisive.   The cost of a relatively high specification PC has fallen to around £400 in recent years, but the software to run on it is no cheaper than it was ten years ago. Many families would be unable to afford the £1000 or so needed to buy and fully equip a PC with the same software as the school is using.

Student licences are made available, but the End User licensing Agreements (EULA) often have unreasonable terms and conditions. For example the EULA for the Student Edition Microsoft Office® 2000 was limited to the named student and so a family with four children needed four licenses if they were to comply with this requirement. Since the Student Edition currently costs nearly £100, this software alone would rival the cost of the PC for a family of four (and the parents would be prohibited from using the software themselves by the terms of the licence (plus they may need four PCs if they wished to install all four copies)).

Note: The author is writing from his own experience of the Office 2000 student licence. The current EULA for Office 2003 may not have this restriction, but experience has shown that there are frequently quite onerous privacy and usage conditions in proprietary licences.

Free and open source software may be installed as many times as needed, for no additional charge, (assuming a charge was made in the first place). Furthermore, free and open source software is generally free from feature-bloat and so will run on computers that are not particularly high specification. Poorer families are much more likely to be able to afford to provide computers for their children, reducing the divide between the children of rich and poor parents. The school can even make copies of the software they use available to parents or pupils, to avoid every family having to find the same software for themselves. If cost is an issue, they are legally allowed to charge for a cheap CD-R or alternatively the parents can supply one in exchange for the copy.

4 OPEN STANDARDS

Proprietary software vendors have little incentive to meet any kind of Open Standard, because that would interfere with their business model which relies on 'encouraging' their users to repeatedly upgrade. If a proprietary file format is created (and kept hidden), then users are forced to upgrade in order to interoperate. A further issue is that if the proprietary company then decides to stop supporting that version, or goes out of business, then in a relatively short period of time (decades or even years) it is unlikely that there will be any software that can read the old files.

This has been an issue for many years, with some of the larger companies, (eg those who have a dominant market share), even going as far as 'hi-jacking' open or de-facto standards, building on top of them, and then releasing their own versions and then patenting elements of the extensions so that anyone wishing to write code to interoperate must licence the patents. Since developers of free and open source software are unable to purchase licences, they are effectively excluded from participation, as are all but the largest proprietary vendors. Users of that software then have to upgrade to the tune of the vendor, who by now now has achieved a monopoly.

Free and open source software does not work in this way. By contrast it makes little sense to do this and conformance with standards is the norm, not the exception. This means that it will always be possible to obtain support for a particular file format. Even if the format has been abandoned and the world has moved on, the specification will be public allowing any competent developer to replicate the capability on a modern system.

A good example of this is ISO/IEC 26300 or Open Document Format. Although only formalised in November 2006, many free and open source Office suites already support this. Microsoft however, consistently refused to support this for much of the time that it was being developed, claiming that it would 'restrict innovation'. As a result, many organisations (including governments) mandated only Open Standards for their documents, effectively excluding Microsoft. Their initial response was to try to formalise their own proprietary Office Open XML into an ISO standard through their membership of ECMA (an industry body), therefore trying to create two standards and defeating the whole object of the exercise. (This is also 'back-to-front'. Normally a standard codifies a specification, which is then used to implement products. The Office Open XML standard codifies the Office 2007 file format, making it difficult to implement for any organisation without access to the source code for Office 2007. The standard is 6000 pages long.)

In July of 2006, Microsoft announced that they were funding and supporting an open source plug-in for Office 2007 to support ODF. At the time of writing some progress has been made, but most reviewers are not happy with the results. At present the conversion between the two formats is particularly poor, with format and layout being seriously corrupted during the process. However, it is to be hoped that this can be improved with more work. What is of greater concern is that is that ODF cannot be set as the default format in Office 2007, so most documents would still be produced using Office Open XML. It is good to see that Microsoft are apparently supporting a free and open source project, but if they were serious about ODF, they would give it the same status as other conversions.

This is a particularly high-profile example of how proprietary companies ignore standards, but by no means the only one. The success of free and open source software depends upon standards, so this kind of lock-in is never an issue and schools and colleges know that material produced today will still be usable in five, ten, fifty or hundreds of years time.

Terry Coles, 7th January 2007

ANNEX

EXAMPLES OF FREE AND OPEN SOURCE SOFTWARE AVAILABLE TO SCHOOLS

Software	Website	Used by	Runs On	Accessible From	Comment
Linux	http://www.linux.org/	Students, Teachers	Various	-	The Linux Operating System can be installed on almost any PC as an alternative to Microsoft Windows® and can be used to provide both server room and desktop capabilities. Most of the software tools described below are available for Linux and it is used by many organisations, including governments, banks, large corporations, educational establishments and web providers. (A large proportion of the internet is powered by Linux.) The system was written as a replacement for Unix, which until recently, was the dominant operating system in corporations. It also uses some BSD code (see below).
FreeBSD	http://www.freebsd.org/	IT Departments	PC	-	FreeBSD is an Operating System based on the Unix-like code written by Berkeley University in California. It was based on elements of Unix that had been made available by AT&T and released under the BSD licence. There are also other OSs based on the BSD code and these are used in many server rooms, including Yahoo!
Ubuntu	http://www.ubuntu.com/	Students, Teachers	PC	-	A Linux 'Distribution' which includes the Operating System and most of the software packages (eg the Applications) needed to run a desktop computer or server.
Edubuntu	http://www.edubuntu.org	Students, Teachers	PC	-	A version of Ubuntu specially crafted to provide a great environment for use in schools.
One Laptop Per Child Project	http://www.laptop.org/	Students, Teachers	N/A	N/A	Project set up by the Massachusetts Institute of Technology to provide a specially designed Laptop computer for children in the developing world. It is based on hardware with wireless access to other laptops or a local server and

Software	Website	Used by	Runs On	Accessible From	Comment
					runs open source software assembled mainly by Red Hat.
					The aim is to get the cost per laptop down to less than $100, although there will be additional costs for infrastructure support. More information may be found at http://news.bbc.co.uk/1/low/technology/6224183.stm.
W3C software	http://www.w3.org/Status	IT Departments	Various	Windows, Linux, Macintosh	The web is the biggest and most open project in the world. The World Wide Web Consortium (W3C) makes all of its own software available through FLOSS licences. Their stated aim is to keep the web open and free from proprietary lock-in.
OpenOffice.org	http://www.openoffice.org	Students, Teachers	Windows, Linux	-	Office Suite with similar functionality to Microsoft Office. It provides a word processor, spreadsheet, presentation tool, vector drawing tool and database tool.
http://www.scribus.net/	http://www.scribus.net/	Students, Teachers	Windows, Linux	-	Desktop publishing tool with support and professional features.
The Gimp	http://www.gimp.org/	Students, Teachers	Windows, Linux	-	Image production and manipulation package.
Firefox	http://www.mozilla.com/en-US/firefox/	Students, Teachers	Windows, Linux	-	Web Browser with particularly good security and functionality
Moodle	http://moodle.org/	Students, Teachers	Windows, Linux	Windows, Linux, Macintosh	Tool for producing internet-based courses and web sites. Has over 150,000 registered users. Very successfully used by the Open University.
Apache	http://www.apache.org/	Students,	Windows,	Windows,	Web server. Used by around 60% of web sites on the internet. (For latest

Software	Website	Used by	Runs On	Accessible From	Comment
		Teachers	Linux	Linux, Macintosh	figures see http://news.netcraft.com/archives/web_server_survey.html)
Wiki	http://www.wiki.org/wiki.cgi?WhatIsWiki	Students, Teachers	Windows, Linux	Windows, Linux, Macintosh	The underlying software and philosophy that makes Wikipedia work.
OpenCD	http://www.theopencd.org/	Students, Teachers	Windows	-	Freely downloadable CD with over 30 free and open source programs for Windows.