Free and Open Source software in schools. A point of view based on experience.

September 2008

Introduction
We continue to believe that Free and Open Source Software (FOSS) offers compelling advantages in the schools sector. We have experience of advocating its use, of installing it and assisting staff to use it. We have moved from an evangelistic to a pragmatic approach based on real-life experience. We are, frankly, cautious about advocating its use in most environments based more on the reactions of staff than of pupils. This paper explains why we take that stance.

Background
Free and Open Source Software (henceforth "FOSS") is produced by many different groups of authors. Some are commercially oriented, some are highly skilled professionals working in their spare time, others are a mixture of both whilst yet other FOSS groups are disorganised or moribund. FOSS embraces many different communities with many different objectives.

The motivations behind the production of FOSS are complex and frankly not important in this context. From the user's perspective what results is

- A large range of software of various kinds
- Levels of quality ranging from astonishingly good through to poor
- Many packages of similar or equal quality to expensive commercial alternatives
- Software that can be freely copied, redistributed and shared at no cost
- Support from users and authors direct as well as commercial support from FOSS support companies
- In many cases, cross-platform software that work on a range of operating systems (e.g. Windows, Mac, Linux, Unix, various PDA and mobile phone systems)

One obvious aspect of most FOSS is that since it's cost-free it has no marketing budget and consequently those who want to take advantage of it need to do some homework. Waiting to have it "sold" to you does not work.

Common Misunderstandings
There are many misunderstandings about FOSS but the most common one is simply lack of understanding of how it is typically packaged and used. Broadly speaking FOSS consists of:

1. Many individual software packages which do a single job. Word processing, web design tools, databases, programming languages and thousands of others
2. Whole "projects" which tend to bundle together associated packages
3. Infrastructure components such as operating systems (think Windows or Mac with no additional software)
4. "Distributions" which select from the plethora of alternatives and put them together as a more or less coherent bundle.

If all you want is, say, a graphical editor or a word processor you would pick an individual package from 1 above. For an office automation suite (word processor, presentation package, spreadsheet) you would most likely use choice 2 and almost certainly would pick OpenOffice.org which most would agree is the best of breed in that application area. Typically you will be using a mainstream

operating system (“platform”) like Windows or Apple's Macintosh in which case you will look for applications that are supported on your chosen platform.

At the infrastructure level things become different. Here you are choosing not to use software as well known as Windows but instead are deliberately picking an alternative. By far the best known of these is Linux, but others do exist. If you do select an alternative platform such as Linux, you will most likely select it by way of choice 4 above, a distribution.

Distributions are usually very large aggregations of software all chosen to work together and bundled with installation programs so they can be installed on a bare PC. The best known are the various misleadingly named “Linux distributions”. The name is misleading since Linux typically is a tiny part of the distribution which then bundles together hundreds or thousands of other software packages all on the same DVD. Since this is easy to do, there are dozens of Linux distributions all with different choices of packages, installation methods and support services – some free, some paid-for. Mainstream examples of Linux distributions include SuSE, Red Hat, Debian and Ubuntu; there are however, numerous others. Their release numbers or names are chosen by those who package them and bear no relationship to one another.

Some distributions come in “live CD/DVD” form which allow you to run them without any need to install anything: the software runs direct from the CD/DVD drive. This allows you to dip your feet before taking the plunge of a proper installation; the drawback is that performance is usually decidedly sluggish compared to an installation on the hard disc.

Sources of Information

For the beginner it can be daunting to navigate the seemingly endless lists of software packages. If you start at one of the better known repositories of FOSS at Sourceforge (http://www.sourceforge.net) you will be baffled in seconds.

A better bet may be to obtain a free copy of a Linux distribution and try that out; unfortunately if your preference is for Windows or Mac software that does not help you much.

We think that one of the best approaches is instead to make links with similar people to yourself who are already further along the curve than you may be. A starting point would be, in the UK, Schoolforge (http://schoolforge.org.uk), Edugeek (www.edugeek.net) and take a good look at http://www.schoolforge.net for the international perspective.

Bear in mind that many of the people you will meet are likely to be extremely enthusiastic and pro-FOSS, but not all have real-life experience of deploying it in the server rooms or classrooms. As well as being inspired by that enthusiasm, try to keep your feet on the ground about what this may be like to use in your own environment and check that your interlocutor has experience to match the enthusiasm.

Claimed Benefits of FOSS

Many of the devotees of FOSS will claim benefits which go far beyond its low price tag. Some of these are hard to substantiate but we are prepared to argue that a strong case can be made for

- Low to zero cost
- Greater achievable reliability compared to commercial alternatives
- Much better support compared to commercial alternatives
- Greater freedom to choose when, if ever, to upgrade
- Reduced susceptibility to hacks, viruses and malware

- Ability to prolong the life of old hardware

There are many other lesser benefits argued but most boil down to improvements in flexibility, sustainability and cost.

FOSS in Schools

We continue to believe that there is a bright future for the use of FOSS in schools. When we first started evangelising the topic we took the view that a big bang was the way to go: replace the ubiquitous Windows with Linux and make sure that the software packages needed for the curriculum were bundled in. Having done this in a number of schools we note and are prepared to defend the following observations

- The students adapt with great ease
- The ability to use the same software at home and in school without needing licences proves to be very useful to students
- Staff tend to be problematic and typically the project founders on the rocks of entrenched attitudes
- Where there is a strong management lead and a determination to succeed, success does follow
- For all the claims of the curriculum to be agnostic when it comes to software, the reality is that there is an unhealthy presumption that well-known commercial software packages are available. It takes guts and determination to work around that.
- Support for FOSS is in practice less easy to obtain than for Microsoft Windows systems since fewer people are presently experienced with it

Reduced costs, higher reliability and availability, freedom from viruses and the ability to extend the life of older hardware do indeed seem to follow.

The factors most hindering wider acceptance of FOSS are, in our view, inertia amongst those with the longest experience of traditional software and platforms. Staff attitudes are often noticeably against change and manifest themselves as active or passive resistance ranging from grumbling to forthright antagonism.

We took the pragmatic view that the education of the students must be the most important thing. If the staff are not happy with a sea-change in platform and software then the educational outcomes are not going to be the best. There are some outstanding examples of schools that have taken the plunge and are doing well, but in the main few schools wish to be pioneers in this particular arena.

As a consequence, we no longer evangelise for a revolution of throw out the old, in with the new. Instead, we recommend that a careful appraisal of the school's budget, desire to break the mould, staff attitude and pupil needs is done. We think that in the majority of cases it's most likely that schools will choose to stick mostly with what they know but gradually to start introducing some of the better-known cross-platform software packages.

The desktop and server systems that we install in schools are specifically designed to be platform-agnostic. In other words, they don't care what operating system or software are in use by the school and support both proprietary Microsoft Windows and Linux alike.

Where it's desirable to run both in parallel or to offer students a mix of both, this is completely practical. Futhermore, Cutter Project Limited is specifically organised as a support company and we are happy and able to provide support for FOSS-based systems if we are involved in specifying and building them.

The Future

We think it's only a matter of time before FOSS becomes well-established at not only the application level but the platform level in UK education. The arguments in favour remain strong whilst the practical difficulties (some real, some perceived) are likely to fade with time as experience grows and people become more used to a wider ecology of software. Firefox, for example, is now well established as a popular web browser and is a poster child for FOSS. More people are starting to use OpenOffice.org and other FOSS applications on their home PCs.

Mobile phones and PDAs are introducing people to the concepts of alternative software platforms and different applications. For many young people the PC already looks strange and dated compared to their phones – and for them, the internet is the application. Anything with a web browser is all they need.

Where there is less inertia or vested interest to displace, FOSS has already made noticeable headway. A leading example is Moodle (http://moodle.org) which is often combined with Elgg (http://elgg.org). Moodle provides a very popular virtual learning environment and is the choice the Open University and many other educational establishments. Elgg provides social networking support to build around the VLE with blogs, file sharing, community and other support.

The focus is now moving away from traditional desktop environments and agility and flexibility are likely to be key requirements of any schools ICT infrastructure in future.

Recommendations

We would encourage anyone responsible for the procurement or specification of school ICT systems to investigate what is possible through FOSS. Schoolforge, EduGeek and others (BECTA to a limited degree) are reservoirs of information and experience that can be drawn upon.

Above all we would recommend a working knowledge of FOSS software and platforms so that any current purchasing decisions do not accidentally constrain or impose unnecessary costs on later choices. Building in flexibility now probably costs nothing and may be very welcome later.

We don't claim unique knowledge here. We can provide support and advice but FOSS is mostly a community led endeavour and the more you get involved yourself the more you are likely to get out of it.

Cutter Project Ltd
18 Lynn Road
Ely
CB6 1DA
Tel 0203 151 0196