

Open source is frequently cited as one of the most important movements in modern software creation. It is supported by the European Commission, the UK government and JISC, and almost every further and higher education institution makes use of open source software.

This briefing paper describes what the term 'open source' means; what the most common licences are; what the impact is on education and where you can get more information.

## What is open source software?

'Open source' describes computer software for which:

- The source code is available to the end-user
- The source code can be modified by the end-user
- There are no restrictions on redistribution or use
- The licensing conditions are intended to facilitate continued re-use and wide availability of the software, in both commercial and non-commercial contexts

In every other respect there is no difference between this and conventionally-licensed software. The key differentiator is the licence. The term 'open source' is reserved for licences which are certified by the Open Source Initiative (OSI) to meet the criteria of the Open Source Definition (OSD).

There are several other features which many, but not all, open source software products have in common:

- The cost of immediate acquisition to the end-user is usually minimal; this is because the right to freely redistribute the software makes selling licences for copies of open source software an unlikely business proposition
- The development methodology of open source projects shares many characteristics with Agile programming, in that releases are frequent, features are added quickly after customer feedback, developers are often distributed geographically, and formal management structures are limited
- Many, but by no means all, open source projects are created and sustained by informal communities of developers, users and evangelists, rather than commercial companies
- Open source projects often serve as apprentice opportunities for junior developers to rapidly learn their trade by engaging in real-world development.

It is important to understand that software which is developed by companies such as IBM, Novell or Sun using conventional methods is being released under an open source licence just as effectively as that developed by a loose-knit community of students working at night to improve their programming skills.

## Licences

At the moment there are more than 50 OSI certified open source licences. The following five are perhaps the most commonly used:

- The GNU General Public Licence (GPL)
- The GNU Lesser General Public Licence (LGPL)
- Modified BSD (Berkeley Software Distribution) Licence (new BSD)
- Apache Licence
- Mozilla Public Licence (MPL)

The difference between them is the extent to which they control the way the code can be combined with other software. At the one extreme, the BSD licence permits open source software to be merged with closed-source code and then sold under a conventional licence. At the other, the GPL licence insists that if the software is combined with other code then that too must be under a GPL licence.

The decision as to which open source licence to use on a new software project should not be taken lightly. In many ways it expresses and shapes the development goals of the project. Guides to the common licences are available from the OSS Watch website.

## JISC and open source advice

OSS Watch is a JISC-funded advisory service providing impartial guidance to the UK HE and FE community on all matters relating to open source. The website provides many resources including focussed briefing notes, studies, reusable teaching materials and a wiki. OSS Watch also runs conferences throughout the year for both the UK and international communities and provides a focus for open source development and exploitation within the UK.

[www.oss-watch.ac.uk](http://www.oss-watch.ac.uk)

## Open source use in education

Open source software is already in use in most UK universities and colleges, whether in the context of mail servers, teaching systems, scientific workstations, or student desktops. It is, however, not yet usually part of an institutional strategy.

Interoperability is the principal reason cited by educational institutions for considering the deployment of open source software. This is because open source development tends to support open standards. Additional factors that can give open source deployments an advantage in HE and FE institutions are:

- No licence surprises. Open source licences are free and perpetual, so a licence fee increase cannot happen
- No incentive for theft. There is pressure on students to have course software available at home, and this can lead to theft. With open source software students can use and copy the software legally
- The lack of secrets. The working of the software is available for anyone to inspect, something of great importance for security auditing

## Open Source Software

- The ability to tailor the system completely to local needs. Both open and proprietary software are typically customisable in a shallow sense - institutions can tailor the interface within the bounds given by the programmers. With open source, if an institution needs the software customised in ways not thought of by the programmers, they can have the program changed, either in-house or by a third party.

### Open source deployment and support

UK government studies show that open source software can, in some instances, be cheaper to maintain than proprietary software. Deployment of open source software should follow the same pattern of evaluation of needs, testing, acquisition from a supplier and so on as would be used with proprietary software. However, there are two additional factors to consider:

- There can be multiple, independent, primary vendors of open source software to choose between
- As well as the usual range of support options (relying on a primary vendor, dealing with a large general support company, employing a specialised consultancy, and developing in-house expertise), there is the additional possibility of bringing development of new features in-house.

The cost of deploying and supporting software, whether open source or closed source, is very often much higher than the simple cost of licence acquisition, and will often be the principal component of the total cost of ownership.

### JISC and open source policies

Strategic bodies within the UK are increasingly providing policies on the use of open source software within the public sector. The UK government published a policy on open source in October 2004, which sets out guidance for the exploitation of publicly funded software development. In response, JISC has published an open source policy to support development activities within its funded projects. It advocates the use of open source as the default for software development as well as providing guidelines on copyright, licensing, trademarks, patents and development practice.

[www.govtalk.gov.uk/policydocs/](http://www.govtalk.gov.uk/policydocs/)

[www.jisc.ac.uk/about\\_opensourcepolicy.html](http://www.jisc.ac.uk/about_opensourcepolicy.html)

### Open source as an exploitation model

HE institutions regularly engage in software development, and sometimes exploit the result by setting up a company to sell licences. There are also business models that have been demonstrated to successfully work with open source licensing and development methodologies. Sometimes this involves dual licensing, an approach which is only viable when the institution controls all of the intellectual property rights (IPR) in the software. The MySQL database is a good example of dual-licensing in the commercial world, and LAMS is a good example in the academic world. Shared development through the creation of consortia, as in the uPortal project, is another successful model.

Open source exploitation models are relatively new to HE knowledge transfer units. Finding a model that is right for a particular institution will take some effort. It is likely that institutions will want to take advantage of a portfolio of different open source licences for different types of project. Sometimes, for example, only part of a software development will be released as open source, while another part may be exploited using conventional licensing - this will have an impact on the type of licence chosen.

Open source exploitation should be a component in any institution's IPR exploitation policy.

HE and FE institutions also need to consider how their staff participate in ongoing open source development. Institutions that

deploy open source software in their infrastructure need to make clear how staff can contribute software code patches, for example, legally to the ongoing development of the software. Such contributions are not merely good open source practice - there are also clear advantages in terms of continuing professional development for staff, and enhancing the reputation of the institution.

### JISC and open source sustainability

All open source outputs require a sustainability model to allow users to confidently commit to uptake and use. Common models are:

- The community model: the costs of sustaining the product or service are covered by building a community of users and industry partners who agree to cooperate on development work and maintenance because of their shared interest in an extended life for the product. Products maintained in this way tend to have a wide applicability, such as Apache.
- The subscription model: users pay subscription costs to an external body in order to support central maintenance and support. SAKAI and Linux Red Hat software are supported in this fashion.
- The commercial model: users choose to adopt and pay for a 'commercialised' version of a piece of software, normally to gain guaranteed support, maintenance and service models.
- The central support model: a central body provides robust releases and support for open source products that are of strategic importance to its community. This is often an interim solution, whilst other sustainability models are under development. The UK OMII and the Globus Alliance are example of this model.

JISC is investing in research in to all of these sustainability options in order to identify the best route for each of its projects and services.

### Common misunderstandings about open source

Open source software is often confused with other software whose minimal cost of acquisition makes it appear similar:

- Software which follows the open source guidance, but whose licence has not been accepted by the OSI: open source is not achieved by self-certification
- Software released with source under special conditions, such as not allowing commercial use; this contravenes one of the fundamental criteria of the OSD
- Software which can be freely redistributed, but without source, or the right to modify or redistribute it; again, such software does not have an open source licence.

### References to open source initiatives

Open Source Initiative: [www.opensource.org](http://www.opensource.org)

Open Source Definition: [www.opensource.org/docs/definition.php](http://www.opensource.org/docs/definition.php)

Linux Red Hat: [www.redhat.com](http://www.redhat.com)

Apache: [www.apache.org](http://www.apache.org)

SAKAI: [www.sakaiproject.org](http://www.sakaiproject.org)

OMII: [www.omii.ac.uk](http://www.omii.ac.uk)

Globus Alliance: [www.globus.org](http://www.globus.org)

This paper has been written by Randy Metcalfe and Sebastian Rahtz of OSS Watch on behalf of JISC, edited by Nicole Harris and produced by the JISC Communications Team.

Alternative formats of the briefing paper can be found at: [www.jisc.ac.uk/publications](http://www.jisc.ac.uk/publications)