

# IPR Issues and Software: A Briefing Document

Naomi Korn, Professor Charles Oppenheim and Professor Sol Picciotto

In collaboration with JISC Legal and OSS Watch

(Sponsored by Ann Borda, JISC Programme Manager)

October 2007

---

## Context

*This research and development briefing paper comprises WP4.12 within the JISC IPR Consultants work plan for 2006-2007. It builds upon the outcomes from the US Licensing and Policy Framework summit for Open Source Collaboration in Higher Education Summit held in November 2006 attended by representatives from the JISC IPR Consultancy, OSS Watch Manager and JISC Legal. It also relates to current initiatives and concerns raised by the patenting of software associated with Blackboard/Web CT. The intended audience for this briefing paper includes JISC Programme Managers, JISC Funded Projects and staff within the broader HE/FE community.*

## 1. Software and IPR

### 1.1 Overview

Software in various forms will be frequently deployed, developed and shared within HE and FE contexts. Various types of software are likely to be used to support live, online and virtual teaching, learning, research and administration within departments, on an institutional level, as well as arising from national and international collaborations. Software development may also result from cross sectoral collaborations including those in partnership with business and commerce.

Software is legally protected by (i) a range of Intellectual Property Rights (IPR), and in many cases by (ii) contractual agreements. Provisions in contracts under which the software is acquired, sometimes referred to as licences, are likely to specify authorised uses and conditions for such use. In UK law they cannot exclude rights which are specified as mandatory (see below). However, such rights depend on legitimate access to the program, which in turn depends on compliance with the contractual licence conditions.

A computer program is automatically protected by copyright as a literary work. It may also be the subject of a granted patent, which in turn may or may not be valid.

Copyright places a number of restrictions upon use, which include:

- Permanent and temporary copying, either of the program as whole or a substantial part, either of the source or object code, including writing code in any language which is substantially based on it
- Communicating to the public (e.g., making it available on a website to other than authorised users) of all or a substantial part
- Making an adaptation, arrangement or other altered version, including a translation into another computer language; this includes repurposing of any elements, e.g. combination or incorporation with other software or other copyright works, e.g., to create a multi-media work or database
- Distribution or rental
- De-compilation of a program, unless it is for the purpose of creating an interoperable program
- Possessing or disseminating an infringing copy.

Copyright is an exclusive right, which means that unless (i) permission has been granted by the rights holder, or (ii) copyright in the work has expired, or (iii) an exception applies, it is an

infringement to commit any of these restricted acts. Due to the special nature of software, a number of specified exceptions and permitted uses are provided, which are discussed in more detail below.

If software is also protected by a patent, and if such patent is valid, it would be an infringement to distribute any software which performs a function covered by the patent claim, even if it is not copied (i.e. the software has been independently produced).

Items associated with the software are also likely to be protected by IPR, notably:

- a manual will be a copyright literary work
- icons and other pictorial elements are likely to be copyright artistic works
- a logo may be a copyright artistic work and/or a protected design and/or a registered or unregistered trade mark
- the name is likely to be protected against passing-off and other restricted uses as a trade-mark.

These rights apply regardless of the form of supply of the software, e.g. on a CD/DVD or by downloading, although the methods by which contracts are formed will differ (e.g. shrink-wrap, click-use). The validity of contractual provisions may depend on both general contract law (including those applicable to consumer contracts) and intellectual property law.

The potential liability for infringement of rights in software can be significant and so people working within the tertiary education sector need to know about them to ensure that risks are suitably considered, and appropriate steps are taken to mitigate them. Typically, the audience for this type of information will include:

- Funding bodies
- Users in Institutions, including academic staff, students and support staff
- Recipients of funding for software deployment and development
- Department Administrators
- Procurement Offices
- Audit and Assurance Offices
- IT Managers or technical staff
- Strategic IT decision makers
- Copyright Officers
- Knowledge Transfer and commercial departments

### *1.2 Copyright in Software and its Ownership*

Software is automatically protected by copyright, as soon as the code is expressed in a material form such as writing. The first owner of the copyright in software is its author, except that in the case of software created during the course of employment, it is the employer rather than the employee. Hence, software created by students<sup>1</sup>, freelancers and contractors and sub-contractors, will belong to the creator, and not the HEI or FEI. It is important to note that the default positions stated here can be varied by a valid agreement.

In the UK, copyright in software lasts for the lifetime of the author plus 70 years from the end of the calendar year in which they died. Duration of copyright will however, vary based upon the following:

- Code created by multiple authors, each having created a distinct section of the code on their own: the copyright in each section expires separately based upon the lifetime of the author plus 70 years from the end of the calendar year in which they died.

---

<sup>1</sup> See <http://www.jisclegal.ac.uk/publications/studentipr.htm> for more information

- Code created by multiple authors where their contributions are not distinct: copyright in the entire work expires 70 years after the end of the calendar year in which the last surviving author dies.
- Employee-created code which does not have identified author(s), such as the code for MS Word: the duration of copyright will last for 70 years from the date of issuing of the code to the public.
- Employee-created code with identified author(s): the duration of copyright will be based upon the life plus 70 years of the employee(s) as above.

Any substantially new version of a computer program is a new copyright work. Thus, in view of the rapid cycles of development and obsolescence in the software world, any software of current interest will be protected by copyright. Criteria for protection include the requirement for the code to display originality, i.e. not to have been copied from or be substantially based on pre-existing code.

The EU Software Directive 1991, implemented in the UK by the Copyright (Computer Programs) Regulations 1992, which amended the Copyright, Designs and Patents Act 1988, confirmed that copyright protection as a literary work applies to a computer program as well as preparatory design work for a computer program.

It also provided a number of exceptions to the copyright restrictions, which allowed certain uses and interactions with software without the requirement to seek permission, some of which are mandatory (i.e. cannot be overridden by contract). These exceptions are:

- *Right to make a back-up copy*  
A lawful user may create a backup copy which is necessary for the purposes of such lawful use; this exception cannot be over-riden by any contractual agreement. However, a backup may not be considered necessary if the program is supplied in a sufficiently robust medium such as a CD/DVD.
- *Right to observe, study and test the program*  
A lawful user may observe, study and test the functionality of the program, or ideas and principles which underpin it, as long as these activities are performed during the course of loading, displaying, running, or storing the program which the user is entitled to do. Like the previous exception, this cannot be over-riden by a contractual agreement.
- *De-compilation right*  
This exception permits de-compilation of a program, i.e. conversion of a program expressed in a low-level computer language into a higher level language, or in the course of so doing copying it, **provided that** this is done in order to create an interoperable program.

This exception cannot be over-riden by contract. However, it is subject to a number of conditions which must be satisfied to prevent any such activities from being potentially disputed by the rights holder. These are:

- Any de-compilation activities are carried out by a lawful user;
- Such de-compilation is necessary in order to obtain the information necessary to create a program which is interoperable with that or with another program, and the information is used only for that purpose;
- Such necessary information is not readily available to the user in some other way, e.g. has not been supplied separately by the rights-owner;
- De-compilation must be strictly limited to what is necessary to achieve the permitted purpose, e.g. it is confined to those parts of the original program

- which are necessary to achieve interoperability, and the information is not supplied to anyone who does not need it for the permitted purpose;
- The information is not used to create a program substantially similar in its expression to the program which has been decompiled, or to do any act restricted by copyright.

### 1.3. Computer-Generated Works

Where a computer or other machine is involved in the production of a work, whether to produce a computer-assisted or an entirely computer-generated work, some skill or labour by a person must be shown in order for the work to be protected by copyright. If this can be shown in relation to the production of the work itself, even though assistance from a machine is involved, the person would be treated as the author. If the work is entirely computer-generated, such that there is no human author, the author is considered to be the person by whom the arrangements necessary for its production are undertaken, which might be a legal or natural person.

### 1.4 Software patents

Software and its underlying code, which satisfy the requirements of being novel and inventive have the potential to be patented in the US and a few other non-EU states. Patenting requires a long and expensive registration process, as well as extensive checks, to check whether it is the subject of an existing patent or an application for a patent, and that it has not been described in publicly available literature prior to the application date for the patent.

In the UK and other EU states, software “as such” cannot be patented. However, if that software is incorporated in some invention, then it can be patent-protected. So, e.g. a machine tool controlled by software may be patentable, in which case both the machine tool and the incorporated software are protected.

In the USA, not only is software as such patentable, the patent office has also been very generous in granting such patents, especially by applying a low standard of novelty. It has thereby allowed many patents which many consider should not have been granted and the validity of which may not be upheld either on re-examination, or by the courts. Anyone can request that a patent be re-examined, although the request is likely to be rejected by the relevant patent office unless it is backed with substantial supporting information. It should be noted that both in the US and the UK, the grant of a patent is not a guarantee that it is valid. The patent owner must challenge any articles of commerce which may be regarded as infringing by bringing a legal action in the courts, which may ultimately decide that the patent was not validly granted.

In the UK and Europe, a much harder line has been taken on software patents, both on the “as such” issue and to prevent non-novel software from being patented. Despite pressure from some software companies, there is no sign of the UK or its EU partners softening their line on this, although the issue remains a live one.

If granted and valid, the patent owner has a monopoly right to prevent third parties from making, using, selling or importing any articles which fulfil the functions described in the patent claim, during the patent’s lifetime. In most countries, including the UK, the maximum lifetime is 20 years from the earliest application date. After that, or earlier if the necessary renewal fees are not paid, the patent expires and anyone can make, use, sell or import the invention without fear of being sued for patent infringement<sup>2</sup>. In addition, the owner can, and often does, license rights under the patent to third parties for a fee. Where a product requires use of patents owned by different persons, it is common to enter into cross licensing agreements. Patents are geographically limited, so to get protection in multiple countries, multiple patent applications have to be made. There are complex international agreements governing how this should be done; details of these are beyond the scope of this report.

---

<sup>2</sup> But note that software might be the subject of both a patent and of copyright. Even if the patent has expired or an application for a patent has been refused, the software will still be protected by copyright.

JISC projects may well encounter software patents in two regards:

- Consideration should be given to patenting (especially in the USA) very successful software, particularly if it is tied to some process.
- There is a risk of infringing a third party's patent in the making, using, licensing or selling of software, or services with software embedded in it, in the USA or other countries where software patents apply, if the software is the subject of a patent.

Patents have wider application than copyright because whilst to prove copyright infringement, one must demonstrate that the copyright work was actually copied, to demonstrate patent infringement one simply has to demonstrate that the infringer was making, using, etc. the invention, even if that invention had quite independently been developed by the infringer, without any conscious or unconscious copying. There is therefore an obligation on an innovator to check existing patents before making, using, selling or importing an invention. This is much easier today as there are numerous patent databases, many of them free of charge, available for searching online. JISC is currently exploring the funding of projects to investigate the potential for specific repository related prior art databases. It is important to note that damages and costs in patent infringement cases can run to millions of pounds, which may be a combination of damages: including your own legal costs and a contribution to the costs of the plaintiff. The calculation of these damages is usually based upon the lost income to the patentee. Punitive damages are rare. However, there are important reasons for paying attention to patents, such as reputation and certainty, in particular.

There are defences against an infringement action, especially that the patent is invalid and should never have been granted in the first place, because, e.g., the invention was known before the application date or the patent owner did not disclose all he/she knew about the invention at the time the patent was applied for, or the making and using is for small-scale research purposes, or the making or using is with the permission of the patentee, or the invention was stolen from the person accused of infringement.

Software patenting caused concern in the HE/FE community when Blackboard/WebCT patented some structural elements of their technology and subsequently started legal proceedings against a competitor for patent infringement. There were fears among the VLE user community that their use of their chosen systems would be disrupted by Blackboard's actions. Such fears applied equally to open source and proprietary systems. Whilst there is a measurable risk of such patent infringements as there is no reason to believe that Blackboard/WebCT will not defend its patent where necessary, it has stated that it is unlikely that it will pursue UK academic institutions.

Certainly, it seems unlikely at this stage that the criteria for patentability will be extended to routinely cover software in the UK, particularly following the recommendations within the Gowers Review of Intellectual Property, as well as steers from the EU. However, the risk is unlikely to remain low if a patent were to be granted in the UK and EU, or software incorporating patented elements is used within the context of international collaborations in jurisdictions where Blackboard has a patent.

### *1.5 Registered designs and trademarks*

Computer icons, trading names and project names, software fonts, software-related peripherals, such as dongles and on screen displays are eligible for registered design protection as well as possible trademark protection.

Different requirements must be met in order to register a design. However, due to the amendments introduced to comply with the European Directive, the definition of designs that can be registered, has in some respects been expanded, e.g. some spare parts may be registered if they are 'complex products', provided they are visible in use (e.g., hub caps, car doors). Registered design right subsists for five years from the date of application (which is deemed to be the date of registration if registration is granted), and may be extended for up to five periods of five years.

Trade marks are symbols, shapes, words etc., associated with particular goods and services and which are used in the course of business or trade. Whilst there is some protection for unregistered trade marks (often called trade names) under UK common law, the strongest protection for such marks is obtained by applying for and obtaining a Registered Trade Mark. As with patents, this involves making a formal application and paying fees at various stages. Unlike patents, it is necessary to demonstrate that the trade marks are well known to the public, or a section of the public, before they can be registered. They must be distinctive and not confusingly similar to any other existing trade mark or trade name. Certain types of symbols, e.g., Red Cross, Olympics, and certain names, e.g., Royal, cannot be registered.

One can register trade marks for services, such as search services, as well as goods. Registered Trade Marks have to be kept in use and renewal fees have to be paid for them to continue, but as long as these conditions are fulfilled, a registered trade mark can last for ever. The owner can sue any third party who uses the mark "in the course of trade" and without permission. Whilst patents and registered designs are limited by geography, registered trade marks are limited by both geography and by class of goods/services. Thus to get extensive protection, the owner has to make separate applications in many countries, and for many classes of goods and services.

Many JISC projects will encounter trade mark issues; for example, service names may well conflict with existing trade marks, or successful projects and services may well wish to protect the name of that project or service. Since trade mark infringement can be inadvertent (i.e., without knowledge of an existing trade mark), advice should be sought from a reputable trade mark agent in all such cases. There may be a number of other types of outputs which are generated by JISC-funded projects, as well as designs by third parties which are used or incorporated into JISC-funded projects. These rights need to be identified and measures employed to ensure that they are managed appropriately.

Trade marks may also play a role in the release of JISC-funded software under open source licences (see below). A standard method of exercising control over an open source licensed software product is to obtain a trade mark on its name and logo. While others can adapt and release the code itself, they would be restrained from describing and labelling such code using the trade mark.

#### *1.6 Databases as Copyright Works, and under the Database Right*

The Copyright Designs and Patent Act 1988, defines a database as:

- `a collection of independent works, data or other materials which –
- (a) are arranged in a systematic or methodical way, and
- (b) are individually accessible by electronic or other means.'

Since this covers all types of works, a multi-media work as a whole may be protected as a database, if the component parts can be accessed individually. However, it does not include a computer program used in making or operating databases.

A database may obtain full copyright protection, if by reason of the selection or arrangement of its contents it constitutes the author's own intellectual creation. Thus, it is likely to have been created by one or more individuals, rather than being computer-generated. If the data-base does not fulfil this test, it may nevertheless qualify for protection under the special database right. It may also be protected both as copyright and under the database right.

The database right applies where there has been 'substantial investment in obtaining, verifying or presenting of the contents of the database'. The investment must be made in 'obtaining, verifying or presenting' the contents, and not in their creation.<sup>3</sup> Investment includes financial, human or technical resources. The maker becomes the first owner, and is defined as the person who takes the initiative in obtaining, verifying or presenting the contents of the database and assumes the risk of investing in those activities. A database made by an employee in the course of an employment

---

<sup>3</sup> *The British Horseracing Board Limited v William Hill Organisation Limited*, decision of the ECJ, Case C-203/02, November 9, 2004

belongs to the employer, in the absence of an agreement to the contrary. The maker must be a national, habitual resident, or legal person formed under the law of, an EEA<sup>4</sup> member state, or a state which provides equivalent protection.

This right is much shorter than full copyright: it extends for only 15 years from the end of the calendar year the database is completed, or made available if that is later. However, if there has been substantial change to a database's contents or their arrangement, which has involved a substantial new investment, it becomes a new database and qualifies for a new term of 15 years. As long as such substantial changes continue, the database right can extend indefinitely.

The database right prohibits the unauthorised extraction or re-utilisation of the whole or a substantial part of its contents. Repeated and systematic extraction of insubstantial parts may amount to a substantial part. No other guidance has been given yet as to what amounts to a substantial part of a database.

There are a number of exceptions to database right, the most important of which is the right to copy insubstantial parts of a database for the purpose of non-commercial research or private study.

JISC-funded projects need to be aware of the protection that is offered to databases, and ensure that any use that is made of third party databases is done either with permission from the rights holder, or within the parameters of what is legally permissible.

### *1.7 Moral Rights*

Whilst moral rights do not apply to computer programs, typeface designs, and any computer-generated work, they can apply to designs such as screen displays or works such as software user/reference manuals. It is therefore important to recognise that these rights may exist and in particular to take note of those that protect the author's right to be acknowledged, to object to false attribution and to object to derogatory treatment of the work. In the case of works created by employees, such as staff of HEIs or FEIs, acts authorised by the employer (who as explained above is the owner) do not constitute derogatory treatment; although if the employee has been identified as the author, there must also be a clear and prominent indication that the author has consented to the treatment. Under UK law, moral rights can be waived by appropriately worded contractual provisions.

### *1.7 Rights in confidential information*

Generally, unless permitted under an open source licence (see below) or other contractual agreement, the high level source code used to write software is kept confidential.

The obligation of confidentiality applies essentially to recipients of confidential information who are in a direct relationship. This would apply to data or information supplied explicitly on confidential terms; or if the method of supply implies confidentiality, e.g., if it is encrypted; or if access to commercially-sensitive material is obtained due to an employment relationship; or if data such as genetic information from samples are obtained without the permission of their owner.

## **2. Software and licences**

### *2.1 Overview of software licences*

Most rights-holders permit use of software by means of licence agreement. Licences can take several forms, ranging from those that permit the user to view, alter and redistribute the source code (Open Source Licences) to those which are much more restrictive (Proprietary Licences).

### *2.2 Open Source Licences*

---

<sup>4</sup> EEA = European Economic Area, comprising the European Union member states plus one or two other European countries

For OSS Watch<sup>5</sup>, the JISC funded service promoting an awareness and understanding of the legal, social, technical and economic issues that arise when educational institutions engage with free and open source software, open source software is defined as software released under an Open Source Initiative (OSI) certified licence.

Each of the licences approved by the OSI meets the conditions of the *Open Source Definition*<sup>6</sup>. That definition includes 10 criteria. The most important of these are the free redistribution of the software, access to the source code, and the permission to allow modifications to the software to be made and distributed. Various licences meet these criteria, including the GNU General Public License, commonly known as the GPL and the MIT licence, which is very different in length and intent from the GPL. Each licence is different and on some level will impose restrictions or obligations upon the user. Some licences, like the GPL, stipulate that modified versions of the covered software must be distributed under the GPL, if they are to be distributed at all. This stipulation is sometimes referred to as a 'copyleft' provision. Other licences, like for example the MIT licence, are 'non-copyleft' in that they do not stipulate a specific licence that must be used for distribution of modified works. Each licence needs to be read carefully to ensure that its terms can be satisfied and that it fulfills all requirements.

### 2.3 Proprietary licences

Microsoft licensed software is a good example of software which is distributed without the source code being made available. Generally, permission will be granted to the user via their "click-wrap" consent to terms and conditions, without which access is denied. This usually takes the form of a dialogue box containing the terms and conditions and a choice of 'Accept' or 'Reject' interface buttons. Other forms of agreement may include a licence on the back of box, terms and conditions on CD and/or DVD or in rare cases, consent to a written agreement. These licences may be supported by technological protection measures, the circumvention of which may itself constitute an infringement of copyright in addition to any infringement that eventually results from the circumvention.

### 2.4 Freeware

Freeware, as opposed to Open Source material does not as standard permit the user to access or alter the source code. The word "freeware" initially applied mainly to hobbyist-written software, but more recently has come to mean any software available for no cost. Examples of freeware in this latter usage include Adobe Reader, the Opera web browser and the Skype voice-over-IP client.

## 3. Checklist for dealing with IPR in software use and development

The layers of rights which are likely to protect software will each impose restrictions upon the user. It is therefore important that these and their characteristics are recognised to ensure that:

- Use of the software is legal
- Obligations to third parties are honoured
- Rights are protected
- Risks to an institution, its partners and funding bodies are minimised
- Ability to exploit rights in developed software is maintained
- Sustainability strategies can be realistically pursued.

To assist in this task, the check list below provides some useful points of reference:

- Carry out a regular software licensing compliance audit: familiarise yourself with the broad range of rights which are likely to protect the software that you wish to use and/or develop, as well as the legal ramifications and restrictions associated with each

---

<sup>5</sup> [www.oss-watch.ac.uk](http://www.oss-watch.ac.uk)

<sup>6</sup> <http://www.opensource.org/docs/definition.html>

- Maintain software and licence registers; there are several software products available for maintaining such a register.<sup>7</sup>
- Incorporate software development and use within your IP Policy and IT Terms of Use
- Assign staff to take responsibility for rights clearance and management
- Allocate appropriate budgets and other resources to rights and licence management
- Carry out prior art searches necessary for software development
- Check terms of open source and proprietary licence agreements to ensure that your required use is permitted. This is especially important when creating new software using pre-existing components.
- Contact OSS Watch for further information about Open Source Licences if you are considering permitting of use of developed software under an Open Source licence
- Negotiate with rights holders for any activities that are beyond the scope of the rights permitted and/or licence agreements which are necessary as part of your project
- Ensure the incorporation of contractual terms relating to ownership and use of IP generated by contractors and freelancers within agreements.
- Carry out risk assessments relating to the use of patentable software, particularly Blackboard/Web CT. Risks are likely to remain low unless a European Patent is granted to the company<sup>8</sup>, or you have international collaborations in jurisdictions where Blackboard has a patent. Other risks that will need to be considered include use of third party designs, trademarks and copyright materials. Bear in mind that making software available on the web may amount to distribution into many jurisdictions, for the purposes of patent litigation.
- Incorporate any identified risks within a risk registry and develop robust policies for minimising these risks.
- Consider the costs and benefits for registering further IP protection (such as patents and trademarks) for software and related goods and services developed in-house.
- Check the terms of any agreements and other contractual obligations such as consortium agreements relating to IPR and the stipulations relating to subsequent access to any software that you might be developing. For example, JISC has an Open Source Policy<sup>9</sup> which requires that any software which it has funded the development must be subsequently licensed for use under an Open Source Licence.

---

<sup>7</sup> See, e.g., <http://www.microsoft.com/uk/business/insights/successstoriesam.aspx>

<sup>8</sup> The company has applied for a European Patent, which is currently under consideration by the European Patent Office.

<sup>9</sup> [http://www.jisc.ac.uk/media/documents/funding/2006/09/jisc\\_osspolicy\\_may2005.pdf](http://www.jisc.ac.uk/media/documents/funding/2006/09/jisc_osspolicy_may2005.pdf)