THE PROCUREMENT DECISION - "OPEN" OR "CLOSED" SOURCE SOFTWARE?

CARLA MICHLER[*]

[*The merits of open source software, particularly in the context of Australian Government procurement, have been the subject of considerable debate in recent times. This paper provides an overview of the distinguishing features of open source software and considers the terms of the licence under which open source software is distributed. Consideration is also given to the merits of open source software through an examination of both the legal and technical risks associated with the product. In concluding, it is determined that there is a positive argument for mandating open source software as a viable alternative to its closed source counterpart.*]

I THE OPEN SOURCE MOVEMENT

In recent times there has been a growing trend for government departments and agencies on a worldwide scale to move toward open source software. The challenge for the Australian government is to explore, through an assessment of risk and benefits and drawing upon domestic and international experience, the availability of open source software solutions to realise the potential to achieve increasing value for money and greater operational activity across the public sector.

II DISTINGUISHING BETWEEN "OPEN" AND "CLOSED" SOURCE SOFTWARE

In order to appreciate the differences between "open" and "closed" source software it is necessary to understand what actually constitutes "software". Software is, in broad terms, constituted by two elements, namely the "source code" and the "object code". The "source code" and the "object code" refer to the "before" and "after" versions of a computer program that is compiled before it is ready to run in a com-

[*] Carla Michler is a lawyer who currently practices in the areas of corporate and commercial law.

puter. The source code consists of a set of human readable programming statements that are created by a programmer with a text editor or a visual programming tool and then saved in a file. The source code cannot itself be executed by a computer but is compiled with a specialised program called a compiler and the resulting output, the compiled file, is often referred to as the "'object code". The object code consists of a sequence of instructions that the computer's microprocessor can understand but that is difficult for a human to read or modify.

In a "closed source" or "proprietary" software model, software vendors retain the source code and sell or licence only the object code component of the program to the software users. Under the terms of the closed source licences, software users are permitted to run the object code but cannot view or modify the source code and, hence, modify the behaviour of the program without consulting with the software vendors. In this closed source model the source code can only be ascertained through recompilation or reverse engineering, however recompilation or reverse engineering of the object code is commonly prohibited by closed source licence terms. As a result, the user can only use whatever is provided to them by the software vendor, along with any modifications that they could prevail upon the software vendor to incorporate. The closed source model continues to be used by the majority of commercial software companies and is the most common software model adopted by both the private and public sector.

Open source software is based on a set of fundamentally different principles than closed source software and provides users with a greater freedom in the way in which they deal with the software. In general terms, open source software is software where the source code is freely distributed and widely available to users so it may be used, copied, modified and redistributed.[1] Open source software is licensed with certain common restrictions which generally differ from closed source software. Frequently, open source licenses require users who distribute open source software, whether in its original form or as modified, to make the source code widely available. The most common of the open source software licences, the GNU General Public Licence is considered in further detail below.

III OPEN SOURCE SOFTWARE LICENCES - THE GPL

Open source software is generally distributed under one of the main "public licences" such as the GNU General Public Licence (the "GPL"),[2] the Apache Software Licence or the Berkeley Licence.[3] The GPL is the most common licence under which open source software is distributed. The GPL allows the user to make verbatim or modified copies of the software provided that, if the user redistributes

[1] The full, formal definition of the term "Open Source Software" published by the Open Software Initiative is available at <http://www.opensource.org/docs/definition.html>

[2] A copy of the GPL is available *at* <http://www.gnu.org/licenses/gpl.html>

[3] Discussion of the full range of open source or "public licences" is beyond the scope of this paper. For further detail in this regard *see* Mark Webbink, *Understanding Open Source Software*, COMPUTERS AND THE LAW 20 (March 2003); and LEGAL ISSUES RELATING TO FREE AND OPEN SOURCE SOFTWARE (Brian Fitzgerald & Graham Bassett (eds.), 2003).

the software (either in original or modified form), certain conditions are met, including:

- a copy of the licence must be distributed with the software;[4]
- a copy of the source code must be included with any distributed copies;[5]
- any modifications made to the software must be identified and documented;[6] and
- the software must be licensed under the same terms under which the original licence was granted.[7]

Therefore, any user who has been granted the right to use, modify and redistribute the software under the GPL is required to extend those same privileges under the same terms to other software developers further down the line. The novelty (and what some may perceive as irony) with open source licensing is that the initial distributor of the source code controls its presentation and further dissemination through the law of copyright. In this process copyright law is used to create what is commonly termed a "copyleft" effect by mandating that the source code should be open and free for all to use in innovation and development of software.

IV BENEFITS OF OPEN SOURCE SOFTWARE

There is merit in the argument that open source software presents a number of benefits to the software user. In particular, one of the key benefits attributed to open source software is that it is significantly less expensive than proprietary software. The main factor that contributes to the price variation between open and closed source software is the lower licensing costs associated with open source.[8] This can be contrasted to closed source software which is usually accompanied by monopolistic licences that often attract high licensing fees.

However, when considering the cost savings offered by the open source model it is important for the procuring software user to bear in mind that low price does not necessarily equate to value for money. Accordingly, other price independent factors must be taken into account when assessing the benefits of open source software.[9]

[4] GPL, *supra* note 2, at Clause 1.

[5] *Id.* at Clause 3.

[6] *Id.* at Clause 2(a).

[7] *Id.* at Clause 3. An overview of the key issues relating to the GPL is available at <http://www.gnu.org/licenses/gpl-faq.html>

[8] Most public licences under which open source software is distributed mandate that the source code should be made available to users at a fee not exceeding the distribution costs associated with the software.

[9] Peter Williams, *Newham chooses Microsoft over Linux for cheap support*, available at <http://www.vunet.com/print/1157384> (last visited Aug. 17, 2004) and; For London Borough Council's assessment that (all factors considered) proprietary software presented a better value for money

Another benefit of open source software is the autonomy associated with procuring what is essentially "supplier independent" software. In an open source model the software user is at liberty to develop indigenous software from the source code; improve existing software by manipulating the source code; or develop the source code to best suit their requirements. The freedom offered by open source software is particularly beneficial from an economic perspective as opportunities are presented for multiple suppliers to bid for services, thus introducing competition which could, in turn, lead to further cost savings to software users. In this regard, the open source model avoids the potential "lock-in"[10] that can often be associated with relying on the one supplier to service and respond to the software user's software requirements.

It is further argued that the open source product offers a substantially more secure, stable and "bug free" alternative to closed source software.[11] The theory behind this argument is that the greater the "brain power" engaged in developing a program, the greater the ability to respond to, isolate and eradicate bugs inherent in the product. This position can be contrasted to closed source software where vulnerabilities are often more difficult to identify because the user does not have access to the source code and, once vulnerability is located, only the licensor has the power to remedy the fault and issue an update to the program. Therefore, open source software provides a substantial benefit to the user by having greater public scrutiny of the source code, faster release times and, if necessary, the problem can be fixed in-house, thus reducing vulnerability and enhancing security capability.

The reality is that while open source software has a number of benefits, it is certainly not free from risk. These risks are explored in further detail below.

V THE RISKS OF OPEN SOURCE SOFTWARE

Open source software is, in many respects, risky business. The risks inherent in the open source model can be classified into two general categories - legal and technical risk.

alternative than open source software *see* Peter Williams, *Microsoft and Government close to three-year deal*, available at <http://www.vunet.com/print/1157419> (last visited Aug. 18, 2004).

[10] The potential for "lock-in" in closed source software models was recognised by the Joint Committee of Public Accounts and Audit Report in its report JOINT COMMITTEE OF PUBLIC ACCOUNTS AND AUDIT REPORT, ENQUIRY INTO THE MANAGEMENT AND INTEGRITY OF ELECTRONIC INFORMATION IN THE COMMONWEALTH, 399 (2004) where AUUG argued that "software vendors may go out of business, may increase prices to an unacceptable level, or may decide that it is no longer in the business plan to support the software".

[11] *See,* Eben Moglen, *Free Software Matters: Security Through Freedom*, available at <http://moglen.law.columbia.edu/publications/lu-21.html>; and Deborah Halbert, *The Open Source Alternative: Shrink-Wrap, Open Source and Copyright*, 10 E-LAW MURDOCH UNIVERSITY LAW 4 (2003).

A Legal Risks

There are many legal issues surrounding open source software and, in particular, the licensing arrangements used to promote an open and free source code.

Unlike closed source software vendors, open source software vendors do not provide warranties or indemnities against intellectual property infringement - in fact, most open source licences explicitly disclaim warranties relating to software performance, merchantability and fitness for purpose.[12] The difficulty with this "blanket disclaimer" is that it falls foul of the Trade Practices Act, 1974, (Cth.) (the "TPA")[13] which explicitly sets out certain warranties that cannot be excluded by contract.[14] The provisions of the TPA are expressed to apply when a corporation or government[15] engages in "trade or commerce". Accordingly, the question arises as to whether the supply of software under an open source licence would satisfy the "trade and commerce" requirement and, as a result, invoke the provisions of the TPA. It is argued that the "trade and commerce" requirement would be satisfied in open source transactions where the software is supplied as part of a commercial dealing or if such supply is connected with advancing or protecting the commercial interests of the supplier.[16] Therefore, on the basis of this argument, the TPA could potentially apply to any commercial or related supply to customers despite any exclusion of warranties or liability that may be provided for in the terms of the open source licence.

It would be erroneous to assume that all open source licences are valid and enforceable. The current debate surrounding enforceability of open source licences focuses on whether the licence may be enforceable under contract law or in copyright law. One view is that the terms of the open source software licences are inherently non-contractual on the grounds that consideration for the grant of the licence is often not supplied.[17] The rationale behind this argument is that since the only promise that an open source software user makes is to redistribute under the GPL, if (and only if) they choose to distribute the derivative work, that promise is not sufficient and there is no consideration to support a valid contract. In the absence of a valid contractual agreement the question would then remain as to whether the licence remains enforceable through the force of copyright law. In considering this issue, it is important to understand that a licence operates to grant specified rights to the copyright owner rather than to grant the licensee a legal right to the property being licensed.

[12] *See,* GPL, *supra* note 2, at Clauses 11 & 12.

[13] Trade Practices Act, 1974, (Cth.). References to the Act apply to its equivalent state and territory fair trading legislation.

[14] The non-excludable warranties include warranties as to title and of quiet enjoyment TPA, s 69; fitness for purpose: TPA, s 74B; and that goods supplied will correspond with sample: TPA, s 72.

[15] TPA, s 2A provides that the TPA applies to the Commonwealth Government and its agencies, when either is carrying on a business, but only Commonwealth agencies can be fined or prosecuted.

[16] Peter C.J. James, *Open Source Software: An Australian Perspective,* in LEGAL ISSUES RELATING TO FREE AND OPEN SOURCE SOFTWARE 78 (Brian Fitzgerald & Graham Bassett eds., 2003).

[17] Ben Giles, *Consideration and the Open Source Agreement,* 12 COMPUTERS AND THE LAW 16 (2002).

Under copyright law, the terms of a licence can only grant the licensee certain rights that are within the ambit of the *Copyright Act* 1968 (Cth) (the "Copyright Act"). In this regard, consideration will need to be given as to whether the terms of an open source licence falls within the scope of the Copyright Act. There is no authority in Australian in relation to the enforceability of open source licences in this context.[18]

A further legal risk arises in circumstances where the licensee seeks to revoke the terms of the open source licence. In the case that only one single entity controls a significant portion of copyright in the source code for an open source software package, that entity may be able to terminate the licence and users will no longer be entitled to copy or redistribute the software. In this regard, it is important to recognise the potential danger that might arise if an upstream developer revoked the licences, causing all derived products to be rendered invalid to the extent that they are derived from the original. However, in the case that a licence is revoked there may be scope to argue that a software developer should be estopped from revoking the licence if the licensor mislead the licensee into relying on the continuance of the existing terms in the knowledge that the licensee would be detrimentally affected by that reliance if the terms were changed.[19] Notwithstanding this argument, the terms of the doctrine of estoppel have not been tested in the context of open source software and, therefore, it is unclear as to whether this doctrine could be invoked in the open source framework.

B *Technical Risks*

The main technical risk associated with open source is the risk that closed source systems will not be interoperable, or that open source systems will not be compatible with, a software user's existing software. Coupled with this risk are a number of inherent problems such as a lack of technical expertise and the absence of an appropriate support framework for the migration from closed to open source software. Software users would need to be confident that they have assessed the risks of moving from a single supplier product to moving to a more technically diverse product such as that offered by open source software.

[18] The most comparable case in this regard is the Federal Court of Australia's decision in: Trumpet Software Pty Ltd. v. OzEmail Pty Ltd, (1996) 560, which is authority for the proposal that licence terms of non-contractual software licences can be enforced. This case also supports the general proposition that a gratuitous licence can be revoked at will.

[19] *See* Walton Stores v Maher, (1988) 164 CLR 387 (High Court of Australia, 1988) regarding the doctrine of equitable estoppel developed to prevent a person from reneging on a promise if the other person would suffer loss as a result of not keeping a promise.

VI OPEN SOURCE SOFTWARE IN GOVERNMENT - THE AUSTRALIAN EXPERIENCE

The merits of open source software, particularly in the context of Australian government procurement, have been the subject of considerable debate in recent times. In September 2003 the *Financial Management and Accountability (Anti-Restrictive Software Practices) Amendment Bill* 2003 (Cth) (the "Bill") was introduced to the Commonwealth Parliament by the Democrats. The Bill aims to redress concerns that "a small number of software manufacturers have disproportionate and restrictive hold on the supply, use and development of software".[20] Under the Bill, the Government would be required to "have regard to the principle that, wherever practicable, an Agency is to use open source software in preference to proprietary software".[21] On 10 December 2003 the ACT Government passed the *Government Procurement (Principles) Guideline Amendment Act* 2003 (ACT) (the "Act") which is essentially a modified version of the Bill (and the first of its kind in Australia). Under this new Act, the ACT Government is required to consider use of open source software in its procurement processes and avoid software that does not comply with open standards.

In addition to the open source legislative movement in Australia, both the public and private sector have embarked on a number of open source initiatives. The NSW Government is particularly active in this regard, having recently announced its intention to spend a minimum of A$40 million on deploying open source systems throughout NSW Government departments. In addition, the Western Australian Government has also indicated its intention to develop a purchasing plan for open source software and to establish a facility to showcase non-proprietary options to the public and public sector.[22]

On a Federal level, the National Office for the Information Economy released an e-government strategy which promotes trials of open source software through "encouraging departments and agencies to consider and assess the merits of open source software as part of their strategic approach to sourcing their information and communication technology expertise".[23] In addition, on 1 April 2004 the Joint Committee of Public Accounts and Audit published a report entitled "Enquiry into the Management and Integrity of Electronic Information in the Commonwealth". Although the Committee did not make specific recommendations in relation to open source software, it observed that "agencies should consider the benefits or otherwise of using open or closed source software, as a normal part of their IT risk management processes and their cost/benefit analysis of new resources".

[20] Financial Management and Accountability Amendment Regs 2003 (Cth.), Schedule 1(1) amendment to Financial Management and Accountability Act 1997 (Cth.) s 44 A (2) (a).

[21] *Id.* at Schedule 1 amendment to Financial Management and Accountability Act 1997 (Cth.) s 43 A.

[22] Simon Hayes, *Open Source Wins Political Space*, THE AUSTRALIAN, Sep. 28, 2004 at 36.

[23] National Office for the Information Economy *Better Services, Better Government*, Nov. 2002 *at* <http://www.agimo.gov.au/admin/search/publications>.

VII OPEN SOURCE SOFTWARE IN GOVERNMENT - THE EXPERIENCE ABROAD

Open source software is favoured by many large organisations, including government departments throughout the world.[24] In as early as 2001 the European Commission announced their "eEurope - An Information Society for all" initiative which addressed the topic of open source software and set a target that "the European Commission and Member States will promote the use of open source software in the public sector and e-government best practice through exchange of experiences across the Union"[25]. In pursuit of this target, the Member States have pioneered the rise of open source software in Europe today.

Today, many Member States require that open source software be considered as part of the procurement process at the Federal, State and local levels of government. In Germany alone, over 500 Government agencies are reportedly using open source and, most remarkably, the local Government in Munich has planned to migrate its systems to open source exclusively. Further, the municipality of Vienna has announced that it will offer half of its government agencies the choice of migrating to open source in 2005 in an attempt to alleviate the current reliance on proprietary systems.[26] In addition, the UK Government is a major user of open source software. In July 2002 the Office of the e-Envoy in the United Kingdom published an open source software policy which encouraged, among other things, the exploration of using open source software as the default exploitation route for government funded R&D Software.

[24] The European Commission Directorate-General Information Society has published a list of cases of OSS recognition, explicitly policy statements or procurement decisions from public bodies worldwide which is available *at* <http://europa.eu.int/information_society/activities/opensource/cases/print_en.htm>. In addition, the Centre for Strategic & International Studies has compiled a table which provides information on the number and type of open source software policies and legislation considered by national, regional or local governments around the world which is available at <http://www.csis.org>.

[25] OFFICE OF THE E-ENVOY, OPEN SOURCE SOFTWARE - USE WITHIN UK GOVERNMENT 2 (JUL. 15, 2002).

[26] Further commentary on the Munich open source initiative is available at: Peter Galli, *Why Munich Dumped Microsoft for Linux*, May. 28, 2003 (online) at <http://www.eweek.com/print_article/0,1761,a=42474,0.asp>; Stephen Shankland, *Munich Breaks with Windows for Linux,* CNET NEWS.COM, May. 28, 2003 (online) *at* <http://news.com.com/2102-1016_3-1010740.html>; Georgina Prodhan, *Munich set to Approve Linux Despite Patent Worries*, REUTERS Sep. 28, 2004 (online) *at* <http://www.reuters.co.uk>; S. Taylor, *Munich delays Linux move by a "few weeks"*, COMPUTER WEEKLY.COM, Aug. 11, 2004 (online) at <http//:www.computerweeksly.com>.

VIII THE PROCUREMENT DECISION

It is difficult to ignore the growing open source "phenomenon" sweeping through government departments and agencies on a global scale. In light of this, the challenge faced by government is to explore the debate between the proponents of closed and open source software and to consider whether open source has the potential to achieve increasing value for money and greater operational activity across the public sector.

In making this assessment, it is crucial for government to fully appreciate the risks, as well as the benefits, of open source software so that an informed decision can be made about its applicability to public sector requirements. Ultimately, government must be confident that it has made the best technology choice at every juncture. Whether the best choice is an open or closed source model is a matter for government to determine, however based on the trend on governments abroad, there is a persuasive argument for mandating open source software which suggests that open source option should not, at this stage, be discounted from the procurement decision.