Open Source vs. Mandatory Restrictions

A submission to the
Inquiry into technological protection measures (TPM) exceptions
of the
House Standing Committee on Legal and Constitutional Affairs

Jiří Baum

26th of September 2005

Executive summary

One of the foundations of Open Source is consumer empowerment; empowering consumers (in this case software users) to make any modification they wish to the software they use.

Mandatory Restrictions (such as copyright protection or regional coding) are based on the idea that a supplier can and should restrict the actions of consumers. They are "mandatory" in the sense that the consumer cannot override them; they can be opposed to Advisory Restrictions, which merely warn the consumer that the action is likely to be illegal.

These two concepts are in natural opposition. Since users are empowered to make any modification they wish to Open Source software, any restrictions it implements cannot be truly mandatory - the user is empowered to modify any aspect of the software, necessarily including any restrictions, thus rendering them merely advisory.

Therefore, Open Source developers need sufficient exceptions (or some other mechanism) to enable them to implement Advisory Restrictions in the place of the Mandatory Restrictions that are not possible in Open Source.

The Author

I am a computer scientist (M.Comp.(Monash)), Open Source developer with current and past participation in several projects, co-owner of a small family business as well as a concerned citizen.

This submission is made on my own behalf as an individual.

Terms of reference

This submission primarily concerns itself with point (e) of the Terms of Reference, with some comments on (f) (*"the activities of open source software developers"* and *"activities conducted in relation to regional coding of digital technologies"* respectively).

Introduction

There is a fundamental conflict between Open Source and Mandatory Restrictions. One seeks to empower consumers, the other to restrict them.

The remainder of this submission is structured as follows: in the next section, a brief description of Open source is presented, especially as it pertains to this conflict, together with its importance for Australia. The following section describes Mandatory Restrictions similarly. Finally, the details of the conflict are described.

Open Source and the consumer

One of the foundations of Open Source is consumer empowerment; empowering consumers (in this case software users) to make any modification they wish to the software they use. This is fundamental to both the current, live usage, and to the history of the movement.

In the Open Source Definition[1], points two and three embody this right:

> 2. **Source Code** The program must include source code, and must allow distribution in source code as well as compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost — preferably, downloading via the Internet without charge. The source code must be the preferred form in which a programmer would modify the program. Deliberately obfuscated source code is not allowed. Intermediate forms such as the output of a preprocessor or translator are not allowed.
> 3. **Derived Works** The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.

Point two enables modification in practice by ensuring all users have access to the source code (the blueprints) for the software on which to build their modifications, and that it's in a useful form. Point three then protects their legal right to make them and to distribute the modified versions to others.

Historically, the details of Open Source — even the label — have changed over the years, but this right has always been fundamental. From the beginning, it was the rights ("freedoms") of the users that were at the heart, and frustrated users the first volunteers.

Even the word 'consumer' is quite alien to Open Source, because of the passivity it implies. In the world of Open Source, it is tacitly assumed that every user is at least potentially a co-creator.

[1] Open Source Definition: `http://www.opensource.org/docs/definition.php`

This framework is also the key to the success of Open Source. It has recently been estimated[2] that the Debian Linux distribution, available for free from its website[3], would cost eight billion US dollars ($\approx 10\frac{1}{2}$ billion AUD) to develop at commercial rates if it were a proprietary product. Every contribution to this group effort is due to the empowerment of the individual and business users of Open Source to take charge of the software they use and make incremental improvements in a way that allows the improvements to build on each other.

Open Source is economically very important. Since several key parts of the Internet are Open Source, it is indirectly involved in practically all e-Commerce. Even just the direct contribution is huge; for instance, in the *LAMP* stack widely used for web-based businesses, each of the four letters stands for an open-source product.

Open Source improves our balance of payments. Currently, a sizeable fraction of ICT spending is destined for the coffers of largely US-based foreign companies. Increased use of Open Source reduces this fraction and increases the fraction spent locally.

Open Source is important for competition, since in many categories of software it is Open Source that provides the only viable competition for what would otherwise be a monopoly or near-monopoly supplier. At the same time, it is not itself prone to monopoly effects due to the way it is licensed.

Mandatory Restrictions and the consumer

Mandatory Restrictions are based on the idea that a supplier can and should restrict the actions of consumers. These restrictions may be intended to prevent copyright infringement (Technological Protection Measures proper), they may be other restrictions not contemplated by copyright law, or a combination of both.

An example of a Mandatory Restriction to prevent copyright infringement would be a restriction of copying to another machine. Examples of non-copyright Mandatory Restrictions include regional coding, which restricts the playing of content to particular geographic regions, or time-based restrictions, for instance recordings which expire on a certain date.

They are "mandatory" in the sense that the consumer cannot override them; the device simply refuses to operate in certain ways. They can be opposed to Advisory Restrictions, which merely warn the consumer that the action is likely to be illegal but permit the action to be completed.

Mandatory Restrictions apply not only or even primarily to DVDs and other entertainment goods. Similar features are present in various document formats (such as password-protected documents or non-printable PDF files), networking protocols (such as Access Control Lists) and other contexts. Their proliferation

[2] "Measuring *Libre* Software Using Debian 3.1 (Sarge) as A Case Study: Preliminary Results", Amor-Iglesias *et al*, UPGRADE, vol. VI nr. 3, June 2005; available at `http://www.upgrade-cepis.org/issues/2005/3/up6-3Amor.pdf`

[3] `http://www.debian.org`

and sophistication is likely to increase in the future. Since these features can be used to protect copyright, they qualify as Technological Protection Measures; however, in many cases, they will be deployed in ways that are quite irrelevant to copyright protection.

For instance, an early version of a server networking program which does not fully implement the details of the Access Control Lists may nevertheless be quite suitable for use on private networks where all users are trusted. However, such an early version would clearly be a circumvention device and therefore illegal under the proposed law.

As can be seen, Mandatory Restrictions will often restrict a range of legitimate uses that have nothing to do with copyright.

Due to the possible commingling of copyright and non-copyright restrictions, it is quite likely that it will be difficult or impossible to remove one without also (at least partially) circumventing the other. This will be especially so in cases where the commingling is intentional on the part of the supplier, so as to extend the protections of copyright law to other, non-copyright restrictions which the law does not contemplate, such as regional coding or preventing interoperability.

For the purposes of interoperability in particular, a developer will often have to first circumvent the original restrictions, especially where they are commingled with other aspects of the format or protocol, then implement a replacement for them. Even where the final product implements the restrictions faithfully, prototypes and development versions may not.

Conflict over the Consumer

Open Source and Mandatory Restrictions are in natural opposition. One seeks to empower consumers, the other to restrict them.

Since users are empowered to make any modification they wish to Open Source software, any restrictions it implements cannot be truly mandatory. The user is empowered to modify any aspect of the software, necessarily including the parts enforcing the restrictions; this renders any such restrictions effectively advisory.

Open Source will therefore effectively be banned from growing areas of software unless there is some mechanism to allow such effectively advisory restrictions in the place of Mandatory Restrictions. Such an effective ban would cripple Open Source (at least in Australia) and relegate it to less important areas of computing.

Another difficulty arises with the bazaar model of development common in Open Source software. An integral part of this model is releasing very early versions to the public, to attract additional developers and early adopter users to the project. However, such very early versions will often not yet implement all of the details of the restrictions (if they implement any at all), or they may implement them incorrectly.

In any case, attempting to implement Mandatory Restrictions in Open Source software would probably be counterproductive as far as protecting copy-

righted content is concerned; if the developer attempts to do so, some of the users will likely modify the software to override them; and such modifications may well remove the restrictions altogether rather than converting them to Advisory Restrictions, or remove all commingled restrictions rather than just the non-copyright ones. Where such modified software is more functional than the original, users will prefer it.

On the other hand, if the restrictions are advisory to begin with, many users will abide by them as they have no wish to break the law.

Conclusion

For these reasons, Open Source developers need sufficient exceptions (or some other mechanism) to enable them to implement Advisory Restrictions in the place of the Mandatory Restrictions that are not possible in Open Source.

Contact

I would be happy to elaborate on these points, whether via e-mail, by way of a Supplementary Submission or in person.

Jiří Baum
P.O. Box 2364
Rowville Vic 3178

Email: jiri@baum.com.au
Phone/fax: (03) 9764 3342