OPEN SOURCE LICENSING IN THE NETWORKED ERA

by

MICHAELA MACDONALD*

The underlying principles of open source software are 'openness' and interoperability. The licensing infrastructure ensures that these values will be preserved. However, the ideology may be under threat in the networked era. Cloud computing is a maturing technology practice whose success is contingent on utilizing open source software. Open source licences apply when software is distributed – mere running or making available does not trigger the protection. The new licence, Affero GPL, aims to prevent a potential proprietary lock-in in the cloud. This article argues that the attempt is futile. The freedom to modify software may be relevant for hobbyists, not for the majority of commercial enterprises. In the current technology set-up the values of open source worth preserving are accessibility, interoperability, and re-usability. And these will not be saved thanks to a licensing scheme, but thanks to an open and flexible architecture.

KEYWORDS

Open source, Affero GPL, cloud computing, distribution, making available

1. INTRODUCTION

A definition of what truly is 'the cloud' or 'open source' will probably never be agreed. However, from an enterprise computing perspective, both terms refer to specific practices. The cloud represents an opportunity for organisations to outsource the computing capability to a third-party provider of networks, servers, storage, applications or services, frequently located in multiple locations. The structure of cloud computing combines both the applications delivered as services and the architecture that provides those services

* Queen Mary University of London; misa.macdonald@gmail.com

(hardware, software, data centres).[1] Open source is the source code of software that can be used, copied and distributed for any purpose whatsoever on a royalty-free basis, provided that the terms of the relevant open source licence are complied with.[2] The idea behind the open source movement is to provide alternative licensing terms to the terms of traditional copyright protection. The restrictive rights of use in traditional contracts and associated licence fees are often perceived, from an enterprise computing perspective, as major obstacles to exploitation, innovation and growth.

Cloud computing is a maturing technology practice whose success is contingent on minimal, if not zero, licensing fees. Open source software meets these criteria. Thus cloud computing utilizes the ideology of 'free' software, but derives its value from the delivery, scalability, integration, configuration, usability and administration of software. It is apparent that open source is a vital factor in the cloud's development and both together have significantly contributed to the innovation and growth in the area of enterprise computing. However, the relationship is a complicated and a complex one. Most importantly, the open source community has voiced concerns that cloud computing threatens the core principles of open source by abusing the benefits of 'free' software without releasing the modified source code back to the community. Stallman believes that the cardinal virtue of open source software is the freedom to access and modify the source code. Reciprocal licences, such as the GNU General Public License (GPL), ensure that the source code remains open through a strictly controlled distribution of derivative works.

Open source commentators emphasize the role open source software played in the growth and development of the Internet as a non-proprietary platform.[3] They fear the values and benefits of the movement will be lost in a new proprietary lock-in in the cloud, because in this context software is no

---

[1] For example, the 'Cloud Legal Research Project' is working on a definition which more closely models the legal relationships which are created within cloud computing. The aim of the project is to reduce the considerable uncertainty as to the legal and regulatory status of several essential aspects of cloud computing. The project started in October 2009 under the leadership of the Centre for Commercial Law Studies (CCLS) at Queen Mary, University of London, more information can be found here at http://www.cloudlegal.ccls.qmul.ac.uk/.

[2] Graham, R. 2011 'The Cloud & Open Source: The New Darlings of Enterprise Computing', The Society for Computers and Law, available at http://www.scl.org/site.aspx?i=ca0&r=2&x=v6nk93wf

[3] O'Reilly, T. 2008 'Open Source and Cloud Computing', available at http://radar.oreilly.com/2008/07/open-source-and-cloud-computing.html

longer distributed,[4] but merely run on a global network. Open source licences apply when the act of distribution takes place – using, running, or making the code available does not trigger protection granted by the licences. As a response, the Free Software Foundation (FSF) introduced a new licence – GNU Affero GPL (AGPL). The AGPL contains an additional requirement regarding network deployments of software. Modified versions of AGPL-licensed software must be released to users who interact with the software over a network. The central argument of this paper is that the industry has moved on from the individual hobbyist modifying software, to a more formal engineering practice, where commercial companies invest in open source software.

It is worth noting, however, that source code is not the only element determining the 'openness' and interoperability of software. The proprietary nature of an application is often determined by the data rather than source code. While open data and open services are highly relevant in the battle for more 'open' cloud, it is important to realize just how much of what is possible is dictated by the architecture of the systems used. As O'Reilly says, "in the era of the cloud, open source without open data is only half the application."[5] The principles of open source software can still thrive in the cloud provided that it is built on federated rather than centralized services. Publishing APIs and language standards, allowing competing implementations of open software, has more relevance in driving technological innovation in today's industry. The different approach can be demonstrated on Twitter and Facebook, two social networking platforms boasting millions of users. Facebook provides application-programming interfaces (APIs) that can be run only on Facebook, while Twitter has built a database of APIs that support third-party applications. Twitter API has enabled people to write both web and desktop applications, mashups and otherwise, using Twitter's services and data. What the cloud has change is a shift from 'opening' the source code to 'opening' the architecture of the system. In other words, "architecture trumps licensing any time."[6]

---

[4] Rosen defines distribution as a "selling or giving copies of software away to others. It also may include such arrangements as incorporating software into consumer or industrial products and selling those products to others. For some software, it may also include making the software available across a network for execution by others." Rosen, L. 2004 'Open Source Licensing', Chapter 3

[5] O'Reilly (2008)

[6] O'Reilly (2008)

Open source movement has to react to these new circumstances and rediscover the values of open source in the networked era. Some values have become more relevant than others. The freedom to modify software may be relevant for hobbyists, not for the majority of commercial enterprises. The (naïve) assumption is that the freedom to modify necessarily implies the improvement of the software, which is often not the case. Modifications only contribute to the quality of the project if they are controlled, assessed, tested and approved. In a business environment, commodity open source software is particularly relevant as it passed quality assurance, such as in the case of Apache or Linux.[7] The article argues that in the current technology set-up the values of open source worth preserving are accessibility, interoperability, and re-usability. And these will not be saved thanks to a licensing scheme, but thanks to an open and flexible architecture. At least for now, that is.

2. OPEN SOURCE MOVEMENT AND THE CLASH OF VISIONS

Software is a type of creative work protected by copyright. In addition, it can attract protection of other intellectual property rights. Proprietary software is released under an exclusive licence protecting legal rights of the copyright holder. The licensee is given the right to use the software under certain conditions, while restricted from other uses, such as modification, further distribution, or reverse engineering.[8] The Open source movement offers alternative to this traditional copyright approach by enabling anyone to use or modify open source software under varying conditions (specified in the licence). The Open source movement uses standardised licensing infrastructure, which is based on two essential ideas: users are allowed to use the software for any purpose and they can modify and redistribute it without prior authorisation from the initial developer.[9]

---

[7] The overall project leader and coordinator of Linux is Linus Torvalds. "The Linux team and Torvalds evaluate the quality of contributions they receive from around the world, and they decide whether to include those contributions as a part of Linux. The Linux project has formal mechanisms for evaluating and testing contributions, and there is a collective rather than dictatorial decision process. In contrast, a board of directors coordinates the development activities of the Apache Software Foundation. Many of the leaders of the Apache project work for software companies that donate their employees' time and software to the Apache Foundation. Important decisions relating to Apache are decided by open vote and consensus." Rosen (2004)

[8] Fontana, R. et al. 2008 'A Legal Issues Primer for Open Source and Free Software Projects', Software Freedom Law Center

[9] Guibalt, L., van Daleen, O. 2006 'Unravelling the Myth around Open Source Licenses: An Analysis from a Dutch and European Law Perspective', Asser Press

The initiative can be traced to Richard Stallman, the founder of the Free Software Foundation (FSF) in 1985, which was to serve as an institutional framework for the development of free software, and in particular of the GNU project.[10] The fundamental freedoms underlying the movement are the freedom to run the program, to study the program, to re-distribute copies of the program and distribute copies of derivate work of the program. Free software and documentation thus allows users to redistribute and change the software under the condition that modifications would have to be published and brought out under the same conditions.[11] These political and ethical principles are encapsulated in the ‚share and share alike' or ‚copyleft' clauses, which are inherent elements of all GNU licences. The following licences have been published by the FSF: GNU General Public License (GPL, the latest versions are the GPL v3 and the GPL Affero); GNU Lesser General Public License (LGPL); and GNU Free Documentation License (FDL). According to Google Code Search, in 2010 GNU GPL and LGPL represented over 70% of the market share of open source products, followed by BSD, Apache and MIT licences ranging between 8-4%.[12]

These political and ethical convictions, however, do not represent the open source movement in its entirety. In 1991 Bruce Perens and Eric S. Raymond established the Open Source Initiative (OSI). their strategy is more pragmatic and market-oriented. They have successfully challenged the term 'free software' used by the FSF and replaced it by 'open source software', which is more appealing to the commercial sector. Some of the leading open source products are Linux with its many distributions, which are released mainly under GPL.[13] Ubuntu is an example of a successful and popular Linux distribution that has gained recognition and acceptance. Apache is web server software, developed and maintained by the Apache Software Foundation and licensed under the Apache License. Google Android, a Linux-based mobile phone operating system, is one of the fastest growing open source products. Although the underlying Linux kernel is licensed under GPL, Google decided to distribute the software infrastructure under the

[10] An operating system created by Stallman as response to commercial software. It means 'GNU is not Unix' and stands for a complete Unix-compatible operating system licensed as free software.

[11] The Free Software Foundation, http://www.fsf.org/about/

[12] Google Code Search, http://license-info.appspot.com/#; Accessed 29 October 2010 and no longer available

[13] This report refers to GNU GPL v2, available at http://www.gnu.org/licenses/gpl-2.0.html

Apache License.[14] In 2009, Microsoft has listed in its annual report for the first time Canonical (the company behind Ubuntu) and Red Hat (distributor of Linux) as serious competitors and threat to Windows. Google, Apple, Opera, and Android were also identified to be competition to the Internet Explorer Web browsing capabilities of Windows products.[15] The most recent annual report identifies an open source business model as a serious threat to Windows products and services. The report describes this business model as based on "modifying and then distributing open source software at nominal cost to end users and earning revenue on advertising or complementary services and products. These firms do not bear the full costs of research and development for the software. Some open source software vendors develop software that mimics the features and functionality of our products."[16]

While the FSF recognises the value of software in the freedom to modify it, the OSI promotes the developer's freedom to utilise the software as he sees fit, including commercial purposes. The clash of values in the centre of the open source movement rises out of different perspectives. The Stallman's of this world want to break free from the technology overlords, such as Microsoft and Apple and end the dark years of legal feudalism represented by copyright and patents. However, not every user, developer and businessman is so deeply concerned with the ever-growing dependency on technology licensed on a proprietary basis.

Open source is now a mature concept and little attention is given to these old battles. The economic potential of open source software has been long recognised by companies including vendors of proprietary software and have been reflected in significant open source project investments. The majority of software companies make profit from the fact that commodity open source software works having passed rigorous quality assurance, can be difficult to use, and does not have quality documentation. In other words, commercialization of open source typically involves documenting

---

[14] Unlike GPL, permissive licenses such as BSD or Apache License are preferred by many companies because they allow using open-source software code without having to turn proprietary enhancements back over to the open source software community. Paul, R. 2007 'Why Google chose the Apache Software License over GPLv2 for Android', Ars Technica, http://arstechnica.com/old/content/2007/11/why-google-chose-the-apache-software-license-over-gplv2.ars [Accessed 3 September 2010]

[15] Microsoft's Annual Report, 2009, p. 4, available from http://www.microsoft.com/investor/SEC/default.aspx?year=2009

[16] Microsoft's Annual Report, 2012, p. 15 , available from http://www.microsoft.com/investor/default.aspx

and distributing modules, but not modifying them. The recent trend is that companies are sponsoring open source gurus to achieve the functionality they require for commercial purposes.

3. CLOUD COMPUTING: A NEW CHAPTER FOR OPEN SOURCE

Cloud computing is a marketing term for technologies that deliver location-independent computing, where servers provide software, data and computational resources to other networked computers as a service on demand rather than a product. The benefits of cloud computing include, for example, maximising capacity utilization, outsourcing the maintenance burden of servers and applications, dynamically scalable and virtualized resources, data accessible from anywhere in the world over the Internet and regular and predictable operational expenditure.

The underlying principle, that treats computational power as a generic utility to be used, is called ‚utility computing'. A parallel to cloud computing can be drawn with commonplace utilities like electricity or water, where end-users consume power on demand without needing to understand the component devices or infrastructure required to provide the service. The idea behind utility computing is not new – it has been around since 1966, but only recently has it become a mainstream phenomenon. It can be said that repackaging of computing services to provide 'on-demand computing' and cloud computing models are by-products and direct consequences of the ease-of-access to remote computing sites provided by the Internet. [17]

Cloud computing services are often characterized in terms of the level of access or type of service that is being offered. Common examples include Infrastructure as a Service (IaaS), such as Google Mail or Google Docs that deliver computing resources like processing power or storage. Platformas-aservice (PaaS) offers a set of lower-level tools such as an operating system, computer language interpreter or web server for the construction of bespoke applications. Microsoft Windows Azure and Google App Engine are examples of PaaS. Software asa Service (SaaS) provides a functionality akin to an end-user application, such as Amazon's Elastic Compute Cloud (EC2). This range of services can be viewed as a spectrum of provision from low-

[17] Judd, W. 'SaaS Threatens Open Source' 2011, available at http://williamjudd.com/2011/01/28/the-threat-of-saas-to-open-source/

level functionality (IaaS) to high-level functionality (SaaS), with PaaS in the middle.[18]

In principle, cloud providers deliver applications via the Internet, which are accessed from web browsers, desktop and mobile apps, while the business software and data are stored on servers at a remote location. For security reasons, especially larger organisations want the benefits of cloud computing but without the risks inherent in trusting their data to a third party. They can achieve this by creating a cloud-like infrastructure in their own data centre. This is called a private cloud. The public cloud refers to providers such as Amazon, Google and salesforce.com, whose shared services are available to all. A hybrid approach uses both public and private services.[19]

At the moment, both the cloud and open source are 'the darlings of enterprise computing'.[20] In private though, their relationship is far from amicable. The open source movement helped to build non-proprietary Internet and thus laid the foundations for a technology like the cloud to succeed. Now they watch, "as the world they've changed picks up on their ideas, runs with them, and takes them in unexpected directions."[21] Cloud providers benefit from the use of open source on the hardware infrastructure, because the licence allows them to build, increase and maintain their hardware infrastructure without repeatedly paying proprietary licence fees.

Open source software is also used as the underlying software for virtualisation technology. Virtualisation enables a physical machine to be subdivided into several different virtual servers or, alternatively, different physical machines can act as one discrete virtual server. Permissive open source licences allow for the software to be used on any machine, in multiple locations and for any purpose.

The architecture of virtual machines providing cloud services rests on open source foundations. Servers frequently run on the LAMP stack, which refers to Linux, Apache, MySQL and PHP, all open source software. This al-

---

[18] Bradshaw, S., Millard, C. and Walden, I., 2010 'Contracts for Clouds: Comparison and Analysis of the Terms and Conditions of Cloud Computing Services', Queen Mary School of Law Legal Studies Research Paper No. 63/2010, available at SSRN: http://ssrn.com/abstract=1662374

[19] Anderson (2010)

[20] Graham, R. 2011 'The Cloud & Open Source: The New Darlings of Enterprise Computing', The Society for Computers and Law, available at http://www.scl.org/site.aspx?i=ca0&r=2&x=v6nk93wf

[21] O'Reilly (2008)

lows the providers to use proprietary application programming interfaces (APIs) or modified open source APIs to build the virtual machines. At this point the end-user looses the freedom to access the source code and thus is exposed to a proprietary lock-in in the cloud.

Issues can arise from the use of reciprocal open source licences, especially where derivative works are created.[22] Linking libraries to computer programs, statically or dynamically, can be under the terms of GPL interpreted as a derivative work. On the other hand, the GPL licences allow aggregation of software on the same storage medium without the requirement of reciprocity. However, the interaction of applications in the cloud brings a new perspective.[23] These and other interactions between the open source and emerging technologies utilizing the open source software for commercialization have caused fears and apprehension in the open source camp.

4. AFFERO GPL ATTACKS OR RATHER TILTS AT WINDMILLS

Due to a legal loophole software run on the cloud was not pursuant to the GPL. The licence conditions protecting open source software are triggered when the underlying source code is distributed. "Where software is provided as a service over the cloud, an arguable interpretation under GPLv2 is that a distribution does not take place – this is manifested in GPL Version 3.0 (GPLv3) which expressly states that mere interaction with a user through a computer network, with no transfer of a copy, is not a distribution … The consequence of this is that developers can use open source and provide access to the functionality of that open source through the cloud for a fee without being required to disclose the source code of the open source. This means that a developer can keep modifications to the open source private, thereby potentially creating a competitive advantage over its competitors."[24]

The open source movement has reacted to combat this 'loophole' and introduced a new version of the GPL – GNU Affero GPL (AGPL). AGPL redefines what is a distribution by expressly stating that interaction over a computer network amounts to a distribution. It is almost identical to GPL v3, except for one provision, section 13:

---

[22] Any derivative work created from a GPL source code must be distributed on the terms of GPL and cannot be licensed under a proprietary licensing model.

[23] Graham asks whether the virtual image or the interaction between applications on a virtual server can be interpreted as a derivative work.

[24] Graham (2011)

"Notwithstanding any other provision of this License, if you modify the Program, your modified version must prominently offer all users interacting with it remotely through a computer network (if your version supports such interaction) an opportunity to receive the Corresponding Source of your version by providing access to the Corresponding Source from a network server at no charge, through some standard or customary means of facilitating copying of software."[25]

It is interesting to note that although cloud computing is perceived to be more evil than Microsoft AGPL addresses the issue in one brief paragraph at the end of the document. Even more surprising is the admission of the author of the licence. The FSF commented the release of AGPL that it is not possible to fend off the potential threat of cloud computing with a software licence. Instead, they foresee a technical solution such as using peer-to-peer (P2P) networks. The private collection of 'modifications' in proprietary 'source safes' for commercial advantage is outmoded. The distributed publication of open source software through P2P networks will provide for easy access to 'open source safes'. In fact, this is to a certain extent how the big Linux distributions work already.

So while cloud computing has become the new paradigm of choice for many software organisations, with Amazon and Google leading the way and Microsoft, Oracle, HP and many others introducing cloud computing branded programs, the uptake of the AGPL has been slow. According to Black Duck statistics, mere 0.16% of open source projects adopt AGPL licence compared to relatively popular GPL v3 (11%). [26] It remains to be seen whether the AGPL licence will gather popular momentum, thereby preventing open source developers developing proprietary-type business models over the cloud. For now, GPLv2 can be used to permit service providers to use the cloud as a platform for commercialising open source software.[27] However, wider adoption of the licence would not stop the development of business models, which include open source software. Google have a perfectly good proprietary business model, using lots of open source components, and actually pay for the open source development. What AGPL might achieve is to stop the commercialization of open source modifications and kill potentially successful open source projects before they even start. There are numerous ways of making money from software, and modifying it is

[25] AGPL, Article 13, full text available at http://www.gnu.org/licenses/agpl.html

[26] Black Duck Licence Data available at http://osrc.blackducksoftware.com/data/licenses/

[27] Guibalt (2006)

only one such way. Ultimately, real-world open source projects need sponsors, usually from the commercial sector, in order to be maintained and improved.

"The Affero GPL is not right for everybody. Some communities do not want the added burden of packaging their code for release to network clients. The Affero GPL creates code distribution requirements for a class of people the FOSS world has traditionally treated more as end-users than developers. Even though some end-users are now sophisticated enough to customize their installations of open source server software, your project might not want to require them to publish those custom changes. Moreover, there are many individual developers as well as corporate commercial users who believe that AGPL takes the idea of copyleft too far."[28]

5. CONCLUSION

Modern software development and business models are predicated on a wide web of ecosystems, where users, developers, testers, customers, businesses and so on interact in a complex web of commercial transactions. This paper argues that the open source movement needs to adapt to the new network era. However, this will not be achieved through licensing. AGPL is a futile attempt to address these emerging technologies. It is futile for couple of reasons. The FSF believes the crucial virtue of open source software remains the freedom to modify the software. Examining current practices and successful business models of enterprise computing it is clear, that the elements of these ecosystems, which depend on the commercialization of modifications of open source products, are minimal and shrinking. The message of the cloud is clear and simple: modification and distribution are last century. What matters this century is ecosystem and interconnectivity. Innovation is proceeding by virtue of interoperable standards, which are published and extensible. In turn, they allow anyone to make a contribution to the architecture of the system, while not necessarily undermining its foundations. It is not the open source licence to modify, but the publication of open standards, which empower collaboration, that are determining the future of software.

[28] Guibalt (2006) A Legal Issues Primer for Open Source and Free Software Projects, op cit note 9 above