Embracing Open Source Software to Empower Potentials of Community

Husni Thamrin

Abstract--**The writing is based on literature study obtained from books and white papers and personal interview with several practitioners, governmental policy makers and students. Past experiences tells that open development or similar concept of activity helped improve a product and gave maximum benefit to the community. Open development of computer software leads to production of open source software (OSS). Embracing OSS helps promote the use of legal software, encourage the production and improvement of available OSS, educate students to good programming techniques, raise the knowledge of students on computer and operating systems, and promote the construction of community that shares and cares. The study concludes that embracing open source software is the best way to empower public potentials.**

Index Terms--**community, culture, development, open source software**

I. NOMENCLATURE

Open source software (OSS) are computer programs that are developed openly by a group of programmers or by the community. Openness in this context can mean that everybody in the world can look into the program code, may contribute to the development, may modify, distribute and sell the software. There are a bunch of licensing models for open source software including the popular GPL (General Public Licence). These licences are applied to ensure that OSS and their derivatives will always be open.

II. OPEN SOURCE AND OPEN DEVELOPMENT

The concept of open source and open development deals with practices in production and development that promote access to the end product's source materials by interested parties. Within the same concept, open source software can be regarded as software that allow people to look into the source code and give contribution to the development and implementation of the software.

Opening the source code of a product enabled a self-enhancing diversity of

production models and interactive communities [1]. Members of the community may study the source code and repeat the experience, modify the procedure and do their own experiments and share the results. Those who is new to the field may learn quickly and may later contribute.

History tells how open development or similar concept of activity helped improve the quality of products and gave maximum benefit to the community. In paper [2], Robert C. Allen presented a detailed case study of technical improvement in the iron industry of Cleveland, United Kingdom during the period of 1850 to 1870. It should be noted that in the period, patenting policy had already been adopted by many countries but the community of Cleveland iron industry developed a model of innovation, which is labelled by Allen as collective invention. Cleveland iron producers freely disclosed to their competitors technical information concerning the construction details and performance of the blast furnaces they had installed. Information was normally shared both through formal and informal channels, such as scientific seminars, talks, site visit and personal communication. New technical knowledge was not protected using patents so that competing firms could freely make use of the released information when they had to construct a new blast furnace. The consequence of this process of information sharing was that the blast furnaces of the district increased their performance very rapidly. Allen noted three essential conditions as the basis of the emergence of the collective-invention model. The first condition refers to the nature of the technology. In the period considered, there was no consolidated understanding of the working of a blast furnace. What the best engineers could do when designing a new blast furnace was to come up with some design guidelines on the basis of previous experiences. Obviously, the sharing of information related to the performance of a large number of furnaces allowed engineers to rely on a wider capita selecta of information in their extrapolations, leading to a more rapid rate of technological progress. Second, blast furnaces were designed by independent consulting engineers who were normally employed on a one-time basis. In this context, the most talented engineers had a strong incentive to disseminate the successful design novelties they had introduced in order to enhance their professional reputation and improve their career prospects. Third, iron producers were often also owners of iron mines. As a consequence, improvements in the efficiency of blast furnaces would have led to an enhancement in the value of the iron deposits of the region. Thus, there was an interest in the improvement of the average performance of blast furnaces, because only improvements in the average performance would made influence to increasing the value of iron deposits.

More recent industry has seen similar activity of open development. In the early of 20th century, Motor Vehicle Manufacturers Association instituted a cross-licensing agreement among US automotive manufacturers, whereby all manufacturers shared between them all of their respective patents openly and without exchange of money. By the time US entered World War 2, for example, Ford had 92 patents that were used freely by other manufacturers and had in turn used 515 patents from other companies without lawsuit. In this way, automotive industry improved to a larger extent with the benefit to more companies and people than it had been previously when the patents were in the hand of a group of capital monopolists [3].

Telecommunication network protocols were also developed in the spirit of open development. Around 1970s, researchers at ARPANET (Advanced Research Projects Agency Network) used collaborative process called Request for Comments to demand feedback from other parties against the first party work. Improvements came by and the work led the birth of the internet. Had the researchers decided to close their work, the improvement would have been slow, the birth of internet would have been delayed, the spread of internet usage would not have been as it is to day and internet would not be as cheap either.

Early software developers around 1960-1970s enjoyed the share and open culture. Those days, computing companies such as IBM often release operating system code and other programs to let people and other industry develop systems based on their machines. Those who learned computers in the late 1970s may well remember that personal computer technical guide included a list of assembly language that constructed the PC Disk Operating System. This openness allowed talented students and developers to create programs for personal computers which had helped spread the usage of computers in public.

III. PROPRIETARY CODE AND SOFTWARE PATENTS

In oppose to the concept of open source and open development there is closed source and secret development of science and technology, which, in software engineering context, leads to the production of what is called proprietary software.

Proprietary code is developed largely in private, albeit its requirements are usually developed with its prospective users [4]. Its source code is generally not disclosed and is typically distributed under the shield of binary executables. Its use is controlled by proprietary software licensing restrictions. The right to copy the program executables is restricted and the user is generally forbidden from attempting to modify and surely from redistributing the code or possible improvements or

attempting to learn how it works by reverse engineering.

In most respects, open source and proprietary application development are polar opposites, though this is not to say there are not many areas where the commercial and open communities have cooperated.

Proprietary software vendors use all of the traditional means of protecting intellectual property — patents, copyright, and trade secrets — in one manner or another to protect software from being used not as intended [5]. In the US, the Congress has even invented a new type for cases in which these may be insufficient, via the Digital Millennium Copyright Act.

Indeed, Intellectual Property Rights is a good tool to reward the creativity and hard work of the patent holder. It grants the holder the right to exclude other people from manufacturing the patented object or using the patented process for a temporary period after the patent has been applied for. However, Intellectual Property Rights has also been used to maximise profit and when it happens, it may create drawback across the whole of our social lives, transforming our ordinary way to interact and to take part in culture and creativity.

Some experts disagree on granting patent to computer software. Klemens, for example, argued that software is not just like any other machine and it is not just Hamlet with numbers [5]. Software is a functional hybrid that can be duplicated at no cost, is legible by computers in some forms and by humans in others, and has a unique mathematical structure. All of these facts have to be taken into consideration in designing any type of protection for software.

There are hundreds of patents that make a competent programmer groan as how it has been granted in the first place. For example in the US, patent 6,389,458, granted May 14, 2002 (filed October 30, 1998, by Brian Shuster), is for pop-up browser windows, which are typically used by advertisers to put ads on top of the content that people actually want to see, and to make it difficult for users to leave a web site. In JavaScript, implementing the patent only needs three lines of computer code, yet the language has been included in web browsers since 1995. This simple implementation had been the idea of JavaScript developers years before Shuster filed for the patent. The U.S. Patent and Trademark Office (USPTO) saw that this combination of one line of code to open a computer window and a subsequent line to focus on the window is a new, non obvious invention, and that no persons may put these three lines of code in sequence in their own work unless they pay Shuster's company a royalty for the privilege of doing so. Only in 2018 will this combination of three lines enter the public domain.

In contrast to the above simple patented work is patent 4,314,081 (granted February 2, 1982, to Bryan Molloy and Klaus Schmiegel): "3-Aryloxy-3-phenylpropyl-amines and acid additions salts thereof, useful as psychotropic agents, particularly as anti-depressants." This patent covers the active ingredient in the formula for the antidepressant Prozac. It is unlikely that even the best chemists could look at the chemical formula and infer that it could alleviate symptoms of depression in certain people. If they could, it would be because they had studied the work of Molloy and Schmiegel. Nor could we ask the best chemists to quickly jot down a chemical compound to alleviate depression and expect them to produce anything like this formula, the product of years of research by Molloy and Schmiegel costing untold amount of money. Conversely, Shuster's invention would make a good quiz question for an undergraduate computer science class. Nonetheless, the patent for Prozac and the patent for pop-ups are entirely equal under the law.

Poorly considered patent granting also occurs in the field outside computing. In 1995, two scientists from the University of Mississippi were granted a patent on a method of increasing the effectiveness of treatments of wounds and cuts by the use of turmeric in a special preparation. Turmeric as a treatment for minor skin cuts and wounds has been used in India as a traditional remedy for hundreds of years. However, intellectual property law does not see anything that constitutes originality or inventiveness in traditional remedies and so is unwilling to grant any protection to traditional knowledges [6].

It needs to be emphasized that Intellectual Property Rights is a tool to reward creativity and hard work but should not be used to maximize profit without considering social needs, as what happened to a nursery in 2002. The story began when the nursery innocently painted pictures of Disney characters on the walls for the amusement of the children, aged between one and five years old, who played there. Somehow, the Disney corporation found out and their lawyers sent a cease and desist letter to the nursery explaining that this represented an infringement of copyright. They warned that the nursery should remove the offending paintings and images from their walls. Failure to comply would mean an expensive and drawn-out court action that would most likely bankrupt the nursery. Even though the children from the nursery went on national television to plead for their beloved nursery walls, the Disney representative claimed that they viewed the nursery as a for-profit organisation and didn't feel a need to distinguish between it and other organisations. They stated that any infringement by anybody else would be dealt with in the same harsh way [7].

It has been the practice of proprietary software industry to claim patent to any "new" idea that come up with them. Similar idea may come to the mind of another programmer without having to look at the work of the first programmer. The two may implement the idea using different programming language, which practically become different work, but the patent law states that the latter must cease his/her work. Apparently, intellectual property rights should be reconfigured so as to grant protection to really original work and not to novice ones. This is in order that creativity flourishes.

IV. OPEN SOURCE SOFTWARE AVAILABLE

In the decade of 1980s began a strong trend from software developers to close their work and get profit from it. Software developers get high apraisal in the job market. Several companies started to develop their own version of Unix system. Despite the trend, the spirit of open development remained on a group of developers who founded Free Software Movement. The movement gained momentum when Linux was created in 1993. Open source development survives and to date, there are a lot of open source software available ranging from operating system for server, desktop and distributed system to office application, to system tools, to internet application and games. This section describes several popular open source software in use and their features.

Office productivity tools probably have the largest number of users with which people are highly aware of the functionalities and usage. Of course an operating system must exists in every computing machine but the existence and functions may be unnoticed or ignored. There are a lot of open source office productivity tools available and the most popular one at the moment is OpenOffice (which is being evolved to LibreOffice). OpenOffice has almost all the facilities to produce office documents including text documents, spreadsheet, presentation and database. Editing mathematical symbols cannot be easier in OpenOffice as oppose to, say, Microsoft Office. Though it is not as interactive as the current version of Microsoft Office, the capability to get job done is not less easy. In addition, OpenOffice is able to access open standard documents and Microsoft style documents. The way OpenOffice handle menu and editing facilities is not the same with other office suite, but the functionalities are mostly available in all suites.

In the desktop operating system world, Linux is probably the youngest guy that was born by the end of the last century. At that time, Microsoft had already dominated the OS market with its graphical MS Windows while Unix kept losing

market and popularity due to its high price. Linux evolved in a completely different way from proprietary software or large freeware. From nearly the beginning, Linux was hacked by huge numbers of volunteers coordinating only through the internet. Quality was maintained not by rigid standards or autocracy but by the simple strategy of releasing every week and getting feedback from hundreds of users within days, creating a sort of rapid Darwinian selection on the mutations introduced by developers [8]. This sort of bazaar development model has produced a stable and working operating system. Together with other toolkits and applications such as the X Window system, Desktop managers and Office suite, Linux are bundled into a software system distribution (or also known as Distro). A lot of distros are available today including the popular Ubuntu, RedHat, and SuSe. Indonesian hackers have also enjoyed the open and share environment and produced their own distros, some of which are IGOS, BlankOn and Tajdid Linux. However, Linux has a major disadvantage. At the moment, only a limited number of hardware companies develop drivers for Linux or release the hardware programming guide to open source developers. For systems with this group of hardware, Linux cannot run well and the open source community has to develop their own drivers without sufficient knowledge from the hardware manufacturers. Developing new drivers in this situation always takes time and sometimes developers have to reverse engineer the drivers for other operating system. Consequently, support for new hardware becomes lagged.

In contrast to the desktop operating system market which sees the domination of a product of proprietary software, hand-held device market does not have a particular software that dominates constituency. Mobile phones, personal devices or other gadgets may come with both proprietary and open source operating systems. Proprietary OS includes Blackberry RIM and Symbian while open source OS includes Java OS and Android. In particular, Android has gained more popularity lately with more and more smart phones go to market installed with the OS. Reportedly, Android market share has jumped to 26% eclipsing iPhone with 25% which is behind Blackberry RIM with 37% [9].

A lot of open source internet based applications have been developed. Internet browser Mozilla Firefox gains 25% market share and becomes a tough competitor to Microsoft Internet Explorer which is bundled in Windows OS. Many developers have enriched the features of the browser by developing add-ons so that everybody may customize their browser with a variety of functionalities to meet his/her needs. Mozilla Firefox is known to its secure and safe browsing feature, which other

browsers suffer more often.

Empathy internet messenger client facilitates chatting and text-based communication with almost all types of account including Yahoo Messenger, Windows Live and Google Chat. Similarly, Gwibber social client may be used to login to popular social websites such as Facebook or Twitter from within a single application.

On the creation of websites, open source community has provided web templates and web authoring software. Millions of web sites have been created using a variety of content management systems. It is even possible to create a social media website with standard facilities including chat, note and image sharing and video posting.

Open source multimedia applications are available to playback music and videos, manage multimedia files and purchase multimedia content from the internet. Multimedia editing application includes bitmap and vector image editing, video editing, animation design, 3D image editing and game development application.

V. OPEN SOURCE SOFTWARE EMBRACE

The relatively cheap deployment of open source software attracts governments and institutions to thinking of using OSS in the offices and sites. Singapore, for instance, is offering tax breaks to companies that use the open source Linux operating system instead of commercial alternatives like Windows [10].

The federal government of Germany has adopted legislation that mandate a new IT environment: Linux on servers, Windows on desktops. Later on, the government has made an agreement with IBM that offers German government offices discounts on IBM machines with pre-installed Linux provided by German Linux distributor SuSE.

Brazil is well-known to its brave step in OSS adoption. Legislation has enforced requirements that open source software be given preference in municipal governments of Recife, Campinas, Solonopole, Amparo, Sao Carlos, and Porto Alegre.

The French national government through its ADAE (Agency for the Development of e-Administration) encouraged the use of open source software licenses in France for administrations starting in December 2002 with the publication of a guide. This policy was reiterated later in The e-Administration Plan of Action 2004-2007 [11].

In addition, the Indonesian government declared "Indonesia go open source" in 2005 and has since encouraged the use of open source software in governmental administration offices and government owned companies. The Ministry of Research

and Technology and the Ministry of Communications and Informatics have aggressively migrated their computing system to open source and open standard. The two ministries have both sponsored the creation of Indonesian distros and declared their readiness in helping other ministries and institution in system migration. In 2009, the Ministry of Empowerment of State Officers announced that by the end of 2011, all national and regional institutions and government-owned companies have to use either open source software or legal software and asked the two aforementioned ministries to help institutions in system migration.

The main reason behind governments action in encouraging the use open source software is probably the economic point of view. Deploying proprietary software is expensive. On the other hand, many common tasks such as office administrative work can be accomplished using open source software. Only very specific and professional computing tasks such as flight simulation may not have open source tools available. Another reason may be the legal threat. Several countries have been warned of legal action if they do not take sufficient measures to reduce software piracy.

Despite the economy and legality of using open source software, the number of people using OSS is still small compared to the total number of computer users. Statistics reported that in 2010, Linux market share is only about 1%, (albeit the accuracy of this number is debatable). In contrast, the percentage of software piracy is estimated at about 30 – 40%. So why people opt to use the expensive proprietary software or use the pirated version rather than embracing the open source alternatives? Is open source software difficult to operate and use?

Harsanto has taught computing to students of elementary up to high schools for years now. On his experience, students do not have any difficulty to use open source softwares for their study. His assistant, Ihsan, teaches basic programming to the sixth grade of elementary school students using open source software. In his experience, students do not have difficulty to understand the materials.

Untung, the governer of Sragen residence, stated that the use of open source software is not a problem technically. People in offices do not find much difficulty to use OSS provided that they get enough training. Dr. Klinik, the right hand person of Untung strongly supports the statement because he has experiences to develop application using Java and open source technology.

The practices in developing Java desktop applications may not be similar to the practices in developing visual based programming. Java based development involves the activity of writing more scripts and creating custom object classes. However, as

Veri described, the overall result is excellent and yet, the product may be executed from desktops that run any operating system.

Therefore, in terms of technical aspects, embracing open source software is not a big problem. However, people are already used to particular softwares. It can be hard to get them use another software which gives similar facilities to the one they are accustomed to, even if they have to use pirated version. Changing habit can be difficult and a big challenge. However, it is not impossible especially if the benefit is clear. For first time users, they may select any software provided that there is someone near them who would teach and support them when they get problems.

It comes now to the question whether it is necessary and beneficial to get people use OSS. It have to be realized that software is expensive. An operating system would typically cost USD 100 and office suite costs about the same figure. Should we expend USD 200 to get basic software set up? We can pay less by using open source software. Moreover, software companies that produce the software may lie outside the country so that at least half of the software expense will go out to other countries.

Somebody may say, "That's ok, I don't pay that much because I use pirated software." On that case we need to remember that using illegal software is not recommended and may not be blessed. For those of you who always try hard to find pennies in Halal way, then it is not funny to work with illegal software which is not Halal according to most Islamic scholars.

Embracing open source software will boost further the spread use of the software. Developers will be more eager to contribute, create or improve the software when they know that their ideas and work are useful.

As described in the section II of this paper, with open source software students are able to see the code of programs by which they learn good programming style and techniques and learn the logic flow of a good program. Thereby, they can quickly become new qualified developers. We will all be keen to see our youngsters become good programmers and contribute to the knowledge and development of the country. With closed software, the chance to become good programmers is reduced because good programming techniques are hidden in the binary code of the software.

The existence of open source software make it possible for the community to develop customized software that fit their need. Creation of local distro is a good example. It also keep us from getting dependent to resources outside the country and enable us to be independent and stand firm. Using open source software and open standard prevent the community from being locked to a certain vendor and certain

application. When an application is felt insufficient and people are not satisfied, they can always find alternatives or even produce one.

VI. Conclusion

A lot of benefits can be mentioned when we embrace open source software and open standard. Technically, open source software is a good replacement to proprietary software. Deploying open source software is usually less costly than that of the proprietary software.

By using OSS, the community can be prevented from being dependent to outside resources. OSS users is not locked in to a particular vendor and may easily switch whenever an application is felt unsatisfactory.

Embracing open source software encourage the development and improvement of the software. On the other hand, open source software development environment encourage the community to share and care. Open source software enable the community to have more qualified developers because those who learn about software development have a good chance to look into the source code of a good computer programs. Later on, the developers may be able to develop more software or improve existing ones which will be beneficial back to the community.

Realizing the benefits of embracing the software, it is a good time to rethinking our decision regarding the utilization of software and start to consider using open source software.

VII. References

[1]   Eric S. Raymond, *The Cathedral & the Bazaar, Musings on Linux and Open Source by an Accidental Revolutionary*. O'Reilly, 2000.
[2]   Allen, R. C., "Collective invention," *Journal of Economic Behaviour and Organization*, vol. 4, pp. 1-24, 1983.
[3]   James J. Flink. *The Car Culture*. MIT Press, 1997.
[4]   David M. Berry, *Copy, Rip, Burn: the Politics of Copyleft and Open Source*, London: Pluto Press, 2008.
[5]   Rosemary J. Coombe, *The Cultural Life of Intellectual Properties: Authorship, Appropriation and the Law*, London: Duke University   Press, 1998.
[6]   Eric S. Raymond, "A Brief History of Hackerdom," in *Open Sources: Voice from the Open Source Revolution*, O'Reilly, 1999.
[7]   Leslie Horn (2011, Jan.), "Android Leap Frogs Apple in Mobile OS Market Share", *PC Magazine*. [Online] Available at http://www.pcmag.com/article2/0,2817,2375417,00.asp
[8]   Fadi P. Deek, James A.M. McHugh, *Open Source Technology and Policy.* Cambridge University Press, 2008.
[9]   Ben Klemens, *Math You Can't Use*, Washington: Brookings Institution Press, 2006.
[10] Robert W. Hahn, "Government Policy towards Open Source Software: An Overview," in

Government Policy towards Open Source Software, Robert W. Hahn, Ed. Washington: Brookings Institution Press, 2002.

[11] Laurence Favier, Joel Mekhantar, Marie-Noelle Terrasse, "Use of OSS by Local e-Administration: The French Situation," in *Handbook of Research on Open Source Software: Technological, Economic and Social Perspectives*, Kirk St. Amant, Brian Still, Ed. New York: Information Science Reference, 2007, pp. 428 – 444.

VIII. BIOGRAPHY

Husni Thamrin was born in Lubuk Linggau, South Sumatera in Indonesia, on July 18, 1970. He graduated from the Universitas Gadjah Mada, Indonesia both for his undergraduate and master degree, and pursued his PhD at the University of Glasgow, Scotland.

He is with Dept of Informatics, Universitas Muhammadiyah Surakarta and his special fields of interest includes control system modelling and simulation, embedded system development, and software engineering. He is a member of ACM since 2010. He involves in promoting the use of OSS and Open Document Format. He uses OSS most of the time which includes Linux and OpenOffice and established UMS Centre for OSS Development (POSS UMS) in 2009.