The Open Source Business Model and its Economic Implications for Nigeria and other Third World Economies

Adesesan Barnabas Adeyemo [1]
University of Ibadan
Nigeria

Abstract

In the last few years open source software products have become mature alternatives to commercially developed software and systems but do not seem to be a viable commercial option for leading software vendors because of its freely available source code. However, market trends seem to paint a different picture. While open source software has matured almost to the point where it is set to challenge Microsoft's dominance in a number of areas, paradoxically, the open source model seems at odds with this reality. Nevertheless, the open source movement has been endorsed by many key industry players who would not do so unless there was a valid commercial reason. This open-minded position is difficult to understand because the business community is still dependent on proprietary software. In this paper, various reasons that have been advanced for and against the adoption of open source systems by developing economies and the open source business model along with its economic implications is presented. The focus is the effect on the end users, software developers and government policies. The paper concludes by examining the Nigerian situation and offering suggestions on how Nigeria and other third world economies can benefit from this seemingly global paradigm shift.

Keywords: Open Source Systems, business models, economic implications.

Introduction

Open Source Software (OSS) has begun to emerge in developing nations as an alternative to proprietary software. Its popularity is dependent on the promise of cost reduction, because open source software is usually available without license costs. However, proponents of OSS argue that there are other long-term benefits, for example the creation of stronger local ICT (Information and Communication Technology) industries. Although many developing countries have recognized the benefits of adopting OSS, formulating policy that favors OSS has met with some resistance.

OSS critics argue that the GNU General Pubic License (GPL) hampers local software development, rather than encouraging business, that there are fewer hardware drivers for OSS operating systems, and proprietary software like Microsoft products are easier to install and

configure. OSS has gradually made its way into the main stream of ICT business and commerce. The proprietary vendors, rather than dismissing OSS as a difficult to configure software put together by an unruly group of hackers on the Internet, have been made to acknowledge the emergence of OSS as strong competition to existing products and traditional business models. Especially on the server computers, OSS has proven its capability to provide stable, high quality applications (Wheeler, 2003).

In developed nations, especially in the United States (U.S.), software has made a significant impact on the economy. If the U.S. experience could be translated to other countries, even on a smaller scale, the potential impact could be large. Hence, the interest in a low-cost means of developing a local software industry is easily understandable. In the 30 years between 1970 and 2000, the U.S. software industry grew at an astonishing rate of 38 percent annually (Wheeler, 2003). By the year 2000, annual software sales to end-users in the U.S. were more than $100 billion (USD). Even after leveling off a bit in the mid-1990s, the software industry grew at roughly three times the rate of the economy as a whole, even while the general economy itself was booming. As a percentage of total GDP, the software industry in the U. S. is actually relatively small. Nonetheless, as a percentage of total exports the effect of the software industry on the economy is much higher. The software industry ran a trade surplus of $13 billion in 1997; without software's contribution, the U.S. trade deficit would have been 36 percent higher (Debroy & Morris, 2004).

However, a digital divide continues to exist between developed and developing countries. Considering for example China's software industry, which is still relatively small, it constitutes roughly 2% of the Worlds software industry, whereas the U.S. market share is 40% and Europe's market share is 31%. In the U. S. 97% of software is provided by local companies whereas in China local companies provide only one third. In 2002, China's total software industry revenue is about $13.3 billion (USD), which is insignificant when compared with developed countries when considering China's 1.3 billion people, which constitutes more than one-fifth of the world's population (Wuqiang, 2003). Some key issues that have been raised that are vital for establishing the worth of Open source in the context of development. These are can developing countries create value through OSS? What are the barriers to OSS? Moreover, what practical approaches can be encouraged? There is a need to address these issues if OSS is to develop in Africa and have the potential to bring benefits to the economy and society.

The OSS Business Model

It is now clear that there are (at least) two discrete models for organizing the production of software. Both appear to be sustainable, although along different tracks. While a number of promising business, models have been developed by companies in the open source market place, only very few have been able to realize economic success compared to that seen in the proprietary software sector. Overall, the impact on the software industry and the effects on software innovation of open source are still unclear. With the growing popularity of open source software, a paradigm shift in how the ICT industry makes money is emerging, away from the traditional model based on protection of intellectual property (IP) rights towards a value-added services model (CATIA, 2003). Many open source companies generate revenue from providing ancillary

services such as support or training. Cygnus Solutions (CATIA, 2003), for example is a large US company providing support services for widely used OSS applications such as the GCC compiler, which bases its business model on providing high quality technical support to a critical mass of clients.

Recently some high profile open source companies have adopted hybrid or dual licensing schemes, whereby the copyright holders sell the software under a proprietary license as well as continue development under an open source license. Examples are SUN Microsystems StarOffice, which is based on the same source code as OpenOffice or the popular open source database MySQL, which is available both under GPL and proprietary licensing terms. Other examples can be found in the scientific sector, such as OSS companies that sell their software, often for a hefty fee, to biotechnology companies and for a reduced fee to universities for non-commercial use, and yet still makes the source code publicly available (CATIA, 2003).

Another common concern with open source is how best to leverage advantage. It seems difficult for most organizations to understand how to leverage open-source projects to their advantage. However, the open-source phenomenon poses organizations with a dilemma: By giving away one's intellectual property (the source code in this case), how do you make money?

The End-user Perspective

Cost saving is often the first argument brought forward by supporters of open source solutions. In Africa, many organizations are struggling with the substantial upfront acquisition costs for software licenses, but donations and price reductions (especially for schools and community-based organizations) often help solve this problem. Although, this itself has created a new set of problems between the software donors and its recipients, such as that contained in an exchange of letters reproduced in Grimshaw (2004) between Microsoft (South and East Africa) and SchoolNet in Namibia. One question is whether potentially higher (intangible) costs for technical support and services for open source software could offset savings on software licenses. To answer this, Total Cost of Ownership (TCO) calculations are used, which attempt to measure the real cost of deploying and using ICT over its lifetime, by looking at all the extra and often unconsidered costs with impact on it. TCO calculations should include an analysis of the risks involved in software deployment. For proprietary software, this may include the risks of price rises in licensing and lock-in to one particular vendor. For OSS, risks may include unexpected interoperability issues with proprietary desktop software and a lack of experienced support staff. According to many TCO studies, initial set-up costs make up only a fraction of product lifetime costs over 3-5 years and the costs of labor for implementation and support often outweighs initial purchase costs. However, most of these TCO studies were conducted in the computing environment in the developed world and are not directly applicable to Africa. Reasons include that the cost ratios of labor, software/hardware acquisition and maintenance may differ significantly in developing countries, where cost of labor tends to be lower. In addition, tariffs and taxes might increase equipment prices, and initial purchasing price factors features more prominently. Both proprietary vendors and independent agencies have carried out various TCO studies. It should however be noted that at present it seems that there is no standardized methodology for determining TCO (Grimshaw, 2004). In the economies of the North, where labor costs are high the costs of

software support, customization and integration are high (reflecting the labor intensity of these components) relative to the license fee for software. Therefore, when the total cost of ownership (TCO) is calculated the cost of the license fee is not a crucial component. However, in developing countries, where labor costs are low the cost of the software license becomes a relatively more important cost component. Figures for piracy from the Business Software Alliance show that there is a correlation between the piracy rate and the effective software license fee (Grimshaw, 2004).

Computer users have had to upgrade their PCs every two or three years, because the operating system is constantly being upgraded and their old PC would not work properly with the new software. However, the home PC user does not require high-end computer systems. Linux and other open source software are a natural response to the monopoly of basic commercial software. Software that is used in specific areas and, therefore, has fewer users, could have a higher price, but basic software such as operating system and office productivity tools, which are installed on every computer system, should be sold at a reasonable price so that everyone can afford them.

With increased emphasis on and pursuit of intellectual property rights enforcement at the international level, the choices available to software users are becoming more distinct. As countries are being discouraged from software piracy toward more strict compliance with standard intellectual property rules, these forces real choices. OSS represents an alternative to having to comply with proprietary intellectual property regulations and users can exercise greater choice on the way their computer systems are implemented and maintained.

The Software Developers Perspective

Unlike physical goods, software is not sold outright. Instead, it is licensed to users with the copyright holders retaining ownership of their software. Licensees gain the right to use the software subject to certain conditions. Both open source and proprietary software follow this model, with the kind of restraints placed on users distinguishing the two forms of software. One of the most frequently used open-source licenses: the GNU GPL is by the Free Software Foundation and is a means of promoting open-source at the expense of proprietary software. If a program is distributed under the GPL, all source code must be made available, free. The GPL also stipulates that any user can modify and distribute the program, either in original or modified form. Any redistribution, though, (whether of the original or modified program) must also come under the GPL. This condition has earned the GPL the label of "viral" because it typically means that once code is licensed under the GPL, any other program that incorporates that code falls under the GPL as well.

The GPL provisions were intentionally aimed at preventing open-source code from being incorporated into proprietary code. One result of the source code distribution requirements is that programmers can charge no more for programs than the cost of reproduction (which are typically quite small). If a programmer tried to charge license fees substantially above the reproduction costs for GPL software, anyone else could acquire the source code and redistribute it on their own, driving the price back down to reproduction costs. With license fees thus foreclosed, the only profit opportunities remaining are for additional services, such as software support or

training, or for complementary proprietary programs that run with or on the open-source program.

The licensing provisions clearly have implications for firms hoping to earn a sustainable return on software production. Some proprietary software firms specializing in open-source software distributions have attempted to distinguish themselves through the skill of their employees and the level of service offered by their support staffs, enabling them to charge recurrent subscriber support fees. Others offer more complicated open-source products that package multiple open-source programs together, such as a complete Linux operating system distribution, along with an easy installation program. These companies can charge for the convenience of their package, but of course, other open-source providers could easily replicate that package and lower the price. Another route taken by proprietary software firms involves combining proprietary software with open-source programs, allowing the company to charge higher licensing fees for the closed software. Regardless of the specifics, the underlying economics imply that pure open-source software production cannot generate sustainable profits, which is an important point to bear in mind when considering the arguments for governments to use open-source as a development tool (Debroy & Morris, 2004).

Open-source advocates often point to IBM as the epitome of a profitable firm with open-source offerings. In 2001, IBM spent $1 billion (USD) backing Linux (Debroy & Morris, 2004). In 2002, it announced that it had recouped this investment in full. Certainly, IBM's highly visible support of open-source software has been profitable for the company. IBM is not, however, primarily a software company. It is a services and hardware company that has successfully deployed Linux as a means to sell its services and hardware as well as its proprietary (not open-source) software. Thus, IBM's experience does not provide developing nations with a road map to large financial rewards via open-source software. To even attempt this route to economic growth, countries would first need to foster a high tech hardware industry, along with a services and proprietary software industry.

The claim that OSS will create value in developing economies rests partly on the assumption that individual software developers will not only be willing but will be positively attracted to working on OSS systems. There are some debates on this issue that question the attractiveness of working on OSS from an individual's point of view. A workshop held in March 2004 in Namibia (Grimshaw, 2004) explored these issues, including some face-to-face interviews with participants. The major barriers identified were prohibitive cost of computers and the prohibitive cost of Internet connectivity. In addition, the low cost of labor in the economies means that computers are not immediately employed as a way of improving productivity. In addition, African developers tend to be physically isolated with just a few in each city in each country (although this might not be a problem if there were fast Internet connectivity). The income from programming is sporadic and therefore unlikely to be able to support those with family responsibilities. Governments and other employees fail to look for locally produced solutions, lack of appreciation of the multiplier effects that the ICT sector can have on the economy and, theft of ideas being a problem for OSS developers (code may be open and shared and this means there is no barrier to another person copying the code and then closing it via intellectual property law).

Role of Government

A number of developing country governments are embracing open source software with an eye toward bringing wide benefits to the economy and society. In some cases, this is motivated by a patriotic desire to support local ICT business and break the power a foreign monopoly has on their ICT sector. For example, the Chinese Government is said to be supporting an initiative to develop their own OSS operating system (Red Flag Linux?), to break the monopoly Microsoft has on their desktop market. Despite generous donations from Microsoft to the Chinese ICT sector, China has decided that the development of a local ICT business based on open systems is a priority.

It is common to have people refer to the Indian software industry and many people think the Indian software industry is successful. The Indian software industry is mainly outsourced work from other countries (Wuqiang, 2003). However, India, which is largely a Microsoft environment, is also adopting OSS because with limited funds to support ICT development, they do not want to become vulnerable to sudden, uncontrollable changes in pricing by being tied to one vendor. In addition, there is the need for Indian scientist and engineers to have the technical knowledge to develop and maintain their own systems without being dependent on any vendor (Sharma, 2004). China and Peru have also raised security and transparency issues as arguments for open source deployment, as they can exert control over the use of open source encryption methods. In the case of countries saddled with large foreign debt payments, such as Argentina, policies that favor OSS have been adopted because of expected cost savings.

Economic growth and development is often an elusive goal. Thus, when a new source of economic development is offered, it attracts attention. In recent years, several policy advisors have offered OSS as a source of economic growth and development. Many developing countries are grappling with what the appropriate level of open-source and proprietary software coexistence should be for a developing economy. Open-source advocates argue that government supported open-source programs can be an important aspect of an economic development program because they can aid nations in creating a local software industry. Since open-source requires so little in the way of initial investments (that is, there are often no up-front licensing fees), proponents claim that the software model's promise is welcome news for resource constrained developing nations seeking an entrée into the technology and computing sector. One of the more qualitative arguments made regarding developing nations is based on the belief that these countries are fundamentally different than economically developed countries and, as such, that they have a "moral" right to adopt different policies. This argument frequently appears in debates over intellectual property rights, a policy area that goes hand-in-hand with knowledge-based products like software. During the 1800s, for instance, the United States took such a stance in an effort to boost local writers and publishers and refused to acknowledge foreign authors' copyrights (Debroy & Morris, 2004). Present day arguments run along similar lines: local computer users cannot afford expensive licenses and thus local governments should not have to enforce piracy rules. However, the main question that demands an answer is whether open-source software has the ability to promote economic growth?

The case for open-source software as a growth and development tool has been described as being weak (Debroy & Morris, 2004). It has been proposed that the arguments muddle reasons

for using the software with reasons the software might promote economic growth. The low initial cost of open-source software, the freedom it affords from Western-based companies and the opportunities it can provide for local programmers may well be valid points, but none of them speaks of the ability of open-source to spur economic growth or even to its ability to establish a viable local software industry. The most prominent example in the developing world of a newly emergent software industry is India, and this was without any open-source contributions. The rapid growth of India's software exports, which comprise 70% of its software industry, is due to its comparative advantage in labor. India has a large reserve of well-qualified, English speaking engineers and technicians that it has parlayed into outsourced proprietary software production for mostly Western clients. These particular circumstances raise the question of whether India's experience can be replicated among other developing countries (Debroy & Morris, 2004).

Conclusion

Decreasing worldwide telecommunications costs and software's inherent portability make software creation a natural candidate for outsourcing to developing countries with a capable supply of software developers. In fact, this is how India began its local software industry by shipping English-speaking engineers for short-term outsourced engagements abroad. While developing nations' interest in open-source is understandable, given its low investment costs and the overall appeal of software, the open-source model on its own does not appear to provide a solid foundation for profitable business operations that can meaningfully contribute to a nations' economic growth. Governments supporting open source software at the expense of proprietary software will not aid the creation of this kind of growth and may even hinder its development. While the interest of developing nations in open source is understandable, given its low investment costs and the overall appeal of software as a source of revenue, the open source model on its own does not appear to provide a solid foundation for profitable business operations that can meaningfully contribute to a nations' economic growth. The pure open-source model is not capable of supporting proprietary software firms. While the service-support model can provide sustainable profits, as the U.S. experience has demonstrated this model can only support a handful of firms at best. Proprietary software applications designed to run on open-source software appear to be the most viable profit-making option (like IBM). Since hardware and software production can be separated, developing nations can complement the progress of developed nations without necessarily entering into direct competition with them.

It is therefore recommended that while Nigeria as a country can adopt Open source technology as a low cost option that will allow Nigerians have access to essential software tools (especially operating system software like Linux, software development tools and Office productivity software) needed to build up the countries ICT capacity, government should assist with Open source Research and Development, Education and Training efforts, and allow Open source software solutions to compete with proprietary options during software acquisition and procurement processes. This will have the effect of freeing the country from overdependence on proprietary software vendors and ensure that sensitive software systems (for example, those used for defense and other high tech systems) are directly under the control of the government. However, while the adoption of a hybrid licensing scheme and the promotion of a service-support open source software industry may be experimented, adopting the pure open source business

model is not recommended for the fledging Nigerian software industry as a means of generating revenue that can sustain the country's economy.

References

CATIA 2a Inception Report, (2003). *Open Source Software and Low Cost Computing*. Retrieved January 2005, from http://www.catia.ws/Documents/Indexpage/Inceptionreport.pdf

Debroy, B., & Morris, J. (2004). *Open to Development: Open-Source Software and Economic Development*. Retrieved January 2005, from http://www.policynetwork.net/pdfs/open-to-development-2004.pdf

Grimshaw, D. J. (2004). *The Intermediate Technology of the Information Age? An Assessment of the Implications of Open Source (Free) Software for Development*, Intermediate Technology Development Group (ITDG). Retrieved January 2005, from http://itdgnet/group/knowledgebase/documents/35513.pdf

Sharma, D. C., (2004). Indian president calls for open source in defense, *CNETAsia*. Retrieved January 2005, from http://asia.cnet.com/news/software/0%2C39037051%2C39186139%2C00.htm

Wheeler, D. A. (2003). *Why Open Source Software / Free Software (OSS/FS)? Look at the Numbers!* Retrieved January 2004, from http://www.dwheeler.com/oss_fs_why.html

Wuqiang, L. (2003). *Open Source Software and China's Software development: Strategies for Building Software Industries in Developing Countries*. Retrieved January 2005, from www.iipi.org/activities/forums/softwareconference/Li%20Paper.pdf

---

[1] Mr. Adesesan Barnabas ADEYEMO is a Lecturer at the Computer Science Department, University of Ibadan, Oyo State, Nigeria. He can be reached at: No 17 Ikale Road, Ijapo Estate, Akure, Ondo State, Nigeria. E-mail: sesan_adeyemo@yahoo.com or sesnet2003@yaho.com; Phone: +234805 210 7367.