Government Policy toward Open Source Software: The Puzzles of Neutrality and Competition

Jyh-An Lee

For a variety of policy reasons, governments throughout the world are now adopting different legislative and administrative strategies that support the development of FLOSS. Some governments have actually begun to procure FLOSS, whereas others have channeled public funds to large-scale FLOSS projects. This study demonstrates both the benefits and the risks of government policy favoring FLOSS from the perspective of economics, technology, and politics, and to further analyze whether these same policy goals can be achieved through government support of FLOSS. The most fundamental argument of the study is that, in lending its support to FLOSS, the difference between a government user and a business user is that the government should take into account society's long-term interests, not merely its own interests as a consumer.

Introduction

In recent years, both private-sector corporate CIOs and public-sector policymakers are beginning to recognize the importance of free, "libre", open source software (FLOSS) in their organizations and refine their attitudes toward it. For different policy reasons, governments all over the world are now adopting various legislative and administrative strategies that support the development of FLOSS. Some governments have begun to procure FLOSS directly, whereas others have begun channeling public funds to large-scale FLOSS projects. However, the issue of whether such pro-FLOSS policies are optimal for society as a whole remains intensely controversial and hotly debated in many countries. This study will analyze the trend and policy considerations behind these governments' preference for FLOSS. The aim of this

---

Jyh-An Lee is a J.S.D. candidate at Stanford Law School. His research interests include free/open source software, intellectual property, and digital technology policy. He may be reached at <jalee@post.harvard.edu>. He would like to gratefully acknowledge the guidance of Jonathan Greenberg, Larry Lessig, Charlie Nesson, and Tony Reese.

Knowledge, Technology, & Policy, Winter 2006, Vol. 18, No. 4, pp. 113-141.

paper is to provide some basis for further discussion on both legislative and economic policies regarding FLOSS.

There are strong network effects in the software market,[1] and the cost of switching between proprietary software and FLOSS is consequently significant for software users. These characteristics are important for any further analysis of government intervention in the software market. The unique features of FLOSS, as well as the incentives associated with developing it, both of which are obviously different from proprietary software, are closely related to government policy toward FLOSS and proprietary software. This study investigates why it is that so many governments are considering supporting the development of FLOSS, weighing the pros and cons of such policies. This study has found there are numerous factors affecting governments' policies toward FLOSS. Such factors may be economic, technical, or political. In conclusion, this study argues that when two systems are equally suitable, government could reasonably choose FLOSS over proprietary software because FLOSS's unique merits will not only help developing countries to build their information technology capabilities, but will also promote competition in the software market. Furthermore, this study holds that, the market failures in the software market will justify governments' support of FLOSS development. However, governments should still cautiously evaluate the social costs and benefits of supporting FLOSS before making such decisions.

This study is also limited in some ways. Since the policy issue regarding FLOSS development is quite new to academics internationally, the analysis underlying this paper is to approach related issues in a comparative national context across geographical regions, socio-economic strata, and political systems. The author chose not to focus merely on a few jurisdictions in order to provide a more complete picture of the underlying policy issues globally. Although this study does not intend to ignore the differences in cultures and socio-economic situations between governments around the world, as well as their different policies for their local needs, the author has decided to approach the underlying issue in a more general way, rather than observing government policies in specific countries. Nonetheless, subsequent research is recommended to focus on government policies in specific economies. Besides, this study does not elaborate on the differences between various types of FLOSS. Different FLOSS projects may have contrasting incentives, norms, and other characteristics. These factors will certainly affect governments' attitudes toward FLOSS. Although this study focuses primarily on operating systems, other applications and server systems are discussed in some sections as well.

Governmental Projects Supporting FLOSS

A variety of regulatory alternatives are available to governments that choose to support FLOSS. Governments may choose to promote FLOSS through legislation, administrative rules, guidelines, procurements, subsidies, industrial policies, or other public measures.

Adopting FLOSS or Open Standard Software in the Public Sector

Using FLOSS in the public sector is, so far, the most widely adopted policy with regard to government support of FLOSS. The market for government software procurement is crucial to software companies, not only because government is usually the largest consumer of software products in a country, but also because a government's use of specific software may encourage individuals and businesses that deal with the government to follow suit.

A government may either "consider" or "prefer" procuring FLOSS for its agencies. If a government begins to "consider" adopting FLOSS, it may thereby signify its desire to establish a level playing field for FLOSS within the public sector's IT procurement policies. However, such policy is not actually pro-FLOSS policy, because it neither constitutes a government preference for FLOSS nor means that the government will necessarily choose FLOSS in a final procurement decision. In deciding between FLOSS and proprietary software, the government still has to be concerned with such factors as efficacy and efficiency. The pending House Bill 2892 in the U.S. state of Oregon provides an example of such legislation. In March 2003, the Oregon legislature introduced this bill, which would require the state's agencies to "consider the use of Open Source Software" when proceeding with all new software acquisitions (Dravis, 2003).

If a government decides to "prefer" FLOSS to proprietary software, the decision will almost certainly be criticized as procurement discrimination by proprietary software developers. The difference between "consideration" and "preference" is well illustrated in the recent Government Procurement Guideline Amendment of the Australian Capital Territory (ACT). The original bill, which was proposed in July 2003, in fact provided that "[each government] entity should, as far as practicable, prefer open source software to proprietary software." The bill was later amended, however, to substitute "consider" for "prefer" (Gedda, 2003). The final usage of "consider" in the bill may indicate the government's intent to remain neutral on the issue. The decision may also reflect a compromise in the exercise of political power by the government to encourage use of FLOSS over proprietary software.

Governments can establish a more modest FLOSS procurement policy by requiring their agencies to procure software that complies with "open standards." But the definition of *open standard* is still unclear. According to the Australia ACT Government Procurement Guideline Amendment of 2003, *open standards* aim to limit "software for which support or maintenance is provided by an exclusive entity that has the right to exercise exclusive control over its sale or distribution," and includes benchmarks recognized by the ISO.[2] The Enterprise Open Standards Policy enacted by the U.S. state of Massachusetts, in contrast, defines an *open standard* as comprising "specifications for systems that are publicly available and are developed by an open community and affirmed by a standards body."[3] The policy gives HTML as an example of such a standard and adds, "open standards imply that multiple vendors can compete directly based on the features and performance of their products. It also implies that the existing information technology solution is portable and

that it can be removed and replaced with that of another vendor with minimal effort and without major interruption" (Turner, 2004).

Although the definition of *open standard* varies in different jurisdictions, it is generally understood to contain the following characteristics: First, the standard is publicly available at a minimal cost. Second, no entity controls the standard, or the standard is licensed on reasonable and nondiscriminatory terms (Kesan & Shah). Examples of open standards commonly found in software and on the Internet include the transmission protocols FTP and HTML, which serves as the language for web pages, and the image format known as JPEG.

Compared to legislation calling for a mandatory adoption of FLOSS, an open standard requirement in government procurement rules seems less partial to FLOSS. If the policy goal of government software procurement is only to promote compatibility or interoperability between different software products, an open standard requirement may be less controversial and more effective than an FLOSS requirement in governments' software procurement rules. However, interoperability is often not the only policy goal in governments' software procurement.

Subsidies for Specific Open Source Projects

Another way for governments to support FLOSS is to subsidize certain FLOSS projects. The target of public subsidies may differ from government to government. Alternatives for public subsidies may include, but are not limited to, the following:

A. The subsidizing of projects for the training of FLOSS developers;[4]
B. The subsidizing of institutions that try to coordinate FLOSS development;[5]
C. Indirect subsidies in the form of tax deduction or other grants to FLOSS projects;[6]
D. The outright funding of specific FLOSS initiatives;[7] and
E. Governments' direct involvement in FLOSS projects.[8]

Lessons Learned

For governments that prefer FLOSS to proprietary software, the most common practice is to support FLOSS through software procurement. Nonetheless, despite the fact that more and more public sectors have begun to migrate from proprietary software to FLOSS, it is quite difficult for government to legislate such a preference. So far, most of the legislative proposals that adopt a "preference" stance either have been defeated or are pending. An increasing number of governments have made known their intention to support FLOSS without any legislation, but rather by promoting FLOSS directly in their procurement decisions. Another interesting trend is that many governments are now beginning to put an open standard requirement, instead of a FLOSS preference, in their procurement guidelines so as to avoid being locked-in by proprietary software companies.

As Evans and Reddy point out, the respective advantages and disadvan tages of FLOSS and proprietary software are the mirror image of each other:

what is advantageous for FLOSS is disadvantageous for proprietary software, and *vice versa* (Evans & Reddy, 2003). The policy considerations behind governments' decisions are extremely complicated and sometimes interdependent. As Lessig states,

> the factors that determine efficiency for governments are fundamentally different from the factors that determine efficiency for private sectors. Governments are not competitors in the sense that private actors are. They have a great interest in externalizing benefits that other competitors might share (Lessig, 2002).

In the following section, this study will analyze why governments nowadays have begun to consider replacing proprietary software with FLOSS, and whether these policy goals can be achieved through government support of FLOSS. This study approach the field of policy issues from the perspective of economics, technology, and politics.

Economic Concerns

Government policy toward FLOSS should have an economic rationale. Much of the recent public-sector interest in FLOSS is motivated by public demands for cost-saving strategies and actions. The expectation is that the absence of up-front license fees and the availability of community-based support can lead to lower costs. Competition in the software market is also an economic concern that governments may have when they support the development of FLOSS. The recent Microsoft antitrust cases have also convinced some governments that supporting FLOSS is the only way to sustain a level playing field in the software market.

Cost

FLOSS and proprietary software are two different systems for producing software. By making source code openly available, FLOSS not only facilitates low costs, large scale, and parallel innovation of software, (Lerner & Tirole, 2002) but also reduces the marginal cost of using it to zero (Evans & Reddy, 2003). Today governments all over the world have noticed that they have been spending enormous amounts for licensing fees on Microsoft software; FLOSS thus becomes an ideal means by which governments can attempt to substantially lower costs in software acquisition.[9] Some commentators believe that FLOSS provides a means to extend the market for software because it serves those consumers who cannot afford to license proprietary software products (Bessen, 2002).

Economists have indicated that, in order to cover significant fixed costs, proprietary software developers cannot price their products at marginal cost. FLOSS, because it can be priced at marginal cost, seems obviously to be more efficient (Schmidt & Schnitzer, 2003; Evans & Reddy, 2003). Furthermore, the ability of FLOSS community members to copy and distribute the code also

constrains pricing for the code itself (McGowan, 2001). Compared to proprietary software, FLOSS could be made more widely and cheaply available. In addition, there are strong incentives for proprietary software companies to charge locked-in consumers prices above cost due to the significant cost of switching between software systems (Schmidt & Schnitzer, 2003). Lessig indicates that, between FLOSS and proprietary software, all things being equal, public policy should favor the former because it brings the cost of information down to its marginal cost (Lessig, 2002).

Benkler provides another argument from the viewpoint of transaction cost economics. According to Benkler, the peer production of FLOSS "has an advantage over firms and markets because it allows larger groups of individuals to scour larger groups of resources in search of materials, projects, collaborations than do firms or individuals who function in markets. This is because when production is organized on a market or firm model, transaction costs associated with property and contract limit the access of people to each other, to resources and to projects, but do not do so when it is organized on a peer production model"(Benkler, 2002). To put it differently, Benkler believes the peer production of FLOSS means "a central input—pre-existing information—could be available to human productive agents without limit," whereas the production of proprietary software "creates a boundary around the set of available agents and the set of available resources that limits the information available about what other agents could have done with these same resources or what else these agents could have done with other resources" (Benkler, 2002).

Schmidt and Schnitzer, however, hold that FLOSS's marginal cost pricing is inefficient because it gives insufficient incentives for software developers to engage in R&D. Schmidt and Schnitzer believe that the limited monetary rewards available to FLOSS developers will tend to limit the effort devoted to these activities (Schmidt & Schnitzer, 2003). Evans and Reddy also contend that the lack of property rights associated with FLOSS results in the fact that firms can charge little more than their distribution costs, and that they thus have little or no incentive to devote substantial resources to the development of new software (Evans & Reddy, 2003). Evans argues further for proprietary developers that, while consumers can benefit from the associated efficiency, the dominant proprietary software company is justified to have monopoly power and may, therefore, charge more than a competitive firm (Evans, 2002).

The arguments above may be true for proprietary software developers because it is unreasonable to expect them to expend—without the promise of any profits—great effort for the production of software products. Thus it is understandable that proprietary software would charge prices above the marginal cost for their software products. Marginal pricing, without providing enough financial incentives for proprietary software development, is, to some extent, an inefficient way to produce proprietary software because it would result in software produced at a less than ideal level. Nonetheless, this study holds that FLOSS must be analyzed within a different context because the incentives for programmers to engage in FLOSS development are entirely different from those for proprietary software developers. Past development of

FLOSS has proven that FLOSS programmers' efforts are not driven by direct monetary incentives. Therefore, the viewpoint that there are insufficient incentives for FLOSS developers may not exactly prevail here. In the final analysis, at least, we should not arbitrarily conclude without providing empirical evidence that marginal pricing provides insufficient incentives for FLOSS development.

Licensing costs are a fraction of the cost of ownership associated with most software products. Additional outlays for customizing, training, maintenance and support may negate licensing cost savings. Because proprietary software companies only distribute object code, users, including governments, are entirely dependent on the software companies to provide debugging and upgrades. It is believed by some that the cost of debugging especially complex software problems are so high that FLOSS can often bear those costs better than proprietary software (Bessen, 2002). FLOSS advocators also argue that the fully modifiable code accompanying FLOSS is better than the proprietary closed code alternative in terms of maintainability and extensibility; this fact may legitimately result in legislation or regulation that make it difficult for proprietary software suppliers to compete against FLOSS for government contracts (Moglen, 2002). Furthermore, FLOSS permits an extremely large labor force (potentially the entire Internet community of programmers) to bring its skill and insight to bear on a problem.

Nonetheless, though adept users are able to create bug fixes for FLOSS more quickly than for proprietary software, whether such bug fixes can easily be put into the hands of general users and governments is still unclear (Evans & Reddy, 2003). Making one's source code available does not guarantee that thousands will flock to view it and fix it. The patching of FLOSS depends on the charitable instincts of volunteer testers (Zittrain, 2004), as well as the willingness of government users to hire someone to fix it for their particular purposes. Furthermore, proprietary software companies argue that FLOSS solutions tend to be more customized than their proprietary counterparts and FLOSS solutions will often require more sophisticated, and thus more expansive, support and maintenance (Smith, 2002). But, at the very least, it is fair to say that the adoption of FLOSS could prevent the market of software debugging from being monopolized by certain software vendors, and government users could thus avoid being locked-in by such vendors.

Network Effects and Switching Costs

Network effects are very important in the software market and such effects may cause the market to favor only one form of software. Therefore governments should assess such effects carefully when they intend to intervene in the software market or when they attempt to promote certain policy goals through software procurement. A government's adoption of specific software may result in strong network effects in the software market, not only because of the government's market share, but also because of its impact on the choices of individuals and businesses that deal with the public sector. Theoretically, a government could make the market tip toward FLOSS if it decided to adopt

FLOSS extensively, but so far we have no empirical evidence to show that governments' adoption of FLOSS would actually lead to market tipping.

In the worst case scenario, if FLOSS were to capture the whole market through strong network effects, current software companies would have few incentives to innovate, and potential proprietary software market entrants would also be discouraged if they felt that they could not compete with FLOSS favored by the government (Schmidt & Schnitzer, 2003). If this is the case, innovation in the proprietary software market will be impeded while incentives for innovation in the FLOSS community may still be the same. Such a result does not benefit government, consumers, or proprietary software developers.

Network effects lie in the adoption of not only identical goods, but also in the adoption of compatible ones. If FLOSS and proprietary software were compatible, the network effects could be enjoyed by all the consumers, instead of just the dominant player(s) that inhabit the market. In that case, governments would not have to worry that their adoption of FLOSS might negatively effect software innovation.

Even if the production and debugging costs of FLOSS are much lower than those of proprietary software, government still has to take other costs incurred by the initial adoption of FLOSS into account. For instance, switching costs are considerable whenever a government procures new software to replace an existing system. In fact, substantial switching costs may deter governments from adopting FLOSS. Such costs are certainly high because the investment in proprietary software has to be duplicated. Furthermore, it may be socially inefficient to switch to superior software if there is already a large installed base and if the switching cost is high (Schmidt & Schnitzer, 2003).

However, if such switching affects the terms of trade for software products and consequently reduces costs for users in the long run, it might still be worthwhile for organizations to make the switch. Recently, the switching costs have become increasingly prohibitive as governments continue to use specific proprietary software. Thus, governments even have begun to worry about being locked-in by Microsoft, and some believe that FLOSS might be a desirable way to seek greater technology independence.[10] If governments do not want to be locked-in even further by Microsoft, switching to FLOSS might be an ideal strategy regardless of functionality and related switching costs. Moreover, compared to business users, governments certainly have some different concerns regarding software procurement, such as national security and industrial policy. These concerns, which will be discussed later in this paper, may alone outweigh the enormous cost of switching.

In addition to the switching cost of moving from proprietary software to FLOSS, governments are also expected to measure the switching cost of moving back to proprietary software once they have adopted FLOSS comprehensively. If the switching cost of moving from FLOSS to proprietary software is too high, FLOSS may capture the entire market, even if it is eventually proven to be inferior to the proprietary software (Schmidt & Schnitzer, 2003). At the same time, proprietary software developers whose products are not used by the government may have fewer incentives to invest in quality improvements

if they have no opportunity to regain the market share. Nevertheless, if governments can provide a level playing field for FLOSS and proprietary software thereafter, there may still be incentives for proprietary software companies to supply a product that is compatible with FLOSS. In this case, switching costs will no longer be an issue, regardless of the training costs; governments, even all consumers, could truly choose the best product for their needs, based on the true value of the products to the institutions or individuals.

Based on the above discussion of switching costs, we can see that software incompatibility is a major obstacle to migrating between software systems. If FLOSS and proprietary software were compatible, the switching costs would be manifestly lower, and governments could truly gain technological independence. Such compatibility could also increase social welfare by allowing consumers to choose the most appropriate software products for their needs, without worrying about significant switching costs. Finally, the goal of efficiency could, in fact, become a reality if genuine competition were to surface between FLOSS and proprietary software. Such concerns of compatibility and lock-in effects are well illustrated in the "Open Source Software Trials in Government Final Report" released by the British Office of Government Commerce (OGC) in 2004. In this report, the OGC claims its priority is to "avoid lock-in to proprietary IT products and services," and maintain "interoperability that support open standards" (OGC, 2004).

Underproduction of Public Goods

Schmidt and Schnitzer argue that public subsidies for R&D should be limited to basic research, rather than applied research, because the former is a public good with strong positive external effects that will not be provided by the market. The authors point out that there are few incentives for firms to conduct basic research, as these same firms are unable to capture the spillover value of that research. Firms usually find it much easier to internalize the positive effects of applied research through intellectual property laws and the market. Thus, Schmidt and Schnitzer reach the conclusion that, because most software products are applied R&D, governments should ensure that subsidized research enters into the public domain or that software be protected by liberal licenses such as the BSD.[11] They argue that the viral nature of the GPL [12] will preclude proprietary software companies from using it. This fact will thus encourage the development of two incompatible networks with significant welfare losses for consumers (Schmidt & Schnitzer, 2003).

Regarding Schmidt and Schnitzer's point, first, the definition of *basic research* is still unclear. If the difference between *basic research* and *applied research* is whether the researcher or the developer can internalize the positive effects of the research, then the development of FLOSS is obviously basic research, while proprietary software is an aspect of applied research. FLOSS developers have never attempted to capture the value of FLOSS through intellectual property laws or market price. Instead, they disseminate FLOSS as widely as possible. Furthermore, the incentives for FLOSS development are more similar to those for basic research, rather than applied research. Basic

research is usually incentivized, so to speak, by a reward system that involves prestigious academic appointments and the prospect of the Nobel Prize and other prizes, whereas applied research is incentivized by intellectual property rights (Landes & Posner, 2003). The incentives associated with the former are somehow more similar to the incentives associated with FLOSS development, which encompass recognition from others and intellectual satisfaction, rather than direct pecuniary compensation. Therefore, it is not reasonable to consider all software development as applied research and, thus, arbitrarily disregard the possibilities of public subsidies for FLOSS.

Moreover, the viral nature of the GPL and the concerns of proprietary software companies do not affect the underlying "public goods nature" of FLOSS at all. In addition, one may argue that the primary goal of subsidies for research is to increase education and the body of knowledge, whereas the secondary goal is to be able to commercialize the research. By keeping the source code open, FLOSS can achieve this primary goal far better than its proprietary counterpart (Clark, 2003).

This study holds that the question of whether governments should support the development of FLOSS is not totally analogous to their funding of basic research. The main reason that governments harness their budgets to incentivize basic research hinges on a desire to avoid the underproduction problem of public goods. One successful example of government's sponsor of the nonproprietary inventions is the basic architecture of the Internet, which includes the TCP/IP standard. During the Internet's early years, the U.S. government supported and encouraged a culture of nonproprietary development that protected the open architecture of the Internet. This development path can be justified by the fact that the TCP/IP and other basic Internet protocols constitute classic public goods that would not necessarily be provided by the market.

However, in light of the burgeoning development of the software industry, most software products are not under-produced at all. The current intellectual property regime has already provided enough incentives for software production. Perhaps the more fundamental problem here rests not on whether software development is a type of basic research, but on whether the current intellectual property regime, as a set of incentives for software production, has created costs that are too high for individual consumers and the society as a whole to bear.

Governments usually have different approaches to solving the problem of the underproduction of public goods. Public subsidies and intellectual property rights are just two of them. With regard to software products, for decades governments have adopted intellectual property rights to solve the public goods problem. Because intellectual property legislation presents no direct, immediate cost to a government, it seems to be a relatively cheap approach to solve the public goods problem (Mergers et al., 2003). Lessig, nonetheless, argues that, by combining the overlap protection with the law and the code, proprietary software companies create too large a gap between the software price and its marginal cost, a gap that may not be justified by the need to solve the product provision problem (Lessig, 2002). Intellectual property rights may, to

some extent, solve the problem of underproduction, but without the attribute of non-excludability, software products may be under-consumed, since some individuals with a need for the product will not purchase it. In other words, the problem may lie in the under-consumption, rather than in the under-production, of public goods. If the costs exceed the benefit they bring, governments certainly should consider developing and implementing alternative incentives, such as public funding, for software production.

A growing body of literature demonstrates that software developers have sufficient incentives to participate in FLOSS development, even though they will not be able to capture the full profit value of what they produce. Eric Raymond contends that there is a "gift culture" in the FLOSS community that incentivizes the production of FLOSS (Raymond, 1998). This study holds that, in the FLOSS community, there are a variety of reasons for gift-giving apart from the altruism identified by Raymond. If we interpret the concept of gift-giving broadly, we may deem the efforts that programmers make for the development of FLOSS as a sort of gift, even if those efforts do not flow from purely altruistic motives. Consequently, we may find some specific reasons for the state to support such gift giving.

FLOSS and basic research have a similar attribute of gift-giving. Professor Shavell has indicated that, if an organization, for example, a university, "is furnishing a public good, providing a benefit to society generally that cannot be provided by the private sector, then one way to finance it is by encouraging those who would give for whatever reason to give more, by subsidizing giving"(Shavell, 2004). In the context of FLOSS development, because FLOSS programmers do not take the value of gifts to donees into full account, but that should be done from the perspective of promoting social welfare, programmers may give too little, and the subsidy of FLOSS may be desirable.

Suppose that if a FLOSS developer A were to give a gift to B, A would obtain an altruistic benefit of 35, that B herself would obtain a benefit of 70 from the gift, but that the gift would cost A 40 owing to the consumption she would forgo. The program developer A therefore would not give the gifts: the altruistic benefit to her of 35 is outweighed by the cost to him of 40. But it is socially desirable for the gift to be given, assuming a sum-of-utilities measure of social welfare; for if the gift is given, the net change in welfare will be positive, 35+70-40=65. A subsidy for FLOSS could introduce A to give the gift, and therefore might be socially advantageous.

This proposal for government support of FLOSS seems plausible, but this example also suggests that through market mechanisms, B could pay A no less than 5 to produce the gift, and public subsidy is, thus, not necessary to produce such software products. Nevertheless, in the real world, B means a huge number of dispersed software users. Given the significant costs of collective action between those users, governments, representing the wide interests of their citizens, may legitimately subsidize FLOSS developer A. A more fundamental question here is, again, whether there is an under-production problem for public goods. If such a problem does not exist, i.e., the market itself could produce enough software, government subsidies for gift giving by

FLOSS developers is superfluous, unless FLOSS is proven to be superior to proprietary software, and the benefits of subsidies outweigh its costs.

Market Competition

It is true that the rapid growth of the FLOSS movement has exerted competitive pressure on proprietary software companies, such as Microsoft, and constrained their pricing.[13] But whether government intervention in the software market—in the form of favoring FLOSS—will promote or impede competition is still fiercely debated. Scholars has suggested that the playing field is not level for proprietary software and FLOSS due to their different natures and current intellectual property regimes. For example, Lessig believes that a system with software patent, which proprietary software developers have taken advantage of to strike at their competitors, is biased toward FLOSS (Lessig, 2002). Zittrain also argues that FLOSS, compared to proprietary software, is much more vulnerable to claims of infringement by proprietary code authors, since the source code of FLOSS is freely available to would-be plaintiffs. In addition, that availability also makes the costs of stealing copylefted software typically lower than the costs of stealing proprietary code because one can always get the source code of FLOSS whereas she or he can never easily get that of proprietary software (Zittrain, 2004).

Some proponents of FLOSS assert that the government should support FLOSS in order to put additional pressure on proprietary software companies, a strategy that might force them to continue lowering their price.[14] Antitrust cases in the software industry raised the profile of Microsoft and, indeed, identified it as the industry's most dominant vendor. Researchers in the antitrust field have asserted that, without government intervention in the software market and government support of OSS, the software market would be monopolized by a very small number of proprietary software companies.[15] They believe that proprietary systems entail a strong tie to a single supplier, and in reality this precludes competition. Lessig provides some powerful rationales for such a viewpoint. He believes that FLOSS "risks none of the dangers of strategic behavior that closed code, or controlled networks, do." He holds that "if the PCs that the government owned ran something other than Windows, then the market for these alternative platforms would be wildly expanded. And if the market for alternatives were strong, then the benefits from building for these alternatives would be strong as well" (Lessig, 2001).

An indirect benefit of the consideration of FLOSS in public sector procurement is that it can be used to boost a user's negotiating position. This is particularly important in product or geographic markets dominated by a single vendor. Lessig believes that government encouragement of FLOSS actually represents a policy favoring diversity, as opposed to a policy against Microsoft (Lessig, 2001). Deploying FLOSS within organizations and promoting a level playing field for FLOSS and proprietary software can break a monoculture, reduce dependency on any one vendor, and, in the process, enable these organizations to negotiate better terms for proprietary software deployment.

Free market advocates, however, oppose public subsidies for FLOSS projects and any other governmental intervention in the software market (Schmidt & Schnitzer, 2003). Some proponents of the open source movement have also come to the same conclusion and believe that the FLOSS "bazaar" will succeed on its own merits. These experts think that technology should compete on its merits in a free market and that governments should not pick winners and losers. If FLOSS is indeed superior, IT specialists in business and government will use FLOSS; they do not need legislation or legislators to make that decision for them. Economists also indicate that governments have a poor record in picking winners through industrial policies (Evans, 2002). They have neither the ability nor the right incentives to determine whether FLOSS or proprietary software is more efficient. Compared to the government, the market is far superior at deciding which products are best for consumers; therefore, we should not be so quick to abandon the current market, unless there is a market failure. When a market failure occurs, there is a potential rationale for government intervention. Nevertheless, most economists believe that government failure resulting from its intervention in a "failed market" may cost the public more than the original market failure. Some economists even hold that the characteristics of the software industry already suggest that competitive health can benefit consumers (Evans & Reddy, 2003).

Based on these arguments, public policy should not favor specific groups or corporate interests. Both a government's preference for FLOSS in procurement and public subsidies for specific FLOSS projects could undermine market mechanisms. Governments' preference for FLOSS might further frustrate proprietary software developers' incentives to innovate, and the price of proprietary software might increase, if proprietary software developers foresee that they cannot fairly compete with FLOSS either in price or in quality. Nevertheless, it might be legitimate for governments to promote compatibility standards to serve public needs.

If we inquire thoroughly into the controversy of government support of FLOSS and its relation with market competition, we will find the fundamental question here lies in what we expect governments' role should be in market competition. The most common arguments with regard the subject debate are government's neutral role in technological development, and its role in promoting competition. Therefore, I will further analyze these two functions regarding government support for FLOSS in subsequent paragraphs.

The principle of *technology neutrality*, which has been globally adopted in legislation regarding electronic signatures, means that government should not stifle the development of certain technologies or unfairly favor one technology over another. According to this principle, between different technologies or implementation schemes, the choice must be that of the subject parties or the users, instead of the governments or the policymakers. This approach exhibits a degree of confidence in the marketplace's capacity to make suitable options available to the market, options that would enable the market to make intelligent choices.

From the perspective of technology neutrality, market competition will decide whether FLOSS or proprietary software will ultimately prevail. What gov-

ernments should do certainly excludes picking the winner, but definitely includes maintaining a level playing field in the software market. That is, the government should not intervene in the market unless there is a market failure. Nonetheless, three questions should be answered before we come to the conclusion that government support of FLOSS will be detrimentally unneutral to different technologies. First, what kinds of FLOSS policies are unneutral? Second, is there a market failure in current software market that can justify government's intervention? Third, can a government, supported by free market advocates, always ideally avoid any kind of intervention into market activities?

The "consideration" type of FLOSS legislation is obviously a neutral policy since it does not give any preference to either FLOSS or proprietary software. If a government decides to adopt FLOSS as a result of costs and quality concerns, such decision is neutral to both FLOSS and proprietary software as well. Nevertheless, the "preference" type of legislation is extremely controversial with regard to neutrality because such legislation clearly places a preference on FLOSS for governmental use. Therefore, the discussion of government's neutral role in software development and its support of FLOSS should be focused on the "preference" type of policy, rather than on the "consideration" one.

Although strong network effects exist, it is difficult to conclude that software market tipping is a market failure. Yet, one can prove that FLOSS is demonstratively superior to proprietary software, but is still not adopted widely in the market. However, other types of market failure may exist in the software market. For example, due to strong intellectual property protection, software products may be under-consumed, since some individuals with a need for the product will not purchase it. Furthermore, software incompatibility can be viewed as a market failure as well. In developing countries, the lack of skilled programmers and programming education may also be deemed as a market failure. These market failures as a whole may, to some extent, justify government intervention into the software market through support of FLOSS.

Since intellectual property system itself is a government intervention, this study holds that it is too idealistic to argue that government should avoid all kind of intervention into the software market. The more fundamental problems here are what is the optimal government intervention and how do avoid government failure, rather than how to avoid government intervention. Nevertheless, these fundamental problems have been extremely arduous for numerous governments around the world.

Technical Concerns

In addition to economic concerns, one of the most important considerations in adopting FLOSS reasons for the adoption of FLOSS is the potential technical problems, including quality issues (such as stability, maturity, security, and adherence to standard) and flexibility (for instance, modularity and ability to integrate or customize through access to source code). Below, this study will analyze related technical issues from the viewpoint of compatibility, security, usability, and availability.

Compatibility

Compatibility is an issue of concern in adopting any software, including FLOSS and proprietary software. Incompatibility between FLOSS and proprietary software is a problem that plagues any government's adoption of FLOSS.[16] Even the FLOSS itself may lead to multiple incompatible versions of the same software. Incompatibility frustrates consumers from replacing their proprietary software with FLOSS because network effects and switching costs already exist for proprietary software. If such incompatibility is not warranted for technological reasons, but is posed by proprietary software companies to deter FLOSS or other potential competitors, it may result in an anti-competitive effect that decreases efficiency.

There are strong network effects in the software market. Users, including governments, should carefully assess whether software interfaces exhibit such effects when they consider switching to different software. According to the view of many economists, by allowing competitors to use program interfaces, users of different programs could benefit from network effects of different networks. Therefore, whether network effects and switching costs limit competition depends on whether vendors have proprietary control of the interfaces. If interfaces are open, competitors can produce compatible software, and users can then choose a software product in accordance with its quality, as well as its price, instead of its switching cost and number of users (Lemley & McGowan, 1998).

To this end, governments could consider imposing expansive compatibility requirements for software interfaces, but such regulation, one can allege, directly expropriates the incumbent's *ex ante* investment. If the law decreases the intellectual property protection on software interfaces, it would not affect the incentives for programmers to participate in FLOSS development at all. After all, the interfaces associated with FLOSS are already open since they are described in the source code. Some scholars thus claim that one of the benefits of supporting FLOSS is the resulting pressure brought to bear on proprietary software developers to open their own interfaces, to the benefit of users (Varian & Shapiro, 2003). From this perspective, FLOSS is better than proprietary software when it comes to increasing compatibility and network effects.

A more moderate alternative for government might be to limit its procurement to open-standard software in order to promote compatibility and network effects. Also, governments could consider subsidizing neutral institutions and coordinating the setting of both standards and the compatibility between FLOSS and proprietary software. Without the government's financial support for collective action effort involving different software developers, it will be difficult for the private sector along to spearhead such efforts. From this standpoint, governments can legitimately coordinate the standards and compatibility between FLOSS and proprietary software to enhance social welfare.

Schmidt and Schnitzer believe that if the FLOSS favored by government is licensed by the GPL or other viral licenses, proprietary software developers would find it legally difficult to make their software compatible with FLOSS. Then, there would be two incompatible networks existing in the market

(Schmidt & Schnitzer, 2003). Evans and Reddy have also ascertained that it is the distribution restrictions of the GPL that makes proprietary software and GPL software uneasy to coexist. Evans and Reddy has argued that since the government-sponsored GPL software can neither go to public domain nor be spun off for commercial purpose, this is a bad policy for the government to support software R&D that is licensed under the GPL (Evans & Reddy, 2003).

Because the GPL license is viral in its nature, proprietary software developers would be much more cautious not to be infected by FLOSS under the GPL license. Nonetheless, proprietary software's license terms may have the same effect, as the GPL license does, on software compatibility. Without license, FLOSS programmers could not legally appropriate the source code of proprietary software just as proprietary software could not appropriate the source code of GPL software without being infected. Hence, this study holds that proprietary software's influence over software compatibility does not obviously differ from GPL software's influence over software compatibility. And it would be too arbitrary to conclude that the GPL makes the coexistence of compatible networks more difficult than proprietary license does.

This study agrees that the government-sponsored GPL license might not be benefit to proprietary software, whereas this does not mean that such GPL software would not be beneficial to the consumers and the society as a whole. At least, arguments against governments' sponsoring GPL say nothing about whether the government should subsidize unviral FLOSS projects, such as projects licensed under the BSD. As Lessig has indicated, arguments against government funding for GPL projects can also be made against government funding the development of proprietary software (Lessig, 2002). According to this line of reasoning, in terms of facilitating software compatibility, perhaps the only software projects that governments should subsidize are those that produce code in the public domain or under unviral FLOSS licenses. However, such arguments may be too simplistic because the policy concerns behind government support for software development is quite complicated and lay far beyond the goal of compatibility.

Security

Security is also an important concern for government software procurement.[17] Modern governments have to maintain a great number of digital files that must be retrievable in perpetuity. Security flaws in the Microsoft operating system and its applications are still often exploited by viruses nowadays. Security concerns have prompted some governments to pause when considering the future use of proprietary software. According to a report released by the British Office of Government Commerce, properly configured FLOSS "can be at least as secure as proprietary systems, and open source software is currently subject to fewer Internet attacks" (OGC, 2004). Some commentators consider that FLOSS's transparency bolsters security because "backdoors" used by hackers can be exposed and programmers can root out bugs from the source code, provided the code is clear and visible (Gillespie, 2000). They

believe that systems built on software from a single vendor are more vulnerable to attack than systems integrating software from different sources.[18]

Proprietary software companies, such as Microsoft, however, assert that the openness of FLOSS makes it insecure and therefore vulnerable to terrorism (Economist, 2003). In fact, FLOSS is easier both for attackers to detect bugs and for defenders to fix them (Varian & Shapiro, 2003). In 2001, Microsoft launched a "shared source" initiative that allows some approved governments and business clients to gain access to most of the Windows software code, but not to modify it. This initiative is aimed, in part, at the alleviation of foreign governments' fears that there may be secret security backdoors in Windows. In 2003, Microsoft announced the Government Security Program (GSP) to provide its government clients with access to the source code of Windows 2000, Windows XP, Windows Server 2003, and Microsoft Office 2003. The GSP is said to provide national governments with information to help them evaluate the security of Microsoft products (Microsoft, 2003).

So far, software developers and governments around the globe still have different views on whether FLOSS or proprietary software can provide better security for customers: however, from Microsoft's shared source initiative and GSP, we can tell not only that proprietary software producers have strong financial incentives maintain the security of their products, but also that security would be continuously enhanced on the strength of market competition.

Usability

Since the functionality, as well as the user-friendliness, of software may be clearly related to both its market share and developers' profit margins, some scholars believe that there are strong incentives for proprietary software developers to meet the needs of all potential users of the software (Schmidt & Schnitzer, 2003). Proprietary software developers are undoubtedly willing to identify the needs of consumers through marketing research and to satisfy them in order to gain a competitive advantage (Smith, 2002). For example, Microsoft has invested heavily in applications usability in recent years, and many usability experts have praised Microsoft XP as a significant advance over previous systems (Varian & Shapiro, 2003).

However, the incentives for proprietary software developers to promote usability may not be shared by FLOSS. Typically, little analysis of consumer needs accompanies FLOSS development (Evans & Reddy, 2003). The open source movement has been fairly successful in the development of operating systems and server application systems that respond directly to the needs of sophisticated users, but they have been much less successful in developing end user applications. Casual observation suggests that FLOSS now is largely aimed at sophisticated users, which may be explained by the fact that FLOSS programmers are seeking recognition from their peers, who are sophisticated users. Developing software for unsophisticated end users may be intellectually unsatisfying and cannot gain recognition from the FLOSS community (Lerner & Tirole, 2002). Moreover, FLOSS projects seldom have the resources to under take market research or otherwise determine customer needs (Smith,

2002). Because most of them have not received training in human-computer interaction and usability engineering, FLOSS developers usually lack a detailed knowledge of the preferences and the practices of end users (Johnson, 2002).

Thus, it seems that there are fewer incentives for FLOSS programmers to cater to mass consumer markets; moreover, FLOSS has not been successful in developing user-friendly software aimed at mass consumer markets. Though some vendors have begun to provide FLOSS for desktop users, most of such FLOSS is still not as functional or as easy to use as Microsoft Office software. Accordingly, usability to end users is often a much greater concern when selecting desktop software to be used through an organization than when selecting software server to be used only by sophisticated IT professionals (Varian & Shapiro, 2003). Moreover, the user-friendliness of FLOSS may be related to the scale of the subject project. For example, SourceForge, a small FLOSS project, supported by only one developer, is arguably less user-friendly than other FLOSS supported by large projects or major companies.

Nonetheless, FLOSS also has its own advantage in serving users' specific needs. Compared to FLOSS, it is respectively harder for users to adopt the proprietary software products for local needs because the source code is protected by intellectual property rights laws (U.K. Commission on IPR, 2002). Thus, Shen argues that, from the perspective of developing countries, the functions and standard technical features of proprietary software products are mainly designed for the developed world. With the source code being secret, proprietary software leaves little space for technological participation by developing countries. Conversely, FLOSS provides local developers with opportunities to develop products for domestic markets, thus utilizing their better understanding of local needs (Shen, 2005).

Availability

The different production models between proprietary software companies and FLOSS community, to some extent, make their products and users dissimilar. Proprietary software companies, which normally own the result of the developers' efforts, define the scope and goals of the project, allocate work, as act as a single point of accountability for the programs vis-à-vis the outside world. In this respect, proprietary software development is more structured than FLOSS development. Therefore, when considering migration to FLOSS, government users have to contemplate whether there are FLOSS products that can serve their various administrative needs.

In addition, FLOSS appeals more to business users than to typical home users because medium or large businesses are likely to have technically adept staff who can fix bugs and tailor FLOSS to their particular needs (Evans & Reddy, 2003). Whereas the vast majority of users cannot program software, the present question for the FLOSS movement is how to have a broad and diverse end-user community whose members are not all technicians.

Industry observers also complain that there is no appropriate open source product available for governmental use (Turner, 2003). Sometimes, people

have the impression that FLOSS is free, so companies are not easily attracted to the idea of developing FLOSS applications. As Evans has pointed out that "[the] fact that government 'demands' does not mean it will get 'supply'" (Evans, 2002). Application providers also have to find a competitive business model for FLOSS. In addition, an FLOSS project may be unattractive to many developers until it reaches a critical mass (Bessen, 2002); developers may choose to wait for others to serve as pioneers. In this perspective, government's initial adoption of current FLOSS can help it to achieve a critical mass, which may, in the long term, foster more competition in the software market.

Yet for those who intend to procure FLOSS for governments and other public sector users, the availability of proper FLOSS products is still a problem to be solved. In the "Open Source Software Trials in Government Final Report" released in 2004, the British Office of Government Commerce (OGC) stated that one of the main obstacles to widespread implementation of open source software business applications is "the lack of Open Source products to compete with large-scale proprietary enterprise-level products" (OGS, 2004).

With the rapid development of FLOSS and the maturity of its related business models,[19] more and more FLOSS applications are available for mass consumer market and are better meeting customer needs. For example, a great number of unsophisticated users have adopted the office suite OpenOffice and web browser Mozilla. Therefore, this study holds that in the years to come, FLOSS developers will continuously improve its availability to compete with proprietary software in different operating and applications markets. As a user, governments will find more and more available FLOSS applications available for various administrative needs. Nonetheless, this study holds that it is reasonable for the government to provide a critical mass to current FLOSS products, provided they are not much inferior to their proprietary counterpart, in terms of quality and TCO. By doing so, government can certainly promote the availability of potential FLOSS products and subsequent competition in the software market.

Both proprietary software and FLOSS have their own advantage in terms of quality; therefore, business and general users' purchasing decision should be based on the different software products' features and their specific needs. According to research conducted by the Berlin-based company Infora, FLOSS is most appropriate not only for mail server and groupware tasks but also for file and print serving. Yet, Microsoft Windows remains most appropriate on the desktop (Lettice, 2002). Therefore, instead of blanket support for procurement of FLOSS, business software consumers should adopt a powerful, twofold strategy: first, rigorous comparisons of the functionalities of FLOSS with the functionalities of proprietary software; and, second, precise targeting of specific needs in discrete areas of business administration. Only after considering total costs of ownership, reliability, performance, scalability, security, identified business requirements, maintenance requirements, legal risks, ease of customization, ease of migration, and other dynamic factors, can a software consumer make optimal software procurement decisions.

However, as a software consumer, a government certainly has more concerns than a business consumer does. The difference between a government

user and a business user is that the government, in lending its support to FLOSS, should take into account society's interests, not merely its own interests as a consumer.

Political Concerns

Political factors, including the nature of the government, the nature, variety, and power of the various special interests working within the nation exercise an important influence on the way that government operates. Sometimes government policy toward whether to adopt FLOSS is based not only on technical and economic considerations, but also on political or ideological ones.[20] Some commentators speculate that countries with strong socialist histories or political movements are more likely to embrace FLOSS, whether by force of the law or force of the norms (Festa, 2001). But such an inference still lacks empirical and academic proof.

FLOSS and Democracy

Attitudes toward democracy may also affect a government's decision to support OSS; however, it is still unclear whether FLOSS benefits or impedes the development of democracy. For example, it has been reported that China, a communist country, favors FLOSS for political reasons that are diametrically opposed to the tenets of democracy: "FLOSS gives a level of control that proprietary software from the likes of Microsoft and HP do not give.... It may be that the authorities want to keep a check on who is using computers and firms like HP might take a dim view of what the Chinese Government wants to do."[21]

On the contrary, the European Working Group on Libre Software stated that one of the reasons for supporting FLOSS concerns the software's capacity to "[provide] a new forum for democratic action" (Working Group on Libre Software, 2000). According to Lessig, "to the extent that code is open code, the power of government is constrained. Government can demand, government can threaten, but when the target of regulation is plastic, it cannot rely on its target remaining as it wants"(Lessig, 1999). Benkler also contends that the way that FLOSS is produced, or the peer-production of information, enables new opportunities for citizens to pursue core political values of liberal societies, which are democracy, individual freedom, and social justice (Benkler, 2003).

So far there is no empirical research on FLOSS's impact on political systems and institutions, but one may argue that the results of governments' promotion of FLOSS and its effects on democracy may be theoretically diverse due to governments' various attitudes toward democracy and their ultimate aims of promoting FLOSS. Nevertheless, because it is the idea of freedom, rather than the idea of control, that is at the heart of the FLOSS philosophy, promotion of FLOSS may help to facilitate a decentralized environment for digital creativity, which is consistent with the underlying values of our democratic system—e.g., limited government and free speech. Some scholars also

contend that, if FLOSS was directed toward a political end, it would sully the "purity" of the technical decision-making process. Political affiliation also might deter people from participating in FLOSS development, thus creating an artificial barrier to entry into this sphere whose ideal and idealized form is a transparent meritocracy (Coleman, 2004). Therefore, this study holds that public authorities that attempt to use FLOSS as a tool of political control may encounter insurmountable conflicts between FLOSS and their governmental policy goals—conflicts that, will finally render the marriage of FLOSS and state control unfeasible.

FLOSS and Anti-U.S. Complexity

The United States government has long made many efforts through various international lobbying activities to promote its software industry, which is one of the most important copyright industries for the country (Drahos & Braithwaite, 2003). Indeed, the United States is completely aware of the global trend of governments' favoring of FLOSS and seeks to counter this trend by doing something beneficial for its robust proprietary software industry.[22] Meanwhile, Microsoft has declared FLOSS to be "un-American" (Story, 2002). In a review of government support for FLOSS, industry observers found that some governments seek to avoid dependence on software whose export is legally controlled by the United States and whose development and licensing is controlled by this country's dominant software industry.[23] It is thus believed by some that there may be certain anti-U.S. sentiments behind other governments' favoring of FLOSS. For example, the BBC News cites Brazilian xenophobia as one possible cause for that country's love affair with Linux (Ashurst, 2004); the *Wall Street Journal* assumes that there are ideological factors affecting European governments' FLOSS policy from the fact that some left-leaning government officials routinely rail about the dangers of being beholden to the U.S. software giant (Bryan-Laow, 2005).

This study argues that, it would be too naïve to explain a government's preference for FLOSS merely on the basis of an anti-U.S. prejudice because some U.S. companies, like IBM, Intel, Hewlett Packard, and Sun, regard governments' positive attitude toward FLOSS as a positive development for their own open-source efforts. Besides, more and more states and other local governments in the United States are now considering the adoption of FLOSS in their public sectors. Hence, it would be too arbitrary to assert that such governmental activities are done purely out of anti-U.S. complexities. A more practical concern for governments is their aim to increase their sovereignty over software products. Governments are worried that Microsoft, a single vendor, exercises far too much power over their government operations. No government wants to submit to so much influence from a single supplier.

FLOSS and Software Industrial Policy

For many developing countries, software may be a more promising field than other technologies because these countries believe that software brings

opportunities for leapfrogging—catching up with or overtaking capabilities in developed countries. Above all, the software industry is labor-intensive and does not require massive investment in fixed plant capacity or infrastructure. Once a software product is developed, manufacturing and transport, respectively, are cheap. Nevertheless, developing countries usually cannot afford the licensing fees of proprietary operating systems, which constitute the core software for computing technologies and which form the basis for software and networks (Shen, 2005). This is one of the legitimate reasons for developing countries to support FLOSS in order to foster their own software industry. Equally important, governments around the globe are now realizing that their technology expenditures have benefited not local players, but foreign, mostly U.S.-based, vendors. In order to ameliorate the ill-conceived reality, FLOSS provide an alternative for developing countries to built up their own software industry.

By promoting FLOSS and decreasing the use of Microsoft software, some governments expect to make computer technology more accessible to their citizens and to aid their respective countries' local economies. Some countries, such as Peru, are proposing extreme preferential legislation for FLOSS, mandating its use wherever possible. Behind this approach lies a long-term strategic objective, often expressed in terms of "national interest." They deem the FLOSS to be a policy tool with which to develop a domestic software industry and to increase local job opportunities.[24] Furthermore, some governments expect that, by promoting FLOSS, they can build a foundation for the export of future services and products. The goal has more immediacy whenever there is a strong need to conserve foreign currency reserves.[25]

The software industry is labor-intensive, but it does require skilled and trained labor. Most developing countries have abundant labor, but rarely abundant skilled labor (Arora & Gambardella, 2005). With regard to the training of skilled programmers, FLOSS provides a peculiar educational function for developing countries. With its culture of cooperation and openness, the FLOSS community brings an extremely valuable opportunity to programmers in developing countries: the opportunity to communicate with, and to learn from, their counterparts in the developed world (Shen, 2005). Governments' promoting FLOSS thus can narrow the technological gap the developing and developed world.

The industrial policy approach to governments favoring FLOSS is criticized by some economists as an unwarranted attempt to interject political considerations into what should be a technological and economic decision (Evans & Reddy, 2003). They argue that, while government support might help new FLOSS projects get off the ground and perhaps positive externalities can be demonstrated in certain cases, many proposed projects are not socially beneficial, and the government possesses no better knowledge than private parties about which proposed projects address unmet private needs (Bessen, 2002).

Nevertheless, such an argument regarding governments' hands-off attitude may be too ideal for the real world. As mentioned previously in this section, FLOSS certainly provides some unique advantages for developing countries. Evidently, it is within a government's capacity to enact the most appropriate

industrial policy to promote social welfare. If FLOSS can serve as a tool to maximize social welfare, governments are legitimately taking advantage of it. Furthermore, free market advocates should bear in mind that IP laws, which have incentivized proprietary software development in the past few decades, are also a type of industrial policy and government intervention. Therefore, the real problem here is discerning the optimal nature of government intervention, rather than figuring out how to avoid such intervention.

FLOSS as an Alternative Approach to the Piracy Problem

For those countries, FLOSS can be used to absorb some of the shock resulting from an anti-piracy clampdown (U.K. Commission on IPR, 2002). Since realizing that its giant software industry made it the biggest exporter of copyrighted material in the world, the United States has been pressuring most developing countries to enforce intellectual property laws seriously (Drahos & Braithwaite, 2003). Thus, the United States has considered the battle to protect software as a significant part of its trade war with developing countries (Lea, 2002). From the perspective of these countries, wide adoption of FLOSS might reduce software piracy and, consequently, lessen U.S. pressures on these countries to stamp out such copyright infringements.[26] Under most of the recognized FLOSS licenses, it is perfectly acceptable either to purchase a single copy of software and install it on any number of machines or simply to download it for free off the Internet.

Maybe the development of FLOSS could alleviate the software piracy problem to some extent in developing countries, but the strategy of supporting FLOSS could only effect a temporary solution to the piracy problem. In addition to software, there are still many information products that can be pirated. In the context of software, FLOSS is currently still unavailable in many application programs; therefore, even if a government were to support specific FLOSS projects or to procure FLOSS for public administration, there would still be some piracy existing in those areas of the software market where no FLOSS is available. It stands to reason, then, that support of FLOSS is, at most, a stopgap measure that only mitigates the piracy problem, whereas it is education and enforcement that can really get at the root of the problem.

Conclusion

Despite the fact that more and more public sectors have begun to migrate from proprietary software to FLOSS, governments find it difficult to legislate such an explicit preference for FLOSS. So far, most of the proposals for preference legislation either have been defeated or are pending. Another interesting trend is that many governments are now beginning to put an open standard requirement, instead of an open source preference, in their procurement guidelines in order to avoid being locked-in by proprietary software companies.

The policy considerations that inform government decisions are extremely complicated and sometimes interdependent. Despite the existence of various levels of use, FLOSS exists largely in a state of policy "limbo" between, on

the one hand, calls for wider adoption and positive action and, on the other hand, approaches to ensure fair treatment and inclusion. This study has attempted to demonstrate both the benefits and the risks of government policy favoring FLOSS from the perspective of economics, technology, and politics, and to further analyze whether these same policy goals can be achieved through government support of FLOSS.

The most fundamental argument of this paper is that, as a software consumer, a government certainly has more concerns than a business consumer does. The difference between a government user and a business user is that, in lending its support to FLOSS, the government should take society's long-term interests into account, not merely its own interests as a consumer. This study holds that the government adoption of FLOSS could prevent the market of software debugging from being monopolized by specific software vendors. Furthermore, this study contends that Microsoft's shared source initiative, GSP, and its support for full indemnification against IP claims, signal its awareness of the need for self-transformation and adaptation to the new technological environment posed by FLOSS.

Like intellectual property rights, government support for FLOSS may be viewed as an alternative approach to solving the under-production problem of public goods. Although strong network effects do not imply software market failure, other types of market failure, such as software under-consumption, incompatibility, and the insufficiency of programming education in the society, may exist in the software market. These market failures on the whole may, to some extent, justify government intervention in the software market through support of FLOSS. Moreover, some researchers claim that national policies, including FLOSS policy, can often override or balance competition concerns when other policy objectives—such as national security and industrial development—are involved. Nonetheless, governments are advised to carefully avoid government failures when intervening into market activities.

Finally, FLOSS is better than proprietary software for increasing compatibility and network effects for consumers. However, if software compatibility is policymakers' only objective, an open standard requirement in government software procurement regulations may be more effective and less controversial in achieving such policy goals. Governments usually have other policy goals than just promoting software compatibility. In some developing countries, promoting FLOSS in the public sector can also have an educational function for software programmers and, consequently, help to develop the domestic software industry. Therefore, this study concludes that, when two systems are equally suitable, governments could reasonably choose FLOSS over proprietary software based on these policy concerns. However, governments should still cautiously evaluate the social benefits and costs of supporting FLOSS before making such a decision.

Notes

1. The expression *network effects* or *network externalities* refers to an economic theory in which "the utility that a user derives from consumption of a good increases with the number of other agents consuming the good" (Katz & Shapiro, 1985).

2. Such language appears in the preceding ACT Government Procurement Guideline Amendment 2003 and in a bill of South Australia (SA) (Turner, 2003).
3. The U.S. state of Massachusetts announced its IT acquisition policy (Enterprise Open Standards Policy) on January 12, 2004, moving from what originally appeared to be a shift toward specifying FLOSS to a greater focus on open standards. The new open standards for IT acquisitions base the criteria for government IT procurements on "best value" and set guidelines to help reduce the total cost of ownership of systems (Taft, 2004).
4. For instance, the Taiwan government is planning to set up six educational centers around Taiwan to train open-source developers. (Noronha & Schlesinger, 2003). In 2000, the South Korean government also set up training programs for GNU/Linux for systems administration (Festa, 2001).
5. For example, BerliOS, a mediator for FLOSS developers and customers, is co-funded by the German federal government and private companies such as Hewlett-Packard and Linux Information system. *See* Klaus M. Schmidt & Monika Schnitzer, "Public Subsidies for Open Source? Some Economic Policy Issues of the Software Market", *Harvard Journal of Law & Technology* 16, no. 2 (2003): 499.
6. For instance, the Singapore government is reported to have offered economic incentives such as tax breaks and grants for Linux-related economic development. *See* David S. Evans & Bernard J. Reddy, "Government Preference for Promoting Open-Source Software: A Solution in Search of a Problem", *Michigan Telecommunications and Technology Law Review* 9, no. 2 (2003): 378.
7. In Thailand and the Philippines, for instance, government-funded computer research centers have created their own FLOSS applications that they are distributing to government users and small businesses (Berger, 2002).
8. For example, the Taiwanese government has planned to pour US $3.4 million into the promotion of FLOSS development in 2003 and 2004. The Industrial Development Bureau (under the Ministry of Economic Affairs), the government's IT think tank, the Information Industry (III), and the Taipei Computer Association will all work together to implement the project (Chuang, 2003).
9. For example, public universities in South Korea, squeezed by the region's 1997 financial crunch, found themselves unable to purchase software. In response, the Ministry of Information and Communication set up training programs for GNU Linux for systems administration (Festa, 2001). Besides, according to research based on interviews with 150 companies and organizations in the private and public sectors in Germany, around 38 percent of the polled companies and organizations identified savings as the main reason for choosing FLOSS (Blau, 2003). The Bundestag of Germany passed a resolution on "Germany's Economy in the Information Society" on November 9, 2001, calling on the government to introduce FLOSS in the federal administration and stating that FLOSS should be used wherever it would lead to cost savings. German experts said the use of FLOSS in public administration would save the federal government US$130 million and US$2.6 billion countrywide (Evans & Reddy, 2003). In Norway, government representatives also have looked into FLOSS as a way to cut costs. The French government, on the other hand, has cut back on expenses since it began replacing 300 of its servers running Windows NT and Unix with to open source alternatives (Berger, 2002).
10. For example, the Munich Mayor Christian Ude said that the city was seeking greater IT independence when Munich decided to migrate 14,000 computers in its public administration to Linux and other FLOSS (Dravis, 2003). The State Secretary in the Federal Ministry of Interior Affairs stated in July 2001 that dependence on a single software provider makes system more vulnerable, and that the federal government would try to reduce its dependence on single software provider by adopting FLOSS. At the regional level, a Green Party Member of the Saxony regional Parliament stated that a Linux platform complemented with FLOSS and commercial products should relieve the government's dependence on a single provider (Evans & Reddy, 2003).
11. The BSD license, perhaps the most historic FLOSS license, was developed by Eric Raymond and other FLOSS developers to represent, "a quieter, less confrontational and more market-friendly strain in the hacker culture." Different from the GPL, BSD is another licensing model of FLOSS that does not prohibit users from turning the source code into proprietary software; it only requires the users to acknowledge the original copyright of the underlying source code.
12. GPL, adopted by the GNU/Linux operating system, is the most famous license term among all the FLOSS license models. The GPL, conceived and written by the Free Software Foundation,

requires that anyone who modifies and redistributes these modified versions be licensing to others according to the GPL terms.

13. For example, in May 2003, the city of Munich in Germany decided to migrate its 14,000 computers to Linux and other open source office applications even though Microsoft dropped its prices to match Linux (Festa, 2001). Moreover, the Thai government initially subsidized a project including Linux TLE, a Thai version of Linux, along with the OpenOffice productivity suite. Microsoft responded by agreeing to deliver its operating system and office suite for about US$300, translate the program into Thai, and develop a new licensing agreement (Dravis, 2003).
14. For example, on November 9, 2001, the Bundestag of Germany passed a resolution on "Germany's Economy in the Information Society." The resolution describes FLOSS as a means by which to secure competition against dominant players in the software markets. The State Secretary in the Federal Ministry of Economics and Technology noted that open-source software played an important role in bringing competition to Germany's software market. Along the same line, the audit study presented to the Budget Committee of the regional parliament of Schleswig Holstein stated that Linux and Linux-compatible applications should bring more competition into the IT arena. The Commission's 2001 IDA study also noted that open source would help competition (Evans & Reddy, 2003).
15. In October 2002, the Danish Board of Technology released the report "Open Source Software in e-Government" in which it is stated that "[the] ordinary market conditions for standard software will tend towards a very small number of suppliers or a monopoly.... It will only be possible to achieve competition in such a situation by taking political decisions that assist new market participants in entering the market." The report further said that open source could help make public sector software procurement more cost-effective by introducing real competition (Broersma, 2003; Dravis, 2003).
16. For instance, in Finland, where the government has been testing StarOffice and OpenOffice, FLOSS developed by Sun Microsystems for office uses, early results reveal some incompatibilities for users trying to open Microsoft Office documents in the open-source alternative. A counselor in Finland's Ministry of Finance said, "We recommended open source only for people who don't exchange documents with other people" (Berger, 2002).
17. According to a research based on interviews with 150 companies and organizations in the private and public sectors in Germany, around 28 percent of the entities cited security and stability as their main reasons for choosing FLOSS (Blau, 2003).
18. For example, the German Federal Ministry of the Interior in Berlin announced a government deal with IBM in 2002 to purchase hardware and software products that support Linux. The official who signed the deal said that the switch to open source would avoid a "mono" IT environment, which is more susceptible to attack (Berger, 2002). Besides, a study at the University of Wisconsin found open source UNIX operation system were more reliable than more mature commercial products (Bessen, 2002).
19. Some FLOSS companies build their distribution and service businesses by assembling collections of FLOSS programs, bundle them, and sell them as "distributions;" payment is thus received not for the software *per se*, but rather for the selection and assembly skill needed to compile a workable distribution. For example, the company Red Hat, which has successfully developed its FLOSS business model, collects a premium for assembling customized versions of Linux and adds value to their product by testing components and using only those that are of the highest quality, thus saving users the cost of making such modifications on their own. This is a business model based on "aggregation" of freely available pieces into a valuable whole. Other FLOSS companies found their true business opportunities lie in follow-on documentation, support, service, and customisation. In many ways, theses FLOSS business models recast software as a service industry, rather than a product industry (Varian & Shapiro, 2003).
20. For example, legislators in the city government of Florence, Italy, passed a motion in June 2001 to warn the public that continued use of proprietary software was leading to "the computer science subjection of the Italian state to Microsoft" (Festa, 2001).
21. This is stated by Dan Kusnetzsky, the vice president of software systems at IDC (BBC News, 2002).
22. For example, it has been reported that the U.S. Ambassador to Peru, John Hamilton, afraid that his host nation might adopt a bill decreeing the use of FLOSS in all government systems, wrote a letter

to the president of the Peruvian Congress, expressing his dismay at the proposed legislation. In his letter, Hamilton said that while the United States does not oppose the development of open-source software, it prefers to support a free market where the quality of the product can determine the issue. Peruvian Congressman Edgar Villanueva, the bill's chief sponsor, said he considers Hamilton's letter to be "overt pressure" on Peru by the United States and Microsoft (d'Empaire, 2002).

23. Some people support FLOSS for political and technical reasons that have yet to be verified. These people believe that some versions of Windows contain backdoors designed to grant the U.S. National Security Agency access to users' data (Perera, 2001).
24. For instance, in the Australia Capital Territory (ACT), Democrat Roslyn Dundas, the member for Ginninderra, who introduced the open source provision to the ACT Government Procurement Guideline Amendment 2003 said, "Open source software was available that could have ... provided local jobs." Helen Cross, a government member, said that the legislation "will encourage open-source software producers in the ACT to develop new products suitable for use by the government, because they will know they have a reasonable chance of winning software tenders" (Gedda, 2003). In China, a vice minister of the Chinese Ministry of Information said in November 2003 that "Linux is an opportunity for us to make a breakthrough in developing software" (Dravis, 2003). The Bundestag of Germany also passed a resolution on November 9, 2001, that declared FLOSS to be a special opportunity for the European Software industry (Evans & Reddy, 2003). The Venezuelan government also announced a policy that exclusively calls for the use of FLOSS in that government. The main reason is that "the government and the people of Venezuela were increasingly concerned that over 75 percent of the funds for software licenses went to foreign nations, 20 percent to foreign support agencies, and only 5 percent to Venezuelan programmers" (Proffitt, 2002).
25. Hancom Linux in South Korea and Conective in Brazil are examples of organizations that are not only serving local users, but also breaking out into new markets. Hancom Linux is promoting its Arabic version of Linux (based on Red Hat Linux) as well as its office suite in the Middle East (Miller, 2002).
26. For example, in Peru, software piracy rates are as high as 60 percent, and members of Congress have proposed a bill that would require government agencies to use FLOSS whenever possible. One of the reasons given for the proposed legislation is to cut down on software piracy. (Berger, 2002). The Pakistani government also announced that Linux and FLOSS products are at the forefront of its initiatives to curb piracy and protect intellectual property (Dravis, 2003).

References

Arora, A. and Gambardella, A. (2005). "The Globalization of the Software Industry: Perspectives and Opportunities for Developed and Developing Countries." *Innovation Policy & Economy*, 5(1), 1-32.

Ashurst, M. (2004). "Brazil falls in love with Linux." *BBC News* <http://news.bbc.co.uk/1/hi/business/3445805.stm>.

BBC News (2002). "Linux takes on MS in China" <http://news.bbc.co.uk/2/hi/science/nature/1749441.stm>.

Benkler, Y. (2002). "Coase's Penguin, or, Linux and the Nature of the Firm." *Yale Law Journal*, 112(3) , 369-446.

Benkler, Y. (2003). "Freedom in the Commons: Towards a Political Economy of Information." *Duke Law Journal*, 52(6), 1245-1276.

Berger, M. (2002). "Microsoft vs. Open Source, Battle Gets Political" <http://archive.infoworld.com/articles/hn/xml/02/06/10/020610hnopensource.xml>.

Bessen, J. (2002). "What Good is Free Software?." In Robert H. Hahn (ed.) *Government Policy Toward Open Source Software* (pp. 12-33). Washington D.C.: AEI-Brookings Joint Center for Regulatory Studies.

Blau, J. (2003). "High Growth Rates for Open Source in Germany" <http://www.infoworld.com/article/03/07/02/HNgermanopen_1.html>.

British Office of Government Commerce (OGC) (2004), "Open Source Software Trials in Government Final Report."<http://www.ogc.gov.uk/oss/Report-v8c.htm >.

Broersma, M. (2003). "Denmark Urges Govt Support for Open Source," *ZDNet UK* <http://news.zdnet.co.uk/software/linuxunix/0,39020390,39117341,00.htm>.

Bryan-Laow, C. (2005). "Governments Prod Microsoft to Slash Prices." *Wall Street Journal*, Jan. 27, 2005, at B1.

Clark, D. (2003). "Counterculture's Gift to Government." *National Journal*, 35(1), 40-41.

Chuang, S. (2003). "Taiwan to Plug Open-Source Software" <http://www.asiacomputerweekly.com/acw_ViewArt.cfm?Magid=1&Artid=21899&Catid=8&subcat=79>.

Coleman, G. (2004). "The Political Agnosticism of Free and Open Source Software and the Inadvertent Politics of Contrast." *Anthropological Quarterly*, 77(3), 507-519.

d'Empaire, A. (2002). "Microsoft's Big Stick in Peru," *Wired News* <http://www.wired.com/news/business/0,1367,54141-2,00.html>.

Drahos, P. and Braithwaite, J. (2003). *Information Feudalism*. New York, NY: New Press.

Dravis, P. (2003). *Open Source Software: Perspectives For Development*. Washington D.C.: The World Bank.

Economist (2003). "Microsoft at the Power Point" <http://www.economist.com/displaystory.cfm?story_id=2054746>.

Evans, D. S. (2002). "Politics and Programming." in Robert H. Hahn (ed.) *Government Policy Toward Open Source Software* (pp. 34-49). Washington D.C.: AEI-Brookings Joint Center for Regulatory Studies.

Evans, D. S. and Reddy, B. J. (2003). "Government Preference for Promoting Open-Source Software: A Solution in Search of a Problem." *Michigan Telecommunications and Technology Law Review*, 9(2), 313-394.

Kesan, J. P. and Shah, R. C. (2003). "Deconstructing Code." *Yale Journal of Law & Technology*, 6, 277-389.

Katz, M. L. and Shapiro, C. (1985). "Network Externalities, Competition, and Compatibility." *American Economic Review*, 75(3), 424-440.

Festa, P. (2001). "Government Push Open-Source Software", <http://news.com.com/2100-1001-272299.html?legacy=cnet>.

Gedda, R. (2003). "ACT Passes Open Source Law" <http://www.arnnet.com.au/index.php?id=792934018&fp=2&fpid=1>.

Gillespie, R. (2000). "German Federal Government to Support Open Source Software" <http://www.internetnews.com/bus-news/article.php/6_408271>.

Ingram, M. (2004). "European antitrust case finds against Microsoft" <http://www.wsws.org/articles/2004/mar2004/micr-m31.shtml>.

Landes, W. M. and Posner, R. (2003). *The Economic Structure Of Intellectual Property Law*. Cambridge, MA: Harvard University Press.

Lea, G. (2002). "Digital Millennium or Digital Dominion? The Effects of IPRs in Software on Developing Countries." In Peter Drahos and Ruth Mayne (eds.) *Global Intellectual Property Right* (pp. 144-59). New York, NY: Palgrave Macmillan.

Lemley, M. A. and McGowan, D. (1998). "Legal Implications of Network Effects." *California Law Review*, 86(May), 479-611.

Lerner, J. & Tirole, J. (2002). "Some Simple Economics of Open Source." *Journal of Industrial Economics*, 50(2), 197-234.

Lessig, L. (1999). *Code And Other Laws Of Cyberspace*. New York, NY: Basic Books.

____. (2001). *The Future of Ideas*. New York, NY: Random House.

____. (2002). "Open Source Baselines: Compared to What?" in Robert H. Hahn (ed.) *Government Policy Toward Open Source Software* (pp. 50-68). Washington D.C.: AEI-Brookings Joint Center for Regulatory Studies.

Lettice, J. (2002). "Report Favours Open Source, Windows Mix for Bundestag" <http://www.theregister.co.uk/content/4/24048.html>.

Merges, R. P. et al. (2003), *Intellectual Property in The New Technological Age*. New York, NY: Aspen Publishers.

Microsoft Corp. (2003). "Government Security Program." <http://www.microsoft.com/resources/sharedsource/Licensing/GSP.mspx>.

Miller, R. R. (2002). "Asia Will Be the Center of Linux Development in 2003", *News Forge* <http://www.newsforge.com/software/02/12/27/0259244.shtml?tid=11>.

McGowan, D. (2001). "Legal Implications of Open-Source Software." *University of Illinois Law Review*, 2001(1), 241-304.

Moglen, E. (2002). "Free Software: Free Government, II" <http://emoglen.law.columbia.edu/publications/lu-24.pdf >.

Noronha, F. & Schlesinger, L. (2003). "Open Asia: Open Source in Tajikistan, Taiwan, and Thailand" <http://www.newsforge.com/os/03/10/10/1814248.shtml?tid=132&tid=2>.

Perera, R. (2001) "German Parliament Considers Linux Switch" <http://www.itworld.com/Comp/2384/IDG011016germanlinux/>.

Proffitt, B. (2002). "Venezuela's Government Shifts to Open Source Software", *Linux Today*, <http://linuxtoday.com/news_story.php3?ltsn=2002-08-30-011-26-NW-LL-PB>.

Raymond, E. S. (1998). "Homesteading the Noosphere" <http://www.firstmonday.dk/issues/issue3_10/raymond/ >.

Schmidt, K. M. & Schnitzer, M. (2003). "Public Subsidies for Open Source? Some Economic Policy Issues of the Software Market." *Harvard Journal of Law & Technology*, 16(2), 473-505.

Shavell, S. M. (2004). *Foundations of Economic Analysis of Law*. Cambridge, MA: Harvard University Press.

Smith, B. L. (2002). "The Future of Software: Enabling the Marketplace to Decide." In Robert H. Hahn (ed.) *Government Policy Toward Open Source Software* (pp. 69-86). Washington D.C.: AEI-Brookings Joint Center for Regulatory Studies.

Shen, X. (2005). "Developing Country Perspective on Software: Intellectual Property and Open Source—A Case Study of Microsoft and Linux in China." *International Journal of IT Standard & Standardization Research*, 3(1), 21-43.

Story, A. (2002). "Don't Ignore Copyright, the 'Sleeping Giant' on the TRIPs and International Education Agenda." In Peter Drahos and Ruth Mayne (eds.) *Global Intellectual Property Right* (pp. 125-43). New York, NY: Palgrave Macmillan.

Taft, D. (2004). "Mass. Softens Stance on Proprietary Software" <http://www.eweek.com/article2/0,4149,1436253,00.asp>.

Turner, A. (2003). "SA Democrats to Tone Down Open Source Bill" <http://www.theage.com.au/articles/2003/08/20/1061261205187.html>.

U.K. Commission On Intellectual Property Rights (2002). *Integrating Intellectual Property Rights And Development Policy*. London, UK: U.K. Commission On Intellectual Property Rights.

Varian H. R. and Shapiro, C. (2003). "Linux Adoption in the Public Sector: An Economic Analysis." <http://www.sims.berkeley.edu/~hal/Papers/2004/linux-adoption-in-the-public-sector.pdf>

Working Group on Libre Software (2000). "Free Software/Open Source: Information Society Opportunities for Europe? Version 1.2"<http://eu.conecta.it/paper.pdf>.

Zittrain, J. (2004). "Normative Principles for the Evaluation of Free and Proprietary Software." *University of Chicago Law Review* 71(1), 265-87.