e-Primers on Free/Open Source Software

Free and Open Source Software (FOSS) and Government

A Policy Primer

Kenneth Wong

International Open Source Network (IOSN)
An Initiative of UNDP's Asia-Pacific Development Information Programme

http://www.iosn.net

Published by
the United Nations Development Programme's
Asia-Pacific Development Information Programme (UNDP-APDIP)
Kuala Lumpur, Malaysia

Web: http://www.apdip.net/
Email: info@apdip.net

ISBN:

Table of Contents

Preface

Free and Open Source Software (FOSS) has grown incredibly in the past few years. Once considered a hobbyists toy, FOSS has grown by leaps and bounds and is now used widely throughout the world, even in critical environments such as financial systems and network backbones.

Governments too have begun to take notice of this phenomenon for various reasons of their own. While the often cited costs and stability benefits of FOSS are attractive, governments often choose to promote FOSS uptake in their own countries for a variety of other reasons. Countries such as Brazil, South Africa, Vietnam, Malaysia and China are implementing nationwide policies or legislation promoting FOSS.

In combination with the other primers in the Free/Open Source Software series, this primer is meant to serve as a resource for nations in the process of formulating their FOSS policies. This particular primer surveys the motivations of other countries in implementing FOSS, summarizes the steps involved in formulating a policy, lists some possible strategies to use in implementing the policy and finally touches on cross-sectoral issues unique to FOSS policies. Its target audience are the policy-makers who set national policies and their advisers.

These primers are brought to you by the International Open Source Network (IOSN), an initiative of the United Nations Development Programme (UNDP)'s Asia-Pacific Development Information Programme (APDIP). We would like to thank all those who have been involved in creating this primer, including the researchers, peer reviewers and production team. In particular, we would like to thank APDIP and the International Development Research Centre (IDRC) of Canada for their generous financial support, without which this primer would never have been written.

http://www.iosn.net/downloads/foss_gov_primer.pdf

[i] This primer is licensed under the Creative Commons Attribution 1.0 license . License is granted to copy, distribute, and make commercial use of this work as long as you give the original authors credit for this work. Full details of the license can be found at: http://creativecommons.org/licenses/by/1.0/

Chapter 1: Introduction

> *"Briefly, OSS/FS programs are programs whose licenses give users the freedom to run the program for any purpose, to study and modify the program, and to redistribute copies of either the original or modified program (without having to pay royalties to previous developers).[1]"*

The above quotation summarizes the underlying principles of FOSS – the freedom to use, understand, modify and distribute the underlying software that controls a computer. While seemingly a simple matter, these principles have the potential to produce a profound impact on the economics and dynamics of the software industry today.

The software industry as we know it today considers the source code of their software to be a trade secret and their primary driver of profit. Their customers do not purchase software but rather license the right to use the software in ways tightly controlled by the software producer. This software cannot be modified by the user should the need arise. Modifications and improvements can only be done by the software producer and often come at an additional cost.

In this industry, software is rarely shared and competitors often reinvent the wheel, re-implementing the same functionality that their competitors have already implemented. Often this reimplementation is done in incompatible ways to ensure that their users experience a high switching cost. In mature software sectors, the cost of re-implementing existing functionality can make the barriers to entry high, greatly reducing competition. e.g. the cost to re-implement an equivalent of Sun's Solaris operating system would prohibitively high.

FOSS, due to its fundamental freedoms, is a dramatic departure from the current proprietary software model. Since source code is easily shared, FOSS becomes a global public resource that anyone can make use of, learn from and improve. Regardless of where it was first produced, the global community at large can take advantage of this knowledge resource and benefit from it.

It is also important to note that FOSS is more than just software or a software development method – it is also a community of diverse individuals who have contributed their time, energies and knowledge to the creation of a global resource. Often vocal, strongly opinionated and technically savvy, the FOSS community can be a powerful contributor towards harnessing the benefits of ICT and FOSS for development.

Ownership of FOSS

It is a common misconception that FOSS applications are not owned by anyone or that it is anti-intellectual property. In reality, all FOSS applications are acknowledged property of and copyrighted by their respective authors. However, these authors have granted liberal licenses that allow anyone to use their intellectual property with limited restrictions.

The copyright holders still have full rights to their work, including the right to sell their work or relicense it under another license. There are several companies successfully utilizing this business model.

A good introductory article on the legal basis of FOSS can be found at[ii]:

http://www.groklaw.net/article.php?story=2004040421042728

It is not possible to cover adequately every detail of FOSS in this primer. For further information, please refer to the first primer in this series, Free/Open Source Software: A General Introduction. It can be downloaded online at:

http://www.iosn.net/downloads/foss_primer_current.pdf

[ii] This is a reprint with permission of the original authors. The original article is available (with much lengthier url at: http://www.freehills.com/CA256AD900137BAA/page/Listing-aip-Open+source+software%3a+What+is+it+and+how+does+it+work%3f00201A3E?opendocument&1=50-Publications~&2=0-Intellectual+Property~&3=~&REFUNID=F4C0F09FA380BBAE4A256B48001B31E6~undefined

Chapter 2: The Strategic Importance of FOSS

There are a large number of countries looking at specifically promoting FOSS via legislative, policy or government procurement methods[iii]. If FOSS were just another method of developing software, governments would have little reason to specifically advocate FOSS. However, FOSS brings many compelling benefits to a nation, especially a developing nation with limited resources.

Various governments have listed different motives behind their FOSS initiatives. Since each country's circumstances are different, there is a wide variance between their motives. Nor are these reasons necessarily similar to the often cited reasons for private sector adoption of FOSS. FOSS brings additional benefits that are not relevant to the private sector but critical to developing nations.

The most common motives given are:

National Values
- National Security
- Developing local capacity/industry
- Conservation of foreign exchange/reduction of imports
- Piracy, intellectual property rights
- Localization

Business Values
- Total Cost of Ownership (TCO)
- Security
- Vendor Independence

Social Values
- Access to information

These motives are further explored below:

National Values

National Security

Because proprietary software is normally distributed in binary format, it is difficult to reverse-engineer and understand exactly what a program does. While the opaqueness of the binary offers limited protection to the intellectual property of the software maker, it also engenders mistrust and suspicion. Could there be hidden back doors

[iii] A listing of some of these countries and their efforts can be found in the first primer, Free/Open Source Software: A General Introduction.

or holes in the software, possibly allowing a remote attacker to easily compromise data? In the case of governments other than the United States of America, there is some mistrust of Microsoft's software, especially after the infamous "NSA Key" incident[iv].

Other recent cases include the acknowledgments by both Netgear[2] and Cisco systems that there were secret users and passwords hard-coded into certain models of their wireless routers[3]. The user could not be disabled and allowed anyone knowing the right username/password combination to connect and take full control of the router. Both Cisco and Netgear later issued a fix but there is no way to verify that other back doors do not exist.

Mistrust of foreign software that they cannot verify has been cited as one of China's reasons for its adoption of FOSS[4] and one of the reasons other governments are considering FOSS[5].

Developing local capacity/industry

A common problem developing nations face is the lack of capacity - where does a country obtain the human capacity required to sustain an ICT infrastructure? In this area, FOSS excels. For most developing countries, it is not that FOSS will make a non-existent industry more competitive but rather it allows a developing nation to kick-start its ICT industry and advance to a stage where it can begin to fully utilize the benefits of ICT internally.

It has been noted that there is a positive correlation between the growth of a FOSS developer base and the innovative software capacities of an economy. A report from the International Institute of Infonomics lists three reasons for this[6]:

- **Low barriers to entry**: FOSS, which encourages the free modification and redistribution, is easy to obtain, use and learn from. Proprietary software tends to be much more restrictive, not just in the limited availability of source code, but also because of licensing, patent and copyright limitations. FOSS allows developers to build on existing knowledge and pre-built components, much like basic research. Existing proprietary software is often the result of years of building and refinement. A software company in a developing country has little chance of developing a competitive system and selling it.

[iv] In 1999, a security researcher found a key in Microsoft Windows code named _NSAKEY. The NSA was assumed to refer to the American National Security Agency and a wide amount of publicity was generated. Microsoft denied the allegations that the key gave the NSA access to data on Windows systems but suspicion still lingers. There are many references to this incident, including on CNN.com, at: http://www.cnn.com/TECH/computing/9909/03/windows.nsa.02/

- **FOSS as an excellent training system**: The open and collaborative nature of FOSS allows a student or software engineer to examine and experiment with software concepts at virtually no direct cost to society. Besides the source code and software tools FOSS provides, there is a wealth of technical manuals, guides and how-tos provided in every FOSS distribution. This documentation is the equivalent of thousands of dollars of manual and textbooks, all available and freely redistributable. A student can also tap into the global collaborative FOSS development network that includes massive archives of technical information and interactive discussion tools. Proprietary systems are usually closed and do not encourage this experimentation and learning.

- **FOSS as a source of standards**: FOSS often becomes a *de facto* standard by virtue of its dominance in a particular sector of an industry. By being involved in setting the standards in a particular FOSS application, a region can ensure that the standard produced takes into account regional needs and cultural considerations For instance, proper word wrapping is still an issue with many non-roman alphabetical languages and holds back the development of word processors, browsers and other software tools for these languages. However, the Chinese language, although based on a radically different system from western languages, is decently supported in FOSS systems due to the involvement of Chinese speaking FOSS developers.

Additionally, current business models based around FOSS are primarily services based, rather than product based. This makes it much more likely that a FOSS based company will have the majority of its staff in the country that it sells to and thus reinvest its profits there.

Reduction of imports/conservation of foreign exchange

A significant portion of the global proprietary software industry today is centralized in only a small number of countries. Companies based in the United States produce an enormous proportion of the world's operating systems (IBM AIX, HP-UX, Solaris, Microsoft Windows, etc) and business applications (Oracle, PeopleSoft, Photoshop, etc). Countries that have to license this software, other than the United States, end up importing software licenses. The large cost of these software licenses in developing nations place an enormous burden on the financial resources of a nation, resources that can be used on other development needs. Fortunately, there are alternatives.

FOSS, by the nature of its licensing terms, can be obtained at little or no cost. This by itself saves a massive amount of foreign exchange

but is not the only benefit. As noted in the European study, "Free/Libre and Open Source Software: Survey and Study"[7]:

> *"The costs of this more service-oriented model of open source are then also normally spent within the economy of the governmental organization, and not necessary (sic) to large multinational companies. This has a positive feedback regarding employment, local investment base, tax revenue, etc."*

Whatever money is spent on FOSS in a country usually stays in that country, which leads to the previously mentioned benefit: the development of local industry.

This is reportedly one of the major motivations behind Brazil's FOSS policy. In its 2002 balance of payments report, Brazil actually spent more money on "royalties and licenses" than it did on "computer and information"[8]. While software licenses only represent a portion of royalties and licenses, any reduction in licensing fees would improve Brazil's balance of payment situation. It has been argued that the current balance of payment situation has negatively impacted Brazil's economic development[9].

Other countries or economies reported to have similar motives include South Africa, Taiwan (province of China) and South Korea.

Piracy, Intellectual Property Rights

Software piracy is a problem in almost every country around the world. The Business Software Alliance estimates that software piracy in 2002 alone cost US$13.08 billion. Even in developed nations where software is affordable in theory, piracy rates were as high as 24 percent in the United States and 35 percent in Europe. Piracy rates in developing countries, where lower incomes make software far more expensive, are upwards of 90 percent[10].

Software piracy and lax laws against it can and does hurt a country in many ways. A country with poor protection for Intellectual Property Rights (IPR) is not as attractive to foreign investors. Membership in the World Trade Organization (WTO) and access to its benefits are strongly influenced by the level of protection given to IPR in a country. Bilateral and multilateral trading agreements with developed nations often involve requirements to reduce software piracy as well. Finally, a culture of software piracy hurts local software development, as there is less incentive for local software developers to create a local product.

FOSS assists in combating software piracy by serving as a low-cost alternative to proprietary software. Proprietary software is far too expensive for most people in developing countries[11] and software piracy is the only way they can access the benefits of ICT if they are not aware of FOSS. Additionally, FOSS can assist organizations that are facing enforcement action by replacing existing pirated proprietary software.

Software piracy issues have been cited as a major motivator for Vietnam's FOSS policy. A trade agreement with the United States signed in 2001 and Vietnam's goal of joining the World Trade Organization by 2005 make it imperative that the piracy rate be greatly reduced.

Localization

Countries where English is not commonly spoken can be at a serious disadvantage when it comes to the uptake and dissemination of Information and Computer Technology (ICT). If the country and language are not deemed to be economically important, proprietary software makers may not choose to produce a localized version of their software, thereby increasing the barriers to ICT usage.

Localization is one of the areas where FOSS shines because of its open nature. Users are able to modify FOSS to suit the unique requirements of a particular cultural region, regardless of economic size. All that is necessary is a number of individuals possessing the technical capability to create a minimally localized version of any FOSS. While the construction of a completely localized software platform is no small feat, it is at least possible. Microsoft's decision in 1998 against producing an Icelandic version of Windows 98[12] would have made computing in Iceland's national language almost impossible if it were not for the emergence of FOSS alternatives.

Few countries cite localization as a motivating factor but localization efforts exist in most non-English speaking countries throughout the Asia-Pacific region.

Business Values

Total Cost of Ownership

FOSS applications save money in several ways. The most obvious is through the lack of licensing fees, since FOSS can be freely redistributed without licensing fees. However, FOSS also lowers costs through means which can be much harder to quantify, such as better security, ease of administration, cross-platform availability, etc.

While the Total Cost of Ownership (TCO) debate still rages between FOSS and proprietary software (mostly Microsoft), many organizations have reported significant cost savings from their own implementations of FOSS. Intel reportedly saved US$200 million from a move to GNU/Linux from Unix, and Amazon reported a savings of US$17 million[13] from switching their servers to GNU/Linux. Major financial institutions such as Credit Suisse First Boston, Morgan Stanley, Goldman Sachs and Charles Schwab are moving a significant portion of their infrastructure to FOSS systems to reap these cost savings[14].

There are only a limited number of TCO studies showing the total cost of running FOSS systems versus proprietary systems. These studies analyze multiple cost factors other than software licensing costs, including maintenance, personnel and opportunity costs from service disruptions. Several have been very positive towards FOSS:

- A TCO study performed by the Robert Frances Group showed that GNU/Linux costs roughly 40 percent of Microsoft Windows and as low as 14 percent of Sun Microsystem's Solaris[15].
- NetProject reported that the TCO of GNU/Linux was 35 percent of Microsoft Window's TCO[16]. Even more interesting was that the savings was due not just to licensing costs but also to various other costs, including reduction in the number of support staff and software updates that results from using GNU/Linux.
- Gartner reported that using GNU/Linux in a "locked" configuration resulted in a roughly 15 percent lower TCO compared to Windows XP[17].

Merrill Lynch, a major financial management company, recently reported that using GNU/Linux could reduce costs dramatically. The unusual part of their TCO study was that the largest costs savings was not from software licensing costs but from personnel and hardware costs[18].

Government institutions that have reported significant cost savings to date have been few and mostly in relatively developed countries. The city of Largo in the United States has reported savings of over US$1 million a year and runs its IT infrastructure on a budget that is only 40% the size of comparable cities[19]. The government of Sweden has identified savings of US$1 billion a year while the government of Denmark has identified savings of between US$480 million to $730 million[20].

Reduced Costs: Sweden

A feasibility study conducted by the Swedish Agency for Public Management concludes:

"Open standards and formats along with free and open source software are important factors in order to be able to arrive at:

– increased competetivity (sic)
– improved interoperability
– reduced costs

for administration in the public sector.

Free and open source software is not any makeshift phenomenon, but instead a fully adequate and dependable competitor to existing proprietary products and solutions."

Full report available online at:
http://www.statskontoret.se/pdf/200308eng.pdf

Security

There is no such thing as a perfectly secure operating system or platform. However, factors such as development method, program architecture and target market can greatly affect the security of a system and consequently make it easier or more difficult to breach. There are some indications that FOSS systems are superior to proprietary systems in this respect:

1. A study found that FOSS applications typically fixed problems faster than their proprietary counterparts. The contrast is much more apparent when the application is one that is typically heavily used by the FOSS developer[21].
2. A study of open source code against proprietary equivalents found that open source software had lower defect densities compared against proprietary software. The open source database MySQL had one sixth of the defects found in proprietary equivalents. The study was performed by Reasoning, a company that produces automated software inspection and auditing tools[22].
3. "Hacker Insurance" issued by J.S. Wurzler Underwriting Managers costs five to 15 percent more if Windows is used instead of GNU/Linux or Unix systems. Walter Kopf, senior vice president of underwriting at J.S. Wurzler Underwriting Managers, says, "We have found out that the possibility for loss is greater using the NT system."[23]

FOSS distributions typically also include a wide variety of security tools that allow a competent system administrator to scan their own network for vulnerabilities, detect outside attacks and protect internal systems. In some cases, FOSS tools prove to be far superior than any proprietary counterparts[24].

The security aspects have already encouraged many public organizations to switch or to consider switching to FOSS solutions. The French Customs and Indirect Taxation authority migrated to Red Hat Linux largely because of security concerns[25]. South Korea, Japan and China also cite security as a major driver behind their FOSS initiatives.

Security: United States of America

A report prepared by the MITRE Corporation for the Defense Information Systems Agency found that in area of security[26]:

"Banning FOSS in this area would have immediate, broad, and in some cases strongly negative impacts on the ability of the DOD (Department of Defense) to analyze and protect its own networks against hostile intrusion ... It would also remove the uniquely FOSS ability to change infrastructure source code rapidly in response to new modes of cyberattack."

Vendor Independence

Over time, many organizations are finding that they are tied to their existing software vendors. Due to intentionally incompatible data formats, large investments in legacy systems and patent restrictions, switching to a different supplier can often be a costly and lengthy undertaking. These organizations are then almost held hostage by their software vendor, forced into purchasing bundled systems that they do not need or upgrading when there is no necessity.

The authors of the paper "Free/Libre and Open Source Software: Survey and Study"[27] produced by the International Institute of Infonomics in the Netherlands argue against use of proprietary software in government. They say:

> *…Consequently one major argument against the implementation of proprietary software in the public sector is the subsequent dependency on proprietary software vendors. Whenever the proprietary standards are established the necessity to follow them is given. Even in an open tender acquisition system, this requirement for compatibility with proprietary standards makes the system*

biased towards specific software vendors, perpetuating a dependency.

Vendors that use FOSS systems have a much harder time locking in their clients. Most FOSS systems use open, documented standards. Even when this is not the case, the availability of the code makes it far easier to reverse-engineer any data format or standard. Purchasers of FOSS systems are not bound to stay with their supplier. They can easily engage another company to develop the systems or build an internal team of technical staff.

Using FOSS systems as a means of gaining vendor independence has been raised in several areas. A report to the UK Government concludes that "the existence of an OSS reference implementation of a data standard has often accelerated the adoption of such standards, and recommends that the Government consider selective sponsorship of OSS reference implementations." The UK health system is all too aware of the dangers of vendor dependency when the insolvency of a major supplier forced it to migrate systems to a FOSS platform[28].

Vendor Independence: United Kingdom

The United Kingdom's policy on the usage of Open Source Software (OSS) in government include the following key points:

1. UK Government will consider OSS solutions alongside proprietary ones in IT procurements. Contracts will be awarded on a value for money basis.
2. UK Government will only use products that for interoperability that support open standards and specifications in all future IT developments.
3. UK Government will seek to avoid lock-in to proprietary IT products and services.
4. UK Government will consider obtaining full rights to bespoke software code or customizations of COTS (Commercial off the Shelf) software it procures wherever this achieves best value for money.
5. UK Government will explore further the possibilities of using OSS as the default exploitation route for Government funded R&D software.

http://www.ogc.gov.uk/embedded_object.asp?docid=2498

Other countries or economies that have cited vendor independence as a driver for their FOSS initiatives include Taiwan Province of China, the German city of Munich, the Australian state of New South Wales, Brazil and South Africa.

Social Values

Access to information

It should be noted that civil society proponents of FOSS often also cite the social value of FOSS. Although often not highlighted by public and private sector proponents of FOSS, the social value of FOSS can be a passionate issue among the many promoters of FOSS.

In sum, software, especially FOSS, represents knowledge – the rules, procedures and methods of manipulating data. In today's knowledge society, knowledge drives our productivity, our economies and the shape of our future. Many argue that such knowledge should be shared as widely and freely as possible. Steve Weber argues[29]:

> *A regime built around the free diffusion of tools has an interesting characteristic: the degree to which a software tool can be utilized and expanded becomes limited only by the knowledge, learning, and innovative energy of the potential users; not by exclusionary property rights, prices, or the power of countries and corporations.*

In the process of considering the most commonly reported benefits of FOSS, it is best to keep in mind that its subtle social values (derived in part from the unfettered sharing of knowledge) are often ignored.

Chapter 3: Why is a FOSS policy needed?

If FOSS systems contain so much promise, then why is it sometimes necessary for governments to intervene and level the field or even actively promote FOSS systems? The answer to this lies in the very nature of FOSS and the computing environment that has evolved today:

Limited marketing

Because FOSS systems are available to almost anyone to use, it is hard for any single company to claim ownership and thus gain competitive advantage from promoting their product. A company that spends millions of dollars promoting GNU/Linux would benefit all the companies that offered GNU/Linux products, including fellow competitors. This is one of the reasons why there are only limited commercial efforts to promote the benefits FOSS.

Being a relatively new movement, this means that many decision makers are unaware of the full benefits that FOSS brings or the issues related to implementing it. The lack of publicity also makes it somewhat vulnerable to the Fear, Uncertainty and Doubt (FUD)[v] that may be produced by competing proprietary software makers.

Non-commercial benefits

Of the FOSS benefits described in the previous section, only three of nine (Total Cost of Ownership, Vendor Independence and Security) were relevant to most commercial entities. Since most decisions on software purchases and implementation are based on "best value" basis, the free market choices may result in a non-optimum choice for a nation.

Policy Justification: Denmark
A report produced by the Danish Board of Technology notes: "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." Full report available at: http://www.tekno.dk/pdf/projekter/p03_opensource_paper_english.pdf

[v] Stands for Fear, Uncertainty and Doubt. A common acronym used to describe the marketing tactics often used by companies to dissuade potential purchasers of competing products. Tactics such as issuing misleading reports, exaggerating problems or making extravagant promises for products that are not delivered in a timely fashion are commonly used.

Entrenched legacy systems

Even in most developing countries, there is an existing ICT infrastructure, no matter how minimal. These investments in hardware and skills development greatly impact future procurements, since decision makers are likely to stay with the systems that they are familiar with and their staff already have the capability to support.

Additionally, a significant amount of proprietary software integrates poorly with other software, be they proprietary or FOSS. This makes the cost of changing or even introducing new infrastructure extremely expensive. In many cases, the poor interoperability is deliberately introduced by the software manufacturer to dissuade customers from switching to a competitor. Proprietary and secretive document standards and communication protocols are common tools in disrupting interoperability.

A FOSS policy formulated by a government, even if it only placed FOSS on an equal standing with proprietary software, could make a large difference on the uptake of FOSS in the local economy and consequently allow the economy to take advantage of the benefits of FOSS where appropriate.

Chapter 4: Policy Formulation Approach

While formulating a FOSS policy is not fundamentally different from creating other national policies, it is worth examining the different stages in the formulation process. Due to its unique nature, a FOSS policy potentially involves a larger segment of society than some of the more focused policies.

What is the policy trying to achieve?

In the first stage of the policy formulation process, the motivations and goals of the policy need to be established in a clear, measurable manner and linked with the larger goals of the nation. Merely saying “FOSS is good for the country and we desire an increase in FOSS usage in the country” is unlikely to produce concrete results.

Clearly articulated motivations for implementing the policy are crucial, especially if they are backed at the highest level of government. Specific motivations will greatly affect implementation strategies. For instance, if the motivation is to reduce imports, then strategies are more likely to emphasize converting existing infrastructure to FOSS. If local capacity building is the motivation, then education is more likely to be emphasized.

Measurable targets should then be set, even if these targets may be adjusted later during the policy formulation process. Targets at this stage set the tone and pace of the policy to be formulated. A typical target that governments have set would be “converting 10% of their civil service infrastructure to FOSS”. Other targets such as “50% of all ICT graduates are proficient in FOSS” or “95% of private sector organizations are aware of the benefits of FOSS” are also possible.

Policy Goals: Vietnam

The Objectives listed in Vietnam's Open Source Software Master Plan (2004-2008) are:

1. Accelerating the application and development of open source software (OSS), enhancing copyrights protection and cutting costs of software purchase, promoting the development of Vietnam's information technology in general and software industry in particular.
2. Forming a base of competent technical experts who master advanced technology and leverage their creativity in OSS application and development.
3. Creating some typical IT products that respond particularly to domestic conditions and practical needs of OSS development.

An English translation of the Vietnamese OSS Master Plan, including details of how the objectives are to be achieved, can be found at:

http://www.asia-oss.org/march2004/hanoi_presentation/mp_oss_v.html

Assessment

Once the basic motivations and goals have been established, an assessment of the existing environment has to be conducted. This involves taking a look at a variety of factors, including but not limited to:

- Existing ICT infrastructure
- Existing human ICT capacity, both in the public sector and in the society as a whole
- Education infrastructure and capabilities within the country
- Existing and proposed national policies
- Regional context

Two of these factors deserve greater attention:

Existing and proposed national policies

FOSS policies may interact greatly with or be affected by other policies, depending on the actual strategies chosen to implement the policy. Some policies may even prevent the implementation of a FOSS policy. A detailed survey of national policies, laws and standards must be undertaken to ensure that a FOSS policy can peacefully coexist with broader socio-economic national objectives, making adjustments as necessary.

In particular, a FOSS policy should not be separate from the national ICT policy if it exists. The overlap in the possible implementation strategies is so great that having two separate policies is likely to be

meaningless.

Other policies that can also greatly affect FOSS policy include education, intellectual property, legal, and international trade policies. The specifics of the more common cross-sectoral concerns are covered in a later section.

Consider the regional context

There are few regions in the world today that do not have active FOSS initiatives and policies in progress somewhere in the region. An assessment of regional efforts is important to ensure that any FOSS policies will work in the regional context and also to identify areas of cooperation or common interests.

For example, countries in the Asia-Pacific region should be aware of a regional effort by Japan, China and South Korea to produce a common, regionally localized and customized FOSS system[30] and strongly utilize this in their own countries. Other economies such as Thailand, Vietnam or Taiwan Province of China have strong FOSS initiatives or formal policies encouraging FOSS. In such a situation, a country that does not have FOSS capabilities would be at a serious disadvantage.

Other issues include regional trade requirements, or even interoperability standards. The European Union is mandating open standards in their inter-government communication and suggesting FOSS as a method for implementing this[31]. On the other hand, existing inter-government systems in other regions may still require proprietary software and an overly strong FOSS policy would be impractical under such circumstances.

Strategy Formulation

Once a proper survey of the existing conditions is complete, strategies to achieve the goals of the policy need to be formulated. FOSS policy level strategies unfortunately do not have a long track record of development, testing and implementation so policy-makers will have to be especially careful with implementation and monitoring issues.

The regional survey performed previously may find many strategies being implemented regionally that can either be adopted as is or modified to suit the local environment. However, these policies are relatively untested and policy makers should not restrict themselves to these options. A popular method of producing strategies involves brainstorming sessions.

Brainstorming

Brainstorming
Brainstorming at its essence is the production of as many and as varied ideas as possible in a non-critical environment. No criticisms or evaluation of ideas are done during the brainstorming process and participants (may be done individually or in a group) are even encouraged to produce ideas that seem a bit odd. This breaks participants out of their standard methods of thinking and allows the formulation of creative and out-of-the-box solutions. Ideas are only evaluated at the end of the brainstorming session and the best ideas chosen. These ideas are then refined further, possibly via group discussions or even further brainstorming, until an optimal solution is found.

Brainstorming can be done by different parties. Different countries have relied on ICT Ministry personnel, industry groups and even cross-sectoral working groups expressly for this purpose. If an active FOSS community exists within the country, their opinions should be tapped as well as they likely have been actively working on promoting FOSS within the country for some time and are aware of issues that policy makers may not be aware of.

After several rounds of brainstorming sessions, the strategies are evaluated, summarized then compiled into a draft policy document ready for the next stage – the consultative process.

Stakeholder and consultative process

FOSS development is typically an inclusive and democratic process. Projects or organizations that operate in a secretive or exclusive manner usually fail in the FOSS environment. This makes it even more important for FOSS policies to undergo a stakeholder and consultative process to ensure that the views, needs and aspirations of all the stakeholders are fully considered. Without support from the FOSS community, few FOSS policies are likely to succeed.

Policy makers may also find that the FOSS community is more chaotic than most other stakeholders. There are few formal structures or organizations, no elected leaders and a diverse array of opinions on any single matter. Discussions on mailing lists often include heated arguments and public disagreements. Yet this movement, often in their spare time, has created software that matches or even surpasses software created by the finest corporations in the world.

There are several different methods commonly used to consult with stakeholders and policy makers can implement one or all of the following:

1) Round tables
2) Focus groups
3) Online consultations

Round tables

Round tables are gatherings (public or by invitation) involving representatives of stakeholder groups. While the FOSS community usually has no formal leader, there are normally individuals who are greatly respected and deferred to within the community. These individuals need to be identified and invited to attend round table sessions. Other affected stakeholders such as government ministries, educational institutes, civil society organizations and private sector representatives should also be included.

Round tables sessions can be very useful. Valuable insights and suggestions can sometimes be gathered and a sense of buy-in and ownership of the policy can be cultivated among the stakeholders. Networks of communication between FOSS practitioners and policy-makers can be forged. At times, the most valuable outcomes of these round tables are born during the breaks between sessions, rather than in the actual sessions themselves.

Round tables should be held regularly and the findings and recommendations of the sessions should be made as public as possible. Unfortunately, round tables can require much energy and resources to organize, especially in a geographically diverse country.

Focus groups

Where round tables cover a broad spectrum of the FOSS community and interests, focus groups tend to concentrate on a narrow area such as education, private sector or legal issues. Participants are usually much smaller in number and focused on these specific fields. The discussions in these sessions, besides being narrower in scope, are often more detailed and lengthier, likely yielding a much greater wealth of information.

Focus groups should be organized for critical areas that the policy targets, to ensure that the policy and implementation strategies are realistic, workable and have community support.

Online consultations

Online consultations leverage the power of the Internet to reach a wider audience and involve more stakeholders in the consultation process. Rather than meeting in person, participants communicate via mailing lists and discussion boards to cover all the various aspects of

the policy.

While the potential of online consultations seem appealing, there are still issues that the policy steering committee should be aware of and be especially careful of.

The most obvious issue is the availability of Internet access within the country often limits the participants of the consultation to only a small subsection of the population-at-large. Online consultations should never be the only stakeholder consultative process employed. However, if the consultation is public, then an online consultation may tap into the worldwide FOSS community, with valuable insights, past experiences and resources to contribute.

Another issue is how public should the online consultation be? Making the discussions fully public on a mailing list or bulletin board is true to the spirit of the FOSS community but can generate a large amount of “noise” on the discussion lists. Nor are all policy-makers comfortable with this. However, too restricted a online discussion, especially with too many restrictions on participant behavior, can discourage any meaningful input. A careful balance has to be found.

Policy-makers should expect some changes to occur during the consultation process, to accommodate stakeholder concerns and new strategies suggested by participants. At times, even significant changes may have to be considered. In many cases, it may be necessary to go through several rounds of consultation to produce a policy that reflects the needs and concerns of all involved.

Chapter 5: Strategies

For any conceivable policy goal, there are numerous strategies to achieve them, each with their own advantages and drawbacks. In this section, we try to list some of the more commonly used strategies, their advantages and drawbacks.

As always, no strategy should be taken and implemented without careful consideration of the local environment. Issues such as language, economic development, legal environment and cultural attitudes can make a particular strategy impossible.

Level of Development

The first and most important factor to note is that the level of economic development and ICT infrastructure greatly affects the strategies employed. Depending on whether a country already has an existing infrastructure, appropriate strategies change significantly.

Relatively advanced nations typically have existing ICT infrastructure and a trained technical pool. The infrastructure and skill set are typically on proprietary systems and this poses significant problems in several ways:

> Institutional resistance – Since decision makers are already familiar with their particular version of proprietary software and their skill sets are all rooted in this system, any FOSS policy is likely to encounter strong resistance. Any institution is resistant to change and ICT departments are no different. In new procurements, decision makers are likely more comfortable staying with the same technology as their legacy systems and biased against new technologies, including FOSS systems.
>
> Migration costs – Since legacy systems are already in place, implementing FOSS systems will eventually involve migrating existing systems over. Migration can sometimes be even more expensive than implementing a system from scratch. This is due to retraining costs and difficulties involved in migrating data from proprietary data formats or interfacing with legacy systems that were not built with interoperability with other systems in mind. Even in a minimally computerized environment, moving simple word processing documents from a proprietary data format to an open standard can take a significant amount of manpower.
>
> Incompatibilities – Even if a total migration to FOSS systems were contemplated, there will be a period of time when FOSS systems and proprietary systems must co-exist peacefully. Unfortunately, a common strategy with proprietary software is to make their systems

not fully compatible with other systems, be they competitor proprietary software or FOSS systems. In some cases, this incompatibility is relatively minor (text documents strangely formatted, etc) but in other cases, data interchange may be quite complicated.

Due to these reasons, implementing a FOSS infrastructure or at least creating a FOSS friendly environment is much simpler when there is only limited existing infrastructure.

With relatively developed countries, FOSS policy issues tend to focus on migration strategies, open standards, new procurement and co-existence issues. Relatively undeveloped countries focus instead on capacity building and legal issues, as they build the infrastructure that is FOSS friendly from the very beginning.

Policy and legal environment

Most countries already have an existing body of policies and laws that have evolved over time. Because the proprietary model of software development has been the predominant model until recently, existing policies and laws are normally favorable towards proprietary software. In some cases, this may mean some policies or laws create a hostile environment for FOSS.

Many countries are finding that they have to reexamine and often tweak their existing policies and laws to align with their FOSS goals. Often, these changes are not just within the ICT sector but within other sectors as well, such as intellectual property, industry promotion and education. For some of the more common issues, please refer to the “Cross-Sectoral Concerns” section later in this primer.

Countries that have directly addressed this problem frequently form a working group or committee to identify the different problems and work with the appropriate government body to address them. This often involves a process of awareness raising and education of policy makers in other sectors of the government. Obtaining consensus can often be a lengthy process and it helps to have solid support behind FOSS from the highest levels of government.

Government Procurement

Governments, especially in developing countries, are among the largest users and procurers of ICT tools. Their influence extends far beyond their direct purchases. Government procurements often have a trickle down effect as consumers of government outputs adjust their own ICT infrastructure for compatibility with government systems. Even minor changes in procurement policies can have significant impact.

To date, FOSS promotion strategies via government procurement throughout

the world fall into four broad categories. They are:

1. Mandating FOSS
2. Preferring FOSS
3. Mandating Open Standards
4. Best Value

Mandating FOSS

This is the most radical approach as it mandates the usage of FOSS systems throughout the government sector. In some countries, this means replacing the entire existing proprietary infrastructure, with large implementation and training costs involved. Other countries have chosen the less painful route of mandating FOSS for all new procurement.

Although quite a few proposals (legislative or otherwise) to this effect have been submitted, to date few have passed. The high costs and risks involved in such a approach have deterred most from trying this approach.

More conservative approaches, such as mandating the change of only a proportion of the infrastructure over to FOSS, are somewhat more common. Countries that fall into this category include Brazil (80% of all systems[32]), South Korea (20-30% of all systems[33]) and Thailand[34].

Such an approach greatly promotes FOSS usage and capacity in the local economy. However, it has been criticized as doing this at the expense of the proprietary software industry. Certain economically advanced countries have also criticized such policies as being protectionist and against the spirit of free trade.

China has a policy of blocking of foreign software usage in government offices[35], which does not mandate FOSS per se but still has a strong stimulating effect. The policy considers locally packaged FOSS systems as local software, even if the majority of its components are produced by the international FOSS community.

Preferring FOSS

Recognizing the difficulty of switching the entire government infrastructure over to FOSS, many governments have moderated their approach by requiring new procurements to prefer FOSS solutions. When all traditional commercial measures are equal (functionality, TCO, risks, stability, etc) then the FOSS solutions are selected in recognition of its social benefits, which can be hard to quantify.

This approach has the benefit of being easier and less risky to implement. It is also much more flexible, allowing procurements to decide on a case by case basis and take into account factors such as the possible lack of a local developer pool. However, the weaker mandate may not be enough to counter the advantage proprietary software enjoys when there is an established proprietary system.

Preferring FOSS: South Africa

South Africa's OSS strategy, sections 10.1 and 10.2 states:

Government will implement OSS where analysis shows it to be the appropriate option. The primary criteria for selecting software solutions will remain the improvement of efficiency, effectiveness and economy of service delivery by Government to its citizens.

OSS offers significant indirect advantages. Where the direct advantages and disadvantages of OSS and PS (Proprietary Software) are equally strong, and where circumstances in the specific situation do not render it inappropriate, opting for OSS will be preferable.

The full strategy can be obtained from:
http://www.oss.gov.za/docs/OSS_Strategy_v3.pdf

Mandating Open Standards

Mandating open standards often has a complementary effect on FOSS systems. One of the strongest weapons that proprietary software has to lock in its users is the usage of proprietary standards. FOSS systems are at a disadvantage in a mostly proprietary software environment due to interoperability reasons. The mandating of open standards would level the playing field and introduce increased competition, not just between proprietary software and FOSS but also between different proprietary software solutions. However, it often requires changing procedures and legacy documents that are still stored using proprietary standards.

The two areas often targeted by open standards advocates are in documents and web standards. Documents such as text documents and spreadsheets are typically stored in proprietary formats and may not be retrievable without the proper proprietary software, impeding the free exchange of information. Proprietary, closed web standards are ironic, since the World Wide Web is primarily based upon open standards. However, the dominance of a single web browser and its complementary web development tools from the same vendor have resulted in many web sites being created in a non-standards compliant fashion that are only accessible using Internet Explorer even though it would take minimal effort to make it cross-platform.

In some cases, mandating open standards would initially preclude certain proprietary software vendors from participating until such time that they added proper support of open standards into their products.

Emphasis on open standards is strongest in countries with mature ICT industries and infrastructure. The European Union, the United Kingdom, certain states within the United States and New Zealand[36] are among the governments looking to open standards.

Mandating Open Standards: Commonwealth of Massachusetts, United States of America

The Commonwealth of Massachusetts in the United States has the following position and policy statements:

Commonwealth's Position

- Effective and efficient government service delivery requires system integration and data sharing.
- Technology investments must be made based on total cost of ownership and best value to the Commonwealth. Component-based software development based on open standards allows for a more cost-effective build once, use many times approach.
- Open systems and specifications are often less costly to acquire, develop and maintain and do not result in vendor lock-in.

Policy Statement

- All prospective IT investments will comply with open standards referenced in the current version of the Enterprise Technology Reference Model.
- Existing IT systems will be reviewed for open standards compatibility and will be enhanced to achieve open standards compatibility where appropriate. Open standards solutions will be selected when existing systems are to be retired or need major enhancements.

The Commonwealth of Massachusetts' Enterprise Open Standards Policy can be viewed at:
http://www.state.ma.us/itd/openstandards.htm

Best Value

This approach focuses mostly on the economic value of FOSS, de-emphasizing the national and social benefits of wide scale FOSS adoption. This approach is the least controversial and is the standard policy in most countries.

However, due to the relative newness of FOSS and lack of general awareness, there have been calls for legislation or policy that

explicitly places FOSS on the same level playing field as established and reputable proprietary software. Multiple legislative initiatives at the state level in the United States were initiated, though none have passed to date. Any policy or legislation that explicitly requires that FOSS be considered on an equal footing with proprietary software is strongly opposed by proprietary software companies and intensely lobbied against.

Capacity Development

Many developing countries have noted that a serious shortage of FOSS capacity in their economies[37] holds back the implementation of major FOSS projects. A part of any major FOSS policy in developing countries will involve building a pool of local FOSS experts and companies to support the projects and users of FOSS in the country.

There are several focus areas for strategies on FOSS capacity development:

Traditional education

The traditional educational structure, starting from primary schools up through to the university level, can often be an excellent training ground for FOSS. There are a wide number of strategies in this sector and are too numerous to list exhaustively. Some of the more common strategies include:

Computer labs with FOSS installed – Projects such as these have been carried out in various countries including Brazil, Spain, Mongolia and Thailand. In some cases, the initial purpose was not to build FOSS capacity but to build ICT capacity in general. FOSS systems were utilized to reduce expenses but also had the added advantage of introducing an entire generation of students to FOSS relatively painlessly. Mongolia, among others, has reported that students had no difficulty learning and using FOSS systems. Implementing an initiative such as this requires a minimum level of FOSS capacity within the country to support the school computers otherwise some schools may find their computers not being used, as was experienced by the Goa Schools Project[38].

Ensuring that curriculum is software neutral – Many ICT literacy and even computer science programs in schools today, even at the university level, are written with a specific proprietary software suite in mind. By ensuring that the teaching of ICT concepts is decoupled from vendor specific skills, a more level competitive field can be achieved. In most cases, ICT skills can be taught on many platforms. For instance, basic ICT literacy skills such as email, web browsing and word processing can be taught on multiple proprietary and FOSS platforms. It may even be beneficial for students to experience two different implementations of a certain concept (one proprietary, one

FOSS) to ensure that the student learns flexible skills that can be easily transferred from one system to another. This does place an extra burden on educators and students alike though.

Some universities justify their teaching of skills tied to specific software packages by noting that their graduates are likely to be more productive immediately upon graduation. However, educators and policy-makers should keep in mind that most software systems have extremely short lifespans. A software package taught today is often outdated within five years and a graduate that has not learned broad, easily transferred concepts will become obsolete.

Scholarships/innovation awards – These are relatively small rewards given to individuals who contribute to the development of FOSS. Meant more as encouragement and recognition of their achievements rather than financial support, these awards raise awareness and interest in FOSS. This will in turn lead to more skilled FOSS practitioners both within the formal education system and the general population at large. While such awards have been implemented in places such as India, it is still too early to assess the long-term outcomes of such a program.

Certification and retraining programs

Most policies, especially those with shorter term goals, normally have a certification and retraining component as well. It typically takes less time to retrain already skilled technical personnel than it takes to train new personnel from scratch through the formal education system.

Retraining programs can either be part of continuing education classes taught in the evenings at colleges and universities or in specially mandated training centers. Some initiatives in the Asia-Pacific region have had government subsidized FOSS training as part of the retraining benefits given to unemployed or downsized workers.

A certification program is also recommended to complement the retraining effort. Certification ensures that all technical personnel possess a consistent and sufficient level of skill before they are allowed to implement FOSS projects. Such minimum standards are necessary to ensure that initial FOSS projects do not fail due to a lack of sufficiently qualified personnel, thereby creating a bad impression for future projects.

An alternative to creating a certification program from scratch (quite a significant undertaking) would be to utilize one of the globally recognized certification programs. Two of the notable certification programs for GNU/Linux (which is only a subset of the entire FOSS

movement) are the Linux Professional Institute (LPI)'s Certification (LPIC) or Red Hat's Red Hat Certified Engineer (RHCE) program.

LPIC – The LPI is a non-profit organization that administers a distribution neutral examination. Supported by a wide number of FOSS companies, the LPI tests the most commonly used FOSS skills via a written (or computerized) examination. The advantages of going with the LPIC are that its questions apply to all and that its examinations are relatively inexpensive (US$100 per exam normally. Under certain conditions, prices can be brought lower).

RHCE – This examination focuses primarily on Red Hat's distribution of Linux but a good proportion of the skills can be easily transferred between different FOSS distributions. The RHCE also includes a hands-on practical examination involving the actual set up a GNU/Linux system. The LPI does not offer a practical exam component at present. However, the RHCE is specific to a single Linux distribution and significantly more expensive.

Both organizations are willing to cooperate in localizing their examinations, so language considerations are not a major issue.

Training: Pakistan

Pakistan is training up to 10,000 end users and 1,000 system administrators on FOSS applications and systems as part of its e-government training project. The training will be focused on employees of the Federal and Provincial Governments in the major cities.

Targeted at increasing the standard and quality of public services through the effective use of ICT in the public sector, this training program is unusual in that it covers both proprietary (Microsoft Windows and Office) and FOSS (Linux, OpenOffice.org) applications. Trainees are then able to compare and choose between the different systems.

Further information can be found at:
http://pcb.gov.pk/project/project.asp

FOSS competency centers

A common strategy of many FOSS policies is the creation of a FOSS competency/research/compatibility center that performs a variety of functions:

1) FOSS awareness raising and promotion
2) Pilot projects implementing FOSS in e-government projects
3) Documenting current and best practices of FOSS usage

4) Technical support for government agencies
5) Training of both end-users and technicians
6) Aggregate news and resources on FOSS

These competency centers typically serve as cores of FOSS competency in a country, supporting the development of competencies elsewhere in the country, especially in the public sector. In larger countries, multiple centers may be established, one per major metropolitan area.

Research

Some policies include a requirement that software produced by publicly funded research should be released under a FOSS license. The rationale being that the fruits of public funds should be freely enjoyed by society as a whole. Releasing it under a FOSS license would ensure this, as opposed to a trend in some countries to have publicly funded research patented and licensed to commercial organizations to commercialize the results of the research.

The United States has a situation that is similar in principle – most of the software created by US departments is released into the public domain, available to all to utilize and take advantage of. As this situation has been in place for many years, some of the results are quite interesting. The VISTA[39] hospital information system released by the Veterans Administration is a widely used FOSS system that runs hundreds of hospitals around the world, saving millions of dollars in software costs in each hospital. The bioinformatics field throughout the world relies very much on software that is created by the National Center for Biotechnology Information (NCBI)[40]. In both cases, the primary beneficiaries are the citizens of the country itself, as adoption is normally quickest there. However, the benefits have ultimately spread globally.

Some critics of such initiatives say that releasing research under a FOSS license is unfair to the private sector as this would prevent the commercialization of this research. However, this criticism applies to only one or two of the 52+ FOSS licenses. There are a variety of licenses that are especially business friendly and some were even created by corporations such as Sun, IBM or Apple. Deciding the appropriate license for research is best left to the local legal experts.

FOSS as an industry

FOSS, while it challenges the traditional proprietary software industry, can still be a large contributor to the local economy. Large multinational corporations such as IBM and Red Hat, as well as countless small to medium sized companies such as MySQL AB, Digital Creations and Trolltech have generated strong profits by focusing on FOSS. While a dominant

traditional proprietary software firm like Microsoft is unlikely with the FOSS model, a vibrant ICT industry based on FOSS is quite possible and several countries have already begun policy initiatives to encourage a FOSS industry.

Larger corporations normally have the resources to adapt to an increased demand for FOSS skills in the economy. It is the Small- and Medium-sized Enterprises (SMEs) that often find it harder to adapt to a new environment or to reskill their workforce. To compensate for this, there are several strategies that governments may employ to encourage the growth of FOSS among SMEs:

Awareness raising

SMEs normally do not have access to the wealth of information resources that larger corporations have and may be unaware of the full benefits of FOSS. Awareness raising activities such as conferences, workshops and training sessions will help SMEs fully utilize this resource and incorporate it into their organizational strategies.

Government procurement

Although government procurement has been covered as a general method of promoting FOSS within a country, it can also serve as a vehicle to promote SMEs as well. Most government procurement contracts are too large for SMEs to compete for and this normally places them at a competitive disadvantage. However, some governments (notably the United States and to a lesser extent the United Kingdom) have requirements that a certain percentage of a government contract must be carried out by SMEs or be subcontracted out to SMEs by the the ultimate winner of the contract.

These initiatives have created strong partnerships between large corporations and SMEs, as well as assisted quite a few SMEs grow to a size where they are able to compete directly against the larger corporations in their respective countries.

Tax incentives

Besides directly creating demand via overt government purchases, governments can also increase the attractiveness of FOSS to SMEs by offering incentives such as tax breaks. Countries such as Singapore have offered tax reductions to companies that use the GNU/Linux operating system[41]. Such guaranteed cost savings make FOSS systems more attractive to SMEs, both for usage internally and as a service provided to clients.

Credit/Financial Assistance

In rare cases, governments can choose to support local FOSS companies via loans, loan guarantees, seed funding, venture capital funds and other financing methods. Assuming there is sufficient demand in the local economy for FOSS products and services, government support can reduce the financial roadblocks to starting and running a successful FOSS business, particularly among SMEs.

Many countries have credit programs for small businesses, including the United States' Small Business Administration (SBA). The SBA has put over US$30 billion in the hands of small business owners since 1958 to finance the growth of the United States' vibrant SME industry.[42] A modified version of such plans can be adopted to assist FOSS SMEs.

SME Incubator: Malaysia

An unusual project for creating a vibrant FOSS industry is the proposed Open Source Software Platform Investment Programme (OSS-PIP) to be funded by the Malaysia Venture Capital Management (MAVCAP).

MAVCAP is reportedly planning to invest between US$18 million to US$36 million into incubating a “full-fledged open source ecosystem” consisting of 20-40 companies. These companies would comprehensively cover the spectrum of systems and solutions that would be demanded by the government, enterprises and consumers[43].

MAVCAP is a venture capital company set up by Malaysia's Ministry of Finance to invest in and nurture the Malaysian technology sector. It was incorporated in 2001 with startup capital of about US$125 million[44].

Chapter 6: Cross Sectoral Concerns

FOSS, due to its unique nature, touches on more than just national ICT policy. Particularly in the area of Intellectual Property Rights (IPR) but also in other areas such as consumer rights and international trade, FOSS can wither or thrive depending on the specifics of the policies implemented. In this section, we look at some of the important non-ICT areas that must be aligned with national FOSS goals.

Intellectual Property Rights

As competition among intellectual property companies (including software companies) intensifies, a variety of legal and policy tactics are being adopted. Some of these tactics, can be disruptive to the creation and usage of FOSS, especially when governments are influenced into creating a legal environment that supports them.

Patents

Software and business process patents are particularly dangerous to FOSS for two reasons – FOSS developers rarely patent anything and paying for patent royalties is impractical with FOSS software. Because of these reasons, software companies threatened by FOSS can utilize software and business process patents to stifle FOSS development.

The nature of FOSS development is typically very open and cooperative, with a diverse number of individuals and organizations involved. Such communities rarely file patents for any of the ideas that they produce. Besides disrupting the cooperative nature of FOSS itself, patents are expensive to file for. In the United States, the absolute minimum for a successful patent hovers around US$1,025[45]. Most patent filers utilize the services of specialized patent attorneys and this greatly increases the costs, with numbers of between US$8,000 to US$30,000 frequently cited. As can be expected, few FOSS patents are filed and FOSS is often defenseless against patents that competitors may possess.

FOSS is also unable to pay royalties on patents held by others. Royalties would restrict the ability to freely redistribute the software, destroying one of the fundamentals of FOSS. Royalties are also incompatible with some of the major FOSS licenses. It becomes extremely costly for distributors such as Debian, a non-profit community organization, to track, collect and pay royalties every time someone shares a copy of Debian's software with someone else.

Note that FOSS is not affected by traditional patents covering inventions, manufacturing processes and most other tangible patents.

It is primarily the patenting of software and to a lesser extent, business practices, that threaten FOSS. If software patents are allowed within a country's legal system, FOSS adoption may be severely threatened.

Software Patents

Software patents are a relatively new concept and have been controversial from the very beginning. Software patents are not currently common outside of the United States, though efforts to introduce them are in progress worldwide, usually lobbied for by large multinational corporations.

In the United States, software could not be patented until the 1980s. Before this, patents could not be granted on scientific truths or mathematical expressions of it. Patents were granted on processes, machines, articles of manufacture and compositions of matter. Software was considered as mathematical algorithms and not patentable.

This changed in 1981, due to the US Supreme Court forcing the Patent and Trademark Office to grant a patent on an invention that utilized computer software in *Diamond v. Diehr*. Since then, software patents have become more common, with thousands of them being issued in the United States every year.

Source: The History of Software Patents
http://www.bitlaw.com/software-patent/history.html

Copyright

Copyright law is fairly standard throughout the world. However, there are a few areas where copyright laws differ somewhat and therefore affect FOSS in different ways.

Derivative works – There is sometimes ambiguity over what constitutes a derivative work in software. This has consequences on the some of most popular FOSS licenses, particularly the Free Software Foundation's General Public License (GPL). The GPL has a provision that requires that all derivative works must be released under the same license as the GPL. The GPL carries its own definition of derivative work but this may not be applicable in every nation in

the world. Clarification of these issues may be necessary to ensure that the legal basis of FOSS licenses are strong in the country.

Reverse engineering – Reverse engineering is the method of taking something apart in order to figure out how it works. Most copyright laws allow some form of reverse engineering, especially for interoperability and compatibility reasons. However, some recent laws, particularly the Digital Millennium Copyright Act (DMCA) in the United States have placed restrictions on reverse engineering. This directly impacts FOSS as FOSS interoperability with proprietary software often comes about through reverse engineering proprietary protocols and data standards. If reverse engineering is prohibited, proprietary software producers are able to erect barriers to keep FOSS (and other proprietary competitors) out of their markets using proprietary communication protocols and data standards. This would make it easier for vendors to lock their customers into their solutions.

Consumer Rights

Overly aggressive consumer protection laws can also destroy the FOSS industry, particularly with regards to implied warranties. This is ironic since FOSS tends to give consumers far more rights than proprietary software. Most proprietary software licenses also explicitly disclaim any liability.

Certain countries are either considering or have enacted laws that do not allow software distributors to totally absolve themselves of liability should their product not work as expected. Doing so would result in almost no software being released as FOSS as individual programmers are unable to handle the legal liability involved when the software can be freely distributed to just about anyone and used for any purpose. Such laws are also incompatible with the majority of FOSS licenses.

It should be noted that nothing prevents a commercial FOSS company from offering warranties and assuming liability risks on software that they sell. Many companies such as Red Hat, SuSE or Mandrake offer industry standard warranties. However, individual projects or non-commercial community distributions such as Debian cannot do the same. This would seriously cripple the vibrant FOSS community as it would be impossible for anyone other than large corporations to participate in FOSS development or even software development in general.

Conclusion

Free and Open Source Software is a proven and mature solution set that offers many opportunities to developing nations building their ICT infrastructure. Numerous corporations, institutions and government agencies are utilizing FOSS throughout their ICT infrastructure and running mission critical applications upon it. Countless national governments are relying on FOSS to kick start their ICT infrastructure, develop local capacity, increase competition in the ICT industry and reduce their dependence on any single vendor.

However, FOSS is not a “magic bullet” that immediately solves all of a nation's problems. Its benefits are real and tangible, but implementing a policy that fully captures these benefits with no significant drawbacks is very difficult. Tony Stanco from the Center of Open Source and Government highlights several strong points in the South African Open Source Strategy[46]:

1. Official Statement of Recognition of the Legitimacy of FOSS. Since FOSS is new and unknown to most decision makers, official recognition and legitimacy has a strong promoting effect on FOSS adoption.

2. Designation of Particular Government Agency to Lead FOSS Program. A designated lead agency is responsible for coordination, communication and execution of the policy. The single point of responsibility reduces the likelihood that a FOSS policy is not issued and forgotten or improperly implemented.

3. Level Playing Field in Government Procurement. By ensuring a truly level playing field between the different options, a government can increase competition in procurement. This involves, among other things, ensuring that open standards and open protocols are used.

4. Appreciation of Social Value of FOSS. Some policies focus too much on the financial benefits of FOSS and leave out additional benefits such as capacity building, transparency in government and greater citizen access. A policy that focuses on all the benefits addresses both the economic and social needs of a country.

5. Phased Implementation. Any large undertaking carries with it a large amount of risk, especially in developing countries where local capacity is highly limited. A phased implementation plan, including pilot projects and and a prudent transition strategy allows a government to build institutional knowledge and capacity while developing best practices and case studies for future projects.

No national policy is easy to formulate or implement. The unique conditions

in each country, the demands of the different stakeholders and the challenges faced present a different problem to each country. Still, the potential benefits to be gained can be enormous and policy makers only need to persevere to reap the full benefits that FOSS can bring to each nation.

Glossary

Free Software
The word “free” in Free Software refers to the users' freedom to run, copy, distribute, study, change and improve the software. It does not refer to the price of the software. More precisely, a program is Free Software if users have the four freedoms:

- The freedom to run the program, for any purpose.
- The freedom to study how the program works, and adapt it to your needs. Access to the source code is a precondition for this.
- The freedom to redistribute copies so you can help your neighbor.
- The freedom to improve the program, and release your improvements to the public, so that the whole community benefits. Access to the source code is a precondition for this.

The definition of Free Software and a more detail explanation is available at http://www.fsf.org/philosophy/free-sw.html.

GNU
GNU is an often heard acronym when discussing FOSS. It is a recursive acronym for “GNU’s Not Unix” and the name of a project started in 1984 by Richard Stallman to develop a complete UNIX-like operating system that is available as Free Software. This is called the GNU operating system.

GPL
The General Public License (GPL) was originally used as the license for “Free Software” distributed by the Free Software Foundation (FSF). Under the GPL, users may run, copy and modify the software, and distribute the modified software. However, users are not allowed to add their own restrictions and the modified software must be released under the same licensing terms. The GPL also requires that the source code be made available to anyone who possess the program binary.

Localization
In the context of software, localization is the process of adapting, translating and customizing a product for a specific market. This means the modification of the interface so that it is meaningful and comprehensible to the local user of the product. Apart from the linguistic issues, localization also needs to address content and cultural issues as well as technical issues.

NSA
NSA stands for National Security Agency. NSA is the United States of America’s cryptologic organization. It coordinates, directs, and performs

highly specialized activities to protect U.S. information systems and produce foreign intelligence information. A high technology organization, NSA is on the frontiers of communications and data processing. It is also one of the most important centers of foreign language analysis and research within the government.

Information adapted from NSA website at:
http://www.nsa.gov/about/index.cfm

Open Source
Open Source software does not only mean access to the source code. To qualify as Open Source software, the distribution terms of the software must comply with the following criteria:

- Free Redistribution
- Availability of Source Code
- Possibility of Derived Works
- Integrity of The Author's Source Code
- No Discrimination Against Persons or Groups
- No Discrimination Against Fields of Endeavor
- Distribution of License
- License Must Not Be Specific to a Product
- License Must Not Restrict Other Software
- License Must Be Technology-Neutral

For further explanation of the definition of Open Source please refer to: http://www.opensource.org/docs/definition.php

Operating System
An Operating System (OS) is a collection of software that controls the hardware and software applications on a computer. The OS manages and allocates the physical resources (CPU processing time, hard disk space, inputs from the keyboard, etc.) among the different applications that run on it. Examples of an OS are Microsoft Windows, GNU/Linux, Solaris and Mac OS X. Most modern OS bundle additional applications (word processors, media players, web browsers) that are not traditionally defined as part of an Operating System.

Reverse Engineer
To reverse engineer a product is to take apart a functioning product to understand how it works and its full functionality. Often used as part of a process to create a separate product that functions in a similar fashion. Reverse engineering for compatibility purposes is protected by law in many countries.

Source Code
Software source code is the set of programming instructions written by the programmer using a particular computer language. Most computer

languages are easily read and understood by a competent programmer. In order for the computer to understand and run the software, the source code must be compiled or "translated" into machine code (also referred to as binary code, executable code or object code). To modify software, the source code must be available for modifications, as the machine code is not human-readable to all but the most exceptional of humans.

TCO

TCO stands for Total Cost of Ownership. This includes all of the costs involved in a technology or business solution. In addition to the initial investment cost, such costs include training, maintenance, support, replacement costs, and the like. In the case of software, the TCO should include the initial cost of the software; upgrade cost; and maintenance, support and training costs.

Useful Resources

Portals focused on FOSS and government

eGovOS - The Center of Open Source and Government. Many useful resources here, including many policies from various government departments and reports.
http://www.egovos.org/

The European Commission's Open Source Observatory:
http://europa.eu.int/ISPO/ida/jsps/index.jsp?fuseAction=showChapter&chapterID=452&preChapterID=0

The United Nations Development Information Programme (UNDP) Networking and Information Technology Observatory (NITO)'s Open Source section:
http://www.sdnp.undp.org/perl/news/articles.pl?do=browse&categories=10

The UNDP's International Open Source Network:
http://www.iosn.net

Policy Related Documents

United Kingdom's Open Source Software policy, as issued by the Office of Government Commerce:
http://www.ogc.gov.uk/embedded_object.asp?docid=2498

Extensive FLOSS survey conducted by the International Institute of Infonomics at the University of Maastricht, The Netherlands:
http://www.infonomics.nl/FLOSS/report/

South Africa's Open Source Software strategy:
http://www.oss.gov.za/docs/OSS_Strategy_v3.pdf

Open Standards policy of Commonwealth of Massachusetts, United States of America
http://www.state.ma.us/itd/openstandards.htm

Report of FOSS usage in the US Department of Defense, prepared by MITRE Corporation:
http://www.egovos.org/rawmedia_repository/588347ad_c97c_48b9_a63d_821cb0e8422d?/document.pdf

Feasibility study conducted by the Swedish Agency for Public Management:
http://www.statskontoret.se/pdf/200308eng.pdf

“Open-source software in e-government” report produced by the Danish Board of Technology:
http://www.tekno.dk/subpage.php3?article=969&survey=14&language=uk&front=1

Endnotes

[1] Wheeler, David, "Why OSS/FS? Look at the Numbers!" http://www.dwheeler.com/oss_fs_why.html

[2] Knienieder, Tom, "Netgear WG602 Accesspoint vulnerability", *securityfocus.com* [home page online]; available from http://www.securityfocus.com/archive/1/365069; Internet, accessed 6 June, 2004.

[3] "Cisco Security Advisory: A Default Username and Password in WLSE and HSE Devices", *cisco.com* [home page online]; available from http://www.cisco.com/warp/public/707/cisco-sa-20040407-username.shtml; Internet; accessed 20 April, 2004.

[4] Loyola, Roman, "China: The Republic of Linux", program aired 13 August 2003, available from http://www.techtv.com/screensavers/linux/story/0,24330,3395670,00.html; Internet; accessed 20 August 2003.

[5] Kettman, Steve, "Germany Denies Microsoft Ban", 19 March 2001, *Wired News*, [home page online]; available from http://www.wired.com/news/politics/0,1283,42502,00.html; Internet; accessed 20 August 2003.

[6] Ghosh, Krieger, Glott, Robles, "Free/Libre and Open Source Software: Survey and Study. Part 2B: Open Source Software in the Public Sector: Policy within the European Union", June 2002, http://www.infonomics.nl/FLOSS/report/FLOSSFinal_2b.pdf

[7] Ghosh, Krieger, Glott, Robles, "Free/Libre and Open Source Software: Survey and Study. Part 2B: Open Source Software in the Public Sector: Policy within the European Union", June 2002, http://www.infonomics.nl/FLOSS/report/FLOSSFinal_2b.pdf

[8] Data downloaded from http://www.bcb.gov.br/ftp/NotaEcon/NI200401sei.zip

[9] Jayme, Frederico Gonzaga Junior, "Balance-of-payments constrained economic growth in Brazil", 2001, Revista de Economia Política, 2003, vol. 23, no. 1.; available from http://ideas.repec.org/p/cdp/texdis/td155.html

[10] Roberts, Paul, "Software Piracy Declines 10 Percent", 03 June, 2003, *Infoworld.com* [home page online]; available from http://www.infoworld.com/article/03/06/03/HNpiracydecline_1.html?security; Internet; accessed on November 8, 2003.

[11] Ghosh, Rishab Aiyer, "License fees and GDP per capita: The case for open source in developing countries", 1st December 2003, Volume 8, Number 12, *First Monday*; [home page online]; available from http://firstmonday.org/issues/issue8_12/ghosh/index.html; Internet; accessed 7 June 2004.

[12] Walsh, Mary Williams. "Microsoft in War of Words", Los Angeles Times. http://www.tungutaekni.is/ymis_frodleikur/war_of_words.html

[13] Shankland, Kane, Lemos, "How Linux saved Amazon Millions", 30 October 2001, *Cnet News.com* [home page online]; available fromhttp://news.com.com/2100-1001-275155.html; Internet; accessed on November 7, 2003.

[14] Sisk, Michael, "Linux Woos Wall St.", August 2003, *Bank Technology News*; available from http://www.banktechnews.com/cgi-bin/readstory.pl?story=20030801BTNC617.xml; Internet; accessed on November 7, 2003.

[15] Orzech, Dan, "Linux TCO: Less Than Half The Cost of Windows", 7 October 2002, *CIO Update*; available from http://www.cioupdate.com/article.php/10493_1477911; Internet; accessed on November 7, 2003.

[16] "netproject – Cost of Ownership" [home page online]; available from http://www.netproject.com/opensource/coo.html; Internet; accessed on November 7, 2003.

[17] Maguire, James, "Windows vs. Linux: TCO Feud Rages On", 01 August 2003, *Newsfactor Network* [home page online]; available from http://www.newsfactor.com/perl/story/22012.html; Internet; accessed on November 8, 2003.

[18] Lemos, Robert, "Merrill Lynch: Linux saves money", 7 June 2003, *CNet News.com* [home page online]; available from http://news.com.com/2100-1016_3-1014287.html?tag=fd_top; Internet; accessed on November 8, 2003.

[19] Miller, Robin, "Largo loves Linux more than ever", 9 December 2002, Newsforge.com; available from http://newsforge.com/article.pl?sid=02/12/04/2346215&mode=thread&tid=19; Internet; accessed on November 7, 2003.

[20] Glover, Tony, "Microsoft losing market grip as rivals go on the offensive", 18 May 2002, *Scotland on Sunday*; available from http://www.scotlandonsunday.com/business.cfm?id=562032003; Internet; accessed on November 8, 2003.

[21] Kuan, Jenniifer, "How to Succeed in Business with Open Source Software", Infonomics.nl [home page online]; available from http://www.infonomics.nl/FLOSS/workshop/papers/kuan.htm; Internet; accessed on February 5, 2004

[22] "Reasoning Study Reveals Code Quality of MySQL Open Source Database Ranks Higher than Commercial Equivalents", [home page online]; available from: http://www.reasoning.com/newsevents/pr/12_15_03.html; Internet; accessed 19 April 2004.

[23] Luening, Eric, "Windows users pay for hacker insurance", 29 May 2001, *CNet News.com* [home page online]; available from http://news.com.com/2100-1001-258392.html?legacy=cnet; Internet; accessed on 8 November, 2003.

[24] Forristal, Shipley, "Vulnerability Assessment Scanners"; 8 January 2001, *Network Computing*; [home page online]; available from: http://www.networkcomputing.com/1201/1201f1b1.html; Internet; accessed 6 June, 2004.

[25] Ghosh, Krieger, Glott, Robles, "Free/Libre and Open Source Software: Survey and Study. Part 2B: Open Source Software in the Public Sector: Policy within the European Union", June 2002; available from http://www.infonomics.nl/FLOSS/report/FLOSSFinal_2b.pdf; Internet; accessed on November 8, 2003.

[26] "Use of Free and Open Source Software in the U.S. Department of Defense", MITRE Corporation, available from http://www.egovos.org/rawmedia_repository/588347ad_c97c_48b9_a63d_821cb0e8422d?/document.pdf; Internet; accessed 6 June, 2004.

[27] Ghosh, Krieger, Glott, Robles, "Free/Libre and Open Source Software: Survey and Study. Part 2B: Open Source Software in the Public Sector: Policy within the European Union", June 2002; available from http://www.infonomics.nl/FLOSS/report/FLOSSFinal_2b.pdf; Internet; accessed on November 8, 2003.

[28] Ghosh, Krieger, Glott, Robles, "Free/Libre and Open Source Software: Survey and Study. Part 2B: Open Source Software in the Public Sector: Policy within the European Union", June 2002; available from http://www.infonomics.nl/FLOSS/report/FLOSSFinal_2b.pdf; Internet; accessed on November 8, 2003.

[29] Weber, Steven, "Open Source in Developing Economies", *Social Science Research Council* [home page online]; available from http://www.ssrc.org/programs/itic/publications/ITST_materials/webernote2.pdf; Internet; accessed on February 6, 2004

[30] Myoung, Seung eun, "Korea, China, Japan start open-source collaboration", 2 April, 2004, *CNETAsia* [home page online]; available from http://asia.cnet.com/newstech/systems/0,39001153,39174180,00.htm; Internet; accessed on 20 April 2004

[31] Williams, Peter, "Europe picks Penguin to link government IT", 18 July 2003, *VNUNet.com* [home page online]; available from http://www.vnunet.com/News/1142411; Internet; accessed on November 8, 2003.

[32] "The Brazilian Public Sector to Choose Free Software", 2 June 2003, *PCLinuxOnline* [home page online]; available from http://www.pclinuxonline.com/modules.php?name=News&file=article&sid=6879; Internet; accessed on November 8, 2003.

[33] Myung, Seung Eun, "Korea jettisons Windows for Linux", 1 October 2003, *silicon.com* [home page online]; available from http://www.silicon.com/software/os/0,39024651,10006225,00.htm; Internet, accessed on February 4, 2004.

[34] "ICT Ministry sets Linux targets for government sector", 23 June, 2003, *Bangkok Post*

[35] "China blocks foreign software use in gov't", 18 August 2003, *CNETAsia* [home page online]; available from http://asia.cnet.com/newstech/applications/0,39001094,39146335,00.htm; Internet; accessed on November 8, 2003.

[36] Brislen, Paul, "Microsoft defends XML patent, offers royalty-free license", 5 February 2004, *Computerworld.com New Zealand* [home page online]; available from http://computerworld.co.nz/news.nsf/NL/AB73832F819E0AF5CC256E30000C7BFE; Internet; accesed on February 6, 2004.

[37] Expressed during the Free and Open Source Software Asia-Pacific conference held February 9-11 in Kuala Lumpur, Malaysia. Summary findings and recommendations are available on the Internet at http://www.iosn.net/downloads/fossap-report-final.pdf; accessed on April 11, 2004.

A second conference in Africa also expressed a similar need. The Idlelo conference was held Jan 11-15 in Cape Town, South Africa. Final report available on the Intenet at: http://www.fossfa.net/tiki-download_file.php?fileId=3; accessed 11 April, 2004

[38] Martyris, Daryl, 'Community – Government partnerships and open source technology for low cost IT access in India – A case study', July 2003; available from http://www.developmentgateway.com/node/133831/sdm/blob?pid=5474; Internet; accessed on November 7, 2003.

[39] More information available on the Internet at: http://www.worldvista.org/vista/index.html; accessed April 11, 2004.

[40] The basic algorithms that are used in biotechnology research is based on NCBI's tools. Although some companies have reimplemented the algorithms, most use the existing algorithms and add value on top of it. Further information can be found on the Internet at http://www.ncbi.nih.gov/; accessed April 11, 2004.

[41] UNCTAD Secretariat, "e-commerce and development report 2003", 2003, United Nations Conference on Trade and Development; available from: http://r0.unctad.org/ecommerce/ecommerce_en/edr03_en.htm#downloads

[42] "Overview & History of the SBA", [home page online]; available from http://www.sba.gov/aboutsba/history.html; Internet; accessed 19 April, 2004.

[43] Peterson, Cynthia, "Is Open Source Worth It?", 22 March, 2004, Computimes; available online from http://www.linuxinsider.com/perl/story/33192.html; accessed 19 April, 2004.

[44] "MAVCAP: The Company", [home page online]; available from http://www.mavcap.com.my/ourstory/thecompany.htm; Internet; accessed 19 April, 2004

[45] "How Much Does A Patent Cost?", [home page online]; available from http://www.ipwatchdog.com/patent_cost.html; accessed 20 April, 2004.

[46] Stanco, Tony, "EGOVOS endorses South Africa's Open Source strategy", 18 June, 2003, *Newsforge.com*; [home page online]; available from http://www.newsforge.com/business/03/06/17/1655215.shtml?tid=19; Internet; accessed 6 June 2004.