

International
Open
Source
Network



Free/Open Source Software

Government Policy

Kenneth Wong



Series Foreword by

RICHARD M STALLMAN



Asia-Pacific Development Information Programme
e-Primers on Free/Open Source Software

Free/Open Source Software: Government Policy

Kenneth Wong



© United Nations Development Programme-Asia Pacific Development Information Programme (UNDP-APDIP) - 2004
Web: www.apdip.net
Email: info@apdip.net

This publication is released under the Creative Commons Attribution 2.0 license.
For full details of the license, please refer to the following:
creativecommons.org/licenses/by/2.0/legalcode

ELSEVIER
A division of
Reed Elsevier India Private Limited
17A/1, Lajpat Nagar IV,
New Delhi 110 024
Tel: 91-11-26447160
Fax: 91-11-26447156
Website: www.asiaelsevier.com

ISBN: 81-8147-755-3

Academic Press, Butterworth-Heinemann, Digital Press, Focal Press, Morgan Kaufmann, North Holland and Pergamon are the Science and Technology imprints of Elsevier.

Printed and bound in India

TABLE OF CONTENTS

FOREWORD	v
PREFACE	vii
ACKNOWLEDGEMENT	viii
OVERVIEW	ix
INTRODUCTION	1
THE STRATEGIC IMPORTANCE OF FOSS	3
Strategic Benefits	3
Developing Local Capacity/Industry	3
Reducing Imports/Conserving Foreign Exchange	4
Enhancing National Security	5
Reducing Copyright Infringements	6
Enabling Localization	6
Economic Benefits	6
Increasing Competition	6
Reducing Total Cost of Ownership	7
Enhancing Security	8
Achieving Vendor Independence	9
Social Benefits	10
Increasing Access to Information	10
WHY IS A FOSS POLICY NEEDED?	11
Limited Marketing	11
Non-commercial Benefits	11
Entrenched Legacy Systems	12
POLICY FORMULATION APPROACH	13
Motivations and Needs Assessment	13
Existing and Proposed National Policies	13
Regional Context	13
Establishing Goals and Targets	14
Strategy Formulation	14
Brainstorming	15
Stakeholder and Consultative Process	15
Round Tables	15
Focus Groups	16
Online Consultations	16
STRATEGIES	17
Mandating FOSS	17
Preferring FOSS	17
Mandating Open Standards	18

Best Value	19
Level of Development	19
Policy and Legal Environment	20
Capacity Development	20
Traditional Education	20
Certification and Retraining Programmes	21
FOSS Competency Centres	22
Research	22
FOSS as an Industry	23
Awareness Raising	23
Government Procurement	23
Tax Incentives	24
Credit/Financial Assistance	24
CROSS SECTORAL CONCERNS	25
Patents	25
Copyright	26
Consumer Rights	26
CONCLUSION	27
GLOSSARY	28
USEFUL RESOURCES	30
Portals Focused on FOSS and government	30
Policy Related Documents	30
ABOUT THE AUTHOR	31

FOREWORD

Free software means software that respects the user's freedom. It means that users are free to run the programs as they wish, free to study and change the software (or hire others to do it for them), free to redistribute copies to others, and free to publish modified versions. As a consequence, users are free to share, and form communities to exercise effective control over the software they use. Free software may also be gratis, zero price, but this is not always the case.

Some users refer to this as open source. That term focuses on the technical benefits that result when software can be reviewed by large numbers of developers and users, the business advantages of using it, and the business models that support its development and use. The term free software refers to the social and ethical importance of freedom, as well as to the practical benefits it brings.

The social and practical imperatives for sharing and changing useful works such as software create a revolution in what it means to publish. The technology of the printing press taught readers to think of written works as fixed – written once, for readers to use passively thereafter. This technology did not make it easy for readers, who generally did not have presses of their own, to adapt, improve, and share copies of books, and they became accustomed to these limitations. So when citizens in large numbers began to use personal computers, many did not question the legal and business systems that placed similar limitations on using software. But this time, the limitation was not a natural consequence of the technology. It was imposed by software developers, who found it profitable to keep users under their control. The developers forbid users to share the software, and by denying users the source code, prevent them from changing it.

Today, however, there is an alternative: free software, the product of a community built by a million willing developers – some volunteers, some paid to contribute. Free software allows users to take full advantage of what their computers can do. It is our door to escape from the limitations of the printing press, into a world where useful works, such as software, are developed among the users, by the users, for the users.

Where software is not made artificially scarce by misguided incentive systems that encourage development of software by locking up its users.

Where the users can collaborate as they see fit in making the software do what they want it to do. Where the users of software are free to run it, share it, and adapt it together or individually. This is the world of Free and Open Source Software (FOSS).

Non-free software keeps users divided and helpless. When a nation increases the use of non-free software, it is not development, it is permanent dependency. Only the use of FOSS permits sustainable development – it is technology that local people are free to learn about, maintain, adapt, and reapply.

How can governments move their countries towards FOSS? In two ways. First, switching to Free Software in schools will teach children the spirit of community cooperation, while producing graduates that are skilled in using and maintaining Free Software. Second, mandating migration to Free Software in government agencies will create demand for these graduates' skills, and build a local economy of Free Software support.

The International Open Source Network (IOSN) is an initiative of UNDP's Asia-Pacific Development Information Programme, and operates under the principle of "Software Freedom for All" (SFA). Its work includes provision of support and assistance, a centre of excellence, and an information clearing-house for Free and Open Source Software in the Asia-Pacific region.

Through the IOSN/SFA initiative, UNDP provides policy support and advisory services to government bodies, non-profit organizations, donor agencies and others. It publishes practical tools and materials,

including simple “how to” primers and guidebooks, training materials, and live CDs of the GNU/Linux operating system for FOSS practitioners and end-users. It also supports FOSS R&D activities in localization and in other areas, and organizes conferences and training programmes to network stakeholders and strengthen local capacities. It welcomes both those interested in benefiting from these services and those who would like to collaborate in extending them.

I’m pleased to cooperate with the work of IOSN/SFA, APDIP and UNDP in taking the message of software freedom to the public and the development sector. Together we can help overcome the digital divide, and replace dependency with development.

Richard M. Stallman

PREFACE

Free and Open Source Software (FOSS) has grown incredibly in the past few years. Once considered a hobbyist's 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. Countries such as Brazil, China, Malaysia, South Africa, and Viet Nam, are implementing nationwide policies or legislation promoting FOSS. While the often-cited cost and stability benefits of FOSS are attractive, governments often choose to promote FOSS in their own countries for a variety of other reasons.

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

The 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).

This primer is licensed under an Open Content license¹ and is also available for download from the IOSN website at: www.iosn.net/downloads/foss_gov_primer.pdf.

¹This primer is licensed under the Creative Commons Attribution 2.0 license. Full details of the license can be found at: creativecommons.org/licenses/by/2.0/.

ACKNOWLEDGEMENTS

The author would like to thank everyone who took the time to share their valuable experience and insights. Peer reviewers for this document include Shahid Akhtar, Colin Charles, Dr. Nah Soo Hoe, Gopi Pradhan, Phet Sayo, Imran William Smith, Richard Stallman, Tan Wooi Tong, Atsushi Yamanaka and Raul Zambrano.

The author would also like to thank Sunil Abraham, Khairil Yusof and Helena Loh. Without their assistance, this primer could never have been written.

Additional thanks goes to the United Nations Development Programme and the International Development Research Centre. Their support has allowed this primer and others like it to be written and shared.

Finally, thanks goes out to the countless individuals and organizations that have contributed their time and resources into making the FOSS movement what it is today.

OVERVIEW

The Benefits of a FOSS Policy

Free/Open Source Software (FOSS) has a wider perspective than a software development methodology. It not only increases access, ownership and control of Information and Communication Technologies (ICT), but also provides a framework for the usage and sharing of intellectual capital in a way that is applicable to many areas of development endeavour. FOSS can play an important role in the application of ICTs for achieving the Millennium Development Goals (MDGs).

This primer covers the main benefits of FOSS. At the national level, FOSS aids in the development of local capacity/industry, reduces imports, conserves foreign exchange, increases the security of the national ICT infrastructure (this is distinct from application level security), reduces copyright infringement and brings localized ICT tools to help develop local knowledge communities.

FOSS provides many socio-economic benefits. The most commonly cited are fostering the ICT industry through increased competition, lowering the ICT application cost and Total Cost of Ownership (TCO), increasing access to powerful yet localized ICT applications, increasing security of ICT applications and providing vendor independence.

Yet for all these clear benefits, many nations find that without a national FOSS policy, the uptake of FOSS in the country is far too slow for their needs. There are a number of reasons why FOSS requires policy intervention, including limited marketing of FOSS, lack of attention to its many non-commercial benefits and the need to overcome entrenched legacy systems.

Policy Formulation Approach

There are several stages in the policy formulation process. Although, this primer does not address the process in totality, it does cover some points uniquely important to FOSS. The stages include:

Establishing motivations and assessing environment: Establishing the motivations behind a FOSS policy and getting support from the top levels of government are critical as FOSS policies often touch many aspects of government and influence various other policies. It is also important to assess how existing national policies and regional initiatives affect the implementation of a national FOSS policy. Certain regions, particularly the Asia-Pacific and Latin American regions, are very active in FOSS initiatives and policies. Regional initiatives not only make it easier to implement FOSS policies but also make it imperative to consider the implications of being surrounded by FOSS-using countries.

Establishing goals and targets : Once the necessity of a FOSS policy has been established, measurable and objective goals set the pace and tone of FOSS implementation. Targets and goals such as, “100 percent awareness of FOSS among CIOs” and “10 percent of all government IT personnel have FOSS certifications” set clear directions for the strategies to be developed at a later stage.

Strategy formulation: Apart from the strategies section of this primer, another excellent source of ideas and strategies for FOSS promotion is the existing FOSS community within the country and region. FOSS advocates have an in-depth understanding of the local ICT environment and are also familiar with both local and international initiatives. Any strategy formulation committee should tap the knowledge and expertise of the local community groups.

Stakeholder consultation process: Stakeholder consultations result in policies that are well thought out and have buy-in from a broader segment of the society. The FOSS communities are used to a more open and transparent decision making process and tend to demand this. Policy-makers are cautioned that the typical FOSS discussion is very vocal and opinionated and often results in disagreements. Still, the open

process has produced software, tools and knowledge that compare favourably with and, in some cases, are superior to those developed by the traditional proprietary development model.

FOSS Policy Strategies

Since, it is impossible to cover all the strategies that other countries have utilized in implementing their FOSS policies, the more common strategies are summarized in this primer. These strategies touch on multiple areas, such as capacity development; the policy and legal environment; procurement strategies; research; and software industry development. Key factors, such as the level of ICT industry development in a country, also affect whether strategies should focus on building capacity or migrating infrastructure.

Cross Sectoral Concerns

There are also various other policy areas interrelated with FOSS policies. Among these are two aspects, commonly referred to as “Intellectual Property” – patents and copyrights. Software patents today pose the single greatest threat to FOSS. In the United States and other countries where software patents are common, proprietary software companies that are unable to compete with FOSS in terms of added innovation and value, can resort to using software patents to cripple or even halt FOSS adoption in a country. To a much lesser extent, recent changes to copyright laws or legislation that affect copyright laws in developed countries can reduce the effectiveness or outlaw certain classes of FOSS. Even consumer protection laws can affect FOSS, albeit in a limited way. Thus, it is important for any country implementing FOSS policies to make a careful survey of their existing policies and regulations.

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).²

The above quotation summarizes the guiding principles of FOSS – the freedom to use, understand, modify and distribute software. Seemingly a simple matter, today, these principles can have a profound impact on the economics and dynamics of the software industry.

The proprietary 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 the software but instead purchase the license or right to use the software, in ways that are tightly controlled by the software producer. This software cannot be modified by the user should the need arise. Modifications and improvements can be made only by the software producer and these often come at an additional cost to customers.

In this industry, software is rarely shared and competitors often reinvent the wheel, re-implementing functionality that their competitors have already implemented. Often this re-implementation 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. For example, the cost to re-implement an equivalent of Sun's Solaris operating system would be prohibitively high.

With its fundamental freedoms, FOSS is a dramatic departure from the current proprietary software model. Since its source code is easily shared, FOSS is 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 self-sustaining community of diverse individuals who have contributed their time, energy and knowledge to the creation of a global resource. Often vocal, strongly opinionated and technically savvy, this community can powerfully contribute 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 FOSS is anti-copyright. In reality, almost all FOSS applications are copyrighted by their respective authors (except for the few that are deliberately placed in the public domain by the authors). However, these authors have used liberal licenses that allow anyone to use their work with limited restrictions.

The copyright holders still have full rights to their work, including the right to sell their work or re-license 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: www.groklaw.net/article.php?story=2004040421042728

²Wheeler, D., "Why OSS/FS? Look at the Numbers!"; available from www.dwheeler.com/oss_fs_why.html.

It is not possible to adequately cover 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 from: www.iosn.net/downloads/foss_gov_primer.pdf.

THE STRATEGIC IMPORTANCE OF FOSS

Many countries are contemplating the promotion of FOSS via legislative, policy or government procurement methods.³ 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 different motives for 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 are:

Strategic Benefits

- ▶ Developing local capacity/industry
- ▶ Reducing imports/conserving foreign exchange
- ▶ Enhancing national security
- ▶ Reducing copyright infringements
- ▶ Enabling localization

Economic Benefits

- ▶ Increasing competition
- ▶ Reducing Total Cost of Ownership (TCO)
- ▶ Enhancing security
- ▶ Achieving vendor independence

Social Benefits

- ▶ Increasing access to information

Strategic Benefits

Developing Local Capacity/Industry

Developing local capacity falls into two main categories: the capacity to use and the capacity to develop software. Although interrelated, FOSS assists in their development in different ways.

Using Software

The local capacity to utilize ICTs is traditionally held back by the high cost of both hardware and software. The ability to participate in the information society and build local knowledge communities is severely constrained when the cost of a basic operating system is equivalent to several years' wages for the average citizen.⁴

FOSS alleviates this situation (though it can never totally eliminate it) in two ways: zero licensing cost and free redistribution. It is possible to obtain FOSS for almost any conceivable purpose without any

³ A listing of some of these countries and their efforts can be found in *Free/Open Source Software: A General Introduction*.

⁴ Ghosh, R. A., "License Fees and GDP Per Capita: The Case for Open Source in Developing Countries", *First Monday*, 28 November 2003; available from www.firstmonday.dk/issues/issue8_12/ghosh/.

licensing costs, thus reducing the cost barrier. FOSS is also freely redistributable, meaning that anyone can share software with a neighbour in need. Free (the F in FOSS) Software, in particular, holds this as one of its fundamental principles. Traditional proprietary software, no matter how low the costs, often cannot be freely redistributed in this manner.

Developing Software

A common problem faced by developing nations is the lack of development 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.⁵

- ▶ **Low barriers to entry:** FOSS, which encourages 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. On the other hand, existing proprietary software is often the result of years of building and refinement that a software company in a developing country has little chance of developing a competitive system and selling it.
- ▶ **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. Apart from the source code and software tools that FOSS provides, there are many technical manuals, guides and 'how-tos' provided in every FOSS distribution. This documentation is the equivalent of thousands of dollars of manuals 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, databases and other software tools for these languages. However, the Chinese language, although based on a radically different system from that of western languages, is decently supported in FOSS systems due to the involvement of Chinese-speaking FOSS developers.

Additionally, the current business models structured around FOSS are primarily based on services, rather than on products. 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 will thus reinvest its profits there.

Reducing Imports/Conserving 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, Adobe Photoshop, etc). Countries that have to license this software, other than the originating country, end up importing software licenses. The large cost of these licenses places an enormous burden on the financial resources of a developing nation, resources that can be used on other development needs. Fortunately, alternatives exist.

⁵Ghosh, R.A., Krieger, B., Glott, R., and Robles, G., "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 www.infonomics.nl/FLOSS/report/FLOSSFinal_2b.pdf.

FOSS, by the nature of its licensing terms, can be obtained at little or no cost, and therefore, saves a massive amount of foreign exchange. But this is not the only benefit. As noted in the European study, "Free/Libre and Open Source Software: Survey and Study":⁶

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".⁷ 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.⁸

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

Enhancing National Security

Proprietary software is normally distributed in binary format; therefore it is difficult to reverse-engineer and to understand exactly what a program does. Although the opaqueness of the binary format offers limited protection to the intellectual property of the software maker, it engenders mistrust and suspicion. Could there be hidden back-doors 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.⁹

Other recent cases include the acknowledgments by both Netgear¹⁰ and Cisco systems that there were secret user names and passwords hard-coded into certain models of their wireless routers.¹¹ The user could not be disabled, allowing anyone with 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.

There have been previous cases where vendors "fixed" such a vulnerability by merely changing the username/password combination and hoping that no one would find out about the new combination.

Mistrust of "black-boxed" foreign software that cannot be audited has been cited as one of China's reasons for its adoption of FOSS¹² and one of the reasons why other governments are considering FOSS.¹³

⁶Ghosh, R.A., Krieger, B., Glott, R., and Robles, G., "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 www.infonomics.nl/FLOSS/report/FLOSSFinal_2b.pdf.

⁷Data downloaded from www.bcb.gov.br/ftp/NotaEcon/NI200401sei.zip.

⁸Jayne, F. G. Jr, "Balance-of-Payments Constrained Economic Growth in Brazil", *Revista de Economia Política*, Vol. 23, No. 1, 2003; available from ideas.repec.org/p/cdp/texdis/td155.html.

⁹In 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 mistrust still lingers. There are many references to this incident, including on *CNN.com*, at: www.cnn.com/TECH/computing/9909/03/windows.nsa.02/

¹⁰Knienieder, T., "Netgear WG602 Accesspoint Vulnerability", *securityfocus.com*; available from www.securityfocus.com/archive/1/365069.

¹¹"Cisco Security Advisory: A Default Username and Password in WLSE and HSE Devices", *cisco.com*; available from www.cisco.com/warp/public/707/cisco-sa-20040407-username.shtml.

¹²Loyola, R., "China: The Republic of Linux", program aired 13 August 2003; available from www.techtv.com/screensavers/linux/story/0,24330,3395670,00.html.

¹³Kettman, S., "Germany Denies Microsoft Ban", *Wired News*, 19 March 2001; available from www.wired.com/news/politics/0,1283,42502,00.html

Reducing Copyright Infringements

Copyright infringements in the form of unauthorized duplication of software (often called software piracy) are a problem in almost every country around the world. The Business Software Alliance estimates that unauthorized software copying in 2002 alone costed US\$13.08 billion. Even in developed nations where software is affordable in theory, software copyright infringement rates were as high as 24 percent in the United States and 35 percent in Europe. The rates in developing countries, where lower incomes make software far more expensive, are upwards of 90 percent.¹⁴

Copyright infringement and lax laws can and often do hurt a country in many ways. A country with a poor track record in copyright protection 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 copyrights and patents in a country. Bilateral and multilateral trading agreements with developed nations often involve requirements to reduce copyright infringements as well. A culture of copyright infringement hurts local software development, as there is less incentive for local software developers to create a local product.

Copyright infringement issues have been cited as a major motivator for Viet Nam's FOSS policy. A trade agreement with the United States signed in 2001 and Viet Nam's goal of joining the WTO by 2005 make it imperative that the copyright infringements in that country be greatly reduced.

Enabling Localization

Countries where English is not commonly spoken can be at a serious disadvantage when it comes to the uptake and dissemination of ICTs. If the country and language are not deemed to be commercially 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 becomes a preferred option 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 needed 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 98¹⁵ 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.

Economic Benefits

Increasing Competition

The software industry often has high barriers to entry, especially in mature software markets. Modern software applications have taken hundreds, if not thousands, of human years to research and develop. This is an investment that no fledgling software company can afford. Yet, once built, the marginal costs of producing additional units of software are miniscule. This grants significant first-mover advantages to initial market entrants and keeps competitors from entering the market at later stages, effectively limiting competition in the market to a few large, established organizations.

FOSS breaks down the barriers to entry by providing to software companies a feature-rich, high-quality base of software to build on. When utilizing FOSS, software companies compete on the services and additional innovations that they add to the existing FOSS base, rather than reaping financial benefits from research and development performed many years ago. Depending on the FOSS license and corporate business model, these benefits often find their way back into the FOSS base, resulting in a larger base of software for new entrants to build on. New entrants into the software industry do not need to spend

¹⁴Roberts, P., "Software Piracy Declines 10 Percent", *Infoworld.com*, 3 June 2003 ; available from www.infoworld.com/article/03/06/03/HNpiracydecline_1.html?security.

¹⁵Walsh, M. W., "Microsoft in War of Words", *Los Angeles Times*; www.tungutaekni.is/yomis_frodleikur/war_of_words.html.

Government institutions that have reported significant cost savings to date have been few and mostly in relatively developed countries. For example, the city of Largo in the United States has reported savings of over US\$1 million a year, with the city running its IT infrastructure on a budget that is only 40 percent the size of comparable cities.²⁴ 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 US\$730 million.²⁵

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 competition.
- ▶ 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: www.statskontoret.se/pdf/200308eng.pdf

Enhancing 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. The following studies indicate that FOSS systems are superior to proprietary systems in this respect:

- ▶ A study found that FOSS applications fixed problems faster than their proprietary counterparts. The contrast is much more apparent when the application is extensively used by the FOSS developer.²⁶
- ▶ A comparative study of FOSS code and proprietary equivalents found that FOSS had lower defect densities. The FOSS database MySQL had one sixth of the defects found in proprietary equivalents.
- ▶ The study was conducted by Reasoning, a company that produces automated software inspection and auditing tools.²⁷
- ▶ “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.”²⁸

FOSS distributions usually include a wide variety of security tools that allow a competent system

²³ Lemos, R., “Merrill Lynch: Linux saves money”, *CNet News.com*, 7 June 2003; available from news.com.com/2100-1016_3-1014287.html?tag=fd_top.

²⁴ Miller, R., “Largo loves Linux more than ever”, *Newsforge.com*, 9 December 2002; available from newsforge.com/article.pl?sid=02/12/04/2346215&mode=thread&tid=19.

²⁵ Glover, T., “Microsoft losing market grip as rivals go on the offensive”, *Scotland on Sunday*, 18 May 2002; available from www.scotlandonsunday.com/business.cfm?id=562032003.

²⁶ Kuan, J., “How to Succeed in Business with Open Source Software”, *Infonomics.nl*; available from www.infonomics.nl/FLOSS/workshop/papers/kuan.htm.

²⁷ “Reasoning Study Reveals Code Quality of MySQL Open Source Database Ranks Higher than Commercial Equivalents”; available from www.reasoning.com/newsevents/pr/12_15_03.html.

²⁸ Luening, E., “Windows Users Pay for Hacker Insurance”, *CNet News.com*, 29 May 2001; available from news.com.com/2100-1001-258392.html?legacy=cnet.

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 to their proprietary counterparts.²⁹

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.³⁰ China, Japan and South Korea also cite security as a major driving force in their FOSS initiatives.

Security: United States of America

A report prepared by the MITRE Corporation for the Defense Information Systems Agency found that in the area of security:³¹

“Banning FOSS ... 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 cyber attack.”

Achieving Vendor Independence

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 need to do so.

The authors of the paper “Free/Libre and Open Source Software: Survey and Study”³² 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 who 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 became aware of the dangers of vendor dependency when the insolvency of a major supplier forced it to migrate systems to a FOSS platform.³³

²⁹ Forristal, S., “Vulnerability Assessment Scanners” *Network Computing*, 8 January 2001; available from www.networkcomputing.com/1201/1201f1b1.html.

³⁰ Ghosh, R.A., Krieger, B., Glott, R, and Robles, G., “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 www.infonomics.nl/FLOSS/report/FLOSSFinal_2b.pdf.

³¹ “Use of Free and Open Source Software in the U.S. Department of Defense”, MITRE Corporation, available from www.egovos.org/rawmedia_repository/588347ad_c97c_48b9_a63d_821cb0e8422d?document.pdf.

³² Ghosh, R.A., Krieger, B., Glott, R, and Robles, G., “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 www.infonomics.nl/FLOSS/report/FLOSSFinal_2b.pdf.

³³ Ibid.

Vendor Independence: United Kingdom

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

1. The UK Government will consider OSS solutions alongside proprietary ones in IT procurements. Contracts will be awarded on a value-for-money basis.
2. The UK Government will only use products that comply with interoperability requirements and that support open standards in all future IT developments.
3. The UK Government will seek to avoid lock-in to proprietary IT products and services.
4. The 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. The UK Government will explore further the possibilities of using OSS as the default exploitation route for Government-funded R&D software.

Source: 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 the Taiwan Province of China, the German city of Munich, the Australian state of New South Wales, Brazil and South Africa.

Social Benefits

Increasing Access to Information

Although the social value of FOSS is not highlighted by proponents from public and private sectors, it can be a passionate issue among the many advocates from civil society.

Software, especially FOSS, represents knowledge – the rules, procedures and methods of manipulating data. In today's knowledge society, knowledge drives not only productivity and economies but also shapes the future. Many experts and opinion leaders advocate that knowledge should be shared as widely and as freely as possible. Steve Weber argues:³⁴

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.

³⁴Weber, S., "Open Source in Developing Economies", Social Science Research Council; available from www.ssrc.org/programs/itic/publications/ITST_materials/webernote2.pdf.

WHY IS A FOSS POLICY NEEDED?

Under ideal circumstances, a specific policy favouring FOSS is not required. However, policy-makers have turned to FOSS policies in order to solve specific problems within their countries. Related to the benefits of FOSS as described earlier, some common indicators of these problems include:

- ▶ Local software economy dominated by monopolies
- ▶ Pressure from foreign trading partners to reduce copyright infringements
- ▶ Large foreign exchange outflows to import software
- ▶ Inadequate and stagnant local ICT capacity
- ▶ Lack of ICT tools in local language

The question has been raised before that if FOSS systems have 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 current computing environment.

Limited Marketing

Because FOSS systems are available to almost anyone, it is hard for any single company to claim ownership and thus gain competitive advantage from promoting FOSS. 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 limited commercial efforts to promote the benefits of FOSS.

FOSS being a relatively new development, many decision makers are unaware of the full benefits that it brings or the issues related to implementing it. The lack of publicity also makes it somewhat vulnerable to the Fear, Uncertainty and Doubt (FUD)³⁵ that may be produced by competing proprietary software vendors.

Non-commercial Benefits

Among the FOSS benefits described in the previous section, only three of 10 (Total Cost of Ownership, Vendor Independence and Security) were favourable to most commercial entities. Since most decisions on software purchases and implementation are based on the “best value” basis, 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: www.tekno.dk/pdf/projekter/p03_opensource_paper_english.pdf

³⁵ 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 that their staff already have the capability to support.

Additionally, a significant number of proprietary software integrates poorly with other software, be they proprietary or FOSS. This makes the cost of changing as well as introducing new infrastructure extremely expensive. In many cases, 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 commonly used to limit interoperability.

A FOSS policy formulated by a government, even one that simply makes FOSS a choice equal to proprietary software, could facilitate a wider uptake of FOSS and consequently allow the economy to take advantage of its unique benefits.

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.

Motivations and Needs Assessment

The first step in the policy formulation process is clearly identifying the motives and needs for a FOSS policy. As FOSS solves certain problems in a more efficient manner, clearly articulated motivations are a must for implementing FOSS policies, especially if they are backed at the highest level of government. Specific motivations would greatly affect implementation strategies. For instance, if the motivation is to reduce high cost imports by using affordable ICT solutions, then strategies are more likely to emphasize converting existing infrastructure to FOSS. If enhancing local capacity is the need, then the emphasis would be on use of appropriate ICT solutions for education.

Once the basic needs and motivations have been established, an assessment of the existing environment has to be conducted to determine the suitability of FOSS policies and their specific benefits to the society as a whole. 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 in fact 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, after having made the necessary adjustments. In particular, a FOSS policy should not be separate from the national ICT policy or ICT4D strategy if it exists, as the likelihood of overlap is great. Other policies/strategies that can also significantly affect FOSS policy include education, intellectual property and international trade. The specifics of the more common cross-sectoral concerns are covered in a later section.

Regional Context

Most regions in the world today have varying degrees of active FOSS initiatives and policies in progress. An assessment of regional efforts is important to ensure that FOSS policies will work in the regional context as well as to identify the areas of cooperation or common interests. For example, countries in the Asia-Pacific region should be aware of a regional effort by China, Japan, and South Korea to produce a common, regionally localized and customized FOSS system³⁶ and utilize this in their own countries. Malaysia, Taiwan, Thailand, and Viet Nam, have strong FOSS initiatives or formal policies encouraging

³⁶Myoung, S. E., "Korea, China, Japan Start Open-Source Collaboration", *CNETAsia*, 2 April 2004; available from asia.cnet.com/newstech/systems/0,39001153,39174180,00.htm

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 and interoperability standards. The European Union is mandating open standards in their inter-government communication and suggesting FOSS as a method for implementing this.³⁷ 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.

Establishing Goals and Targets

Once the benefits and suitability of a FOSS policy have been established, goals and targets of the policy need to be stated 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.

Measurable targets should be identified, 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 is “converting 10 percent of civil service infrastructure to FOSS”. Other possible targets are “50 percent of all ICT graduates are proficient in FOSS” or “95 percent of private sector organizations are aware of the benefits of FOSS”.

Policy Goals: Malaysia	
The 2005 targets listed in Malaysia’s Master Plan for implementing OSS in the public sector are listed below. Details available at: opensource.mampu.gov.my/	
Awareness	▶ 100% of CIOs and IT personnel are OSS literate
Re-skilling	▶ 60% of IT personnel are OSS trained ▶ 10% of IT personnel are OSS certified ▶ 20% of teachers responsible for school IT labs are OSS trained
Education	▶ 40% of institutions of higher education utilize OSS education and teaching tools
Procurement	▶ 60% of all new servers (hardware) procured are able to run open source operating systems
Implementation	▶ 20% of school IT labs have OSS installed (e.g., office productivity) ▶ 60% of web servers (software) use OSS ▶ 30% of office infrastructure (email, DNS, Proxy) use OSS ▶ 30% of desktop solutions (e.g., web browser, email reader) use OSS

Strategy Formulation

Once tentative targets have been set for the FOSS policy, strategies to achieve the goals of the policy need to be formulated. FOSS policy level strategies unfortunately do not have a long history of development, testing and implementation. Policy-makers, therefore, will have to be especially careful with implementation and monitoring issues.

The regional survey mentioned earlier, 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 coming up with strategies involves brainstorming sessions.

³⁷Williams, P., “Europe Picks Penguin to Link Government IT”, *VNUNet.com*, 18 July 2003; available from www.vnunet.com/News/1142411.

Brainstorming

Brainstorming is producing 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. In fact, participants (individuals or groups) are encouraged to come up with ideas that seem a bit odd. Participants are encouraged to break out of their standard methods of thinking, which allows the formulation of creative and out-of-the-box solutions.

At the end of the brainstorming session, ideas are evaluated and the best ideas are chosen. These ideas are then refined further, possibly via group discussions or even further brainstorming, until an optimal solution is found.

Brainstorming

Different parties can do brainstorming. 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 they are likely to have been promoting FOSS within the country for some time and would be aware of issues that policy-makers may not be familiar with.

After several rounds of brainstorming sessions, the strategies are evaluated, summarized and 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 (including corporations, academic institutions and developers), few FOSS policies are likely to succeed.

Policy-makers may 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 has created software that matches and even surpasses the software created by the finest corporations in the world.

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

- ▶ Round tables.
- ▶ Focus groups.
- ▶ Online consultations.

Round Tables

Round tables are gatherings (public or by invitation) of representatives of stakeholder groups. The FOSS community usually has no formal leader, there are usually 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.

In the round table sessions valuable insights and suggestions can sometimes be generated 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 public. Organizing round tables, however, can require much energy and resources, especially in a

geographically diverse country.

Focus Groups

While round tables cover a broad spectrum of the FOSS community and interests, focus groups tend to concentrate on narrower areas such as education, private sector or legal issues. The number of participants is smaller but they are focused on these specific fields. The discussions in these sessions, though narrower in scope, are often detailed and lengthier, tending to yield greater information.

Focus groups should be organized for the critical areas targeted by the policy, 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 of the various aspects of the policy.

However, the policy steering committee should be aware and especially careful of certain issues with respect to online discussions. The most obvious issue is that the availability of Internet access within the country often limits the participants of the consultation to only a small subsection of the population. For this reason, 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, which has valuable insights, past experiences and resources to contribute.

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

Policy-makers should expect some changes 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.

STRATEGIES

There are numerous strategies for achieving any conceivable policy goal. Each strategy has its own advantages and drawbacks. This section discusses considerations affecting strategy formulation, and some of the more commonly used strategies with their advantages and drawbacks.

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.

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, which involves large implementation and training costs. Although a number of proposals (legislative or otherwise) to this effect have been submitted, to date few have passed. The high costs and risks involved are the main deterrents in this approach.

Other countries have chosen the less painful route of mandating FOSS for all new procurement. More conservative approaches such as this are somewhat more common. Countries that mandate the change of only a proportion of the infrastructure over to FOSS include Brazil (80 percent of all systems),³⁸ South Korea (20-30 percent of all systems)³⁹ and Thailand.⁴⁰

This approach greatly promotes FOSS usage and capacity in the local economy. However, the criticism is that this is done 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.⁴¹ This does not mandate FOSS *per se* but it has a strong stimulating effect. The policy considers locally packaged FOSS systems as local software, even if the international FOSS community produces the majority of its components.

Preferring FOSS

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

³⁸"The Brazilian Public Sector to Choose Free Software", *PCLinuxOnline*, 2 June 2003; available from www.pclinuxonline.com/modules.php?name=News&file=article&sid=6879

³⁹Myung, S. E., "Korea Jettisons Windows for Linux", *silicon.com*, 1 October 2003; available from www.silicon.com/software/os/0,39024651,10006225,00.htm

⁴⁰"ICT Ministry Sets Linux Targets for government Sector", *Bangkok Post*, 23 June 2003.

⁴¹"China Blocks Foreign Software Use in Govt", *CNETAsia*, 18 August 2003; available from asia.cnet.com/newstech/applications/0,39001094,39146335,00.htm

This approach has the benefit of being easier and less risky to implement. It is also more flexible, allowing procurements to be decided on a case by case basis, taking 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 that proprietary software enjoy when there is an established proprietary system.

Preferring FOSS: South Africa

Sections 10.1 and 10.2 of South Africa's OSS strategy 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: 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 most effective ways in which software vendors lock in their users is the use of proprietary standards. FOSS systems are at a disadvantage in a mostly proprietary software environment due to the lack of interoperability. 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, this often requires modifying procedures and legacy documents that are still stored using proprietary standards.

The two areas often targeted by open standards advocates are documents and web standards. Text documents and spreadsheets are typically stored in proprietary formats and may not be retrievable without the proper proprietary software, thus impeding the free exchange of information. Proprietary,

Mandating Open Standards: Commonwealth of Massachusetts, USA

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: www.state.ma.us/itd/openstandards.htm

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 using non-standards compliant HTML tags that are only accessible using Internet Explorer even though it would take minimal effort to make these sites cross-platform.

In some cases, mandating open standards would initially preclude certain proprietary software vendors from participating until such time that they add proper support for open standards in 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 Zealand⁴² are among the governments supporting open standards.

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 were started at the state level in the United States but, to date, none have passed. 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.

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 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 to be 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 be a significant amount of work.

Incompatibilities: Even if a total migration to FOSS systems were contemplated, there will be a period of time when both FOSS and proprietary systems must coexist 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 (for example, text documents that are strangely formatted). But in other cases, data interchange may be quite complicated.

For these reasons, implementing a FOSS infrastructure or at least creating a FOSS-friendly environment

⁴²Brislen, P, "Microsoft Defends XML Patent, Offers Royalty-Free License", *Computerworld.com* New Zealand, 5 February 2004; available from computerworld.co.nz/news.nsf/NL/AB73832F819E0AF5CC256E3000C7BFE

is much simpler when the existing infrastructure is limited.

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

Policy and Legal Environment

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

Many countries are finding that they have to re-examine and often tweak their existing policies and laws to align them with their FOSS goals. Often, the changes are not just within the ICT sector but within other sectors as well, such as copyrights, patents, 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 support for FOSS from the highest levels of government.

Capacity Development

Many developing countries have noted that a serious shortage of FOSS capacity in their economies⁴³ holds back the implementation of major FOSS projects. 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, too many to be listed 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, Mongolia, Spain, 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 this 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.⁴⁴

Ensuring that curriculum is vendor neutral: Many ICT literacy and 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

⁴³ Expressed during the Free and Open Source Software Asia-Pacific conference held between 9-11 February in Kuala Lumpur, Malaysia. Summary findings and recommendations are available at www.iosn.net/downloads/fossap-report-final.pdf. A second conference in Africa expressed a similar need. The Idlelo conference was held on 11-15 January in Cape Town, South Africa. The final report is available at www.fossfa.net/tiki-download_file.php?fileId=3.

⁴⁴ Martyris, D., “Community -Government Partnerships and Open Source Technology for Low Cost IT Access in India – A Case Study”, July 2003; available from www.developmentgateway.com/node/133831/sdm/blob?pid=5474.

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 students learn flexible skills that can be easily transferred from one system to another. This does place an extra burden on both educators and students.

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 who has not learned broad, easily transferred concepts will become obsolete.

Scholarships/innovation awards: These are relatively small financial rewards given to students 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 within the formal education system. Such awards have been implemented in places such as India, but it is still too early to assess the long-term outcomes of such a programme.

Certification and Retraining Programmes

Most policies, especially those with short-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 programmes can either be part of continuing education classes taught in the evenings at colleges and universities or in specially mandated training centres. 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 programme 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. These 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 programme from scratch (which is quite a significant undertaking) would be to utilize one of the globally recognized certification programmes. Two of the notable certification programmes for GNU/Linux (which is only a subset of the entire FOSS movement) are the Linux Professional Institute's Certification (LPIC) and Red Hat's Red Hat Certified Engineer (RHCE) programme.

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 its examinations are relatively inexpensive (US\$100 per exam normally; under certain conditions, prices can be brought lower).

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 programme is unusual in that it covers both proprietary (Microsoft Windows and Office) and FOSS (GNU/Linux, OpenOffice.org) applications. Trainees are then able to compare and choose between the different systems.

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

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 of 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.

Other organizations offering certification programmes include CompTIA with their Linux+ certification and SuSE with their certified engineer programme.

FOSS Competency Centres

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

1. FOSS awareness raising and promotion.
2. Implementing pilot projects in e-government.
3. Documenting current best practices of FOSS usage.
4. Providing technical support for government agencies.
5. Providing training of both end-users and technicians.
6. Aggregating news and resources on FOSS.

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

FOSS Competency Centres: Malaysia	
Malaysia's Master Plan for implementing FOSS in the public sector includes the creation of an Open Source Competency Center (OSCC). The target activities are listed below. Further details available from: opensource.mampu.gov.my/	
Policies and Guidelines	▶ Formulate OSS policies, guidelines and standards to facilitate OSS implementation
Knowledge Bank	▶ Maintain Knowledge Bank for sharing of knowledge and experience
Promotion and Awareness	▶ Create greater awareness, promote OSS and the OSCC
Training and Certification	▶ Conduct and coordinate OSS training and certification programmes for public sector personnel
Technical Support	▶ Facilitate, coordinate, advise and assist Government agencies in OSS implementation
Research and Development	▶ Conduct, facilitate, coordinate and monitor OSS Research and Development programmes

Research

Some policies include a requirement that software produced by publicly funded research should be released under a FOSS license. The rationale is that the fruits of public funds should be freely enjoyed by the 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, made 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 VISTA⁴⁵ 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 created by the National Center for Biotechnology Information (NCBI).⁴⁶

In both cases, the primary beneficiaries are the citizens of the country itself, as adoption is normally quickest there. But the benefits have 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

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

While a large 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.

In many developing countries there is insufficient local capacity to properly support widespread FOSS usage (or ICT usage of any kind) throughout the economy. Thus, FOSS policies can and should be linked with the creation and development of Small and Medium-sized Enterprises (SMEs) to supply support services. In combination with a FOSS government procurement strategy, this can create a vibrant ecosystem with strong supply and demand.

There are several strategies that governments may employ to encourage the growth of FOSS among SMEs, as discussed below.

Awareness Raising

SMEs normally do not have access to the wealth of information resources that larger corporations have and they 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 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 ultimate winner of the contract.

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

⁴⁵ More information available on the Internet at www.worldvista.org/vista/index.html.

⁴⁶ The basic algorithms that are used in biotechnology research are based on NCBI's tools. Although some companies have re-implemented the algorithms, most use the existing algorithms and add value on top of it. Further information can be found at www.ncbi.nih.gov.

Tax Incentives

Besides directly creating demand via overt purchases, governments can increase the attractiveness of FOSS to SMEs by offering incentives such as tax breaks. Countries like Singapore have offered tax reductions to companies that use the GNU/Linux operating system.⁴⁷ The guaranteed cost saving makes FOSS systems more attractive to SMEs.

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 programmes for small businesses. An example is 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.⁴⁸ A modified version of such plans can be adopted to assist FOSS SMEs.

⁴⁷UNCTAD Secretariat, "e-Commerce and Development Report 2003", 2003, United Nations Conference on Trade and Development; available from r0.unctad.org/e-commerce/e-commerce_en/edr03_en.htm#downloads.

⁴⁸"Overview & History of the SBA"; available from www.sba.gov/aboutsba/history.html.

CROSS SECTORAL CONCERNS

Due to its unique nature, FOSS touches on more than just national ICT policy. Particularly in the area of Intellectual Property Rights (IPR) as well as 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.

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 this, software companies threatened by FOSS can utilize software and business process patents to stifle FOSS development.

The nature of FOSS development is open and cooperative, with many individuals and organizations involved in the process. These 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. In the United States, the absolute minimum for a successful patent hovers around \$1,025.⁴⁹ Most patent filers utilize the services of specialized patent attorneys and this greatly increases the costs to between 8,000 and US\$30,000 per patent. 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

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 when the US Supreme Court compelled 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 and common, with thousands of them being issued in the United States every year.

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

⁴⁹"How Much Does A Patent Cost?"; available from www.ipwatchdog.com/patent_cost.html.

adoption may be severely threatened.

Richard Stallman of the Free Software Foundation notes that patents are also a threat to small and medium-sized companies that develop or use software. The Foundation for a Free Information Infrastructure (FFII), a European grouping of over 1,000 small companies and 60,000 individuals, is fighting the introduction of software patents into the European Union and has collected an extensive array of resources arguing against software patents. One of the more telling quotations in their resources is attributed to Microsoft's Bill Gates in 1991:

If people had understood how patents would be granted when most of today's ideas were invented and had taken out patents, the industry would be at a complete standstill today. ... The solution is patenting as much as we can. A future startup with no patents of its own will be forced to pay whatever price the giants choose to impose. That price might be high. Established companies have an interest in excluding future competitors.

Full details can be found at the FFII site at: www.ffii.org/

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 some of the most popular FOSS licenses, including the Free Software Foundation's GNU General Public License (GNU GPL). The GNU GPL has a provision requiring that all derivative works of GNU GPL software must be released under the GNU GPL. The definition of derivative works is not consistent worldwide and 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 is 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 its 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 by using proprietary communication protocols and data standards. This would make it easier for vendors to lock their customers into their software solutions.

Consumer Rights

Overly aggressive consumer protection laws can also destroy the FOSS industry, particularly with regard to implied warranties. This is ironic since FOSS tends to give consumers far more rights than proprietary software does.

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, since software producers 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 many 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. Mandating such a warranty for all would seriously cripple the FOSS community as it would then be impossible for anyone other than large corporations to participate in FOSS development.

CONCLUSION

Free/Open Source Software is a proven and mature solution set that offers many opportunities to developing nations that are 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 jumpstart 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 difficult. National leaders must be able to make balanced and well-informed decisions pertaining to FOSS policies. Tony Stanco from the Center of Open Source and Government highlights several strong points in the South African Open Source Strategy:⁵⁰

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 Programme.** 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 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 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 benefits to be gained are enormous and policy-makers need to be well informed to make optimal choices so that each nation can fully realize the benefits that FOSS can bring to each.

⁵⁰ Stanco, T., “EGOVOS Endorses South Africa's Open Source Strategy”, *Newsforge.com*, 18 June 2003; available from www.newsforge.com/business/03/06/17/1655215.shtml?tid=19

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 following 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 neighbour.
- ▶ 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 www.fsf.org/philosophy/free-sw.html.

GNU

GNU 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.

GNU GPL

The GNU General Public License (GNU 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 possesses 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 becomes meaningful and comprehensible to the local user of the product. Localization needs to address linguistic, content, cultural and technical issues.

NSA

National Security Agency (NSA) is the United States of America’s cryptologic organization. It coordinates, directs, and performs highly specialized activities to protect US 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 centres of foreign language analysis and research within the United States government.

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

Open Source Software

Open Source Software does not only mean access to the source code. To qualify as Open Source Software, a software must have distribution terms that comply with the following conditions:

- ▶ 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 endeavour.
- ▶ 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: 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. This is 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 "translate" into machine code (also referred to as binary code, executable code or object code). To modify software, the source code must be available, as the machine code is not human-readable.

TCO

Total Cost of Ownership (TOC) includes all the costs involved in a technology or business solution. In addition to the initial investment cost, TCO includes training, maintenance, support, replacement costs, and the like. In the case of software, the TCO should include the initial cost of the software; upgradation cost; and maintenance, support and training costs.

USEFUL RESOURCES

Portals Focused on FOSS and Government

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

The European Commission's Open Source Observatory: 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: www.sdn.undp.org/perl/news/articles.pl?do=browse&categories=10

The International Open Source Network (IOSN): www.iosn.net

Policy Related Documents

United Kingdom's Open Source Software policy, as issued by the Office of Government Commerce: 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: www.infonomics.nl/FLOSS/report/

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

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

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

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

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

ABOUT THE AUTHOR

Kenneth Wong is a Free/Open Source Software advocate with a strong interest in utilizing FOSS to widen the reach of ICT in developing communities worldwide. Ken is currently the head of software development projects at Exosoft, a FOSS focused software development firm in Beijing, China. Ken has been making a living from FOSS throughout his entire professional career. He has BBA and MBA degrees from the University of Hawaii and a MS degree in computer science from Boston University. Ken can be reached by email at ken@exosoft.com.

APDIP

The Asia-Pacific Development Information Programme (APDIP) is an initiative of the United Nations Development Programme (UNDP) that aims to promote the development and application of Information and Communication Technologies (ICTs) for poverty alleviation and sustainable human development in the Asia-Pacific region. It does so through three core programme areas, namely, Policy Development and Dialogue; Access; and Content Development and Knowledge Management.

In collaboration with national governments, APDIP seeks to assist national and regional institutions in the Asia-Pacific through activities that involve awareness-raising and advocacy, building capacities, promoting ICT policies and dialogue, promoting equitable access to tools and technologies, knowledge sharing, and networking. Strategic public-private sector partnerships and opportunities for technical cooperation among developing countries (TCDC) are APDIP's key building blocks in implementing each programme activity.

www.apdip.net

IOSN

The International Open Source Network (IOSN) is an initiative of UNDP's Asia-Pacific Development Information Programme (APDIP). Its overall objective is to serve as a Center of Excellence and a Clearinghouse for information on Free and Open Source Software (FOSS) in the Asia-Pacific region. In order to accelerate the world-wide adoption of FOSS, IOSN seeks to raise awareness, facilitate networking of people, strengthen capacities, and conduct R&D.

The beneficiaries of IOSN are governments, IT professionals, software developers, the FOSS R&D community, academics and the NGO community. IOSN serves as a resource centre to help policy-makers and decision-makers in the public sector, educational institutions, businesses and others develop policies and plans for the use of FOSS in their respective organizations. Much of IOSN's activities are undertaken online and the IOSN portal has been developed for this purpose and serves as a comprehensive online resource center on FOSS. The IOSN portal also provides a means for the FOSS community in the region to contribute to its efforts and to interact with each other.

www.iosn.net

Also available from UNDP's Asia-Pacific Development
Information Programme

e-Primers on Free/Open Source Software

1. Free/Open Source Software–A General Introduction
2. Free/Open Source Software–Education
3. Free/Open Source Software–Localisation

www.iosn.net

e-Primers for the Information Economy, Society and Polity:

1. The Information Age
2. Nets, Webs and the Information Infrastructure
3. e-Commerce and e-Business
4. Legal and Regulatory Issues in the Information Economy
5. e-Government
6. ICT in Education
7. Genes, Technology and Policy
8. Information and Communication Technologies for Poverty Alleviation

www.eprimers.org



International Open Source Network

An Initiative of the
UNDP's Asia-Pacific
Development Information
Programme and supported
by the International
Development Research
Centre, Canada

Asia-Pacific Development Information Programme

www.apdip.net

United Nations Service Building
3rd Floor, Rajdamnern Nok Avenue,
Bangkok 10200 ,
Thailand

Tel: +66 (0) 2288 1234, (0) 2288 2129
Fax: +66 (0) 2288 3032