

Open Source Adoption in Indonesia Enabling the Culturability Approach in Developing Countries

I Made Wiryana¹, Ernianti Hasibuan¹, I Wayan Simri Wicaksana¹, and

Gunadarma University, Jl. Margonda Raya 100, Depok, Indonesia, phone +62 21 7888 1112, fax: +62 21 787 2829
{mwiryana,ernianti,iwayan}@staff.gunadarma.ac.id

Summary. The shift from system oriented to human context oriented design causes system development process is not only dictated by developers. Culturability, the merging of culture and usability, has implications for system design. Usability must be redefined in terms of a cultural context, as what is user friendly for one culture can be vastly different for another culture. However, customization to satisfy the culturability consideration sometimes may not be established due to the license limitation.

Developing countries always face software piracy problem due to high cost of software license. The use of proprietary and closed software also create some hidden cost and hidden problems, such as the hardware upgrade, file format, the software piracy, and in some specific cases, the fact that the program code cannot be read leaves doubts regarding the reliability, particularly in terms of confidentiality. Free and Open Source Software (FOSS) in Indonesia has been introduced since 1995 and now slowly is adopted by various government bodies, including the Information System of President of Republic of Indonesia. After Tsunami, the Aceh local government persuades to exploit FOSS to accelerate ICT development. Openness and flexibility for providing a secure system and possibility to provide a sustainable solution as well as independent in providing solution for local problem are also factor being considered in adopting the Open Source solution.

This paper will descriptively describe the process and approaches that has been applied in promoting the FOSS in Indonesia, as well as the adoption process. The first example is the progress of localization projects in Indonesia. Secondly, the progress of GNU/Linux distro development in Indonesia. The third is the adoption of FOSS in Information System of President of Republic of Indonesia. The proliferation of distro development in Indonesia shows the wide adoption of Open Source as well as the availability of human resource for providing the open source solutions.

Key words: culture, developing model, multi language, operating system

12.1 From machine to context oriented system design

The shift from system oriented to human context oriented design causes new considerations in choosing the appropriate system. A machine oriented design is mostly a deterministic system based on automated data processing. System is designed with deterministic behaviour and definite boundary. Role of user is only described as behaviour at the human-computer interface and ignored otherwise. Mismatches between technology and user preferences usually is interpreted as the failure of users to understand or learn a well-working technology. Applying human context approach, system development process is not only dictated by developers. Users play more active and important role. User acceptance plays more important role in evaluating the quality of system. Furthermore, user preferences are determined by the culture and the local context. Software must fit with this cultural and local context, without considering the cultural aspects some software products fail to satisfy the users [7]. Cultural aspect plays important role as critical success factor for implementation of ICT [12]. Culturability, the merging of culture and usability, has implications for system design. Usability must be redefined in terms of a cultural context, as what is user friendly for one culture can be vastly different for another culture. Many of the icons, metaphors, shapes, colors of text and background, frame/text locations on screen, etc. employed in systems design are relevant to the culture of origin of the software.

However, most User Centered Design (UDC) approaches are still focusing on how human interacts with computers, rather than looking the other way around, how the technology can be shaped to support enrichment of human skill and socially useful product. The cultural model deals will considering the issues how strategies, attitudes, habits and norm are influencing the use of artefact. Human context approach means accepting that any social entity has its own history

As culture becomes critical issues, designers have begun to realize the role of culture in design and to develop the method and process of applying cultural factors on design. However, the customization of software sometimes may not be performed due to the license limitations, especially for proprietary software. The Open Source license, enables the customization, can be performed freely without a complicated legal process. Developers in developing countries may develop a specific customization to fulfill the local needs without asking the permission of a company. Therefore, FOSS enables the possibility to customize, redesign or re-implement system according to culturability, as well as for doing research in culturability in developing countries. Without open source, major methods of cultural design were limited only in pencil and paper survey that requires significant time, effort, and cost. With FOSS researchers can choose a mature programs and redesign It according to culturability particular users.

Localization (providing user interface to other languages) is a basic effort in accommodating users who has different culture than the original developer.

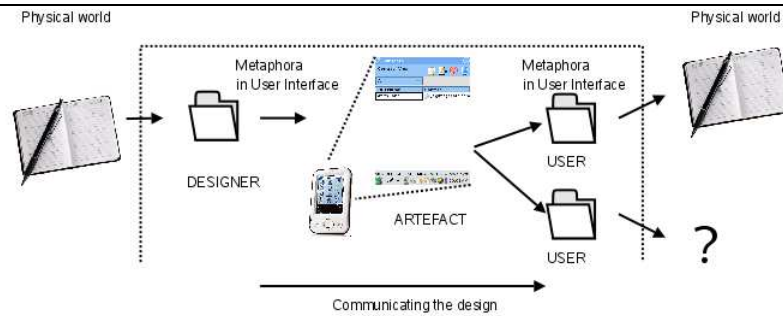


Fig. 12.1. Mapping to unknown object

However, only translating the GUI into the user language sometimes is not enough. Understanding the user context is an important aspect in customizing or designing a system. As shown in Fig. 12.1, an icon or menu for "Address book" cannot be translated easily into Indonesian as "Buku alamat". For particular users they do not know the real object such as Address Book, because they never experience or have that real object.

Language is used to describe and report the reality. However, language is not only content but also provide a context and way to re-contextualize content [1]. Therefore, only translating into their language without considering context does not solve the metaphor problem. We have to redesign the user interface to bring the concept of real object such as Address Book to an understandable object. Some user preferences for software also influence by the culture, the Open Source enable the local developers to perform more than translating the menu to accommodate this culture preference.

12.2 Localization Project in Indonesia

The Indonesia language is a normative form of the Malay language, an Austronesian (or Malayo-Polynesian) language which has been used as the lingua franca in the Indonesian archipelago for centuries and was elevated to the status as an official language of the Republic of Indonesia by the Indonesian Declaration of Independence in 1945, drawing inspiration from the *Sumpah Pemuda* (Youth's Oath) event in 1928. Indonesia language is used by approximately 200 million people. It is used by some people in East Timor, Netherlands, Philippine, Brunei Darussalam, Singapore, Southern Thailand, Saudi Arabia, Malaysia and Suriname.

Basically, the Indonesian language is relatively simple because there are no case of gender or definite article, no conjugation, no future or past tense, and no even a verb "to be". On the other hand, this simplicity could also bring complexity to speaker because lack of strict rules make difficult to write Indonesia text correctly. Furthermore, the Indonesian language is very con-

textual which means that the preference of words shall be accustomed with the purpose of articles as well as the audience. Over the years, the Indonesian vocabularies are also influenced and expanded by several languages such as Dutch, Portuguese, Hindi or Sanskrit, Arabic, English and Javanese, and other local languages in Indonesia, mostly through trade contacts and international media. The use of computer also influences the adoption of many English computer terminologies into Indonesian languages.

The localization effort which has been done in Indonesia firstly performed by Prof Dr. Dali. S. Naga, in 1984. This project translated the BASIC command into Indonesia, and it was known as KILANG[3]. However, this project did not take the usability into their translation consideration. Nowadays, some software vendors have tried to provide the menu also in Indonesia language. The vendor such as Nokia, Ericsson have already provided the menu in Indonesia language in their products.

In 1996 GNU/Linux was getting more attention in Indonesia. more people want to try and to install this operating system, as an alternative operating system. However, there are still exists many problems for disseminating the Linux distribution for users in Indonesia. One of the biggest obstacles is the language. Most of Linux documentations or user interface are not not written in Indonesia, There were many Indonesian developers who are working in translating the Linux HOW-TO document and the other related document to Indonesia language. However, based on our observation, users in Indonesia start to install and use soft ware directly without reading any documentation. They read the manual or the documentation after facing a problem. Thus, the translation of the installation and administration of the Linux Distribution will provide a big assistance for the new users. Our translation effort has objectives :

- To provide a FOSS system which has an installation or administration in Indonesia language, including the help system.
- To promote FOSS to the public in Indonesia through the Indonesia distribution.

To start the localization effort, in 1997 a group of lecturers and students in Gunadarma University was formed to translate the SuSE Linux distribution. SuSE 5.3 was chosen because is the most complete distribution and has complete documentations. It will be a big help for the user in Indonesia, due to the lack of Internet bandwidth. SuSe also had a nice system management program called YaST. Getting the source file and submitting the result of translation was not a big problems.

Instead of trying to use the "correct Indonesia terminology" for translation we use "more familiar terminology". There is an on-going effort performed by *Pusat Pengembangan Bahasa Indonesia*, to translate the scientific and engineering terminologies into Indonesia terminologies. However, we found that many translated word are not popular for the users. If we use this correct

terminology, the user will get confused and will not be able or more difficult to understand the sentence.

Before performing the translation we had to determine the "audience" or users. The type of users determine the strategy of the terminology selection, as well as the narration style of the sentences. Furthermore, it will determines how we choose from the translated terminology of the original terminology. For this first project, we decided as our target are the students, or the people in government institution. As the consequences, we assumed, that they have been familiar with the English or adapted terminologies such as : diskette, *konfigurasi* (configuration), boot, etc. We want to provide a smooth transition from the users who have only MS-Windows background. It is important since most of this typical users do not have any idea about these following terminologies, *mount*, *file device*, *boot manager* etc.

For example we will not try to translate "window" to "*jendela*". Users will understand faster and easier if we use "window" rather than "*jendela*". We are aware that it is not the correct way in translating a terminology into Indonesia language. However, we have a reason based on the usability of the result. We consider more about the usability of the document than the correctness of the translation from linguistic point of view. Many translated books in Indonesia are useless. It is due the "awkwardness" of the translation. They tend to use the "new" terminology which are correct from the Indonesia grammatical point of view, but this word is very "strange" to the normal user.

For example:

Mouse is a device which requires a device driver.

Many authors try to translate this English sentence into Indonesian :

Tetikus suatu perangkat yang membutuhkan sebuah penggerak perangkat.

However, many computer users who have been familiar with term "mouse", "device drive" will be hard to understand when they read the translated text. Thus, for users, the translated version is useless, because it is hard to understand. They tend to read the English version again, because it is easier to understand.

Based on this evaluation, we kept to balance between the familiarity of a terminology and the correctness of a terminology from the linguistics point of view. There is always a trade off in this consideration. To tackle this problem sometimes we used a new term but with a more familiar term in the bracket. For example : *mount*, *load*

These words are translated as :

mengaitkan (mount), *memuatkan* (load)

After 3 months of part time work the translation are finished and was part of SuSE 5.3 distribution. Thus, at that time SuSE has installation menu in Indonesia. We also continued by translating the man/info pages. The translation of KDE started on April 1999 but it was so slow. During this first attempt we found that the project management model for translation is an area which has not been explored thoroughly. This project is the first project for us which

has to be coordinated and managed via Internet. However, this project has shown a very promising result that we can apply to the next projects.

In deciding a translated terminologies as new terminology we faced some some cases :

- The original terminologies are used and not being translated. In some cases, the English terminologies are widely used by Indonesian user, for example, *modem*, *mouse* etc. We did not attempt to translate them into Indonesian terminologies.
- The original terminologies are adapted. Some English terminologies are adapted directly, only how they are written is different. It appears in the case of *device* (*divais*).
- The terminologies are translated. Some terminologies are possible to be translated into Indonesian language. Furthermore the translated terminologies are popular enough to the user. For example *berkas* (*file*).
- Introducing a new terminologies.
- Some terminologies which cannot be translated, such as *default*.

12.3 Distro development in Indonesia

Distro (distribution) is a collection of kernel, libraries, application and utilities that make the installation and uses of Linux easier for the normal users. Customizing distro or remastering distro usually is performed due to the specific requirements of the users, such as the localizations of menu and installation process. According to Computer Literacy Survey in 2001 from Agency for Assessment and application of Technology (BPPT), Indonesia users still needs the GUI which is written in Indonesia language. Other programs such as dialer for the local Internet Service Provider also a local specific requirement.

Distro plays important role in introducing Open Source widely, because it makes users can easier to install and used the new system. Since the Internet connection in Indonesia has not been in high capacity, Linux distribution also plays role as knowledge delivery media. Thus the availability the system in one media (CD-DVD) is very useful for users. Thus distro in Indonesia has to include the documentation in Indonesian. The first distro which was developed in Indonesia was Trustix Merdeka [5]. This distro was developed by Indonesian developer who is working with Trustix, a Linux company in Norway.

Translation for a complete distro is more complex than translating a program such as a part of SuSE installation, because some works should be translated such as :

- **Graphical User interface components**, such as menu, but tons, title of windows. It can be done by translating the contents of PO files. Some terminologies should be translated consistently.

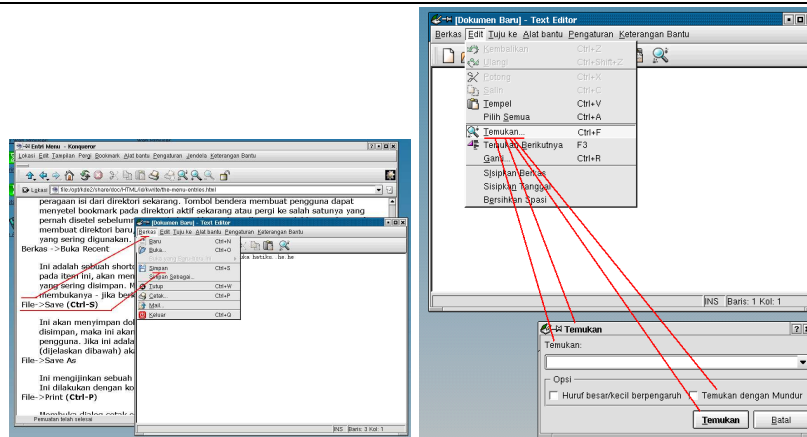


Fig. 12.2. Consistency problem

- **Warning message**, such as error and warning messages. The text should be translated according to context. By ignoring the text the user cannot understand the meaning of the text.
- **Online help text**. There are some formats which are used for the online help such as troff, info, DocBook or just basic HTML. Therefore, before starting the translation works, we should be familiar with those formats.
- **Documentation**. Fortunately, the open format which are used in Open Source programs, make them easier to be transform into other format, for example from DocBook to HTML, or DocBook to $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$, or troff to HTML.

In performing translation some consideration should be taken into account, such as :

- **Usable translation**. The translation is not only correct according to grammar and guidance for Indonesia language. . We try to avoid the use of a new terminology which is very “awkward” for typical users. Some English terminologies has been used widely, although there is translated version of the terminologies.
- **Consistent translation**. The files that should be translated are spread in hundreds of files. For example, “File” or “Find” should be translated into the same word for all files or program components. It should be avoided the use of different translated terminologies for a same word.
- **Consistency between the menu in the program, online help and manual as well as consistency among programs in the distro**. As shown in Fig. 12.2, some explanations in online help refer to a GUI component. For example “Push the Cancel button”, should be translated to “Tekan tombol Batalkan”. Some GUI components in a programs also refers to other GUI components in other programs. In a distro there are

more than 100 programs and about 1000 files. We have to keep it consistent among the difference group of translators. Therefore we decided to start with Trustix Merdeka which has a translated version of some components KDE, and building a special tools which combine the translated phrases into a database. This tools will help to establish rough translation, and to keep consistency between components

- **Size of translated text with the GUI components.** In many cases, the translated text in Indonesia are longer than the English version. so we change the program itself or we can just modify the translated text. For example "Help" will be translated as "Bantuan" or "Keterangan bantu". Therefore we have to developed a tools which calculate the length of text before translating it. Availability the source code is a big help.
- **The accelerator keys.** For example in the English version "New" has accelerator key `Ctrl-N`, the Indonesian version is "*Baru*", However if we keep the accelerator key as `Ctrl-N` (there is no relation with "*Baru*", then we have to change into `Ctrl+B`). Furthermore, we have to check there is no clash of accelerator key.

The work in developing WinBI was about three months and there were 30 translators and 3 developer who are involved in the development. Most of them are students and staffs of Gunadarma University. The works was started by using Trustix Merdeka as the starting point, since the Trustix Merdeka has already many component which has been translated. Therefore, before starting the translation project we evaluated the Trustix Merdeka. The distro development was performed by different groups such as translator for PO and DocBook files, programmers, graphic designer, and evaluators.

The development is performed as shown by Fig. 12.3.

- The first PO translator team, translated the PO files which has not been translated in Trustix Merdeka. The second team evaluated the translated PO files in Trustix Merdeka and build the phrase table in order to check the consistency of translation. This table are used in the translation process. To edit the PO files we used KBabel.
- There was no Docbook files which had been translated, thus the translator can start to translate it. DocBook is an open format, it is easy for the student who has been familiar with it to understand the format and build the tools that can make the translation faster. The English terminologies were not translated, because we have to wait the final version from the GUI Component translation.
- In the installer there are some PO files for menu and online help of installer. These files were translated by the translator team and DocBook. Graphic designer creates the icons, pictures which are used during the installation process as well as design the layout and the cover for manual books. \LaTeX is chosen as format to prepare the publication in order to keep the consistency between document and make the workflow of document production faster. The programmer develop the installation program

12.3. DISTRO DEVELOPMENT IN INDONESIA

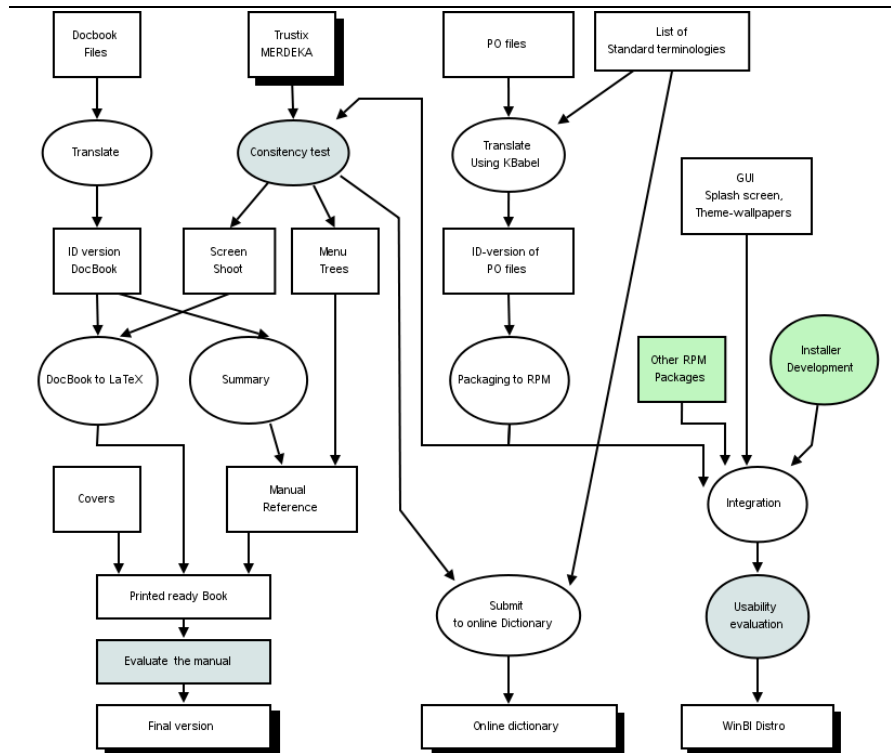


Fig. 12.3. Translation process

by modifying an available installation program. Paper based prototype are employed to evaluated the installation steps.

- After finishing translation process, consistency check are performed for all PO files. Each phrase will be collected and store into database. By comparing with the guideline of Decree of President 02/2001, and the other translated terminologies, we evaluated the translation. By using the integrated table, we can change a phrase and the other component of GUI which has this phrase will be affected as well. We also have to consider the length of text due to the size of GUI.
- After deciding the suitable translated terminologies, we checked again trough the whole PO files, and we performed "search -replace" in online Help document, because the English terminologies has not been translated. This process is performed for all PO files and all DocBook files. This is the second cycle of translation process.
- After that screen-shots are taken and will be used in the document and online Help. To make the checking process easier we built the menu trees automatically using script which used GraphViz [<http://www.graphviz>].

org]. This menu trees assist developers, whenever they want to refine the translation works.

- After finishing all translated, the result will be packaged into RPM package, because the WinBI using RPM as the package manager. The installer and the init script will be integrated into the distro and the iso image is generated. After the ISO image is ready, we performed the usability test according to the testing form. Our focus is on the translation aspects of GUI.
- For the online help we convert the DocBook using db2Latex and than processing it into the printed ready format by applying the class that has been prepared. By using this workflow, which uses the same source files we can keep the printed book, online help, and the menu are consistent. To produce the online help in the web, the DocBook files were translated directly to HTML. This process was easy due to the use of Open Format, such as DocBook.

Therefore, the results of this projects was not only the WinBI, but the PO files which can be used for other translation project or other Distro. It also submitted to the KDE as Indonesia localization. The translated phrase collection are also available for public in a form of parallel corpus.

12.4 Sustainability

There are some localizations efforts in Indonesia which were performed for the proprietary products. Such as the translation of MS DOS 3.3. However, the result was never delivered to public. This is the main difference with the localization effort in Open Source project. Public can use for different purpose the result of works. As shown in Fig. 12.4, the open source project has a greater change to be sustain. The development model of Open Source project, makes the other groups can continue the project without the involvement of the initiator. For example in the distro development of Indonesia :

- Trustix merdeka evolved into WinBI. Than it is continued by other groups and becoming Komura (Komputer Murah - Cheap Computer) and also RimbaLinux.
- BlankOn Linux [11] which has some packages from RimbaLinux now is being continued by other groups using different distro as basis. Now BlankOn switches using Ubuntu instead of Fedora.
- The other nice features is the translation results or other products of each project can be used freely by other developer, such as other distro developer or other translator. The OSS development model can accelerate the bootstrap phase during the development.

The proliferation of distro development in Indonesia shows the wide adoption of Open Source as well as the availability of human resource for providing the open source solutions.

12.5. WEB SITE OF THE PRESIDENT OF REPUBLIC OF INDONESIA

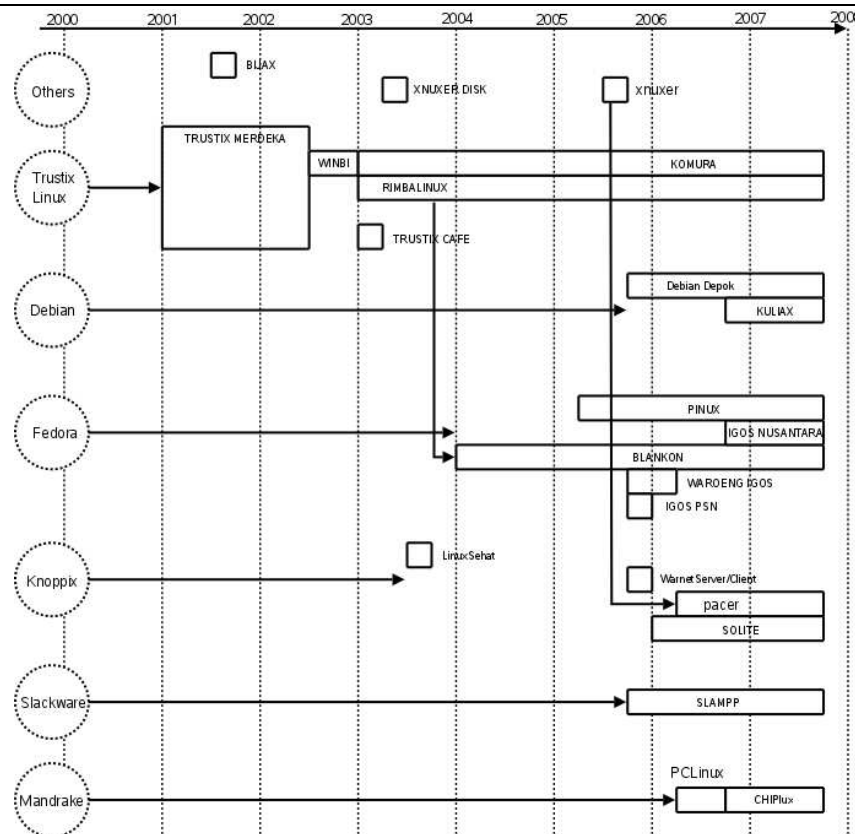


Fig. 12.4. Evolution of distro development in Indonesia

12.5 Web site of the President of Republic of Indonesia

On November 2005, President of Republic of Indonesia decided to develop the official web site. A developers team was formed to define the requirement and develop the system. Performance, reliability and security of system are the main consideration. To achieve the performance goal some strategies are employed such as the make use of web cluster, the dominant static page which are generated on demand from the dynamic content, as well as the use of cache technology. To reach the reliability goals, we employ cluster system with fail-over and live migration capability, mirroring system for disaster recovery.

Security is a big consideration for this web site, because this web site publish the official documents of President of Republic of Indonesia. An attack which changes the content will result big impact to the public. To reach the security goal some strategy are implemented such as : layering structuring for accessing the system, VPN and Firewall as basic requirement, integrity control, prevention of attack of the dynamic sites such as Cross Site Scripting

12.5. WEB SITE OF THE PRESIDENT OF REPUBLIC OF INDONESIA

(XSS) and Sql Injection, continuous monitoring and alert. Furthermore, to strengthen security of system we applied non monoculture system and we do not implement security by obscurity.

In developing the web content, we do not consider only a particular popular browser such as Internet Explorer. Different type of client such as text based browser or mobile browser also are considered. We try to adhere the open standard. As the document format used in the system, we decided not to be locked into a single vendor. Sustainability of system is the main consideration.

The first prototype with the actual hardware should be demonstrated in front of President of Republic of Indonesia and his staffs at the end of December 2005. The web site should content many type of file, has the facility to audio and video streaming, and support the notification via short messaging system (SMS). There is also specific security requirement, for example the content providers of the web site should be able to access the back end system securely everywhere such as in Internet Cafe. Thus the isolation and integration control is a requirement to protect from tempering. There was only 1 month to design, prepare, hardware acquisition and deploy the system. The process of the development follows the agile principles.

1. Developers gather the functional and non functional as well as technical and non technical requirements by sitting closely with the users. We employ participatory design and as result we listed requirements and customizations that we need according to local needs. We employs the prototype based design during the requirement phase.
2. After the technical characteristics being defined we request to the proprietary vendor to submit their budget and possible solutions using the proprietary operating system. At the same time, the same solution but using the open source solution was also developed by developer team. Most of the developers are familiar with proprietary and open source solutions.
3. After considering the security, the flexibility as well as the cost the proprietary solution cannot satisfy the requirement. Proprietary solution cannot satisfy the requirement for virtualization as well as live migration. Also the isolation for integrity control of client is weak area of the proprietary solution. Some additional cost should also be considered due to additional security features which are not built in proprietary operating system, such as firewall, intrusion detection, integrity control, anti virus. Other weak area for is the log managements. Since the proprietary solution is too "closed" we cannot change it as flexible as the open source to suit our needs.
4. We decided to have to develop the system based on the Open Source solution. We employed several Linux distributions and many Open Source programs. The flexibility of Open Source and the choice make it easier to choose and benchmarking the solution. For example for web server, we can choose different web server such as Apache[<http://www.apache.org>], Lighttpd [<http://www.lighttpd.net>], Yaws[<http://yaws.hyper.org>]

12.5. WEB SITE OF THE PRESIDENT OF REPUBLIC OF INDONESIA

etc. For the virtualization we can choose between OpenVZ [<http://www.openvz.org>] or Xen [<http://www.xen-source.com>] and Qemu [<http://fabrice.bellard.free.fr/qemu/>]. The flexibility of Open Source solution enabled us to create a special system for log monitoring and management based on available Open Source solution such as Nagios [<http://www.nagios.org>], Argus [<http://www.qosient.com/argus/>], Awstats [<http://awstats.sourceforge.net/>], Snort [<http://www.snort.org>] etc.

On 14th of February 2006 the web site was officially launched. The system survived from the big hit as well as attacks on the first week. After operating the website 1 years some interesting results in comparing the used of open source and proprietary software can be demonstrated in this paper. We only presents two data sets which are related with the spread of the Open Source usage in Indonesia. Since most of the visitors come from the Indonesia [8], the data can represent the user in Indonesia.

- **Browser used by visitors.** As shown in Table 12.1 the IE users are about 65% of visitors. Many users in Indonesia only used browser which is included in their operating system, to download other browser via Internet is still big problem. However the number of users who used IE is decreasing. On the other hand, the open source browser such as Gecko based (Firefox, Galeon etc), Konqueror is going up. Although users have to do extra works to download or install the open source browser, but they expect the benefit for doing it, such as security. It shows the increasing the acceptance of open source as alternative solution. Therefore, it can be concluded that there is a positive trend in the use of Open Source browser in Indonesia.

Table 12.1. Browser statistic of visitors

Month	Browser agent (%)					
	IE	Gecko	Opera	Safari	Konqueror	Others
2006-02	65.626	19.882	7.068	0.427	0.152	6.845
2006-03	64.449	17.135	7.774	0.315	0.244	10.083
2006-04	63.528	16.434	7.435	0.411	0.188	12.004
2006-05	64.931	17.612	6.517	0.426	0.212	10.302
2006-06	62.266	17.706	7.172	0.509	0.220	12.128
2006-07	61.561	17.803	7.230	0.474	0.194	12.739
2006-08	62.299	19.025	7.244	0.572	0.147	10.713
2006-09	61.559	19.602	6.710	0.595	0.163	11.371
2006-10	61.237	19.761	6.162	0.772	0.144	11.925
2006-11	60.627	20.793	6.574	0.793	0.279	10.934
2006-12	57.820	20.301	6.522	0.689	0.304	14.365
2007-01	58.303	20.992	6.613	0.602	0.270	13.220
2007-02	56.307	22.746	7.178	0.712	0.337	12.720
2007-03	52.002	29.281	6.817	0.543	0.281	11.078

- **Operating systems used by the visitors.** It is still dominated by MS Windows, and follows by Linux and others as shown in Table 12.2. The percentage of users who use Linux as desktop operating system is small compared to MS Windows. However, it cannot be neglected and increasing from time to time. The other operating system such as BSD and SunOS are mainly used for the proxy server in many corporate networks or Internet cafes in Indonesia. Some operating system which cannot identified are also mostly Linux-based system as well.

Table 12.2. Operating Systems

Month	Operating System (%)					
	MS Windows	Linux	Mac	Free/NetBSD	SunOS	Others
2006-02	89.888	2.241	0.758	0.015	0.007	7.090
2006-03	87.203	2.254	0.699	0.010	0.003	9.831
2006-04	84.702	2.348	0.814	0.007	0.004	12.125
2006-05	84.947	3.535	0.832	0.037	0.007	10.643
2006-06	84.935	2.468	0.900	0.007	0.002	12.228
2006-07	84.588	1.618	0.947	0.017	0.014	12.816
2006-08	86.342	1.730	0.982	0.029	0.011	10.906
2006-09	85.489	1.591	1.046	0.031	0.005	11.839
2006-10	84.590	1.467	1.229	0.044	0.022	12.647
2006-11	85.699	1.615	1.446	0.002	0.000	11.237
2006-12	82.299	1.725	1.238	0.007	0.002	14.728
2007-01	83.386	1.828	1.022	0.004	0.000	13.759
2007-02	82.920	2.422	1.337	0.013	0.005	13.304
2007-03	84.182	2.549	1.085	0.020	0.000	11.534

The make use of Open Source solution in web site of President of Republic of Indonesia shows that Open Source is enabling the local developers to provide a suitable solution in shorter time with limited budget. Open Source is also a better alternative than available proprietary solutions in Indonesia by comparing the features and costs.

12.6 Open Source adoption in Indonesia government

In adopting ICT, Indonesia as a developing country need a breakthrough strategy which can effectively accelerate the use of ICT. It is believed that ICT has a big potential to improve the quality of life and accelerate the development. However, to reach this goal Indonesia government should be more active and has a good commitment to build the political awareness, developing a good competitive business environment, as well as preparing the readiness of public in adopting the ICT. The cost of software as well as the

piracy rate are still a big problem that has to be encountered by government and public in adopting the ICT for daily operation.

However, many institutions developing their IT system without considering the software cost. The use of proprietary and closed software do not only make the cost very high but also create some hidden cost and hidden problems, such as the hardware upgrade, file format, the software piracy, and in some specific cases, the fact that the program code cannot be read leaves doubts regarding the reliability, particularly in terms of confidentiality.

After the economic turmoil in 1997, some government bodies started to consider and used open source solutions for their intranet and Internet servers. This decision was due to only the cost consideration. The increasing of the currency rate of dollar made software license cost could not be covered by budget of most of the government projects. Slowly, Open Source has been being adopted by various government bodies, including the Information System of President of Republic of Indonesia. On 2001, the government released The Presidential Decree No. 02/2001 about the development of User Interface in Indonesia language. The implementation of this decree produced the Linux distro called WinBI. In the same year government release Presidential Decree No. 06/2001, which stated that government bodies have to start to consider the open source solution. On 2004, five ministries launched an effort called **Indonesia, Go Open Source !** (IGOS). Through this program several projects has been established, such as the IGOS Desktop, Open Source Research Center, and various open source projects. IGOS has goals as follow :

- The use of legal software in all government bodies
- Informing and encouraging the public for the use of FOSS in Indonesia
- Preparing the migration guidelines and the using of FOSS guidelines.
- Encouraging the development of training centers, competency center, and business incubation center which based on FOSS
- Encouraging the increase of coordination, capability, creativity, motivation and participation of government bodies in implementing the FOSS

After the adoption of Open Source in a high profile web site such as the President of Republic of Indonesia, the public is more aware about the possibility of the use of Open Source. Currently, there is migration projects in the government offices in Aceh. After Tsunami, the local government wants to exploit the Open Source in order to accelerate ICT development in Aceh.

Indonesia as a developing countries, needs a IT solution which can minimize the total expenses of the IT system. It can be established by exploiting the FOSS based solution, or combination between closed/proprietary or both of them. The decision should be made case by case. Interoperability between systems is a must. For example, the eGov system which is extensively developed by government bodies should put the interoperability between system as a requirement in order to provide a maximum services to fulfill the society needs. The development of IT should be based on the cost reduction.

The adoption of Open Source in government bodies shows that :

- The cost of IT deployment will be reduced
- Local software developer will play more important role in providing the customization and supports
- Increasing the innovations and creativity of local software developer
- Providing the flexibility for software development and system integration. The proprietary solution sometime is difficult to integrate.
- Security in IT is an important consideration. It is not only about the attack and security threat, but also about the dependency to a single vendor. Being dependent to a single vendor in providing the software solution is very fragile to a developing country such as Indonesia.

However, the price of license is not only the driving factors in choosing the Open Source solutions. The openness and the flexibility of Open Source for providing a secure system is a major factor that in adopting the Open Source as a major components in Information System of President of Republic Indonesia. The possibility to provide a sustainable solution as well as the independency in providing solution for local problem are also factor being considered in adopting the Open Source solution.

12.7 Sustainable development model for culturability approach

As a comparison, in 2004 there was a collaboration project between Microsoft with Bandung Institute of Technology to translate MS Windows XP. It took about 1 year to produce translation of Microsoft Windows XP menus (language pack of Windows XP). However, this translation have not produced online help or other programs in MS Windows, only the menu in MS Windows XP desktop. Comparing with WinBI, which also produces online Help of desktop system and various application programs localization result from Microsoft was smaller [9].

Comparing time required by Microsoft localization project, WinBI project produced more results in shorter time. The translation Windows XP could not performed as fast as WinBI because translator team cannot compile the language packs, or performing the test by their own. Translators had to send the result to Microsoft before getting the final system. Translators also cannot build the tools by themselves. Thus, depends on the tools provided by Microsoft. Moreover, the translators might not use the translation results from various localization FOSS due to the license restriction. Beside faster, WinBi was also done with smaller budget. Cost consideration is a critical factor for developing countries.

WinBI results such as the translations, translated phrase database, tools, user and developer documentations are always available for public. It creates

possibility for other developer groups to continue the works of WinBI. It is a necessary causal factor to support the sustainability of the works. It is demonstrated by the continuation of WinBI works without the involvement of the initiator. The result of localization project from proprietary software may not be used by other groups without getting permission from the vendor. For developers in developing countries establishing the vendor permission is not an easy task. Thus, whenever the initiator or the vendor do not want to continue the project, the work cannot be sustained. FOSS provides a better platform for delivering sustainable localization solutions.

The web sites system of the President of Republic Indonesia is developed by small number of team in short time with limited budget. However, the developed system can satisfy the requirements. It also demonstrates the flexibility of FOSS to provide solutions that are difficult to be fulfilled by proprietary solution. In this project, FOSS shows strong point in some area such as virtualization, security, and reliability. FOSS always is capable to fulfill the additional requirement during the operation of system. In other word, FOSS is possible to provide a sustainable solution.

Sustainable development model attempts to meet the current and future need without compromising environment. The sustainability model has three main considerations : social (people), environment (interoperability), and economics (cost). For a sustainable development, understanding the current user requirements and the future requirements is important aspect. However, in many development methodologies and approaches user are still viewed as anonymous object that is going to use the system [6]. In a sustainable development model, understanding the user and the context of the use of system can produce a better system which can sustain to satisfy the future requirement. Most requirement engineering approaches still disregard the active user involvement during the whole project.

Participation of stakeholders should take place throughout the strategy and planning process in defining problems, setting vision, developing solution. Employing development methods which have their origin in the socio-technical such as co-operative, collaborative, participatory or participative design, can accommodate better the user requirement. Technology shapes society, but society also shapes technology [4]. Therefore, increasing the awareness of social expectation concerning technology in use is important to develop a sustainable solution. Balance the different aspect of requirements of people, organization and technology which make up complex socio-technical system within their social and natural environments, within processes of global networking.

Software engineering cannot be seen as pure development of technology, because it also induces a change process in the environment where it take places. The critical success factor is to adapt constantly the planning to any changes. FOSS provides a socio-technical infrastructure which can be adapted easily in the future. OSS model is good candidate for sustainable system development model for software engineering.

12.8 Conclusion

Open Source is enabling the developing countries to fulfill their local needs. Furthermore, the specific requirement due to the local or cultural context can be fulfilled by using the local human resource. The localization project of FOSS shows this phenomenon. By using smaller team, and require shorter time and smaller funding, better solution can be provided comparing to the same solution from proprietary vendor. From the economical consideration it is a good point for developing countries such as Indonesia.

The proliferation of distro development in Indonesia shows the wide adoption of Open Source as well as the availability of human resource for providing the open source solutions. The freedom to adopt, adapt and modify the software to suit its needs is paramount to the government initiative. One major lessons learned from the distro development project such as WinBI, BlankOn is that without enough funding for full time developer a workable solution can be provided.

Government information system are public information system which should be analyzed from several perspectives as regards users, usages, data contents, and technical, organizational, and legal aspects [2]. Therefore, in developing the web site of President of Republic of Indonesia, the FOSS is chosen instead of proprietary solution. Moreover, some requirements cannot be fulfilled economically by proprietary solution. For the future requirements, FOSS provides a flexible socio-technical infrastructure which can be always adapted.

However, before the OSS adoption can be done widely in government bodies. The dissemination of idea of FOSS should be established. The FOSS capacity building is an important steps in a successful FOSS adoption. The open source dissemination model also inspired a new model of distribution of scholar works. Distro plays important role as media for delivering the knowledge and the FOSS. Therefore, in promoting the use of FOSS in Indonesia, we started by building a starter kit which consists of distro and sets of documentations that are developed by collaboration among universities, communities as well as the companies. Universities are important component because they have supported, and have enough human resource as well as they owned the business incubation model, which can uses the FOSS solution as the basic of their offered services.

The guidelines and starter kit will have 3 difference audience or targets : IT decision maker in government bodies, IT solution and service providers for government bodies, and public, as well as the education institutes and training providers. The guidelines consists of general FOSS guideline, technical guideline for migration work, legal guidelines for the use of FOSS and also the guidelines for business and economic models. It will assist the companies or public to asses the advantages, cost of the use of FOSS. In the FOSS adoption government should take position more or less acts as facilitator in this work rather than performing every thing by themselves.

References

1. Birgit Renzl (2006). Language as a vehicle of knowing: the role of language and meaning in constructing knowledge. *Knowledge Management Research & Practice*, 5:44–53.
2. Bo Sundgren (2005). What is a public information system. *International Journal of Public Information System*, 1:81 – 99.
3. Dali S. Naga (1984). *KILANG 002: BASIC dalam Bahasa Indonesia*. Kesaint Blanc.
4. Dietrich Brand (2006). The global technology laboratory. *AI & Society*, 21:453–470.
5. eBizz Asia (2003). Perang apel dan orange. *eBizz Asia*, 1(11).
6. Hannakaisa Isomaki (2005). Nuances human-centredness in information system development. In *Proceedings of the 38th HAWAII International Conference on System Science 2005*.
7. Huigang Liang, Yajiong Xue, William R. Boulton, and Terry Anthony Byrd (2004). Why western vendors don't dominate china's erp market. *Commun. ACM*, 47(7):69–72.
8. I Made Wiryana (2006). Laporan tahunan situs presiden sby. Technical report, Konsultan Teknis Situs Presiden SBY.
9. Merry Magdalena (2004). Windows xp bahasa indonesia kali ini, bill gates kalah start dengan linux. *Sinar Harapan*, 27/05/2004, 27/05/2004.
10. Minaa Koskinen, Eleni Berki, Katja Liitamainan, and Mikko Jakala (2005). The human context of Information system *Proceedings of the 38th Hawaii International Conference on System Science 2005*
11. Nah Soo Hoe (2006). *Breaking barriers : the potential of Free and Open Source Software for Sustainable Human Development*. UNDP Asia-Pacific Development Information Programme.
12. William Boulton Kittipong Laosekathul (2006). Critical success factors for ecommerce in thailand : cultural and infrastructural influences. *The Electronic Journal on Information System in Developing Countries*, 30(2):1–22.