Association for Information Systems
AIS Electronic Library (AISeL)

ECIS 2006 Proceedings

European Conference on Information Systems (ECIS)

2006

Critical issues associated with adoption and use of open source software in public sector: insights from Tanzania

J. Lungo
University of Oslo, jumal@ifi.uio.no

CRITICAL ISSUES ASSOCIATED WITH ADOPTION AND USE OF OPEN SOURCE SOFTWARE IN PUBLIC SECTOR: INSIGHTS FROM TANZANIA

Juma Hemed Lungo, Department of Informatics, University of Oslo, Gaustadalléen 23, P.O.Box 1080 Blindern, 0316 Oslo, jumal@ifi.uio.no

Abstract

Two Open Source Software (OSS) projects in Tanzania are discussed as exemplar efforts for the adaptation and use of OSS in public sector. The projects investigated in this Action Research Study were from the Health and Education Sectors. The two projects are of important in that, they use two different approaches of embracing OSS where the Health Sector project uses an approach of getting existing and working open source software framework and customizes it with the help of contributors from Open Source Community, while the Education Sector project developed software from scratch under the auspice of OSS development approach. A comparative analysis of the two projects with the terms of the Open Source Software definition indicates that the two developed software did not take into consideration of all OSS terms. The study concludes that the deviation from the terms of OSS definition reflect incompatibility between free software and open source software philosophies which results into difficulties to comply with one of them. As a result the focus is to achieve the freedom to use the software for any purpose and for any number of computers and the freedom to maintain the software without depending on the author of the software.

Keywords: Information Systems, DHIS, Open Source, Zalongwa, Action Research

1 INTRODUCTION

Nowadays software is rapidly becoming one of the most fundamental building blocks of human interaction and activity (Klang 2005). This fact indicates that there is a need of bringing about an understanding of the computer software because this software belongs to someone. Klang (2005) emphasizes that the software that fills our homes and our lives is, in almost all cases, is a property of someone else and therefore we are dependent upon the property of others for our everyday lives to a much greater extent than we may previously have imagined.

Proprietary computer software are experiencing criticism from many organizations that they demand high total cost of ownership (TCO), they lock organizations to one vendor, they suppress innovations, and that they ship with background spy-wear software. To address their concern, many countries are now adopting Open Source Software (OSS) (NAGARAJ 2002; Cardoso 2004; Desk 2004). Open source is fundamentally a model of distributed, shared, open software development. Open Source Software development philosophy is gaining momentum in many developing countries as it is believed to be cost effective as it often comes free of charge, can be customized, helps progress of technology, prevent monopoly over ICT markets, and it benefits to the poor nations where software license costs accounts for government expenditure budget(Weerawarana and Weeratunga 2004). Countries like India, Brazil and China have attempted to adopt OSS and their decisions were largely backed by political leaders. Most of the reported OSS cases are mainly based on programs developed for computer professionals such as Operating Systems (e.g. Linux distributions) and server side programs (e.g. Sendmail, Apache, PHP, MySQL, etc.) There is little said on Computer Application Programs developed for specific application domain in organizations and in public sector in particular where most of the users are not computer professionals. This study aims to shed light on practical OSS projects in public sector in order to understand its adoption and use.

This paper underpin important features which drive public sector's decision making on the adoption and use of open source software. I bring arguments drawn from OSS projects at the University of Dar es Salaam and at the Ministry of Health in Zanzibar, Tanzania. At the University of Dar es Salaam, the project is the design of an Academic Register and Exam Databank Management Information System (AREDMIS). At the Ministry of Health Zanzibar, the OSS project involved customization of software in order to be used in the Health and Management Information System (HMIS) of Zanzibar.

In this first section, the paper introduced concepts of OSS and the study case studies. The second section presents theoretical perspectives on proprietary, free and open source software. In the third section, I present the research design, methodology, and the innovation projects. The fourth section presents analysis of the case studies. The fifth section presents discussion on issues considered as of important in adopting OSS. The sixth section concludes the paper by presenting summary of open source theories and the author's stands. The last section presents useful cited works, the reference list.

2 THEORETICAL PERSPECTIVE ON SOFTWARE

2.1 Proprietary Software

The Free Software Foundation (FSF), an organisation whose goals is to promote the computer users' right to use, study, copy, modify, and redistribute computer programs, describes software that is not free as a Proprietary Software. Technically, the term means software that has an owner who exercises control over the software. Proprietary Software is software that has restrictions on using and copying it, usually enforced by a proprietor. The prevention of use, copying, or modification can be achieved by legal or technical means. Technical means include releasing machine-readable binaries only, and withholding the human-readable source code. Legal means can involve software licensing, copyright and patent law. The FSF uses the term to highlight that the owner is of prime importance, in contrast to "free software", where the freedom of computer users is of prime importance. Well known examples of proprietary software include Microsoft Windows, RealPlayer, Winzip, and Adobe Photoshop. Generally, proprietary software does not give a user any freedom to distribute original or derivative works and that to view, to attempts to view or to recreate the code is strongly prohibited.

2.2 Free Software and Open Source Software

When computers first reached universities, they were research tools. Software was freely passed around, and programmers were paid for the act of programming, not for the programs themselves. Only later on, when computers reached the business world, did programmers begin to support themselves by restricting the rights to their software and charging fees for each copy. 'Free Software' as a political idea has been popularized by Richard Stallman since 1984, when he formed the Free Software Foundation (FSF 2005a) and its GNU Project (FSF 2005b). Stallman's premise is that people should have more freedom, and should appreciate their freedom. He designed a set of rights that he felt all users should have, and codified them in the GNU General Public License (GPL). More specifically the freedom envisioned by FSF which was formalised in the GPL concerns four freedoms, which are the freedom to: (0) run the program, for any purpose; (1) study how the program works, and adapt it to your needs; (2) redistribute copies so you can help your neighbour; and (3) improve the program, and release your improvements to the public, so that the whole community benefits(FSF 2005c). The software freedom envisioned by the FSF on its GPL does not satisfy all people. (Rosen 2004) sees the FSF's GPL license being constricting the user since any derivative works must be licensed under the same terms. The GPL reads "you must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License"(FSF 2005c). This implies that widespread commercial development cannot take place under the GPL license. The conflict involved in the term free is depicted in the free software definition when it ask readers to think of 'free' in terms of free speech rather than 'free beer'. To address this concern, in 1998, the Open Source Initiatives (OSI) movement was launched with the term 'Open Source Software'.

The Open Source Definition includes many of Stallman's ideas, and can be considered a derivative of his work. The Open Source Definition coined by Perens (2005) defines ten terms of rights that a software license must conform to be certified as Open Source Software (OSS). Those ten terms are free redistribution; source code availability; derived works allowed; 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 contaminate other software; and finally it give example licenses (Perens 2005). Open Source Software rights addresses the rights of users as well as the rights for programmers (sometime called 'hackers')(Raymond 1999). Perens (2005) sees these rights to be more important to the software contributors because they keep all contributors at the same level relative to each other. The notion with OSS is that anyone who invests the time to build knowledge in an Open Source program can support it. This provides users with the option of providing their own support, or the economy of a number of competing support providers. The fact that any programmer can tailor an Open Source program to specific markets in order to reach new customers adds strength to the open source movement because people who do these things are not compelled to pay royalties or license fees.

The two terms used interchangeably, "Free Software" and "Open Source Software" are interpreted differently according to individual or society ideological perspectives. Klang (2005) describe the term "free software" that it includes a philosophy, an understanding that software is an important building block in the information society and that the control of this infrastructure needs to remains accessible to all. Raymond (1999) insists that "Free software" refers not to price but to freedom. Klang (2005) describe the essence of open source that it was an attempt to promote open source as a software development model acceptable to corporate developers, those who had been reluctant to adopt a methodology connected to the ‘free software’. This resulted into a shift of the fundamental ideological perspectives from ‘correct software’ to ‘good software’ where software does not need to be free to be considered good software. It is important to stress that both FSF and OSI refer to the same types of software products and license, the different lies in the ideological judgement on whether software is correct or good for the society. Though they have a common enemy, proprietary software, Richard Stallman made it clear that OSI and FSF movements are separate movements with different views and goals, although they can and do work together on some projects (Stallman 2002). This concludes that the Open Source Software was created to meet the needs of both those developers who were disenchanted with the view that software needed to be ideologically pure as opposed to functional and of those developers who were attempting to entice more traditional software manufacturers to join the free software/open source movement (Klang 2005). This vision of supporting commercial activities in open source led the open source software to grow massive in both economically and politically.

Explaining open source and the way open source work, Weber argues that, the philosophy of open source is not the software. It is the ‘process’ by which software is created (Weber 2004). Thus we need to think of the software itself as an artefact of the production process because production processes, or ways of making things, are of far more important than the artefacts produced because they spread more broadly. Open source has proved itself as a way of making software, and as we develop more sophisticated understanding of what the open source production process is, there is a potential of extending the process to other new physical goods. In order to understand the open source process, it is required to look into four aspects (Weber 2004, p.56): “(1) who are the people who write open source code? (2) what do these people do, exactly? (3) how do they collaborate with each other? and (4) how do they resolve disagreements and deal with conflicts?” As the study focuses open source development in public sector where there are non-ICT intensive establishments featured by specific domain which requires hackers to be familiar with the application domain and usually there are limited ICT professionals, the interest is on the 3rd question, how do people collaborate with each other?

The collaboration on open source development process is guided by the use of technology, license schemes and architecture of the software. Hackers form ‘virtual teams’ where communication is made through the internet where they share codes, technological tools such as bug database and code versioning systems and also new users and programmers join the teams through the internet. An open source licensing scheme generally creates a social structure because it empowers users by ensuring access to source codes, passes a large proportion of the rights regarding use of the code to the user rather than reserving them for the author, and it constrains users from putting restrictions on other users in ways that would defeat the original goals. The architecture of opens source software is not pre-formulated instead the architecture tracks organisations. That is the architecture of the software is formulated as codes goes on hence the codes defines the architecture because in most cases the technical decisions are influenced by beliefs about effective ways to organise development and thus is about a reflections on social practices.

2.3 Open Source Software in Public Sector

Many governments are avoiding using proprietary software because it is relatively easy to hack into, and then the governments have to depend on programmers of the commercial software to fix the problem. Governments also like OSS because by using it, they will not be locked into a contract as they would be with a company like Microsoft. Munich and Brazil are the first leading initiatives on open source followed by others like China, Japan, India and Korea who have stressed the importance of forming a cooperation around implementation of open source software (Bentz, Boeckman *et al.* 2004).

As Open Source gets more acceptances in public sector, there is expanding market for it. This is because the open source movement believes strongly that rather than being a trade secret, the source code beneath the software should be open for anyone to play with. The result is software like the Linux operating system, created through collaboration that involves thousands of programmers. Reid

(2005 argues that the open source movement does not object to making money, the source code may be free, but there is gold in software support, training and publishing. Reid adds that with businesses keen on open source software and the support it offers, there is a business opportunities and that the software, is now designed by, but not necessarily for, programmers (Reid 2005). When presenting Open Source Software business model lecture, Isene tells that many IT companies make business with establishments where establishment pay developers to customise or develop their open source systems through installing, customising, and developing from scratch. Establishments are relieved from paying annual license fees and from being locked to one vendor (Isene 2005). The discussion shed light that when it comes to non-ICT intensive establishments especially in the public sector what matters is the availability of source codes which gives the freedom to maintain the program obtained on what ever development process.

It is difficult to draw an exact picture of the usage of open source software in the public sector because there are huge differences in the implementation of open source software between country to country and also within a single country. Software implementation and usage is normally part of the responsibility of the single governmental institutions themselves. Due to the lack of political policy directives in the context of open source software, IT managers in public institutions are normally - within their budgets - free to buy and install the software they consider to be the most appropriate for their situation. The lack of policy directives on a supra-institutional level causes an absence in the monitoring of installation and usage of open source products on a large scale(Ghosh, Krieger *et al.* 2002). In many OSS studies there are rare available statistics on the level of single institutions. These institutions are in a phase of official policy-driven implementation of open source software and in those institutions where migration and implementation not following a specific policy is usually are not tracked. In many cases open source software is not used as a single platform, but as an additional feature on a proprietary software e.g. operating system. These quite frequently mixed configurations normally do not appear in statistics at all. So if you count all these misreported cases, you may find a handful of OSS cases in the public sector.

In a survey study by Wichmann (2002) on the use of OSS on Firms and Public Institutions in three countries: German, Sweden and UK, the researcher found many activities of adaptation and use of open source software in the public sector. In a recent study, the Bridges.org(Schmidt 2005) scanned OSS use in Africa in a 'Comparison Study of Free/Open Source and Proprietary Software in an Arican Context' focused on Implementation and Policy-Making to Optimise Public Access to ICT and found many cases including several National ICT Policies citing Open Source Software interests.

Wilchmann's study revealed two critical issues driving establishments in their decision-making for adopting OSS that "(1) establishments use OSS to become more independent from pricing and licensing policies of large software companies (2) establishments are using open source software because they want to support the open source community" (Wichmann 2002). The study further investigated the importance of OSS as part of an establishment's ICT infrastructure and revealed that ICT intensive sector consider their OSS to be important constituent of their ICT infrastructure than the public sector. This suggests more in-depth investigation in order to understand important features of OSS driving public sector to adopt open source software. Some studies point higher stability and better access protection than proprietary solutions are most important decision factors in favour of an OSS server operating system (Wichmann 2002; Weerawarana and Weeratunga 2004). The point here is that end-users see critical issues in software differently from software programmers. For example, from technical stand it is always argued that security and total cost of ownership (TCO) is the main reasons for opting Linux over Microsoft Windows. Vaughan-Nichols (2005) reports a survey conducted at Rackspace Managed Hosting Ltd (which manages more than 13,000 servers, with an approximately fifty-fifty Linux and Windows split) that the choice between Linux or Windows operating system is driven largely by available developer talent and individual application requirements. The operating system choice is based on middleware, database and programmer availability considerations (Vaughan-Nichols 2005).

The terms free software and open source software refers to two different software perspectives and non of them is without shortcomings. Free software is seen as anti-commercial while open source is supporting commercial initiatives. In this study an exploration of the motive behind and critical elements of Open Source Software driving public sector towards adaptation and use of OSS are examined. Specifically the study explores the elements of Open Source software considered to be raising OSS over proprietary software by analysing the openness of the development process of the two projects and the resulting software.

3 RESEARCH METHODOLOGY

3.1 Study Design

The study took place from December 2004 to August 2005 in the field where the author was actively involved in two projects: (1) Design and Implementation of Student Academic Register and Examination Databank at the University of Dar es Salaam (UDSM) known as ZALONGWA and (2) customisation of a health data analysis tool in the Ministry of Health in Zanzibar known as District Health Information Software (DHIS). In the two projects, the researcher played two roles: Programmer and Team Leader. These dual roles gave a wide angle of view to observe, interact, take actions, manage the process and witness all happened in the field. Though, the study involved two projects from different domains (education and health sectors) geographically located apart (Zanzibar and Dar es Salaam), the two projects share common theory of great interest to study, *adoption and use of open source software in the public sector*. This is because Open Source Software (OSS) has recently gained increasing attention in Information Systems Research but there is still a lack of empirically sound and analytically novel approaches that show the multifaceted nature of OSS phenomena.

3.2 Research Approach

This study falls under "Action Research Paradigm". Action research has been typified as a way to build theory, knowledge, and practical action by engagement with the world in the context of practice itself (see, e.g.(Kock 1997);(Whyte, Greenwood et al. 1991)). (Dick 2002) explains an action research as a research approach, which has the dual aims of action and research:

- action to bring about change in some community or organisation or program;
- research to increase understanding on the part of the researcher or the client, or both.

In this study, five phases of action research approach are adopted, as Baskerville and Wood-Harper (2002, p.133) argue that, "the most prevalent description of action research details a five phase, cyclical process which can be described as an 'ideal' exemplar of the original formulation of action research"(Baskerville and Wood-Harper 2002). This ideal approach first requires the establishment of a 'client-system infrastructure' or research environment. Then, five identifiable phases are iterative: (1) diagnosis, (2) action planning, (3) action taking, (4) evaluating, and (5) specifying learning. The key assumptions of action research are that "social settings cannot be reduced for studying and that action brings understanding" (Baskerville 1999). In our particular case, the client system infrastructures are the Health and Management Information System (HMIS) of Zanzibar and the Academic Registration and Examination Databank Management Information System (AREDMIS) of the University of Dar es Salaam.

Action research aims to contribute both to the practical concerns of people in an immediate problematic situation and to the goals of social science through joint collaboration within a mutually acceptable ethical framework. The ideal domain of action research is therefore revealed in three distinct characteristics (Baskerville and Wood-Harper 2002 p.136) of the approach: The researcher is actively involved, with expected benefits for both researcher and the research client. The knowledge obtained could be immediately applied. The research is a cyclical process linking theory and practice. Inline with this action research practices, the author of this paper has played three roles: first, he was among of programmers of the systems and he coordinated all analysis, design and implementation activities of the project. Second, he was a researcher where he wanted to see how we can use open source theoretical perspectives to implement systems of this nature in the specialised domain. Finally, he was a system user where as a University Lecturer and Departmental Examination Officer, he wanted to publish examination results and get feedback on human errors made while processing examination results and that he wanted to enable university students to access their examination results where ever they are. The arguments and analysis of this paper is on the process of developing the systems as open source projects. Specifically the author aims to contribute to both the research approach and on understanding open source processes. In terms of the research approach the contribution will be the application of action research approach in analysing software development processes instead of the dominating practice of evaluating outcomes of actions as successful or failed.

4 THE CASE DESCRIPTION AND ANALYSIS

4.1 The District Health Information Software (DHIS)

Since 2004, the Ministry of Health in Zanzibar is introducing a comprehensive Health Managing Information System (HMIS) reform by implementing national standardized health data reporting forms and a computer database system known as District Health Information System (DHIS). The DHIS has several functionalities such as maximum and minimum ranges, validation rules, data definitions, indicators, report generator, a number of modules ranging from PHC to hospital and TB, organizational unit infrastructure, and annual surveys. DHIS was designed to capture health data at the district level of the health information system. The DHIS was designed in South Africa and is distributed under Open Source Software philosophy in that it is distributed with its source codes; it is distributed for free in a website and in CDs, and you can get a support from the developers through a mailing list. In this project, the main activities were to setup the DHIS database and customizing DHIS reports and data entry forms. A team of programmers was hired to customize the DHIS, though the software was obtained for free. The link between the DHIS developers in South Africa and the customization team in Zanzibar was maintained primarily through a mailing list. The DHIS issues and bug track website was sometimes used. To download the DHIS, the team needed to visit expensive hotels during the night because of the low bandwidth and took at least two hours to download a 14MB zipped file. To connect the Zanzibar team to the DHIS community, bugs were communicated through emails, SMS, and instant messages but in most of the time bugs were reported to private email addresses of the most DHIS experts in South Africa. At the time of writing the DHIS in Zanzibar was customized to the extent that Zanzibar health data collection forms were successfully mimicked in the DHIS electronically and that the DHIS is installed in four (4) districts in Zanzibar and initial DHIS user training took place. In the four districts where the DHIS runs, no single computer is connected to the internet a situation which forced the DHIS to make its way to the districts through CDs and that DHIS end-users are completely blocked from the DHIS's issues and bug tracks website. Apart from free (non-paid) contributions from the DHIS developers in South Africa, there are no evidences of hackers' contribution around the world in general and all the team members in Zanzibar work under monthly based paid contracts. Apart from the DHIS programmers, six health works (employees of the Ministry of Health) form the Zanzibar form the taskforce for implementing the HMIS in general and the DHIS in particular.

4.2 The Academic Register and Exam Databank (ZALONGWA)

Academic Register and Examination Results Databank Management Information System (AREDMIS) was among of the system envisioned in the University of Dar es Salaam ICT master plan approved in

1995 (Nfuka 2004). AREDMIS is aimed to cater for electronic student information system which stores course registration and examination results. The University of Dar es Salaam has incurred several expenses in the past on its attempts to build its AREDMIS. In onetime, the university has received a contract of USD $120,000/= from a local company to develop AREDMIS (UDSM-ARIS 2004), however, the company did not deliver the software until the time of writing. In 2003, the Department of Computer Science designed a limited version of AREDMIS (called ZALONGWA Software). Zalongwa is a web based application developed using Open Source Software Technologies which came pre-installed in a Linux Fedora Core 3 server.

In the academic year 2003/2004, Zalongwa was used to compile and display examination results of all students at the Computer Science Department. Due to its performance, other Departments from the Faculty of Science and other Faculties at the University developed interest to adopt Zalongwa, but in order to support all the requirements, the software had to be redesigned. In January 2005 a development team was set to upgrade Zalongwa from departmental to University level specifications. Different from other Open Source Software projects, in developing Zalongwa (the projects did not make use of website to track bugs and issues).

The development team (one lecturers and three systems administrator) are university employees and were not given incentives for the act of developing the system. During the process of testing Zalongwa, old examination data from hardcopies, MS Excel and Word files were extracted and imported to Zalongwa. The development team then was paid to Extract Transform and Load all examination records from the academic year 2001/2002 to 2003/2004 to Zalongwa (The University has paid only USD $2,000 for data extraction transformation and loading at its Faculty of Law). In July 2005, all students from the Faculty of Arts and Social Sciences, Faculty of Engineering, Faculty of Law, Faculty of Commerce and Management have seen their examination results on an internet café, for the first time in the history of the University of Dar es Salaam.

At the time of writing, Zalongwa was adopted by the second largest university in Tanzania, the Open University of Tanzania (OUT 2005) and NetTel@Africa, a network of universities in Africa which offers collaborative degree programs. In order to adopt Zalongwa, the Open University of Tanzania and NetTel@Africa had to organize local teams at their institutes for customizing Zalongwa with a help of the original developers of Zalongwa. Although the original developers were give paid contracts, but the institutions have made large serving. For example, the Open University of Tanzania has paid only USD $6,000/= to adopt Zalongwa while its ICT policy estimates USD$420,000/= for developing similar system.

Though, the DHIS and Zalongwa are in a substantial use, Zalongwa has over 13,000 users (students and lecturers) at the universities and the DHIS is already deployed at the districts where users have started to enter health data, neither Zalongwa nor DHIS come with an ‘End User License Agreement’ (EULA). Even though, there is a missing EULA, users seem to be not concerned at all. The DHIS and Zalongwa are distributed with its source codes and there is un-written freedom to customize the software, run the software in any number of computers and for any purpose. For example, apart from Zalongwa being an Academic Register and Examination Databank Management Information System, the universities also use the software as student accommodation system, student financial information system, and e-learning system. The same with DHIS, in the Tanzanian mainland the DHIS has been used as a pilot health data analysis tool in 5 districts and also is being used for teaching health workers at the University of Dar es Salaam in a Health and Management Information System (HMIS) NORAD-Sponsored course. The two projects have started in a small scale and in the form of cathedral approach but did not grow up to the bazaar stage (Raymond 1999). That is, in the two projects there is limited contribution of anonymous hackers.

The Action Research phases were followed cyclically during the execution of the projects. Analysis of the action research phases is now presented:

- **Diagnosing:** Baskerville and Wood-Harper (2002, p.134) describes diagnosing phase as it “corresponds to the identification of the primary problems that are the underlying causes of the organization's desire for change". According to Baskerville and Wood-Harper, diagnosing involves self-interpretation of the complex organizational problem to develop certain theoretical assumptions about the nature of the organization and its problem domain. In the HMIS case, we wanted to solve the problem of health data storage, compilation and analysis by implementing a computer database while in the AREDMIS case the problem was the storage, retrieval and dissemination of academic records. Due to the nature of the client systems infrastructure, it was observed that open source solutions would enable wide sharing of development process and sharing tools and experiences on similar application domains and will reduce the cost of implementation.

- **Action Planning:** After the diagnosing phase, researchers and practitioners then collaborate in the next activity, action planning. The discovery of the planned actions is guided by the theoretical framework, which indicates both some desired future state for the organization, and the changes that would achieve such a state. The plan establishes the target for change and the approach to change. As we have seen through theoretical perspectives that Open Source Software is a suitable approach, we planned to develop the AREDMIS using Open Source Software called MySQL database System that will run on Linux (Fedora), the business logic of the system is implemented using PHP and Java scripts. In the case of the HMIS, an existing health data analysis tool known as District Health Information Software (DHIS) was acquired as an open source framework which could be customized according to the HMIS requirements. It was further planned to engage programmers who will actually execute the projects. This is because the projects required programmers who are familiar with application domains.

- **Action Taking:** This phase implements the planned action. The researchers and practitioners collaborate in the active intervention into the client organization, causing certain changes to be made. In the HMIS case, the DHIS framework was obtained from a website free of charge. The database was customized, health data elements were introduced in the database, and the user-interface (forms and reports) were customized to suit the HMIS in Zanzibar. During customization of the DHIS the DHIS authors in South Africa were consulted from time to time through a mailing list. The AREDMIS was developed as a combination of six modules namely: (1) Nominal Roll – which keeps students biographic data; (2) Accommodation – which keeps information about students halls of residents and student room allocation; (3) Academic – which keeps academic related information including examination results; (4) Accounting - for payments managements; (5) Student Information – which combine different information from other modules that is relevant and personalized to each student and (6) Database Maintenance – which enable system administrator to manage users of the system including granting and revoking user rights.

- **Evaluating:** After the actions are completed, collaborating researchers and practitioners evaluate the outcomes. Evaluation includes determining whether the theorized effects of the action were realized, and whether the effects relieved the problem. A reflection of the development process was developed in order to determine the fitness of the projects. In this system, evaluation was done in two approaches: (1) to evaluate whether the university community is using the system or not. This was done through logging system use. The system has a feature which records every user who logs in and records every page a user has accessed. With this, it is possible to log how many web hits have been performed per day. (2) The DHIS was evaluated through entering old health data to the system, progress meeting, training sessions and interviewing of health workers. A comparison of the development process and the resulted software with the open source definition was performed in order to determine the extent the projects have deviated from the open source definition

- **Specifying Learning:** According to Baskerville, knowledge gained in the action research (whether the action was successful or unsuccessful) can be directed to three audiences First, restructuring of organizational norms to reflect the new knowledge gained by the organization during the research. Second, where the change was unsuccessful, the additional knowledge may provide foundations for diagnosing in preparation for further action research interventions. Finally, the success or failure of the theoretical framework provides important knowledge to the scientific community for dealing with future research settings (Baskerville, 1999 p.14). This paper is aimed to specifying learning where it tells which has worked well, which part did not work and provide future development recommendations.

5 DISCUSSION

While many Open Source Software projects are featured by virtual communities around the internet. Usually a website is set where hackers have full access to the project source codes. However, in these two projects no one is maintaining a project website (at the time of writing) and a communication with outside hackers have maintained through mailing lists. That is to say, external hackers helped the local teams through debugging the pieces of source codes and in sometime by answering logic questions. When a serious need of external helps arises, the project team members used to send queries to a specific contact person. This concludes that, the developers were bonded to deliver the products through a formal contract. Binding developers with a formal agreements is one of acceptable open source business model where although clients pay developers, but once the product is delivered the client will have a freedom to maintain the software without necessarily relying on same developers.

The two software products come without licenses. In the case of Zalongwa, it can be argued that each organization is developing different system according to its use and therefore is like developing in house software. However, each organization uses a substantial amounts of original source codes of Zalongwa, a situation called software 'forking' in Open Source Philosophy, thus it could be nice to tell even in which license these source code files follow under. The same applies to the DHIS; the software is deployed into use without any license. In a progress meeting, one of the DHIS developer tried to bring the issue of DHIS license, and another developer claimed,

> *"the software was developed from a non open source software platform [Microsoft Access] but we distribute it under open source software philosophy, so as long as the users understand the philosophy, there is no need of stamping license documents"* (DHIS Developer, July 16 2005, DANIDA Office-Zanzibar).

The interpretation of the excerpt is that some aspects and features of open source software are not perceived as critical in the process of adaptations and use of open source software in public sector. Because of the mixed nature of IT infrastructures in public sector, choosing licenses become difficult and this drives the establishments to not opt for any license wherever possible.

It is useful to argue the openness of the software in order to develop a theory on adaptation and use of open source. To construct the theory of adaptation and use of open source software, I present a comparative analysis of the two projects against the open source definition in order to determine the extent to which the two projects have deviated from the ideal open source software development approach.

Perens (2005) argues that, to be Open Source Software, all of the ten terms of open source definition must be applied together, and in all cases**.** For example, they must be applied to derived versions of a program as well as the original program. It is not sufficient to apply some and not others, and it is not sufficient for the terms to only apply some of the time. Table 1 presents the comparative analysis of the two project features against the terms of the open source software definition.

Terms of OSS definition	DHIS and Zalongwa Software	Remarks
Free Redistribution	Redistribution allowed	The software cannot be used in another setting without being customised
Source Code Availability	Distributed with source code	Source codes available for both software but the DHIS requires users to have a proprietary software, Microsoft Access 2003

Derived Works Allowed	Modifications and derived works allowed	Zalongwa derivatives is in use at other academic institutions in Tanzania
Integrity of The Author's Source Code	Undefined	There is no defined restrictions on users who wants to change codes
No Discrimination Against Persons or Groups	Undefined	Any establishment can use the software
No Discrimination Against Fields of Endeavor	Undefined	The software are specific for health and education settings
Distribution of License	Verbally available	Need to specify specific license
License Must Not Be Specific to a Product	Not specific to a product	Again the DHIS requires a user to have Microsoft Access 2003
License Must Not Contaminate Other Software	Can be packed with other programs	Can be packed with other programs
Example Licenses	Not specified	Need to be specified

Table 1: Comparative Analysis of DHIS and ZALONGWA Software against Terms of OSS Definition

The comparison made in Table 1 is not straight forward because the OSS definition is focusing on software licenses which requires that all software must be flagged a license. However, this is not always the case in other setting where software are regarded as tools for just doing the work. In this case license is not perceived a critical issue otherwise the software deployment could have not been possible. The tendency of overlooking the importance of some aspects listed on the terms of open source definition is seen as the second threat to the open source movement the first being the *Halloween Documents,* an internal Microsoft memos which outline strategies to fight Open Source Movement (Perens 2005). Perens (2005) argues that overlooking some terms of OSS definition is an attempt to dilute the definition of Open Source by including partially-free products.

Thus public sector sees the promised freedom under the FSF's GPL license cannot be achieved because of too much restriction imposed to developers. This is because in most of the cases Open Source products need to be customised to accommodate specific requirements in the public sector. Customisation needs a developer to understand the context of the system clients, so it is not easy to be achieved through widely distributed hackers. Open Source Software and Free Software enlighten the public sector that with good software public sector gets: (1) the freedom to use the software for any purpose and for any number of computers and (2) the availability of source code for maintainability. The open source business model of contracting developers gives establishments an opportunity of owning and managing the process of software development a privilege not available in proprietary software. In what ever software development process is adopted, establishments will be relieved from annual license fees and are guaranteed to get the aforementioned two freedoms.

The two cases are complementing each other in theorising the way public sector systems adopt and use OSS for they are presenting two different approaches of acquiring OSS products. Tanzania as one of the low income countries is an eligible case study in these particular cases where there are many unanswered questions on how poor countries can adopt OSS development approaches and products. The insights from Tanzania now are that despite the poor quality of internet connectivity, developing IT skills, and financial constraints, yet it is possible to implement OSS projects. These findings are of beneficial to both developed (donors) and developing countries (recipients) for outlines that OSS projects can be realised even in less resources counties.

6 CONCLUDING REMARKS

This study explored the adaptation and use of open source software in the public sector through following up two projects conducted in Tanzania. The study investigated critical issues driving public sector toward adaptations and use of open source software taking into consideration that the public sector establishments are not ICT intensive organizations. Software and ICT related tools are seen as just a tool for performing the work and thus the development of software in this sector may not necessary follow the same path in the same way software for ICT intensive establishments like IT companies take. These facts calls that as Open Source Software make its way to the public sector, the software, is designed by, but not necessarily for, programmers.

The contribution of this study is mainly in two categories: practical and theoretical contributions, with some methodological contributions. The practical contributions are that, through these projects, DHIS-Tanzania and ZALONGWA software were born. The study also contributes knowledge to the client-system infrastructures (HMIS and AREDMIS) that their open source solutions do not conform fully to the open source philosophy, thus they need to fit-in their projects. It also argues that because of the nature of public sector domains, open source development is re-invented which leads to two critical issues of important with open source software namely the freedom to use the software for any purpose and for any number of computers and the freedom to maintain the software without depending on the original author of the software. The study also contribute methodologically that, though dominant conclusions of action researches focuses on the project outcomes; this study uses action research to investigate the process of open source software development hence contributes that not only action research seeks to argue the impact of actions applied but also can be used to reflect on the process of applying actions in the problematic domain.

The two projects presented reflect that the software freedom envisioned by the free software foundation is acceptable but the proposed process of achieving it does not fit in public sector. Open source software developed is in most cases on top of proprietary software a situation which make the public sector to find difficulties to comply with the open source definition. Adaptation and use of open source software in public sector then is borrowing ideas from both free software and open source software but it is difficulty to comply with the two wings.

References

Baskerville, R. L. (1999). "Investigating Information System with Action Research." Communications of The Association for Information Systems [online] 2(19).

Baskerville, R. L. and A. T. Wood-Harper (2002). A critical perspective on action research as a method for information systems research. Qualitative research in information system. D. Avison and M. D. Myers. London, Sage: 129-145.

Bentz, L., M. Boeckman, et al. (2004). Government initiatives regarding open source and their successes and failures, [online] Open Source Development and Documentation Project (OSDDP) http://osddp.org/node/186. [Accessed November 13 2005].

Cardoso, M. (2004). How Do You Say, 'Bye, Microsoft', in Brazil? Brasilia.

Desk, J. N. (2004). Brazil Prepares to Put Open Source Java Into Use. Brasilia.

Dick, B. (2002). Thesis resource paper: you want to do an action research thesis?, [online]. Available from: http://www.scu.edu.au/schools/gcm/ar/art/arthesis.html,. [Accessed November 11 2005].

FSF (2005a). Free software is a matter of liberty not price. You should think of "free" as in "free speech". Free Software Foundation, Inc. [Online]. Available at http://www.fsf.org/ [Accessed 10th November 2005]. 2005.

FSF (2005b). The GNU Operating System, Free Software Foundation, Inc. 2005.

FSF (2005c). GNU General Public License. Boston, Free Software Foundation, Inc. 2005.

Ghosh, R. A., B. Krieger, et al. (2002). Open Source Software in the Public Sector: Policy within the European Union. Free/Libre and Open Source Software: Survey and Study, International Institute of Infonomics, University of Maastricht, The Netherlands.

Isene, G. (2005). Open Souce Software Business Model. Olso, University of Oslo: A lecture on a course INF5750.

Klang, M. (2005). "Free software and open source: The freedom debate and its consequences." First Monday 10(3): 1-15.

Kock, N. F. J. (1997). "Myths in organisational action research: reflections on a study of computer-supported process redesign groups." Organizations and society 4(9): 65-91.

NAGARAJ, S. (2002). Open IT: Government to rewrite source code in Linux. New Delhi.

Nfuka, E. (2004). ICT Developments and Applications at UDSM in the last Ten Years. Proceedings of the 11th Annual Consultative Meeting on the UDSM Transformation Programme, 8th - 9th September 2004. DPD. Dar es Salaam, University of Dar es Salaam: 208-215.

OUT (2005). The Open University of Tanzania Student Information System. Dar es Salaam, Open University of Tanzania. 2005.

Perens, B. (2005). The Open Source Definition, http://perens.com/Articles/OSD.html. 2005.

Raymond, E. (1999). The Cathedral and the Bazaar. Cambridge, Mass, O'Reilly & Associates.

Reid, D. (2005). Open Source. OnlineClick. UK.

Rosen, L. (2004). Open source licensing: Software freedom and intellectual property law. Upper Saddle River, N.J., Prentice Hall.

Schmidt, P. (2005). Implementation and Policy-Making to Optmise Public Access to ICT. Comparison Study of Free/Open Source and Proprietary Software in an Arican Context. Cape Town, Bridges.Org: 52.

Stallman, R. (2002). Free software, free society: Selected essays of Richard M. Stallman. Boston, Free Software Foundation.

UDSM-ARIS (2004). University of Dar Es-salaam ARIS Proposal V2.0. Dar es Salaam, University of Dar es Salaam: 1-6.

Vaughan-Nichols, S. J. (2005). Enterprise War: Microsoft vs Linux - Apps Availability Drives Choice Between Linux and Windows, eWeek.com. 2005.

Weber, S. (2004). The Success of Open Source. United States, Harvard University Press.

Weerawarana, S. and J. Weeratunga (2004). Open Source in Developing Countries. Stockholm, Edita Sverige AB.

Whyte, W. F., D. J. Greenwood, et al. (1991). Participatory action research: through practice to science in social research. Participatory action research. W. F. Whyte. Newbury Park, CA, Sage: 19-55.

Wichmann, T. (2002). Use of Open Source Software in Firms and Public Institutions: Evidence from German, Sweden, and UK. Free/Libre and Open Source Software: Survey and Study. Berlin, International Institute of Infonomics, University of Maastricht, The Netherlands: 64.