USING OPEN SOURCE SOFTWARE IN GOVERNMENT

DISCUSSION DOCUMENT

FEBRUARY 2002

PREFACE

The strength and following of open source software (OSS) is growing. The potential advantages that it is reported to offer led to this investigation. The aim of this document is to possible ways of giving guidance on future use of OSS in government in South Africa. The report covers an investigation conducted during 2001 and 2002. It is intended to convey information and stimulate discussion.

The Work Group was appointed by the Government IT Officers Council (GITOC) in June 2001 to investigate the potential use of OSS in government. It consists of representatives of –

Department of Arts, Culture, Science and Technology

Department of Public Service and Administration

CSIR

State IT Agency (SITA)

Department of Land Affairs

Government Communication and Information System

CONTENTS

LIST OF ABBREVIATIONS

DACST	Department of Arts, Culture, Science and Technology
DPSA	Department of Public Service and Administration
DTI	Department of Trade & Industry
GCIS	Government Communication and Information System
GITO	Government IT Officer
GITOC	Government IT Officers Council
NACI	National Advisory Council on Innovation
OSS	Open source software
SITA	State IT Agency

1 INTRODUCTION

1.1 Background to this study

The quality and utility of OSS is increasing rapidly. In the public service there are several instances where OSS is already being used with success. The time was ripe to investigate the actual and potential use in the South African government. The Government IT Officer's Council (GITOC) concluded that it was the appropriate body to oversee the investigation, hence the formation of a work group.

GITOC's investigation is set in an array of related initiatives extending over many institutions and indeed many countries. That fact simplified the task of the work group of GITO, who could draw on results of other studies and the practical experience of others. The reader will notice that we were able to quote liberally from texts reporting on the work of others.

The study supplements a report on a study by the National Advisory Council on Innovation (NACI), which examines the use of OSS at national level, i.e. in government as well as non-government sectors. GITOC appreciates being able to use some material produced by the NACI project. We trust that some results of our study will in turn enhance the output of the NACI study.

Information is available on the state of play regarding OSS in many other countries and some of it is used in this document.

1.2 Background to OSS

We quote from the NACI study mentioned above[1]:

> "Unlike the physical device that it may lend functionality to, software is a coded embodiment of ideas and knowledge. Like all knowledge (and unlike physical artefacts), to share it is not to lose it, whether or not a fee is involved. Indeed, to share knowledge is to enhance it because it enables others to build further upon it. The principle of open disclosure and rigorous peer scrutiny has underpinned the advancement of human knowledge for centuries.
>
> This is a principle that today underpins the development of **open software** – software that is free of proprietary restrictions (see formal definition below). An intimately related issue in ICT is that of **open standards** for communication. Open software implementations of specified standards are available to anyone without incurring prohibitive licence fees or other proprietary restrictions. The Internet, and its associated applications such as the World Wide Web and e-mail, is the most visible triumph of such openness in ICT.
>
> The other side of the openness coin is that an individual or organisation may not wish to freely disclose software that is commercially strategic or sensitive in some way. This choice has no less merit than the choice to be open. One cannot decree that all software must be open any more than one might insist that all knowledge must be shared freely. Thus, one should have no qualms about the co-existence of open software alongside closed or proprietary software. However, it is necessary to take issue with any attempt to restrict non-proprietary software use or development (through sweeping software patents, say) just as vigorously as it is necessary to defend freedom of expression and the open exchange of ideas.

[1] "Open Software and Open Standards in South Africa – A Critical Issue for Addressing the Digital Divide", www.naci.org.za

1.2.1 Formal Definition and Brief History

1.2.2 Definition

The term "open software" is used as a shorthand for "open source software" (OSS), hence the terms will be used interchangeably[2].

OSS is typically developed through public collaboration, it is available to anyone (usually at little or no cost), it does not require proprietary license fees and it may be freely re-distributed. Users also have access to the human readable version of the software called the "source code", revealing the inner workings of the software and allowing its modification, hence the term "open source"[3]. The use, modification and re-distribution of the source code is governed by rules specified in associated non-proprietary open source licenses.

Access to source code has the potential to empower people in ways that proprietary software simply does not allow. It offers people the freedom to probe, modify, learn from and customise the software to suit their needs. Hence open source software is also known as free software, where "free" refers, first and foremost, to this freedom rather than to "free" in the monetary sense[4].

1.2.3 History

a Origins

Free software has existed since the invention of the first computers. However, its production, distribution and use was limited to a few engineers, scientists and others who had access to the then expensive computing facilities.

More recently, the initial development of the Internet was propelled by the use of open software and the adoption of related open standards. For example, the most basic Internet applications such as e-mail, FTP, Gopher and the World Wide Web all make use of open software for their successful deployment and overall adoption. More than half of all web servers use open software[5].

As increasingly rich standards are being developed to cater for a variety of Web services, it becomes all the more important that associated protocols and tools remain freely accessible if the Web is to retain its non-proprietary pedigree.

b Adoption and Impact

There is a groundswell the world over of adoption of open software by individuals, small and large enterprises, schools and other public and private institutions. This reflects a growing acceptance of and confidence in open software and open standards, and thus contributes to their further development. It is no idle matter that IT industry big names, notably IBM[6], have expressed such confidence and commitment.

Where the open software message has been promoted in South Africa, it has mostly

[2] "Open software" is preferable because it does not make constant reference to the rather technical concept of "source" and it has a symmetry with "open standards"
[3] See http://www.opensource.org for a full definition.
[4] See http://www.gnu.org/ for fuller discussion.
[5] Netcraft's survey http://www.netcraft.com/survey shows that some 60% of web sites use the open source Apache Web server
[6] "We believe very, very strongly that open standards and open-source software are absolutely critical foundations for the IT business going into the future." IBM, http://www.ibm.com/news/us/2001/08/15.html

been by individuals in academia, NGOs and small software companies. It is timely for such bottom-up initiatives to be complemented by initiatives led by government policy and action.

Indeed, many governments are now developing national policies to promote the use of open software. Examples include China, Thailand, Brazil, Argentina, Germany, France and the United Kingdom. Information technology professionals are also encouraging their governments to adopt policies that support open software, New Zealand being a notable case in point[7]. Some of these countries recommend that software used by government and its associated agencies must be open software, unless proprietary software is the only available option.

The initiatives in Germany, France and the UK are a result of, or are closely allied to, the European Commission initiative "eEurope – An Information Society for all". The initiative's action plan set the target: "During 2001 the European Commission and Member States will promote the use of open source software in the public sector and e-government best practice through exchange of experiences across the Union".

1.3 Situation in other countries

1.3.1 European Union

From the EU (IDA)[8] report we glean the following information:

a Global use of Open Source (private and public sectors)

ai The Internet connected hosts:

About 61% of the active IP sites have Open Source web servers and about 40% have Open Source operating systems. Without any contestation, the Internet (web servers, and now application servers, tomorrow office servers etc.) is the main demonstration area of the success of the Open Source development model.

Of course, the IP connected hosts represent only a **limited part** (10%) of the global server market: most of the enterprise and public sector servers are LAN or Intranet servers, that cannot be counted automatically, and do not generally use Open Source operating systems.

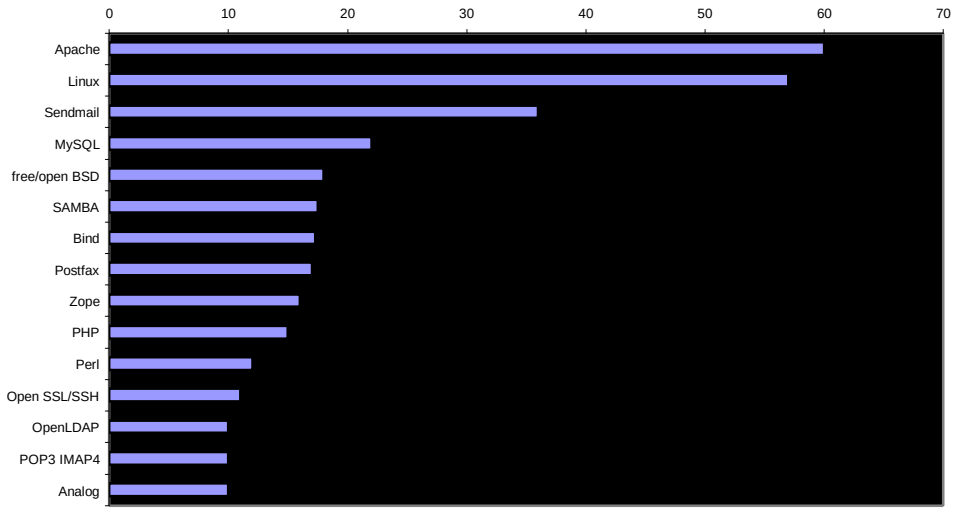
In April 1999 already (year of the strongest Linux growth) the six countries observed in the present study, (taking in account the predominant part of Germany) had 46% Linux & BSD operating systems on the Internet connected hosts (counted by Ripe)[9].

aii The most used OSS on servers

The report later goes into further detail:

The duo "Apache/Linux" is cited first. The Apache web server is even more used than Linux (Apache runs also on other platforms), with a rate of 60% in our public sector panel - corresponding to the global market share (61% in 2001 according

[7] See http://www.openz.org/
[8] European Commission, Study into the use of Open Source Software in the Public Sector, June 2001
[9] Analysis based on http://leb.net/hzo/ioscount/



Netcraft).

According to the fact that 63% of our panel use some kind of OSS solution, the conclusion is that nearly all public sector OSS users run Apache somewhere.

Linux comes after (55% against 18% for BSD variants).

MySQL seems to be the most used database.

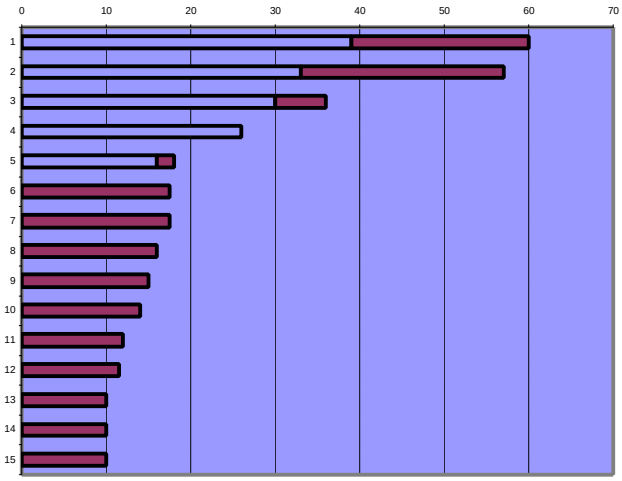
The use of Zope is growing in several new projects (in France, Spain, Germany).

The most used solutions on desktop

If for the servers, some solutions (Apache, Samba) are now really considered as well established “market standard”, the desktop situation is quite different: the current use of OSS is still low and the situation may change rapidly, with the emergence of new products (especially office suites). Contrary to the server solutions, desktop products are also more subject to criticism: some solutions are missing, have insufficient functionalities, or are “too heavy”, or encountering interoperability problems…

Linux of course, the Mozilla-Netscape browser and the KDE desktop graphical user interface, are currently the most used solutions in a desktop environment.

Concerning the office suites, StarOffice, released as Open Source by Sun in October 2000, starts now to find its developer’s community and is currently the most used. This is mainly because it requires minimal additional training, compared to its proprietary rival, and because it is the most interoperable solution, running also on multiple platforms (including Windows and Linux). The main general reproach against StarOffice is that it is still quite “heavy” and resource consuming, with all components loaded at the same time.



(In % citation)

1. Linux (OS)
2. StarOffice Suite
3. KDE (GUI)
4. Netscape (Browser)
5. Gnome (GUI)

The main issue regarding "local desktop running office suites" is that, generally, their interoperability is not measured according to a neutral and published standard, but versus the dominant Microsoft Office suite.

This measure corresponds to the "user" point of view: importing a document from Office 2000 for example, modifying it, then re-exporting it to Office 2000 and see if everything works.

A measure example was given by "01 informatique" in May 2001[10]:

MS Office Interoperability	
Suite	**Rating out of 10**
StarOffice	7,2
WordPerfect Office	6,2
SmartSuite	6

The main encountered problems were:

- Especially from MS-Excel, many elements cannot be imported /exported: VBA (Visual basic for application) macros, Cross dynamic sheets, Validation fields, List controlled fields etc.

- From MS-Word, VBA again and problems with tables, table of content, index and illustrations.

1.3.2 Support by major players

OSS's credibility has been established as market giants such as HP, IBM and Sun have thrown their weight behind it.

[10] 01 informatique, 4 May 2001p.34 – "Suites bureautiques: des alternatives à Office" by Ludovic Arbelet

1.4 Situation in South Africa

1.4.1 Usage in the public sector at large in South Africa

There are certain pockets of vigorous OSS use in government. This includes network/internet software, as well as pilot implementation of other packages in certain departments such as the DPSA.

As part of the work group's investigation, the CSIR embarked on a survey of actual and potential use of OSS in the South African government in general. In many organisations the survey encountered a low level of knowledge and understanding of OSS, which inhibited in-depth questioning on potential scope for application of OSS in the particular environment.

This low level of knowledge greatly influence the choice of approach for the future.

1.4.2 Usage in the private sector

There are prominent private sector companies that use OSS extensively. Full analysis of the situation could not be completed in time for this report.

1.4.3 User groups

Some groups have been established in South Africa and offer support to potential new users.

1.5 The future of OSS

1.5.1 A view from the North: The UK

Within five years, 50% of the volume of the software infrastructure market could be taken by OSS. We expect that OSS's position in the small server market (file and print servers and Web servers) will grow fastest.

OSS's position in large servers (e.g. those managing massive multi-user databases), such as those that underpin many large Government procurements, will grow from its current position of near zero penetration, to a position where OSS is a viable option, within 2 - 3 years.

Within the developed world, we as yet see no sign that OSS will become a viable alternative to Microsoft Windows, for users' (general purpose) desktop machines in the corporate or home PC markets. However, OSS on the desktop may soon become a significant player on the desktop in the developing world. For these reasons the study recommends against any preference for OSS on the desktop, but also recommends that this issue be reassessed by the end of 2002, by which time early trials of the use of OSS desktops may have generated sufficient evidence to warrant a reassessment.

OSS is already suited to restricted functionality desktops, such as those used in industry for point-of-sales and point-of-service terminals; and in these areas OSS's market share is likely to grow significantly.

We expect OSS to rapidly become the market leader in consumer computing devices.

We expect the market for new portable and consumer computing devices (such as set-top boxes and smart mobile phones) to remain very dynamic, with no dominant market leader emerging. OSS is however likely to be a significant player in this market.

We expect that the software infrastructure that is implemented on top of operating systems (so-called middleware) will move gradually from proprietary products towards OSS.

All of the above predictions relate to the software infrastructure market. As yet, it is not possible to predict that, within the developed world, OSS will make such a major contribution to the software applications market. There are a few OSS applications that are becoming significant, but it is too early to say if a trend is developing.

The software industry is very fast moving, and frequently throws up promising new developments that initially promise to make great changes in the marketplace, but which ultimately fail to live up to their initial press hype. Our first key conclusion is that OSS is indeed the start of a fundamental change in the software infrastructure marketplace, and is not a hype bubble that will burst. This is perhaps surprising because OSS does at first sight appear to be a bit of a paradox.

1.5.2 Another view from the North: European Commission[11]

The European Commission describes three possible scenarios and goes on to discuss them:

No action is taken by the European Commission and governments.

Limited support is granted by these institutions to open source software.

``Aggressive'' support to open source software is provided by these institutions.

a No action scenario

Up to the present moment, almost no explicit support has been granted by governments in any part of the world to open source development. Therefore, this scenario assumes that things are going to progress following the same path.

Our consideration in this respect is that currently open source systems are already a noticeable part of our computing environment, especially in the server and development areas, where they are already playing an important role in the information technology infrastructure. As companies are discovering, it is possible to live off open source.

And as the Red Hat and VA Research IPOs have showed [15], the investors are already betting with their money on their capability to succeed.

If no action is taken by the European Commission and other governments, we can expect that only private companies and individuals will try to walk this route. Since the movement is already proving to be self-sustaining in both the economic and the technical field, and it is competing head to head with the leaders in several market niches, it seems unreasonable to doubt its future health.

Probably open source software will be used as a standard part of the infrastructure by most governments, without the need of any specific endorsement. Instead of buying some proprietary systems, some government offices will decide to buy open source systems and servers. In fact, this trend is already starting to take shape in several countries (at least in Europe). Open source will be preferred in many situations simply on economic or technical grounds, as a lower cost or technically better alternative to proprietary systems. In this case no specific benefit will be reaped by governments, or by the societies they serve, except for the direct cost savings on licences.

[11] European Commission, Free Software / Open Source: Information Society Opportunities for Europe?, April 200, http://eu.conecta.it/paper/

In this scenario (and in part also in the others), some threats for the open source movements will also have to be dealt with. Among them, some which are already being discussed within the open source community are:

FUD (fear, uncertainty, doubt) techniques, used by companies or organizations committed to proprietary software, afraid of losing their market share or supremacy. Until now those techniques have not been a real problem, but the situation could change in the future, as more and more resources are devoted to this effort.

Dissolution, due to systems and licenses which can be confused with open source software, causing divisions in the community and in the code base, and in the long term, the loss of some of the advantages of the open source model.

Ignorance, or loss of global vision, by the people composing the open source community. As more and more people come to use and produce open source software, there is the risk that a significant part of them don´t really understand how open source really works, and all the lessons learned by the community over the last few decades. This could lead to the conversion of open source in just another buzzword, without any real meaning for many people, and therefore to the loss of the real advantages of the model.

Legal impediments. The legal framework could make more difficult, or even impossible, the progress of the open source movement. The issue of software patents (discussed in subsection 6.2) is one of the more obvious legal impediments that could be found in the next years

Some of these problems could be alleviated with the help of responsive governments, as will be discussed later.

b Limited support scenario

In a limited support scenario, probably it makes most sense to ask how society, administrations and governments in general, and the European Commission in particular, can benefit from open source software, and how they can use it for their own advantage, rather than the other way around (how can the open source movement benefit from governmental support).

In this case, they will probably invest some time and resources to assess the feasibility of open source software in their areas of interest, and to identify the barriers which could impede their adoption of open source technologies. With time, they will identify some strategic open source projects where they want to contribute, and will understand more clearly some of the benefits in terms of flexibility, usefulness in the entire life cycle, and adaptability. They may be interested in trying open source software for several (especially mission critical) components of their infrastructure.

In the long term, this will give results, especially in the form of greater acceptance of open source software in society in general, because of the amplifier effect that its use in governments has on society. If they finally recognize the benefits of the open source model, probably they will also help to overcome the future problems (specially those related to the legal framework) that have been described.

The recent case of the German government funding the development of GPG (GNU Privacy Guard) because it is found to be of great benefit for society shows how this scenario may develop in the short to medium term.

c Aggressive scenario

We are completely unable to predict what exactly `aggressive' may mean, because legislation may preclude some specific kinds of action, and because what today may seem `aggressive', tomorrow can be considered as just common sense and normal business practice.

As an example, we can think of legislative actions on the part of the European Commission

and the national governments to give preference to open source solutions whenever they are technically feasible. Another interesting action could be the active promotion (by direct or indirect funding) of the development of open source alternatives to proprietary systems in those areas where it is identified that this is convenient (because of strategic, social or economic reasons). This would create an enormous market for open source consulting and solutions, improve significantly the skills of the European information technology work force, and probably increase the usefulness of information technology systems. Also, these measures should have some measurable impact on the import/export balance for information technology products, currently very biased against Europe because the majority of widely distributed (usually shrinkwrapped) software systems come from the United States.

In fact, the whole matter of the level of support that open source deserves is mainly a matter of betting on its future. If open source software is really going to change the whole landscape of the information technology industry, the support that Europe (or any other country) gives to it can only be transformed into more benefits. If open source is not a passing fashion, but is here to stay, the impact could be similar to that of Internet technologies during the last decade. In this case, the more radically that a society adopts it as a technological enabler, the more benefits that society will get from it. In our opinion, if the open source community becomes strong in any given area of the world, that area has a far greater possibility of competing in a software market with changing rules, and the society in that area can benefit earlier from reduced costs, greater economic activity, and widespread diffusion of new technologies.

1.6 Advantages and disadvantages

1.6.1 The UK analysis

a Advantages to software developers

"This section goes a long way to explaining why Open Source has been so successful. Starting with the advantages to software developers:

Open Source software has a unique advantage in "crossing the chasm" . The reason that products drop into the chasm (i.e. they fail to establish a sustainable market share) is that companies choose to cut their losses rather than keeping on funding the product in the hope that it achieves acceptance in a niche or mainstream market. Proprietary developers solve this problem by having the resources and management commitment to continue pushing products that they believe in (e.g. Microsoft with Windows and Windows NT) for as long as it takes for them to take off. Open Source solves this by having a zero cost base - so running out of money is not a problem - as long as the group of developers maintain their interest they cn keep on going;

A second major issue in crossing the chasm is the need to offer customers a total solution to their IT problems, not just a component of the solution which the customer has to integrate themselves. Typically, a company trying to get a commercial product across the chasm needs to recruit a significant number of service providers who perform the necessary integration for the customers. A useful analogy here would be that the manufacturers of central heating boilers do not, in general, sell directly to customers - customers buy the boilers recommended by the heating engineers who are installing or upgrading their central heating systems. Service providers will like the fact that Open Source software, such as Linux, is free (helping their margins); that it can be tailored to the requirements of the installation; that it is very reliable; and that if it goes wrong they are not dependent on the manufacturer to fix it;

The ability of users to deploy the software without having to sign licenses, or make

financial cases to their management, aids initial take up;

Open Source developers have access to the existing body of Open Source software to include in their programs (the Open Source community have been one of the first communities to be able to exploit the potential offered by widespread software reuse);

The Open Source community attracts very bright, very motivated developers, who although frequently unpaid, are often very disciplined. In addition, these developers are not part of corporate cultures where the best route to large salaries is to move into management, hence some Open Source developers are amongst the most experienced in the industry. In addition all users of Open Source "products" have access to the source code and debugging tools, and hence often suggest both bug fixes and enhancements as actual changes to the source code. Consequently the quality of software produced by the Open Source community sometimes exceeds that produced by purely commercial organisations;

The size of the Open Source developer community is very large. We would estimate that there are many tens of thousands of active Open Source software developers; hundreds of thousands of active beta testers; and a non-commercial user base of about 5 million Open Source supporters (and many more millions of users). As we have said earlier the size of the development community is growing all the time. A successful Open Source project can attract (and use effectively) a size and quality of developer and testing community that no company (even giants like Microsoft or IBM) can hope to match. An example of this effect was that when Oracle announced in July 1998 that they were going to port their database to Linux, they had 20,000 developers sign up to Oracle's development programme - an unprecedented amount of interest for a future port of the Oracle RDBMS;

Open Source developers are not constrained by corporate product development processes or ISO 9001-style software development and Quality Assurance processes. Although some Open Source software is unreliable, many of the most popular OSS products have a rate of evolution, robustness and responsiveness to bug reports that much commercial software can only dream of;

Open Source is an obvious subject for projects in Universities and research institutes. There is growing interest amongst Governments in using Open Source as a mechanism for exploiting research results. The Research Community gives Open Source developers free access to a large community of the brightest and freshest minds. This provides a major (free) source of analysis and incremental enhancements to Open Source developers;

Many students get involved with Open Source software at university and when they go into industry and obtain positions of power they have a natural tendency to favour the software they worked on in their student days. This effect is widely credited with being one of the reasons why UNIX first made an impact in the enterprise market. It would also help explain the current surge of commercial interest in Linux.

b Disadvantages to software developers

The disadvantages to software developers who produce software within the Open Source model are:

Commercial companies have to find new places to make money - e.g. selling services or books. For some markets (e.g. games) it is difficult to see where commercial companies can "make a buck". At the current time investors do not really understand the Open Source model. The possible relationships between the Open Source culture (motivated by individuals desire for recognition) and

commercial models (where there will be a profit motive) are still developing and are consequently unpredictable;

There is no marketing budget to push the product;

There is no funded product development budget. In practice, this means that Open Source software "products" tend to get ease-of-use features and user-oriented documentation significantly later in their lifecycle than commercial products. For example, early Linux distributions attracted justifiable criticism for the complexity and lack of standardisation of their installation and management processes;

Open Source developers tend to be very passionate about technical issues. Consequently, without a project leader with good people skills an Open Source project can break up in acrimony.

c The advantages to users of Open Source software

Although, in reality, software licensing costs are not a major part of the Total Cost of Ownership (TCO) of IT, the attraction of the software being free should not be underestimated. A corporate license for even cheap software can, in absolute (not relative) terms be serious money, and niche software can be dauntingly expensive. In addition, users are not faced with user-unfriendly license managers that can lock out users when more than the licensed number of users try to use a product simultaneously. Users do not face costly management of licenses - including: the legal costs and risks of checking and signing licenses, ensuring that license conditions are adhered to, and ensuring that all relevant licenses have been purchased and are up to date. Users are also not locked into having to buy future upgrades;

The market greatly values robustness, and the Open Source model, particularly as practiced by Linux, encourages a large market of early adopters (compared to the size of the early market for commercial products) who actively help debug the software. Consequently much Open Source software becomes highly robust at a surprisingly early stage of its development, and mature Open Source products are setting new industry standards for bulletproofness;

Paradoxically, the evolution of Open Source software can often be much more responsive to user requirements than commercial software. The upgrade cycle for Open Source is usually much faster than the typical 12 - 18 month cycle of commercial products. In addition Open Source products such as Linux often provide the most rapid turnaround on urgent issues, such as patches against newly found vulnerabilities to external attack;

Open Source software tends to be written portably and hence is available on a wide range of platforms. In addition, because the source code is openly available, people interested in availability on another platform can do the port themselves or pay someone else to do it. As a result, Open Source allows a wider choice of computing platforms and potentially easier upgrade to new technology;

Open Source allows the user (and their service providers) to control vulnerabilities themselves.

Open Source software tends to be free of dependency on related products. Purchasers often perceive that the product works best with other products from the same manufacturer. Open Source software offers its users greater freedom to purchase other products, avoiding lock-in to particular manufacturers;

Open Source software means that there is no single proprietary source of software support and upgrades. This has a double advantage - firstly, there is no risk that the one company that supports the software stops supporting it or goes out of business;

secondly, there can be a competitive market in companies offering support services (reducing cost and increasing quality of service). Users also have access to an Internet community, which includes both the developers and users of the software, so that in-depth advice (and possibly source code fixes) can often be obtained rapidly, and at no charge. An additional advantage is that often there are many people available for recruiting who are capable of supporting the software. In addition the source code is available to system integrators making system integration much easier and cheaper than having to rely on the originator to make cosmetic or interfacing changes;

Companies developing in-house applications on Linux, and service providers who use Linux, will like the fact that if they hit a bug in Linux there is a possibility that they can go into the source code and fix it. Encountering a bug in a proprietary operating system can stop a project or service supplier dead in their tracks;

Commercial software may become unmaintainable once its originators leave the company. By comparison, Open Source software often maintains a vibrant life for much longer as it is, in effect, the property of a community. This property is enhanced by the fact that Open Source software is often better structured and with better program documentation than commercial software - after all, everyone can see an Open Source developer's code so personal pride (and the need to maintain the respect of one's peers) usually ensures it looks pretty. Indeed large, geographically dispersed teams can only work well if the software design is highly modular.

d The disadvantages to users of Open Source software

There is no single organisation with a vested interest in supporting it;

As mentioned above, ease-of-use features tend to arrive later than for commercial products;

There are lots of negative perceptions that Open Source still has to overcome (note: in our opinion most of these perceptions have little actual substance, but it will require lots of additional publicity about deployment of Open Source in large, respected companies to overcome them). These perceptions include:

Senior managers in companies are likely to equate "free" with "unreliable";

There is no commercial organisation who you can sue if something goes wrong;

Because I do not pay the software developers I have no control over them;

Because the developers are motivated by recognition rather than money, they are unpredictable, for example they might all rush off and work on a new, more exciting Open Source project;

Open Source developers will not understand commercial imperatives like backwards compatibility, and the need for interoperability.

1.6.2 NACIs view:

General comparison of advantages and disadvantages, with conclusions regarding the merit of pursuing the use of OSS further.

The NACI report argues as follows on strengths and weaknesses of OSS:

… many people now believe that the future impact of open source software in the ICT industry and in society in general will be so huge that the current rules by which the software industry behaves will completely change. But what precisely are the

strengths and weaknesses of the open software model?

As pointed out in [12], software falls into two broad categories:

Software infrastructure - the plumbing of ICT systems and the Internet - includes operating systems, databases, Web servers and other components that enable software applications to run. This category is taken to include Web and system middleware - the increasingly important intermediate layer between low level infrastructure (operating systems) and user level applications.

Software applications, including generic desktop business applications such as word processors, spreadsheets, financial and management systems, as well as various other (often highly specialised) applications that an organisation may need to run its business.

There has been significant penetration of open software in the infrastructure category. Reference [12] comments that "OSS is indeed the start of a fundamental change in the software infrastructure marketplace, and is not a hype bubble that will burst". The authors proceed to predict that "within five years, 50% of the volume of the software infrastructure market could be taken by OSS".

Open software has not yet had a significant impact in the generic desktop applications category, where Microsoft Office, running on the Microsoft Windows operating system, is the *de facto* standard.

Nevertheless, there are already many open source office suites, such as KOffice, Gnome Office and OpenOffice/StarOffice, which already offer equivalent functionality and ease of use. Appendix A in [12] gives a detailed discussion of OSS alternatives to MS Office. Hence the significant obstacle to adoption of OSS desktops is neither technical nor is it any longer a matter of ease of use - it largely has to do with familiarity and to some extent compatibility.

It is not possible to ensure full and sustained compatibility with proprietary MS Office document formats that evolve from time to time. The matter of compatibility needs to be embedded in the wider context of interoperability of different systems across a network. Proprietary document formats impede public access to online government documents.

This strongly suggests a commitment to open standards for interoperability in government use, together with a commitment to the use of non-proprietary formats for document exchange. Indeed, this is a fundamental recommendation of this document.

Familiarity breeds resistance, at least for the user who is not too disaffected by the current offering. Newcomers to computers – students and others – will be more receptive to a different, open model. Furthermore, because the source code is available, the OSS desktop can be freely customised to suit local needs. An obvious candidate is support for local languages. This alone makes a compelling case for the OSS desktop in South Africa. Accordingly, it would make sense to pilot the OSS desktop in various public sector institutions as a possible precursor to more widespread use.

The language support problem does not arise in a country that is entirely English speaking (or other "mainstream" language officially supported by MS Office or other packaged commercial software application). Furthermore, in a developed country, familiarity with MS Office is more entrenched than in a developing country. Against

[12] "Analysis of the Impact of Open Source Software", http://www.govtalk.gov.uk/library

this background, [12] recommends against the OSS desktop in the UK, subject to reassessment by the end of 2002, but adds: “However, OSS on the desktop may soon become a significant player on the desktop in the developing world”. That said, the OSS desktop is being adopted or piloted by some institutions in the developed world. For example, the US Defence Information Agency recently committed to StarOffice on the desktop[13].

In the light of the preceding discussion, another fundamental recommendation is that government ICT procurement policy should allow open software solutions to compete “on a level playing field” with proprietary software - at both the infrastructure and the application level. Clearly, if there are no established open software alternatives (such as certain business process related solutions) the issue of choice does not arise."

2 IMPLEMENTATION

2.1 Security

The UK report goes into a detailed analysis of security aspects and the security record of both proprietary and open source software applications. The final conclusion that it comes to is "that there is no great security benefit or disbenefit between proprietary and OSS software. Issues such as properly designed, and rigorously maintained security architectures are much more important than the choice between OSS and proprietary systems".

We therefore conclude that security considerations should not play an overriding role in the South African government's policy decisions on implementing OSS.

2.2 Categories of applications

OSS has become a popular and viable choice in the software infrastructure arena. We have seen above that Linux is becoming a prominent server operating system. For some applications, such as storing Web Pages, OSS software has become a market leader.

We have already said that we think that the current immaturity of OSS on the desktop means that there is a clear reason to not express any preference for OSS on the desktop at present.

2.3 Issues of interface and interoperability

Base on the UK Report:

"We would advocate that … standards should be openly defined standards as far as practical (as is currently the case for the e-GIF), rather than proprietary ones. This is a weaker recommendation than mandating Open Source, as proprietary implementations of the standards would conform. In many cases (e.g. jpeg or png image formats), an Open Source reference implementation is available, which has helped the dissemination and uptake of the standard. The use of proprietary *standards* (for example, the Microsoft Office file formats) locks in dependency on proprietary implementations, and would limit the choices in all connected components. If the interchange standards between components are open, it is much easier to replace components, allowing stronger competition as components and the business processes they serve advance. The components may be very large, in this vision something as large as the whole of the tax system might be a component.

[13] See “http://www.zdnet.com/zdnn/stories/news/0,4586,2781914,00.html”

2.4 Research and development

2.4.1 Software developed by government contractors

The NACI report states that, as also suggested by the UK Report, it is reasonable for government to seek full rights to bespoke software rather than allow the contractor to claim full rights. Where appropriate, government can then choose to release the software under an open source license and hence open the contractor's work to general scrutiny. Maintenance and upgrade might then be conducted by different contractors and hence minimise supplier lock-in.

2.4.2 Research in academic institutions

NACI states that groundbreaking software often originates in academic research. In the section "Improving the competitiveness of UK industry", [12] notes that "Open source has been the *de-facto* standard for the exploitation of academic software in the US for many years. It is hard to over-state the beneficial effect that this has had on the technology and the wider computer industry." It recommends that open source software should be the "default exploitation route for Government R&D software" in the UK.

This is a well-founded recommendation, one that South Africa also ought to adopt for government funded software R&D conducted at universities and research councils. Such a move would be in the interest of South Africa's industrial competitiveness.

It may make sense for government agencies to develop software as open software from the outset, and take best advantage of the open software development model of voluntary collaboration.

This NACI view could be made applicable to most central and provincial government departments.

2.5 Cost

Whereas there are specific items such as license fees where significant savings can be effected through OSS usage, the variety of environments within which OSS can be introduced is considerable. This study can not yet come to a conclusion as to which of the environments offer distinct cost benefits.

An important factor is however the fact that usage of OSS will lead to a greater portion of funds spent on IT remaining inside the country,

2.6 CONCLUSIONS

1. OSS is a significant and growing phenomenon within many governments. It should not be ignored.
2. In certain areas of application OSS has reached a high level of maturity. Its use can safely be allowed, even encouraged, and it has the potential of generating significant efficiencies.
3. In other areas the performance of OSS has not been proved to be superior to proprietary software.
4. Academic institutions are often enthusiastic, successful champions of OSS. Partnerships between them and government institutions may be beneficial.
5. Challenges regarding orderly, responsible implementation, such as security, interoperability, and providing user support are generally not unrealistically

high and should not in themselves constitute reasons for not choosing OSS.

6. The level of knowledge and understanding of OSS in the South African government is currently inconsistent. Pressure to opt for OSS may be inappropriate.
7. The UK policy quoted below can serve as a legitimate vision for OSS usage in the South African government.

3 A possible vision

The proposed policy in the UK may be a suitable vision for GITOC:

Government will consider OSS solutions alongside proprietary ones in IT procurements. Contracts will be awarded on a value for money basis.

Government will only use products for interoperability that support open source standards and specifications in all future IT developments.

Government will seek to avoid lock-in to proprietary IT products and services.

Government will obtain full rights to bespoke software code that it procurers and all custodians of commercial off the shelf packages that it uses wherever this achieves value for money.

Government will explore further the possibilities of using OSS as the default exploitation route for government-funded R&D software by academic research institutions.

This may represent a sound guideline for an eventual South African OSS policy. Work undertaken by GITOC should be aimed at positioning the South African government to be able to adopt such a policy. Some of the issues that need to be addressed in this process are discussed below.

4 Measures to promote usage of OSS

At the forthcoming strategy workshop, discussion on the issues below will be welcomed. Based on the discussion the work group could then proceed to formulate definite proposals for adoption by GITOC soon thereafter.

4.1 Active support by SITA

Strategies implemented by SITA for promoting the greater deployment of OSS in government.

1. SITA is committed to play an active role in ensuring continuity in OSS Research and Development through its current Technology Assessment Unit.
2. Debates central to open source software are amongst those prioritised within SITA's Institute of Electronic government solutions

4.2 Regulatory steps

Possible adaptation of tender procedures to favour tenderers that support/use OSS.

4.3 Amending criteria for incentive/support schemes for innovation

Providing easier access to financial support from schemes such as those

administered by DACST and DTI.

4.4 On-going monitoring of OSS developments globally

The rate of development is fast and new opportunities for using OSS emerge regularly. GITOC should be in a position to keep track of developments.

4.5 Monitoring and reporting on OSS implementation in government in South Africa

GITOC should keep track of OSS applications in various segments of government and ensure sharing of learning.

4.6 Generating more knowledge and understanding of OSS in government

Before a firm policy can be meaningfully implemented, wider knowledge and understanding should possibly be generated. A concerted effort should be made.

4.7 Partnerships between government and institutions from other sectors

The academic and private sectors are applying OSS in many ways. Productive partnerships should be explored.

4.8 Developing OSS advice and support capacity

Such capacity will be crucial for the maximum exploitation of OSS's potential. Issues to be considered include whether government should develop the capacity on its own or in partnership with other sectors.

4.9 OSS in the context of NEPAD

The possibility of contributing to NEPAD through promoting OSS in other partner countries should be explored.

5 Way forward

It is anticipated that the issues in par 4 above will be enriched through the GITOC discussion. The work group can then be tasked to develop proposals on how to tackle each of the issues that are raised.

oss report/2002-09-16/12:20:06/DACST