

Using Free and Open Source Software

(FOSS)

in government, military and business projects

Prepared by :



&



Overview

Concept

The Concept of using "Free and Open Source Software V. Proprietary Software Raises a host of questions from cost to migration options, from compliance to security. The following is a our analysis or feasibility study regarding exploiting closed source software platform vs open source software. This study presents general overview of the open source platforms and current tendencies using Free and Open Source Software (FOSS). This report is focused or referenced on government and military simply because their references are available, while internal corporate policy is often unavailable for public scrutiny. The justification for FOSS in the military is clearly transferable to adoption in the private sector.

Summary

Free and open source software gives the user the freedom to use, copy, distribute, examine, change and improve the software. These rights are stipulated in licenses. Development takes place within the form of projects, and those who participate are mostly professional software developers who either take part on paid time with the permission of their employer, or on their own time. Many large suppliers (IBM, HP, CISCO, Sun, etc) use free and open source software in their own products and solutions and also contribute with their own resources in the various projects and free distributions.

Free and open source software entails a new kind of competition, separated from that of traditional business in that the product generally is not owned by any single company and therefore cannot be purchased of the market. Furthermore, the software itself is not constricted by any cost or fee. It can be obtained free of charge on the Internet. The conclusion of the working group is that free and open source software in many ways, both functionally and qualitatively, is quite equivalent to – or better than – proprietary products.

Free and open source software should therefore be judged on an even par with proprietary software in a procurement process in order to establish better market competition. It is also necessary to place demands on open standards and file formats in order to achieve interoperability between different systems. Proprietary

software is mainly sold nowadays without any guarantees, and producers exclude responsibility for, more or less, all errors and bugs in the product.

The producer dictates conditions for changes and corrections, and the lifetime of a product is usually short before new versions are released. If the software producer goes out of business, all development and support disappear and the user has to look for alternative solutions. With free and open source software these dependencies can be avoided.

The concept of free and open source software has existed for almost twenty years, and Linux has been around for about ten. Many of today's Internet functions are almost completely based on free and open source software, functions such as e-mail and the translation of computer names to IP addresses, i.e. DNS. More than 65% of all web servers on the Internet are based on open source software.



Conclusion

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

- increased competitiveness
- improved interoperability
- reduced costs

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

FOSS

What is Free and Open Source Software

One could illustrate the idea of free and open source software compared to proprietary software with a situation where a person buys a house. Upon purchasing the house he also receives technical drawings and blueprints so that he can make changes himself to the structure. One alternative would be for the seller to retain ownership. All drawings would remain in his possession and therefore all changes to the house would have to be made by the same vendor or seller.

Another case could be the purchase of a car, where only the dealer's repair shop retains all car manuals and repair guides for exclusive rights to repair or make any changes in the performance of the car.

Richard M Stallman, founder of Free Software Foundation, uses the illustration of cookery and food. How would we experience the world around us if recipes were not freely available or free to change and modify? What would it be like if we committed a crime every time we made a copy of the recipe or gave it to someone else?

Free and open source software means that source code is freely accessible, that the software can be freely used, changed, improved, copied and distributed by all who wish to do so.

Free and open source software does not have to be free of charge. Besides being able to construct business models around the software based on commercial aspects, a company can receive direct payment by the use of a large number of licensing schemes and models. These models can also be included in the overall definition of what we mean by free and open source software. What is important here is that the source code is available to the customer.

FOSS background

When Unix was introduced it was free of charge for use in university and academic context. Since much of the development of Internet technologies took place within the walls of universities and research facilities, Unix became the operating system that was used for this development.

It was commonplace for application developers to share and distribute source code between each other. At the beginning of the eighties, licensing conditions were changed for Unix. Prices increased and the possibilities for sharing source code were restricted substantially.

In 1984 Richard M. Stallman started a project called GNU with an idea of recreating the open environment, which he had experienced as a member of the application development team at the Massachusetts Institute of Technology (MIT). But this time his ambition was greater. He wanted to create an operating system, which was completely open and free. He also created a special type of software license called the GNU license (GNU GPL – General Public License) in order to guarantee that software developed within the GNU project would continue to be open and free for all to use.

GNU GPL became the object of wide attention, but not everyone was attracted to the puristic and ideological spirit in which Stallman worked. Bruce Perens, together with Eric S. Raymond, started therefore the Open Source Initiative (OSI) in 1997. They came with their own definition of open source code, "The Open Source Definition". All licenses that hold themselves to the guidelines set forth in the definition can be called "Open Source Software". The main difference between GNU GPL and OSI is that OSI allows commercial use and sales of the software.

FOSS licensing

Proprietary software is sold without any access to source code and is therefore not possible to change, improve or further distribute. A license for proprietary software entitles a user only to use the software under certain conditions.

Free and open source software gives the user the right to use, copy, distribute, examine, change and improve the software. These rights are stipulated in the licenses, which apply to free and open source software. GNU GPL is the most widespread license, but also the one which places the most demands on openness and lack of restrictions.

The licensing model in GPL is very much different than other software licenses, mainly due to the intentions of the licensing agreement and its protective clauses. GNU GPL is based on the freedom to make an agreement or contract and that the originator has the right to make the software available on his own conditions. Copyleft is a mechanism, in among others GPL licenses, which stipulates that free software remains free, even when modified or changed. Copyleft does not allow someone



who further distributes the software, in original or modified form, to add any restrictions to the license. This means that all copies of the software, even modified, continue to be free.

It is this mechanism, which precludes reuse of GPL licensed software in proprietary software that Microsoft has heatedly opposed.

Besides GPL there are 42 other OSI approved licenses for open and free software.

FOSS motivation

There are several studies about how free and open source software is developed, who takes part in the development and their motives for developing this kind of software.

One observation made is the high level of creativity in development projects for free and open source software. Another observation is that most of those taking part in projects are themselves professional developers who either take part on their own time or with the permission of their employer on paid, company time.

Characteristic for those taking part is a strong feeling of solidarity with the developer community. Intellectual stimulation and the chance to develop one's own competency is also an important impetus. Participating members of the community want to learn from each other by contributing themselves and being able to learn from the knowledge of other developers.

Professional developers, who create both proprietary and free and open software, often prefer the latter model because of more room for creativity. There are also not the same time restraints with scheduled deadlines. One often receives quick feedback on one's contributions. It is also important to "be seen". This also contributes to the wish to arrive at a good result. In a closed environment, where an individual programmers work is not noticed in the same way, commitment on the part of the programmer can slack and can even contribute to mediocre program code.

A typical community consists of a core group of dedicated and knowledgeable individuals, sometimes taking part on their own initiative and sometimes through an elective process. This group structures the on-going work and chooses the best contributions. Anyone can submit a contribution, both companies and individuals, but it is the core group which decides the design of, for example, system architecture and also which contributions are to be used. Linus Torvalds is an example of a very successful "leader" for such a community – the development of Linux.

Thus, the community is responsible for both development, further development and maintenance of the product. Almost all development projects put up their own portal on the Internet, a web site for the project, where others outside the community can follow ongoing development, read documentation, download new versions (both stable versions for use and beta versions for testing), comment on ongoing work and contribute themselves to the project.

FOSS examples

There are a large number of products based on free and open software. At one of the most popular portals for the publication of free and open source software there are over 60,000 different projects of which 7,000 are adapted to various operating systems.

The most popular and successful projects also have their own web sites.

Linux is probably the most well-known example of open source software, but there are also a large number of other successful software products. A few examples:

Apache is one of the world's most utilized web servers. Apache is used in more than 65% of all web servers on the Internet today.

Bind is, without question, the most widely used name server (DNS) on the Internet. A name server's foremost task is to translate computer names to IP addresses and vice versa.

DHCP is software for automatic configuration of nodes on an IP-based network

FreeS/WAN is a popular open source application for building VPNs (Virtual Private Networks). It uses IPSec and common encryption algorithm such as 3DES and AES along with the option to compress data in traffic.

GIMP is used for picture editing. It can be used for editing and touching up of pictures from digital cameras, but also as a general purpose drawing program and for converting graphics files between different formats.

Mozilla is a web browser which has some program code identical to that within Netscape Communicator 6. Mozilla is a complete suite of programs with support for both Linux and Windows.

MySQL is an SQL based database server with accompanying applications for administration, control and clients. MySQL is the most frequently used open source database on the Internet. Support for almost all platforms including Linux and Microsoft Windows is available.

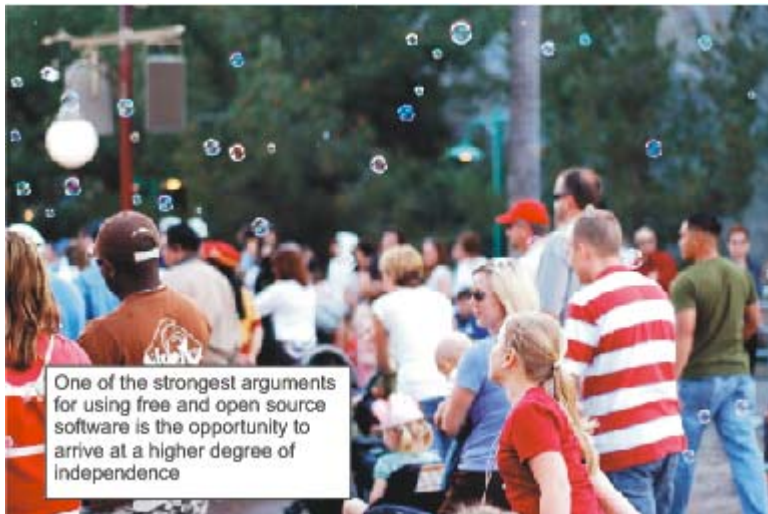
OpenOffice.org is a complete office suite for word processing, presentations and spreadsheet. Representatives from Sun administer the project and Sun contributes to a large part of the development. The project has developed their own XML-based file format and also support for MS Office file formats, although not 100%. Works on both Linux and Windows platforms.

Samba is a product which emulates an NT 4-server on a Unix-based platform. Samba works as both a file and a print server for both Windows and Linux clients.

Sendmail is the most common of all mail servers on the Internet. Sendmail includes functions such as mail routing, mail relay, anti-spam and also supports protocols such as SMTP, POP and IMAP. Sendmail can also function as a client e-mail server.

Why FOSS ?

One of the strongest arguments for using free and open source software is the opportunity to arrive at a higher degree of independence regarding price and



One of the strongest arguments for using free and open source software is the opportunity to arrive at a higher degree of independence

licensing conditions. In a situation with economic restraint, new and more rigorous licensing conditions, and with software that becomes replaced by newer versions more often than before, the software environment becomes increasingly more expensive.

Free and open source software enjoys a significant market share in many areas. Notably, Apache is used for more than 65% of all Internet web servers,

often with Linux as operating system. Free and open source software often has higher dependability, and in many cases better performance when directly compared to its proprietary counterpart.

Scalability and flexibility within the model for the development of free and open source software enables it to be developed for a large number of platforms and environments.

An area difficult to measure and compare is security, but it has been found that it is just as good, if not better, than proprietary alternatives. Free and open source software is less prone to attacks and virus over the Internet. As far as costs are concerned, it is to the advantage of free and open source software, especially if one looks exclusively at direct costs.

Pros and cons

Advantages of free and open source software are as follows:

- higher stability
- high level of security
- none or low licensing fees
- possibility to modify source code
- ample access to IT specialists
- independence from major software vendors

One considerable advantage of free and open source software is of course that one can customize and modify a product for a certain target group of users, i.e. make the software simple and functional. For example, a specific development environment can be created, an application for electronic services aimed at the general public can be developed or a desktop computer with adapted functionality can be set up.

The products are extremely dependable in production and easy to administrate and maintain. There is an initial cost of development, but running costs are very low and the total cost is considerably lower than comparable proprietary alternatives.

Positive effects:

- simpler license management
- reduced dependence on a product, less risk for "locking in" effects
- lower costs overall
- increased competition
- increased quality and stability

- increased activity on part of local/domestic businesses
- increased security
- open formats simplify communication with general public

Negative effects:

- possible need for extensive migration
- could be difficult finding the right product
- psychological resistance among decision makers

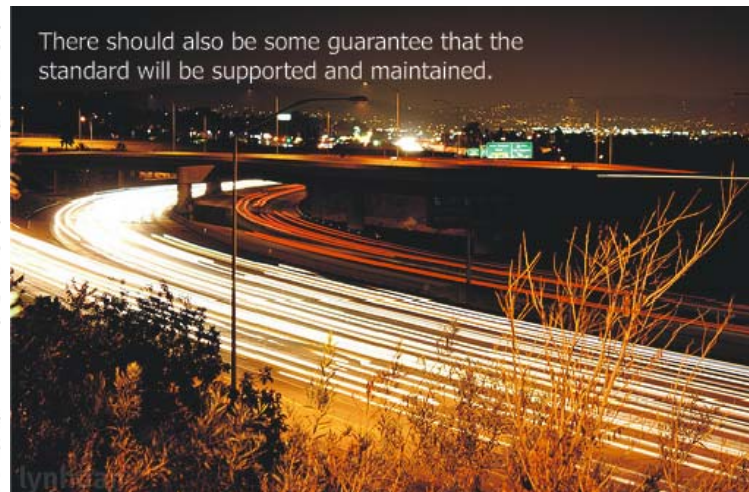
Several of the negative effects could take place at an initial stage of a project and are usually eliminated by vendors whose core expertise is FOSS and it's implementation for various operating environments.

Open standards

Free and open source software preferably makes use of open standards, but there is nothing explicitly said about this in the various types of licenses for the software. The standards that are of interest in this context are definitions of file formats and standards for the exchange of information, i.e. protocols and formats for communication between different applications and systems.

The term standard means an accepted norm for a certain activity, event or occurrence, or a variation thereof, which is commonly used or accepted. There is, on the other hand, no clear definition of the term open standard, but generally a standard is considered open if:

- Anyone can use it or take part in work on creating the standard.
- It is arrived at in generally accepted and open environment.
- Development takes place according to a model of consensus, and not led by one single player.



- The specification is published freely and is available without cost or only at prime production cost.
- Documentation is published freely and is available without cost or only at prime production cost.

There should also be some guarantee that the standard will be supported and maintained.

Examples of open standards are HTML and TCP/IP. HTML is published by World Wide Web Consortium (W3C), an independent body of companies and organizations and TCP/IP is published by Internet Engineering Task Force (IETF). These organizations also accept the responsibility for the further development of their standards. Both W3C and IETF fulfill the requirements listed above.

It is important to check to what degree a product is really based on open standards. If the product does not follow the open standards one is at a risk of having in time to change over to another solution, with increased costs as a result.

Interoperability

Proprietary office applications create a type of document file, which can only be completely recreated by an office application that can recognize the original file format.

It is these proprietary document formats, which tend to be an obstacle for companies and government agencies. Many different applications need to be able to use what is contained in the document files, and it results in serious problems if these are not compatible with the document formats of other applications. Another aspect concerns the legal requirements for archiving, where information should be kept for as long as possible, preferably for "a limitless time". This, in principle, demanded that an open, standardized document format be used in order to avoid readability problems in the future.

There is, therefore, a great need of a standardized and open document format for storage, document exchange, archiving, etc., rendering possible an exchange of information between applications and systems.

Inadequate interoperability between applications and systems are one of the main reasons for why businesses and organizations hesitate to change over to free and open source applications in the computers in their working environment.

Today, Microsoft, with a market share within certain product areas of around 85-90%, enjoys a distinctly dominating position and the document formats in MS Office have become something of a de facto standard.

Vendors lock-in

Becoming dependent on a leading vendor can result in adverse effects of upgrading, where new versions of the product are not always initiated by the actual needs of the user, but rather by the product cycle of the vendor or supplier. A software upgrade also often leads to a need to buy new hardware, resulting in even higher costs.

Any attempt at breaking out of this vicious circle entails considerable endeavour, resulting in a state of affairs where many users just accept the situation as it is. The more dependent a user becomes, the more difficult it is for him to break away and a situation arises where a vendor can dictate his own conditions.

The IT sector is no different than most other business sectors: Suppliers and vendors attempt to arrive at a deliberate degree of locking in with the purpose of keeping a customer and securing his continued business. As a customer, it is impossible not to be affected in some way or the other, but increased awareness also means increased possibilities of containing and limiting possible damage.

Locking in is a serious problem in situations concerning the procurement, operations, and maintenance of complex IT based systems. The problems are caused by costs for upgrading and system changes, the choices made in one's surrounding environment along with how one deals with previously made decisions.

A careful review of the costs for locking in together with relevant requirement specifications when procuring new systems – before the effects of locking in can take place – constitutes an efficient way of lessening the negative effects of locking in.

Furthermore, open standards offer a possibility of minimizing vendor or product dependence, given that the implementation of the standard does not include one's own additions. Free and open source software can decrease the risks of locking in by using open standards and interfaces.

More and more users seem to have taken notice of the situation, and a sharp and distinct protest, focused on increased costs, worsened terms of contract and increased locking in, have resulted in many users during the last six months starting to look for an open file format for office applications and also becoming interested in alternative products based on free and open source software.

Using FOSS

Several different studies show that Linux is becoming more and more common as an operating system within companies and government agencies all over the world. In the first place, Linux replaces proprietary Unix and Windows platforms on servers. Studies show that Linux is the fastest growing operating system on the market today.

Major systems suppliers and integrators, such as Hewlett-Packard and IBM, put Linux on an even par with earlier traditional operating systems and offers support and consultancy services at the same level. IBM goes as far as choosing Linux as a Unix based operating system for certain system solutions instead of its own AIX. Even Apple abandoned it' own direction developing it's current and future Operating Systems on a Linux kernel.

Most often asort of partnership is established with one of the leading Linux distributors such as RedHat or MandrakeSoft and tests are carried out to ensure that applications and hardware are compatible and work together without any glitches. Both IBM and HP dedicate considerable resources in the development of Linux.

Even other application providers develop Linux based versions of their own software. Oracle is one such example. At Oracle it is considered just as important to make available software versions for Linux as well as any other operating system. In some areas, Linux dominates the market for Internet servers: Web servers and large server clusters with stringent requirements for calculation capabilities.

Recent developments in better user interfaces and office programs such as OpenOffice have made Linux a viable alternative even for desktop computers in the workplace.

The following areas of use are of interest, partly because they represent a considerable portion of software already used within government agencies and partly because there already are competitive free and open source software alternatives or they could easily be set up and effectuated within these areas.

Infrastructural applications – name servers, catalogue servers, network servers, mail servers, etc.

- Server operating systems
- Workplace computers (PC)
- Workflow systems
- Business Intelligence tools
- Data migration and transformation
- Reporting



- Collaboration
- Analytic and Modeling Tool
- Office applications
- Web servers and browsers
- Development environments
- Databases
- Business-specific systems
- Security software

The current FOSS market

One common motive for not using free and open source software on a broad front is the fact that there is not the same range of vendors supplying services such as software and operational support.

Nowaday all major software vendors including Oracle, Apple, IBM, HP, SAP etc sell the majority of their products based on various FOSS tools. Also those vendors often sponsor FOSS products development providing a firm support for FOSS marketplace.

FOSS in government and military projects



In the United States, the Department of Defense (DoD) is a long-standing proponent of free and open source software. In 2009 DoD [launched](#) it's own open-source portal [Forge.mil](#), stimulating usage of FOSS in military projects. According to DoD, [«The open-source development model works for everybody».](#)

The United States Department of Defense (DoD) has launched a new site - Forge.mil - as its own open source repository. Built on the same technology used by SourceForge.net, the Forge.mil project does have an added layer of security to meet DoD standards, including the use of smart cards for system authorization. The early access version of the Forge.mil and software.forge.mil collaboration capability is for internal DoD testers only, reports Guy Martin, Open Source Community Manager at CollabNet, but is a first step in the government's shift toward open source technology.

Defence Information Systems Agency (DoD) [will open source](#) administrative software by allowing 3rd party developers to view, modify and reuse the source code.

According to [Open Technology Development roadmap](#) , DoD urges open source adoption and «combines salient advances in the following areas:

- Open Standards and Interfaces
- Open Source Software and Designs
- Collaborative/Distributive culture and the and online support tools
- Technological Agility

In 2008, [the National Defense Authorization Act for fiscal year 2009](#) included language that, for the first time, in section «SEC. 143. REQUIREMENT FOR COMMON GROUND STATIONS AND PAYLOADS FOR MANNED AND UN MANNED AERIAL VEHICLES.» instructed the DoD to favor open source software.

According to [DoD Open Source Software FAQ](#), the most frequently used FOSS tools are as follows below.

Quick List of FOSS Software Used in the U.S. DoD

Quick List of FOSS Software Used in the U.S. DoD

As of publication 2/2010,

A

ACE, ACE ORB (TAO), ACID, AMANDA, Apache, Autoconf, Automake



B

bash, Bastille, BIND

C

CVW, Cygwin, C++, Boost, CIS, Benchmarks, Colt, Condor, COPS, Crack, CVS

D

DDD, DjVuLibre

E

EADSIM, Emacs, eTrust, Expect

F

FreeBSD

G

GateD, gawk (awk), GCC, GDB, Ghostscript, GNAT, GnuPG, gnuplot, grep

H

h2n, HOSTS

I

ImageMagick

J

JADE, Jakarta, Jaxen, JBoss, JDOM, Jikes, jSIP

K

Kaffe

L

LaTeX, Linux, Linux (Red Hat), Linux firewalls, Lsof



M

M4, Majordomo, make, Maxima, MIMESweeper, MRTG MTR

MySQL

N

Nessus, NetBSD, NetSaint, nload, Nmap, ntop, NTP

O

Octave, OpenBSD, OpenMap, OpenOffice, OpenSSH, OpenSSL

P

Perl, Perl CGI scripts, PerLDAP, PHP, PingScan, Procmail

Q

Qmail

R

RealSecure, RRDtool, RTLinux, RWhois, RXVT

S

Samba, SARA, SATAN, Saxon, SCA, sed, SELinux, Sendmail, SNARE, Snort, Squid

T

Tcl/Tk, TCP Wrappers, Tomcat, Top, Tripwire

U

V

VisAD, VOCAL, VTK

W

Webmin, WebTAS, Weka, WU-FTPD

X

Xalan, Xerces, XFree86, XGobi, Xpatch

Y

Z



zlib, Zope

A firm adoption of FOSS by U.S. Government was clearly shown at [GOSCON](#) conference took place in November 2009 in Washington, DC. During the conference a new document «[FAQ about Copyright and Computer Software](#)» was issued serving «as a useful resource as federal agencies increasingly adopt the use of open source software in their IT architectures.»



Conclusion

Using FOSS in client projects, both commercial and government projects and own products allows:

- adhere to open standards
- inter-operate in various system environments
- avoid vendors lock-in
- apply the latest tendencies in software development for U.S. Government and DoD.

About the authors



Paul Swengler

His experience in computers dates back to the early 70's and his international business is highlighted with a successful track record with Addressograph's Documentor Division - Pacific Area, RCA in Asia, and Wang Asia Marketing Support Manager.

He led a number of highly successful operations in computer development and in end user applications.

His core belief has always been that computers are to serve people not the other way around. He is an experienced high ticket - high tech professional with a domestic and international track record including VAR development.

He can transcend technology into human terms.



Michael Lyubchenko

Michael was in the core team launched [WebGalileo Faces™](#) , an open source tool for RIA and Java Faces. When he was at architecture and management positions in the company, Michael participated in design, architecture and delivery phases of projects for several Fortune 500 clients, including British Telecom, Siemens, Hyundai, Hitachi, and Thomson.

He holds MD in Computer Systems and Networking from Kharkov State University and Sun Java Certified Programmer.



SoftAspects LLC is U.S. based privately owned company with headquarters in Tampa/St.Peterburg area and development offices located in Ukraine.

It's core competency is open source software, Rich Internet Applications and Web 2.0 technologies. One of the company's product, WebGalileo Faces™ , is used around the globe by different clients ranging from SMBs to international corporations.

SoftAspects can be found online at www.softaspects.com

