WYŻSZA SZKOŁA BIZNESU
NATIONAL-LOUIS UNIVERSITY

WYDZIAŁ PRZEDSIĘBIORCZOŚCI I ZARZĄDZANIA
KIERUNEK: Zarządzanie i marketing z wykładowym j. angielskim

Lena Wojewódzka
Nr albumu: 7427

OPEN SOFTWARE SOLUTIONS: PHILOSOPHICAL FOUNDATIONS, DEVELOPMENT AND APPLICATION IN BUSINESS

ROZWIĄZANIA OPEN-SOURCE: PODWALINY FILOZOFICZNE, ROZWÓJ ORAZ ZASTOSOWANIE W BIZNESIE

Praca licencjacka
Promotor: dr Włodzimierz Moczurad

Nowy Sącz, 2006

One line of code at a time,
application by application, Web server by Web server,
the data centers of a growing number of
major companies are taking on a new personality,
one that smells of the ocean and waddles when it walks.
The trend is open-source software (…)

Larry Greenemeier
Open Source Goes Corporate

Table of contents

Introduction

The last century is definitely the era of rampant capitalism. The capitalistic concept took the lead in the battle of economic systems and influences nowadays – directly or indirectly – the whole globe. In the world of aggressive Western values and increasing profit as the ultimate basis for existence, a fully unexpected phenomenon emerged. The trend called open-source software is one of the most important movements in today's global technology. It has changed the face of software, creating an alternative powered by community not motivated by money, but the need of contribution, innovation spreading and, what may seem unusual, fun.

At the same time, the business environment always seeking decrease of costs and increase of efficiency became interested in the open software. The solution apparently created for the ones interested in computer technology, developed into an everyday tool used and praised by many. What once used to be a vision of a narrow group of computer geeks, took on a new personality and turned into a methodology that goes far beyond IT.

This bachelor thesis is to explain the philosophy behind the open-source software, its development and application in business. The main question to be answered is whether open-source software is mature enough to be implemented in enterprises.

The work is divided into two parts – theoretical and empirical. The theoretical part provides overall understanding of the open-source software concept and its aspects. The empirical part refers to the observation of practical applications of open-source software in world-wide known companies.

The first chapter aims at introducing the overall concept of open-source initiatives and the definition of open-source software. The history of the open-source software movement and the importance of the contributing community follow. The chapter finishes with making the reader familiar with the ways of making profit from open-source products and services.

The second chapter concentrates on the economical, technical and legal aspects of open-source software in the business context, as well as describes the main drawbacks of open solutions.

The third chapter is dedicated to the research into 10 companies: Lapeyre, LVM, Renault, Skanska, Statoil, Lycos Europe, neckermann.de, Sony, Suzuki and Yahoo!

Finance, that have chosen open-source software as the answer to their IT needs. Two open solutions will be analyzed on the example of the before mentioned companies – Red Hat Enterprise Linux and MySQL database. The study will aim at presenting the reasons for implementing open-source software in companies, in addition to proving that open-source software is an alternative to proprietary solutions.

The resources used to complete the work include academic books, magazine and newspaper articles, online articles, corporate websites, materials published by the analyzed companies and software vendors. The thesis is licensed under the Creative Commons Attribution-Non-Commercial-Share Alike 2.5 Poland License.[1]

[1] To view a copy of the Creative Commons Attribution-Non-Commercial-Share Alike 2.5 Poland License, see *Attachment 1*, visit http://creativecommons.org/licenses/by-nc-sa/2.5/pl/ or send a letter to Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA

Chapter 1:
The concept of open-source software

1.1 The open attitude

Nowadays the term open-source refers to a philosophic view as well as a methodological approach. The advocates of the philosophic view claim that open-source products contribute to the evolution of the whole humanity as well as increase its well-being. The methodological approach concentrates on the way a product ought to be created in order to delight with its quality and reliability. The expression *open-source* is generally understood as practices in production and development that encourage access to the end product's sources.[2] Moreover, the model permits the usage of various schemas and approaches in production. The expression means also that everyone who wishes to modify a given system is free to do it. Open-source products became the fascination of economists, sociologists, political scientists and many more who are interested in the methodology and ethos surrounding the phenomenon.

Open-source software became the most known example of open-source. However, the open-source concept influences many other fields of study, such as nourishment, health, politics, management and technology. Let us take medicine as an example. An open-source pharmaceutical development - the Tropical Disease Initiative, is a web-based effort where scientists from laboratories, universities, institutes and corporations can work together in order to develop drugs for tropical diseases.[3] Synaptic Leap that is also the advocate of collaborative on-line research concerning tropical diseases claims that it sees no reason why to keep research results secret, as the profit driven pharmaceutical world is not interested in its projects aiming at tropical regions.[4] CAMBIA – an Australian non-profit research organization, develops mechanisms for plant improvement using technology based on the open-source model. In order to develop and market new biotechnologies, CAMBIA has created technologies, patents and licenses that are to give innovators greater freedom.[5] Moreover, CAMBIA works on a system that would enable

[2] *Open-source* [Online] Available at: http://en.wikipedia.org/wiki/Open_source [Accessed on 21.08.2006]
[3] *Tropical Disease Initiative* [Online] Available at: http://www.tropicaldisease.org/ [Accessed on 21.08.2006]
[4] *The Synaptic Leap* [Online] Available at: http://www.thesynapticleap.org/ [Accessed on 21.08.2006]
[5] *CAMBIA* [Online] Available at: http://en.wikipedia.org/wiki/CAMBIA [Accessed on 21.08.2006]

contributions assessment in order to help researchers to identify promising techniques.[6] The before mentioned open-source health initiatives aim at using open-source licenses to keep the discoveries available not only to researchers, but also manufacturers if they become interested in the products.

Open-source politics uses Internet technology such as e-mails and blogs to provide rapid feedback between political organizations and their supporters, at the same time, definitely increasing political transparency.[7]

When it comes to management techniques, Toyota creates teams that stress decentralization, flexibility and autonomy that are also the characteristics of the Linux community.[8] Sharing knowledge widely, establishing reputation systems and working for peer recognition became the characteristics discovered lately by companies that gain by giving up some of their proprietary knowledge.

As far as computer technology is concerned, not only software may be open-source. Hardware specifications of Sun Microsystems' OpenSPARC T1 Multicore processor are published in order to enable its modification and encourage innovation. Moreover, the OpenSPARC initiative focuses on making the newest intellectual property available for free and promote evolvement through cooperation.[9]

1.2 Defining open-source software

Open-source software has its source code available under an open-source license in order to study, change and improve its design by practically anyone. Such a software allows anybody to make a new version, move it to other operating systems and processor architectures, share it with others or market it.[10] Open-source software aims at the product to be more understandable, modifiable, duplicable and accessible. What is more, open software evolves at a speed that seems astonishing when compared to the pace of

[6] Special Report: Open, but not as usual - Open-source business *The Economist*. London. Mar 18, 2006
[7] *Open-source politics* [Online] Available at: http://en.wikipedia.org/wiki/Open_source_politics [Accessed on 21.08.2006]
[8] Evans P., Wolf B. Collaboration Rules. *Harvard Business Review.* July 2005
[9] *OpenSPARC* [Online] Available at: http://opensparc.sunsource.net/nonav/index.html [Accessed on 21.08.2006]
[10] *Open-source software* [Online] Available at: http://en.wikipedia.org/wiki/Open_source_software [Accessed on 21.08.2006]

conventional development.[11] It is said that the open-source model creates software that is more secure, more easily integrated and, at the same time, of higher quality. It became a *live example of user innovation moving across the entire spectrum from design and development to execution to support and upgrades [Karim Lakhani]*.[12] The process of open-source software development is very often compared to the scientific progress which is largely based on free exchange of knowledge and theories among participants who may easily use and modify them.[13] It is said that open-source software democratizes technology and is itself a social change concept. Open-source software is claimed to bring actual benefit to the society, as it enables the creation of participation based economy around technology.[14] The key components and characteristics of open-source software are:

1. decentralization,
2. cooperativeness,
3. participation cost close to 0,
4. international involvement,
5. resource friendliness.

Moreover, open-source software has a great impact - political, economic and cultural - on societies. It assures informational freedom, democracy and cultural equity. At the same time it supports ethical business practice, innovation and the right of people to choose. Since communication is the basis for societies, its mechanisms should remain open and free. People should not be made to pay for speaking the language of technology so open-source software and open standards create a forum for everyone who wants to participate. Open communication drives democracy and cultural equity is ensured by giving everyone the possibility to participate on their terms. Thanks to open-source projects competition is open and vendors are more accountable towards users. Furthermore, the ones not being customers of a given company are not penalized. Innovational environment is supported by open-source projects as they do not require short-term gains and with open innovation more people innovate. Additionally, open-source provides the community with natural selection in software.[15]

---

[11] *Basic idea behind open-source* [Online] Available at: http://opensource.org/ [Accessed on 21.08.2006]
[12] *Executive guides: open-source software* [Online] Available at: http://guide.darwinmag.com/technology/program/open/index.html [Accessed on 21.08.2006]
[13] Vermeir D. *Open-source: an overview* [Online] Available at: http://tinf2.vub.ac.be/~dvermeir/software/oss-overview-slides.pdf [Accessed on 21.08.2006]
[14] Seigo A. *How open-source software improves society* TPOSSCON 2006
[15] Seigo A. *How open-source software improves society* op. cit.

However, one cannot be blindly idealistic when it comes to open-source software – its greatest benefit may be its greatest undoing at the same time. When everyone can contribute it leaves the project open to abuse, intentional or not, and constant self-policing needs to be implemented.[16]

Open-source software is often referred to as the opposition of proprietary software. Proprietary software has restrictions on using and copying that may be achieved by legal or technical means. Technical means consist of including machine-readable binaries only and legal means include licensing, copyright as well as patent law. Such software has an owner who fully controls it. Though, there exists proprietary software which source code is available in order to study and modify it with redistribution of modifications or sharing the software restricted by a license.[17]

1.3 The history of open-source software: developers and evangelists

The term *open-source software* as such was officially used for the first time in 1998 when Netscape released most of the code base of Mozilla Suite under an open-source license.[18] The suite was to become the base of many applications such as Firefox web browser and Thunderbird e-mail application. Some say that the open-source movement started with the before mentioned event, however, it is also claimed that the birth of Internet or early 1960s when software was developed and passed between academics freely were the real beginning. The truth is that till the middle of 1970s software was intensively shared and programmers collaborated widely regardless of the employer. This model was similar to open-source, still, nobody used the exact name. With time computers were becoming more and more accessible. Software companies wanted to control their key assets such as intellectual property and kept the source code of their products secret. The customers were buying the product but the whole process stayed in the company. In the 1970s UNIX was getting more and more popular in the academic and research centers, as well as in the business world. It was originally open-source but some companies started selling its proprietary versions (of course incompatible). Collaboration was no longer that

[16] Special Report: Open, but not as usual - Open-source business *The Economist*. London. Mar 18, 2006
[17] Above paragraph according to *Proprietary software* [Online] Available at: http://en.wikipedia.org/wiki/Proprietary_software [Accessed on 21.08.2006]
[18] *Open-source software* [Online] Available at: http://en.wikipedia.org/wiki/Open-source_software [Accessed on 21.08.2006]

easy. Richard Stallman decided to create a free, open-source operating system - GNU. In the 1980s he founded the Free Software Foundation that was aiming at supporting GNU and other similar projects.[19]

The GNU Public License was later on created in order to protect the idea and practice of free software. It has become the most popular free software license that uses copyleft mechanism requiring derivative works of programs licensed under GPL also to be licensed under the GPL.[20] The copyleft as an idea preserves freedoms making them inseparable with the concerned piece of software. Many open-source programs are licensed under the GPL including Linux OS, MySQL database engine and JBOSS application server. The Lesser GNU Public License was developed specially for software libraries. The LGPL does not require the entire application – libraries and the program using the libraries - to be licensed under the GPL. Still, changes to the library itself are said to be published under the LGPL if distributed.[21]

As the GNU project was not able to produce a kernel, in the early 1990s a college student from Helsinki – Linus Torvalds, created the Linux kernel using development tools built by the Free Software Foundation.[22] He did not expect people to be interested in his idea. At the present time, his kernel is normally used in combination with the GNU system and forms a complete, functional, free software operating system referred to as *Linux* which became one of the most famous examples of open-source software.

Since the Internet grew in popularity in the 1990s, new open-source projects were developed. In 1997 one of the open-source authorities, Eric S. Raymond, in his essay "The Cathedral and the Bazaar"[23] based primarily on observations of the Linux kernel development, explained the popularity of community-created product when compared to proprietary solutions. The author claimed that traditionally software is developed by as few architects as possible in a centralized way with clearly defined roles. In such a model – the cathedral, the source code is available with each release, however, the development between releases is limited to a small group of architects. On the contrary, the bazaar model assumes following patterns:

---

[19] Above paragraph according to *Brief history of open-source* [Online] Available at: http://www.netc.org/openoptions/background/history.html [Accessed on 22.08.2006]
[20] *General Public License* [Online] Available at: http://en.wikipedia.org/wiki/GNU_General_Public_License [Accessed on 21.08.2006]
[21] *Understanding free and open-source licenses* [Online] Available at: http://stephesblog.blogs.com/papers/Optaros_Und_FOSS_Lic_SWalli_PartII_051906.pdf [Accessed on 21.08.2006]
[22] *Linux* [Online] Available at: http://en.wikipedia.org/wiki/Linux [Accessed on 21.08.2006]
[23] Raymond E.S. *The Cathedral & the Bazaar*. O'Reilly. October 1999

1. users should be treated as co-developers, since more developers equals higher rate of evolvement as well as additional testing environments,
2. early releases, since the chance of finding co-developers early increases,
3. frequent integration, since large number of bugs should not be fixed at the end of the project life cycle,
4. at least two versions – one with more features and one more stable,
5. high modularization – allowing parallel development,
6. dynamic decision making structure – depending on changing environment and requirements.[24]

Eric S. Raymond believes that only software built according to the bazaar model is able to fully evolve and because of public testing and experimentations, bugs may be discovered and fixed earlier than under the cathedral development (*given enough eyeballs, all bugs are shallow*[25]*)*. In the cathedral model a huge amount of time needs to be spent to find bugs since the working version of the code is restricted to a small group of people. The open-source community according to E. S. Raymond *resemble[s] a great babbling bazaar of differing agendas and approaches*[26] that forms a unique environment for creation with no direct limitations. The leader in the bazaar model rewards the contributors with fame and gratitude, which are emphasized in the essay in the context of the gift/reputation culture theory that explains the eagerness of programmers to work on software and afterwards give it away.[27] On contrary, in the cathedral model extrinsic reward such as money and promotions are used by the leader who is setting goals and controlling the product. Additionally, E. S. Raymond in his essay pays close attention to the act of placing the principles of community over selfishness as inspired by R. Stallman and his idea of copyleft.[28] Still, it is important to mention that the cathedral is a typical model for proprietary software development and the commercial community, however, the original essay concerned free software only.[29]

---

[24] *Open-source model* [Online] Available at: http://en.wikipedia.org/wiki/Open_source_software#Open_source_model [Accessed on 21.08.2006]
[25] Raymond E.S. *The Cathedral & the Bazaar*. O'Reilly. October 1999
called also the Linus' law; more formally: "Given a large enough beta-tester and co-developer base, almost every problem will be characterised quickly and the fix obvious to someone."
[26] Raymond E.S. *The Cathedral & the Bazaar*. op. cit.
[27] *Brief history of open-source* [Online] op. cit.
[28] *Brief history of open-source* [Online] op. cit.
[29] Cathedral and the Bazaar [Online] Available at: http://en.wikipedia.org/wiki/Cathedral_and_the_Bazaar [Accessed on 22.08.2006]

In 1998 to prevent Microsoft from controlling the browser market and inspired by E.S. Raymond, Netscape planned to create an open-source browser - mentioned earlier Mozilla Suite. At the same time E.S. Raymond, B. Perens and others launched the Open-source Initiative. The initiative manages and promotes the Open-source Definition (based on "The Debian Free Software Guidelines" by B. Perens[30]). According to the definition, open-source does not mean the access to the source code only but also complying with the following rules:

1. free redistribution,
2. source code included,
3. modifications and derived works allowed,
4. integrity of the author's source code,
5. no discrimination against persons or groups,
6. no discrimination against fields of endeavour,
7. no need of additional license in case of redistribution,
8. license must not be specific to a product,
9. license must not restrict other software,
10. license must be technology-neutral.[31]

The Open-source Initiative gives out an open-source certification mark (i.e. it approves licenses).

The difference between the Open-source Initiative and the Free Software Foundation is rather vague. In general, the Free Software Foundation is considering freedom, political, social and philosophical ideals, whereas the Open-source Initiative concentrates on practical issues. The biggest difference between the approaches lays in motivation for developing and using such software. The free software movement sees the technical excellence as a by-product of the ethical standard while the open-source movement sees the technical excellence as the goal.[32] The Free Software Foundation has also distinct criteria of evaluating whether software can be described as free. According to the Free Software Foundation all free software is open-source but not all open-source

[30] More on Debian Free Software Guidelines may be found at http://en.wikipedia.org/wiki/Debian_Free_Software_Guidelines

[31] *Open-source definition* [Online] Available at: http://en.wikipedia.org/wiki/Open_Source_Definition [Accessed on 21.08.2006]
The complete definition is available at: http://www.opensource.org/docs/definition.php

[32] *Open-source movement* [Online] Available at: http://en.wikipedia.org/wiki/Open_source_movement [Accessed on 21.08.2006]

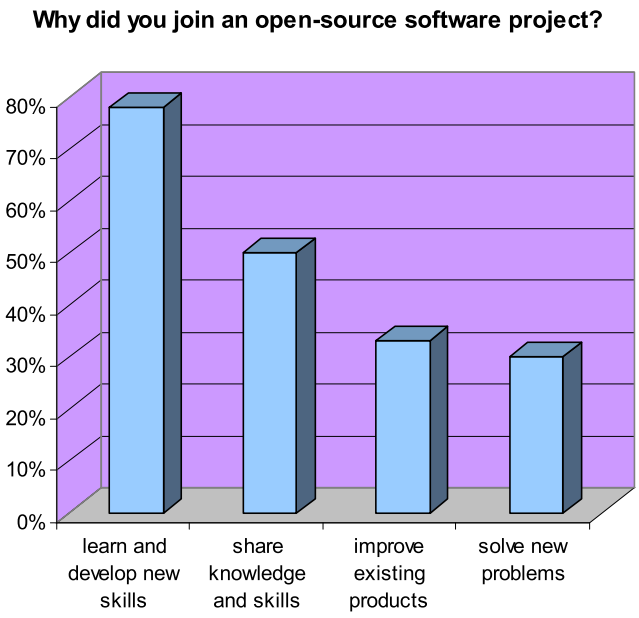
software is free.[33] Though, in practice most open-source licenses satisfy the Free Software Foundation rules.

1.4 The open community and its motivation

Open-source software tends to be developed by loosely-organized ad-hoc communities with contributors from all over the world who have never met face-to-face (...) this mish-mash of people coheres and effectively accomplishes an extremely complex task: building high-quality software.[34] A typical open-source software developer is: male (98%), young (70% between 22-37 years old), living in Europe or the USA (80%), an IT professional (more than 50%) or a student (20-30%), spending less than 5 hours a week on the project (34-48%).[35] They contribute because of need for learning and developing new skills (78%), sharing knowledge (50%), improving existing products and solving new problems (each around 30%).[36] Likewise, it is often said that getting involved in an open-source project may be a way to meet likeminded people. It is also described as a way to contribute to a larger body and by giving away knowledge and skills - getting enjoyment, fun and a feeling of belonging to a community. Moreover, the contributors stay to work in the project because of the need to learn (48%) and since they have a personal sense of accomplishment and contribution (25%).[37]

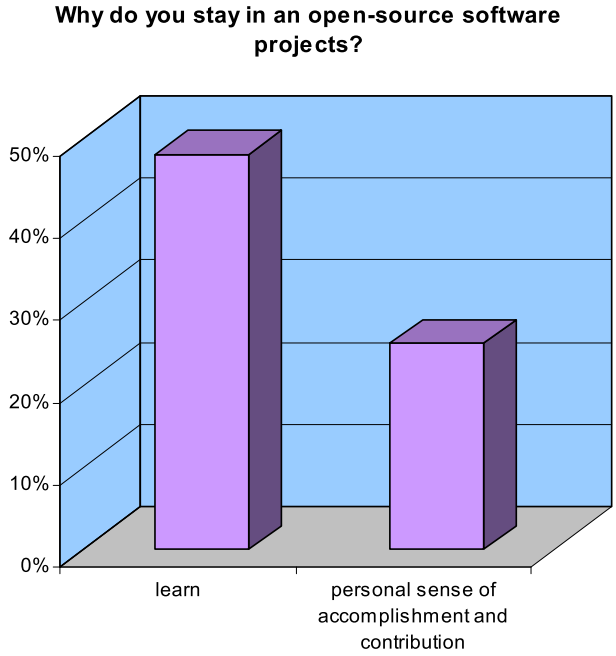
---

[33] *Free software* [Online] Available at: http://en.wikipedia.org/wiki/Free_software [Accessed on 21.08.2006]
[34] Kim (2003) cited by Vermeir D. *Open-source: an overview* [Online] Available at: http://tinf2.vub.ac.be/~dvermeir/software/oss-overview-slides.pdf [Accessed on 21.08.2006]
[35] Free/Libre/Open-Source Software (2002), Boston Consulting Group (2002), WIDI (2001), Kim (2003) cited by Vermeir D. *Open-source: an overview* [Online] Available at: http://tinf2.vub.ac.be/~dvermeir/software/oss-overview-slides.pdf [Accessed on 21.08.2006]
[36] Free/Libre/Open-Source Software (2002), Boston Consulting Group (2002), WIDI (2001), Kim (2003) cited by Vermeir D. *Open-source: an overview* [Online] op. cit.
[37] Free/Libre/Open-Source Software 2002, Boston Consulting Group 2002 cited by Vermeir D. *Open-source: an overview* [Online] Available at: http://tinf2.vub.ac.be/~dvermeir/software/oss-overview-slides.pdf [Accessed on 21.08.2006]



Graph 1.4.1 Motivation for joining an open-source project

Based on Free/Libre/Open-Source Software 2002, Boston Consulting Group 2002 cited by Vermeir D. *Open-source: an overview* [Online] Available at: http://tinf2.vub.ac.be/~dvermeir/software/oss-overview-slides.pdf [Accessed on 21.08.2006]



Graph 1.4.2 Motivation for staying in an open-source project

Based on Free/Libre/Open-Source Software 2002, Boston Consulting Group 2002 cited by Vermeir D. *Open-source: an overview* [Online] Available at: http://tinf2.vub.ac.be/~dvermeir/software/oss-overview-slides.pdf [Accessed on 21.08.2006]

It is said that they are intrinsically motivated as *People will be most creative when they feel motivation primarily by the interest, enjoyment, satisfaction and challenge of the work itself (...) and not by external pressures or inducements.*[38]

Aaron Seigo in his lecture at Trans-Pacific Open-Source Software Conference in 2006 mentioned 10 ways of getting involved in an open-source software project.[39] The most obvious way is writing code. Moreover, one may donate money and make purchases of the software, as well as gadgets connected with a given project. What is important, when open-source software is bought from huge corporations – the powerhouses of capitalism such as IBM or Novell – some of the money is still feed back into the cycle and invested into the direct project development. The third way of getting involved according to Aaron Seigo is creating an enterprise selling open-source software and services connected with it, and getting this way funds that may be given out to the community of developers. The fourth way mentioned is artwork creation and contribution, as all the icons, desktops and templates are separate pieces of art that obviously need to be designed. Furthermore, documentation in various languages also needs to be donated. Supporting other users by joining online forums and mailing lists is mentioned by Aaron Seigo in his speech, as well as creating and maintaining project infrastructure such as servers, mail lists and web sites. Being the advocate of open-source software and letting other people know that you use it becomes one of the most important issues nowadays according to Aaron Seigo, as not enough information and fear of unknown are the factors that may discourage from using open software.

The community of open-source software users and direct or indirect developers may be broadly divided into two categories: the core or inner circle – people modifying code that constitutes the project, and the peripheral – software users who report bugs and provide new suggestions. Those people may be further divided into:

1. project leaders – people involved in coding the first release,
2. volunteer developers – doing actual coding for the project,
3. everyday users – performing testing, bugs identification etc.,
4. posters – participating in discussions but not involved in coding.[40]

---

[38] Amabile (2003) cited by Vermeir D. *Open-source: an overview* [Online] Available at: http://tinf2.vub.ac.be/~dvermeir/software/oss-overview-slides.pdf [Accessed on 21.08.2006]
[39] Seigo A. *How to get involved* TPOSSCON 2006
[40] *Participants in OSS development projects* [Online] Available at: http://en.wikipedia.org/wiki/Open-source_software [Accessed on 21.08.2006]

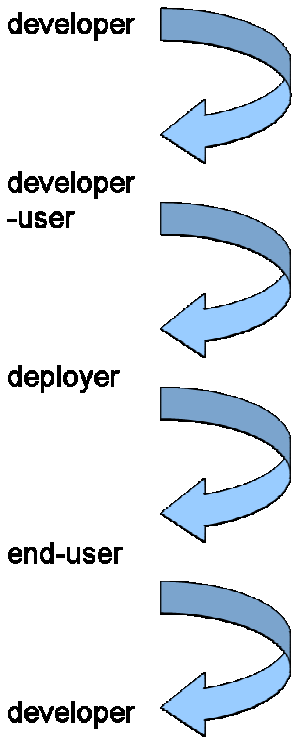


Illustration 1.4.1 The open-source community

Based on Metcalfe R. *Open-source: an Introduction* [Online] Available at: http://www.oss-watch.ac.uk/talks/2006-04-28-rsc-eastern/2006-04-28-rsc-eastern.pdf [Accessed on 21.08.2006]

Naturally, software companies contribute from time to time as well (just to mention Sun's Openoffice) in order to, for example, foster acceptance of a new technology, pool development resources, grow their consulting business (IBM and Novell), boost hardware sales (AMD and Intel) or support, grow and promote the market for the main business.[41]

Although most open-source software projects follow the cave model with very few people contributing and controlling the task along with requests and bug reports coming from the users (e.g. TouchGraph started by A. Shapiro to which direct contributions are difficult because of the system not being modular enough), the community model may be met in widely known projects such as SquirrelMail or the Linux kernel. SquirrelMail is divided into 7 projects (stable release, development release, internationalization, plug-ins, user support, documentation and system administration) with 10 project leaders, 5 active contributors and an overall project leader. Furthermore, each branch of the Linux kernel is

[41] *Who contributes to open source software? And why do they give it away?* [Online] Available at: http://www.oss-watch.ac.uk/resources/whowhy.xml [Accessed on 21.08.2006]

controlled by one person that decides which changes will be implemented but contribution may be done by anyone.[42]

Since the participants are situated in various geographic locations, a need for a tool enabling collaboration exists. The Concurrent Versions System released under the GNU General Public License helps to manage the project files when they are worked on at the same time by many people. Clients may connect to the server that stores the current version of the project and its history. They can check-out a copy of the project, work on it and than check-in introduced changes. When the work is done, the server attempts to merge the files. It may fail and reject the changes (e.g. two users aiming at changing the same line) - the second check-in operation is stopped and the client is informed about the problem which has to be resolved by hand. A possibility exists of updating user's local copies with the newest version on the server. The Concurrent Versions System is also able to sustain various branches of the project, e.g. a released one and one still being developed.[43]

In order to help while testing the software integration, a tool such as Tinderbox is used. Tinderbox enables error detection as it runs a continuous build process and informs about damaged parts on a given platform. It holds people accountable for their actions by allowing everyone to see what is happening in the source tree.[44]

With the aim of bug tracking, records of all reported bugs, versions in which they occurred, whether they were fixed or not must be kept. It is advised that in order to effectively track bugs a daily software build should be made. Such a build enables users to report bugs in the newest version and not concentrate on those that probably have already been fixed. Moreover, the person who reported the bug should verify that it has been fixed. A bug report ought to include a description on how to reproduce the bug so as to reduce the amount of time needed to do it by the developer. GNU GNATS is a tool that tracks bugs reported by users to a central site. It stores all information about problems and enables querying, editing and maintenance of the databases. GNATS is not limited to a single user

[42] Above paragraph according to Vermeir D. *Open-source: an overview* [Online] Available at: http://tinf2.vub.ac.be/~dvermeir/software/oss-overview-slides.pdf [Accessed on 21.08.2006]
[43] Above paragraph according to *Concurrent Versions System* [Online] Available at: http://en.wikipedia.org/wiki/Concurrent_Versions_System [Accessed on 21.08.2006]
[44] Above paragraph according to *Tinderbox* [Online] Available at: http://en.wikipedia.org/wiki/Tinderbox_%28software%29 [Accessed on 21.08.2006]

interface and it is easy to use and flexible thanks to possibility of storage in plain text files. What is more, it may be expanded by adding own utilities using standard GNU tools.[45]

1.5 The open-source business

Right now, one may find over 100 000 open-source products that easily substitute proprietary software as operating systems, internet infrastructure, database management systems, office applications or multimedia. As for most types of application alternative open-source solutions exist, it leads to competition on quality, creation of products that fit even niche user needs and take-up of functional novelties. The main advantage of proprietary software for an everyday user is compatibility, amount of extra features and user-friendliness. In contrast, users of open-source software will appreciate flexibility, lower costs, orientation towards innovation and avoidance of vendor lock-in. Another advantage of open-source software is quick support thanks to direct contact with the author(s) or various forums. When one compares an open-source forum to commercial support it turns out that forums are much faster, much more specific and cheaper. Though, if a user is the first one to experience the problem, they may be left with solving it themselves.[46]

Obviously, large and small companies discovered that it is possible to make a profit from open-source products and their theoretical disadvantages. The recipe for success in the open-source world seems to be rather simple: put the software on a CD-ROM in a ready-to-install format, add documentation and related applications, in the end – charge a service and support fee. However, one may be more successful extensively using one's imagination. Providing special extensions and customization, selling documentation and courses, consulting, as well as lowering prices of ready-to-use hardware are in nowadays.

Franck Hecker in his paper "Setting Up Shop"[47] described eight business models that may be used while aiming at making profit from open-source software. The models are described as follows:

---

[45] Above paragraph according to *GNATS* [Online] Available at: http://www.gnu.org/software/gnats/ [Accessed on 21.08.2006] and *Bug tracking guidelines* [Online] Available at: http://bug-tracking-guidelines.com/ [Accessed on 22.08.2006]

[46] Vermeir D. *Open-source: an overview* [Online] Available at: http://tinf2.vub.ac.be/~dvermeir/software/oss-overview-slides.pdf [Accessed on 21.08.2006]

[47] Hecker F. *Setting up shop: the business of open-source software* [Online] Available at: http://www.hecker.org/writings/setting-up-shop [Accessed on 21.08.2006]

1. support-sellers: revenues generated by selling physical goods and services related to the open-source software, competition based on offering more complete packages,
2. loss-leader: free of charge open-source product offered with hope that it will lead to profits from other products offered by the company, increase brand recognition and encourage to purchase proprietary products,
3. widget-frosting: intended mainly for hardware companies, open-source software distributed at no charge along with the hardware, most of the revenue generated by sales of the hardware,
4. accessorizing: distributing books and other physical items informing or teaching about specific open-source products, adding to the product free of charge open-source software,
5. service enabler: creation and distribution of open-source software to support access to revenue-generating on-line services,
6. sell it, free it: release of a proprietary product and changing it to open-source when that becomes more profitable,
7. brand licensing: software created as an open-source, however, the company retains the exclusive right to the trademark and brand name, charging for the right to use the name when making derivative products using the free product source code, the original company has the reputation of continuous testing and security,
8. software franchising: authorizing other developers to use the established brand names and trademarks, supplying the company with training in aspects of developing an open-source product and any other relevant information, revenue coming from sales of franchises and royalties.[48]

It is said that the most successful models are the support sellers and the loss-leader. The biggest part of revenue of the top Linux provider – Red Hat, that is utilizing the support-sellers model, comes from the enterprise business. The company is one of the first to make real money on open-source. Red Hat deploys, integrates, updates, manages and supports open-source solutions. It is selling Linux on a CD with access to services, support on different levels and a comprehensive users-manual. Red Hat is an expert from which one may buy Linux with a guarantee offered, get information and press releases.

[48] *Open-source* [Online] Available at: http://cci.mccombs.utexas.edu/research/white/open-source.htm [Accessed on 29.07.2006]

SuSE – the leading Linux distributor and support provider in Europe, aims at selling operating systems and application packages that would be needed while working in a Linux environment. The provider tries to enhance Linux with additional services and charge a fee for them. Turbolinux with Asia-Pacific as the number one market, goes beyond the open approach and sells closed software to run on open platforms.

Software giants such as Oracle begin to buy open-source start-ups which creates a huge controversy in the open-source community. The programmers who give up their free time to create software according to the open philosophy fear that their work will be now used to bring the rich companies profit and not benefit the community.[49] Also OEMs such as HP, Dell and Compaq do not want to stay behind the trend. Especially IBM sees revenues in open-source software with the use of widget-frosting model as it makes Linux servers, mainframes as well as offers applications and support.

For those companies, choosing between Linux and Windows is often very simple – *[OEMs] have got a lot to win. They never did make any money off the operating system because Microsoft was making the money there. So what difference does it make to them if they sell Linux or Windows?*[50]

When enterprises are concerned, there is a doubt whether open-source software will satisfy their needs. Support, training, documentation, integration and services play a very important role for them. Many organizations take a soft approach to open-source. They are still afraid to use open-source software to critical applications and choose proprietary software in their place. According to the InformationWeek magazine, 44% of all companies work in the mixed Windows/Linux environment.[51] Still, interest in open-source software increases every year. According to the Gartner analysis open-source software will generate during the next 5 years revenue 22% higher than the one today.[52] Moreover, the biggest demand is said to be created by small and medium size European enterprises.[53] Just

[49] Lacy S. Open season on open-source? *Business Week*. New York. Mar 13, 2006
[50] Gill L. *Who's making money from open-source?* [Online] Available at: http://www.newsfactor.com/perl/story/17137.html [Accessed on 21.08.2006]
[51] *Novell promuje open-source dla firm* [Online] Available at: http://www.egospodarka.pl/12361,Novell-promuje-open-source-dla-firm,1,14,1.html [Accessed on 22.08.2006]
[52] *Wzrosną przychody z open-source* [Online] Available at: http://wirtualnemedia.pl/document,,1275149,Wzrosna_przychody_z_Open_Source.html [Accessed on 21.08.2006]
[53] *Wzrosną przychody z open-source* [Online] op. cit.

to mention Poland where according to the Professional Linux Association and Pentor 94% of all Polish companies uses open-source software.[54]

The time has come to answer the most important question concerning implementation of open-source software in small and medium size companies – is open-source software really ready for the enterprise? And, what is even more important, is there a winner in the battle between open and close solutions for companies?

[54] *Oprogramowanie open-source stosuje 94 proc. polskich firm* [Online] Available at: http://wirtualnemedia.pl/document,,977690,Oprogramowanie_Open_Source_stosuje_94_proc._polskich_firm.html [Accessed on 21.08.2006]

Chapter 2:
Aspects of open-source software in the business context

2.1 Economical aspects

When it comes to economical aspects of open-source software, it is said to provide cost reduction and market health. The market share data concerning the period 1996-2006 clearly shows a transformation of software market structure with the growing popularity of open-source software that is to create an environment encouraging competition and, at the same time - innovation.

Cost reduction is an important motivator for those wanting to spend money wisely. The true cost of software ownership includes the sale price, hardware upgrades and maintenance, software upgrades and maintenance, downtime, administration of licenses, training and staffing cost.

Open-source software has a much lower initial price than proprietary software, in most cases it may be acquired freely by downloading. Those who wish to buy a CD-ROM version with paper documentation and support can pay a fee that costs still less that the one needed to be paid for proprietary systems. Just to compare the numbers, in October 2006 Microsoft Windows Small Business Server 2003 Premium Edition with Polish interface for 5 clients cost 8462 PLN and the unlimited version of SUSE Linux Enterprise Server 10 with Polish interface cost 1234 PLN.[55] At the same time, the cost of proprietary software increases with the number of computer clients, whereas open-source software may be used on as many computers as one wishes with no performance limits[56]. The upgrade costs of open-source software are usually lower. Upgrading of a proprietary system will cost around half of its initial price, at the same time the user being mostly dependant on the top pricing.[57]

Open-source software enables savings on license fees that swallow a significant chunk of every IT budget.[58] License management costs are in the case of open-source solutions close to zero. Proprietary licenses need to be managed, since customers must

[55] According to the price list of ABC DATA [as for 08.10.2006]
[56] Taking into account most distributions
[57] Wheeler D. A. *Why open-source software? Look at the numbers!* op. cit.
[58] Nairn G. Promise of cost-savings driver take-up of Linux. *Financial Times.* September 21, 2005

prove payment for every copy or become severely penalized. The organizations must track licenses and employ people who will perform audits to avoid a lawsuit.

Open-source software usually has lower hardware requirements, so it yields smaller hardware costs by eliminating the need for new components. Taking into account the previous example, the hardware requirements for Microsoft Windows Small Business Server 2003 Premium Edition and SUSE Linux Enterprise Server 10 are as follows:[59]

Server type	Hardware requirements
Microsoft Windows Small Business Server 2003 Premium Edition	▪ CPU speed – minimum: 750 MHz, recommended: 1 GHz ▪ 512 of RAM minimum required ▪ 16 GB of available hard disk space
SUSE Linux Enterprise Server 10	▪ 512 MB of RAM ▪ 1.5 GB hard disk space for software and user data

Table 2.1.1 Hardware requirements

According to *SUSE Linux Enterprise Server 10* [Online] Available at: http://www.novell.com/products/server/sysreqs.html and *System Requirements for Windows Small Business Server 2003 R2* [Online] Available at: http://www.microsoft.com/Windowsserver2003/sbs/evaluation/sysreqs/default.mspx [Both accessed on 08.10.2006]

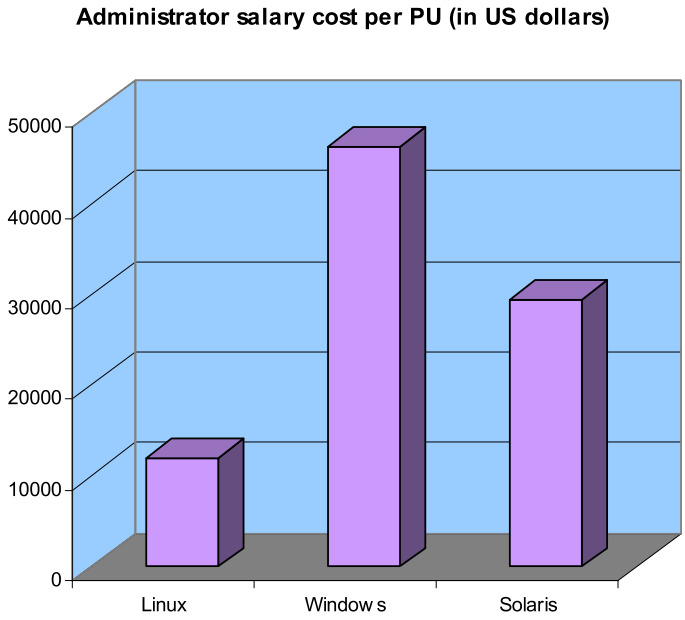
Open-source software maintenance is argued to be cheaper, as fewer – however, better paid - administrators are needed. According to the Robert Frances Group long-term study of 2002[60], despite the fact that Windows systems administrators are less expensive to hire, they were not able to manage more than 10 machines each compared with Linux and Solaris administrators. The administrator salary cost per processing unit for Linux was equal to $12 010, Solaris - $29 509 and Windows - $46 360. Moreover, also Netproject reported very significant savings in the number of support staff.[61] The before mentioned

[59] According to *SUSE Linux Enterprise Server 10* [Online] Available at: http://www.novell.com/products/server/sysreqs.html and *System Requirements for Windows Small Business Server 2003 R2* [Online] Available at: http://www.microsoft.com/Windowsserver2003/sbs/evaluation/sysreqs/default.mspx [Both accessed on 08.10.2006]

[60] Robert Frances Group *Total Cost of Ownership for Linux Web Servers in the Enterprise* [Online] Available at: http://www.rfgonline.com/subsforum/LinuxTCO.pdf [Accessed on 17.09.2006]

[61] Wheeler D. A. *Why open-source software? Look at the numbers!* op. cit.

cost reductions may be additionally based on the fact that it is easier to automate over time manual tasks in Unix-like systems.[62]



Graph 2.1.1 Administrator salary cost per processing unit

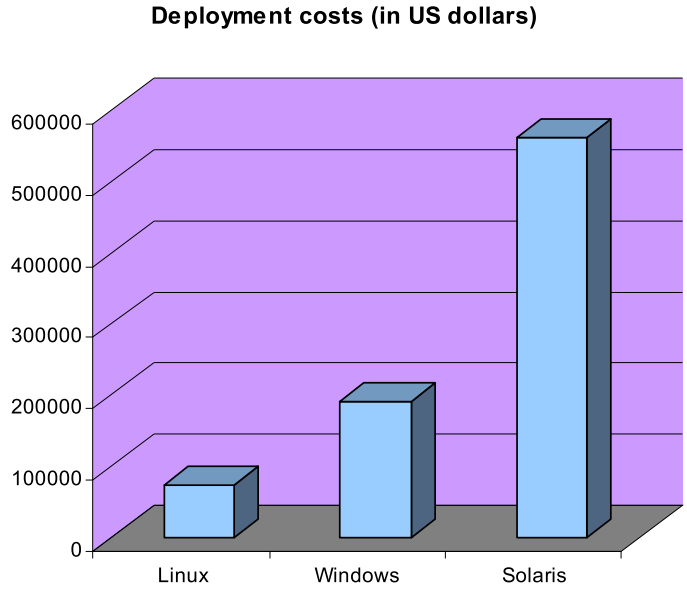
Based on Wheeler D. A. *Why open-source software? Look at the numbers!* [Online] Available at: http://www.dwheeler.com/oss_fs_why.html [Accessed on 17.09.2006]

Open-source software limits the possibility of vendor lock-in that may be described as the lack of compatibility between software versions and data formats.[63] With open-source software it is easier for users to switch from one project to the another, as many of those are developed. This way the users are not dependent on one vendor – a monopolist – who may raise prices or stop providing upgrades. In the case of open-source software always a possibility exists that if one group originating the code stops development, another group may continue without legal or practical limitations.

According to the findings of previously mentioned Robert Frances Group[64] in 2005, Linux has a lower total cost of ownership than Microsoft Windows and Sun Solaris. The Robert Frances Group states that using open-source operation systems enables to save up

[62] Wheeler D. A. *Why open-source software? Look at the numbers!* op. cit.
[63] *Vendor lock-in* [Online] Available at: http://en.wikipedia.org/wiki/Vendor_lock-in [Accessed on 17.09.2006]
[64] Orzech D. *Linux TCO: Less Than Half The Cost of Windows* [Online] Available at: http://www.cioupdate.com/article.php/10493_1477911 [Accessed on 17.09.2006]

to 40% of the amount that was to be spent on proprietary software and its maintenance. The study was conducted on production deployments of Web servers running on Linux, Microsoft Windows and Sun Solaris at 14 Global 2000 enterprises. It was calculated that Linux deployment cost $74 475, Windows $190 662 and Solaris $561 520. The Robert Frances Group found out that most Linux savings came from the software licensing fees.

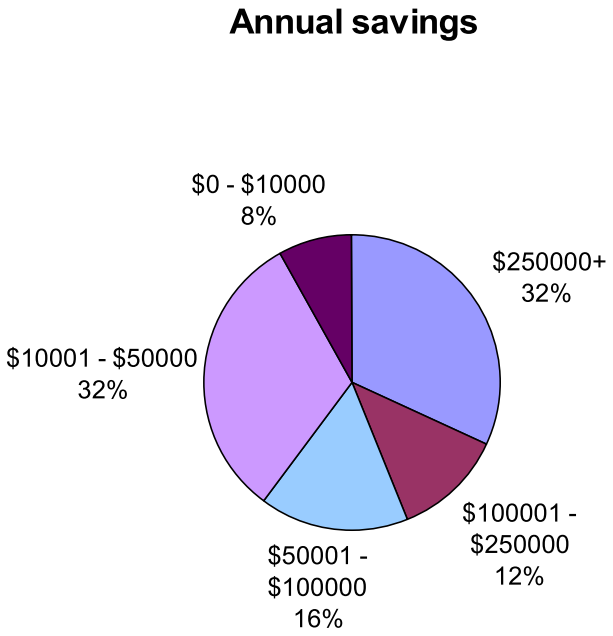


Graph 2.1.2 Deployment costs

Based on Orzech D. *Linux TCO: Less Than Half The Cost of Windows* [Online] Available at: http://www.cioupdate.com/article.php/10493_1477911 [Accessed on 17.09.2006]

Another survey conducted in 2001 by InfoWorld[65] on 40 Chief Technical Officers who were the members of InfoWorld CTO network reported that 32% using open-source solutions saved more than $250 000 annually. As a most important benefit of using open-source software, 93% of the CTOs were mentioning reduced cost of application development or acquisition.

[65] Wheeler D. A. *Why open-source software? Look at the numbers!* op. cit.



Graph 2.1.3 The annual savings using open-source solutions

Based on Wheeler D. A. *Why open-source software? Look at the numbers!* [Online] Available at: http://www.dwheeler.com/oss_fs_why.html [Accessed on 17.09.2006]

What may sound peculiar, Microsoft has fully admitted in 2002 that its products are in total more costly than open-source solutions. The Microsoft CEO Steve Ballmer stated that the company is trying to *figure out how to be lower-priced than Linux*.[66] However, one time the company claims that its products cost more, since they are of higher quality, than it sponsors a report claiming the opposite.[67]

2.2 Technical aspects

Open-source software ensures high quality of the products, transparence and, foremost, security. Open-source solutions are in a way the technical heritage of generations as the new and innovative software is built upon the previous knowledge. Moreover, it is claimed that to extend the lifetime of an application, it needs to be available in a source

[66] Greene T. C. *Ballmer 'fesses up to Linux/Windows cost FUD* [Online] Available at: http://www.theregister.co.uk/2002/07/16/ballmer_fesses_up_to_linux/ [Accessed on 17.09.2006]
[67] Bozman J. and others *Windows 2000 Versus Linux in Enterprise Computing* [Online] Available at: http://www.microsoft.com/windows2000/docs/TCO.pdf [Accessed on 17.09.2006]

form.[68] Most information needs to be accessible for many years and it is essential that data retrieval is permanently enabled. When given the source code, the way of storage is publicly known and traceable. Although, the development process of open-source projects looks rather chaotic, studies show that the applications are of higher quality than the commercial ones. The development process of open-source software outpaces that of closed source code. The software may be improved and repaired more quickly, as many programmers correct mistakes and add features. If an internal software malfunction takes place, the user may fix it himself or have it fixed by an expert, which is impossible in the case of proprietary software. What is more, the product is delivered when it is ready and when the development team decides that its quality is good enough. Marketing pressures in the case of open-source software are minimized because no commercial entity sees profit in speeding up the launch.

It is easier to find weaknesses and bugs given the transparent structure, which leads to increased security. All flaws identified in the open-source software by the Fuzz Random Testing Study measuring reliability of proprietary and open-source software by feeding them with random characters and seeing whether they crash or freeze up, were removed in the period between the tests. In contrast, the proprietary software vendors did not fix the problems during the 5 years time.[69] When a serious flaw was found in the Apache Web Server, a patch was available two days after the announcement. In the case of OpenSSL, a patch was available the same day. When the responses of Microsoft and the KDE Project to a serious SSL vulnerability are compared, it turns out that the proprietary vendor downplayed the problem, whereas the patch for Konqueror was accessible the same day.[70] Moreover, in June 2004 the U.S. Department of Homeland Security's Computer Emergency Readiness Team (CERT) recommended not using Microsoft Internet Explorer browser, since it cannot protect against vulnerabilities.[71] According to Scanit's Browser Security Test, Internet Explorer was safe from known and unstoppable remote attacks only 7 days in the year 2004.[72] In 2001 Internet Information Services was attacked 1400 times

[68] *Free Software/Open Source: Information Society Opportunities for Europe?* [Online] Available at: http://eu.conecta.it/paper/Advantages_open_source_soft.html [Accessed on 17.09.2006]
[69] Wheeler D. A. *Why open-source software? Look at the numbers!* op. cit.
[70] Wheeler D. A. *Why open-source software? Look at the numbers!* op. cit.
[71] Naraine R. *US-CERT: Beware of IE* [Online] Available at: http://www.internetnews.com/security/article.php/3374931 [Accessed on 17.09.2006]
[72] *Browser security test* [Online] Available at: http://bcheck.scanit.be/bcheck/page.php?name=STATS2004 [Accessed on 17.09.2006]

more frequently than Apache, taking into account that there are twice as many Apache systems on the internet.[73]

Proprietary software security concerns were one of the reasons of the open-source movement creation, however, safety of all software depends on regular monitoring, upgrades and modifications. Open-source software indeed gives the attackers more information, but at the same time, gives it also to the defenders. Linux operating system definitely leads the industry in defensive design.[74] After the Spring 2002 Linux Developer Survey including over 400 GNU/Linux developers, it turned out that Linux systems are relatively immune from outside attacks. 78% of the surveyed developers have never experienced an unwanted intrusion and 94% operated virus-free.[75] Even Bruce Schneier – the computer security and cryptography guru argues that one should *demand open source code for anything related to security*.[76]

According to various studies, the most popular open-source software packages are extremely reliable. IBM conducted 30 and 60-days stressful tests on Linux and found out that all core components of the operating system operated consistently. Moreover, the operating system had 0 critical system failures. It was pointed out that *the Linux kernel and other core OS components are reliable and stable (...) and can provide a robust, enterprise-level environment for customers over long periods of time.*[77] A 10-month ZDnet experiment was concluded with the findings that Microsoft Windows NT with Service Pack 3 crashed on average once in every six weeks with 30 minutes of fixing, while Caldera Systems OpenLinux and Red Hat Linux did not go down.[78] A study by Reasoning found out that MySQL database had 6 times fewer defects than 200 compared proprietary programs.[79] Coverity found out during a 4-year research that there are only 985 defects in 5,7 million lines of the Linux code, while a typical program of that size would have more than 5000 defects.[80] According to a 3-month Syscontrol AG analysis, sites using Microsoft Internet Information Services web serving software were on average twice more time

---

[73] Costello S. *RSA: Security in 2002 worse than 2001, exec says* [Online] Available at: http://archives.cnn.com/2002/TECH/internet/02/25/2002.security.idg/index.html [Accessed on 17.09.2006]
[74] *Features and quality* [Online] Available at: http://www.netc.org/openoptions/pros_cons/features.html [Accessed on 17.09.2006]
[75] Wheeler D. A. *Why open-source software? Look at the numbers!* op. cit.
[76] Wheeler D. A. *Is open-source good for security?* [Online] Available at: http://www.dwheeler.com/secure-programs/Secure-Programs-HOWTO/open-source-security.html [Accessed on 17.09.2006]
[77] Wheeler D. A. *Why open-source software? Look at the numbers!* op. cit.
[78] Wheeler D. A. *Why open-source software? Look at the numbers!* op. cit.
[79] Wheeler D. A. *Why open-source software? Look at the numbers!* op. cit.
[80] Lemos R. *Security research suggests Linux has fewer flaws* [Online] Available at: http://news.com.com/Security+research+suggests+Linux+has+fewer+flaws/2100-1002_3-5489804.html [Accessed on 17.09.2006]

offline than the ones using Apache. As the difference between Apache and Netscape was insignificant, it was proved that open-source solutions have reliability as good as the most reliable proprietary software.[81]

Downtime	Apache	Microsoft	Netscape	Other
September	5.21	10.41	3.85	8.72
October	2.66	8.39	2.80	12.05
November	1.83	14.28	3.39	6.85
Average	3.23	11.03	3.35	9.21

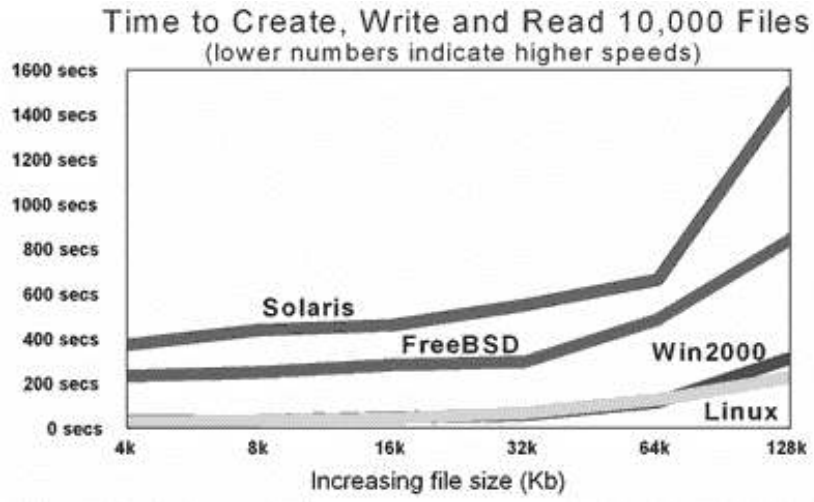
Table 2.2.1 Web sites downtime

Sourced from Wheeler D. A. *Why open-source software? Look at the numbers!* [Online] Available at: http://www.dwheeler.com/oss_fs_why.html [Accessed on 17.09.2006]

After an in-depth analysis conducted by the authors of "Open Source Software Development Should Strive for Even Greater Code Maintainability", it was concluded that the open-source software *code quality appears to be at least equal and sometimes better than the quality of [closed source software] code implementing the same functionality.*[82]

Open-source programs are generally faster and scale larger.[83] Linux has a huge ability to run large networks with many users. It is used in 78% of the world's 500 fasters computers.[84] According to the tests performed by Sys Admin magazine, GNU/Linux beat Solaris, Windows 2000 and FreeBSD on exactly the same hardware.[85]

[81] *Presserelease, 7th February 2000* [Online] Available at: http://web.archive.org/web/20011011215009/http://www.syscontrol.ch/e/news/Serversoftware.html [Accessed on 17.09.2006]
[82] Wheeler D. A. *Why open-source software? Look at the numbers!* op. cit.
[83] *Features and quality* op. cit.
[84] Lyons D. *Linux Rules Supercomputers* [Online] Available at: http://www.forbes.com/home/enterprisetech/2005/03/15/cz_dl_0315linux.html [Accessed on 17.09.2006]
[85] Rothman J. B. and Buckman J. *Which OS is Fastest for High-Performance Network Applications?* [Online] Available at: http://www.samag.com/documents/s=1148/sam0107a/0107a.htm [Accessed on 17.09.2006]



Graph 2.2.1 Time to create, write and read 10000 files

Sourced from Rothman J. B. and Buckman J. *Which OS is Fastest for High-Performance Network Applications?* [Online] Available at: http://www.samag.com/documents/s=1148/sam0107a/0107a.htm [Accessed on 17.09.2006]

eWeek's test proved that MySQL was comparable to the proprietary Oracle database program, although the open solution costs much less.[86] What is more, the comparisons of Mac OS X and GNU/Linux showed that systems based on Linux are 5-8 times faster on server tasks, specifically using MySQL.[87]

Adapting to the changing conditions and improvement is no longer an issue when one uses open-source software. Inspections and verifying the correctness of algorithm becomes possible, as well as changing everything that does not suit the needs. When the program needs to be re-done or tuned, as the source code is freely available everyone is allowed to do it. Each distribution of Linux offers a customized operating system dependant on the targeted market, not to mention the existence of two graphical interfaces for Linux – KDE and GNOME.

The possibility of forking the open-source project offers the ability of greater technical customization. The code base may be divided into subgroups managed in various ways. In the case of reached consensus, a reunification is always possible. A perfect example of forking are the stable and experimental versions of the GNU/Linux operating system that

[86] Dyck T. *Server Databases Clash* [Online] Available at: http://www.eweek.com/article2/0,3959,293,00.asp [Accessed on 17.09.2006]
[87] *No more mysteries: Apple's G5 versus x86, Mac OS X versus Linux* [Online] Available at: http://www.anandtech.com/mac/showdoc.aspx?i=2436 [Accessed on 17.09.2006]

enable trying out edge technologies or just tested features. Furthermore, forking introduces competition between versions, as branches compete for users with very similar products.[88]

Software solutions always need some expertise to deploy and maintain. Open-source software has changed much since the 1980s. It is no longer that challenging when it comes to installation. Open solutions are nowadays as easy to deploy as proprietary applications, since they are equipped with installers, suggested configurations and graphical interfaces. Furthermore, the overall user-friendliness of open-source software increased since much of open solutions derive from the interface of popular proprietary solutions. Open-source training programs of all possible levels are provided not only by professional vendors such as Novell or Red Hat, but also by independent training centers. Having in mind various forums, mailing lists and enormous amounts of books, gaining knowledge about open-source solutions has never been so easy.

2.3 Legal aspects

Legislation provides mainly four mechanisms for protecting intellectual property - copyright, patents, trademarks and trade secret. Copyright is a set of rights regulating the form of material expression and not the idea or concept itself.[89] It is literally the right to copy. The exclusive rights granted by a state to an inventor in exchange for public disclosure of a process which is new and industry applicable are called patents.[90] A trademark is a sign – usually a logo, phrase or image - used by a business to uniquely identify itself and its products/services.[91] The fourth mechanism – trade secret – is the practices and designs used by business in order to obtain advantage over direct competitors.[92] Trade secret may not be applicable to the software as a product, since the before mentioned compatibility would be not possible to implement.

Open-source licenses are based on the copyright law. The programmers decide that their specific rights are conveyed to everyone, giving them the right to copy, modify and distribute the software under specific conditions. Some state that one may find plagiarized code in open-source software. However, it is claimed that copying code from proprietary to

[88] *Free Software/Open Source: Information Society Opportunities for Europe?* op. cit.
[89] *Copyright* [Online] Available at: http://en.wikipedia.org/wiki/Copyright [Accessed on 17.09.2006]
[90] *Patent* [Online] Available at: http://en.wikipedia.org/wiki/Patent [Accessed on 17.09.2006]
[91] *Trademark* [Online] Available at: http://en.wikipedia.org/wiki/Trademark [Accessed on 17.09.2006]
[92] *Trade secret* [Online] Available at: http://en.wikipedia.org/wiki/Trade_secret [Accessed on 17.09.2006]

open software is unlikely, since it is easy to detect violation due to the open format and to prosecute the lawbreaker due to the record of code submission.

Most of the open-source software is dependant on the validity of open-source licenses. Some claim that the GPL-alike licenses – the most popular ones in the open-source environment – are unenforceable. However, courts find those licenses valid. For example, in April 2004 the German court decided that distribution of a Sitecom product must be stopped, as the product was derived from the GPL but did not comply with the license.

The EULA click-through licenses that are described as being contrary to open-source software licenses may contain terms dangerous for the end-users. According to Electronic Frontier Foundation, EULAs may forbid comparing a proprietary product with another software and publicly criticizing it. Moreover, it may contain clauses concerning automatic updates through contacting a third party without notifying the user, potentially compromising privacy and security. EULAs often limit the usage of products that evaluate the performance of the software or that can be used to uninstall all or part of it. Furthermore, agreeing to the written terms may mean one agrees to terms that will appear in the future.[93] Licenses including such terms limit user rights and ought to be carefully revised before agreeing. Moreover, most proprietary software licenses forbid lawsuits in case of software malfunction.

Bryan Pfaffenberger in his article "A Senior Microsoft Attorney Looks at Open-Source Licensing" states that *With open-source software (...) you are, in principle, walking into the deal with your eyes wide open. You know what you're getting, and if you don't, you can find someone who does. Open-source licenses enable the community of users to inspect the code for flaws and to trade knowledge about such flaws, which they most assuredly do. Such licenses allow users to create derivative versions of the code that repair potentially hazardous problems the author couldn't foresee. They let users determine whether the program contains adequate safeguards against safety or security risks.*[94]

Open-source software is said to protect users from licensing litigation. Proprietary vendors impose complex license management mechanisms and create increased legal risk for users who do not comply – intentionally or not - with the given rules. For example, the Business Software Alliance (BSA) operating in the USA and sponsored by Microsoft,

[93] Above paragraph according to Newitz A. *A User's Guide to EULAs* [Online] Available at: http://www.eff.org/wp/eula.php [Accessed on 17.09.2006]
[94] Pfaffenberger B. A Senior Microsoft Attorney Looks at Open-Source Licensing [Online] Available at: http://www.linuxjournal.com/article/5073 [Accessed on 17.09.2006]

Macromedia and Autodesk may even get a court order to storm the offices with marshals looking for unregistered software.[95] With open-source software there is no risk of illegal copies, license audits and anti-piracy measures. However, open-source software needs limited license management, as its quality depends on updates.[96]

Furthermore, using proprietary software may involve getting the permission from software vendor to sell a business unit using the applications. In the case of Kmart selling Bluelight.com to United Online Inc., Microsoft stated that *The licenses that debtors [Kmart] have of Microsoft's products are licenses of copyrighted materials and, therefore, may not be assumed or assigned with[out] Microsoft's consent*.[97]

The best known suit against open-source software vendor is SCO Group v. IBM. The SCO Group demanded $5 billion for IBM contributing SCO intellectual property to the code base of the Linux operating system. The SCO Group has threatened the members of Fortune 1000 and Global 500 companies with liability when using Linux. Legal actions against SCO were started by Red Hat and IBM, while SCO additionally sued Novell, AutoZone and DaimlerChrysler. The proper trial will begin in February 2007 as till now no evidence in favor of SCO was produced.[98]

Having in mind the SCO v. IBM case, one should remember that the most of open-source software is not involved in any legal dispute. Business analysts often perceive the claims towards Linux as a way to stop a competitor through courts. Moreover, Pillsbury Winthrop LLP stated: *The suggestion that users of [open-source] software are more likely to be sued for patent infringement than those that use proprietary software (...) does not appear supported by actual experience. It is interesting to note that while Microsoft has had several dozen patent infringement lawsuits filed against it in the past few years, none have been reported against Linux, the most popular of all [open-source] programs*.[99] Concluding, there is a larger risk of being sued over license counting by a proprietary vendor than over intellectual property issues connected to open-source software. The issue of intellectual property is solved between vendors, and not vendors and customers,

---

[95] Wheeler D. A. *Why open-source software? Look at the numbers!* op. cit.
[96] *Deployment and maintenance* [Online] Available at: http://www.netc.org/openoptions/pros_cons/deployment.html [Accessed on 17.09.2006]
[97] *Microsoft says MS Licenses prohibit company sale* [Online] Available at: http://librenix.com/?inode=2402 [Accessed on 17.09.2006]
[98] *SCO v. IBM* [Online] Available at: http://en.wikipedia.org/wiki/SCO_v._IBM_Linux_lawsuit [Accessed on 17.09.2006]
[99] *Pillsbury Winthrop Attorney Clarifies Latest Open Source Myths Following Government Guidance and Microsoft Posturing* [Online] Available at: http://www.prnewswire.com/cgi-bin/stories.pl?ACCT=104&STORY=/www/story/12-15-2004/0002633825 [Accessed on 17.09.2006]

although it could be technically proceeded this way. Proprietary software vendors do not aim at suing everyone who infringes their property, as litigation against users might discourage further product purchase. Because of that unwritten rule Eolas Technologies sued Microsoft over the web browser patent and not the Internet Explorer users.[100]

2.4 The main drawbacks of open-source software

Nowadays one of the biggest obstacles to the adoption of open-source solutions is the switching cost. It may be even high enough to outweigh savings on license fees. Bernard Golden in his article "The ROI of open-source" argues that return on investment in the case of switching from proprietary to open-source software requires retraining and hiring new personnel, which makes it the worst return-on-investment scenario.[101] It is said that a company should deploy Linux in a new operation rather than as a replacement for Windows. Gartner's analysts show that the Linux TCO drops radically when one avoids migration and installs Linux in the first place.[102] The figures show that migration from another operating system and porting software written for the old one are the two largest costs. *Knowledge workers use PCs to run diverse combinations of applications. For these users, migration costs will be very high, because all Windows applications must be replaced or rewritten.*[103] Many organizations already bought specialized software that will only run on Windows[104] and strong financial grounds will be needed to persuade the organizations that they should throw away proprietary software in which they already invested huge amounts of money. Enormous amount of applications and documents that are found in computers need proprietary software to run effectively and as Gartner research group predicts *Through 2008 Linux will not be suitable for all desktops in 80% of companies*.[105] In many organizations there is a lack of open-source expertise which also

---

[100] *Open-Source Legal Risk Management in the Enterprise* [Online] Available at: http://stephesblog.blogs.com/papers/Optaros_FOSS_Risk_Mgmt_SWalli_051906.pdf [Accessed on 17.09.2006]
[101] Golden B. *The ROI of open-source* [Online] Available at: http://www.computerworld.com/developmenttopics/development/webservices/story/0,10801,102638,00.html [Accessed on 17.09.2006]
[102] Miller R. *IT analysts' influence on open source adoption* [Online] Available at: http://trends.newsforge.com/trends/04/11/03/181215.shtml [Accessed on 17.09.2006]
[103] *Linux on the Desktop: The Whole Story* [Online] Available at: http://www.gartner.com/DisplayDocument?id=406459 [Accessed on 17.09.2006]
[104] *Linux on the Desktop: The Whole Story* op. cit.
[105] Cane A. Migration costs are the hitch., *Financial Times*. September 21, 2005

complicates implementation of the trend. Open-source solutions are still seen as a risk that cannot be taken just because of low-cost computing.

It is noted by Gartner analysts that *lost productivity stemming from learning curves and compatibility can eat up direct-cost savings when moving to Linux on the desktop.*[106] Time needed to master a new program is usually longer as it may be difficult for some to get used to open solutions.

Putting together open and closed solutions may be a problem because some software is not compatible with open-source applications. Backward (or downward) compatible software may replace the older one and interoperate with products designed for the older product. Forward compatible products have the ability to accept input intended for future versions of themselves. A typical and highly expected behavior of forward compatible systems is ignoring data or instructions not recognized, e.g. a web browser ignores HTML tags that are not recognized. Backward compatibility is easier to achieve since the input data format is already known, whereas forward compatible products need to cope with unknown future formats and features. What is more, it is possible to distinguish between binary and source compatibility. Source compatibility requires recompilation, however, no changes to the source code need to be implemented. Binary compatibility means that programs work correctly with the newer version of the library without additional recompilation. Software incompatibility may be dealt with by the means of additional tools, e.g. emulators that simulate an older platform or programs that convert files from one format to the another. In the heterogenic internet environment incompatibility is resolved by the implementation of so called virtual machines. Furthermore, as open-source solutions are still a minority, they may limit the amount of hardware that may be ported to them.

Unfortunately, there is also no guarantee that the development of an open-source project will happen. The life of a project is mainly based on the interest of developers. One is not able to predict whether it will reach a usable stage or live long. If there is no backing of a company or enough programmers, the project easily dies in the stage when source base is immature. A self-sustaining level needs to be reached in order to the project proceed by itself.[107]

---

[106] *Linux on the Desktop: The Whole Story* op. cit.

[107] *Free Software/Open Source: Information Society Opportunities for Europe?* [Online] Available at: http://eu.conecta.it/paper/Perceived_disadvantages_ope.html [Accessed on 17.09.2006]

Since open-source project are not widely advertised, it may be difficult to know whether a project exists and if so, what is its current status. However, a new business opportunity emerges since there is a need for services maintaining information about location and evaluating open-source projects. The web services such as Ohloh[108] or SpikeSource[109] are there to facilitate choosing an open-source project.

When there are problems arising with open-source software, as most of it having no contractual obligations or full commercial license, one is not able to point a single liable person or organization that is responsible. Significant problems may be also connected with the intellectual property issue. It is rather difficult to know whether an algorithm or software is patented in other countries. Moreover, the lag between application and issue of the patent might result in not knowing whether someone else's claims have been infringed. Because of that an open-source community may be accused of intellectual property infringement. To deal with that problem special switches and patches that disable questionable in some countries code parts are enclosed in the software.[110] Some say that the source code itself is not an executable device and because of that is not covered by patent law. The viability of this idea is not proved and the problems are still left existing for the everyday user.

Robert Gomulkiewicz in his essays states that distribution without any warranties shifts all the risk from the open-source licensor to the user. What is more, Gomulkiewicz claims that the enormous amount of licenses is confusing and holds open-source software developers at a disadvantage.[111]

Scott A. Hissam and Daniel Plakosh's in their article "Trust and Vulnerability in Open Source Software" point that attackers can learn about vulnerabilities in a closed source program from patches made to an open-source program performing the same function. They described an example of Linux developers providing a patch for vulnerability and at the same time letting attackers assume that similar problem might occur in Windows.

In order to guarantee open-source software security, the code needs to actually be reviewed. Amount of review may be reduced by: 1) being a niche or rarely used product, 2) having few developers, or 3) using a non-popular programming language. Furthermore,

---

[108] http://www.ohloh.net/
[109] http://www.spikesource.com/
[110] *Free Software/Open Source: Information Society Opportunities for Europe?* op. cit.
[111] Frost J. J. *Some Economic & Legal Aspects of Open Source Software* [Online] Available at: http://opensource.mit.edu/papers/frost.pdf [Accessed on 17.09.2006]

people reviewing and developing it need to have the skills to write secure programs. Programs are secure when the fixes are quick and well-distributed. Unfortunately, distribution in the case of open-source software may not be smooth.[112]

2.5 Making open-source ready for the enterprise

The adoption of open-source software in big companies is growing very rapidly. According to Gartner's vice president – Mark Driver – open-source software is being more and more often recognized as an important part of the IT landscape. Crucial is not the fact that companies use open solutions, but the fact that they are doing it formally.[113] As open-source software is becoming institutionalized, it *is starting to look much less like a curiosity of digital culture and more like an enterprise, with its own risks and rewards.*[114]

Definitely, open-source software enables organizations to save on license fees and hardware. It is a reliable solution that offers faster resolution of bugs that is strictly connected with security, transparency, scalability and speed. Moreover, open software gives users freedom and flexibility, as well as helps in avoiding vendor lock-in.

However, some issues are still to be solved so that enterprises feel 100% confident when using open solutions. The before mentioned Mark Driver claims that to avoid costly problems, a company should use open software coming from a reliable vendor that is able to guarantee long-term professional support. Still, the biggest obstacle for companies to the adoption of open-source solutions is the switching cost that may outweigh potential future savings.

The fact remains that adopting a soft approach to open-source is the best idea for companies that want to take advantage of the benefits but are afraid of the risks that the software implies. Trying open software on non-critical applications and keeping the mainstream ones on proprietary systems seems the best, as co-existence makes a lot of sense, keeping in mind that future will bring development of both – open and proprietary software solutions.

---

[112] Paragraph according to Wheeler D. A. *Is open-source good for security?* op. cit.
[113] *Does Open Source Deserve a Place in Your IT Portfolio?* Robin Miller http://trends.newsforge.com/article.pl?sid=04/10/19/1518218&tid=139&tid=150&tid=111&tid=37&tid=132&tid=18&tid=29
[114] Special Report: Open, but not as usual - Open-source business *The Economist*. London. Mar 18, 2006

Nevertheless, according to the InformationWeek research, there is a long road ahead of open-source software to win businesses' hearts.

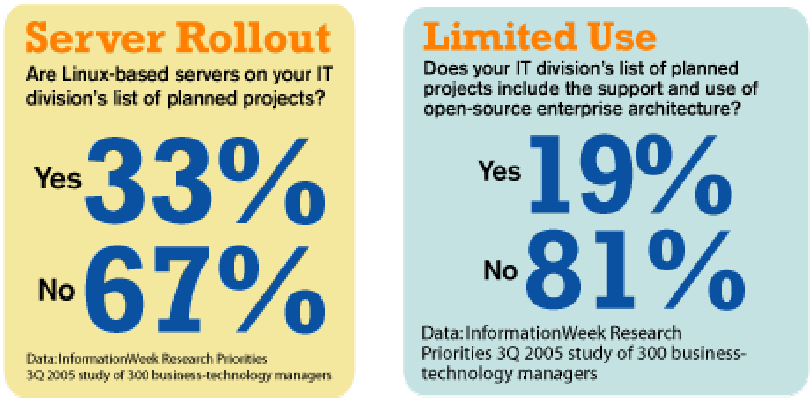


Illustration 2.5.1 Planned projects involving open-source software

Sourced from *Open Source Goes Corporate* [Online] Available at: http://www.informationweek.com/software/showArticle.jhtml?articleID=171200352&pgno=1&queryText= [Accessed on 23.10.2006]

Chapter 3:
Practical applications of open-source software in business

3.1 Methodology of the research

The conducted research is based on the analysis of 10 companies: Lapeyre, LVM, Renault, Skanska, Statoil, Lycos Europe, neckermann.de, Sony, Suzuki and Yahoo! Finance. Application of two open-source solutions will be examined – Red Hat Enterprise Linux and MySQL database. The first five enterprises will be investigated with respect to the implementation of Red Hat Enterprise Linux; the further five will be investigated with respect to the implementation of MySQL database.

The enterprises are examined in alphabetical order for each of the open-source solutions. They were chosen so that different industries are represented. They are all worldwide known, representing Europe, Asia, as well as the United States of America. The companies discussed are pioneers of implementing the newest technology, not afraid to look for more efficient IT solutions. Open solutions became the answer to their problems and helped them reach the financial and managerial goals.

Each case study will be examined according to a following schema:

1) basic information about the company,
2) identification of the problem,
3) implementation of the open-source solution,
4) results and future plans.

The analysis will be based on secondary sources including corporate web sites, annual reports, case studies presented by Red Hat and MySQL, as well as press releases concerning the enterprises.

Moreover, each set of case studies will be preceded by a description of all offered solutions of the open-source vendor, accompanied by a short description of the software company and its major successes, as well as a table presenting the overview of analyzed enterprises.

3.2 Red Hat Solutions[115]

Red Hat is the leading Linux provider and at the same time the most recognized Linux brand in the world. Founded in 1993, it employees approximately 1150 employees worldwide. Red Hat provides operating system platforms with middleware, applications and management solutions, support, training programs that operate in 60 worldwide locations and consulting services. The solutions offered by Red Hat include:

- Red Hat Enterprise Linux AS for high-end and mission-critical systems,
- Red Hat Enterprise Linux ES for small- and mid-range servers,
- Red Hat Enterprise Linux WS for technical/design workstation clients,
- Red Hat Desktop for volume client system deployments,
- Red Hat Network – a complete systems management platform for Linux,
- Red Hat infrastructure solutions that extend the benefits of open-source (Red Hat Global File System, Red Hat Cluster Suite, Red Hat Developer Suite),
- JBoss middleware solutions - JBoss Enterprise Middleware Suite, JBoss Application Server, Hibernate, JBoss Subscription.

The server features include among others: leading open-source databases - PostgreSQL, MySQL; environments for C, C++, Java, Fortran with Perl, Python, CVS and Emacs; FTP, NFS, and Samba (CIFS) file servers; IMAP/POP mail servers with Cyrus, Sendmail, spamassassin; a complete suite of network servers and firewall; features including SSL, IpSec, MAC/DAC; Apache web server. The client features include OpenOffice.org office suite, extensive Microsoft interoperability, several mail clients, applications for multiple audio/video formats, plug-and-play device support, GNOME-based interface with optional KDE, Firefox web browser, X Windows system.

Among the customers of Red Hat are Amazon.com, AOL, Merrill Lynch, Credit Suisse First Boston, DreamWorks, Lithonia Lighting, VeriSign, Charles Schwab, Lehman Brothers, UBS Warburg, Morgan Stanley and Goodyear. At the same time, its key partners are IBM, Dell, HP, Oracle, Sun, Fujitsu, Intel, NEC, Hitachi and BEA.

---

[115] Section according to *Corporate Fact Sheet for Red Hat* [Online] Available at: http://www.redhat.com/about/companyprofile/facts/ and *Red Hat Enterprise Linux Details* [Online] Available at: http://www.redhat.com/rhel/details/ [Both accessed on 21.10.2006]

The company has been awarded by the CIO Insight Magazine with the Most Valued Vendor award in 2006 and 2005. In 2002 the CNET survey among IT professionals concerning relevance of companies to their business in the next 5 years showed that Red Hat has been ranked as number 2, ahead of IBM, Sun, Dell, Cisco, HP, Oracle and Apple.

Table 3.2.1 Overview of the analyzed enterprises (Red Hat solutions)[116]

Enterprise	Industry	Goal	Solution
Lapeyre	Retail	Deploying a highly-available platform with long-term support for a critical sales and customer services tool	*Platform:* Red Hat Enterprise Linux ES v 2.1; *Hardware:* Dell Poweredge 2650; *Applications:* Cameleon, Oracle9*i*; *Systems Management:* Red Hat Network Satellite
LVM	Financial services, insurance	Finding a Linux-solution backed by enterprise-level support; vendor-independence	*Clients:* Red Hat Desktop and IBM ThinkPad notebook computers; *Servers:* Red Hat Enterprise Linux and Fujitsu Siemens, Sceneo & Futro hardware; *Systems Management:* Red Hat Network
Renault	Automobile	Reconfiguration and consolidation of micro-sites on a single platform	*Platform:* Red Hat Enterprise Linux; *Web Hosting Implementation:* Rackspace
Skanska	Construction	Reduction of hardware costs and improvement of performance	*Platform:* Red Hat Enterprise Linux v. 3; *Hardware:* Dell PowerEdge servers, including 6650s; *Applications:* Oracle E-Business Suite 11i, Oracle10g RAC
STATOIL	Oil and gas	Standardizing from seven Unix variants to Linux	*Platform:* Red Hat Enterprise Linux v. 2.1 and 3; *Hardware:* Intel and AMD; *Applications:* Oracle Database, BEA WebLogic, Schlumberger Eclipse, Landmark drilling software

[116] Table according to selected case studies presented at *Success Stories* [Online] Available at: http://www.redhat.com/rhel/informationcenter/successstories/ [Accessed on 21.10.2006]

3.3 Behind the scenes with Red Hat[117]

The company

The Lapeyre Group is a French manufacturer and distributor of home improving products. The company is a leader in Europe with 2600 employees and 250 retail outlets in 6 European countries. It offers pre-made and custom-designed fixtures and fittings for inside and outside parts of buildings. Its success is based on close relationships with customers and the understanding of their needs. Lapeyre uses sales management tools such as Cameleon Configurator to deliver made-to-order products. Cameleon is a solution helping to sell complex products in multi-channel environments. Cameleon may be used to browse catalogs, customize products and capture orders. Moreover, the application transfers data such as names and ranges for a customer order in the Enterprise Resource Planning system from the points of sale to one of the 13 factories.

The open-source solution

In order to benefit from the advantages of the new version of ACCESS COMMERCE Cameleon – a critical tool for Lapeyre sales force, the company needed to migrate to Oracle9i database. At the same time Lapeyre decided to introduce more changes and support the Oracle environment by Red Hat Enterprise Linux ES to be able to manage the increase of catalogue data. As Cameleon is the basic application for Lapeyre, it is essential that the configurator is highly available and excels in performance. Those were

---

[117] Section according to case studies presented at *Success Stories* [Online] Available at: http://www.redhat.com/rhel/informationcenter/successstories/,
Cameleon Lapeyre [Online] Available at: http://www.access-commerce.com/automne_modules_files/standard/public/p12_2795f03268b3d5a054dcbc40ded2719cLAPEYRE_EN.pdf,
Renault in brief [Online] Available at: http://www.renault.com/renault_com/en/main/10_GROUPE_RENAULT/10_Renault_en_bref/,
Brief Facts [Online] Available at: http://www.skanska.com/skanska/templates/page____270.aspx,
Statoil in brief [Online] Avaiable at: http://www.statoil.com/STATOILCOM/SVG00990.nsf?opendatabase&lang=en&artid=3FED33ECC77666314125665D004E05E3 [All accessed on 21.10.2006]

only possible while implementing Red Hat Enterprise Linux, that was chosen because of its stability, availability, effective support and full compatibility with already existing solutions.

The IT culture of Lapeyre is mostly Unix-based. The company's technicians tested various solutions as they needed to find a reliable, high performance system accompanied by long-term support. Eventually, they have chosen Red Hat Enterprise Linux ES that was specially packaged for Lapeyre. Automatic installation has been included, so that the technicians needed only 15 minutes to connect to a server, roll out a master and network the hardware.[118] All the stores were equipped in no more than 6 months. Lapeyre installed also Red Hat Network Satellite – a program that enables downloading operation system updates during the night and catalogues them so that the network has accurate data.

The Head of Lapeyre Systems and Database Management System – Philippe Rennes, says the company appreciates that the Linux operating system was provided by professionals. The service part of the installation was for Lapeyre very important – availability of Red Hat technicians, response time and relevance were the key factors. Furthermore, Lepeyre sales force was not heavily impacted by the changes.

It was explained that Lepeyre decided to implement Red Hat Linux because the enterprise wanted to be independent when it comes to software and avoid the proprietary model. Moreover, the company was aiming at having one reliable collaborator that would provide long-term support. The technicians with previous Unix experience were able to develop their skills with support guaranteed by Red Hat, making the implementation even more pleasant.

After the first implementations in 2004 all stores have now the same computing systems that enable efficient and cost effective system management.

[118] According to Philippe Rennes, Head of LVM Systems and Database Management System

The company

LVM Versicherungen is one of the largest insurance companies in Germany. It was founded in 1896 and has headquarters in Muenster, where it employs 2200 people. The three million customers of LVM are locally supported by 2100 agencies across Germany, what is described as one of LVM's biggest strengths. The company is a pioneer in using Linux on desktops. In the year 2000 it developed its own version of Linux suited to the needs of an insurance company.

The open-source solution

LVM decided in the late 1990's that to obtain a basic competitive advantage, it needs to centralize the data management system. Till that time every agency held own database for which it was responsible. Data comparison and harmonization took lots of time and effort. Moreover, information was not identically available for headquarters and agencies; no one could avoid discrepancies. By using central data management system LVM wanted to avoid duplication of procedures and applications, as well as enable live access to information.

In 1998 LVM bought 1400 IBM 2800 Network Stations that were, however, no longer developed and supported after 2000. The company decided to implement Linux as it was ideally suited for Thin Client environment. Though, as the available distributions were too complex, LVM decided to create own distribution.

With time the number of work stations and functions needed has grown. It became difficult to maintain such a complex system. LVM decided that it is time to move to a new solution. After examining the alternatives, Linux was still the best option, however, the project was too big to be based on a community version. LVM needed professional support and a commercial service provider. In the first phase of the project 8500 systems had to be migrated.

LVM did not want to be dependent on one provider, which was fully possible with Red Hat that works closely with the Linux community and uses only code supported by it. The

desktops were supplied with Red Hat Network for systems management to centrally administer the roll out and management process. Red Hat Desktop was used on PCs and mobile computers. All staff was able to always have the same information thanks to connecting the agencies to headquarters by the WAN.
LVM felt more confident as Red Hat offers Open Source Assurance Program that enables using Red Hat Enterprise Linux and Red Hat Desktop even if patent disputes arise. It was easy for the insurance company to decide on standardizing areas of its IT infrastructure to Red Hat products.

The company

Renault Group is present in 118 countries designing, developing, manufacturing and selling vehicles. Renault employs 130 000 people who share a common vision of competitiveness, customer satisfaction, product quality and innovation. The company expands internationally with Dacia, Renault Samsung Motors and the Alliance with Nissan. Renault is the most popular automobile brand in Europe and its vehicles are considered the safest in the world.

The open-source solution

The primary source of information for the customers of Renault are websites tailored for each market. In the United Kingdom the automobile manufacturer decided to consolidate impractical UK individual micro sites that were hosted by creative agencies. The front-line web-based marketing strategy needed to be managed more efficiently in order to cut costs and use server space more economically. The outdated server technology that did not provide security, stability or future-proofing needed to be replaced in order to keep control of the web-based operations.
Renault decided to use a solution from Rackspace Managed Hosting that is based on Red Hat Enterprise Linux. Rackspace has one of the biggest bases of Red Hat technology and

provides assistance and expertise thanks to Red Hat Certified Engineers. The web servers configurations were consolidated and full scalability accompanied by the ease of IT infrastructure management was achieved. Red Hat technology was able to provide Renault with a system that guarantees full-time availability of online information to customers.
Ian Collins – the Operations Manager of Renault UK, states that with the new solution probability of web-based problems has been reduced. Rackspace uses the before mentioned Red Hat Network to manage its Red Hat servers, so that its clients use systems with the latest security patches and management tools. All in all, Renault UK has lowered long-term IT costs and increased control over its web presence.
With the new solutions Renault UK was able to completely focus on its core business and use the money saved in the IT department to improve its products and services.

SKANSKA

The company

Skanska is a Swedish company founded in 1887, a leader in the constructions industry. The company's home markets are Sweden, the USA, the UK, Denmark, Finland, Norway, Poland, the Czech Republic and Argentina. Skanska employs around 54 000 people and aims at providing high quality accompanied by low costs.

The open-source solution

The IT needs of Skanska are served by wholly-owned subsidiary Skanska IT Solutions. The subsidiary operates and manages Oracle E-Business Suite for Skanska Sverige AB that serves 5000 total and 1000 simultaneous users performing 6000 tasks every day. Because of that, high availability and security are the most important for Skanska IT Solutions.
In the late 1990's Skanska deployed Oracle E-Business Suite on Sun Solaris. With upgrading to Oracle11i, the company decided to take full advantage of the new database and change the operating platform, as well as hardware. Upgrading the Solaris servers would be too costly, so a cost-effective alternative was to be found.

Skanska IT has already had some Unix experience, so migrating to Linux required minimized training. Red Hat's reputation was of course one of the factors helping in convincing Skanska of migration to Linux. The combination of Oracle11i, Red Hat Enterprise Linux and Dell hardware enabled increase in performance and cost-effectiveness. What is more, Veritas Storage Foundation and Veritas NetBackup provided security measures.

The biggest challenge for Skanska was the alignment of support matrices of vendors, so that the company was sure that suppliers and versions are fully compatible, as well as the Service Level Agreements are met.

Skanska decided to introduce modules standardization, in order to eliminate the risk of system conflicts and unanticipated expenses. The company noticed not only increase in performance, but also estimated future operating costs reduction by 30%.

The company

Statoil is a Norwegian gas and oil company represented in 33 countries. The company employs around 26 000 people, out of which nearly 50% work outside of Norway. Statoil is a leading retailer of petrol and oil products in Scandinavia, Poland, Ireland, Russia and the Baltic states. The company is one of the largest sellers of crude oil and a substantial supplier of natural gas to the European market. Its goal is to excel in the environmental, social and financial areas.

The open-source solution

In the late 1990's Statoil used 7 different versions of proprietary Unix and Microsoft Windows. Standardization of the server platforms became the ultimate goal. It was decided that migration to Linux was possible and a group of experts advised conducting it first on application severs and databases. The next step would be using Linux to analysis and simulation based on huge data sets. The standardization was to bring more

efficient system administration, reduction in costs, but maintain the strict requirements concerning application quality and availability.

Statoil has chosen Red Hat because of the openness of the operating system and the big market share. The competition was not able to beat Red Hat because they lacked competency among third party suppliers and documentation. What is more, the number of vendors basing its products on Red Hat Linux source code was high enough to guarantee independence.

In one year Red Hat Linux became the preferred platform. First the internet/intranet services were migrated, then Linux became the base for HPC needs. Statoil owns over 1000 nodes that utilize Red Hat Enterprise Linux to analyze data and create simulations. Its largest cluster was created in 2004 with 512 nodes to process seismic data and ranked among 100 most powerful computers in the world.

Naturally, Statoil had some unexpected technical issues that were, however, corrected with time. When the company entered the first stage of migration, there were some problems with third party software being ported to Linux. Though, with time more and more software companies realized that Linux is a strategic platform that should not be excluded.

Statoil has learned that well-tested kernels are better than the latest technology and that one should carefully define version handling, platform management and certification practices.

Statoil used the regular support and the assistance of Technical Account Manager which helped much in technical discussions between the company and application software vendors.

70% of Statoil services are now run on Linux with four out of seven Unix versions used. Not only 50% cost savings were reported, but also significant performance improvements with applications used for reservoir simulation working 10-40 times faster.

3.4 MySQL Solutions[119]

MySQL AB – a company founded in Sweden – offers a family of high performance, affordable database servers and tools. It is an owner of the MySQL server source code, the MySQL trademark and the mysql.com domain.

The company aims at providing a less complicated solution for widespread application at a reduced total cost of ownership. MySQL with its superior speed accompanied by reliability and ease of use, eliminates the problems associated with downtime, maintenance, administration and support. It is said to provide the reduction of database licensing costs by over 90%, to cut system downtime by 60%, to lower hardware expenditure by 70%, reduce administration, engineering and support costs by up to 50%.

MySQL is a part of open-source enterprise software stack called LAMP (Linux, Apache, MySQL, PHP / Perl / Python).

The solutions offered by MySQL include:

- MySQL Enterprise - MySQL database software, services and support (MySQL Enterprise Server, MySQL Network Monitoring and Advisory Services, MySQL Production Support),
- MySQL Cluster - fault tolerant database clustering architecture for deploying highly available mission-critical database applications,
- MySQL Embedded Database – solution for OEMs/ISVs wanting to embed or bundle a reliable and high-performance relational database,
- standards-based drivers for JDBC, ODBC, and .Net enabling developers to build database applications,
- open-source visual database tools (MySQL Administrator, MySQL Query Browser, the MySQL Migration Toolkit),
- MaxDB - open-source database certified for SAP/R3,
- MySQL training courses,
- MySQL Certification Program,
- consulting services,

[119] Section according to *About MySQL AB* [Online] Available at: http://www.mysql.com/company/, *MySQL Products* [Online] Available at: http://www.mysql.com/products/, *MySQL Services* [Online] Available at: http://www.mysql.com/services/ and *MySQL Awards* [Online] Available at: http://www.mysql.com/why-mysql/awards/ [All accessed on 21.10.2006]

- MySQL Technical Support.

Among the customers of MySQL are companies and organizations ranging from travel to defense, just to mention Yahoo!, Lloyds TSB Bank, ClassMates, UNICEF, DaimlerChrysler, Braun, Toyota, Yamaha, CNET Networks, Hoover's, Yves Rocher, Intel, Motorola, Lufthansa Systems, Nokia and many more.

MySQL 5.0 has been awarded Software Development Magazine Productivity Award in 2006 as the product *that have "jolted" the industry with their significance and made the task of creating software faster, easier, and more efficient.*[120] Moreover, in 2005 MySQL has been awarded the Members Choice Award: Database of the Year by LinuxQuestions.org and in 2004 MySQL AB has been chosen by Red Herring Magazine as one of the industry's most innovative companies.

[120] *MySQL Awards* [Online] Available at: http://www.mysql.com/why-mysql/awards/ [Accessed on 21.10.06]

Table 3.4.1 Overview of the analyzed enterprises (MySQL solutions)[121]

Enterprise	Industry	Goal	Solution
Lycos Europe	Online products and services	Decrease of the total cost of ownership while keeping the quality at an unchanged level	*Hardware:* Dell 6650; *OS:* Debian Linux; *Database:* MySQL Server; *Biggest database size:* 25 GB; *Number of concurrent users:* max 25 000
neckermann.de	Retail	Finding a solution that would add performance, scalability and reliability	*Hardware:* Dell PowerEdge 1855 Blade; *OS:* Suse Linux Enterprise Server; *Web server:* Apache; *Database:* MySQL Cluster 4.1; *Language:* PHP; *Database Size:* 6 GB; *Number of Users:* 500 000 per day
Sony	Electronics	Moving testing records from paper to electronic version, minimizing time needed to submit testing requests and look up certificates	*Hardware:* Genesi Pegasos *OS:* Debian GNU/Linux *Database:* MySQL Server *Web Server:* Apache *Language:* PHP, Visual Basic, C *Database Size:* 1 GB, 70 000 records
Suzuki	Automobile	Creating a tool enabling the sales people to obtain all needed product information and integrating it into the sales process	P.R.O. Kiosk Appliance *Hardware:* Dell Optiplex 260s and 270s; *OS:* Windows XP; *Database:* MySQL Server; *Language:* Visual Basic .NET; *Database Size:* 600+ MB, 200 000+ records, 50 tables Master Databases *OS:* Red Hat Linux (version); *Database:* MySQL Server
Yahoo! Finance	Financial online services	Finding a flexible and scalable enough solution to create Jointly Administered Knowledge Environment (JAKE)	*OS:* FreeBSD, Linux; *Database:* MySQL Server; *Database Size:* 25 GB

[121] Table according to selected case studies presented at *Case Studies* [Online] Available at: http://www.mysql.com/why-mysql/case-studies/ [Accessed on 21.10.2006]

3.5 Behind the scenes with MySQL[122]

The company

Lycos Europe operates a network of European web sites in nine languages – English, German, French, Italian, Spanish, Russian, Swedish, Danish and Dutch. Lycos offers online services and products, such as search, communication, content channels, Internet access, homepage building and online communities. The company generates revenue from advertising and fees for services. Every month it serves 3 billion pages to 25 million users.

The open-source solution

Lycos Europe was using a proprietary database on a cluster of HP Tru64 Unix servers. After some time it was clear that the company needs to find a hardware and database solution that would decrease its total cost of ownership. That is why it decided to migrate to the Linux operating system and MySQL database.

Lycos Europe was spending € 100 000 annually on database support and maintenance. With the development of ISP-services and WebServices it has chosen to architect the solutions on LAMP open-source stack (Linux, Apache, MySQL, PHP) accompanied by additional Java applications. After replacing the existing database solution with MySQL, 90% overall savings were observed. Because of that fact, all portal services were migrated from a proprietary solution to MySQL. What is very important, the biggest database having 25 GB of data was completely moved in less than one day. However, the whole migration process took 6 man-months of work; fortunately, without disturbances or unexpected surprises.

Nowadays Lycos Europe uses 100 MySQL servers to keep 1 Terrabyte of data. At peak time 1 GB of data is delivered to users per second and not more than 25 000 concurrent

[122] Section according to case studies presented at *Case Studies* [Online] Available at: http://www.mysql.com/why-mysql/case-studies/, *Suzuki* [Online] Available at: http://en.wikipedia.org/wiki/Suzuki,
2005 Annual Report [Online] Available at: http://www.shareholder.com/shared/dynamicdoc/YHOO/1183/YHOO.pdf [All accessed on 21.10.2006]

users are online. In order to guarantee its users 100% availability, Lycos paired MySQL servers on different machines. That way if one server goes down, the twin takes over all tasks without any disruption noted.

Thanks to the implementation of MySQL, Lycos Europe was able to move its databases from expensive symmetrical multi-processing machines to commodity boxes from Dell. What is more, the number of machines needed went down from 13 to 4. Lycos freed around € 200 000 annually and still expects to reduce administration costs. As MySQL may be maintained with minimal specialized resources and fewer machines, overall costs are reduced.

The company

neckermann.de is a German mail order company which launched its site in 1995. It is one of the first companies in Germany to offer products and services online. neckermann.de offers more than 140 000 items and additional services such as interactive product advisors, interior design advisors, online videos and 3D product viewers. It is one of the most trafficked German sites with 500 000 visitors every day.

The open-source solution

neckermann.de was running its web site on a proprietary database and SMP computers. The existing solution was running out of capacity, as the number of online shoppers was growing every day. A cost-effective way had to be found that would add performance, scalability and reliability, as the company was facing the Christmas 2005 season. neckermann.de decided to migrate to MySQL Cluster architecture that run on commodity dual-CPU Intel blade server hardware.

The Symmetric Multiprocessing Unix servers were storing user profiles and managing session data. If neckermann.de was to upgrade the existing system, it would have to purchase more SMP hardware and proprietary software licenses. Moreover, it would have to invest in people with high-cost DBA skills to administer the solution.

MySQL solution delivered the three most important factors demanded by neckermann.de: scalability, high availability and professional services. The MySQL Cluster parallel server architecture enables scalability in a near linear way, so that huge investments may be made to increase capacity. MySQL Cluster using synchronous replication enables replication of session data across multiple database nodes, making it easier to successfully complete a transaction even if a node fails. Professional Services offered by MySQL helped neckermann.de in deploying the solution as fast as it was possible and fine-tune the performance to meet the requirements.

Thanks to MySQL, neckermann.de was able not only to deliver customers a positive shopping experience, but also upsell additional products basing on customer shopping behavior. neckermann.de is able to provide fast and easy navigation, as well as ensure smooth completion of a transaction thanks to tracking user activity. Wait times that are critical for e-commerce have been reduced, as the company relies on MySQL Cluster for instant read and write data operations.

The company

Sony International GmbH in Germany runs the most modern electronic measuring facility. All electronic devices must be tested in accordance with emissions regulations, as only those that pass the tests may be sold in the European Union. Sony tests products not only for itself, but also for other large electronic companies. Annually thousands of products are tested ranging from Tvs to computers.

The open-source solution

The testing records of all tested electronic products were kept in a paper form so it was very time-consuming to submit a testing request or look up the certificate. Sony needed to find a solution that would satisfy its out-of-the-box requirements while keeping the records management simple.

Sony has chosen FileNet to solve the problem, however, it turned out that FileNet needed to be externally customized using Microsoft SQL and Active Server Pages, which would take months, cost much more that the original estimations and not be reliable enough. Because of the fact, Sony decided to evaluate further alternatives.

The company decided to create an extensive workorder system (FIS) built on the LAMP stack. Sony selected MySQL because of its reliability, price, low operating cost, multi-platform availability, ease of use, full text search that makes accurate and intuitive search possible, replication and excellent support. FIS was continuously available for employees, partners and resellers thanks to the extreme stability of MySQL. The implementation saved more than 90% that would be spent on licensing, maintenance and support of the initial Content Management solution. The system which took one developer 3 months to complete, is running on Debian GNU/Linux and one external website that enables verification of certificates by resellers and testing request submission.

Over the years Sony tested over 70 000 products which results, certificates and reports are stored in the FIS system.

The company

Suzuki Motor Corporation is a Japanese company producing automobiles, motorcycles, outboard motors and small combustion-powered engine products since 1909. Moreover, the company builds cars for other manufacturers, such as Subaru, Nissan and Chevrolet. Production facilities of Suzuki are located in 22 countries all around the world

with distributors in 140 countries. In the United States Suzuki markets motorcycles, ATVs and scooters. It has a network of 1150 dealerships in 49 states.

The open-source solution

In 2003 American Suzuki Motor Corporation decided to create Suzuki Sales Professional Retail Outlet (PRO) to provide detailed product information for the sales force and automate the sales process. Moreover, the PRO Kiosk System was aiming at letting prospective customers build their custom motorcycle in the dealership.

The company teamed up with Matrix Consultants who have chosen MySQL as the database with proven scalability, reliability and zero administration. The most critical requirements were ease of use and deployment. The project needed to be ready for Spring, so a prototype was developed in the Fall. After the success of the prototype, production and full deployment in dealerships followed. The prototype used Microsoft Access that, however, was not able to deal with a growing customer database reaching hundreds of thousands at each dealership. The database was further to be migrated to Microsoft SQL Server but the total cost of ownership did not meet the requirements even with the volume discounts. Moreover, it was very important for Suzuki that the database finally selected was dependant and did not need on-site administration, since the dealerships did not have appropriate technical resources.

The PRO application uses MySQL to store product configuration and prices, automatically update product catalogs, provide a secure login for sales representatives and managers, enable sales managers to track the sales staff performance, maintain a customer relations database, track popularity of models and accessories. The system also enables customization of product and price information to suit the needs of the local market. Matrix Consultants maintain a database of 85 000 records and use the rsync and MySQL Administration Utilities to distribute changes from the master database to dealerships. The updates of 5-20 MB are available to dealers 24 hours a day.

The company

Yahoo! is a leading global internet brand developed and made available in 1994 by two Stanford University graduates. It provides online properties and services, as well as a range of tools and marketing solutions for businesses. Yahoo! services are mainly free to users, as revenue is generated by providing marketing services and collecting fees for premium services. Yahoo! Finance provides the users with financial resources ranging from company information to personal finance management tools. Moreover, it offers fee-based services including real-time stock quotes package, company conference call transcripts and analyst research reports.

The open-source solution

MySQL powers many web properties of Yahoo!, Yahoo! Finance being the first one in the year 2000. Jeremy Zawodny of Yahoo! states that the company runs many services on MySQL because it is cost-effective, easy-to-use and reliable. Moreover, the open-source database has proven many times that it is capable of the most demanding high-traffic applications. Nowadays, MySQL is used by more than 200 Yahoo! applications.

Before implementing MySQL, Yahoo! Finance managed its data through flat files and Berkeley DB databases. However, those solutions were inflexible and not scalable enough. MySQL was the only one able to handle high-volume record tables as well as the small ones. *One of our servers handled over a quarter of a billion queries in a month and-a-half, and it still has capacity to spare* said Jeremy Zawodny.

After the success of MySQL in Yahoo! Finance, it was applied in Jointly Administered Knowledge Environment (JAKE) – a publishing system handling language translations and local media issues. The platform was developed initially to be used in Europe, yet, it was adopted very widely.

According to J. Zawodny, *over the past few years open source has become the technology we* [Yahoo!] *consider when there's something we need.*[123]

[123] *Open Source Goes Corporate* [Online] Available at: http://www.informationweek.com/software/showArticle.jhtml?articleID=171200352&pgno=1&queryText= [Accessed on 23.10.2006]

3.6 Research summary

No matter the reason for implementing open-source solutions, all the 10 companies' future plans were taking into account further introduction of open-source software. The factors named by all companies as the most important when implementing software were: professional support, drop of total cost of ownership, scalability, reliability, quick deployment and performance increase. All of them were present in the applied open solutions.

The immediate conclusion from the research conducted on companies migrating to Red Hat Enterprise Linux is that a migration is easier for a company with Linux- or Unix-based culture, as training needs are in those cases minimized. For Lapeyre the most important became customization of the Red Hat Enterprise Linux, compatibility and long-term support. LVM appreciated extra services such as Open Source Assurance Program that allows to use Red Hat Linux even if patent disputes occur. Renault was aiming at full-time availability of on-line information while keeping the costs low. Total cost of ownership was important also for Skanska which decided that upgrading of proprietary solutions is too costly for the company. After evaluating open-source alternatives, Statoil has chosen Red Hat because of its complete documentation. When all the reasons for choosing Red Hat Linux are pulled together, a picture of professional open-source software emerges with the emphasis put on vendor independence accompanied by the highest possible quality.

It appears that the five companies that migrated to MySQL were choosing the solution because of its ease of deployment and high capacity. Moreover, MySQL database enabled moving from expensive hardware to commodity PCs, as in the case of Lycos. neckermann.de was able to save money that would be spend on upgrading the old system, new hardware and licenses. Synchronous replication made the on-line service available all the time and the reduced wait time improved the customer experience. Sony appreciated the small amount of time and work that needed to be devoted to creation of the data management system. For Suzuki it was most important to have a database with zero on-site administration required. Yahoo! stated clearly that MySQL is capable of high-traffic and most demanding applications. No wonder that the biggest enterprises that need reliable and powerful databases choose an open-source solution – MySQL.

Of course, it is impossible not to have some technical problems while migrating to new software or implementing one. However, with the source code given, customizations that is possible and excellent service, no company is left alone to handle the malfunction.

To sum up, the research proved that open-source software is an alternative to proprietary software that should be considered before choosing more costly solutions. Observing the analyzed companies one may notice that open solutions may not only be cheaper, but also more efficient when it comes to technicalities. One should avoid at all cost the reasoning that open solutions provide only software without needed services. Time and time again the big open-source software vendors excel in making the open experience as pleasant as possible, not only for IT professionals.

Conclusions

Although some may say that open-source software does not fit into the picture of the capitalistic 21st century, the movement is gaining recognition and momentum. Long were we to wait for an alternative to proprietary solutions that could be considered by enterprises while software purchasing. More and more companies are tempted by lower costs, freedom and transparency. However, proprietary solutions are still ahead of the products stemming from the "community over selfishness" philosophy.

It is estimated that proprietary solutions for long will dominate the market, as they are required by already existing documents. One should not delude oneself with the vision of open solutions replacing the proprietary ones. Companies ought to aim at finding the right software mix that will enable profit maximization, as costs and benefits of implementing open-source are not the same for various enterprises. It is argued that universal answers do not exist.

Open-source software has definitely much to offer to business. Its biggest advantages are in the cost area that is strictly connected with profit accumulation. The total cost of ownership that may roughly be divided into the price of purchase, licensing fees, hardware requirements and overall maintenance, is proved to be much lower than the cost of ownership of proprietary solutions. Moreover, open-source software ensures security, scalability and reliability. It enables total customization according to the company's needs and constantly proves that its deployment is no longer that challenging. The legal disputes concerning open software may be summarized as a desperate attempt to stop a prospering competitor through courts. Concluding, as it was stated in the section concerning legal aspects of open software, there is a much larger risk of being sued by a proprietary vendor than over intellectual property issues connected to open-source.

Of course, open-source software is not an idyllic solution free of drawbacks. One of the biggest obstacles to the adoption of open software by enterprises is the switching cost that may outweigh potential savings. That is why companies are encouraged not to implement open solutions as a replacement to proprietary software. What is more, the human factor and learning curves should be taken into account, as well as software and hardware compatibility.

The empirical part of the thesis revealed that all 10 analyzed companies implementing open solutions coming from a professional vendor appreciated most comprehensive support, decrease of total cost of ownership, scalability, reliability, quick deployment and performance increase. Not surprisingly, the companies that previously used Linux or Unix were in a better position while deploying Red Hat Enterprise Linux and could enjoy lower costs, as well as vendor independence not exposed to significant performance drops.

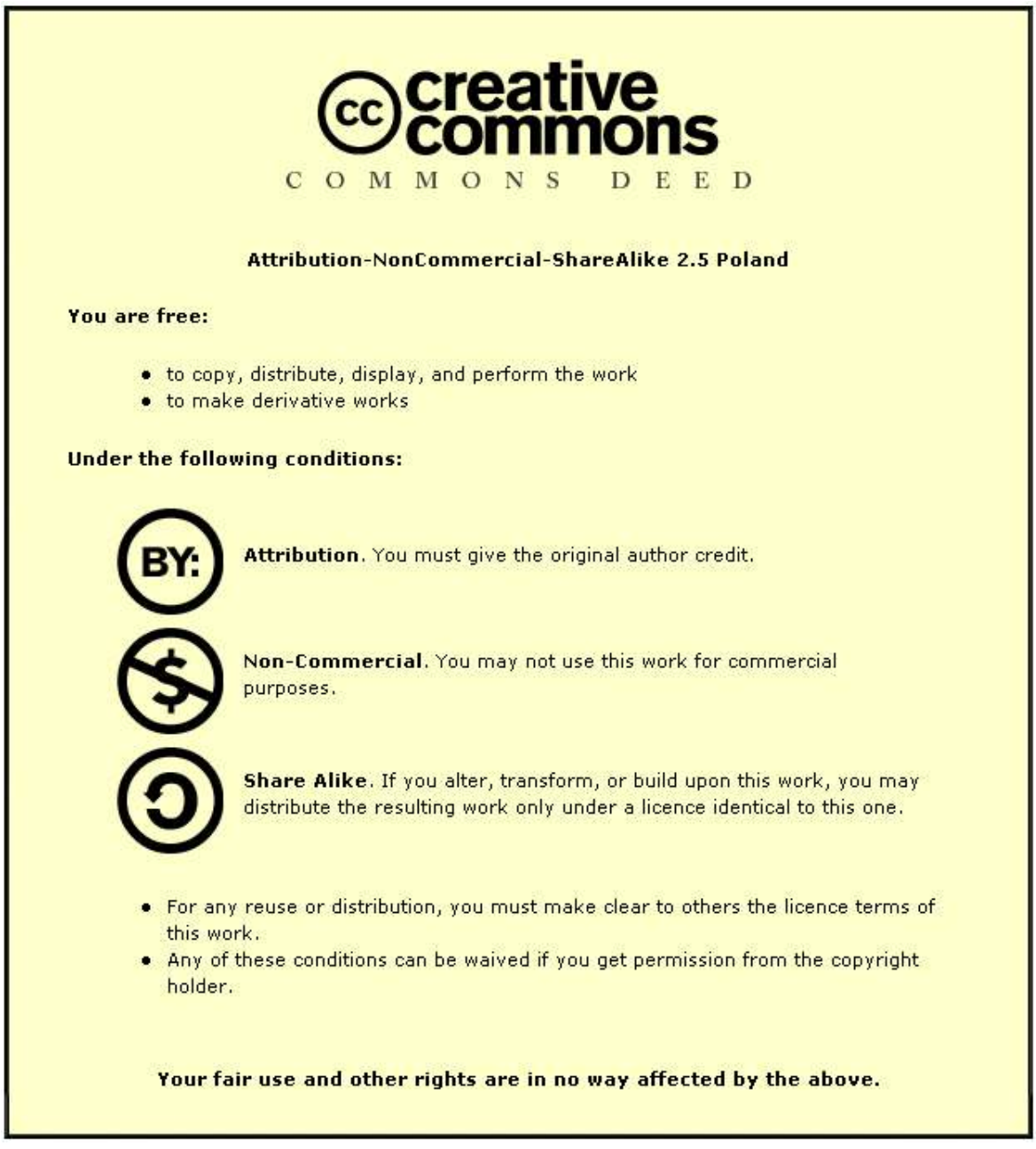
Another important conclusion stemming from the conducted study was that the open-source database – MySQL – was chosen because of higher capacity than the proprietary solutions. Thanks to synchronous replication it enabled the services to be full-time available, which was what the companies were aspiring to.

This work was to state whether open-source software is mature enough to be implemented in enterprises. Even though the empirical chapter provided the reader with examples of successful open software implementation, the author believes in adopting a soft approach to open-source. Additionally, as it was emphasized, enterprises need professional service to feel confident while trying out new technologies. That is why it is stated that one should choose professional open-source vendors whenever the budget allows one to.

Bearing in mind that no two companies are identical, implementing open software should be an individual decision of every IT manager. Still, the new personality smelling of ocean is becoming more and more tempting as the affection towards proprietary solutions very often turns out to be too costly and incapacitating.

Attachment 1:

Creative Commons License*



creative commons

COMMONS DEED

Attribution-NonCommercial-ShareAlike 2.5 Poland

You are free:

- to copy, distribute, display, and perform the work
- to make derivative works

Under the following conditions:

BY: **Attribution**. You must give the original author credit.

Non-Commercial. You may not use this work for commercial purposes.

Share Alike. If you alter, transform, or build upon this work, you may distribute the resulting work only under a licence identical to this one.

- For any reuse or distribution, you must make clear to others the licence terms of this work.
- Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

* Sourced from *Creative Commons Deed* [Online] Available at: http://creativecommons.org/licenses/by-nc-sa/2.5/pl/ [Accessed on 25.10.2006]

Attachment 2:

GNU General public License*

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the

* Sourced from *GNU General Public License* [Online] Available at: http://www.gnu.org/copyleft/gpl.html [Accessed on 25.10.2006]

original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

> **a)** You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
> **b)** 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.
> **c)** If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the

user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

> **a)** Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
> **b)** Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
> **c)** Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

(…)

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

Attachment 3:

Possible scenarios for the future of open-source software*

The working group on Libre Software created by the Information Society Directorate General of the European Commission describes the possible scenarios of how the support of governments and administration could influence the future of the open source movement. The authors assume three possible scenarios:

1. No action is taken by governments,
2. Limited support is granted by governments,
3. Aggressive support is granted by governments.

In the case of the *No action* scenario one can expect that mainly private companies and individuals will try to implement OSS. The movement has already proved that it is able to be self-sustaining in the economic and technical field and compete with the software development leaders. Probably open source software will be used as a standard part of the infrastructure by most governments. It will be preferred on economic and technical grounds, as a lower cost or technically better alternative to proprietary systems.

However, the threats for the OSS movement will be as follows:

1. FUD (fear, uncertainty, doubt) techniques, used by companies producing proprietary software,
2. Dissolution due to systems and licenses, causing divisions in the community and in the code base, as well as the loss of some of the advantages of the open source model,
3. Ignorance or the loss of global vision by the open source community,
4. Legal impediments disabling the progress of the open source movement.

In the case of *Limited support* scenario, governments will invest a limited amount of time and resources to assess the feasibility of open source software and to identify the barriers to adoption. When some strategic open-source projects will be identified, contribution will begin as well as the understanding of benefits in terms of flexibility, usefulness and adaptability. It is believed that this will result in greater acceptance of open-source software in society because of the effect that governments have on society. If the

* Attachment according to *Free Software / Open Source: Information Society Opportunities for Europe?* [Online] Available at: http://eu.conecta.it/paper/ [Accessed on 25.10.2006]

governments recognize the benefits of the open source model, they will probably also help to overcome the future problems.

Taking into account the *Aggressive support* scenario, the governments might give preference to open-source solutions whenever they are technically feasible or fund the development of open-source alternatives to proprietary systems. Such actions would create a market for open-source consulting and solutions, as well as have impact on the import/export balance for information technology products.

List of graphs

List of illustrations

List of tables

Bibliography

1) *2005 Annual Report* [Online] Available at: http://www.shareholder.com/shared/dynamicdoc/YHOO/1183/YHOO.pdf [Accessed on 21.10.2006]

2) *About MySQL AB* [Online] Available at: http://www.mysql.com/company/ [Accessed on 21.10.2006]

3) *Basic idea behind open-source* [Online] Available at: http://opensource.org/ [Accessed on 21.08.2006]

4) Bozman J. and others *Windows 2000 Versus Linux in Enterprise Computing* [Online] Available at: http://www.microsoft.com/windows2000/docs/TCO.pdf [Accessed on 17.09.2006]

5) *Brief Facts* [Online] Available at: http://www.skanska.com/skanska/templates/page____270.aspx [Accessed on 21.10.2006]

6) *Brief history of open-source* [Online] Available at: http://www.netc.org/openoptions/background/history.html [Accessed on 22.08.2006]

7) *Browser security test* [Online] Available at: http://bcheck.scanit.be/bcheck/page.php?name=STATS2004 [Accessed on 17.09.2006]

8) *Bug tracking guidelines* [Online] Available at: http://bug-tracking-guidelines.com/ [Accessed on 22.08.2006]

9) *Business Ready Open Source Solutions* [Online] Available at: http://www.spikesource.com/ [Accessed on 25.10.2006]

10) *CAMBIA* [Online] Available at: http://en.wikipedia.org/wiki/CAMBIA [Accessed on 21.08.2006]

11) *Cameleon Lapeyre* [Online] Available at: http://www.access-commerce.com/automne_modules_files/standard/public/p12_2795f03268b3d5a054dcbc40ded2719cLAPEYRE_EN.pdf [Accessed on 21.10.2006]

12) Cane A. Migration costs are the hitch., *Financial Times*. September 21, 2005

13) *Case Studies* [Online] Available at: http://www.mysql.com/why-mysql/case-studies/ [Accessed on 21.10.2006]

14) *Cathedral and the Bazaar* [Online] Available at: http://en.wikipedia.org/wiki/Cathedral_and_the_Bazaar [Accessed on 22.08.2006]

15) *Concurrent Versions System* [Online] Available at: http://en.wikipedia.org/wiki/Concurrent_Versions_System [Accessed on 21.08.2006]

16) *Copyright* [Online] Available at: http://en.wikipedia.org/wiki/Copyright [Accessed on 17.09.2006]

17) *Corporate Fact Sheet for Red Hat* [Online] Available at: http://www.redhat.com/about/companyprofile/facts/ [Accessed on 21.10.2006]

18) Costello S. *RSA: Security in 2002 worse than 2001, exec says* [Online] Available at: http://archives.cnn.com/2002/TECH/internet/02/25/2002.security.idg/index.html [Accessed on 17.09.2006]

19) *Creative Commons Deed* [Online] Available at: http://creativecommons.org/licenses/by-nc-sa/2.5/pl/ [Accessed on 25.10.2006]

20) *Debian Free Software Guidelines* [Online] Available at: http://en.wikipedia.org/wiki/Debian_Free_Software_Guidelines [Accessed on 26.10.2006]

21) *Democratizing innovation* [Online] Available at: http://en.wikipedia.org/wiki/Democratizing_Innovation [Accessed on 21.08.2006]

22) *Deployment and maintenance* [Online] Available at: http://www.netc.org/openoptions/pros_cons/deployment.html [Accessed on 17.09.2006]

23) *Drugs for Neglected Diseases Initiative* [Online] Available at: http://www.dndi.org/ [Accessed on 26.10.2006]

24) Dyck T. *Server Databases Clash* [Online] Available at: http://www.eweek.com/article2/0,3959,293,00.asp [Accessed on 17.09.2006]

25) Evans P., Wolf B. Collaboration Rules. *Harvard Business Review*. July 2005

26) *Executive guides: open-source software* [Online] Available at: http://guide.darwinmag.com/technology/program/open/index.html [Accessed on 21.08.2006]

27) *Features and quality* [Online] Available at: http://www.netc.org/openoptions/pros_cons/features.html [Accessed on 17.09.2006]

28) *Finding Cures for Tropical Diseases: Is Open Source an Answer?* [Online] Available at: http://medicine.plosjournals.org/perlserv/?request=get-document&doi=10.1371/journal.pmed.0010056 [Accessed on 26.10.2006]

29) *Free Software / Open Source: Information Society Opportunities for Europe?* [Online] Available at: http://eu.conecta.it/paper/ [Accessed on 25.10.2006]

30) *Free software* [Online] Available at: http://en.wikipedia.org/wiki/Free_software [Accessed on 21.08.2006]

31) *Free Software Foundation* [Online] Available at: http://www.fsf.org/ [Accessed on 21.08.2006]

32) *Frost J. J. Some Economic & Legal Aspects of Open Source Software* [Online] Available at: http://opensource.mit.edu/papers/frost.pdf [Accessed on 17.09.2006]

33) *General Public License* [Online] Available at: http://en.wikipedia.org/wiki/GNU_General_Public_License [Accessed on 21.08.2006]

34) Gill L. *Who's making money from open-source?* [Online] Available at: http://www.newsfactor.com/perl/story/17137.html [Accessed on 21.08.2006]

35) *GNATS* [Online] Available at: http://www.gnu.org/software/gnats/ [Accessed on 21.08.2006]

36) *GNU General Public License* [Online] Available at: http://www.gnu.org/copyleft/gpl.html [Accessed on 25.10.2006]

37) Golden B. *The ROI of open-source* [Online] Available at: http://www.computerworld.com/developmenttopics/development/webservices/story/0,10801,102638,00.html [Accessed on 17.09.2006]

38) Goldman R. Gabriel R. P. *Innovation Happens Elsewhere* [Online] Available at: http://www.dreamsongs.com/IHE/IHE-6.html [Accessed on 26.10.2006]

39) Greene T. C. *Ballmer 'fesses up to Linux/Windows cost FUD* [Online] Available at: http://www.theregister.co.uk/2002/07/16/ballmer_fesses_up_to_linux/ [Accessed on 17.09.2006]

40) Hecker F. *Setting up shop: the business of open-source software* [Online] Available at: http://www.hecker.org/writings/setting-up-shop [Accessed on 21.08.2006]

41) *Institute for OneWorld Health* [Online] Available at: http://www.iowh.org/ [Accessed on 26.10.2006]

42) Lacy S. Open season on open-source? *Business Week*. New York. Mar 13, 2006

43) Lemos R. *Security research suggests Linux has fewer flaws* [Online] Available at: http://news.com.com/Security+research+suggests+Linux+has+fewer+flaws/2100-1002_3-5489804.html [Accessed on 17.09.2006]

44) *Linux* [Online] Available at: http://en.wikipedia.org/wiki/Linux [Accessed on 21.08.2006]

45) *Linux on the Desktop: The Whole Story* [Online] Available at: http://www.gartner.com/DisplayDocument?id=406459 [Accessed on 17.09.2006]

46) Lyons D. *Linux Rules Supercomputers* [Online] Available at: http://www.forbes.com/home/enterprisetech/2005/03/15/cz_dl_0315linux.html [Accessed on 17.09.2006]

47) Metcalfe R. *Open-source: an Introduction* [Online] Available at: http://www.oss-watch.ac.uk/talks/2006-04-28-rsc-eastern/2006-04-28-rsc-eastern.pdf [Accessed on 21.08.2006]

48) *Microsoft says MS Licenses prohibit company sale* [Online] Available at: http://librenix.com/?inode=2402 [Accessed on 17.09.2006]

49) Miller R. *Does Open-Source Deserve a Place in Your IT Portfolio?* [Online] Available at:http://trends.newsforge.com/article.pl?sid=04/10/19/1518218&tid=139&tid=150&tid=111&tid=37&tid=132&tid=18&tid=29 [Accessed on 26.10.2006]

50) Miller R. *IT analysts' influence on open source adoption* [Online] Available at: http://trends.newsforge.com/trends/04/11/03/181215.shtml [Accessed on 17.09.2006]

51) *MySQL Awards* [Online] Available at: http://www.mysql.com/why-mysql/awards/ [Accessed on 21.10.06]

52) *MySQL Products* [Online] Available at: http://www.mysql.com/products/ [Accessed on 21.10.2006]

53) *MySQL Services* [Online] Available at: http://www.mysql.com/services/ [Accessed on 21.10.2006]

54) Nairn G. Promise of cost-savings driver take-up of Linux. *Financial Times*. September 21, 2005

55) Naraine R. *US-CERT: Beware of IE* [Online] Available at: http://www.internetnews.com/security/article.php/3374931 [Accessed on 17.09.2006]

56) Newitz A. *A User's Guide to EULAs* [Online] Available at: http://www.eff.org/wp/eula.php [Accessed on 17.09.2006]

57) *No more mysteries: Apple's G5 versus x86, Mac OS X versus Linux* [Online] Available at: http://www.anandtech.com/mac/showdoc.aspx?i=2436 [Accessed on 17.09.2006]

58) *Novell promuje open-source dla firm* [Online] Available at: http://www.egospodarka.pl/12361,Novell-promuje-open-source-dla-firm,1,14,1.html [Accessed on 22.08.2006]

59) *Ohloh Home Page* [Online] Available at: http://www.ohloh.net/ [Accessed on 25.10.2006]

60) *Open Source Goes Corporate* [Online] Available at: http://www.informationweek.com/software/showArticle.jhtml?articleID=171200352&pgno=1&queryText= [Accessed on 23.10.2006]

61) *Open-source* [Online] Available at: http://cci.mccombs.utexas.edu/research/white/open-source.htm [Accessed on 29.07.2006]

62) *Open-source* [Online] Available at: http://en.wikipedia.org/wiki/Open_source [Accessed on 21.08.2006]

63) *Open-source definition* [Online] Available at: http://en.wikipedia.org/wiki/Open_Source_Definition [Accessed on 21.08.2006]

64) *Open-Source Legal Risk Management in the Enterprise* [Online] Available at: http://stephesblog.blogs.com/papers/Optaros_FOSS_Risk_Mgmt_SWalli_051906.pdf [Accessed on 17.09.2006]

65) *Open-source model* [Online] Available at: http://en.wikipedia.org/wiki/Open_source_software#Open_source_model [Accessed on 21.08.2006]

66) *Open-source movement* [Online] Available at: http://en.wikipedia.org/wiki/Open_source_movement [Accessed on 21.08.2006]

67) *Open-source politics* [Online] Available at: http://en.wikipedia.org/wiki/Open_source_politics [Accessed on 21.08.2006]

68) *Open-source software* [Online] Available at: http://en.wikipedia.org/wiki/Open_source_software [Accessed on 21.08.2006]

69) *OpenSPARC* [Online] Available at: http://opensparc.sunsource.net/nonav/index.html [Accessed on 21.08.2006]

70) *Oprogramowanie open-source stosuje 94 proc. polskich firm* [Online] Available at: http://wirtualnemedia.pl/document,,977690,Oprogramowanie_Open_Source_stosuje_94_proc._polskich_firm.html [Accessed on 21.08.2006]

71) Orzech D. *Linux TCO: Less Than Half The Cost of Windows* [Online] Available at: http://www.cioupdate.com/article.php/10493_1477911 [Accessed on 17.09.2006]

72) *Participants in OSS development projects* [Online] Available at: http://en.wikipedia.org/wiki/Open-source_software [Accessed on 21.08.2006]

73) *Patent* [Online] Available at: http://en.wikipedia.org/wiki/Patent [Accessed on 17.09.2006]

74) Pfaffenberger B. *A Senior Microsoft Attorney Looks at Open-Source Licensing* [Online] Available at: http://www.linuxjournal.com/article/5073 [Accessed on 17.09.2006]

75) *Pillsbury Winthrop Attorney Clarifies Latest Open Source Myths Following Government Guidance and Microsoft Posturing* [Online] Available at: http://www.prnewswire.com/cgi-bin/stories.pl?ACCT=104&STORY=/www/story/12-15-2004/0002633825 [Accessed on 17.09.2006]

76) *Presserelease, 7th February 2000* [Online] Available at: http://web.archive.org/web/20011011215009/http://www.syscontrol.ch/e/news/Serversoftware.html [Accessed on 17.09.2006]

77) *Proprietary software* [Online] Available at: http://en.wikipedia.org/wiki/Proprietary_software [Accessed on 21.08.2006]

78) Raymond E.S. The Cathedral & the Bazaar. *O'Reilly*. October 1999

79) *Red Hat Enterprise Linux Details* [Online] Available at: http://www.redhat.com/rhel/details/ [Accessed on 21.10.2006]

80) *Renault in brief* [Online] Available at: http://www.renault.com/renault_com/en/main/10_GROUPE_RENAULT/10_Renault_en_bref/ [Accessed on 26.10.2006]

81) Robert Frances Group *Total Cost of Ownership for Linux Web Servers in the Enterprise* [Online] Available at: http://www.rfgonline.com/subsforum/LinuxTCO.pdf [Accessed on 17.09.2006]

82) Rothman J. B. and Buckman J. *Which OS is Fastest for High-Performance Network Applications?* [Online] Available at: http://www.samag.com/documents/s=1148/sam0107a/0107a.htm [Accessed on 17.09.2006]

83) *SCO v. IBM* [Online] Available at: http://en.wikipedia.org/wiki/SCO_v._IBM_Linux_lawsuit [Accessed on 17.09.2006]

84) Seigo A. *How open-source software improves society* TPOSSCON 2006

85) Seigo A. *How to get involved* TPOSSCON 2006

86) Special Report: Open, but not as usual - Open-source business, *The Economist*. London. Mar 18, 2006

87) *Statoil in brief* [Online] Available at: http://www.statoil.com/STATOILCOM/SVG00990.nsf?opendatabase&lang=en&artid=3FED33ECC77666314125665D004E05E3 [Accessed on 21.10.2006]

88) *Success Stories* [Online] Available at: http://www.redhat.com/rhel/informationcenter/successstories/ [Accessed on 21.10.2006]

89) *SUSE Linux Enterprise Server 10* [Online] Available at: http://www.novell.com/products/server/sysreqs.html [Accessed on 08.10.2006]

90) *Suzuki* [Online] Available at: http://en.wikipedia.org/wiki/Suzuki [Accessed on 25.10.2006]

91) *System Requirements for Windows Small Business Server 2003 R2* [Online] Available at: http://www.microsoft.com/Windowsserver2003/sbs/evaluation/sysreqs/default.mspx [Accessed on 08.10.2006]

92) *The open source definition* [Online] Available at: http://www.opensource.org/docs/definition.php [Accessed on 25.10.2006]

93) *The Synaptic Leap* [Online] Available at: http://www.thesynapticleap.org/ [Accessed on 21.08.2006]

94) *Tinderbox* [Online] Available at: http://en.wikipedia.org/wiki/Tinderbox_%28software%29 [Accessed on 21.08.2006]

95) *Trade secret* [Online] Available at: http://en.wikipedia.org/wiki/Trade_secret [Accessed on 17.09.2006]

96) *Trademark* [Online] Available at: http://en.wikipedia.org/wiki/Trademark [Accessed on 17.09.2006]

97) *Tropical Disease Initiative* [Online] Available at: http://www.tropicaldisease.org/ [Accessed on 21.08.2006]

98) *Understanding free and open-source licenses* [Online] Available at: http://stephesblog.blogs.com/papers/Optaros_Und_FOSS_Lic_SWalli_PartII_051906.pdf [Accessed on 21.08.2006]

99) *Vendor lock-in* [Online] Available at: http://en.wikipedia.org/wiki/Vendor_lock-in [Accessed on 17.09.2006]

100) Vermeir D. *Open-source: an overview* [Online] Available at: http://tinf2.vub.ac.be/~dvermeir/software/oss-overview-slides.pdf [Accessed on 21.08.2006]

101) *What is CAMBIA?* [Online] Available at: http://www.cambia.org/ [Accessed on 26.10.2206]

102) Wheeler D. A. *Is open-source good for security?* [Online] Available at: http://www.dwheeler.com/secure-programs/Secure-Programs-HOWTO/open-source-security.html [Accessed on 17.09.2006]

103) Wheeler D. A. *Why open-source software? Look at the numbers!* [Online] Available at: http://www.dwheeler.com/oss_fs_why.html [Accessed on 17.09.2006]

104) *Who contributes to open source software? And why do they give it away?* [Online] Available at: http://www.oss-watch.ac.uk/resources/whowhy.xml [Accessed on 21.08.2006]

105) *Wzrosną przychody z open-source* [Online] Available at: http://wirtualnemedia.pl/document,,1275149,Wzrosna_przychody_z_Open_Source.html [Accessed on 21.08.2006]


SOME RIGHTS RESERVED


BY: