ANALYSIS OF THE BENEFIT FROM COSTLESS AND COMMERCIAL SOFTWARE THROUGH COMPARISON BETWEEN OPEN SOURCE SOFTWARE AND CLOSED SOURCE SOFTWARE

Goran Stanoev

University American College Skopje

goranstanoev@gmail.com

Adrijan Bozhinovski

University American College Skopje

bozinovski@uacs.edu.mk

Eva Blazhevska

University American College Skopje

blazevska@uacs.edu.mk

Abstract

This paper examines the differences between free and commercial software, from the viewpoint of the final profit, through comparison of both types of software. The analysis has been performed through comparison of the differences in the software types, as well as the viewpoint of the feasibility of the technologies, regardless of whether they're free or not. Conceptual foundations of the open source and closed source software are given, as well as examples for such software and an overview of certain differences through a comparison of operating systems, and also an overview of the profit of companies that produce free software and companies that produce non-free software. The approach of choosing software based on personal experience is given. According to the arguments presented, as well as personal experiences, a conclusion is reached about which approach should be taken when starting with software creation, from the viewpoint of available resources, the technology that would be used, and the ultimate goal of the company that is producing the software.

Keywords: open source software, closed source software, free software, commercial software

I. Introduction

Over the past few years, there has been an increasing interest for a new paradigm for software development - the open source software development. The term refers to a process of software development in which the source code is public and available for many developers that use open standards, share their knowledge and experience, and collaborate with each other and with the worldwide users of the built software, in order to build more powerful and client-oriented software, identify and correct errors, and make software enhancements [1]. Unlike the traditional proprietary paradigm of software development, users have free access to the source code, which they can modify to correct software bugs. By the open source philosophy, software is a public good and should be shared for free, used, and developed by everyone. On the other hand, the source code and development standards for proprietary software are private property, kept in secret and protected with patents. Proprietary software belongs to some company that sells licenses to the end-users for profit.

Software exists in two forms. It is typically sold in binary form and hence is "closed source" (the source is kept secret). Binary code prevents humans from looking at or understanding how the software is done and works. Binary code makes it impossible to modify the software or to use its parts in other software products. Software is also sold with a copyright clause that says you may not copy or give away or sell the (binary) software for free or without company accordance. Open Source software is the opposite— it is available in original source code (format), and, once you have it, you are free to make copies, give it away, make modifications and use parts of in other open-source programs or products [2].

This paper elaborates the usage of free and binary (proprietary) software for commercial purposes. Through the comparison of concepts, as well as the software itself and companies that

produce it, the paper draws a conclusion for the cost-effectiveness of the usage of each of these types of software.

II. Concepts

Even though there are several approaches for free and proprietary software, there are some key concepts that characterize and distinguish these two groups. The development of information technology is constantly changing, but the approach and principles of creating software, whether free or not, usually is permanent.

As suggested by the examples listed in Table 1, the process of creating an open source and proprietary source software differs in several segments. The approaches like method for creating software, cooperation between users and developers and the time of producing are quite different.

Advantages and disadvantages of the concepts of free software:

- Developers are in a constant "race" to express their abilities and knowledge. In order to be recognized, the programmer enforces himself/herself for better acceptance of his/her solution. With this approach, the programmer is constantly improving his/her skills and knowledge, but the time to achieve the benefit (profit) is longer and without warranty that it will ever be achieved.
- The popularity of the project attracts a large number of developers working on it. This contributes a major expansion of the project, but increases the risk of various unstable and unstandardized code segments because of the diversity of the approaches of developers.
- End users use the software for free, but user support is minimal or absent altogether and at the same time the user is not protected from any consequences that could arise from inappropriate behavior of the program.

Advantages and disadvantages of the concepts of proprietary software:

- Every programmer has a guaranteed benefit (profit) from the company that employed him/her and has certain responsibilities. This approach allows the programmer to perform his/her job duties calmly, but may limit his/her progress and improvement.

Table 1: Conceptual bases for open source and proprietary software [3]

Open source	Proprietary source
Non-paid programmers provide software improvements with intention of personal recognition among the community if their improvement is accepted	Programmers are hired by a software company and work for compensation (salary)
If the program is popular, a very large number of programmers may work on it	The number of programmers working on the code is limited by the company (or department) resources
A coordinating committee or an open forum for experience exchange selects the best software improvements and gives permission for its incorporation in the next software release	Coordination of software improvements is imposed by the company team to reduce overlap of efforts
Programmers compete with each other for personal satisfaction and recognition among the community members (software development is treated as tournament)	Programmers do not compete with each other
Users can use the developed software for free, i.e. nothing is charged for its usage; particularly attractive to users	Users pay a price for software usage (usually license). The price is set so it can return the company investment in software research and development
Less time and effort is spent by an individual user/programmer on research and development, but there are more programmers	More time and effort are spent on research and development by the software company, but there are fewer programmers

- The number of developers depends on the company's resources. With the company's increased workload, more programmers may be employed. Planning of resources is very important in order to avoid unnecessary costs and thus increase the final price of the product or reducing the number of employees.
- End users pay for software that, besides guaranteed performance during use, has an appropriate warranty, technical support for a period of time and are legally protected from consequences that can arise from inappropriate

behavior of the program for the duration of technical support.

The main concept for realization of the target benefit from open source software is creating a free platform through which companies would sell their or other services. The Android operating system is open software built on this concept. The company that develops this free operating system allows its free usage on mobile devices and installation of commercial products on it. Google Play is another example of open source service for commercial mobile applications trading and marketing [4].

III. Software

Operating systems that belong to the group of free software have different approaches in terms of resource planning and achieving ultimate profit. Examples are Ubuntu Linux and Red Hat Linux. Canonical, the company that develops Ubuntu, offers two versions of the operating system (free and commercial). Both versions are identical, but the commercial version also includes technical support. Red Hat offers two Linux OS versions: Fedora, which is free, and Red Hat, commercial, but these two products are different. Fedora is not sufficiently tested and is not certified, unlike Red Hat that you have to pay for in order to use it [5]. The Linux operating systems are usually specialized and therefore they have smaller financial profit, compared to systems with proprietary software.

On the other hand, operating systems that are parts of the proprietary software in general have similar development planning approaches in order to achieve the ultimate profit. Marketing and introduction of new technologies (patents) are the main means for achieving profit. Proprietary operating systems are developed for general purpose, so that the end user gets what the company team thinks is necessary to be included in the product, so it will provide full and adequate usage of the device the operating system is installed on.

When it comes to companies that use different types of software, overview of the profits of the 4 companies that are among the biggest and financially most powerful IT companies is shown in Figure 1. Two of them offer free software, and the other two offer proprietary software.

Figure 1 shows that the companies that offer proprietary software have higher profits per year.

Table 2: Examples of open source and proprietary software products with comparable usage

Open source software	**Proprietary software**
Operating systems	
Linux	Windows OS/2 HP-UX Solaris
Programming languages	
Perl Python PHP	ASP
Programming tools	
GNU Project	Visual C++ Cold Fusion
Web Servers	
Apache	Internet Information Services IIS
Web browsers	
Mozilla	Internet Explorer
Other	
Gimp Open Office	Adobe Photoshop Microsoft Office

This is a result mainly from the strong commercialization of their products and the fact that their products are very familiar to the general population.

IV. Companies

In general, companies that produce open source software have fewer employees than other companies. This is due to the fact that free software is developed by the community, and therefore fewer developers for implementation in the company's final product are required. Although statistically the number of employees is lower, the real number of developers is much higher, taking into account the fact that the software is developed globally.

There are several differences about what type of software should be used, depending on the needs, resources, customer demand and the company's ultimate goal. During the preparation of special purpose software for a particular customer, client requirements have great impact. Often the

client chooses the software type: open source or proprietary. According to that, the company defines the price and the number of employees who will work on the project.

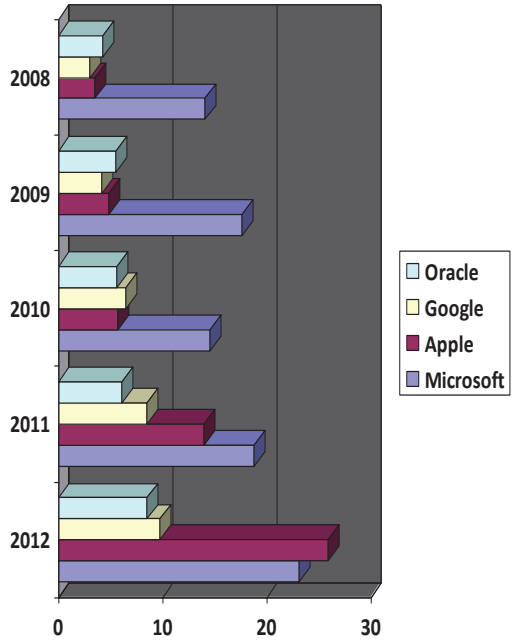

Figure 1: Profit comparison per years [6]

Software development of a product that will be published, as a company service for general purpose, requires detailed analysis about what assets are available, how many developers would work on the project, as well as its deadlines and constraints. The use of commercial software development platforms can significantly accelerate the development of software. This is due to the fact that the license fee for the use of certain software development platforms provides additional services in which the company (software development platforms producer) invested, so it achieves reliability and productivity for its customers. When the security and stability of the software is an important issue, commercial software is the better and the only choice, because it offers warranty and support for the offered services.

Sometimes the type of software is selected according to the favor of the developers to specific technologies. This approach can often be wrong because the technology chosen may not be the most appropriate technology for product development. The result of this wrong decision is possible time extension for the development of the product, or creation of a product that is hardly sustainable.

V. Conclusion

Before starting a new software development, detailed analysis of the project requirements should be made in all aspects. Adequate knowledge of the concepts of free and proprietary software may guarantee the right choice of technology. However, companies or individuals are not always able to implement the right technology for a particular project, due to available resources. A well planned development process can provide better software distribution. The use of proprietary software requires payment for appropriate user license, but productivity and safety are on a higher level, and thus provides a better potential profit from the final product.

Today in the IT world, the most popular products belong to the group of free software, but profitability is greater in proprietary products. This is the result of the powerful company marketing and awareness of product protection (patents, licenses etc.). Product popularity doesn't always come with its profitability.

References

[1] O. Reilly, 1999. Lessons from Open-Source Software Development, *Communications of the ACM* (42:4), April 1999, pp. 32-37.

[2] P. Dasgupta, 2002. School of computing, informatics, decision system engineering, *Open Source, Closed source*, http://cactus.eas.asu.edu/partha/columns/05-28-OpenSource.htm [last accessed on 14.2.2013].

[3] B. Mishra, A. Prasad and S. Raghunathan, 2002. Quality and Profits Under Open Source Versus Closed Source. *International Conference on Information Systems,* Orlando, Florida.

[4] Google Inc, 2012. *Transaction Fees, http*://support.google.com/googleplay/android-developer/answer/112622?hl=en&ref_topic=15867 [last accessed on 03.2.2013].

[5] S. Shankland 2006. C|Net - American tech media website, *Canonical seeks profit from free Ubuntu*, http://news.cnet.com/canonical-seeks-profit-from-free-ubuntu/2100-7344_3-6123249.html [last accessed on 16.2.2013].

[6] CNN Money, 2012. *Fortune 500*, http://money.cnn.com/magazines/fortune/fortune500/ [last accessed on 01.02.2013]