

Open Source Software Vs Proprietary Software

Sakshi Singh, Madhvik Verma, Naveen Kumar N

Abstract— Open Source software are the most revolutionizing concept that has been put forth in the Information Technology spectrum. This new release has raised quite a lot of brows, similar to when the Internet was launched. For a very long time, there has been a tiff going on between the Proprietary software and the open source software and how the open source software are taking over their ancestors. The technical experts often indulge in conversations as to which of the two software is better and why. People also get confused as to which type of software can be listed as Open source. In this paper we aim to emphasize that the open source development cycle has been there in the market for quite some time and has a huge user base now, thus giving it a strong hold in the software market. It has encouraged the use of new strategically placed actions which has in turn made a huge base for open source software systems. On the other hand, many experts also suggest that this is just momentary and it will pass over a period of time whereas the proprietary software are there to stay. This paper gives the understanding and recommendations for both the kind of software thus giving the user a proper understanding of which software is suitable for their use and thus they can make a better choice for their purpose.

Index Terms— Open Source Software, OSS, Free Software, Proprietary Software

1 INTRODUCTION

Just like the trend of desktop computers and our famous and beloved Internet, Open source software have been both praised and criticized since it first appeared in the markets due to its unexpected arrivals with proprietary software in opposition. There has been a lot of uproar against the Open Source Software(OSS) due its wide availability and the various new aspects it brings forth. Open Source Software(OSS) was developed as a workstation project because it's source code is procured within a license which allows the users to use it, modify/change it so as to improve the services and to reallocate it in a new framework or in the previous one only. It is constantly processed in an open, user-friendly and transparent environment. Open source software gives information, required skill set and the knowledge to use it to make it user-friendly which is its main ideology. The idea behind it is that when an individual is going through the source code certain mistakes and bugs are easily spotted by the person. Due to the large number of people using it, bugs are found at a very fast rate.

We begin this paper by plainly putting forth a demonstration of the brief yet interesting history of the open source software. We also show that the open source software is here for the long run and that, their heritage is important for people to know since people have been perceiving it negatively also. The new emerging technologies that come along with open source software open a plethora of opportunities for the developers and also give us, the consumers, the power to change and use the software as we like. In our paper, we aim to compare different aspects of Open source and proprietary software and also list down the ups and downs of using each kind of software.

It is a known fact that since the arrival of open source software, the sales of proprietary software has tremendously gone down. This is due to the wide availability and ease of use of the open source software. In a study conducted by Amazon, they reached a conclusion that their technological costing went down by 25 percent since they started to adopt open source software. In Munich, a city in Europe, people are actively

transitioning from proprietary to open source by going from Windows (Internet Explorer) to Linux (Mozilla Firefox).

In this war of survival of the fittest, both the software platforms have been worked on day in and day out to become even better and thus increasing their profits tremendously. There are various kind of strategies that the proprietary software is adopting in order to beat their opponents which only proves the growing effectiveness of open source software which has shaken an already established structure.

2 LITERATURE REVIEW

The most frequently cited question on the theory of open source software is to find out what tangible and intangible profit and motivation are the developers of open source software getting from working so hard on giving the world a free software .The possible set of reasons have been divided into two categories, first being the internal reasons like gaining knowledge and learning new things on the way ,and the external reasons being uncertainty in career and being recognized by the fellow members .

The second major concern that was proposed is how effective are open source software and the growing rivalry between the open source and proprietary platforms. Some researchers have reached a conclusion that open source software is equally efficient and equally good in quality as compared to the proprietary software. They have reached this conclusion by testing a particular software in both a closed and open environment and checking its behavior. Mukund stated in his works that the scope of improvement is way more in open source as compared to proprietary because its source code is being critically examined by each new user getting added to the community. Menell reached a conclusion that for open source software to stay top-notch in its quality they need to have a presentable structure and should be highly compatible with the upcoming technologies.

The completion between open source and proprietary has been a widely talked about topic. The fight can be visualized

by seeing the example of Linux vs Windows wherein we see a non-profit organization competes with a for-profit organization. Linux aims at maximizing the welfare and providing the software to the masses, whereas Windows aims only at maximizing its own profit. This rivalry can have two possible outcomes as predicted by the experts. The first one being that both of these software get a separate user base which ends their fight and they co-exist in the market or Linux gets kicked out of the market because of the strong-hold of Windows over the market.

3 SOFTWARE AND ITS TYPES

Software can be defined as the code that when executed controls and conducts operations of a system/machine to provide different functionalities depending on what purpose it solves. Software is a complete sphere consisting of basic to very extensive engineering techniques to improve its efficiency and to keep up with the upcoming technologies. It acts as a functional unit of the system which is responsible for supervising and managing the hardware of the system and to tell the other software that the particular hardware will be used for its own work. It also keeps a copy of the low-key system functions that it might need to call during its course of execution.

There are different kinds of software available in the market like:

A. Freeware software

These are the software that are available for free on the internet and these can be used by anyone irrespective of the region they are living in. Andrew Fluegelman came up with the term "Freeware" in 1982 to tell his fellow researchers about a project he worked on called the PC-Talk. Every person using a freeware software gets a duplicate of the copyright laws that are involved with the software to keep the sanctity of the developer intact. Freeware software do not accept altering programming and their developers are the ones solely responsible for how they work and for fixing the bugs.

B. Shareware software

This kind of software gives the consumer/user a slight sneak-peak into the real software. This trial version has many functions hidden. To access the whole software, the user needs to buy the software from a reliable source. It also consists a set of copyright laws which enable the makers to be the sole authority responsible for the software and all the rights are in the hands of usually one person. Some functionalities might be restricted to the administrator only whereas the others can be enjoyed by anyone who actually buys the software. This software doesn't have the ability to change and is often rendered useless after a period of time. Thus, the makers publish new versions regularly to keep it functional. The final call regarding any matter what-so-ever remain in the hands of the creator.

C. Open Source Software

This kind of software has the flexibility of being modified, passed on or shared and viewed by anyone who follows a particular licensing agreement that is set by the creator. This is so because its source code is public. This helps it to remain bug-free and increases the efficiency of the software.

The software package consists of:

- a. Free Rights to distribute
- b. Source Code free access
- c. Modification control to the user
- d. Distribute the new Modifications done
- e. Forbidding the right to modify to a person who doesn't follow the licensing agreement

D. Proprietary Software

For this software a set of advantages are entrusted only in the hands of the person or organization that holds the copyright to it. It can be made by an individual or an organization who can either use a pre-existing framework or build the software from scratch to match their ideologies. They exercise certain advantages and privileges over the software. Unlike open source, proprietary software keeps their source code a hidden mystery. They also do to not make the structure of their program a topic of public knowledge. There are a set of limitations that are put on the software based on the EULA (End-User license agreement) which the user has to accept before getting into using the software. This is done to avoid copyright infringement by any individual over the software.

E. Public Domain Software

The owner or the developer of the software completely gives it to the public domain without any copyright or any limit over the accessing of features in the software.

F. Trial Ware software

It is a time based software meaning that the user can access the functionalities only within the restricted time period. The user is given a list of the programs or processes which will get blocked after a certain time period or a certain number of times. After paying the owner organization, the user gets unlimited and unrestricted access to all the functions of the software and also gets new updates (if any).

demonstrated in this document, the numbering for sections upper case Arabic numerals, then upper case Arabic numerals, separated by periods. Initial paragraphs after the section title are not indented. Only the initial, introductory paragraph has a drop cap.

4 OPEN SOURCE AND PROPRIETARY SOFTWARE

Many people have been using proprietary software over the widely available open source software because they have a misconception that they are at risk while using a software which can be accessed by anyone. What they don't understand is the changes being made in the software pose no threat to their personal data. They often live under a misconception that since the proprietary software is paid, they are relatively easier to learn and understand. Whereas on the contrary, the truth is that Open source software is fairly simple with all its functions. Here we list out the differences between both Open source and proprietary software based on various aspects to reach a better conclusion as to which one suits which kind of individual/organization.

Table 1- Open Source Software Versus Proprietary Software

Factor	Open Source Software	Proprietary Software
Expense	<p>OSS is a free software which makes it the best choice for the in-house works. If a person is running a business or any work from home, the open source software can prove to be highly capacitive with least expenses on technological aspect. Certain skill set is required to work with an OSS. One must keep in mind that although the software is free, expenses can be incurred once the software is set up and starts working.</p> <p>Open source software is expected to be free but nowadays for specific add-ons the creators have started charging people which might end up being equal to other paid software thus deviating from its purpose. People should always benefit from the open source software and optimize the software in the process without worrying about any additional charges or subscription money.</p>	<p>The costing of proprietary software starts from some thousand to some hundred thousand Rupees (INR) depending on the complexity of the software and the purpose it is used for.</p> <p>This cost is an amalgamation of some base charges for the actual programming involved, administration rights and the permit charges (can be monthly or yearly). The consumer in turn gets an efficient and robust software which bountiful security measures to protect data from attackers, constant changes in the software through updates to make it better and free of glitches, packing costs and additional costs for certain special skills that the software possess.</p> <p>The final goal of such software is to back the components and maintain a good access to the product sites. The additional charges that the open source software are asking for , to use the add-ons is only giving the proprietary software an edge over OSS because the roundabout cost comes somewhat same only.</p>
Services and support provided	<p>The main problem with open source software is that of Administration. The bugs and glitches in the software are communicated via online blogs and discussion forums. These means might be super-fast at certain times, whereas very slow on the other. Consumers have to depend on certain unreliable sources to get help in case of any technical issue and the updates can't be trusted blindly since they might come from a completely unknown source. The software might become unresponsive if a lot of people are trying to cause changes which might decrease efficiency.</p>	<p>Good administration and regular updates are the highlighting features of any proprietary software. The users are provided with immediate and effective help in case of need which is 100% secure and safe. This saves both time and effort from the user side thus making him/her favor this software. The exclusive programming is the main part where the service and support part of proprietary software becomes strong. Backup is also provided to the clients to safeguard the data from being lost. Along with this, a certain demo is provided to make them familiar with the software and its functions.</p>

<p>Scope to Innovate</p>	<p>Open source software gives wings to the imaginative minds for them to innovate and get used to the software environment and also to help them make the software better suited for their own use.</p> <p>It also empowers the users to spread the word about the new innovations they bring about using online blogs or they can keep it to themselves and not sure. This choice is completely given to the user depending on their desire to share or not.</p> <p>Since the source code is being readily modified on each person's individual system, to bring about updates and to implement them is a major issue of concern. If the developer aims to fix any issue or bugs or if they want to enhance the software, the new changes might make certain functions unresponsive due to the different codes.</p>	<p>Proprietary software has their source codes hidden from the users. This in turn provides undivided security and amazing quality of services which is tough to compete with. Certain software have different modes of operation depending on the type of user. These different modes help in the optimization of the software for the needs of that particular person. R&D department looks into doing it and also provides certain offers.</p> <p>Unlike OSS, the online discussion groups of proprietary software collect grievances and complaints from the users and thus try to make changes to solve their issues. Certain suggestions made on such platforms are also put into consideration and launched in the following updates. Before launching any new updated, it is fully tested for bugs and are fixed before the public version is released and any client can access it.</p>
<p>Ease of use</p>	<p>Open source software are often looked down at when it comes to the ease of use. They do not usually come with any manual or help and the user has to figure it out on their own. The innovation put forth t]by the users online may not very well sync in with a new user thus making it tough for them to use.</p> <p>Open source software's entire structure is based on the designers and developers without a prefixed framework. They don't come in with the support to learn any additional language that might be used during running the software thus giving user an unpleasant experience. People who know how to solve small bugs are the ones who benefit the most but the newbies usually have to rely on others to fix their issues.</p> <p>Open source software is not very legally binding in some aspects and the user formatting can easily cause problems in the device. The documentation that comes along with the software is usually just a brief introduction of the software but doesn't tell the user anything about the additional learning required. Hence when the user has certain issue he/she has to turn to online help to solve the query. This is both time consuming and inconvenient. This</p>	<p>Proprietary software comes with personalized guide or instruction to go about the software. The experts working behind the scene evaluate and work on increasing the functionalities that the software has to offer. They also provide user group based services which can be used for the greater good. The ease of usability for proprietary software is quite high.</p> <p>The manuals and instruction leaflets come in handy when the user is stuck in a problem or wants to learn some new program. The manuals are completely safe to follow and doesn't hamper the system as well thus helping the user to learn quickly.</p> <p>The organizations behind this software also hold seminars and sessions to introduce new updates and to make the users friendly with the changes been made in the software. The proprietary software packages in the recent times offer a lot more to the user as compared to the amount they charge.</p>

	also puts the user's data at risk.	
Security Issue	<p>Open source software is not compulsorily programmed in a closed space. The giants usually have a dedicated team responsible for the development and programming of the software. But usually the software is being made and tested by people sitting miles apart in different corners of the world and they may or may not be a part of the development of the software for the entire course of time.</p> <p>This disrupts the continuity and the needs of each person might be different resulting in the development going in various directions. This can be advantageous but usually is problematic for the user base.</p>	<p>Security wise, Proprietary software is a step ahead of OSS owing to the fact that they have a dedicated team working day in and day out to make the software problem free and efficient. This is done in a very controlled environment to get best results.</p> <p>Since the source code is only accessible to these people only, it is completely in their hands as to how they wish to modify the software. Also they work hard to avoid any third-party Trojans or any other bugs from reducing the software's efficiency or security.</p>
Software License and copyright	<p>Any software is required to have a license to run. This license is basically an indication of how the software can be distributed and used. OSS are often referred to as "copyleft" software in an attempt to mock how OSS do not follow the guidelines under a copyright.</p> <p>Open Source software is free and can be used and modified by any person. Usually there's no organization in-charge of this and thus the license usually states that there is absolutely no limitation on who downloads the software and how that person manipulates it.</p> <p>This only means that there are no other set of limitations posed on the Open source software and it can be used by anyone who follows the guidelines provided in the license agreement.</p>	<p>The proprietary software on the other hand have a proper license in order to protect their years of hard work. The owner organization is the sole entity responsible for the development of the software whereas the consumers can pay to get the license to only use the software and not modify it. Licensing a proprietary software collects revenue which is used for the furthermore development of the software.</p>
How reliable is it?	<p>A major drawback of the internet is that there is no full-proof way to check if a website is trusted or not. Since a particular Open source software can be available on many websites, the novelty of these downloads is ambiguous. They might or might not be secure, even a very technically sound person might not be able to make out the difference between a healthy and corrupt software. But since the high amount of changes being made to tackle the above mentioned problems, online groups take up the task of extensively going through the code and making it safe for use. They also try to include changes they feel are suitable for the greater masses.</p>	<p>Proprietary Software, as mentioned earlier, have a dedicated team for its development. Each product is tried and tested and since the user is getting the software directly from an authorized organization it is highly trustworthy and safe to use.</p>

Ease of availability	Open Source Software is free to be downloaded from multiple websites on the internet. As discussed earlier, there are online blogs and forums where people can post their issues and grievances which are usually immediately solved.	The software is obtained from the parent organization which own the right to the software. But the trial version of any proprietary software is available free of cost, and is available on the internet too.
How flexible is it?	The system of Open source software is very flexible in its approach. It has proved to reduce the IT cost of companies significantly and gives the scope of innovation a major head start. This flexibility is provided at a very fast rate to the users.	Using a proprietary software is a tedious job since it requires the updating of not just the software but also the hardware in order to keep up with the needs of the new software version. Each new version aims towards solving bugs and issues that arose in the previous version.

5 PROS AND CONS OF PROPRIETARY SOFTWARE

5.1 Pros

a. High effectiveness

Commercial products of proprietary software are designed with a smaller target of capabilities and characteristics. They mainly aim on a small market of users as compared to the open source counterparts. Commercial vendor's customer maybe a developer using a company's utilities like libraries, in fact application users also concern with easy utility and functionality.

b. Powerful

For survival in the market with all the competitors, a rigid hold on the product's blueprint is required. The products are made from the beginning keeping in mind the future aspects and the updates which will come along in the way. It is more robust than the open source counterparts.

c. Full control and authority

It is expensive to have the proprietary software but you get full authority over the software, its future journey with vendors, its upgrades, bug fixes and revised paper work and that it will be sold as an updated version.

d. Support is provided based on your needs

The support system for the software is designed for the company's production in coming years. As they have a small target as compared to their counterparts, preparation and customer support after the software has been sold, is more complete and easy to use. When we compare the forums, the proprietary software company gives direct help from proper consultants and technicians, not just having a question and answer session like in the open source software firm.

5.2 Cons

a. Highly dependent

Customers don't have much of an influence on the proprietary software company so the company may not pay a good attention to the demands and ideas presented by the customer. And if a switch in company has been seen as a necessity, it can turn out to be expensive.

b. Background process hidden

We cannot see the insides and outs of the software. Hence, a little adjustment or modification, is unlikely. If any problem arises, we cannot solve it on our own. The users can only ask for help from the company. Therefore, optimizations are not possible as a proprietary software is closed for viewing internally.

6 PROS AND CONS OF OPEN SOURCE SOFTWARE

6.1 Pros

a. You can try them for free before buying them

Anyone can download the community open source easily, source code too, so that it can be evaluated. It is possible that some COSS(Commercial Open Source Software) vendors even have a free version of software packages including the source code. In fact, most of the open source software, mainly Operating systems, are available as "live" media, so that you don't need to install the software and directly run it from the DVD or USB flash drive.

b. It provides free support

It can be commercially integrated or evaluated but an open source package has the same global community of users and for user support, developers are there. Many things like forums, updates, newsgroups, email list, sophisticated documentation and more is available for free.

c. Open Standard support is provided

Communities with open source packages doesn't adapt to proprietary standards. They prefer open standards of community protocols and data formats. Conformity is improved by doing so, between open source and proprietary software. Operating system is also improved that leads to better conformity for business and customer applications too.

d. It has Fewer Bugs and faster repair rate

Chance of exposing of glitches is pretty high as compared to proprietary vendor's code as bigger open source software projects have a lot of viewings to evaluate the code. Also, open source communities are fast to execute a report. Moreover, as there is a software for source code, customers can easily edit it.

e. More secure than its counterparts

As there is fast repair, security provided is well and good. A proprietary software vendor's code can be cryptic or dense, it is hard to find the security bugs. It can be possible that there is an error but we won't be able to see it.

f. Problem of vendor lock-in is avoided

Keeping aside the case of COSS, there is a very less dependency on single vendor for enhancements, repairs and backup for open source software. Moreover, there is a very less chance of threat that the software may be a left-out software as the open source society is scattered and assorted. This can be a case where the proprietary vendor had to cancel their task.

6.2 Cons

a. Reduces the competitive aspect

If a company also sells a software, then making products with open source code accumulates the proceeds from the sell, for the company's software relying on the license agreement for the open source software. When the proprietary code's worth is depended on the open source platform is not sufficient to set the decreased constraint for allowing challengers to make similar software on the same open source code.

b. Not much support is provided

Big projects on open source have a large community that helps with the paper work, techniques and help mechanisms to support the users. Free support is not always efficient, sometimes the company wants a way to solve a problem with random code bugs. Big companies with which can pay for the best support available can have a good attention that is very unique from open source community.

c. Not a very friendly user interface

Open source projects are not that simple to understand. They target the users with less knowledge like the proprietary counterparts. Those users will not even care about the source code.

d. Can cause a huge economic risk

Except Red Hat, big vendors of open source software have no competition with equals. Sometimes it is possible if you require a product, there is a risk that the small, less experienced companies may not fulfill the requirements.

7 WHY CHOOSE OPEN SOURCE OVER PROPRIETARY?

Money-wise, Open source software is obviously the better choice because if we're getting the same functionalities for free, one should obviously go for it. The money spent of proprietary

software every year is only increasing exponentially. This money can be used to expand the IT sector if we replace proprietary software with open source software. The overhead generated by proprietary software is something that needs to be eliminated since India has a load of work load but low advancements on the monetary side. Factors like low maintenance, high scalability, high flexibility, open access to the source code, great performance etc. are usually credited for open source software to be such a huge success. Business where there are no constraints exercised on the resources, OSS might not solve any issues but in business' where the need to compete and be better than the others is felt strongly, open source software is a cost effective solution because it gives the scope to expand business using the funds saved.

8 CONCLUSION

The recent trends have shown and demonstrated the wide and abundant use of Open source software over the Proprietary software. The widespread popularity of OSS if forcing the proprietary software makers to either step up their product so that the OSS cannot compete with it or reduce the costing so that more people continue to buy them. Both ways, the consumers are the ones gaining benefit from the whole scenario which is the ultimate aim.

Success stories of Open source software can be seen in the case of IBM which will be spending \$1 Billion in order to use Linux in their workspaces. Amazon has already shifted their work base from Windows to Linux and has cut down approximately 25 percent of their technological expenses.

We don't know if Open source software is here to stay or not but one thing is for sure that they are key holders in the market in the present times.

ACKNOWLEDGMENT

The authors wish to thank VIT University, Vellore.

REFERENCES

- [1] Amandeep Singh, et.al, "Open source vs proprietary software" (2015)
- [2] Robert Merges, Peter Menell, "Intellectual Property in new technological age" (2006)
- [3] <http://ben.balter.com/open-source-alternatives/>
- [4] Mukund Raj et.al, "Proprietary software vs Open source software for education purposes" (2013)
- [5] <https://www.3pillarglobal.com/insights/proprietary-vs-open-software-the-pros-and-cons-explored>
- [6] <https://www.cmswire.com/cms/information-management/open-source-vs-proprietary-software-there-is-no-clear-winner-021752.php>

IJSER