DQ1: LAMP

[Collapse](https://elearning.uol.ohecampus.com/webapps/discussionboard/do/message?action=list_messages&forum_id=_952804_1&nav=discussion_board&conf_id=_441678_1&course_id=_1470762_1&message_id=_15931440_1)

Top of Form

Summarise the advantages and disadvantages of using open source technologies such as LAMP (Linux-Apache-MySQL-PHP).

Bottom of Form

1. Introduction
   1. What is open source software
      1. OSI, 2014)
         1. Open source software is software that can be freely used, changed, and shared (in modified or unmodified form) by anyone. Open source software is made by many people, and distributed under licenses that comply with the Open Source Definition.
      2. Opensource.com, 2014
         1. The term "open source" refers to something that can be modified because its design is publicly accessible.
         2. Open source software is software whose source code is available for modification or enhancement by anyone.
         3. "Source code" is the part of software that most computer users don't ever see; it's the code computer programmers can manipulate to change how a piece of software—a "program" or "application"—works. Programmers who have access to a computer program's source code can improve that program by adding features to it or fixing parts that don't always work correctly.
         4. Open source software is different. Its authors make its source code available to others who would like to view that code, copy it, learn from it, alter it, or share it. LibreOffice and the GNU Image Manipulation Program are examples of open source software. As they do with proprietary software, users must accept the terms of a license when they use open source software—but the legal terms of open source licenses differ dramatically from those of proprietary licenses. Open source software licenses promote collaboration and sharing because they allow other people to make modifications to source code and incorporate those changes into their own projects. Some open source licenses ensure that anyone who alters and then shares a program with others must also share that program's source code without charging a licensing fee for it. In other words, computer programmers can access, view, and modify open source software whenever they like—as long as they let others do the same when they share their work. In fact, they could be violating the terms of some open source licenses if they don't do this.
      3. Jaisingh, J, See-To, E, & Tam, K 2008
         1. The primary difference between OSS and cSS is that the source code of an OSS is accessible to everyone whereas the source code is proprietary in the case of a cSS.
   2. What are examples of open source software?
      1. (University of Liverpool/Laureate Online Education, 2014)
         1. Linux, Apache, MySql, PHP
      2. Filezilla
2. Advantages
   1. Jaisingh, J, See-To, E, & Tam, K 2008
      1. One consequence of keeping the source code open is that the OSS can benefit from modifications and improvements made by programmers from all around the world. The cSS, on the other hand, can only be improved or modified by programmers hired by the firm developing the cSS (henceforth referred to as the firm when there is no risk of confusion). It is well known that OSS such as Linux and apache have been developed and enhanced by contributions from thousands of volunteer programmers. another consequence of keeping the source code open is that the firm cannot charge a price for purely selling the OSS—the open nature of the source code will drive the price down to zero. Most of the OSS such as Linux, apache, SendEmail, etc., can be obtained free of charge.
   2. Updegrove, A 2009
      1. Access to code: When a user installs proprietary software, they become entirely dependent on the vendor for it's quality, improvement and performance, because the customer has neither the technical means (access to source code) nor the right (legal permission) to alter the code. If the customer needs new or different features, the vendor may or may not be willing to customize the program (either at all or at a price the customer is willing to pay), and if the vendor discontinues support for the product, or goes out of business, the customer is stranded.“ In contrast, a customer with a FOSS alternative has the ability as well as the legal right to change the code any time that it wants to. It can also hire anyone it wants to help it change or maintain the code, and if the project that created the code goes dormant, it may be disappointed, but it will not be stranded.
      2. Freedom from lack in: While open standards increasingly give customers protection from “lock in" dependency on a single vendor, and the certainty of signiﬁcant switch g costs if they wish to change vendors), changing from one product to another can still be difficult and expensive in many situations. In the case of systems based on Linux, the increasingly popular FOSS operating system (05), there are currently over we independent “distributions,” all based on the same core software (the |Jnux kernel). While application software will not always automatically run across all versions, all of the major distributions certify their products to the |Jnux standards Base (LSB), a set of standards currently maintained by the Linux Foundation, in order to permit application software to run more interoperably across compliant distributions.
      3. Security: While it may seem counterintuitive that code Vlslble to anyone anywhere would be safer to use, popular FOSS programs are acknowledged to be more secure, largely for the same reasons just stated: because anyone can see the code, anyone can tmck down the source of a vulnerability, let project managers know of the cause of concern, and/or propose a ﬁx herself. As a result, security issues can typically be identiﬁed, ﬁxed, and propagated to all users faster than ﬁaws in proprietary code.‘ As a result, FOSS is increasingly being used by defense, ﬁnancial and other types of users where security concerns are greatest.’
   3. K Ven, J Verelst, H Mannaert - Software, IEEE, 2008
      1. Proponents argue that making source code available lets everyone peer review the code, resulting in higherquality software. It’s also suggested that it gives users more choice and control because it lets them read and modify the source code. Although many OSS advocates have proclaimed these advantages, several authors have questioned or cast doubt on them.
3. Disadvantages
   1. Hammond, S 2009
      1. Disadvantages: the issue of ongoing maintenance and support.
      2. Support and IP issues: Lee pointed out another issue with OSS: the open nature of the code which can cause confusion on intellectual property issues. "With the GPL (GNU General Public License, see sidebar "What's a GNU?"), be careful if you develop some of your IT on top of the public code. But dynamic linking means that you link your code in runtime rather than compile time, and there are services for OSS audits, to ensure that you're in compliance. You don't want to give up your competitive edge."
      3. Another issue that concerned Lee is that "the OSS community seems to be interested in fashionable tech—they develop a 'hot area' for a few months, then it's out of fashion and they work on something else, and what can you do? It's like you're left with an obsolete technology— you can't control your destiny."
   2. K Ven, J Verelst, H Mannaert - Software, IEEE, 2008
      1. In the first scenario, the source code’s availability is neither an advantage nor a disadvantage for the organization. Half of the organizations in our sample indicated that they didn’t consider source code availability to be an advantage and that they never used the source code. This is consistent with other studies in this field.2,4–6 Joseph Feller and Brian Fitzgerald labeled this the “Berkeley Conundrum,” which questions the importance of the source code’s availability if no one actually uses it.13 At least two factors can account for this observation:
         1. Our study focused on highly mature infrastructure software such as Linux and Apache. Organizations have little need to modify the source code of such applications.2,5,6 Other types of OSS might give different results.
         2. Few—even experienced—programmers can modify the source code of mature software such as Linux and Apache.2
      2. In this scenario, OSS serves as a black box, and its advantages and disadvantages are comparable to proprietary packaged software.
4. Conclusion

References:

Hammond, S 2009, 'Open source 2009. (cover story)', Computerworld Hong Kong, 26, 6, pp. 12-14, Computers & Applied Sciences Complete, EBSCOhost, viewed 14 June 2014.

Jaisingh, J, See-To, E, & Tam, K 2008, 'The Impact of Open Source Software on the Strategic Choices of Firms Developing Proprietary Software', Journal Of Management Information Systems, 25, 3, pp. 241-275, Business Source Complete, EBSCOhost, viewed 14 June 2014.

Opensource.com (2014) *What Is Open Source*? [Online]. Available from: <http://opensource.com/resources/what-open-source> (Accessed: 14 June 2014)

Open Source Initiative (2014) *The Open Source Initiative* [Online]. Available from: <http://opensource.org> (Accessed: 14 June 2014)

University of Liverpool/Laureate Online Education (2014) *DQ2 from Programming the Internet Module Seminar 4* [Online]. Available from: University of Liverpool/Laureate Online Education Blackboard (Accessed: 12 June 2014).

Updegrove, A 2009, 'A Concise Introduction to Free and Open Source Software', Standards Today, 8, 5, pp. 8-46, Business Source Complete, EBSCOhost, viewed 14 June 2014.

Get from refworks:

Should you adopt open source software?

K Ven, J Verelst, H Mannaert - Software, IEEE, 2008 - ieeexplore.ieee.org

... Table 1 Claims and counterclaims about open source software ... is neither an advantage nor a

disadvantage for the organization. ... In this scenario, OSS serves as a black box, and its advantages

and disadvantages are comparable to proprietary packaged software.14 In the ...

Cited by 78 Related articles All 8 versions Cite Save More