

# Study of Maintainability Process in Open Source Software

Sheenam Batra

**Abstract**— The focus of this paper is to study about maintenance processes of International Standard Organization (ISO) and Open Source Software (OSS) As Open Source Software is becoming more popular, hence its maintenance has become most important issue. So in this paper we study the maintenance processes and also focus on differences between them. In future we are going to calculate the maintainability index of various versions of open source software to know the variations and modifications in each of the version.

**Keywords**— maintenance, maintainability index, open source software, open source software development process.

## 1 INTRODUCTION

Open Source Software (OSS) is software whose source code is freely available for anyone to inspect and study. Open source software not only provides the right to inspect and study the source code but also allow us to use it for any desired purpose without monetary or other restrictions. These other purposes include making as many copies as desired, installing on as many computers as desired, modifying in any desired way, and redistributing in its original or modified form [1].

The main features that characterize open source software is the freedom that users have to:

1. Use the software on many computers as they desire, in any technically appropriate situation.
2. Use the software to fit to their needs. Of course, this includes improving it, fixing its bugs, augmenting its functionality, and studying its operation.
3. Redistribute the software to other users, who could themselves use it according to their own needs. This redistribution can be done for free, or at a charge, not fixed beforehand.

### 1.1 Examples of Open Source Software, Servers and Operating Systems

1. OpenOffice.org – office suite,
2. Mozilla and Firefox – web browsers,
3. Thunderbird – email client,
4. MySQL–Relational Database Systems
5. GNU/LINUX – Operating Systems
6. Apache web server – server [2].

### 1.2 Some of the common open source software licenses are as follows:

1. BSD (Berkeley Software Distribution). The BSD license imposes almost no conditions on what a user can do with the software. Redistributors can do almost anything with the software, including using it for proprietary products. The

authors only want their work to be recognized [3].

2. GPL (GNU General Public License).It allows modification without restrictions (if the derived work is also covered by GPL); and complete integration with other software is only possible if that other software is also covered by GPL [3].

3. MPL (Mozilla Public License). This is the license made by Netscape to distribute the code of Mozilla, the new version of it is network navigator. It is in many respects similar to the GPL, but perhaps more "enterprise oriented" [3].

### 1.3 Benefits of using Open Source Software:

1. Reliability: OSS is reliable software. We can take it to mean the absence of defects which cause incorrect operation, data loss or sudden failures [4].
2. Cost: By using OSS we can minimize cost as its source code is freely available.
3. Freedom: OSS gives us freedom to study, change and redistribute the source code [4].

### 1.4 Disadvantages of Open Source Software:

1. No support exists for open source software: There are no help documents and manuals are made available for OSS.
2. Higher installation costs: Most of the open source applications are incompatible with the present day gadgets. Technical support in this case is costlier compared to commercial software, so it increases the installation cost.
3. No guarantee of updates: Since we are not paying for the open source software, nobody is bound to give regular updates. We can get stuck with the same old version without ever getting an update [5].

## 2 OPEN SOURCE DEVELOPMENT COMMUNITY MODEL (OSSD)

The OSSD model has the following features:

1. Development Community includes those developers which want to contribute their work in open source software development.

• Sheenam Batra is currently pursuing masters degree program in Computer science and applications in Thapar University, India.

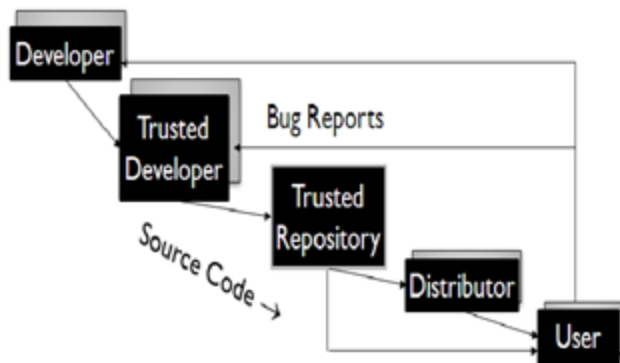


Fig. 1.OSS Development Community Model [6]

2. Trusted Developers includes those developers which are expertise in a specific domain.

3. Trusted Repository is a repository where source code resides. Whenever trusted developer develop a new code or made some changes in code, they put the code in trusted repository.

4. Now user can directly access the source code from trusted repository as well as through distributors.

5. While using the software, if user will find any bug in the source code then user can report it to either developer or trusted developer.

### 3 SOFTWARE MAINTENANCE

In a changing environment, the software is also supposed to change. Software maintenance is one of the processes in the software life cycle. According to the Software Body of Knowledge (SWEBOK) [IEE04], software maintenance is defined in the IEEE Standard for Software Maintenance as follows

“Software product after delivery to correct faults, to improve performance or other attributes, or to adapt the product to a modified environment” [7]

The four main maintenance categories are:

#### 1. Corrective maintenance

Corrective software maintenance involves developing and deploying solutions to problems ("bugs") that arise during use of a software program.

#### 2. Perfective maintenance

Perfective software maintenance involves computer programmers working to improve the way a software program functions or how quickly it processes requests.

#### 3. Adaptive maintenance

The field of technology constantly changes through both hardware and software developments. Adaptive software maintenance addresses these changes.

#### 4. Preventive maintenance

When computer programmers engage in preventative software maintenance they try to prevent problems with software programs before they occur. Computer programmers test software to make sure the software can handle high data loads and other stressful operations without problems.

#### 3.1 Maintainability Index (MI)

The MI [8] is a composite metric, based on several metrics. It is based on following metrics:

1. Halstead Volume (HV) metric
2. Cyclomatic Complexity (CC) metric
3. Average number of lines of code per module (LOC)
4. Percentage of comment lines per module (COM).

Halstead Volume is a composite metric based on the number of (distinct) operators and operands in source code.

Cyclomatic Complexity is the number of linearly independent paths through a program.

Lines of code is a software metric used to measure the size of the source code.

Comments per module are the number of comment lines in the source code of a module.

The formula for maintainability index is:

$$MI = 171 - 5.2 * \ln(\text{aveV}) - 0.23 * \text{aveV}(g') - 16.2 * \ln(\text{ave-LOC})$$

Where

aveV = average Halstead Volume V per module

aveV(g') = average cyclomatic complexity per module

aveLOC = the average count of lines of code (LOC) per module

The higher the MI, the more maintainable a system is.

### 4 ISO SOFTWARE MAINTENANCE PROCESS:

An overview of the ISO maintenance process [9] is presented in Figure 2

ISO primary software maintenance activities are broken down into following tasks.

#### 1. Process Implementation:

In process implementation, first of all maintenance plans and procedures are developed for modification request and finally implement the configuration management process.

**2. Problem and Modification:**

In problem and modification step, tasks are performed for initial analysis and verify problem. After verifying problem, options are developed for implementing the modification and results are documented. Finally obtain approval for modification option.

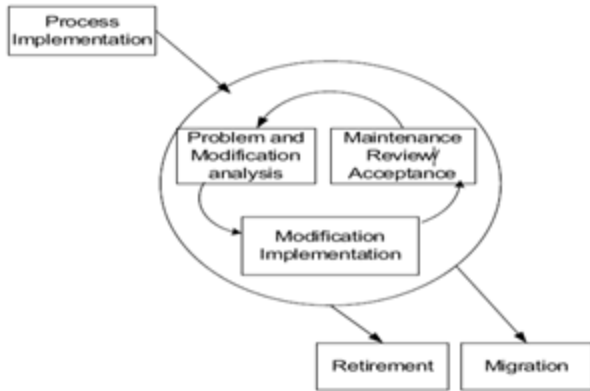


Fig 2: ISO Software Maintenance Process

**3. Modification Implementation:**

In this step, main task is to perform detailed analysis and then develop the code and test the modification

**4. Maintenance Review/Acceptance :**

In this task, various reviews are conducted and then obtain approval for modification.

**5. Migration:**

In migration task, we ensure that the migration should be in accordance with ISO. After then migration plan is developed. Users are notified about migration plans and conduct parallel operations. Then users are notified that migration has started and finally a post-operation review is conducted and ensure that old data is accessible.

**6. Software Retirement:**

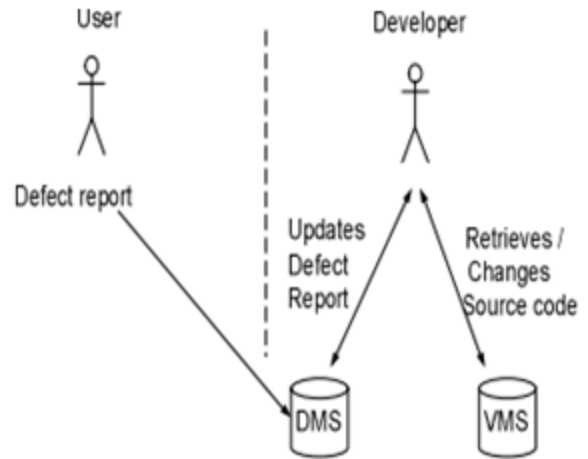
In software retirement task, retirement plan is developed. After developing the retirement plan users are notified about retirement plans and conduct parallel operations. Finally users are notified that retirement has started and ensure that old data is accessible.

**5 OPEN SOURCE SOFTWARE MAINTENANCE PROCESS (OSSPs)**

OSSPs [10] use two kinds of management systems  
 Defect Management System (DMS)  
 Version Management System (VMS)

Defect management system is used to store the defect reports.

Whenever source code is modified in software develop-



ment, sometimes changes are not good. In that case developers are able to restore the older version in VMS.

Fig 3: OSS Software Maintenance Process

Following Activities are involved in Open Source Software

1. Whenever user finds any defect, he/she can report it to the DMS
2. Developers can retrieve the information reported by users and analyze it.
3. The developer can retrieve the source code from the VMS and modifies the source code to remove the defect.
4. The modification is reviewed before it is accepted and merged to the main version of the source code in the VMS.
5. The status of the defect is updated into the DMS.
6. Then, a new version of the software is available from the VMS or from the project web site. Then the users can retrieve the modified version.

**6 DIFFERENCES BETWEEN ISO AND OSS MAINTENANCE PROCESSES**

The major difference in ISO and OSS maintenance processes is of retirement activity and migration activity. In ISO maintenance process, Retirement and Migration plans are first developed and then notified to the users. Also, it conducts parallel operations to inform users that plans have been started. While in OSS maintenance process there are no such activities. Another difference between ISO and OSS maintenance process is in modification review and acceptance. Modifications are accepted before implementation in the ISO maintenance process but after implementation in the OSS maintenance process. In OSS maintenance process developers can retrieve any of the defects and start to implement the required modifications. After implementations, the modification is reviewed and can still be discarded. [11]

**CONCLUSION AND FUTURE SCOPE**

Open Source Software is software whose source code is freely

available for study, modification and redistribution with few restrictions. Maintenance of open source software is very important. In this paper we have studied the ISO maintenance process and OSS maintenance process using defect management system and version management system. Defect management system is basically used to store defect reports submitted by users and version management system acts as repository for source code.

Matter presented in this paper is only the preliminary review study of open source software and importance of maintenance process in open source software development. In future aim is to study the flow of maintenance process using open source software having large versions. Tool used to study the process is JHawk.

## REFERENCES

- [1] [http://www.linfo.org/open\\_source.html](http://www.linfo.org/open_source.html)
- [2] Prof. Dr. Sabina Jeschke, Dr. Claudia Muller, Sven Grottko, "Lecture on Free/Libre and Open Source Software", Institute of Information Technology Services, University of Stuttgart, 2009.
- [3] <http://www.opensource.org/licenses/agpl-v3.html>
- [4] [http://open\\_source.gbdirect.co.uk/migration/benefit.html](http://open_source.gbdirect.co.uk/migration/benefit.html).
- [5] [http://open\\_sourcetechnologies.blogspot.com/2008/06/disadvantages-of-open-source-software.html](http://open_sourcetechnologies.blogspot.com/2008/06/disadvantages-of-open-source-software.html)
- [6] W.W.Ming, Y.D.Lin, "Open Source Software Development", In proceeding of the international symposium on digital object identifier, pp. 33-38, 2001.
- [7] IEEE: *Guide to the Software Engineering Body of Knowledge (SWEBOK)*. IEEE Computer society 2001, Los Alamitos, California.
- [8] T.Kuipers, Joost Visser "Maintainability Index Revisited"
- [9] T. Koponen, V.Hotti, "Open Source Software Maintenance Process Framework", In proceeding of 5th workshop on open source software engineering,
- [10] T.koponen, Heli Lintula, V.Hotti, "Exploring the Maintenance Process through the Defect Management in the Open Source Projects-Four Case Studies", In proceeding of International conference on software engineering, 2006.
- [11] T.Koponen, V.Hotti "Evaluation Framework of Open Source Software", In Proceedings of The International Conference on Software Engineering Research and Practice Vol. II., pp. 897-902, 2004