 ISSN: 2321-8169
1116– 1118

Cloning in Reused Components

[1]Ritu Garg

CSE Department
DeenbandhuChhotu Ram
University of Science and Technology,
Murthal, Haryana, India

Abstract -- Software systems are newly created or modified as per demand. The system can be created or modified with or without reusing the built in or user defined candidate components from system library which makes the process ease. In both the cases cloning can occur in software which needs detection and removal accordingly. Its removal can be further easily done with the help of reusing. This decreases time and cost related to the software development.

Keywords-- Software clones, duplication, components, system library, reusing

*****

I. INTRODUCTION

A software system has many components. The reused components are those components that are available in system library so there is no need to rebuild those components but they can be reused from system library. This process reduces time and cost of the software [1].Cloning refers to the duplication that exists within the software or between two or moresoftware's [2, 3, 4].Cloning on its own has many advantages and disadvantages [5, 7].

II. BACKGROUND

Cloning can be removed by reusing but reused components may itself have cloning sometimes. So, there is a need to detect clones among the reused component and then avoid or remove the clones accordingly as shown in figure 1. This occurs mainly when a software require not the inbuilt library component as a whole but a part of it. In that case there is a need to remove clones between the library component and the software under development.

Cloning can occur in four types [6]

1. Type-1: Exact clones
2. Type-2: Modified clones
3. Type-3: clones
4. Type-4: Semantic clones

The various techniques associated with the detection of clones are [7]-

1. Text based clone detection
2. Token based clone detection
3. Tree based clone detection
4. Graph based clone detection
5. Metric based clone detection
6. Hybrid based clone detection

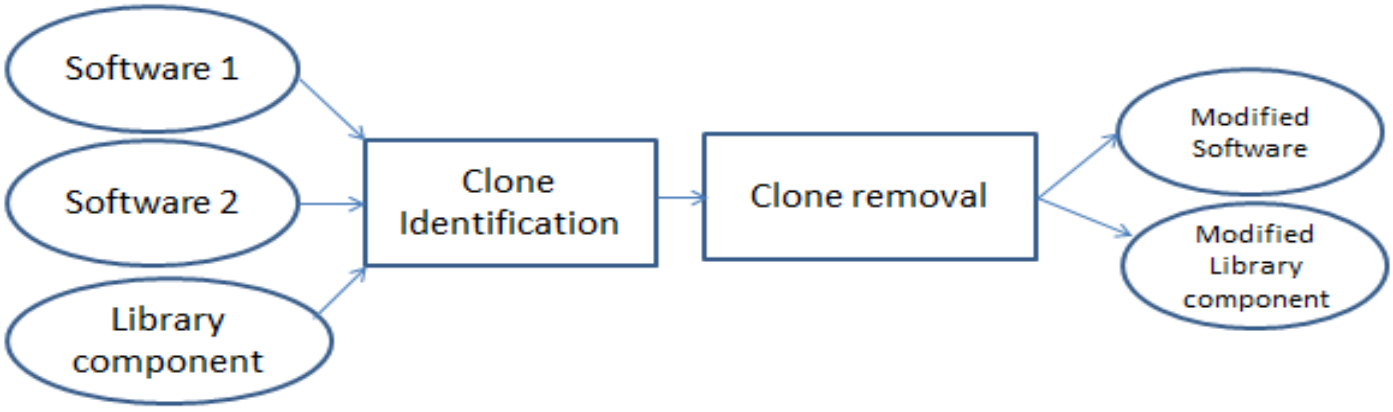

Figure 1. Representation of phases of clone handling

 ISSN: 2321-8169
1116– 1118

III. TECHNIQUES

Cloning can not only occur between different software's under construction but also between the software component and library candidates as shown in figure 2. In that case both the software and library component can be modified after the removal of clones. The clone pairs can be treated further as a library component and placed in the library as a candidate component and then reused at all the places where it's required as shown in figure 3. This process can occur only when the size of library candidate after division should be large enough that it should decrease the cost and time then the library candidates must not be divided.

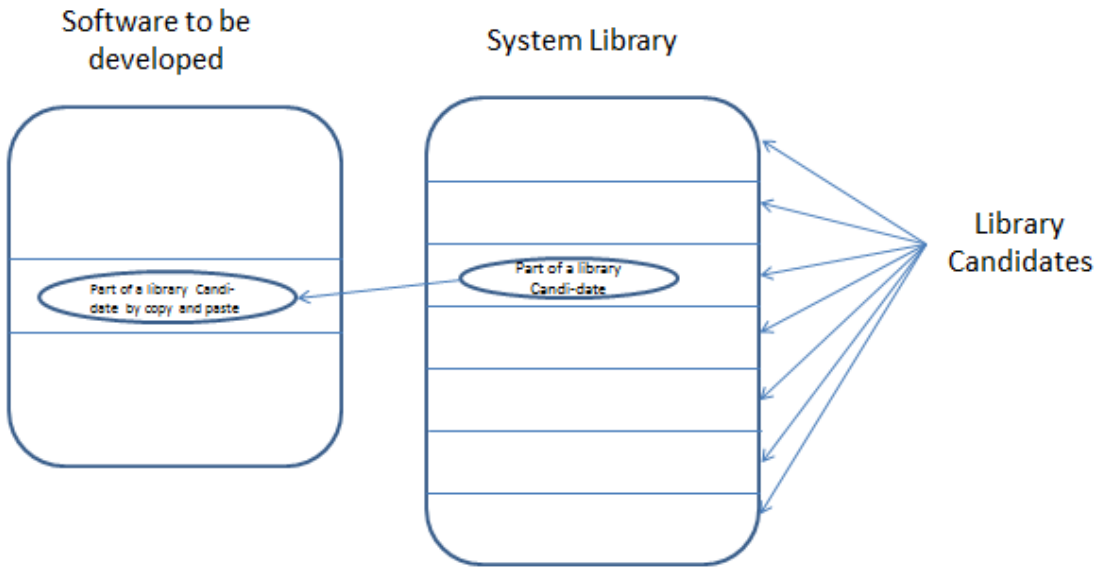

Figure 2. Representation of copy and paste approach in software to be developed from a part of system library

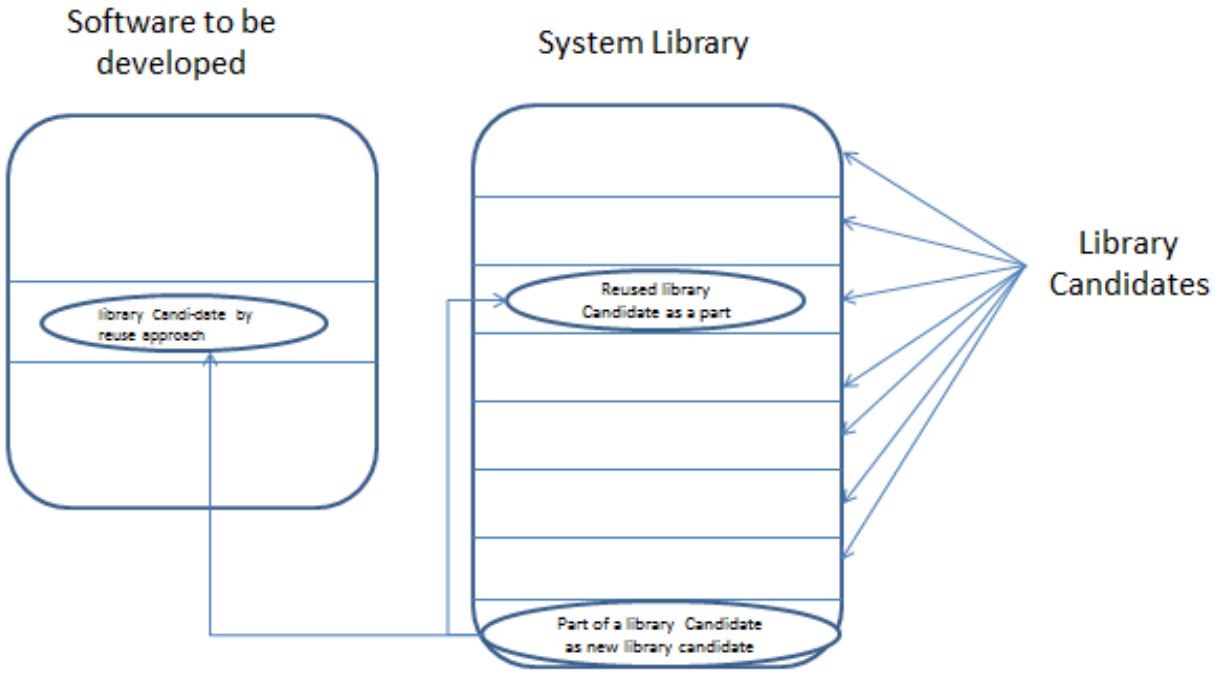

Figure 3.Representation of reuse approach in software to be developed from a part of system library after making the reused part a new component of system library

IV. CONCLUSIONS

The clones can occur within the reused components present in the library as a candidate component. So, those clones should be identified and removed if the cost of the software under construction can decrease using it. It also decreases the risk associated with the software. It should be done to decrease the effort, time and cost associated with software maintenance. The quality is thus improved due to this approach.

V. FUTURE WORK

Much more work is requiredwith the clone relevancy.It should takedecision on basis of whether clones are relevant with respect to the software or not. If the clones are relevant, then they should be removed. This decreases time and cost of software maintenance while the clone irrelevancy means clones that should not be removed. Such clones on their removal can instead of decreasing, increases the time and cost of the software. There needs an improvement in terms of clone management.

REFERENCES

[1] Garg, R., Bhatia, R. (2014). Code Clone v/s Model Clones: Pros and Cons , International Journal of Computer Applications (IJCA), 89, 15, 20-22

[2] Baker, B. , (1995). On finding duplication and near-duplication in large software systems, Proceedings of the 2nd Working Conference on Reverse Engineering , 86–95.

[3] Kapser, C. J., Godfrey, M. W. (2008). Cloning considered harmful:patterns of cloning in software,Empirical Software Engineering, 13 (6), 645–692.

[4] Sharma, Y., Bhatia R. ,Tekchandani , R . K. (2011). Thesis on Hybrid technique for object oriented software Clone detection , Electronic Theses & Dissertations @ TU

[5] Rattan, D., Bhatia, R., Singh, M. (2012). , Model clone detection based on tree comparison, India Conference (INDICON),2012 Annual IEEE, 1041-1046.

[6] Störrle, H. (2010). Towards clone detection in UML domain models, European Conference on Software Architecture (ECSA'10) , 285–293.

[7] Rattan, D., Bhatia, R., Singh, M. (2013). Software clone detection: A systematic review, Information & Software Technology, 55, 1165-1199