Implementation of Model Cloning in Software Models using UML diagrams

Jashandeep Kaur[a,*], Rasbir Singh[b]

[a] *MTech Student,,RIMT-IET, Mandigobindgarh,Punjab,India*
[b] *Assistant Professor, RIMT-IET, Mandigobindgarh,Punjab,India*

IJTC.ORG

ABSTRACT

As model based development in the software field is increasing rapidly. Clone detection is emerging as an active research area. Models are integral part of the software development. The maintenance of the model designed is very important. One of the major problems that occur when the model is designed is cloning. Cloning is process of creating the copies of various elements of the model. Clones are divided into two code clones and model clone. Model cloning is the process of development of the duplicate parts of the models. Code cloning is the process of duplication of the code .The cloning results in the increase in the maintenance cost of the model and also increases the probability of bugs in the system. So the detection of the clones is the important process. In this paper a detailed study of the cloning is presented. Along with this the major causes of the creation of clones and various methods of clone detection have been discussed.

Keywords—Clone detection ,software development ; code clones , model clones .

I. INTRODUCTION

In software development the elements of one model are copied and are pasted in the other section without any changes. This process is termed as the cloning. The cloning occurs at two stages either at designing level or at implementation level. So on the basis of this cloning can be defined as the code cloning or the model cloning. The cloning that exists at the implementation level that is in the source code which is termed as the code cloning. The cloning that exist during the designing of the model is termed as the model cloning

Clones are harmful as they increase the redundancy of the system which will in turn affect the maintenance of the system. Due to cloning the maintenance cost and the probability of the bugs increases. So clones are quite harmful for the system so they needs to be detected and removed before they could harm the system

When the software is developed the system is models by using the ULM diagrams. In the ULM diagram the model clones are detected at the initial stages when the system is developed. These model clones are removed and thus they are prevented at the earlier stages.

There are many challenges in identifying model clones. In this paper various causes of the model clones have been discussed and the types according to which the model clones are classified are also defined.

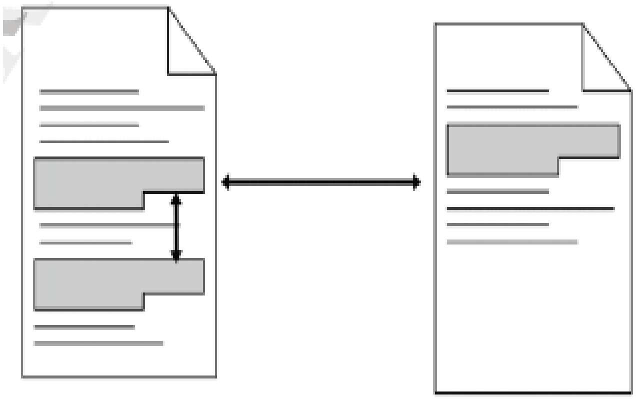

Fig. 1. Example of cloning

II. REASONS OF CLONES

IJTC.ORG

The main reason that causes the clones is described below:

• **Copy /pasting of the elements of the model**: This is one of the major reasons that cause the clones. As the elements are copied /paste in order to create a new copy of the model element. These are basically created to reuse the model elements. This method is mostly used by the software designer that causes model cloning.

• **Language loopholes:** This is another method that causes the model cloning. This is basically due to the limitation of some languages that some of the models are repeated that causes the model cloning.

• **System complexity:** There is difficulty in understanding the large system as the copying the elements of the existing model make it complex.

• **Time limits assigned to software designers:** This is also major cause of the model cloning. The time allotted to eth designers for the completion of the task also play an important role as if the task required more time and less time is allotted to them they will find easy way for completing it . They just copy and paste the exiting one and adapt to their current needs

• **Lack of knowledge in a problem domain.** This is also considering as the major reason of cloning**.** Some time the designers are not familiar with the problem domain .if the solution is found they will easily adapt that solution as per there need without actual knowledge of the problem. So this also leads into the model cloning.

• **Clones by the intention of designer:** sometime the programmer intentionally creates the clones to create the part of the model.

III. TECHNIQUES OF CLONING

Clone detection is the important on order to decrease the probability of the bugs in the system . Various techniques of clone detection have been proposed. Some of the technique has been described below:

1) **String-based technique**: This technique is also known as the text based technique. In this the source code that is used either goes under less or no transformation. if the case of codes are considered the white space that occur in code and the comments they are all ignored . The codes are divided into lines. These lines are then compared to the algorithm to find the duplicate data .

2) **Token-based technique**: In this the lexical analysis is done, in this the stream code are used as the basis for the clone detection. This stream code is compared with the source code.

3) **PDG-based technique**: In this the graphs are used to represent the source code. These are program dependence graphs that depict the nature of the source code.

4) **AST-based technique:** In this the parsing are used to represent the source code. The detection algorithm is used is then compare with the sub-trees .

5) **Metrics-based techniques**: In this technique the hashing algorithm are used. In this the metrics are calculated for each code. Then, code fragments are compared and detected from the given data .

IV. RELATED WORK

Clone detection is the important role in the software evolution process in which the same entity is present over multiple versions. Various contributions have been made in literature regarding the detection of the clone. Some paper has been discussed blow that were used for studying the literature:

Anisha Rani et al [1] present the approach for the reduction of clones in the software models. Code clones are considered to be harmful. This clones are basically the unwanted and unnecessary multiple segment of the code. The cost of the maintenances is also increased due to the cloning. These duplicated segments are harmful when the model is designed. The UML are used for the designing of the models. So to remove the duplicity in the models, clone detection techniques are used so to make system more efficient. In this paper a new technique is proposed for the reduction of the clones in the system in which the software models are designed using the XLM models. From the results obtained it is concluded that this method is efficient for the reducing the cloning

Harjot Kaur et al [2] presents an approach for the detection of the clones in the class diagram. In this basically the redundant elements that are present in the class diagram are removed that increases the complexity of the class diagram. The ULM models are used for designing and developing the software. The various unexpected copy of the elements can cause harm to the software as the similarity level of the element can make system complex. So these clones are to be detected. In this paper the XLM files are used that are encoded as the class diagrams. From these diagrams the tokens are extracted that are matched. So the similarities in the tokens are detected that are termed as the clones

Dhavleesh Rattan et al [3] presents a new approach for the computing of similarity in object oriented program at different levels. This approach is used for the detection of the similarity in the ULM diagrams. This is done by measuring the distance between the two models. In the traditional approaches the high level similarity design were designed and used for the better maintenance of the system. So in this proposed

IJTC.ORG

approach the traditional method is extended to improve the system Form the results obtained it is concluded that this proposed method is better than the traditional methods of clone detection

Girija Gupta et al [4] introduce a new method for the detection of the cloning. Clones are the harmful for the maintenance of the software as it also increase the complexity of the system. As if the size of the cloning is increased the designing problem of the system will also increase. Clones codes are referred as the duplication of the same element in the system. The detection of the clone is important as it will decrease the complexity of the system. For the detection of the clone various techniques have been proposed earlier so that the maintenance cost and the complexity of the system is decreased.. The mostly commonly used approaches are the text base comparison and the token based detection but still the system was not as required . So in this paper a new approach that is code clone detector is used for the detection of the clone. From the results obtained it is concluded that this method is efficient than the traditional method of the clone detection.

Cory J. Kapser et al [5] presents the method of the detection of the clone. The duplication of the code is termed as the cloning that is major problem in the industrial software systems. This cloning results in the increase the complexity of the system and also result in the increase in the maintenance cost of the system. For the detection of the clone various techniques have been proposed. This paper presents the in depth study of the various cloning technique. Form the result obtained it is concluded that this method is efficient.

Chanchal K. Roy et al [6] presents the generic clone detection process in which the comparison between the techniques has been compared. This approach is divided into two parts. In first section the techniques used are compared and evaluated in the two dimensions. In second section on the basis of the classified the technique are classification is made on the basis of the creation of Type-1 , Type-2 , Type-3 and Type - 4. On the basis of this the clone detection technique is defined. From the experiments preformed it is concluded that this method is better and efficient than the traditional approach.

V. CONCULSION AND FUTURE SCOPE

Model clones are basically defined as the model that is segmented into various model elements that have some similar properties. Model clones are the segmented model of the model that consists of high degree of similar data. Model clones are harmful as it increases the maintenance cost and the probability of the bugs in the system. So clone detection is the important task for which various methods have been proposed. This paper mainly focuses on the detailed study of the model cloning. The main reasons of the model cloning is discussed, along with this types of model cloning are also studied. In future various other solutions can be found for the clone detection. Till date there is no specific classification of the clones. So the clones should be specified in the proper manner so that it is detected easily. Along with this the methods should be found for clone detection so that the maintenance cost and probability of occurring errors are reduced .

IJTC.ORG

REFERENCES

[1] Anisha Rani , “Reducing Cloning in Software Models Using UML Parsing” , International Journal of Engineering Research & Technology , Vol. 2 , Issue 11, November – 2013, Pp 3800-3805

[2] Harjot Kaur,” Detecting Clones in Class Diagrams Using Suffix Array ” International Journal of Engineering and Advanced Technology, Volume-3, Issue-4, April 2014,Pp 243- 246

[3] Dhavleesh Rattan, “Detecting High Level Similarities in Source Code and Beyond” ,International Journal of Energy, Information and Communications Vol.6, Issue 2 (2015), Pp.1-16

[4] Girija Gupta, ‘‘Novel Approach Towards Code Clone Detection and Redesigning “International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 9, September 2013 ,Pp 331-338s

[5] Cory J. Kapser, “Supporting the Analysis of Clones in Software Systems: A case study”, journal of software maintenance and evolution: research and practice, 2005, Pp 1-21

[6] Chanchal K. Roy, “Comparison and Evaluation of Code Clone Detection Techniques and Tools: A Qualitative Approach”, Preprint submitted to Science of Computer Programming, February 24, 2009

[7] Balwinder Kaur , “Clone Detection in UML Sequence Diagrams Using Token Based Approach “International Journal of Advanced Research in Computer Science and Software Engineering , Volume 5, Issue 5, May 2015 ,Pp 34-41

[8] S.MYTHILI “A Novel Weight Assignment Approach for Detection of Clones in Simulink Diagrams“Volume 14, 2015, Pp 253-263

[9] Harjot Kaur, Detecting Clones in Class Diagrams Using Suffix Array “International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-3, Issue-4, April 2014, Pp243-246

[10] Nam H. Pham,” Complete and Accurate Clone Detection in Graph-based Models”, IEEE, May 16-24, 2009, Pp286-276

[11] Zhengping Liang (2014) ,” A Novel Optimized Path-Based Algorithm for Model Clone Detection” JOURNAL OF SOFTWARE, VOL. 9, NO. 7, JULY 2014,pp 1810-1817

[12] Robert Tairas, “Visualization of Clone Detection Results”

[13] Matthew Stephan , “Identifying Instances of Model Design Patterns and Antipatterns Using Model Clone Detection “

[14] Nicolas Bettenburg , “ An empirical study on inconsistent changes to code clones at the release level “ ELSEVIER ,2010, Pp 1-17

[15] Ginika Mahajan., “Implementing a 3-Way Approach ofClone Detection and Removal using PC DetectorTool”, in Proceedings of IEEE 2014, InternationalConference on Program Comprehension, pp.242-245,

[16] Choi E., “Finding Code Clones for Refactoring with CloneMetrics: A Case Study of Open Source Software”, in Proceedings of The Inst. Of Electronics, Information and Communication Engineers (IEICE) July2011. , pp. 53-57,

[17] Mondal M., “An Empirical Study of the Impactsof Clones in Software Maintenance”, in Proceedings ofIEEE International Conference on Program Comprehension , 2011. , pp. 242-245

[18] Störrle H., “Towards Clone Detection in UML Domain Models”, in Proceedings of the FourthEuropean Conference on Software Architecture:Companion Volume, , 2010., pp. 285-293

[19]Mahajan G., “Software Cloning inExtreme Programming Environment”, InternationalJournal of Research in IT and Management, , February 2012., vol. 2, no.2, pp. 1906-1919

[20]Benjamin Hummel, “Index-Based Model Clone Detection” May 23 2011,Pp 1-7

IJTC.ORG

[21] Eytan Adar: “Visualization and Exploration of CodeClones in Context. In the proceedings of the 29th International Conference on Software Engineering (ICSE'07),, Minneapolis, MN, USA, May 2007 ,Pp.762-766

[22] Dhavleesh Rattan, “Software clone detection: A systematic review” Information and Software Technology 55 , 2013, Pp 1165–1199

[23] B. Baker, “Sparse dynamic programming for longest common subsequence from fragments Journal Algorithms” Vol 42 issue no 2, (2002), Pp. 231–235

[24] Ritu Garg, “ Code Clone v/s Model Clones: Pros and Cons “International Journal of Computer Applications (0975 – 8887) Volume 89 – No 15, March 2014, pp 20-22

[25] Meena Bharti, “Software Cloning and Its Detection Methods” IJCST Vol. 5, Issue 4, Oct - Dec 2014,Pp 191-194

IJTC.ORG