Study of Software Reusability in Software Components

M.H.Arifa Banu [#1], N.Mohamed Thoufeeque[#2],K.Archana[#3]
[# 1]Computer Science & Engineering, School of Computing,
SASTRA University, Tirumalaisamudram, Thanjavur-613401, Tamilnadu, India.
arifabanucse@gmail.com
[# 2]Assistant System Engineer,Tata Consultancy Services,Chennai
mthoufeeqcse@gmail.com
[# 3]Computer Science & Engineering, School of Computing,
SASTRA University, Tirumalaisamudram, Thanjavur-613401, Tamilnadu, India.
karchana.cse@gmail.com

***Abstract*— Software engineering is a disputation for the technical person, as it is difficult for them to produce software which is very cheap, fast functions better. The benefit of Software resembles in the growth of the computer hardware. Component based software system is reuse based method. According to the vast variety of functionality present in software system it accentuates the splitting vexation. Component model is implemented by using the component technology by means of standards and guidelines. Software reuse is a method of reusing the existing software by the incorporation of their functionality. In order to reuse software several techniques and algorithm has been used such as Neural Network, Genetic Algorithm, Power Builder, and Support Vector Machine, Cyclometric complexity, Fuzzy logic, Hybrid genetic algorithm and fuzzy logic. This paper concerns the study of some of the techniques, design, approaches involved in software reuse.**

Keywords: Support Vector Machine, Neural Network, Genetic Algorithm, Fuzzy.

I. INTRODUCTION

Reuse of software is a melioration crusade of the fecundity of the software systems. By the reusability the component can have better qualified, cheaper cost, improved performance. The reusable software component works better than the existing software as they are created with overcoming of the existing software module. The reusable software components are delivered within the shorter time. The major advantage of software reuse is it reduces the development time and cost, increases the software productivity, improves system interoperability, Provide more standardized system. Software reuse solves the several software crisis. In order to reuse a software system component technology is the important technology. Components are nothing but the smaller module which consist of classes and services which is defined in an application software system. Component form goes through the different form such as development, package, deployment and execution. Component assessment consists of various steps as find the component, verify the component, and finally store the component in the repository.

Components are betrothed for reuse. Component Based software systems cogitate in decay of software system into function and logical component into distinct interface. This system hence the better maintainability of the software component. McClure has provided the technique to build the reusable component such as generalize, Standardize, Automate, certify, document. The approaches to evaluate the software reliability are qualitative, empirical relation of the component is described. As substantial manual effort is needed for the quantitative approach. This method is based on the attachment of prejudiced value. The empirical approach is based automatic collection of the objective data. Based on the following sections the section II concern with the CBS development process section III with approaches of reuse the section IV with design of the reuse system, section V with techniques and followed by conclusion in section.

II. CBS DEVELOPMENT PROCESS

Component Based Software Development process the software organization is split up into several categories based on the role: component developer, tool provider, environment provider component broker. The component developer who is responsible for dealing with the business logic. Developer requests the component from component repository the component broker in the middle help to select the appropriate component from the repository. Component broker evaluate and procure the component from the developer. After the selection of the component the tool provider which is nothing but the human or organization responsible for the creation, development, assembling and packaging tool .design develop the component with the tool. Then the environment provider which is an operating system, database, application or web server provides the environment to implement the tooled component. Then the role of software developer is to collect the source

from the component developer, broker, tool and environment provider and foregather them in order to meet the customer satisfaction. The above process is explained in the figure 2.1

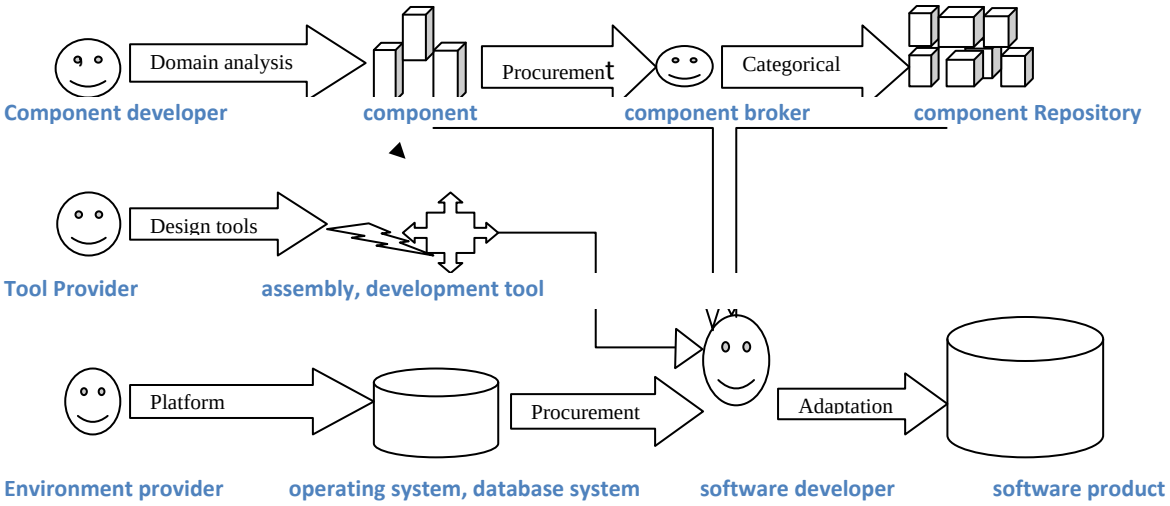

Figure 2.1 CBS development process

The main aim of the software reuse is to exploits the software crisis, as this crisis is responsible for the major failure of the software. The crisis of the reuse is given below:

- Project run over bridge
- Project run over time
- Software inefficiency
- Software low quality
- Software does not meet the requirement
- Project is unmanageable

III. REUSE APPROACHES

Code reuse is extracting the partial or complete written programming code from the application software system and reuses it to write a new code in order to adopt in the existing application software system. Reusing limits the development time and reduces the energy to develop the code by reusing the programming code it is a vernacular approach. Software reuse consists of two approaches: Development of code from abrasion. Extraction of reusable code from the exiting code.

3.1 Reuse type

In order to reuse a software component first thing pauperism to engender identifies the reusable element in the application software. As that reusable element forms the library by this the software development efficiency can be meliorated. Software reuse can be done in two ways such as direct and inheritance reuse.

3.1.1.Direct Reuse

Selection of reusable element from the reusable library and by declare instance of object type to the element, the element can be directly reused.

3.1.2 Inheritance Reuse

Inheritance reuse is based on the object oriented property. Creating the new ones from the existing ones. New object type is derived from the existing object. As this to make a sense of same object to work with the specific operation by using the instance variable in the new object type.

IV. REUSE DESIGN

Reuse design wells with four major processes such as Abstraction, Selection, Instantiation and Integration.

4.1.1 Abstraction

Abstraction is a vital technology in software reuse. The probability of abstractions describes about the probability of reusability. Higher the level of abstraction higher the level of reusability. Domain analysis method is used to identify the reusable part of the application in abstraction. Management information system has to communicate with other external systems to exchange data.

4.1.2 Selection

Selection process is also knows classification; retrieval and descriptions. After abstraction the selection is by searching and comparing the abstracted component with the component in the library so as to make a note that it satisfies the requirement of the application.

4.1.3 Instantiation

Instantiation is the process of direct reuse. It is a process of converting, constraining, passing parameters to the reusable object. As it satisfies it has been converted into executable code.

4.1.4 Integration

Integration is an optimal step. By the use of software framework it builds the software systems with the reusable components. Integration is engaged with the reusable discipline.

4.2 Reuse indefinite quantity

Reuse is a manifestation of compounding elucidation to the crisis based on predefined elucidation to the sub crisis. For the preparation of one stage to another stage reuse activity undergoes several indefinite quantity. The indefinite quantities includes designing the plan, arriving the solution, and analyzing the solution, acquiring the predefine components and finally transformation from component to product. Then verify it, as it satisfies the customer needs. The following chart explains the quantity:

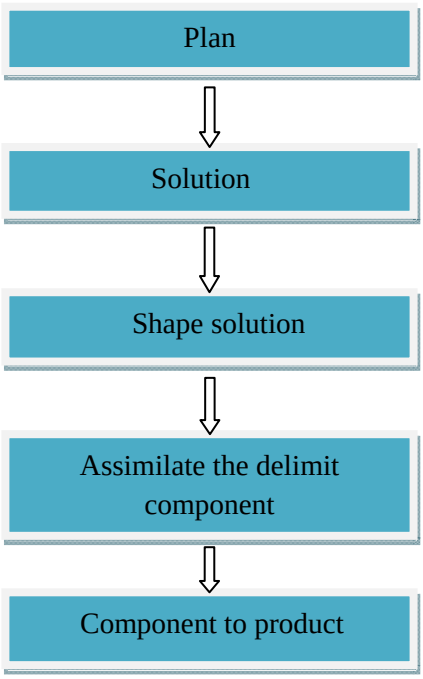

Figure 4.2 explain the reuse indefinite quantity

V. TECHNIQUES OF SOFTWARE REUSE

In order to reuse a software system several techniques has been used. The technique helps to identify the reusable component and consider the various parameters and metrics involved in the reusability.

5.1 Support Vector Machine

Support vector machine (SVM) is generally a classifier model that forges the assortment task. For reusability method SVM based classifier is more reclaimable. Data classification, support vector machine is the very effective technique. Data instance is involved in the training and testing data which is the more bases for the SVM classifier. The data instance consists of several attributes and one target value for each training unit. The aim of Support Vector Machine is to exhibit a model that venture the data instance target value in the testing set. Both regression and classification is foster with the support vector machine. Multiple continuous and categorical variables can be appendage in SVM. Dummy variable is generated for the categorical data with 0 or 1 case value.

In order to minimize the error function SVM pertain the iterative training algorithm for the purpose of erection of hyper plane.SVM model has been crack into four different group based on the error function

(a) The C-SVM classification
(b) The nu-SVM classification

(c) The epsilon-SVM regression
(d) The nu-SVM regression

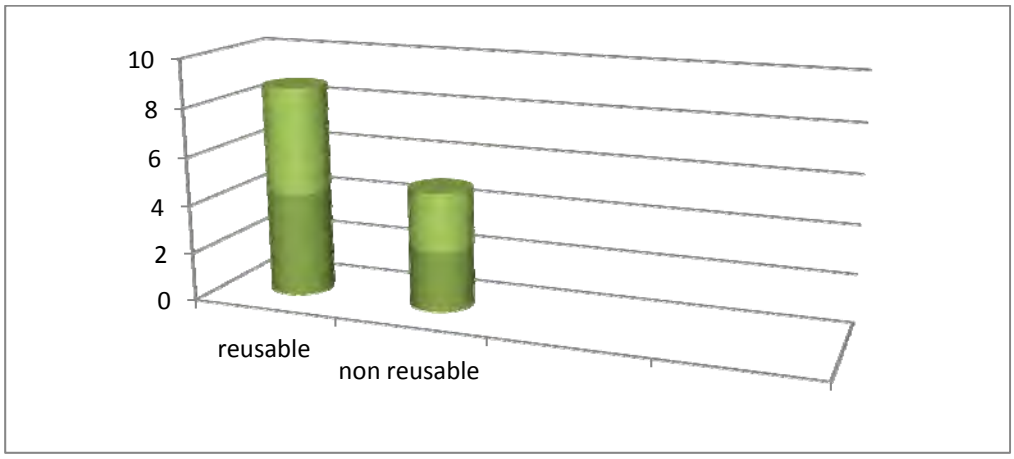

Figure 5.1describes the count of reusability output attribute in the dataset.

5.2 Neural Network

In order to design and evaluate the neural system it has to undergo several neural network algorithms such as (a) Batch Gradient descent (b) Batch Gradient Descent with momentum (c) Variable Learning Rate (d) Variable Learning Rate training with momentum (e) Resilient Back Propagation. In neural network the training dataset are trained with different neural network. Based on different comparison technique such as Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) the trained neural networks are cerebrating against the testing dataset. Neural network are easy to use but it produce inadequate result sometimes.

5.2.1 Mean Absolute Error (MAE)

Mean Absolute Error (MAE) is an average predicted value. It is a measure of different between the average predicted and actual value in all test cases. The equation to solve the MAE is given as

$$\mathbf{MAE = |a1\text{-}o1|+|a2\text{-}o2| +-------+|an\text{-}on| \mid}$$

5.2.2 Root Mean Square Error (RMSE)

Root Mean Square Error is the frequent measure of different between the predicted value and the absolute value predicted or observed by the model or estimator. The final decision is based on the result predicted at last session. The equation to calculate the RMSE is given as,

$$\mathbf{RMSE = \mid a1\text{-}o1|+|a2\text{-}02|+-------+|an\text{-}on| \mid n}$$

5.3 Genetic Algorithm

Genetic algorithm is hired to puzzle out the optimization problem because it has a capacity to provide good performance in combinatorial explosion. To solve the large instance of optimized problem customized genetic algorithm is used. Based on the problem attribute Rc- position based representation is created which is known as special encoding scheme for genetic algorithm. By this encoding, each chromosome of the given gene provides the integer number that specifies the index of AC in order to specify the selected RC. The technique involved in genetic algorithm is two namely crossover and mutation.

5.3.1 Crossover

Crossover is the process of interchanging the gene this is done by randomly selecting the point meant as crossover points then swapping their parent's gene within one another. If crossover is occurred between two separate application verification need not be done in this case. If it occurs inside the two different place of the application it needs to be verified so as the check whether the same available component is present in both the place.

5.3.2 Mutation

As of crossover strategy mutation can also appear at any orientation of the chromosomes. The mutation index needs to be checked off that it hasn't adapted from the same available component after the operation is successful.

2	10	6	3	4	8	5	7	4	3
a	b	c	d	e	f	g	h	i	j

Figure 5.3 Representation of chromosomes in genetic algorithm

5.4 Fuzzy Logic

Fuzzy logic is to solve the large problem based on the fuzzy set theory. Fuzzy is based on the truth value/Boolean value such as 0 and 1/true or false, member ship values. Estimation of reusability component can be done using a fuzzy logic. Fuzzy logic has several advances than that of the neural network; the major highlight is it works without data. Fuzzy logic helps in best inclination of reuse. Fuzzy takes the input as several factors like customization, interaction complexity, understandability, portability and produce the reusability as the output.

5.4.1 Fuzzy Logic Approach

Fuzzy works with several approaches such as fuzzy based approach, Neural fuzzy based approaches, and Hybrid fuzzy-GA based approach

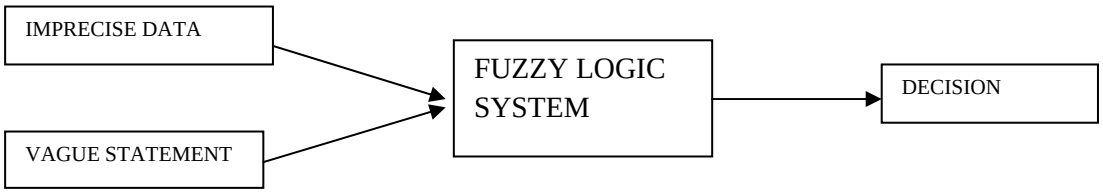

Figure 5.4 explains the Fuzzy logic system

5.4.1.1 Fuzzy based approach

The fuzzy based approach works by first identifying the membership function. After the identification an optimized rule base for initial process is selected for the fuzzy interference. Then the measure of the fuzzy interference is done. Finally the 100% accurate result is attained.

5.4.1.2 Neural fuzzy based approach

The neural fuzzy based approach involves the following process. The fuzzy interference is created as that of the fuzzy based approach. Then the trained data of neural fuzzy system is added with the training set then the hybrid learning algorithm uses both the least square method and back propagation to perform the training NF based system. Least Square is used to picks up the consequent parameter in forward passes and back propagation picks up the premise parameter in backward passes. Finally with the testing data trained neural fuzzy system is examined and then the result is attained in % accuracy.

5.4.1.3 Hybrid Fuzzy-GA based approach

The Hybrid fuzzy-GA based approach works as if it is a combination of both fuzzy system and genetic algorithm. This process follows as, first it takes metric value as the input then based on the Euclidean distance is catch the nearest matching value. The output for the fuzzy interference is obtained by the gathered input set. Up to this the fuzzy system is over then the genetic algorithm part starts as the selected nearest value and fuzzy interference value is considered as a chromosomes then these two values are multiplied with a 100 in order to convert it into binary value. Finally the crossover point is selected so as to crossover the two value and compare the result.

5.4.1.4 Defuzzification

The final step of fuzzy is defuzzification. In the fuzzy system based on the knowledge base the membership function is reckoned on the fuzzy domain by the inference engine provided by the domain expert then the final step of changing the fuzzy value into single numerical value is known as defuzzification, in case of reusability the defuzification provide the appropriate vale of reusable component according to the input set. This helps in estimating the quantity and development of the software component.

VI. CONCLUSION

The software component is the building block of composition with their own interfaces. This component can be get accessed any where through the interface. The internal configurations of the components are hidden from the user. The component can be only be reused by the organization to which it belongs, through the internet and other web source the object code of the component is available it can be reused in any environment through the retrieval. Reusability is the major characteristic of the component. The component that as the higher key of reusability that can be reused very faster with better quality and less cost of development. The higher the level of abstraction greater is the degree of reusability. As the reuse the best way to overcome the prone and cones of the software system that exist already. More over it saves the time the maintenance of the software is more flexible and feasible that the one created new. So the software reuse in more useful and upgrading technique in recent year. This paper concerns the few techniques involved in the software reuse process in the software component.

VII. REFERENCES

[1] Ajay Kumar,"Measuring of Software Reusability Using SVM based Classifier Approach", International Journal of Information Technology and Knowledge Management, Volume 5, No.1, pp 205-209 ,January – June 2012.

[2] C.K.Kwong, L.F Mu, J.F. Tang, X.G. Luo ,"Optimization of Software Component Selection for Component –Based Software System Development", Computer and Industrial Engineering, pp 618-624,2010.

[3] Anupama Kaur, Himanshu Monga, Mnupreet Kaur, "Performance Evaluation of Reusable Software Component", International Journal of Emerging Technology and Advanced Engineering, Volume 2, Issue 4, April 2012.

[4] M.Mustafa Rafique, Ali R.Butt, Eli Tlivich, "Reusable Software Components for accelerator based cluster" Journal of Systems and Software, pp 1071-1081, 2011

[5] J.F. Tang, L.F Mu,C.K.Kwong, X.G. Luo, "An Optimization Model for Software Component Selection under Multiple Applications development", European Journal of Operational Research, pp 301-311, 2011

[6] Zhenjun Tang, Xianquan Zhang, Liyan Huang, "Design and Implementation of Reusable Software Components using Power builders", Sciverse Science Direct, pp 584-588,2012

[7] Anupama Kaur, Himanshu Monga,Mnupreet Kaur, Pravinder,S.Sandhu, "Identification and Performance Evaluation of Reusable Software Components Based Neural Network", Volume 1, No.2, March 2012

[8] Hironori Washizaki, Hirokazu Yamamoto, Yoshiaki Fukazawa, "Metrics Suited for Measuring Reusability of the Software Components".

[9] Kevin Kum Fung Yuen, Hendry C.W.Lau, "A Fuzzy group analytical hierarchy process approach for software quality assurance management: Fuzzy logarithmic least squares method", Expert System with Application, pp 10292-10302, 2011

[10] Sonia Manhas, Rajeev Vashisht, Pravinder S.Sandhu, Nirvair Neeru, "Reusability Evaluation Model for Procedure based Software Systems", International Journal of Computer and Electrical Engineering, Volume 2,December,2010

[11] Parvinder Singh Sandhu, Dalwinder Singh Salaria and Hardeep singh,"A comparative Analysis of Fuzzy, Neuro Fuzzy, Fuzzy-GA based Approaches for Software Reusability Evaluation", World academy of science engineering and technology, 2008.

[12] Vidhu Bharadwaj, "Estimating Reusability of Software Component using Fuzzy Logic",2010.