EVALUATING IMPACT OF COMPONENT REUSABILITY WITH NEW HIERARCHICAL COST ESTIMATION MODEL

Parul Gandhi  
Department of Computer Science & Business Administration  
Manav Rachna International University  
Faridabad, 121001, India  
gandhi2110@gmail.com

Pradeep Kumar Bhatia  
Department of Computer Science & Engineering  
G. J. University of Science and Technology  
Hisar, 125001, India  
pk_bhatia2002@yahoo.com

ABSTRACT

Software development cost estimation has been the subject of vigorous study over the past few decades. Software reusability provides opportunity for reducing the overall development cost and time of the software and improving the overall quality. Although significant progress has been made in the areas of reusability. The growing trend towards using CBSE approach to include COTS component has taken the original concept of reuse into a completely different arena. The use of Commercial-Off-The-Shelf, software components is increasing in today’s software development environment. Component-based reuse is widely accepted as an important reuse strategy and component-based reuse programs heavily depend on software reuse repositories for achieving success [1] [2] [3]. Present paper discusses impact of reusability in relation to overall development cost of quality based software. The main objective is to measure the relation of reused and developed software, the cost for obtaining reuse and the cost avoided by reusing software during development and maintenance. In our research we propose a Hierarchical Cost Estimation Model, to evaluate to what extent reusable components lead to increased savings in term of cost and quality.

Keywords: Reusability, CBSE, Hierarchical Cost Estimation Model, Quality
1. INTRODUCTION

Software reuse is only relevant when it has positive economical impacts in organizations and cost estimation models are fundamental in assessing these impacts. Component-based reuse is widely accepted as an important reuse strategy and component-based reuse programs heavily depend on software reuse repositories for achieving success [1] [5] [3], however the main focus on the reuse repository area is on classification and retrieval problems. Existing repositories have their efficiency measured by a sum of features [7], but lack mechanisms to automate the task of collecting and maintaining trace links between reused components and their clients. There are various ways to achieve reusability. Many researchers have specified the benefits of reuse in their reports[15,16]. Different forms and extents of reuse are observed in the software development process, including ad-hoc reuse, planned reuse, systematic reuse, inter-organizational reuse.

Developing software from scratch is the most complex part of the process. Reusability reduces this need and hence allows a greater focus on quality. Investments in information technology by U.S. businesses continue to spiral, with estimates topping $400 billion for 1997 [8]. The software development process in many organizations has been associated with cost and quality. There should be some Economic models quantify the costs and benefits of reusability. Several authors have modified the cost models that are today used to estimate time and effort and for the development both of components and of applications using components [9][10][11]. In this paper we discuss the use of CBSE approach to develop components and also propose the Hierarchical Cost Estimation Model that calculate the cost to make component reusable and savings in avoided work which will eventually reduced the overall cost of software development and increase the product quality.

This paper is organized as follows. In the next section, we discuss various existing economic oriented model developed by various researchers. In section 3 we propose Hierarchical Cost Estimation Model to evaluate component reusability. In section 4 we discuss the relationship between overall development cost and reusability and in section 5 we draw some conclusions and give directions for future work.

2. EXISTING ECONOMICS MODELS

Several Cost Estimation models have been presented in the literature. The Balda and Gustafson (1990) model allows estimating the software project cost with reuse based on the COCOMO Model [13]. Existing models can only address limited-scope questions; each focuses only on a particular aspect of reuse [12]. The Reuse-Level Metrics models are concerned with how much reuse is achieved and the Economic Models answer how many projects must be developed to reach a break-even point after the introduction of a reuse program. The literature has not presented a comprehensive model of reuse that includes all elements of the reuse process and their interaction with one another.

In the literature, there is ample anecdotal evidence that describes reuse success stories [4, 6, 17, and 18]. Existing models can help companies to assess the success of their reuse programs. They might even help to improve reuse methods that are in place. However, they don’t tell organizations ahead of time how to best implement reuse into the development process. The adoption of a reuse component would be less costly and more successful as compare to the development of component from scratch, if researchers could provide a guideline to potential adopters describing how reuse is implemented best.
3. HEIRARCHICAL COST ESTIMATION MODEL

Developing the system by reusing the components made for earlier products is the promising way to save valuable development time and cost. A component is built to be reused and reusability implies generality and flexibility. Models for software reuse economics try to help us answer the question, “when is it worthwhile to incorporate reusable components into a development and when is custom development without reuse preferable?”[14]. Software developed by the means of component should require less cost than traditional development of software from the scratch and will improve the functionality and quality of overall software. When developing the software with the help of CBSE approach, estimation of cost become bit more complicated. Therefore we proposed Hierarchical Cost Estimation Model to evaluate the benefits of component reusability which eventually reduced the overall cost of software development. The main component of proposed cost estimation model is Reusable Component Repository (RCR) which is used to store the reusable components. Proposed model categorize the Actual software development cost into Search cost, Reusability cost with or without modification and Development cost.

3.1 Search Cost

The main idea of the component based approach is building systems from pre-existing components which requires search for the component from Reusable Component Repository (RCR) to check the availability of component required for the overall development of the software. Therefore search cost is the cost of performing a search operation of the required component from Reusable Component Repository (RCR). Search for a component for reusability is illustrated in figure1.

![Figure 1. Cost to search Component from RCR](image1)

3.2 Development Cost

Development cost is the cost of developing the software component from scratch. Development cost assuming no reuse at all that means the required component is not available in Reusable Component Repository (RCR). Development cost is directly proportional to the actual cost of overall software development. Figure 2 shows the contribution of development cost in overall development of the software.

![Figure 2. Cost to develop the component from scratch](image2)
3.3 Reusability Cost
We can Categorize the component reusability cost as follows:

3.3.1. Component Reuse with Modification
This cost occurs when the component required is available in RCR but not exactly match the specified requirement. We first Check components according to the requirement, analysis, and design from RCR, If components specifications match 70 to 80% according to software analysis, specifications and design then we start development using reusable components with some modifications. This cost have a great contribution in reducing the overall cost of the software. Figure 3 indicates the process of developing the software with reusable component after some modification to reduce the overall cost of the software.

![Figure 3: Cost to Reuse Component after Modification](image)

3.3.2. Component Reuse without Modification
Development of software without modification in reusable component plays a very important role in improving the quality and reducing the actual cost of developing the software. It includes following phases, namely, domain analysis and specification, domain engineering, frame working. Figure 4 indicates the process of developing the software with reusable component without modification. This cost also has the great contribution in the actual cost required for the overall development of the software.

![Figure 4: Cost to Reuse Component without Modification](image)

3.4 Actual Cost
Actual cost depends on the percentage of components reused and percentage of components not reused to develop the overall software. Figure 5 indicates the contribution of all the costs in overall software development process.

Eg if the software need to be build with n number of modules then Actual cost can be calculated as:

$$ Acost = Scost + Dcost(n-i) + RMcost(i-m) + RWMcost(m) $$

529
Where
n: Number of modules of the required software.
I: Number of modules available in RCR but require some modification
m: Number of modules available in RCR and can be used without modification
Acost: Actual cost to develop the software.
Scost: Cost required to search the component from RCR
Dcost:: Cost required to develop the component from scratch
RMcost: Cost required to modify the component available in RCR.
RWMcost: Cost required to Adapt the available component without any modification.

4. DISCUSSION
The effectiveness of the model is largely determined by its robustness and its reasonableness. All the costs discussed above plays a very important role to calculate the overall development costs of the software. Fig 6,7 and 8 indicates the relationship between all the costs required to calculate the actual costs. The more reusable components available reduced the actual cost and hence improve the overall quality as the developer can more concentrate on the other parts of the software. Thus the actual cost is inversely proportional to reusability cost and directly proportional to development cost.

**Acost 1/\alpha Rcost**

![Figure 6 Relationship between Actual Cost and Reusability Cost](image)

**Acost \alpha Dcost**

![Figure 7 Relationship between Actual Cost and Development Cost](image)
5. CONCLUSION

Software reuse is only relevant when it has positive economical impacts in organizations. Component-based software engineering is successful if the reusable and certified component available for development of CBSS and decreases the overall cost as well as improve the quality as compared to traditional software development. Cost and quality plays a very important role at every phase in developing a software and if cost can be decreased, a lot can be gained. The broader context of discussion is the cost associated with the different stages of software development. It is found that reuse density (the ratio of number parts reused to the size of the system) play an important role in estimating cost savings due to reuse. The proposed Hierarchical cost estimation Model provides a framework to estimate the effect of component reusability in the overall development process.

REFERENCES


