

## International Journal of Computer Science and Mobile Computing



A Monthly Journal of Computer Science and Information Technology

ISSN 2320-088X

*IJCSMC, Vol. 3, Issue. 4, April 2014, pg.1116 – 1121*

### **REVIEW ARTICLE**

# **A Review Paper on Effort Estimation and Model Based Regression Testing with SOA**

**Yogesh Bhardwaj<sup>1</sup>, Dr. Manju Kaushik<sup>2</sup>**

<sup>1</sup>(M.Tech.) Department of CSE & JECRC University, Jaipur, India

<sup>2</sup>(Associate Professor) Department of CSE & JECRC University, Jaipur, India

<sup>1</sup>yogesh.bhardwaj12345@gmail.com; <sup>2</sup>manju.kaushik@jecrcu.edu.in

---

*Abstract— Regression testing is expensive and essential part of an effective testing process, for achieving quality of the software and for gaining confidence in modified software. This is a very expensive activity work in maintenance phase. In regression testing model based regression testing used for modification in models. According to model based regression testing the scalability problem in code based technique in unit testing is removed. Many researchers are working on model based testing approaches. The next era of model based regression testing used in Service Oriented Architecture. A Service Oriented Architecture is an application architecture in which services are described using a description language. So in this paper a brief review given for SOA in model based regression testing. To enhance the functionality we give approach of effort estimation of Web application in SOA in model based regression testing.*

*Keywords— Testing; Regression Testing; Effort Estimation; Model Based Regression Testing; Service Oriented Architecture*

---

## **I. INTRODUCTION**

Testing is a process to find out errors or bugs in a software. Software testing is the process of evaluation a software item to detect differences between given input and expected output. Also to assess the feature of a software item. Testing assesses the quality of the product. Software testing is a process that should be done during the development process. In other words software testing is a verification and validation process. There are many models are used for developing the software. Every model has same levels of development like feasibility study, requirement analysis, design, coding and implementation, testing and maintenance etc. Different types of testing used for find out the quality of software. Regression testing work in a maintenance phase. Regression testing is a more effective and essential testing for ensuring software quality. The main motive behind regression testing to test the new version of software or system to verify that after modifications the functionalities and attribute before changes or before modification have not been affected by the new changes. [1] So the new changes have not introduced new bugs is the main motive of regression testing. So this is the method or criteria of verification of the functionality, verification of attributes, and verification of newly added features that have not created in problem in previous working version of software. Regression testing is an important activity to gain confidence in re-used components. **The purpose of regression testing** is to confirm that a recent program or code change has not adversely affected existing features. Regression testing can be applied in various ways code based, specification-based and model-based. Code-based techniques are

white-box method that is they select test cases based on the difference between original and modified code. It uses relationships between code parts and test cases that traverse them to locate test cases for retest when code is modified. An important issue with unit-testing is scalability problem. As software systems grow in size and complexity, so does the need for higher level models and abstractions in their development. Model centric development creates opportunities to drive regression testing processes at higher abstraction levels. A model-based technique is a black-box method. It selects test cases based on model modification, so it uses relationships between model elements and test cases that traverse those elements to locate test cases for retest.

**Need of Regression testing:-** Regression Testing is required when there is a Change in requirements and code is modified according to the requirement.  
 New feature is added to the software.  
 Defect fixing and Performance issue fix.

Regression testing in Service Oriented Architecture [2] is one of the major matters of concern because of the dynamic nature of the Service binding and the adaptability to accommodate the changed requirement. Service Oriented Architecture [3] provides a very flexible and controlled architecture that uses web service technologies to redefine functions within all underlying applications as independent services. A very common technology for Service Oriented Architecture implementation is web services. Effort estimation [4] in regression testing is done mathematically and practically. The amount of effort in either person days or person hours necessary for conducting tests.

There are many papers are published regarding regression testing and Service Oriented Architecture but the main motive behind this paper presents the combination of regression testing with Service Oriented Architecture. This paper present a reduce effort conducting in Service Oriented Architecture based application in regression testing using tool.

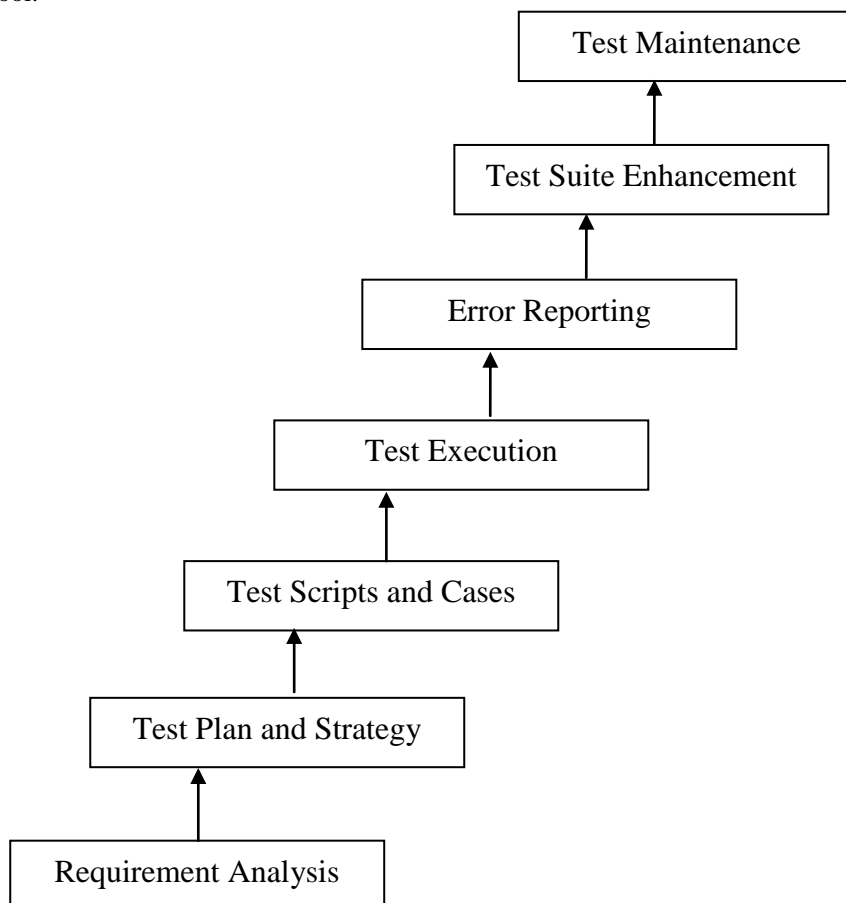


Fig. 1:- Flow Diagram of the Regression Testing

This paper is structured as follow. In section 2 for work closely related to us. Section 3 discusses model based regression testing with SOA that is available in literature. Section 4 gives overview of effort estimation. Section 5 for result and Conclusion and future work are summarized in section 6.

## II. RELATED WORK

SOA based systems are form of software and so they should be tested. Numerous research works have been presented related to web service regression testing; a class of SOA based applications [5]. Service-Oriented Architecture [3] (SOA) is a booming computing paradigm that utilizes services as fundamental elements for developing software applications/solutions. The concept of a service is proposed as a self-describing, platform-independent computational element that performs functions ranging from simple requests to complicated business processes and supports rapid, low-cost composition of distributed systems. In practice, the implementation of the service concept is Web services that allow organizations to expose their reusable business process fragments over the Internet using open standards and protocols. A regression testing approach for composite web services is proposed [6], which makes it possible for the tester to locate the fault in the respective service, ultimately differentiating test data from test behaviour. Several researchers publish their research paper on effort estimation in SOA based services. In which there are many techniques are used for estimating effort they are mathematically techniques like Work Flow Model [7] in which estimation of effort is done by function point analysis for individual service. A Work Flow algorithm is used for effort estimation. A framework also developed for Costing Service-Oriented Architecture (SOA) Projects Using Work Breakdown Structure (WBS) Approach [8]. But there is some limitation in Functional Point method. This method is not designed (yet) to estimate mathematically-intensive software such as: expert systems, simulation software, forecast software, Artificial Intelligence, etc. It also has limitations on measuring very small pieces of software or which processes continuous variables such as audio sounds or video images, such as, for instance, in computer game software, musical instruments, etc. Local extension may be defined for this kind of functionality and also to overcome some other limitation with aspects not taken into account like the influence of complexity (however defined) that contribute to software 'size'. Another approach of effort estimation using Classification matrix [9] is also defined. Model based regression testing [10] is a technique of regression testing the others techniques are code based, risk based, state based etc. in model based regression testing the classes are state machine based, activity diagram based, model checking based and hybrid approaches. Model based regression testing [11] is applied when the code is unavailable. In model-centric approaches, modifications are first done to models rather than code. In model based testing test cases are selected for modified parts of software. UML classes and sequence diagrams are used to implement model based regression testing.

## III. MODEL BASED REGRESSION TESTING WITH SOA

- A. Model based techniques:** - As software system grow in size and complexity then there is no code based technique is used due to unit testing scalability problem. It is a black box technique in this it selects test cases based on model modification, so it uses relationships between model elements and test cases that traverse those elements to locate test cases for retest.

**Advantages:-**

- ✓ Traceability - It is easier to maintain traceability between the design artifacts and the test cases than maintaining traceability between code and the test cases. It is also easier to identify changes between across different versions of design artifacts as compared to analyzing changes across code versions.
- ✓ Scalability - Code-based regression testing becomes very expensive when applied to large programs. A model being a simplified representation of a code, model-based testing is comparatively more efficient.
- ✓ Language independence - Different parts of software may be developed using different programming languages.

**Disadvantages:-**

- ✓ Change Identification
- ✓ Change Propagation
- ✓ Baseline test suits generation
- ✓ Validity of test cases
- ✓ Test automation

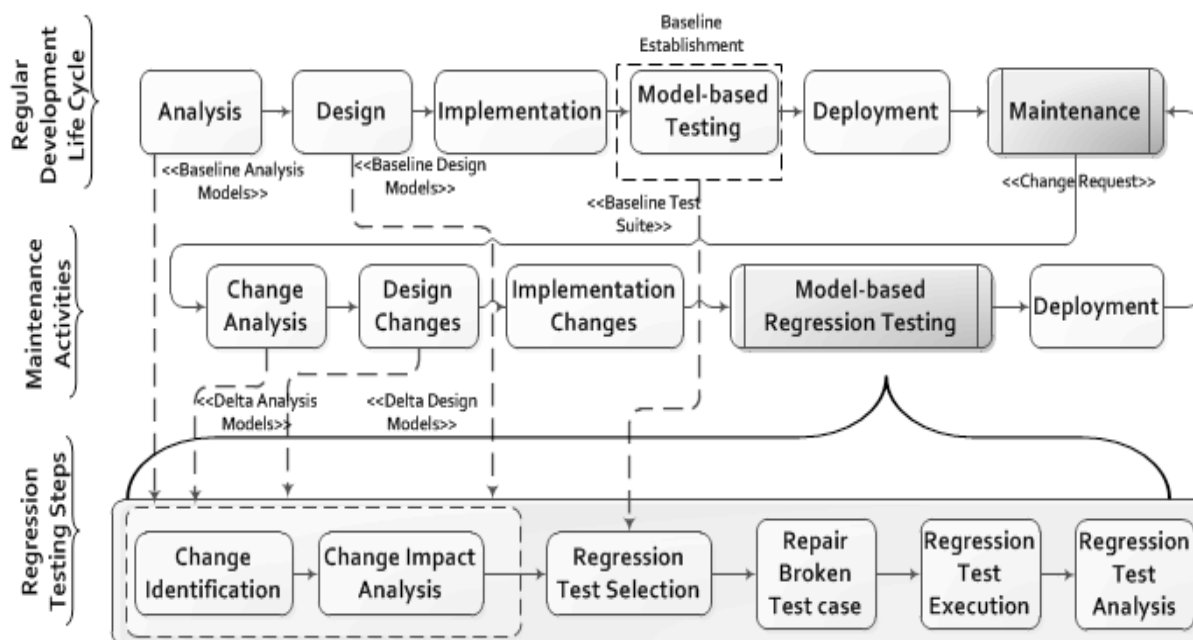


Fig. 2:- Model Based Regression Testing

According to Figure 2, a software system is constructed using the steps of any regular software development life cycle. When a change request is triggered probably due to a changed requirement, the maintenance activities are performed to entertain it. Once this new requirement is implemented, the system should be tested to detect the faults introduced by the changes. Model-based regression testing is used in this phase to test the changed software system.

**B. Service Oriented Architecture (SOA):-** Regression Testing in Service Oriented Architecture is one of the major matters of concern because of the dynamic nature of the Service binding and the adaptability to accommodate the changed requirement. The actual configuration of the service is known only at run time, so It becomes very complex to verify whether the changes made in earlier version of the system are correct or not and it does not affect the functionality and performance of the existing system. Regression testing is essential for ensuring software quality. It is the process of validating modified software to provide confidence that the changed parts of the software behave as intended and that the unchanged parts of the software have not been adversely affected by the modification. Regression testing help identify changes between a selected product release and a previous release of the product called a baseline. A baseline is recorded snapshot of desirable product behaviour. This expected behavior is then used to ensure that nothing has been broken in the system as a result of changes introduced in a program module.

#### IV. OVERVIEW OF EFFORT ESTIMATION

- A. **Definition of Test Estimation:** - Test Estimation is the estimation of the testing size, testing effort, testing cost and testing schedule for a specified software testing project in a specified environment using defined methods, tools and techniques.
- 1) **Testing Size** – the amount (quantity) of testing that needs to be carried out. Sometimes this may not be estimated especially in Embedded Testing (that is, testing is embedded in the software development activity itself) and in cases where it is not necessary
  - 2) **Testing Effort** – the amount of effort in either person days or person hours necessary for conducting the tests
  - 3) **Testing Cost** – the expenses necessary for testing, including the expense towards human effort
  - 4) **Testing Schedule** – the duration in calendar days or months that is necessary for conducting the tests

**B. Approaches to Test Effort estimation**

Now the following approaches are available for carrying out Test Effort Estimation

- a) **Delphi Technique:** - Under this method of software estimation, the project specifications would be given to a few experts and their opinion taken. The actual number of experts chosen would depend on their availability. A minimum of three is normally selected to have a range of values.
  - b) **Analogy Based estimation:** - Analogy Based Software Estimation is based on the principle that actual values achieved within the organization in an earlier and similar project are better indicators and predict the future project performance much better than an estimate developed a fresh from scratch. It also facilitates bringing the organizational experience to bear on the new projects.
  - c) **Software Size Based Estimation:** - By the time a testing project is in its initiation phase, software size would have been available. Now we adopt this software size as the testing project size. Then we assign a Productivity figure (rate of achievement) for the software size to arrive at the required effort to execute the testing project.
  - d) **Test Case Enumeration Based Estimation:** - The following steps describe this technique –
    - 1) Enumerate the test cases – list down all the test cases
    - 2) Estimate testing effort required for each test case – use person hours or person days - consistently
    - 3) Use Best Case, Normal Case and Worst Case scenarios for estimating effort needed for each test case
    - 4) Compute Expected Effort for each case using Beta Distribution  
$$\text{Best Case} + \text{Worst Case} + (4 * \text{Normal Case}) / 6$$
- Sum up the –**
1. **Expected times** to get Expected effort estimate for the project
  2. **Best-Case** times to obtain best-case effort estimate
  3. **Worst-Case** times to obtain worst-case effort estimate
  4. **Normal-Case** times to obtain normal-case effort estimate

## V. RESULT & DISCUSSION

This paper gives a brief overview of Regression testing and its subclass Model based regression testing. We also give some idea of Service Oriented Architecture (SOA) and how it is closely related to regression testing. Estimation of effort and its impact on testing also discussed to relate with software development.

## VI. CONCLUSION & FUTURE SCOPE

Regression testing in Service Oriented Architecture is one of the important testing activities which require a lot of effort to test the system because of the dynamic nature of the system. In this paper we present overview of MBRT and SOA with effort estimation. By reviewing this estimated effort is compared to the traditional estimation techniques and explore more ideas regarding estimation of cost and size of system. In future we intend to focus on reduction of effort of business application and others. We would like to work on web services of SOA and explore the new method to reduce further effort and cost of system using an approach or method. We would also like to combine all these terms and use them for an application or a case study.

## REFERENCES

- [1] Mr. Rohit N. Devikar, Prof. Manjushree.D.Laddha, "Automation of Model-based Regression Testing", *International Journal of Scientific and Research Publications*, **2(12)**, Pages 121-125, (2012).
- [2] Prachet Bhuyan, Chandra Prakash Kashyap, Durga Prasad Mohapatra, "A Survey of Regression Testing in SOA", *International Journal of Computer Applications*, **44(19)**, Pages 22-25, (2012).
- [3] Prachet Bhuyan, Abhishek Kumar, "Model Based Regression Testing Approach of Service Oriented Architecture (SOA) Based Application: A Case Study", *International Journal of Computer Science and Informatics*, **3(2)**, Pages 11-16, (2013).
- [4] Murali K. Chemuturi, "Test Effort Estimation", (Emerge), *Software Estimation Best Practices, Tools, & Techniques*, J Ross India, 197, (2009).
- [5] Rajani kanata Mohanty, Binod Kumar Pattanayak and Durga Prasad Mohapatra, " UML based web service regression testing using test cases: A Case Study". *ARPN journal of engineering and applied sciences*, **7(11)**, Pages 1416-1423, (2012).
- [6] Bo Yang, Ji Wu, Chao Liu, Luo Xu, "A Regression Testing Method for Composite Web Service", *International Conference on Biomedical Engineering and Computer Science (ICBECS)*, Wuhan, 1(2010).
- [7] Khalid Mahmood, M. Manzoor Ilahi, Shakeel Ahmad, Bashir Ahmad, "Integration Efforts Estimation in Service Oriented Architecture (SOA) Applications", *International Journal on Information and Knowledge Management*, **1(2)**, Pages 23-27, (2011).

- [8] Yusuf Lateef Oladimeji, Olusegun Folorunso, Akinwale Adio Taofeek, Adejumobi, A. I, “A Framework for Costing Service-Oriented Architecture (SOA) Projects Using Work Breakdown Structure (WBS) Approach”, *Global Journal of Computer Science and Technology*, **11(15)**, Pages 36-48, (2011).
- [9] Zheng Li, Liam O’Brien, “Towards Effort Estimation for Web Service Compositions using Classification Matrix”, *International Journal on Advances in Internet Technology*, **3(3, 4)**, Pages 245-260, (2010).
- [10] Bharat Choudhary, Vineet Ricchariya, “An Approach of Regression testing of Service Oriented Architecture”, *International Journal of Research in Computer & Communication Technology*, **2(7)**, Pages 352-356, (2013).
- [11] Sabah Tamimi, Muhammad Zahoor, “Analysis of Model Based Regression Testing Approaches”, *10th WSEAS international conference on communications, electrical & computer engineering*, USA, 65(2011).