



GLOBAL JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY

Volume 11 Issue 3 Version 1.0 March 2011

Type: Double Blind Peer Reviewed International Research Journal

Publisher: Global Journals Inc. (USA)

Online ISSN: 0975-4172 & Print ISSN: 0975-4350

# Design of Requirement Engineering Model during Reengineering

By Ashok Kumar, Anil Kumar

*Kurukshetra University*

**Abstract-** Requirement Engineering<sup>[1,2,3]</sup> is the most effective phase of software development. Its aims is to collect required requirement from stakeholders in the right way. It is very important of every organization to develop quality of software product, that can satisfy the users needs. It will focus number of viewpoints, roles, responsibility and objectives. Therefore it becomes necessary to apply requirement engineering in every phase of reengineering modules.

Requirement engineering depends on customers request and planner knowledge.

Requirement engineering model help in conceptual design and implementation design, also referred to as planning stage and implementation planning.

Requirement engineering provides an incremental approach for reengineering. It concerned with the requirement elicitation, analysis, specification, validation and requirement management.

Systematic requirements analysis is also known as requirements engineering. It is sometimes referred to loosely by names such as requirements gathering, requirements capture, or requirements specification.

**Keywords:** Reengineering, Requirement Engineering, Architectural Design, Review and Analysis

**Classification:** GJCST Classification: D.2.2



*Strictly as per the compliance and regulations of:*



# Design of Requirement Engineering Model during Reengineering

Ashok Kumar<sup>1</sup>, Anil Kumar<sup>2</sup>

**Abstract-** Requirement Engineering<sup>[1,2,3]</sup> is the most effective phase of software development. Its aims is to collect required requirement from stakeholders in the right way. It is very important of every organization to develop quality of software product, that can satisfy the users needs. It will focus number of viewpoints, roles, responsibility and objectives. Therefore it becomes necessary to apply requirement engineering in every phase of reengineering modules.

Requirement engineering depends on customers request and planner knowledge.

Requirement engineering model help in conceptual design and implementation design, also referred to as planning stage and implementation planning.

Requirement engineering provides an incremental approach for reengineering. It concerned with the requirement elicitation, analysis, specification, validation and requirement management.

Systematic requirements analysis is also known as requirements engineering. It is sometimes referred to loosely by names such as requirements gathering, requirements capture, or requirements specification.

**Keywords:** Reengineering, Requirement Engineering, Architectural Design, Review and Analysis

## I. INTRODUCTION

As we know that reengineering<sup>[4,5,6,7]</sup> is the revolution in software industry, it will reduces almost more than 60 percent cost as compared to maintenance of any software modules.

And we already know that maintenance is one of the critical activity of any software modules, but continue maintenance of software modules is not a good choice to increase the quality of the software. Because while continue maintenance takes place it will result increases the complexity of software modules.

Therefore, Reengineering is good choice than maintenance. The fundamental goal of reengineering was to minimize the impact of implementation expected system changes to the system.

---

*About<sup>1</sup>- Professor, Department of Computer Science & Application Kurukshetra University, Kurukshetra, India*

*About<sup>2</sup>- Asst. Professor, Computer Science & Engg. Vaish College of Engineering, Rohtak Rohtak, Haryana (124001), India  
Bestanil2005@rediffmail.com*

The purpose of this paper is to revisit the existing design methodology and develop a **requirement engineering model**, that will help to determine actual requirement of stakeholders before reengineering of software projects takes place. This data strongly suggests that, reengineering through new methodology significantly reduce full life cycle costs, schedules and risk.

## II. LITERATURE REVIEW

The research is based on the process models found in<sup>[8,9,10,11]</sup> as these are the models that have been designed keeping in view the GSD context for streamlining the phase of RE. The process models given in sideline of any case study as in<sup>[12,13]</sup> are excluded because such processes are only for specific case study and situation. Moreover, the generic process models for RE that are not designed specifically for GSD(Global Software Development) context such as that proposed in<sup>[14]</sup> have also been excluded because they are not suitable to this research

In evolutionary model<sup>[8]</sup> , requirements elicitation, requirements analysis and negotiation, requirements specification, and requirements validation are covered but this model addresses only few of the major GSD issues including a cultural issue which covers cultural diversity. Knowledge management issues cover knowledge sharing with users and knowledge acquisition techniques whereas technical issues over artifact sharing. Communication issues, time-zone difference, communication media and strategic issues are not covered in this model. Though this model is proposed to support the management of telecommunication network topologies but the reason for including this model is that it is documented process for DSD and is a good candidate for the GSD as well and can be applicable to other projects and domains as well.

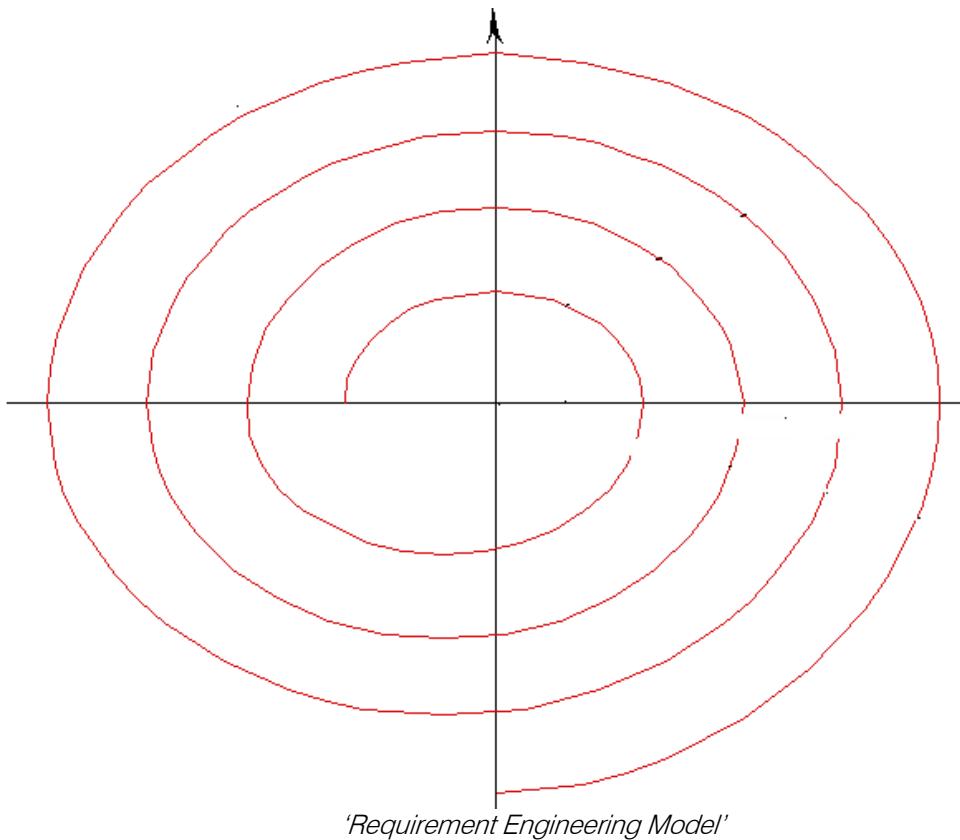
In requirement traceability model<sup>[9]</sup> , requirement traceability and requirements management are covered. Though this model does not cover all RE activities but the reason for including this model in this paper is that it can be very helpful in elicitation of correct requirements in GSD and the other reason for including this model is the level of importance attached to the requirements traceability and how this model can assist for better understanding of requirements in GSD



environment is also illustrated in this model. Requirement traceability helps in collecting rationale behind the requirements and this model also helps in indicating Value attached to the requirement. The GSD issues covered by this model are knowledge management, which includes knowledge sharing, and knowledge acquisition techniques. Technical issues (documentation sharing) and communication mode (asynchronous) are also covered. Other important issues like cultural issues, communication issues, strategic issues and time-zone difference are not addressed in this model.

In [10], though all activities of RE are not covered but major issues of GSD and three important activities of RE are covered. The other reason for including this

process model is that it is documented process and it is in the context of GSD. The important RE activities which are covered by this model include requirements analysis and negotiation, requirements specification and requirements validation. This process model does not address the problems related to geographically distributed specification team. For example if some of the specification team members are in country A and some are in country B then, requirement speciation is quite a hectic tasks as team members at both sites need to validate SRS. This model address important GSD issues including cultural, technical, knowledge management and communication issues (poorly defined).



In the above model, there is four phase that play very crucial role during reengineering of software module before actual process takes place, because if we know each and every requirement of customers before actual process takes place then we can reduce overall complexity of module in later phase.

The phases of requirement engineering model given below:

1. *Concept of formulation / Problem Understanding*

This phase generally proposed by people outside the system. E.g. Senior manager in the organization, politician, chairman, or other member of

the organization who want to get benefit directly or indirectly.

Here we will try to know, why reengineering takes place and the impact that the proposed concept is likely to have. This process should involves primarily domain experts rather than the system engineers. Although some system engineering input is essential. The above activity is iterative process

- all stakeholders involved are identified together with their goals (called win conditions);
- Conflicts between these goals are captured together with their associated risks and uncertainties;
- Goals are reconciled through negotiation to reach a mutually agreed set of goals,

constraints, and alternatives for the next iteration.

## 2. Requirement Engineering

Requirement engineering<sup>[1,2,3]</sup> play very important role in identification of need:

- Identify need, want or desire
- New or improve capability
- Based on perceive or real deficiency
- Statement of need
  - New system requirement may be defined
  - Date when new capability introduced or used
  - Estimate need resources
  - Specific quantitative qualitative terms with enough detail to proceed to next step
- Evaluate approaches in terms of performance, effectiveness, maintenance, logistic support and economics
  - Type and technological maturity
  - Stability and growth potential
  - Technology life time
  - Number of suppliers
- Recommend likely course of action such as
  - Feasibility study, determine objective, investigate in depth, possible way of implementing
    - Technology Assessment: hardware and software needed to implement
    - Solution of proposal
    - Cost/benefit analysis: estimate cost of developing, determine benefit
    - On completion of it, a decision is made on a preferred solution
  - Result of feasibility analysis should include life cycle consideration
  - Need should derive technology

## 3. Architectural Design

Evolve through a series of stages and each stages is evaluated, before proceeding to next stage

Here one of very important activity takes place that to determine information need, determine different sources from where information will be obtained.

Here two types of design takes place

- Logical Design: is a more detail design, which includes all major components and entities plus their relationship. The data flow and connection are detailed in this stage
- Physical Design: has all major components and entities are identified within specific physical servers and location or specific software services, objects or solution, include all known detail such as OS, Ver. No. etc.

## 4. Review and Analysis

Review is one very important activity, in the requirement engineering model, it is essential to cross verification of functionality. Once anywhere requirement changes it is very important to review it according to change requirement.

Review and analysis is again a iterative process in the requirement engineering model, it help in

- Conceptual Design (Preliminary Requirement Review)
- System Design Review (Preliminary Design Review)
- Equipment / software Design Review
- Critical Design review (Detail Design and development)
- Risk analysis is vital activities of analysis

Again evaluation function play important role, for day to day project coordination and data review, formal design review

## III. RESULT & DISCUSSION

There is no doubt that requirement engineering is first activity of software development life cycle which play very important role in successes or failure of software projects, however there are some problem that we will face during requirement engineering during software development.

The purpose this paper is design requirement engineering model during reengineer that will solve the inconsistency of management for requirements engineering that will suffers these problems during software development:

- The specific kind of inconsistency being considered is not always clear. In fact, there is no common agreement on what a conflict between requirements does really mean. The lack of precise definition is a frequent source of confusion. Moreover, most techniques consider binary conflicts only, that is, conflicts among pairs of requirements. As we will see, there may be conflicts among three requirements, say, that are non conflicting pair wise.

- There is no systematic support for detecting conflicts among non operational specifications of goals or requirements—if one excepts theorem proving techniques for logically inconsistent specifications. Current conflict management techniques take it for granted that conflicts have been identified in some way or another.
- There is a lack of systematic techniques for resolving inconsistencies through goal/requirement transformations—one notable exception is<sup>[15]</sup>, which proposes a set of operators for restructuring objects involved in conflicting goals.

The above requirement engineering model solve above problem that we will face during software development, as well as it play important role during reengineering of module.

#### IV. CONCLUSION

This model play very important role in reengineering, it not only reduces the risk to meet specification, but it also ensure to avoid immediately assuming you know what the problem and solution are, Going directly to try to solve the problem before thinking about it.

It provides systematic and organized approach that allows us to focus on achievable goal and to attain the best possible results from available resources. Therefore it will set the system objective, where reengineering takes place in the organization, so that management and employees agree to the objective and understand what they are in the organization.

This model play very important role in reengineering because it play very important part in testing of reengineered modules according to new requirement as well as existing functionality

#### REFERENCES RÉFÉRENCES REFERENCIAS

1. J.A. Espinosa and E. Carmel, "The impact of time separation on coordination in global software teams: a conceptual foundation", *Software Process: Improvement and Practice*, Wiley InterScience, vol.8, no.4, pp. 249-266, 2003.
2. D. Damian and D. Zowghi, "Requirements Engineering Challenges in Multi-Site Software Development Organizations", *Requirements Engineering Journal*, vol. 8, pp.149-160, 2003.
3. D. Damian, "Requirements Engineering in Distributed Projects", *International Conference on Global Software Engineering*, 2006.
4. Tenner, A. R., and Detoro, I. J., 1992. "The process redesign—the implementation guide for managers", Prentice Hall, New Jersey.
5. Cook, S., 1996. "Process Improvement: A Handbook for Managers", Gower Publishing, Hampshire, England.
6. Davenport, H. T. and Short, E. J., 1990. "The New Industrial Engineering: Information Technology and Business Process Redesign", *Sloan Management Review*, 34(4), pp. 11-27.
7. Davenport, T. H., 1993. "Need radical innovation and continuous improvement? Integrate process reengineering and TQM", *Planning Review*, 21(3), pp. 6-12.
8. C. Maidantchik, M. Montoni and G. Santos, "Learning organizational knowledge: an evolutionary proposal for requirements engineering", *Proceedings of the 14th international conference on Software engineering and knowledge engineering*, Ischia, Italy, pp. 151 – 157, 2002.
9. M. Heindl and S. Biffl, "Risk Management with Enhanced Tracing of Requirements Rationale in Highly Distributed Projects", *International Conference on Software Engineering (ICSE)*, Shanghai, China, pp. 20-26, 2006.
10. L. Lopes, R. Prikladnicki, J.L.N. Audy, and A. Majdenbaum, "Distributed Requirements Specification: Minimizing the Effect of Geographic Dispersion", in *Proc. ICEIS (3)*, pp.531-534, 2004.
11. G. N. Aranda, A. Vizcaino, A. Cechich and M. Piattini, "Strategies to Minimize Problems in Global Requirements Elicitation", *CLEI electronic journal*, vol.11, no.1, June 2008.
12. Hanish Jo, C. Brian and T. Theerasak, "Differentiating Local and Global Systems Requirements Gathering Processes in IS Software Development Projects", *PACIS Proceedings*, 2005.
13. Hanisch Jo and C. Brian, "Requirements Engineering During Global Software Development: Some Impediments To The Requirement Engineering Process-A Case Study", *European Journal of Information Systems (ECIS)*, vol. 16, no. 6, pp. 793-805, Dec 2007.
14. D.A.James and K.G.Markus, "An operational model for structuring the requirements generation process", *Springer- Verlag London Limited*, vol. 10, no.1, pp.45-62,2004.
15. W.N. Robinson and S. Volkov, "A Meta-Model for Restructuring Stakeholder Requirements," *Proc. ICSE19—19th Int'l Conf. Software Eng.*, pp. 140-149, Boston, May 1997.