CHARACTERIZATION AND VALIDATION OF REQUIREMENTS MANAGEMENT MEASURES USING CORRELATION AND REGRESSION MODEL.

S. Arun Kumar[1] and T.Arun Kumar[2]

[1]Assistant Professor, SCSE, VIT University, Vellore, Tamil Nadu, India
sarunkumar@vit.ac.in
[2]Professor, SCSE, VIT University, Vellore, Tamil nadu, India
arunkumart@vit.ac.in

ABSTRACT

Requirements engineering is one of the challenging and crucial phase in the development of software products. One of the key reasons found in literature survey the failure of software projects due to poor project management and requirement management activity. This paper mainly addresses 1. Formulate a mixed organization structure of both traditional approaches and agile approaches, to apply KM practices for both the approaches to achieve requirements issues such as missing and inconsistency of requirements and improve the project management activities in a global software development environment. 2. Propose requirements metrics to measure and manage software process during the development of information systems. The major contribution of this paper is well-founded methods to manage the project and effective requirements management metrics to measure changing requirements while giving particular attention to the requirements engineering issues such as completeness and consistency. Two hypotheses have been formulated and tested this problem through statistical techniques and validate the same.

Keywords: *Requirements Engineering (RE), Requirements Management (RM), Requirements Metrics, Knowledge Management (KM).*

1. INTRODUCTION

In the software engineering literature project management and requirements management activities are crucial in order to achieve the high quality software system. According to the National Association of Software and Service Companies (NASSCOM) report the outsourcing is grow up to 53% yearly rate of the companies outsourced software development to India. Due to increase the globalization of software development deriving many benefits to the software companies at the same time it will leads to several problems/issues occurring in the requirements engineering phase which includes

1. Understanding the requirements among different groups of people (geographical distributed development teams).
2. Problems in deals with frequently enhancing the requirements.
3. Cross culture relationship, communicational issues.
4. Coordination among employees.

Traditional approach at the global level, be agile approach at the local level in organization i.e., mixed organization structure is proposed to handle these concerns. To apply KM practices to

DOI : 10.5121/ijcsit.2011.3217

the proposed organization structure is helps to reduce the various requirements engineering issues in global software development. Requirements engineering activity is broadly divided in to two categories first one is requirements definition which deals with elicitation, analysis, documentation and review and second one is requirements managements which deals with change management and traceability. This paper mainly concentrates on requirements change management is mentioned in fig.0. Propose requirements management metrics to measure the requirements changes, requirements schedule, and average rate of defects during the development of information systems in a multi site environment. This paper mainly covers the following application areas software process improvement, requirement management, software project management and software measures is clearly mentioned in fig.6

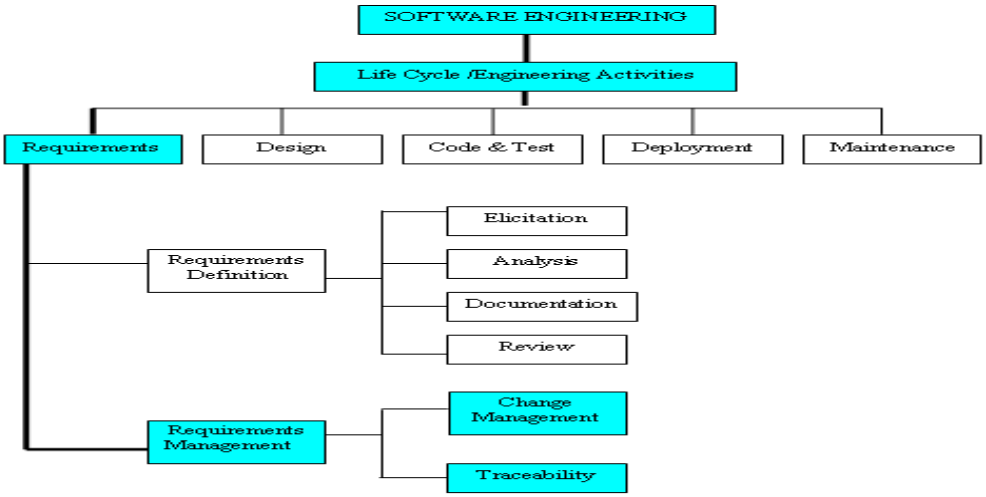

Fig.0: RE activities

2. LITERATURE SURVEY

The some of the empirical evidence are described based on previous researches of global software development: RE challenges due to geographical distribution of stakeholders in a multi-site organization, cultural diversity, language and time zone.[1] The survey includes 70% of the problems occurred in the requirements engineering phase due to frequent changes in customer requirements,[2] 52% of problem due to cultural differences among development team, 88% problem due to loss of communication richness,[7] 70% problem due to lack of requirements management,[6] 70% problem due to co-ordination break down among people [8] and 65% of the communication problems that exists due to lack of understanding the requirements. [3] The various social issues would exist while managing the customer requirements. [4] The reason beyond these issues is no proper unified model; organization structure exists currently to manage the requirements engineering issues while developing a software product in a multi site environment.

3. REQUIREMENTS ENGINEERING TECHNICAL PROBLEMS

The survey said there number of technical problems arising in requirements phase in global software development is shown in Fig.1.

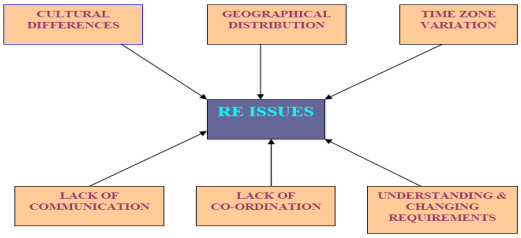

Figure 1: RE ISSUES

The surveys gives 44% to 80% of all defects are found in the requirements phase [21]. The software projects regularly fail due to problems with requirements. Frequent changes in requirements may result in incomplete, wrong, ambiguous requirements. The poor requirements management is leads to increase overall cost, decrease quality of the system or fail altogether.

4. REQUIREMENTS MANAGEMENT MEASURES

Requirement is need of customer. Requirements are capabilities and intention to any product or service must conform which is common to all development of the system and other engineering activities. The term "engineering" also suggests that the outputs of a requirement engineering process need to be carefully engineered, where those "outputs" are usually understood to be detailed specifications. [12].

Requirements engineering is defined as follows: *"Requirement engineering (RE) is a set of activities concerned with Identifying and communicating the purpose of a software-intensive system, and the context in which it will be used. Hence, RE acts as the bridges between the real world needs of user, customer, and other constituencies affected by a software system, and the capabilities and opportunities affected by software-intensive technologies"*. [12]

Requirements management is defined as follows: *"Requirement management is a set of activities that helps the project team to identify, control, and track requirements and changes to requirements at any time as project proceeds*". [13]

Software metrics used in requirement management process to provide the information, needed to take key project decisions and to take suitable actions. Metrics are used for continuous project improvement and control in requirement management process. The general and comprehensive set of requirement management measures is defined for implementation of the goals of the requirement management KPA (Key Process Area) within the SW-CMM level2 is mentioned in table 1. [14]

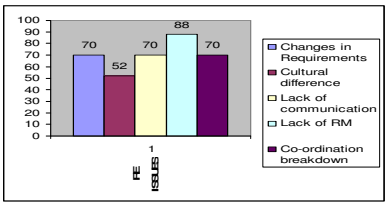

Figure 2: RE SURVEY

4. METHODOLOGY TO BE ADOPTED

Phase I Analyze the performance issues in traditional approach and agile approach in global software development.

Phase II Design an optimized framework model to integrate traditional approach at the global level and agile approach at the local level of a organization structure which is arrived at phase I.

Phase III Propose requirements metrics to measure the requirements changes, average rate of defects and requirements schedule.

Phase IV Formulate the hypotheses and tested the problem using statistical techniques and validate the same.

5. PROPOSED FRAME WORK

The researcher will collect the materials related to the key areas of requirements engineering with respect to process. Through comparative study will be carried out to achieve an integrated approach for the software development organization. Diagrammatic representation of the proposed organization structure is shown Fig.3

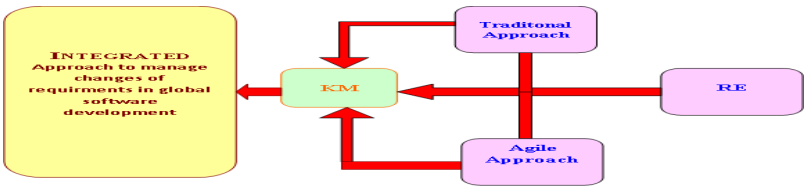

Table 1. Requirements Management Measure [14]

S.No	Measurement Type	Measure	Entity
1.	Direct	Requirements affected by Change	Requirement management process
2.	Direct	Inconsistent requirements	Requirement management process
3.	Direct	Missing requirements	Requirement management process
4.	Direct	Incomplete requirements	Requirement management process
5.	Direct	Initial requirements	Requirement management process
6.	Direct	Final requirements	Requirement management process
7.	Indirect	Requirement schedule	Requirement management process

Figure.3 Globally Traditional, Locally Agile

6. TRADITIONAL APPROACH VERSUS AGILE APPROACH

The survey said 1,027 projects in the United Kingdom 13% did not fail, and waterfall-model was the one of "major contributing factor for failure, problem being addressed in 82% of the projects. The United States Defense Department projects concluded that "46% of the systems really did not meet the real needs as well as user specified requirements, that they were never successfully used, and another 20% required for extensive rework" to be usable. The survey said

traditional approaches improve the quality and productivity of the software system incase of agile approach satisfy the real needs of the customers and project success. The requirements keep on changing while the project is progress. The requirements changes due to improve the effectiveness of the system to satisfy the real needs of the users. It is difficult to respond the customer changes once the requirements have been accepted in the traditional approaches but agile approach is adoption of changing requirements dynamically even the project is progress. Our proposed model is creating mixed organization structure of both traditional approaches at the global level to finalize the requirements for system development and be agile approach at the local level to elicit, analyze, and review the software requirements dynamically.

Reference	Type of Measures	Attributes	Entity	Theoretical /Empirical Valid	Environment
Loconsole and Börstler [15,19]	Total number of requirements, number of Initial, current, and final requirements, status of requirements, number of changes per requirement, status, type, reason, and cost of change to requirements	Size and status of requirements specifications, size and status of changes to Requirements	Requirements management	Kitchenham representational theory [18]	Academic case study
Loconsole and Börstler [16]	Num lines, words, actors, use cases, subjective volatility Volatility (num changes, major, moderate, minor, revisions)	Requirements specifications	-	Spearman correlation	Industrial software
Ambriola and Gervasi [17]	Measures of stability (amount of information contained in requirements at time t) and efficiency	Amount of rework, perceived work efficiency	Requirements analysis process	Plot of data	Academic experiment
Our proposed Methodology	Number of Requirement Changes, Average Rate of Defects, Requirements Schedule	Changes to requirements, Schedule Slippage, and rate of defects	Requirements Management	Pearson Correlation coefficient and Regression Analysis	Web Based Project

Table 2. Measures Defined and Validated this Paper

The reason beyond to go for this mixed organization structure is now a days traditional way of software development has faced several challenges/issues like understanding ,changing requirements in a multi site environment , the agile approach successfully to manage the challenges in a global software development environment. To implement both traditional and agile method in an organization, to take the benefits of these two methods to improve accuracy, productivity, and quality of software development projects. To apply KM practices for both the approaches to achieve requirements engineering issues like missing requirements and inconsistency and improve the project management activities in a global software development environment. We propose requirements management metrics to measure the requirements changes, average rage of defects and requirements schedule in order to improve effectiveness of the software development process. The Analysis Center for Software [10] and software development researchers in Israel [11] said agile methods are 29% better cost, 91% better schedule, 97% better productivity, 50% better quality, 400% better satisfaction, and 470% better return on investment compare than traditional approaches is mentioned in fig.4. The survey said agile software development deriving many benefits to the software companies which includes:

- Early and continuous delivery of software system.
- Dynamically keep on changes in the requirements even progress in the development.
- More on customer and end users involvement towards system development.
- Return on investments in earliest manner.
- Meet the stakeholders expectations.
- Simplicity and visibility of actual progress of the project is available.
- Face-to-Face conversation.
- Risk reduction in the overall development of the software system.

7. REQUIREMENTS MANAGEMENT METRICS

We propose a set of metrics for requirements phase. The purpose of this metrics is in order to improve the effectiveness of the software development process.

Actual Requirements Count (ARC): The numerical count of total number of original requirements is present in the project.

Number of New Requirements (NNR): The numerical count of the actual number of requirements in the project, compared to the count of the number of new requirements introduced in the project cycle: analysis, design, code, test, and implementation.

NNR = Number of Original requirements + newly added requirements – Deleted requirements

Number of Requirements Changes (NRC): To calculate total number of requirements change requests (insert/delete/update) during the project cycle.

NRC = newly added requirements + deleted requirements + revised requirements. (ARC – Number of Requirements changes (insert/delete/update))

Average Rate of Requirements (ARR): The average number of requirements change requests required in the entire life cycle of the project.

Number of Requirements changes (NRC) /Actual Requirements count (ARC) *100 per phase.

Average Rate of Defects (ARD): The total number of defects is detected in each phase of the project per month.

Requirements Schedule (RS): The numerical count total effort is required to complete the task**.**

Actual completion date- Estimate completion date / actual effort

8. KNOWLEDGE MANAGEMENT ISSUES IN GLOBAL SOFTWARE DEVELOPMENT

Knowledge management is all about the people to test how the people are working towards in order to achieve business and organizational objectives. Knowledge management is defined as "delivering the right knowledge to the right persons at the right time." To apply knowledge management practices the proposed organization structure of software development deriving many benefits this includes:

- KM helps to ensure the right information and to make right decision.
- On time to delivery of the system
- Increase the profitability

- To improve the project management and communication among team members.
- KM helps to reduce overall development time and cost.

Global software development really requires effective knowledge management techniques to fulfill the user expectations. Knowledge management techniques are taking major initiative for the success of the geographical distributed software development. Knowledge management mainly focusing on knowledge sharing of globally distributed development teams, cross culture issues, communication issues (meet time difference time zone and distance) and technical issues such as lack of synchronization.

Our main aim is successfully manage and resolve these issues in distributed software development environment. Our proposed model is framing an effective organization structure in a current software industry practice.

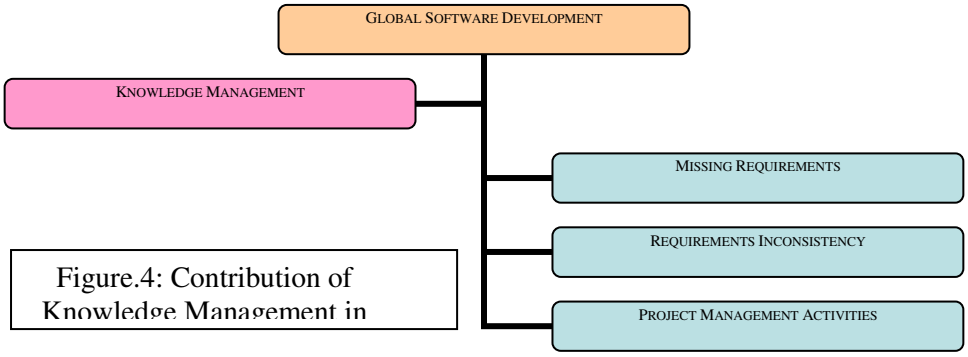

Figure.4: Contribution of Knowledge Management in

9. EXPECTED OUTCOME

An integrated approach and a validated model for implementing the requirements engineering framework involve managing the requirements changes in an effective manner.

To analyze some of the issues related to this problem currently arising in global software development project having some remarks: reduce cost, increase productivity manage, measure the requirement changes, average rate of defects and requirements schedule and improve product and services.

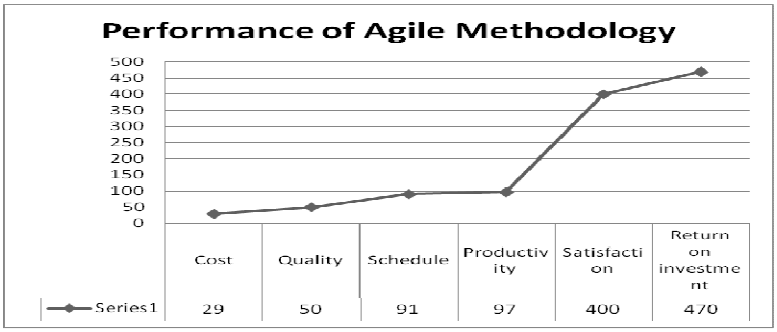

Figure.5: Performance metrics of Agile approach

10. METHODOLOGY AND DATA ANALYSIS

In this paper a set of requirements management measures has been presented and validated based on table 1 and table 2.

Our observation is that the majority of existing measures are neither theoretically nor empirically validated for different academic case studies and industry case studies. Table 3 provides the data were collected based on web based project according to this data we have measured and validate via the statistical analysis and find correlation coefficient and forming regression model of NRC, ARD and RS.

The main aim of this study is to establish the importance of requirements management metrics for global software development projects. We use correlation and multiple correlations, regression analysis to test the two hypotheses formulated above to study the impact of one metric over another.

Table 3

Project	NRC	ARD	RS(Days)
A	**600**	**792**	**210**
B	**750**	**862**	**265**
C	**816**	**937**	**350**

We have formulated the following two hypotheses focusing our study and to assist the statistical analysis:

Hypothesis 1: The metric NRC influences the metric average rate of Defects (ARD)

Hypothesis 2: The metrics NRC, ARD decide the requirements schedule (RS) in person-hours.

Correlation is a statistical measurement of the relationship among two variables. The correlation ranges from +1 to –1. There is no relationship among the variables then the correlation is said to be zero correlation If one variable goes up, and another variable goes down then the correlation is said to be prefect negative correlation(-1) . A correlation of +1 denotes a perfect positive correlation; it means the both variables progress mutually in the same direction.

Regression analysis is a statistical method where the mean of one or more random variables is predicted based on other measured random variables. Regression analysis is mainly used to understand how the actual value of the dependent variable changes when any one of the independent variables is varied. Here RS metric is dependent variable of NRC metric and ARD metric.

We have used IBM SPSS analytical software to find Pearson coefficient of correlation and regression analysis of above requirements metrics.

Based on the available data shown in Table 3 provides the NRC, ARD and RS values from a web based projects. The coefficient of multiple correlations 'R' between NRC (r_1) versus ARD (r_2), NRC (r_1) versus RS (r_3), and ARD (r_2) versus RS (r_3) is computed below

- R_{12}= 0.971
- R_{13}= 0.941
- R_{23}= 0.995

The value of 'R' is greater than zero is denotes a perfect positive correlation between this three variables. Here the relationship between two variables (NRC metric and ARD metric) change in one variable result in a positive or negative change in the other variable and also greater change in NRC metric result in corresponding greater change in other metric ARD and RS.

Say NRC = x_1, ARD = x_2, RS = x_3 when RS is dependent on NRC, ARD then the regression model will be

$$RS_{predict} = -7.1429+1.376_{x2} - 0.275_{x1}$$

The RS metric is predicted based on NRC metric and ARD metric which is clearly stated in the above regression model. The regression $R^2 = 1$ correlation coefficient determination = 1 so this regression model is prefect fit for the above requirements metrics. Through this data analysis gives a great deal of support to our two hypotheses formulated in this study. The clearest result observed form the value x_1, x_2 and x_3 is the strong relationship between NRC metric influence ARD metric and NRC and ARD metric decide the RS metric.

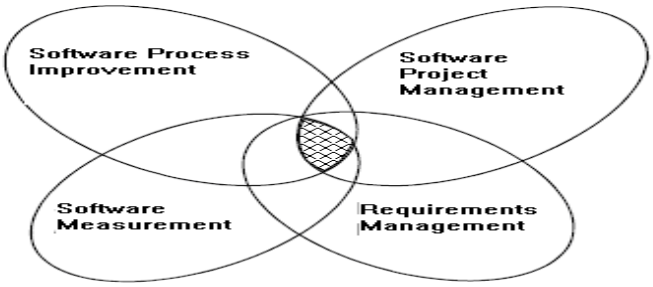

Figure. 6 Major Application areas

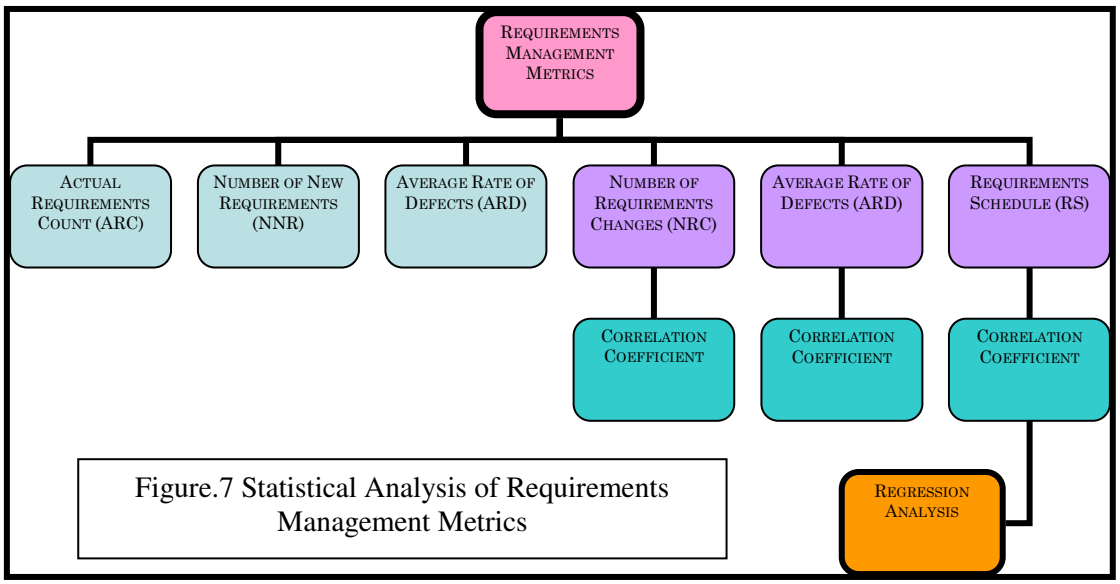

Figure.7 Statistical Analysis of Requirements Management Metrics

11. Conclusion

Though there are number of requirements engineering models being practiced in most of the organizations, no unified model, organization structure exists currently for implementation in managing requirements while developing a software product. KM methods, tools are required to achieve better requirements engineering practice in global software development. The proposed model would facilitate the organizations to effectively manage requirements engineering issues into various levels that will result in forming a benchmarked approach.

The conclusion of this paper based on statistical analysis of requirements metrics and benefit of proposed organization structure in a multi-site environment. In our future research is based on

development of software metrics to reflect on the quality factors to improve the software development process.

12. REFERENCES

1. Daniela E. Damian Didar Zowghi (2007). *"Requirements engineering challenges in multi-site software development"*Requirements engineering journal 8, pp. 149-160.

2. Esmaeil Kheirkhan, Aziz Derman (2009). *"A Framework for Requirments Engineering in End-user Computing",* IEEE World congress on computer science and engineering, 701-705.

3. Alter, S (2005*). "A Broad View of Systems Analysis and Design Research"*Implications for Research. Communications of the ACM, 21-26.

4. Jamaludin Sallim (2009) *"Requirement Engineering for Enterprise application development",* World academic science, engineering and technology www.waset.org/journals/waset/v4/v4-27.pdf.

5. Ebert C, Parro C H, Suttels R and Kolarczyk H (2001) *"Improving validation activities in a global software development",* In Proceedings IEEE Software: 545-554

6. F. Prikladnicki, J. Audy and R. Evaristo, "Requirements Management in Global Software Development: Preliminary Findings from a Case Study in SW-CMM context", *The International Workshop on Global Software Development*, ICSE, 2003, pp.53–5

7. S.Komi-Sirvio and M. Tihinen, "Lessons Learned by Participants of Distributed Software Development", *Knowledge and Process Management*, vol.12, no.2, Wiley InterScience, 2005, pp.108-122.

8. E. Carmel, "Global Software Teams Collaborating Across Borders and Time Zones", Prentice Hall, 1999.

9. Aybuke Aurum and Claes Wohlin(2000)."*Engineering and Managing Software Requirements",* Springer Publications, 2000.

10. Viktor Clerc (2005). *"Towards architectural knowledge mangement practices for global software development"* ACM, ISBN: 978-1-60558-038-8.

11. Berenbach.B, *"Global Software Development Handbook",* www.ism-journal.com/ITToday/AU9384_C002.pdf.

12. Steve Easterbrook, *"What are Requirements"*,2004. http://jasonnolan.net/kmd1002/easterbrook.pdf.

13. Pressman S Rogers, "Software Engineering, A Practitioner's *Approach"*, Fifth Edition, McGraw Hill Higher Education, 2001

14. Loconsole Annabella, "Measuring the Requirements Management *Key Process Area"*, Proceeding of ESCOM – European Software Control and Metrics Conference, London, UK, April 2001

15. Loconsole, A. and Borstler, J. 2003. Theoretical Validation and Case Study of Requirements Management Measures, Department of Computing Science, Umea University, Technical Report UMINF 03.02, (Jul. 2003).

16. Loconsole, A. and Borstler, J. 2005. An Industrial Case Study on Requirements Volatility Measures. In Proceedings of the 12th Asia-Pacific Software Engineering Conference (Dec. 15–17, 2005). IEEE Computer Society, Washington, DC, 249–256.

17. Ambriola, V. and Gervasi, V. 2000. Process Metrics for Requirements Analysis. In Proceedings of the 7th European Workshop on Software Process Technology (Feb. 21–25, 2000). R. Conradi, Ed. Lecture Notes In Computer Science, 1780. Springer- Verlag, London, 90–95.

18. Kitchenham, B.A., Pfleeger, S.L., and Fenton, N. 1995. Towards a Framework for Software Measurement Validation. IEEE Transaction on Software Engineering, 21, 12 (Dec. 1995), 929–944.

19. Loconsole, A. and Börstler J., (2007) A Correlational Study on Four Size Measures as Predictors of Requirements Volatility, Submitted to *Journal of Software Measurement*.

20. Eberlein A. (1997). Requirements Acquisition and Specification for Telecommunication Services, PhD Thesis, University of Wales, Swansea, UK.

21. Nafon Assawamekin.(2010) An Ontology-Based Approach for Multiperspective Requirements Traceability between Analysis Models, 9th IEEE/ACIS International Conference on Computer and Information Science.