

Effort Reduction in RUP using CRM for Project Development: Mapping the Best Practices of CRM into RUP

Muhammad Javed, Bashir Ahmad, Muhammad Ali Abid, Muhammad Ahmad Jan
Sheikh Muhammad Saqib and Muhammad Ali Shah

¹*Institute of Computing and Information Technology Gomal University,
D. I. Khan, Pakistan*

*javed_gomal@yahoo.com, bashahmad@gmail.com, abid.gu@gmail.com,
mr_ahmadjan@yahoo.com, saqibsheikh4@hotmail.com,
mohd5_ali14@yahoo.com*

Abstract

It has been observed that with ever changing technologies need for different software methodologies also evolved. Different methodologies are actually developed to cater the needs of different software development technologies. These methodologies are so technology specific that it becomes a hazard whenever these are applied as generic methodologies for all type of applications. For example it is evident that none of these methodologies can fully cope with the development of expert system, soft computing systems or scientific applications. Moreover Rational Unified process, besides its numerous disadvantages, is very effective uniform and having wide range of applicability. On the other hand Cleanroom software Engineering is an efficient methodology which is aimed develop zero defect software. At the very mundane level it is very much unrealistic to think of a man-made product without any fault or any error, but a still believe that the efforts involved in Cleanroom software engineering at utmost level, or the ultimate efforts to remove the possibility of errors, with the mapping of RUP with CRM the effort for development is to be decreased. However the Cleanroom methodology is not the sole solution for the efficient and effective development of software.

Keywords: *Cleanroom methodology (CRM), Rational Unified Process, Software Methodologies*

1. Introduction

RUP is another software development approach. It is an organized and sophisticated art of allocating the assignments and jobs within the development team. RUP ends up producing delivering a quality software product which is aligned with the requirements of its end users in a controlled schedule and finances. RUP is basically Process Product in nature. RUP is followed with the various available collection of software development tools. It is also a process framework which is followed and enriched in order to best qualify the requirements of the organization [1].

RUP is an organized and regulated method of software development. It is basically process product is nature which is developed and adopted by Rational Software. RUP is used in variety of ways for different type of software projects. It enables to utilize other rational tools for software development rather than its effective application and integration with other vendors' offerings [2].

RUP is an iterative and incremental software development approach in which a project is broken down into various modules which is developed one after the other. Each iteration starts with a milestone to achieve and similarly ends up with a deliverable which takes the product into the final stages gradually. Every iteration of RUP is constituent of various steps like requirements gathering, requirements analysis, implementation, assimilation and test phase [3].

RUP is use case driven, iterative and incremental, and architecture centric approach. In order to design the user interface in RUP, the phases of user interface modeling and user interface prototyping are carried out. Use case model serves as an input to these activities which tells how to use the system [4].

Clean room is a software development method which confront the classical point of view that zero defect software is either extremely expensive or is next to impossible for commercial software systems. It is a complete software development methodology which covers all the activities required to develop a full furnished quality product [5].

Cleanroom software engineering is a collection of prescribed specifications which visualize the behavior of the system to the outside world. It involves spending more time on developing the error free system rather than quickly creating the system and then spending time on debugging.

It follows a rigid incremental approach of software refinement and verification utilizing a box-structured approach which clearly defined user requirements and the architecture of the system. The productivity is targeted to be increased by incrementally developing the product. It is normally said that 50% of the product is 100% complete instead of 100% being 50% complete.

Clean room uses this method using the object based technology of box structures called black, state, and clear boxes [6].

The main advantage of Cleanroom software engineering over the traditional software development is, the traditional software development methods are wide open to the errors which takes a valuable amount of time in exploring and correcting these errors whereas in contrast this approach Cleanroom methodology is less error prone which means less debugging and rework so there is higher level of productivity [7]. As the less number of errors in the software always result in less cost, the same and the ultimate objective of Cleanroom software engineering process which aims to reduce errors up to maximum limit or even zero errors in deliverable [8].

2. Facts about CRM and RUP

The basic objective of this research is to map the best practices of Cleanroom software engineering with RUP model of software development .As it has been discussed that both of these approaches are of the most applicable and predictable approaches for the software development having their own merits and demerits. If we go into deeper insights in both of these models then some interesting facts are explore which include the following similarities.

- These models provide guidance to the order of team activities.
- Both of these models direct the task of individual developer in the team as a whole.
- Both of these models specify what artifacts should be developed.
- Both of these models emphasize on the criteria on the monitoring and measuring activities during all phases of development since its inception to completion.

- Both of these models are not just probable but a purely practitioner approach is followed in both of these models to insure higher level of reliability, maintainability and overall quality.

Besides all these similarities there are also considerable differences between these two models.

- The Unified software development process model is an outcome of 30 year experience which provides a complete solution to the software development problem whereas Cleanroom methodology was introduced in order to bridge the gap between heavy weight and light weight methodologies.
- RUP is component based which means that the software system being built is made up of software components interconnected via well-defined interfaces. Each and every component in RUP is treated as a complete subsystem whereas on the other hand Cleanroom software methodology stress on the totality of system.
- Distinguishing aspects of RUP are captured in three key words: use case driven, architecture centric, iterative and incremental. This is what makes the RUP unique. On the other hand Cleanroom methodology is based on iterative rectification of requirements to eliminate the errors completely and there is no reversal in this methodology.
- The RUP is a process which derives the development process where as Cleanroom methodology is a process which controls the development process.
- RUP is completely iterative and incremental with a wide range of flexibility such that not only intra-face iteration is permitted but interface iteration is also permitted .On the other hand Cleanroom methodology is rigid in nature having controlled iterations that reduce cost risk to the expenditure on a single increment.
- RUP, due to its natural tendency towards human behavior speed ups the tempo of whole development effort because developer work more efficiently towards results in clear and short focus schedule whereas the Cleanroom focus on the long term planning of designing and reserving a short span for the development.
- RUP acknowledges a reality ignored by the Cleanroom software engineering that the user requirements and corresponding needs cannot be fully defined up front. They are typically refined in successive iterations.
- RUP is very flexible model whereas Cleanroom stress on the rigidity with assumption that the development should not be unless the design guaranties zero error.

In the view of above comparison it can be easily understood that having similarities in major areas both of these models still have the huge difference in their approach. It is worth mentioning that there is sufficient scope available to map Cleanroom software methodology with RUP model to improve to overall efficiency of RUP model.

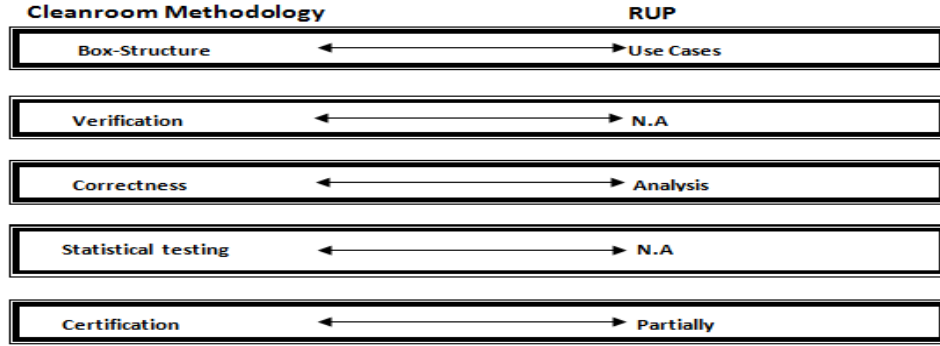


Figure 1. Overview of Comparison of Cleanroom Methodology and RUP

Figure 1 illustrates only the two phases of RUP Inception phase and Elaboration phase can be mapped with Cleanroom methodology whereas Construction phase and Transition Phase play the same role for development as in Cleanroom methodology. For first phase of RUP, the Use cases can be mapped with Box structure of Cleanroom methodology and for second phase of RUP, Analysis can be mapped with Correctness process of Cleanroom methodology.

In Cleanroom methodology box structure is used for defining the behavior of system and it is used to collect the system requirements. In Cleanroom methodology refinement process for requirements specification is also performed by box structure.

3. Comparison Effort for the Development of Project using CRM, RUP and Proposed Model

The presentations of comparison results of the three models on the basis of experiments for the effort required to complete a project.

In first experiment the project is developed by using Cleanroom methodology and in second experiment the project is developed by using RUP and in third experiment the project is developed by using RUCM.

3.1 Experiment-1:

The effort required to develop a project using Cleanroom methodology is:

$$E_C = N(S + x + y + z) + D_C$$

Whereas

E_C = Required effort for the development of project by using Cleanroom methodology.

S = Effort required for stage.

N = Total number of stages (Features) used in the project.

x = Effort required for mathematical modeling and quantification process in Person/hour.

y = Effort required for statistical testing in Person/hour.

z = Effort required for certification process in Person/hour.

D_C = Development effort without stages in Person/hour.

Table 1. Effort for the Development of Project by using CRM

Stage No.	Effort for Stages(S) in (P/H)	Effort for Mathematical Modeling (x) in (P/H)	Effort for Statistical testing(y) in (P/H)	Effort for Certification process(z) in (P/H)	Effort for Development (D _c) in (P/H)	Total Effort for Development (E _c) in (P/H)
1	48	1	1	1	2400	4338
2	52	1	1	1	2400	4490
3	58	1	1	1	2400	4718
4	63	1	1	1	2400	4908
5	78	1	1	1	2400	5478
6	90	1	1	1	2400	5936
7	105	1	1	1	2400	6504

Table-1 illustrates that in Cleanroom methodology, the effort is needed in person/hour for mathematical Modeling and quantification, statistical testing and for certification process for any stage, because these all characteristics are available in CRM.

3.2. Experiment-2:

The effort required to develop a project using RUP is:

$$E_R = N(S) + D_R$$

Whereas

E_R = Required effort for the development of project by using RUP.

N = Total number of stages in used in the project.

Each stage is in fact a feature of Cleanroom methodology which has been discussed in Chapter-4.

S = Effort requires to complete a stage in Person-hour.

D_R = Development effort for RUP without stages.

Table 2. Effort for the Development of Project by using RUP Model

Stage No.	Effort for Stages(S) in (P/H)	Effort for Mathematical Modeling (x) in (P/H)	Effort for Statistical testing(y) in (P/H)	Effort for Certification process(z) in (P/H)	Effort for Development (D _R)	Total Effort for Development (E _R) in (P/H)
1	48	-	-	-	5400	7224
2	52	-	-	-	5400	7376
3	58	-	-	-	5400	7604
4	63	-	-	-	5400	7794
5	78	-	-	-	5400	8364
6	90	-	-	-	5400	8820
7	105	-	-	-	5400	9390

Table-2 illustrates that in RUP, no effort is needed in person/hour for mathematical Modeling and quantification, statistical testing and for certification process for any stage, because these all characteristics are not available in RUP and also for RUP process the effort for development is double but slightly greater than the twice of CRM.

3.3. Experiment-3:

The effort required to develop a project by using RUCM (for half) because 50 % mapping has been done for RUP with Cleanroom for first two phases where as other two phases have the same role in development as in Cleanroom methodology:

$$E_{RUCM} = N/2 (S + x + y + z) + N/2 (S) + D_{RUCM}$$

$$E_{RUCM} = N/2 (S) + N/2 (x) + N/2 (y) + N/2 (z) + N/2 (S) + D_{RUCM}$$

$$E_{RUCM} = N * S + N/2 (x + y + z) + D_{RUCM}$$

Whereas

E_{RUCM} = Required effort for the development of project by using RUCM which is proposed model.

N = Total number of stages in used in the project (adopted from CRM).

S = Effort required to complete a stage in person/hour.

x = Effort required for mathematical modeling and quantification process in Person/hour.

y = Effort required for statistical testing in Person/hour.

z = Effort required for certification process in Person/hour.

D_{RUCM} = Development effort without stages.

Table 3. Effort for the Development of Project by using RUCM

Stage No.	Effort for Stages(S) in (P/H)	Effort for Mathematical Modeling (x) in (P/H)	Effort for Statistical testing(y) in (P/H)	Effort for Certification process(z) in (P/H)	Effort for Development (D_{RUCM}) in (P/H)	Total Effort for Development (E_{RUCM}) in (P/H)
1	48	1	1	1	3000	4881
2	52	1	1	1	3000	5033
3	58	1	1	1	3000	5261
4	63	1	1	1	3000	5451
5	78	1	1	1	3000	6021
6	90	1	1	1	3000	6477
7	105	1	1	1	3000	7047

Table-3 illustrates that in RUCM, Effort is needed in person/hour for mathematical Modeling and quantification, statistical testing and for certification process for any stage, because these all characteristics are also available in RUCM and also the Effort required to complete the process for development in RUCM is 1.25 % greater as compared to CRM.

4. Conclusion

The overall comparison effort for the development of project by all of these three models, the tables show that the RUCM is efficient as compared to RUP, because all the best features of CRM are also used in development for the project and the deficiencies found in RUP related to documentation improvement, effort required for the development and quality of the product are also have been improved Via RUCM. It has been also observed that, the effort is

always dependent on development time, here the development Effort for CRM, RUP and RUCM is represented in the given relation

$$D_C < D_{RUCM} < D_R$$

and effort for the development of project can also be represented on the basis of these tables as

$$E_C < E_{RUCM} < E_R$$

References

- [1] P. Kruchten, "Rational Fellow 'What Is the Rational Unified Process?'"', Rational Software Canada, (2001) January.
- [2] G. Pollice, "RUP and XP, part I: Finding Common ground", Evangelist, The Rational Unified Process, Rational Software, (2001).
- [3] G. Tattersall, et. al., "Supporting Iterative Development Through Requirements Management", (2002) October.
- [4] C. Phillips and E. Kemp, "In Support of User Interface Design in the Rational Unified Process", Institute of Information Sciences & Technology Massey University, (2001).
- [5] P. Murphy, "A Review of Cleanroom software Engineering", (1996).
- [6] H. Ananthpadmanabhan, et. al., "Cleanroom Software Development", Arizona State University, (1996).
- [7] C. M. Wolak, "Taking the Art out of Software Development An In-Depth. Review of Cleanroom Software Engineering", (2001) April.
- [8] B. Ahmad, et. al., "Exploring Documentation: A Trivial Dimension of RUP", Computer Engineering and Intelligent Systems ISSN 2222-1719 (Paper) ISSN 2222-2863 (Online), vol. 2, no. 6, (2011).

