International Journal of Computer Trends and Technology (IJCTT) - volume4Issue4 –April 2013

The Issues with Reengineering of Agile System's Assets

Asfa Praveen[#1], Bilquis Fatima [#2], Shamimul Qamar[#3], Shahanawaj Ahamad[#4]

[1 & 2] *Ph.D. (Computer Sc.) Research Scholars, Faculty of Science & Technology, Shri Venkateshwara University, Gajraula, (U.P.), India.*
[3]*Professor of Electronics & Computer Engineering, Noida Institute of Engineering & Technology, Greater Noida, (U.P.), India.*
[4]*Assistant Professor, Dept. of Comp. Sc. & Software Engg, College of Computer Sc. & Engg., University of Ha'il, K.S.A.*

***Abstract*— Currently the abilities of companies have to be improving to compete the market challenges, so their all assets are under consideration for up-gradation including human resources, structures, and computer based system, technically here focus is on software system which have become agile because emerging requirements and functional changes have been incorporated continuously to fulfill the demands and ultimately it has itself become an issue. Reengineering has come up with a useful and stable tool for managing and realizing these up-gradations, but with the issues and risks that accompany traditional process of reengineering and having a question whether it can achieve the desired benefits. This paper addresses several issues which should be undertaken while the system is under up-gradation with long term maintenance; this paper focuses on the risks and issues of reengineering of agile software assets as code, data and structure with its implementation challenges.**

Keywords: **Agile, CORBA, Code, Reengineering, Redesign, Process.**

I. INTRODUCTION

Agile system's usability, stability, maintainability and throughput can be improved by using reengineering of it because many other requirements have forced to keep it in the updated direction of evolution because business and its process are continuously adopting the dynamically changing environment, so there must be a faithful and effective strategy to be chosen for proactive, dynamic information infrastructure that must have strength to support both changes in the business environment and technologies.

Reengineering approaches are very effective to rejuvenate I.T. based infrastructure in order to survive in a very competitive age because with the invention of new form of technologies, there has been a need to redesign and re-implement existing systems.

Redesign and restructuring of have some of the floating issues and challenges while using existing system as much as possible, these methods are to transform the existing system into the format needed for the adoption of new technologies, instead of make waste or left away the old heritage system, these kind of processes are obviously give benefits with supported applications of semi-automated, automated tools.

II. LITERATURE REVIEW

Recent business process applications developments have been widely influenced by CORBA [1], DCE [2], for e-commerce it is needed to access wide range data and to manage and buildup MIS and DDS [3]. Many organizations are purchasing new information systems to take benefits of the advanced technologies, longer established enterprises have to modify pre-existing system to fit new requirements and exploit emerging technologies so this is a challenging task because many systems have developed over several generations of programmers and containing only obsolete or no documentation, such system and applications are classified agile systems [3].

Recently the researchers have been increasing efforts to develop methods, processes and concepts reengineering, redesign agile system. To reduce the challenges and complexities some of the methods have been implemented in computer-aided reengineering (CARE) tools to automate risky activities. The existing data structure, databases are the core part of an agile systems [4], most methods targets on database schema analysis [5], [6] with reverse engineering and schema translation and redesign with forward engineering [7],[8].

III. Issues With Agile Code

There are several issues while reengineering the agile code, which have been listed as follows.

1) It is hard to maintain and enhance the functionalities of most of agile code, especially if the code is written in formal languages such as C, COBOL and FORTRAN;
2) Rewriting the code needs huge time and money with inputs of expert programmers;
3) There are a lot of duplications in the code and it's functionalities;
4) Agile code lack to get the advantages of use of advanced processors, and distributed computing features;
5) Most agile code cannot tackle memory leak problems including dynamic memory allocation and de-allocation;
6) Conditional statements: "If-then-else" and "switch" statements are major issues as Object-Oriented program design try to remove the use of switch control structure;
7) Short of abstractions: it is hard to use it to implement the proper abstractions for under short term maintenance system;
8) Lack of future scope of updating: These codes are having very low scope for future up-gradation and possibilities to enhance features of concern.

COBOL is a language of large syntax rules, a widely known issue in specifying COBOL syntax is that the many syntactic variations with many of the same semantics, if the size of the COBOL code definitions is reduced then it is giving easiness and better performance but depends on the grammar free programming style. There are so many conflicts in the understanding the semantics and syntax of COBOL code due to their connection and associations with other code segments, the issues with the priority of code functionalities and syntax rules are handled by COBOL system defined formats. However there are many remaining conflict cannot be handled which are context dependent as given in the example code that the X = Y OR Z depending on the status of Z the root of the parse tree is either = or OR, as in fig. 1. When the parse tree will be drawn on the basis of this then it depends on whether Z is a numeral or a predicate, also depending on the values of X, Y, and Z so ultimately depends on the value of Z.

III. Structural & Architectural Issues

Agile code architecture is based on structure of system that consist of small components and other observed features of components with representing relationship among them [9]. Architecture is a basic structural organization to which system is incorporated in its components, interrelationships; the relations to their environment, and the principles guiding its design and evolution for the components [10]. System architectures are main abstractions at various levels, so system architectures enable various entities to handle with the complexity of system and the architectures are a means to incorporate the a variety of views of code structures [11].

```
00001  IDENTIFICATION DIVISION.
00002  PROGRAM-ID. XisYorZ.
00003  DATA DIVISION.
00004  WORKING-STORAGE SECTION.
00005  01 X PIC 9 VALUE 1.
00006  01 Y PIC 9 VALUE 2.
00007 * 88 Z VALUE 2.
00008  01 Z PIC 9 VALUE 2.
00009  PROCEDURE DIVISION.
00010  MAIN.
00011    IF X = Y OR Z
00012      DISPLAY "s"
00013    ELSE
00014      DISPLAY "f".
00015
00016
00017 *       X | 1 | 1 | 1 | 1 |
00018 *       Y | 2 | 2 | 2 | 2 |
00019 *       Z | 2 | 1 | 2 | 1 |
00020 *  -------|---|---|---|---|
00021 *  output | f | s | s | f |
00022 * comment | 7 | 7 | 8 | 8 |
```

(Fig.1: Example COBOL Code)

An architectural framework is discussed which is alternate of the famous systems architecture of Zachman [12], he described systems architecture as a collection of architectural models in two dimensions; first as description type and other as perspective, are also challenges for them. The famous perspectives are agree with the benefits and interests of different persons in a system development effort as system manager, developers and designer's view may differ completely from both. Each of the participant's views is, however may be relevant to develop the system successfully.

There are following issues for different perspectives:

1) The business perspective: This is for the fulfillment of the objective of the organization;
2) The logical perspective: This supports and focus on the logical aspects of system with it intended functions;
3) The technical perspective: This is intended for the achievement the desired functionalities and ultimate objectives of the systems;
4) The infrastructural perspective: This is for hardware and software related issues that required for the efficient function and execution of system.

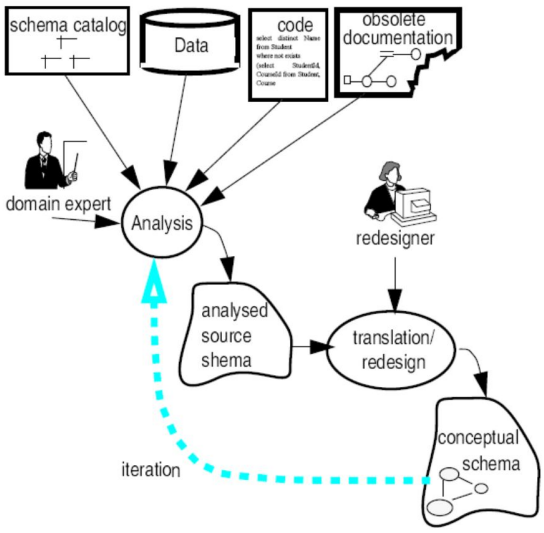
The second dimension of the risks involves the different types of descriptions which can make of a system. These types are applied for many perspectives, study suggested the general 'what', 'how' and 'where' questions as the important types of challenges. If distinguishing the data, the functions, and the process type of descriptions on the one hand and components on the other hand so it can find inclusive management of the most important issues and challenges:

1) Data model: This is for definitions of the relevant entities or objects;
2) Function model: This is in concern of the on a description of the services and function;
3) Process model: This describes how data and functions are incorporated into network;
4) Component model: This presents object's hierarchy and encapsulation to remove the system complexity.

IV. REENGINEERING

The DoD organized a workshop in 1992 to define reengineering terminologies, the workshop defined software reengineering as which can be perceived and presented in the perspective of systems as it is an investigation, examining and modification of a subject system to redesign, reframe, reconstitute it in a new form reimplementation of the new form beside this a general view of reengineering presented by Robert S. Arnold [14], a typical reengineering environment is shown in Fig.2, commonly reengineering is an umbrella activity [15] based on technologies as:

1) Reverse engineering: This is to understand the system functionality from the source code and interfaces, applications;
2) Forward engineering: This is for general engineering process and implementing the design information from back steps;
3) Re-documentation: This is to recover the vanished documentation of the system by semi-automated tools from source code;
4) Restructuring: This is for redesign of system architecture remains unchanged the business logic and functions;

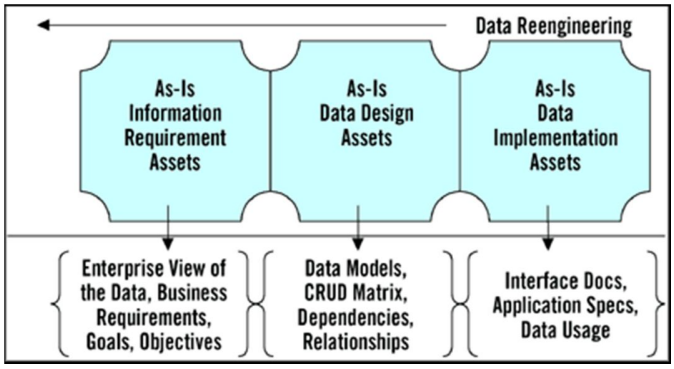


(Fig. 2: Reengineering environment)[9]

5) Retargeting: Migrating the system to a modern infrastructure, platform or environment;
6) Source code translation: This is for program language translation to other target language;
7) Database reengineering: Re-architect the database and files.

V. DATA REENGINEERING

At the time of agile reengineering, data reengineering is a crucial task, it include data analysis, understanding documentation, models, data structures, this task is highly depends on the system interaction with other modules and flow of information with better understanding of the meta-data, this process has to ensure the recovery and preserving of data organization and availability. Fig. 3 depicts reengineering oriented classification focusing exclusively on data assets.



(Fig. 3: Data Reengineering) [17]

VI. The Risks & Challenges to REengineering

This section of investigation reports the major challenges, because of them why most agile system base organizations are not ready or feeling risky to turn to reengineering options, generally it is because of track record of reengineering projects is not very admirable. Poor planning, understanding and missing applications of many tools are major reasons behind failure reengineering projects [16], so no organization is ready to face the risk. On the other hand if the existing agile system is being replaced by a newly reengineered or developed from information base based on existing system and any small error occur at the time of functioning or operation them the user and manager become unsatisfied because major reason of it is they are highly practical with old system and not fitting well to learn new thing of function but it is reality which is a major concern many time to resolve.

The identified risks are too high but based on studies and investigations the reengineering tasks may face some of the risks as follows:

1) Cost / benefit analysis to justify the return of system cost;
2) Incremental reengineering for massive systems;
3) Integration with business process engineering;
4) It is targeting the more common users;
5) Integrating heterogeneous tool sets.

Studies identified some major challenges as:

1) In upgrading the business processes technical support and infrastructure;
2) Developing a target architecture based on business and technical reviews;
3) Changes in development and maintenance processes;
4) Increasing the future growth in system;
5) Minimize processing time;
6) Minimize the operating costs with use of cheaper hardware & software;
7) To lower development costs and faster system development time;
8) To increase in user interest, GUI & database features;
9) Faster response to fulfillment of market demands;
10) Up gradation of existing hardware;
11) Increasing in revenues, more satisfied user and operation of services.

VII. Conclusion

It is a reality that due to the emergence of new technologies, the existing software system may lose its intended efficiency and become a question for maintenance and updating and then after some time it become classified as agile assets. This paper presented a study to highlight some of the critical technical issues and challenges based on user input, opinions and cost implications that if these issues are not considered well to resolve at the time of reengineering projects for agile systems then it can degrade the desired results and intended functions.

References

[1] S. Vinoski; “CORBA: Integrating diverse applications within distributed heterogeneous environments”. IEEE Communications Magazine, 14(2), Feb.1997.

[2] Open Software Foundation; “Introduction to OSF/DCE’, Prentice Hall, New Jersey, 1992.

[3] Amjad Umar; “Application (Re)Engineering Building Web-Based Applications and Dealing with Legacies”, Prentice-Hall International, London, UK, 1997.

[4] P. Aiken; “Data Reverse Engineering: Slaying the Agile Dragon”, McGraw-Hill, 1995.

[5] O. Signore, M. Loffredo, M. Gregori, and M. Cima; “Reconstruction of ER schema from database applications: A cognitive approach”, in Proc. of 13th Int. Conference of ERA, Manchester, pages 387–402, Springer, 1994.

[6] Himel Inc; President Drive, Castro valley, CA 94546, USA, DBInformer User’s Manual, 1997.

[7] F. Husemann, et.al; “Datenbank Gateways”, in 10. Workshop "‘Grundlagen von Datenbanken"’, pages 52–56, Konstanz, June 1998. Konstanzer Schriften in Mathematik, Informatik Nr. 63, Universität Konstanz.

[8] J. Fong; “Converting relational to object-oriented Databases”, ACM SIGMOD Record, March 1997.

[9] Shaw, M. and D. Garlan; “Software Architecture: Perspectives on an Emerging”, Discipline, Prentice Hall, 1996.

[10] IEEE-2000; “Recommended Practice for Architectural Description of Software-Intensive Systems”, IEEE Standard 1471, 2000.

[11] P. Kruchten; "The Rational Unified Process: An Introduction", Addison Wesley, 2000.

[12] J.A. Zachman; "A Framework for Information Systems Architecture", IBM Systems Journal, Vol. 38, No. 2-3, pp.454-470, 1999.

[13] J. Fong and S. Huang; "Information Systems Reengineering", Springer Verlag, ISBN 981-3083-15-8, 1997.

[14] R.S. Arnold; "Software Reengineering", IEEE Computer Society Press, Los Alamitos, CA, 688 pp., 1993.

[15] R. G. Waters and E. Chikofsky; "Reverse engineering: progress along many dimensions", Communications of the ACM, 37(5), 22–25, 1994.

[16] R. E. Johnson; "Software Reengineering Assessment Handbook", Version 3.0,Washington DC, Hill Air Force Base, Ogden UT, 1997.

[17]http://www.information-management.com/infodirect/20050225/1020767-1.html.

AUTHOR'S BIOGRAPHY

Asfa Praveen has six years of experience with good practical, academic and research projects exposures after completion of three years Master of Computer Applications (M.C.A.) degree in year 2007 from Punjab Technical University, Jalandhar with very good grades; Advanced 'A' level (P.G.) Diploma in Computer Science in year 2003 from Department of Electronics, Ministry of I.T., Govt. of India; Oracle Certified Professional (O.C.P.) Examination in year 2003 from Oracle Corporation, U.S.A.; she is currently pursuing Ph.D. in Computer Science from Faculty of Science & Technology of Shri Venkateshwara University, Gajraula, (U.P.), her area of research includes Service Oriented Reengineering of Monolithic Legacy Software.

Bilquis Fatima has received Bachelor of Science (B.Sc.) from Ch. Charan Singh University, Meerut followed by three years Master of Computer Applications (M.C.A.) degree from Punjab Technical University, Jalandhar in year 2011 with very good grades, currently she is full time research scholar for Ph.D. in Computer Science in Faculty of Science & Technology of Shri Venkateshwara University, Gajraula, (U.P.), her area of research includes Software Reengineering and Rejuvenation.

Prof. (Dr.) Shamimul Qamar has sixteen years of wide experience in research, academics and administration, held various positions as Director, Professor, Consultants in universities and engineering colleges after completion of Ph.D. in Electronics and Computer Engineering from Indian Institute of Technology (I.I.T.) Roorkee with excellent grade; he has completed B.Sc. from Ch. Charan Singh University, Meerut; Bachelor of Engineering (B.E.) in Electronics & Communication Engg. from Madan Mohan Malviya Engineering College, Gorakhpur in the year 1996; M.Tech. (Information & Communication Systems) from Aligarh Muslim University, Aligarh; he has published more than 35 research papers in his credits and supervised many master projects and Ph.D. thesis; currently he is designated as Professor of Electronics and Computer Engineering in Noida Institute of Engineering and Technology, Mahamaya Technical University, Noida, U.P. (Delhi-N.C.R.), India.

Dr. Shahanawaj Ahamad is an active academician and researcher in the field of Computer Science, Software Reverse Engineering with twelve years of research and academic experience including five years in abroad, working with College of Computer Science & Engineering of University of Ha'il, K.S.A. before joining UoH he has worked with King Saud University, Al-Khraj University of K.S.A. and Shobhit University, Meerut (Delhi-NCR) and Uttar Pradesh Technical University of INDIA as HoD-I.T., Asstt. Professor etc. He is the professional member of British Computer Society, U.K., senior member of Computer Society of India including membership of various national and international academic and research organizations, member of research journal editorial board and reviewer. He is currently working on Service-Oriented Migration, Multi Agent System Reverse Engineering, published more than twenty five research articles in his credit in national and international journals and conference proceedings. He holds M.Tech. followed by Ph.D. in Computer Science with specialization in Software Engineering from Jamia Millia Islamia Central University, New Delhi, India. He has supervised many bachelor projects, master and Ph.D. dissertations.