ISSN 1746-7659, England, UK
Journal of Information and Computing Science
Vol. 8, No. 4, 2013, *pp*. 256-263

Square model is better than Spiral Model to improve the process of IVR Software

Devesh Kumar Srivastava

Manipal University Jaipur

(*Received February 24, 2013, accepted June 30, 2013*)

Abstract. Software process improvement (SPI) has emerged as the dominant approach for improving quality and productivity in software development organizations. It is an important activity which starts when an organization plans to enhance/purify the capabilities of its on-going processes. When improvement or change is planned or initiated, there are a number of process-improvement-factors which originate and affect the effectiveness of software process improvement. Therefore the biggest challenge is to find a route for appropriate SPI technologies to realize their company's improvement goals. A software Industry focused on two different research fields work on the issues of software process: first - software process modeling and second - software process evaluation and improvement. In this paper, the most relevant results of second approaches are presented to evaluate the process improvement for development of IVR software using Square Model. This paper also suggests to IVR application software companies that require collaboration and strengthening to transform their current perspective into inseparable global IT scenario.

Keywords: Interactive Voice Response (IVR), Voice Response Software (VRS), Line of code (LOC), Computer Telephony Integration (CTI), Business Process Outsourcing (BPO).

1. Introduction

Currently Process Improvement is widely used in software industry. This activity starts when the deficiencies are identified in the current processes and finished when a certain satisfactory or defined level is attained. It is being used in order to improve the quality, productivity and organizational on-going processes [1]. Currently, most of the organizations have limited resources everywhere in the world. If full resources and complete process may use then quality of the product can better improved in order to exist in the market with the passage of time [2]. SPI involves the understanding of the software processes as they are used within an organization and suggests areas for improvements in achieving specific goals such as increasing product quality, operation efficiency and cost reduction [3, 8]. A process is known to be mature if they pass at least five levels. These five levels of process maturity are [5]:1. Initial - until the process is under statistical control, no orderly progress in process improvement is possible.2. Repeatable - a stable process with a repeatable level of statistical control is achieved by initiating rigorous project management of commitments, cost, schedule, and change.3. Defined - definition of the process is necessary to assure consistent implementation and to provide a basis for better understanding of the process. At this point, it is probable that advanced technology can be usefully introduced.4. Managed - following the defined process, it is possible to initiate process measurements. This is where the most significant quality improvements begin to appear.5. Optimized - with a measured process, the foundation is in place for continuing improvement and optimization of the process. The most popular reasons for introducing SPI are - improving software quality due to best-practices; reducing costs; and reducing timescales. It is fact that the use of best-practices in a software project development improves productivity in organizations, projects planning and the quality of the software products. Currently, they are deployed because of the broad usability of project management tools. However they present several constraints to manage and transfer the knowledge of these best practices. The knowledge is obtained through the software engineering experts' experiences. Based on the use of best –practices and knowledge management with the help of square model the process of IVR software may be improved better than use of spiral model. Currently the IVR software is in huge demand by BPO industries to make inbound –outbound calls among the customers and product information. This is based on Gensys technology and IVR technology. In this paper we use the square process model for better improvement of IVR software [7]. The main capabilities of this square model are: to reuse and manage of process assets; to

improve the efficiency of use of the processes; to reduce costs in software process improvement programs; to work collaboratively in the phases of software projects. It has been observed that while developing the IVR software the iterative and prototype software model are being followed with risk management. The components are being tested through manual testing and testing by tools but the interactive voice testing are not tested at each component level. Therefore it is advisable to test the interactive voice testing at each component level to avoid the fault and failures during development of IVR application by square model.

2. Existing Models

The purpose of this research work is to give a very brief introduction to some of the most commonly recognized SPI models like SPICE,SW-CMM, CMMI, ISO 9000 - 9001,Trillium, BOOTSTRAP and Six Sigma. Based on the results of preliminary study on the literature, several studies related to frameworks for measurement and evaluations of SPI were identified. The studies are briefly discussed in the paragraphs below.

A. Capability Maturity Model

The process was developed by the Software Engineering Institute at Carnegie Mellon University in 1986 The Capability Maturity Model for Software is a model for judging the maturity of the software processes of an organization and for identifying the key practices that are required to increase the maturity of these processes. Problems typically reported with the CCM when used by these organizations were: Documentation overload, Unrelated management structure, High resource requirements, High training costs, Lack of need guidance, Unrelated practices The SW-CMM has been developed by the software community with stewardship by the SEI [14].

B. Capability Maturity Model Integrated CMMI-SW

Capability Maturity Model Integrated for Software builds on and extends the best practices of the Capability Maturity Model for Software [5]. CMMI is based on the SW-CMM and could be considered as version 2.0 of the SW-CMM. While some new information has been added in this version of the CMM, most of the old information have been reused -simplified and extended.

C. SPICE

SPICE stands for Software Process Improvement and Capability Determination. It provides a reference model for focused self-assessments and includes a capability scale that is simple to understand. The main objective of SPICE is to provide a framework for the assessment of software processes and to decide if the performance of the process/processes is satisfying and if the processes are effective in achieving their goals, and to distinguish and determine the capability of the process [29]. The result of the assessment is analyzed to determine the weaknesses, strengths and risks of the process. This can be used as a base for process improvement. The definition of the term “process assessment” is “A disciplined evaluation of an organization's software processes against the process model or variant model described in this International Standard.

D. Bootstrap

Bootstrap is a methodology that originated in a European Community project that took place between 1991 and 1993. The project consisted of developing the Bootstrap model and stage 60 trials in the industry. Since the projects ended Bootstrap has been further developed. The main goal with the project was to speed up the application of software engineering technology in the European software industry. In a related article, Stienen described the main characteristics of the BOOTSTRAP method [30]. These included the reference framework, the assessment procedure, the structure of the questionnaires, and the rating and scoring mechanisms employed. The BOOTSTRAP method adopted a process model which addresses processes and practices for both the software producing unit and the project. Process areas were divided into organization, methodology, and technology.

E. Six Sigma

“ó” (sigma) is a Greek letter that stands for standards deviation – a measure of dispersion, variation or spread [6]. Six Sigma is a methodology for eliminating defects, waste, or quality control

problems that originated at Motorola in the early 1980's. Key features of the methodology are; statistical quality control techniques, data analysis methods, and systematic training of people in the organization that is affected or targeted by Six Sigma. Six Sigma is a data driven methodology that addresses a variety of business activities such as manufacturing and management.

F. ISO 9000

ISO 9000 consists of a number of standards related to quality management systems and related supporting standards. It was created by the ISO. The ISO 9000 family is there to represent "an international consensus on good quality management practices". Issues that are covered in the standards include [31]:

- Procedures for key business processes
- Processes monitoring
- Keeping records
- Defect control and preventive techniques
- Review of specific processes and measuring effectiveness
- Continual improvement

G .ISO 9001

ISO 9001 is an international standard for quality assurance in design, development, production, installation, and service. It is broken down into twenty elements. ISO 9001-3 relates to the development, supply, and maintenance of software. Almost 90 percent of the companies that completed ISO 9001 implementation reported improved internal documentation as one of the most important benefits of registration. Other benefits included higher product quality, greater internal quality awareness, and increased competitive advantage [35].

H. Trillium

The Trillium model was initially designed for use with embedded software systems which is based on the CMM. Trillium is comprised of five levels (1-5). These are unstructured, repeatable and project oriented, defined and process oriented, managed and integrated, and fully integrated [36]. Trillium can be used in a number of ways. For example it can be used to benchmark an organization's product development process against industry best practices or to self-assess and identify opportunities for improvement.

3. Research Method

Software Process Improvement is a difficult activity to initiate because of its complex and changing nature of processes. It is intensive and time consuming activity which consumes a lot of time and efforts of the responsible team. The effectiveness from SPI shows the success or failure of this activity. No matter, what the size of the organization is but we need to take care of many other activities and factors which are directly or indirectly dependent on SPI. Organizations with fewer resources need better process improvement and profitability measures. All activities must be run in a structured and systematic way so that other ongoing activities must not get disturbed. Once process improvement is initiated properly, it shows positive aspects in the end results in the shape of improved processes and better profit margin. Organizational structure, road maps, assessed methods and a good plan lead towards successful results of process improvement [2]. Customer satisfaction, continuous improvement and less staff turnover show the strong business profitability aspects of an organization. One major characteristic of process improvement is to emphasize the continuous improvement of products as well as of organizational processes in terms of performance, stability, compatibility [19]. By doing so, a good quality can be achieved; products can be developed in low cost and high productivity. After the application of process improvement the organization feels the competitiveness, increase in performance, profitability and innovations in the processes which shows the benefits taken from the successful processes. Process improvement focuses towards the development of the practices, improved quality of the products, reliability, productivity, and customer and employee satisfaction. The change occurs in the shape of good staff, improved technical system, organized structure and better management practices. Software process improvement always facilitates to identify and apply the changes to current processes so that the new processes can be helpful in producing the high quality product. With the advancement in the software requirements and technology, we intend to make a change to

meet the demands of the user and technology. This change leads towards highly competitive market because there is always some higher organization ready for competition. Hence, improvement leads towards continuous change. Software process improvement is an effective way to improve product quality, meet market needs, and reliability.

3.1 Process improvement at component level for call (voice) testing

To develop the software applications, software professionals use the life cycle of software engineering which end at system testing level. After it voice testing is performed against calls. By following these strategies, faults and failures are more created as usual. When IVR software is developed using square model, it is found that the fault and failures are very less than earlier traditional methods. Therefore it is advisable to use square model in development of IVR software to make process better than earlier method. In this figure the IVR software life cycle is shown at each component level. After deployment of component the call is made through phone system and each call is recorded to monitor the status of all calls. Each call is verified that how many calls are true and how many calls are false. Its validly is checked.

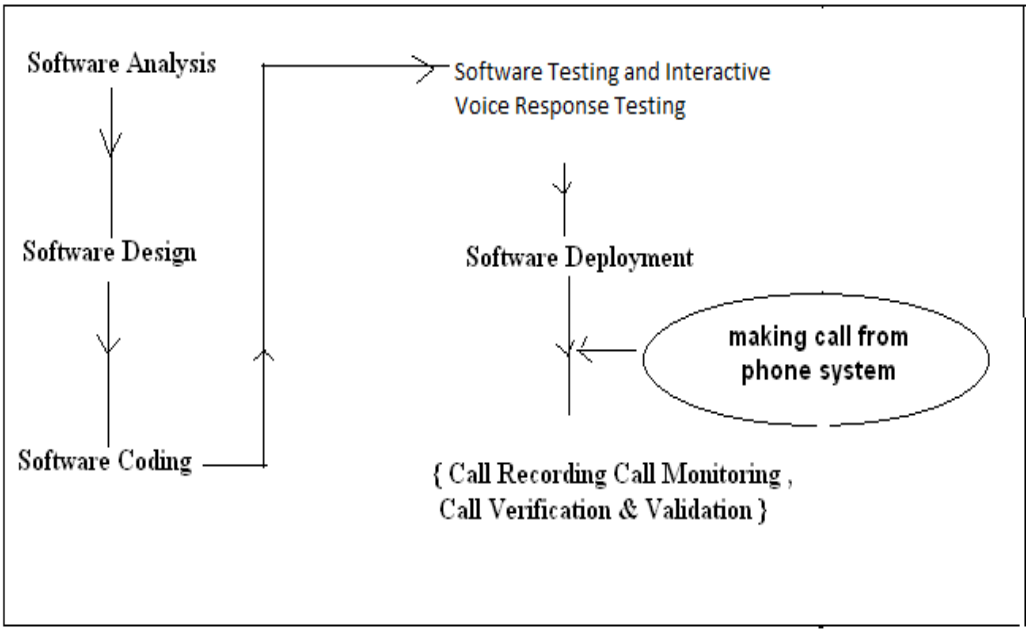

Fig 1: IVR Software life Cycle covering risk at each level

For process improvement following objectives should be applied to build IVR application software.

a. A detailed understanding of software process improvement.
b. Better understanding of change in an organization.
c. Importance of change in process improvement activities.
d. Understanding how change can be beneficial and effective during software process improvement.
e. Identification of factors involved during organizational change in Software Process improvement.
f. Understanding of systematic review.
g. Conducting a systematic review to identify and prioritize the factors affecting software process improvement change.

3.2 Effects of results using Spiral Model versus Square Model

Table 1: Collection of data

Project No.	Time (Month) using Spiral Model	Time (Month) using Square Model
1	14.0	13.0
2	16.0	14.6
3	17.0	16.0
4	20	18.0
5	23.8	21.9
6	26	25.0
7	10.20	9.8
8	19	17.5
9	8.0	7.0
10	12	11.5

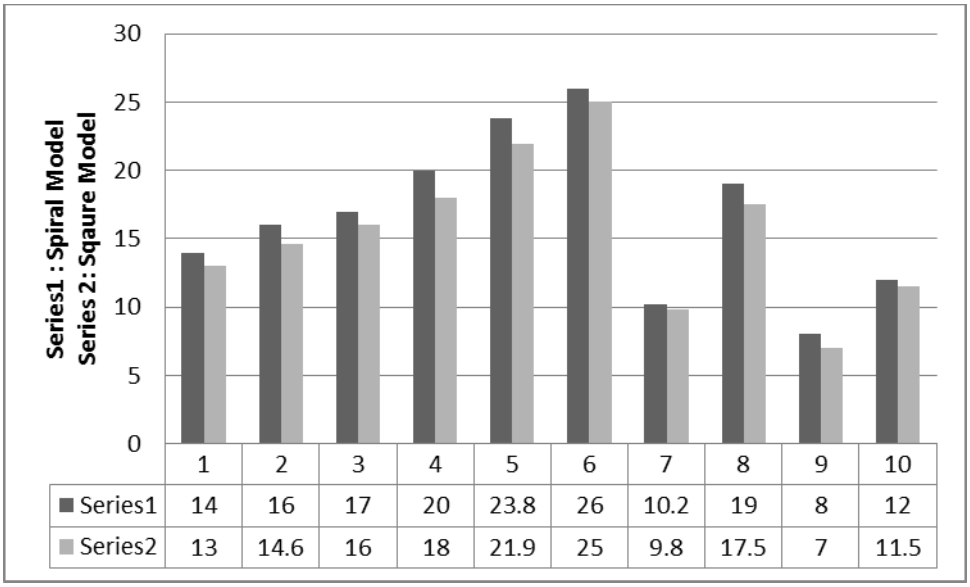

Fig 2: Effects of curve of Spiral versus Square Model using column chart

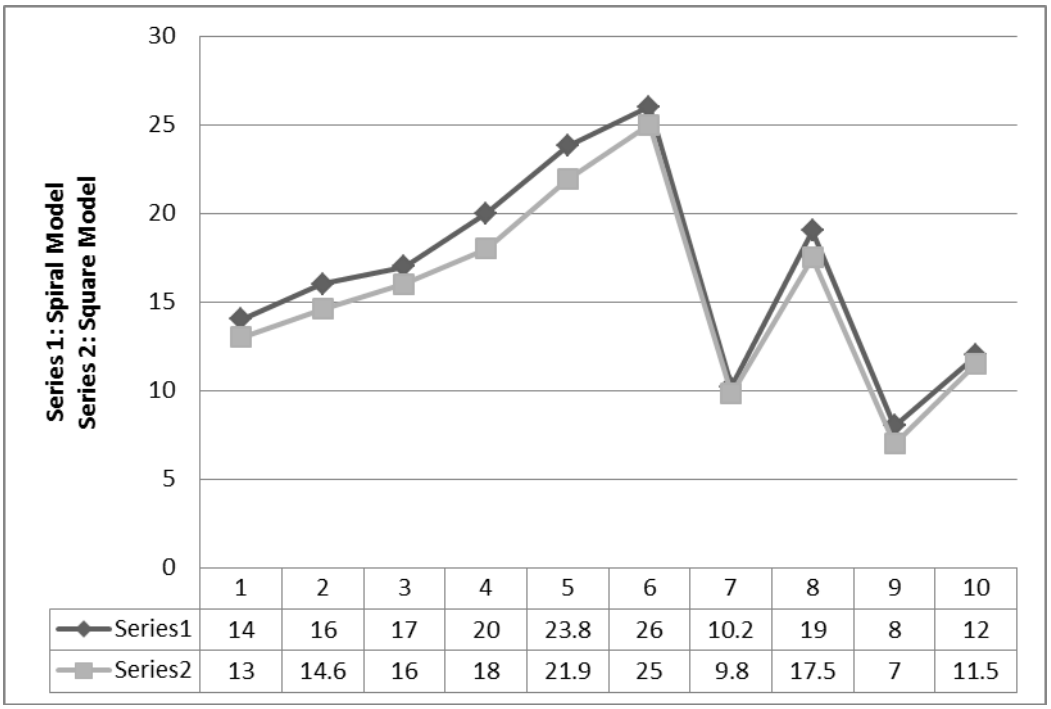

Fig 3: Effects of curve of Spiral versus Square Model using line chart

From Figure 2 and Figure 3, it is found that curve of Square model is few inclined than curve of Spiral model. When both model were applied it is empirically found that while Using Spiral Model:

a. Much time is taken at system level.
b. It may possible alpha version software produce lot of errors at voice testing level. Usually voice may not recognized by IVR software. In such case reverse engineering and reengineering is applied to find out the issues to test the calls successful for IVR software.
c. Total Bugs found is more
d. System Complexity is major while testing.
e.

Using Square model:

a. Much time is taken at each component level.
b. Less time is taken at system level testing.
c. Few chances to get the software failure.
d. Quality is better improved than earlier.
e. System Complexity is minor while testing.

4. CONCLUSION

Improvements introduced by SPI efforts are, however, measured through informal and non-objective processes, based on the employees' perception and not through formal measurement processes. There are a smaller percentage of other improvement proposals which have as a focal point the definition, assessment and support of the software process. On the basis of above results, it is finally concluded that Square Model is better than Spiral Model to develop IVR software. It has been suggested from present study that such IVR software must use the square model to improve the process improvement that effort is aligned with corporate business and technical objectives.

5. References

[1] Ashrafi, Noushin., The impact of software process improvement on quality: in theory and practice, Elsevier, 2002.

[2] Allison, I., Merali, Y., Software process improvement as emergent change: A structurational analysis. ELSEVIER, 2007.

[3] A.R.M. Reddy, P. Govindarajulu, M. M. Naidu, A Process Model for Software Architecture, International Journal of Computer Science and Network Security, 2007, VOL.7, No. 4.

[4] Bill C. Hardgrave and Deborah J. Armstrong (2005), Software Process Improvement: It's a Journey, Not a Destination, Communications of ACM, Vol. 48 (11), pp. 93-96.

[5] Capability Maturity Model Integration (CMMI), Version 1.1 (Staged Representation), Pittsburgh, PA: Carnegie Mellon Software Engineering Institute, 2002.

[6] C. Redzic and J. Baik, "Six Sigma Approach in Software Quality Improvement," in *Fourth International Conference on Software Engineering Research, Management and Applications* 2006, pp. 396-406.

[7] Devesh Kumar Srivastava, Durg Singh Chauhan, Raghuraj Singh ,Square Model- A Software Process Model for IVR Software System. - International Journal of Computer Applications Volume 13– No.7, 33-36 January 2011 New York, USA.

[8] Dyba, Tore.,An Empirical Investigation of the Key Factors for Success in Software Process Improvement, vol. 31, no. 5, IEEE, May 2005.

[9] Emam, Khalid el, James, Herbsleb, James, *Success or Failure? Modeling the Likelihood of Software Process* Improvement, International Software Engineering Research Network technical report ISERN-98-15.

[10] E. Emam, J. Drouin,W.Melo, SPICE: The Theory and Practice of Software Process Improvement and Capability Determination, IEEE Computer Society Press, 1998.

[11] Hannay, J. E., Sjøberg, D. I. K., & Dyba, T. A systematic review of theory use in software engineering experiments. IEEE Transaction on Software Engineering, 33(2), 87–107. 2007.

[12] Ita Richardson and Christiane Gresse von Wangenhelm, Why are small software organizations different?, IEEE Software, January/ February 2007.

[13] Iversen, Jakob., Ngwenyama, Ojelanki, Problems in measuring effectiveness in software process improvement: A longitudinal study of organizational change at Danske Data., ELSEVIER, 2006.

[14] J. Herbsleb, D. Zubrow, D. Goldenson, W. Hayes, and M. Paulk: Software Quality and the Capability Maturity Model, *Communications of the ACM*, Vol. 40, No. 6 (June 1997), 30-40.

[15] Jones, Lawrence G,Software Process Improvement and Product Line Practice: Building on Your Process Improvement Infrastructure., SEI, 2004.

[16] Jørgensen, M., & Shepperd, M. J. (2007). A systematic review of software development cost estimation studies. IEEE Transaction on Software Engineering, 33(1), 33–53.

[17] Kathleen Coleman Dangle, Patricia Larsen, Michele Shaw, and Marvin V. Zelkowitz, Software Process Improvement in Small Organizations: A Case Study, IEEE Software, November / December 2005.

[18] Kusters, Rob J., Trienekens Jos, J.M., *On the Business Impact of Software Process Improvement*, Proceedings of the 26th Annual International Computer Software and Applications Conference IEEE, 2002.

[19] K. El Emam, D.R. Goldenson, J. McCurley, and J. Herbsleb, Modeling the Likelihood of Software Process Improvement: An Exploratory Study, Empirical Software Eng., vol. 6, no. 3, pp. 207-229, 2001.

[20] M. Biro´, R. Messnarz, and A.G. Davison, "The Impact of National Cultural Factors on the Effectiveness of Process Improvement Methods: The Third Dimension," ASQ Software Quality Professional, vol. 4, no. 4, pp. 34-41, 2002.

[21] Mathiassen, Lars, Ngwenyama, Ojelanki K, Aaen, Ivan,Managing Change in Software Process Improvement. IEEE 2005.

[22] N. M. A. Munassar, A. Govardhan, Comparison Between Five Models Of Software Engineering, International Journal of Computer Science Issues, 2010, Vol. 7, No. 5.

[23] ISO9000-3: Guidelines for the application of ISO9001 to the development, supply and maintenance of software, International Standard, 1991.

[24] Rainer, Austen. Hall, Tracy. A quantitative and qualitative analysis of factors affecting software processes. Elsevier March 2002.

[25] Rainer, Austen, Hall, Tracy, Key success factors for implementing software process improvement: a maturity-based analysis, Elsevier, august 2001.

[26] Richardson, I. (2001). Software process matrix: A small company SPI model. Software Process: Improvement and Practice, 6(3), 157–165.

[27] Software Engineering Institute (SEI).The Capability Maturity Model: Guidelines for Improving the Software Process. Addison Wesley Longman, 1999.

[28] Substrate interfaces (Translation Journals style)," *IEEE Transl. J. Magn. Jpn.*, vol. 2, Aug. 1987, pp. 740–741 9th Annual. Conference. Magnetics Japan, 1982, p. 301.

[29] SPICE: The Theory and Practice of Software Process Improvement and Capability Determination, K. El Emam et al., eds., CS Press, 1998.

[30] Stienen, H., Software Process Assessment and Improvement: Five Years of Experiences with BOOTSTRAP. In K. Emam & N. Madhavji (Eds.), *Elements of Software Process Assessment and Improvement* (pp. 57-75). Washington, DC: IEEE Computer Society Press 1999.

[31] Stelzer, D., Mellis, W., & Herzwurm, G. (1996) Software Process Improvement via ISO 9000? Results of Two Surveys among European Software Houses," Proc. 29th Hawaii Int'l Conf. Systems Sciences.

[32] T. Dyba, A Dynamic Model of Software Engineering Knowledge Creation, Managing Software Eng. Knowledge, A. Aurum et al., eds., Berlin: Springer Verlag, pp. 95-117, 2003.

[33] T. Dyba, "Factors of Software Process Improvement Success in Small and Large Organizations: An Empirical Study in the Scandinavian Context, Proc. Joint Ninth European Software Engineering Conference and 11th SIGSOFT Symposium Foundations Software Engineering (FSE-11), pp. 148-157, 2003.

[34] Tore Dyba, An empirical Investigation of the key factors for success in software process improvement, IEEE Transactions on Software Engineering, vol. 31, no.5, May 2005.

[35] Weissfelner, S., ISO 9001 for Software Organizations. "*Elements of Software Process Assessment and Improvement* Washington" 1999.

[36] Zahran. , Software Process Improvement: Practical Guidelines for Business *Success*. Reading, MA: Addison-Wesley 1988.