International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 7 | ISSN : 2456-3307

Sustainability of Software Process Maturity

Muhammad Awais*[1], Izaz Khan*[1], Anum Zahoor[1], Muhammad Ikram[1], Hussain Khan[1]
*[1]Department of Computer Science, Abdul Wali Khan University Mardan, KPK, Pakistan

ABSTRACT

A lot of work has been done on Process improvement but still organizations facing a number of problems during software process improvement. For getting process maturity the organizations should make much effort. This paper includes a literature study of different articles about software process, process improvement, and sustainability and how we will achieve sustainable process. We investigated the issues and difficulties occur during process improvement and then identified different factors to improve the software process. Moreover, in this paper we pointed out problems during process improvement and described some suggestions for achieving sustainable software process.

Keywords : Software Process, Software Improvements, Sustainable Process, Software Maturity, Software Process Model.

I. INTRODUCTION

Software process is a set of activities and resources which changes the specified input to output. Resources mean techniques, facilities, equipment and finance etc. another definition of software process as it is collection of activities like methods, practices and transformation which are used by the people for maintenance and development of software and associated products like test cases, designing documents, code and manuals[10]. There are many software processes but the following activities are common to all process [14].

- Software Specification: this specifies the functions of software and constraints on it.
- Software design and implementation: software is developed and installed.
- Software validation: software is tested and checked whether it fulfills the customer requirement or not.
- Software evaluation: different types of changes are made to satisfy the customer.

Software process improvement means bringing of changes in the existing process after it has been analyzed for improvement i.e. to minimize the cost and increase the quality of product [14]. Software process is very complex and has different characteristics. It is not easy to optimize all the characteristics by improving the software process. These characteristics or attributes are understandability, visibility, supportability acceptability, reliability, robustness, maintainability and rapidity. Process improvement is a cyclical activity and has three basic stages [14]:

- Process measurement: attributes of the current project are measured
- Process analysis: the current process is analyzed and weaknesses and bottlenecks are identified
- Process change: in this stage changes to the existing process is implemented and improved the process.

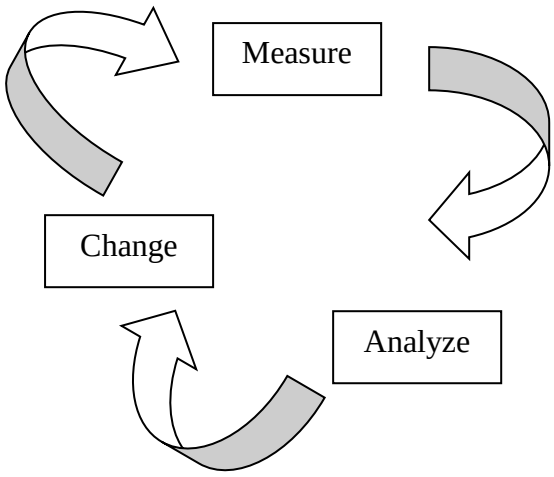

Fig 1. The process improvement cycle [14]

Process change stage in software improvement is an important and difficult one as many organization encounter problems while implementing changes because they did not manage change process well [11]. Furthermore, process change has five sub stages [14].

- Improvement identification: the result of the analysis stage is studied and identified different problems, and set some goals and then applying some new tools, methods and procedures to remove the problems.
- Improvement prioritization: there may be number of changes to be implemented so, in this stage the changes are implemented according to their priority.
- Process change introduction: the introduction of new methods, tools and technique but should be made sure that these compatible with organization standards and the process as well.
- Process change training: training is necessary as without training full success is not possible.
- Change tuning: in this stage minor problems are identified and modification is proposed and introduced.

Software process model: a software process model is the abstract representation of software process [1]. Process represented by a software model is different from others software models as every software model represents the process in his own perspective. There is no ideal software model as every model has his own pros and cons and can be selected according to the project nature i.e. complexity of the project, size of the project and deadline for the project etc. there are different models used for process improvement for example CMM, CMMI, ISO 9000, SPICE and Trillium etc.

Software process improvement is composed of two steps. After identifying the organization process needs and goals then a process model or methodology is selected. To combine these two it is necessary to carefully analyzed and examine the situation and then combine the two. This is represented by the following Fig.

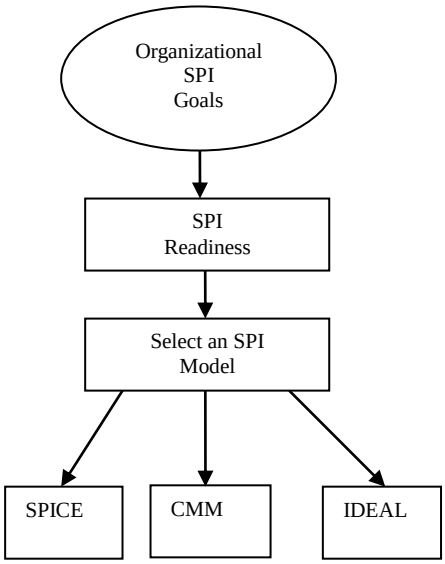

Fig 2. [13].

II. RESEARCH METHODOLOGY

We collected different papers on process improvement, process maturity and sustainable process and then selected twelve most relevant papers for our research. All these papers were downloaded from ACM, IEE and Willy InterScience.

Then We thoroughly studied Research papers and noted what efforts and steps have been taken for software process improvement and how much they got success so far, and what type of problems have been faced during Software Process Improvement. Our research work is composed of four sections. In section 3 we discuss various success factors for process improvement and section 4 having common problems with Software Process Improvement. After that in section 5 we mentioned some factors for sustainable process likewise, in section 6 there are some suggestions for sustainable process and finally i.e. section 7 contains conclusion.

III. STATE OF THE ART

Researcher worked a lot on software process to minimize the errors in software process and to get the product which meets the requirements of the customers and several approaches has been adopted for this purpose. They developed models for setting goals and priorities of the organizations to achieve software process improvement. Organizations assessments are also done by these models and frameworks. Some of these frame works are CMM, SPICE, ISO-9000, Trillium and Bootstrap [5] [14].

Detailed description of the Success Factors in Process Improvement

CMM: Software Institute of Engineering (SEI) developed capability maturity model (CMM) in the mid of eighty's. The SEI just initiated a way for judging the software contractors. CMM provides a standards to assess the software capability and maturity for software process development. CMM has eighteen key process areas (KPA's) which are divided into five levels. These levels are given in the figure [1], [5].

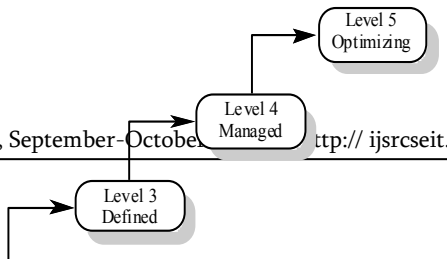

Fig 3:Level of optimization

In the above figure it is shown that Process improvement is achieved going from level 1 to level 5. These levels have been discussed shortly below.

Initial level: organization does not have any organized and disciplined process and can use any technique and method.

Repeatable level: in this level organizations have schedule management and cost procedures.

Defined level: this level has a standard software process for organizations as process for engineering and management activities are standardized and completely documented.

Managed level: through KPA's product quality and process management is achieved in this level.

Optimizing level: continue process improvement strategies are present in this level.

For achieving a level in CMM it takes one or two years so it is time consuming secondly quantitative measurement is placed in the fourth level while the measurement is important activity in the process improvement and should be kept in the initial levels. Let us take an example of DataStream Content Solution (DSCS) which got software process improvement by implementing CMM. DSCS assigned dedicated persons to relevant tasks and secondly, by taking periodic review of the process i.e. there were monthly meeting about project progress and

discussion on different issues and what have been done and what are the progress and what would be done, all this make a good software process improvement. Furthermore, there were standards for process and terminology as well the top management divided their authority in subordinates thus it reduce workload on top management and others also started interest and took their responsibilities. After all these initiatives defined in CMM DSCS got software process improvement [1].

CMMI: It is the abbreviation for integrated capability maturity model (CMMI). In this model SEI attempted to integrate different models including CMM. It overcomes some of the weaknesses in the CMM. There are two versions of CMM, staged and continuous. Staged version is Compatible with CMM and the continue version has 24 process areas and divided into five categories [5], [1].

ISO-9000-3: Detailed quality guidelines have been given in this part for development and maintenance of software products. There is also interpretation of ISO 9000 for software organizations. It not only provides guidelines for development activities but also for quality activities. Includes guidelines for quality practices in both development aims primarily to establish an acceptable baseline for software process. ISO-9000-3 mainly focuses on specific software development.ISO-9000 certification cannot provided by internal auditors [5].

SPICE Model: It means Software Process Improvement and Capability determination (SPICE), aka ISO-15504, and it is an International standard for software process assessment. SPICE is more flexible from assessment point of view then SEI models. This model supports other models and allows the organizations to keep continuity without changes in the previous set up. SPICE has a reference model of practices and an assessment method but this model presents much details and giving good assessment for software process improvement [5].

BOOTSTRAP: It is a methodology which is used for software process assessment and improvement and developed in a European Community project (ESPRIT). The basic purpose was evaluation of investments in technology. The basic purpose of bootstrap was to introduce the technology in the small organizations. Process management issues are address through this methodology. Furthermore, BOOTSTRAP provides assessment of the practices and then to analyze the assessment result it provides specific tools and methods. For improving the process maturity the BOOTSTRAP help to change the assessment result into o the action [5].

IV. Common problems during software process improvement

Software engineers introduced new methods, technologies and process models for SPI but when organizations tried to implement these new models and methodologies they faced a number of problems which were obstacles to get the required SPI. We are going to discuss in this section some of the hurdles occurred during the SPI.

- Inter organizational instability: it is necessary for process improvement. Organizational changes affect the SPI badly. These changes may be in the top management or may be in subordinates i.e. programmer, developers or even the changes can be introduced to the structure of the organization i.e. to restructure the organization. Then there will be some SPI measures related to the previous set up so by changing that set up the SPI is affected. There is example of a company Raytheon who took eight years for maturity while other organization like Telecordia get maturity very quickly comparable to Raytheon although it is a large organization. It is noted that the reorganization of Raytheon affected the SPI of Raytheon. Paulish and Carleton also reported the problem of reorganization within Siemens [2].

- Intra organizational instability: Another key factor which affects the SPI is intra organizational instability. The example of intra organizational instability is staff turnover which has been noted by researchers. Paulish and Carleton discussed that during SPI implementation in Siemens the staff turnover caused problems [2].
- The Data stream content solutions (DSCS) faced various problems during software process improvement. These are discussed here. They had to do more training and arrange workshop and information sessions because they realized that the resources which were assigned different tasks of software improvement their knowledge and expertise were not sufficient that is why there was a delay in software process improvement. Other problem was that they did not ensure that the resources were seriously involved. There should be a formal review to judge the progress of the resources [1].
- Krasner and Ziehe (1995) have discussed problems at software process improvement initiatives of US American software suppliers. They found that the following problems were the major challenges to the Software Process Improvement: Shortage of software engineers and expert managerial staff; lack of commitment and realistic goals; assigning high priority to software improvement, paying not much attention to change process and importantly lack of will to implement the change and little progress in learning (Krasner and Ziehe 1995), [11].

V. Achieving sustainable process

Sustainable process achievement cannot be archived easily because for sustainability organizations need to carry on their efforts and focus on the weaknesses they have in software improvement process. During our literature study we found some problems while organization were trying to get SPI therefore, concentrating and avoiding those problems can bring sustainability to the software process. There are some points by which sustainability can be achieved.

As we discussed in section 4 that inter organizational and intra organizational instability and organizational changes affect the SPI. Therefore by avoiding organizational, business changes where the management should pay their concentration to prevent organizational and intra organization instability for getting sustainability [1], [2].

We observed that most of organizations suffered a lot due to the lake of knowledge, expertise in process change, training and overrunning budget and schedules [11]. Therefore, by the arrangement of workshop, training session, discussion, making schedules on ground reality and moreover, by staff professionalism and organizational recruitment policies like in the SERVICECO, sustainability can be achieved [3], [11].

- Some organizations like DSCS, Telecordia Technologies, faced problems as their employees were not committed and some of them were trying to resist the changes for the improvement because they were not included in the all process and they worried about their positions. Hence by including all the employees in the process and make sure that all resources are committed and devoted. Furthermore, it is also confirm that all resources are involved in the process. These measures can bring software sustainability [11].

VI. Analysis

In this we have given two sections, parts one is analysis of the result and the second part contains information about sustainable process.

A. Analysis of the results

According to our study various organizations tried to improve software process by adopting CMM, CMMI, SPICE model, ISO-9000-3, BOOTSTRAP and Agile methodology etc. Many organizations like Siemens, Telecordia Technologies, and Data Stream Contents solutions faced problems during SPI, these problems occurred because of business organizational changes, instability, insufficient training, and lake of interest, expertise and knowledge and management etc. these problems were obstacles in achieving software improvement. Hence avoiding the above mentioned mistakes and weaknesses the process stability may be achieved [2], [1].

Our suggestions for sustainable software processes

Software sustainability is a laborious task and organizations must keep their struggle to keep it up. They should regularly arrange workshop and training sessions for their employees, so that all employees will be up to date and able to accept any challenge for sustainability.

Software organizations should allocate a separate budget for getting sustainability because organizations have to pay continuously for training of employees and will hire experts and trained persons and good management people as well [12-32].

Software sustainability is a challenge for software industry. The researchers should focus on this area of software. Researcher should explore new methods, techniques and tools for software sustainability.

VII. CONCLUSIONS

After, studying fifteen research papers about process improvement, factors for getting software process improvement, problems occurred during software process improvement and sustainable process achievement. We noted that SPI started in 1980,s and much of the contribution has been done by SEI and CMM was a milestone for SPI but still it is not always possible to get a sustainable process and we concluded that the criteria for SPI is changing regularly so, S.E must pay much concentration to SPI and a lot of efforts is required in this direction.

VIII. REFERENCES

[1]. Khan, F., ur Rehman, A., Usman, M., Tan, Z., & Puthal, D. (2018). Performance of Cognitive Radio Sensor Networks Using Hybrid Automatic Repeat ReQuest: Stop-and-Wait. Mobile Networks and Applications, 1-10. https://doi.org/10.1007/s11036-018-1020-4

[2]. Alam, M., Trapps, P., Mumtaz, S., & Rodriguez, J. (2016). Context-aware cooperative testbed for energy analysis in beyond 4G networks. Telecommunication Systems. doi:10.1007/s11235-016-0171-5

[3]. Khan, F., Rahman, F., Khan, S., & Kamal, S. A. (2018). Performance Analysis of Transport Protocols for Multimedia Traffic over Mobile Wi-Max Network Under Nakagami Fading. In Information Technology-New Generations (pp. 101-110). Springer, Cham.

[4]. Alam, M., Albano, M., Radwan, A., & Rodriguez, J. (2013). CANDi: context-aware node discovery for short-range cooperation. Transactions on Emerging Telecommunications Technologies, 26(5), 861–875. doi:10.1002/ett.2763

[5]. Khan, F., & Nakagawa, K. (2013). Comparative study of spectrum sensing techniques in cognitive radio networks. In Computer and Information Technology (WCCIT), 2013 World Congress on (pp. 1-8). IEEE.

[6]. Alam, M., Mumtaz, S., Saghezchi, F. B., Radwan, A., Rodriguez, J. (2013). Energy and Throughput Analysis of Reservation Protocols

of Wi Media MAC. Journal of Green Engineering, 3(4), 363–382. doi:10.13052/jge1904-4720.341

[7]. Alam, M., Yang, D., Huq, K., Saghezchi, F., Mumtaz, S., & Rodriguez, J. (2015). Towards 5G: Context Aware Resource Allocation for Energy Saving. Journal of Signal Processing Systems, 83(2), 279–291. doi:10.1007/s11265-015-1061

[8]. Jan, M. A., Nanda, P., He, X., & Liu, R. P. (2014). PASCCC: Priority-based application-specific congestion control clustering protocol. Computer Networks, 74, 92-102.

[9]. Jan, M. A., Nanda, P., He, X., & Liu, R. P. (2015, August). A sybil attack detection scheme for a centralized clustering-based hierarchical network. In Trustcom/BigDataSE/ISPA, 2015 IEEE (Vol. 1, pp. 318-325). IEEE.

[10]. Jan, M. A., Nanda, P., He, X., Tan, Z., & Liu, R. P. (2014, September). A robust authentication scheme for observing resources in the internet of things environment. In Trust, Security and Privacy in Computing and Communications (TrustCom), 2014 IEEE 13th International Conference on (pp. 205-211). IEEE.

[11]. Jan, M., Nanda, P., Usman, M., & He, X. (2017). PAWN: a payload-based mutual authentication scheme for wireless sensor networks. Concurrency and Computation: Practice and Experience, 29(17).

[12]. Jan, M. A., Nanda, P., He, X., & Liu, R. P. (2013, November). Enhancing lifetime and quality of data in cluster-based hierarchical routing protocol for wireless sensor network. In High Performance Computing and Communications & 2013 IEEE International Conference on Embedded and Ubiquitous Computing (HPCC_EUC), 2013 IEEE 10th International Conference on (pp. 1400-1407). IEEE.

[13]. Khan, F. (2014, May). Fairness and throughput improvement in multihop wireless ad hoc networks. In Electrical and Computer Engineering (CCECE), 2014 IEEE 27th Canadian Conference on (pp. 1-6). IEEE.

[14]. Jan, M. A., Nanda, P., He, X., & Liu, R. P. (2018). A Sybil attack detection scheme for a forest wildfire monitoring application. Future Generation Computer Systems, 80, 613-626

[15]. Jan, M. A., Nanda, P., & He, X. (2013, June). Energy evaluation model for an improved centralized clustering hierarchical algorithm in WSN. In International Conference on Wired/Wireless Internet Communication (pp. 154-167). Springer, Berlin, Heidelberg.

[16]. Usman, M., Jan, M. A., & He, X. (2017). Cryptography-based secure data storage and sharing using HEVC and public clouds. Information Sciences, 387, 90-102.

[17]. Usman, M., Jan, M. A., He, X., & Nanda, P. (2016, August). Data sharing in secure multimedia wireless sensor networks. In Trustcom/BigDataSE/I SPA, 2016 IEEE (pp. 590-597). IEEE

[18]. Khan, F., Khan, M., Iqbal, Z., ur Rahman, I., & Alam, M. (2016, September). Secure and Safe Surveillance System Using Sensors Networks-Internet of Things. In International Conference on Future Intelligent Vehicular Technologies (pp. 167-174). Springer, Cham.

[19]. Usman, M., Yang, N., Jan, M. A., He, X., Xu, M., & Lam, K. M. (2018). A joint framework for QoS and QoE for video transmission over wireless multimedia sensor networks. IEEE Transactions on Mobile Computing, 17(4), 746-759.

[20]. Khan, F., ur Rahman, I., Khan, M., Iqbal, N., & Alam, M. (2016, September). CoAP-Based Request-Response Interaction Model for the Internet of Things. In International Conference on Future Intelligent Vehicular

Technologies (pp. 146-156). Springer, Cham.

[21]. Fida, N., Khan, F., Jan, M. A., & Khan, Z. (2016, September). Performance Analysis of Vehicular Adhoc Network Using Different Highway Traffic Scenarios in Cloud Computing. In International Conference on Future Intelligent Vehicular Technologies (pp. 157-166). Springer, Cham.

[22]. Jan, M. A., Khan, F., Alam, M., & Usman, M. (2017). A payload-based mutual authentication scheme for Internet of Things. Future Generation Computer Systems.

[23]. Usman, M., He, X., Lam, K. K., Xu, M., Chen, J., Bokhari, S. M. M., & Jan, M. A. (2017). Error Concealment for Cloud-based and Scalable Video Coding of HD Videos. IEEE Transactions on Cloud Computing.

[24]. Yang, N., Usman, M., He, X., Jan, M. A., & Zhang, L. (2017). Time-Frequency Filter Bank: A Simple Approach for Audio and Music Separation. IEEE Access, 5, 27114-27125.

[25]. Usman, M., Jan, M. A., He, X., & Alam, M. (2018). Performance evaluation of High Definition video streaming over Mobile Ad Hoc Networks. Signal Processing, 148, 303-313.

[26]. Jabeen, Q., Khan, F., Hayat, M. N., Khan, H., Jan, S. R., & Ullah, F. (2016). A Survey: Embedded Systems Supporting By Different Operating Systems. International Journal of Scientific Research in Science, Engineering and Technology (IJSRSET), Print ISSN, 2395-1990.

[27]. Usman, M., Jan, M. A., He, X., & Nanda, P. (2018). QASEC: A secured data communication scheme for mobile Ad-hoc networks. Future Generation Computer Systems.

[28]. Jan, M. A., Tan, Z., He, X., & Ni, W. (2018). Moving Towards Highly Reliable and Effective Sensor Networks.

[29]. Jan, M. A., Jan, S. R. U., Alam, M., Akhunzada, A., & Rahman, I. U. (2018). A Comprehensive Analysis of Congestion Control Protocols in Wireless Sensor Networks. Mobile Networks and Applications, 1-13.

[30]. Alam, M., Ferreira, J., Mumtaz, S., Jan, M. A., Rebelo, R., & Fonseca, J. A. (2017). Smart Cameras Are Making Our Beaches Safer: A 5G-Envisioned Distributed Architecture for Safe, Connected Coastal Areas. IEEE Vehicular Technology Magazine, 12(4), 50-59.

[31]. Jan, M.A., Usman, M., He, X., &, Rehman, A.U. (2018). SAMS: A Seamless and Authorized Multimedia Streami ng framework for WMSN-based IoMT. IEEE Internet of Things Journal, doi: 10.1109/JIOT.2018.2848284

[32]. M. A. Jan, S. R. Jan, M. Usman, M. Alam. (2018), State-of-the-Art Congestion Control Protocols in WSN: A Survey, IoT EAI, DOI: 10.4108/eai.26-3-2018.154379