

Role of Software Configuration Management in Software Engineering

^[1]Aruna Malik

^[1]Department Of Computer Science and Engineering, Galgotias University, Yamuna Expressway Greater Noida, Uttar Pradesh

^[1]aruna.malik@galgotiasuniversity.edu.in

Abstract- Software configuration management is a quite essential functionality in the field of software engineering and in certain areas of reviewing. Software Configuration Management is known as the law for monitoring the development of conceptual software structures. Also it needs to support software activities performed using process as a simple job for professional software development. SCM is commonly utilized, and promotes technology and procedures that allow monitoring of changes and distribution of versions. Modern Software Configuration Management concept is monitoring the development of multiple structures. It is the regulation that helps everyone to continue to develop the scheduling element within supervision, and to meet the criteria of consistency and pause. The International Organization for Standardization defines configuration management to be one of functional fields of system and system control, including error monitoring, performance monitoring, reporting monitoring, and protection strategic planning as well. The paper gives a comprehensive overview on SCM, its activities, its tools followed by its challenges.

Keywords- Activities of SCM, Challenges of SCM, SCM (Software Configuration Management), Tools of SCM.

INTRODUCTION

SCM is the regulation of complicated system development. More rationally, it is the methodology that helps everyone to keep in monitor changing software products, thereby adding to the satisfaction of performance limitations and delays. SCM arose as a specialty shortly after the so-called software problem is established, i.e. until it is realized that scripting does not encompass everything within Software Engineering (SE), and other problems hindered SE growth, such as design, construction, development, etc. The phase of evolution of software engineering is marked by constant adjustments[1]. A team of people usually produce, alter, and share popular and specific parts of the program, operating towards a common purpose. The target is not always just a specific static object but a complex array of elements intended to operate together. Not all configurations will end in a computationally efficient unit, and the set is sometimes made up of a wide number of models, with many people managing and modifying them

at various sites; the whole cycle of creation is often a constant background of modifications, changes and enhancements. SCM is the methodology where emerging software systems are structured, regulated, and managed. The aim of using SCM is to ensure the integrity of a software entity and to make its development more bearable; such goal is accomplished by defining the software configuration at specific times and consistently monitoring modifications to the configuration defined in order to sustain software authenticity, quality control and transparency across the existence of the software. The use of Software Configuration Management obviously means an increased expense over time, money, and other elements of the software lifecycle. It is generally accepted, nevertheless, that the implications of not using SCM will cause many difficulties and shortfalls[2]. For any design function, SCM is essential, namely project management, web browsing, database activities, hypertext composing, software-aided design and production,

**International Journal of Engineering Research in Computer Science and Engineering
(IJERCSE)
Vol 5, Issue 4, April 2018**

servers, and other implementations where information is regularly inserted and updated. In the software engineering and software-aided development repositories, SCM is especially examined. Software configuration management is a methodology for handling the creation of software engineer goods, both in the initial design phases and in all servicing stages.

Documentation- A Documented design aids in constructing the object, evaluating sections and sorting them that give them access in some framework.

Control- Managing the delivery of an object and making changes throughout the lifespan by setting up control to ensure consistent programming via the creation of a gauge product.

Status Accounting- Uploading and documenting section status and changing requirements, as well as key social event perspectives on element sections.

Audit and Review- Recognizing the execution of an element and preserving continuity between the parts by checking that the product is an all-around collection of elements regarded. The SCM process is explained below in Fig. 1 Software Configuration Management

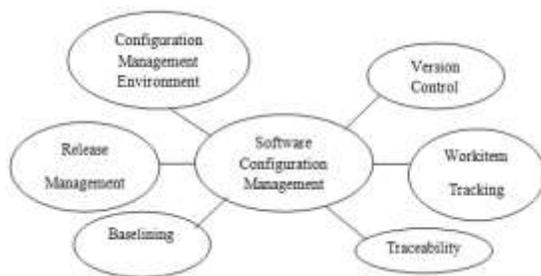


Fig.1: Software Configuration Management

Functions of SCM:

Controlling a collection of Components- The different elements of a software package and all the variants need to be stored in a safe manner. Such focus covers model control, software design and handling complicated objects[3].

Assist Technicians in the daily activities- Software Engineering involves the application of resources to items (documents). SCM devices aim to provide right entities to technicians at the right place. It is often related to as power over space. Collection and monitoring of derivative entities is a big problem.

Process Monitoring and Support- It became apparent that a key issue, if not the one, affects people. Generally, change management is an essential part of the SCM commodity; at present the trend is to expand the capacity of system and beyond.

Participants of SCM:

The various Participants of SCM are shown below in Fig. 2 Participants of SCM

Configuration Manager- Configuration Manager is the head accountable for defining configuration objects. CM maintains team implements the SCM cycle. Requirements for changes need to be accepted or denied.

Developer- The designer has to adjust the code as per normal design practices or demands for modification. He is in charge of keeping software setup. The designer will test the adjustments and settle disputes.

Auditor- The auditor is liable for reviews and assessments carried out by SCM. Need to make sure launch is consistent and full.

Project Manager- Guarantee that the product is created within a certain period Track the ongoing development and identify problems in the SCM system. Generate updates on the computer system progress. Make sure procedures and practices are implemented to build, update, and check.

User- The consumer will grasp the key words of the SCM and insure they have the new software update.

**International Journal of Engineering Research in Computer Science and Engineering
(IJERCSE)
Vol 5, Issue 4, April 2018**

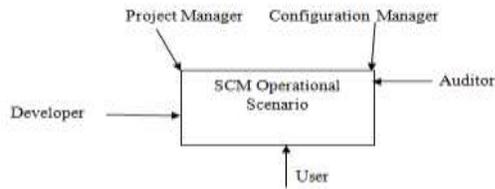


Fig.2: The Figure Portrays the Participants of SCM

SCM ACTIVITIES

System Configuration Management (SCM) is an activity arrangement that regulates progress by identifying objects for shift, establishing relations between objects for alter, creating / classifying equipment for monitoring various variations, regulating adjustments in the existing framework, examining and disclosing / reporting improvements made[4]. It is important to monitor the adjustments in view of the fact that if the modifications are not properly verified then they may end up undermining a well-run scheduling. The various activities involved in SCM are-

Recognition and Establishment- Defining the specification objects from goods that constitute parameters at the particular time points (a reference is a collection of generally compatible specification objects that have been systematically evaluated and decided upon and that form the basis for more growth.

Version Control- Creation of current product versions / requirements to create new goods with the assistance of SCM scheme. The current trend is to make a clear distinction between the structures (for instance, the list of divisions and modifications) and the nature of the variants; the objective is to cover the structures of the lower level (sects and modifications). One of the benefits of the update set method is that the client does not have visibility into how iterations are processed, he / she understands only the conceptual adjustments.

Change Control- A change request (CR) will be requested and reviewed to determine the technological validity, possible side effects, total effect on other design objects and device operations and the change's estimated price. The assessment results are viewed as a progress document that is used by a change "control board

(CCB)" — a person or organization that makes a decision about the change's nature and priority. For each authorized change an "Engineering Change Request (ECR)" is made. CCB will also alert the applicant in event the request is denied for clear reason[5]. The ECR defines the shift to be created, the limitations to be complied with and the examination and audit requirements. The object to be modified is "checked out" of the server of the venture, the changes are made and then the entity is again evaluated.

Configuration Auditing- The systematic technological analysis of the method and product is complemented by software configuration review. It relies on the functional consistency of the modified specification entity. The review checks the comprehensiveness, accuracy, and integrity of SCM program objects, and monitors action plans from review to completion.

Reporting- Providing programmers, testers, consumers, clients and shareholders with correct progress and existing setup information through user manuals, user guidebooks, FAQs, Update Notes, Memoranda, Setup Manual, and Setup Guide etc. The Process of SCM Activities is shown below in Fig. 3 SCM Activities

**International Journal of Engineering Research in Computer Science and Engineering
 (IJERCSE)
 Vol 5, Issue 4, April 2018**

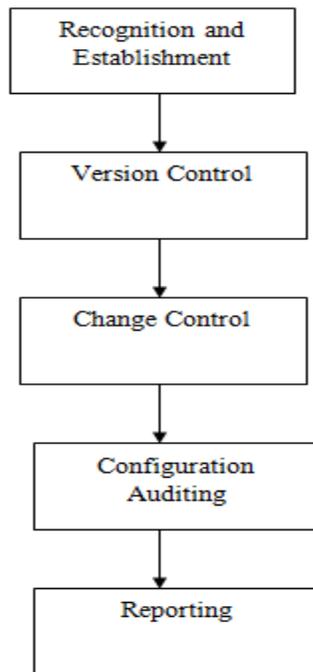


Fig.3: The Figure Portrays the SCM Activities

SCM TOOLS

The various tools for SCM are-

SVN (Subversion) - It is the most influential open source configuration management tool used for adjustment and regulate. Being free software, it is freely available through the web. It supports nuclear requirement, which ensures that if the origin phase is intruded in the centre, it will only come into effect once the whole commitment has been successful[6].

VSS (Virtual Source Safe) - Is a Windows Concentrated Type Control System. North Carolina Company developed it called One Tree Technology. All the store data is stored in Microsoft SQL Server to ensure the data is professional. Because it is designed in C++ and Sql, it has a very insignificant system specification. It is possible to implement Software in 32 bit and 64 bit.

GIT- Git is a distributed adjustment peripheral device created by "Linus Torvalds" to support Linux's

development. Git was first used in 2005 and Linus gave the first conference with software for Linux. GIT is an effective and versatile tool. The decentralized version management system takes a peer-to-peer method to version control as compared to the decentralized network customer database method. It has many useful features and divisions can be easily generated and combined. GIT is a common continuous integration tool where everybody has their own database and between designers it is possible to pull and move improvements.

Perforce- Perforce is a prominent tool in the theoretical community, perhaps in view of the fact that the company gives the system to allow public access practices. Perforce is recognized for its frank construction layout and unique propagation proof that progresses towards the root of the tree rather than internally[7].

BENEFITS OF SCM

One of the most important sources of inspiration for implementing such a SCM system is the need for details in the software to be updated in two specific countries. Such a SCM is needed as a result of software development at take places. Previously, more open door for production of unrealistic capacities, comparable to the potential to have variance leakage for dual-stage products and to create cross-sectional variations over them and have competencies close to design coverage for specific customers[8]. Luminosity in the state of product development and maintenance activities, and in relation to all progressions, knowledge stuff; background of all things, for instance, who made huge changes, where, how, why a transition. Healthier estimation of the date of release on the basis of period of the migration process is measurable and therefore more disconcerting. Improve leverage inquiry to further assess the volume of exertion required in a change, and then how long it will take. Less germs in recorded discharges multiple problems found in discharges could also been held back if SCM had been established.

SCM CHALLENGES

It defines SCM in reclusion; SCM systems are too often homogenous, and restricted their capacity to modularize. Things changed drastically over the last few years; SCM

**International Journal of Engineering Research in Computer Science and Engineering
(IJERCSE)
Vol 5, Issue 4, April 2018**

methods are one of a rising number of other methods, and are thus tested in a number of elements[9].

Functional Challenges:

The first task is to answer the above-mentioned questions including discovering or bringing complex concepts and frameworks into action.

- Software template, with dynamic entities, specific associations, documents.
- Namespaces, monolithic for all entity and interaction types.
- Specification control, automated choice and compatibility requirements.
- Reconstruction, dual-platform, elevated-level, effective formalism
- Work areas, decentralized, modular, diverse and independent
- Parallel architecture with high-level prototypes and scalable approaches.
- Numerous-view process management, hybrid solutions and high-level designs.

All of the above-mentioned issues are daunting by nature, but recognizing that a big constraint nowadays is linked to performance is of vital importance. Verifying or marking a system takes seconds to weeks, restoring takes days to days; complex choice or management of simultaneous equipment greatly slowed down the job, etc. Computational time is a major burden in software engineering and professionals do not consider any further slowing down. The main challenge is finding extremely efficient, accessible, effective and flexible, innovative, versatile and stylish approaches. A great number of software and frameworks maintain a regional, often versioned database of data. Usual instances provide an area for coding, or 4GL tools[10]. The challenge is to find some way to inter-operate all these databases so as to prevent (too much) redundancy and ambiguity. The main problem is making the several varying definitions, layouts, and versioning methods consistent (enough). It is a difficult problem for a long time. The trend is to include ever more programs, and to tackle (try to) more areas. Distributed software has been implemented in recent times, then mobile production, then change command,

then process help, then Internet promote, and so forth. This is a sign of progress, but also a troubling one. The task is to find a new SCM architecture that allows you to split the frameworks into individual bits, and design the SCM framework from the functionality that is currently required.

CONCLUSION

Software Configuration Management contains both the operational and the administrative functions. Rather large amounts of factors are involved in the new situation during the lifecycle of software development, without the implementation of SCM software becoming an impossible job. SCM is a way to handle various objects in the program. Now the users' expectations for a day are evolving very quickly, streamlined to satisfy these different procedures. The choice of SCM software depends on the specifications of a specific company. There is not the need to use all the functionality of a SCM tool; most SCM systems offer some level of personalization. It is uncertain what the potential of SCM analysis and software is. The essential services will become comprehensible, established and secure enough yet to optimize. These would collapse into the public sphere, as essential services that everyone can demand from a network, such as versioning, restoring, vendor support for work area, etc. Manufacturers may lose control over those facilities of reduced level. SCM tools will continue to grow, suggest tool chains and continue to view SCM as an independent field. SCM analysis must take this opportunity that many SCM problems are already being worked on in other Software Engineering domains, to promote collaboration between these analysis spheres and SCM, putting in the experience and know-how that has made SCM solid, providing a way to the valuable and effective software development requirements of tools. The paper gives the overview of Software Configuration Management, its roles, activities, advantages and disadvantages.

REFERENCES

- [1] B. Summers, "Software Configuration Management Plan," in *Software Engineering Reviews and Audits*, 2011, pp. 201–218.
- [2] L. P. Taylor, "Developing a Configuration

**International Journal of Engineering Research in Computer Science and Engineering
(IJERCSE)
Vol 5, Issue 4, April 2018**

-
- Management Plan,” in *FISMA Compliance Handbook*, 2013, pp. 153–165.
- [3] S. S. Mohd Fauzi, P. L. Bannerman, and M. Staples, “Software Configuration Management in Global Software Development: A systematic map,” in *Proceedings - Asia-Pacific Software Engineering Conference, APSEC*, 2010.
- [4] S. S. M. Fauzi, A. J. Suali, and W. A. W. M. Sobri, “A State of the Art: Software Configuration Management Tools for Global Software Development,” in *Journal of Physics: Conference Series*, 2018, vol. 1049, no. 1.
- [5] A. Bartusevics and L. Novickis, “Models for implementation of software configuration management,” in *Procedia Computer Science*, 2015.
- [6] A. Bartusevics, L. Novickis, and S. Leye, “Models and Methods of Software Configuration Management,” *Appl. Comput. Syst.*, vol. 17, no. 1, pp. 53–59, 2015.
- [7] H. Saleem and S. M. A. Burney, “Imposing Software Traceability and Configuration Management for Change Tolerance in Software Production,” *IJCSNS Int. J. Comput. Sci. Netw. Secur.*, vol. 19, no. 1, pp. 145–154, 2019.
- [8] S. S. M. Fauzi, “Mapping of software configuration management issues with the area of responsibility,” in *8th IEEE International Conference on Application of Information and Communication Technologies, AICT 2014 - Conference Proceedings*, 2014.
- [9] A. Bartusevics, L. Novickis, and S. Leye, “Implementation of Software Configuration Management Process by Models: Practical Experiments and Learned Lessons,” *Appl. Comput. Syst.*, vol. 16, no. 1, pp. 26–32, 2015.
- [10] M. WasimBhatti and I. Anjum Manarvi, “Architecture of a Software Configuration Management System for Globally Distributed Software Development Teams,” *Int. J. Comput. Appl.*, vol. 67, no. 19, pp. 1–5, 2013.
- [11] Vishal Assija, Anupam Baliyan and Vishal Jain, “Effective & Efficient Digital Advertisement Algorithms”, CSI-2015; 50th Golden Jubilee Annual Convention on “Digital Life”, held on 02nd to 05th December, 2015 at New Delhi, published by the Springer under ICT Based Innovations, Advances in Intelligent Systems and Computing having ISBN 978-981-10-6602-3 from page no. 83 to 91.
- [12] Vishal Jain and Dr. S. V. A. V. Prasad, “Analysis of RDBMS and Semantic Web Search in University System”, International Journal of Engineering Sciences & Emerging Technologies (IJESSET), Volume 7, Issue 2, October 2014, page no. 604-621 having ISSN No. 2231-6604.
- [13] Vishal Jain and Dr. S. V. A. V. Prasad, “Evaluation and Validation of Ontology Using Protégé Tool”, International Journal of Research in Engineering & Technology, Vol. 4, No. 5, May, 2016, page no. 1-12 having ISSN No. 2321-8843.
- [14] RP Shermey, S Balamurugan, “Certain Investigation on Context Aware Knowledge Discovery Strategies for Healthcare Systems”, Asian Journal of Research in Social Sciences and Humanities, Volume 6, Issue : 8, 2016
- [15] S Balamurugan, RP Shermey, Gokul Kruba Shanker, VS Kumar, VM Prabhakaran, “An Object Oriented Perspective of Context-Aware Monitoring Strategies for Cloud based Healthcare Systems”, Asian Journal of Research in Social Sciences and Humanities, Volume : 6, Issue : 8, 2016
- [16] S Balamurugan, P Anushree, S Adhiyaman, Gokul Kruba Shanker, VS Kumar, “RAIN Computing: Reliable and Adaptable Iot Network (RAIN) Computing”, Asian Journal of Research in Social Sciences and Humanities, Volume : 6, Issue : 8, 2016.
-