Study of Different Aspects of Software Configuration Management

Shivanand S. Rumma
Dept. of P.G. Studies and Research in Computer Science,
Gulbarga University, Kalaburagi.
Karnataka, India
Email id: shivanand_sr@yahoo.co.in

Kavita Manikrao
Bidar, Karnataka State
kavi_sachhi@yahoo.co.in

Abstract – Software configuration management (SCM) is an important activity in the software engineering life cycle. Today, most of the software project teams understand the need for SCM to manage change to their software systems. Most of the companies use the software configuration management and it is essential during the process of software development as rules to control and manage the evolution of software systems. In the Software Engineering, the Software Configuration System (SCM) is used for the tracking & controlling the changes in the software. Since evolution and change is unavoidable in software systems configuration management is considered as an integral element of software development and activity maintenance. The SCM activities are change management system, version management, and then system construction and release management. This paper is the study of software configuration management, the change control, CM process, steps and tools of SCM. The purpose of this paper is to give an idea about software configuration management and how this influences the software development process.

Keywords*:* Software Configuration Management, Controlling, Software Tools, change control

*****

I. Introduction

SCM enhances productivity and boosts application quality because it automatically versions files, labels and organizes them as they change during the development process[1]. Software Configuration Management (SCM) can be defined as the control of the evolution of complex software systems. The SCM is the discipline that enables a software development organization to keep evolving software products under control and helps in accomplishing software quality assurance goals. Developers are expected to produce quality software that integrates with existing applications and other projects. If control of the versions of software released into a production environment is lost huge costs can be incurred backtracking to a working solution. The Software Configuration Management (SCM) aids development by controlling who has access to versions of code and managing the particular versions that go together to create a release. The dependency management between versions and products can also improve the quality and reliability of developed software. The knowledge of the SCM is essential to ensure that development teams take maximum advantage of SCM. The software development team should have clear understanding of the tools and the benefits of using them.

II. Terminology

The following are the some terminologies regarding Software Configuration management.

➢ Configuration item: The configuration item is each single software entity that is individually identifiable the documents, the modules and libraries are all configuration items.

➢ Version: Its is one specific instance of a configuration item. This provides a stable referencing mechanism for the management of configuration items. They cannot be changed without creating a new version.

➢ Revision: The revision is the kind of versions that is a step in the evolution of a configuration item. This supersedes an earlier version and may contain arbitrary changes compared with its predecessor.

➢ Variant: Variants are alternative versions of a configuration item. Variants are remain valid at a given instant in time and are typically created to handle environmental differences

Software Configuration Management is the capability to controlling and managing the changes occur in a software project. The changes are natural and required in any of the software project.

III. WHY SCM IS REQUIRED?

Almost everyone in an organization plays a role in effective and efficient configuration management. SCM provides a common point of integration for all planning, oversight, and implementation activities for a software project or product line.SCM provides the framework labeling and identification for interfacing different activities and defining the mechanisms like change control. Also, provides framework for controlling computer program interfaces with their underlying support hardware and coordinating software changes when both hardware and software may be evolving during development or maintenance activities. It gives management with the visibility through accounting and audits.

The SCM system has the following advantages,

- The effective management in the system.
- The team organization can be improved

- The unnecessary work can be reduced
- Using this, we can avoid the problems that occur in the configuration.
- The tracking of the fault: This is one of the important features of SCM. Using this we can track any defect occur in the system.

IV. CHANGE CONTROL

Changes are normal in the evolution of a system. Proper change control is the essence of good configuration management. Software configuration control evaluates proposed changes to configuration items and coordinates the implementation of approved changes.[2]

The purpose of the Change Control process is to ensure that the cost, schedule, and technical impacts of the proposed changes are developed and considered by all affected parties and collected evaluations are considered in the approval or rejection of the proposed changes.Then, All affected parties are informed of proposed changes and their dispositions. The Baseline documentation is controlled and updated as appropriate to reflect approved changes. The process is initiated with the generation of a Change Request.

The table below specifies the sample risk required for each change request.

	No Risk	Low Risk	Medium Risk	High Risk
Low Impact	Immediately	Immediately	1 Day	1 Day
Medium Impact	Immediately	1 Day	7 Days	7 Days
High Impact	1 Day	1 Day	7 Days	14 Days

V. STEPS REQUIRED IN SOFTWARE CONFIGURATION MANAGEMENT

Steps used in Software Configuration Management are as follow:

1. Preliminary working
2. Base lining
3. Changing management
4. Configuration status accounting
5. Configuration review

Preliminary Work: In the initial working a small research and communication is done by the some people. In this time some part of the files are placed confidentially by the person and a few part of done work make publically.[3]

Base lining: It is this process some set of items officially become publicly available at the standard location for the people, who are authorized for using it. Each of the given files assigned a configuration id and a unique version number. [3]

Changing management: In a configuration system after the configuration, changes take place due to the checked in & base lining. These changes occur due to the changes in requirements, changes in the designing, changes in the environment. [3]

Configuration Status Accounting: In the large software development projects, it is likely possible that if the bugs occur the developer become focused on the specific enhancements and after bug fixing the larger picture gets lost and everyone unable to know the affects of changes. [3]

Configuration review: A configuration review means of instances for ensuring the process compliance. A successful compliance is done after the sufficiently random. [3]

VI. SOFTWARE CONFIGURATION MANAGEMENT PROCESS

Conventional Software Configuration Management process is best suitable solution for handling the changes in some software projects. Conventional Software Configuration Management process recognize the physical and functional attributes of a software and performs organized control of changes in the identified attributes for the maintaining software integrity throughout the software development life cycle [4].

The conventional Software Configuration Management identifies four methods which can be defined for the each software project to make sure a good SCM process. They are

- Configuration Control
- Configuration Status Accounting
- Configuration Identification
- Configuration Authentication

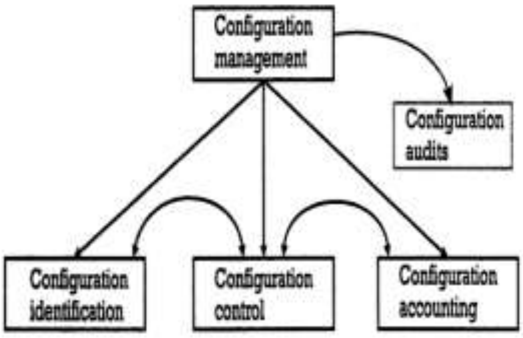

Fig. 1: Software Configuration Management Process

In this part most of the section covers up conventional Software Configuration Management.

VII. SCM Tools

There are tools are available in the market. The following are the examples of SCM Tools.

GNU Arch: Arch is an advanced SCM specification with multiple and independent but compatible implementations. it does not work well on Windows-based systems

ClearCase: ClearCase is the market leader and provides change management functionality in addition to the standard version control.It is used to Build management and auditing and user can able to retrieve old versions easily. Here, we can Labelling of product releases can done. The main disadvantage of ClearCase is the costly one.

Some other tools for software configuration management

- VSS Visual source safe
- CVS Concurrent version system
- Quma version control system
- IBM Configuration management version management

VIII. CONCLUSION

Modern IT organizations can have hundreds or thousands of servers supporting business users around the world. The optimized onfiguration management allows organizations to control and protect the deployment of their IT resources it can be a very powerful step in ensuring proper control of your IT resources. The information allows you to make informed decisions on maintaining and upgrading your IT environment.

IX. References:

[1] Merant,"The Benefits of Software Configuration Management",2001

[2] ESA Board for Software Standardisation and Control," Guide to Software configuration management",1995

[3] Managing Global Software Projects, Ramesh McGraw-Hill Education (India) Pvt Limited, 01-Jan-2005

[4] Software Quality Assurance: From Theory To Implementation By Daniel Galin.