Software Configuration Management

Software Configuration Management

- "Change Management" is one of the fundamental activities of software engineering.
- This is because the business environments for which software applications are developed, themselves are very dynamic.
- And business requirements , expectations from software applications keep changing.

Some definitions

- Configuration: the arrangement of a computer system or components as defined by the number, nature and interconnections of its constituent parts.
- Configuration Management: a discipline applying technical and administrative direction to: identify and document the functional and physical characteristics of a configuration item, and control changes to those characteristics, report change processing and implementation status and verify compliance with specified requirements.

Why Configuration Management?

- Consider the following aspects:
- Changes are frequent
- 1. Different customers have different requirements
- 2. Changes get clarified at a later date
- 3. Business Environment, too, changes
- 4. Technology changes
- 5. Personal whims!

- Software projects: Personal turnover
- 1. Developers leave
- Users change
- 3. Customer single point contact changes Another problems are:
- Double maintenance problems
- Shared data
- Simultaneous updates
- Missing/unknown version problems

- All this only means that change is inevitable when computer software is built.
- And changes increases the level of confusion among software engineers.
- When changes are not analyzed and documented before implementing then, confusion will result.
- "the art of coordinating software development to minimize confusion is called Configuration Management. Configuration Management is the art of identifying, organizing and controlling modifications to the software being built by a programming team. The goal is to maximize productivity by minimizing mistakes..."

SCM activities are developed to:

- Identify change
- Control change
- Ensure that change is being properly implemented
- Report changes to others who may have to do with it.
- Configuration Management includes six point:
- It is a Management discipline
- It identifies the proposed or actual configuration of a system
- This is done at discrete points in time.

- Systematically records and traces changes to all system components
- Provides tools for controlling changes; and finally
- Allows everything happening with the system, throughout the entire life-cycle of the system, to be verified via auditing and reporting tools

Configuration Management is a process used for efficiently developing and maintaining software. This is accomplished by improving: accountability, reproducibility, traceability and co-ordination.

Nomenclature

- System: the package of all the software that meet's the user's requirements.
- Subsystem: large systems can have many subsystems, such as communication, display and processing.
- Product: subsystems typically contain many products. Example an operating system might contain a control program, compilers, utilities and so forth.
- Components: at the next level, a control program could be made up of such components of the sub-system.
- Module: at the lowest level, components consist of a number of modules. Modules typically implement functions that are relatively small and self-contained.

SCM activities

SCM has four major activities:

- Identification
- Control
- Status Accounting
- Auditing

Software Configuration Identification

 Configuration Identification: an element of Configuration Management, consisting of selecting the configuration items for a system and recording their functional and physical characteristics in technical documentation.

In addition to this, we must maintain:

- Documentation
- The specifications and designs
- The test designs, data and reports
- The change and problem reports
- The 'make' files which direct and document the way the system is constructed.

Rules governing the process of SCI are:

- SCI defines the 'granularity' of the SCM
- SCI defines what is needed to be seen i.e. the visibility to all those who need to know what is happening on the project.
- SCI ensures that the identification scheme selected reflects the structure of the product, the project and the organization.
- The identification process must always be coupled with a parallel process of labeling the item with a distinct and unique label
- SCI is a critical project management task.

 SCI can be defined as work product or information that is created as part of the software engineering process.

Example of work products identified as configuration items:

- Requirement specifications
- Design specifications
- Test plans and Test cases
- Source code
- Software tools
- Software Configuration Management plans
- Software verification and validations plan
- User manuals

Software Configuration Identification

Important identification principles

- Labels
- Baselining
- 1. Partitioning a baseline is always subjective
- 2. Create some partition and maintain it
- 3. Baselines live forever
- 4. Relations v/s Versions
- 5. Visibility and Traceability

Configuration Identification numbering

Software configuration control

- Configuration control: An element of configuration management, consisting of the evaluation, coordination, approval or disapproval and implementation of changes to configuration item after formal establishment of their configuration identification.
- Software configuration control solves four of the most common problems facing software developers:

- The double maintenance problem
- The shared data problem
- The simultaneous update problem
- The missing/unknown version problem

Software configuration status accounting

- Main issue of Software Configuration Management is acquisition and maintenance of all information concerning a project's status and that of its parts.
- For each configuration item, regardless of whether the item is an individual item or an aggregate, a separate 'logical account' is maintained and transactions are recorded.
- In addition to transactions, actual programming information is recorded upon the configuration item's archive.

Software configuration status accounting

Configuration Status Accounting Reports: there are three reports, which must be called the 'most basic of reports'.

- Transaction log
- Change log
- Item 'delta' report

Some other common reports are:

- Resource usage
- Stock status
- Changes in progress
- Deviation agreed upon etc.

Software configuration status accounting

• Change control board: a group of people responsible for evaluating and approving/disapproving proposed changes to configuration items, and for implementation of approved changes.

CCB members are:

- Project Managers
- User Representative
- Quality Controller
- Configuration Controller

Change log

- This log is implemented as an informal / formal notebook.
- This log should contain all information relevant to, and regarding requested changes in the system or any part thereof.

The following minimum information is expected in the change log:

- Change number
- The change number will be assigned by the CCB
- Change request status type
- Request opened
- Authorized for implementation
- In implementation
- In pre-release testing
- Change request denied

- Change request originator details
- Software documents affected by this change request
- Change request originator date
- Change description
- Details of the person who will carry out the Change Request when approved
- Impact Analysis Details
- Follow-up action required to implement this change request

Delta report

 This report is generated to summarize progress of the development and to compare this progress with status presented in the previous report.

Report should include:

- Dates
- Narrative
- format

Stock status report

- This report should summarize the status of each and all system changes which are in a defined state of 'open'; and should indicate the following items:
- Inventory
- Description
- Status
- Analyst/programmer/software engineer
- Completion date

Transaction log

 This report should clearly show the effect and relationships resulting from each and every event, which occurred during the course of the software project.

It should have the following minimum entries:

- Transaction number
- Date and time stamp
- Originator
- Software elements affected
- Description of the activity
- Change in progress report
- Participants
- Impact

Project Change Request Form

General Information	
Request Number:	Date:
Name:	Phone:
Project Name:	
Date Requested:	
Proposed Change	
Description:	
Reason for Change:	
Business Partner Review	
Background:	
Performance Impact:	
Technical Impact:	
Schedule Impact:	
Budget Impact:	
Enterprise Impact (to other projects):	
Requesting Project Manager Revie	ew
Approved Denied	Approved with specific conditions
Title/Role:	
Reason for Denial:	
Specified Conditions for Approval:	
Signature:	Date:

Configuration Auditing

- A software configuration audit should periodically be performed to ensure that the SCM practices and procedures are rigorously followed.
- Every major baseline must be audited. For in-house systems, an informal audit is usually sufficient.
- The intention of audit is to provide project visibility.
- Generally, configuration audit is performed by an external auditor.

Control over suppliers and subcontractors

- It should be noted that the items or subsystems, which are developed by sub-contractor or supplied by vendor, must always be controlled and audited for Configuration Management.
- The determination of sub-contractor Configuration Management requirements should be governed by a consistent concept of the needs of the agency issuing the contract.

Guidelines for Configuration Audits

- SCM audits are periodically needs to ensure the integrity of the software baselines.
- An audit is performed before every major baseline change or major release and non-conformances if any are tracked to closure and reported to all stake holders.
- The configuration audit team consists of qualified technical people who are not involved in the specific tasks being audited.
- The audit verifies that changes to the baseline are implemented as intended.
- The auditing function is an integral part of the SCM system.

A phase review process then ensures that the proper actions are taken as follows:

- Requirements
- Functional
- High-level design
- Detailed design
- Product
- operational

Process Audit Checklist Process: Configuration Audits					
Are process inputs used and the correct					
versions, where applicable?					
CM Plan, schedule					
Baseline documentation					
Baseline product					
• Final test results					
Is the configuration audits process performed					
when scheduled and when ready?					
Is configuration verification performed of					
requirements and design baselines?					
Are functional configuration audits performed					
of accepted product?					
Are final physical configuration audits					
performed of completed product?					
Does the configuration audits process comply					
with its defined documentation?					
Are the correct outputs produced?					
Baseline configuration verification					
results and actions					
• Functional configuration audit results					
Physical configuration audit results					
Additional Notes:					
QA:				Date:	

Personnel in SCM Activities

- Number of personnel
- Expected skills levels
- Job titles
- Security clearances needed.

SCM: Some pitfalls

Anti-patterns are the syndromes to be watched for:

- Silver Bullet
- Developer-software configuration management
- Failure to audit
- Software configuration management experts
- Postmortem planning