5 Configuration Management

Syllabus

Software Configuration Management (SCM) - Baselines, Software Configuration Items (SCI), SCM Process, Identification of Objects in the Software Configuration, Version Control, Change Control, Configuration Audit, Status Reporting, Goals of SCM.

Contents

5.1 Software Configuration Management

5.2 Advantages of SCM

5.3 Need for SCM

5.4 Baseline

5.5 Software Configuration Items

5.6 Software Configuration Management Process

5.7 Goal of Software Configuration Management

5.1 Software Configuration Management

- Software Configuration Management (SCM) is a component of SQA.
- This is the discipline that applies a rigorous approach to ensure
  - Different items produced in software systems are all identified and tracked
  - Changes to the various items are recorded and tracked
  - Completion and proper integration of all the various modules
- SCM can help determine the impact of change as well as control parallel development
- SCM deals with issues related to control of software changes, proper documentation of changes, registering and storing the approved software versions, tracking registered versions and more **throughout the software system's life cycle.**
- It can track and control changes in all aspects of software development
  - Requirements
  - Design
  - Code
  - Tests
  - Documentation

5.2 Advantages of SCM

The SCM system has the following advantages :

- Reduced redundant work.
- Effective management of simultaneous updates.
- Avoids configuration-related problems.
- Facilitates team coordination.
- Helps in building management; managing tools used in builds.
- Defect tracking: It ensures that every defect has traceability back to its source.

5.3 Need for SCM

- As software evolves - many resources make changes to the system
  - CM prevents avoidable errors that arise from conflicting changes

- Often many versions of the software are released and require support
  - CM allows a team to support many versions.
  - CM allows changes in sequential versions to be propagated
- CM allows developers to track changes and reverse any fatal changes to take a software system back to its last known safe state
- Good SCM increases confidence that we are :
  - Building the right system
  - Testing the system enough
  - Changing it correctly and carefully
- It also :
  - Restrains non-essential changes
  - Ensures that decisions and changes are traceable
  - Increases accountability
  - Improves overall software quality
  - Provides a fall back position when things do not work

5.4 Baseline

- A product that has been formally approved, and consists of a well-defined set of consistent configuration items
- A specification or product that has been formally reviewed and agreed to by responsible management,that thereafter serves as the basis for further development, and can be changed only through formal change control procedures.
- As the system is developed a number of baselines are created :
  - Developmental baseline (RAD, SDD, integration test, ...).
    - ✦ **Goal :** Coordinate engineering activities.
  - Functional baseline (prototype, technology preview, alpha, beta release).
    - ✦ **Goal :** Obtain customer experiences with functional system.
  - Product baseline (GA with a version - win95, word 2000).
    - ✦ **Goal :** Coordinate sales and customer support.

❑ Baseline Levels

- The software system can be tagged at various stages of its evolution with a baseline number
  - Development baseline "n" (where the "n" can be indicative of the 10 % of functionality implemented)
  - Testing baseline (where a specific build is created for the specific purpose of testing)
  - Release baseline (where the software is built for GA)
- There is no rule on when to baseline - but a good guideline is to have one a week

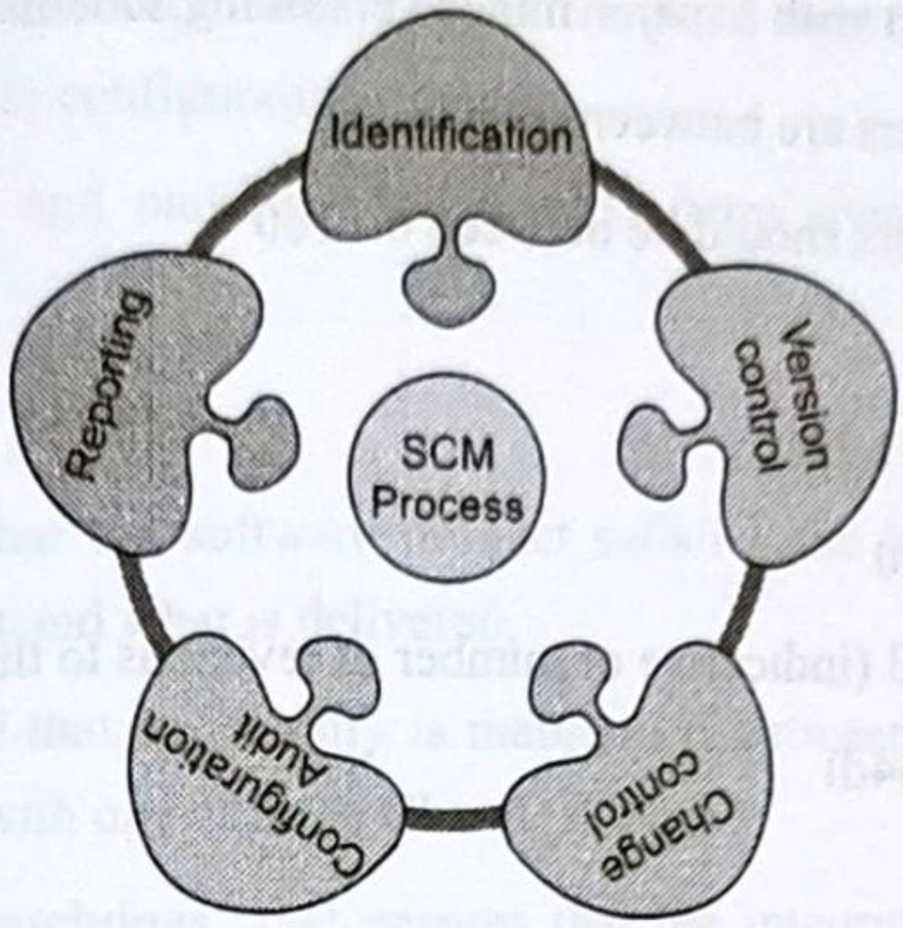
🕮 5.5 Software Configuration Items

- A large number of items are part of the software development process :
  - Source and binary modules
  - Hardware and operating systems
  - Documentation
    - ✦ Requirements
    - ✦ Design
  - Test cases
  - Etc...
- Key is to identify the items that need to be under SCM

🕮 5.6 Software Configuration Management Process

- It uses the tools which keep that the necessary change has been implemented adequately to the appropriate component. The SCM process defines a number of tasks :
  - Identification of objects in the software configuration
  - Version control
  - Change control
  - Configuration audit
  - Status reporting

Software Configuration Management Process



5.6.1 Identification of Objects in Software Configuration

- All configuration requirements on a project should be identified and recorded. This includes functionality requirements, design requirements, and any other specifications. The completion of this process results in the configuration baseline for the project.
- **Basic object :** Unit of Text created by a software engineer during analysis, design, code, or test.
- **Aggregate object :** A collection of essential objects and other aggregate objects. Design Specification is an aggregate object.
- Each object has a set of distinct characteristics that identify it uniquely : a name, a description, a list of resources, and a "realization."

5.6.2 Version Control

- Version Control combines procedures and tools to handle different version of configuration objects that are generated during the software process.
- Configuration management allows a user to specify the alternative configuration of the software system through the selection of appropriate versions. This is supported by associating attributes with each software version, and then allowing a configuration to be specified [and constructed] by describing the set of desired attributes.

Version Allocation

- Once a Configuration Item (CI) has been identified - a proper version number must be allocated.

- The best option is to start with a major-minor versioning scheme
  - Major version numbers are between 0 - n
  - Minor version numbers should be between 0 - 100
- Examples :
  - Report_Java (version 1.23)
    - ✦ Major version : 1.0
    - ✦ Minor version : 23 (indicative of number of revisions to this file)
- Project plan (version 6.34d)
  - Major version : 6
  - Minor version : 34
  - The "d" is indicative of "draft"
- Versioning scheme is developed by the company to suite their needs
- Often many companies prefix the configuration item based on its type.
  - Documentation may be prefixed "doc"
  - Source code can be "src"
  - Example: doc-pmp-2.34
    - ✦ Project management plan document (version 2.34)
- New versions of software can be :
  - Maintenance releases
  - Minor upgrades
  - Technology refresh or major upgrades
  - Technology insertion

5.6.3 Change Control

- Change control is a procedural method which ensures quality and consistency when changes are made in the configuration object. In this step, the change request is submitted to software configuration manager.
- Activities during this process :
  - Control ad-hoc change to build stable software development environment. Changes are committed to the repository.

- The request will be checked based on the technical merit, possible side effects and overall impact on other configuration objects.
- It manages changes and making configuration items available during the software lifecycle.

5.6.4 Configuration Audit

- SCM audits to verify that the software product satisfies the baselines requirements and ensures that what is built and what is delivered.
- SCM audits also ensure that traceability is maintained between all CIs and that all work requests are associated with one or more CI modification.
- SCM audits are the "**watchdogs**" that ensures that the integrity of the project's scope is preserved.
- The two main types of audit are :
  - **Physical audit :** Are all identified items have a correct version and revision, this helps us remove old and unnecessary items.
  - **Functional audit :** Verifies that the items under SCM satisfy defined specifications.

5.6.5 Status Reporting

- Configuration status reporting (sometimes also called status accounting) providing accurate status and current configuration data to developers, testers, end users, customers and stakeholders through admin guides, user guides, FAQs, release notes, installation guide, configuration guide, etc.

5.7 Goal of Software Configuration Management

- ✦ **Goal 1 :** Software configuration management activities are planned
- ✦ **Goal 2 :** Selected software work products are identified, controlled, and available
- ✦ **Goal 3 :** Changes to identified software work products are controlled
- ✦ **Goal 4 :** Affected groups and individuals are informed of the status and content of software baselines.

❑❑❑