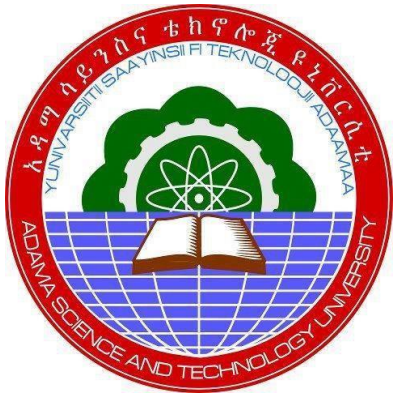


Chapter -4

Configuration Management

SEng4203

Lecture notes
ASTU, 2023
Begna D.
ibmbegna@gmail.com

CHAPTER OUTLINES

- Configuration Management
- Change Control
- Management and organizational issues
- Management responsibilities
- Documentation

CONFIGURATION MANAGEMENT

- Large, complex systems undergo many more changes than relatively small systems, and management of changes in large systems is nontrivial.
- Therefore, the concept of CM was developed to manage changes in large systems.
- The goal of CM is to manage and control the numerous corrections, extensions, and adaptations that are applied to a system over its lifetime.
- It handles the control of all product items and changes to those items.

CONFIGURATION MANAGEMENT

- Software Configuration Management (SCM) is the process of managing and controlling changes to software systems throughout their lifecycle.
- It involves the systematic management of software configuration items (SCIs) and the associated documentation, ensuring that changes are planned, implemented, tested, and tracked in a controlled manner

CONFIGURATION MANAGEMENT…

- **SCM accrues two kinds of benefits to an organization as follows:**
- SCM ensures that development processes are traceable and systematic so that all changes are precisely managed.
  - Consequently, the product is always in a well-defined state
- SCM enhances the quality of the delivered system and the productivity of the maintainers

CONFIGURATION MANAGEMENT…

- The objectives of SCM are to:
- Uniquely identify every version of every software at various points in time.
- Retain past versions of documentations and software.
- Provide a trail of audit for all modifications performed.
- Throughout the software life cycle, maintain the traceability and integrity of the system changes

CONFIGURATION MANAGEMENT…

- **Projects benefit from effective SCM as follows**

1. Confusion is reduced and order is established.
2. To maintain product integrity, the necessary activities are organized.
3. Correct product configurations are ensured.
4. Quality is ensured and better quality software consumes less maintenance efforts.
5. Productivity is improved, because analysts and programmers know exactly where to find any piece of the software.

CONFIGURATION MANAGEMENT…

- 6. Liability is reduced by documenting the trail of actions.
- 7. Life cycle cost is reduced.
- 8.Conformance with requirements is enabled.
- 9.A reliable working environment is provided.
- 10.Compliance with standards is enhanced.
- 11.Accounting of status is enhanced.

CONFIGURATION MANAGEMENT...

SCM Spectrum of Functionality

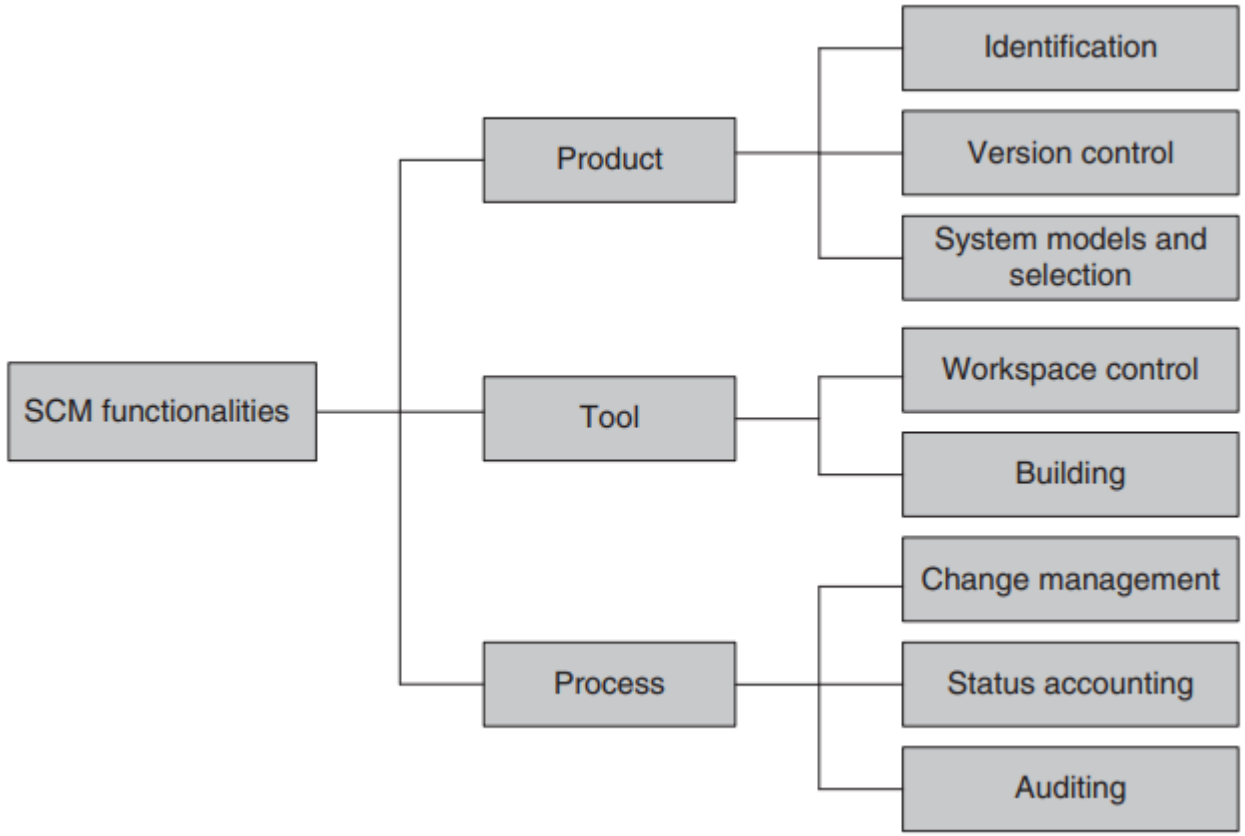


Fig 4.1Technical dimensions of SCM systems

CHANGE REQUEST

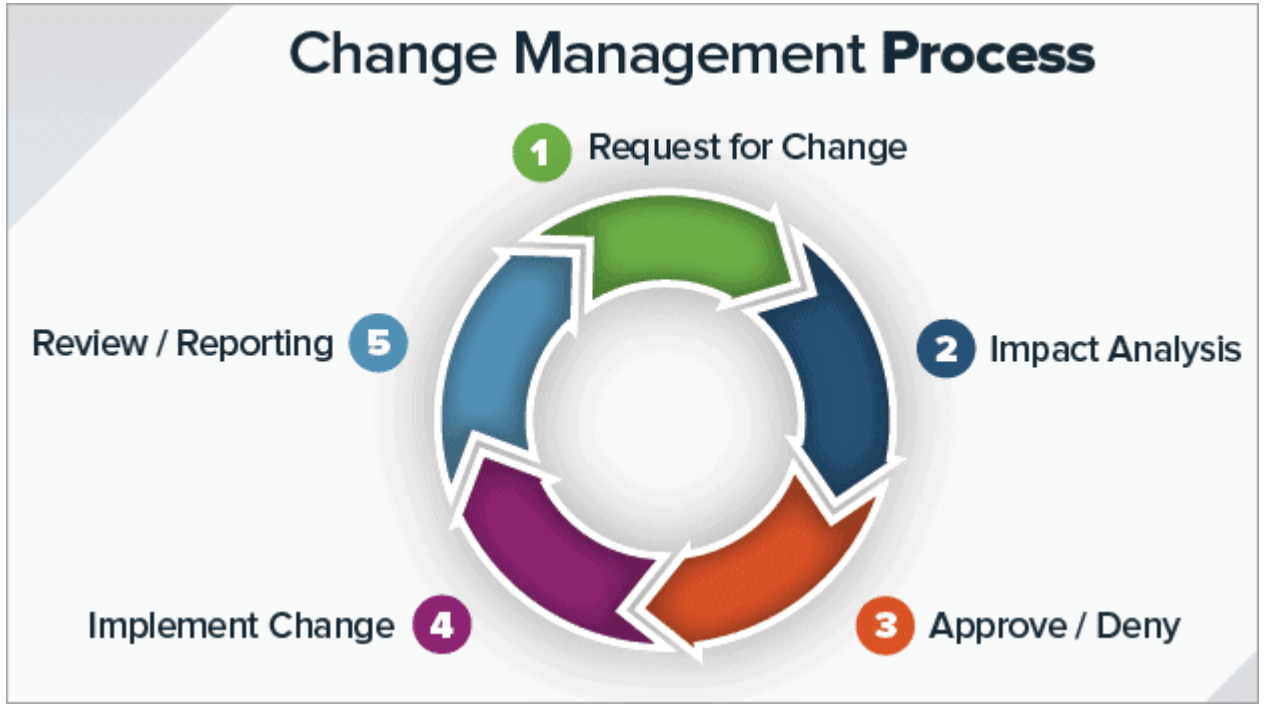
- A history of CR includes the answers to the following questions:

1. Why are changes made?
2. When are the changes made?
3. Who makes the changes?
4. What changes are made?

CHANGE CONTROL

- Change management in software maintenance refers to the process of managing and controlling changes made to software systems after they have been deployed and are in use.
- It involves **planning, evaluating, implementing, and tracking** changes to ensure they are carried out in a controlled and systematic manner.
- The goal of change management in software maintenance is to minimize the risks associated with changes, ensure the stability and reliability of the software, and meet the evolving needs of users and stakeholders.

CHANGE CONTROL……



Change Control……

- change control refers to the process of managing changes to software systems, applications, or codebases.
- It involves a systematic approach to controlling modifications to software to ensure that they are properly evaluated, approved, implemented, and tracked.
- The primary objective of change control is to minimize risks, maintain software quality, and ensure that changes are effectively managed throughout the development lifecycle.

Change Control……

- The change control process typically includes the following steps:

- **Change identification:** Changes can arise from various sources, such as user feedback, bug reports, new requirements, or improvements.
  - The first step is to identify and document the proposed change.

- **Change request submission:** The change request, often in the form of a formal document or ticket, is submitted to the change control authority or team.
  - It should include relevant details, such as the reason for the change, desired outcomes, impact analysis, and supporting documentation.

Change Control……

- **Change evaluation and impact analysis:** the change control team evaluates the proposed change and performs an impact analysis.
- This analysis assesses the potential effects of the change on the software system, including functionality, performance, security, and dependencies.
- It also considers resource requirements and potential risks.

- **Change approval or rejection**: based on the evaluation and impact analysis, the change control authority approves or rejects the change request.
  - Approval may be conditional, with specific criteria or constraints that need to be met before implementation.

Change Control……

- **Change implementation:** If the change is approved, a plan for implementation is developed.
  - This plan includes details such as the development tasks, testing requirements, deployment considerations, and rollback strategies.
  - It ensures that the change is implemented in a controlled and systematic manner.
- **Change testing:** Depending on the nature of the change, appropriate testing activities are performed.
  - Testing helps identify any issues or unintended consequences introduced by the change.

Change Control……

- **Change deployment:** once the change has been tested and deemed ready for deployment, it is introduced into the production environment.
  - This may involve activities such as code deployment, database migrations, configuration updates, or infrastructure changes.
  - Proper version control and release management practices are essential to track and manage the deployed changes.

- **Change monitoring and review:** after deployment, the change is closely monitored to ensure that it performs as expected and does not introduce any negative impact.
  - Performance metrics, error logs, user feedback, and other monitoring mechanisms help identify and resolve any issues that arise from the change.

Change Control……

- **Change documentation and communication:** throughout the change control process, documentation is updated to reflect the implemented changes.
  - It includes updating system documentation, release notes, user manuals, and any other relevant artifacts.
  - Communication with stakeholders, such as users, developers, and project managers, is crucial to keep them informed about the change and its implications.

- **Change closure and post-implementation review:** once the change has been successfully implemented and reviewed, it is formally closed.
  - A post-implementation review is conducted to assess the effectiveness of the change, gather feedback, and identify lessons learned.
  - This information helps to improve future change control processes.

Management and organizational issues

- Configuration management involves managing and controlling the changes made to software systems, including their documentation, code, and related artifacts.
- While configuration management aims to streamline the development process and ensure the stability and integrity of software systems.

Management and organizational issues…

- There can be various management and organizational issues that need to be addressed. Some of these issues include:
- **Lack of Clear Configuration Management Plan**: Organizations may face challenges if they don't have a well-defined configuration management plan in place.
- Without clear guidelines and procedures, it becomes difficult to manage and control software changes effectively.
- A comprehensive plan should outline roles and responsibilities, change control processes, versioning and release strategies, and tools to be used.

Management and organizational issues…

- **Inadequate Change Control Processes:** Without robust change control processes, organizations can encounter issues such as unauthorized or uncontrolled changes, conflicting modifications, and difficulty in tracking and auditing changes.
  - It is essential to establish a structured change control process that includes proper evaluation, approval, and documentation of change requests.

Management and organizational issues…

- **Lack of Collaboration and Communication:** Configuration management requires collaboration and communication among various stakeholders, including developers, testers, project managers, and end-users.
  - Inadequate communication between teams can lead to misunderstandings, delays, and inconsistencies in managing software changes.
- **Resistance to Change:** Introducing configuration management practices into an organization may face resistance from team members who are accustomed to existing development processes.
  - Some individuals may resist adopting new tools or following prescribed procedures.
  - It is important to address this resistance through training, clear communication of benefits, and demonstrating how configuration management can streamline development efforts and improve software quality.

Management and organizational issues…

- **Tooling and Infrastructure Challenges:** Configuration management often relies on specialized tools and infrastructure to manage software versions, track changes, and facilitate collaboration.
  - Integrating configuration management tools with other development tools and workflows can be complex and require careful planning and coordination.
- **Scalability and Flexibility:** Configuration management practices should be scalable and adaptable to accommodate the evolving needs of the organization and the software systems.
  - As projects grow in complexity or the development team expands, the configuration management process should be able to handle increased demands without introducing bottlenecks or excessive overhead.
  - Flexibility in accommodating different development methodologies, such as agile or waterfall, is also crucial.

Management and organizational issues…

- Training and Skill Gaps: Effective configuration management requires skilled personnel who understand the principles, tools, and best practices associated with managing software changes.
  - Organizations may face challenges if there is a lack of trained personnel or a knowledge gap within the team.
  - Providing comprehensive training and mentoring programs can help address these skill gaps and ensure that the team members are equipped to handle configuration management tasks effectively.

Management Responsibilities

- Management responsibilities play a crucial role in overseeing and ensuring the effective implementation of configuration management practices within an organization.
- Key management responsibilities in configuration management:
- **Establishing Configuration Management Policies:** Management is responsible for defining and establishing configuration management policies and guidelines that align with organizational objectives and industry best practices.
- **Developing a Configuration Management Plan:** Addresses version control, change control, release management, documentation management, and other aspects of configuration management.
- **Resource Allocation:** Management is responsible for allocating appropriate resources, including personnel, tools, and infrastructure, to support configuration management activities.

Management Responsibilities…

- **Providing Training and Education:** Management is responsible for ensuring that team members receive adequate training and education on configuration management practices.
- This includes providing training on configuration management tools, processes, and methodologies.
- **Enforcing Change Control Processes:** involves establishing a structured change management process, ensuring that change requests are properly evaluated, approved, and implemented.
- **Monitoring and Auditing:** includes periodically reviewing configuration management activities, conducting audits to identify gaps or deviations, and taking corrective actions as necessary.
-

Management Responsibilities…

- **Continuous Improvement:** involves analyzing the effectiveness of existing processes, identifying areas for improvement, and implementing changes to enhance efficiency and quality
- **Collaboration and Communication:** includes promoting cross-functional collaboration, facilitating communication channels, and ensuring that information regarding software configurations and changes is shared transparently among relevant stakeholders.
- **Risk Management:** includes assessing the impact of changes, understanding potential risks and dependencies, and implementing measures to mitigate risks that could impact software development and delivery.

Reading Assignment

- Documentation and tools in Configuration management

Discussion..