

Course: Software Configuration Management

Week 7: Software Configuration Identification – Change Management

Lecturer: Yimer Amedie (MSc.)

Addis Ababa Science and Technology University, Ethiopia

October, 2025

Contents



SOFTWARE CONFIGURATION MANAGEMENT

- Software Configuration Change Management
- Change Request Process and Documentation
- Change Evaluation and Impact Analysis
- Change Approval and Implementation Procedures
- Deviations & Waivers in Configuration Control
- Summary

Figure 1: Concepts of SCM
(Source: OpenAI, 2025)

Learning Outcomes

After completing this lesson, you will be able to:

- Define the concept and purpose of change management in SCM.
- Explain how change requests are initiated and documented.
- Evaluate changes through impact analysis and assessment criteria.
- Describe the approval and implementation steps of changes.
- Distinguish between deviations and waivers in configuration control.

What is Change Management?

- A systematic process to handle modifications.
- Ensures orderly control over changes to baselined items.
- Protects product integrity by avoiding unauthorized changes.
- Involves documentation and traceability of every modification.
- Promotes communication among developers, testers, and stakeholders.

Objectives of Change Management



Maintain software integrity throughout evolution



Ensure visibility and accountability for all changes



Control the impact on schedule, cost, and quality



Provide traceability from request to implementation



Support auditing and compliance with standards

Why Change Management is Important



Changes are inevitable in software development

Uncontrolled changes cause instability and confusion



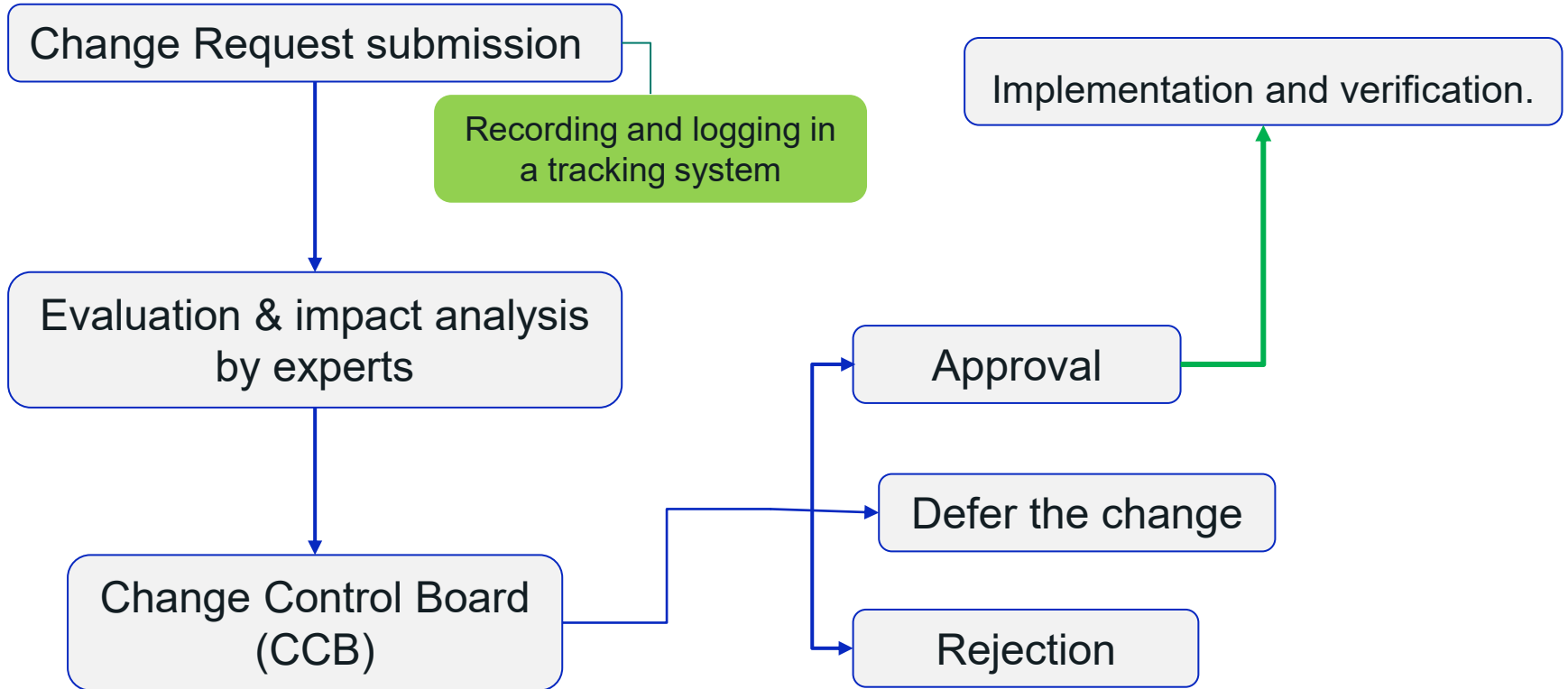
Ensures alignment with business goals and priorities

Protects the baseline from accidental modifications



Enhances customer confidence and regulatory compliance

The Change Management Process



Change Request (CR) – Definition and Role

- A change request:
 - A formal proposal to modify a configuration item.
 - It records the reason and justification for change.
 - Identifies affected items and stakeholders.
 - Contains priority, risk, and expected benefits.
 - Serves as the entry point to Change Management.

Information Included in a Change Request

- Each Change Request must contain comprehensive details to support evaluation and decision-making
 - ! Unique identification number for tracking.
 - ! Description of the proposed change and rationale.
 - ! Impacted items or systems.
 - ! Estimated effort, resources, and cost implications.
 - ! Priority level and risk category.

Sources of Change Requests



Developers identify technical improvements or defects.



Testers raise CRs for failed test cases or quality issues.



Customers or Users propose enhancements.



Managers initiate CRs for compliance or cost reasons.



Automated Tools generate CRs for detected anomalies.

Categories of Change Requests

Corrective

Fix identified defects or errors.

Preventive

Avoid potential future issues.

Adaptive

Adjust software for new environments

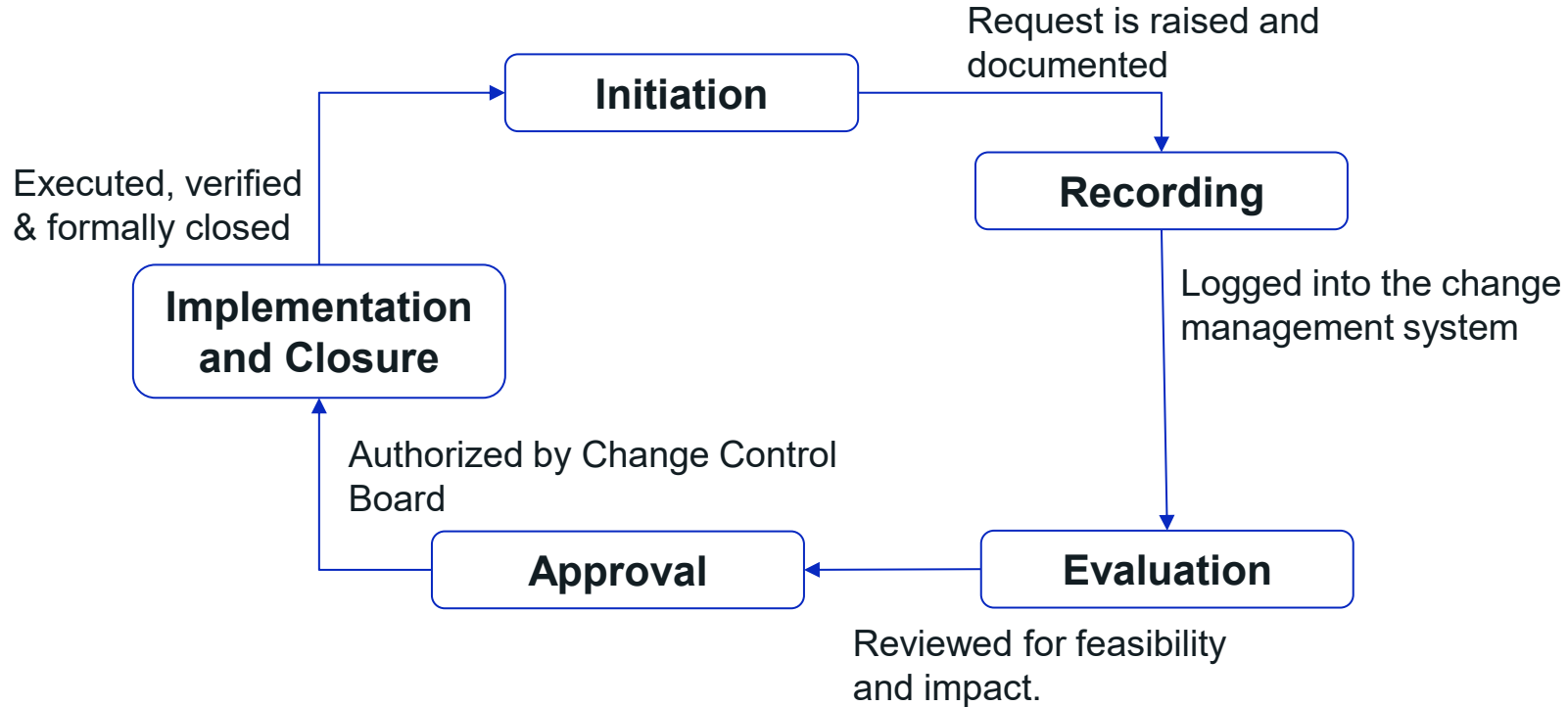
Emergency

Urgent changes requiring immediate attention.

Perfective

Improve performance or maintainability.

Change Request Lifecycle



Purpose of Change Evaluation

- The goal of change evaluation is to:
 - Determine the necessity of the proposed change.
 - Assess technical feasibility and resource availability.
 - Analyze cost, schedule, and risk impact.
 - Identify dependencies and affected components.
 - Provide data for informed decision-making.

Change Evaluation Activities

- The activities in change evaluation are:
 - ❑ Conduct impact analysis on affected modules.
 - ❑ Estimate development and testing effort.
 - ❑ Assess risk level and propose mitigation.
 - ❑ Review consistency with project requirements.
 - ❑ Prepare evaluation reports for CCB review.

Impact Analysis in Change Management

- Impact analysis is a systematic approach to understand the consequences of proposed modifications.
 - Identifies components affected by a proposed change.
 - Examines dependencies and interfaces between modules.
 - Evaluates potential side effects on performance and behavior.
 - Helps estimate testing scope and cost.
 - Ensures traceability from requirements to implementation.

Criteria for Change Approval



Alignment with project goals and requirements



Feasibility and technical soundness of the proposed change



Risk evaluation and mitigation adequacy.



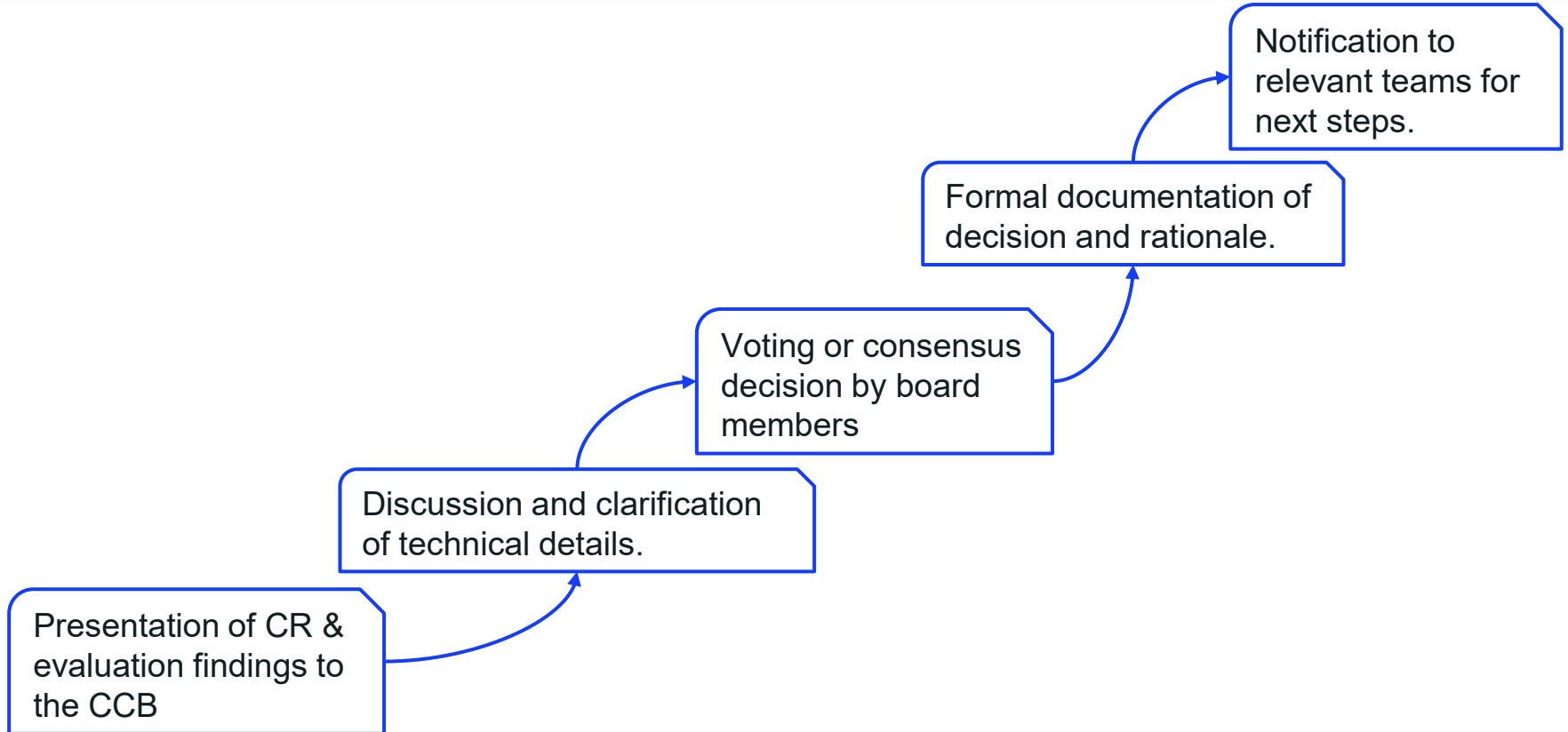
Impact on schedule and cost.



Availability of resources and tools.



The Approval Process



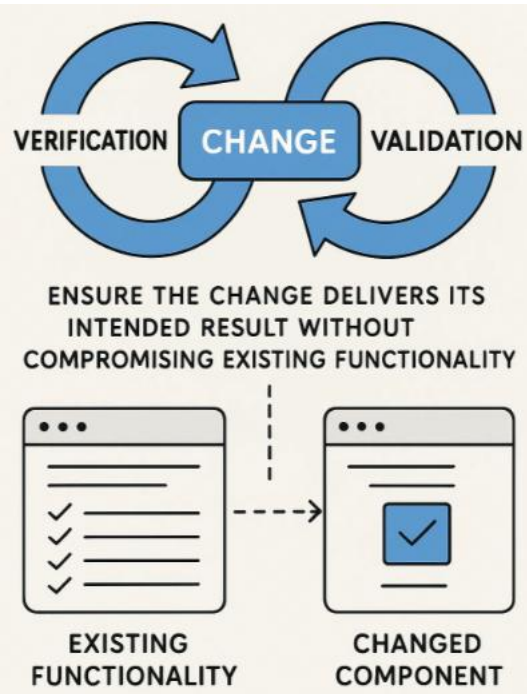
Implementation Phase Overview

- The implementation stage transforms the approved request into a tangible software modification:
 - Begins after formal approval by the CCB.
 - Involves planning, coding, and testing of the change.
 - Ensures version control of modified items.
 - Requires coordination across teams for consistency.
 - Ends with integration and baseline update.

Key Principles of Change Implementation

- Change implementation must follow strict configuration control principles to ensure quality and traceability
 - Maintain traceability from CR to code changes.
 - Apply controlled check-in/check-out procedures.
 - Perform peer reviews before committing changes.
 - Execute unit and integration testing after modification.
 - Document lessons learned and outcomes.

Verification and Validation of Changes



- ✓ Conduct verification testing against change objectives.
- ✓ Perform regression testing to detect side effects.
- ✓ Update test documentation and results.
- ✓ Ensure traceability to original requirements.
- ✓ Obtain formal acceptance before baseline update.

Figure 2: Software change assurance
(Source: OpenAI, 2025)

Implementing Approved Changes – Planning Phase

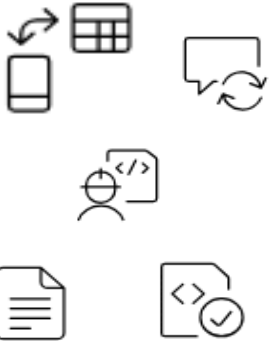
- Once a change is approved, the implementation process begins with thorough planning



- Review approved change documentation.
- Identify affected configuration items (CIs).
- Develop a detailed implementation plan.
- Assign responsible personnel or teams.
- Define rollback or contingency procedures.

Implementing Approved Changes – Execution Phase

- During execution of change:
 - Apply changes to the baseline version.
 - Update configuration documentation.
 - Perform necessary builds and integrations.
 - Conduct testing and verification.
 - Communicate status updates to stakeholders.



Post-Implementation Review

- After implementation, a post-implementation review (PIR) is conducted to validate the results.
 - Verify the change achieves intended goals.
 - Assess side effects or unintended impacts.
 - Confirm updates in configuration records.
 - Obtain final approval from stakeholders.
 - Document lessons learned.

Deviations in Configuration Management

- In configuration management deviation (Leon, 2015):
 - Temporary departures from baseline requirements.
 - Typically, pre-approved under specific conditions.
 - Used when strict compliance is impractical.
 - Must not compromise system integrity.
 - Requires formal documentation and tracking.

Waivers in Configuration Management

- In configuration management deviation (Leon, 2015):
 - Permanent authorization for nonconformance.
 - Issued when a requirement cannot be met.
 - Requires detailed justification and approval.
 - Becomes part of the configuration baseline.
 - Documented in configuration records.

Change Tracking and Traceability

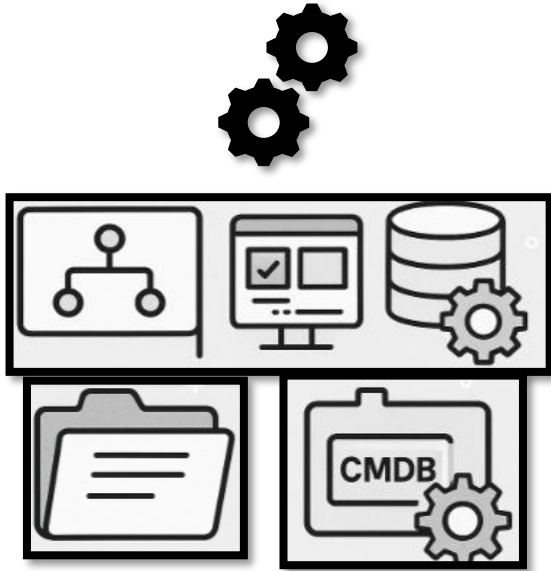


- Change tracking and traceability are core principles in software configuration management.
 - Record all approved and implemented changes.
 - Maintain links between requests, approvals, and code.
 - Use version control tools for documentation.
 - Enable audit trails for accountability.
 - Support rollback and recovery operations.

Configuration Auditing after Change Implementation

- Configuration auditing is conducted after change implementation to ensure full compliance with configuration management policies.
 - Verify implementation aligns with approved request.
 - Ensure documentation and baselines are updated.
 - Confirm testing and verification results.
 - Check for unauthorized changes.
 - Conduct both functional and physical audits.

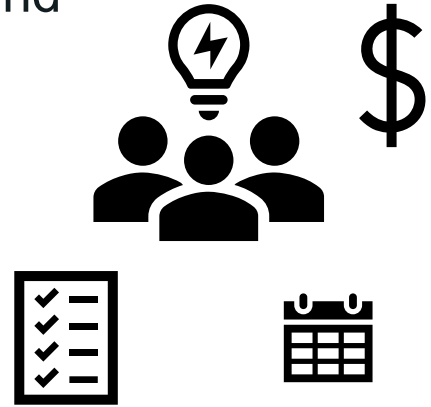
Tools Supporting Change Management



- Version control systems (Git, SVN, CVS).
- Issue tracking tools (JIRA).
- Configuration databases (CMDB).
- Automated build and deployment systems.
- Document management repositories..

Role of the Configuration Control Board (CCB)

- The governing authority responsible for managing and approving changes.
 - Reviews and approves all change requests.
 - Evaluates technical, cost, and schedule impacts.
 - Ensures consistency with project objectives.
 - Authorizes deviations and waivers.
 - Maintains change documentation.



Communication in Change Management

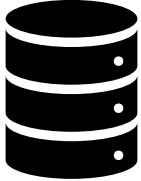
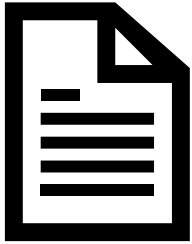


- Keep all stakeholders informed.
- Use standardized change reports.
- Schedule periodic review meetings.
- Update affected documentation and manuals.
- Encourage feedback and collaboration.

Handling Emergency Changes

- Emergency changes occur when immediate action is needed to resolve issues
 - Arise from critical failures or security issues.
 - Require fast evaluation and approval.
 - Follow expedited implementation procedures.
 - Must still be documented post-implementation.
 - Reviewed retrospectively by the CCB.

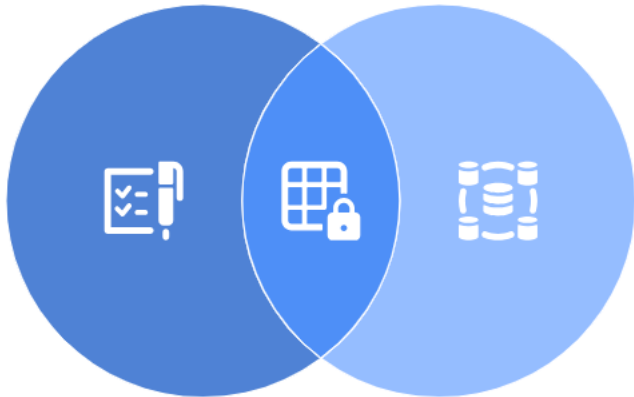
Change Documentation and Reporting



- Maintain formal change logs and reports.
- Record request details, evaluations, and approvals.
- Track implementation progress and results.
- Include deviations and waivers in reports.
- Share summaries with management and auditors.

Integrating Change Management with Configuration Identification

Change management & configuration identification are deeply interconnected



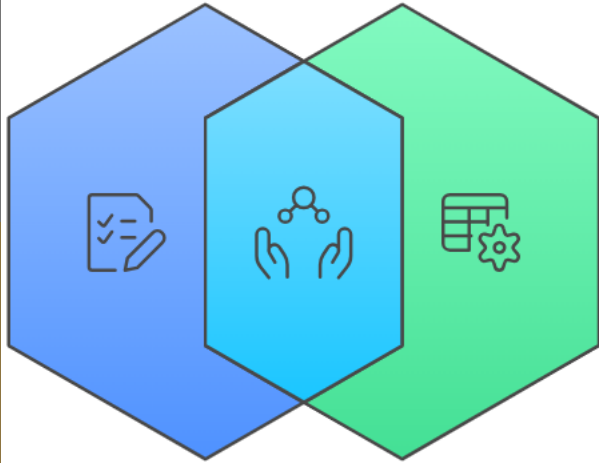
- Links changes to specific configuration items.
- Updates item identifiers and baselines.
- Ensures consistency across repositories.
- Prevents orphaned or redundant changes.
- Promotes traceability throughout lifecycle.

Integration with Configuration Control

- Configuration control
 - Enforces formal approval processes.
 - Ensures changes meet quality and compliance standards.
 - Aligns with baseline control mechanisms.
 - Coordinates between technical and managerial teams.
 - Reduces risks of unauthorized modifications.

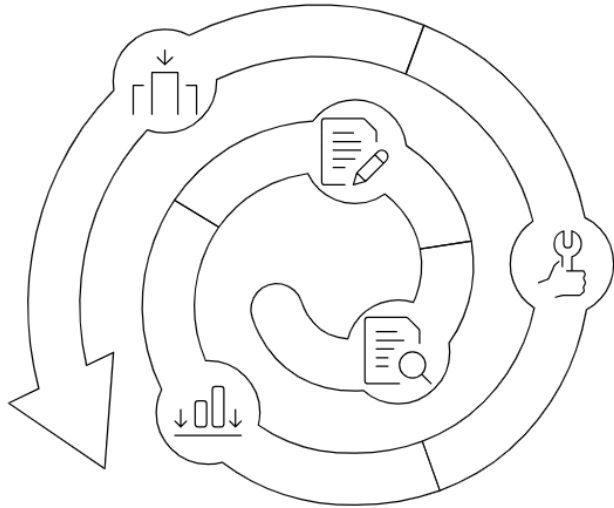
Relationship Between Change Management and Configuration Status Accounting

- Configuration status accounting (CSA):



- Records current state of all configuration items.
- Reflects implemented and pending changes.
- Supports reporting and audit readiness.
- Provides visibility for stakeholders.
- Facilitates decision-making and resource planning.

Auditing Deviations and Waivers



- Verify justification and approval documentation.
- Confirm validity and expiration of deviations.
- Ensure waivers are correctly recorded in baselines.
- Review impact on overall system integrity.
- Report audit findings and recommendations.

Best Practices in Change Management

- Adopting best practices ensures an efficient and consistent change management process
 - Standardize change request procedures.
 - Automate workflows using tools.
 - Maintain traceability and version control.
 - Ensure stakeholder engagement.
 - Conduct periodic reviews and audits.

Challenges in Change Management

- Some of the major challenges of change management are:
 - ≠ Resistance to process discipline.
 - ≠ Poor documentation or tracking.
 - ≠ Unclear communication channels.
 - ≠ Lack of tool integration.
 - ≠ Time and resource constraints.

Summary

- ❖ Change management ensures controlled evolution.
- ❖ Includes evaluation, approval, and implementation.
- ❖ Deviations and waivers provide flexibility.
- ❖ Integration with other SCM elements is essential.
- ❖ Promotes quality, consistency, and accountability

References

1. Leon, A. (2015). Software Configuration Management Handbook (3rd ed.). Norwood: Artechhouse. Retrieved September 4, 2025.
2. OpenAI. (2025, September 4). SCM history and concepts [AI-generated image]. ChatGPT (Sora). <https://chat.openai.com/>
3. OpenAI. (2025, October 16). Software change assurance [AI-generated image]. ChatGPT (Sora). <https://chat.openai.com/>

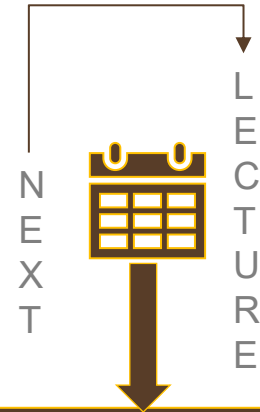


Thank You!

SOFTWARE CONFIGURATION MANAGEMENT



Figure 3: Concepts of SCM (Source: OpenAI, 2025)



**Software Configuration
Status Accounting**