Course: Software Configuration Management

Week 10: Software Configuration Auditing

Lecturer: Yimer Amedie (MSc.)
Addis Ababa Science and Technology University, Ethiopia

October, 2025

Contents

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

- Introduction and Purpose of Configuration Audits
- Verification and Its Relationship with Auditing
- Types of Configuration Audits
- Configuration Audit Process and Roles
- Audit Reporting and Post-Audit Activities
- Summary

Learning Outcomes

After completing this lesson, you will be able to:

- Define software configuration audit and explain its purpose.
- Differentiate between baseline, functional, physical, and compliance audits.
- Describe the roles and responsibilities in conducting configuration audits.
- Explain how audits ensure baseline integrity and compliance.
- Apply configuration auditing concepts to a sample project

Introduction to Configuration Audit

- **What is Configuration Audit (CA)?**
  - A **formal examinations** ensuring that **software and documentation conform** to approved **baselines and compliance standards**.
  - A **check** to verify that the **product package contains** all of the **components** it is supposed to **contain and performs as promised** (Leon, 2015).

Introduction to Configuration Audit … (2)

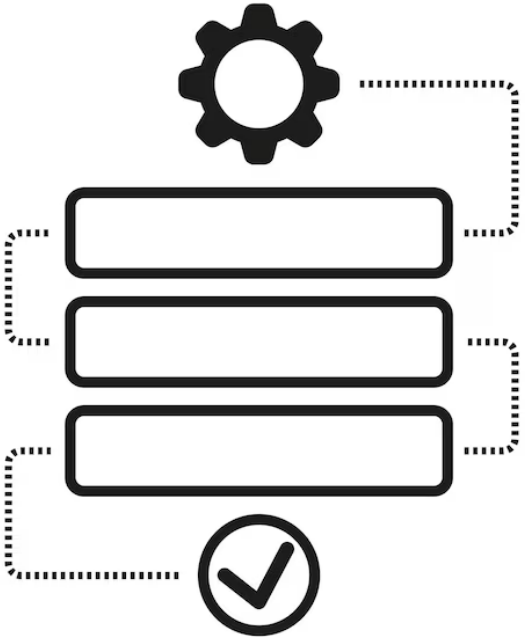
What is Configuration Audit (CA)?

- Conducted before system delivery or release
- Confirms product and documentation consistency
- Supports compliance and certification readiness
- Strengthens configuration control discipline

Purpose of Configuration Audit

- **Configuration Audit (CA):**
  - Verify conformance to approved baselines
  - Ensure completeness and correctness of items
  - Detect discrepancies and unauthorized changes
  - Maintain traceability and accountability
  - Support quality assurance and acceptance

Relationship b/n Verification and Audit



- Verification precedes audit activities
- Audits confirm verified results and records
- Verification checks implementation accuracy
- Audit ensures documentation consistency
- Together ensure full configuration integrity

Scope of Configuration Audits

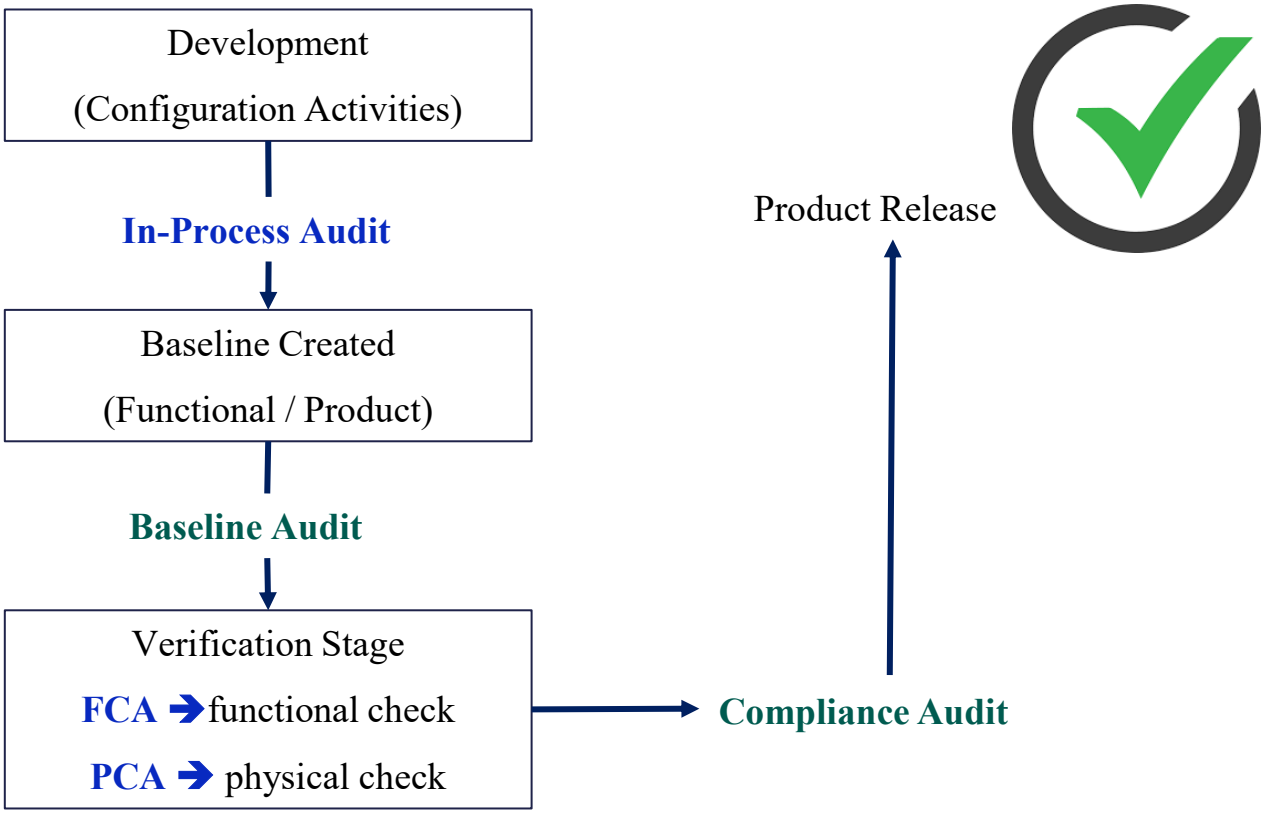
- Applies to all baselined items
- Involves technical and administrative review
- Conducted before release milestones
- Focuses on conformance and completeness
- Supports lifecycle quality assurance

Types of Audit

1. Functional Configuration Audit
2. Physical Configuration Audit
3. In-process Configuration Audit

    - ❖ Baseline Audit
    - ❖ Compliance Audit

Types of Audit … (2)

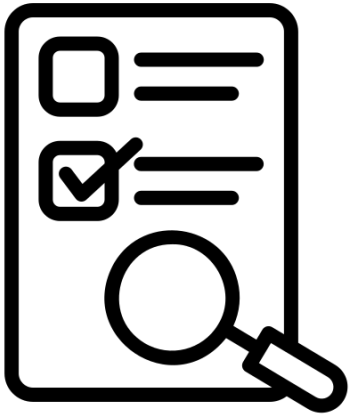


Baseline Audit

- Baseline audit:
  - Conducted before Functional or Physical Audit
  - Confirms all approved items are included
  - Detects missing or unauthorized elements
  - Validates readiness for detailed audits
  - Reinforces baseline integrity

Baseline Audit Process

- Identify configuration baselines for review
- Compare baseline records with repository data
- Validate approvals and signatures
- Record audit findings and discrepancies
- Approve baseline readiness for FCA/PCA



Importance of Baseline Audit

- Baseline audit
  - ✓ Prevents unapproved configuration changes
  - ✓ Ensures accuracy of project documentation
  - ✓ Builds confidence for subsequent audits
  - ✓ Enables controlled version management
  - ✓ Supports quality and compliance assurance

Baseline Audit: Example OLMS Project

- LMS course management baseline verified
- Repository matched with configuration log
- Unauthorized changes detected and reported
- Audit report approved by configuration board
- LMS baseline certified for Functional Audit

Functional Configuration Audit (FCA)

- FCA is a formal review to verify that the system or software meets all functional and performance requirements in the approved specifications.
  - Evaluates system functionality and performance
  - Conducted after system verification and testing
  - Confirms system meets functional requirements
  - Checks that changes are properly implemented
  - Provides input for compliance certification

Does the product do what it's supposed to do?

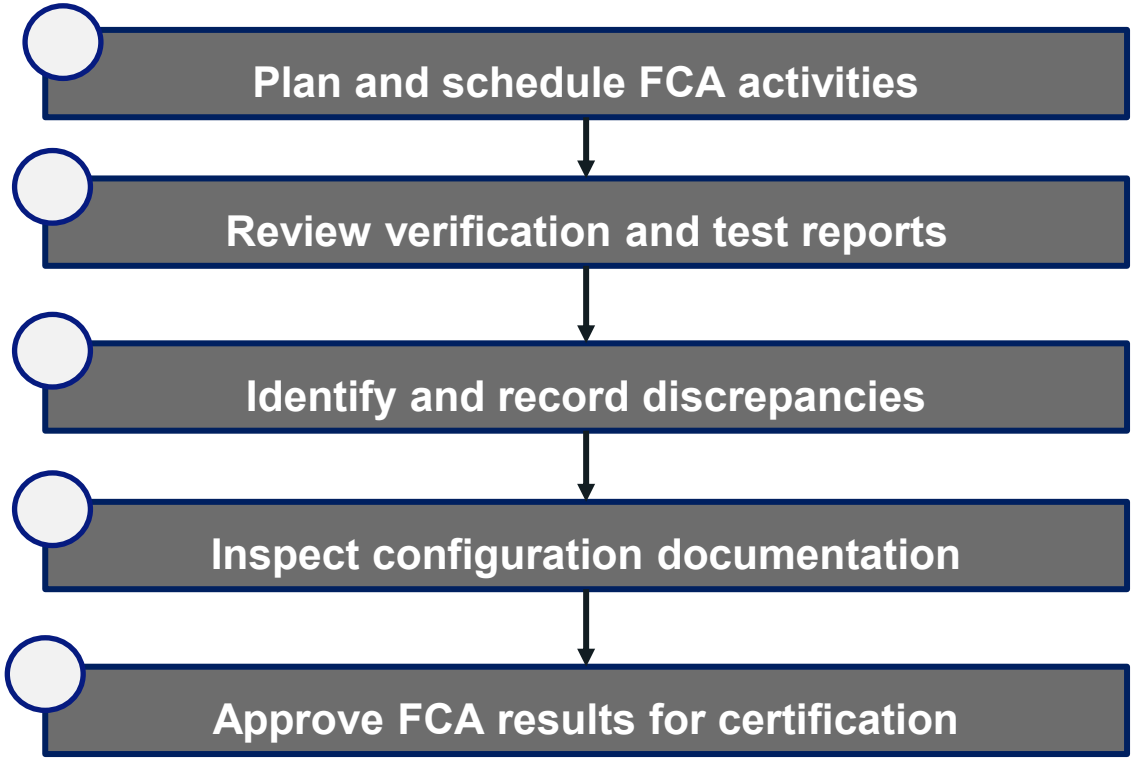
Objectives of FCA

- **The objective of FCA is to:**
  - ✓ Verify fulfillment of functional requirements
  - ✓ Confirm defect corrections are validated
  - ✓ Evaluate completeness of test results
  - ✓ Assess documentation accuracy and traceability
  - ✓ Ensure readiness for release or delivery

Scope of FCA

- FCA covers a wide scope
  - Examine all functional baseline items
  - Includes software modules and subsystems
  - Validates test procedures and outcomes
  - Checks interface and integration behavior
  - Reviews performance against specified limits

FCA Process


Plan and schedule FCA activities
Review verification and test reports
Identify and record discrepancies
Inspect configuration documentation
Approve FCA results for certification

FCA Checklists

- An FCA checklist helps auditors ensure completeness and accuracy
  - Verify implementation of all approved requirements
  - Check functional traceability matrix completeness
  - Validate all test cases executed successfully
  - Confirm configuration documentation accuracy
  - Ensure corrective actions are fully implemented

FCA Roles & Responsibilities

- Several key roles support the FCA
  - **Lead Auditor:** directs FCA execution
  - **Developers:** provide evidence and clarifications
  - **QA Staff:** supply test and verification data
  - **SCM Representative:** ensures baseline control
  - **Project Manager:** approves audit results

FCA and Compliance Relationship

- **The FCA directly supports compliance**
  - It validates contractual and functional compliance
  - Demonstrates adherence to standards and policies
  - Ensures functional baseline integrity
  - Confirms customer-specified deliverables
  - Supports compliance audits and certifications

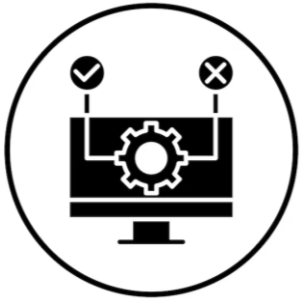
FCA Findings and Reporting

- Record nonconformances and observations
- Classify findings by severity and type
- Recommend corrective actions and re-tests
- Summarize FCA report and approval status
- Archive findings in configuration records

FCA Example: OLMS Project

- FCA conducted after testing LMS modules
- Verified user roles and access functionalities
- Reviewed grading and attendance management
- Checked interface consistency across components
- Approved LMS functional baseline for release

Key Benefits of Functional Audits



- ✓ Confirms correctness of system functionality
- ✓ Ensures documentation and performance alignment
- ✓ Improves user and stakeholder confidence
- ✓ Reduces risk of post-release defects
- ✓ Strengthens configuration control and traceability

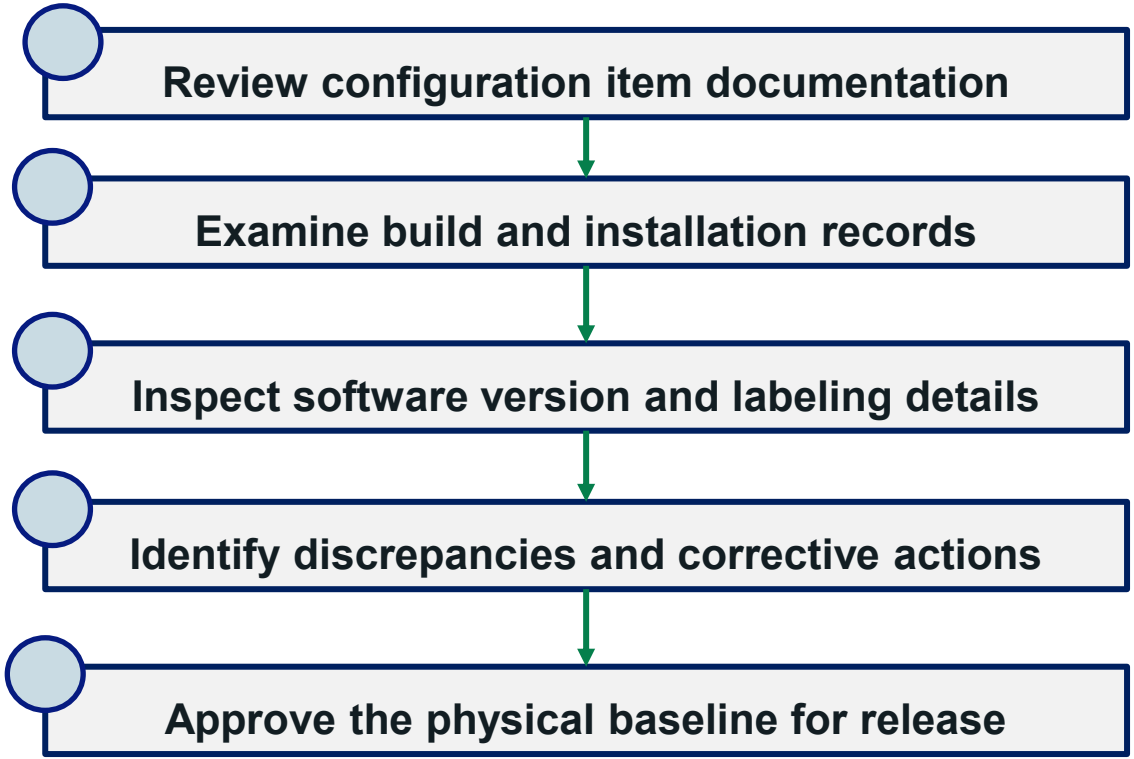
Physical Configuration Audit (PCA)

- PCA is a formal review to verify that the **as-built product** and **documentation** exactly match the approved configuration baseline
  - Conducted after Functional Configuration Audit (FCA)
  - Verifies the product matches design documentation
  - Confirms inclusion of all approved changes
  - Ensures product completeness and correctness
  - Certifies the final physical baseline

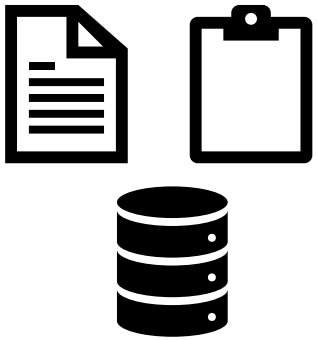
Purpose of PCA

- Ensure the delivered system is built and documented as approved
  - Validate product build consistency with design
  - Confirm documentation accuracy and revision control
  - Verify configuration identifiers and part numbers
  - Ensure no deviations from authorized design
  - Establish product readiness for release

PCA Process

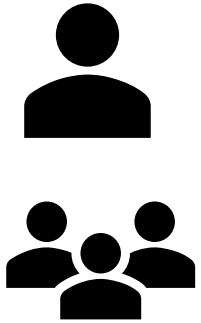

Review configuration item documentation
Examine build and installation records
Inspect software version and labeling details
Identify discrepancies and corrective actions
Approve the physical baseline for release

PCA Documentation and Records



- PCA documentation includes all materials that describe the **final system configuration**
  - ✓ Includes as-built configuration documentation
  - ✓ Contains installation and user manuals
  - ✓ Lists verified components and versions
  - ✓ Provides traceability between documents and product
  - ✓ Retained as part of audit evidence

Roles and Responsibilities in PCA



- Several key roles support the PCA
  - **Lead Auditor:** manages PCA execution process
  - **Developers:** provide build and release evidence
  - **QA Team:** validates product quality and records
  - **SCM Representative:** ensures configuration control
  - **Project Manager:** approves audit completion

PCA Example:
OLMS Project

- Conducted after FCA completion
- Verified LMS source code and build outputs
- Cross-checked release documentation accuracy
- Confirmed deployment readiness on test servers
- Certified physical baseline for production use

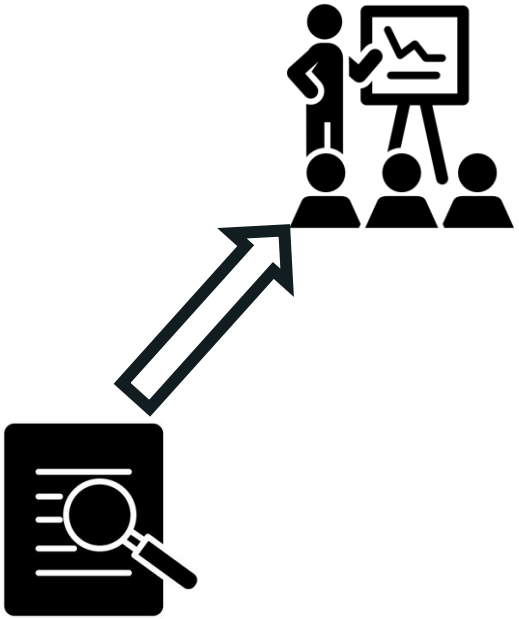
Compliance Audits

- Ensure adherence to standards and regulations
- Verify conformity to contractual requirements
- Confirm process compliance with SCM policies
- Identify gaps or deviations for correction
- Provide assurance of organizational accountability

Relationship b/n Compliance & Configuration Audits

- Compliance forms the foundation for configuration audits
- Audits verify compliance through baseline assessments
- FCA & PCA validate implementation against standards
- Noncompliance triggers corrective action processes
- Supports certification and accreditation readiness

Compliance Verification Process



- Identify applicable standards and regulations
- Develop compliance checklist and criteria
- Review artifacts and test evidence
- Record compliance gaps and findings
- Report compliance status to CCB

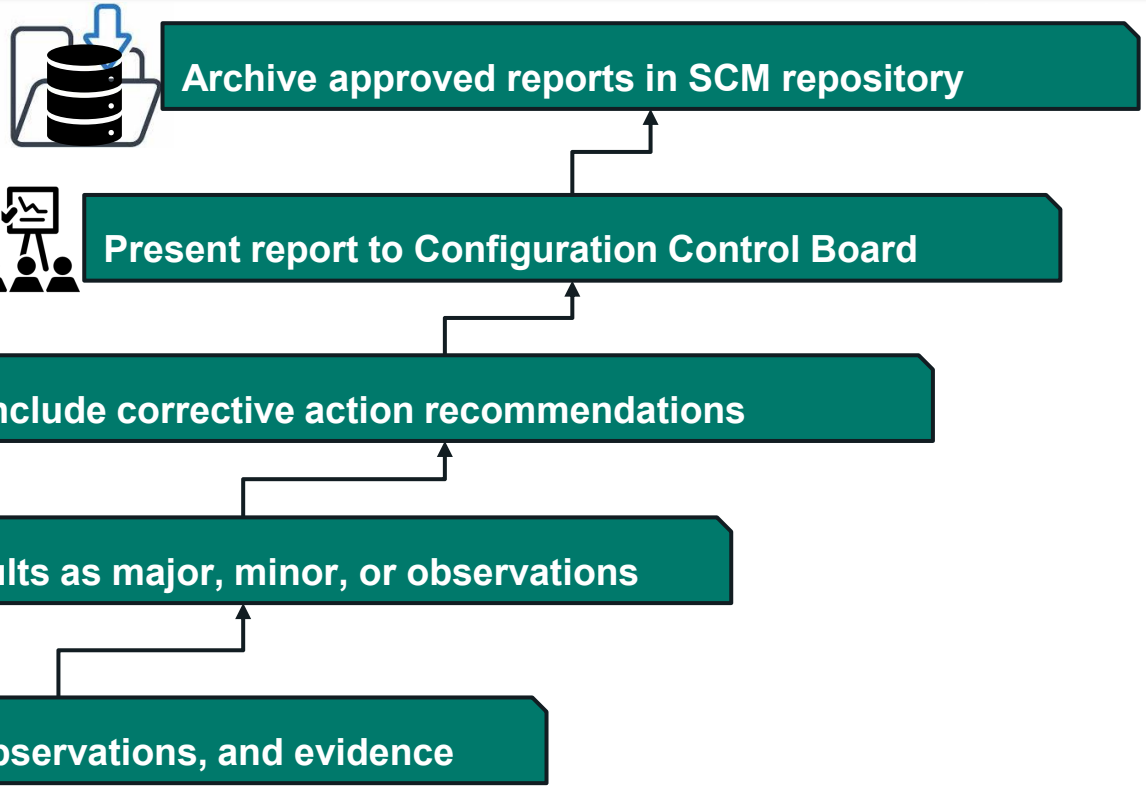
Benefits of Compliance Audits

- Compliance audits offer significant benefits
  - Reinforces adherence to SCM and QA policies
  - Improves audit readiness and process maturity
  - Builds client and stakeholder confidence
  - Reduces risks of nonconformance penalties
  - Ensures continuous improvement through feedback

Conducting Configuration Audits

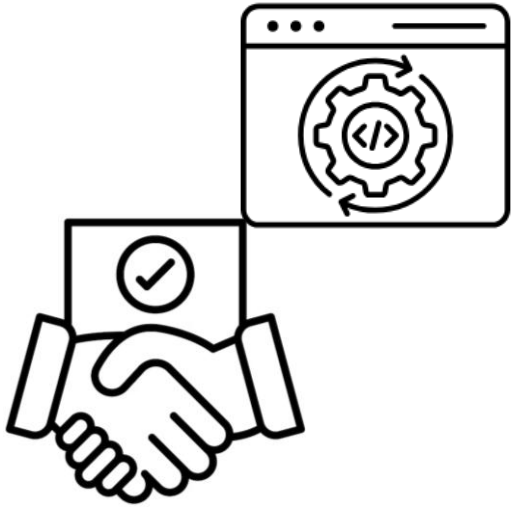
- Conducting configuration audits involves several structured steps
  - Develop detailed audit plan and scope
  - Assign audit team and roles
  - Collect configuration documentation and evidence
  - Execute review and verification activities
  - Record findings and observations

Audit Reporting Process

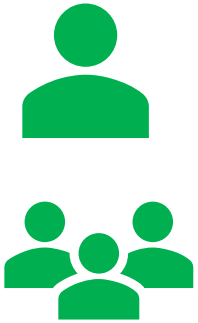


Post-Audit Activities

- ✓ Implement corrective actions and re-tests
- ✓ Verify closure of audit findings
- ✓ Update baseline documentation and records
- ✓ Conduct follow-up audit if needed
- ✓ Communicate results to stakeholders



Roles in Audit Execution



- Several key roles support the Audit Execution:
  - **Auditor:** Conducts independent assessment
  - **QA Manager:** Oversees quality compliance aspects
  - **SCM Lead:** Maintains baseline and evidence records
  - **Project Manager:** Supports logistics and resolution
  - **CCB:** Approves audit outcomes and actions

Integrating Functional and Physical Audits

- FCA and PCA complement each other
- FCA ensures functionality correctness
- PCA ensures documentation and build consistency
- Combined audits enhance compliance accuracy
- Integration strengthens overall configuration integrity

Integrated Audit Example: OLMS Project

- Baseline audit verified approved LMS modules
- FCA confirmed learning workflow and grading accuracy
- PCA validated source code and release documentation
- Compliance audit reviewed standard adherence
- Audit closure certified LMS for deployment

Importance of Configuration Audits

- Prevent discrepancies between product and documents
- Maintain baseline accuracy and consistency
- Support regulatory and quality certifications
- Enhance system reliability and traceability
- Reinforce disciplined SCM practices

Challenges in Configuration Auditing

- ≠ Incomplete or outdated documentation
- ≠ Limited audit resources or time constraints
- ≠ Poor version control practices
- ≠ Resistance to corrective actions
- ≠ Managing multiple audit findings

Summary

- Configuration audits ensure software and documentation alignment.
- FCA and PCA verify both functionality and physical integrity.
- Compliance audits uphold standards and regulations.
- Audits strengthen traceability and configuration discipline.
- Integrated audits improve overall software quality assurance

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/

Thank You!

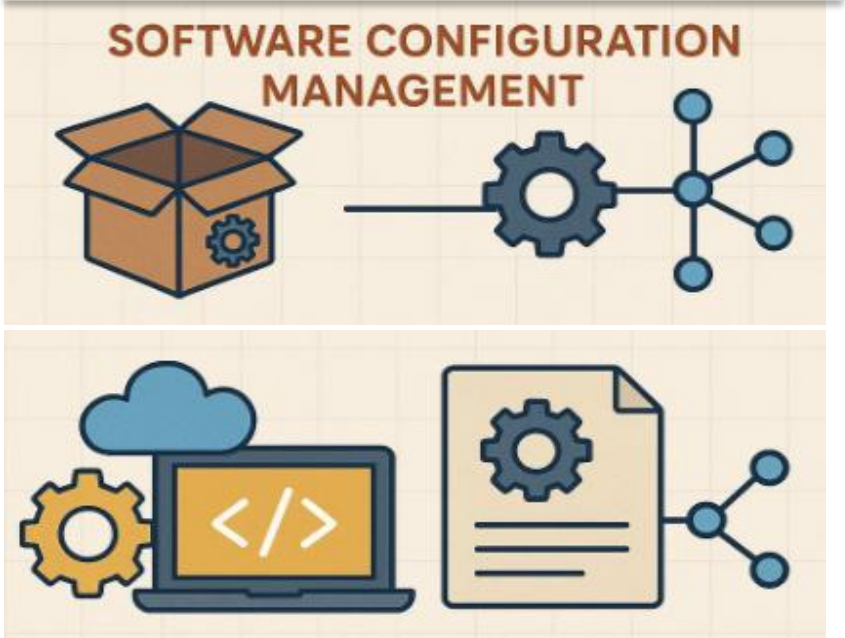


Figure 2: Concepts of SCM (Source: OpenAI, 2025)

NEXT LECTURE

Software Release Management and Delivery