

CONFIGURATION MANAGEMENT

A Support Process Area at Maturity Level 2

Purpose

The purpose of Configuration Management (CM) is to establish and maintain the integrity of work products using configuration identification, configuration control, configuration status accounting, and configuration audits.

Introductory Notes

The Configuration Management process area involves the following activities:

- Identifying the configuration of selected work products that compose baselines at given points in time
- Controlling changes to configuration items
- Building or providing specifications to build work products from the configuration management system
- Maintaining the integrity of baselines
- Providing accurate status and current configuration data to developers, end users, and customers

The work products placed under configuration management include the products that are delivered to the customer, designated internal work products, acquired products, tools, and other items used in creating and describing these work products. (See the definition of “configuration management” in the glossary.)

Examples of work products that can be placed under configuration management include the following:

- Hardware and equipment
- Drawings
- Product specifications
- Tool configurations
- Code and libraries
- Compilers
- Test tools and test scripts
- Installation logs
- Product data files
- Product technical publications
- Plans
- User stories
- Iteration backlogs
- Process descriptions
- Requirements
- Architecture documentation and design data
- Product line plans, processes, and core assets

Acquired products may need to be placed under configuration management by both the supplier and the project. Provisions for conducting configuration management should be established in supplier agreements. Methods to ensure that data are complete and consistent should be established and maintained.

Refer to the Supplier Agreement Management process area for more information about establishing supplier agreements.

Configuration management of work products can be performed at several levels of granularity. Configuration items can be decomposed into configuration components and configuration units. Only the term “configuration item” is used in this process area. Therefore, in these practices, “configuration item” may be interpreted as “configuration component” or “configuration unit” as appropriate. (See the definition of “configuration item” in the glossary.)

Baselines provide a stable basis for the continuing evolution of configuration items.

An example of a baseline is an approved description of a product that includes internally consistent versions of requirements, requirement traceability matrices, design, discipline-specific items, and end-user documentation.

Baselines are added to the configuration management system as they are developed. Changes to baselines and the release of work products built

from the configuration management system are systematically controlled and monitored via the configuration control, change management, and configuration auditing functions of configuration management.

This process area applies not only to configuration management on projects but also to configuration management of organizational work products such as standards, procedures, reuse libraries, and other shared supporting assets.

Configuration management is focused on the rigorous control of the managerial and technical aspects of work products, including the delivered product or service.

This process area covers the practices for performing the configuration management function and is applicable to all work products that are placed under configuration management.

For product lines, configuration management involves additional considerations due to the sharing of core assets across the products in the product line and across multiple versions of core assets and products. (See the definition of “product line” in the glossary.)

In Agile environments, configuration management (CM) is important because of the need to support frequent change, frequent builds (typically daily), multiple baselines, and multiple CM supported workspaces (e.g., for individuals, teams, and even for pair-programming). Agile teams may get bogged down if the organization doesn't: 1) automate CM (e.g., build scripts, status accounting, integrity checking) and 2) implement CM as a single set of standard services. At its start, an Agile team should identify the individual who will be responsible to ensure CM is implemented correctly. At the start of each iteration, CM support needs are re-confirmed. CM is carefully integrated into the rhythms of each team with a focus on minimizing team distraction to get the job done. (See “Interpreting CMMI When Using Agile Approaches” in Part I.)

Related Process Areas

Refer to the Project Monitoring and Control process area for more information about monitoring the project against the plan and managing corrective action to closure.

Refer to the Project Planning process area for more information about developing a project plan.

Specific Goal and Practice Summary

SG 1 Establish Baselines

- SP 1.1 Identify Configuration Items
- SP 1.2 Establish a Configuration Management System
- SP 1.3 Create or Release Baselines

SG 2 Track and Control Changes

- SP 2.1 Track Change Requests
- SP 2.2 Control Configuration Items

SG 3 Establish Integrity

- SP 3.1 Establish Configuration Management Records
- SP 3.2 Perform Configuration Audits

Specific Practices by Goal

SG 1 Establish Baselines

Baselines of identified work products are established.

Specific practices to establish baselines are covered by this specific goal. The specific practices under the Track and Control Changes specific goal serve to maintain the baselines. The specific practices of the Establish Integrity specific goal document and audit the integrity of the baselines.

SP 1.1 Identify Configuration Items

Identify configuration items, components, and related work products to be placed under configuration management.

Configuration identification is the selection and specification of the following:

- Products delivered to the customer
- Designated internal work products
- Acquired products
- Tools and other capital assets of the project's work environment
- Other items used in creating and describing these work products

Configuration items can include hardware, equipment, and tangible assets as well as software and documentation. Documentation can include requirements specifications and interface documents. Other documents that serve to identify the configuration of the product or service, such as test results, may also be included.

A "configuration item" is an entity designated for configuration management, which may consist of multiple related work products that form a baseline. This logical grouping provides ease of identification and controlled access. The selection of work products for configuration management should be based on criteria established during planning.

Example Work Products

1. Identified configuration items

Subpractices

1. Select configuration items and work products that compose them based on documented criteria.

Example criteria for selecting configuration items at the appropriate work product level include the following:

- Work products that can be used by two or more groups
- Work products that are expected to change over time either because of errors or changes in requirements
- Work products that are dependent on each other (i.e., a change in one mandates a change in the others)
- Work products critical to project success

Examples of work products that may be part of a configuration item include the following:

- Design
- Test plans and procedures
- Test results
- Interface descriptions
- Drawings
- Source code
- User stories or story cards
- The declared business case, logic, or value
- Tools (e.g., compilers)
- Process descriptions
- Requirements

2. Assign unique identifiers to configuration items.
3. Specify the important characteristics of each configuration item.

Example characteristics of configuration items include author, document or file type, programming language for software code files, minimum marketable features, and the purpose the configuration item serves.

4. Specify when each configuration item is placed under configuration management.

Example criteria for determining when to place work products under configuration management include the following:

- When the work product is ready for test
- Stage of the project lifecycle
- Degree of control desired on the work product
- Cost and schedule limitations
- Stakeholder requirements

5. Identify the owner responsible for each configuration item.

6. Specify relationships among configuration items.

Incorporating the types of relationships (e.g., parent-child, dependency) that exist among configuration items into the configuration management structure (e.g., configuration management database) assists in managing the effects and impacts of changes.

SP 1.2 Establish a Configuration Management System

Establish and maintain a configuration management and change management system for controlling work products.

A configuration management system includes the storage media, procedures, and tools for accessing the system. A configuration management system can consist of multiple subsystems with different implementations that are appropriate for each configuration management environment.

A change management system includes the storage media, procedures, and tools for recording and accessing change requests.

Example Work Products

1. Configuration management system with controlled work products
2. Configuration management system access control procedures
3. Change request database

Subpractices

1. Establish a mechanism to manage multiple levels of control.

The level of control is typically selected based on project objectives, risk, and resources. Control levels can vary in relation to the project lifecycle, type of system under development, and specific project requirements.

Example levels of control include the following:

- Uncontrolled: Anyone can make changes.
- Work-in-progress: Authors control changes.
- Released: A designated authority authorizes and controls changes and relevant stakeholders are notified when changes are made.

Levels of control can range from informal control that simply tracks changes made when configuration items are being developed to formal configuration control using baselines that can only be changed as part of a formal configuration management process.

2. Provide access control to ensure authorized access to the configuration management system.
3. Store and retrieve configuration items in a configuration management system.
4. Share and transfer configuration items between control levels in the configuration management system.

5. Store and recover archived versions of configuration items.
6. Store, update, and retrieve configuration management records.
7. Create configuration management reports from the configuration management system.
8. Preserve the contents of the configuration management system.

Examples of preservation functions of the configuration management system include the following:

- Backup and restoration of configuration management files
- Archive of configuration management files
- Recovery from configuration management errors

9. Revise the configuration management structure as necessary.

SP 1.3 Create or Release Baselines

Create or release baselines for internal use and for delivery to the customer.

A baseline is represented by the assignment of an identifier to a configuration item or a collection of configuration items and associated entities at a distinct point in time. As a product or service evolves, multiple baselines can be used to control development and testing. (See the definition of “baseline” in the glossary.)

Hardware products as well as software and documentation should also be included in baselines for infrastructure related configurations (e.g., software, hardware) and in preparation for system tests that include interfacing hardware and software.

One common set of baselines includes the system level requirements, system element level design requirements, and the product definition at the end of development/beginning of production. These baselines are typically referred to respectively as the “functional baseline,” “allocated baseline,” and “product baseline.”

A software baseline can be a set of requirements, design, source code files and the associated executable code, build files, and user documentation (associated entities) that have been assigned a unique identifier.

Example Work Products

1. Baselines
2. Description of baselines

Subpractices

1. Obtain authorization from the CCB before creating or releasing baselines of configuration items.
2. Create or release baselines only from configuration items in the configuration management system.

3. Document the set of configuration items that are contained in a baseline.
4. Make the current set of baselines readily available.

SG 2 Track and Control Changes

Changes to the work products under configuration management are tracked and controlled.

The specific practices under this specific goal serve to maintain baselines after they are established by specific practices under the Establish Baselines specific goal.

SP 2.1 Track Change Requests

Track change requests for configuration items.

Change requests address not only new or changed requirements but also failures and defects in work products.

Change requests are analyzed to determine the impact that the change will have on the work product, related work products, the budget, and the schedule.

Example Work Products

1. Change requests

Subpractices

1. Initiate and record change requests in the change request database.
2. Analyze the impact of changes and fixes proposed in change requests.

Changes are evaluated through activities that ensure that they are consistent with all technical and project requirements.

Changes are evaluated for their impact beyond immediate project or contract requirements. Changes to an item used in multiple products can resolve an immediate issue while causing a problem in other applications.

Changes are evaluated for their impact on release plans.

3. Categorize and prioritize change requests.

Emergency requests are identified and referred to an emergency authority if appropriate.

Changes are allocated to future baselines.

4. Review change requests to be addressed in the next baseline with relevant stakeholders and get their agreement.

Conduct the change request review with appropriate participants. Record the disposition of each change request and the rationale for the decision, including success criteria, a brief action plan if appropriate, and needs met or unmet by the change. Perform the actions required in the disposition and report results to relevant stakeholders.

5. Track the status of change requests to closure.

Change requests brought into the system should be handled in an efficient and timely manner. Once a change request has been processed, it is critical to close the request with the appropriate approved action as soon as it is practical. Actions left open result in larger than necessary status lists, which in turn result in added costs and confusion.

SP 2.2 Control Configuration Items

Control changes to configuration items.

Control is maintained over the configuration of the work product baseline. This control includes tracking the configuration of each configuration item, approving a new configuration if necessary, and updating the baseline.

Example Work Products

1. Revision history of configuration items
2. Archives of baselines

Subpractices

1. Control changes to configuration items throughout the life of the product or service.
2. Obtain appropriate authorization before changed configuration items are entered into the configuration management system.

For example, authorization can come from the CCB, the project manager, product owner, or the customer.

3. Check in and check out configuration items in the configuration management system for incorporation of changes in a manner that maintains the correctness and integrity of configuration items.

Examples of check-in and check-out steps include the following:

- Confirming that the revisions are authorized
- Updating the configuration items
- Archiving the replaced baseline and retrieving the new baseline
- Commenting on the changes made to the item
- Tying changes to related work products such as requirements, user stories, and tests

4. Perform reviews to ensure that changes have not caused unintended effects on the baselines (e.g., ensure that changes have not compromised the safety or security of the system).
5. Record changes to configuration items and reasons for changes as appropriate.

If a proposed change to the work product is accepted, a schedule is identified for incorporating the change into the work product and other affected areas.

Configuration control mechanisms can be tailored to categories of changes. For example, the approval considerations could be less stringent for component changes that do not affect other components.

Changed configuration items are released after review and approval of configuration changes. Changes are not official until they are released.

SG 3 Establish Integrity

Integrity of baselines is established and maintained.

The integrity of baselines, established by processes associated with the Establish Baselines specific goal, and maintained by processes associated with the Track and Control Changes specific goal, is addressed by the specific practices under this specific goal.

SP 3.1 Establish Configuration Management Records

Establish and maintain records describing configuration items.

Example Work Products

1. Revision history of configuration items
2. Change log
3. Change request records
4. Status of configuration items
5. Differences between baselines

Subpractices

1. Record configuration management actions in sufficient detail so the content and status of each configuration item is known and previous versions can be recovered.
2. Ensure that relevant stakeholders have access to and knowledge of the configuration status of configuration items.

Examples of activities for communicating configuration status include the following:

- Providing access permissions to authorized end users
- Making baseline copies readily available to authorized end users
- Automatically alerting relevant stakeholders when items are checked in or out or changed, or of decisions made regarding change requests

3. Specify the latest version of baselines.
4. Identify the version of configuration items that constitute a particular baseline.
5. Describe differences between successive baselines.
6. Revise the status and history (i.e., changes, other actions) of each configuration item as necessary.

SP 3.2 Perform Configuration Audits

Perform configuration audits to maintain the integrity of configuration baselines.

Configuration audits confirm that the resulting baselines and documentation conform to a specified standard or requirement. Configuration item related records can exist in multiple databases or configuration management systems. In such instances, configuration audits should extend to these other databases as appropriate to ensure accuracy, consistency, and completeness of configuration item information. (See the definition of “configuration audit” in the glossary.)

Examples of audit types include the following:

- Functional configuration audits (FCAs): Audits conducted to verify that the development of a configuration item has been completed satisfactorily, that the item has achieved the functional and quality attribute characteristics specified in the functional or allocated baseline, and that its operational and support documents are complete and satisfactory.
- Physical configuration audits (PCAs): Audits conducted to verify that a configuration item, as built, conforms to the technical documentation that defines and describes it.
- Configuration management audits: Audits conducted to confirm that configuration management records and configuration items are complete, consistent, and accurate.

Example Work Products

1. Configuration audit results
2. Action items

Subpractices

1. Assess the integrity of baselines.
2. Confirm that configuration management records correctly identify configuration items.
3. Review the structure and integrity of items in the configuration management system.
4. Confirm the completeness, correctness, and consistency of items in the configuration management system.

Completeness, correctness, and consistency of the configuration management system's content are based on requirements as stated in the plan and the disposition of approved change requests.

5. Confirm compliance with applicable configuration management standards and procedures.
6. Track action items from the audit to closure.