COMPANY NAME

SOFTWARE CONFIGURATION MANAGEMENT PROCEDURE

Revision History

Update the revision information in this section each time the procedure is changed and re-approved.

Document Version	Revision Date	Originator	Revision Description
1.0	xx-xx20xx	Flo Samuels	Initial Release.
1.1	xx-xx20xx	Flo Samuels	Reason for Revisions
1.2	xx-xx20xx	Flo Samuels	Reason for Revisions
1.3	xx-xx20xx	Flo Samuels	Reason for Revisions
1.4	xx-xx20xx	Flo Samuels	Reason for Revisions

COMPANY NAME

SOFTWARE CONFIGURATION MANAGEMENT PROCEDURE

Table of Contents

COMPANY NAME

SOFTWARE CONFIGURATION MANAGEMENT PROCEDURE

1. Introduction

1.1. Purpose

This procedure details the software configuration management activities that are required to ensure the integrity of a project's work products is properly established and maintained throughout the software's life cycle.

1.2. Objectives

1. The components of a system SCM Plan.
2. How the SCM Plan is used to identify and manage configuration items (CIs).
3. How ClearCase is used in software configuration management.
4. The purpose of status accounting reviews and status audits.

1.3. Definitions

Build: Compiling a baseline configuration into a testable work product.

Configuration Item (CIs): A work product that is placed under software configuration management control.

Configuration: A group consisting of a single version of one or more CIs.

Configuration Control: The recording and tracking of project change requests, as well as maintaining versions of CI's and Baselines throughout the life of the project.

Configuration Identification: The recording of the CI's (work products treated as a single entity) to be controlled for the project and the libraries in which they will be retained.

Configuration Management: A discipline applying technical and administrative direction and surveillance to identify and document the functional and physical characteristics of a configuration item, control changes to those characteristics, record and report change processing and implementation status, and verify compliance with specified requirements.

Configuration Release: The planning and scheduling of releasing applications into the production environment.

Development Manager: Responsible for the design, code development, testing, and defect correction of new and existing systems prior to deployment to production. May also be responsible for use cases and/or requirements.

Manager: Indicates the Product, Project, or Development Manager.

Product Manager: Responsible for interface with internal and external customers relative to the development of use cases and requirements for new systems, for changes to systems under development, or for enhancements to systems in production. Responsible for managing the overall product direction, scope, and market timing of product releases.

Project Manager: Responsible for ensuring the project plan is developed according to written procedures. Organizes, tracks, updates, and communicates the status of a

project. Maintains the balance between scope, schedule, and resources. Manages issues and their timely resolution. Advises the appropriate senior management on potential deviations in scope, schedule, and resources, and elevates unresolved issues to ensure the appropriate action plans are developed to alleviate problems. Negotiates commitment changes with affected parties and reports the results to senior management. Ensures negotiated action plans are implemented and project plans are adjusted accordingly.

Release: A Baseline that has been formally approved for move to production.

Release Manager: Manages source code configurations, change control, and the source code control system. Responsible for ensuring that all software source code builds are performed according to defined standards. Responsible for ensuring release-ready packages include release notes, setup scripts, and are properly deployed to the implementation team.

Software Configuration Management: A means to establish and maintain the integrity of the work products of the software project throughout the project's life cycle.

Version: An instance or occurrence of a Configuration Item, Baseline or Release.

VOB: Versioned Object Bases. Repositories, implemented as VOB storage directories, which hold all versions of individual CI's in a database.

2. Procedure Description

2.1. Overview

When Software Configuration Management (SCM) involves identifying selected software work products to be placed under version control (configuration), controlling changes to the configuration, and maintaining the integrity and traceability of the configuration throughout the software life cycle. The work products that may be placed under SCM control include software products (e.g., software requirements document, code) and the items identified with or required to create these software products (e.g., the compiler). A work product might not be placed under configuration control if it is not critical to the system.

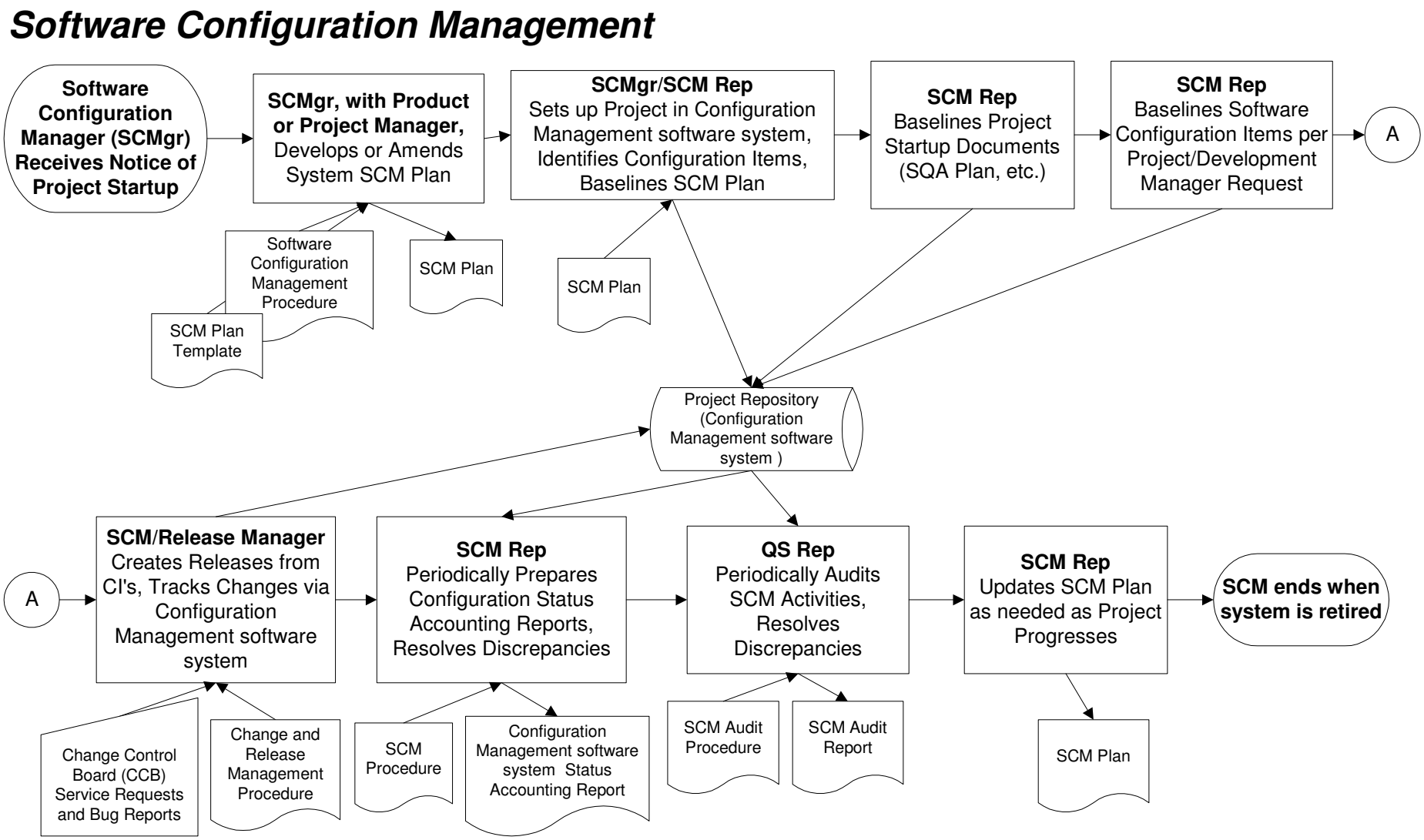
The process starts with preparation of a SCM plan for the System's SCM activities. This plan is updated with each new system-related project and used to identify and control configuration items throughout the project cycle. At Project Closure, the SCM process ensures the completeness of all project deliverables for release (deployment to production). SCM control continues until the system is retired.

SCM also includes reporting on the status and content of controlled items; logging, tracking and resolving discrepancy items, and identifying an approach to mitigating future discrepancies. SCM Configuration Audits and Reviews are used to identify and prevent problems from occurring at the time of release.

SOFTWARE CONFIGURATION MANAGEMENT PROCEDURE

Trigger	When preparation of Use Cases is started.	
Who	The Product and/or Project Manager, the Development Manager, and the SCM Representative develop/amend the SCM Plan and implement the Plan's activities. The SCM Representative prepares status accounting reports and final project SCM housekeeping prior to deployment to production. The QS Representative audits SCM activities for compliance with procedures and standards. After deployment, the Development Manager in Production is responsible for maintaining and implementing the SCM Plan.	
Inputs		*Doc control #*
	SCM Plan Template	
	SCM Procedure	
	SCM Audit Form	
	SCM Audit Findings Form	
	Discrepancy Log Template	
	Configuration Management software system Manual	
	Change and Release Management Procedure	
Outputs		*Doc control #*
	SCM Plan	
	Identification of project Configuration Items	
	Library Allocations (VOBs in Configuration Management software system)	
	Status Accounting Reports (Configuration Management software system)	
	Status Audits and Findings	
	Discrepancy Log	
	Baseline Release Inventory (Configuration Management software system)	

2.2. Flow Chart



Software Configuration Management

3. Procedure Steps

3.1. SCM Plan

3.1.1. Purpose

3.1.1.1. Used to direct SCM activities by defining, integrating, and updating the details for the system and the individual project's software configuration management.

3.1.1.2. Required for all projects and in force until the system is retired. The plan is updated as projects for the system are added and as configurations change.

3.1.2. Process

3.1.2.1. Each system will have a SCM Plan based on the SCM Plan Template. During the Requirements stage of a project, the SCM Representative updates the SCM Plan with project specific data.

3.1.2.2. The SCM Rep submits the updated SCM Plan to the Product/Project and Development Managers.

3.1.2.3. Based on feedback received, the SCM Rep revises the SCM Plan.

3.1.2.4. When approved by the Product/Project and Development Managers, the SCM Plan is saved as a CI in the System Plans library.

3.1.2.5. The SCM Rep verifies that the MS Project schedule has the standard SCM activities for each stage.

3.1.3. Updates to SCM Plan

3.1.3.1. The SCM Rep receives changes resulting from SCM activities (e.g., new version of an external CI, status accounting report discrepancies) and updates the SCM Plan.

3.2. Completing the Plan

3.2.1. Configuration Administration, Section 2.1.

3.2.1.1. Establishes standards for responding to requests, retaining versions on-line versus archive, tracking change requests, configuration control, and status accounting report and audit schedules.

3.2.1.2. The SCM Rep and Product/Project and Development Managers review the existing standards relative to the new project. Changes, if needed, are made to the table with notations indicating the project to which the changes apply.

3.2.1.3. The Product/Project or Development Manager indicates the path for storage of the SCM Plan (Section 2.1.2).

3.2.2. Configuration Identification and Control, Section 2.2

3.2.2.1. Identify SCM Libraries, Section 2.2.1

3.2.2.1.1. Establishes how libraries, called VOBs in the Configuration Management software system, will be set up so that project teams have a repository for storing and sharing

software work products (identified as configuration items), and their associated information, as they are developed.

3.2.2.1.2. Ensures that new libraries have unique identifiers linking them with the correct projects.

3.2.2.1.3. Provides control of configuration items and their availability, including archived versions.

3.2.2.1.4. Defines separate repositories or unique identifiers for the software baselines and software releases, including version numbers.

3.2.2.1.5. Controls builds and releases by ensuring that products from the software baseline library, for both internal and external use, are built only from CIs in the software CI library and their release is controlled according to the **Change and Release Management Procedure**.

3.2.2.2. Configuration Item Identification and Control, Section 2.2.2

3.2.2.2.1. Provides a consistent means to identify work products that will be placed under SCM.

3.2.2.2.2. Establishes CI category identification through a predetermined directory structure (Appendix A in the SCM Plan, which includes definitions of CI categories.)

3.2.2.2.3. The Configuration Management software system enables loading, naming, and numbering original and subsequent versions of CI's to provide for the storage and retrieval of CI versions and their archived versions.

3.2.2.2.4. Allows selecting additional CI categories for projects with unique development requirements.

3.2.2.2.5. Describes naming conventions to be used for storing project CIs.

3.2.2.3. External Configuration Item Identification, Section 2.2.3

3.2.2.3.1. Defines external programs that are used to develop or test the CIs but are not a part of the system.

3.2.2.4. Baseline and Release Identification and Control, Section 2.2.4

3.2.2.4.1. Ensures only configuration items approved by the Development Manager are identified as software baselines and included in releases.

3.2.2.4.2. Ensures changes to baselines and the archiving of baselines are controlled consistently across projects and/or application development efforts.

3.2.2.4.3. Ensures that each release can be traced to the correct baseline version used to create the release.

3.2.2.4.4. Describes the baseline and release naming conventions, and versioning system used to identify baselines and

releases in the Configuration Management software system.

3.2.2.5. Change Request Tracking, Section 2.2.5 (Configuration Control)

3.2.2.5.1. Controls Configuration Items and Baselines by tracking all change requests and their resulting impacts in a way that consistently identifies and versions CI's, Baselines and the libraries containing them.

3.2.2.5.2. Establishes how change requests, in the form of reported bugs or requested enhancements, are logged through the Defect Management software system to create an audit trail for reporting all SCM items and activities.

3.2.3. Configuration Status Accounting, Section 2.3

3.2.3.1. Ensures the following data elements are tracked and reported for each CI and Baseline:

- its initial approved version
- the status of requested changes
- the implementation of approved changes

3.2.3.2. Reports the status of work products placed under SCM control in a way that informs affected groups of the status and content of controlled items (Change Requests) compared to actual status (Configuration Management software system changes).

3.2.3.3. Enables status accounting reviews, and logging, tracking, and resolving discrepancy information.

3.2.3.4. Standard reports developed as a result of the status accounting include:

- Change Request reports from the Defect Management software system that provide summary information on the status of change requests
- Summary reports from the Configuration Management software system of changes made to controlled items
- Revision history of CI's

3.2.3.5. The SCM Rep compares the Defect Management software system and the Configuration Management software system information prior to holding a scheduled Configuration Status Accounting Review. If any status discrepancies are identified in the comparison or the review, the SCM Rep records each discrepancy in the Discrepancy Log (see SCM Discrepancy Log.doc).

3.2.3.6. The SCM Rep attempts to resolve each discrepancy with the Project Team and records resolutions in the Discrepancy Log.

3.2.3.7. The SCM Rep and project team take preventative action to ensure similar discrepancies do not occur in the future.

3.2.4. Software Configuration Management Audits, Section 2.4

3.2.4.1. Auditing a project's SCM activities and/or deliverables to identify any discrepancies between the SCM Plan and the projects actual activities and/or deliverables.

3.2.4.2. Summarizes each audit.

3.2.4.3. Information on the Audit Process, logging, tracking, and resolving audit discrepancies, and identifying an approach to mitigating future discrepancies is described in the SCM Audit document.

3.2.5. Configuration Release, Section 2.5

3.2.5.1. Prepare project deliverables for release to production in a way that ensures all the necessary release information pertaining to a release is available and accounted for.

3.2.5.2. Ensure release information is prepared consistently across projects and/or application development efforts

3.2.5.3. Control changes to releases in a way that allows only those baselines approved by the Release Manager to be identified as releases. Approved releases are administered per the **Change and Release Management Procedure** and are summarized as follows:

- The release stored in the release library has an identifier and version number, which follows the Configuration Management software system standards.
- The release identifier and location of the release is recorded in the SCM Plan.
- The contents (Baseline) of each Release are recorded in the Release Notes and the location of the Release Notes is recorded in the SCM Plan.
- Required Release version(s) have been archived.

3.2.6. SCM Plan Implementation, Section 3

3.2.6.1. Assigns role responsibilities for deliverables.

3.2.6.2. Describes the tools used to perform SCM activities.

3.2.6.3. Describes key assumptions and risks.

3.2.6.4. References the SCM Project Schedule location.

3.2.6.5. Defines SCM Effectiveness and Efficiency metrics.