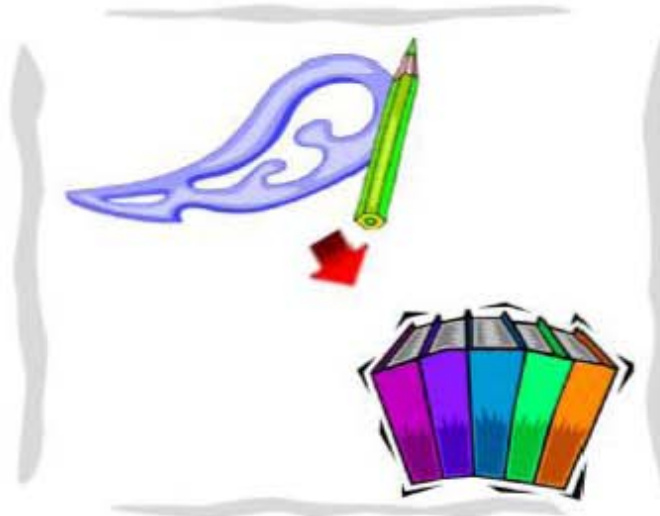


Smartwire Templates



Templates For Software Configuration Management Documents Version 6.0

SEPT Product Number 76 (Replaces 53)

Author: John M. Neorr

ISBN Number: 098-1952291

ISBN 13 Digit Number: 978-098195522-9-1

Produced by Software Engineering Process Technology (SEPT)
2725 NW Pine Cone Drive, Issaquah, WA. 98027
E-mail: Stanmagee@smartwire.net
Web Site: www.12207.com and www.15288.com

© 2004-2012. Software Engineering Process Technology (SEPT)
All rights reserved.

Templates For Software Configuration Management Documents

Version 6.0

ISBN Number: 098-1952291

ISBN 13 Digit Number: 978-098195522-9-1

SEPT Product Number 76 (Replaces 53)

Produced by Software Engineering Process Technology (SEPT)

2725 NW Pine Cone Drive

Issaquah, WA. 98027

Tel: 425-391-2344

E-mail: Stanmagee@smartwire.net

Web Pages: www.12207.com and www.15288.com

Author: John M. Neorr

©2004-2012 Software Engineering Process Technology (SEPT).

 **Documents**
Document Sharing Hub

Templates For Software Configuration Management Documents

Version 6.0

Abstract

This document contains a set of templates that enable engineers and managers to implement effective, standards-compliant software configuration management (CM). The templates found in this document conform to the software CM requirements specified in standards ISO/IEC 12207:2008, *Software life cycle processes*, ISO/IEC 15288:2008, *System life cycle processes*, and IEC 62304, *Medical Device Software – Software Life Cycle Processes*. They also provide a process framework consistent with ANSI/EIA-649-B, *Configuration Management Standard*. These templates provide generic descriptions and instructions that are compliant with the above standards and provide a structure within which you can include additional CM standard requirements as well as project-specific information. Once tailored, the templates become a complete set of documents that allow implementation of software configuration management compliant with both widely used standards and project-specific requirements.

Thirteen templates are provided. They are:

<u>Template #</u>	<u>Template Name</u>
1	Software Configuration Management Policy
2	Software Configuration Management Strategy
3	Software Configuration Management Organization Charts
4	Software Configuration Management Plan
5	Software Configuration Management Process
6	Software Configuration Identification Procedure
7	Software Configuration Change Control Procedure
8	Software Configuration Status Accounting Procedure
9	Software Configuration Item Check-in and Check-out Procedure
10	Software Configuration Audits and Reviews Procedure
11A	Subcontractor Software Configuration Control Procedure
11B	External Interface Configuration Control Procedure
12	Software Product Release and Delivery Procedure

In addition to the templates shown above, this document provides five different aids to help tailor the templates.

What's New in Version 6.0

Version 6.0 CM templates have been reviewed and updated to ensure consistency with the Software CM requirements of ANSI/EIA-649-B, *Configuration Managements Standard*. An appendix was added to describe the 649B standard and minor modifications were made to several of the templates to ensure they are compatible with the 649B standard while still remaining compliant with ISO/IEC 12207:2008, ISO/IEC 15288:2008, and IEC 62304.

Version 6.0 still contains the many illustrative diagrams introduced in Version 3.0. These diagrams will continue to provide added value in the implementation and communication of CM processes.

Version 6.0 Enhancement	Description
Reviewed and updated CM templates to ensure compatibility with the ANSI/EIA-649-B standard, <i>Configuration Management Standard</i> .	• Added a matrix in the introduction that maps the ANSI/EIA-649-B standard to our CM templates • Added a background and context section to the CM strategy template • Added a project overview section and a training section to the CM plan template • Added text in the status accounting template to ensure that process metrics are gathered and utilized to measure, monitor, and improve CM processes • Added activities in the audits and reviews template to ensure that: - Specification documents are complete and consistent - Test results and/or test reports confirm that the software meets all requirements
Added a review of ANSI/EIA-649-B standard	Added Appendix A, which provides an overview of the ANSI/EIA-649-B standard and discusses how it relates to the ISO/IEC 12207:2008 standard and our CM templates
Added a list of tables	Following the table of contents, a list of tables was added to facilitate rapid access to key data

Software Configuration Management Templates - Table of Contents

Table of Contents

1. Introduction.....	6
1.1 About Software Configuration Management	6
1.2 How to Use This Document	7
References to Standards	7
1.3 The Software Domain	9
1.4 Traceability to Configuration Management Standards	9
Overview	9
ISO/IEC 12207:2008 and ISO/IEC 15288:2008 Traceability	10
IEC 62304 Traceability.....	13
ANSI/EIA-649-B Traceability	14
1.5 Product Support.....	14
1.6 Warranties and Liability	14
2. Software Configuration Management Templates.....	15
2.1 Software Configuration Management Policy	17
2.2 Software Configuration Management Strategy	21
2.3 Software Configuration Management Organization Charts	29
2.4 Software Configuration Management Plan	33
2.5 Software Configuration Management Process	44
2.6 Software Configuration Identification Procedure	54
2.7 Software Configuration Change Control Procedure	60
2.8 Software Configuration Status Accounting Procedure	67
2.9 Software Configuration Item Check-in and Check-out Procedure	71
2.10 Software Configuration Audits and Reviews Procedure.....	76
2.11 External Interfaces Procedures	81
2.12 Software Product Release and Delivery Procedure.....	89
3. Software Configuration Management Aids.....	95
3.1 Aid 1: Acronyms	96
3.2 Aid 2: Standards	101
3.3 Aid 3: Software Change Request Data Elements	101
3.4 Aid 4: Software Status Accounting Data Elements.....	102
3.5 Aid 5: Software Documentation Approval Matrix.....	103
4. About the Author	107
Appendix A: Review of the ANSI/EIA-649-B Standard	108

Software Configuration Management Templates – Figures and Tables

List of Figures

Figure 1: <i>Software CM templates, and aids, and how they relate.</i>	8
Figure 2: <i>Standard's document sections and relationships.</i>	11
Figure 3: <i>ISO/IEC 12207:2008 Software CM Requirements Clauses.</i>	11
Figure 4: <i>CM Policy provides high-level guidance and control.</i>	17
Figure 5: <i>CM Strategy is guided by policy and focuses plans and procedures.</i>	21
Figure 6: <i>Software CM organization within a typical large company.</i>	30
Figure 7: <i>CM organization within a small company.</i>	31
Figure 8: <i>Organization of a typical software CM group.</i>	31
Figure 9: <i>The configuration management process showing all its inputs and outputs.</i>	45
Figure 10: <i>Controlled objects and their relationships and attributes.</i>	46
Figure 11: <i>Conceptual hierarchy of CM policies, plans, processes, and procedures.</i>	47
Figure 12: <i>Change control process flow.</i>	62
Figure 13: <i>Check-in/Check-out used by CM processes.</i>	71
Figure 14: <i>The check-in/check-out process.</i>	73
Figure 15: <i>Product release and delivery – the final step.</i>	89

List of Tables

Table 1: ISO/IEC 12207:2008 Paragraph/Template Number Cross-Reference	12
Table 2: IEC 62304 Paragraph/Template Number Cross-Reference.....	13
Table 3: ANSI/EIA-649-B Function and Principle/Template Number Cross-Reference	14
Table 4: Examples of Tool Strategy Granularity	26
Table 5: ISO/IEC 12207:2008 and ANSI/EIA-649-B Cross Reference.....	109

1. Introduction

In the past decade, the need for establishing effective software configuration management policies and procedures has grown in conjunction with the need for systems that are complex, fast, accurate, secure, and safety-critical. From simple everyday systems that track bank accounts to the complex systems in airplanes, cars, medical devices, and industrial, or military products, there is a demand for highly reliable, error-free software that either supports the development and maintenance of these systems or is an integral part of them.

This document provides a set of templates for implementing software configuration management processes. These templates are easy to use, self-explanatory, and do not require expensive training or extensive experience. They meet the requirements of ISO/IEC standards 12207:2008, *Software life cycle processes*, 15288:2008, *System life cycle processes*, IEC 62304, *Medical Device Software – Software Life Cycle Processes*, and ANSI/EIA-649-B, *Configuration Management Standard*.

Section 1.4 of this document provides a brief discussion of the ISO/IEC 12207:2008, ISO/IEC 15288:2008, IEC 62304, and ANSI/EIA-649-B standards. It also provides mappings between the various standards' requirements and the templates that address them. Note that in many cases, the name of a standard document is shortened to enhance readability. Unless otherwise noted, any reference containing "649B," "12207," "15288," or "62304" refers to ANSI/EIA-649-B, the 2008 version of ISO/IEC 12207, the 2008 version of ISO/IEC 15288, or the 2006 release of IEC 62304 respectively.

1.1 About Software Configuration Management

The inclusion of software configuration management as a top-level process in the software life cycle has proven to be a necessary and cost effective step in producing and delivering quality products on time and within specified time and dollar limitations. Effective configuration management of a system's software elements is accomplished by:

- Establishing a policy that requires the implementation of software configuration management.
- Defining a software configuration management strategy that drives subsequent plans and procedures.
- Building a comprehensive configuration management plan.
- Establishing a scheme for software item identification.
- Identifying the controlled items and item types that are designed, developed, and delivered.
- Agreeing to a baseline configuration.
- Enabling the development of and changes to that agreed upon baseline.

Software Configuration Management Templates - Introduction

- Recording the events of an item's evolution via status accounting records.
- Verifying via audits and reviews that:
 - Configuration management processes are measured, monitored and improved
 - The configuration of the product meets the appropriate specification documents

It is imperative for companies that build software to have a clear and concise policy on software configuration management as well as the procedures to implement it. A CM policy should be established by executive level management to provide top-level guidance and constraints to lower level plans and procedures. This guidance and constraint helps ensure consistency across lower level procedures including compliance with key corporate objectives.

Because software CM procedures may affect the utilization of resources across company organizations, these procedures should be reviewed and approved by an appropriate level of management in those organizations. These organizations can include Quality Management, Engineering, Testing, Project Management, and the customer (when applicable). The responsibility for defining and executing CM procedures must be clearly assigned down an organization as well as across it.

In large companies or organizations, different groups or projects may have their own unique CM requirements and resultant policies and procedures. In such cases, each organization must ensure that their unique software CM plan, strategy, and procedures are aligned with higher-level CM documents.

1.2 How to Use This Document

This document is designed to aid a person, with some knowledge of software configuration management, implement a sound software CM system. Expertise in software configuration management is not required. This document provides templates and aids for software CM that can be applied to manual or automated methods and can be implemented by one or more persons depending on the scope and complexity of the project or organization. These materials are applicable to projects of all sizes, and are applicable for all types of products during their life span.

Figure 1 shows the thirteen software CM templates and five CM aids contained in this document and how they relate to one another. Note, for example, how higher level documents (organization, strategy, policy, plan) provide the overall guidance and direction for the establishment of lower level procedures.

References to Standards

In many cases, the name of a standard document is shortened to enhance readability. Unless otherwise noted, any reference containing "649B," "12207," "15288," or "62304" refers to ANSI/EIA-649-B, the 2008 version of ISO/IEC 12207, the 2008 version of ISO/IEC 15288, or the 2006 release of IEC 62304 respectively.

Software Configuration Management Templates - Introduction

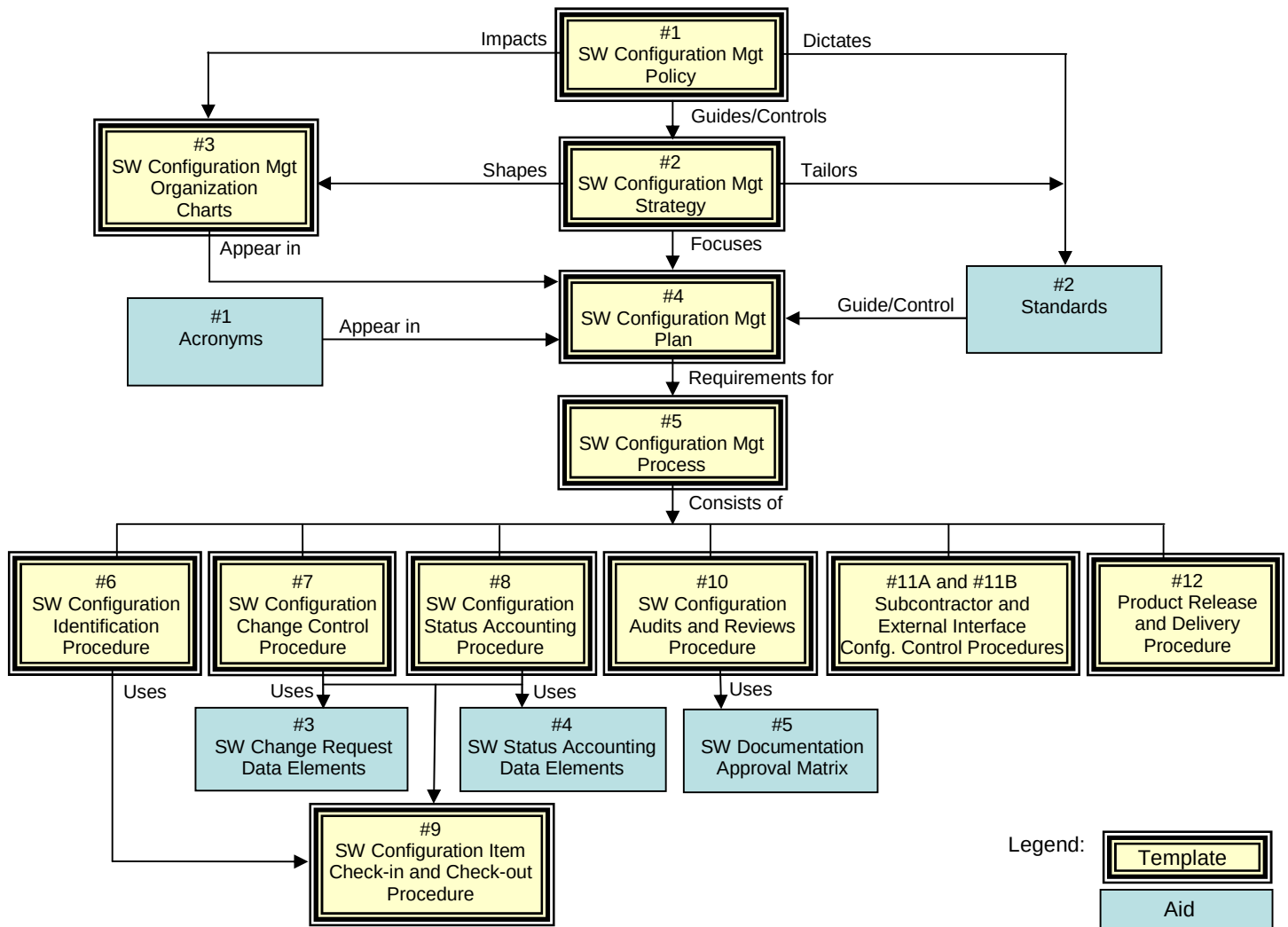


Figure 1: Software CM templates, and aids, and how they relate.

Software CM Templates

Each subsection of Section 2 (2.1, 2.3, etc) is devoted to one of the thirteen templates shown in Figure 1. Preceding each template, which begins with a title page, is a brief overview of the area being addressed by that template. Guidance notations in the templates are underlined and should be stripped from the implemented template. Template pages can be distinguished from other types of pages in this document by the occurrence of the word “Template” in their page header. Pages that are not part of a template do not have that word in their header.

The template serves as a framework for meeting the unique CM requirements of your organization. Thus, you will have to add text that provides the necessary detail to carry out the activities described in each template. Specific tools, data, and geographic distribution of work products and people will all affect the definition of an activity and will necessitate the tailoring of each template. The size and experience of your software staff will also affect the level of detail required. You may also have to modify text for specific domains and end-item types (embedded software, government software, classified software, etc).

Software Configuration Management Templates - Introduction

Although specific details to a template may be added to make its resulting procedure useful, do not be afraid to start with minimal text. The important thing is to *document your procedures*! Once they are documented, you can begin the more important task of *executing* them along with measuring, monitoring, and improving them throughout the system life cycle.

Software CM Aids

Section 3 of this document provides additional aids for software configuration management. These aids and their relationship to the templates are also shown in Figure 1. These aids provide help concerning acronyms, standards, CM data records, and document approval. Documentation found in Section 3 can be incorporated into your CM procedures or simply used as reference material.

1.3 The Software Domain

Note again that all of the templates contained in this document are for *software* policies, plans, procedures, etc. At times, to streamline the text, the word *software* does not precede the words “configuration management” or the abbreviation “CM.” Nevertheless, it is to be assumed.

1.4 Traceability to Configuration Management Standards

Overview

The templates in this document satisfy the software configuration management requirements of ISO/IEC standards 12207:2008, *Software life cycle processes* and 15288:2008, *System life cycle processes* as well as IEC standard 62304, *Medical Device Software – Software Life Cycle Processes*. The templates provided are also consistent with the principles of ANSI/EIA-649-B, Software Configuration Management, thus enabling implementation of the processes compliant with that standard.

The 2008 versions of the ISO/IEC 12207 and ISO/IEC 15288 were released in tandem to better align them. As noted in both standards, they were released to:

- “Provide a common terminology between the revision of ISO/IEC 15288 and ISO/IEC 12207,
- Where applicable, provide common process names and process structure between ISO/IEC 12207 and ISO/IEC 15288,
- Enable the user community to evolve towards fully harmonized standards ... while maximizing backward compatibility.”

IEC 62304 was released in 2006 and has not been revised since then.

ANSI/EIA-649 was developed in response to the DoD’s decision in 1994 to cease using MIL-STD-973 as a configuration management standard and to adopt best commercial practices. It was developed by the Electronics Industry Alliance (EIA) to provide CM guidance to DoD contractors without prescribing a strict set of CM procedures and deliverables. The original version of the ANSI/EIA-649 standard was released in 1999, updated in 2004 (ANSI/EIA-649-A), and then again in 2011 (ANSI/EIA-649-B). During this time frame (in 2009), the DoD officially declared that “The EIA-649: ‘National Consensus Standard for

Software Configuration Management Templates - Introduction

Configuration Management' as adopted by DoD can be used as a replacement since it is almost a complete duplication of MIL-STD-973.”

The 649B standard was approved by ANSI, hence its formal designation as an “ANSI/EIA” standard. EIA ceased operations in February 2011; the 649B standard is now owned and maintained by TechAmerica.

ISO/IEC 12207:2008 and ISO/IEC 15288:2008 Traceability

As part of the tandem release of these two standards, an identical set of CM *system* process requirements now appears in both the 15288 and 12207. 12207 has an additional set of CM *software* process requirements that are a specialization of the system requirements. Figure 2 shows an overview of how CM process requirements fit into each standard and how they relate to one another. Suffice it to say - if your software CM processes are compliant to 12207:2008, they will be compliant to 15288:2008.

As noted above, meeting the 12207 software CM standard will also ensure meeting the 15288 CM standard as it applies to software. Accordingly, to show how our templates address the software CM requirements of both 12207 and 15288, we have mapped each one of our templates to the 12207 clause that they address. This mapping is shown in Table 1 below following an itemization (in Figure 3) of 12207 requirements clauses. Because the standard is written at a high level and the procedures are inter-related, each requirement is addressed by multiple templates. Depending on how you adapt your templates, you may need a different set of cross-referenced templates to meet a standard.

Note that neither ISO/IEC 12207 nor ISO/IEC 15288 specifically requires the development of a CM policy, but they refer to the *use* of a policy to implement CM procedures. This is consistent with the hierarchy of policy, strategy, plan, and procedures as shown in Figure 4 and Figure 5 in Sections 2.1 and 2.2 respectively. Accordingly, the CM policy template is shown as being required to meet selected standards.

Software Configuration Management Templates - Introduction

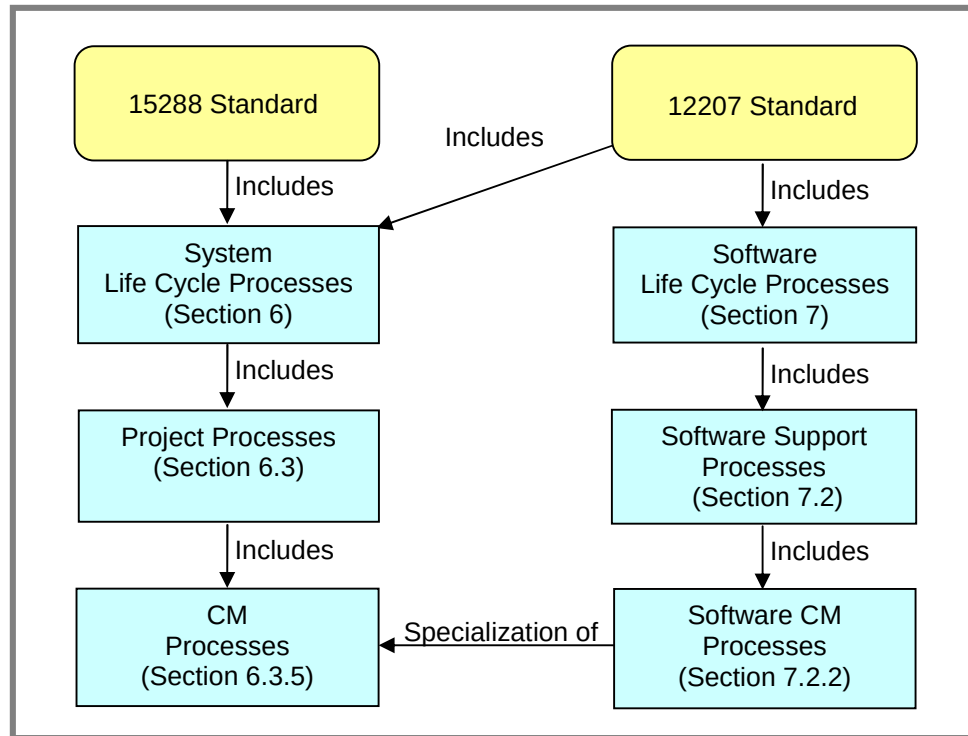


Figure 2: Standard's document sections and relationships.

As shown in Figure 2 above, the system requirements that are in the 15288 standard are also included in the 12207 standard. The 12207 standard also includes an additional set of requirements clauses that meet the *system* CM processes as well as provide some requirements specific to *software*.

- 7.2.2 Software Configuration Management Process
 - 7.2.2.1 Purpose
 - 7.2.2.2 Outcomes
 - 7.2.2.2a CM strategy
 - 7.2.2.2b Identified, defined, baselined work products
 - 7.2.2.2c Controlled work product changes and releases
 - 7.2.2.2d Modifications and releases available to affected parties
 - 7.2.2.2e Work product status recorded and reported
 - 7.2.2.2f Work product completeness and consistency ensured
 - 7.2.2.2g Work product storage, handling, delivery controlled
 - 7.2.2.3 Activities and tasks
 - 7.2.2.3.1 Process implementation
 - 7.2.2.3.2 Configuration identification
 - 7.2.2.3.3 Configuration control
 - 7.2.2.3.4 Configuration status accounting
 - 7.2.2.3.5 Configuration evaluation
 - 7.2.2.3.6 Release management and delivery

Figure 3: ISO/IEC 12207:2008 Software CM Requirements Clauses.

Software Configuration Management Templates - Introduction

Table 1: ISO/IEC 12207:2008 Paragraph/Template Number Cross-Reference

12207 Requirements Paragraph	Template Number												
	1	2	3	4	5	6	7	8	9	10	11A	11B	12
7.2.2.1	X	X		X							X	X	
7.2.2.2a	X	X	X	X							X	X	
7.2.2.2b				X	X	X			X		X	X	
7.2.2.2c					X				X		X	X	
7.2.2.2d					X		X		X		X	X	
7.2.2.2e					X			X			X	X	
7.2.2.2f					X			X	X	X	X	X	
7.2.2.2g					X				X		X		X
7.2.2.3	X												
7.2.2.3.1	X	X	X	X							X	X	
7.2.2.3.2					X	X			X		X	X	
7.2.2.3.3					X		X		X		X	X	
7.2.2.3.4					X			X	X		X	X	
7.2.2.3.5					X					X	X	X	
7.2.2.3.6					X	X	X	X	X	X	X	X	X

Legend: X = Template required to satisfy this standard requirement.

Software Configuration Management Templates - Introduction

IEC 62304 Traceability

The CM requirements found in the IEC 62304 standard are compatible to ISO/IEC 12207; thus, compliance with 12207 should ensure compliance with 62304. However, unlike 12207, the 62304 standard specially references Software of Unknown Provenance (SOUP). As noted in the 62304 standard, SOUP is also called off-the-shelf software (OTS). Accordingly, to help ensure compliance with both 62304 and 12207, SEPT has added specific references to both OTS and SOUP in Templates 6 (Configuration Identification) and Template 12 (Software Release).

Although you may choose to use all thirteen templates in this document, as shown in the Table 2 below, only four templates are needed (6, 7, 8, and 12) to address 62304 software CM requirements specifically. Note also that besides addressing 62304 CM requirements, templates 7 and 12 in this document address 62304 process requirements for problem resolution and software release respectively.

Table 2: IEC 62304 Paragraph/Template Number Cross-Reference

IEC 62304 Requirements Paragraph	Template Number												
	1	2	3	4	5	6	7	8	9	10	11A	11B	12
5.8													X
8.1						X							X
8.2							X						
8.3								X					X
9.0							X						X

Legend: X = Template required to satisfy this standard requirement.

Software Configuration Management Templates - Introduction

ANSI/EIA-649-B Traceability

Our templates provide a baseline that allows you to easily create CM processes and plans adapted to your unique project requirements while still meeting the requirements specified in the ANSI/EIA-649-B standard. Table 3 below show where 649B functions and principles are addressed by our templates. Because Table 3 shows ID numbers only for 649B functions and principles, you need to have the actual 649B standard in hand to view its function names and principle descriptions. Refer to Appendix A for more information about the 649B standard and how it relates to the 12207 standard and to our CM templates.

Table 3: ANSI/EIA-649-B Function and Principle/Template Number Cross-Reference

649B Function	649B Principle	CM Template(s)
5	CM-1	1,2
5.1	CMP-1,CMP-3	4*
5.1	CMP-2	3,4*, 11A,11B
5.1	CMP-4 thru CMP-6	4*,5
5.1	CMP-7	4*, 5, 11A, 11B
5.1	CMP-8	4*,5
5.2	CI-1 thru CI-12	5, 6*, 9, 11A, 11B
5.2	CI-13	11A, 11B
5.3	CCM-1 thru CCM-9	5, 7*, 9, 11A, 11B
5.4	CSA-1 thru CSA-3	5, 8*, 9, 11A, 11B
5.5	CVA-1 thru CVA-3	5, 10*, 11A, 11B

* This template is the main template addressing the function and principle

1.5 Product Support

Within the first 60 days following your purchase of this document, SEPT will provide 3 hours of free consulting to help you better understand and apply these templates. This consultation is available at the SEPT Help Desk. Contact us at 425-391-2344 between the hours of 8 AM to 5 PM PST, Monday through Friday, or email your questions to Stanmagee@smartwire.net.

1.6 Warranties and Liability

SEPT makes no warranties, implied or stated with respect to this template, and it is provided on an “*as is basis*”. SEPT will have no liability for any indirect, incidental, special or consequential damages or any loss of revenue or profits arising under, or with respect to the use of this document.

2. Software Configuration Management Templates

Listed below are the templates contained in this chapter. Each one of these templates is contained in a subsection of this chapter.

- **Section 2.1 Template 1: Software Configuration Management Policy**

CM policy provides the top-level instructions and directives that must be followed in the development and subsequent implementation and execution of configuration management strategy, plans, and processes. It reflects an organization or project's commitment to configuration management.

- **Section 2.2 Template 2: Software Configuration Management Strategy**

Strategy describes specific approaches to be followed, adaptations to be made, and objectives to be achieved in the implementation and execution of CM plans and procedures. Your CM strategy must fit within the dictates and constraints of your CM policy while defining, in more detail, how effective CM is to be achieved.

- **Section 2.3 Template 3: Software Configuration Management Organization Charts**

This section provides example charts to illustrate where configuration management organizations or functions can fit into companies of various sizes in order to ensure the effective oversight and implementation of CM.

- **Section 2.4 Template 4: Software Configuration Management Plan**

Guided and constrained by the higher level CM Policy and CM strategy, this plan describes what software items shall be placed under configuration management and it provides constraints and criteria for software CM procedures. This plan tells project and customer management *what* will be done to ensure the integrity of the software. Lower level procedures describe the *how*.

- **Section 2.5 Template 5: Software Configuration Management Process**

This process provides a top level "parent document" for configuration management. Included in this parent document are the functions: Software Configuration Identification, Software Configuration Change Control, Software Configuration Status Accounting, Software Configuration Audits and Reviews, Subcontractor Software Configuration Control, External Interface Configuration Control, and Software Product Release and Delivery

Section 2.6 Template 6: Software Configuration Identification Procedure

This procedure defines activities necessary to ensure that each configuration item and its related technical data can be uniquely identified and tracked through each release cycle.

- **Section 2.7 Template 7: Software Configuration Change Control Procedure**

This procedure describes the activities required to ensure that software changes are properly recorded, analyzed, reviewed, approved (or rejected), and implemented. This procedure assures that changes are properly reviewed and approved (or rejected) by a Change Board and that change records and change impacts are documented as part of each release.

Software Configuration Management Templates

- **Section 2.8 Template 8: Software Configuration Status Accounting Procedure**

This procedure defines activities that track the state of software configuration items as they move through the software life cycle.

- **Section 2.9 Template 9: Software Configuration Item Check-in and Check-out Procedure**

This procedure describes the activities, controls, and reports used in order for developers to remove configuration items (check-out) from the CM repository and then later return them (check-in). This procedure provides versioning of configuration items as well as verification of change authorization.

- **Section 2.10 Template 10: Software Configuration Audits and Reviews Procedure**

This procedure defines activities that help ensure project processes are being followed properly and that configured end items are being developed per their specifications.

- **Section 2.11 Template 11A: Subcontractor Software Configuration Control Procedure**

This procedure defines activities to ensure that subcontractors are following agreed-upon configuration management processes.

- **Section 2.11 Template 11B: External Interface Configuration Control Procedure**

This procedure defines activities to document, monitor, and control specifications of external interfaces.

- **Section 2.12 Template 12: Software Product Release and Delivery Procedure**

This procedure describes the activities that result in the creation of the final software product and documentation that is delivered to the customer or distributor.

Software Configuration Management Policy

2.1 Software Configuration Management Policy

An organization's CM policy reflects the top level guiding principles it wants to be followed while achieving desired results from software development and maintenance. CM policy provides key directives that provide boundaries and constraints within which lower level strategies, plans, and procedures must conform. It is a cornerstone of an organization's software strategy and plans. An organization desirous of implementing a configuration management process must first determine the organization's policy for that process.

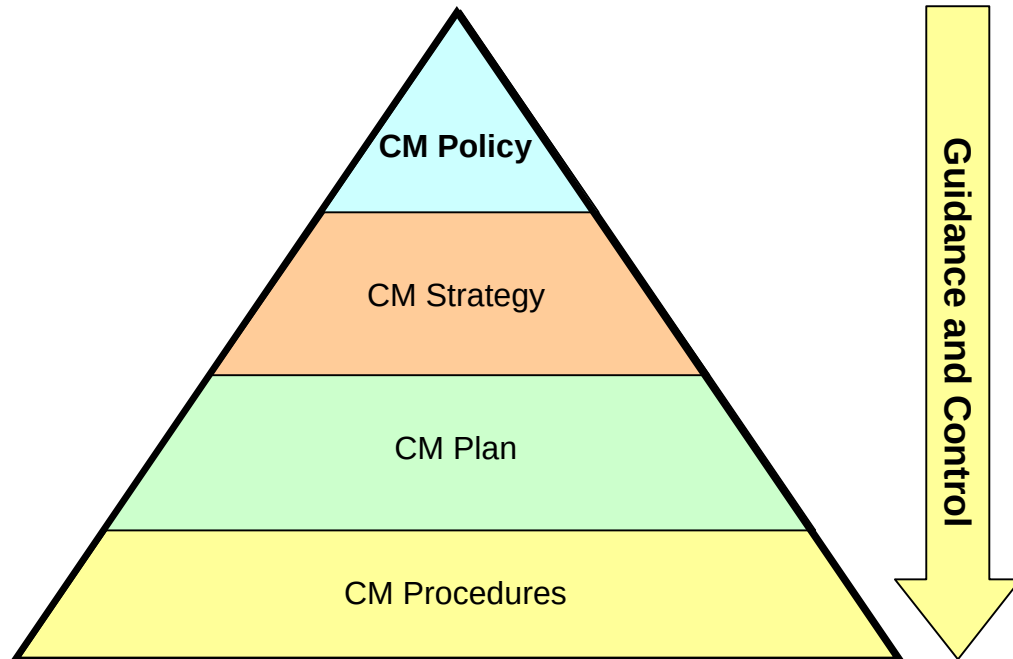


Figure 4: *CM Policy provides high-level guidance and control.*

CM Policy is applied at all levels of the organization. For larger organizations this can be far reaching – from the boardroom to the shop floor. For small companies, this can mean from the owner/operator to his or her small group of employees. A configuration management policy provides the authority for lower level operating strategies, plans and procedures. It can be specific in some cases, but generally leaves most of the details for lower level operating documents. For example, your policy can require following a specific standard, but tailoring of that standard can be part of your configuration management strategy. Similarly, the CM policy defines basic responsibilities. For example, configuration managers are responsible for defining, implementing, and monitoring configuration management processes. Specific as to the “how to” are left to subsequent procedures and instructions.

Absence of a CM policy (and the authority it provides) can hinder the effective operation of your organization's other key functions such as finance, engineering, manufacturing, marketing, and quality control.

The text in this policy template is written as if the policy is for the entire company. In many large companies and organizations, lower-level organizations and/or projects may require their own unique policy that not only conforms to their corporate policy, but also incorporates

Software Configuration Management Policy

elements dictated by specific organization, product or contractual requirements. This template can be modified and adopted for such lower-level policy statements.

Figure 11 in Section 2.5 illustrates a hierarchy of policies. The CM policy template follows.

Template 1: Software Configuration Management Policy

Title Page for Software Configuration Management Policy

Software Configuration Management Policy

As appropriate, add:

- *Version/Revision Number*
- *Company/Organization Name*
- *Program/Project Name or Product Name*
- *Contract Reference*
- *Prepared by/Approved by and Dates*

Example:

Version 1.0

The XYZ Company

Project: Ace Software Project

Contract 001

Prepared by	Signature	Signature Date
Joe Smith		Jan 5, 2012
Approved by		
Sue Jones		Feb 1, 2012

Template 1: Software Configuration Management Policy

Policy Statement

The [XYZ] Company shall provide for the implementation of a configuration management process on all projects designed, developed, built or procured, tested and delivered, or sold, whether required by contract or purchase agreement. This process will meet or exceed all applicable industry-wide and organizational standards concerning configuration management.

Objective

The objective is to know at all times what the designed, developed, built or procured, tested and delivered configuration is on all products produced by the company. There will be traceability from one stage of the product to the next. The customer will have confidence that (XYZ) company has a system for controlling all stages of a product throughout the product life cycle.

Responsibility

<u>Function</u>	<u>Responsibility</u>
Organization's Chief Executive	• Ensure the implementation of a configuration management process through this policy and subsequent configuration management procedures.
Executive Management (Depends on company size/organization)	• Ensure that the configuration management process is performed on all projects for which they are responsible.
Project Managers	• Plan and provide for implementation of the configuration management process on their assigned project. • Authorize process tailoring as appropriate to meet unique project requirements while still meeting the objectives of this policy
Configuration Managers	• Develop appropriate configuration management procedures and instruction material in accordance with project requirements as well as applicable industry-wide and organizational standards. • Implement CM procedures • Provide training and education to users of CM procedures. • Maintain, monitor, measure and improve implemented processes
Software Development and Maintenance	• Understand, utilize, and adhere to company, organization, and project CM requirements.

Software Configuration Management Strategy

2.2 Software Configuration Management Strategy

CM strategy can vary widely from company to company and even from project to project. As indicated on Figure 1, CM strategy is guided and controlled by CM policy. Once a strategy is defined, it further focuses operating plans and procedures for CM.

Figure 5 is a conceptual chart showing how strategy sits between CM policy and plans helping to sharpen CM focus.

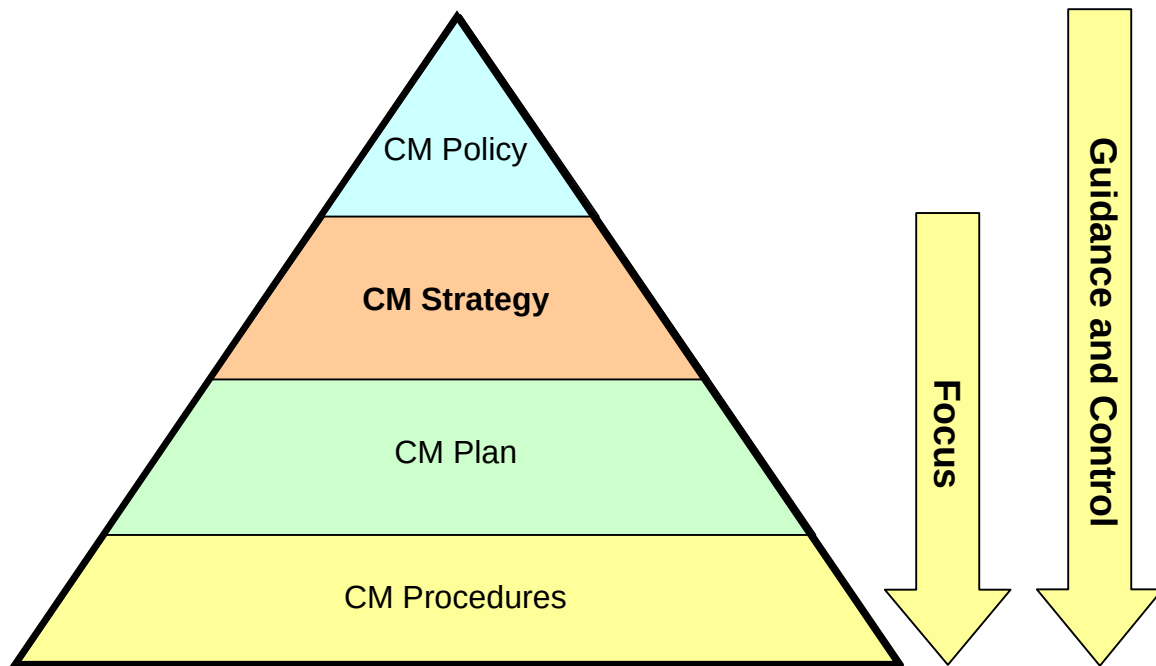


Figure 5: *CM Strategy is guided by policy and focuses plans and procedures.*

Within the boundaries of the CM policy’s high-level constraints, various strategies can be employed to implement CM. For example, your CM policy may dictate the use of ISO/IEC standards 12207 and 15288. Both of those standards allow tailoring, thus one project may choose to adapt implementing procedures one way and another project may adapt their procedures differently.

Some organizations may find it difficult to draw the distinction between their strategy and their plan. It is acceptable to have only a CM plan, as long as that plan addresses those issues documented in clause 5.4.7.3a of the 15288 standard. Because the CM plan and subsequent CM procedures provide the definitive approach used by an organization or project, the CM strategy should be brief – providing only that level of detail needed to focus the plan and subsequent procedures.

Note that the template that follows is a template for the strategy document. It is not a procedure template.

Template 2: Software Configuration Management Strategy

Title Page for Software Configuration Management Strategy

Software Configuration Management Strategy

As appropriate, add:

- *Version/Revision number*
- *Company/Organization Name*
- *Program/Project Name or Product Name*
- *Contract Reference*
- *Prepared by/Approved by and Dates*

Example:

Version 1.0

The XYZ Company

Project: Ace Software Project

Contract 001

Prepared by	Signature	Signature Date
Joe Smith		Jan 5, 2012
Approved by		
Sue Jones		Feb 1, 2012

Template 2: Software Configuration Management Strategy

1.0 Executive Summary

This executive summary will briefly outline your strategy and the key considerations driving that strategy. The purpose of this document is to define the configuration management strategy that will be employed on the [Ace Software Project]. The configuration management strategy for the [Ace Software Project] falls within the guidelines established by company policy and is implemented in the configuration management plan and lower level configuration management procedures defined for this project. Add more detail as appropriate.

2.0 Responsibility

<u>Function</u>	<u>Responsibility</u>
Configuration Management	• Define and implement a configuration management strategy.
Project Manager	• Review and approve the configuration management strategy.
Software Development and Maintenance <u>“Software Development and Maintenance” is used here to refer to anybody authorized to design, build, or change a software configuration item.</u>	• Develop and maintain software utilizing CM procedures as required. • Provide feedback to the configuration management function to help modify strategy based on experience.

3.0 Inputs and Controls

- The XYZ Company’ Software CM Policy.
- Other company development policies and standards. You may choose to enter those standards here.
- The Ace Software Project contract.
- List a specific statement of work or set of requirements as appropriate

Template 2: Software Configuration Management Strategy

4.0 Strategies

The subsections below contain guidelines for developing a strategy for key factors affecting CM processes. You may choose to define strategies for all or some of these factors and as well as other factors unique to your company or project. Using these guidelines, you will have to define the specific strategy you intend to employ. Note that as you move from policy, to strategy, to plan, you become more definitive. Thus, your CM strategy document need only contain enough information to focus the CM plan and subsequent procedures, which contain your definitive approach to CM.

Because each one of these sections will be unique to your organization and project, for readability purposes, we have chosen not to underline the text below.

Background and Context

This section should describe the nature of your project and the constraints that might not be otherwise obvious. The information contained in this section would provide information that might help explain why a particular strategy is being employed. This information may also appear in the CM plan.

Staffing

Typically, you will have staffing constraints already defined prior to developing a strategy. You may have been given a staffing level or cost and you need to fit your CM approach within that level or you may have been simply told to “minimize CM labor costs.” If you have not received this directive, you should assume you have! Remember that CM requirements will impact software developers. Highly paid developers should not be burdened with time-consuming clerical tasks.

Given the fact that you want to minimize the clerical CM burden on software developers, you need to tradeoff hiring CM clerical staff against hiring technical staff to build CM tools for the developers. Typically, you will have both, so your strategy should be to get the right blend of technical and clerical staff to perform the CM function. Too often, CM is looked upon as solely a non-technical, clerical task. This is a mistake!

On large projects, you may want to have several programmers and/or software engineers working in the configuration management organization. This can greatly improve the communication between the development organization and the configuration management organization. Because these individuals understand the development tasks, they can readily develop small unobtrusive CM tools that enhance overall project CM while not slowing down the developers.

You may want to consider rotating software developers through the CM function for two reasons:

- Working in CM will give them a better appreciation of what is required.
- If a developer works too long in the CM area, their software development skills may languish as well as their paycheck!

Template 2: Software Configuration Management Strategy

As alluded to in the above bullet, if you do have software developers on your CM staff, you need to ensure that they are treated equally to their software development brethren in salary issues.

Documentation

Define the formal CM documentation that you will create as well as its media and controls. Some documents may be database-driven while others may be classic text-oriented documents. Access to documents should be considered within your strategy. Should they be on the web? How widely distributed should they be? What kinds of access controls are required? How will you notify people when a document changes?

Some of the answers to these questions may be dictated by the contract or other company policies or standards. CM information should be:

- Transparent
- Readily available to the widest possible audience
- Easy to use

How you plan to achieve the above three objectives is part of your strategy.

Tools

Based on the discussion in the above two sections, it is clear that tools will be part of your approach to CM. Specific tools may be defined in those sections, left to this section, or even in some cases delayed until the definition of a specific process. However, enough detail must be present in your strategy to begin to quantify cost and technical feasibility.

Items to consider and tradeoffs concerning CM tools include:

- Are they a supported commercial product or are they freeware?
- How large is their user base?
- How easy are they to tailor?
- Are they being used elsewhere in the company?
- What are the contractual requirements?
- What does the contractor/subcontractor use?
- How scalable is the tool?
- Does it interface with the internet?
- How does it handle multiple users?

Shown below is a table with several examples of a specific tool strategy. Note that although the granularity of the policy and plan are the same, the granularity of the strategy is different in each example. All of these examples are acceptable – again depending on the requirements levied on the strategy definition.

Template 2: Software Configuration Management Strategy

Table 4: Examples of Tool Strategy Granularity

Example	Policy	Strategy	Plan
#1	There will be traceability from one stage of the product to the next.	This project will maintain multiple versions of the product as it moves through the software development life cycle using a proven open source CM product.	Using the [XXX] open source product the CM group will maintain a separate version of the software product for development, test, and release.
#2	Same as above.	This project will maintain multiple versions of the product as it moves through the software development life cycle using open source tools used successfully elsewhere in the XYZ company.	Same as above.
#3	Same as above.	This project will maintain multiple versions of the product as it moves through the software development life cycle using the open source tool, [XXX].	Same as above.

CM Standards and Procedures Usage and Tailoring

The best approach to standards and procedures usage is to do what you have always done, *but make it better*. There is no point in reinventing the wheel when it comes to CM. However, it is also foolish to ignore lessons learned on previous projects.

Sometimes you do not have the luxury of simply reusing already existing procedures or sometimes, you may decide that existing procedures can be greatly simplified because the project itself is limited in scope (size, complexity, where used, etc). At other times, you may face a contractual requirement that necessitates a change to your existing procedures. In any case, often the best approach is to tailor already existing procedures. As part of your strategy, to ensure that you have met contractual and company CM requirements, you may elect to develop a matrix that shows how your CM procedures meet required standards or other directives.

Although you may not have a requirement to meet specific ANSI, ISO, and/or IEC CM standards, we recommend that as part of your strategy you review these standards and ensure that your procedures map back to them or that you have good reasons to depart from them. These widely-used standards leave a great deal of leeway to their implementation. Consequently, as part of your strategy, *what* standard is not as important as *how* you implement them. For small, geographically centralized projects, your procedures can be less formal and more streamlined than for projects that are large and geographically dispersed.

Template 2: Software Configuration Management Strategy

Audits and Audit Reports

As part of your strategy, you need to determine the most effective way to perform and support audits and their necessary follow-up. Your detailed CM audit procedures will be shaped by the strategies you articulate. Shown below are some questions that your audits should address. Note that your specific project may have different questions.

- Can the traceability of a configuration item to a requirement be verified?
- Should audits be done via formal reviews, informal reviews, or both?
- Can auditing be done via automated reporting
- How often should reviews/reports be conducted/generated?
- Who gets the report?
- Who follows up?
- What measurements need to be in place?
- Are the audits effective?
- Are changes managed properly?
 - Is change impact documented?
 - Are changes noted on each release?
 - Are all changes properly authorized?
- Upon completion, has the right product been built?
- Are all related documents and other support products properly packaged with the completed ready-to-release software product?

Remember that it is in your procedures where the details of these questions are answered. Thus, you can keep your strategy statement simple. Here is an example: “CM audits and their follow-up will be designed to be effective, yet as unobtrusive as possible. The CM organization will put activities in place to ensure that audits are performed as required and that action items spawned by audits are being accomplished in a timely manner. The CM organization will record and track audit information on a web-accessible database, discuss audit reports as part of normal project management meetings, and meet quarterly with the customer to review CM issues.”

Template 2: Software Configuration Management Strategy

Items Placed Under Configuration Management

Your configuration management plan will specify what work product will be controlled via CM procedures. In your strategy document, you need to state a simple policy that will help focus the plan. For example, “All work products that will become a part of the deliverable product will be managed by our CM procedures. All other supporting products will be subject to CM procedures on a case-by-case basis as determined by the Software Project Manager.”

Access Control, Security, and Backup

In most contracts, access to your software is of major concern and must be addressed by your strategy and subsequently in your CM procedures. If possible, your strategy should be to use or adapt processes and procedures already approved and in place.

Questions that your strategy needs to address include:

- Are some work products classified?
- Is multi-user update required?
- Is remote access required?
- How often will backups be performed?
- Where will backup copies be stored?
- Are different access authorities needed for different types of work products?
- Are physical security measures needed?

Training/Communication

Successful implementation and execution of CM procedures requires that all impacted organizations and personnel understand them. Sometimes organizations and personnel play a supporting role, and at other times (such as check-in/check-out), they actually perform CM-required activities. Consequently, an important part of the CM strategy is how to educate impacted organizations about using and support CM processes. Your specific strategy will depend on the experience of impacted organizations, expected turnover, and their need for ongoing support. In addition to experience of impacted personnel, the size and scope of the project can affect your approach to training project personnel.

Once an organization begins the software development process, ongoing feedback is needed to ensure that procedures are being followed properly and that needed CM data is readily available. Again, the size and scope of your project will generally drive your approach to filling this need.

Software Configuration Management Organization Charts

2.3 Software Configuration Management Organization Charts

The organization charts found in this section are centered on how software configuration management fits within an organization. One of the key elements in establishing a robust configuration management function is to ensure that it is positioned correctly within the overall organization. If it is not positioned correctly, it will cause confusion and political infighting when serious configuration management problems occur. For example, if configuration management reports to the person responsible for software design and testing, it may be difficult for the voice of configuration management to be heard when there is a conflict between CM requirements and other organizational objectives. The first two organizational charts shown in this section depict the software configuration management function and how it fits into the overall organization. The first chart (Figure 6) shows a typical organization structure for a large firm. The second chart (Figure 7) is more typical for smaller firms comprised of less than 25 people. This second chart can also apply to programmer teams that are normally part of larger groups. The ready availability of charts like this enables customers or potential customers to see how their configuration management requirements are or will be managed, respectively.

In the example chart for a large company (Figure 6), the CM manager reports on a dotted line basis to the Quality Management manager to help ensure that CM issues are addressed from a quality standpoint as well as a technical, cost, and schedule standpoint. In a small firm, as shown on Figure 7, it may be advisable to have the person responsible for configuration management reporting to the chief programmer for everyday tasks, but reporting to the chief executive officer for policy issues concerning configuration management. This same person would also be responsible for reporting to the chief executive office any serious matters that affect the configuration of a product to be delivered to a customer.

Note that we have put program and project management together. For very large programs, you may have multiple projects and consequently project managers. With the exception of the chart on Figure 6, we have opted to use “project manager” throughout the document. You may have to change this nomenclature depending on your specific reporting hierarchy.

Organization charts are presented below as a template. Note that unlike other templates, no cover page is included. That is because this template (with your specific adaptations) should be included in the configuration management plan. You may also choose to include it in your configuration management strategy document.

Template 3: Software Configuration Management Organization Charts

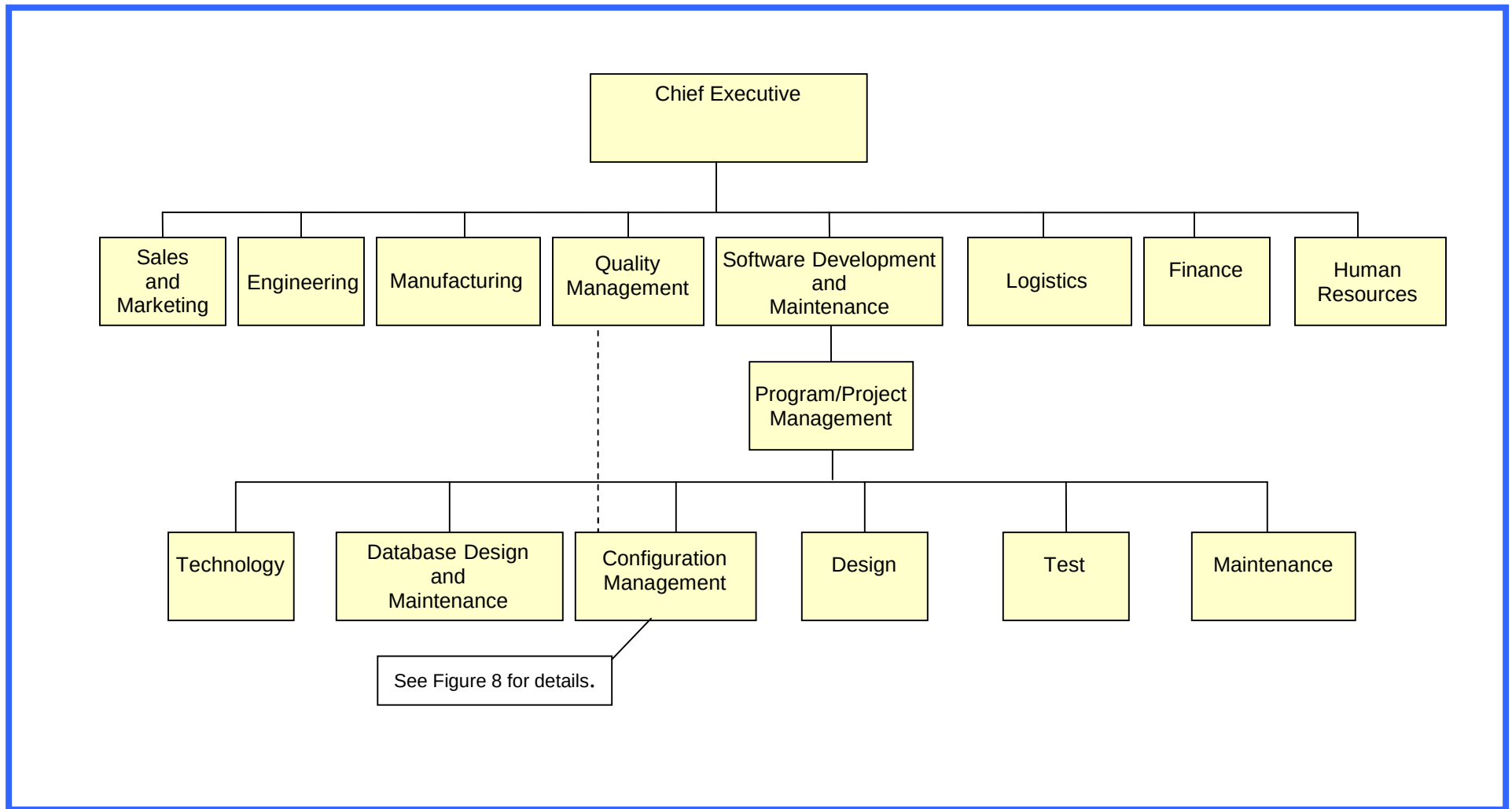


Figure 6: *Software CM organization within a typical large company.*

Template 3: Software Configuration Management Organization Charts

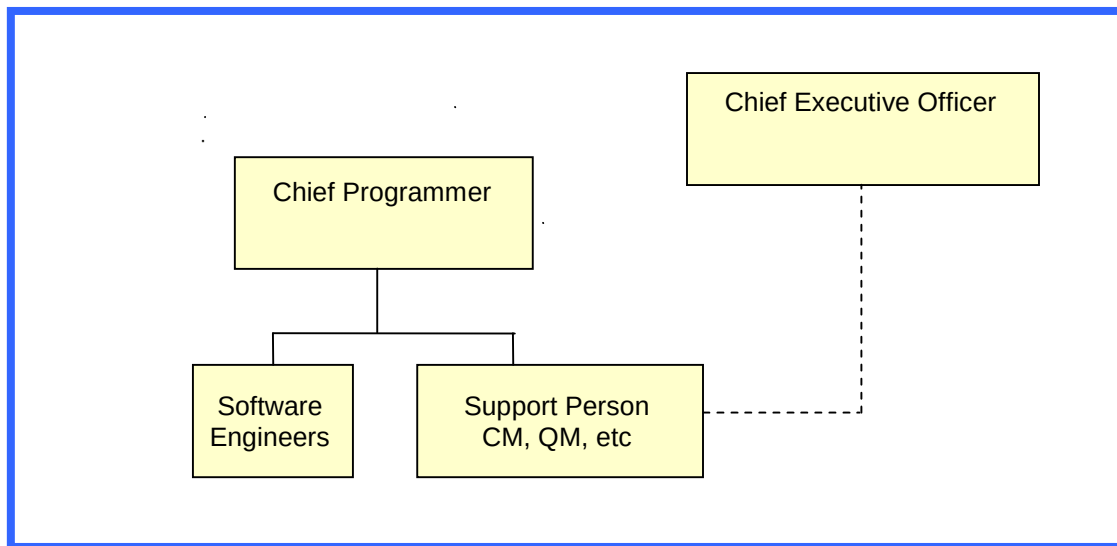


Figure 7: *CM organization within a small company.*

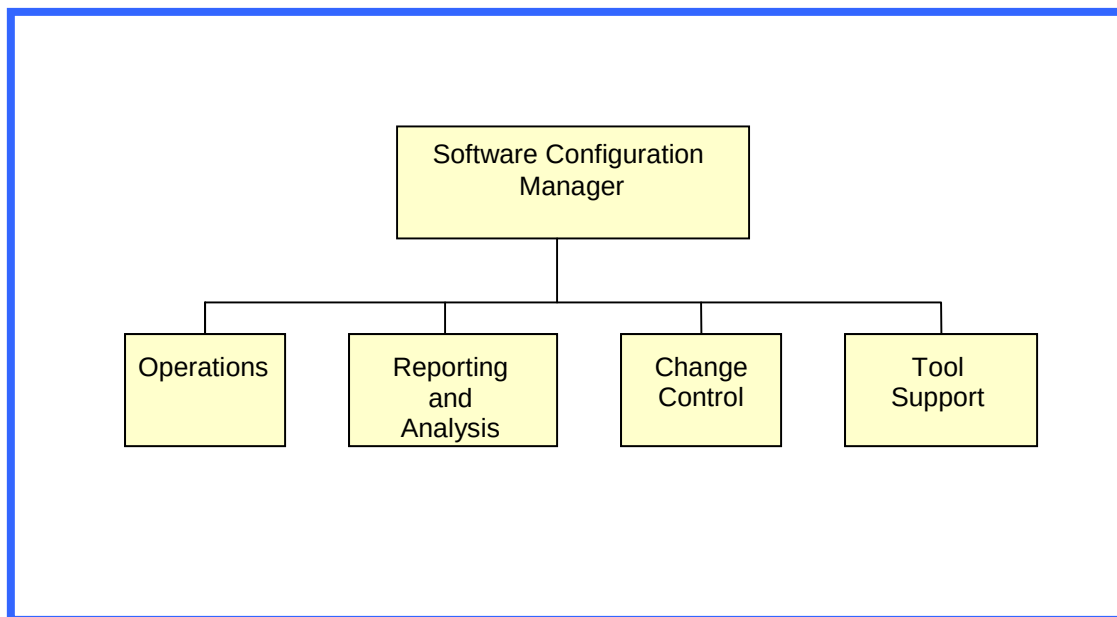


Figure 8: *Organization of a typical software CM group.*

The third organization chart (Figure 8 above) shows how a configuration management organization for large projects or organizations can be broken down. This type of organization might be staffed with 5 to 10 people. Although Figure 8 depicts a hierarchical structure, people assigned to one area may have to perform work in a second area depending on workloads and priorities. Likewise, because all of the CM procedures are interrelated, close communication and cooperation among the groups is essential. Each group shown in Figure 8 is listed below together with their primary responsibilities.

Template 3: Software Configuration Management Organization Charts

- **Operations**

This function encompasses the normal day-to-day operations that support users of the CM process. This would include support to the assignment of unique identifiers as well as support to daily check-in and check-out operations. This functional area would also include product release, subcontractor CM control and external interface control. Accordingly, this group focuses primarily on Templates 6, 10, 11, and 12.

Reporting and Analysis

This functional area specializes in communicating CM information within the CM group and, more importantly, to other functional areas. This group produces audit reports, status reports, and CM review reports. The group also follows up anomalies and deficiencies noted in any of the reports. This group focuses primarily on activities found in Templates 8 and 9.

- **Change Control**

This group is responsible for developing and operating the change management process. They provide support to the Change Board and provide informational reports regarding change status. This group focuses primarily on the activities found in Template 7.

- **Tool Support.**

This group develops or procures and then supports the tools necessary to support the CM process. This group would provide support to all the other CM groups.

You may choose to organize your CM group differently from what is shown in Figure 8; but, whatever your organization structure, the CM processes to be supported are the same.