



**<Project Name>  
Configuration Management Plan**

**Version <1.0>**

- *[Note: The following template is provided for use with the Rational Unified Process. Text enclosed in square brackets and displayed in blue italics (style=InfoBlue) is included to provide guidance to the author and should be deleted before publishing the document. A paragraph entered following this style will automatically be set to normal (style=Body Text).]*

<Project Name>	Version: <1.0>
Configuration Management Plan	Date: <dd/mmm/yy>
<document identifier>	

### Revision History

Date	Version	Description	Author
<dd/mmm/yy>	<x.x>	<details>	<name>

CTOSystem.com

<Project Name>	Version: <1.0>
Configuration Management Plan	Date: <dd/mmm/yy>
<document identifier>	

## Table of Contents

1.	Introduction	4
1.1	Purpose	4
1.2	Scope	4
1.3	Definitions, Acronyms, and Abbreviations	4
1.4	References	4
1.5	Overview	4
2.	Software Configuration Management	4
2.1	Organization, Responsibilities, and Interfaces	4
2.2	Tools, Environment, and Infrastructure	4
3.	The Configuration Management Program	5
3.1	Configuration Identification	5
3.1.1	Identification Methods	5
3.1.2	Project Baselines	5
3.2	Configuration and Change Control	5
3.2.1	Change Request Processing and Approval	5
3.2.2	Change Control Board (CCB)	5
3.3	Configuration Status Accounting	5
3.3.1	Project Media Storage and Release Process	5
3.3.2	Reports and Audits	5
4.	Milestones	6
5.	Training and Resources	6
6.	Subcontractor and Vendor Software Control	6

<Project Name>	Version: <1.0>
Configuration Management Plan	Date: <dd/mmm/yy>
<document identifier>	

# Configuration Management Plan

## 1. Introduction

- *[The introduction of the **Configuration Management Plan** should provide an overview of the entire document. It should include the purpose, scope, definitions, acronyms, abbreviations, references, and overview of this **Configuration Management Plan**.]*

### 1.1 Purpose

- *[Specify the purpose of this **Configuration Management Plan**.]*

### 1.2 Scope

- *[A brief description of the scope of this **Configuration Management Plan**; what model it is associated with and anything else that is affected or influenced by this document.]*

### 1.3 Definitions, Acronyms, and Abbreviations

- *[This subsection should provide the definitions of all terms, acronyms, and abbreviations required to properly interpret the **Configuration Management Plan**. This information may be provided by reference to the project Glossary.]*

### 1.4 References

- *[This subsection should provide a complete list of all documents referenced elsewhere in the **Configuration Management Plan**. Each document should be identified by title, report number (if applicable), date, and publishing organization. Specify the sources from which the references can be obtained. This information may be provided by reference to an appendix or to another document.]*

### 1.5 Overview

- *[This subsection should describe what the rest of the **Configuration Management Plan** contains and explain how the document is organized.]*

## 2. Software Configuration Management

### 2.1 Organization, Responsibilities, and Interfaces

- *[Describe who is going to be responsible for performing the various Configuration Management (CM) activities described in the CM Process Workflow.]*

### 2.2 Tools, Environment, and Infrastructure

- *[Describe the computing environment and software tools to be used in fulfilling the CM functions throughout the project or product lifecycle.*
- *Describe the tools and procedures required used to version control configuration items generated throughout the project or product lifecycle.*
- *Issues involved in setting up the CM environment include:*
  - *anticipated size of product data*
  - *distribution of the product team*
  - *physical location of servers and client machines]*

<Project Name>	Version: <1.0>
Configuration Management Plan	Date: <dd/mmm/yy>
<document identifier>	

### 3. The Configuration Management Program

#### 3.1 Configuration Identification

##### 3.1.1 Identification Methods

- *[Describe how project or product artifacts are to be named, marked, and numbered. The identification scheme needs to cover hardware, system software, Commercial-Off-The-Shelf (COTS) products, and all application development artifacts listed in the product directory structure; for example, plans, models, components, test software, results and data, executables, and so on].*

##### 3.1.2 Project Baselines

- *[Baselines provide an official standard on which subsequent work is based and to which only authorized changes are made.*
- *Describe at what points during the project or product lifecycle baselines are to be established. The most common baselines would be at the end of each of the Inception, Elaboration, Construction, and Transition phases. Baselines could also be generated at the end of iterations within the various phases or even more frequently.*
- *Describe who authorizes a baseline and what goes into it.]*

#### 3.2 Configuration and Change Control

##### 3.2.1 Change Request Processing and Approval

- *[Describe the process by which problems and changes are submitted, reviewed, and dispositioned.]*

##### 3.2.2 Change Control Board (CCB)

- *[Describe the membership and procedures for processing change requests and approvals to be followed by the CCB.]*

#### 3.3 Configuration Status Accounting

##### 3.3.1 Project Media Storage and Release Process

- *[Describe retention policies, and the back-up, disaster, and recovery plans. Also describe how the media is to be retained—on-line, off-line, media type, and format.*
- *The release process should describe what is in the release, who it is for, and whether there are any known problems and any installation instructions.]*

##### 3.3.2 Reports and Audits

- *[Describe the content, format, and purpose of the requested reports and configuration audits.*
- *Reports are used to assess the “quality of the product” at any given time of the project or product lifecycle. Reporting on defects based on change requests may provide some useful quality indicators and thereby alert management and developers to particularly critical areas of development. Defects are often classified by criticality (high, medium, and low) and could be reported on the following basis:*
  - *Aging (Time-based Reports): How long have defects of the various kinds been open? What is the “lag time” of when in the lifecycle defects are found versus when they are fixed?*
  - *Distribution (Count Based Reports): How many defects are there in the various categories by owner, priority or state of fix?*

<Project Name>	Version: <1.0>
Configuration Management Plan	Date: <dd/mmm/yy>
<document identifier>	

- *Trend (Time-related and Count-related Reports): What is the cumulative number of defects found and fixed over time? What is the rate of defect discovery and fix? What is the “quality gap” in terms of open versus closed defects? What is the average defect resolution time?]*

#### **4. Milestones**

- *[Identify the internal and customer milestones related to the project or product CM effort. This section should include details on when the CM Plan itself is to be updated.]*

#### **5. Training and Resources**

- *[Describe the software tools, personnel, and training required to implement the specified CM activities.]*

#### **6. Subcontractor and Vendor Software Control**

- *[Describe how software developed outside of the project environment will be incorporated.]*