***Software Configuration Management Plan***

# Introduction

## Scope and Intent of SCM Activities

The main focus of software configuration management (SCM) is to identify and control major software changes, ensure that the changes are implemented correctly, and other employees or customers who may be interested in them. The goal of the SCM is to limit the impact of changes on the entire system. This allows you to avoid unnecessary changes and monitor and control the changes you need. As a result, software development can continue without significant backtracking, despite major or insignificant changes. This reduces development time and results in higher quality products. The SCM team oversees these activities and any changes to existing code or architectural design should be reviewed before execution.

## SCM Organizational Role

The SCM group will paintings intently with the SQA (Software Quality Assurance) group, go inspecting some of the submitted files and software program requests. Software Engineers will publish requests without delay to the SCM group for his or her inspection and approval.

The SCM Lead is appointed to oversee all SCM activities. He receives all change requests and makes all final decisions regarding those changes, including the software engineer who makes the approved changes. SCM Manager also maintains a library of all submitted requests, including those that have been rejected.

# SCM Tasks

## Identification

*Description*

The SCM Manager analyzes all current design specifications and breaks down the software into subsystems. All subsystems consist of important software features or interface components. All submitted changes are linked to the appropriate subsystem and tracked through the system to determine its impact.

*Work Products*

The SCI document contains a breakdown of the subsystems and their relationships. All approved changes are returned to the software developer with a change approval sheet, a list of potentially affected subsystems, and additional information that the software engineer may need before initiating a code change.

## Configuration Control

*Description*

The software engineer sends a change request to the SCM manager. The SCM manager then analyzes the request using SCI documents, project draft documents, and current prototypes of the software. He makes decisions based on the extent to which changes affect the entire system, especially the corresponding subsystems.

Once the decision is made, you need to present the changes to the selected software engineer, update the SCI documentation to reflect the changes, and update the SCM library to record the change requests and decisions

*Description of Configuration Control Task: Submitting Change Requests*

The software engineer sends a change request to the SCM manager. PA Software is a very small development company, so all employees work closely together to create a more casual atmosphere. For this reason, software engineers do not need formal application documents when submitting change requests. These inquiries can be made by email or verbally (if the SCM Manager has documented the inquiries). Formal requests are always accepted, but PA software does not maintain a standard form. All requests, whether approved or not, are recorded in the SCM request library. The record contains the name of the requesting engineer, the date of the request, the subject of the request, and if applicable, whether it was approved.

*Description of Configuration Control Task: Request Analysis*

The SCM manager then analyzes the request using SCI documents, project draft documents, and current prototypes of the software. SCI documents are used to track impacts across relevant subsystems and ultimately throughout the system. Refer to the original design document to ensure that the requested changes are within the requirements specification and overall concept. The SCM Manager can look at the current prototype if you are not sure if you need to make changes. This is required if you want to change the appearance of the interface. He makes decisions based on how the changes affect the entire system, especially the corresponding subsystems

*Description of Configuration Control Task: Request Disapproval*

If the SCM Manager determines that the change is not necessary, it will contact the software engineer who made the request and explain why the request was rejected. The software engineer can discuss this decision with the SCM manager at this point and resend the modified change request if another understanding is obtained.

*Description of Configuration Control Task: Request Approval*

If SCM Manager determines that changes are needed, it updates the SCI document to reflect the changes. This may include related subsystem changes or system-wide implications. After making changes to the SCI document, it will be sent back to Software Engineer to notify you of all possible subsystems for monitoring after the changes have been introduced. If these changes significantly change the functionality of the software or the way the user interacts with the software, the customer will be contacted for approval before the software engineer is given approval.

*Work Products and Documentation*

*Submitting Change Requests*

SCM Library is updated to reflect the request.

*Request Analysis*

None.

*Request Disapproval*

SCM Library is updated to reflect the disapproval.

*Request Approval*

SCM Library is updated to reflect the approval. The SCI is amended.

## Version Control

*Description*

Each time a system or subsystem is updated, the program build number (version number) is updated to reflect the changes. The version numbers follow a standard x.x.x input mask (for example, version 1.2.7) and each digit corresponds to an increased severity of the change. One hundredth (x.x.x) reflects a very small change to the software. The 10th place (x.x.x) reflects a wider range of software changes. The location of 1 (x.x.x) reflects a significant change to the software. Version control is done manually. No source code tracking / version control tools are used.

*Description of Version Control Task: Minor Version Change*

When a small bug in the software is fixed, it will increase by a factor of 100. This change reflects a single error being fixed, or a small design change that has little or no impact on the surrounding subsystems. These changes are usually peculiar and can be large enough to affect thousand digits (eg).

*Description of Version Control Task: Substantial Version Change*

If there are major changes in the software, the 10th place will increase. This change is significant enough to affect how the software works internally or to slightly change the way the user interacts with the user interface, with many minor errors or design changes. It reflects the simultaneous correction.

*Description of Version Control Task: Severe Version Change*

If there is a significant change in the software, the unit digits will increase. This change is a serious enough interface revision to make design decisions that affect the manners in which most of the subsystems interact, major changes in software functionality, or changes in the way users interact. It reflects. Use the device to completely change the interface

*Work Products and Documentation*

*Minor Version Change*

The SCM library will note the version change and the bug that was

fixed.

*Substantial Version Change*

The SCM library will note the version change and the bugs that were fixed or design changes were made. If design changes are made, the SCI document will also be updated.

*Severe Version Change*

The SCM library will note the version change and the design decisions that were made. All affected subsystems will be updated in the SCI document.

## Configuration Status Accounting (CSA)

*Description*

As soon as the SCI document changes, it will be returned to the software engineer, and after the changes are introduced, the SCM Manager will notify you verbally or by email for monitoring of all potentially affected subsystems. To do. If these changes significantly change the functionality of the Software or the way you interact with the Software, we will contact you before any changes are made. A draft of the changes will be sent to you for review. If you decide that you need to make changes, you can contact the SCM Manager with your consent.

*Work Products and Documentation*

A draft of the changes will be sent to client. Once approved, the final change document will be changed to an SCI document.

## Audits and Reviews

*Description*

Software configuration audits and formal technical reviews are performed to ensure that the changes are implemented properly. The SCM team adheres to the standard process established by PA software. Regularly reviewing the behavior of your SCM team will help you determine if your SCM team is performing its activities properly and thoroughly. This is done to ensure that the changes made maintain the original quality of the design and do not introduce new or design flaws. If you find a violation, contact your SCM manager. It is his responsibility to take corrective action to resolve the issue

*Work Products and Documentation*

A review document is created that catalogs the performance of the SCM team and identifies issues with the following standard processes.

## Data Collection and Evaluation

All change requests are sent to the SCM manager, which documents them and saves them in the SCM library. Hard copies are stored in the SCM office and electronic copies are stored in the PA software network.

Due to the informal atmosphere of the PA software, all information except the most important information can be sent informally by email or word of mouth. Unless otherwise specified in the circumstances, all contact with customers may be unofficial.

# Standards, Practices and Conventions (SPC)

It is the responsibility of all software engineers to inform the SCM manager of major design changes and implementation deviations. This process takes into account the impact of changes to the rest of the software. All major changes, including the requested software engineer and the requested date, are recorded by the SCM Manager.

All software engineers are expected to thoroughly test each complete subsystem of the software according to the guidelines provided in the test specifications. This is to ensure that each subsystem is functioning properly and efficiently. All critical defects found will be reported to the SCM Manager. All SCM team members roughly follow some criteria in carrying out their activities.

# SCM Resources

### Personnel

SQA team leader

SQA team members (3)

### Hardware

No special hardware is required.

### Software

No special software is required, but access to a central database containing the SQA defect log would be preferable in communicating between the SQA team and software engineers.

### Tools

No special tools are required.

# Software Quality Assurance Overview

The goal of the SQA team is to ensure that the product does not deviate significantly from the original design specifications. If it is determined that a discrepancy has occurred, the SQA team will notify the development team to prevent future discrepancies and correct previous discrepancies. In addition, the SQA team will carry out walkthroughs to analyze product quality at each specific development phase. Error detection and possible improvements are also announced to the development team.

The organizational role of SQA is to review the product at specific times during product implementation. After the review, the task of the SQA team is to evaluate the software at the current stage of development and identify defects in the next phase (design or implementation). The SQA team will interact directly with the software engineering team in group discussions to discuss identified errors or possible improvements. In addition, the SQA team ensures that the software engineering team does not deviate from the original design specifications.

# SCM Tools, Techniques, and Methods

All SCM activities follow the same guidelines and methods (see Appendix). All SCM meetings include all group members. Each group member is expected to participate in the discussion. Group members who do not participate in the review will be notified by the SCM leader of what happened during the review. The SCM leader oversees the discussion and notes any errors or improvements that need to be analyzed.

The SCM team analyzes errors or improvements and determines their complexity, system impact, and priorities. After prioritization, the SCM manager assigns each point to the software engineer along with the priority. After the error has been fixed or an extension has been added, the software engineer will notify the SCM manager at the next SCM check.

The SCM manager will note the modification of the SCM library. The SCM team does not require any special tools, but it is recommended to access a central database accessible to all group members to avoid time and duplicate errors.