

**CONFIGURATION MANAGEMENT PROCEDUCE TEMPLATE**

**Software Process And Quality Management**

**Team 5 K16T1**



|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Author | Description |
| 25/12/2013 | 1.0 | Big Five Team | Create Document |
| 30/12/2013 | 1.1 | Big Five Team | Update Document |

**REVISIONS**

Table of Contents

[List of table: 4](#_Toc376254948)

[1. INTRODUCTION FOR USE AND TAILORING OF THIS DOCUMENT: 5](#_Toc376254949)

[2. INTRODUCTION: 5](#_Toc376254950)

[2.1. SCOPE: 6](#_Toc376254951)

[2.2. REFERENCE MATERIALS 6](#_Toc376254952)

[2.3. AUDIENCE 6](#_Toc376254953)

[2.4. ROLES AND RESPONSIBILITIES 6](#_Toc376254954)

[2.5. REFERENCE DOCUMENT: 9](#_Toc376254955)

[3. SCM PROCEDUCES 10](#_Toc376254956)

[3.1. CONFIGURATION IDENTIFICATION 10](#_Toc376254957)

[3.2. SOFTWARE PRODUCT IDENTIFICATION AND CLASSIFICATION 11](#_Toc376254958)

[3.3. TEST IDENTIFICATION 13](#_Toc376254959)

[3.4. RELEASE IDENTIFICATION: 14](#_Toc376254960)

[3.5. BUILD IDENTIFICATION: 15](#_Toc376254961)

[3.6. BASELINE IDENTIFICATION 16](#_Toc376254962)

[3.7. SOURCE FILE IDENTIFICATION 17](#_Toc376254963)

[3.8. DOCUMENT IDENTIFICATION 17](#_Toc376254964)

[3.9. CHANGE REQUEST IDENTIFICATION 17](#_Toc376254965)

[4. CONFIGURATION/CHANGE CONTROL 18](#_Toc376254966)

[4.1. ROLES AND RESPONSIBILITIES 19](#_Toc376254967)

[4.2. SCM CHANGE TRACKING 21](#_Toc376254968)

[4.2.1. CHANGE RECORD DEFINITION AND TYPES 22](#_Toc376254969)

[5. SCM STATUS ACCOUNTING AND SCM STATUS REPORTING 23](#_Toc376254970)

[6. CONFIGURATION AUDIT/VERIFICATION 24](#_Toc376254971)

[6.1. AUDITS 24](#_Toc376254972)

[6.2. BUILD AUDITS: 25](#_Toc376254973)

[7. DEVELOPMENT ENVIRONMENT 25](#_Toc376254974)

[7.1. PRODUCT CONTROL WITH SCM TOOLS 25](#_Toc376254975)

[7.1.1. DEVELOPMENT BUILDS 26](#_Toc376254976)

[7.1.2. FORMAL BUILDS 27](#_Toc376254977)

[7.1.3. <SCM TOOL> IMPLEMENTATION 27](#_Toc376254978)

[7.1.4. DOCUMENTATION REPOSITORY 29](#_Toc376254979)

[7.1.5. DEVELOPMENT TOOLS 34](#_Toc376254980)

# List of table:

[Table 1: Roles and responsibility 9](#_Toc376254896)

[Table 2: List configuration Item 11](#_Toc376254897)

[Table 3: Current Status of Project Template 13](#_Toc376254898)

[Table 4: List Testing Configuration Items template 14](#_Toc376254899)

[Table 5: List Configuration Items Name Template 15](#_Toc376254900)

[Table 6: List Configuration Items Name Template 16](#_Toc376254901)

[Table 7: List baseline format Template 17](#_Toc376254902)

[Table 8: List SCM Tool Template 18](#_Toc376254903)

[Table 9: List authority Name Template 19](#_Toc376254904)

[Table 10:Roles and responsibility 21](#_Toc376254905)

[Table 11: CM Tool tracking template 22](#_Toc376254906)

[Table12: Changed CM Tool tracking template 23](#_Toc376254907)

[Table 13: Changed CM Tool attribute tracking template 23](#_Toc376254908)

[Table 14: CM Tool status report template 24](#_Toc376254909)

[Table 14: Implemented CM Tool template 26](#_Toc376254910)

[Table 15: Implemented CM Tool Description 27](#_Toc376254911)

[Table 16: Implemented CM Tool Description 28](#_Toc376254912)

[Table 17: Implemented CM Tool Description 28](#_Toc376254913)

[Table 18: Implemented CM Tool Description 28](#_Toc376254914)

[Table 18: Implemented CM Tool Description 29](#_Toc376254915)

[Table 18: Implemented CM Tool Description 29](#_Toc376254916)

[Table 10: Implemented CM Tool Description 30](#_Toc376254917)

[Table 21: Implemented CM Tool Description 30](#_Toc376254918)

[Table 22: Implemented CM Tool Description 34](#_Toc376254919)

[Table 23: Implemented CM Tool Description 34](#_Toc376254920)

# INTRODUCTION FOR USE AND TAILORING OF THIS DOCUMENT:

*The Software Configuration Management (SCM) Procedures provides a uniform approach to SCM for VA software products which could be developed in house, embedded, purchased, or outsourced software, third party frameworks or packages. The SCM Procedures template is in compliance with the SCM Plan Standard published within ProPath. Each software development project may have minor variations in their procedures in order to best suit the needs of the specific project working on a VA product. All SCM Managers should consult the template when updating an existing SCM Procedures document.*

*This document is intended to be used as a template to assist SCM Managers in developing a product specific SCM Procedures. There will be a unique SCM Procedures document for every single VA Product/Application/Module, the VA Product depending on how the scope was defined in the VA Software Product Registration process. If it does not exist, the SCM manager will create the SCM Procedures for each Product within the scope of their Project. If it exists, the SCM Manager will update the SCM Procedures for each Product within the scope of their Project. The ownership of this document will be shared by all SCM Managers where the Product is within their Projects scope.*

*The blue text provides instructions on the content of each section. The black text provides examples of the section content. The blue text must be removed from the product specific SCM Procedures. The SCM Procedures will serve as the basis for detailed work instructions that each SCM Manager will author for each project creating or modifying the Software Product. If sections or subsections of this template are not applicable to the specific product, remove those section(s) with explanation in the Revision History Description of the document. Throughout the Black text there will be blue brackets with blue text <text> that will need to be modified with the actual information and the <> deleted, changing the font and color to regular font and color; for example “It describes the procedures for managing and controlling the development, delivery, and maintenance of the specific Software Product <Product name>”, will become, “It describes the procedures for managing and controlling the development, delivery, and maintenance of the specific Software Product Patch Module”. The words Product name is replaced with Vista Patch Module within the SCM Product created to manage the Patch Module within the system.>*

# INTRODUCTION:

As identified in the Software Configuration Management (SCM) Plan Standard, the implementation of a formal and structured SCM environment ensures that all Software Development product artifacts are base lined and maintained in a stable environment.

This SCM Procedures identifies the procedures that conform to the requirements identified in the SCM Plan Standard. This document is intended to provide a uniform approach to SCM for the software product being developed or modified by projects regardless of location or staffing model. It describes the procedures for managing and controlling the development, delivery, and maintenance of the specific Software Product *<Product name>*.

## SCOPE:

*<This section defines the extent of applicability of the document for a product. If certain activities are to be specifically excluded, then those activities must be identified as out of scope. The following standard text is to assist the person fulfilling the role of SCM Manager, who is responsible for performing the SCM responsibilities outlined in the OIT ProPath Process Maps and the SCM Plan Standard. This section will describe the scope of the SCM efforts pertaining to the VA Product that is being created or modified by a project or multiple projects.>*

The SCM Procedures applies to *<Product Name>* under development or maintenance. It also applies to all documentation products and other project or program initiative documentation that VA management communicates now or in the future as required to be controlled by SCM procedures. Each project associated with the product will develop work instructions for the implementation of these procedures.

## REFERENCE MATERIALS

## AUDIENCE

*<This section identifies the staff to which the SCM Procedures applies and/or is responsible for implementation of SCM procedures. This is a general definition and is not intended to define roles and responsibilities.>*

The primary audience for this document consists of staff assigned to projects where *<Product Name>* is within scope are required to implement and apply SCM procedures.

## ROLES AND RESPONSIBILITIES

*<This section identifies the specific roles and responsibilities as they relate to SCM, each VA Project will identify the role that will be responsible for the Product. The SCM Manager will create work instruction documents to assist the project team members with the responsibilities within their assigned role. Each Project will identify who is assigned to each role by having one roles and responsibilities table below per project by coping table for each project and pasting directly below the previous table provided.>*

The table below is a specific list of the personnel who may be members of Project teams and SCM teams along with their assigned roles and responsibilities as they relate to SCM. The Roles defined herein can sometimes be overlapped with other roles and responsibilities depending on the environment. In addition, one person allocated for a specific role as listed below may often have the responsibility of other roles.

|  |  |
| --- | --- |
| *<Named Project(s)>* | |
| Role | Responsibility |
| Program Manager/  Project Management  *< Assignee>* | * Develops and maintains artifacts following proper version control procedures using the SCM Procedures and work instructions for each VA Product being worked as part of the Program/Project. * Ensures proper execution of the SCM Plan Standard. * Oversees the SCM process. * Assesses and evaluates all other change requests. * Establish appropriate Change Control Board (CCB). * Submit CCB baseline information. * Identify dependent projects. * Establish/revise required artifacts. * Creation of SCM Procedures and work instructions for each VA product they are assigned. |
| Software Configuration Manager  *<Assignee>* | * Educates project team members in SCM “best practices.” * Develops and maintains SCM Procedures and work instructions for each VA product they are assigned. * Establishes, promotes, and releases baselines. * Works with the Enterprise Tools Team to create and administer the SCM repositories. * Works with the Enterprise Tools Team to build and manage the SCM tool environment. * Performs or validates interim and final builds. * Prepares release package, release archives and Version Description Documents (VDD). * Accountable for instituting the established processes and reporting progress statistics based on change requests. * Attends CCB meetings. * Identifies product baselines as necessary of all products within their assigned Projects. * Responsible for SCM audits and necessary status accounting related to the product. * Conducts audits at scheduled milestones. |
| Development Manager/Leads  *<Assignee>* | * Develops and maintains artifacts following proper version control procedures using the SCM Procedures and work instructions. * Submits build/release requests. * Coordinates development activities and assigns tasks. * Ensures all SCM Procedures and work instructions are implemented and followed for all software, documentation, and/or any other components for which they are responsible. * Ensures all developers’ work within the specified SCM process and related guidelines as specified in the SCM Procedures and work instructions. * Attends the CCB meetings and provide technical details, as required. |
| Developers/System Administration/Functional/ Technical Analysts/DBAs/System Administration  *<Assignee>* | * Develops and maintains artifacts following proper version control procedures using the SCM Procedures and work instructions. * Maintain accurate, detailed information for all assigned change requests (CRs), in the CR database, related to the applicable development detail of the CRs lifecycle. * Provide impact analysis reporting for the CCB approved problems or changes, including documentation of suggested solutions to facilitate CCB disposition activities. * Documentation of build, release, and installation instructions. |
| Software Change Manager | * Develops and maintains artifacts following proper version control procedures using the SCM Procedures and work instructions. * Governing body for reviewing and approving change requests under the SCM Procedures and work instructions. |
| Technical Writer  *<Assignee>* | * Develops technical deliverable documentation to support the software deliverables. * Provides editing, formatting, and graphics support for documentation. * Develops and maintains artifacts following proper version control procedures using the SCM Procedures and work instructions. |
| Software Quality Assurance Manager  *<Assignee>* | * Develops and maintains artifacts following proper version control procedures defined in the SCM Procedures and work instructions. * Ensures all SQA Analysts work within the SCM Procedures and work instructions. * Verifies that only SCM-approved deliverables are installed into the test environment(s). * Ensures that SQA Analysts are always testing from official SCM deliverables. * Attends CCB meetings and provides testing details, as required. * Reviews status accounting related to the project. * Reviews deliverable artifacts. |
| Software Quality Assurance Analysts/Testing Analyst/Testing Service Analyst  *<Assignee>* | * Develops and maintains artifacts following proper version control procedures using the SCM Procedures and work instructions. * Responsible for testing installed releases, as SCM provides releases from development. * Update CRs assigned to them according to test activity results. * Determines Pass/Fail for each CR scheduled for a release. * Opens CRs (defect and or enhancements) for any newly discovered problems during testing. |
| Release Manager/Implementation Team/EVS/Operations Team/Deployment Team-Enterprise Product Support (EPS)  *<Assignee>* | * Develops and maintains artifacts following proper version control procedures using the SCM Procedures and work instructions document. * Coordinates the release and deployment of software to the existing sites and the newly activated sites following SCM Procedures and work instructions. * Assures products meet all exit criteria prior to release * Assures change control and SCM processes have been followed as defined in the SCM Procedures and work instructions. |
| Process Engineer  *< Assignee>* | * Develops and maintains artifacts following proper version control procedures using the SCM Procedures and work instructions. * Guides the Team members in following the EPG published process maps. |

Table 1: Roles and responsibility

## REFERENCE DOCUMENT:

*<This section identifies reference documents that, although not a part of this document, serve to strengthen and clarify its contents. The listed documents provide reference material for background information only. Contact the SCM team for instructions on how to obtain copies of the released versions of these documents. The Project assigned SCM Manager will list any additional documents that pertain to the project(s), that are within their scope of responsibility.>*

# SCM PROCEDUCES

*<The following sections defines the procedures used for configuration identification, SCM repositories, configuration change control, configuration status accounting, configuration audits, release administration, and archive and storage. It is expected that each Project SCM Manager working on the named product will modify the information as being implemented by their project*

## CONFIGURATION IDENTIFICATION

*<This section describes the Configuration Identification of the Software Product and providing a unique identity to the product, it’s components, and associated documentation, including the definition of appropriate level of identification.  In order to identify the configuration item(s)(CI)s that are to be placed under SCM control, the SCM Manager must understand that Configuration Identification is the process of selecting the CIs and the development items subject to Change Control for a product, assigning unique identifiers to them, and recording their functional and physical characteristics in technical documentation.*

*The following items are subject to configuration identification for software products as per the SCM Plan Standard and are to be placed under SCM control:*

* *Products that are delivered to the customer*
* *Designated internal work products, including source code used to generate the deliverable*
* *Commercial off the Shelf (COTS) products*
* *Government off the Shelf (GOTS) products*
* *Non Developmental Items (NDI) products*
* *Tools*

*Other items that are used in creating and describing these work products, including documentation describing the function and physical requirements and characteristics of the product*

*These items consist of the set of currently approved or conditionally approved technical documentation, modeling artifacts, source code, executable images, and object files that identify and describe the functional and physical characteristics of the application.*

*The Project assigned SCM Manager will work with the respective Program/Project Manager(s) with regards of the Product that is in scope and retain compliance with ProPath process maps in order to select, and identify the appropriate CIs. The SCM Manager will then document the items in this section of the SCM Procedures and maintain the items under configuration control, including and/or related to the following:*

* *Documentation*
* *Change Requests*
* *Software Versions*
* *Repositories*
* *VistA Patches*
* *COTS/GOTS/NDI Product Identification*
* *Test Configurations and Test Documentation*
* *Builds*
* *Baselines*
* *Releases*

*The SCM Manager will create work instruction documents to assist the project team members with how they will maintain the CIs that have been identified in this section for their assigned role. Each Project will document the CI categories that are being placed under SCM control by duplicating the table provided for each project below the previous table. The CI categories listed in the table(s) below are only examples; the SCM Manager must list the actual CI categories that will be placed under control for the named product.*

The SCM Manager(s) have worked with the Program/Project Manager(s) in compliance with ProPath process maps to select, and identify the appropriate CIs to be placed under configuration control for *<Named Product>*, these CI categories will be maintained under configuration control, included in the table below:

|  |  |  |  |
| --- | --- | --- | --- |
| *<Named Project(s)>* | | | |
| <Named Product> | Analysis and Design Documentation | Business Modeling Documentation | Configuration and Change Management Documentation |
| Source Code | Project Management Documentation | Test Documentation | Implementation Documentation |
| Configuration Files | Builds | Release Candidates | Releases |
| Database Scripts | Database Seed Data | Environments | User Access and Permissions |

Table 2: List configuration Item

## SOFTWARE PRODUCT IDENTIFICATION AND CLASSIFICATION

*<This section defines how the software is identified and classified.*

*Each software product is given a name or namespace within the VA Product Registration Process which is outside of the scope of this document. The Program/Project Manager is responsible for obtaining the product registration and will communicate that information to the SCM Manager for it to be recorded in this section of the document. The Program/Project Manager and the SCM Manager should use the following information to determine the classification of the product to be one of the following:*

* ***Commercial Off-the-Shelf (COTS)***

*A COTS item is defined as a commercial item that is of a type customarily used by the general public or by non-governmental entities for purposes other than governmental purposes, and:*

* *Has been sold, leased, or licensed to the general public; or has been offered for sale, lease, or license to the general public*
* *Has been sold or offered for sale in substantial quantities in the commercial marketplace*
* *Has been offered to the Government, under a contract or subcontract at any tier, without modification, in the same form in which it is sold in the commercial marketplace*

*COTS items shall be identified within the system configuration by the manufacturers name, item identification, and version in sufficient detail to allow re-acquisition of the identical item. If a COTS item is changed in such a manner that it no longer meets the definition of COTS, the item must be reclassified by its new classification.*

* ***Government off-the-shelf (GOTS) Item***

*A GOTS item is defined as any previously developed item of supply used exclusively for governmental purposes by a Federal agency, a State or local government, or a foreign government with which the United States has a mutual defense cooperation agreement.*

*GOTS items shall be identified within the system configuration by the Government item identification and version in sufficient detail to allow re-acquisition of the identical item. If a GOTS item is changed in such a manner that it no longer meets the definition of GOTS, the item must be reclassified by its new classification*

* ***Non-Developmental Item (NDI)***

*An NDI is defined as any COTS item that requires only minor tailoring of a type customarily available in the commercial marketplace, and is within the normal function of the COTS item. This tailoring does not include modification or customization beyond what is normally provided in the commercial marketplace and is outside of the provider’s normal pricing structure.*

*NDI items shall be identified within the system configuration by the manufacturers name, item identification, version, and tailoring in sufficient detail to allow re-acquisition of the identical item. If a NDI item is changed in such a manner that it no longer meets the definition of NDI the item must be reclassified by its new classification*

* ***Modified Item***

*A modified item is defined as a COTS or GOTS item which is customized for a specific purpose and to meet specific requirements beyond the normal function of the COTS or GOTS item is defined as a Modified item*

* ***Third Party Item***

*A Third Party Item is defined as a new item or modified item developed by a subcontractor for a specific purpose and to meet specific requirements.*

* ***Developmental Item***

*A Developmental Item is defined as a new item or modified item developed for a specific purpose and to meet specific requirements.*

*Each SCM Manager will need to update the table below if the product is new, being rebranded or reclassified within their respective project. Each SCM Manager will document their project where this product is in scope for the new, rebranding (rename of existing product), classification, and/or status information. Each SCM Manager will document their project, classification, and status by appending to the table below. The project classification and status listed in the table below are examples only.>*

The table below describes the current status of project(s) for the identified Named Product and its classification.

|  |  |  |
| --- | --- | --- |
| *<Named Product>* | | |
| Project | **Classification** | **Status** |
| *<Named Project>* | *<Classification>* | *<Active or In Active>* |
|  |  |  |
|  |  |  |

Table 3: Current Status of Project Template

## TEST IDENTIFICATION

*<This section defines how test procedures will be uniquely identified. This assists in the association to change requests for requirements and testing traceability. The Test tool should assign unique identifiers to each test artifact such as the test plan, test case and procedures. The published test results will be CIs managed with the identified SCM tools. The project will document what tools and techniques will be used to achieve and maintain unique references.*

*The SCM Manager will create work instruction documents to assist the project team members with how they will maintain the testing CIs using the identified SCM and Testing tools for their assigned role. Each Project will document the Testing tools that are being used to manage the test CIs that are under SCM control by duplicating the table provided for each project below the previous table. The Testing tools listed in the table below are examples of the approved Testing tools to be used within the VA. The Program/Project Manager must seek a waiver for use of non-approved Testing tools. The SCM Manager must list the actual Testing tool and waiver status (if applicable) of tools used to perform development testing activities for the Named Product. >*

The Testing tools in conjunction with the SCM tools that have been identified will be used to maintain and manage all selected testing CIs for the *<Named Product>* are listed in the table(s) below.

|  |  |  |  |
| --- | --- | --- | --- |
| *<Named Project(s)>* | | | |
| Testing Tool | **Purpose** | **Status** | **Waiver**  **(if applicable)** |
| *<Testing Tool>* | *<Purpose>* | *<Active or InActive>* | *<Yes or No>* |

Table 4: List Testing Configuration Items template

## RELEASE IDENTIFICATION:

*<This section defines the mechanism by which releases and their contents will be identified by the Product. A release may contain software, documentation or a combination of software and documentation. A software release is made up of builds that become release candidates. A release candidate may or may not become a release. Once the release candidate has gone though all the required steps it becomes the official release.*

*A release includes all CIs for the release and/or changed or new CIs since the last release.*

*A release is identified by a release number and a version number. The different types of releases are:*

*Release Identification is achieved through the use of a five character identifiers that determines the type of release. The Release, Major.Minor.Maintenance.build composition is as follows:*

1. *Release: New features or capabilities where the deliverable is no longer interchangeable with the previous release*
2. *Major: New features, capabilities and accumulated bug fixes where interchangeability is maintained*
3. *Minor: Fixes, enhancement or known problems.*
4. *Maintenance: Patch, to fix unanticipated problems*
5. *Build: The identifying build number that was promoted throughout the testing phase to be released.*

*In the previous example, if a minor release was the next product release, the Release Identification would be 1.0.1.0.4. Notice the maintenance release position was reset to ‘0’.*

*Release Identification assignment is determined by the Program/Project Manager(s) and Integrated Project Team(s) (IPT)s in compliance with ProPath process maps, and then controlled through the SCM procedures.*

*The SCM Manager will create work instruction documents to assist the Project Team members with how they will maintain the release CIs using the identified SCM tools for their assigned role. Each Project will document the how the Named Product will be assigned by duplicating the table provided for each project below the previous table. The SCM Manager must work with the Program/Project Manager(s) and Integrated Project Team(s) (IPT)s in compliance with ProPath process maps to determine the Release Identification assignment for the Named Product.>*

The SCM Manager(s) have worked with the Program/Project Manager(s) and the IPT(s) in compliance with ProPath process maps to select, and identify the appropriate Release Identification for the *<Named Product>*. The naming convention will be as follows:

|  |  |
| --- | --- |
| *<Named Project(s)>* | |
| Release | **Definition** |
| *<Release#.Major#.Minor#.Maintenance#.build#>* | *<Purpose>* |
|  |  |

Table 5: List Configuration Items Name Template

## BUILD IDENTIFICATION:

*<This section defines the build identification format that builds for the Named Product that Project(s) will be utilizing. A build marks a reference point in a software development life cycle that is associated with a products release candidate.*

*A build is a collection of files that represents a useable state of the product. A build is an operational version of a baseline. Only final builds will be in the controlled environment by executing the formal build procedures. Reiterative builds are an important part of the development process. Throughout development, application components are collected and repeatedly compiled for testing purposes, to ensure the reliability of the final product.*

*This is an example of two types of builds:*

*Developer Build: Takes place during the development phases of the Software Development Life Cycle (SDLC). The developer’s goal is to quickly create an executable program so that coding can continue.*

*Product Build: Takes place during all test phases of the SDLC. Product builds are deployed on a build machine or centralized server. The focus is to provide a stable product for testing.*

*Product builds are identified through the use of a build identification (ID). Build IDs will be added as the fifth numeric position to the Release Identification. The build ID is incremented by 1 for each build executed; a build number should not be reused.*

*The SCM Manager will create work instruction documents to assist the project team members with how they will maintain build identification that has been identified in this section for their assigned role. Each Project will document the build identification formats by duplicating the table provided for each project below the previous table. The build identification formats listed in the table below are only examples; the SCM Manager must list the actual build identification formats that will be used for the Named Product.>*

The SCM Manager(s) have worked with the Program/Project Manager(s) and the Developer Team Lead in compliance with ProPath process maps to select, and identify the appropriate build identification formats to be used for *<Named Product>*, these build identification formats will be maintained under configuration control, each SCM build will result in a baseline labelbeing applied within the SCM tool to all CI’s making up that build using the following:

|  |  |
| --- | --- |
| *<Named Project(s)>* | |
| Build Type | **Format** |
| *<Component, Product, Environment, or other build type the Projects determine>* | *<This should be the format that the build baseline label is going to follow, it can be as simple as digits or as detailed as giving the product name, component, date/time and build number>* |
| Component Builds | *<Component>\_bld\_<#>* |
| Product Builds | *<Named Product>\_BLD\_<#>* |
| Environment Builds | *<Enviroment>\_bld\_<#>*  SQA\_BLD\_9 |

Table 6: List Configuration Items Name Template

## BASELINE IDENTIFICATION

*<This section defines how baselines for the Named Product will be identified. A baseline is a snapshot of the developing CIs at a specific point in time, usually applied with the SCM tool.*

*A baseline is a group of CIs. Baselines may be a single work product (requirements document), or a set of work products (code, test case, test results). At a minimum a project should have three baselines: Design Baseline, Operational Baseline (internal for developers & testers), and Product Baseline (for external customer). Each work product or Release Identification will be associated with the baseline IDs. A baseline may be identified by a date code, build identification or other unique identifier. Each SCM baseline that is applied will identify each CI and version of the CI that is part of that baseline.*

*The SCM Manager will create work instruction documents to assist the SCM team members with how to maintain baseline identification that has been identified in this section. Each Project will document the baseline identification formats by duplicating the table provided for each project below the previous table. The baseline identification formats listed in the table below are only examples; the SCM Manager must list the actual baseline identification formats that will be used for the Named Product.>*

The SCM Manager(s) have worked with the Program/Project Manager(s) and the Developer Team Lead in compliance with ProPath process maps to select, and identify the appropriate baseline identification formats to be used for *<Named Product>*, these baseline identification formats will be maintained under configuration control, using the following:

|  |  |
| --- | --- |
| *<Named Project(s)>* | |
| Baseline Type | **Format** |
| *<Component, Product, Environment, or other baseline type the Projects determine>* | *<This should be the format that the baseline label is going to follow, it can be as simple as digits or as detailed as giving the product name, component, date/time and build number>* |
| Component build status | *<Component>\_<date>\_<status>\_bld* |
| Official Product Release | *<Named Product>\_Release\_<Release Identifier>* |

Table 7: List baseline format Template

## SOURCE FILE IDENTIFICATION

*<This section defines how source files will be identified. Source Files will go through a series of changes, in order to ensure proper Source File Identification, a SCM tool will be used. The SCM tool should automatically generate the version number of each CI, such as ClearCase, that is called out in the ProPath Maps.*

*The SCM Manager will create work instruction documents to assist the project team members with how they will maintain the Source File Identification using the identified SCM tools for their assigned role. SCM(s) will document how Source Files are identified that are under SCM control by reviewing the information below in this section and modifying as appropriate. >*

Source files for *<Named Product>* will be identified by its unique name and version.

## DOCUMENT IDENTIFICATION

*<This section defines how documents are to be identified.*

*Documents may go through a series of changes. In order to manage these changes, an initial version of the document must be checked in the SCM Repository. Documents are identified by their unique name and revision/version.*

*The document naming convention should follow Product Document Standards as outlined in the ProPath Maps. The standardization of naming documents is not within scope of this document.*

*The SCM Manager will create work instruction documents to assist the project team members with how they will maintain the Documentation Identification using the identified SCM tools for their assigned role. SCM(s) will document how Documentation Files are identified that are under SCM control by reviewing the information below in this section and modifying as appropriate. >*

Documentation files for *<Named Product>* will be identified by its unique name and version.

## CHANGE REQUEST IDENTIFICATION

*<This section defines how Change Requests (CR)s will be identified. An approved SCM tool will be used to identify all CRs. The SCM tool should automatically generate the unique identifier that will be used to identify the CRs, such as ClearQuest, that is called out in the ProPath Maps. The automatic numbering and assignment of the electronic change request form is handled within SCM tool. This unique number will be tracked in the SCM tools to ensure the change is traceable from approval to implementation and in the case of disapproval; the unique number will remain with the change record showing the disposition.*

*The SCM Manager will create work instruction documents to assist the project team members with how they will identify CRs using the identified SCM tools in this section for their assigned role. Each Project will document the SCM tools that are being used to identify the CRs that are under SCM control by duplicating the table provided for each project below the previous table. The SCM tools listed in the table below are examples of the approved SCM tools to be used within the VA. The Program/Project Manager must seek a waiver for use of non-approved SCM tools. The SCM Manager must list the actual SCM tool and waiver status (if applicable) of tools used to perform development SCM activities for the named product. >*

The SCM tool(s) that will be used to identify all selected CRs for *<Named Product>* pertaining toall development, modifications and customizations for projects are listed in the table(s) below.

|  |  |  |  |
| --- | --- | --- | --- |
| *<Named Project(s)>* | | | |
| SCM Tool | **Purpose** | **Status** | **Waiver**  **(if applicable)** |
| *<SCM Tool>* | *<Purpose>* | *<Active or InActive>* | *<Yes or No>* |
|  |  |  |  |

Table 8: List SCM Tool Template

# CONFIGURATION/CHANGE CONTROL

*<This section contains an introduction to configuration/change control. It should define the purpose of configuration/change control and its benefits to the project. Sub-sections should also contain process instructions including identifying the tools used as well as steps to retrieve the information stored within the tool.*

*Formal change control will be implemented and will be followed to manage changes and versions to the CIs that are introduced to the baseline. This will provide stable, traceable, dependable, and secured development, test, and production environments.*

*A key principle in the change control process is the ability to push the change authority to the lowest possible level while still maintaining sufficient control; change authority will depend upon the phase and the state of the configuration item. Each role in the change process has a set of guidelines to determine authority for either approving change or escalating the change authority to the next level. This will result in a compromise between a stringent, centralized change authority and the flexibility of rapid change required by a software development process.*

*All changes are subject to the Change Control Process and must follow a change request review cycle. As defined in the ProPath OIT Change Control Process, there are different levels that have the ultimate approval authority for changes that impact budget, schedule, overall product functionality, quality or risk to other efforts.  The different OIT CCBs have delegated the authority for specific approvals to other Change Management Governance Boards (CMGB)s and/or individuals and groups. This delegation will follow the guidelines and thresholds for approval and are documented in the PMAS Guide. Evaluation Criteria, Change Request Allocation and Escalation are now part of the PMAS Processes and not within scope of this document.*

*The SCM Manager will create work instruction documents to assist Project team members with how the to follow the Change Control Process to be maintained in this section for their assigned role. Each Project will document the Change Control levels of approval that are being used by duplicating the table provided for each project below the previous table. The level of OIT CCBs and CMGBs listed in the table below are examples. The SCM Manager must work with the Program/Project Manager(s) and IPT(s) to determine the correct levels of approval authority. The SCM Manager must list the actual levels of approval authority that are to be consulted for SCM change activities for the named product.>*

The SCM Manager(s) have worked with the Program/Project Manager(s) and the IPT(s) in compliance with ProPath process maps to define the select the approval authorities be consulted for *<Named Product>*, these baseline identification formats will be maintained under configuration control, using the following:

|  |  |  |
| --- | --- | --- |
| *<Named Project(s)>* | | |
| Approval Authority | **Purpose** | **Context** |
| *<Authority Name>* | *<Purpose>* | *<Organization Level>* |
|  |  |  |

Table 9: List authority Name Template

## ROLES AND RESPONSIBILITIES

*<This section provides examples of roles and responsibilities that may comprise the IPT CHGB.*

*Each Project will document the roles and responsibilities by duplicating the table provided for each project below the previous table. The roles and responsibilities listed in the table below are examples. The SCM Manager must work with the Program/Project Manager and Integrated Project Team (ITP) to determine the correct roles and responsibilities. The SCM Manager must list the actual roles and responsibilities and identify who is assigned to each role SCM change activities for the named product. >*

The SCM Manager(s) have worked with the Program/Project Manager(s) and the IPT(s) in compliance with ProPath process maps to document the roles and responsibilities for *<Named CHGB>* responsible for *<Named Product>*, in the table following:

|  |  |
| --- | --- |
| *<Named Project(s)>* | |
| Role | Responsibilities |
| Senior Project Manager, Product Manager or Development Manager  *<Assignee>* | * Reserves VETO power. * Serves as chair with authority to commit resources and make decisions within escalation criteria. * Senior Program Manager will designate an alternate if unable to attend. * Submit CRs, as appropriate. * Review, evaluate, and coordinate agenda items. |
| Project Manager  *<Assignee>* | * Voting Member * Serves as CCB member responsible for input from assigned project(s) or areas of responsibility. * If Senior PM is not attending, the PM serves as chair with authority to commit resources and make decisions within escalation criteria. * Ensures scope, schedule and budget impacts are fully understood by the board * Submit CRs, as appropriate. * Review, evaluate, and coordinate agenda items. |
| Configuration and Change Manager  *<Assignee>* | * Non-voting Member * Provides subject matter expertise. * Ensures Configuration or Change Management concerns are addressed by the board for each change. * Supports boards by managing PCCB activities to include, but not be limited to: administration, implementation, training, and reporting. * Submit CRs, as appropriate. * Review, evaluate, and coordinate agenda items. |
| Software Quality Assurance (SQA)  *<Assignee>* | * Voting Member * Provides subject matter expertise. * Ensures that SQA concerns are addressed by the board for each change. * Submit CRs, as appropriate. * Review, evaluate, and coordinate agenda items. |
| Functional Personnel as needed (developer, analyst or other)  *<Assignee>* | * Non-voting Member * Provides subject matter expertise in functional area. * Ensures that functional area concerns are addressed by the board for each change. * Submit CRs, as appropriate. * Review, evaluate, and coordinate agenda items. |
| Application Architect  *<Assignee>* | * Voting Member * Provides subject matter expertise. * Ensures scope, schedule and budget impacts are fully understood by the board, * Submit CRs, as appropriate. * Review, evaluate, and coordinate agenda items. |
| Scribe  *<Assignee>* | * Non-Voting Member * Documents discussion items and action items of the CCB meetings and documents in minutes. * Assists in the development of CRs being escalated to the HPS CCB. |

Table 10:Roles and responsibility

## SCM CHANGE TRACKING

*<This section identifies SCM change tracking.*

*This section defines how SCM Change Requests (CR)s will be tracked. An approved SCM tool will be used to track all CRs. The SCM tool should automatically generate the unique identifier that will be used to track the CRs, such as ClearQuest, that is called out in the ProPath Maps. The automatic numbering and assignment of the electronic change request form is handled within SCM tool. This unique number will be tracked in the SCM tools to ensure the change is traceable from approval to implementation and in the case of disapproval; the unique number will remain with the change record showing the disposition.*

*The SCM Manager will create work instruction documents to assist the project team members with how they will use the SCM tool to track the CRs in this section for their assigned role. Each Project will document the SCM tools that are being used to track the CRs that are under SCM control by duplicating the table provided for each project below the previous table. The SCM tools listed in the table below are examples of the approved SCM tools to be used within the product. The Program/Project Manager must seek a waiver for use of non-approved SCM tools. The SCM Manager must list the actual SCM tool and waiver status (if applicable) of tools used to perform development SCM activities for the named product. >*

The SCM tool(s) that will be used to track all selected CRs for *<Named Product>* pertaining to all development, modifications and customizations for projects are listed in the table(s) below.

|  |  |  |  |
| --- | --- | --- | --- |
| *<Named Project(s)>* | | | |
| SCM Tool | **Purpose** | **Status** | **Waiver**  **(if applicable)** |
| *<SCM Tool>* | *<Purpose>* | *<Active or InActive>* | *<Yes or No>* |

Table 11: CM Tool tracking template

### **CHANGE RECORD DEFINITION AND TYPES**

*This section defines SCM Change Requests (CR)s Record Types. An approved SCM tool will be used to standardize the CR records used for Software development. The SCM tool should contain Change Records for the change process instead of the change type. A change record should follow a defined process that allows for the reason of the change to be modified within the change record, instead of having to close and open records if the change reason is modified.*

*The SCM Manager will create work instruction documents to assist the project team members with how they will use the SCM tool to track the CRs in this section for their assigned role. Each Project will document the SCM tools that are being used to track the CRs that are under SCM control by duplicating the table provided for each project below the previous table. The information listed in the table below is examples directly related to the approved SCM tool ClearQuest that is called out in the ProPath Maps and to be used within the VA. The Program/Project Manager must seek a waiver for use of non-approved SCM tools. The SCM Manager must list the actual SCM tool and waiver status (if applicable) of tools used to perform development SCM activities for the named product. >*

The SCM record to be for CRs for *<Named Product>* pertaining to all development, modifications and customizations for projects are listed in the table(s) below.

|  |  |  |  |
| --- | --- | --- | --- |
| *<Named Project(s)>* | | | |
| SCM Tool | **Record** | **Purpose** | **Change Type** |
| *<SCM Tool>* | *<Purpose>* | *<Description of what the record is used to track>* | *<Description of what the change type is used to track>* |
|  |  |  |  |
|  |  |  |

Table12: Changed CM Tool tracking template

#### **SCM CR Attributes**

*This section defines SCM Change Requests (CR)s Attributes. An approved SCM tool will be used to standardize the CR records used for Software development. The SCM tool should contain Change Records with each record having a set of attributes such as fields, behaviors, states and permissions.*

*The SCM Manager will create work instruction documents to assist the project team members with how they will use the attributes for each CR record in this section for their assigned role. Each Project will document the attributes that make up each CR record that are under SCM control by duplicating the table provided for each project below the previous table. The information listed in the table below is examples directly related to the approved SCM tool ClearQuest that is called out in the ProPath Maps and to be used within the VA. The Program/Project Manager must seek a waiver for use of non-approved SCM tools. The SCM Manager must list the actual SCM tool and waiver status (if applicable) of tools used to perform development SCM activities for the named product. >*

The SCM Attributes for each CRs type for *<Named Product>* pertaining to all development, modifications and customizations for projects are referenced in the table below.

|  |  |  |
| --- | --- | --- |
| *<Named Project(s)>* | | |
| SCM Tool | **Record** | **Attributes** |
| *<SCM Tool>* | *<Purpose>* | *<Description of Attributes>* |
|  |  |  |
|  |  |

Table 13: Changed CM Tool attribute tracking template

# SCM STATUS ACCOUNTING AND SCM STATUS REPORTING

*<This section contains an introduction to Status Accounting and Status Reporting. It defines the purpose of Status Accounting and Status Reporting. Status Accounting and Status Reporting is the process of creating and organizing a knowledge base necessary for the performance of SCM. This includes identifying the tools used as well as steps to retrieve the information stored within the tools.*

*This section defines how the SCM repositories are used for Status Accounting of CIs throughout the development lifecycle. SCM tools such as ClearCase and ClearQuest, which are called out in the ProPath Maps should be implemented as the SCM Repositories to enable reporting on the status of CI(s),CRs and health of the overall Named Product or projects.*

*The SCM Status Accounting repository will be used for the recording, tracking, and reporting of information. Data and logs that identify the configuration status, configuration item identification, and change status of all CRs will be maintained to enable reporting. SCM Status Accounting addresses the recording, tracking, and reporting of information needed to effectively manage software development. CRs provide the foundation for SCM Status Accounting but do not stand alone. Therefore additional status reporting should be conducted where appropriate to include: tool and resource usage, progress and the overall product SCM Environment conditions.*

*The processes that enable Status Accounting are critical in the ability to perform Status Reporting. Status Reporting is defined as being able to identify what needs to be tracked, who is responsible, how it is tracked, when it is tracked, how the information is stored and anomalies escalated to management.*

*The SCM Manager will create work instruction documents to assist SMC team members with how the repositories will be used and how to generate reports for assessing the status of CI(s), CR(s) and the health of the Named Product in this section for their assigned role. Each Project will document the SCM tools that are being used as SCM Status Accounting repositories by duplicating the table provided for each project below the previous table. The SCM tools listed in the table below are examples of the approved SCM tools to be used within the VA. The Program/Project Manager must seek a waiver for use of non-approved SCM tools. The SCM Manager must list the actual SCM tool and waiver status (if applicable) of repositories used for SCM activities for the named product. >*

The SCM repositories that will be implemented using the SCM tools for *<Named Product>* to report on CI, CR, Product Status Accounting, and Status Reporting for all products are listed in the table(s) below.

|  |  |  |  |
| --- | --- | --- | --- |
| *<Named Project(s)>* | | | |
| Repository | **Purpose** | **Status** | **Waiver**  **(if applicable)** |
| *<SCM Tool>* | *<Purpose>* | *<Active or InActive>* | *<Yes or No>* |

Table 14: CM Tool status report template

# CONFIGURATION AUDIT/VERIFICATION

## AUDITS

*<This section explains the purpose of audits.*

*The SCM Manager will create work instruction documents to assist SMC Team Members with how to perform SCM audits of the Named Product in this section for their assigned role.>*

SCM Audits ensure the integrity of all CIs from initiation to release of CIs for *<Named Product>*. Audits identify traceability of product changes to the authorization for the change, and ensure that only authorized changes are implemented in the controlled system; Auditing documents authenticate the current configuration of the system and procedures. If any discrepancies are noted they will be corrected and signed off by management and archived in the central repository.

## BUILD AUDITS:

*<This section explains the purpose of Build Audits.*

*The SCM Manager will create work instruction documents to assist SMC Team Members with how to perform SCM audits on builds, the tools used, steps to perform the audit, and defines instructions on reporting the audit results of the Named Product in this section for their assigned role* **>**

Build Audits ensure that configuration control and integrity is maintained at the end of the deployment process for *<Named Product>*. There are many tasks that are performed upon release and deployment of a product. This audit ensures that each task is performed and the results and action items, if any, are documented. A Build Audit must be performed and documented.

# DEVELOPMENT ENVIRONMENT

*<This section outlines the Product Control, SCM Tools, and Development Tools for the Named Product.>*

## PRODUCT CONTROL WITH SCM TOOLS

*<This section defines how the CIs will be controlled and what tools will be used to support SCM activities. The SCM tools such as, ClearCase, ClearQuest, and BuildForge are called out in the ProPath Maps should be implemented as the SCM tools pertaining to the control and management of CIs that have been identified within the categories defined in the Configuration Identification section of this document.*

*A SCM tool for version management control, change traceability control and baseline control of all project artifacts must be implemented and should automate the SCM tasks that software development teams perform daily to track the files and directories used to create the Named Product.*

*A SCM tracking tool for management and notification of CRs for the Named Product should be implemented.*

*A SCM Build tool should implemented for build management for software assembly of the Name Product.*

*The SCM Manager will create work instruction documents to assist the Project Team Members with how they will control and manage the CIs using the identified SCM tools in this section for their assigned role. Each Project will document the SCM tools that are being used to manage the CIs that are under SCM control by duplicating the table provided for each project below the previous table. The SCM tools listed in the table below are examples of the approved SCM tools to be used within the VA. The Program/Project Manager must seek a waiver for use of non-approved SCM tools. The SCM Manager must list the actual SCM tool and waiver status (if applicable) of tools used to perform development SCM activities for the named product.>*

The SCM tools that will be implemented to control and manage all selected CIs and associated documentation for all development, modifications and customizations for *<Named Product>* are listed in the table(s) below.

|  |  |  |  |
| --- | --- | --- | --- |
| *<Named Project(s)>* | | | |
| SCM Tool | **Purpose** | **Status** | **Waiver**  **(if applicable)** |
| *<SCM Tool>* | *<Purpose>* | *<Active or InActive>* | *<Yes or No>* |
| ClearCase | SCM of CIs (version management control, change traceability control and baseline control of all project artifacts) | Active |  |
| ClearQuest | SCM of CRs (tracking, management and notification of all changes) | Active |  |
| BuildForge | SCM of builds (build management for software assembly) | Active |  |
| CruiseControl | SCM of builds (build management for software assembly) | InActive | No |
| CVS | Software Version Control | InActive | No |

Table 14: Implemented CM Tool template

### **DEVELOPMENT BUILDS**

Builds are performed in the development branch(s). Each developer builds portions of the product within their private workspace area, without affecting other developers. Developers can use the integrated development environment (IDE) to build; however, the best practice is to use a build automation tool, following the work instructions for building *<Named Product>*.

### **FORMAL BUILDS**

To ensure consistency, repeatability, and product stability, the only official builds are the ones produced by the SCM Team using the documented build procedures for the <Named Product>.

### **<SCM TOOL> IMPLEMENTATION**

*<This section defines the SCM Tool implementation of non-documentation repositories, repository location, branches, and SCM Tool integrations. Documentation repositories are documented in the following section.*

*The SCM tool, such as ClearCase, should be implemented to support repositories and branching. It must also have the ability to track change comments and baselines capabilities.*

*The SCM Manager will create work instruction documents to assist the Project Team Members with how they will use the SCM tool implementation in this section for their assigned role. Each Project will document the SCM tool implementation that is being used to manage the CIs that are under SCM control. Each SCM Manager will document their repositories, branches, and SCM Tool integrations by appending to the tables below. The tables below are examples of an approved SCM Tools implementation to be used within the VA. The SCM Manager must document the actual SCM tool implementation used to perform development SCM activities for the named product.>*

The SCM Tool is implemented (except Documentation) for the *<Named Product>* as described in the table(s) below.

|  |  |  |
| --- | --- | --- |
| *<SCM Tool>:<Server Location Information>:<SCM Tool Integration (If applicable)>*  ClearCase: Hines <http://vhaishvob1:81/ccweb/>: Soft ClearCase integration using Activities | | |
| Repository | **Status** | **Definition** |
| *<SCM Tool>* | *<Active or InActive>* | *<Purpose>* |
| HAIISS\_pvob | Active | CC Project VOB that keeps track of the metadata pertaining to the CC projects and baselines. |

Table 15: Implemented CM Tool Description

|  |  |  |
| --- | --- | --- |
| *<Repository>* | | |
| Component | **Status** | **Definition** |
| *<SCM Component>* | *<Active or InActive>* | *<Purpose>* |
| HAIISS\_websrv | Active | CC component to store web services artifacts |

Table 16: Implemented CM Tool Description

|  |  |  |  |
| --- | --- | --- | --- |
| *<SCM Tool Project Branching Strategy>* | | | |
| Projects | **Status** | **Definition** | **Components** |
| *<SCM Tool Project>* | *<Active or InActive>* | *<Purpose>* | *<SCM Tool/Named Product Component>* |
| HAIISS\_Websrv | Active | CC project for managing and baselining of web service activities conducted on code produced for the VA product. | HAAIIS\_websrv |

Table 17: Implemented CM Tool Description

|  |  |  |  |
| --- | --- | --- | --- |
| *<SCM Tool Parallel Development Branching Strategy>* | | | |
| Projects | **Status** | **Definition** | **Project** |
| *<Branch or Stream Name>* | *<Active or InActive>* | *<Purpose>* | *<SCM Tool/Named Product Component>* |
| HAIISS\_Websrv\_Int | Active | This stream is reserved for the SCM/Rational Tools Team to perform administrative task such as delivering or syncing other CC projects and or streams and or to create the official release baseline. This steam is for managing and storing all artifacts that have been delivered from the BL (Baseline stream) in accordance of the SCM procedures. Only in extreme cases will any checkout or check-in activities take place on this stream. The check in as well as any delivers to this stream will automatically become part of the next baseline. | HAAIIS\_Websrv |

Table 18: Implemented CM Tool Description

### **DOCUMENTATION REPOSITORY**

*<This section defines the SCM Tool implementation of Documentation Repositories, repository location, branches, and SCM Tool integrations. Non-documentation repositories are documented in the above section.*

*The SCM tool, such as ClearCase, should be implemented to support repositories and branching. It must also have the ability to track change comments and baselines capabilities.*

*The SCM Manager will create work instruction documents to assist the Project Team Members with how they will use the SCM tool implementation in this section for their assigned role. Each Project will document the SCM tool implementation that is being used to manage the CIs that are under SCM control. Each SCM Manager will document their repositories, branches, and SCM Tool integrations by appending to the tables below. The tables below are examples of an approved SCM Tools implementation to be used within the VA. The SCM Manager must document the actual SCM tool implementation used to perform development SCM activities for the named product.>*

The SCM Tool is implemented for documentation of the *<Named Product>* as described in the table(s) below.

|  |  |  |  |
| --- | --- | --- | --- |
| *<SCM Tool>:<Server Location Information>:<SCM Tool Integration (If applicable)>*  ClearCase: Hines <http://vhaishvob1:81/ccweb/>: Soft ClearCase integration using Activities | | | |
| Repository | **Repository** | **Repository** | **Repository** |
| *<SCM Tool>* | *<SCM Tool>* | *<SCM Tool>* | *<SCM Tool>* |
| HAIISS\_pvob | HAIISS\_pvob | HAIISS\_pvob | HAIISS\_pvob |

Table 18: Implemented CM Tool Description

|  |  |  |
| --- | --- | --- |
| *<Repository>* | | |
| Component | **Status** | **Definition** |
| *<SCM Tool>* | *<Active or InActive>* | *<Purpose>* |
| HAIISS\_doc | Active | CC component to store documentation that includes all VA Product information that is not captured in the Req\_Pro database or the Models component. The standard directory structure for OED is included in this document with the description and examples. |

Table 18: Implemented CM Tool Description

|  |  |  |  |
| --- | --- | --- | --- |
| *<SCM Tool Project Branching Strategy>* | | | |
| Projects | **Status** | **Definition** | **Components** |
| *<SCM Tool Project>* | *<Active or InActive>* | *<Purpose>* | *<SCM Tool/Named Product Component>* |
| HAIISS\_Doc | Active | CC project for managing and baseline documentation that is to have development activities conducted by various VA Projects. | HAAIIS\_doc |

Table 10: Implemented CM Tool Description

|  |  |  |  |
| --- | --- | --- | --- |
| *<SCM Tool Parallel Development Branching Strategy>* | | | |
| Projects | **Status** | **Definition** | **Project** |
| *<Branch or Stream Name>* | *<Active or InActive>* | *<Purpose>* | *<SCM Tool/Named Product Component>* |
| HAIISS\_Doc\_Int | Active | This stream is **reserved** for the SCM/Rational Tools Team to perform administrative task such as delivering or syncing other CC projects and or streams and or to create the official release baseline. This steam is for managing and storing all artifacts that have been delivered from the BL (Baseline stream) in accordance of the SCM procedures. Only in extreme cases will any checkout or check-in activities take place on this stream. The check in as well as any delivers to this stream will automatically become part of the next baseline. | HAAIIS\_Doc |
|  |  |  |  |

Table 21: Implemented CM Tool Description

The documentation is typically maintained within the SCM Tool as an independent repository. A standard directory structure within the documentation repository is used to standardize implementation across Software Development. This document structure supports the performance of data mining of artifacts for reporting on an as needed basis to report product/project status of execution of project management requirements and the health of the product development and implementation. Further, the standard directory structure supports the creation of release packages that will require linkages back to project documentation artifacts.

All document repositories regardless of the tool used to maintain the documents shall use the directory structure shown below:

|  |  |  |  |
| --- | --- | --- | --- |
| Directory Name | Full Name | Purpose | Examples |
| A&D\_Docs | Analysis and Design Docs | Documentation that will assist and be used to transform the requirements into a design of the system. | Software Architecture Document  Analysis Documents  Interface Control Documents |
| BM\_Docs | Business Modeling Docs | Business Modeling documentation serves as input to and reference for the requirements of the system. | Business Glossary  Business Vision  Business Architecture Document  Businesses Rules |
| CCM\_Docs | Configuration and Change Management Docs | Documents capture and present information related to the configuration and change management discipline | SCM Procedures  Configuration Audit Finding  Version Description Document |
| Depl\_Docs | Deployment Docs | Documentation that presents information related to transitioning the system presented in the Implementation set into the production environment. | Deployment Plan  Release Notes  Installation Guide  Security Guide  Developer's Guide  System Management Guide |
| Env\_Docs | Environment Docs | Environment documentation capture and present information used as guidance throughout the development of the system to ensure consistency of all artifacts produced. | Development Processes  Development Infrastructure Documentation |
| FOIA\_Docs | Freedom of Information Act Docs | Freedom of Information Act Docs | Freedom of Information Act Docs |
| Impl\_Docs | Implementation Docs | Implementation documents capture and present the realization of the solution presented in the Analysis & Design Set. | Implementation Plan Integration Plan  Implementation Structure Document  Analysis of Runtime Behavior Document  Interface Control Registration (ICR)s; in the case of interfaces with VistA |
| Obsolete\_Docs | Obsolete Documents | Documents that have been replaced by other documents because they are so outdated, they are considered to no longer exist. | HSD&D document that have been rewritten for the new Systems Development organization  Documentation describing a solution that was replaced with another implementation |
| PM\_Docs | Project Management Docs | Project Management documentation capture the artifacts associated with project and process planning and execution | Project Management Plan  Project Schedule  Iteration Assessment  Iteration Plan  Status Reports  Software Development Plan  Risk Management Plan  Issue Management Plan  Problem Resolution Plan  Product Acceptance Plan  Security control documents |
| Reference\_Docs | Reference Materials | Reference Materials | Reference Materials |
| Released\_PDFs | PDFs that have been released | PDFs that have been released | PDFs that have been released |
| Req\_Docs | Requirements Docs | Requirements documentation capture and present information used in defining the required capabilities of the system that are not managed in the Req\_Pro VA Product Repository. | Soda reports generated from Req\_Pro  Crystal reports generated from Req\_Pro  Captured emails from stakeholder dealing with requirements  Documentation used to generate requirements |
| Test\_Docs | Testing, and QA, Docs | Testing, and QA, documentation is developed as products of the test and evaluation activities grouped by responsible role | Master Test Plan  Test Logs  Test Strategy  Test Evaluation Summary |
| Third\_Party\_Docs | Third Party Documentation | Documents pertaining to COTS and or Vendor specifics that do not fit into one of the other defined directories. | Custom install guides  Captured email from 3rd parties  Technical solutions documentation provide from 3rd parties |
| Unique\_Docs | Unique Documents | Documents not fitting in one of the above categories | Developer Notes that are not part of one of the official documents  Meeting minutes that do not belong in one of the other folders. |

Table 22: Implemented CM Tool Description

### **DEVELOPMENT TOOLS**

*<This section outlines any expectations and restrictions imposed on the use of software development tools.*

*In order to meet the minimum functional, architectural, and performance requirements, the Project will use the recommended tools authorized by the VA Technical Reference Model (TRM) as referred to in the SCM Plan Standard. If the tool is not on the list, they must seek a waiver prior to its use until approval has been obtained. At the time of approval, the tool may be removed from the table below.*

*Each Project will document the non-approved tools being used for the Named Product by duplicating the table provided for each project below the previous table. The non-approved tools listed in the table below are examples of the non-approved tools within the VA. The Program/Project Manager must seek a waiver for use of non-approved tools. The SCM Manager must list the actual non-approved tools, use, approval request, and waiver status for the Named Product. >*

The non-approved tools and the waiver status for *<Named Product>* for all projects are listed in the table(s) below.

|  |  |  |  |
| --- | --- | --- | --- |
| *<Named Project(s)>* | | | |
| Tool | **Use** | **Approval Requested** | **Waiver**  **(if applicable)** |
| *<Non-Approved Tool>* | *<Purpose>* | *<Yes or No>* | *<Yes or No>* |

Table 23: Implemented CM Tool Description