|  |
| --- |
| NST LOGO.jpg |
| Software Configuration Management Process |
|  |
|  |

**Version History:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Ver.** | **Date** | **Description of Change** | **Authored / Revised By** | **Reviewed By** | **Approved By** |
| 1.0 | 3rd June, 2008 | Initial Draft | Aaftab Brar |  | Mr. Sudhir Saxena |
| 2.0 | 28th May, 2009 | Release & Update in Roles  | Abhishek Rautela | Ausaf Khan | Mr. Sudhir Saxena |
| 2.1 | 26th August, 2010 | Configuration Audit changed | Abhishek Rautela | Saket Madan | Mr. Sudhir Saxena |
| 2.2 | 23rd Jan 2012 | Update section 6.1.2 | Saket | Dhananjay | Mr. Gurinder Dua |
| 3.0 | 26th March, 2012 | Update in Change Control Board Process and update section 6.1.3 | Saket Madan | Dhananjay Kumar | Mr. Gurinder Dua |
| 4.0 | 22nd Aug 2013 | Update section 6.3, 6.4 & 6.7 for ISMS requirement and change NST logo | Rahul Raj | Saket Madan | Dhananjay Kumar |
| 5.0 | 16th May 2016 | Update section 6.2, 6.6 | Rahul Raj | SEPG Team | Ajay Kumar Zalpuri |
| 5.1  | 29th Nov 2018 | Update section 6.7 for CSA (Configuration Status Accounting Sheet) | Rahul Raj | SEPG Team | Ajay Kumar Zalpuri |

Table of Contents

[1. Purpose 4](#_Toc375161219)

[2. Entry Criteria 4](#_Toc375161220)

[3. Glossary / Abbreviation / Definition 4](#_Toc375161221)

[4. Inputs 5](#_Toc375161222)

[5. Roles and Responsibilities 5](#_Toc375161223)

[6. Tasks 6](#_Toc375161224)

[*6.1 Configuration Management Planning* 6](#_Toc375161225)

[*6.2 Change Control Board Process* 7](#_Toc375161226)

[Figure 1: Change Management Process 8](#_Toc375161227)

[*6.3 Software Change Control Management* 9](#_Toc375161228)

[*6.4 Infrastructure Change Control Management:* 9](#_Toc375161229)

[*6.5 Version Control* 10](#_Toc375161230)

[*6.6 Naming Convention* 10](#_Toc375161231)

[*6.7 Configuration Status Accounting* 11](#_Toc375161232)

[*6.8 Configuration Audit* 12](#_Toc375161233)

[7. Outputs 12](#_Toc375161234)

[8. Validation 12](#_Toc375161235)

[9. Exit Criteria 12](#_Toc375161236)

[10. Related Documents 12](#_Toc375161237)

## 1. Purpose

Changes are inevitable in any system development / support project. Software configuration management provides a discipline for identifying items to be managed, controlling changes, maintaining integrity and traceability of the components that are required to develop the final software product, throughout its life cycle. It is an integral part of the software development process and supports the management and control of the project.

The main objective of software configuration management is to manage the change process and track changes to ensure that the configuration of the software product is accurately known at any given time.

## 2. Entry Criteria

* When Project planning starts.
* Change Request has been accepted. The changes may be of type:
	+ - Change request in environment
		- Change request for functionality including enhancements / modifications
		- Change request for database enhancements
* When a Configuration item is ready for baselining.

## 3. Glossary / Abbreviation / Definition

|  |  |
| --- | --- |
| **Term** | **Definition** |
| **SCCB** | Software Change Control Board |
| **CMP** | Configuration Management Plan |
| **SDS** | Software Design Specification |
| **PP** | Project Plan |
| **SOW** | Statement of Work |
| **SPMP** | Software Project Management Plan |
| **SRS** | Software Requirements Specification |
| **VSS** | Visual SourceSafe |
| **TFS** | Team Foundation Server |
| **Baseline**  | A revision of a work product that is considered complete with respect to a certain set of requirements.  |
| **Configuration Item**  | A work product that requires configuration control. A configuration item may be a single piece of work or a group of files that together form the basis for a single software program or document. There is a one-to-one correspondence between configuration items and modules in the archive. A question to ask when determining whether files should be grouped into a single configuration item is whether or not they should eventually be base lined together |
| **Label**  | A specific form of a tag that is used to identify the revisions of a configuration item. Specific labels will be used to mark a baseline version of a work product. |
| **Major Defect** | A defect that could cause a change to the code or design. |
| **Minor Defect** | Any comment on a work product that is not a major defect. |
| **Non-Trivial Change** | A change to a work product that affects the work product's meaning. Also, in the case of code, any change that affects the code's behavior. |
| **SEPG** | Software Engineering Process Group |
| **Trivial Change** | A change to a work product that does not affect the work product's meaning, including formatting changes in a document, or a change in code that does not affect the code's behavior. |
| **Work Product**  | Documentation, pictures, and code that are produced as a result of work done for the project.  |
| **Version**  | A unique number assigned to each revision of a work product. |

## 4. Inputs

* System Enhancement Request / Problem report
1. Configuration items (CIs) to be baselined.

## 5. Roles and Responsibilities

The Configuration Manager has primary responsibility for defining the software configuration management plan for the project. Project team members may be assigned to perform software configuration management on a full-time or part-time basis depending on the size of the project. The Project Manager may also perform the role of the Configuration Manager. The Configuration Manager is the central control point for changes in the project. Typically, his / her responsibilities include:

* Establishing baselines.
* Ensuring that no unauthorized changes are made to the baseline
* Ensuring that all baseline changes are recorded in sufficient detail.
* Configuration status accounting

## 6. Tasks

### *6.1 Configuration Management Planning*

 The Configuration Manager prepares the Configuration Management plan (may be a part of Project Plan) which will be reviewed and then baselined. The baselined Configuration Management plan is used to perform Configuration Management activities. The major contents of the Configuration Management Plan are:

* + 1. Roles and Responsibilities

This section should identify

1. Configuration Manager
2. Composition of SCCB and when an Enhancement Request / problem report has to be escalated to SCCB
	* 1. The Configuration Items (CI) in the project. A Configuration Item is any item that affects the quality of the software product, and is also subject to change. They may be technical documents, plans, programs, data etc. Examples of CIs are:
* Configuration Management Plan (CMP)
* Code
* Change request (CR)
* Risk Management Plan (RMP)
* Design Standard
* Statement of Work (SOW)
* Team Policy, Plan, and Process Documents like PMP
* Software Requirements Specification (SRS)
* Software Design Specification (SDS)
* Testing Documents
	+ 1. Baseline, working and archive directories and their locations are identified. The directories and rules for keeping the files created that are used by other members of the project are also established in VSS or TFS.
		2. Required Tools needed for Software Configuration Management are identified.
		3. QMS inclusion …

### *6.2 Change Control Board Process*

**Change control** is a formal process used to ensure that changes to a product or system are introduced in a controlled and coordinated manner. It reduces the possibility that unnecessary changes will be introduced to a system without forethought, introducing faults into the system or undoing changes made by other users of software.

The goals of a change control procedure usually include minimal disruption to services, reduction in back-out activities, and cost-effective utilization of resources involved in implementing change.

The Software Change Control Board (SCCB) will be responsible for the authorizing changes to configuration items.

The SCCB will consist of the Managing Director, project manager, configuration manager and Delivery Head.

The composition of the SCCB shall be clearly defined in the configuration management plan (may be a part of the project plan) of the project and below also

 ******

### Figure 1: Change Management Process

### *6.3 Software Change Control Management*

* + 1. Log Enhancement / change requests / problem reports in the Change Request Log. Please refer to the Change Request Log template.
		2. Carry out impact analysis and prepare an impact analysis report. Please refer to the Impact Analysis Report section in the Change Request template below. TP-07-CRF-Change Request.doc
		3. Get the approval from Delivery Manager or the Project Manager as defined in the Configuration Management Plan.
		4. If the change is approved, check out the CIs that will be affected from the baseline directory. Prioritize the change. Carry out appropriate changes in the CIs, get the changes reviewed / tested and thereafter check in the CIs into the baseline directory with a new version number.
		5. The CIs may be baselined now.
		6. Report the status of the change request whether change is implemented or rejected, to the person/group that raised the Change Request.
		7. Update the Change Request Log.

### *6.4 Infrastructure Change Control Management:*

1. Change Request will be initiated by IT Department/or Manager IT itself and will be forwarded to Senior Management via E-mail.
2. Manager IT will work on inputs which he receives for Change Request like reason/Requirement of Change (can be Bandwidth usage reports, new requirements, Reviews/Issues faced for existing antivirus for Antivirus Upgrade, Network/Connectivity set up on new location for accommodation, etc.).
3. Manager IT will identify the areas where requested change can impact. After this analysis, all stakeholders will be informed about the changes and impacted areas. Each changes will be recorded in Change Log Template. A back-out plan will be laid and analyzed for each change.
4. After consultation with all the stakeholders for critical changes and approval from Security Council, IT will execute implementation on the required change. In case any stakeholder has any objection regarding the timing or impact related to the activity, then this is brought in notice of Senior Management who will then take decision on it. IT team will then act accordingly. IT department is responsible for effecting necessary changes.
5. All stakeholders will be informed about the down time (Servers Downtime, Network Connectivity downtime etc.) required for the application of the change so that they can plan their work accordingly. IT team will try to execute these changes preferable on a weekend or holiday unless and until it’s a critical change. There can be changes which need to be implemented in coordination with multiple functions. So, all function owners will be involved in planning and execution of the change.
6. Any initial input from end users (like Software list during Workstation replacement, Server restart procedure during Server shut down/maintenance) will be taken by IT team and other function owners (in case change is to be implemented in coordination with multiple functions) before implementation of change. This will be input when Security Council will analyze, evaluate and take decision on implementation of the change.
7. After effecting the changes, all stakeholders will be intimated via. Email (wherever stakeholder needs to be informed).
8. IT department (and other function owners if they are involved in the implementation) will make necessary arrangements to revert to earlier state, if change is unsuccessful.
9. During all phases of the change, the evidence of Change Requirement/ Reason, Approvals, will be maintained.

### *6.5 Version Control*

Check in the new version of CIs with the promoted version numbers. It is suggested that M.nn is the structure maintained for keeping the version control. M and nn are integers. ‘M’ is the major version number and is incremented when there is either a change in the base environment or a major change in the functionality. The other changes are represented by incrementing the minor version ‘nn’.

 For example:

* The baselined version of a document is doc1.0
* If a minor change is made to the document, then its version is changed to doc1.1
* If a major change is made to the document, then its version is changed to doc2.0

### *6.6 Naming Convention*

* 1. The Naming convention **for labelling project** artifacts used will be: ProjectName\_DocName\_versionNumber
	2. The Naming convention **for Document names** will be abbreviated like for Software Requirement Specification, DocName will be SRS and SSD for Software Design Document.
	3. The Naming convention **for** **labeling project Code** used will be:

Code Build for\_Version Number\_MMDDYY

For ex-

1. If build is for System Testing then Code labeling should be like this; SYS\_Rel1\_1.0\_MMDDYY
2. If build is for UAT Testing then Code labeling should be like this; UAT\_Rel1\_1.0\_MMDDYY
3. If build is for Production then Code labeling should be like this; PROD\_Rel1\_1.0\_MMDDYY

* 1. The Naming convention **for labeling Quality management system document** used on the basis of document type will be:

 **Document Type**

* Checklist: for this use CK
* Guidelines: for this use GD
* Policies & Procedures: for this use PO
* Processes: for this use PR
* Templates: for this use TP

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Common Policy/Procedure** | **Document type** | **Document Serial No.** | **Document 3 to 4 letter short form** | **Document Full name** | **Final Label Number** |
|  | XX | 00 | YYY | XXXXXXXXXXXXXXXX |   |
|  | CK | 01 | GNG | GO No GO checklist | CK-01-GNG-GO No GO checklist |
| NST | PO | 01 | IQP | Integrated Quality Policy | NST-PO-01-IQP-Integarted Quality Policy |
| NST | CP | 01 | CAPA | Corrective & Preventive Action Procedure | NST-CP-01-CAPA-Corrective & Preventive Action Procedure |
|  | PR | 01 | TRA | Training Process | PR-10-TRA-Training Process |
|  | TP | 01 | SRS | Software Requirements Specifications | TP-01-SRS-Software Requirements Specifications |
|  |  |  |  |  |  |

* 1. The naming convention for **Assets Labeling** Number Guidelines are as follow:

|  |  |  |
| --- | --- | --- |
| **Category Name** | **Number** | **Final Label Number** |
| **Asset Category** | **Asset Type** |   |   |
| X | XX | X (Incremental) |   |
| H | SR | 1 | HSR1 |

* 1. The naming conventions for **Backups** are as follow:

|  |
| --- |
| **Backup Name** |
| **Automatic** |   |
| **Date (mmddyyyy)** | **Type of Back up** | Server Name | Extension |
| 01022014 | NA | VSS | .bkf |
|  |  |  |  |
| **Manual** |  |
| **Server Name** | **Type of Back up** | **Date** | Extension |
| VSS/TFS | Incremental/Full | 01022014 | .bkf |

### *6.7 Configuration Status Accounting*

The objective of Configuration status accounting is to maintain a continuous record of the status of all baselined items so that there is clear communication of Configuration Management activities and contents of software baselines to the related persons. Accounting is done by maintaining a master list of all baselined CIs.

 The entry for each baselined item shall contain:

* CI Name
* Version No.
* Date Created or Changed
* Status (Current denotes that it is the active version of the CI, Checked Out denotes that the CI is undergoing modification and a new version will be available soon, Obsolete denotes old version of the CI that is not in use.)
* Author or the person who modified
* Path where CI is stored

Ref to TP-36-CSA-Configuration Status Accounting Sheet.xls

The tasks in configuration status accounting are:

* + 1. Make an initial entry when a CI is baselined for the first time. Make sure the date on which the document becomes effective i.e. baselined, should be greater than or equal to the Review Date.
		2. Update when the status of the CI changes after approval based on the change control process.
		3. The Access rights are given to all the team members for the respective code, documents. Ref to TP-37-ACR-Access Right.xlsx
		4. The Master list need not have an individual entry for code files. The location of code where the baselined version exists needs to be mentioned.

### *6.8 Configuration Audit*

Configuration Audits are generally conducted as a part of Internal Audits during which the Configuration Management process for the project is also audited. However if required, the Project Manager may commission special configuration audits over and above the Internal audits. This shall be defined in the Configuration Management Plan.

Several areas are to be audited.

1. All software releases must be controlled to ensure the accuracy of the change implementation and continued integrity of the CI.
2. All updates to baselined documents are audited to ensure accuracy and continued integrity of the documentation set.
3. All approved change requests are audited upon incorporation to ensure accuracy and integrity of the documentation set.

Any audit deficiencies are incorporated in the audit report.

## 7. Outputs

* + Baselined CIs
	+ Configuration management plan
	+ Updated Configuration Status
	+ Updated Change Request Log

## 8. Validation

* Baselined configuration items.
* Approval or reject decision of request after impact analysis.
* Project Closure

## 9. Exit Criteria

* Baselined configuration items.
* Approval or reject decision of request after impact analysis.
* Project Closure

## 10. Related Documents

* Impact Analysis Report template
* Change Request Log template
* VSS/TFS Guidelines
* Configuration Management Accounting Sheet
* Access Rights

## 11. Guidelines for Code Maintenance in TFS

Before sending the request for the check in of files, please perform following step to avoid any compile time error:

1. Perform Get Latest on all files from Code Folder ($/)
2. Perform Clean Solution using Build menu of the Visual Studio.
3. Perform Re-Build Solution using Build menu of the Visual Studio.
4. If error rectify it/escalate it and then again perform above step ii and iii.
5. If no error found on build then send the request/check in your files into Source Control with proper labeling.