# Annex E
## to Tender Specifications

# Code upgrade and Code merging procedures

EMSA

# TABLE OF CONTENTS



# LIST OF FIGURES



# LIST OF TABLES

# 1. Introduction

Given the adoption of an ITIL-based framework for all the project deliveries associated with the ICT operational services of the Agency all the code developers working SSN Ecosystem applications should align their deliveries to the procedures provided as appendices to Annex F to the tender specifications namely:

***Service Transition***

1. Change Evaluation Management (refer to Appendix A of Annex F)
2. Release and Deployment Management (refer to Appendix B of Annex F)
3. Service Verification, Validation and Testing (refer to Appendix C of Annex F)
4. Service Asset and Configuration Management (refer to Appendix D of Annex F)

***Service Operation***

1. Event & Incident Management (refer to Appendix E of Annex F)
2. Problem Management (refer to Appendix F of Annex F)

ITIL (see figure below) is considered as an approach of "best practices", which means that has been developed based on the experience of experts from a large number of organizations. An approach like this, carries a great number of benefits like having a starting point, pointing guidelines, or providing a common vision and language, etc. Basically, it leads to a better quality of service, improving the moral of the team as well as implementation times, improving business support, quality and costs information together with raising the decision capacity along with reducing licence and support costs.

While ITIL promotes quality IT services, it also emphasis on the need of an efficient resource usage and that the delivered services fulfil the business requirements.

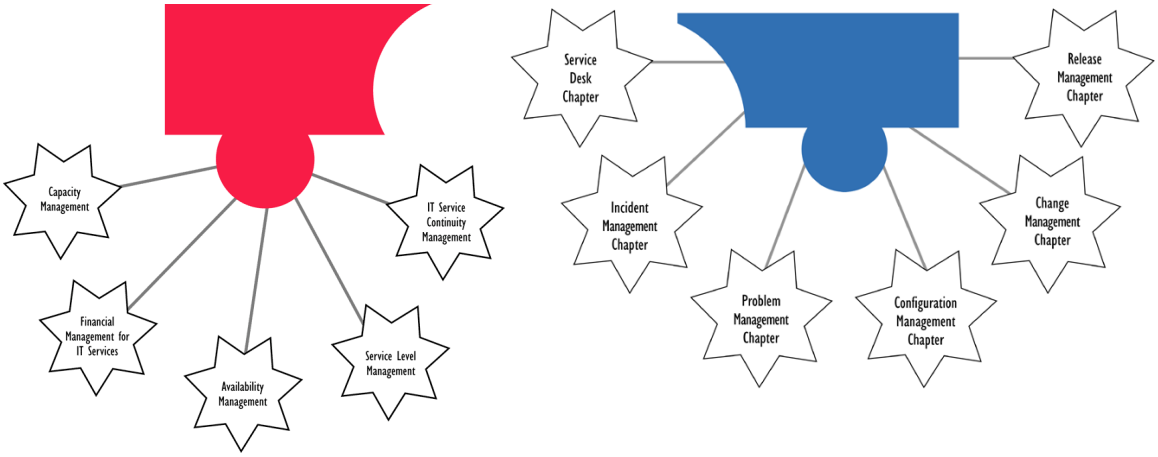


Figure 1: ITIL processes

## 1.1. SACM goals

SACM is a process of managing changes in hardware, software, documentation, etc. (plan, do, act, control). The objectives of Configuration Management are to define and control the components of a service and infrastructure and maintain accurate configuration information:

- Identification

  The configuration management must ensure to identify the current configuration baseline (say, document based), and the as-built configuration (say product based) are known, in order to deal with discrepancies detected during production, delivery or operation of the product. That is the same to be able to know at any time the technical description of a product using approved documentation;

- Control

  The configuration management process must be able to track the modification/evolution of the system baseline and associated technical description.

SACM includes the organization, implementation and follow up of the following tasks:

- Configuration Identification

  Configuration items should be selected by decomposing the service and infrastructure according to agreed criteria. This should divide the total configuration structure into logically related and subordinate groups of hardware, software or combination of them. Controls items may be intermediate and final outputs of: Software (source code, configuration files, database scripts, third party libraries, software design artifacts) COTS, Documents (Test Cases, Test Reports, Test Plans, Use Cases, User Manuals), Release (the executable files produced)

- Configuration Control

  Ensures that only authorized and identifiable configuration items are accepted from receipt through to disposal. Items should be added, modified, replaced or removed with appropriate controls e.g. version control, approved change request, service request or other similar documentation).

- Configuration status and accounting reporting

  Status accounting should capture, correlate, store, maintain and provide views of the assets and configurations.

- Design Review

  The configuration management process should ensure that assets and configurations that fall within scope of configuration management are identified, conform to their specifications and are documented in sufficient detail to support other processes.

- Configuration Audit and Verification

## 1.2. Assumptions

The following assumptions are of uttermost importance in the scope of the SACM and RDM procedures::

- EMSA will be the ultimate responsible for the Change/Configuration/Release Management processes;
- Configuration Management Manager (CM Manager):
  - EMSA will nominate one of his employees for the role of SSN EIS CM manager;
  - Each contractor will nominate their own CM Manager that will interact with EIS CM Manager;
- Configuration Management Infra-Structure:
  - Will be located at EMSA premises or will be supported by EMSA-supplied (and managed) hardware and software;
  - EMSA will provide access to required tools and repositories to all involved companies:
    - All the contractors involved in a software delivery contract which is on-going at the time of launch or evaluation of this tender
    - [Any new contractor awarded work related to any of the SSN Ecosystem applications under a new framework contract, including the one to be established based on this tender]

- Tools to be used:
  - CollabNet TeamForge:
    - To track issues and problems;
  - Maven (http://maven.apache.org/):
    - Build & Test;
  - Apache Subversion (SVN):
    - Software versioning and revision control for:
      - Source Code;
      - Documentation;
      - Maven POMs, Build & Test scripts;
      - Libraries and other required files;
    - Due to licensing issues the SVN repository to be used will not be the one provided by TeamForge tool;
  - Oracle JDeveloper:
    - Integrated Development Environment;
- Archiva for artifacts management (e.g. libraries, COTS, …)
- Setup of initial SVN repository:
  - Under the responsibility of EMSA;
  - SafeSeaNet components to be under Configuration Management:
    - EIS and Central databases (COD, CSD, CLD) components;
    - STIRES (key components of this application are known as SSN GI and SSN SI) until to be deprecated and STAR/ SEG;
    - “VMS Integration” and BlueBelt components;
    - Accident/Incident Module;
  - For the above components, has to include all of the below listed data:
    - Source Code;
    - Documentation;
    - Maven POMs, Build & Test scripts;
    - Libraries and other required files;
  - The division of the EIS and STIRES components into more granulated packages and modules will be performed by EMSA with the support of EX1 and EX3;
- Build of SafeSeaNet’s versions/patches will be performed by EMSA, based on the source code and build & test scripts provided by Contractors;
- Patches (either “Normal Patches” or “Emergency Patches”) for SSN software shall be addressed the same way that those new modules to be developed under a new service contract (including those contracts to be awarded to the successful bidder of this tender)
- Regression Tests required for new SafeSeaNet’s versions/patches will be performed by EMSA or a contractor chosen by EMSA;
- In order for the contractors to setup the necessary development and test environments in their own premises, EMSA will provide to them:
  - Copies of the required Virtual Machines files (VMs);

- Instructions about the setup and configuration of those VMs;
- Instructions about the setup and configuration of the involved SafeSeaNet components;
- Data (test and/or real data);

## 1.3. Repositories Structure for SACM

The high-level SVN project repository structure is presented in the following table:

Table 1 Project Repository structure

| Directory | Folder | Sub-Folder | Sub-Sub-Folder | Description |
|---|---|---|---|---|
| SafeSeaNet Teamforge Project | Management | Schedules | | Project scheduling plans and charts |
| | External Documents | Current baseline | | Current contractual baseline documents |
| | | History | | Previous versions of contractual documents |
| | Internal Documents | {Class of documents} | | Current project documentation, divide according to the class of documents as defined by EMSA and respective history folders. |
| | Records | Reviews | | Project reviews forms and records |
| | | Audit Reports | | Self-explanatory |
| | | Correspondence | | Formal received /sent correspondence and control register |
| | | Progress Reports | | Self-explanatory |
| | | Metrics | | Management and SW metrics database |
| SafeSeaNet Code Main Trunk | Code | {Package id} | {Module id} | The set of source code files, build & test scripts and other required files/libraries necessary for creating the target software |
| SafeSeaNet Code Tags | {Version id} | {Package id} | | The set of SVN soft links to the appropriate package modules |
| SafeSeaNet Code Branches | {feature id} | {Package id} | {Module id} | The set of source code files, build & test scripts and other required files/libraries necessary for implementing the required feature/patch |

## 1.4. CM activities – division of responsibility

The following figure (refer also to section 2.2) depicts the main Configuration Management activities, with the following colour coding:

- With a Blue Background: Activities to be performed by EMSA;
- With a Red Background: Activities to be performed by contractors;

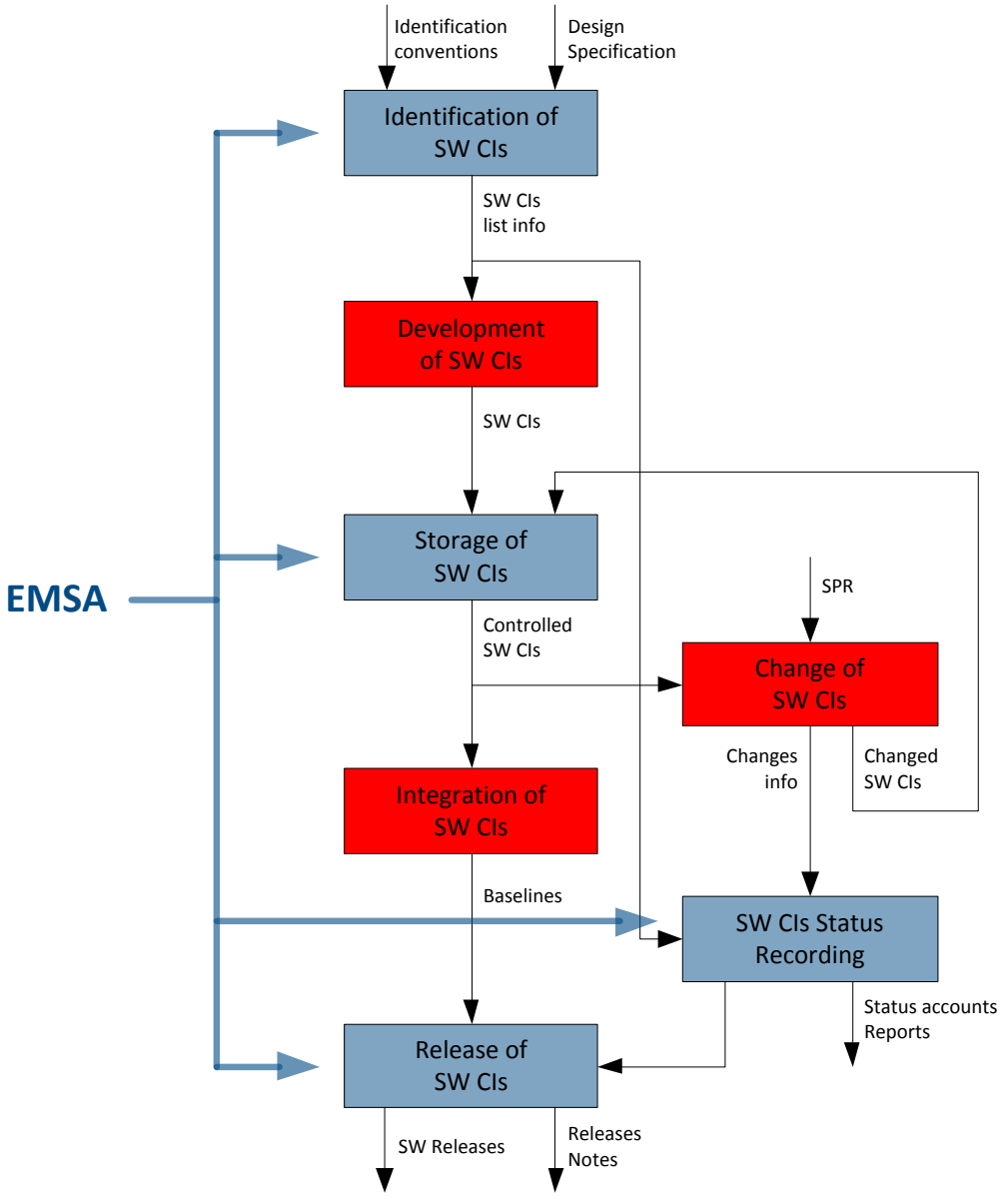


Figure 2: Configuration Management Activities

## 1.5. Backup and Archive

The CM repositories are all backed up periodically according to project standard procedures. After the end of the project the full content of the repositories will be archived in optical support and will be preserved according to EMSA's quality management system.

### 1.5.1. Records Collection and Retention

The Configuration Management records will be collected by the SSN EIS CM Manager and retained for a period of 1 year after the completion of the project (starting at the end of warranty period).

### 1.5.2. Software Archive, Backup and Retrieval

As far as Software is concerned, CM is automatically maintained by SVN and queries can be made through any standard SVN client interface.

The SVN infrastructure will be secured via the regular SVN security mechanisms based in the network identification of the users and the repositories will be backup according with the regular EMSA backup policy.

The regular backup policy of EMSA should follow a Grandfather-Father-Son rotation scheme where daily incremental backups and bi-weekly full backups are made. One full backup per month is kept for three months; the other full backups are rotated on a four week basis. Additionally, one full backup is sent, every three months, to a physically independent archive outside of EMSA premises in Portugal, in order to enhance the protection of the data.

The evolution state and multiples version of each SCI and system versions will be managed by EMSA (the CM Manager) using the regular SVN mechanisms.

All records will be kept for 2 years after release.

The archive of software configuration items shall contain the following metadata:

- Date
- Name of person doing the archive
- Shall be placed in a dedicated area of the project
- Summary description of the archive content
- Discriminate whether it is development or target platform
- List directories and files archived

The backups shall be performed as already mentioned and stored taking into consideration the following:

- Date
- Name of person doing the backup
- Shall be placed in a dedicated area of the project
- Summary description of the backup content
- Discriminate whether it is development or target platform
- Record in the Backup list this action

Retrieval must be done in a temporary area and verified in terms of integrity and correctness of the archive prior to overwrite (if that is the final purpose) into the project folder.

Please note that any of these operations, in particular the back-up, can only be done in a "compilable" software code. Exceptions must be duly justified and approved in advance by the SCM.

# 2. Configuration Management in the event of code merging from different contractors

**Note:**
Referring to the procedure in section 5.4 (point 5.4.2.0) of the SACM (refer to Appendix D in Annex F), the text below aims in clarifying the procedures to be followed in those rather rare cases where code merging by different contractors is to be required.

## 2.1. CM Organization

To facilitate decision making in case of situations where code merging for code delivered by different contractors is required, the following management organization shall be established and followed for all the projects delivering code for the SSN Ecosystem.

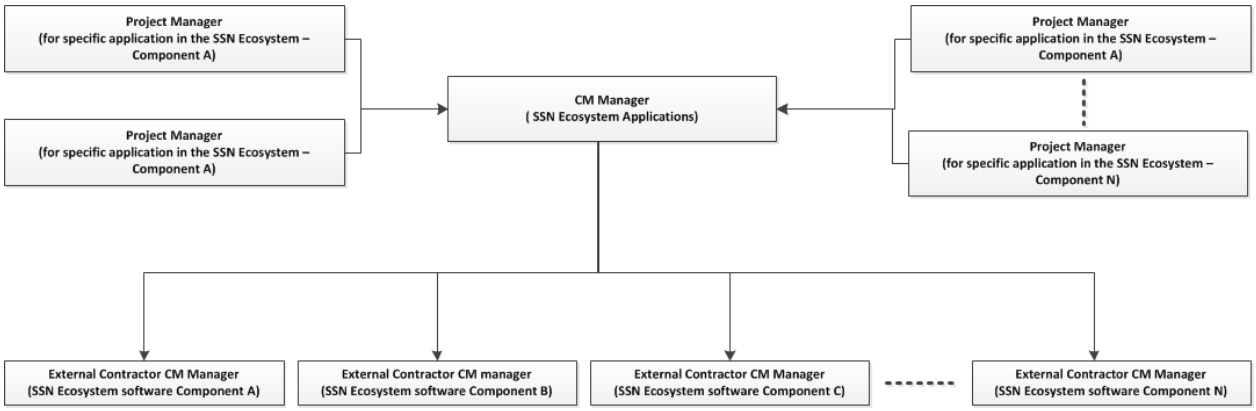


Figure 3: CM Organization

A Configuration Control Board (CCB) shall be established (meeting usually via Teleconference, if a need will arise). It comprises the EIS CM manager (for the SSN Ecosystem EIS applications), the CM managers of EMSA for other applications of the SSN Ecosystem (if required), CM managers of each contractor delivering code and the project managers of the contracts implementing components of the SSN Ecosystem.
The CCB secretariat and overall coordination will be under the EMSA responsibility, and will be vested in the Configuration Management Manager (CM Manager for the application primarily affected by the change) responsibility.

CCB will be organised into several “virtual” teams, each one related to a software modules of the SSN Ecosystem. Each team (occasionally meeting usually via Teleconference, if a need will arise) shall be led by the EMSA CM manager and will include, e.g. in case of the “Module A” team:

1. The CM managers of the contractor tasked to implement a software upgrade or patch to the code for the Module A
2. The CM manager of the company that previously delivered the code for which is be upgraded/ patched.
3. (if the contracted upgrade/ patch shall create impacts to other modules interfaced with module A) The CM managers of the companies responsible for the code of otter modules interfaced or sharing code base with the module A

The team members of a team within the CCB will be responsible to address and resolve issues that may arise in the code merging process

## 2.2. Branching and tagging Strategy

Software libraries will be established (by EMSA) for the Software CI (SCI) in SVN. See figure 5 for the proposed structure.

The following rules will be applied for Branching and Tagging associated with software delivered under this contract:

- Tagging and Branching will be managed by EIS CM Manager;
- Tagging fall solely under the responsibility of EIS CM Manager ;
- The stable, production-ready code and related libraries, built & test scripts will be placed in the “Main Trunk” (SafeSeaNet Code Main Trunk\Code);

The steps to be performed for the implementation of a feature or a patch to existing CIs:

1. Whenever a new branch is required by a Contractor, EMSA shall provide the branch name to be used;
2. Contractor checks out a copy of “Main Trunk” module that needs to be changed/affected for every change and creates a branch for it;
3. Contractor works locally: contractor performs development and unit testing of feature/patch;
4. After finalizing his local work (Step 3), Contractor will agree with EIS CM Manager the strategy and timing for merging:
    a. If necessary, “Rebase to Branch” if “Main Trunk” has been changed. If this step is performed, Contractor will have to perform new Unit Tests on the new merged version (in other words, go back to Step 3);
    b. “Merge from Branch to Main Trunk”;
5. The integration of the “branched code” back into the “Main Trunk” will fall under the responsibility of the Contractors;
6. The build and regression tests for the new “Main Trunk” fall under the responsibility of EMSA;
7. The establishment of a release version for the pre-Production and Production environments will fall under the responsibility of EMSA;

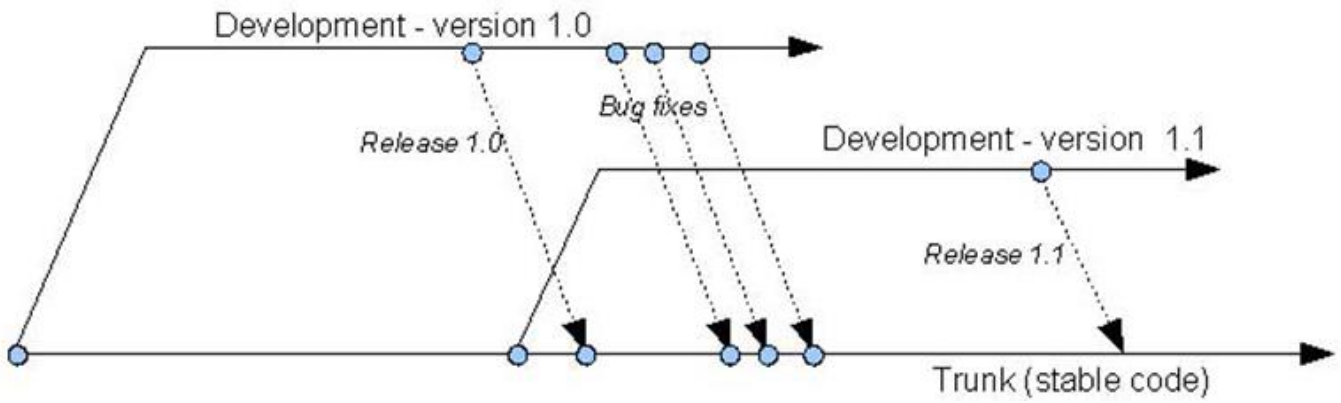


Figure 4 Branching and Tagging using SVN: Overall Policy (1 of 3)

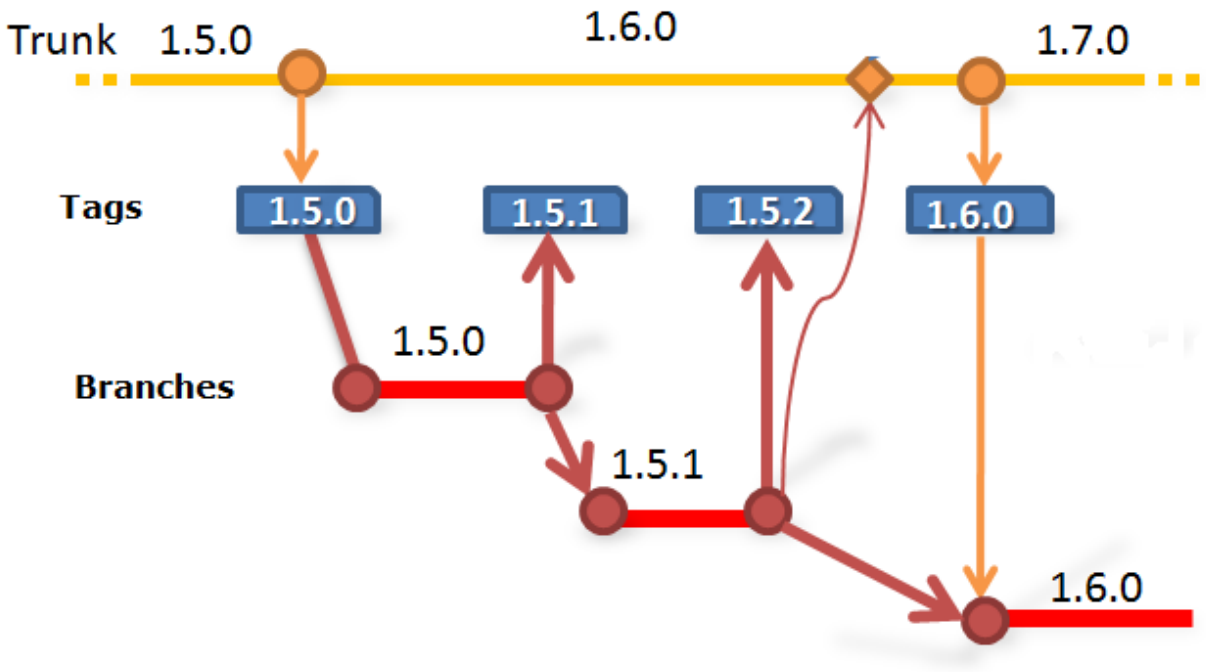


Figure 5 Branching and Tagging using SVN: Overall Policy (2 of 3)


Figure 6 Branching and Tagging using SVN: Overall Policy (3 of 3)