1 Configuration Management Strategy

The purpose of the configuration management process is to control and monitor the system elements and configuration during the life cycle. This section describes the projects configuration management strategy that is used to identify how the project's products will be controlled and protected [1]. The strategy is strongly inspired by the configuration management procedure outlined by PRINCE2 Agile [2].

1.1 Planning

Configuration management is a collection of all the activities that maintain and control changes for each product throughout the life cycle of the project and after the project is completed [2].

CrisBag(r) is the product delivered to the customer, which consists of several critical functional components. Each of these components are listed below:

```
Main components:
    Tote
    Transport
    Curve
    90 Transfer
    Divert
    Merge
    Top-load
    Side-load
    Discharge
    Vertical Sort
    Walk-through
    Stacker
```

The list of components above will be referred to as 'main components' and must all be versioned and labelled after a specific identification system, which is described in the next section. Sub-systems/sub-modules of each main component will not be forced to follow same versioning/labelling. Updating a baseline for a main component can only be done by following the change procedure described in section **1.3 Change Management**. If there is a change in a sub-module in a main component it will only be 'persisted / approved' if it follows previous mentioned procedure.

Most of the main components are delivered by Beumer or 3rd parties and will most likely not be up for revision during the systems life cycle. The products / solutions that are prone to change during the system's life cycle is the mechanical solution (and associated software solution). These solutions must also be versioned and labelled and will be explained in section **1.4 Status Accounting**. All major components/systems can be seen in figure 1.1

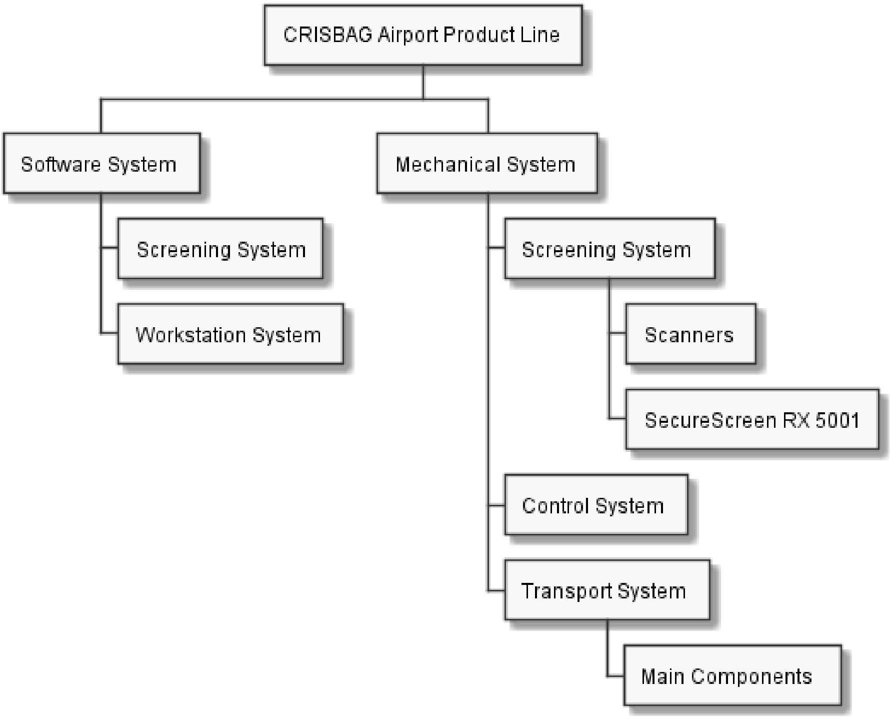


Figure 1.1: Components and subsystems that must follow the configuration management procedure. Components are also referenced as configuration items.

1.2 Identification

This activity gives the proper identification of configuration items (CIs) to be maintained under configuration control. The activity covers the identification of such items as well as defining unique identifiers for the CIs. This also includes establishing baselines for the CIs as well as agreements of the baselines by the acquirer and supplier.

Identification of CIs
The CIs of this project are mechanical modules (including hardware items), software modules, communication interfaces and project documentation. The main components of this system mentioned in **1.1 Planning** are the CIs to be tracked. The main components can be divided into additional sub products for which the identification can be specified if needed.

Unique identifiers
The purpose of the unique identifiers for each CI is to maintain version control to support and track the baselines of each product.

Identification of a product is done by naming the product by the following unique identifier:

```
<Product Code>
<Sub Product Code>
<Initials Of Owner>
<Version Number>
<Latest Modified Date>
```

The documentation of this product would be named:

```
01-00-SE-v01-20211802.pdf
```

The version number is significant for project deliverables and is assigned and changed when needed (e.g. for different iterations). The version number should not be changed for every product modification however the <Latest Modified Date> must always be updated.

The <Product Code> must indicate the relationship between the products (e.g. naming the adjacent products with adjacent or nearby numbers).

Establishing baselines
The baselines of the products are to be held as flexible as possible for as long as possible, but are to be approved by the acquirer and the supplier according to both the requirements and the associated designs.

A design baseline is described in terms of physical constraints and additional requirements (R1-R12, see case description [3]) including security and outsourcing of several system blocks. The design of the outsourcing is incomplete and is to be defined but are limited to 5 cabinets to hold electronics only.

1.3 Change Management

During the project it is unavoidable that multiple changes might be needed along the way. For this reason it is important that changes happen in a controlled and systematic way to avoid the change to effect cost, schedule etc. In this section a process will be described for changes to the individual components that are described in section 1.2. The change process begins with a ECP (Engineering Change Proposal) that is the baseline for a change. It is important for this ECP to have been classified for the impact it might have. The change should also be approved by a CCB (Configutration Control Board) that includes different stakeholders, which also includes the customer. When the ECP is approved by the CCB the change can be implemented. When the change has been implemented a change request must be submitted to merge it into the previous baseline. The steps of the process is illustrated on figure 1.2.

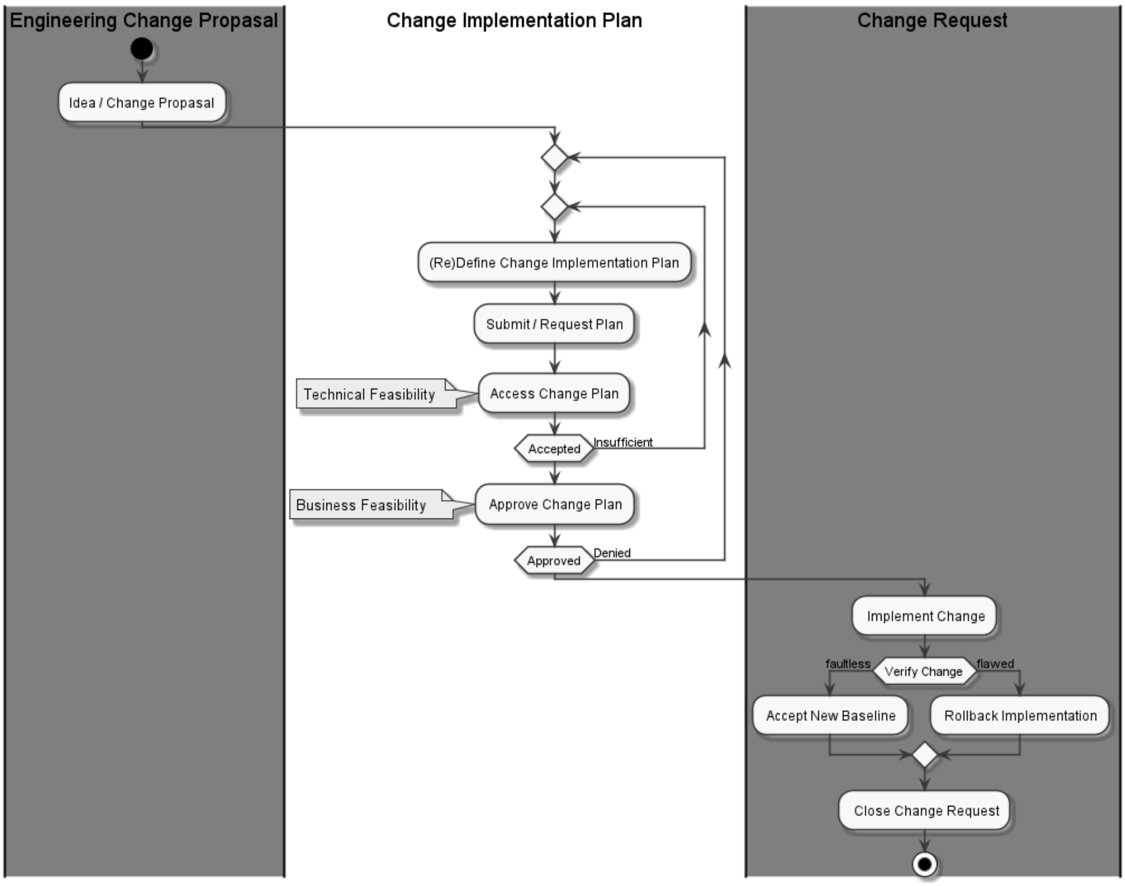
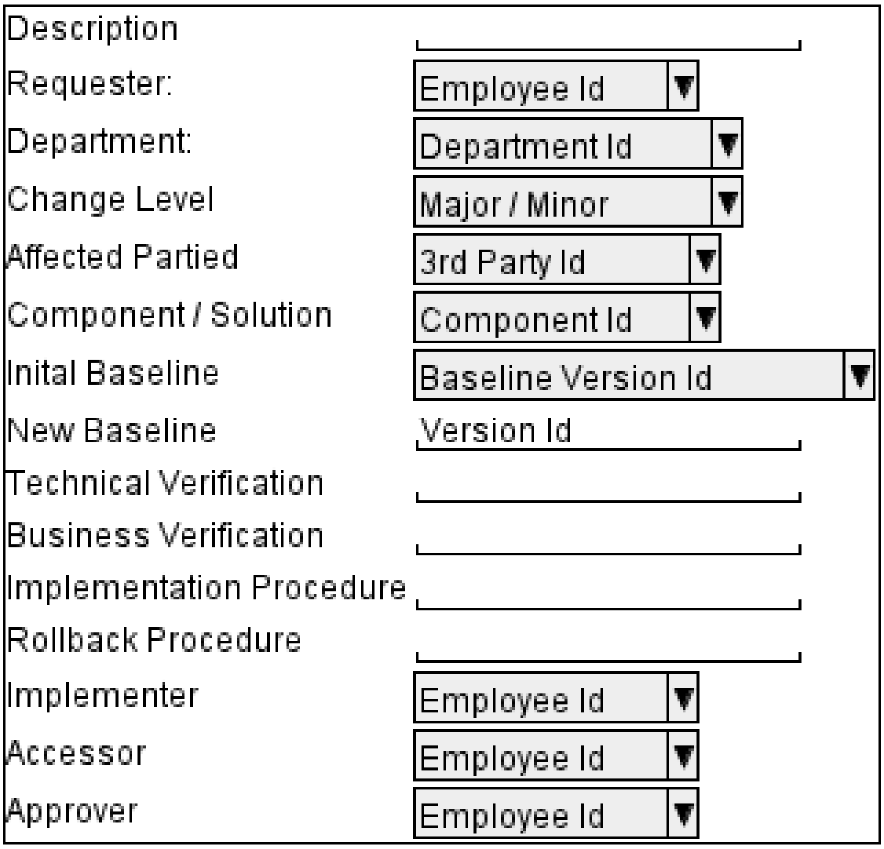


Figure 1.2: Change Management Plan

The first step after implementation is to create a 'Change Request' that is the new baseline for a product. Figure 1.3 illustrates what a change request must contain.



Description	
Requester:	Employee Id ▼
Department:	Department Id ▼
Change Level	Major / Minor ▼
Affected Partied	3rd Party Id ▼
Component / Solution	Component Id ▼
Inital Baseline	Baseline Version Id ▼
New Baseline	Version Id
Technical Verification	
Business Verification	
Implementation Procedure	
Rollback Procedure	
Implementer	Employee Id ▼
Accessor	Employee Id ▼
Approver	Employee Id ▼

Figure 1.3: Graphical representation of change request. Non-dropdown elements should contain a detailed description of the specific task, e.g. 'Technical Verification' should describe how the change is verified after implementation.

The next step of the process is to send the 'Change Request' to a verification (if it is interactive component, e.g. a part of the mechanical solution, it should be verified in its production environment). This could be done by one or multiple people from the same team or department and do not have to include the management or stakeholders. The next step is to have the change verified by manager or stakeholders (e.g. customer). Only after these steps the changes gets a new version number and replaces an older version. The new version should be verified by managers and stakeholders to make sure that the change is correct. It is possible to roll back the change after this. In this case a new 'Change Request' should be created and the steps should be followed from the beginning with a new version being the result of the change.

1.4 Status Accounting

This activity is the process of reporting all activity related to the CIs of this system in all life cycle stages. The report shows the current and historical data for each product including all changes and the cause of change when renewing the version of a product.

Status accounting for this project includes the following status information for each product in the system:

```
Identifier
Version
Latest modification
Cause of Change
Current status
Initials of Owner
Date of next baseline
```

The products included in the status accounting system is the components identified in **1.2 Identification**. This covers all changes of versions which are to be recorded in the status accounting system.

The reports should be documents available to any stakeholder during all life cycle stages and should be in a format such as .docx or .pdf depending on who is requesting the report.

All additions must be approved by the configuration manager, to be sure that they have reviewed the correct protocol, cf. change management.

1.5 Verification and Audit

Verification and audit is the act of ensuring that the configuration management procedure outlined in the change management section is followed. There is dedicated two steps in the change management to facilitate this: ’Access’ and ’Approve’. The ’assessor’ is usually a group of engineers from the same team or department that ensures that the change implementation plan is technical feasible and ’approver’ should be a group of people with extensive business domain knowledge and experienced in change management verification and auditing (CCB).

Verification and audit management also includes the alignment of the project with stakeholders, consumers and customers. Below are some examples of elements that must be considered:

- Do the customer / stakeholder have access to the correct product versions?
- Change request aligned with end goal, e.g. is new baseline aligned with business goals?