Configuration Management implementation on industrial environments

Jaime Larumbe Chacón

University of Deusto

This dissertation is submitted for the degree of

Doctor of Philosophy

November 2020

Dissertation written by

Jaime Larumbe Chacón

Approved by

Dr. Diego López de Ipiña González de Artaza, Thesis Director

Dr. Jon García Barruetabeña, Thesis Director

A Dios por haberme dado unos padres maravillosos.

مع النداء الأول للصلاة في عزلة الصحراء ، يداعبني نسيم الصباح اللطيف

بنظرة الزمرد

Abstract

Configuration Management [CM] is a Systems Engineering process for establishing and maintaining consistency of a product´s performance, functional, and physical attributes with its requirements, design, and operational information throughout its life. CM is the practice of controlling changes methodically so that a system keeps its traceability over time. CM programs and plans provide technical and administrative direction to the development and implementation of the procedures, functions, services, tools, processes, and resources required to successfully develop and support a complex system. During system development, CM allows program management to track requirements throughout the life-cycle through acceptance and operations and maintenance. Ideally, the CM process is applied throughout the system lifecycle.

Currently CM is widely applied in the most engineered complex and competitive products of the word from its inception until the disposal such as space crafts, satellites, information systems, planes, trains and vehicles, apart from that, in the civil engineering and other industrial engineering segments such as roads, bridges, canals, dams, and buildings.

However, nor a scientific method neither systematically structured knowledge is followed in its application. The costs associated with a proper CM can be avoided by not bothering to employ it in an enterprise. If this is done, however, the company will probably pay instead in costs for unexpected failures from mistaken system components, because it was not possible to exactly identify which components were impacted by a change. Therefore, the lack of a proper CM, or its insufficient implementation, can be very expensive and sometimes can have catastrophic consequences such as failure of equipment or loss of life.

Hence, throughout this research a systematically structured procedure for the practical implementation of CM is defined. This is done via the definition of a standardized procedure for identifying, selecting and categorizing the Configuration Items that will lead to the practical implementation of Configuration Management. The term configuration item (CI) has reference with the basic structural unit of a CM system. The CM system commands the life of the CIs through a combination of processes and tools by applying and allowing the fundamental elements of identification, change management, status accounting, and audits. This system avoids the introduction of failures related to lack of testing as well as of conflicts with other CIs.

As a result, the proposed procedure guarantees quality and reproducibility on Configuration Management implementation while minimizing CM manager bias. Its goal is for CIs selection to be conducted in a fair, unbiased and repeatable manner.

Keywords: *Systems Engineering, Configuration Management, Configuration Items, Product Life Cycle, Agile Methodology, Quality*

Preface

Except where otherwise stated in the text, this dissertation is the result of my own work and is not the outcome of work done in collaboration.

This dissertation is not substantially the same as any that I have submitted for a degree or diploma or other qualification at any other university.

This dissertation does not exceed 65,000 words, including appendixes, bibliography, footnotes, tables and equations. It does not contain more than 150 figures.

Jaime Larumbe Chacón
November, 2020

Acknowledgements

I would like to thank the University of Deusto through the person of Diego López de Ipiña and his proactive behavior for giving me the possibility of carrying out a dream that has been this doctorate in the field of Configuration Management.

Continuing in the academic world in chronological order my special thanks to Jon García Barruetabeña. His intelligence and courage arrived when they were most needed.

This PhD is the culmination of a wonderful professional adventure as an engineer of more than 20 years in the industry that gave me the chance of living in remote countries taking part of rewarding cultures and participating in such human teams as:

- GENERAL DYNAMICS-US
- BOMBARDIER-Canada
- AIRBUS-KSA
- MITSUBISHI-Qatar

Publications

Aspects of the work described in this dissertation feature in the following publications:

Jaime Larumbe, “Configuration Items Selection Process Review”. ISSN: 2169-0316 Industrial Engineering & Management 6: 216. doi:10.4172/2169-0316.1000216. Published Date: August 28, 2017

Jaime Larumbe, Jon García Barruetabeña, Diego López De Ipiña Gonzalez De Artaza. "The importance of configuration management on complex industrial environments". DYNA New Technologies 7, no. 1 (2020). DOI: http://dx.doi.org/10.6036/NT9654

Jaime Larumbe, Jon García Barruetabeña, Diego Lopez De Ipiña Gonzalez De Artaza. "METHODOLOGY FOR THE IMPLEMENTATION OF CONFIGURATION MANAGEMENT ON LARGE SCALE PROJECTS". DYNA 96, no. 1 (2021). DOI: http://dx.doi.org/10.6036/9806

Content

List of Figures

List of Tables

Glossary

BRM	Benefit Realization Management
CI	Configuration Item
CM	Configuration Management
CMDB	Configuration Management DataBase
CMMI	Configuration Management Model Integration
CMP	Configuration Management Plan
CSA	Configuration Status Accounting
DCD	Design Control Document
DWO	Deepwater organization
FFF	Fit Form and Function
GoM	Gulf of Mexico
HW	Hardware
ILS	Integrated Logistics Support
IT	Information Technology
ITIL	Information Technology Integration Library
LMX	Leader Member Exchange
LSP	Large Scale Project
NASA	National Aeronautics and Space Administration
NPP	Nuclear Power Plant
PBS	Product Breakdown Structure
PLM	Product Lifecycle Management
PM	Project Management

SE	Systems Engineering
SME	Small and Medium Enterprise
SSC	Standards Systems and Components
SW	Software
USDoD	United States Department of Defense
WBS	Work Breakdown Structure

"Change is the only constant" Heraclitus

Chapter 1

Introduction

One of the first things this chapter will do is to provide general statements that outline the importance of the discipline of Configuration Management and provide enough background information so that the reader can understand the context in which the research sits. In doing so, existing studies will be referenced. The introduction will also discuss the gap that the research will fill. Then, the research questions and the problems the study addresses will be outlined.

The introduction will also discuss the value that this study brings to the broader field or discipline (Project Management and Systems Engineering). This will be done by detailing the central argument, the research aims, the structure of the discussion, the methods employed, the study's limitations and the layout of the thesis.

Apparently, the interaction between the evolution of technology and the development of economy and society has always been an important dimension of human history. This applies to the Iron and Bronze Ages as well as to modern times. The transition from the agricultural society towards the industrial society provides the most pertinent illustration of the profound implications which the full diffusion of new technologies can have on family structures, work relations, settlement patterns, economic and political power configurations, and also on behavior patterns and value systems.

The relationship between technology on the one hand and economy and society on the other is not unidirectional. Not only does technological progress result in the continuous change of economic and social structures, but the latter, including the evolution of attitudes and values, has at the same time a major impact on the direction and the speed of technology

development. The industrial society of today, characterized by mass production, mass consumption and mass government, is in many ways a complex incarnation of the technologies of the 21st century. But there is no doubt that the profound change in political, economic and social structures has provided the conditions to enable the transition to a new paradigm.

Looking at technology developments at the turn of the 21st century, there seems to be once again a broad range of new technical breakthroughs in reach. Further rapid progress is expected in information technology, new materials, genetics technology, environment protection and energy technologies, to name just a few.

New possible combinations and interactions of the various technologies will also be of major importance. Prominent examples include information technology and telecommunications as well as energy and environment technologies. However, only a few of these technologies appear to be pervasive enough or to provide mankind with new basic capacities for them to have a major impact on society.

In a thirty-year perspective, genetics technology as well as energy and environment technologies could hold this potential. But looking ahead towards the next ten years or so, the main driving force for economic and social change will be information technology. After a quarter of a century of gradual development and diffusion, many believe that information technology is on the verge of a new take-off.

This is partly due to genuine technology evolution; however, it is also partly the result of changing economic and social structures. These are increasingly adapting to the new organizational and institutional patterns required for the full and most effective use of the new technology, thereby contributing now to the push for further technological progress.

Everything in industry involves change. Product design, development, integration, test, production, deployment, delivery, maintenance, support and obsolescence are all manifestations of change process. So, CM, the process of managing changes, takes on a new light. CM embraces change as an opportunity for continuous growth and learning.

1.1 Definition of Configuration Management

Configuration management (CM) is a Systems Engineering (SE) process commonly applied in multidisciplinary large-scale engineering projects of complex systems. It is used for establishing and maintaining consistency of a product´s performance, functional, and physical attributes with its requirements, design, and operational information throughout its life (Aiello & Sachs, 2010).

In other words, CM is the practice of handling changes systematically, so that a system maintains its integrity over time. CM implements the policies, procedures, techniques, and tools that manage, evaluate proposed changes, track the status of changes, and maintain an inventory of system and support documents as the system changes (Hass, 2003).

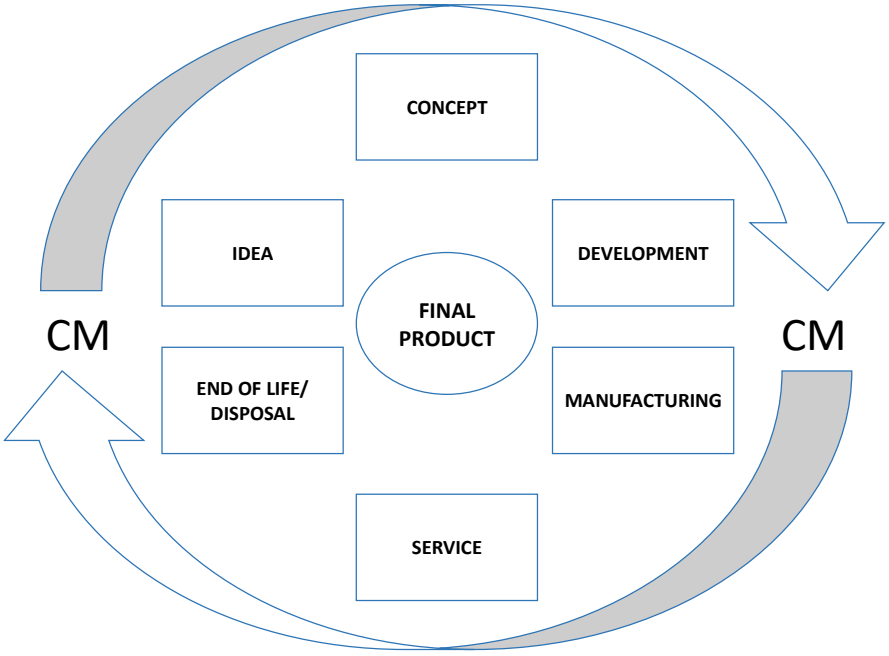


Figure 1.1. All product related information and activities integrated by CM.

Figure 1.1 depicts how CM integrates all the information and development, manufacturing, service and disposal activities along the product life cycle from the initial idea until end of life of the product. In addition, CM programs and plans constitute a technical and an administrative tool that contributes to the development and implementation of procedures, functions, services, tools, processes, and resources required to successfully design,

development, manufacture, service and maintenance of a complex engineering system. Therefore, during system development, CM allows program management to track the initial requirements throughout all the changes and modifications during the life-cycle from the acceptance trough manufacturing and maintenance. As changes inevitably occur not only in the requirements, but also in design, it is of upmost importance that they are approved and properly documented.

Hence, changes traceability is guaranteed, creating an accurate record of the current system status. Hence, the CM strategy should be applied throughout the system lifecycle (Quigley & Robertson, 2015). The CM process is widely used by military engineering organizations to manage changes throughout the system lifecycle of complex military systems, such as weapon systems, vehicles, and information systems (Wateridge, 1999).

Outside the military, the CM process is also used with IT service management as defined by ITIL (Information Technology Infrastructure Library). Likewise, it is also applied in civil engineering and other industrial engineering segments such as cars, planes, trains roads, bridges, canals, dams, and buildings (Watts, 2015).

1.2 The importance of Configuration Management

However, nowadays the application of CM is somehow the combined outcome of experience and company knowledge and no scientific method neither systematically structured knowledge is followed in its application (Ali & Kidd, 2013). The consequences of this lack of organized procedures for CM application in complex engineering projects leads to several problems with relevant economic consequences.

As an example of lack of CM, the current BOEING 737 MAX pair of accidents which claimed 346 lives (Bronk, 2020). It may be somewhat counter-intuitive, but the more efficient LEAP engines used on the 737 MAX are much larger than the engines used on previous versions of the aircraft. To make these significantly larger engines fit on the wing, they had to be mounted not only higher but farther forward than previous generations of the 737. It was found that this new positioning of the engines caused the 737 MAX to pitch up slightly during certain manoeuvres, especially when the aircraft was already at a high

angle-of-attack (AoA). In other words, when the nose of the aircraft was raised to gain altitude, the plane would start to climb higher than the pilot intended. If left unchecked, this tendency could potentially lead to a disastrous stall condition; where the aircraft has pitched up so far that it's no longer able to produce lift.

To counteract this quirk of the design, the MCAS system was introduced (Matthews & Choi, 2019). Put simply, MCAS detects when the 737 MAX is at risk of this pitch-up tendency, and compensates by using the aircraft's rear stabilizer to bring the nose back down. When operating as intended the pilot shouldn't even know that MCAS was engaged. It was designed to be a system that operated in the background, automatically providing the pilot with the ideal aircraft performance.

In fact, it was since been revealed that 737 MAX operators were not informed about MCAS, or trained on its operation. Documentation detailing the changes made between the two generations of aircraft didn't mention the automatic system, as Boeing believed it wasn't something pilots would need to be consciously aware of. As such, many pilots didn't learn about MCAS until the first fatal accident had already occurred.

Another example, the issue that occurred to Toyota car manufacturer with the gas pedal, can be mentioned (Finch, 2009). In this case, Toyota changed during the development process, the floors mats without considering the impact in other components, in this case the gas pedal. As a consequence, once that the car was in service, the floor mats could jam the gas pedal down.

Although the jam of the blocking of the gas pedal when driving is an extremely serious situation, fortunately there was no regret for human losses. However, this lack of CM caused to the Japanese firm not only millionaire losses but a huge loss of credibility. As a result, there was a recall of 4.2 million on Toyota and Lexus vehicles with very important costs due to the replacements (Finch, 2009). On one hand, there was a cost of spare parts without labor cost of approximately 120 million of dollars. On the other hand, there was a cost of claims and legal charges of approximately 1.7 billion of dollars.

In this context, the pace of adding software-driven system and driver features has been accelerating rapidly. Carmakers are using software controlled engine management to

improve gas mileage and reduce emissions, replacing bulky mechanical systems by “x by wire” software, and introducing an array of active safety functionality enabled by highly sophisticated software. There is no official source of statistics about software-driven failures and vehicle recalls, or design defects that can be corrected strictly via software update, such as the now famous software patch Tesla applied remotely to increase ground clearance in order to prevent incidents that have caused Tesla cars to catch fire (Ivory, 2014).

Therefore, identifying and categorizing each and every part that composes the final product, as well as, the relationship between them is of key importance. This information should be provided by CM through the Configuration Items (CI).

1.3 Definition of Configuration items

The term CI refers to the fundamental structural unit of a CM system (J. Larumbe, 2017). The CM system oversees the life of the CIs through a combination of processes and tools by implementing and enabling the elements for identification, change management, status accounting, and audits. Figure 1.2 shows that the CM process for any hardware and software configuration items comprises five distinct disciplines: planning, identification, change control, status accounting, and audit. (See Figure 1.2)

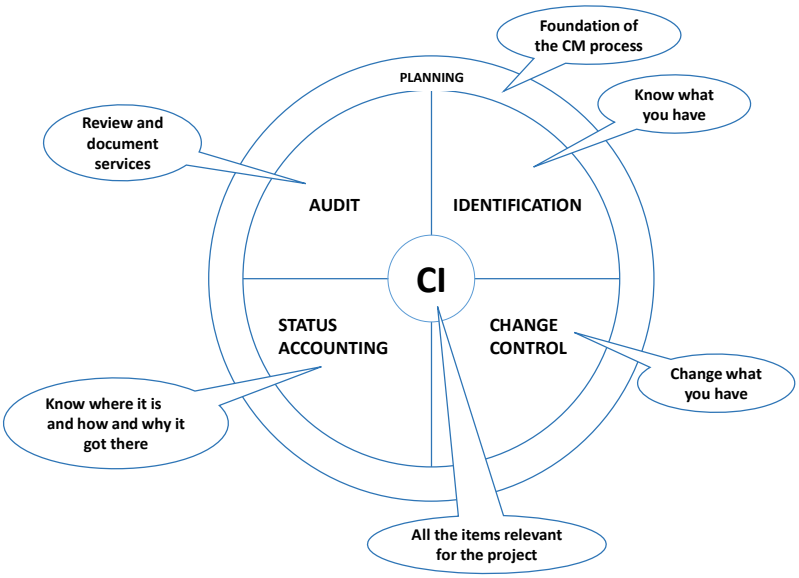


Figure 1.2. Configuration Management-Overview

As seen in Figure 1.2, CIs can be considered as the Gordian knot of all this five disciplines composing the CM. CIs are the basic units of configuration management. They may vary widely in complexity, size and type, from an aircraft, ship, tank, electronic system or software program to a test meter or a round of ammunition. Regardless of form, size or complexity, the configuration of a CI is documented and controlled. CI selection separates system components into identifiable sub-sets for the purpose of managing further development. For each CI:

- There will be associated configuration documentation (which may range from a performance specification to a detailed drawing to a commercial item description.
- Configuration changes will be controlled.
- Configuration status accounting records will be maintained.
- Configuration audits will be conducted to verify performance and product configuration (unless the CI has an already established product baseline).

1.4 Definition of Baseline

On the other hand, in configuration management, a "baseline" is an agreed description of the attributes of a product, at a particular date, which serves as a basis for tracing changes (J. Larumbe, 2017). Hence, an `as required baseline´ is a snapshot in time that represents

the agreed-upon, reviewed, and approved set of requirements committed to a specific product release.

Accordingly, an `as designed baseline´ is a snapshot in time that represents the agreed-upon, reviewed, and approved items identification describing functional and physical characteristics, interface requirements, design constraints and verification required committed to a specific product release. Without this information, it is impossible to ensure that the impact of a change has been taken into account, as in the case of the example where neither the design constrains nor the interface requirements were properly identified.

Therefore, Boeing nor Toyota were not able to identify the impact that the engines would have with the stability system nor floor mats would have with the gas pedal respectively.

1.5 Statement

As a conclusion, it is necessary to develop a CM application methodology that allows minimizing the risk in management of large and complex engineering projects by ensuring the complete identification of the impact of all changes as well as the traceability of the requirements. Furthermore, this work attempts to complement the development of a scientific methodology with an increasing difficulty couple of real case studies in industry in order to validate the following statement:

"It is possible to define a scientific method to select the highest priority CIs to establish a procedured CM regardless the sector and the complexity of the product in order to provide testable explanations and predictions about the final product".

1.6 Scope

Therefore, this dissertation presents a systematic procedure to build and organize knowledge in the form of testable explanations and predictions about the final product. Selection and identification of CI for a particular project can be seen as the first step in developing an overall architecture of the product from the top-down.

This procedure is the result of over 10 years of experience of the first author in the domain of Configuration Management in some of the most ambitious projects and for some of the most competitive multinationals of the world like the Doha metro project for MITSUBISHI, KSA NBS Programme (Kingdom of Saudi Arabia Northern Border Security) for AIRBUS, TWINDEXX Swiss Express for SBB (Swiss Federal Railways) for BOMBARDIER, and the SCOUT SV Programme (Family of Armoured Fighting Vehicles for the British Army) for GENERAL DYNAMICS.

1.7 Outline

The outline is as follows:

Chapter 2 explains the background of CM through its methods, tools and technics.

Chapter 3 overviews the state-of-the-art in CM. Firstly, this survey identifies how this discipline increases the general consistency and as a result, the success of large scale projects. Secondly, the study will identify those gaps that CM is able to cover in any LSP.

Chapter 4 presents the lack of a scientific methodology in the selection of the CIs as-yet unresolved issue in the application of CM. This chapter describes the scientific method to be applied on the selection of the CIs to solve the lack of a scientific method to select the CIs.

Chapter 5 describes the validation procedure. A real industrial example will be used to validate this method with a real configuration management case. More concretely, one of the most important subsystems and with the highest added value of the Doha last generation metro construction project will be used.

Chapter 6 presents a summary of the main contributions of this research and suggests potential future work. The benefits of a systematically structured procedure for the practical implementation of Configuration Management will be described.

Chapter 2

Background

The background of the study will provide context to the information discussed throughout the dissertation. Background information will include both important and relevant studies. In addition, the background of the study will discuss the problem statement, rationale, and research questions. It links introduction to the research topic and ensures a logical flow of ideas.

Thus, it helps readers understand the reasons for conducting the study. The purpose of the dissertation background/history section is to give the reader the relevant facts about the discipline of Configuration Management, so that they understand the material and how it links to the theoretical question. It aims to contextualize the study and to explain its relevance. The background chapter will explain how this work adds to and builds upon existing academic studies.

Configuration Management is the science arisen in large-scale projects management to turn its spotlight from the final product to the configuration items and as a result grow the possibility of holistic success. The expression configuration item applies to the basic entity that conforms a configuration management approach. The configuration-management approach supervises the configuration items along the life cycle via the following processes:

- Identification.
- Change management.
- Status accounting.
- Audits.

Besides of that, this section will focus on CM techniques and tools that can increase the consistency of large-scale projects delivered in complex environments.

2.1 Configuration Management disciplines

CM is based on a set of high-level disciplines and practically with a set of techniques and tools. The target of this section is to define and concisely present these disciplines and to introduce a selection of all the different techniques and tools. Tools and techniques shown and discussed in this section are those to assist the consistency of projects delivered in complex environments. For more information they are explained with all the necessary details in CM standards. In the next section all these elements will be collected in a holistic view (Schuster *et al*., 1999).

2.1.1 Configuration Management Plan

It has been said that nearly all engineering mishaps, apart from those originated from natural causes or operator mistake, can lately be traced to at least one CM concern. Failures in CM happen in every industry and are frequently splashed across the news headlines all around the word when things go wrong. The role of CM weakness in product failures over the last half century traditionally had an adverse influence well in excess of $30B in today's dollars. Among the more famous examples incorporate the following mishaps:

- Apollo1-1967 (Kauffman, 1999)
- Apollo13-1970 (Chaikin, 1995)
- Solar Max Capture 1980 (Bell, 1980)
- Challenger 1986 (Vaughan, 1989)
- Mars Climate Orbiter 1999 (Sauser et al., 2009)
- Columbia 2003 (Gehman, 2003)
- Equifax Data Breach 2017 (Thomas, n.d.)

The application of CM begins with a well-conceived Configuration Management Plan, often abbreviated as CMP. The CM Plan is usually the first item that program auditors or accident investigators will inspect when they assess the adequacy of a company's CM

application, as well as when seeking participating factors to a product defect or system failure. CM Plan specifies the pattern that the Program Manager (PM) and systems engineer will put into practice to control program documentations and the program baseline (Technical, Functional and Allocated). As necessary, this CMP maybe upgraded regularly to be evidence of the changes in procedures, rules, regulations, and best practices to preserve correct CM practices (Niknam *et al*., 2013).

For the aerospace & defense industry, the CMP pin points how a company will fund and accomplish the PM, SE and other operation requirements mentioned in the contract along the life of the program's CIs for every serial or lot number. This scope of a CMP is far wider than delivery of required data depicted in the:

- Contract Data Requirements List (CDRL), which is a list of authorized data requirements for a specific procurement that forms part of a contract.
- Mission Assurance Requirements (MAR) which are the requirements for a disciplined application of SE, risk management, quality and management principles to achieve success of a design, development, testing, development and operations process.
- Statement of Work (SOW), which defines project-specific activities, deliverables and timelines for a vendor providing services to the client. It spreads to requirements mentioned in the terms and conditions and overall supplying of the contract, as well as those pin pointed in requirements and operation fulfilment specifications (Cantamessa *et al*., 2012).

2.1.2 Identification

Configuration identification increasingly pin-points and preserves the definitive present basis for control and status accounting of a system and its CIs along their lifecycle (development, production, deployment and operational assistance, until demilitarization and disposal). The configuration identification process guarantees that all acquisition and supporting management disciplines have common collections of documentation as the foundation for creating a new system, changing an existing component; buying a product for operational use, and providing assistance for the system and its components.

The configuration identification process also contains identifiers that are shorthand references to items and their documentation (Sciore, 1994). Good configuration control procedures ensure the continuous consistency of the configuration identification (Ali & Kidd, 2015). Mistaken or erroneous configuration identification may cause inadequate products, schedule delays, and higher maintenance costs after dispatch (Rickabaugh, 1994).

2.1.3 Configuration Control

When developing and maintaining a product, changes are unavoidable. People do errors, clients require modifications, and the conditions in which the product runs develops. Besides, people regularly develop their understanding of the problem and their capacity to sort it out. In software development, it's commonly said that the solution of a problem will generate new problems. In other words, we get wiser all the time. Figure 2.1 pinpoints the different steps downwards in the validation and for the change management upwards.

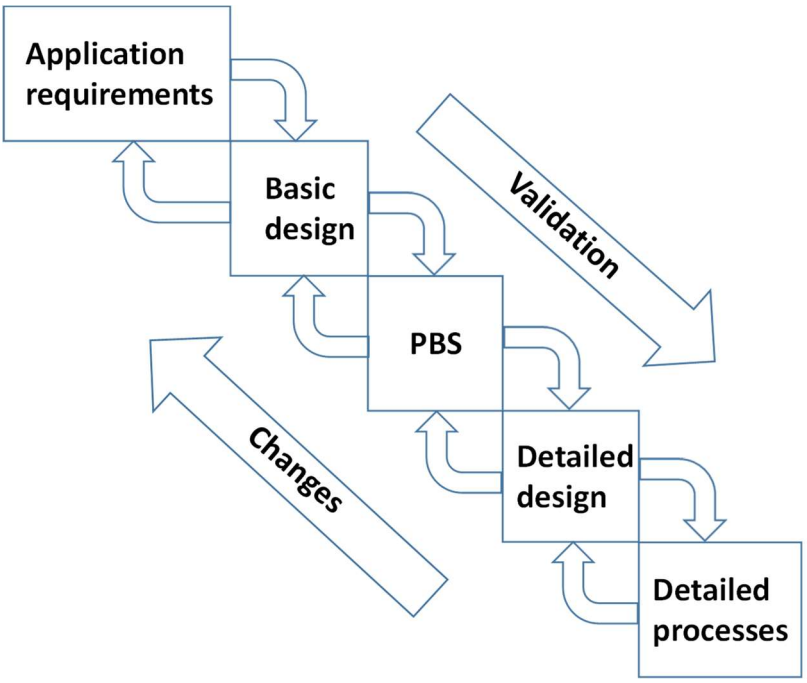


Figure 2.1. Change management waterfall

As can be seen in the Figure 2.1 each time a change arises, configuration management must go upstream in the product development process in order to ensure that the impact of the mentioned change on the whole project is completely known and traced. The aim of change

control is to be completely in control of all change requests for a product and of all implemented changes (Joeris, 1997). For any CI, it must be feasible to identify changes in it relative to its predecessor. Any change should be traceable to the item where the change was applied (Whyte *et al*., 2016).

Table 2.1. Change management phases

Change control phases	Description
Creation of the event registration	The event registration is created and the event is described
Analysis of the event	Configuration item(s) affected by possible changes are determined and the extensiveness of these changes is estimated
Rejection or acceptance of the event	If the event registration is accepted, a change request is created for each configuration item affected
New CI initiation	A new configuration item is identified and created and the change is implemented. In the course of accepting the new item and 'placing it in storage, feedback is given to the configuration control board
Closure of the change request	The change request can be closed when the change has been implemented and accepted
Closure of the event registration	The event registration can be closed when all corresponding change requests are closed

Change management is originated by an incident. An incident may also be denominated a wish for correction but need not be designated as a certainly formulated wish. In this context, an incident is any monitoring of something surprising, unexpected, inconvenient, or directly wrong during running of the CI. It may, for example, be

- An incorrect formulation, captured during the review of a document
- A coding error discovered along a walk-through of a piece of source code
- An improvement request appeared from a new idea from the client during work on the project
- An error discovered in the integration test
- A wish to extend or improve the finished product, emerging if the product is in operation
- An request for information to a helpdesk about a problem in regards with usage of a system
- A change required in the code due to an upgrade to a new version of the middleware assisting the system, which may not be backward consistent (Ward *et al*., 2007).
- An incident should be documented in an incident log, which is the input to the change control activity. Some changes, such as those because of a review, can be expected and planned, while those because of, for example, a new client request cannot (Y.-M. Wang *et al.*, 2004).

As Figure 2.2 depicts the necessity of a PDCA analysis for all the changes affecting the project.

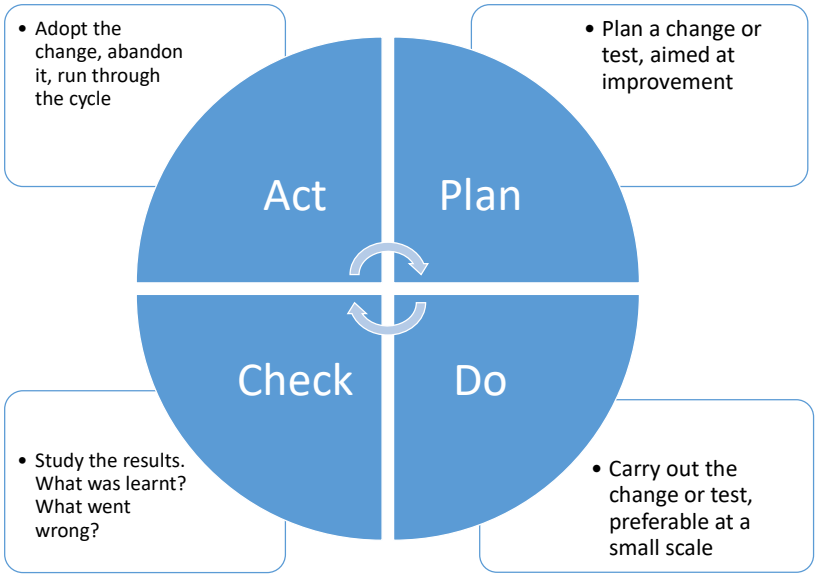


Figure 2.2. Change event PDCA (Tague, 2005)

As can be seen in the Figure 2.2, the change management flowchart is designed taking into account the rules of the PDCA analysis. Very frequently the change request is connected with the incident log, so no independent change requests are raised. This is not a very good idea, except if it remains possible to obtain statistics and status information on individual change requests as well as on the incident. This is mainly true if an incident produces changes in many CIs, which is frequently the case (Jarratt *et al*., 2011) (Su *et al*., 2007).

2.1.4 Configuration status accounting (CSA)

It is the process of developing and planning the knowledge base needed for the fulfilment of CM. Apart from facilitating CM, the aim of CSA is to supply a highly reliable source of configuration information to assist all program/project activities, together with program management, SE, manufacturing, SW development and maintenance, logistic support, modification, and maintenance (Sarma *et al*., 2003).

Apart from using automated CM tools, the process is supported or assisted by the documented CM process and open communications. The results from this activity give

visibility into CM document, activity status, and configuration information regarding the product and its documentation. They also contain "metrics" created from the information gathered in the CSA system and management "prompts" coming from inspection of the CM database (Anchuri *et al*., 2012). The Figure 2.3 depicts the Status accounting process and the inputs of the process.

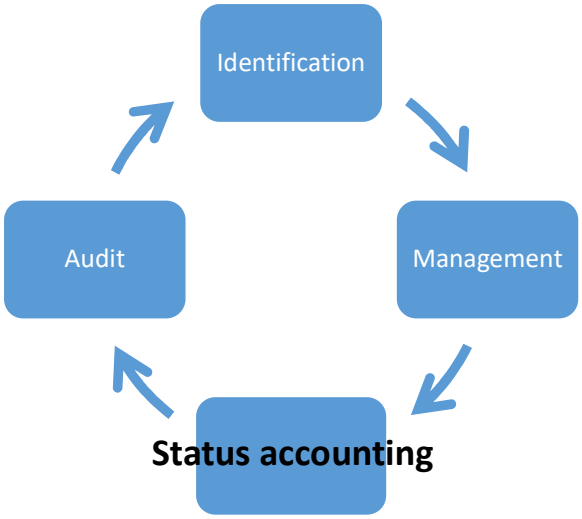


Figure 2.3. Configuration Status Accounting process (EIA, 1998)

As can be seen in the Figure 2.3, the CSA process is enriched with information as product development progresses. CSA inputs should be preserved in such range and depth as to be responsive to the requirements of the different assistance activities for access to configuration information. The data repository is the central point for the gathering, storage, processing, and publication of this data. Configuration information should be suitable on a request basis, either by hard copy or online computer access.

The CSA records are used as "best source" input data for purchasing data packages, design studies, and management analyses requested by the supporting/design activities. More concretely, the CSA metadata records must exactly represent the status of the configuration documents (specifications, drawings, lists, test reports, etc.) preserved in the document repositories (Monticolo *et al*., 2015).

2.1.5 Configuration Verification and Audit

The Functional Configuration Audit (FCA) is applied to prove that the concrete fulfilment of the CI meets the requirements declared in its fulfilment specification and to guarantee that the CI has met those requirements. After all of the accumulation have been fulfilled, a final (summary) FCA may be held to address the status of all of the action items that have been identified by the additional meetings and to document the status of the FCA for the system or CI in the minutes and certifications. Therefore, the audit is effectively fulfilled with a least amount of difficulties (Sun & Chang, 2018)

The Physical Configuration Audit (PCA) is applied to assess the concrete configuration of the CI that is representative of the product configuration in order to confirm that the related design documentation matches the design of the deliverable CI. In cases where the customer does not plan to control the detail design, it is still key that the contractor direct an internal PCA to define the starting point for controlling the production design and to create a product baseline (Erne & Knippenberg, 2008). Further PCAs may be fulfilled later during CI production if conditions such as the following take place:

- The production line is "shut down" for some time and then production is restarted
- The production contract for manufacture of a CI with a fairly complex, or difficult-to-manufacture, design is awarded to a new contractor or vendor.

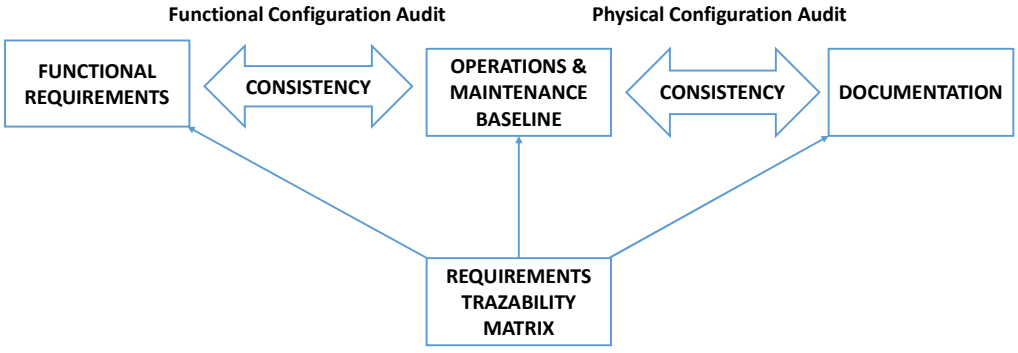


Figure 2.4. Configuration audits

As can be seen in Figure 2.4 the configuration audits ensure the consistency among functionality and documentation. Examples of audit types include the following:

- Functional configuration audits (FCAs): Audits conducted to verify that the development of a configuration item has been completed satisfactorily, that the item has achieved the functional and quality attribute characteristics specified in the functional or allocated baseline, and that its operational and support documents are complete and satisfactory.
- Physical configuration audits (PCAs): Audits conducted to verify that a configuration item, as built, conforms to the technical documentation that defines and describes it.

2.2 Configuration Management techniques and tools

CM contains a suitable mixture of techniques and tools (Ali & Kidd, 2013). The literature examination lists a large number of tools that can assist CM: quality functional deployment; test engineering management plan; failure modes and effects analysis etc. This section focuses on CM techniques and tools that can increase the consistency of large-scale projects delivered in complex environments. Next section compares the PM tools with CM tools (M. Burgess & Couch, 2006).

2.2.1 Requirements Management tools

Requirements management is the process of documenting, analyzing, tracing, prioritizing and agreeing on requirements and then controlling change and communicating to pertinent stakeholders. It is a constant process along a project. A requirement is an ability to which a project result (product or service) should be consistent.

The target of requirements management is to guarantee that an organization documents, conforms, and fulfils the needs and forecast of its customers and internal or external stakeholders (Stellman & Greene, 2005). Requirements management starts with the analysis and drawing out of the objectives and restrictions of the organization. Requirements management further contains assisting planning for requirements, incorporating requirements and the organization for working with them (features for requirements), as well as relationships with other information delivering against requirements, and changes for these (Lapouchnian *et al.*, 2007).

Requirements management includes communication between the project team members and stakeholders, and adjustment to requirements changes along the course of the project. To protect one class of requirements from overriding another, constant communication among members of the development team is key. For instance, in SW development for internal applications, the business has such strong needs that it may ignore user requirements, or believe that in creating use cases, the user requirements are being taken care of.

Modern requirements management tools enable rail users to adopt a unified approach to compliance and assurance, providing greater visibility into project health and flagging issues much earlier than traditional methods allow. It is possible to trace high-level project requirements and their decomposition through design and delivery to ensure compliance. Because requirements and their inter-dependencies are fully traceable, it is possible to:

- Quickly assess the impact and cost of change as it occurs across the project.
- Mitigate risks associated with the project lifecycle process to achieve a fully assured and compliant project
- Improve delivery performance by demonstrating the project meets reliability and maintainability objectives
- Reduce costly delays and overruns by continuously assuring the project throughout the lifecycle
- Speed up the entire project by ensuring a readily available flow of information
- Demonstrate compliance with recognized safety standards

For the concrete case of Doha Metro Construction Project the requirements management tool chosen was Comply Pro. Manage rail project requirements. With the support of this modern requirements management tool MHI managed to clearly define, record, and manage project requirements. Besides, we managed to raise, log, track, and resolve changes throughout the Doha metro construction project lifecycle and maintain them in a register so they are recognized and managed efficiently.

Figure 2.5 depicts that this requirements management tool manages project-wide safety hazards. Concretely, it identifies, records, analyzes, and implements measures to mitigate and control the effects of safety

Operating Mode	Hazard Description	Hazard Cause	Effect/Consequence	Exposed Group	Linked Safeguard ID	Linked Safeguard Owner	Linked Safeguard Description
Normal	Collision between trains	Non-communicating train enters reserved route	Collision with UTO train. Multiple fatalities and/or injuries to passengers.	Passengers	CM-165	S&TC	All works trains and self-propelled vehicles to be fitted with UTO equipment.
Degraded	Collision between trains	Driver or Control Room error	Injuries to passengers and/or Train Operator.	Passengers Staff	CM-166	Maintenance Vehicle	Speed restricted to 20 kph to allow safe line of sight driving.
					CM-167	ET	Crashworthiness specification.
Normal	Collision with power cables	Power cables fall on track	Damage to train. Possible injuries to passengers. Possible derailment.	Passengers System	CM-175	V201	Detailed design to inpower line to achieve the required clearance.
					CM-176		Design protective framework.
Normal	External fire	None specifically identified	Smoke inhalation injuries to passengers.	Passengers	CM-183		Smoke detection system capable of identifying source
					CM-188		Tunnel ventilation.
					CM-189		Station and tunnel fire alarm monitoring by Signalling. Prevent train from entering fire zone
Emergency	Toxic fumes and smoke	Fire in station	Fatalities and Smoke inhalation injuries to passengers.	Passengers	CM-731		LSOH cables used.
					CM-732		Design free-standing machines in stainless steel cases to reduce risk of spread of fire; internal fire constrained by lack of oxygen within case.
Emergency	Toxic fumes and smoke	Fire on train in tunnel	Train evacuation leads to Smoke inhalation injuries to passengers	Passengers			
Normal	Overcrowding in stations / rushing into trains	Crossing passenger flows, peak hours or major festivals	Fatalities and possible injuries to passengers (Crush injuries, fall down stairs / escalator, fall from platform).	Passengers	CM-689		Option to close fare gates, stop escalators.
					CM-690	CMMS	Passenger flow predictions addressed in architectural design.
					CM-169		CCTV surveillance
					CM-691	APG	Control of lifts and escalators from Station Control Room.
Normal	Unauthorised access to viaduct	Access via platform end door or PSD	Person hit by train. Electrocution injuries to passengers from Conductor Rail.	Passengers	CM-177		Platform Screen Doors installed on above ground stations.
					CM-358		Electro-magnetic locks.
					CM-1490		EAC control on Platform Screen Door.

Figure 2.5. Comply Pro hazard management rail project specifications

ID	Requirement Type	Requirement Category	Requirement Text	Status	Verification Criteria	Verification Status
DP04.243	Heading		**1 Introduction**			
DP04.1	Information	Construction	The Rail Upgrade Program requires 4 new viaducts to be constructed and 3 existing viaducts to be upgraded.			
DP04.2	Requirement	Construction	New viaducts shall be built in the following locations: o Aylestone Valley and Kimberley Plate o Crossing the Victoria River at Chesterton o Crossing the Maple River 1 mile to the south of Latticeham o Crossing the Ribble Estuary to the south of Preston.	Allocated	Confirm that each bridge has been built in the correct locations. When the initial site is being prepared and inspector shall be sent to each site to confirm the correct locartion is being used.	VC Agreed
DP04.3	Requirement	Construction	The following viaducts shall be upgraded: o Slingford Crossing o Oldacre Viaduct o The Union Lane Viaduct	Allocated		VC Not Ready
DP04.259	Heading		**2 Design 1: 110m Steel I-Girder 3 Span Viaduct**			
DP04.4	Information	Construction	The Viaduct consists of 3 spans. Horizontally curved steel spans present many unique challenges. Despite their challenges,curved girder spans have become widespread and are commonly used at locations that require complex geometries and have limited right-of-way, such as urban interchanges. Some of the important issues that differentiate curved steel girders from their straight counterparts include the effects of torsion, flange lateral bending, their inherent lack of stability, and special constructibility concerns. Also, the complex behavior of horizontally curved bridges necessitates the consideration of system behavior in the analysis.			
DP04.5	Information	Construction	However, horizontally curved girder spans require special attention during design and construction. Fabrication can require additional labor or material, and shipping costs may be greater than for a straight girder. Due to torsional behavior during lifting of the girders during erection, additional lifting points and temporary supports may be required, leading to increased costs. Nevertheless, curved girder bridges are typically more economical than kinked or chorded girder bridges that are on a horizontally curved alignment.			

Figure 2.6. Comply Pro viaduct rail project specifications

As observed in Figure 2.6 for the particular case of a viaduct through a modern requirements tool like Comply Pro it is possible to develop rail project specifications. Concretely, it advances project specifications from the earliest conceptual stage – so that cross-discipline expertise and knowledge are merged to deliver smarter, better-engineered solutions.

2.2.2 Product Breakdown Structure (PBS)

In project management, a product breakdown structure (PBS) is a tool for analysing, documenting and communicating the results of a project, and forms part of the product based planning approach. As can be seen in Figure 2.7 the PBS represents "an exhaustive, hierarchical tree structure of deliverables that make up the project, arranged in complete-part relationship" (Alvarez-Rodríguez *et al*., 2015). This diagrammatic representation of project outputs provides a clear and unambiguous statement of what the project is to deliver. It disintegrates a "Main Project Product" into its composing parts in the form of a hierarchical structure.

Once completed the PBS gives certainty that what is required is clearly understood and that the relationship between a product and its composing parts has been defined (Fossa & Sloman, 1996). From the PBS a Product Flow Diagram can be produced to establish the order in which sub-products build into the main project product. It supports describe the logic of the delivery plan.

The PBS is exactly alike in format to the work breakdown structure (WBS), but is a separate entity and is used at a different phase in the planning process. The PBS goes ahead the WBS and focuses on recording all the desired outputs (products) needed to fulfil the target of the project (Feng *et al.*, 2003).

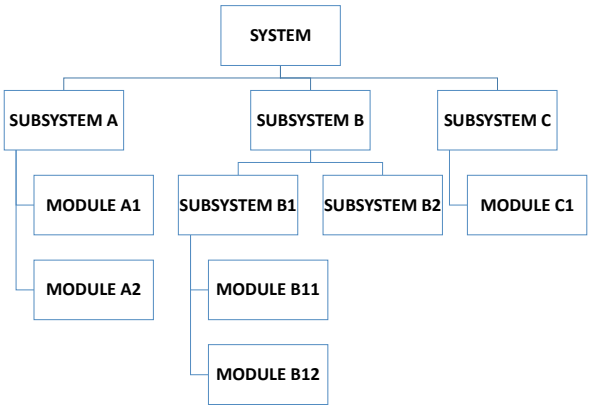


Figure 2.7. Product Breakdown Structure (PBS)

As can be seen in the Figure 2.7, by means of the PBS the final product is split into systems and subsystems to facilitate its handling.

2.2.3 Baseline

In CM, a "baseline" is an agreed description of the features of a product, at a point in time, which supports as a basis for defining change (Handbook, 2001). A change is a movement from this baseline state to a next phase. The identification of important changes from the baseline state is the main goal of baseline identification (Team, 2002).

Typically, important states are those that receive an official approval status, either explicitly or implicitly. An approval status may be registered separately, when a previous definition for that status has been established by project leaders, or denoted by association to a position above or below the created baseline.

However, this approval status is usually recognized publicly. Therefore, a baseline may also mark an approved CI, e.g. a project plan that has been signed off for accomplishment. The same way, incorporating multiple CIs with such a baseline depicts those items as being approved (Van Der Hoek *et al.*, 1996).

Usually, a baseline may be a single work product, or group of work products that can be applied as a logical foundation for comparison. A baseline may also be founded as the basis for subsequent select activities when the work products meet specific criteria. Such

activities may be attributed with official approval. A baseline may be a concrete type of baseline, such as the body of items at a specific certifying review (Abran *et al*., 2004)

In Table 2.2 can be observed the main features of the main baselines of a project

Table 2.2. Features of the main baselines

Name	Purpose	Subject matter	Owner	User	Freeze
As-required baseline	baseline the requirements	physical and functional requirements	Requirements management	Engineering	after requirements approval before design kick off
As-designed baseline	baseline the design	design drawings, specifications, BoMs,…	Engineering	Manufacturing	after critical design review and before manufacturing kick off
As-built baseline	baseline the product	manufacturing drawings, source code, licenses…	Manufacturing	Maintenance	after product audits and before maintenance kick off

Figure 2.8 depicts the main baselines' composition.

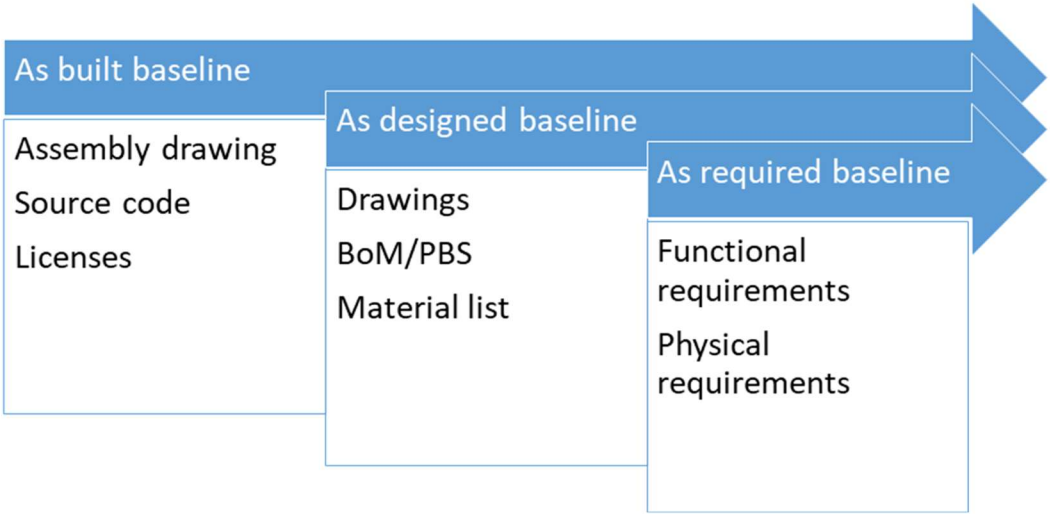


Figure 2.8. Composition of the main baselines

2.2.4 Configuration Management Database

The Configuration Management Database (CMDB) is an ITIL database applied by an company to store information about HW and SW assets (commonly referred to as Configuration Items). This database is a data warehouse for the company and also keeps

information regarding the relationships among its CIs (Brenner *et al*., 2006). The CMDB gives a means of understanding the organization's critical CIs and their relationships, such as information systems, upstream sources or dependencies of CIs, and the downstream targets of CIs (Earl, 1992).

CMDBs can be applied for many topics, containing but not limited to: business intelligence, SW and HW builds, inventory (Benson *et al.*, 2016), impact analysis for change management (Sauvé *et al.*, 2006) and incident management. In the conditions of ITIL, the application of CMDBs is as part of infrastructure operations and support. The CMDB describes the authorized configuration of the important components of the IT environment. As a summary, the CMDB includes and records data that are also called CIs. It also gives details about the essential attributes of CIs and the relationships among them (Graves & Mockus, 1998).

2.2.5 Form fit function analysis

The expression form fit function, also referred to as FFF, is applied in manufacturing to describe the identifying features of an item (a single component that goes into the final build of your product, usually maintained on an item master). Form fit function can be described as:

- Form: the shape, size, dimensions, mass, weight and other visual parameters that uniquely distinguish an item. For instance, you might describe a screw that will be used in your product as ‘SCREW, PAN HEAD, M3 x 0.5, 2mm Lg, 316 SS.’
- Fit: the ability of an item to physically interface with, connect to, or become a fundamental part of another item. For instance, if the previous screw is wanted to fit to properly in the final product, it must follow the rules set by engineering in the design phase. This might contain specifications for the space around the screw relative to a faceplate hole or the location of the screw’s top location relative to the product surface.
- Function: the action or actions that an item is designed to perform. In the example, the screw is planned to hold other items of the product together.

A correctly applied and rigorously enforced part numbering scheme coupled with well understood form fit function rules will ensure that the changes required and started by

manufacturing or engineering are correctly tracked and easily followed by the whole team. Typically upon receiving new part specifications (whether in the initial build or due to a change), manufacturers will attribute each newly specified part a unique number based on a planned part numbering scheme and register the information into an item master. This guarantees that purchasing, engineering and design teams all have the correct form fit function requirements for every part (Pikosz & Malmqvist, 1998).

2.2.6 Effectivity analysis

Elements that are effectivity-managed cover parts, documents, and the following manufacturing process objects: process plans (manufacturing process plans, sequences, and operations) and manufacturing resources (plants, resource groups, skills, process materials, tooling, and work center).

Effectivity is the scheduled date, lot, or serial number at which old versions of the element are substituted by new versions in production. Changes can be made to planned effectivity settings from a change notice or a change task. When the change notice has been approved, the scheduled effectivities get copied over as concrete effectivities on the mentioned element (Wright, 1997).

Effectivity set on a variance is descriptive information only. No system processing happens if the effectivity is set on a variance. When adding an effectivity data, you can specify multiple data and separate intervals. For example, you could set Part 123, version B to be effective on Product ABC, serial numbers 1 through 5, and 9 through 11. The Table 2.3 shows all the necessary terms to understand and apply successfully the effectivity analysis

Table 2.3. Effectivity terminology

Effectivity terminology	
Term	Description
Effectivity level	The product element that provides the context for setting serial number effectivity
Effectivity structure propagation	The system action where an effectivity statement is written on child versions in a product structure to a defined configuration specification
Effectivity qualifier	Additional information that further defines the meaning of an effectivity statement. For example, Exact or Not later than
Effectivity type	The kinds of effectivity that can be set on an object version. The choices include serial number, manufacturing sequence number (MSN), lot number, block number and date. The allowable types are determined by default trace code recorded on a traceable part
End item	The top part in a product structure that represents what is delivered to a customer in fulfilment of a contract
End item effectivity	An effectivity statement written on a part version with the context of an end item
Sibling propagation	The system action where the effectivity ranges of previous object versions are modified to reflect the effectivity of a new object version
Traceable part	A component or assembly part that has been designated as traced by serial number.

As can be observed in Table 2.3 effectivity is a key feature in CM and because of that has its own terminology and the terms have to be agreed and understood by the main stake-holders of the project.

2.2.7 Configuration Management tools vs. Project Management tools

CM incorporates many tools and techniques and the ones debated in the aforementioned paragraphs have been chosen because are appropriate to transform the consistency of projects delivered in complex environments (Dart, 1991). Even if some of the previous tools are applied in PM (e.g. the PMBOK (PMI, 2004)) the CM methodology to them is radically distinct. Key conceptual differences between CM and PM tools are due to their different focus, CI versus project.

In other words while the PM applies those tools to the project the CM enlarges the view to the CI and its whole life cycle. This holistic approach radically changes the point of view and the result of the analysis Table 2.4 represents the CM tools from the CM and PM point of view.

Table 2.4. CM tools vs PM tools

CM tools vs PM tools	CM	PM
Requirement Management tools	In addition to the traditional techniques for requirement management, SE uses models, software tools and the "cascade" model	PM captures requirements in documents and uses a Requirements Traceability Matrix to ensuring the traceability.
Product Breakdown Structure	PBS focuses on cataloguing all the desired outputs (products) needed to achieve the goal of the project	PBS feeds into creation of the WBS, which identifies the tasks and activities required to deliver those outputs
Baseline	Baselines themselves are valued not only to use to identify the notable state of work product(s) but also provide historical views of how work product elements have proceeded together over time	Milestones are tools used in PM to mark specific points along a project timeline. These points may signal anchors such as a project start and end date, or a need for external review or input and budget checks
CM database	It is an Information Technology Infrastructure Library (ITIL) database to store information about hardware and software assets (Configuration Items) and their relationships in order to build a final product	Project management software (PMS) has the capacity to help plan, organize, and manage resource tools and develop resource estimates
Form fit function analysis	It helps companies decide whether a product change it is interchangeable or not and as a result if it should require a new item number or simply a revision of the same item number in order to ensure the trazeability	According to lean project management the main principle of PM in any product change is delivering more value with less waste in a project context.
Effectivity analysis	It helps companies to decide the optimum point/time to incorporate a concrete change in the product	

As can be noticed in Table 2.4 CM provides the tools to support the project management.

2.3 Whole Configuration Management process

The CM disciplines and tools discussed in this section and summarized in Figure 2.9 are the main CM elements that can transform and improve the consistency in large scale project environments. Table 2.5 based on the literature presented in the previous sections, introduces the main elements of CM and highlights how these elements interact each other. Their implementation and their interrelationship positively impact on the consistency of large scale project (Goldsack *et al*., 2009).

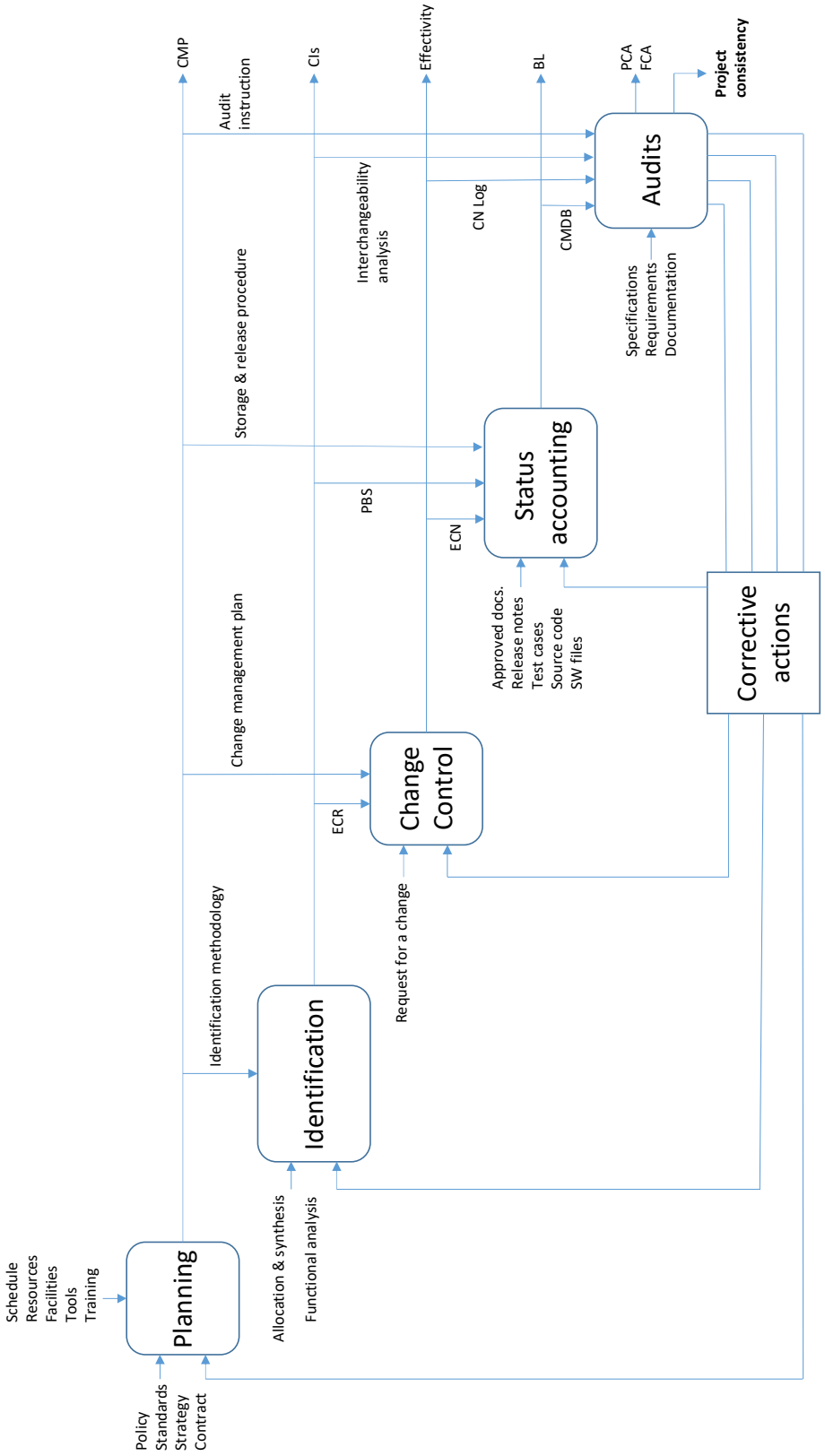


Figure 2.9. CM as a whole (Army, 2003)

The main CM elements are:

- Configuration Management Plan
- Configuration Identification
- Change Control
- Change Control Board
- Status Accounting
- Verification and audit
- Configuration Management Database
- Product Breakdown Structure
- Baselining; Interchangeability analysis
- Physical and Functional Audit
- Requirements management tools

Each interaction is discussed in Table 2.5.

Table 2.5. CM disciplines relationship

Relationship			Description
CM	Planning	Identification	CM planning founds the criteria and methodology for Cis selection and identification
		Change control	CM planning establishes the methodology for control of releases and changes to CIs
		Status accounting	CM planning creates the process for recording and reporting CI descriptions (e.g. hardware, software, firmware, etc.) and all departures from the baseline during design and production
		Audit	CM planning settles the process for verifying the completeness and correctness of Cis throughout the system life cycle
	Identification	Change control	CM identification is the basis by which changes to any part of a system are identified, documented and later tracked through design, development, testing and disposal
		Status accounting	In the event of suspected problems, CM identification makes the quickly verification of baseline configuration and approved modifications possible
		Audit	CM identification is the process of identifying the attributes that define every aspect of a CI. These attributes are recorded in configuration documentation and baselined. Baselining attribute forces formal CM audit to be effected at delivery
	Change control	Status accounting	CM change control creates the set of processes and approval stages required to change a configuration item´s attributes and to re-baseline them
		Audit	CM change control ensures that functional and performance attributes of a CI are achieved prior a functional configuration audit and ensures that a CI is installed in accordance with the requirements of its detailed design documentation prior a physical configuration audit
	Status accounting	Audit	CM status accounting records and reports the configuration baselines associated with each CI according to the PBS as a fundamental condition for successful audit

As can be noticed in Table 2.5, all the CM disciplines are interconnected and they enrich each other making a network that ensures the complete traceability of the items composing the final product.

2.4 Conclusion

CM transforms and improves the consistency with disciplines centered on the Configuration Identification approach and the CIs. Configuration Identification takes into account the environment, and its interactions, in which the project is accomplished. The PBS including the CIs defining the project consistency, enables the definition of a complete

and accurate plan with a multidisciplinary and systemic approach. Configuration Management Plan provides the best definition of roles, responsibilities, requirements, interfaces and objectives.

The communication among the involved organizations is supported by a common CMDB. The strategic tools that support the CMDB consistency are requirements management, PBS, baselining, FFF and effectivity analysis, which ensures the delivery of the project with a focus on the advantages over the subsequent life-cycles.

In this chapter, it has been described how all the techniques and tools available in configuration management work alone and integrated throughout the process to carry out their work. It would be necessary after this to review the achievements that this discipline can get in the field of project management.

Chapter 3

State of the Art

The state of the art will describe the current knowledge about the discipline of the Configuration Management through the analysis of similar or related published work in the area of Large-Scale Project Management. CM provides the alignment between project management and SE which is critical to project success. CM contributes a critical ingredient to successful change.

The Project Management Plan (PMP) and the Systems CM Plan (CMP) are key documents used to define the processes and methodologies the project will employ to build and deliver a product or service. The PMP is the master planning document for the project. It describes all activities, including technical activities, to be integrated and controlled during the life of the program.

The configuration management plan (CMP) is developed to define, document, control, implement, account for, and audit changes to the various components (CIs) of this project. Although they vary in terms of focus and approach, each discipline is essential to moving projects and people from the transition state to the desired future state. It will provide a comprehensive overview of what has been done in the field of CM as a support of LSP and what should be further investigated, in order to help formulating the problems and hypothesis the thesis intends to address. This will involve analysing, comparing, evaluating and linking different sources.

This chapter of the dissertation shows how configuration management is able to steer the large-scale projects management focus from the final product to the configuration items increasing the consistency of the project. Besides of that, it will be reviewed the LSP success definition according to CM. As a result, the consistency will be identified as the

main factor for success of the LSP and it will be reviewed the role of CM in order to add general consistency to any project.

However, not only does CM add consistency to the projects, but to the rest of the issues inherent to the common characteristics of the LSP. Therefore, several famous large-scale projects are reviewed and as a result from that, some characteristics in common are identified. Next, all the common characteristics related to them are investigated one by one and some issues related to each of them are drawn. In this context, the discipline CM emerges as a scientific method to face and consequently to solve all of those concerns by means of the CM tools.

Throughout the discussion of related work, issues not addressed by previous research that have prevented a more widespread adoption of Configuration Management, are identified. Table 3.1 shows 3 examples of large-scale projects, depicting at each example, the characteristics of the LSP, the problems associated to each characteristics, the solutions to those problems, how the CM helps to solve those issues and finally the tools applied to get those goals.

Table 3.1. Large scale project vs CM.

Example	Definition	Characteristics	Problem	Solution	CM	Tool
Dubai Aviation Engineering Projects (DAEP)	LSP	High-tech	Risk of cyber incidents	Identification and correction of misconfigurations	Identification	BL
Boeing 737 assemble		Capital intensive	Economic slowdowns	Right degree of project control	Change management	Change flowchart
Doha metro construction		Long duration	Loss of focus	Backwards management	Configuration management planning	CMP
		Many companies	Latest version uncertainty	Knowledge organization	CSA	CMDB

As can be seen in the Table 3.1 CM raises as a network of support facing some of the problems associated to the characteristics of LSP through the specific tools of the disciplines of CM described in the previous chapter: baselines, change management flowcharts, CM plan and CM database.

3.1 Large scale projects

Large scale projects are a group of projects that have particular defining characteristics:

1. high-tech,
2. capital intensive engineering projects
3. long duration
4. require different companies to work together across firm limits in project delivery (Brady et al., 2007; Hobday, 1998)

Such projects deliver complicate product systems, such as aircraft, satellites, and trains. Their completion requires systems integration capabilities, as complex product systems are designed and integrated through a network of component and sub-system suppliers (Davies & Mackenzie, 2014) (Davies & Hobday, 2005).

Within these projects information about complex product systems is developed across multiple companies, involving many professions and trades, as these organizations communicate through the digital systems. The largest construction project in the world is — no surprise — the massive new air hub planned in Dubai, Al Maktoum International Airport. Here's a list of the Top Ten:

1. Al Maktoum International Airport, Dubai
2. Jubail II, Saudi Arabia
3. DubaiLand, Dubai
4. International Space Station, Space
5. South-North Water Transfer Project, China
6. London Crossrail Project
7. High Speed Railway, California
8. Chuo Shinkansen (High Speed Railway), Japan
9. Beijing Airport, China
10. Great Man Made River Project, Libya

3.1.1 Al Maktoum International Airport

The Dubai Aviation Engineering Projects (DAEP), an autonomous entity responsible for the planning and development of Dubai's airports, has embarked on a new expansion project at Al Maktoum International Airport. Approved in 2014, the AED120bn ($32.67bn) expansion project is devised to make the airport the world's biggest, in terms of size and passenger capacity, by 2050. It will also ease pressure on the nation's first airport, Dubai International Airport, which is expected to witness 100 million passengers by the end of 2020, and has limited scope for further growth. Figure 3.1 shows a scale model of the expansion of the airport.

Figure 3.1. Al Maktoum International Airport (*Dubai Aviation Engineering Projects (DAEP)*, n.d.)

As can be guessed from Figure 3.1 Al Maktoum International Airport will become the biggest worldwide. The expansion will be executed in two phases over the following years. After the project's completion, the airport will be able to handle more than 220 million passengers a year. Biometrics and smart passenger tracking will be used during the expansion to facilitate seamless passenger movement through the terminals (Dubai Aviation Engineering Projects (DAEP), n.d.).

3.1.2 Boeing 737

Boeing Co. is one of the US's biggest and best-known manufacturers, employing 158,000 workers and producing tens of billions of But to assemble just one model of airplane — the 737, its smallest and most popular commercial jet — the company relies on a complex web of hundreds of suppliers providing everything from engines and fuselages to seats and exit signs.

In part because Boeing serves so many large international airline customers, the company also relies heavily on international suppliers. Among major suppliers, China's Xian Aircraft Co. makes 737 vertical fins, and Japan's Mitsubishi Heavy Industries provides the wing's inboard flaps. The 737, which is made up of 367,000 parts, is assembled at a factory in Renton, Wash., south of Seattle. dollars worth of commercial jets and defense systems annually for customers worldwide. Figure 3.2 shows the deliveries of the model 737 per year from 1967 to 2018

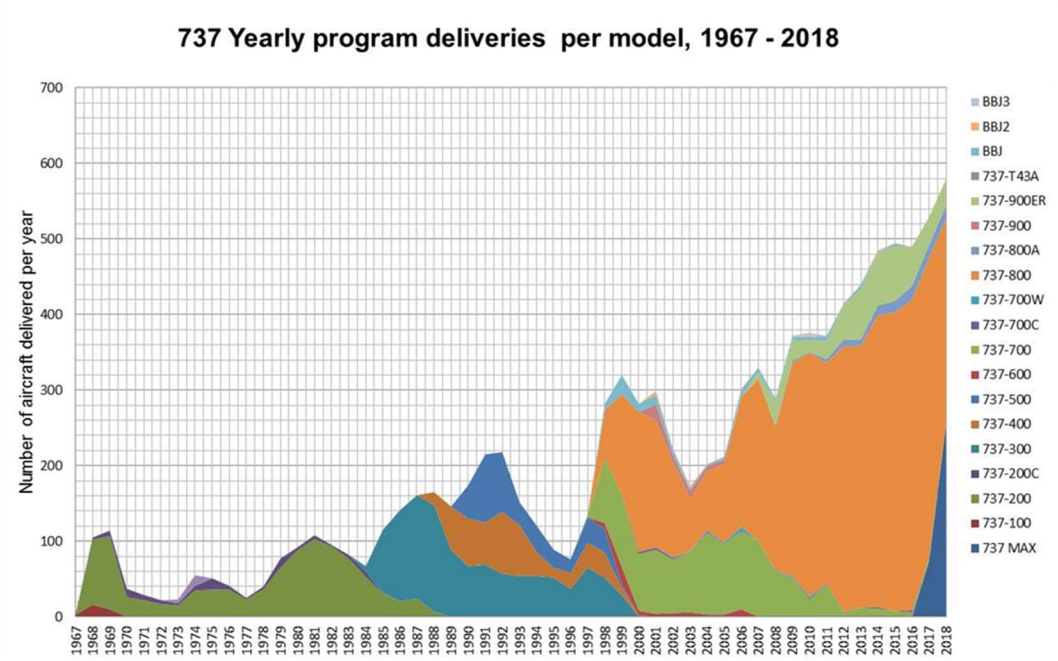


Figure 3.2. Boeing 737 deliveries along the history (*Boeing*, n.d.)

As can be seen in Figure 3.2 the number of aircrafts delivered per year has increased progressively. Boeing delivered 372 of the single-aisle 737s last year — a little more than one a day (msnbc.com, 2010).

3.1.3 Doha metro

Doha Metro in Qatar's capital city will be one of the most advanced rail transit systems in the world when Phase I becomes operational by the end of 2019. The rail transit system is part of the Qatar Government's €130bn diversification and modernization strategy involving public and private investments. The campaign also aims to reduce its dependency on natural gas exports. The Doha metro network is estimated to cost $36bn. Figure 3.3 shows the 3 lines of the phase 1 of Doha Metro

Figure 3.3. Phase 1 of Doha Metro project (*Página Web de Qatar Rail*, n.d.)

As can be seen in Figure 3.3 in the Phase 1, which finished at the end of last year, only 3 lines have been finally constructed. It is planned to be built in phases and will become one of the most modern driverless railway networks in the world. It will have four lines, namely the Red, Green, Gold and Blue metro lines, with an overall length of 300km and 98 stations. Bids for the lines were submitted in 2012 and early 2013.

Doha Metro will be operated and maintained for a duration of 20 years by RKH Qitarat, a joint venture formed by Hamad Group (51 %) and French transit operators Keolis and RATP Dev (49 %), on behalf of system owner Qatar Rail. A consortium of Mitsubishi Heavy Industries (MHI), Mitsubishi Corporation, Hitachi, Kinki Sharyo and Thales

submitted a bid for the turnkey construction of a fully automated driverless metro system in March 2014 and received a letter of conditional acceptance (LCA) in February 2015 (*Doha Metro, Qatar*, n.d.).

3.2 Features of Large Scale Projects

As depicted previously the LSP have the characteristics described in this section:

3.2.1 High-tech

High technology, or high-tech is technology that is at the cutting edge: the most advanced technology available. Here are some of the highlights in the rail industry:

Recent analytical studies have shown that technology businesses are playing an important role in driving the nation's economy, and they are especially potent contributors to the growth of regional economies. High technology businesses, particularly in software, computers and the Internet, are creating many new, high paying jobs, and transforming a wide range of traditional economic sectors.

High technology is propelling the economy, growing four times faster than the overall economy in the 1990s (DeVol, 1999). Computers and information processing equipment accounted for over 40 percent of the growth in private, non-residential investment since 1995 (Pakko, 2000). Information technology industries accounted for a third of U.S. economic growth between 1995 and 1999 (Economics *et al*., 2000).

For those metropolitan areas hosting significant concentrations of high technology industries, the beneficial impacts have been tremendous. Internet companies, software developers, biotech concerns, and computer and electronics companies pay high wages to programmers, scientists and engineers, and the computer and electronics companies have provided many opportunities for entry level jobs.

Moreover, the importance of high technology reaches beyond its role in triggering recent economic growth. High technology is also an "indicator species" of the process of growth in a knowledge-based economy. The key roles played by continuous innovation and "speed

to market" in high technology today are increasingly coming to characterize the rest of the economy. In the years ahead, the process of development in all industries will more closely resemble the dynamics of high-tech industries. The classification of manufacturing industries is based into categories based on R&D intensities (Hatzichronoglou, 1997). As a result of that, high-technology industries Medium-high-technology industries are:

- Aircraft and spacecraft Electrical machinery and apparatus.
- Pharmaceuticals Motor vehicles, trailers and semi-trailers.
- Office, accounting and computing machinery Chemicals excluding pharmaceuticals.
- Radio, TV and communications equipment Railroad equipment and transport equipment, Medical, precision and optical instruments.

3.2.1.1. Problem associated to high-tech

The drive to increase productivity and reduce costs in industrial environments has led to an exponential increase in the adoption of automation. This integration of computation, networking and physical processes is what has characterized the 4th industrial revolution of today, known as Industry 4.0 or the Industrial Internet of Things (IIoT). The Figure 3.4 shows how the integration of physical industrial equipment and machinery with software defines Industry 4.0. (Lasi *et al*., 2014)

Figure 3.4. Industry 4.0 (Roblek et al., 2016)

As can be seen in Figure 3.4 the intersection of Industry 4.0 with the Industrial IoT (IIoT) adds sensors, connectivity, cloud, applications, big data and analytics, and intelligent

systems, brings to life real time automation and management across dispersed deployments:

1. In the case of Al Maktoum International Airport it will have full connectivity with all the stake-holders and superior proactivity and reactivity to adapt to the real time solicitation of the airport (operational needs, customer requests, …) in order to become a 4.0 airport.
2. For Boeing 737 the aviation 4.0 concept will significantly increase the safety levels in aviation through:
    - Automatic flying in predefined situations in a rule-based way.
    - Developing a robust aircraft predictive maintenance.
    - Cockpit safety cognitive computing aid systems.
    - Real-time weather information update.
    - Improved search and rescue services especially in the oceanic or remote area.
    - Real-time human performance monitoring and alerting based on nonintrusive physiological sensors/signals and contextual information.
3. In the case of the Doha metro the rail transport 4.0 will bring innovation within the railway industry for more climate protection and more customer quality

This is where real business value is being created. Although this trend provides clear business advantages, it has also led to manufacturing networks which are more interconnected, complex and heterogeneous. As a result, today's manufacturing networks are operating with an increased risk of cyber incidents understood as changes to a systems firmware, software or hardware without the system owners consents. (Bilge *et al*., 2017)

3.2.1.2. Solution to the high-tech related problem

Identifying and Correcting Misconfigurations: Network misconfigurations include unstable or broken links, commands that don't reach the destination, a network service or IP misconfiguration, "noisy" devices causing traffic floods and devices not configured correctly by the vendor or system integrator (Choi *et al*., 2004).

These misconfigurations happen frequently and can have a moderate impact on your bottom line, as the network will not always operate reliably or correctly. To maintain

operational continuity, you need to have visibility of all assets and communications within your network and leverage this visibility to identify any existing misconfigurations (Miller & Rowe, 2012).

3.2.1.3. Misconfigurations vs identification

In configuration management, a "baseline" is an agreed description of the attributes of a product, at a point in time, which serves as a basis for defining change. An "alteration" is a movement from this baseline state to a next state. The identification of significant changes from the baseline state is the central purpose of baseline identification. Typically, significant states are those that receive a formal approval status, either explicitly or implicitly. An approval status may be marked individually, when a prior definition for that status has been established by project leaders, or signified by association to a position above or below the established baseline.

Nevertheless, this approval status is usually recognized publicly. Thus, a baseline may also mark an approved configuration item, e.g. an engine version that has been signed off for implementation. In a similar manner, associating multiple configuration items with such a baseline indicates those items as being approved.

Generally, a baseline may be a single work product, or set of work products that can be used as a logical basis for comparison. A baseline may also be established as the basis for subsequent select activities when the work products meet certain criteria. Such activities may be attributed with formal approval.

Conversely, the configuration of a project often includes one or more baselines, the status of the configuration, and any metrics collected. The current configuration refers to the current status, current audit and/or current metrics. Similarly, but less frequently, a baseline may refer to all items associated with a specific project. This may include all revisions of all items, or only the latest revision of all items in the project, depending upon context. Figure 3.5 shows the different baselines that a LSP contains along the product life cycle

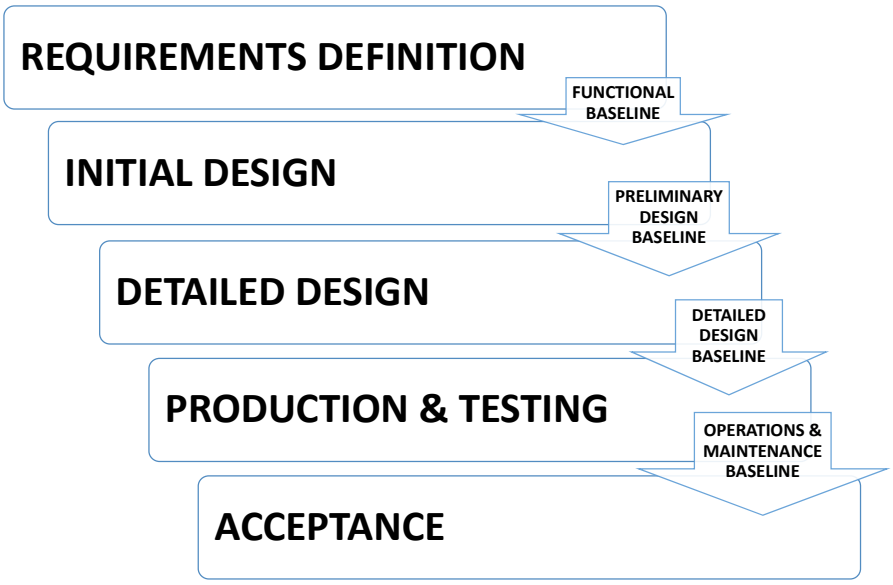


Figure 3.5. Baselines

As can be seen in Figure 3.5, as the development of the product progresses, the different baselines ensure that all the information related to the project keeps consistent and it is approved and traced. Some examples include:

- Functional Baseline: The approved functional configuration documentation (documentation describing the system's functional, performance, interoperability, and interface requirements and the verifications required to demonstrate the achievement of those specified requirements).
- Allocated Baseline: The approved allocated configuration documentation (documentation describing a CI's functional, performance, interoperability, and interface requirements that are allocated from those of a system or higher level configuration item; interface requirements with interfacing configuration items; and the verifications required to confirm the achievement of those specified requirements).
- Developmental Baseline: state of work products amid development.
- Product Baseline: contains the releasable contents of the project.
- Others, based upon proprietary business practices.

3.2.2 Capital intensive

Capital intensive projects require large amounts of investment to produce a good or service, and therefore have a high percentage of fixed assets, which are also known as property, plant, and equipment (PP&E). Companies in capital-intensive industries are often marked by high levels of depreciation. All these industries require massive amounts of capital expenditure. Over half of capital intensive projects globally underperform, with those in rail, defence and infrastructure in particular having poor track records. European projects have the joint lowest rate of reaching objectives, on 42% alongside Africa.

The Figure 3.6 shows the 'Global 2019 Project Controls Survey' (*Logikal Projects Global Projects Controls Survey Report 2019*, n.d.) that has researched nearly 700 project professionals from around the world to find out the state of the project profession, and in particular the state of project controls. According to the cross-industry survey, internationally 60% of projects have not fully met their objectives, with 18% failing to meet any objectives.

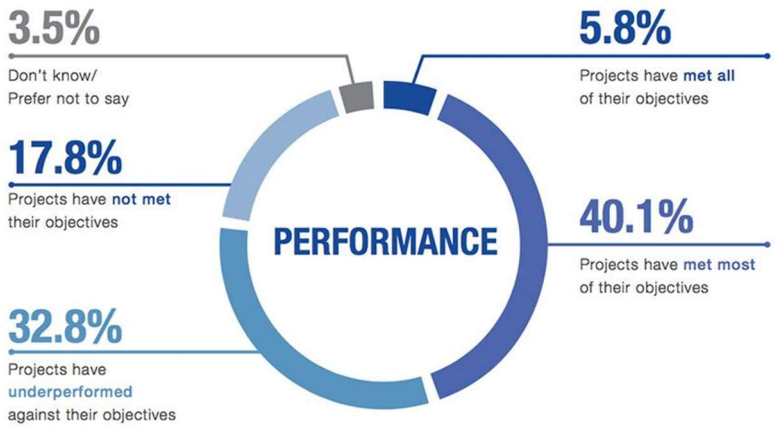


Figure 3.6. European capital intensive projects underperformance (*Logikal Projects Global Projects Controls Survey Report 2019*, n.d.)

As can be seen in Figure 3.6 Logikal found that just 5.8% of all projects met every single one of their objectives. 40.1% hit most of their objectives, but this stands in stark contrast to the 17.8% which did not meet their objectives, or the huge 32.8% that underperformed against their objectives. Projects that cost more than US$1 billion are proliferating.

According to estimates from the Organization for Economic Co-Operation and Development (OECD), by 2030, the combined annual infrastructure investment requirements for telecommunications, road, rail, electricity and water are likely to total an average of 2.5 percent of world gross domestic product (GDP). If electricity generation and energy-related infrastructure investments in oil, gas and coal are included, the annual share of GDP rises to around 3.5 percent.

3.2.2.1. Problem associated to capital intensive

Capital intensive industries tend to have high levels of operating leverage, which is the ratio of fixed costs to variable costs. As a result, capital intensive industries need a high volume of production to provide an adequate return on investment. This also means that small changes can lead to big changes in profits and return on invested capital.

Their high operating leverage makes capital intensive industries much more vulnerable to economic slowdowns compared to labor-intensive businesses because they still have to pay the fixed costs such as overhead on the plants that house the equipment, depreciation on the equipment and other fixed costs associated with a capital intensive businesses, even when the industry is in recession (Paté-Cornell & Dillon, 2001).

3.2.2.2. Solution to the capital intense related problem

Because of the complexity and high financing costs of such projects, it is critical for project management to have transparent, easy-to-understand, consistent and comparable information within the project. At each stage, budgets should be presented clearly via indicators and should be part of “go/no-go” decisions. Global companies that have built strong methodological processes and used high-powered models find that these tools and methods prove their worth in helping management make key decisions.

They can be invaluable, for example, in determining when to abandon a project whose profitability targets seem weak. Ultimately, these procedures provide visibility into long-term, capital intensive projects that are fraught with many kinds of quantifiable risk.

3.2.2.3. Economic slowdowns vs control

Applying a formal process gives project teams a way to improve results in the near term while also setting the stage for repeated success across future projects. Established approaches for managing change on projects use Configuration Management, a systems engineering approach with origins in the mid-20th century. In its original form, Configuration Management is characteristics of what (Morris, 2008) describes as 'project management 1.0'.

It involves hierarchical, sequential and a synchronous processes; managing change against a baseline. Its use focuses attention on assets as configuration items: sub-systems or components that have value to the organization, in which changes will often have systemic consequences on the function or layout of other items within the product structure and hierarchy. The Figure 3.7 shows all the steps that have to be followed in order to ensure a proper control of all the changes

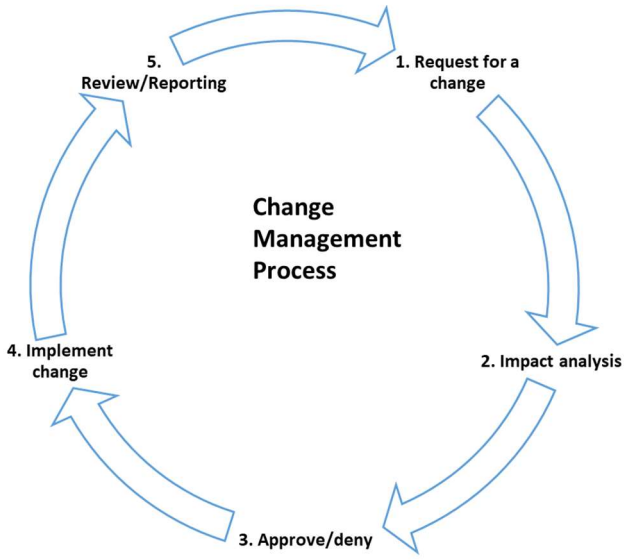


Figure 3.7. Change management process

As can be seen in Figure 3.7 all the changes of the project should be subject to those 5 steps:

1. Creation of the request.
2. Impact analysis.
3. Approval/rejection.
4. Implementation.
5. Reporting.

New practices of managing change in large projects might be expected as we face large projects, in which internal and external information sets become linked and configuration item information becomes a project deliverable. (Morris, 2008) argued that methodological change control is fundamental to good project management intuitive change control is one of the issues that limits managers' ability to execute viable project plans. Others see projects, themselves, as information processing systems. As project management information systems are increasingly used, altering the pace and complexity of projects, there are challenges to the no customized heavy formality of several of the techniques to manage large-scale, one-time, non-routine projects. Here, Morris, like Levitt, points to software projects, in particular, as rebelling, using agile methodological forms of management, through small projects with close developer-customer relationships.

3.2.3 Long duration

Duration is defined as the time to complete the project. Time is often the most frequent project oversight in developing projects. This is reflected in missed deadlines and incomplete deliverables. Proper control of the schedule requires the careful identification of tasks to be performed and accurate estimations of their durations, the sequence in which they are going to be done, and how people and other resources are to be allocated. Generally speaking, short-term projects take a few hours, days or weeks to complete. Long-term projects take months or even years to finish.

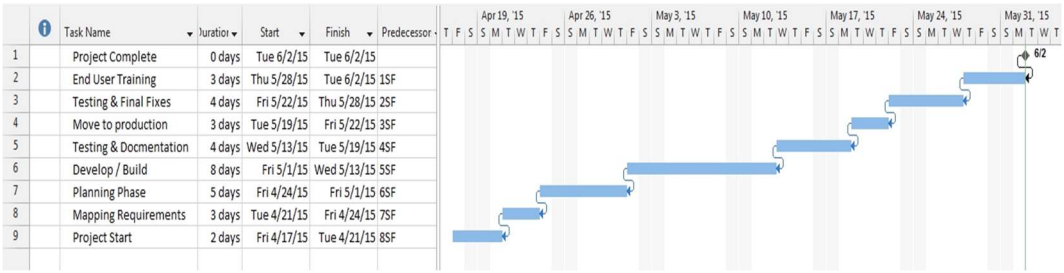
3.2.3.1. Problem associated to the long duration

When it comes to long term project planning and management, several challenges and difficulties can arise which require a total rethink of the project's future or scope, all the way up to having to abandon it or adopt a whole new working approach. Loss of focus, the slowing of progress, a lack of immediate results, poor diligence and losses in productivity

are only a few of the perils that can strike long-term projects. To avoid these, a good project management needs to be able to prepare for almost every eventuality successfully.

3.2.3.2. Solution to the long duration related problem

When planning for long term projects, there are many factors and decisive events that one might not be able to foresee from the beginning of the project. This creates a need to plan ahead and create a comprehensive timeline. One effective approach is to start from the end. The Figure 3.8 shows the steps that have to be followed to do a general backwards planning from the kick-off until the complexion



	Task Name	Duration	Start	Finish	Predecessors
1	Project Complete	0 days	Tue 6/2/15	Tue 6/2/15	
2	End User Training	3 days	Thu 5/28/15	Tue 6/2/15	1SF
3	Testing & Final Fixes	4 days	Fri 5/22/15	Thu 5/28/15	2SF
4	Move to production	3 days	Tue 5/19/15	Fri 5/22/15	3SF
5	Testing & Docmentation	4 days	Wed 5/13/15	Tue 5/19/15	4SF
6	Develop / Build	8 days	Fri 5/1/15	Wed 5/13/15	5SF
7	Planning Phase	5 days	Fri 4/24/15	Fri 5/1/15	6SF
8	Mapping Requirements	3 days	Tue 4/21/15	Fri 4/24/15	7SF
9	Project Start	2 days	Fri 4/17/15	Tue 4/21/15	8SF

Figure 3.8. Backwards planning

The Figure 3.8 shows the nine steps valid to any project starting in the complexion date and calculating all the lead times necessary to fulfil all the milestones. As the final deadline is generally one of the most solid facts you can build upon, it is sometimes easier to consider the development of a project backwards. Pinpoint the final deadline and start building the necessary steps that need to be made to complete a project, starting from the last and moving backwards to the first.

Keep in mind that it is very important to make reasonable time estimations, update timeframes regularly as the project evolves and keep all stakeholders informed and on board. This method is proven to be effective in boosting the attitude of those involved as it is easier to keep track of successes, celebrate milestone deliveries as well as helping overall time management (Salama *et al*., 2008).

3.2.3.3. Uncertainty vs. planning

When we're facing a deadline or getting ready to tackle a big project, the standard approach is to plan from the beginning and then work our way forward, starting with the first step and ending with our final task. In recent years, however, a method known as backwards or reverse planning has been receiving a lot of attention, and it's exactly what it sounds like.

When you plan in reverse, you start with your end goal and then work your way backwards from there to develop a plan of action. As counterintuitive as it may seem, working backwards in this way can give you a much clearer picture of what and how much must be accomplished during each phase of a project. It can also help you identify and avoid unnecessary activities.

Motivation tends to be at its highest at the start and near the finish of a project. But reverse planning provides a kind of map to the finish line by outlining each step we will need to take along the way, and this can help us stay focused throughout the whole project, even at times when motivation might otherwise be lagging. These effects don't seem to be driven by the plan itself, however, because once finished, backwards and forwards plans tend to look pretty similar. Instead, it seems to be the process of reverse planning that can increase motivation and goal expectancy.

One reason for this is that focusing on the end goal allows us to use our imagination to think of future events as if they already happened, which makes it easier for us to visualize the steps we will need to take. The researchers refer to this as "future retrospection."

Future retrospection can help increase our anticipation of pleasure from achieving our goal, which brings about goal-directed behaviors. Another reason backward planning may be more effective than forward planning is that it helps us focus on a positive outcome. When we plan chronologically, it's easier to get caught up thinking about all the obstacles that might prevent us from reaching our goal. The Figure 3.9 depicts the steps that have to be followed to create the CMP.

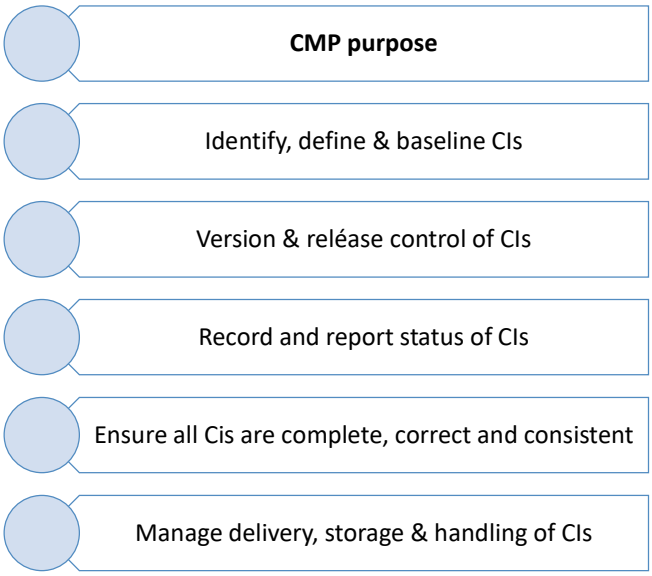


Figure 3.9. CMP steps

As can be seen in Figure 3.9 all the steps are subject to the CIs which have to be selected from the very beginning of the project. In fact, the Configuration Management Plan is a backwards or reverse plan. It has to be created from the very beginning of the project and it has to foresee the way that the target is going to be accomplished. The purpose of the Configuration Management Plan is to describe how configuration management (CM) will be conducted throughout the project lifecycle.

This includes documenting how CM is managed, roles and responsibilities, how configuration item (CI) changes are made, and communicating all aspects of CM to project stakeholders. Without a documented configuration management plan it is likely that CIs may be missed, incomplete, or unnecessary work is done because of a lack or version and document control. While a configuration management plan is important for all projects, this is especially so for software and other information technology (IT) projects (Feiler & Downey, 1990).

3.2.4 Different companies to work together

A consortium allows 2 or more businesses to combine their capabilities when developing and delivering a tender. The primary driver of a consortium approach is that it allows for greater economies of scale, efficiency and effectiveness. A consortium can be made up of

delivery partners from different industry sectors and this offers a great source of competitive advantage.

Airbus Industries was formed in 1970 as a consortium of aerospace manufacturers. The retention of production and engineering assets by the partner companies in effect made Airbus Industries a sales and marketing company. This arrangement led to inefficiencies due to the inherent conflicts of interest that the four partner companies faced; they were both shareholders of, and subcontractors to, the consortium.

The companies collaborated on development of the Airbus range, but guarded the financial details of their own production activities and sought to maximize the transfer prices of their sub-assemblies. In 2001, EADS (created by the merger of French, German and Spanish Airbus partner companies) and BAE Systems (the British partner company) transferred their Airbus production assets to a new company, Airbus SAS. In return, they got 80% and 20% shares respectively. BAE would later sell its share to EADS. The consortiums in general present the following advantages:

- A consortium lets its partners share relevant skills, experience and expertise in such a way that every business complements each another (i.e. in terms of the tender roles and responsibility, and in relation to service delivery).
- Being in the consortium will give you the opportunity to access partner experiences or competencies that you might not have, and which you cannot afford to 'buy in' just to secure the contract.
- Other business partners will have unique selling points (USPs) that you do not have. They may have the capability to deliver in other geographical areas or may have access to data that can give the tender submission a competitive advantage over other submissions.
- Sharing your expertise and capabilities, via the consortium, can increase your chances of success at tender evaluation.
- Business partners can share development costs, which can reduce overheads and the amount of resources required of each business.
- Risk can be spread across business partners.

3.2.4.1. Problem associated to sharing the project

As the scale of our systems increases, and the emergent properties become increasingly important, we now face difficult problems. The structural complexity of our large organizations makes the development process less efficient. Since large-scale systems are generally developed by large and complex organizations, and since these large organizations generally distribute large projects across multiple organizational units and locations, these large projects tend also to have complex structures. This added complexity both complicates the work and takes added resources and time.

The implication for both projects and organizations is that, as they grow, their structure gets progressively more complex, and this increasingly complex structure makes it harder and harder for the developers to do productive work. Finally, at some point, the organization gets so big and so complex that the development groups can no longer get their work done in an orderly, timely, and productive way. Since this is a drastic condition, it is important to understand the mechanisms that cause it.

While companies have always handled big jobs by breaking them into numerous smaller jobs, this can cause problems when the jobs' parts have inter-dependencies among companies. System performance, for example, has always been a problem in these cases. A typical scenario could be the coding from an out-of-date design:

A development team spends two months coding from a baselined design description that, unbeknown to them, has outstanding change requests. When the change requests are finally implemented the design is substantially different. Ten thousand lines of code have to be re-worked.

3.2.4.2. Solution to sharing the project related problems

This loss could have been avoided if the configuration status of the design description was known to the software developers. In order to avoid those issues, the ability to identify the configuration of operational items such as infrastructure, vehicles, equipment and software at a point in time is key. This capability is useful for processes such as problem management, audits, compliance and investigations.

It is crystal clear that a process of recording and organizing the knowledge base necessary for the performance of the project is needed. Besides, this process should provide a highly reliable source of information to support all program/project activities including program management, systems engineering, manufacturing, software development and maintenance, logistic support, modification, and maintenance (McElroy, 2000).

3.2.4.3. Organizational complexity vs recording

In Configuration Management the Configuration Status Accounting (CSA) receives information from the other CM and related activities as the functions are performed. It is constrained only by contractual provisions, which establish the program life cycle phase, tasks to be performed and the organization (Government or contractor) tasked to perform them. In addition to the use of automated configuration management tools, the process is aided or facilitated by the documented CM process and open communications.

The outputs from this activity provide visibility into CM document, activity status and configuration information concerning the product and its documentation. They also include "metrics" developed from the information collected in the CSA system and management "prompts" resulting from analysis of the CM database. CSA provides the recording and reporting of information needed for configuration management including the status of configuration items (CIs), proposed changes and the implementation status of approved changes. The Figure 3.10 depicts the evolution of the information recorded in the configuration management database

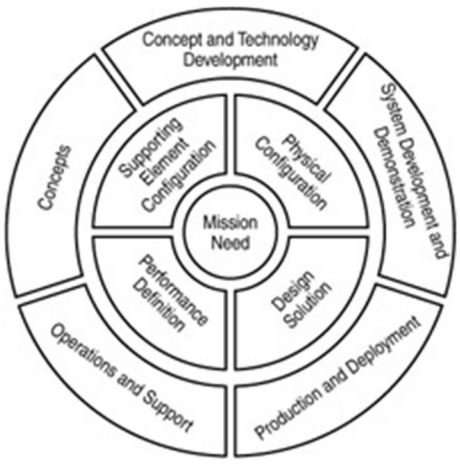


Figure 3.10. CSA Evolution over the System/CI Life Cycle

As can be observed in Figure 3.10 the first stage of the CSA is the storage of the information for the Mission the second one is regarding the physical and the functional configuration for the design and the third one the one related to production operations and deployment. Some of these tasks also may not span the entire life cycle. The allocation of responsibilities within these functions (tailoring) must be accomplished during the CM planning activity and should take into account the degree to which the information technology infrastructure has been upgraded.

All of the information required to accomplish the complete CSA function can be captured and supplied using commercial configuration management and product data management tools. Status accounting provides the means by which the current state of the development can be judged and the history of the development life cycle can be traced. A typical Configuration Status Report might include:

- A list of the configuration items that comprise a baseline.
- The date when each version of each configuration item was baselined.
- A list of the specifications that describe each configuration item.
- The history of baseline changes including rationales for change.
- A list of open change requests by configuration item.
- Deficiencies identified by configuration audits.
- The status of works associated with approved change requests by CI.

Configuration status information is maintained in a Configuration Item Register

3.3 Configuration Management: an step forward towards consistency

As previously mentioned in this section it will be shown how configuration management is able to steer the large-scale projects management focus from the final product to the configuration items increasing the consistency of the project.

3.3.1 Large Scale Project success definition according to Configuration Management

CM is a scientific discipline for establishing and maintaining consistency of a product's performance, functional, and physical attributes with its requirements, design, and operational information from its inception until disposal. The project management was considered successful if the delivered project is performed within the agreed upon deadline, fulfil the agreed upon requirements and within the agreed upon cost.

Although the leaders of small and medium-sized enterprises (SMEs) involved on small projects, still nowadays follow this theory, the academic word and big companies involved in large-scale projects have left it .A 2017 study suggested that the success of any project depends on how well four key aspects are aligned with the contextual dynamics affecting the project, these are referred to as the four P's: (Mesly, 2017)

- Plan: The planning and forecasting activities.
- Process: The overall approach to all activities and project governance.
- People: Including dynamics of how they collaborate and communicate.
- Power: Lines of authority, decision-makers, organograms, policies for implementation and the like.

According to (Ogunlana, 2010) investigations show that other performance indicators such as safety, efficient use of resources, effectiveness, satisfaction of stakeholders, and reduced conflicts and disputes are key. (Serrador & Pinto, 2015) in their work focused on the description of project success introduce the Benefits Realization Management (BRM) which is a group of procedures built to offer consistency between strategy arrangement and performance through guaranteeing the application of the most advantageous proposals.

Their work shows the results of a study to project managers in Brazil, United Kingdom and United States assessing the influence of BRM on project success level. Their outcomes depict that BRM pattern can forecast the project success on the creation of strategic value for the business. Hence, these outputs propose that BRM can be useful in the assistance of the successful performance of business strategies.

(Serrador & Pinto, 2015) checked the result of Agile implementation in companies on two perspectives of project success: efficiency and stakeholder satisfaction versus organizational targets through the information of around 1000 projects covering many industries and countries. They also analyzed the modest influences of topics like recognized quality of the targets of the project, project complexity, and project team knowledge. Their incomes propose that Agile approaches do have a favorable effect on both perspectives of project success (Enck *et al*., 2009).

On the other hand, some global surveys connected project success to concrete elements like leader-member exchange relationship theory (LMX), project culture, and team communication. (Qubaisi *et al*., 2015) spread the work to analyze consistent team communication as an intermediary for the project success. The outputs depict to what extent consistent team communication, uniform project leadership, and harmonious organizational culture affect the project success.

3.3.2 Consistency as a key feature for the Large Scale Project success

Almost 40% of projects do not succeed according to (Mark, 2016, p. 16). For other experts the percentage is even bigger, e.g. (GIORGIO Locatelli *et al*., 2014) states this number of failures reaches around 80% of large-scale energy projects. (Giorgio Locatelli *et al*., 2017) reports that 40% of large scale transport infrastructure projects are over budget and have different performance based on a dataset populated by almost 30 of these megaprojects in Europe.

What is more, when the projects are over they provide lower satisfaction than predicted, e.g. (Voulgaris, 2019) says that in rail projects the number of trips is around 50% smaller than forecasted. Besides of that, for roads the percentage of error in traffic predictions is around 10%. Taken into account the previous point success description, it is necessary to wonder “Why do large scale projects in XXI century fail still?”

According to Flyvbjerg and Green the high rate of megaprojects delivered with cost and time increase as well as low satisfaction has to be admitted. More concretely, (Flyvbjerg, 2014) states that project organization inconsistency is behind the exceeding budgets and

deadlines as well as the decreasing profits. On the other hand, (Green *et al.*, 2015) justifies that a big data project forecast is successful if it is consistent with cumulative knowledge about the present and the past.

Regarding consistency and complexity (Kardes *et al.*, 2013) takes an illustrative method to recognize essential features of megaprojects, components producing unwanted outcomes, and provides a reliable risk management framework and managerial direction for reaching success. Building on the prospect theory, self-justification theory, and sunk cost effect, the author inspects the performance of decision making under risk in large scale projects. He settles that by fostering a consistent risk management method and following regular best practice, level of success and the productivity of multinational projects can be improved. According to (Smith & Lewis, 2011) as projects are more multinational, active, and ambitious, inconsistent requests arise.

The authors analyze the paradox literature, classifying and underlining basic discussions. They introduce a dynamic paradox model of organizing, which pinpoints how consistent cyclical feedbacks to contradictory tensions allow sustainability—peak performance in the present that allows success in the future. That analysis and the model contribute to the creation of a theory of paradox. In the end, the authors arrive at conclusion that a failed project is mainly produced by inconsistent planning and inception.

3.3.3 Definition of Project consistency in Large Scale Project

According to (Qian *et al*., 1990) the project consistency is the especially correspondence or uniformity among all the parts of a complex system. Project consistency is a no so recent concept. Its systematic investigation can trace its roots back to (Bowman, 1963) where some rationale about consistency for decision making in companies and criteria surfaces is provided to support to understand the major ideas introduced. (Tarr & Clarke, 1998) explained that agreement, harmony and compatibility are key characteristics for consistency in the management of requirements and discussed the requirements in terms of both functionality and cross-cutting issues that influence how this operationality is supplied.

Inconsistencies create a serious concern to control in combining modelling scenarios, more concretely in environments with diverse contexts engaged. (Dávid *et al*., 2016) suggests a method for managing inconsistencies. They use a case study of the design of an automated guidedvehicle (AGV) to motivate their work. The AGV is designed to transport payload on a specific trajectory between a set of locations. The drivetrain is fully electrical, using a battery for energy storage and two electric motors driving two wheels. Being a complex mechatronic system, the requirements of the AGV are specified by stakeholders of the different involved domains.

Motor and battery selection activities use databases maintained in Excel files. Since these tools work with different modeling formalisms, reasoning over the consistency of the system as a whole properties poses a complex problem to overcome. By explicitly modeling linguistic and ontological properties and associating them with the engineering activities, patterns of inconsistencies can be identified and handled. They discuss that to completely follow the logic of their incident and effect on the global design, inconsistencies should be figured out in the context of the procedure they arise in.

Therefore, authors suggest a language for modelling processes in agreement with the features of the engineered system. The design of large-scale complex engineered systems requires many developers making consistent decisions over several companies and at diverse positions of the organization chart

Requirements command the development process and are tracked over organizations and along the hierarchical structures of the organization. (Kannan *et al*., 2017) search for essential elements of getting improved system consistency by a means of a value based systems engineering strategy, applying a commercial satellite system as a testbed. Their work proves that guaranteeing system analysis consistency by application of system sensitivities can get through problems related to: feature relationship; wrong system interaction acquisition; and straight change of features to calculate value impact.

The engineering of control systems is a key topic within the engineering of production systems gathering several previous engineering actions. Thus, a high data quality of the previous action has to be guaranteed particularly eluding inconsistencies between supplied groups of engineering data. (Lüder *et al*., 2018) draw an approach relevant to model

engineering area crossing a uniformity pattern to assist an automatic assessment for consistency management. (Durey *et al*., 2019) understands consistency as standardization Based on that, this study will standardize the application of CM in order to provide consistence and as a result success to the project. The author presents the attempts and results introducing consistency to Deepwater projects in the Gulf of Mexico (GOM) and to Shell's broader global Deepwater Organization (DWO). Consistency attempts contain full End-to-End well delivery from engineering design documents, recommended/best practices, operational procedures, workflow processes, after-action-reviews, knowledge sharing, and refreshing standards as required. They stress how the consistency is a key aspect to explain the project performance.

As a summary, the essential characteristics of the consistency of a project introduced in the literature are:

- Consistency is not a new area of discussion, additional research is missed, though;
- Consistency is applicable for the project success, more concretely for large scale projects and projects accomplished in complex engineering environment.

The consistency described in (Kaliappan & Ali, 2018) can be considered as the state of the art in SW development engineering projects. It has the point of view of Model Driven Engineering, where inconsistency between model and its implementation n has huge impact on the software development process in terms of added cost, time and effort. This view is probably adequate for projects delivered in non-complex environments. In this case the company has a deep understanding of the project scope as well as the technologies, the stakeholders and the risks involved.

However, for projects delivered in complex environments, the governance needs to be transformed from “project perspective” to “CI perspective”. CIs are a means to represent software variability As explained in the next section, CM is the methodology to achieve this transformation. By taking advantage of CM the project consistency needs a broader view that lasts more than the project team and the duration of the project itself. Due to their logical grounding, such representations allow for automated reasoning about specific model properties. (Felfernig *et al*., 2018) show how the concepts of conflict and

inconsistency detection and model-based diagnosis can be applied to analyze and improve the quality of a feature model.

The example feature model used in this context is based on the variability information of a real-world event management environment. As in the emblematic example of Software and systems engineering described by (Egyed *et al*., 2018) there is a search on how incremental consistency checking is able to automatically and continuously detect discrepancies among these artefacts. In complex project environments, continuous consistency checking can be a starting point but is not enough.

3.3.4 Consistency in the XXI century

According to above mentioned, still in the XXI century, in spite of the progression of project management (PM) tools and techniques, many projects show inconsistent completion.

Failed projects are frequently handed over in an scenario with quick evolution of inventions; shortened invention life-cycle; high threat of quick obsolescence of the inventions (Kerr et al., 2008), rising level of interoperability between all the subsystems (Geraldi & Adlbrecht, 2007), disproportionate focus on cost decrease (Iacovou *et al.*, 2009); integration concern due to the big amount of companies and technologies engaged (Drouin *et al*., 2012) and finally high number of competitors (Rhodes & Ross, 2010).

These characteristics are normal in megaprojects. According to the Oxford Handbook of Megaproject Management, "Megaprojects are large-scale, complex ventures that typically cost $1 billion or more, take many years to develop and build, involve multiple public and private stakeholders, are transformational, and impact millions of people"

CM is the exercise of dealing with changes methodically thus a system keeps its consistency along the life-cycle. CM applies the policies, procedures, techniques, and tools that manage, assess requested changes, monitor the situation of changes, and keep a record of systems and support documents as the system evolves.

CM programs ensure technical and administrative non-conflicting guidance for the development and compatible application of the procedures, functions, services, tools, processes, and resources necessary to successfully develop and support a complex system. During system development, CM allows program leaders to monitor requirements from product inception until disposal. Due to the fact of changes unavoidably arise in the requirements and design, they must be agreed and reported, providing a precise registration of the system situation.

CM program should be implemented from the system inception until disposal. During the missile competition of the 50s, a lot of prototypes were produced, but these were not consistently documented. This supposed that parts were needless created and included, which caused big expenses and mixed success. The US Dept. of Defense (DoD) took the lead in imposing Configuration Management for hardware. US DoD had to accomplish military programs complying with deadlines, costs, quality. Therefore, it was necessary to be ready to fulfil long term goals with a consistent strategic management of stakeholders.

For the first time, it was not sufficient to fulfil the requirements of the project or final product in the limits of time and money, it was necessary to accomplish it taking into account the whole product lifecycle: interchangeability, operations, maintenance, obsolescence, recycling, etc. In order to fulfil this complicate goal the traditional project management with a final product focus was no longer sufficient. It was essential to change the focus of the project from final product to CI.

CIs management rises the chance of project success. Actually, in spite of failing many projects, there are megaprojects successfully accomplished too: the Doha metro (Shandas *et al*., 2020); Dubai metro and tram (Adela, 2019); Riyadh-Dammam Corridor (Almujibah & Preston, 2019). A common feature of these projects is the implementation of CM fundamentals and methods. Hence, due to the importance of CM for large scale projects, this chapter´s purpose is to describe how CM can improve the consistency of the project by transforming its focus from the final product to the CI.

After the Fukushima mishap in 2011, safety has arisen as a key political and social concern in the nuclear sector. As a result, the relevance of CM is highlighted as a means to guarantee the reliability and safety of nuclear centrals. In spite of that, CM application in the nuclear

business is in a theoretical phase so far. As an effort to address that concern, (Kang et al., 2019) did the following achievements:

- Present an evaluation approach for prioritizing the relevance of CM implementation sections depending on the physical subjects of Nuclear Power Plants (NPP) structures, systems, components (SSCs).
- The presented approach consists of an 'spread CM framework' with more details and 'assessment model' based on CM functions.
- That work created a SSCs record by assessing Design Control Document (DCD) of Westinghouse AP1000.
- The theory of CM in that research covers the whole NPP project life-cycle as a means to foster the practical CM application.
- The outputs of case-study conducted in by Kang and Jung would bring the SSCs priorities and rules for practical CM for sustainable NPP facilities.

This example shows (1) that the lack of implementation of CM principles and practices can produce many problems and (2) that when CM is properly implemented such issues can be sorted out.

3.3.5 Configuration Management as a support of the Large Scale Project Management

Configuration Management started in the United States Department of Defense in the 1950s as a technical management scientific field of endeavor for hardware material items—and it is now an everyday business in nearly each industrial sector. The CM scientific field's topics contain systems engineering (SE), Integrated Logistics Support (ILS), Capability Maturity Model Integration (CMMI), ISO 9000, Prince2 project management method, COBIT, Information Technology Infrastructure Library (ITIL), product lifecycle management, and Application Lifecycle Management. Many of these functions and models have re-described CM from its conventional holistic discipline to technical management.

Some managers deal with CM as being parallel to a librarian activity, and start change control or change management as an isolate discipline. It is quite complicate to generate just one definition of CM because the literature supplies many different definitions, even

if the most comprehensive and agreed is that of MIL-HDBK-61A: “Configuration management (CM) is a systems engineering process for establishing and maintaining consistency of a product's performance, functional, and physical attributes with its requirements, design, ad operational information throughout its life.” (*MIL-HDBK-61 A CONFIGURATION MANAGEMENT GUIDANCE*, n.d.)

The project that is ruled by CM has a hierarchical nature with the final product split into CIs. The CIs can be hardware, software, firmware, personnel, facilities, data, materials, services, and processes (Handbook, 2001). CM guarantees that the engagement and interfaces between them are consistent (Army, 2003, p.). Even if CM is an iterative and recursive multidisciplinary discipline capable to control every phase of system's life cycle (Command, 2015), the main advantages are at the earlier project stages (Doreswamy, 2013). These phases are the project definition (scope management), project stakeholder management and project planning. Thus, CM is especially helpful in large-scale projects as described in section 2.3. Many experts have studied the importance of CM on project consistency, highlighting that it has a decisive influence on the project accomplishment as can be observed in the Table 3.2.

Table 3.2. Impact of CM

		Impact of CM on project´s							
		Consistency	**Scope**	**Budget**	**Quality**	**Schedule**	**Risk**	**Satisfaction**	
								Customer	**Team**
Authors	Wateridge (1999)		X						
	Ali & Kidd (2014)	X	X	X	X		X		
	Berland (2002)		X	X		X		X	X
	Friedrich (2005)	X				X			
	Globerson (1994)	X	X						
	Kerravala (2004)	X	X	X	X	X			
	Xu et al. (2013)	X	X	X	X	X		X	
	Do (1999)	X	X	X		X			
	Kidd (2001)	X							

As it can seen in the Table 3.2, Configuration Management has a positive influence on project management in many areas. However, according to most of the authors where its greatest contribution comes is in consistency understood as an agreement or harmony of parts or features to one another or a whole.

3.3.6 Configuration Management standards

The CM discipline evolved into its own technical approach around the late 1960s when the Department of Defense created a series of military standards called the "480 series" (i.e., MIL-STD-480, MIL-STD-481 and MIL-STD-483) that were consequently released in the 1970s. In 1991, the "480 series" was merged into a single standard known as the MIL–STD–973 that was then substituted by MIL–HDBK–61 in accordance with a general Department of Defense target that decreased the amount of military standards in favor of industry technical standards assisted by standards developing organizations (SDO). This meant the starting point of what has now developed into the most broadly spread and agreed standard on CM, ANSI–EIA–649–1998.

MIL-STD-973 defines CM requirements which are to be selectively implemented, as required, during the life cycle of any CI:

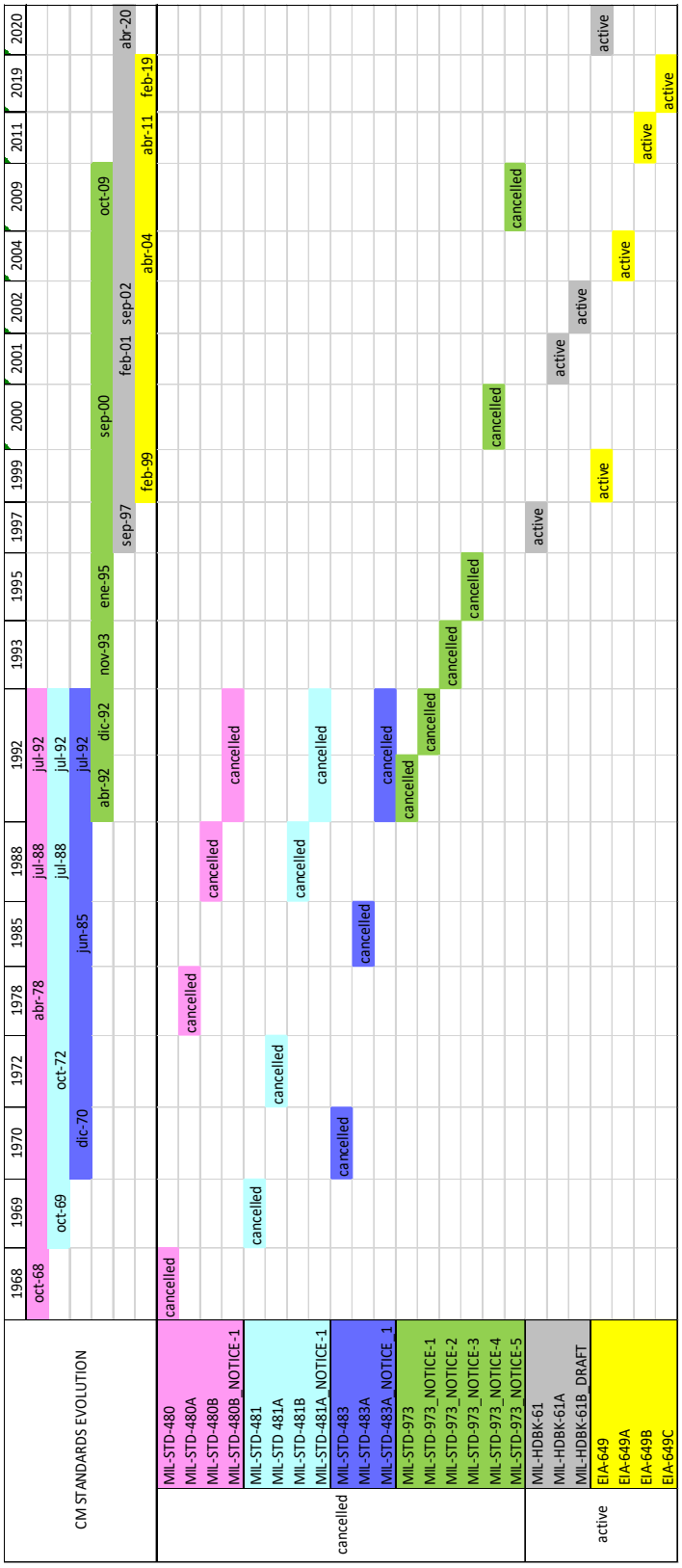
- Created entirely or partially with Government finance, including non-developmental items when the development of technical data is necessary to assist commercial equipment of software
- Selected for CM for reason of integration, logistics support, or interface control.

MIL-HDBK-61A supplies direction and information to Department of Defense procurement managers, logistics managers, and other groups attributed responsibility for CM. Its target is to support them in planning for and implementing efficient CM activities and practices throughout all life cycle phases of defense systems and CIs. It assists acquisition founded on performance specifications, and the application of industry standards and methods to the greatest practicable degree.

ANSI/EIA-649 was created in 1994, when the Electronic Industries Alliance's (EIA) G-33 Committee on Data and Configuration Management started a duty to create an industry CM standard. Their target was to handle overall requirements, principles, and best practices in CM without forcing the application of specific nomenclature or methods on how CM should be applied in any concrete environment. This standard affirms that when CM fundamentals are implemented using efficient practices, return on investment is maximized and product life cycle costs are decreased.

When CM is efficiently and consistently implemented, it brings a positive influence on product quality, cost, and schedule by guaranteeing consistency of a product's performance, functional and physical attributes with its requirements, design, and operational information. Table 3.3 shows the evolution of all the CM standards since its earliest beginnings to contemporary times.

Table 3.3. CM standards evolution



As can be seen in the Table 3.3, the standardization process began more than 50 years ago and in this time many standards have been created. Currently just the MIL-HDBK-61 for military projects and the EIA-649 for commercial projects are active.

3.3.7 Impact of Configuration Management

There are a lot of intersections between CM and other disciplines, more concretely with PM. (Allan, 1997) discusses in detail the critical relationships and interconnections between the computer system development projects and CM. According to (Ali & Kidd, 2014) CM is an efficient pattern for project managers to take advantage of a formalized approach in order that they can manage status and changes to it along the lifecycle. (T. F. Burgess *et al.*, 2005) discovers an intersection between CM and PM in risk management within the aeronautic sector. According to the author the impact of CM will be positive if information technology and people are activated to upgrade CM.

Since the Fukushima mishap in 2011, nuclear safety has arose as a key political and social concern. Under this situation, (Kang & Jung, 2015) stress the weight of CM in order to guarantee the reliability and safety of a nuclear program. However, CM is in its initial phase in the nuclear industry so far due to the ambiguity of CM definitions, deficit of CM procedures, lack of computerized systems and insufficiency of CM professionals.

Shortly CM is a multidisciplinary discipline dealing with both technical and managerial features. The technical concern is based on the CI, which has substantial technical components. CM also has a management concern, addressing the consistency of the technical work (Myrodia *et al.*, 2019). This issue is seen in the integration in CM of processes to manage the projects (ISO, n.d., p. 10007). CM joins traditional, technical and managerial disciplines into a holistic CIs approach (Van Ommering, 2001). Nearly all of the profitable projects in complex environments have implemented several fundamentals and practices which have the roots in CM. In addition to the examples introduced the most well-known and best tested project enhanced in agreement with CM is the International Space Station. The International Space Station is a research facility in space in which the spacemen do experiments close to zero gravity, or “microgravity” (Oman *et al.*, 2003). This project is the output of the collective effort of 5 aerospace agencies from 15 countries (Robinson *et al.*, 2007). Very quickly, NASA (the leading organization) and its partners

found how to arrange and incorporate all of the international partners and their strongly interconnected modules, diverse cultures, technical languages and operational viewpoints on risks and safety. Meantime NASA had to deal with concerns related to technical obsolescence, logistics, technology gaps and important career progression of its employees. NASA had to develop expertise in supporting its CM discipline while aligning to the realities of a complex external environment together with international politics across many partner nations. CM requires more in the early stages of the project life cycle. It is key to obtain a pristine definition of objectives, roles, responsibilities and requirements (Doreswamy, 2013).

For example in the West Coast Route Modernization a new documentation approach was introduced to promote readability, using a simple and concrete language with unique identifiers and traceability (Loubersac & Halliday, 2001). CM needs the robust implications of all stakeholders throughout the whole project.

3.4 Conclusions

CM is a managerial and technical methodology developed in the last 50 years to improve the consistency (and hence the performance) of large scale-projects designed and delivered in complex environments. CM achieves these results, transforming the consistency from the project and pure "project management" to a more holistic items view of "CI management". The literature review reveals that despite consistency being one of the key factors influencing project performance, its optimal form has not been identified yet.

Nevertheless, CM has emerged as an important technique to transform the consistency in complex project environments. Configuration management generally brings consistency to projects as seen in the previous chapter. In addition, and more specifically, this discipline manages to cover with its tools and methodology certain deficiencies inherent in large projects. Despite all these advantages, the aforementioned discipline is based on the configuration items and an erroneous selection of them would ruin any positive contribution to the project. Therefore, and given the lack of methodology that exists when identifying these elements, the proposal of this research is to process the selection of configuration elements in order to provide tighter control of configuration management.

Chapter 4

Methodology

This chapter, on reporting and discussing the findings, deals with the core of the thesis. This is particularly important as it allows demonstrating how the research contributes to the science (by creating a different method to procedurize a particular activity. This chapter is mainly the presentation, description and interpretation of the created methodology in the discipline of the CM.

Configuration management (CM) is a Systems Engineering process commonly applied in multidisciplinary large-scale engineering projects of complex systems (Brouse, 2009). It is used for establishing and maintaining consistency of a product´s performance, functional, and physical attributes with its requirements, design, and operational information throughout its life (Carnduff & Goonetillake, 2004).

Therefore, during system development, CM allows program management to track the initial requirements throughout all the changes and modifications during the life-cycle from the acceptance trough manufacturing and maintenance (Chen *et al*., 2010). When CM is used over the life cycle of a system, gives clarity and control of its performance, functional, and physical features. CM proves that a system develops as planned, and is identified and documented in detail to guide its projected life cycle (Tang *et al*., 2015).

The CM discipline facilitates orderly management of system information and system changes for such beneficial purposes as to revise capability, improving performance, reliability, or maintainability; increasing life; decreasing cost, risk and liability; or amending defects (Dart, 1991). As we can see in Figure 4.1 all the main project stakeholders are involved in the CM along the product lifecycle

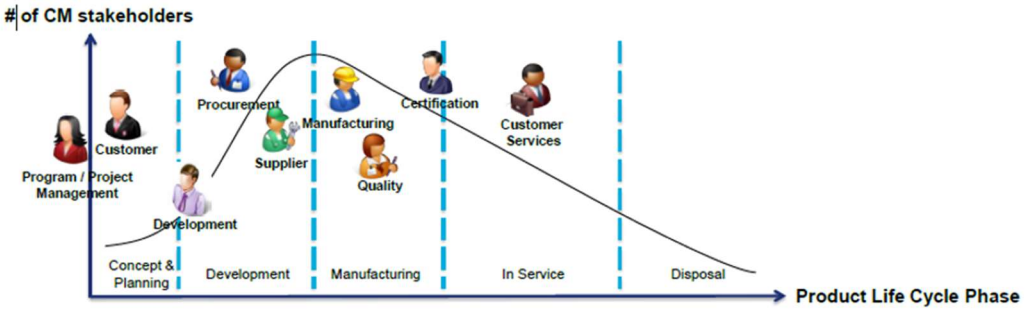


Figure 4.1. CM stakeholders during the product life cycle

However, nowadays the application of CM is somehow the combined outcome of experience and company knowledge and no scientific method neither systematically structured knowledge is followed in its application (Ambriola *et al*., 1990).

The consequences of this lack of organized procedures for CM application in complex engineering projects leads to several problems with relevant economic consequences (Jaime Larumbe *et al.*, 2020). The Figure 4.2 shows the different engineering efforts in the development phase both with adequate CM and without.

Figure 4.2. Engineering efforts with adequate CM versus engineering efforts without CM

Therefore, identifying and categorizing each and every part that composes the final product, as well as, the relationship between them is of key importance (Larsson & Crnkovic, 1999). This information should be provided by CM through the Configuration Items (CI) (J. Larumbe, 2017).

The term CI refers to the fundamental structural unit of a configuration management system (Larsson, 2000). The CM system oversees the life of the CIs through a combination of processes and tools by implementing and enabling the elements for identification, change management, status accounting, and audits(Larsson & Crnkovic, 2001).

CIs can be considered as the Gordian knot of all this five disciplines composing the CM. CIs are the basic units of configuration management. They may vary widely in complexity, size and type, from an aircraft, ship, tank, electronic system or software program to a test meter or a round of ammunition.

Regardless of form, size or complexity, the configuration of a CI is documented and controlled. CI selection separates system components into identifiable subsets for the purpose of managing further development. For each CI:

- There will be associated configuration documentation (which may range from a performance specification to a detailed drawing to a commercial item description)
- Configuration changes will be controlled
- Configuration status accounting records will be maintained
- Configuration audits will be conducted to verify performance and product configuration (unless the CI has an already established product baseline)

On the other hand, in configuration management, a "baseline" is an agreed description of the attributes of a product, at a particular date, which serves as a basis for tracing changes.

Hence, an `as required baseline´ is a snapshot in time that represents the agreed-upon, reviewed, and approved set of requirements committed to a specific product release. Accordingly, an `as designed baseline´ is a snapshot in time that represents the agreed-upon, reviewed, and approved items identification describing functional and physical characteristics, interface requirements, design constraints and verification required committed to a specific product release (Lin & Reiss, 1996).

Without this information, it is impossible to ensure that the impact of a change has been taken into a full consideration.

The requirements baseline is an official document of the project signed by the client and by the development company that determines and defines the physical and functional requirements that the final product will meet. It must be signed after the requirements freeze and before starting the design.

On the other side, the design baseline is an official document also signed by both parties, but which in this case determines and defines the documentation according to which the final product will be developed (plans, specifications, bills of materials and so forth). The proper time to sign this document is after critical design and before starting the manufacture or development. The design baseline must contain the requirements baseline since it must outline the requirements and offer a document that shows that the aforementioned requirement will be fulfilled.

CM's job is to ensure that previously mentioned traceability is maintained and any non-compliance in this biunivocal relationship between requirements and documentation must be justified and defined by means of a change through the official change management process with the subsequent final approval from the client.

As a conclusion, it is necessary to develop a *configuration management application methodology* that targets minimizing the risk in management of large and complex engineering projects by ensuring the complete identification of the impact of all changes as well as the traceability of the requirements (Ackoff, R. L., n.d.) (Narain, 2005).

Therefore, this chapter presents a systematic procedure to build and organize knowledge in the form of testable explanations and predictions about the final product. Selection and identification of CI for a particular project can be seen as the first step in developing an overall architecture of the product from the top-down (Nguyen *et al*., 2005).

This procedure is the result of over 10 years of experience of the first author in the domain of Configuration Management in some of the most ambitious projects and for some of the most competitive multinationals of the world like the Doha metro project for MITSUBISHI, KSA NBS Programme (Kingdom of Saudi Arabia Northern Border Security) for AIRBUS, TWINDEXX Swiss Express for SBB (Swiss Federal Railways) for

BOMBARDIER, and the SCOUT SV Programme (Family of Armoured Fighting Vehicles for the British Army) for GENERAL DYNAMICS.

4.1 Method description

Next a method to calculate the configuration elements in a methodological and procedural way will be described. This method is mainly based on 3 stages. The 1st one is based on the functional requirements or in other words *understanding what customer wants*. The 2nd one is based on the allocated or as designed baseline or in other words *how the customer wants are going to be satisfied*. The last stage will collect data from the previous steps and it will proceed to *calculate the value as a configuration item of all the elements that make up a final product* regardless of the complexity and nature (HW, SW, FW ...).

More concretely, the methodology consists of the following stages:

1. *Creation of the functional requirements matrix.* This is done by the mapping of the `as designed baseline´ with the `as required baseline´. More concretely, it consists on the mapping of the Bill of Material (BoM) with the functional requirements (FC). The bill of materials is a list of the raw materials, sub-assemblies, intermediate assemblies, sub-components, parts, and the quantities of each needed to manufacture a product. All manufacturers building products, regardless of their industry, get started by creating a bill of materials (BoM). As defined in requirements engineering, functional requirements specify concrete results of a system. Therefore, the necessary functional requirements for the creation of the different items will be identified. Besides, how these functional requirements are affected by time and money will be calculated. The result of this matrix is the quantification of time and budget needed to develop the items of the final product from the functional point of view.

2. *Creation of the configuration requirements matrix.* For this matrix, it will be necessary to identify the configuration requirements (CR). For example, these requirements should give the criteria to identify:

    - Items to be placed under government program office CM.

- Associated artifacts to be contained in the program's technical baselines.
- Systems or sub-systems to be recycled by other programs.
- …

Once the CR have been agreed and even quantified, the configuration manager will proceed to map the `as designed baseline´ with these CR. In other words, it consists on the mapping of the BoM with the CR of the project. The output of this matrix will be the categorization of the relevance of different items (F) in the final product from the point of view of the configuration.

3. *Calculation of the weight (W) that every item composing the final product has as CI.* It means the weight that all the items have from the Configuration Managenet perspective or in other words how susceptible are to be under the most strict Configuration Mangement control.

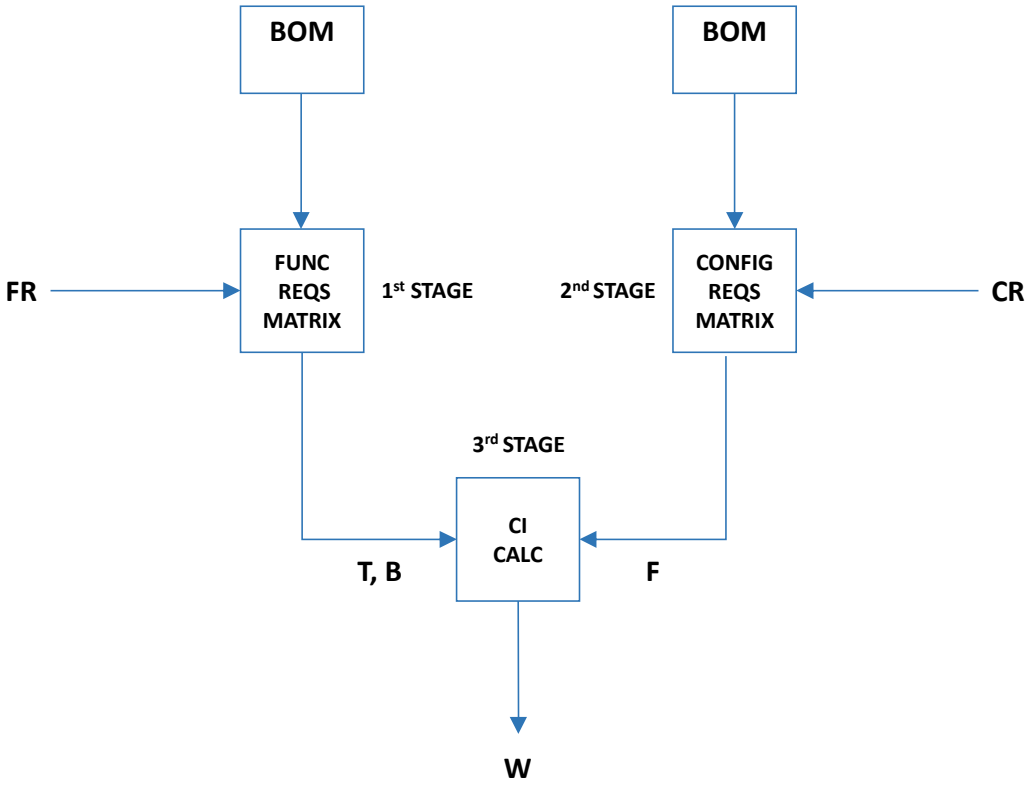


Figure 4.3. Methodology flowchart

Figure 4.3 shows the flowchart of the methodology. As can be seen, there are 3 stages in the methodology where *the final output is the categorization of the CI based on the corresponding calculated weights*.

The categorization of the CI according to the importance and potential impact is a useful tool for determining the appropriate level of identification marking of all the components of the product depending on the value W. All the items identified as CIs in the project will have a unique identity so they will have to be serialized and the rest will be identified with lot numbers.

Next, each point of the methodology will be discussed on detail. To illustrate the explanation, the sample case of a coffee maker will be used (see Figure 4.4), where all the information and activities along the product life cycle of the coffee maker will be outlined. The coffee maker is selected because it is a regular day product that anyone can understand easily the functioning of its components and subsystems and at the same time contains certain level of complexity that make necessary the CM.

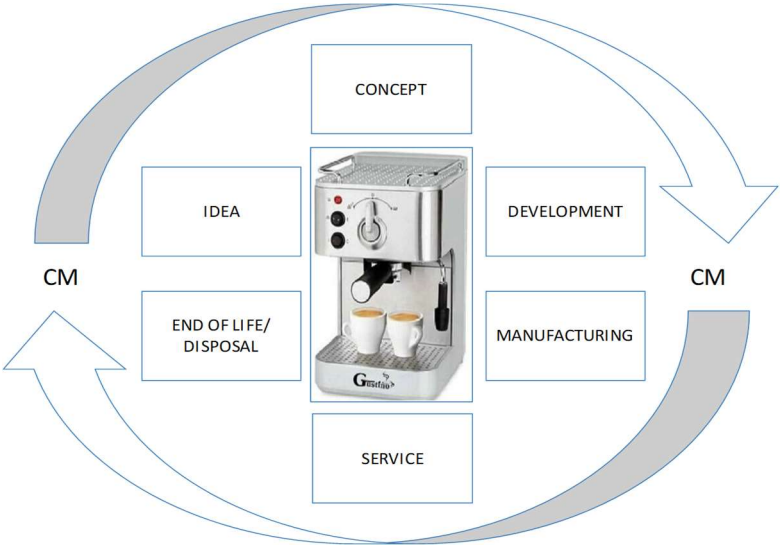


Figure 4.4. CM through the life cycle of a coffee maker

As can be seen in Figure 4.4 is applied all over the life cycle of the product from its inception until the end of life. All along this period traceability among requirements among requirements, specifications, drawings, versions, parts and changes must be assured since

at the end of life product, CM manager should be capable of identifying which parts can be re-used, recycled, and so on.

The Figure 4.5 shows the coffee maker label example. The coffee maker itself is a CI because of that it has serial number.

Figure 4.5. Label of a coffee maker

As can be noticed in Figure 4.5 all the CIs should include in their labels at least the S/N, identification name and P/N.

4.1.1 Functional requirements matrix

As explained, the first stage in this methodology is the calculation of the functional requirements matrix. For that, the `as required baseline´ and the `as designed baseline´ have to be created and then both baselines have to be mapped. Next, the procedure is described in detail. In the following picture Figure 4.6 can be seen the baselining process of the coffee maker.

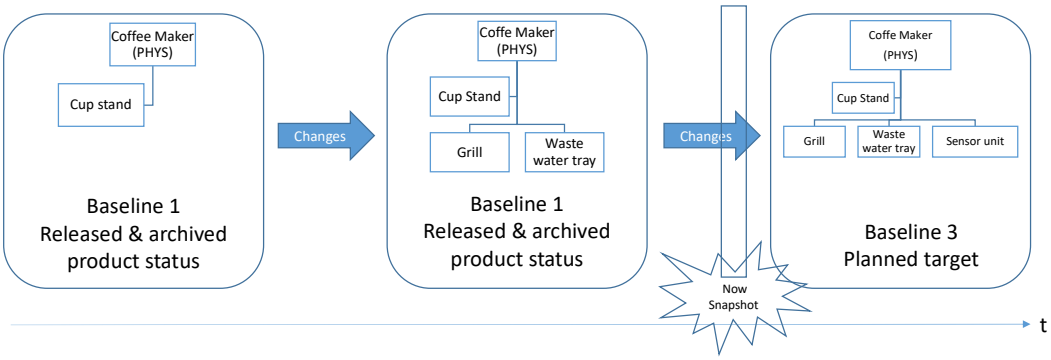


Figure 4.6. Baseline of a coffee maker

As can be noticed in Figure 4.6 baseline 3 is the planned target baseline. The development process of the coffee maker started in the baseline 1 where there was just a cup stand. After some changes the coffee maker development process moved to baseline 2 where it was added the grill and the waste water tray. As per the example the development process is supposed to be somewhere between the baseline 2 and the planned target.

4.1.1.1. As required baseline

When stakeholders "sign off" on requirements, what they're really doing is agreeing and committing to a specific requirements baseline (whether they think of it in those terms or not). Typically, a baseline is also given a unique name so that all the project participants can refer to it unambiguously. Besides good configuration management practices allow the team to reconstruct accurately any previous baseline and all its components. The `as required baseline´ explicitly identifies only those requirements that the project will implement. A baseline is not a tangible item, but rather a defined list of items.

The requirements baseline consists of a number of specifications that are harmonized to articulate the expected functional, performance, physical, and quality characteristics to be exhibited by the product, system or sub-system. This set of requirements focuses on the activities that elaborate the system and the definition of the acceptance test scenarios and procedures. The requirements are utilized during the functional and physical configuration audits as the standard against which system implementation is assessed to determine its quality of workmanship. The requirements baseline involves the following relationships and dependencies with stakeholder needs or other elements of the PBS:

a) *Development costs versus timeliness (Stakeholder Needs).* The requirements will ultimately determine the cost of the development effort and will impact the product delivery milestones. Determining how to balance the system requirements so that the development project remains stable and can proceed decisively toward conclusion is contingent on the accuracy and reasonableness of the requirements baseline to establish the conditions for the project to succeed.

b) *System requirement appropriateness (functional architecture).* The suitability, correctness, thoroughness, and precision of product requirements will impact the development effort. The breadth of the requirements will affect the product success in the marketplace. The system product requirements are interpreted and converted into the functional architecture. Vague or ambiguous requirements may obscure the optimal composition of the functional architecture in terms of orderly arrangement, complexity, and suitability. Requirements that are overstated will drive up development costs and threaten to increase product complexity.

c) *Scope of the test and evaluation effort (test and evaluation).* The requirements baseline establishes the scope of the test and evaluation effort for both the system and the post-development system sustainment processes. Permitting the requirements to be excessive will significantly increase the number and complexity of test cases and scenarios. The testing effort may consume a significant amount of project resources in terms of budget, tools, equipment, and schedule.

d) *Scope of the post-development processes (post-development processes).* Increased product complexity impacts ease-of-use and training demands for users, as well as product and customer support costs.

Once the requirements have been defined based on the baselined scope, they must be agreed by all key stakeholders and formally baselined and controlled. Baselining the requirements before the design phase puts the requirements under configuration control and the resulting configuration control process. Any changes proposed to the requirements then must go to the configuration control board who will evaluate each change for feasibility; impacts to cost, schedule, and risk; validate the change against the base lined scope, and formally

approve/reject the change. A change control process is not about stifling change; it's about providing decision-makers with the information that will let them make timely and appropriate decisions to modify the planned functionality. That planned functionality is the baseline. Whereas the scope definition distinguishes what's in from what's out, the requirements baseline explicitly identifies only those requirement specifications that the project will implement.

4.1.1.2. Functional requirements

In Systems Engineering and in Software engineering, a functional requirement defines the functionality of a system or its components. A function is described as a set of inputs, usability, and outputs. Functional requirements may be calculus, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish (Dart, 1990).

As defined in requirements engineering, functional requirements specify particular results of a system. Functional requirements drive the application architecture of a system. A typical functional requirement will contain a unique name and number, a brief summary, and a rationale. This information is used to help the stakeholders understand why the requirement is needed, and to track the requirement through the development of the system. Figure 4.7 shows the functional requirements and the physical requirements of the coffee maker.

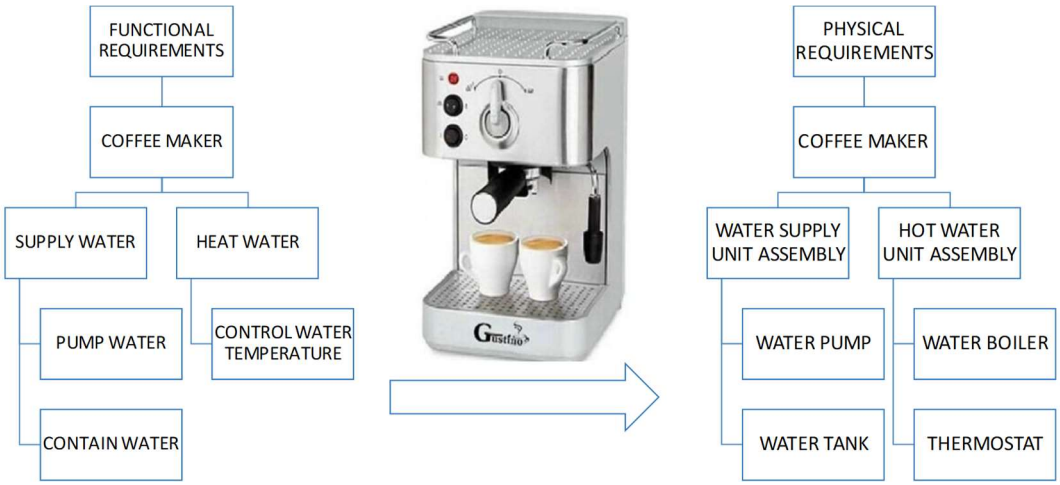


Figure 4.7. Functional requirements for a coffee maker

As can be seen in Figure 4.7, the coffee maker has functional and physical requirements that have to be mapped. As an example, the water supply unit assembly which is a physical requirement is supposed to supply the water which is a functional requirement of this unit.

4.1.1.3. Functional requirements prioritization

During requirements prioritization all stakeholders assess the requirements and estimate the value, risk, and effort needed to fulfil each requirement. The result of this step is an ordered list of ranked requirements. The value assigned to each functional requirement represents the importance of each requirement to the stakeholders; high-value requirements need more detailed tracing than low-value requirements.

This is because they represent the core functionality of the system and trace information of the latter is more important than trace information of low-value functional requirements. Hence, in order to prioritize the functional requirements, time and budget are estimated for every single requirement. In Figure 4.8 can be seen the main baselines of the coffee maker.

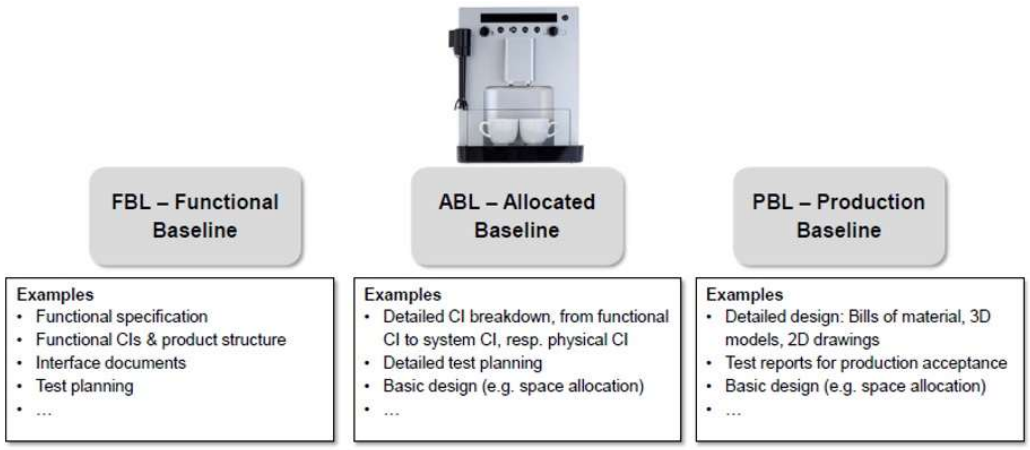


Figure 4.8. Main baselines in a coffee maker

As can be observed in the Figure 4.8 the typical documentation that the functional baseline should contain is: functional specification, functional CIs and product structure, interface documents, test planning and so on. On the other hand, the typical documentation that the allocated baseline should contain is detailed CI breakdown, detailed test planning, basic

design and so on. Finally, the production baseline should contain at least the following documentation: detailed design (bill of materials, 3D models, 2D drawings…), test reports for production acceptance, basic design (e.g. space allocation.)

4.1.1.4. As designed baseline

The `as designed baseline´ or allocated baseline is the approved performance-oriented configuration documentation for a CI to be developed. It describes the functional, performance, and interface characteristics that are designed to demonstrate achievement of those specified characteristics. The `as designed baseline´ extends the top-level performance requirements of the functional baseline to sufficient detail for defining the functional and performance characteristics and for initiating detailed design for a CI.

The as designed baseline is usually controlled by the design organization until all design requirements have been verified. The as designed baseline is typically established at the successful completion of the Preliminary Design Review. Prior to Critical Design Review, the company normally reviews design output for conformance to design requirements through incremental deliveries of engineering data. Company control of the as designed baseline occurs through review of the engineering deliveries as data items.

Complex products such as automobiles, aircraft, and major capital equipment and systems are made of thousands to hundreds of thousands of parts. In addition, there are related tooling, fixtures, gauges, templates, test equipment and software. This fact leads to the conclusion that a company may need to evaluate and process several thousands of engineering changes for a single complex product.

Besides, the manufacturer must assure at any time of the life cycle that the as-designed configuration satisfies every single functional requirement. Figure 4.9 depicts that for every major change it is necessary to a PDCA analysis:

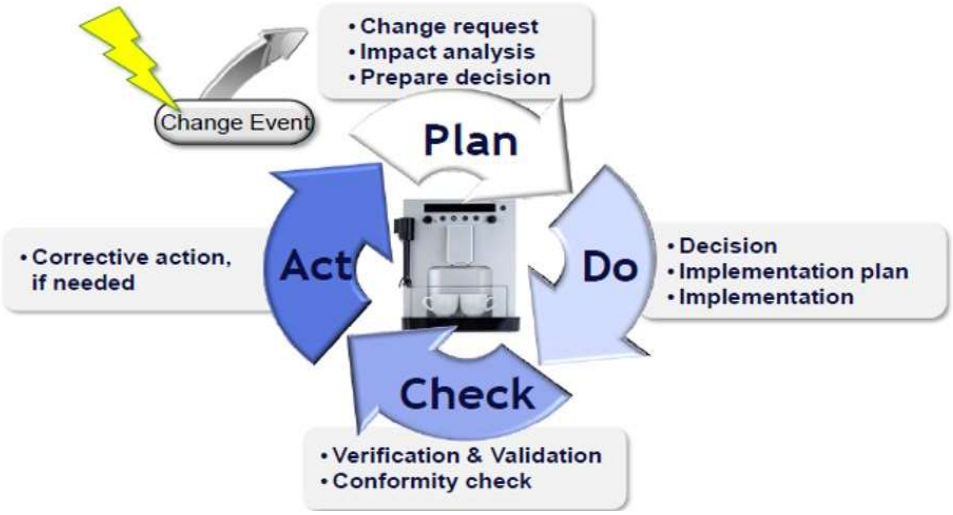


Figure 4.9. PDCA for the change process in a coffee maker

As it was advanced in the previous chapter and can be seen in Figure 4.9, once a change event occurs a change request must be raised, an impact analysis must be done and a decision must be taken. This 3 steps corresponds to the `plan´ of the PDCA analysis. After that, once the decision is taken the implementation plan has to be created and the change has to be implemented. This part corresponds to the `do´ of the PDCA analysis. After implementing the change it is time for the verification, validation and conformity check. This activities are involved in the `check´ of the PDCA. Finally, the corrective action should be contained in the `act´ part of the PDCA.

The coffee maker might be affected by the following changes:

- Law: Reduce lead concentration
  "A safety study shows: Doses of lead are released into drinks after being cleaned. These doses have to be reduced due to possible risks to health."
- Market: Use tall cups/ glasses
  "A market study shows: End users would like to use a broader range of cups and glasses, e.g. for tall latte macchiato."
- R&D: Improved milk foam unit
  "R&D proposes an improved milk foam unit which is easier to clean and produces finer foam."

The change impact analysis is “the heart” of the change process: All impacts of a change shall be considered in order to prepare the implementation decision to guarantee that the hardware and software being delivered (`as-built configuration´) corresponds to the approved as-designed configuration according to Table 2.2 and Figure 2.8. The engineering bill of material or as-designed configuration is a collection of engineering drawings and parts lists linked together in a drawing tree or engineering product structure.

This representation of how materials, components, sub-assemblies and assemblies come together in the final product is based on the designers’ visualization of the product and its stages of production or based on a functional subsystems-oriented view of product assemblies. An example of consistency among the requirements and the specifications is presented in Figure 4.10

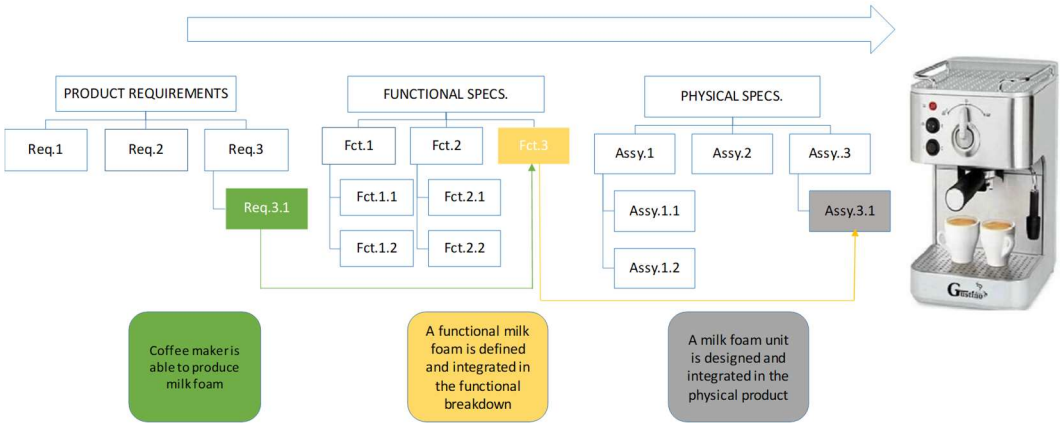


Figure 4.10. Consistency in a coffee maker

As can be understood from Figure 4.10 requirement 3.1 consists on producing milk foam. Therefore, it is created the functional specification 3 defining a functional milk foam and integrating it in the functional breakdown. As a result a milk foam unit has to be designed and integrated in the physical product. The relation between requirements, specifications, and parts must be bi-univocal since all the requirements and functions must be fulfilled by the specifications of the parts and the other way around all the specifications of the parts must fulfil a requirement or function according to the requirements baseline.

4.1.1.5. Bill of materials

A bill of materials or product structure (sometimes bill of material, BoM or associated list) is a list of the raw materials, sub-assemblies, intermediate assemblies, sub-components, parts, and their respective quantities needed to manufacture a product. A BoM may be used for communication between manufacturing partners or confined to a single manufacturing plant. Besides, a BoM is often tied to a production order whose issuance may generate reservations for components stock or requisitions for components that are not in stock. An engineering bill of materials (EBoM) is a type of bill of materials (BoM) reflecting the product as designed by engineering, referred to as the "as-designed" bill of materials.

4.1.1.6. Functional requirements mapping

The first step for the functional requirement mapping is to assign time to every single functional requirement. The objective is to know how long it will take to meet every functional requirement and it can be computed as

$$\boldsymbol{t_{FR_T} = \sum_{j=1}^{j=n} t_{FR_j},} \qquad \textbf{4.1}$$

where t_{FR_T} is the necessary time to meet the j_{th} functional requirement, n is the total number of functional requirements and t_{FR_j} is the total time necessary to meet all the functional requirements. Once the time needs have been considered, functional requirements and items have to be mapped. Therefore, time has to be allocated to every single item as

$$\boldsymbol{t_{FR_j} = \sum_{i=1}^{i=m} t_{ij},} \qquad \textbf{4.2}$$

here t_{ij} is the particular time that i_{th} item needs to meet j_{th} requirement and where m is the total number of items composing the BoM. Hence, the total time to develop every single item according to the functional requirements can be computed as the sum of all the partial times necessary to fulfil with all the requirements according to

$$\boldsymbol{t_{I_i} = \sum_{j=1}^{j=n} t_{ij},} \quad \textbf{4.3}$$

where t_{I_i} is the total time that i_{th} item needs to meet all the functional requirements. Therefore, the total time dedicated to achieving the functional requirements can be calculated as the sum of all the individual times that each item needs to fulfil the functional requirements. This can be calculated as

$$\boldsymbol{t_{FR_T} = \sum_{i=1}^{i=m} t_{I_i},} \quad \textbf{4.4}$$

Once the needed time is calculated, the functional requirements are mapped with the items of the BoM producing the functional requirements matrix shown in Table 3.1. The output of this Table 4.1 is the time needed by all the items to meet the functional requirements.

Table 4.1. Functional requirements matrix with times

	FR_1	**FR_2**	**FR_3**	**...**	**FR_{n-1}**	**FR_n**	**TOTAL**
I_1	t_{11}	t_{12}	t_{13}	...	$t_{1,n-1}$	$t_{1,n}$	t_{I_1}
I_2	t_{21}	t_{22}	t_{23}	...	$t_{2,n-1}$	$t_{2,n}$	t_{I_2}
I_3	t_{31}	t_{32}	t_{33}	...	$t_{3,n-1}$	$t_{3,n}$	t_{I_3}
⋮	⋮	⋮	⋮	... ...	⋮	⋮	⋮
I_{m-1}	$t_{m-1,1}$	$t_{m-1,2}$	$t_{m-1,3}$	...	$t_{m-1,n-1}$	$t_{m-1,n}$	$t_{I_{m-1}}$
I_m	$t_{m,1}$	$t_{m,2}$	$t_{m,3}$	...	$t_{m,n-1}$	$t_{m,n}$	t_{I_m}
TOTAL	t_{FR_1}	t_{FR_2}	t_{FR_3}	...	$t_{FR_{n-1}}$	t_{FR_n}	t_{FR_T}

As can be seen in Table 4.1, all the necessary times to meet all functional requirements are identified in the functional requirements matrix. Once the influence of time has been evaluated, the cost in terms of money/budget has to be evaluated as well, according to

$$\boldsymbol{b_{FR_T} = \sum_{j=1}^{j=n} b_{FR_j},} \quad \textbf{4.5}$$

where b_{FR_j} is the budget necessary to meet the j_{th} functional requirement and b_{FR_T} is the total budget necessary to meet all the functional requirements. Next, functional requirements and items have to be mapped and budgets have to be allocated to every single item as

$$\boldsymbol{b_{FR_j} = \sum_{i=1}^{i=m} b_{ij},} \quad \textbf{4.6}$$

where b_{ij} is the budget that ith item needs to meet i_{th} requirement and $b_{FR_j}$is the budget necessary to meet the j_{th} functional requirement. Hence, the total budget to develop every single item according to the functional requirements can be computed as the sum of all the partial budgets needed to fulfil all the requirements according to

$$\boldsymbol{b_{I_i} = \sum_{j=1}^{j=n} b_{ij},} \quad \textbf{4.7}$$

where b_{I_i} is the total time that j_{th} item needs to meet all the functional requirements. Consequently, the total budget dedicated to the functional requirements b_{FR_T} can be calculated as the sum of all the partial budgets that all the items need to fulfil the functional requirements b_{I_i} according to

$$\boldsymbol{b_{FR_T} = \sum_{i=1}^{i=m} b_{I_i},} \quad \textbf{4.8}$$

Hence, the functional requirements are mapped with the items of the BoM producing the functional requirements matrix shown in Table 4.2. The output of this Table 3.2 being the functional requirements matrix is the needed budget in order that every item meets the functional requirements.

Table 4.2. Functional requirements matrix with budgets

	$\mathbf{FR_1}$	$\mathbf{FR_2}$	$\mathbf{FR_3}$	…	$\mathbf{FR_{n-1}}$	$\mathbf{FR_n}$	TOTAL
I_1	b_{11}	b_{12}	b_{13}	…	$b_{1,n-1}$	$b_{1,n}$	b_{I_1}
I_2	b_{21}	b_{22}	b_{23}	…	$b_{2,n-1}$	$b_{2,n}$	b_{I_2}
I_3	b_{31}	b_{32}	b_{33}	…	$b_{3,n-1}$	$b_{3,n}$	b_{I_3}
⋮	⋮	⋮	⋮	… … …	⋮	⋮	⋮
I_{m-1}	$b_{m-1,1}$	$b_{m-1,2}$	$b_{m-1,3}$	…	$b_{m-1,n-1}$	$b_{m-1,n}$	$b_{I_{m-1}}$
I_m	$b_{m,1}$	$b_{m,2}$	$b_{m,3}$	…	$b_{m,n-1}$	$b_{m,n}$	b_{I_m}
TOTAL	b_{FR_1}	b_{FR_2}	b_{FR_3}	…	$b_{FR_{n-1}}$	b_{FR_n}	b_{FR_T}

As can be seen in Table 4.2, all the necessary budgets to meet all functional requirements are identified in the functional requirements matrix. As a result of this first stage, an allocation of resources would have been achieved; a full picture of time and budget needed to guarantee that each and every single one of the items composing the final product meets the functional requirements. This is done regardless the nature (HW, SW…) as a previous step to the final assignment of a value as CI to all the items composing the final product.

4.1.2 Configuration requirements matrix

The second step of the methodology is the calculation of the items-configuration requirements matrix. To obtain this configuration requirements matrix, the configuration requirements have to be identified. Then, the configuration requirements and the 'as designed baseline' will be mapped as it will be described subsequently. Figure 4.11 shows how the configuration requirements affect some of the subsystems/items.

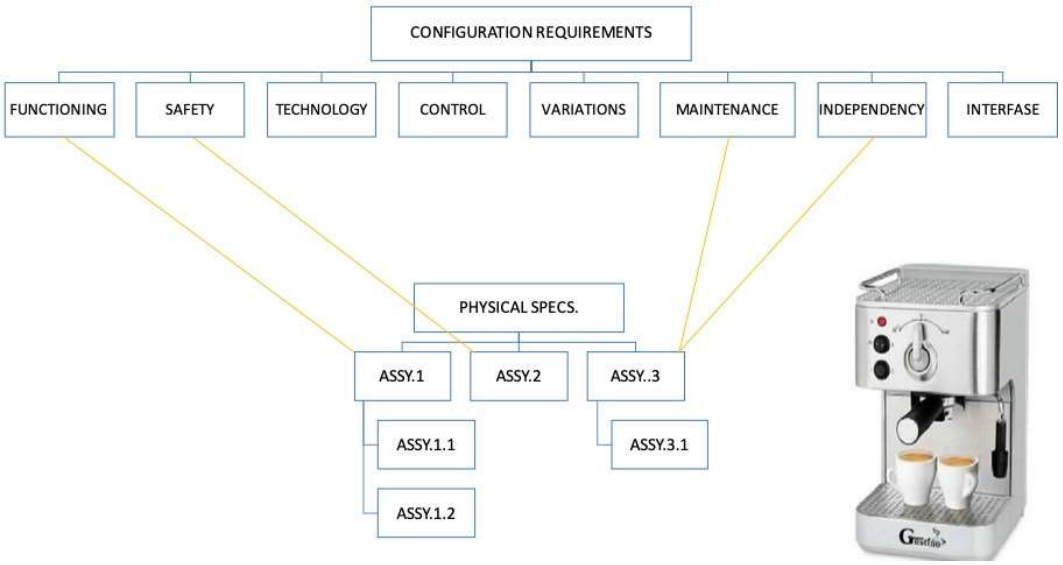


Figure 4.11. Configuration requirements

As can be seen in Figure 4.11, the configuration requirements have to be allocated to the subsystems/items. For instance, the subsystem supplying water is key for the functioning of the coffee maker. Besides, this assembly needs a maintenance since the water pump has to be changed regularly. Another example is the fact the integrity or safety of the user can be affected if the hot water unit assembly does not work properly.

In Systems Engineering the configuration requirements constitute configuration management concerns. These requirements are based on the management needs related to the system to be develop. The aim of these requirements is to identify the criteria that has to be followed in order to define and to control the performance of a system.

Some of the primary configuration requirements are: Independent end use functions; Sub-assembly factors such as the need for separate configuration control or a separate address for the effectivity of changes; Common components to several systems; Interface with other systems, equipment or software Level at which interchangeability must be maintained; Separate delivery or installation requirement x Separate definition of performance.

By means of this method, transparency and order is ensured when allocating resources to the coffee maker during its life cycle. In line with this, the importance and the risk within the project of all the components of the coffee machine are measured and quantified. In addition, the impact of any change, dysfunction or failure in the coffee machine can be unequivocally detected.

On the other hand, traceability between the requirements of the coffee machine and components throughout the life cycle is ensured. Finally, the unequivocal identification of all the components of the coffee machine is ensured according to its importance and risk.

4.1.2.1. Configuration requirements identification

The identification of the configuration requirements is jointly raised between the customer and the manufacturer. This can be done via some standardized questions separated into seven different categories as those shown in Table 4.3. These categories are; Functioning, Safety, Technology, Control, Maintenance, Independency, Interface. The standards of Configuration Management give clues as to which the categories should be, however the final decision is given by the experience in similar projects of the configuration manager and the characteristics of the project.

Table 4.3. Configuration requirements identification

Category	Question
Functioning	Is the item´s functioning critical or high risk?
	Would failure of the item have significnat functioning impact?
	Is the item required for functioning?
Safety	Does the item implement critical capabilities (e.g. security, protection, collision avoidance, human safety, nuclear safety)?
	Would CI designation enhance the required level of control and verification of these capabilities?
Technology	Is the item computer hardware or software?
	Will the item require development of a new design or a significant modification to an existing design?
	Does the item incorporate unproven technologies?
Control	Would CI designation enhance the required level of control and verification of these capabilities?
	Can the item be readily marked to identify it as a separate controlled item?

	Will it be necessary to have an accurate record of the item´s exact configuration and the status of changes to it during its life cycle?
	Is the item at an appropriate level for Government configuration control?
Maintenance	Does it have an independent maintenance?
	Is the item in the maintenance technical manuals?
Independency	Can (or must) the item be independently tested?
	Do all subassemblies of the item have common mission, installation and deployment requirements and common testing?
	Does the item have separate mission, training, test, maintenance and support functions or require separately designated versions for such purposses?
	Is it, or does it have the potential to be designated for separate procurement?
Interface	Does the item interface with a CI controlled by another design activity?
	Does the item have an interface with a CI developed under another contract?
	Have different activities been identified to logistically support various parts of the system?

The list shown in Table 4.3 is not exhaustive and it can be extended an adapted to cover special cases. However, it is a guide for configuration items identification. These are some of the questions presented in the Military Standards (*MIL-HDBK-61 A CONFIGURATION MANAGEMENT GUIDANCE*, n.d.) to identify the configuration requirements. The responsibility of the CM manager is to identify which of those affect to the project, as well as, the creation of new ones depending on the nature of the product and the project. From that moment onwards, all the questions will be ordered and classified as in Table 4.3.

4.1.2.2. Configuration requirements categorization

An example of configuration requirements can be found in Table 4.4. It must be clarified that there could be others since at the end of the day, they are the result of the negotiation between client and final constructor. The categories are those mentioned in Table 4.4.

Table 4.4. Configuration requirements

Category	Configuration requirements
Functioning	Critical to the functioning of the system
	Whose failure might result in primary system capability
	Implement critical capabilities

Safety	Whose failure might result in serious injury or loss of life
	Critical or high risk and failure would have significant safety impact
Technology	Whose technology is complex
	Whose technology is critical to meeting acquisitions, operational or supportability goals in program
Control	Enhance the requierd level of control and verification of critical capabilities
	Requires separate designated variance, versions or nomencaltures
	Are expected to change over time either because of error or changes in requirements
Mintenance	Requires technical manuals
	Planned maintenance system (PMS)
	Maintenance index pages (MIP)
	Maintenance required cards (MRC)
Independency	Can be independently tested
	It has a separate mission, test, maintenance and support functions or requires separately designated versions for such purposes
	Separate definition of performance and test requirements
	Separate delivery or installation
Interface	Interface with CI controlled by another design activity
	Interface with other systems equipment or software

This last table shows the configuration requirements of the project and arises as a result of the questions formulated in the previous table (*MIL-HDBK-61 A CONFIGURATION MANAGEMENT GUIDANCE*, n.d.). By means of this last table the configuration manager will proceed to calculate the degree of involvement of each component of our final product in the fulfilment of the configuration requirements, being 1 total implication in the fulfilment of the requirement, 0 null implication and 0.5 a certain degree of implication.

4.1.2.3. Configuration requirements prioritization

The assigned value to configuration requirements is the measure of the importance of each one to the stakeholders. Hence, as previously mentioned, the high-value configuration requirements need more detailed tracing than low-value configuration requirements, since they represent higher value requirements for the final product. In fact, they represent the management needs of the program and trace information of the latter is more important than trace information of low-value configuration requirements.

Hence, in order to prioritize the configuration requirements, a relative value or factor 0<f<1 has to be allocated to every single configuration requirement. This value will be given to an item according to the degree of compliance of a certain configuration requirement. These values are shown in Table 4.5.

Table 4.5. Configuration requirements prioritization

Value	Configuration requirement priorization
0	The item does not meet configuration requirements
0,5	The tem meets the configuration requirements partially
1	The item fully meets the configuration requirements

This prioritization will help to identify the importance that every item has for the fulfilment of the configuration requirements.

4.1.2.4. Configuration requirements mapping

In order to do the configuration requirements mapping, the configuration requirements need to be identified. Once they are identified, the mapping between the items of the `as designed baseline´ and the `*configuration requirements´* is done according to the features and composition of Table 2.2 and Figure 2.8 respectively. Hence, next step is to assign a degree of compliance of the configuration requirement to every item as

$$f_j = \sum_{i=1}^{i=m} f_{ij}, \qquad \textbf{4.9}$$

where *j=1…p*; *p* is the total number of configuration requirements; f_{ij} represents the level of compliance of j_{th} configuration requirement (CR) of i_{th} item and f_j represents the total level of compliance of the j_{th} configuration requirement of all the items composing the final product. The goal is to prioritize one versus the other items. Finally, the degree of compliance will be obtained with the configuration requirements of each element of the as design baseline

$$f_{I_i} = \sum_{j=1}^{j=p} f_{ij}, \qquad \textbf{4.10}$$

where $i=1...m$ and f_{I_i} represents the total level of compliance of i_{th} item of all the configuration requirements. It means, the total factor f_{CR_T} dedicated to the configuration requirements can be calculated as the sum of all the partial factors that all the items need to meet the configuration requirements f_{I_i}according to

$$\boldsymbol{f_{CR_T} = \sum_{i=1}^{i=m} f_{I_i},} \qquad \textbf{4.11}$$

where f_{CR_T} represents the total factor dedicated to the configuration requirements. Once the levels of compliance of all the items has been calculated, the configuration requirements are mapped with the items of the BoM producing the configuration requirements matrix shown in Table 4.4. Table 4.6 is the configuration requirements matrix showing the degree of compliance of the configuration requirements of every item.

Table 4.6. Configuration requirements matrix

	$\mathbf{CR_1}$	$\mathbf{CR_2}$	$\mathbf{CR_3}$	…	$\mathbf{CR_P}$	$\mathbf{CR_{P-1}}$	**Total**
I_1	f_{11}	f_{12}	f_{13}	…	$f_{1,p-1}$	$f_{1,p}$	f_{I_1}
I_2	f_{21}	f_{22}	f_{23}	…	$f_{2,p-1}$	$f_{2,p}$	f_{I_2}
I_3	f_{31}	f_{32}	f_{33}	…	$f_{3,p-1}$	$f_{3,p}$	f_{I_3}
…	…	…	…	…	…	…	…
I_{m-1}	$f_{m-1,1}$	$f_{m-1,2}$	$f_{m-1,3}$	…	$f_{m-1,p-1}$	$f_{m-1,p}$	$f_{I_{m-1}}$
I_m	$f_{m,1}$	$f_{m,2}$	$f_{m,3}$	…	$f_{m,p-1}$	$f_{m,p}$	f_{I_m}
Total	f_{CR_1}	f_{CR_2}	f_{CR_3}	…	$f_{CR_{p-1}}$	f_{CR_p}	t_{CR_T}

As can be seen in Table 4.6, all the degrees of compliance of the configuration requirements of the items are identified in the configuration requirements matrix. As a result of this second stage, an identification of configuration requirements to each and every one of the items composing the final product is achieved in order to comply with the configuration requirements of the project regardless the nature (HW, SW…). This step is the previous one before the final calculation of the value as CI of all the items of the final product.

4.1.3 Configuration items calculation

As was said in the introduction, the last stage is the calculation of the value as CI of each and every item composing the final product from the final values calculated in Table 4.1, Table 4.2 and Table 4.6. A definition of Configuration Item (CI) is "an item aggregation of hardware or software that satisfies an end function and is designated for a separate configuration management".

As can be observed in the Figure 4.12 Configuration Management shall allocate a unique identifier (consisting of numbers, letters or symbols, or combination thereof), to uniquely identify the CI, and cross-reference to other drawings or documents as appropriate, subsequently recording in the project specific 'Documentation Plan'.

All documentation defined as CIs, and any supporting documentation derived from CIs, shall for release be distinguished by the application of issue numbers. Any deviation away from this standard shall be documented in the associated project CM Plan.

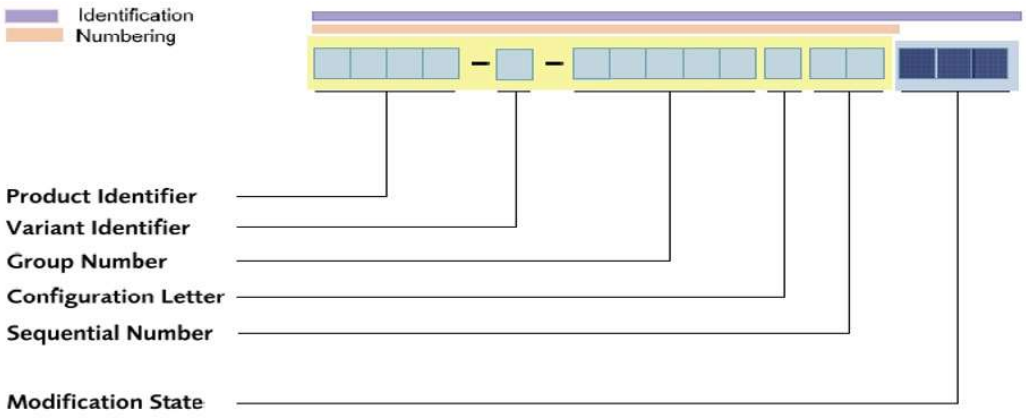


Figure 4.12. Example of P/N identification

As shown in Figure 4.12, P/N whole identification may consist in a numbering plus a modification state. As a part of the identification it might be included a product identifier, a variant identifier, a group number, a configuration letter and a sequential number.

The calculation of CIs is a bottom-up approach; it means that the method will start from the lowest level of the BoM to the subsystem upstream. A subsystem is a group of

interconnected and interactive items that performs an important job or task as a component of a larger system. Hence, the value as CI of one subsystem is the addition of the values of all the items composing the mentioned subsystem.

Therefore, if one item reaches the value of CI, the subsystem in which the item is included will be a CI immediately. Figure 4.13 shows all the items composing a coffee maker susceptible to be CIs.

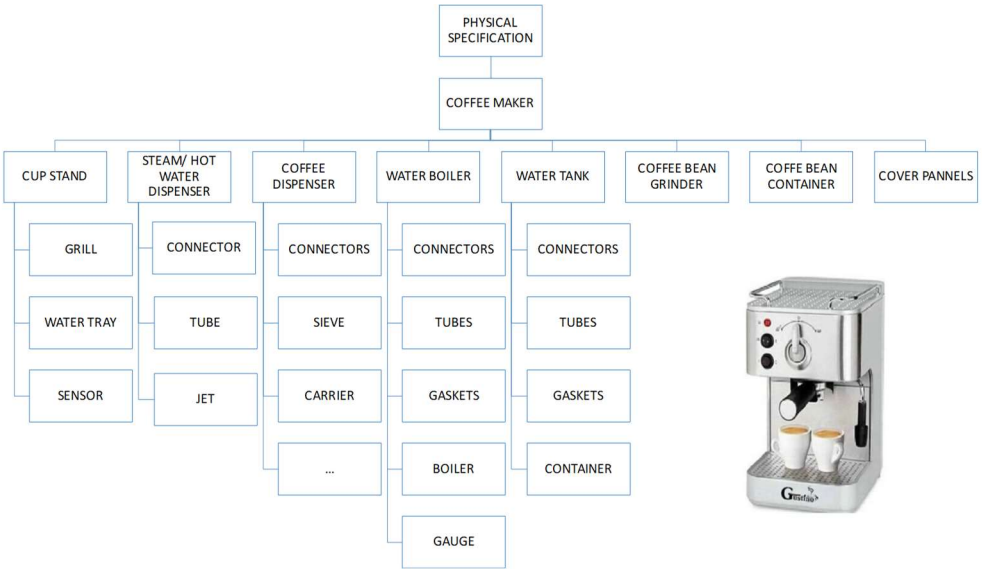


Figure 4.13. Configuration items calculation

As can be seen in Figure 4.13, all the items from the BoM of a coffee maker are likely to be a CI in first instance. However, not all the items have the same importance in terms of CM. Therefore, in the following section it will be depicted a method to distinguish from each other from the CM perspective.

4.1.4 Configuration items categorization

To categorize the CI, the stake-holders agree on the relative importance of each of the factors: time, budget and configuration considered through 3 constant values *x, y* and *z* respectively. From there, the final weight of the time factor for every item will be calculated as

$$\boldsymbol{T_{I_i} = x\frac{t_{I_i}}{t_{FR_T}},} \quad \textbf{4.12}$$

where x is the relative importance of the time in comparison with the budget and the configuration requirements and T_{I_i} is the absolute total value of the budget for i_{th} item. Analogously, it will be done with budget factor as

$$\boldsymbol{B_{I_i} = y\frac{b_{I_i}}{b_{FR_T}},} \quad \textbf{4.13}$$

where y is the relative importance of the budget in comparison with the time and the configuration requirements; B_{I_i} is the absolute total value of the budget for i_{th} item and finally with the configuration factor

$$\boldsymbol{F_{I_i} = z\frac{f_{I_i}}{f_{FR_T}},} \quad \textbf{4.14}$$

where z is the relative importance of the configuration requirements in comparison with the time and the budget and F_{I_i} is the absolute total value of the configuration requirements for i_{th} item. The final value of each element as a CI will be calculated by multiplying the absolute total value of time, budget and configuration requirements obtained in equations (4.12), (4.13) and (4.14) as

$$\boldsymbol{W_{I_i} = T_{I_i} \times B_{I_i} \times F_{I_i},} \quad \textbf{4.15}$$

where W_{I_i} is the total value as CI of i_{th} item.

From this moment onwards, it is possible to select the CIs according to their value in a justified and reasoned way. The procedure is summarized as follows:

- Calculate the time needed by every item to meet the functional requirements.
- Calculate the budget needed by every item to meet the functional requirements.

- Calculate the degree of compliance of the configuration requirements by every item.
- Calculate the absolute value that every item has a CI. It means the weight that all the items have from the Configuration Managenet perspective or in other words how susceptible are to be under the most strict Configuration Mangement control.

4.2 Conclusions

A systematically structured procedure for the practical implementation of Configuration Management has been defined. This was done via the definition of a standardized procedure for identifying, selecting and categorizing the Configuration Items that leads to the practical implementation of Configuration Management. As a result, the proposed methodology guarantees traceability, explanation and reproducibility on Configuration Management implementation as its main advantages.

In order to facilitate understanding, the description of the method has been supported by the simple and didactic case of a coffee maker. Next chapter validates this methodology by applying it into a real project case where its foreseen advantages are proven.

Chapter 5

Validation

This chapter will demonstrate the applicability of the devised methodology, described in Chapter 4, in a real industrial use case exemplary of a highly complex CM process.

This chapter meant to serve as a validation since the chosen system is one of the systems with the greatest added value technologically speaking that there may currently be worldwide. It does indeed encompass elements of various nature (HW, SW, FW…) all of them interconnected with the most modern architectures of communication.

The studies real industrial example is used to validate this method with a real configuration management case. Not only is the subsystem a real case, but the scenario is also real. The BoM of the product is the one under the responsibility of MITSUBISHI which is the leader of a construction consortium for the last generation metro. From this level downwards, it will be the responsibility of the different subcontractors according to Figure 5.1.

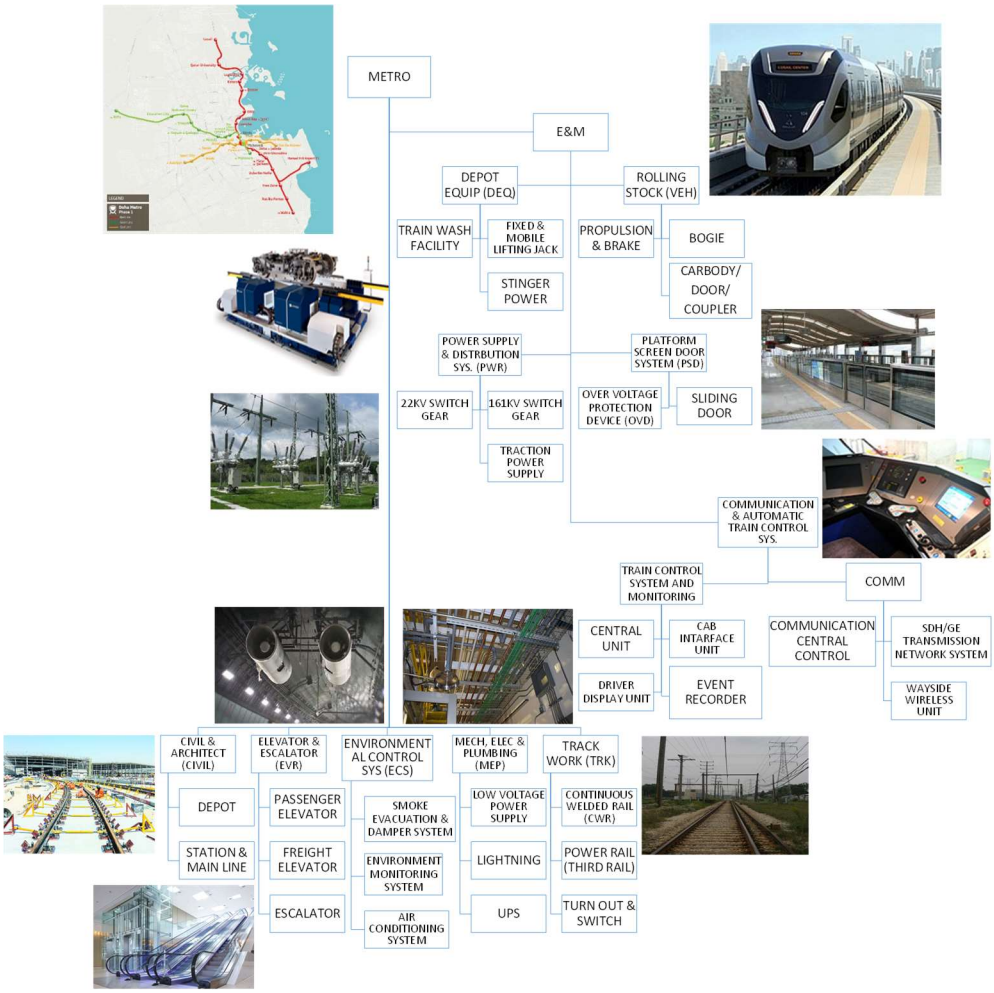


Figure 5.1. Product Breakdown Structure (PBS) of the Doha metro

For our case study, we consider the subsystem Train Control & Management System (TCMS) which is a train-borne distributed control system. It comprises computer devices and software, human-machine interfaces, digital and analogue input/ output (I/O) capability and the data networks to connect all these together in a secure and fault-resistant manner. (Neil, 2014).

Typically, the TCMS provides data communications interfaces to other train-borne systems, and also telecommunications to supporting systems operating remotely on the wayside. TCMS will usually be physically separated from Wi-Fi networks available to passengers for security reasons. TCMS is often referred to as the “brain of the train” due to its central role in coordinating control and monitoring across disparate systems (J. Wang et al., 2014).

5.1 Train Control Monitoring System functional requirements matrix

As explained in the methodology, the 1st stage is the calculation of the functional requirements matrix. Therefore, in order to obtain the requirement matrix of the TCMS, it is necessary to map the functional requirements and the BoM of the TCMS. As shown in the methodology, there are 2 types of functional requirements matrix:

a) one is relative to deadlines
b) relative to budgets

The first one shows all the necessary development times that every component of the final product needs to meet all functional requirements and the second one the money that every component needs to meet the functional requirements.

The functional requirements that have been considered are those that correspond to the main subsystems of the TCMS since below this first level, it would be in the different suppliers where the responsibility to ensure the identification and traceability of its components would fall.

5.1.1 Train Control Monitoring System as required baseline

The functional requirements define all the functions of the TCMS and its mayor components. Table 5.1 shows all the functional requirements that are going to be considered for the TCMS at the level of the metro construction consortium leader

Table 5.1. TMCS Functional requirements

	TMCS Functional requirements
FR_1	Provide Control Device for driving control auxiliary control and fault detection and Ethernet switching device
FR_2	Provide interface conversion device of Ethernet link and RS485 transmission of Digital Input/output interface
FR_3	Provide train equipment status and fault information for the train driver via touch screen display
FR_4	Provide event system to ensure critical data is recorded

As can be seen in Table 5.1, there are 4 functional requirements for the construction consortium leader for the TCMS.

Controllers interface to the real world through electrical I/O and sensors to obtain train status and drive the operation of connected equipment. They also communicate with each other and central servers through on-board networks.

A communication interface conversion device is suitable for electrically connecting a detecting unit and a vehicle electronic control unit (ECU) with different communication interfaces. The communication interface conversion device includes a first circuitry, a second circuitry, and a conversion module.

The Train Equipment Control (TEC) is a powerful user control. Through the touchscreen interface it provides access to essential operational data and detailed maintenance information in a single location. The TEC is designed for easy installation and integration with the train's Ethernet network and on-board subsystems. It operates over a wide range of voltages for direct connection to the vehicle's low-voltage system.

The Event Recorder (ER) collects critical information about train systems, driver activities, and vehicle performance and stores it in a Hardened Memory Module. Recorded data provides vital information following an incident to determine root cause and identify preventive measures. Redundant network connections allow the ER to obtain data reliably from train subsystems over the Ethernet network, reducing the need for discrete wired connections (Bar-Am, 2015).

5.1.2 Train Control Monitoring System `as designed baseline´

In this case the `as designed baseline´ is a snapshot in time that represents the agreed-upon, reviewed, and approved items identification describing functional and physical characteristics, interface requirements, design constraints and verification required committed to the TCMS release. As explained, from the `as designed baseline´ it will be extracted the BoM of the TCMS.

Accordingly, Figure 5.2 shows the BoM of the TCMS.

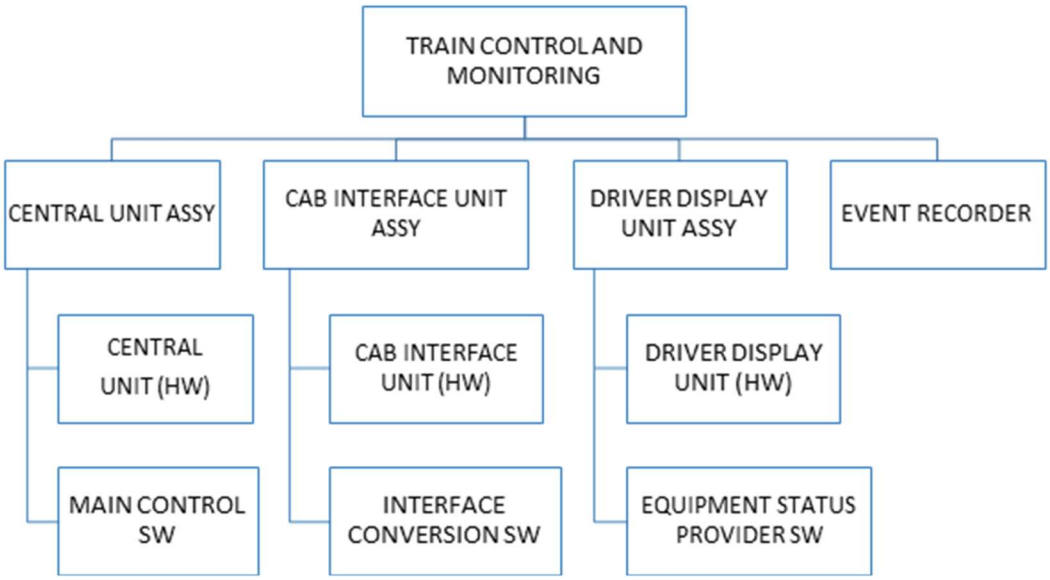


Figure 5.2. PBS of the TMCS

As can be seen in Figure 5.2 the BoM has 4 major subsystems. Table 8 shows all the items in the lowest level of the BoM of the TCMS.

Table 5.2. List of items of the TCMS

Items label	Items description
I_1	Central unit (HW)
I_2	Main control SW
I_3	Cab interface unit (HW)
I_4	Interface conversion SW
I_5	Driver display unit (HW)
I_6	Equipment status SW
I_7	Event recorder

As can be seen in Table 5.2 there are 7 items or subsystems under the responsibility of the construction consortium leader that are susceptible to be considered as CI. The next step is the application of the equation (4.1) to (4.4), in order to create the TCMS functional requirements matrix with time allocation. Table 5.3 shows the TCMS functional requirements matrix with time allocation.

Table 5.3. TCMS functional requirements-time matrix (weeks)

PBS			Functional requirements				Total
Level 1	**Level 2**	**Level 3**	**FR_1**	**FR_2**	**FR_3**	**FR_4**	
TMCS	Central Unit assy	Central Unit (HW)	10				10
TMCS	Central Unit assy	Main control SW	20				20
TMCS	Cab interface Unit assy	Cab interface unit (HW)		10			10
TMCS	Cab interface Unit assy	Interface conversion SW		20			20
TMCS	Driver display unit assy	Driver display unit (HW)			10		10
TMCS	Driver display unit assy	Equipment status SW			20		20
TMCS	Event recorder unit	Event recorder unit				10	10
		Total	30	30	30	10	100

As can be seen in Table 5.3 the functional requirements and the items of the BoM have been mapped and the time has been allocated. According to the results, the project will allocate 10 weeks to the central unit and 20 weeks to the TCMS SW to fulfil the functional requirement of providing control device for driving control, auxiliary control and fault detection and Ethernet switching device.

Besides, 10 weeks will be allocated to the Cab Interface Unit and 20 to the TCMS SW to Provide interface conversion device of Ethernet link and RS485 transmission or Digital Input/output interface. Further, 10 weeks will be allocated to the driver display unit HW and 20 weeks to the TCMS SW to provide train equipment status and fault information for the train driver via touch screen display.

Finally, 10 weeks to the event recorder to provide event recorder system to ensure critical data is recorded critical or high risk and failure would have significant financial impact. Therefore, the TCMS SW will take twice the time the HW components of the TCMS. In order to create the TCMS functional requirements matrix with budget, the equation (4.5) to (4.8) are applied as per Table 5.4.

Table 5.4. Train Control and Monitoring functional requirements matrix ($10,000)

PBS			Functional requirements				Total
Level 1	**Level 2**	**Level 3**	**FR_1**	**FR_2**	**FR_3**	**FR_4**	
TMCS	Central Unit assy	Central Unit (HW)	5				5
TMCS	Central Unit assy	Main control SW	25				25
TMCS	Cab interface Unit assy	Cab interface unit (HW)		5			5
TMCS	Cab interface Unit assy	Interface conversion SW		25			25
TMCS	Driver display unit assy	Driver display unit (HW)			5		5
TMCS	Driver display unit assy	Equipment status SW			25		25
TMCS	Event recorder unit	Event recorder unit				10	10
		Total	30	30	30	10	100

According to the results, the project will allocate $50,000 to the central unit and $250,000 to the TCMS SW to fulfil the functional requirement of providing a control device for driving control, auxiliary control and fault detection and an ethernet switching device. Besides, $50,000 will be allocated to the Cab Interface Unit and $250,000 to the TCMS SW to Provide interface conversion device of Ethernet link and RS485 transmission or Digital Input/output interface.

Further, $50,000 will be allocated to the driver display unit HW and $250,000 to the TCMS SW to provide train equipment status and fault information for the train driver via touch screen display. Finally, $100,000 to the event recorder to provide an event recorder system to ensure critical data is recorded critical or high risk and failure would have significant financial impact.

Therefore, the TCMS SW components will request more than 4 times the money of the HW components of the TCMS. The hierarchical physical and functional product break-down structures of the final TCMS with the relationships of its components are completely defined through the BoM and the different functional requirements matrixes, where the resources time and money are distributed unequivocally among all the components of the different subsystems.

The method facilitates management of information since the traceability between the functional requirements of the TCMS and components throughout the life cycle is ensured by means of the functional requirement matrixes

5.2 Train Control Monitoring System Configuration requirements matrix

As explained in the methodology the 2nd stage is the calculation of the functional requirements matrix. Therefore, in order to obtain the configuration requirements matrix of the TCMS the configuration requirements have to be identified. All the degrees of compliance of the configuration requirements of the items are identified in the TCMS configuration requirements matrix.

Therefore, the degree of compliance of the following components will be calculated:

a) Central Unit (HW)
b) TCMS Software
c) Cab Interface Unit (HW)
d) TCMS Software
e) Driver Display Unit (HW)
f) TCMS Software
g) Event Recorder

For such estimation the degree of compliance with the configuration requirements of the following elements will be calculated using the aforementioned prioritization scale: 0 does not comply, 0.5 partially complies and 1 absolutely fulfils the considered requirement.

5.2.1 Train Control Monitoring System Configuration requirements

The configuration requirements define configuration management concerns. These requirements are based on the management needs of the program to develop the TCMS.

These requirements are the criteria that has to be followed in order to define and control the performance of a TCMS and all of its components namely:

a) Central Unit (HW)
b) TCMS Software
c) Cab Interface Unit (HW)
d) TCMS Software
e) Driver Display Unit (HW)
f) TCMS Software
g) Event Recorder

In order to identify these requirements, the configuration management the rail standards CENELEC and the nature of the rail industry in Qatar are taken into account. As a result, the configuration requirements are categorized according to the following aspects functioning, safety, control, technology, control, variance, maintenance, independency and interface.

According to Figure 5.3 for the particular case of safety, the questions are:

If the component implements critical capabilities like for instance:

- security protection.
- collision avoidance.
- human safety.
- nuclear safety.

…

Would CI designation enhance the required level of control and verification of these capabilities?

As the answer to this question is affirmative, we will transform this question into two statements that will be mapped with all the components of the TMCS:

- Whose failure might result in serious injury or loss of life
- Critical or high risk and failure would have significant safety impact

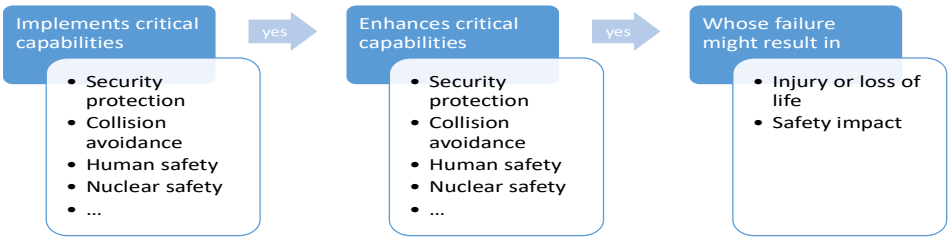


Figure 5.3. Configuration requirements agreement process

As depicted in Figure 5.3 at the end of the negotiations between the stake-holders, these were configuration requirements that were agreed upon for the TCMS. The list of configuration requirements that the CIs have to meet is shown in Table 5.5. Such table shows the way that the configuration requirements are going to be cetegorized for the TMCS.

Table 5.5. Configuration requirements for the TCMS

Configuration Requirements	
Functioning	Critical to the functioning of the system
	Whose failure might result in primary system capability
	Implement critical capabilities
Safety	Whose failure might result in serious injury or loss of life
	Critical or high risk and failure would have significant financial impact
Technology	Whose technology is complex
	Whose technology is critical to meeting acquisition, operational or supportability goals of the program
Control	Enhance the required level of control and verification of critical capabilities
Variance	Requires separate designated variance versions or nomencaltures
	Are expected to change over time either gbecause of errors or changes in requirements
Maintenance	Requires test equipment (e.g. General Purpose, Special Purpose or Built-in)
	Requires technical manuals
	Maintenance Requirement cards (MRC)
Interdependency	Can be independently tested
	Independent end of use functions
	It has a separate mission, training, test maintenace and support functions or requires separately designated versions fior such purpose
	Separate definition of performance and test requirements
Interface	Interface with a CI controlled by another design activity
	Interface with other systems, equipment or software

5.2.2 Train Control Monitoring System Configuration requirements mapping

In this step the TCMS configuration requirements matrix will be created through the calculation of the degree of compliance of the configuration requirements by every item described in Table 5.5, equation (4.9) to (4.11) and Table 4.6. As output, the configurations requirements matrix shown in Table 5.6.

Table 5.6. TCMS configuration requirements matrix

Configuration Requirements		I_1	I_2	I_3	I_4	I_5	I_6	I_7	Total
Functioning	Critical to the functioning of the system	0.5	1	0.5	1	0.5	1	0.5	5
	Whose failure might result in primary system capability	0.5	1	0.5	1	0.5	1	0.5	5
	Implement critical capabilities	0.5	1	0.5	1	0.5	1	0.5	5
Safety	Whose failure might result in serious injury or loss of life	0.5	1	0.5	1	0.5	1	0.5	5
	Critical or high risk and failure would have significant financial impact	0.5	1	0.5	1	0.5	1	0.5	5
Technology	Whose technology is complex	0.5	1	0.5	1	0.5	1	0.5	5
	Whose technology is critical to meeting acquisition, operational or supportability goals of the program	0.5	1	0.5	1	0.5	1	0.5	5
Control	Enhance the required level of control and verification of critical capabilities	0.5	1	0.5	1	0.5	1	0.5	5
Variance	Requires separate designated variance versions or nomencaltures	0.5	1	0.5	1	0.5	1	0.5	5
	Are expected to change over time either gbecause of errors or changes in requirements	0.5	1	0.5	1	0.5	1	0.5	5
Maintenance	Requires test equipment (e.g. General Purpose, Special Purpose or Built-in)	0.5	1	0.5	1	0.5	1	0.5	5
	Requires technical manuals	0.5	1	0.5	1	0.5	1	0.5	5
	Maintenance Requirement cards (MRC)	0.5	1	0.5	1	0.5	1	0.5	5
Interdependency	Can be independently tested	0.5	1	0.5	1	0.5	1	0.5	5
	Independent end of use functions	0.5	1	0.5	1	0.5	1	0.5	5
	It has a separate mission, training, test maintenace and support functions or requires separately designated versions fior such purpose	0.5	1	0.5	1	0.5	1	0.5	5
	Separate definition of performance and test requirements	0.5	1	0.5	1	0.5	1	0.5	5
Interface	Interface with a CI controlled by another design activity	0.5	1	0.5	1	0.5	1	0.5	5
	Interface with other systems, equipment or software	0.5	1	0.5	1	0.5	1	0.5	5
	Total	10	20	10	20	10	20	10	100

As can be seen in Table 5.6 all the configuration requirements have been mapped with the items of the TCMS and prioritized.

Therefore, the method identifies the attributes (performance, safety, technology, control, variance, maintenance, and independency) of the TCMS and all of its components: through the configuration requirements matrix.

5.3 Train Control Monitoring System Configuration items calculation

As explained, the last stage is the calculation of the value as CI of each and every item composing the TCMS from the final factors calculated in the previous points:

1. Calculate the time needed by every item to meet the functional requirements equation (4.1) to (4.4) and Table 4.1. As output, the functional requirements matrix considering time for the TCMS shown in Table 5.3.

2. Calculate the budget needed by every item to meet the functional requirements equation (4.5) to (4.8) and Table 4.2. As output, the functional requirements matrix considering budget for the TCMS shown in Table 5.4.

3. Calculate the degree of compliance of the configuration requirements by every item described in Table 4.5, equation (4.9) to (4.11) and Table 4.6. As output, the configurations requirements matrix shown in Table 5.6.

4. Calculate the absolute value that every item has a CI according to the following point.

5.4 Train Control Monitoring System Configuration items categorization

In this particular case, the importance of the time is twice the importance of the budget. On the other hand, all the configuration factors have the same importance as the budget.

The reason for all this is that for the final customer, Qatar Rail, the delivery times are more important than the budget. In fact, the goal of this rich country is to have at least the red line (which goes from the airport to the centre of the capital) of the Doha metro available for the Qatar 2022 FIFA World Cup.

It means that $x = 2y = 2z$.

Next step is the application of equation (4.12) to (4.15) in order to calculate the value as CI of every single item of the TCMS.

From Table 5.3 and equation (4.12) the absolute total values of the budget for all the components of the TCMS are obtained as per Table 5.7.

Table 5.7. Weights according to the time factor

I	t	T
$\mathbf{I_1}$	10	$20.\mathbf{10^{-2}}$
$\mathbf{I_2}$	20	$40.\mathbf{10^{-2}}$
$\mathbf{I_3}$	10	$20.\mathbf{10^{-2}}$
$\mathbf{I_4}$	20	$40.\ \mathbf{10^{-2}}$
$\mathbf{I_5}$	10	$20.\ \mathbf{10^{-2}}$
$\mathbf{I_6}$	20	$40.\ \mathbf{10^{-2}}$
$\mathbf{I_7}$	10	$20.\ \mathbf{10^{-2}}$

From Table 5.4 and equation (4.13) the absolute total value of the budget for all the components of the TCMS are obtained as per Table 5.8.

Table 5.8. Weights according to the configuration requirements factor

I	f	F
$\mathbf{I_1}$	10	$10.\ \mathbf{10^{-2}}$
$\mathbf{I_2}$	20	$20.\ \mathbf{10^{-2}}$
$\mathbf{I_3}$	10	$10.\ \mathbf{10^{-2}}$
$\mathbf{I_4}$	20	$20.\ \mathbf{10^{-2}}$
$\mathbf{I_5}$	10	$10.\ \mathbf{10^{-2}}$
$\mathbf{I_6}$	20	$20.\ \mathbf{10^{-2}}$
$\mathbf{I_7}$	10	$10.\ \mathbf{10^{-2}}$

After that, equation (4.15) will be applied in order to obtain the final values of every component of the TCMS as a CI.

Table 5.9. Final value as CI of the items of the TCMS.

I	ITEMS DESCRIPTION	T	B	F	W	CI
$\mathbf{I_1}$	CENTRAL UNIT (HW)	20. $\mathbf{10^{-2}}$	5. $\mathbf{10^{-2}}$	10. $\mathbf{10^{-2}}$	10. $\mathbf{10^{-4}}$	No
$\mathbf{I_2}$	TCMS SOFTWARE	40. $\mathbf{10^{-2}}$	25. $\mathbf{10^{-2}}$	20. $\mathbf{10^{-2}}$	200. $\mathbf{10^{-4}}$	Yes
$\mathbf{I_3}$	CAB INTERFACE UNIT (HW)	20. $\mathbf{10^{-2}}$	5. $\mathbf{10^{-2}}$	10. $\mathbf{10^{-2}}$	10. $\mathbf{10^{-4}}$	No
$\mathbf{I_4}$	TCMS SOFTWARE	40. $\mathbf{10^{-2}}$	25. $\mathbf{10^{-2}}$	20. $\mathbf{10^{-2}}$	200. $\mathbf{10^{-4}}$	Yes
$\mathbf{I_5}$	DRIVER DISPLAY UNIT (HW)	20. $\mathbf{10^{-2}}$	5. $\mathbf{10^{-2}}$	10. $\mathbf{10^{-2}}$	10. $\mathbf{10^{-4}}$	No
$\mathbf{I_6}$	TCMS SOFTWARE	40. $\mathbf{10^{-2}}$	25. $\mathbf{10^{-2}}$	20. $\mathbf{10^{-2}}$	200. $\mathbf{10^{-4}}$	Yes
$\mathbf{I_7}$	EVENT RECORDER	20. $\mathbf{10^{-2}}$	10. $\mathbf{10^{-2}}$	10. $\mathbf{10^{-2}}$	20. $\mathbf{10^{-4}}$	Yes

5.5 Conclusion

The more components that are identified as CI the more budget and time is needed. Because of that, it is necessary to keep a balance between control and resources dedicated. Due to the impact in time that it would have an absolute control of all the pieces of the TCMS subsystem, it was decided to consider configuration items only those whose value as an element of configuration W is above $10.\ 10^{-4}$.

As expected, all the SW components of the TCMS have a high value as CI because by definition SW components are CI since every SW meets an end use function. The other way around, HW components containing SW have low value since they do not fulfil any end of use, their objective is to contain the SW and because of that they use to be COTS (commercial off the shelf).

Table 5.9 shows the value as CI of all the items of the TCMS at the main contractor level W. As can be seen in Table 5.9 only items with value higher than $10.\ 10^{-4}$ are considered CIs.

Through the proposed method in chapter 4 the importance and the risk within the project of all the components of the TCMS have been measured and quantified.

By means of the method the impact of any change, dysfunction or failure in the final product can be unequivocally detected through the different matrixes.

Besides, through this method, the appropriate level of identification marking of all the components of the product depending on the value W will be reached. This is so, since all the items identified as CIs in the project will have a unique identity so they will have to be serialized and the rest of items will be identified with lot numbers.

Through this method, the client is involved in the project from the beginning, being an active part of the selection of the CIs. In this way, possible dissatisfaction on the part of the client due to uncertainty at the time of the delivery of the project is avoided, as demonstrated by the greater number of complaints raised by the client of the Dubai metro.

In addition to the above, the development time of the product decreases since the method helps to adapt the physical requirements to the elements to be developed, as demonstrated by the lower number of changes requested by production in this metro compared to the construction of the Dubai metro.

On the other hand, as a consequence of a selection of configuration elements more adjusted to the needs of the project, functional audits are more successful, as demonstrated by the lower number of non-conformities during functional audits compared to the construction project of the subway. Dubai

Chapter 6

Conclusion and future work

This dissertation has contributed to create a standardized procedure for identifying, selecting and categorizing the Configuration Items, which is the first step in any practical implementation of Configuration Management.

The aim is to foster a wider transition of Configuration Management from its current chaotic and lawless practice to a systematic and structured labour in the regular basis.

Thus, the developed final product could have guaranteed traceability of all its components throughout its life cycle, justification in the project for decision-making and reproducibility after delivery as the main advantages.

The aforementioned feasibility and viability has been validated by using it to undertake Configuration Management in the TCMS which is one of the main subsystems of the Doha last generation metro construction project.

Unfortunately, the methodology has not been applied in other sectors or type of products given the lack of sensitive data and information that it would be necessary for an adjusted application in other projects.

The contributions put forward by this dissertation address these limitations and are deemed to make possible a more widespread adoption of the methodology in more type of projects. This chapter summarizes these contributions and identifies areas that deserve further investigation.

6.1 Summary of Contributions

This dissertation has made several important contributions in project management and systems engineering by bringing order and procedure to configuration management through the implementation of this method:

1. **Appropriate CI interfaces to be defined, and documented**. If they are all baselined by the final customer (usually governments) early in the Engineering & Manufacturing Development (EMD) Phase, this will ensure the right level of the contractor's freedom to evolve his design solution, solve problems expeditiously, and implement advantageous changes without contractual consequences.

2. **CI functionality defined at the right level**. It includes appropriate design constraints requiring formal test, and technical reviews, assuring what is required for the final customer to achieve reasonable assurance of system performance. Correlates individual product units to warranties and service life obligations through the configuration requirements matrix. This is even more important if performance specifications for the lower-level CIs are baselined too early in the EMD phase.

3. **Convenient number of configuration requirements**. This provides a balance between technical requirements and control.

4. **Proper fragmentation**. It will actually offer final customer visibility and understanding of system performance since correct fragmented description of functionality decreases the overall volume of requirements. Hence it is more easy to understand, and facilitates the document review, approval, and control process. Opportune complexity of each CI. This results in useful management insight and ability to assess progress leading to decreased complexity in managing and accounting for common assemblies and components. Through pertinent level designated CI (implementing related functions, containing only hardware or software components, etc.) the formal testing of critical capabilities will be well-timed and easier.

5. **The potential for reuse of the CI is increased**. Besides, re-procurement of the CI and components is easy and the potential re-procurement sources are numerous. Apart from

that, the ability to account for the deployment of a CI, whose component parts are disbursed to different locations

6. **Ease in addressing the effectivity of changes and retrofit actions**. Particularly when there are different quantities or separately deliverable components.

7. **Convenient cost**. All the CI has to be serialized and an excessive number of serialized items will affect the budget of the project

The application of the method described in Chapter 5 has served to verify the benefits mentioned above:

- Thanks to the application of the methodology to the TCMS all the changes affecting the HW part of the system went through the quick change flowchart management process. Therefore, it was avoided to spend superfluous time controlling and documenting the changes of these items.

- As a consequence of the application of the methodology, the SW elements of the TMCS led the development of the system, better targeting resources to achieve 100% functional compliance with the final system.

- By means of the method it was easier for the client to understand the functional architecture of the system since a tighter segmentation was achieved

- On the other hand, only the elements identified as configuration elements were assigned a serial number with the consequent savings in resources that this implies since those numbers must be kept in databases and, further than that, the labels must be created so that each configuration item is identified by these numbers.

Unfortunately, project information is not available without the application of the aforementioned methodology. Therefore, it is very difficult to make a comparison of scenarios with and without the method.

It should be noted that the scenario of identifying the configuration items without the method would depend entirely on the personal knowledge of the configuration manager. In other words, a different configuration manager could identify other different configuration items so the scenario would be different as well.

6.2 Future work

Although the method has been applied to diverse nature systems and with a very high technological level, firstly it would be necessary to apply it to systems belonging to other sectors that are also cutting-edge technologically such as aerospace, aeronautics

The objective of this proposal for the future would be to see the response capacity of the method in other sectors and thus in this way approach a complete validation of the method.

A possible avenue of work would be to use "blockchain" (Meng et al., 2018) blockchain to record in an immutable way all the decisions made in the configuration process, so it could not be the case that a client blames that details have been changed in the configuration management, avoiding fines, penalties and so forth.

Secondly, it would be interesting the application of the methodology autonomously and independently by different configuration managers with different training and professional career. In this way, the results could be compared and the level of repeatability or normalization provided by the method could be measured in a more practical way.

Thirdly, and taking advantage of the involvement of configuration managers with experience in the experiment, it would be advisable that the selection of the mentioned configuration elements were made with own or personal models and then compared the results with those produced by the method for the same project.

Later, in-depth study could be made in order to identify which model is more enriching according to the new trends in project management.

Finally, it would be of great help for project management if this method was integrated within the configuration and general project management tools and thus automatically receive all the information that would otherwise have to be downloaded manually.

More specifically, the method should be embedded within the configuration management database as well as configuration management tools such as Windchill (Matta et al., 2011), *SAP* (Bancroft & Cunningham, 1996) and requirements management tools and project management. In this way, an automatic supply of information would be achieved, such as functional requirements, cost of development and development lead-time of all the elements involved, general requirements of the project and the client and so on.

If the integration was complete, a common repository would be available accessible to all the stake-holders of the project.

Bibliography

Abran, A., Moore, J. W., Bourque, P., Dupuis, R., & Tripp, L. (2004). Software engineering body of knowledge. *IEEE Computer Society, Angela Burgess*.

Ackoff, R. L. (n.d.). *Scientific method: Optimizing applied research decisions.*

Adela, H. A. (2019). The Role of the Roads and Transport Authority (RTA) in the Transformation of Dubai into a Smart City. In *Smart Technologies and Innovation for a Sustainable Future* (pp. 403–412). Springer.

Ali, U., & Kidd, C. (2013). Critical success factors for configuration management implementation. *Industrial Management & Data Systems*.

Ali, U., & Kidd, C. (2014). Barriers to effective configuration management application in a project context: An empirical investigation. *International Journal of Project Management*, *32*(3), 508–518.

Ali, U., & Kidd, C. (2015). Configuration Management maturation: An empirical investigation. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, *229*(2), 321–327.

Allan, G. (1997). Configuration management and its impact on businesses that use computer platforms. *International Journal of Project Management*, *15*(5), 321–330.

Almujibah, H., & Preston, J. (2019). The Total Social Costs of Constructing and Operating an High-Speed Rail Line Using a Case Study of the Riyadh-Dammam Corridor, Saudi Arabia. *Frontiers in Built Environment*, *5*, 79.

Alvarez-Rodríguez, J. M., Llorens1, J., Alejandres1, M., & Fuentes2, J. M. (2015). Semantic-based representation, enrichment and computation of product breakdown structures. *INCOSE International Symposium*, *25*(1), 35–53.

Ambriola, V., Bendix, L., & Ciancarini, P. (1990). The evolution of configuration management and version control. *Software Engineering Journal*, *5*(6), 303–310.

Anchuri, P., Zaki, M. J., Barkol, O., Bergman, R., Felder, Y., Golan, S., & Sityon, A. (2012). Graph mining for discovering infrastructure patterns in configuration management databases. *Knowledge and Information Systems*, *33*(3), 491–522.

Army, U. S. (2003). *MIL-STD-973 Configuration Management*.

Bancroft, N. H., & Cunningham, F. (1996). *Implementing Sap R/3*. Prentice Hall PTR.

Bar-Am, M. (2015). *On-train rail track monitoring system*. Google Patents.

Bell, P. M. (1980). Flare build-up study provides new data on solar flares. *Eos, Transactions American Geophysical Union*, *61*(34), 585–585.

Benson, J. O., Prevost, J. J., & Rad, P. (2016). Survey of automated software deployment for computational and engineering research. *2016 Annual IEEE Systems Conference (SysCon)*, 1–6.

Bilge, L., Han, Y., & Dell'Amico, M. (2017). RiskTeller: Predicting the Risk of Cyber Incidents. *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 1299–1311. https://doi.org/10.1145/3133956.3134022

Boeing: The Boeing Company. (n.d.). Retrieved 14 July 2020, from https://www.boeing.com/

Bowman, E. H. (1963). Consistency and optimality in managerial decision making. *Management Science*, *9*(2), 310–321.

Brady, T., Davies, A., Gann, D., & Rush, H. (2007). Learning to manage mega projects: The case of BAA and Heathrow Terminal 5. *Project Perspectives*, *29*, 32–39.

Brenner, M., Garschhammer, M., Sailer, M., & Schaaf, T. (2006). CMDB–yet another MIB? On reusing management model concepts in ITIL configuration management. *International Workshop on Distributed Systems: Operations and Management*, 269–280.

Bronk, C. (2020). Operation of Transport and Logistics in a Time of (Cyber) Insecurity. In *Computation and Big Data for Transport* (pp. 33–49). Springer.

Brouse, P. S. (2009). Configuration management. *Systems Engineering and Management for Sustainable Development-Volume I*, 214.

Burgess, M., & Couch, A. L. (2006). Modeling Next Generation Configuration Management Tools. *LISA*, 131–147.

Burgess, T. F., McKee, D., & Kidd, C. (2005). Configuration management in the aerospace industry: A review of industry practice. *International Journal of Operations & Production Management*.

Cantamessa, M., Montagna, F., & Neirotti, P. (2012). Understanding the organizational impact of PLM systems: Evidence from an aerospace company. *International Journal of Operations & Production Management*.

Carnduff, T. W., & Goonetillake, J. S. (2004). Configuration management in evolutionary engineering design using versioning and integrity constraints. *Advances in Engineering Software*, *35*(3–4), 161–177.

Chaikin, A. (1995). *Lunar Odyssey*. JSTOR.

Chen, X., Mao, Y., Mao, Z. M., & Van der Merwe, J. (2010). Declarative configuration management for complex and dynamic networks. *Proceedings of the 6th International COnference*, 1–12.

Choi, M.-J., Choi, H.-M., Hong, J. W., & Ju, H.-T. (2004). XML-based configuration management for IP network devices. *IEEE Communications Magazine*, *42*(7), 84–91.

Dart, S. (1990). *Spectrum of functionality in configuration management systems*.

Dart, S. (1991). Concepts in configuration management systems. *Proceedings of the 3rd International Workshop on Software Configuration Management*, 1–18.

Dávid, I., Denil, J., Gadeyne, K., & Vangheluwe, H. (2016). Engineering Process Transformation to Manage (In) consistency. *COMMitMDE@ MoDELS*, 7–16.

Davies, A., & Hobday, M. (2005). *The business of projects: Managing innovation in complex products and systems*. Cambridge University Press.

Davies, A., & Mackenzie, I. (2014). Project complexity and systems integration: Constructing the London 2012 Olympics and Paralympics Games. *International Journal of Project Management*, *32*(5), 773–790.

DeVol, R. C. (1999). America's High-Tech Economy. *Santa Monica, CA: Milken Institute*.

Doha Metro, Qatar. (n.d.). Railway Technology. Retrieved 12 July 2020, from https://www.railway-technology.com/projects/doha-metro/

Doreswamy, R. (2013). *Configuration Management at NASA*.

Drouin, N., Besner, C., Piperca, S., & Floricel, S. (2012). A typology of unexpected events in complex projects. *International Journal of Managing Projects in Business*.

Dubai Aviation Engineering Projects (DAEP). (n.d.). Dubai Aviation Engineering Projects. Retrieved 10 July 2020, from http://daep.gov.ae/

Durey, D. A., Murphy, D. J., & Rueda, B. E. (2019). Standardization: Successful Implementation of Driving Consistency in Deepwater Operations. *SPE Annual Technical Conference and Exhibition*.

Earl, M. J. (1992). Putting IT in its place: A polemic for the nineties. *Journal of Information Technology*, *7*(2), 100–108.

Economics, U. S., Administration, S., Telecommunications, U. S. N., & Administration, I. (2000). *Falling Through the Net: Toward Digital Inclusion: a Report on*

Americans' Access to Technology Tools. US Department of Commerce, Economic and Statistics Administration.

Egyed, A., Zeman, K., Hehenberger, P., & Demuth, A. (2018). Maintaining consistency across engineering artifacts. *Computer*, *51*(2), 28–35.

EIA, E. (1998). 649-National Consensus Standard for Configuration Management. *Electronic Industries Alliance*.

Enck, W., Moyer, T., McDaniel, P., Sen, S., Sebos, P., Spoerel, S., Greenberg, A., Sung, Y.-W. E., Rao, S., & Aiello, W. (2009). Configuration management at massive scale: System design and experience. *IEEE Journal on Selected Areas in Communications*, *27*(3), 323–335.

Erne, R., & Knippenberg, C. (2008). Der Erfolg von Geschäftsprozessen in der technischen Systementwicklung. *Zeitschrift Führung Und Organisation*, *77*, 249–258.

Feiler, P., & Downey, G. (1990). *Transaction-Oriented Configuration Management: A Case Study.* CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST.

Felfernig, A., Gruber, T., & Stettinger, M. (2018). Consistency Management Techniques for Variability Modelling. *Enterprise Modelling and Information Systems Architectures (EMISAJ)*, *13*, 207–215.

Feng, T., Dan, B., Lan, L. C., & Zhang, X. M. (2003). Product family structure and configuration management for mass customization. *Jisuanji Jicheng Zhizao Xitong/Computer Integrated Manufacturing System(China)*, *9*(3), 210–213.

Finch, J. (2009). Toyota sudden acceleration: A case study of the national highway traffic safety administration-recalls for change. *Loy. Consumer L. Rev.*, *22*, 472.

Flyvbjerg, B. (2014). What you should know about megaprojects and why: An overview. *Project Management Journal*, *45*(2), 6–19.

Fossa, H., & Sloman, M. (1996). Implementing interactive configuration management for distributed systems. *Proceedings of International Conference on Configurable Distributed Systems*, 44–51.

Gehman, H. W. (2003). *Columbia accident investigation board report* (Vol. 6). Columbia Accident Investigation Board.

Geraldi, J. G., & Adlbrecht, G. (2007). On faith, fact, and interaction in projects. *Project Management Journal*, *38*(1), 32–43.

Goldsack, P., Guijarro, J., Loughran, S., Coles, A., Farrell, A., Lain, A., Murray, P., & Toft, P. (2009). The SmartFrog configuration management framework. *ACM SIGOPS Operating Systems Review*, *43*(1), 16–25.

Graves, T. L., & Mockus, A. (1998). Inferring change effort from configuration management databases. *Proceedings Fifth International Software Metrics Symposium. Metrics (Cat. No. 98TB100262)*, 267–273.

Green, K. C., Armstrong, J. S., & Graefe, A. (2015). Golden rule of forecasting rearticulated: Forecast unto others as you would have them forecast unto you. *Journal of Business Research*, *68*(8), 1768–1771.

Handbook, M. (2001). *Configuration management guidance*. MIL-HDBK-61A (SE) Department of Defense–United States of America.

Hass, A. M. J. (2003). *Configuration management principles and practice*. Addison-Wesley Professional.

Hatzichronoglou, T. (1997). *Revision of the high-technology sector and product classification*.

Hobday, M. (1998). Product complexity, innovation and industrial organisation. *Research Policy*, *26*(6), 689–710.

Iacovou, C. L., Thompson, R. L., & Smith, H. J. (2009). Selective status reporting in information systems projects: A dyadic-level investigation. *MIS Quarterly*, 785–810.

ISO, U. (n.d.). 10007: 2003 Sistemas de gestión de la calidad. *Directrices Para La Gestión de La Configuración*.

Ivory, D. (2014). Federal safety agency ends its investigation of Tesla fires. *The New York Times, Mar*, *28*.

Jarratt, T. A. W., Eckert, C. M., Caldwell, N. H., & Clarkson, P. J. (2011). Engineering change: An overview and perspective on the literature. *Research in Engineering Design*, *22*(2), 103–124.

Joeris, G. (1997). Change management needs integrated process and configuration management. In *Software Engineering—ESEC/FSE'97* (pp. 125–141). Springer.

Kaliappan, V., & Ali, N. M. (2018). Improving Consistency of UML Diagrams and Its Implementation Using Reverse Engineering Approach. *Bulletin of Electrical Engineering and Informatics*, *7*(4), 665–672.

Kang, M.-Y., Jeong, Y., & Jung, Y. (2019). Assessment Methodology of Practical Configuration Management (CM) for Sustainable Nuclear Power Plants (NPPs). *Sustainability*, *11*(8), 2391.

Kang, M.-Y., & Jung, Y. (2015). Framework & Functions of Configuration Management (CM) in Nuclear Power Plants (NPP). *Korean Journal of Construction Engineering and Management*, *16*(3), 101–112.

Kannan, H., Mesmer, B. L., & Bloebaum, C. L. (2017). Increased system consistency through incorporation of coupling in value-based systems engineering. *Systems Engineering*, *20*(1), 21–44.

Kardes, I., Ozturk, A., Cavusgil, S. T., & Cavusgil, E. (2013). Managing global megaprojects: Complexity and risk management. *International Business Review*, *22*(6), 905–917.

Kauffman, J. (1999). Adding fuel to the fire: NASA's crisis communications regarding Apollo 1. *Public Relations Review*, *25*(4), 421–432.

Kerr, C. I. V., Phaal, R., & Probert, D. R. (2008). Technology insertion in the defence industry: A primer. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, *222*(8), 1009–1023.

Lapouchnian, A., Yu, Y., & Mylopoulos, J. (2007). Requirements-driven design and configuration management of business processes. *International Conference on Business Process Management*, 246–261.

Larsson, M. (2000). *Applying configuration management techniques to component-based systems* [PhD Thesis]. Uppsala University.

Larsson, M., & Crnkovic, I. (2001). Configuration management for component-based systems. *Proceedings of International Conference on Software Engineering (ICSE)*.

Larsson, M., & Crnkovic, I. (1999). New challenges for configuration management. *International Symposium on Software Configuration Management*, 232–243.

Larumbe, J. (2017). Configuration Items Selection Process Review. *Ind Eng Manage*, *6*(216), 2169–0316.

Larumbe, Jaime, Barruetabeña, J. G., & Artaza, D. L. D. I. G. D. (2020). THE IMPORTANCE OF CONFIGURATION MANAGEMENT ON COMPLEX INDUSTRIAL ENVIRONMENTS. *DYNA New Technologies*, *7*(1). https://doi.org/10.6036/NT9654

Lasi, H., Fettke, P., Kemper, H.-G., Feld, T., & Hoffmann, M. (2014). Industry 4.0. *Business & Information Systems Engineering*, *6*(4), 239–242.

Lin, Y.-J., & Reiss, S. P. (1996). Configuration management with logical structures. *Proceedings of IEEE 18th International Conference on Software Engineering*, 298–307.

Locatelli, Giorgio, Invernizzi, D. C., & Brookes, N. J. (2017). Project characteristics and performance in Europe: An empirical analysis for large transport infrastructure projects. *Transportation Research Part A: Policy and Practice*, *98*, 108–122.

Locatelli, GIORGIO, Littau, P., Brookes, N. J., & Mancini, M. (2014). Project characteristics enabling the success of megaprojects: An empirical investigation in the energy sector. *Procedia-Social and Behavioral Sciences*, *119*, 625–634.

Logikal Projects Global Projects Controls Survey Report 2019. (n.d.). Major Projects Knowledge Hub. Retrieved 12 July 2020, from https://www.majorprojectsknowledgehub.net/resources/logikal-projects-global-projects-controls-survey-report-2019/

Loubersac, J., & Halliday, B. (2001). Case Study–Creating the WCRM Concept of Operations. *Idea*, 1.

Lüder, A., Pauly, J.-L., & Wimmer, M. (2018). Modelling consistency rules within production system engeering. *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)*, 664–667.

Mark, A. L. (2016). PMI's Pulse of the Profession: The High Cost of Low Performance. How will you improve business results. *Project Management Institute*.

Matta, N., Ducellier, G., Charlot, Y., Beldjoudi, M. R., Tribouillois, F., & Hibon, E. (2011). Traceability of design project knowledge using PLM. *2011 International Conference on Collaboration Technologies and Systems (CTS)*, 233–240. https://doi.org/10.1109/CTS.2011.5928692

Matthews, M., & Choi, C. Q. (2019). Compass course. *ASEE Prism*, *29*(4), 24–29.

McElroy, M. W. (2000). Integrating complexity theory, knowledge management and organizational learning. *Journal of Knowledge Management*.

Meng, W., Tischhauser, E. W., Wang, Q., Wang, Y., & Han, J. (2018). When Intrusion Detection Meets Blockchain Technology: A Review. *IEEE Access*, *6*, 10179–10188. https://doi.org/10.1109/ACCESS.2018.2799854

Mesly, O. (2017). *Project feasibility: Tools for uncovering points of vulnerability*. CRC Press.

MIL-HDBK-61 A CONFIGURATION MANAGEMENT GUIDANCE. (n.d.). Retrieved 28 July 2020, from http://everyspec.com/MIL-HDBK/MIL-HDBK-0001-0099/MIL-HDBK-61A_15177/

Miller, B., & Rowe, D. (2012). A survey SCADA of and critical infrastructure incidents. *Proceedings of the 1st Annual Conference on Research in Information Technology*, 51–56.

Monticolo, D., Badin, J., Gomes, S., Bonjour, E., & Chamoret, D. (2015). A meta-model for knowledge configuration management to support collaborative engineering. *Computers in Industry*, *66*, 11–20.

Morris, S. B. (2008). Estimating Effect Sizes From Pretest-Posttest-Control Group Designs. *Organizational Research Methods*, *11*(2), 364–386. https://doi.org/10.1177/1094428106291059

msnbc.com, A. L. S. writer. (2010, April 28). *Hundreds of suppliers, one Boeing 737 airplane*. Msnbc.Com. http://www.nbcnews.com/id/36507420/ns/business-us_business/t/hundreds-suppliers-one-boeing-airplane/

Myrodia, A., Randrup, T., & Hvam, L. (2019). Configuration lifecycle management maturity model. *Computers in Industry*, *106*, 30–47.

Narain, S. (2005). Network Configuration Management via Model Finding. *LISA*, *5*, 15–15.

Neil, G. (2014). *On board train control and monitoring systems*.

Nguyen, T. N., Munson, E. V., & Boyland, J. T. (2005). An infrastructure for development of object-oriented, multi-level configuration management services. *Proceedings. 27th International Conference on Software Engineering, 2005. ICSE 2005.*, 215–224.

Niknam, M., Bonnal, P., & Ovtcharova, J. (2013). Configuration management maturity in scientific facilities. *International Journal of Advanced Robotic Systems*, *10*(12), 404.

Ogunlana, S. O. (2010). Beyond the 'iron triangle': Stakeholder perception of key performance indicators (KPIs) for large-scale public sector development projects. *International Journal of Project Management*, *28*(3), 228–236.

Oman, C. M., Howard, I. P., Smith, T., Beall, A. C., Natapoff, A., Zacher, J. E., & Jenkin, H. L. (2003). The role of visual cues in microgravity spatial orientation. *The Neurolab Spacelab Mission: Neuroscience Research in Space*, 69–82.

Página web de Qatar Rail. (n.d.). Retrieved 10 July 2020, from https://corp.qr.com.qa/English/Pages/default.aspx

Pakko, M. R. (2000). Capital deepening. *National Economic Trends*, *may*.

Paté-Cornell, M.-E., & Dillon, R. L. (2001). Success factors and future challenges in the management of faster-better-cheaper projects: Lessons learned from NASA. *IEEE Transactions on Engineering Management*, *48*(1), 25–35.

Pikosz, P., & Malmqvist, J. (1998). A comparative study of engineering change management in three Swedish engineering companies. *Proceedings of DETC*, *98*.

PMI, A. (2004). Guide to the Project Management Body of Knowledge (PMBOK). *Project Management Institute*.

Qian, X., Jullig, R., & Daum, M. (1990). Consistency management in a project management assistant. *ACM SIGSOFT Software Engineering Notes*, *15*(6), 34–43.

Qubaisi, J. M. M. L. F. A., Elanain, H. M. A., Badri, M. A., & Ajmal, M. M. (2015). Leadership, culture and team communication: Analysis of project success causality-a UAE case. *International Journal of Applied Management Science*, *7*(3), 223–243.

Quigley, J. M., & Robertson, K. L. (2015). *Configuration management: Theory, practice, and application*. CRC Press.

Rhodes, D. H., & Ross, A. M. (2010). Five aspects of engineering complex systems emerging constructs and methods. *2010 IEEE International Systems Conference*, 190–195.

Rickabaugh, J. (1994). CONFIGURATION MANAGEMENT: THE HIDDEN FRIEND IN BUSINESS REENGINEERING-Configuration management assures the conformance of a product, service or facility to its definitive documentation and allows. *Industrial Engineering-Norcross*, *26*(8), 20–25.

Robinson, J. A., Thumm, T. L., & Thomas, D. A. (2007). NASA utilization of the international space station and the vision for space exploration. *Acta Astronautica*, *61*(1–6), 176–184.

Roblek, V., Meško, M., & Krapež, A. (2016). A complex view of industry 4.0. *Sage Open*, *6*(2), 2158244016653987.

Salama, M., El Hamid, M. A., & Keogh, B. (2008). Investigating the causes of delay within oil and gas projects in the UAE. *24th Annual ARCOM Conference*, 1–3.

Sarma, A., Noroozi, Z., & Van Der Hoek, A. (2003). Palantír: Raising awareness among configuration management workspaces. *25th International Conference on Software Engineering, 2003. Proceedings.*, 444–454.

Sauser, B. J., Reilly, R. R., & Shenhar, A. J. (2009). Why projects fail? How contingency theory can provide new insights–A comparative analysis of NASA's Mars Climate Orbiter loss. *International Journal of Project Management*, *27*(7), 665–679.

Sauvé, J., Rebouças, R., Moura, A., Bartolini, C., Boulmakoul, A., & Trastour, D. (2006). Business-driven decision support for change management: Planning and scheduling of changes. *International Workshop on Distributed Systems: Operations and Management*, 173–184.

Schuster, H., Neeb, J., & Schamburger, R. (1999). A configuration management approach for large workflow management systems. *Proceedings of the International Joint Conference on Work Activities Coordination and Collaboration*, 177–186.

Sciore, E. (1994). Versioning and configuration management in an object-oriented data model. *The VLDB Journal*, *3*(1), 77–106.

Serrador, P., & Pinto, J. K. (2015). Does Agile work?—A quantitative analysis of agile project success. *International Journal of Project Management*, *33*(5), 1040–1051.

Shandas, V., Makido, Y., & Ferwati, S. (2020). Rapid Land-Cover Change in Doha. In *Urban Adaptation to Climate Change* (pp. 13–29). Springer.

Smith, W. K., & Lewis, M. W. (2011). Toward a theory of paradox: A dynamic equilibrium model of organizing. *Academy of Management Review*, *36*(2), 381–403.

Stellman, A., & Greene, J. (2005). *Applied software project management*. O'Reilly Media, Inc.

Su, Y.-Y., Attariyan, M., & Flinn, J. (2007). AutoBash: Improving configuration management with operating system causality analysis. *ACM SIGOPS Operating Systems Review*, *41*(6), 237–250.

Sun, H., & Chang, Z. (2018). An approach to configuration management in the aero-engine overhaul process. *Aircraft Engineering and Aerospace Technology*.

Tague, N. R. (2005). Plan–do–study–act cycle. *The Quality Toolbox*, 390–392.

Tang, C., Kooburat, T., Venkatachalam, P., Chander, A., Wen, Z., Narayanan, A., Dowell, P., & Karl, R. (2015). Holistic configuration management at Facebook. *Proceedings of the 25th Symposium on Operating Systems Principles*, 328–343.

Tarr, P., & Clarke, L. A. (1998). Consistency management for complex applications. *Proceedings of the 20th International Conference on Software Engineering*, 230–239.

Team, C. P. (2002). Capability maturity model® integration (CMMI SM), version 1.1. *CMMI for Systems Engineering, Software Engineering, Integrated Product and Process Development, and Supplier Sourcing (CMMI-SE/SW/IPPD/SS, V1. 1)*, *2*.

Thomas, J. E. (n.d.). *A Case Study Analysis of the Equifax Data Breach*.

Van Der Hoek, A., Heimbigner, D., & Wolf, A. L. (1996). A generic, peer-to-peer repository for distributed configuration management. *Proceedings of IEEE 18th International Conference on Software Engineering*, 308–317.

Van Ommering, R. (2001). Configuration management in component based product populations. In *Software Configuration Management* (pp. 16–23). Springer.

Vaughan, D. (1989). Regulating risk: Implications of the Challenger accident. *Law & Policy*, *11*(3), 330–349.

Voulgaris, C. T. (2019). Crystal Balls and Black Boxes: What Makes a Good Forecast? *Journal of Planning Literature*, *34*(3), 286–299.

Wang, J., Wang, J., Roberts, C., & Chen, L. (2014). Parallel monitoring for the next generation of train control systems. *IEEE Transactions on Intelligent Transportation Systems*, *16*(1), 330–338.

Wang, Y.-M., Verbowski, C., Dunagan, J., Chen, Y., Wang, H. J., Yuan, C., & Zhang, Z. (2004). Strider: A black-box, state-based approach to change and configuration management and support. *Science of Computer Programming*, *53*(2), 143–164.

Ward, C., Aggarwal, V., Buco, M., Olsson, E., & Weinberger, S. (2007). Integrated change and configuration management. *IBM Systems Journal*, *46*(3), 459–478.

Wateridge, J. (1999). The role of configuration management in the development and management of Information Systems/Technology (IS/IT) projects. *International Journal of Project Management*, *17*(4), 237–241.

Watts, F. B. (2015). *Configuration management for senior managers: Essential product configuration and lifecycle management for manufacturing*. Butterworth-Heinemann.

Whyte, J., Stasis, A., & Lindkvist, C. (2016). Managing change in the delivery of complex projects: Configuration management, asset information and 'big data'. *International Journal of Project Management*, *34*(2), 339–351.

Wright, I. C. (1997). A review of research into engineering change management: Implications for product design. *Design Studies*, *18*(1), 33–42.