Much ado about Everything: The Configuration Management Story

Christopher A. Irvine

Abstract

200 words of text here about how awesome this is

1 Introduction

Configuration Management (CM) has helped many organisations across the globe manage the development and maintenance of complicated systems. However, the influence of CM does not stop in the office. The techniques used to incorporate the CM functions into a project have seeped out into every day life. In a similar manner, CM has taken many every day organisational techniques to heart.

In this paper we will look at where CM originated from and how it has evolved to be relevant in the industries of today (see Section 2). Then we will look how the functions of CM (see Section 3) are used both in the office and at the home (see Section 4). Before performing a critical analysis on the benefits gained from incorporating CM into a project or personal life, to try to justify the costs associated with that incorporation (see Section 5).

1.1 Defining Configuration Management (CM)

Before we continue to answer the questions outlined in Section 1, we should have an understanding of what the modern definition of CM is.

According to the *Association of Project Management (APM)*:

> "Configuration Management encompasses the administrative activities concerned with the creation, maintenance, controlled change and quality control of the scope of work." (Association for Project Management (2017))

Expanding upon this definition; CM is a collection of principles, techniques and characteristics that aim to control the execution of a project. This allows for the Project Manager to ensure that all work regarding the project is of a high quality by deploying CM techniques, ensuring that short-term targets and long-term goals are achieved.

2 History of CM

In this section we will take a brief look at where CM came from and why it was created (see Section 2.1), and how CM spread across the globe (see Section 2.2). Finally we will learn how the CM has evolved to remain relevant in the IT industry, and what standards have emerged as a result.

2.1 Origins

The origins of CM can be traced back to the U.S. Department of Defence (DoD), where the need for universal hardware standards was required in order to make maintenance of the equipment manageable. This was driven due to the mass of weapons and vehicles built during the second

world war that were each unique and had its own set of issues. No two guns or tanks were the same. This meant the DoD was spending an excessive amount of money on training mechanics to maintain the equipment. The standards was created to save money and increase efficiency.

Figure 1: William J. Perry, 19th Secretary of Defence for the United States under President Bill Clinton. Perry was instrumental in streamlining the US military infrastructure. Which formalised the discipline known as Configuration Management in private sector industries.

It was not until the 1960s that CM became a technical discipline of its own, when the DoD released a series of military standards known as the *"480 series"*. These standards were regularly updated and eventually consolidated into MIL-HDBK-61 in 1991, which contained a series of technical standards supported by standards developing organisations (SDO)(personal communication, Perry (1994)) [1].

2.2 Adoption into Industry

SDOs regulated their own industries through a collection of standards publications starting in the late 80s and early 90s. These various issues have evolved into a widely distributed an accepted standard on CM known as *ANSI-EIA-649-1998 (EIA-649)*, a venture helmed by the Electronic Industries Alliance (EIA). EIA-649 has provided the base for many specialised CM techniques since the 90s, but this document describes the five primary functions of CM (Lager et al. (2004)).

2.3 Software Configuration Management (SCM)

Software Configuration Management evolved from the practices of the late 60s when Professor Leon Pressor wrote a thesis on change and configuration control, whilst working with the DoD. He took the some of the basic concepts of CM, which are discussed in Section 3, and revised others into tools and techniques that solve the issues that were occurring during software projects. This created a set of distinct emphases that were separate from the emphases in traditional CM.

3 The 5 Functions of CM/SCM

EIA-649 (see Section 2.2) outlines 5 functions (sometimes known as disciplines) that should be enforced in both hardware and software projects. Together they establish a standard for managing the development of a project (U.S. Department of Defence (2002)).

This section will take a brief look at the 5 traditional functions of CM before looking how they differ in SCM, as detailed by the latest version of EIA-649.

3.1 CM Planning and Management

Each project should have a formal document that outlines, in detail, all process that are required for the project. This should be used as a reference for everyone involved. Such a document will include procedures for:

- personnel
- responsibilities and resources

[1] A Standards Developing Organisation (SDO) is a body whose primary activities revolve around the improvement and development of technical standards within a given field (Wang (2011)).

- training
- administrative meeting guidelines
- base lining resources (see Section 3.2)
- configuration control and configuration status accounting (see Sections 3.3 and 3.4)
- naming conventions
- audits and reviews (see Section 3.5)
- subcontractor/vendor CM requirements

These processes will be created before development of the project has begun. Any changes to this document during the life-cycle of the project should be avoided, to prevent confusion. But if changes are needed, then a Document Change Notice (DCN) should be issued to all personnel. The details of a DCN can be found within a CM document.

3.2 Configuration Identification (CI)

CI involves defining the system and subsystem architectures through the use of baselines. The procedures for changing any part of the system - how the change is identified, documented and tracked through design, development, testing and delivery - is created during this function.

3.3 Configuration Control

Configuration Control is the evaluation of all change-requests and change-proposals to the project and its documentation; including the process for approving or rejecting the changes.

3.4 Configuration Status Accounting

Configuration Status Accounting dictate the process for recording and reporting configuration item descriptions and all changes from the original designed baseline.

3.5 Configuration Verification and Audit

The final function of CM is the independent review of hardware and software to ensure that the product meets the established performance requirements outlined at the start of the project.

3.6 Alterations for SCM

SCM was created to better suit the challenges posed by Software Projects, as stated in Section 2.3. SCM places a greater emphasis on the need to trace changes in the system with the ability to verify that the delivered software does not exceed the original intent for the program. As a result the four functions for SCM are:

1. Configuration Identification (CI)
2. Configuration Change Control
3. Configuration Status Accounting
4. Configuration Audits

The first three functions of SCM are closely linked to their traditional counterparts, see Sections 3.2, 3.3 and 3.4 for details about SCM 1, 2 and 3.

Configuration Audits for SCM are separated into two halves, functional and physical, both of which can occur at either delivery or when a change is implemented. Functional audits look at a configuration item to ensure that it meets both the functional and performance requirements. A physical audit ensures that a configuration item is installed within the wider system in accordance with the system design documentation .

Notice that the CM Planning and Management function of traditional CM has been entirely removed from SCM. That is because Design Document, a Software Engineering practice, covers the majority of the same topics [2](McElrath (2007)).

4 Using CM

In this section we will examine how the functions of CM can be utilised within work and personal projects, touching briefly upon the techniques used to accomplish CM.

4.1 In the Office

Historically configuration management has required a dedicated employee, or team of employees for larger projects, in order to function. Typically they would be known as Administrators. This is still true today for complicated projects carried out by large organisations. Outsourcing is another method in which configuration management can be deployed, where an external companies will handle the administration of a project. Small start up firms will incorporate the task of configuration management into a senior member of the development team. In any case, the tools and techniques used will remain the same (Aiello and Sachs (2010)).

Each function of CM produces a formal product. The first stage (see Section 3.1) creates a document, much like a design document, that informs individuals how to contribute to a project in a way that allows others to understand what was changed. The Administrator of a project creates this document and will ultimately decide how tasks will be completed. There are a range of tools to assist in this task such as *Chef* (Chef Software Inc. (2018)) or *Puppet* (Puppet (2018)) which take a lot of the burden away from the administrator.

Functions two through five produce a variety of reports. The creation of these documents is detailed by the CM document produced in the first stage. For example; when developing software in a modern formal setting there are two versions of the project at any given time (in most cases), a *master* copy and a *development* copy. When an edit is created it is tested according to the baselines detailed in CI (see Section 3.2) within the development build. The results of the test are recorded, typically by a tool such as *Codeship* (CloudBees (2018)). Assuming the tests pass, a request is made to update the Master copy with the new edits. The industry standard is to use a *GitHub pull request* (GitHub, Inc. (2018)) to accomplish and record this, including any last minute edits. Finally at the end of each completed feature a review (or audit) of the project is performed to ensure that the work accomplished is relevant to the original specification.

The above example shows that by producing reports from CM functions two through five and adhering to the guidelines set out by the CM document (see Section 3.1) the functions of CM will be automatically applied to any project. However, the tools will cited in the example will differ from the tools used in a non-software based project.

[2]A Design Document is a technical guide for Developers to use before, during and after Developing a piece of Software.

4.2 At Home

As discussed in Section 2, the primary function of CM is to ensure that tasks which affect a project are performed in a near identical manner so that individuals can collaborate with minimal training. In the office this could be a trivial task such as how a testing report is filed, to a critical task as to what testing parameters will be used.

But the lessons that are learned from CM in industry can be applied to home life as well. Keeping the cleaning supplies in a certain cupboard, writing groceries on a piece of paper as they are used up and having a filing system for important documents are examples of how CM has infiltrated many homes. The majority of families will not realise that they are using tried and tested CM techniques to improve the efficiency of their households. Is it possible to incorporate CM further into our daily lives?

Figure 2: An example shopping list that is a prime example of CM techniques being used in home life (Tumisu (2017)).

A lot of CM revolves around lists; to do, in progress, audit, done lists are all common place in CM projects. This is because if we tried to remember everything about a project without the aid of lists we would be lost. This is especially true when executing complicated tasks like engineering a bridge or designing logic to control a function. The same can be said for home life as shopping lists (as seen in Figure 2) are common place but why not have a general *to do list*. Expand that list in to a weekly plan on your calendar (app or physical). It would be foolish to execute an industrial project without a detailed plan, personal lives are no less complicated.

Another key component of CM is knowing where everything related to a project is. Being able to look quickly located bug and testing reports can save a lot of time. The same can be true for knowing exactly where the bleach or light bulbs are kept in the house. It is not enough to know which cupboard these items may be located in, as you would not expect the bug reports to be left in one general folder, the reports may be kept in a dated folder and the files timestamped with the summary of the issue. So why not have baskets in your cleaning supplied cupboard, each with their own dedicated label. Benjamin Franklin once said "A place for everything and everything in it's place", this is very true for keeping an efficient household (Akhila K E (2016)).

5 Embracing CM

We have already discussed what CM is and how it came to be in Sections 2 and 3. Section 4 covered how CM is utilised in a project. This section will discuss the benefits of CM and the costs of incorporating it into a project.

5.1 Benefits of CM

As stated throughout this paper the principle benefit of CM is that it allows for individuals to contribute to a project without disrupting other contributors with minimal training. Increased efficiency and communication can be experienced within teams, both internally and externally (Walker (2016)). Section 4.1 states that each function of CM produces a formal product, usually in the form of a report. These reports form a paper trail that records every detail of a project, which in turn can be reviewed to see where the project (and the development methodologies used) can be improved for future use. Products that were built using CM tend to be more

reliable due to the higher level of scrutiny that is inherited from the use of CM techniques (Giva (2016)).

5.2 Costs

CM is not free, there are associated costs with it. Creating the CM document from the first function is a huge undertaking (see Section 3.1). We can either purchase a licence to a tool that will create our CM rules, or hire skilled administrators to write the document from scratch. No matter what we do it will cost both money and time. We may also need additional licences for tools to generate the reports for the other five functions of CM, and administrators to ensure that the CM system is working in an optimal manner. So there is a huge time and cost investment required to get CM off the ground (UpGuard (2017)). Chef, a popular automated CM tool has a licence cost of $137 per node (per computer using Chef) (Chef Software, Inc (2018)).

6 Conclusion

References

Aiello, B. and Sachs, L. (2010). *Configuration Management Best Practices: Practical Methods that Work in the Real World*. Pearson Education.

Akhila K E (2016). Life under configuration control.

Association for Project Management (2017). Configuration management.

Chef Software Inc. (2018). Chef.

Chef Software, Inc (2018). Chef pricing.

CloudBees (2018). Codeship.

GitHub, Inc. (2018). Github.

Giva (2016). The benefits of configuration management.

Lager, A., Brummer, B., Noble, D., et al. (2004). Eia-649: A national consensus standard for configuration management.

McElrath, R. (2007). Xml legal document utility software design document.

Perry, W. J. (1994). Specification & standards: A new way of doing business.

Puppet (2018). Puppet.

Tumisu (2017). Shopping list.

UpGuard (2017). 5 things about configuration management your boss needs to know.

U.S. Department of Defence (2002). *Military Handbook: Configuration Management Guidance*.

Walker, A. (2016). Unexpected benefits of configuration management software.

Wang, P. (2011). A brief history of standards and standardization organizations: A chinese perspective.