AN INFORMATION SYSTEMS PROJECT MATURITY FRAMEWORK FOR LEVEL 2 COMPLIANCE

by

MARK JONATHAN CORRIGAN

submitted in accordance with the requirements

for the degree of

MASTER OF SCIENCE

in the subject of

COMPUTING

at the

UNIVERSITY OF SOUTH AFRICA

Supervisor: Professor John Andrew van der Poll

Co-supervisor: Dr. Emmanuel Samuel Mtsweni

February 2022

DECLARATION

Name: Mark Jonathan Corrigan

Student number: 04151690

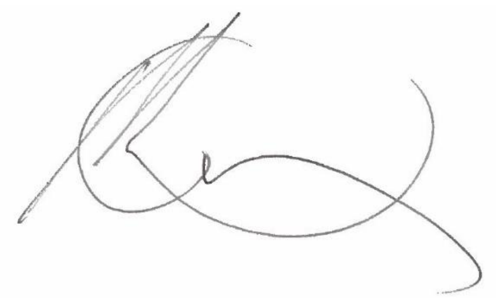
Degree: MSC Computing

Title: An Information Systems Project Maturity Framework for Level 2 Compliance.

I declare that the above dissertation is my own work and that all the sources that I have used or quoted have been indicated and acknowledged by means of complete references.

I further declare that I submitted the dissertation to originality checking software and that it falls within the accepted requirements for originality.

I further declare that I have not previously submitted this work, or part of it, for examination at Unisa for another qualification or at any other higher education institution.



08 October 2021

SIGNATURE **DATE**

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

PUBLICATION

The following publication emanated from the research in this dissertation:

Corrigan M.J., van der Poll J.A., Mtsweni E.S. The Project Management Information System as Enabler for ICT4D Achievement at Capability Maturity Level 2 and above. In: Krauss K., Turpin M., Naude F. (eds) Locally Relevant ICT Research. IDIA 2018. *Communications in Computer and Information Science*, vol 933. Springer, Cham. https://doi.org/10.1007/978-3-030-11235-6_19

(Corrigan, Mtsweni & van der Poll, 2018)

ABSTRACT

From a perception of project management being challenged, current trends often promote agile methods as a solution for problems surrounding software project delivery, focusing more on individuals and interactions than on processes and tools. The Capability Maturity Model from Carnegie Mellon University explains that any project method if undertaken with limited process focus below Capability Maturity Level 2, may often, despite best intentions, result in project delivery disappointment and failures.

With the dynamic global growth of the internet, including advancements in internet-enabled systems facilitating virtual collaboration, the utilization of an internet-enabled Project Management Information System (PMIS) emplacement could be an essential tool to facilitate improved process application by project teams who want to perform at Capability Maturity Level 2 and above. Correctly installed and operating optimally, the PMIS emplacement could be a launchpad for the attainment of software project management excellence above Level 1, irrespective of the project methodology used.

The output of this research is a PMIS Capability Maturity Improvement Framework to enhance the current PMBOK® 6 Process Group and Knowledge Area matrix. The framework is envisaged as an implementation guide and checklist to achieve project management process stability at Level 2 and above. While compliance with project processes at Level 2 is this dissertation’s primary concern, other processes for software engineering, the IT Infrastructure Library (ITIL) processes and others could also be introduced and managed in the PMIS emplacement. The researcher argues that access to Capability Maturity Level 2 process stability could remain elusive unless, and until, the PMIS emplacement becomes a strategic driver for success. Optimization of the PMIS emplacement will be achieved if driven down into the organisation with executive oversight.

KEY TERMS

Capability Maturity Model integrated, Project Management Body of Knowledge, Project Management Information System, Project Planning, Project Execution, Project Monitoring and Control, Continuous Improvement, DevOps, Game Theory, Performance Metrics.

OPSOMMING

Huidige tendense behels dikwels die bevordering van buigsame metodes (*agile methods*) as ’n oplossing vir probleme rakende die afhandeling van sagtewareprojekte, wat behels dat meer klem op individue en interaksies as op prosesse en hulpmiddels geplaas word. Die Volbringingsvermoëmodel (Capability Maturity Model) van die Universiteit van Carnegie Mellon is gegrond op die veronderstelling dat enige projekmetode wat met ’n beperkte prosesfokus onder vlak 2-volbringingsvermoë toegepas word, dikwels, ongeag goeie bedoelinge, op teleurstelling en mislukkings ten opsigte van projekafhandeling uitloop.

In die lig van die dinamiese uitbreiding van die internet wêreldwyd, insluitend vooruitgang op die gebied van internetgebaseerde stelsels wat virtuele medewerking aanhelp, kan ’n internetgebaseerde projekbestuur-inligtingstelselinplasing dien as ’n noodsaaklike hulpmiddel om verbeterde prosesuitvoering deur projekspanne wat op vlak 2-volbringingsvermoë en hoër wil funksioneer, te bevorder. Indien ’n projekbestuur-inligtingstelselomgewing korrek in werking gestel word en optimaal funksioneer, kan dit ’n wegspringplek wees vir die bereiking van sagtewareprojekbestuur-uitnemendheid bo vlak 1, ongeag die projekmetodologie wat toegepas word.

Hierdie navorsing het gelei tot die ontwikkeling van ’n inligtingstelsel-projekvolbringingsraamwerk vir ’n projekbestuur-inligtingstelselomgewing wat daarop gerig is om die huidige PMBOK® 6-prosesgroep-en-kennisarea-matriks (PMBOK® 6 Process Group and Knowledge Area *matrix*) aan te vul. Dit word beoog dat die raamwerk as ’n implementeringsriglyn en -kontrolelys vir die bereiking van projekbestuurprosesbestendigheid op vlak 2 en hoër sal dien. Hoewel hierdie verhandeling hoofsaaklik op die nakoming van projekprosesse op vlak 2 toegespits is, kan ander prosesse vir sagteware-ingenieurswese, die prosesse van die IT-infrastruktuurbilbioteek (IT Infrastructure Library) en ander prosesse ook in ’n projekbestuur-inligtingstelselomgewing in werking gestel en bestuur word. Die navorser doen aan die hand dat prosesbestendigheid wat aan vlak 2-volbringingsvermoë voldoen, buite bereik sal wees tensy, en totdat, ’n projekbestuur-inligtingstelselomgewing ’n strategiese aandrywer van sukses binne ’n organisasie

word. Die optimalisering van ’n projekbestuur-inligtingstelselomgewing in ’n organisasie kan met behulp van uitvoerende bestuurstoesig bewerkstellig word.

HOOFTERME

Volbringingsvermoëmodel, Kenniskorpus aangaande projekbestuur, Projekbestuurinligtingstelsel, Projekbeplanning, Projekuitvoering, Projekmonitering en -kontrole, Deurlopende verbetering, DevOps, Spelteorie, Prestasiemetriek.

NGOBUFITJHAZANA

Izenzo ezisiqhelo zagadesi kanengi zifaka ukuphakanyiswa kwemethodo enjengesisombululo semiraro emalungana nokwenziwa kwephrojekthi ye-*software*, ngokutjheja khulu abantu kanye nokuhlangana kwabantu kunokuthi kutjhejwe amahlelo kanye namathulusi. IModeli yokuVuthwa kweKghono lokuSebenza (*Capability Maturity Model*), lokhu kuvela e-Carnegie Mellon University, ehlathulula bona omunye nomunye umethodo wephrojekthi, nangabe wenziwa ngehlelo elinemikhawulo eliqale ngaphasi kwezinga le-*Capability Maturity Level 2*, naphezu kokuba neenhloso ezingcono, ngokuvamileko, lidala ukuphoqakala kokwenziwa kwephrojekthi kanye nokuhluleka kwephrojekthi.

Malungana nobujamo obutjhugulukako bokukhula kwe-inthanede ephasini, kufakwa phakathi ukuthuthukiswa kwamasistimu asizwa yi-inthanede ekghonakalisa isebenziswano lehlelo le-inthanede, ukusetjenziswa kweSistimu yeLwazi lezokuPhathwa kwePhrojekthi (*Project Management Information System (PMIS*) kungaba lithulusi eliqakathekileko lokusiza ukusetjenziswa kwehlelo elithuthukiswe ziinqhema zephrojekthi ezifuna ukusebenza eZingeni lesi-2 lokuVuthwa kweKghono lokuSebenza (*Capability Maturity Level 2*) nangaphezulu. Nayifakwe kuhle begodu yasebenza kuhle, ukusetjenziswa kwe-PMIS kungaba lithulusi elisiza ukufikelela ngaphezu kweZinga loku-1 elihle khulu lokuphathwa kwephrojekthi nge-*software*, ngaphandle kokuqala imethodoloji yephrojekthi esetjenzisiweko.

Umphumela waleli rhubhululo ku-*PMIS Capability Maturity Improvement Framework* kukuqinisa ihlelo lagadesi le-PMBOK® 6 Process Group kanye ne-*Knowledge Area matrix*. Isakhiwo sithathwa njengomhlahlandlela osetjenziswako kanye nerhelo lokutjhejisisa (*checklist*) ukuphumelela ukunzinzisa ihlelo lokuphathwa kwephrojekthi ukobana libe seZingeni lesi-2 begodu nangaphezulu. Njengombana ukulandelwa kwamahlelo wephrojekthi eZingeni lesi-2 kumnako omkhulu wedizetheyitjhini, amanye amahlelo wobunjiniyere be-*software,* amahlelo woMthangalasisekelo weLayibhrari ye-IT (*IT Infrastructure Library (ITIL*) kanye namanye bekangathulwa begodu aphathwe ekusetjenzisweni kwe-PMIS. Umrhubhululi uyatjho ukuthi ukutholakala kwenzinzo le-*Capability Maturity Level 2* kungabonakala kukhohlisa, ngaphandle kokuthi, ekugcineni, ukusetjenziswa kwe-PMIS kube mtjhayeli omkhulu wepumelelo..

Ukusetjenziswa kuhle kwe-PMIS kuzokuphunyelelwa nangabe kusunduzelwa ukobana kube yihlangano enesigungu esitjhejisisako.

AMAGAMA AQAKATHEKILEKO

IModeli yokuVuthwa kweKghono lokuSebenza, Iziko leLwazi lezokuPhathwa kwePhrojekthi, ISistimu yeLwazi lezokuPhathwa kwePhrojekthi, UkuHlelwa kwePhrojekthi, UKwenziwa kwePhrojekthi, UkuTjhejisiswa kanye nokuLawulwa kwePhrojekthi, UkuThuthukiswa okuRagela Phambili, IHlanganisela yeSiko lokuSebenza kanye namaThulusi (*DevOps*), IThiyori emalungana nesayensi yobujamo bomphakathi (*Game Theory*), IMethriksi yeZinga lokuSebenza/lomSebenzi.

KEY DEFENITIONS

Bidirectional Traceability of requirements	Bidirectional traceability is the ability to trace forward (from requirement to test case) and backward (from test case to requirement). Traceability should be bidirectional. It establishes a relationship between two artifacts. And it's important to be able to trace from one item to the next and back again.
CMMi	The Capability Maturity Model integrated. This model comes out of work undertaken by the Carnegie Mellon University Software Engineering Institute.
CMMi Level 2 Processes	The Level 2 processes of the Capability Maturity Model (at Level 2) are: Project Planning (PP), Project Monitoring & Control (OMC), Requirements Management (REQM), Configuration Management (CM), Measurement & Analysis (MA), Process & Product Quality Assurance (PPQA) and SAM (Supplier Agreement Management).
CMMi Level 1 - heroic	CMMi Level 1 means processes defined in CMMi Level 2 to 5 of the Capability Maturity Model are not being followed. At an essential level these processes are Planning, Monitoring & Control, Verification & Validation against an approved project Requirement. Essentially no plan is being followed. Or the plan is at such a high level of abstraction (lacking detailed project tasking on a developed and managed project schedule) that the plan has no value.
Developed schedule	A developed schedule is one that has tasks on it which are based on a plan. The idea is that the schedule is baselined first to lock down the plan and only then does project execution begin. In traditional project management in the Project Management Body of Knowledge there are 24 planning processes. In agile (Scrum method as an example of agile) the plan is based on an agreed next

	sprint which project team board is full of accepted User Stories in the form of yellow post it notes.
Project Task (Time Ask)	A project task or “time ask” refers to the act of a project manager or team tasking a willing human resource to do work on a project. Typically, a task has a cost associated with it and work needs to be started, changed, and stopped by the estimated time (obtained from the person being tasked) when the goal of the task is achieved.
Red Beads	Edwards Deming’s Red Bead Experiment (Annexure F) refers to Red Beads which is understood by this work as a lack of or unproductive processes. Work is expended but a productive result is not being achieved.
Silver Bullet	Magical solutions are solutions that are fantastic and not based on logic, truth, or reality. In this regard, the term “silver bullet” is also used in software engineering to indicate that a magical solution pretending to be a silver bullet or quick fix will be needed to slay the magically protected werewolf (problem), which is not realistic.
Tar Pit	Brooks made a comparison of software engineering complexity being like the danger of being trapped in a “tar pit.” Examples are found of the bones of ancient beasts who wondered into tar pits, unaware of the dangers, who became stuck and thrashed around desperately trying to escape but eventually died there.
PDCA (Plan, Do, Check Act) Cycle	PDCA is an iterative design and management method that is used in business for the control and continuous improvement of processes and products. It is also known as the Deming circle/cycle/wheel, the Shewhart cycle, the control circle/cycle, or plan–do–study–act.

CHAPTER 1:
INTRODUCTION

1.1 Overview

This is a dissertation on the use of the Project Management Information System (PMIS) emplacement to assist project teams to achieve improved levels of productivity and quality. The dissertation focuses on the potential of the PMIS to enable project management process improvements for stability at Capability Maturity Level 2 (CM L2).

Motivation for the research came from Rosser et al. (2005:213), who sparked interest in the PMIS with their finding that 75% of large IT projects that were managed with the support of a PMIS would be successful while 75% without would fail. According to Kostalova, Tetrevova & Svedik (2015:98), the PMIS has value for a project team as it facilitates the ability to track projects from conception through execution to conclusion. Benefits should increase when multiple projects and teams coordinate and collaborate on a PMIS system. Raymond & Bergeron (2008:7) contend the PMIS contributed significantly to project success and should continue to receive attention in project management research efforts. The research instrument presented in Chapter 4 could be another step on this journey.

The dissertation aims to show that through the underpinning support and firm base provided by the system, a PMIS could promote improved project management process execution if its operation was tightly coupled with a process focus and managed systematically in a "stepwise"[1] manner. The coupling of the PMIS with process focus should bring about quality and productivity improvements, which could enhance project transparency and stability. Ultimately, the dissertation asks if the PMIS could be a panacea to counter negative perceptions and criticism of traditional project management, especially when planning and estimating fails to achieve the goals and value that has been promised.

---

[1] "Stepwise" application of processes ensures that earlier process outputs are managed to feed into later processes.

Edwards W. Deming (1986:248) indicates that in his experience, most of the troubles, and most areas or possibilities for improvement, which he estimates at being above 80%, belong to the system, which is the responsibility of management. Therefore, once the problem arrives at the shop floor, workers will have little ability to control or improve anything. Deming's well-known Red Bead Experiment and 14 Observations for Management, synthesized for the reader in Annexure F and G, explains his thinking and approach in this regard. Consequently, managerial insistence on heroics[2], if through managerial exhortations[3] and targeted within a CM L1 environment, would simply imply that effort expended by under-empowered and inadequately skilled personnel, despite their best efforts, would not have the desired effect and may result in project failures. Furthermore, the well-intentioned use of agile, or any project method undertaken at CM L1, could also result in project products having little value, especially where complex software solutions are the endeavour. The characteristics of the capability maturity levels, where defined processes are installed systematically at a specific level to improve productivity and quality, are illustrated in Annexure C.

Since the original Standish Group Chaos Report (1995), of which insights are updated regularly in subsequent reports, these appear well-received and garner interest and attention. To address the unique demands in Information Technology, whilst responding to criticisms levied against the reports by Everleens & Verhoef (2010) and Glass (2005), software projects often utilize agile methodologies instead of traditional project methodologies in attempts to produce value faster and to circumvent the problems identified in the reports. The agile approach aims to empower the tenets of the Manifesto for Agile Software Development from Sutherland, Schwaber, Highsmith, Cockburn *et al*., (2001) through the use of time boxing, iterative and incremental releases and other techniques. Recent articles have started to question if Agile can do the job it purports, as is found in *Dear Agile, I'm Tired of Pretending* (2019), and some sceptics of Agile have begun to refer to the method as “fragile” or “aino” (agile in name

---

[2] In this dissertation, “heroics,” or being pressured by management to “just do it,” is understood as operating without due consideration for process application at Capability Maturity Level 1. If Deming's wisdom discussed on this page is accepted, then management, rather than disempowered workers, needs to change the system. The concept of heroic effort is illustrated in the model in Annexure C.

[3] Edwards W. Deming refers specifically to the problems pertaining to managerial exhortations to do better. Annexure G refers to this in point 10.

only), there is no denying the new focus and interest in more lean and agile ways of working.

On the other hand, the Dunning-Kruger effect identified by Kruger & Dunning (1999:1) illustrated with Figure 2.5 explains the finding that people tend to hold overly favourable views of their own abilities and expertise in many social and intellectual domains. The Dunning-Kruger effect may have implications for the critics of Traditional Project Management because a shift away from a reliance on following project process could inadvertently be causing some of the chaos reported in the Standish Reports. Ultimately, in terms of project management and software engineering process competence, while it may be challenging to operate and sustain process quality and consistency at CM L2, the use of the PMIS to focus and automate process improvements for success, regardless of the project method selected, must be desirable.

Section 1.2 aims to present a closer look at how the PMIS could be used to facilitate Capability Maturity improvements.

1.2 The PMIS to facilitate Capability Maturity improvements

The recent dynamic growth globally of the internet must unlock the numerous benefits of a Project Management Information System (PMIS) emplacement, accessible in real-time by a collaborating team of software engineering professionals. The PMIS can definitely assist with improvements in the application of project processes at Capability Maturity (CM) Level 2 and above, and it can also be used to focus on improvements elsewhere, such as software engineering processes, ITIL 4 processes, COBIT processes, and others. As the research instrument (a PMIS CM Improvement Framework) in Chapter 4 suggests, the PMIS could also be used to offer decision support in the project management areas of staffing, training, administration, good governance, etc.

As Capability Maturity models are designed around ranked processes to assist with quality improvements, increased process proficiency at CM L3, which leverages off L2, should be possible if L2, as a foundation, is firmly in place and sustained. This should further assist with increased productivity and higher quality of projects whilst reducing risk, rework, and waste; an idea which is explained in Annexure C. As several CM models exist, the Capability Maturity integrated (CMMi) model from Carnegie

Mellon University is the preferred model for the purposes of this dissertation, as it incorporates not only software development processes, but also amalgamates processes from the services and acquisitions constellations shown in Figure 2.4. To this end, Annexure C is a simplified version of the CM model, and Annexure D contains more detail on the processes.

As CM L1 is an “Initial” state of process maturity (Figure 2.3) the assumption is that it does not focus on the installation of project management processes, which only starts formally at CM L2. If the CMMi model is followed, then the challenge for a collaborating project management team is to move out of CM L1 to achieve stable operations through the application of the project processes that are defined at CM L2. The researcher understands that from the initial CM L2 processes the first, and recommended order of implementation should be: Requirements Management (REQM), Project Planning (PP) and Project Monitoring and Control (PMC). With these processes under control and operating nominally, the team should be well-situated to implement the other CM L2 processes, which include Configuration Management (CM), Measurement Analysis (MA), Process and Product Quality Assurance (PPQA), and Supplier Agreement Management (SAM).

Ultimately, as any project management method could be used on a PMIS at CM L2, the premise is that both Traditional and Agile Project Management methods, with project processes that are defined, can benefit from a PMIS emplacement. The PMIS emplacement, if implemented successfully and operating well, should enhance the ability of project teams to use the project processes of their chosen methodology more efficiently and effectively to deliver value more consistently.

Sections 1.3 and 1.4 will discuss why the PMBOK® Guide is a valuable resource from the perspective of understanding traditional project management processes and being able to use them correctly to achieve higher standards of excellence in project execution. The Project Management Institute also has an Agile Certified Practitioner (ACP) qualification, which focuses on delivering projects successfully when using agile methods and processes. The ACP focuses primarily on the Scrum project method, Extreme Programming, and the use of a Lean mindset and Kanban to understand and improve feature flow through a targeted information system.

1.3 **The PMBOK® 6 Guide**

The Project Management Body of Knowledge PMBOK® Guide, produced by the Project Management Institute (PMI), can be viewed as an official ANSI standard for project management (2019). Annexure B explains how one could place the PMBOK, the Capability Maturity Model, and other methods and frameworks within a constantly evolving Standards, Frameworks and Methodologies landscape.

From a project management perspective, unlike “Business as Usual” (BAU), the sixth (6th) edition of the PMBOK® Guide continues to define a project as “a temporary endeavour undertaken to create a unique product, service, or result” (2017:41). Essentially this means that any project work that falls within the auspices of “BAU” may undermine the tenets of the PMBOK Guide. The goal of the PMBOK is understood by Project Management Professionals as the ability to hit the project performance target (Figure 2.8) with appropriate use of project management processes to manage project delivery within project constraints, as is illustrated in the figure below.

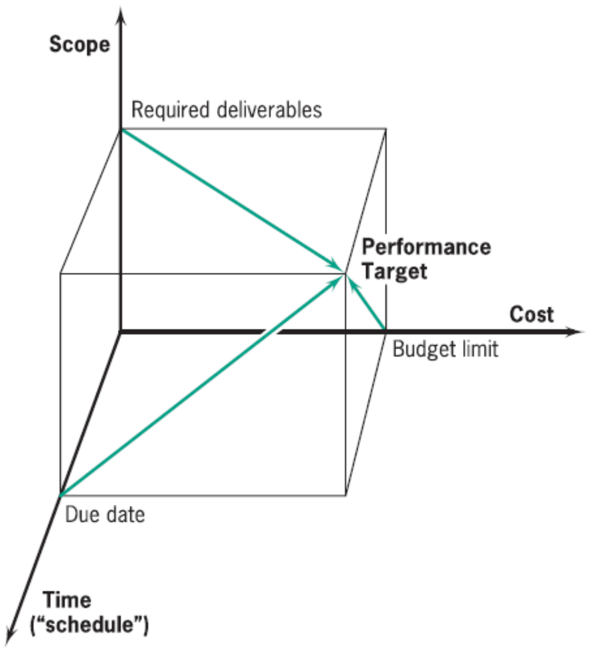


Figure 1.1: The Triple Constraint

Source: Meredith & Mantel.(2012)

Figure 1.1 explains that project management, as a temporary endeavour, must consistently manage the effects of scope, time, and cost amongst other constraints if it is to break even from a budget perspective and deliver value. Ultimately, if cost, time, and scope were not constrained, then work undertaken is not a project and must rather be classified as BAU.

It is the researcher's experience that a trend appears to be emerging, where agile projects are used to house and channel BAU work. Also, in the researcher's experience, as the annual budget allocation for agile projects is depleted and delivery is not forthcoming, milestones for urgent delivery can be imposed by management onto agile projects. This undue pressure on the software development team can disrupt the team's ability to produce quality outputs. In addition, the tenets of the Agile Manifesto will also be undermined as an agile team's ability to produce value, without a detailed plan in place, should proceed iteratively and incrementally, learning the best way forward through trial and error. For a Scrum team, this is the ability to apply the principle of empirical process control[4], amongst others.

Section 1.4 focuses specifically on the Process Group and Knowledge Area Matrix of the PMBOK® Guide. The Matrix (as a table) simplifies project process complexity to facilitate understanding of the PMBOK processes; how they fit into and flow through the Knowledge Areas and Process Groups.

1.4 The PMBOK® 6 Process Group & Knowledge Area Matrix

The 49 processes of the PMBOK® 6 are collated in a matrix that combines five Process Groups horizontally with ten Knowledge Areas (including one Integration Area) vertically. Supporting each node within this matrix are Process Inputs, Process Tools and Techniques, and Process Outputs. The PMBOK matrix is regarded as an invaluable tool and checklist for Project Management Professionals as the matrix clearly shows the processes and process flow required to run projects professionally and successfully.

The matrix in Annexure S is adapted from the PMBOK® 6 Guide (2017:25 & 556). The researcher inserted the arrows on the matrix to show that the starting and focus point for a PMIS emplacement must be the maturation of the developed schedule (PMBOK

[4] Empirical Process Control forms one of the six Scrum Principles: Scrum prescribes making decisions based on observation and experimentation rather than detailed upfront planning.

process #6.5) in the Planning Process Group and moving this finalized output into the Execution Process Group.

The Developed or Final Schedule (as opposed to a Preliminary Schedule still being finalized), when ready, receives attention from a collaborating project team, who then moves the project into the Execution Process Group as Directed and Managed Project Work (PMBOK process #4.3) (Key Definitions). Owing to the requirement of the Developed Project Management Plan (PMBOK process #4.2 - collating all processes in this Planning Process Group) and the developed schedule (PMBOK process #6.5) as essential finalizing steps, and the fact that less than half of the total number of processes in the matrix are planning processes, it follows that some planning ought to precede Execution in project management. This idea is expanded in Section 1.6 below: The Deming Plan Do Check Act (PDCA) Cycle.

If the detail in these 49 processes may appear complex, perceived by many as contradicting the concept of agility, it is central to the understanding of the PMBOK; that a Project Management Professional be able to tailor processes to achieve speed and agility within project constraints. The concept of tailoring, explained in the PMBOK guide (2017:2), is the ability to determine the appropriate combination of processes, inputs, tools, techniques, outputs, and life cycle phases to manage a project. The process node Close Project or Phase (PMBOK process #6.7) on the PMBOK matrix in Annexure S illustrates that, if required, a project phase can easily be created, conforming to the requirements of a short agile iteration, which in agile or scrum methodology is referred to as a time box, iteration or sprint.

Ultimately, if the planning process to produce a project plan (PMBOK process #4.2) from a Traditional Project Management perspective is an amalgamation of the other planning processes by Knowledge Area, then shorter agile plans can either approximate or ignore this important step. As proponents of the Agile Manifesto prefer responding to change over following a plan, it is understood that even a two-week agile iteration, or sprint as it is called in the Scrum methodology, while short, should be based on a two-week plan of prioritized User Stories the team are prepared to build. Scrum methodology and the processes used to create and manage a Scrum sprint is found in Annexure O**.**

If the core CM L2 processes of Requirements Management (REQM), Project Planning (PP) and Project Monitoring and Control (PMC) are considered, and project management did not involve planning, then project activities undertaken without planning would mean that these activities occur at CM L1. In addition, if a complex software solution receives limited attention from the perspective of adequate and correct requirements gathering and planning, it follows that the inputs and outputs of the Project Planning processes pertaining to the Project Scope Management Knowledge Area, in Annexure S, have been neglected. If this occurs, then the inputs and outputs of the corresponding Project Monitoring and Control (PMC) processes will be limited in their ability to operate successfully, thus compromising the value the project can produce. Figure 2.10 explains this idea as a quality gap that will need to be navigated.

The PMBOK Process Group and Knowledge Area Matrix also gives an overview of where a project is at any time within its project management lifecycle via its process groups, be it Initiation, Planning, Execution, or Closing. This overview of project status within the process groups reinforces the concept that a project should always be regarded as a temporary endeavour with a clearly defined start and finish. In addition, if a project is being managed, the matrix can be used to focus the Knowledge Areas as possible constraints to project success, consistently monitored and controlled for safety and success.

Often the PMBOK matrix is simplified to the concept of the Iron Triangle, focusing primarily on the Time (Schedule), Cost and Scope processes. Despite opinion stating that the Iron Triangle is no longer relevant for project measurement of success, Pollack et al. (2018) found that the criteria of being on time, within or under budget, and delivering project scope to a defined quality, still holds to our understanding as to a project’s current status and its ability to deliver the planned results.

Section 1.5 focuses on the crucial importance of being able to manage the project task, which the PMIS can assist us with. The project task is seen as the lowest unit of project control. The project task is like the way a single financial transaction is regarded in financial management. Project tasks, like financial transactions, can be individually costed on the PMIS Schedule and each, like building blocks, will contribute towards the overall success of the project or endeavour.

1.5 The Project Task

Tasks are understood to be the next project steps to be undertaken to achieve project goals. Tasking assumes initiative and accountability for completion is transferred to a willing, competent, and professional human resource. Traditionally this was undertaken on a command-and-control basis by a delegating project manager. However, now, a responsible, self-organising agile team, who takes the initiative, can decide which tasks should be prioritised.

Oncken & Was (1999:1) discuss the problems associated with tasking. They state it is a “monkey on-the-back” analogy, where managers can transfer initiative back to their subordinates and keep it there. Oncken & Was state that before developing the initiative in subordinates, the manager must see to it that they have the initiative. From this article, the researcher understands that if the manager takes the initiative back, they will no longer have discretionary time, giving this up to subordinate-imposed time. Izmailov, Korneva & Kozhemiakin (2016:97) were also aware of the negative effect of what they refer to as “bad multitasking” on project success from the theory of constraints perspective. Accordingly, companies appeared happy to accept that multitasking happened. Also, that project resources were often forced to have open tasks focused on work happening at the same time, but for different projects. Unmanaged though this can lead to what is referred to as the "cascade effect", which will decrease project efficiency and increase project duration, delaying the project.

From the Project Management Information System (PMIS) perspective, the project task is understood to be an essential unit of effort captured in the system to deliver value in software (or other) project types. The project task is defined as a unit of work undertaken (started, changed, and stopped) by a willing human resource on a developed schedule (PMBOK process #6.5, Annexure S). Project tasking (Key Definitions) on a project schedule essentially asks: who will do what, with what, by when? The time estimate to do a task of work is typically entered into the Preliminary Project Schedule (PMBOK #6.5, Annexure S) by a person agreeing to do the work. Thereafter, as a final step in creating the developed schedule, the estimate is agreed upon, and a baseline of the schedule is saved. The PMIS should be able to automate this process.

Example 1.1 shows how tasking could work on the PMIS.

Example 1.1

Figure 1.2 graphically illustrates how project task management can be undertaken on a PMIS emplacement. The Gantt chart view of the task is a construct of the typical project constraints. Information can be assimilated briefly through the visual representation.

Task Name	Duration	Start	Finish
Task 1 75% complete, behind schedule	1.5 days	Thu 19/03/15	Fri 20/03/15
Task 2 completed on schedule	2 days	Thu 19/03/15	Fri 20/03/15
Task 3 behind schedule, 50% complete	5 days	Thu 19/03/15	Wed 25/03/15
Task 4 very behind, not started	2 days	Fri 20/03/15	Mon 23/03/15
Task 5 ahead of schedule	5 days	Fri 20/03/15	Thu 26/03/15
Task 6 on track	2 days	Tue 24/03/15	Wed 25/03/15

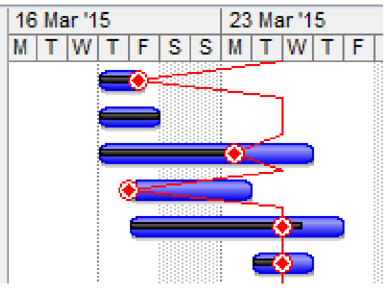


Figure 1.2: The Slipped Task in a Gantt chart view

Source: Stakeholdermap.com

The figure shows several estimated tasks as horizontal bars on the Gantt chart. Completed tasks are blue, with a black bar running from left to right. A red line runs vertically down, indicating the current or a selected date. Uncompleted, late and overdue tasks are identified by the slip line that kinks to the left. The slipped project task, one of the various views that can be configured on a PMIS, is a problem for the project team that requires urgent attention. From the researcher’s perspective, the ease of use and benefits of rapid feedback via visual telemetry at task level are key features of the PMIS emplacement. In the same way that a driver can respond quickly to visual feedback from dials of a console in a motor vehicle to avoid accidents, so too a project team can use the PMIS emplacement to keep their project operating nominally and on track against the plan.

Section 1.6 uses Deming’s Quality Cycle to tie Project Planning (PP) and Project Monitoring and Control (PMC) back to the PMBOK Matrix (Annexure S). The challenge is that if PP and PMC are not implemented effectively by the PMIS emplacement, then project delivery at CM L2 will not be achievable. In addition, without the Quality Cycle, the concept of adjustments for improvements in quality may not be possible.

1.6 The Deming Plan Do Check Act (PDCA) Cycle

The PMBOK matrix illustrates the importance of the Project Planning (PP) and Project Monitoring and Control (PMC) Process Groups and how PP and PMC are needed on either side of, and to bolster, the Execution Process Group. Keeping a project on course and continually improving project quality is understood by using PP and PMC in much the same way that drivers position their hands on either side of a steering wheel, at the 10 o'clock and 2 o'clock position, ensuring the car maintains its direction at 12 o'clock, safely on the road, and on track, moving towards the driver's goal. Deming, whose thoughts underpin much of the Capability Maturity Model integrated, appears to have broadened the application of PP and PMC towards continuous quality improvements, with his Plan Do Check Act cycle.

Essentially, as 'Do' is an important activity, without undertaking planning beforehand and checking afterwards, just doing work on its own with no direction or goal can quickly become superfluous. Therefore, it is self-evident that consistent application of the PDCA cycle is required for the achievement and maintenance of CM L2 and above.

The goal of Schedule Management specifically (Knowledge Area 6 in the PMBOK matrix), is to schedule tasks so they can be completed in correct order to complete the project successfully at its agreed destination by and at the end of the project phase or lifecycle (PMBOK process #4.7, Annexure S). This journey should ensure that project value, created during and by the end of the journey, via quality products in scope, is delivered on time and within budget, understood as managing the triple constraints. Therefore, as CM L2 is "basic project management" (Monitoring, 2010:3) at its core, which in essence is the challenge of implementing Project Planning (PP) and Project Monitoring and Control (PMC), it is not easy to achieve or maintain this state at CM L2.

The challenge for the project manager (or self-organizing agile team) concerning the tasking and management of work at CM L2 is therefore to:

- Plan (develop a schedule by entering project tasks for a willing and empowered resource).
- Do (the resource is able and strives to complete the agreed task on time, within budget and quality amongst a host of other constraints).

- Check (actual task progress against the Plan), and, if necessary.
- Act on Plan (task) variances.

Due to the inherent complexity in monitoring and controlling tasks in Execution at CM L2, which requires PP and PMC, from a Plan Do Check Act perspective, the power of a PMIS emplacement can greatly assist to minimize complexity. The same can be said of the complexity of managing financial transactions without the assistance of a Financial Management System.

Deming's Plan Do Check Act Cycle also has implications for game theory. Success in the game depends on implementing formulated plans, be they high-level strategic plans or tactical plans. The PMIS emplacement can empower a team by providing rapid feedback on whether tactics employed are successful in implementation when these are seen against the overall game plan. If not, feedback, as an early warning, may indicate that a change in direction or tactics is required. The problem is, without a PMIS emplacement highlighting that project tasks are not being delivered against the team's game plan, the ability of the project team to be responsive and successful could be constrained.

Section 1.7 shows that the developed schedule utilized for Earned Value is an essential tool to manage project health. Earned Value Management could assist the executive in avoiding the chaos mentioned in the Standish Chaos Reports (2014).

1.7 **The developed schedule for Earned Value**

According to Fleming & Koppelman (2010), Earned Value Management (EVM) should be of importance to any organisation if it is concerned about improving its project management processes to provide value to the business. Baber, Thaheem & Ayub (2017:3) note that EVM does not specifically address important aspects of health and safety, stakeholder satisfaction, and quality, their model aims to introduce various key performance indicators into the risk performance index (RPI). The researcher has factored in the ability to tightly manage risk and EVM metrics in the PMIS, to act on them immediately via his research instrument in Chapter 4.

From a game theory perspective, if statistics are produced by a PMIS emplacement, this can be used by a project team to modify behaviour to improve performance for success and to avoid failure. Without the capability to focus on managing earned value at task level using the developed schedule, a project team could be hard-pressed to keep their project healthy, delivering value and firmly on track.

Section 1.8 touches on the fact that, of the many project types, software project management could be regarded as one of the most difficult to run safely and successfully.

1.8 Safely traverse the Tar Pit

Software engineering principles lie at the heart of software projects. Frederick Brooks (1987:2) stated that software design, development, and the deployment of technology are inherently complex due to the "essential nature" of the medium. Project processes required to build a bridge, or a skyscraper will differ from those used to build software. Meedeniya, Rabashinghe & Perera. (2019:107) identify that there are limitations in the ability to manage software artifacts, especially from a change, continuous integration (CI) and continuous deployment (CD) perspective with specific regard to high traceability cost and efforts in a DevOps environment due to the frequent CI tasks. They note the urgent need of having a generalized traceability solution to cope with the maximum types of artifacts, with a minimum cost at an optimum level of automation.

According to Brooks (1995:18) 3 decades ago, building complex software solutions were regarded as being one of the most challenging undertakings. This research adopts Brooks's comparison of software to a 'tar pit' (Key Definitions) where the development of programming systems products can often become mired in ever-increasing difficulties. Based on the Chaos Reports, is appears that Brooks's quote about large-system programming mired in such a tar pit, it is a situation where many great and powerful beasts have thrashed violently and then expired. Twenty years later from the OPSLA Conference in 2007 and the paper "No Silver Bullet - Reloaded" Fraser D. Steven, Brooks P. Frederick, Fowler Martin, *et al.*, (2007) contend that the danger remains and software engineers should ignore it at their peril. Based on the discussion on Software Engineering in Chapter 2, this is a very real threat we continue to face.

Section 1.9 introduces the Research Questions that are addressed in this dissertation.

1.9 Research questions

Based on the introduction, various research questions have been raised that deal with software project management complexity and the challenges faced when project teams gear up to build valuable software.

- What is the status quo with respect to Software Project Management in Agile? (Research Question (RQ) 1)
- To what extent can Agile be successful below Capability Maturity Level 2? (RQ2)
- To what extent may the use of a Project Management Information System facilitate Software Project Management at Capability Maturity Level 2, thereby enhancing the ICT value proposition for software projects? (RQ3)

1.10 Research objectives

The objective of the research is to develop and test a Project Management Information System Capability Maturity Improvement Framework to assist with the implementation and stable running of software projects at Level 2 and above. With reference to Section 1.9, the following are sub-objectives.

- Align the Project Management Information System (PMIS) Capability Maturity (CM) Improvement Framework with respect to the current status quo regarding Software Project Management in agile (Research Objective: RO1).
- Determine whether a PMIS CM Improvement Framework can be used below Capability Maturity Level 2 (RO2).
- Determine the position at which the PMIS should facilitate Software Project Management at Capability Maturity Levels above 2, thereby enhancing the ICT value proposition for software projects (RO3).
- Propose a new diagram to be embedded into the PMBOK Process Groups and Knowledge Areas dashboard (RO4).

1.11 Layout of the dissertation

The layout is as follows:

- Chapter 2 contains a literature review.
- Chapter 3 deals with the Research Design and Methodology.

- Chapter 4 presents the contribution of this work – the PMIS CM Improvement Framework.
- Chapter 5 describes a case study and validates the PMIS CM Improvement Framework presented in Chapter 4 against the case.
- Chapter 6 contains conclusions and future work.
- References and several other annexures follow Chapter 6, concluding the work.

1.12 Summary

Chapter 1 unpacked the problem identified by the Standish Chaos Reports (2014), where it appears that projects across the globe are often not managed successfully for earned value. A general overview placed software project management in context while proposing that a focus on process management using the PMIS emplacement may alleviate many of the challenges faced. **Chapter 1** also explained the problems and resultant inability to unlock capability maturity requirements needed to move out of CM L1 behaviour.

1.13 Conclusion

In this chapter, the statement is made that if project processes for traditional project management or agile project management methods are not consistently applied and sustained at CM L2, it could cause some of the many problems identified in the Standish Chaos Reports (2014). Expanding upon this idea, the Project Management Information System (PMIS) emplacement could be considered an essential tool to manage project complexity. Efficient and effective use of a well-running PMIS can empower project teams to rise out of CM L1 behaviour to begin focusing on process excellence at CM L2. Even if the application of traditional processes appears to be understood by many as the core reason for project management in chaos, and agile with less attention to process application rigour appears the logical solution, this chapter alludes to the fact that the PMIS used to run any project framework well at CM L2+ is another option to consider. The PMIS CM Improvement Framework in Chapter 4 is presented by the researcher as a possible approach to achieve a PMIS CM improved project performance improvement solution.

The next chapter contains a literature review for this research. A diagram (Figure A.1 in Annexure A) has been constructed by the researcher to synthesize several dimensions that are envisaged to exist within this research. Figure A.1 is therefore presented as a high-level model or overview which includes a research statement for Chapter 2.

CHAPTER 2:
LITERATURE REVIEW

2.1 **Overview**

Chapter 1 unpacked the problems associated with an inability to unlock potential and value, primarily due to Capability Maturity (CM) Level 1 or “initial” behaviour. The Project Management Information System (PMIS) emplacement is presented to apply, automate, and manage project and software engineering and other processes better to meet the software project management challenges to operate at CM L2 and above.

Chapter 2 is a literature review that will be used to expand the discussion started in Chapter 1 for the theoretical case study in Chapter 5. The introduction below explains the theoretical dimensions that underpin the work conducted in this chapter.

Figure 2.1 below is a research dimensions model to give a visual overview of the research approach. A larger and easier to read version of Figure 2.1 is found in Annexure A.

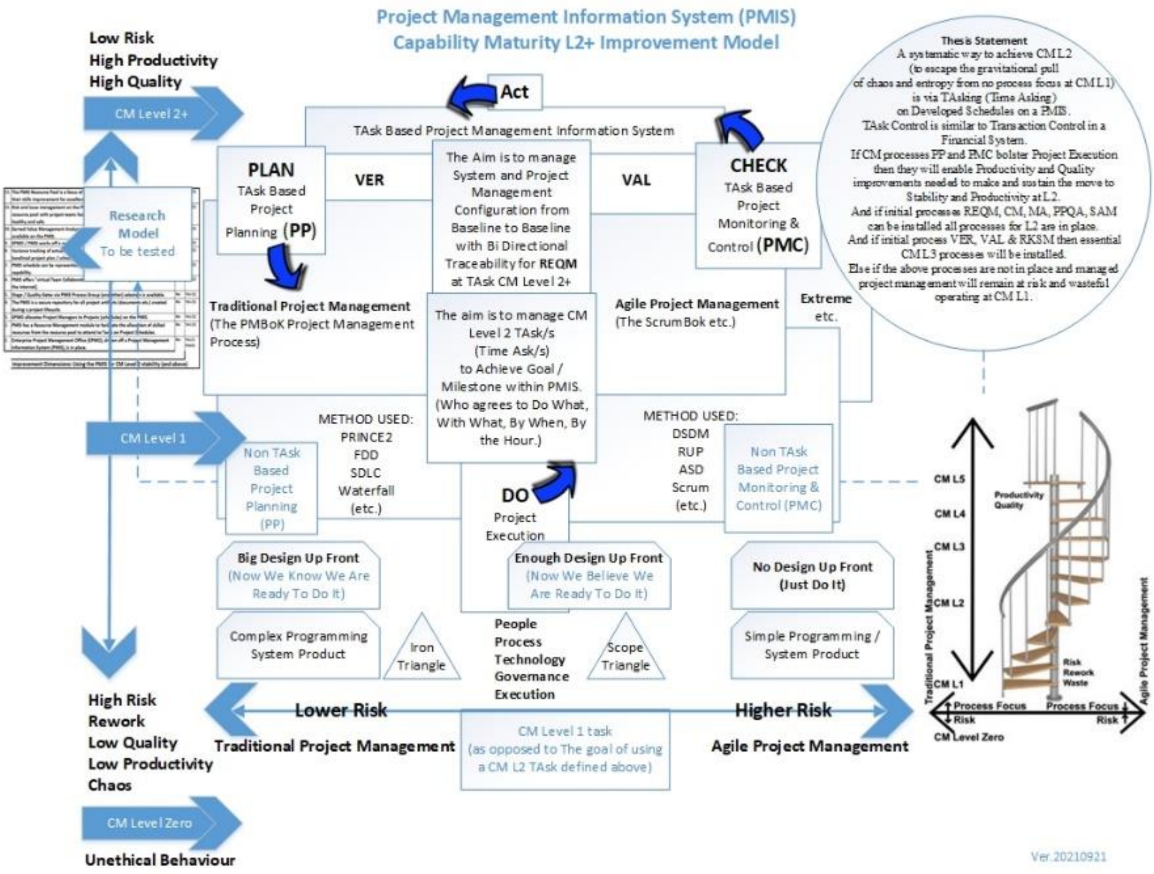


Figure 2.1: Research Dimensions Model

2.2 Introduction

As this work is based on a theoretical case study in Chapter 5, to reduce complexity, Figure 2.1 and Annexure A presents a Research Dimensions Model, compiled by the researcher, to guide the research approach and direction. The model will also be used to assist with the structure of this chapter. The model, a high-level overview, is constructed around 3 dimensions: vertical, horizontal and a depth or spiralling upwards and downward dimension. The horizontal, vertical axis and depth axis are like the X, Y and Z axes in the Cartesian system.

Vertical Y-Axis:

The vertical axis is associated with the Capability Maturity Levels found in the Capability Maturity model integrated from Carnegie Mellon University. Low risk, high productivity, and high quality are found at the top of this axis, while high risk, low productivity, rework, and waste is found at the bottom.

Horizontal X-Axis:

A horizontal axis is also available and focuses on Traditional Project Management on the left and Agile Project Management on the right. The researcher's understanding is that lower risk could be found further to the left, with higher risk existing further to the right of the X-axis. The assumption is that the horizontal and vertical axis or dimensions can affect each other, and risk will be more prevalent in "initial" behaviour at CM L1 than behaviour focused on process application at CM L2 and above.

Spiralling Up and Down Z-Axis:

A third axis (z) of depth, height, or spiralling up or down dimension is based on the application of the Plan Do Check Act (PDCA) Cycle to create improvement in a system, as shown by Figure V.1 in Annexure A. The assumption is that PDCA applied will allow the achievement of sustainable gains. In this regard, the idea is of a bird that can soar or spiral upwards and then maintain that height. An assertion is made that a PMIS emplacement could be an essential enabler to achieve successive stable improvements in project and software engineering or other process application, such as CMMi, ITIL, COBIT process application as an improvement trend to CM 2 or higher.

Based on the above approach, Chapter 2 will start by placing Software Engineering complexity in perspective. We will then move to the Capability Maturity model focusing on specific process improvements sought in moving out of CM L1, the vertical or Y-axis. Finally, the second part of Chapter 2 will look more closely at Traditional and Agile project processes as a continuum along the horizontal or X-axis.

To compensate for limited research material available on Capability Maturity Levels, especially where these can be applied to a PMIS, the researcher intends to extend his analysis into several adjacent areas of study, methods and frameworks using the manuals or other available material to support all assertions made. Ultimately, this research is aimed at illustrating that any project method can benefit from process application improvement focus on a PMIS emplacement set up and operating correctly, as is recommended by the PMIS CM Implementation Framework in Chapter 4**.**

The researcher will now discuss the literature relating to the research that has been conducted in this focus area. Section 2.3 below begins this process with a look at Software Engineering and why it is regarded as complex.

2.3 **Software Engineering Complexity**

Kotter (1996:13) predicted that change would accelerate rapidly with companies needing to reduce costs, increase productivity, improve the quality of products and services, and locate new opportunities to provide value in order to ensure their survival and growth. Kotter believed the rate of change would not slow down any time soon. Twenty years later, Haverkort & Zimmerman (2017:8) found that Kotter's observations were correct.

Clearly businesses are still grappling to absorb the speed of change and rate of innovation in a worldwide information and communication technology (ICT) revolution, the challenge remains, how exactly to integrate or coordinate largely internet-based information and communication technology (ICT) offerings into their service value chains. According to Haverkort & Zimmermann, (2017) the ICT change drivers appear to be focused on the industrial Internet of things and how to best use these for competitive advantage in the smart industry, or as some now refer to it: Industry 4.0.

The researcher notes that towards the end of 2020, with the international COVID-19 pandemic, business reliance on the ability to operate continuously, effectively, and profitably across the internet, with staff working from home, has become crucially important. Now, more than ever, the internet is being relied upon as an ICT backbone to ensure efficient, adaptive production and supply-chain processes; businesses with the ability to capitalise on the new generation of always-connected products must have a competitive advantage. To remain viable and competitive, businesses subjected to this paradigm shift must understand how to support innovation to deliver increasingly valuable software and ICT systems. Larger enterprises especially will struggle to innovate fast enough to keep up with the increasingly digital world. They will need to learn how to adapt quickly to changing economic conditions and technology or become extinct.

However, speed or agility may not always be the best approach. According to technologist Jeff Bizos in 2020, his preferred approach is summed up in the phrase "Gradatim Ferociter", which is Latin for step by step ferociously. Applying this motto to the challenges of spaceflight, Bizos says that one cannot cut corners when building a flying vehicle. Bizos says that cutting corners is an illusion, that it will only appear faster (Boyle, 2016). For this researcher, focused on the Project Management Information System, to facilitate a stepwise approach to the application of project management and other processes at CM Level 2+, this advice appears to be sound.

The next section deals with the speed that technology is changing which is encapsulated in the concept of the VUCA world.

The VUCA world

Johansen (2017:5) coined the term VUCA. According to Johansen, the ultimate dilemma and requirement for leaders in the future will be the ability to flip, or change, away from the frightening VUCA (volatility, uncertainty, complexity, and ambiguity) to a hopeful VUCA (vision, understanding, clarity and agility). In Chapter 1, the discussion of the importance of project planning and project monitoring and control to support project execution illustrates what could be a sound approach if the chaos described in the frightening VUCA definition, also touched upon in the Standish Chaos reports, is to be avoided.

The next section explains that software complexity, no matter the many ideas to tame the medium, to enable software engineers to be more productive when building solutions, remains complex.

Software is a complex medium

Frederick Brooks, in “The Mythical Man-Month: Essays on Software Engineering” (1975), mentions the illusive (essential) nature of software. Hughes & Cotterell (2009) make the clear distinction between normal projects and software projects. Unlike structural engineering, where requirements and material components are elucidated upfront, with software it will always be difficult to pin down exact requirements or best ways to standardise project methods and software engineering approaches.

The benefit of running software projects well means that effort expended is converted quickly into value for the organisation. However, due to the nature of the medium, software projects can just as rapidly become bogged down with difficulties, often mired to such an extent they cease to be viable.

Software design, development, and deployment of technology are complex undertakings, and according to Brooks (1987, 1995:5), these can be regarded as “the most difficult of undertakings.” It is noted that in Brooks’s Anniversary Edition, some 20 years after the original publication of the Essays on Software Engineering, he again emphasised software complexity. Additionally, Brooks mentioned how software development projects, to create valuable “programming systems products”, often, if not carefully managed, could become mired in ever-increasing difficulties he referred to as the “tar pit of failed software projects”.

To this end, the metaphor of the “tar pit” is borrowed from Brooks (1987, 1995:5), and is used to illustrate what happens to software projects whose forward momentum has slowed down and become stuck.

Figure 2.1 explains how Brooks understood the ‘Programming System Product’ in terms of complexity and effort.

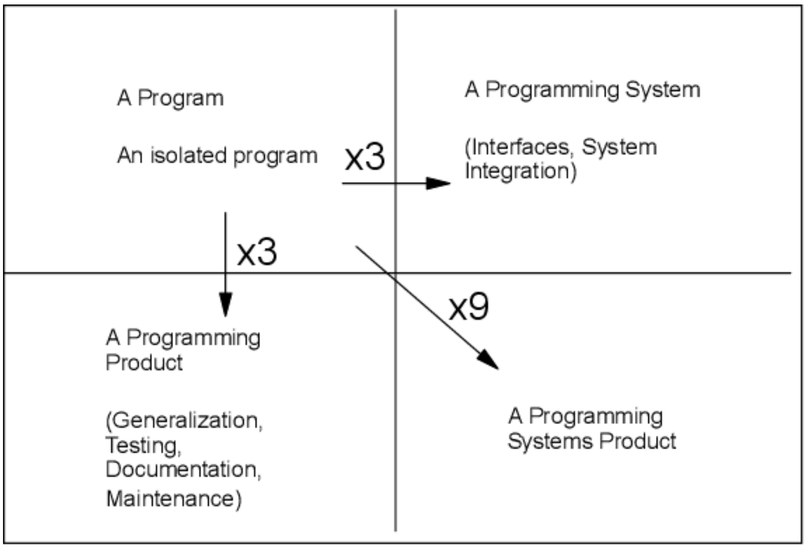


Figure 2.2: Complexity of Programming Systems

Source: Brooks (1995); Hailpern & Padmanabhan (2002)

The figure explains that small software programs or modules take effort to build, which Brooks estimated at 1-fold complexity, as soon as these need to be combined into programming products, the complexity increases 3-fold due to testing, maintenance, interfaces. However, when programming system and programming products are combined into programming systems products, effort based on the increased complexity could increase 9-fold.

Brooks (1987, 1995:5) description of ancient lumbering beasts of old ensnared, is as true today as it was four decades ago: "Yet the fiercer the struggle, the more entangling the tar, and no beast is so strong or so skilful but that he ultimately sinks." Brooks (1987, 1995:7) also comments that large-system programming has been a tar pit over the previous decade in which many strong beasts have fought ferociously to escape (1995:5). Johnson & Ekstedt (2016) reiterated that the main problem is the communicative difficulty between the architecture of human cognition and the architecture of computing systems. Brooks (1995:184) summarizes software complexity into four distinct qualities that make it seem like tar. These qualities are complexity, conformity, changeability, and invisibility.

Hong Zu (2005) classifies design errors into four types, namely Incorrectness, Inconsistency, Ambiguity and Inferiority. Zu explains that inefficiency and inflexibility in software engineering often relate to poor qualities of software design. Accordingly, adding incorrectness as an obvious design error is by no means less common than other types of design errors. Often the perspective of incorrectness of the design is that the design does not meet the user requirements about its functionality and features. This indicates that a programmer may have done excellent work, but due to the complex medium of software, the error could appear in the form of misinterpretation or omission of user requirements. Inconsistency problems relate to multiple design statements that could make conflicting assumptions about the functionality of a component or the meaning of a data item, with the result that the design does not work. From an ambiguity perspective, this occurs when the design specification is interpreted in several different ways or is not clear enough. Ambiguity causes errors in the implementation of the design due to inconsistent interpretations made in the implementation process. Inferiority of design means the design does not address quality requirements adequately. Consequently, the designed software is of poor quality with respect to users' quality requirements.

Schubert, Tsitsipas & Jeffery (2018) found that a new basis of thinking is required which is removed from traditional perceptions involving systems that must run distributed, in parallel, on heterogeneous environments, share distributed data. Hughes & Cotterell (2009) indicate that, unlike traditional engineers, software developers have to conform to the requirements of human clients who may not appreciate the complexity of the medium. They comment that it is not just individuals who can be inconsistent, but also organisations, because of lapses in collective memory, internal communication, or effective decision-making. Consequently, developers often must cater for "remarkable organisational stupidity". Dooley (2017:305), concluding his book on software engineering, states that software development is difficult. Dooley adds that not everyone can do it, and of those who can, few do it extremely well all the time.

Zaman, Jabber, Nawaz & Abbas (2019:456) identify linkages among complexities, socials skills, political skills and project performance. The authors recognise the mobility factor of human resources as an issue that requires attention. The training and maintaining of skilled software engineers will require greater emphasis on workforce planning, recruitment and boarding based on project complexity demands, and this will require socio-political skills. Alawad, Panta, Zibran, *et al.*, (2018:34) established empirical confirmation of the existing wisdom; that code readability and software complexity had a significant impact on software quality when they proved that readability and complexity were found to be negatively correlated. Where programmers are concerned, the goal must be to simplify their code so that others can easily read it. By doing so, they reduce code complexity and improve its quality and robustness.

If software products are regarded as inferior, this could involve problems pertaining to inefficiency and inflexibility. Inefficiency relates to the software solution being slow or difficult to use, inflexibility causes the designed software to be difficult to change. This viewpoint is supported by Parnas & Weiss (1987), who classify software complexity into inconsistency, ambiguity, inefficiency, and inflexibility. Parnas & Weiss (1987) reportedly explained at the conference that software development was a conceptually tough business and that magical solutions[5] were not just around the corner. Parnas & Weiss (1987) also reiterated at the conference that it was time for the practitioner to examine evolutionary improvements rather than to wait, or hope, for revolutionary ones. Additionally, he said that this was a discouraging picture if the thinking was that breakthroughs were near at hand. However, Parnas & Weiss (1987) said, for those realists, older and with more experience, this was a breath of fresh air as now the focus could be to get on with the incremental improvements in software productivity that is possible, rather than waiting for the breakthroughs that are not likely to ever come.

Thirty years later, the work by Rahmati (2016) ratifies that the problems of software design complexity still exist, stating that the problem of being unable to correctly quantify the cost of software development has not changed. His total cost evaluation is based on applying his complexity function points to the application of known and unknown parameters that are mainly estimated based on designers' experiences. To this end, the idea of a silver bullet (Key Definitions), or easy solution, that will solve the software complexity problem could be regarded as a quest for the holy grail, ultimately doomed to failure.

Even the statement that there is no silver bullet to slay the software werewolf was reiterated by Brooks (1995) himself and in the paper "No Silver Bullet – Reloaded"

---

[5] Magical solutions are solutions that are fantastical and not based on logic, truth or reality. In this regard, the term "silver bullet" is also used in software engineering to indicate that a magical solution pretending to be a silver bullet or quick fix will be needed to slay the magically protected werewolf (problem), which is not realistic.

Fraser D. Steven, Brooks P. Frederick, Fowler Martin, *et al.*, (2007), many on the panel held firm to Brooks's original assertion. At this Conference, Doctor Parnas framed Brooks's paper "as a challenge for us to continually try to address productivity". Parnas said that people have a very natural tendency to look for easy answers to hard questions and added that "designing software is hard and it will always be hard". Recently, Ozkaya (2019) posed the question, asking whether DevOps and automation were the new silver bullets. Some companies had successfully adopted effective tooling and continuous delivery infrastructures and could describe gains in reducing rework costs (Dora, 2017), and others reported on overheads and failures. According to Ozkaya (2019:5), failure stories are often similar and include "not understanding the context, dropping key activities between communication barriers, failing to collaborate, and having problems in some activities that could not be automated", which is where many of the processes can fail in their adoption. Ozkaya (2019:7) also concluded her paper, stating that "as Brooks has told us in 1975, there is no silver bullet. Neither DevOps nor automation is one, either".

As more can be said on this subject, the statement from Parnas at the 'No Silver Bullet Retrospective' talk at the OOPSLA Conference in 2007 (2008) was particularly apt when he said that "there were things called lead bullets, plain old ordinary bullets, that are disciplined, hardworking, and require a lot of training to use them. And we do not do it", and he questioned why we did not use these lead bullets.

The researcher maintains these lead bullets could be Capability Maturity processes, understood and correctly applied, fired repeatedly from the stable base of the PMIS emplacement. Owing to the software engineering and project complexities involved in producing the next versions of valuable software, a PMIS emplacement could improve productivity and quality and reduce risk and waste, supporting the software development team apply CM Level 2+ processes to ensure that they can safely and systematically traverse the tar pit.

Section 2.4 below will start to look at the Capability Maturity model from the point of view of the dimensions that was explained in Figure 2.1 at the beginning of this chapter and which is also found in Annexure A.

2.4 CM Levels as a Vertical Dimension:

Annexure A explains how Capability Maturity (CM) levels can be viewed as a vertical dimension. Annexure A can also be considered as a staircase where one starts at the bottom and moves upwards step by step, from CM L1 to CM L5, to reach higher levels of quality and productivity through the correct use of process application. The move upwards from CM L1 to CM L2 entails the installation, application, and adherence of project management processes. The vertical CM dimension discussed in this section is contrasted to the horizontal dimension shown in Annexure A, to be discussed later.

The researcher, when working with Traditional and Agile projects, has found that some proponents of agile assume that agility precludes the next process. In the Scrum agile method in Annexure O, Table O.1 shows that Scrum has nineteen processes that should be followed systematically and in order from the first process through to the 19th process for sprint success. These processes operate similarly to the processes that are used in Traditional Project Management, but the detail that is expected in planning is far less, and the timeframe for an iteration is much shorter. The assumption is that by reaching the final step following the Scrum process, the quality and productivity goals of the Scrum sprint, as from a production line, would have been achieved, and working software will be ready to be released and shipped.

Hirsch (2012), discussing the Project Management Institute's Pulse of the Profession, found a positive relationship between higher maturity levels and on-time and on-budget delivery of projects. Rahmani, Sami & Khalili (2016) demonstrated many similarities between the CMMI-DEV, ISO 9001, and the PMBOK and that these could be harmonized. Their study concluded that a unified model (CIP-UQIM) could be beneficial to resolve or reduce SPI issues, especially in the case of software solutions for SMEs. Selleri Silva, Soares, Peres, *et al.*, (2015) found that agile methodologies could be used by companies to reduce the effort needed to attain levels 2 and 3 of CMMI. However, agile methodologies alone were not sufficient to obtain a rating at a given level or to maintain it.

As mentioned previously, the researcher decided to focus on the Capability Maturity Model integrated (CMMi), which comes from the Software Engineering Institute (SEI) at Carnegie Mellon University. Work on Capability Maturity modelling originated at Carnegie Mellon to assist the U.S. Defence Department to improve software development processes. The CMMi caters for all areas of the organisation and does not focus solely, as was the case previously, on software development. More information on the CMMi and the Carnegie Mellon University Software Engineering Institute (SEI) is available in Annexure M.

The characteristics of the maturity levels in Figure 2.2 show the high-level goals for each of the five capability maturity levels. The two variations of the Capability Maturity Model integrated, found in Annexures C and D, offer essentially the same information as the figure below. However, each of the Annexures has more detail. Overall, the identification, application, and correct use of all the CMMi processes should assist the organisation to move towards higher levels of productivity and quality, endeavouring to reduce risk, rework, and waste.

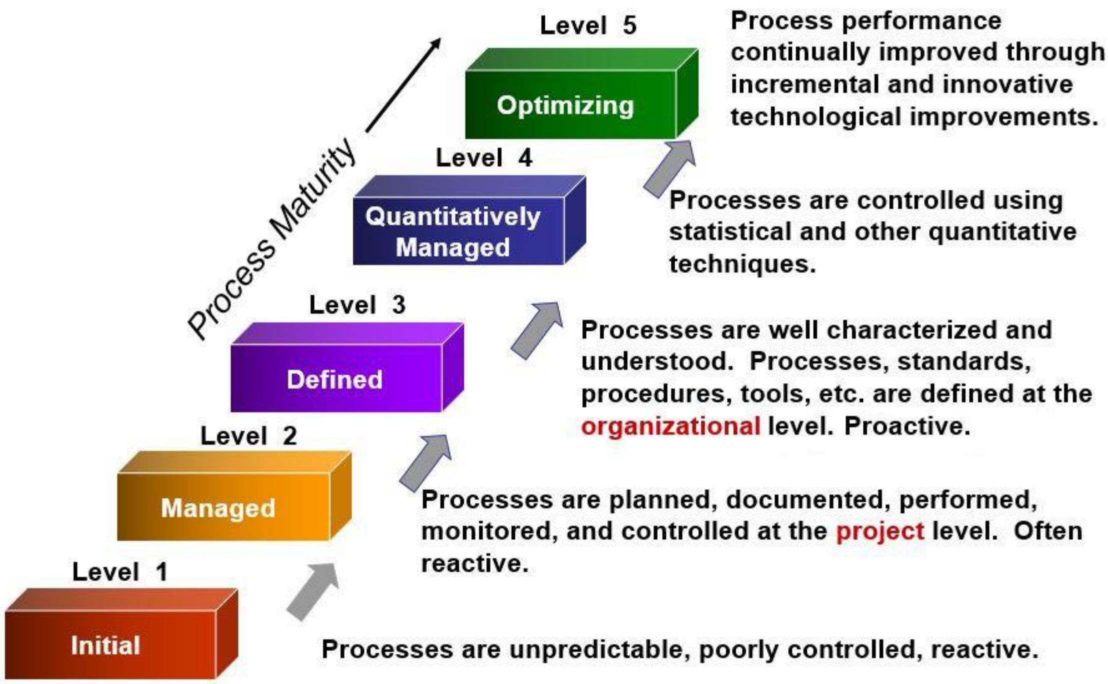


Figure 2.3: Characteristics of the Maturity Levels

Source: S. Lanjewar and further synthesized by the researcher

The figure illustrates that activity at Level 1 is Initial, which means that processes are unpredictable, poorly controlled, and reactive, or that no process is being followed at all. The assumption from the researcher in this dissertation is that processes at this level can be considered as non-existent unless effort is being expended knowingly to move away from CM L1 to implement a specific project process at CM L2. Level 2 is characterised by the installation of project management processes. Project management processes can be utilised in any project methodology, as long as it has processes and if possible, inputs, tools and techniques, and outputs documented for each process. In addition, at this level processes will be planned, documented, performed, monitored, and controlled at the project level. Unfortunately, for this level, the behaviour could often be uninformed and reactive.

Level 3 processes, on the other hand, are well-characterised, documented by the organisation, and understood. Processes, standards, procedures, and tools are proactive and clearly defined at the organisational level. This means that if PRINCE2 project methodology is the organisation's method of choice, then these processes are understood and used corredtly. If the level 3 processes listed in Annexure D are analysed, it will be evident that these processes are typically focused on advanced project management as well as software engineering activities and other processes.

Level 4 focuses on quantitative management, and processes are controlled using statistical and other quantitative techniques.

Level 5 focuses on continuously improving process performance.

Table 2.1 lists the seven processes at CM L2 from the Capability Maturity Model integrated (CMMi) for Software Development constellation. These processes are those that are needed to run projects successfully. With a total of twenty-two processes are found in the CMMi for Software Development constellation, the list in Table 2.1 shows the processes that need focus if the goal is to move up from CM L1 to CM L2.

Table 2.1: Capability Maturity level 2: Managed (First Steps)

PA CODE	CAPABILITY MATURITY L2 PROCESS AREAS
CM	– Configuration Management
MA	– Measurement and Analysis
PMC	**– Project Monitoring and Control**
PP	**– Project Planning**
PPQA	– Process and Product quality Assuranc
REQM	– Requirements Management
SAM	– Supplier Agreement Management

Source: Synthesized by the researcher

The table shows the Process Area Code (PA CODE) and Capability Maturity L2 Process Areas associated with each code. Project Planning (PP) and Project Monitoring and Control (PMC) are emboldened in the table to explain the essential importance of these two core processes in preparing the solid base for projects in Execution at CM L2, as discussed in Section 1.4 (Chapter 1).

In addition to PP and PMC in Table 2.1, it stands to reason that a project requirement is also needed in order to run any project. To satisfy this, the CM L2 Requirements Management process (REQM) is also important. The Process Area of CM and MA focus on configuration and measurement, which are understood to be essential for CM L2 compliance and as a way to improve from a previous baseline. Process and Product Quality Assurance (PPQA) focuses on tying together the project processes required to produce value through the delivery of quality products, essentially providing staff and management with objective insight into processes and associated work products. Supplier Agreement Management (SAM) is self-explanatory, and the importance of this process area is validated by having its own Knowledge Area called Procurements in the PMBOK Knowledge Area and Process Group matrix, as shown in Annexure S.

If the CM Level 2 processes PP and PMC above are analysed, it is clear that these closely resemble the core PMBOK 6 processes found in the Planning and Monitoring and Control Process Groups in Annexure S. To this end, Annexures C and D (the CMMi model) and Annexure S (the PMBOK Process Groups and Knowledge Areas matrix) contain the same project processes at CM L2, thus validating the findings of (Rahmani, Sami & Khalili, *et al.*, 2016) mentioned previously.

The PMBOK 6 matrix in Annexure S contains processes within the Project Planning and Project Monitoring and Control Process Groups, each process broken up into Inputs, Tools and Techniques, and Outputs, and each aligned to a specific Knowledge Area. The fact that Capability Maturity Model integrated (CMMi) processes (Annexure C) leverage off the PMBOK 6 project management processes at CM L2 (Annexure S) must be invaluable to an organisation that needs to rapidly install the Capability Maturity Model integrated for Development V1.3 (Annexure D) for Level 2 compliance.

Figure 2.3 shows the latest version of the CMMi, V3.1. The figure is subdivided by the three constellations of Development, Services, and Acquisition and illustrates the 22 common processes that lie between each of the constellations.

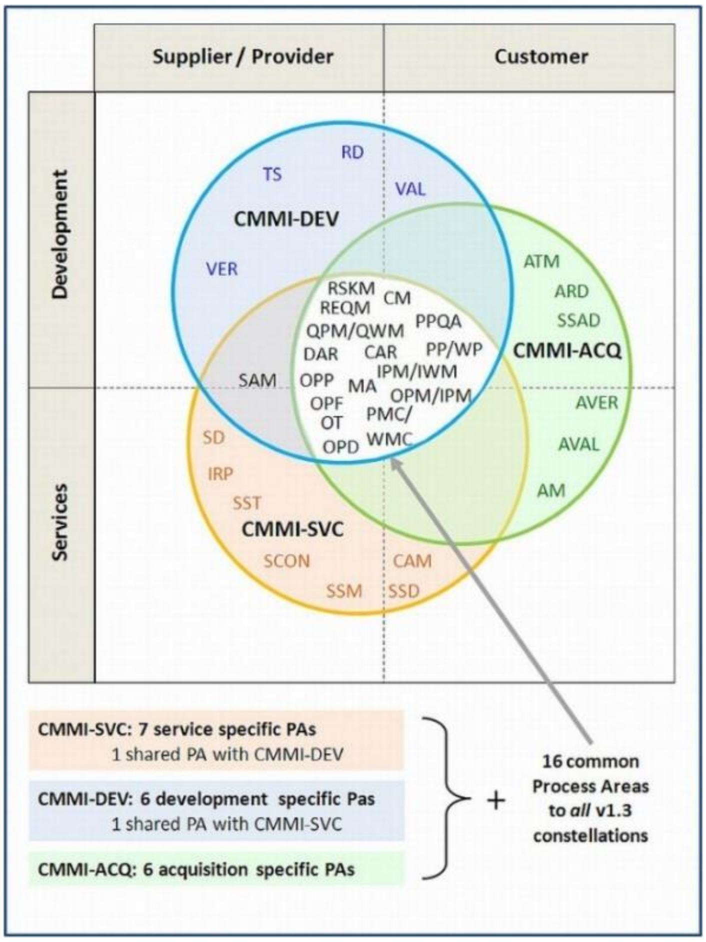


Figure 2.4: CMMi Constellations & Processes

Source: Tarnovski (2014)

The figure shows the three constellations of the CMMi, which are the CMMi for Development (Dev) constellation, CMMi for Services (SVC) constellation and CMMi for Acquisition (ACQ) constellation. The Process Area Codes list 16 common processes that are shared by the three constellations at its centre and the other processes that are unique to each constellation individually.

It is worth noting that the People Capability Maturity Model (PCMM) in Annexure H, the Human Resource component of the CMMi, does not yet belong to a constellation. The researcher observes that the PCMM will straddle the DEV, SVC and ACQ, guiding the implementation and development of professional human resources required by all the Constellations. From a PCMM perspective, Level 1 is inconsistent human resource management, Level 2 people-focused human resource management, Level 3 competency-based human resource management, Level 4 capability-based human resource management, and Level 5 is focused on change management so that human resources working together can move together to implement change. In a mature project-focused environment, capable staff would be required to understand and correctly apply project management processes to support projects and programs within the organisation that will make it more competitive, and able to consistently deliver value to its clients.

The PMBOK 6 identifies several organisational structures that range from Projectized, Matrix or Functional (PMI, 2017:47). According to the PMBOK 6 (2017:82), projects run in functional structures may find general resistance to collaboration across its organisation. Functional structures may also insist that staff maintain focus on business activities as a priority, and at the same time, staff will be expected to participate in projects.

Table 2.2 offers a summarised list of the CMMi process.

Table 2.2: Twenty-two Process Areas of CMMi Dev 3.1

MATURITY LEVEL (ML)	ML DESCRIPTION	PROCESS AREA	PROCESS AREA ABBREVIATION
2	Repeatable	Configuration Management	CM
2	Repeatable	Measurement and Analysis	MA
2	Repeatable	Project Monitoring and Control	PMC
2	Repeatable	Project Planning	PP
2	Repeatable	Process and Product Quality Assurance	PPQA
2	Repeatable	Requirements Management	REQM
2	Repeatable	Supplier Agreement Management	SAM
3	Defined	Decision Analysis and Resolution	DAR
3	Defined	Integrated Project Management	IPM
3	Defined	Organizational Process Definition	OPD
3	Defined	Organizational Process Focus	OPF
3	Defined	Organizational Training	OT
3	Defined	Product Integration	PI
3	Defined	Requirements Development	RD
3	Defined	Risk Management	RSKM
3	Defined	Technical Solution	TS
3	Defined	Validation	VAL
3	Defined	Verification	VER
4	Quantitatively Managed	Organizational Process Performance	OPP
4	Quantitatively Managed	Quantitative Project Management	QPM
5	Optimizing	Causal Analysis and Resolution	CAR
5	Optimizing	Organizational Performance Management	OPM

Source: R.D. Levy CMMI-DEV tool

The table lists all 22 CMMi processes for Development 3.1. The Process Area Abbreviation of the table lists the short acronyms used to identify each of the twenty-two processes. The Process Area states what the process is called and which area it is used in. The Maturity Level (ML) identifies if the process is an L2, L3, L4 or L5 process. The ML Description explains if the process is Repeatable, Defined, Quantitatively Managed or in Optimising.

The researcher understands that the CMMi can be used to guide the organisation on a quality and productivity improvement journey. The use of the CMMi entails process identification, understanding, implementation and correct use as the steps from a precedence perspective, as needed if productivity and quality gains are an organisation's goals.

To assume there is an easier approach could be an indication that "silver bullet solutions[6]" are being sought, which could be an indication that the organisation is operating within an immature mindset beneath CM L2. In Section 2.3, Parnas referred to lead bullets[7], plain old ordinary bullets that are disciplined, hardworking, and need a lot of training to use. The researcher perceives these lead bullets as the CM processes we have been discussing. Parnas questions why these lead bullets are not used. A possible answer is that the ability to obtain a deeper understanding of the CM processes could be a challenge. On the other hand, if the lead bullets were understood and appreciated, and, like following a recipe, they could be applied and used at the right time, then teams of professionals could start to focus on how and when to use the processes correctly rather than wasting time looking for easy alternatives. The PMIS emplacement, installed correctly according to the PMIS CM Improvement Framework in Chapter 4, could be the way (recipe) needed to roll out the processes and be a stable base from which to fire these lead bullets to hit project targets.

Furthermore, the Dunning-Kruger effect (1999:1) states that people tend to hold overly favourable views of their abilities in many social and intellectual domains. The Dunning-Kruger effect has implications for the project management profession, as is explained in Figure 2.4. Project process stability at CM L2 for both traditional and agile

---

[6] Silver bullet solutions are solutions that sound like easier approaches but are not.

[7] Lead bullets are simple, tried-and-tested processes that can be applied to problems.

project management can be viewed as the start of a move up the slope of enlightenment, towards the goal of sustainability of ICT projects.

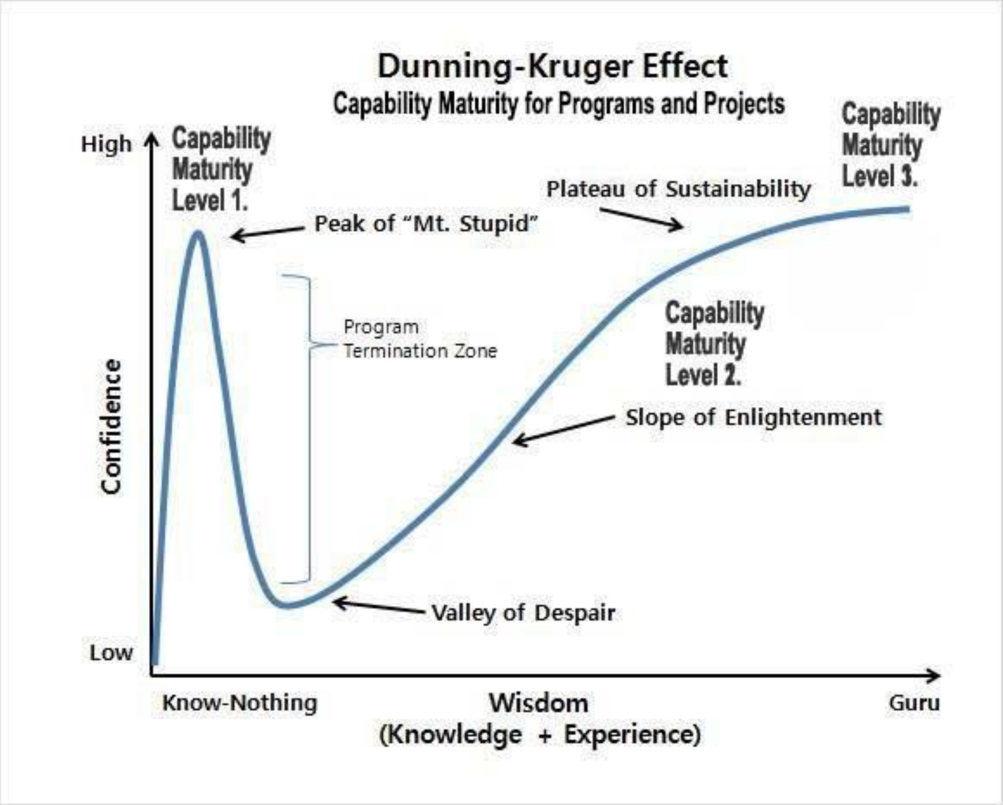


Figure 2.5: Dunning Kruger Effect on Projects and Programs

Source: Adapted from Dudley (2019)

The figure uses the Dunning-Kruger effect and combines it with the Capability Maturity levels. The figure illustrates that project activity undertaken at CM L1, which ignores the CM L2 project processes, could be misguided and may stand a good chance of introducing chaos and ultimately failure. Confidence at CM L1 may be high, but that could be due to a lack of experience. The termination zone is where projects or programs fail, and confidence also falls as reality dawns on the actual complexity involved, which was previously unappreciated. It is only when project processes are diligently applied, similarly, to following a recipe in a kitchen, that success can be achieved, allowing a sustainable move up the slope of enlightenment. CM L2, essentially a base project management process set, if applied with Wisdom (Knowledge and Experience), ought to be the starting point to facilitate project success and stakeholder satisfaction. Consequently, a complex, agile-driven project at CM L 1 runs the risk of being severely challenged, thereby compromising good ICT intentions.

2.4.1 CMMi Complexity and Installation Time

According to Staples, Niazi, Jeffery, *et al.*, 2007 the most frequent reasons given for non-adoption of CMMi are because the organisation is small, it will be too costly to install, there is no time, or they are using some other Software Process Improvement (SPI) approach so that adopting the processes will be infeasible, but not unbeneficial. Khurshid, Bannerman & Staples, 2009 stated that companies were often, due to the complexity of the CMMi, unsure of the actual benefits or gains, and/or the organisation had other more pressing priorities. Industry 4 roadmap research by Issa, Hatiboglu, Bildstein, *et al.*, 2018 showed where CMMi could be applied to digital transformation projects. The research showed how individual organisations could utilise the proposed processes in order to assess their current position and use these as a starting point to build their roadmap for the future. However, they recognised that their work was new and at an early stage of a relatively new topic. Work by Pane & Sarno, 2015 shows the benefits of using the Capability Maturity Model integration (CMMi) as an integrated software process improvement standard and approach to be applied to optimising Object Orientated Applications Development. They concluded that optimising object-oriented analysis and design results by using the CMMi is a relatively new research area in software process improvement, which could be used to reduce organisation cost of failure in the next stages of the software development process.

Sun & Liu, 2010 found that some common limitations of existing SPI models and standards are the specifications of what to do but not how to do it. The fact that CM L2 is all about project management processes indicates that stability at CM L2 must be all about project management capability and the ability to run projects successfully to assist with prioritisation, focus, and project delivery. The PMBOK matrix (Annexure S), a summary of the latest project processes from the PMBOK guide by the PMI, clearly shows how to do it and also gives the steps that are needed in a very specific order, including inputs, transformations, and outputs for each step. In addition, these steps are spread out across the Project Management Knowledge Areas.

Diaz-Ley, Garcia & Piattini, (2010) state that an important reason for the failure of CMMi, and the measurement program needed to facilitate its successful implementation, pertains to the maturity of companies with regard to measurement, which has not been considered during the requirements definition phase. In CMMi, the process MA (measurement analysis) focuses specifically on measurements of projects and other processes to ensure they operate optimally. This could be seen to have special relevance to software engineering and the cost of building quality software. In this regard, if the measurement was based on professional project management outcomes, with a PMIS emplacement focused specifically on Earned Value Management and Good Governance, these target measurements can easily be compiled for each project within the Initiation and Planning processes, as found in Annexure S, and then measured for accuracy and the ability of the plan to hit its delivery target in the Monitoring and Control processes.

As estimating is always difficult, the researcher does not see a problem in this regard because any estimate or budget based on limited information must be better than using none. Using the PMIS emplacement and being able to adjust actuals in flight means that negative variances against budgets or estimates can be quickly analysed and updated as appropriate remediation steps are put into place. This activity will align the project to continue its journey to deliver by readjusting its plan with new information learned since the project started.

Gonçalves (2012) agrees that the creation and correct use of project management remains a challenge in many software organisations. In striving to address these needs, best practice models, such as the Capability Maturity Model or the PMBOK, are being consistently improved to better assist organisations interested in quality and accurate project management. Sauter (2013) states that when comparing the PMBOK and ISO21500, the PMBOK guide goes a long way towards the unification of the new standards. In addition, the PMBOK, with its track record and established body of knowledge, is like “a planet with greater mass, pulling ISO standards into its gravitational orbit.” Annexure B gives an overview of how many of the standards and frameworks interrelate. It also shows where the PMBOK and CMMI, and other methodologies fit into an overall IT landscape.

2.4.2 CMMi Installation Time

According to Entinex (2014), a move from CMMi L1 to L2 is estimated to take over two and a half years. When asked for more detail on this, Carnegie Mellon University used this analogy: Say you are carrying around about 40lbs (18.18kg) of excess body fat. How long will it take you to lose the fat? A year? Two? Six months? Can one person do in six months what another person needs two years to do? We all know the answer to these questions; it depends!

Ultimately, if much of CM L2 detail is found in the PMBOK, summarized by Annexure S, then project managers who understand these processes and how they work, their input tools and techniques, outputs, etc., could help to radically reduce the CMMi installation time.

2.4.3 Sustaining CMMi processes at CM L2

As it may not be easy to install the processes that are required for CM L2, if a stable operation is achieved at L2, this state will have to be maintained through process focus based on consistent measurement of process productivity and project productivity. Unfortunately, it is possible to drop back down to L1 if process application measured as quality outputs is allowed to deteriorate.

2.4.4 CMMi not working. Time for another silver bullet

CM or SPI models in general, and especially the Carnegie Mellon University CMMi, is all about the systematic improvement of Capability and Maturity to run essential processes at a particular CM target level with competence, stability, and sustainability.

If CM models are regarded as too difficult, and the time arrives to move to some other approach, it could be a clear indication that Capability and Maturity within the organisation could be low and could be operating at CM L1. The act of rejecting CM models and grasping for another silver bullet must surely illustrate a lack of understanding and immaturity, for the only way out of the chaos referred to in the Standish Chaos Reports (2014) must surely be competency at process application.

The researcher is told regularly that Traditional Project Management has failed and Agile is a better way to proceed due to an inability to plan in software engineering. This may well be the case. However, even Agile has processes that, if understood and competently applied, must realize higher levels of quality and productivity than rejecting the CM approach at Level 2 or deciding not to follow processes at all. Competency should always be backed up with measurement statistics based on comparisons between a prior and new state as the primary reliance. The CMMi processes MA and QPM in Table 2.2 are used to achieve this knowledge. To this end, the acronym from the IT Infrastructure Library, DIKW (Data, Information, Knowledge, Wisdom), where wisdom and knowledge derive primarily from information based on data, is worthy of consideration.

It is of comfort to know that many of the processes found in the CMMi will also appear in other bodies of knowledge, methodologies, and frameworks. Therefore, to confront and master them systematically via the process steps that comprise the CMMi should provide solid and measurable improvements when considering other frameworks for those who are able to do so. An example of this is the COBIT 5 process Build, Acquire and Implement (BAI) 01: Programmes and Projects. Knowledge of the PMBOK and CMMi processes is fully aligned with BAI01. Table I.1 in Annexure I shows the COBIT 5 processes, including BAI01.

Looking at the continual improvement process from another perspective, the concept of Experience Curves in Figure 2.5 shows that a competitor who is first to embark on a new journey, solving problems as they go, will reap benefits sooner than competitors who delay or avoid embarking on the journey. Being able to take advantage of new and innovative ideas to be able to release them quickly to market is a key difference between successful companies and those that fail. Competitors who operate at CM L2 must surely have an advantage over those who operate at CM L1 as they should be able to move quickly to deliver quality and value, especially if they are operating off a PMIS CM emplacement that is working well.

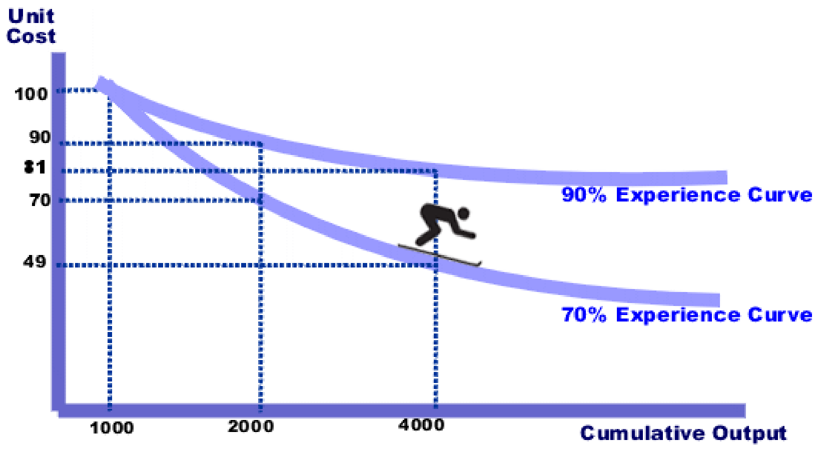


Figure 2.6: Experience Curves

Source: Synthesized by the researcher

Experience Curves in the figure explain that a company that can capitalise rapidly on an idea will gain a competitive advantage over and be ahead of its competitors. Experience gained should put them at a lower unit cost and should also enable them to produce a higher cumulative output at that cost level. In the example in the figure, the lower unit cost of 49 with an output of 4000 on the 70% Experience Curve has the advantage over a competitor situated on the 90% Experience Curve with a unit cost of 100 and output of 1000.

It is also worth noting that Experience Curves include two types of effects, namely the learning curve effect and the experience curve effect. The learning curve effect refers to experience and the fact that the more often a task is performed, the less time will be required on each iteration. The experience curve effect is more all-encompassing and states that the more often a task is performed, the lower the cost of doing it will be. This effect was discovered in the late 1960s by Bruce Henderson at the Boston Consulting Group. Robinson (1982:112) succinctly explains that “competitors taking advantage will ride the experience curve with glee.”

Mindful of Experience Curves, if a company can successfully install CM L2 processes to run projects more successfully on a PMIS emplacement, and they are able to sustain this advantage, then they should be able to produce successful software projects at a lower cost and more often than competitors. Ultimately, if experience is gained through focus and improvement on the use of the essential CM L2 processes, which should include Requirements Management (REQM), Project Planning (PP) and Project Monitoring and Control (PMC), first, this could mean that an organisation might be able to solve the CMMi complexity problem ahead of their competition and reap the productivity and quality benefits of this when also reducing the increased levels of risk, rework, and waste associated at CM L1. Applying the wisdom of Bizos mentioned previously in the section on Software Complexity, they will be able to move carefully but in a ferocious and focused manner through the tar pit, delivering newer and more valuable versions of their product without becoming ensnared with competitors operating at CM L1 not being able to be so fortunate.

2.5 Strategic direction and earned value from projects

Strategy planning, business planning, and project planning processes all have the same goal, to create value and a competitive advantage for the organisation. Planning is not worth much if ideas cannot be put into action rapidly. It is for this reason that Project Management Offices (PMOs) will attempt to convert an organisation’s strategic direction in value streams through projects, programs and portfolios.

The strategy wall in Figure 2.6 has been synthesized by the writer from a strategic processes model created by Davis et al. (2010:25). The researcher has incorporated a new Project Management Process (PMP) and Capability Maturity (CM) brick into the top right-hand side of the wall. This brick, required to focus on Earned Value Management, relies primarily on the installation and correct use of the PMIS emplacement to manage and measure strategic implementation success against the plan. This should assist an executive board to manage strategic plan success.

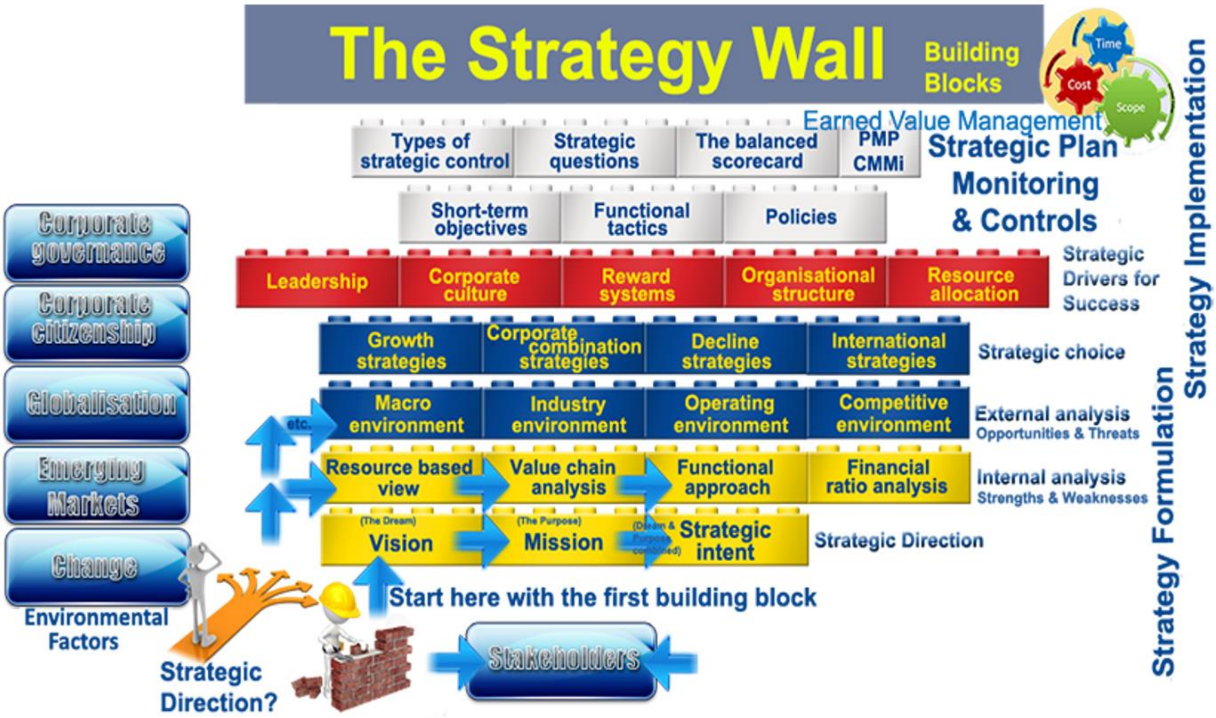


Figure 2.7: The Strategy Wall

Source: Synthesized by the researcher from Davis et al. (2010)

Figure 2.6 shows that a process approach must be followed when creating strategic plans. To this end, a user of the model above will enter at the bottom with a Vision, then move to Mission and then Strategic Intent. With the Strategic Direction pinned down, they will move to an Internal Analysis, working through each of the bricks in the second level, proceeding from there, upwards, until finally arriving at the top of the wall. As with a project plan, the ability to implement a strategic plan is an important phase of the strategic management journey. To the right of the figure, this is illustrated with the words Strategy Formulation, representing the first four levels and then Strategy Implementation as the next three levels. Put another way: an inability to implement a formulated strategy is highly wasteful. To be successful, any plan delivered must be implemented within its constraints.

As all of the five strategic drivers for success, found on the fifth row of the strategy wall in the figure, are important, leadership and resource allocation are vital from a project management perspective, as these ensure the resource loaded schedule is driven into place by top management aimed at achieving the clear goal in mind. Earning value from effort must involve the ability to apply and then effectively and efficiently use allocated resources. The resource loaded schedule is an essential enabler for strategic and project management success. It is worth noting that the resource-loaded schedule is achieved at the end of the PMBOK Schedule Knowledge Area and Planning process #6.5, as is shown in Annexure S. Essentially, a project schedule that is not resource loaded should not be moved into the Execution Process Group.

2.6 **Strategy and Planning Success**

According to Rensburg & Davis, 2010:14 organisations are very much like human beings who also find it easier to dream and make plans than putting these dreams and plans into action. Strategic planning is one dimension of planning. However, this research understands that Business Planning, Project Planning, and even Balanced Scorecard all utilize planning. Rensburg & Davis, 2010:141 mention that the Balanced Scorecard Collaborative found that nine out of ten organisations fail to implement strategies, and as few as 10 per cent of formulated strategies are implemented. According to Pienaar et al. (2008), this understanding is borne out by the Economist Intelligence Unit of Marakon Associates and the Corporate Strategy Board, who found that as much as 37% of the potential value of a strategic plan was lost due to unsuccessful strategy implementation. The reasons for this loss in potential value include inadequate resources, poor leadership, unclear accountability for implementation actions, inadequate performance targets and rewards, organisational silos, cultural resistance to change, and a poorly communicated strategy.

Beer, Eisenstat, University., *et al.*, 2000 stated that the problem associated with failed strategy implementation pertains to poor management in respect of unclear strategy or conflicting priorities, overly controlled environments or a non-directive laissez-faire management style, lack of teamwork, poor coordination across units, and no commitment of middle management staff. It is worth noting that this is very similar to the finding of the original Standish Group Chaos Report (2014:9), which indicated that software development companies often experience numerous project challenges and impaired factors, including limited support from the Executive, a lack of vision and objectives cumulating in unclear statements of requirements, limited user involvement, an inability to plan correctly, competent and trained staff, and unrealistic expectations.

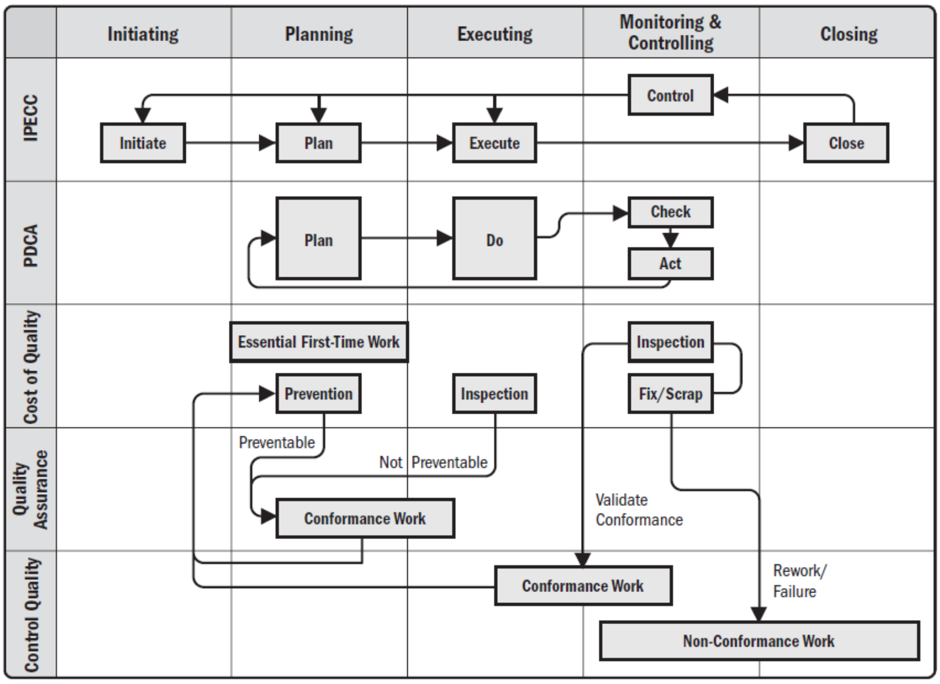
Schach (2011:5) identified managerial challenges as a main result of the Chaos Reports findings. They found that only 35 per cent of projects were completed successfully, and 19 per cent were either cancelled before completion, were over budget, were late, had fewer features and functionality than initially specified, or were never implemented. They also found that just over one in three software development projects was successful, and almost half the projects displayed one or more symptoms of the software crisis. According to Guess (2006:25), the reasons why software development projects failed or were challenged were attributed to poor user inputs, vague requirements, poor cost and schedule estimates, failure to plan, and late or ignored failure warning signals. According to Hirsh (2012:6), despite criticisms, the Standish Chaos Reports have caused many organisations to question their approach to project management. There appears to be agreement that renewed focus has caused project management maturity to rise in certain areas.

More recently, Musawir, Serra, Zwikael, *et al.*, 2017 found that project governance for improving strategic and project success included the development and monitoring of a high-quality project business case. Furthermore, research conducted by the Project Management Institute (2017) found that a majority of senior leaders acknowledged that their organisations often struggled to bridge the gap between strategy formulation and its day-to-day implementation. In addition, the executive leaders surveyed reported that in the last 12 months, only 60 per cent of their strategic initiatives met their goals.

2.7 **The Plan Do Check Act Cycle and Game Theory**

Deming's Plan Do Check Act Cycle was discussed in Chapter 1. It is mentioned again to illustrate its relationship to the discussion in Sections 2.5 and 2.6 and to place it in context via Table 2.3 with Quality Assurance and Control Quality to the IPECC, Plan Do Check Act, Cost of Quality Models, and the Project Management Process Groups from the PMBOK.

Table 2.3: Plan Do Check Act and Quality Control



Source: PMI (2013:257)

The table above compares several quality assurance methods that are based around the PMBOK 6 Process Groups, of which PDCA is one. For each of the methods, the ability to manage quality depends on Project Planning (PP) and Project Monitoring and Controlling (PMC) being in place. As previously mentioned in Section 1.6**,** keeping a project on course and continually improving project quality is understood by using PP and PMC in much the same way that drivers position their hands on either side of a steering wheel, ensuring the car maintains its direction safely on the road and on track towards its goal.

In Figure 2.7, the researcher has built a model based on the PMBOK 6 Knowledge Areas, which considers project management from a game theory perspective. Essentially a project's success or failure is measured against its ability to produce the value envisioned in the project charter and plan. The model shows that the ability of a project to hit its planned objective should be its measurement of success if game theory is to be used to promote and focus productivity. As both traditional and agile project methods have defined processes, game theory can be used to measure and improve productivity and quality outputs. An important difference between the traditional and agile approaches is that agile has a much smaller planning horizon.

Also, agile keeps itself open to the possibility of rapid changes in direction, if these are required.

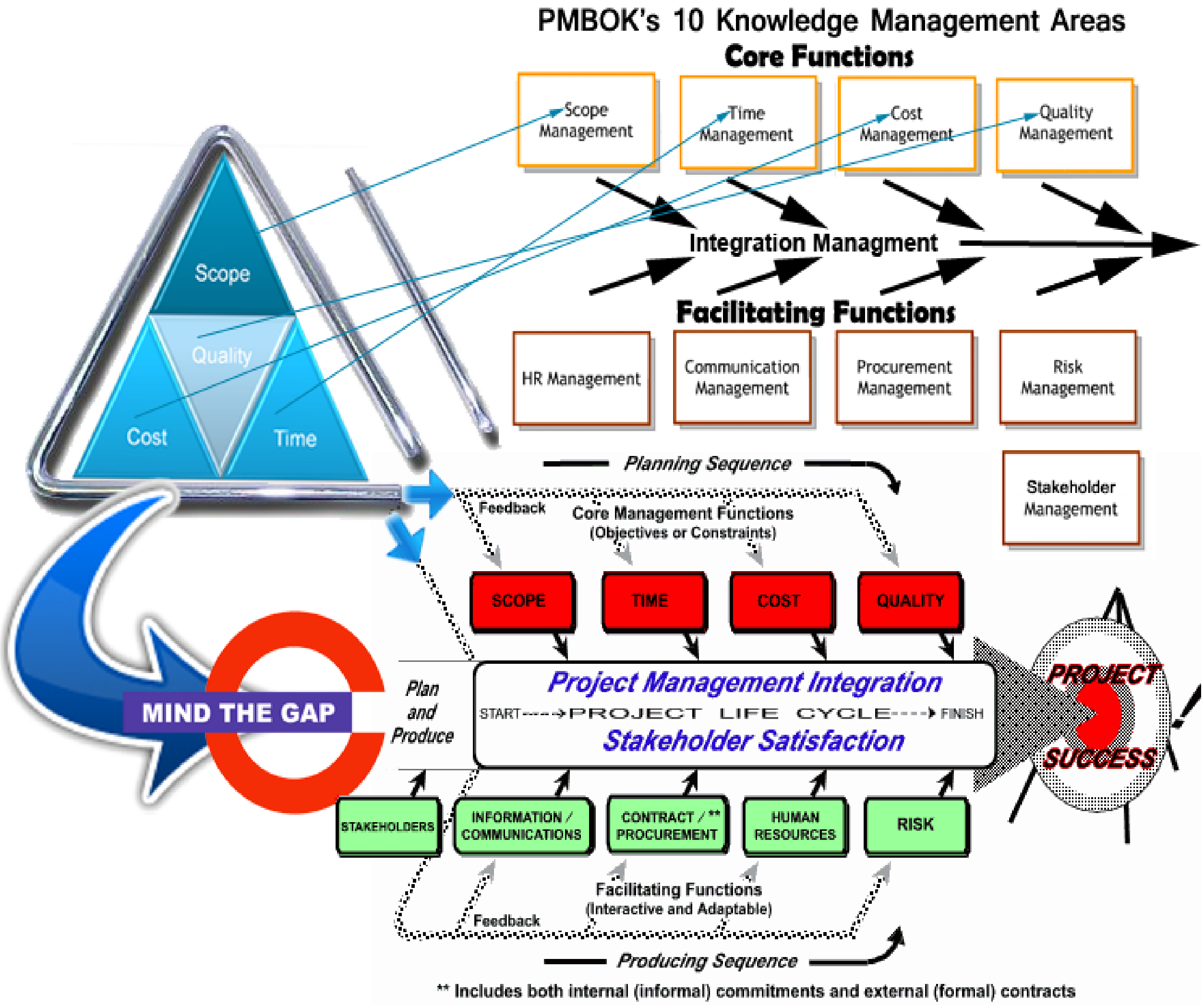


Figure 2.8: The PMBOK as a game

Source: Synthesized by the researcher from Wideman (2001)

The figure is synthesized from the PMBOK guide process group and knowledge area matrix in Annexure S**.** Even though the PMBOK Process Groups are not specifically shown, they are inferred, and the model assumes the full project lifecycle or phase from Initiating to Closure. Project success is represented by integrating the PMBOK Knowledge Areas in order to hit a target. Typically, this involves being on time, within cost and having the ability to deliver value when compared to the planned scope. Time, cost and scope are the key constraints in any project. Consequently, the Iron Triangle has been superimposed on the figure. If Iron Triangle's points of time, cost, scope in Figure 1.1 at the beginning of this document are regarded as the core constraints, it is understood by those who understand the PMBOK processes, that the interaction of these three points in the Iron Triangle, if expanded or contracted, will have a corresponding positive or negative effect on project quality.

In addition, project management requires the careful use of time, cost and scope to mind the planned gap from a current or existing state to a new state in order to be able to achieve project success. In software, states are versions of software that are released. The dynamic interaction between time, cost, scope and quality is summed up in project management circles by the following statement: Fast, cheap or good - you can have any two.

2.8 The Project as Budgeted Requirement

This section offers more detail about PMBOK Cost Management Knowledge Area and also Earned Value Management (EVM). Detail is included to illustrate the complexity that will need to be considered if the CM L2 process PP and PMC, and EVM are to be installed correctly and what may be needed for optimal operation.

Software engineering appreciates that according to Parkinson's Law, "work expands to fill the time available for its completion”. This seems to imply that other constraints can be applied to better manage project work, rather than simply relying on the essential constraints of Time, Cost and Scope. To this end, Project Management Professionals understand that the constraints of Time, Cost, Scope and Quality must be expanded to include Risk, Human Resources, Procurements, Communications, and Stakeholders; essentially the other Knowledge Areas from the PMBOK (Annexure S), if projects are to be governed successfully. This could be another reason why forsaking the PMBOK processes for agile run heroically at CM L1 could be counterproductive.

The movement of project tasks through a project lifecycle from an Earned Value Management perspective, once the budget and schedule are set in execution, can either deliver successfully or will fail in the attempt. As budget is often an essential driver for project health, and budget has a direct effect on time and scope (schedule), nominal telemetry derived from this metric would be highly beneficial for Project Management Quality Assurance. Figure 2.8 is the Earned Value Management graph, which forms the essential management output in the Practice Standard for Earned Value Management from the Project Management Institute (2011:2). Earned Value

Management was initially created by the United States Department of Defence. They imposed 35 criteria on a contractor's management control system at any time costs or incentives were referred to. As this was complex the system was radically revised into ten fundamental steps to implement Simple Earned Value Management (Fleming & Koppelman 2010:30). This Simple EVM is Earned Value for the Masses, and the refined approach contains only the minimum requirements necessary to employ simple earned value (Fleming & Koppelman 2010:46).

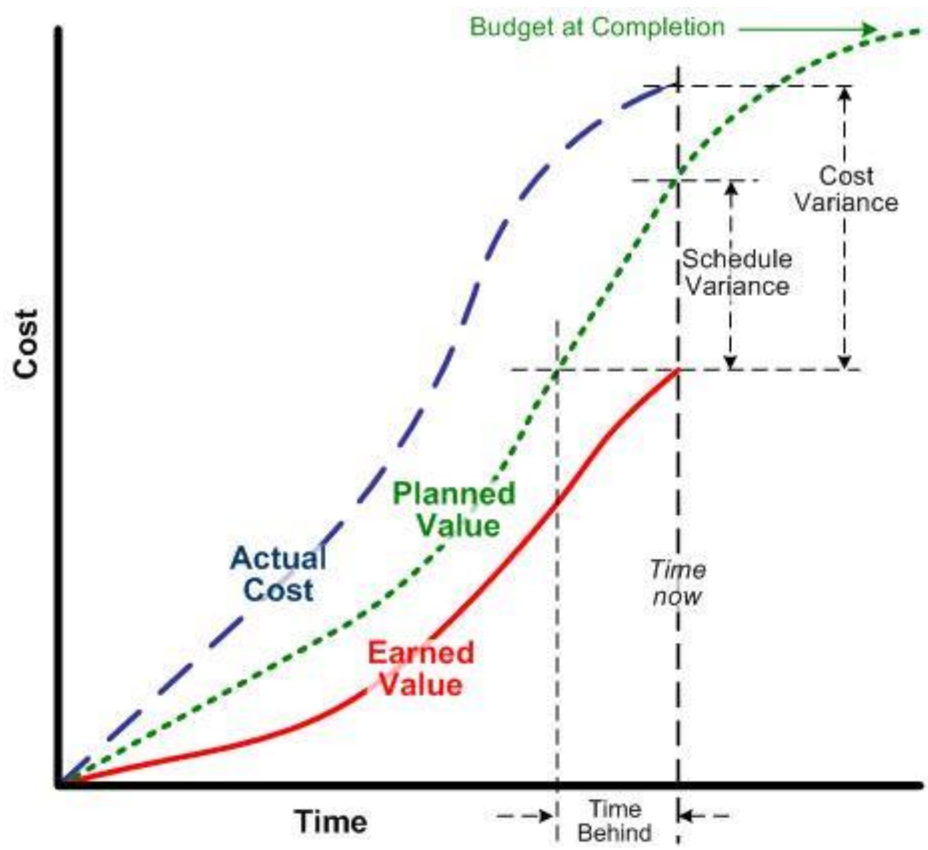


Figure 2.9: Earned Value Management

Source: Project Management Institute (2019)

The list below shows some of the more important calculations that are available from the figure:

1. Cost Variance (CV) = Earned Value (EV) – Actual Cost (AC)

2. Schedule Variance (SV) = Earned Value (EV) – Planned Value (PV)

3. Cost Performance Index (CPI) = EV / AC

4. Schedule Performance Index (SPI) = EV / PV

5. Estimate At Completion (EAC) = AC + Bottom-up ETC

6. EAC = Budget At Completion (BAC) / Cumulative CPI

7. EAC = AC + (BAC – EV)

8. EAC = AC + [BAC – EV / (Cumulative CPI ´ Cumulative SPI)]

9. To Complete Performance Index (TCPI) = (BAC - EV) / (EAC - AC)

It would appear from the researcher's experience that many projects are run with a disregard for the mentioned essential controls. These often unused controls, and the resultant inability to fully control the project while it is in execution based on feedback from them, could be regarded as a key driver of chaos with regard to the Standish Group (2014). The PMIS emplacement, correctly set up for CM L2 by following the PMIS CM Improvement Framework developed in Chapter 4, if just to install and use the above project management control formulas, should greatly help to reduce the chaos referred to in the Standish Chaos Reports.

According to Fleming & Koppelman (2010:175), it does appear that many executives are likely to measure project performance with only two dimensions: Projected costs vs Actual costs. Thus, if the budget is spent, the project is regarded as being on target. If less is spent, then there has been a cost underrun and if more, an overrun. This is not cost performance but rather funding performance. What is missing is the measurement of the value of the work performed for the finances spent, which is earned value management. The benefit of Earned Value Management is that, apart from the ability to utilize an array of formulas, a single number called the To Complete Performance Index (TCPI, point 9) can be used at any time throughout the project lifecycle to measure project performance at that point. Naturally, the TCPI and other ratios can only be produced after a project budget or baseline has been defined and committed to a baseline.

As CM L2 focuses on Project Planning (PP) and Project Monitoring and Control (PMC) as primary measures for success, if a PMIS emplacement is used, then this could facilitate the management of actual task status against planned status in order to apply the cost of time spent as a computerized calculation in real-time. Without the PMIS emplacement, due to the complexity of trying to manage these metrics manually, the effort expended will be unlikely to provide project stability at CM L2. From the researcher's experience, if Earned Value Management must rely on manual calculations, it will not be successfully implemented.

The criticism levied against earned value is that if it is so good then why is it not used on all projects? The fact is that it is very difficult to manage the complexity of Earned Value Management without a system to monitor and manage the calculations. It is only recently, with advancements in the internet, collaboration, and the arrival of PMIS's, that Earned Value Management could begin to be considered more seriously as a viable option. The researcher believes that a well-run PMIS emplacement database should be able to automate all Earned Value Management graphs if the "developed schedule" is constructed and baselined. These graphs, formed in real-time as the project unfolds, would be easy to understand and be a snapshot of actual project status, facilitate better project management. With updated Earned Value Management available in real-time from the PMIS emplacement, a project manager and team can ensure project safety and health at any time during a projects' lifecycle. In addition, with the ability of the PMIS emplacement to monitor and control variances on a developed schedule, the field is set up to run projects from a game theory perspective.

Furthermore, the researcher understands that Earned Value Management graphs can only function correctly if the developed schedule is baselined prior to a move to the PMBOK Execution Process Group, as is illustrated in Annexure S. Using the PMIS emplacement for Earned Value Management to track tasks by resource responsible, measured on an hourly basis (or in smaller increments) against value produced from the tasks, could be a natural extrapolation of the PMIS emplacement capability. Even as the sticky note on a physical agile board may seem like it is promoting agility, when compared to the requirements needed for the developed schedule in Annexure S, sticky notes even on an electronic board must be limited in their ability to provide sufficient detail to operate successfully at CM L2 and above. This is due in part because sticky notes on an agile board capture points in time of expected delivery and do not include start and end dates and the time that it takes to deliver value.

Minding the gap in Figure 2.7 above refers to achieving project goals within constraints. The idea of minding the gap is derived by the researcher from the British Underground symbol and safety mantra advising commuters to be careful when boarding or leaving trains. The gap is clearly understood to be between the project plan or iteration and actual work produced in project execution during the plan or iteration. As the project management community reminds itself constantly, if the gap is widened, trying to jump across it, chances are good that the project team will land in it. The project community are also aware that a project becomes late, a day at a time. Therefore, the sooner a negative variance between actuals and planned performance metrics are acted upon, the sooner appropriate corrective action can be taken. The danger of scope creep and the importance of locking down the constraints between project lifecycles or project iterations are essential aspects of professional project management at CM L2+.

2.9 The Project Software Product Configured

This section presents some detail pertaining to the Configuration Management (CM) process in the CMMi. It is included to illustrate the complexity if the CM L2 process REQM and CM is to be installed correctly and what may be needed for optimal operation.

Minding the gap, from a project management perspective, is also understood as the gap between a current state and a new state. In software projects, this is the state of one system software baseline or version and the next. The V-Model in Figure 2.9, also called the Verification and Validation Model has been used for many years. As the V-Model stems from early Software Development, Life Cycles can be applied to Software Project Management as well. It has been synthesized by the researcher to illustrate how the V can also be understood as the gap to be successfully crossed to achieve the next System Configuration Baseline. Moving successfully to the next System Configuration Baseline means that one has made progress through the tar pit without becoming ensnared or trapped in the tar.

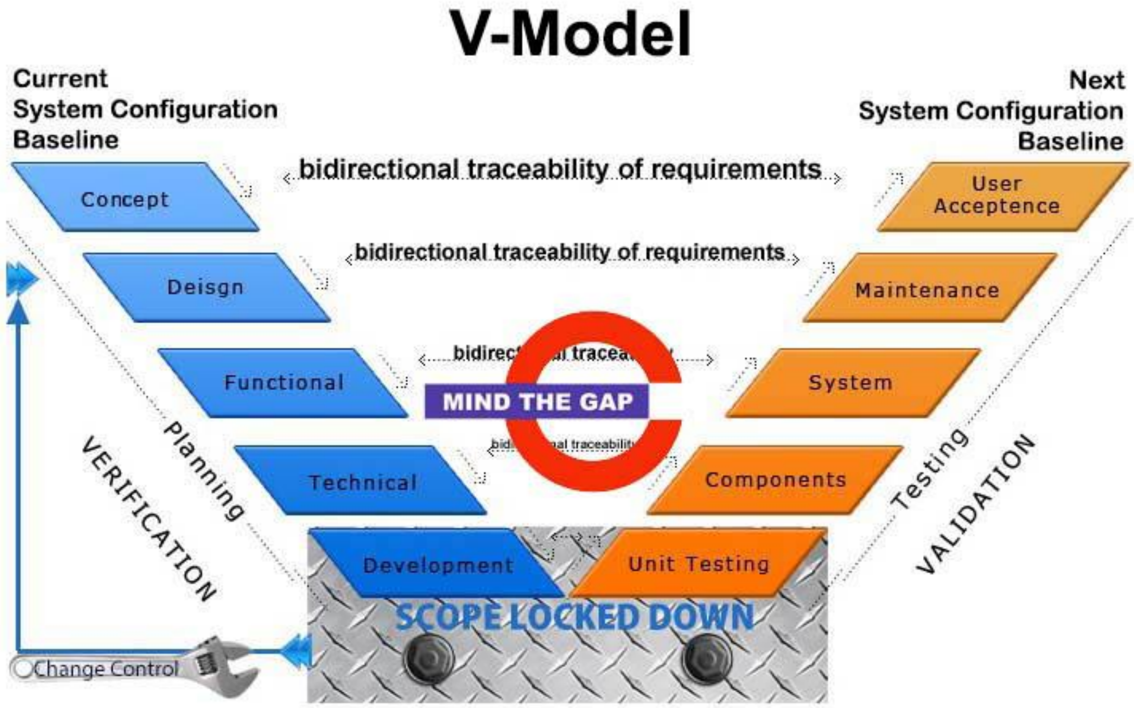


Figure 2.10: The V-Model

Source: Synthesized by the researcher from Tutorialspoint

The figure illustrates that project scope comprised of defined project requirements is verified on the left of the V. Verification is the business agreeing that a requirement document accurately reflects what they need. In agile, or the Scrum method, typically a User Story is used to pin down requirements. After development, these should be validated through testing and final user acceptance approvals before a “next software version” is finalized and rolled out. We can see that Figure 2.9 synthesizes several CMMi processes that are found at CM L2 & 3. Essentially the V-Model shows that Requirements produce a next version of value (CM L2 process REQM) to be verified (CM L3 process VER) before work begins. Validation (CM L2 process VAL) after work completion means that improvements to newly built software have been fully tested and approved. Minding the gap for approval at a next stable version can be seen as a segregation of duties between those who are custodians for the current system version and those striving to replace the existing system baseline with an improved baseline. Configuration Management (CM L2 process CM) is used along with the REQM process to maintain bidirectional traceability between successive software versions.

According to Chrissis, Konrad, Shrum, *et al.*, 2011:531 & 541 from the CMMi for Development model V1.3, the purpose of Verification (VER) is to ensure that selected work products meet their specified requirements. The purpose of Validation, according to the CMMi, is to demonstrate that a product or product component fulfils its intended use when placed in its intended environment. There is a close connection between Requirements Management, Configuration Management and bidirectional traceability of requirements Chrissis, Konrad, Shrum, *et al.*, 2011:576 Bidirectional traceability of requirements ensures that traceability exists between baselines with the goal to be able to maintain the bidirectional traceability of requirements for each level of product decomposition. Therefore, the V-model and the CM processes it encapsulates, as a step-by-step approach to minding the gap, should be able to introduce higher levels of safety traversing the tar pit.

According to the PRINCE2 methodology (Projects IN Controlled Environments Annexure K), a core principle of the method is Continued Business Justification. The Business Case in PRINCE2, authorised by the Project Board, is the way a PRINCE2 project is initiated. PRINCE2 Manual (2009:25) explains three basic business options concerning any investment, namely, do nothing, do the minimum, do something. Do nothing should always be the starting option to act as the basis for quantifying the other options – the difference between do nothing and do the minimum or do something is the benefit that the investment will buy. The starting point, from a PRINCE2 perspective, if given a choice to run a project, is not to run a project.

Configuration Management, from the Configuration Model Figure 2.10, aims to put into place a current baseline upon which new (beneficial) requirements are implemented to form a new baseline. An existing configuration baseline is needed to measure configuration conformance after a change is completed. Guess (2006:21) refer to Configuration Management as the process of managing an organisation's products, facilities and processes by managing their requirements, including changes, and assuring that results conform in each case. Process improvement, per Capability Maturity II, is measured by the ability to "change faster and or document better."

From a Capability Maturity II perspective, Guess (2012:6) explain that configuration occurs from one baseline to the next. The CM process model starts at a Strategic Business Plan, then moves to As-Planned and As-Released Baselines, 9-Step Development Process, Naming, Numbering and Reuse, Data and Record, Integrity, Validation and Release Records, Changes and Revision Records, and finally As-Built Records. If Configuration is not adequately addressed, the organisation may remain at Configuration Management level 1, stuck in corrective action mode, as the figure below illustrates.

Figure 2.11: CMII Implementation model

Source: CMII Research Institute (2010:7)

Figure 2.11 shows that the goal of CMII Level 5 is to move out of corrective action mode. Robust improvements happen at this level, and the return on investment is large. From a configuration management perspective, no change to an information system with its baseline defined must occur without valid justification and authorisation. All change must be quantifiable via an audit trail of configuration records from the current baseline to the new baseline.

With the complexity involved in Configuration Management, it stands to reason that the PMIS emplacement should be able to assist to reduce this. As changes to a system baseline will be in the form of software features, the PMIS can be configured to include the necessary configuration records. In addition, as part of the action of closing out the task, the PMIS can ensure that this does not happen until the configuration system is fully updated.

2.10 A Model to manage projects within the Public Sector

The Meril-De Model (Olivier 2015), of which Figure 2.11 is a part, presents a 9-component model for successful strategy execution. It is designed with the Public Sector in Africa in mind. This model is included to support the Case study in Chapter 5, which is based in the Public Sector.

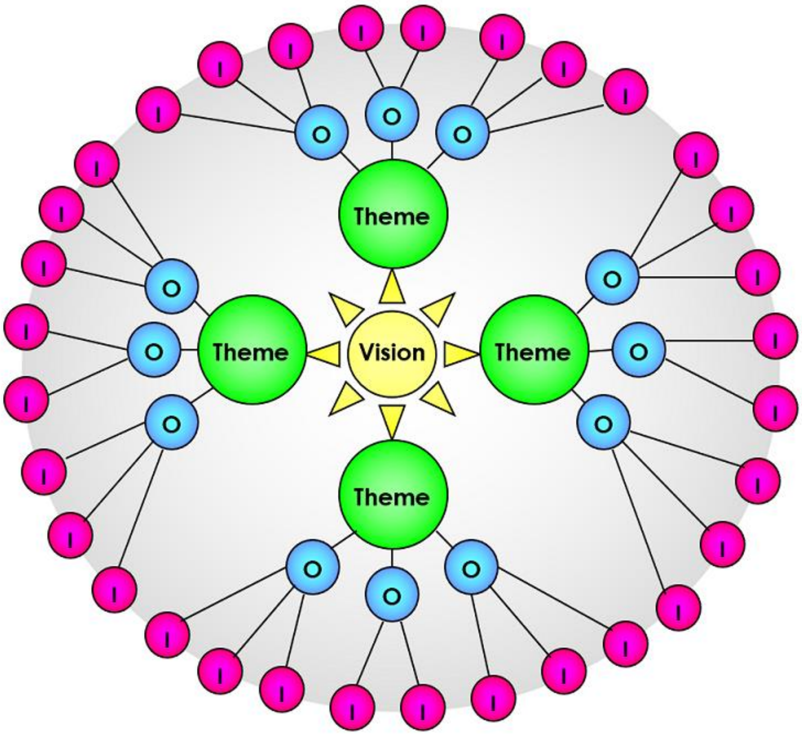


Figure 2.12: Meril-De Programme & Project Initiation Processes

Source: Olivier (2015:5)

Figure 2.11 explains the approach of using strategic planning with themes and objectives, which are then used to define projects. While project methods do attempt to do this naturally, the figure above and Olivier's research show clearly the importance of initiators when running a project or multiple projects in a program. The projects, either on their own or combined into programs or portfolios of work, are kicked off via project initiation processes. The initiators (I's) from the figure compliment the Initiation Process Group, or the starting point for project management, with regard to the PMBOK process matrix, as seen in Annexure S. With the PMIS emplacement supporting EVM telemetry in place and nominal, and following the Olivier (2015:5) project approach, project teams operating off a PMIS emplacement and utilizing game theory could be well-positioned to deliver value well for stakeholders in any sector.

The fact that a project needs to be initiated officially also ties into the concept of a project being a temporary endeavour that has a start and a finish. Often, when the environment is at Capability Maturity Level 1, projects are “spun up” on executive whim and pressure is exerted to do them, but often without much planning. Under the guise of agile it is the researchers experience that more often than not these types of projects will not be successful. A key reason ties back to the Initiation process of the PMBOK which asks for a Project Charter and empowered Stakeholders to approve the project, which is a formal process that needs to be completed, regardless if the project is Traditional or Agile in construction.

2.11 **People-Process-Technology Triad**

Annexure A explains that software projects need to operate with an awareness of the People-Process-Technology triad. Figure 2.12 shows the People, Process and Technology triad and how the positive interaction of each outer node can facilitate success. This is often referred to as the People, Process and Technology golden triangle.

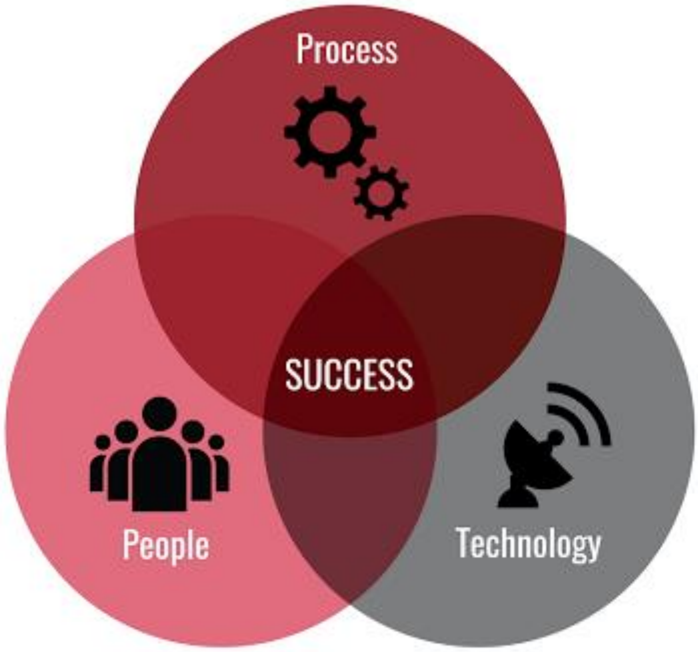


Figure 2.13: The People Process Technology Triad

Source: Google Images

The People-Process-Technology triad is understood as a cornerstone of the CMMi. In addition to the People-Process-Technology dimensions, there are also Data, Governance and Execution dimensions that are interconnected. Wagner (2017) proposes the following questions that can be asked of the triad:

1. People:
    - Do the employees have the rights skills?
    - Are the staffing levels appropriate?
    - Does the organisational culture encourage desired behaviour?
    - Is behaviour based on best practice?
2. Process:
    - Is process defined and understood?
    - Is following process a workflow pain point?
    - Are all staff using the intended processes?
    - Are appropriate incentives in place to ensure process compliance?
3. Technology:
    - Are the needed tools and equipment in place?
    - Do all staff have this technology?
    - Do the tools have the functionality staff need to perform their jobs efficiently and effectively?
4. Data:
    - Is data, the lifeblood of the triad above, able to be converted to information that should then create knowledge and wisdom?
5. Governance:
    - Is governance, as a direction and vision from appropriate, empowered leadership, in place?
    - Does decision-making align with the strategic goals of the organisation?
6. Execution:
    - Are there management, planning, technology and staff resources to ensure projects are appropriately executed?

From the perspective of Total Quality Management, Lean and the Capability Maturity Model integrated, ideas derived from Deming are often mentioned. According to Deming (1986:121), there is a supposition that is prevalent the world over, that there would be no problems in production or service if only our production workers would do their jobs in the way they were taught. Unfortunately, this is not so. The workers are handicapped by the system, which belongs to management. Deming (1986:8) also says it is not enough that management commits themselves to quality and productivity, they must know what it is they must do. Such a responsibility cannot be delegated. A process can be stable, in control, and producing defective items 100% of the time. Production standards and numeric goals can be meaningless. Extrinsic motivation is not effective; according to Deming, people are already doing their best. The problem is most often with the system. By using rewards and punishment, management, in fact, is often tampering with a stable system. Only management can change the system, and it is the system that must be fixed and not the workers.

The researcher understands that Deming's influence essentially requests behaviour at CM L2 rather than a managerial insistence of "just do it" and heroics[8], as was explained in Chapter 1. Chrissis, Konrad, Shrum, *et al.*, 2011:4 elucidates Deming's wisdom by stating that while process is often described as a node of the People-Process-Technology triad, it can also be considered the glue that ties the triad together. Essentially while everyone realises the importance of having a quality and motivated workforce, even the most talented staff might not perform at their best if the system and its processes are not clearly understood or operating optimally.

The researcher postulates that a PMIS correctly installed will put the onus and responsibility for project delivery back onto the executive, where it ought to be. This is especially relevant from a Risk Management perspective. In this regard, when a project becomes stuck, and the project team is unable to intervene, the PMIS will ensure that a rapid risk response is elicited from the executive. The 12 dimensions of the PMIS Improvement Framework in Chapter 4, if used correctly by the executive who is ultimately accountable for project success and good governance, should go a long way to ensuring project management is run successfully at CM L2+.

[8] In this dissertation, "heroics," or being pressured by management to "just do it," is understood as operating without due consideration for process application and at Capability Maturity Level 1. If Deming's wisdom discussed on this page is accepted, then it is management that needs to change the system rather than disempowered workers. The concept of heroic effort is illustrated in the model in Annexure C.

2.12 Resource Allocation Syndrome

The People Capability Maturity Model in Annexure H explains how to move towards the “Empowered Team.” Resource Management is handled by the PMBOK in the Execution Process Group within processes #9.3 and #9.5, as resources are monitored and controlled through process #9.6.

According to Tromp & Homan, 2015 the main challenge in a multi-project environment is the lack of dedicated teams on one project. The human resources are expected to be focused, and results demanded, sometimes simultaneously on different projects at the same time. Although different projects may not be related, simultaneous demand on employees could interconnect projects and create interdependencies. This will surely contribute to the complexity that will be experienced. Engwall & Jerbrant, 2003:408 suggest that resource allocation and over-allocation are consequences of flawed organisational procedures rather than poor project management practices. They state that addressing the root cause of this syndrome is a task for executive management. In their conclusion, they found that the resource allocation syndrome of multi-project management is not an issue in itself but rather “an expression of many other, more profound, organisational problems of the multi-project setting.”

Ponsteen & Kusters, (2015) found that a different method was needed to allocate resources than was typically used for a single project. Ponsteen & Kusters point to several dimensions in their study, one dimension classifies how the scheduling problem is approached, relying on human insights or optimization algorithms. From the perspective of this work, the researcher understands that while staff can be appointed to work on different projects and project schedules in a PMIS emplacement, they cannot be over-allocated due to system controls that prohibit it.

While project teams may appear happy to be allocated to project schedules and may even agree to undertake tasks on a project schedule, it is only when task delivery can be approved, an automated process on a PMIS emplacement, that real commitment and productivity can be ascertained. Tromp & Homan, 2015 concede that despite an indication that team leaders can claim that they were on board and were working towards new routines and procedures, “it turned out that they didn’t change at all.” They also warn about the difference between on-stage and off-stage behaviour. Scott (1990) describes this as the difference between public discourse and hidden transcripts. This means that people can pretend to go along with a new approach but may not be convinced and can be resistant or subtly undermine efforts towards change. In this regard, the PMIS emplacement tasks are completed in a pending state by the project resource working on them and then sent to the project manager for approval via PMIS workflow, which then fully updates the task to be complete. This approach facilitated by the PMIS is more in line with operations at CM L2+ than CM L1.

Zaman *et al.*, (2019) appreciate that there is a soft side to project management where prioritised focus and attention to behavioural features of employees while finalising project teams could be a requirement for project success. Luthans (2012:339) understands that problems associated with demotivated and dysfunctional teams typically involve a range of phenomena, including conflict, norm-violation, role ambiguity, groupthink, conformity problems, and social loafing. These are exacerbated by ineffective leadership, lack of clear goals, direction or priorities, lack of skills, a lack of cooperation, and poor communication.

Developing a schedule according to process #6.5 on the PMBOK can facilitate that project tasking involves staff who agree that they are able to do the work. Staff need to accept the task as the right thing they can do next in a collaborative environment in order to ensure the schedule is developed, rather than being delegated a task by a project manager. Often the staff themselves will create their own tasks and agree to them as part the developing a scheduled process that the members conduct and construct on behalf of their team. Staff who are over-allocated should not be able to task themselves on a preliminary schedule at the same time as they are allocated elsewhere on other schedules. The fact of tasking on a schedule, moving towards an agreement that the schedule is fully developed, happens in the planning process. This implies that teams should work better together, knowing that they agree upfront in the Planning process and before fully developed schedules are baselined and moved into execution on the PMIS. In this regard, there is agreement that work-allocation for the next phase or agile sprint is fully understood, fair, and achievable by the team member that is taking on the work.

PMIS emplacement facilitates team collaboration and cohesion even if teams are separated globally, especially if unique skills from specific resources need to be used on different projects. As the PMIS emplacement empowers global connectivity to projects by anyone who has access rights and an internet connection, the system is also able to tie the project task to its owner within a project at an agreed time in the day, ensuring over allocation does not happen.

2.13 EVM on the PMIS for Good Governance

From a project management perspective, good governance means excellence in execution. The Project Management Institute, for example, insists on projects being managed with ethics and good governance and, similar to the auditing and accounting profession, if there is impropriety, a Project Management Professional's certification can be revoked.

The Execution Process Group lies between the Project Planning Process Group (PP) and the Monitoring and Control Process Group (PMC). Therefore, correctly applied PP and PMC processes are essential for good management, ensuring Execution remains on track and able to deliver. Many practices and frameworks incorporate and address the concept of good governance. In Section's 2.13.1 to 2.13.4, a discussion on GAAP, SOX, the PFMA, and King III explains the concept of good governance in more detail. These all underpin the ability, desire, and legal necessity to achieve the state of good governance.

2.13.1 GAAP and IFRS

GAAP (Generally Accepted Accounting Principles), understood to incorporate IFRS (International Financial Reporting Standards), is focused on financial statements in general and assumes that the following ten principles are in place and operate well:

- Principle of Regularity
- Principle of Consistency
- Principle of Sincerity
- Principle of Permanence of Methods
- Principle of Non-Compensation
- Principle of Prudence
- Principle of Continuity
- Principle of Periodicity

- Principle of Materiality / Good Faith
- Principle of Utmost Good Faith

The IFRS set common rules so that financial statements can be consistent, transparent, and comparable globally. IFRS are issued by the International Accounting Standards Board (IASB) (Standards 2020).

The utilisation of GAAP principles and processes can fine-tune the implementation of the PMIS emplacement towards a tighter application of financial monitoring and control for the Project Cost Management Knowledge Area per Annexure S. Project tasks on a developed schedule, which are also tracked against Earned Value Management statistics, can promote higher levels of excellence in project execution. Overdue or late projects will drain project budgets, which translates into debt the organisation will need to carry without producing the value for which it had planned. Ability to utilise Earned Value Management means the project is monitored and controlled correctly per PMBOK's 4th Process Group offering support for the Project Cost Management Knowledge Area processes.

The PMIS emplacement, run correctly from an Earned Value Management perspective, will facilitate automation of the CM L2 Measurement and Analysis (MA) process. Earned Value Management techniques can also assist with the installation of the CM L3 Risk and Issue Management (RSKM) process, upon which it depends. Therefore, if challenges arise, based on risks identified, then corrective action can occur timeously. In turn, this fully supports GAAP, which aims to facilitate the ability to manage the financial position of the organisation better. Annexure J, which deals with the process requirements of Control Objectives in Information Technology (COBIT 5), deals with the need to produce planned value, facilitated through the use of Earned Value Management.

Section 2.13.2, about the PFMA is touched upon as the researcher is in South Africa and has worked for a number of years in the Public Sector. In the New South Africa there has been a crucial imperative to use software engineering in the public sector to improve the lives of millions of previously disenfranchised citizens. In a similar way that international standards like regulations such as FINRA and Sarbanes Oxley (SOX) are used outside South Africa the PFMA is in place in South Africa to ensure that public funds are used correctly and with good governance. From a

Capability Maturity level perspective misuse of any public funds could be seen as operating underneath Capability Maturity Level 1.

2.13.2 PFMA

The regulation of the management of finances for national and provincial government resorts under the Public Finance and Management Act. The Act sets out the procedures for efficient and effective management of revenue, expenditure, assets, and liabilities. It also establishes the duties and responsibilities of government officials in charge of finances. The Act aims to secure transparency, accountability, and sound financial management in government and public institutions.

The National Treasury, consisting of the Minister of Finance and National Departments responsible for financial and fiscal matters, is the main body that oversees the implementation of the Act. The accounting authority (either a board or other controlling body, or a CEO) protect the assets and records of the public entity and aims to prevent damaging the financial interests of the State. In addition, the Minister of Finance sets up systems for addressing financial misconduct and criminal charges. According to the National Treasury (2010:66), criminal offences include the following:

- wilful or gross negligence on the part of an accounting officer,
- wilful or gross negligence on the part of an accounting authority, and
- rationalise loans or entering a binding financial contract on behalf of a department, public entity or constitutional institution without permission.

2.13.3 King IV and Good Governance

The King IV Report (Shango 2016) was released in 2016 and is the latest release in a code of Governance Principles for South Africa. In the overview of the report, King IV can be understood in a single word, ‘transparency.’ Expanding on previous work, King IV held that solid corporate governance was a crucial component for corporate citizenship. Corporate governance asks that an organisation not to see itself operating in a vacuum. Rather it must position itself as a part of its society, accountable towards it and to future stakeholders.

The Institute of Directors (2002:17) explained how corporate governance was important for the development of the nation. Accordingly, governance will position the value delivery systems of society where business finds itself operating. Furthermore, corporate governance can also help to prevent corporate disasters, such as Enron, WorldCom, Parmalat and recently the Fidentia crisis in South Africa, which obliteration of stakeholder and shareholder wealth largely happened through bad governance and corporate malpractices.

The PMIS emplacement, facilitating transparency in tasking (Who, What, by When, at What Cost), can be used to measure and track success or failure against agreed baselines. Without the PMIS emplacement, it is doubtful that the project manager will be able to adequately manage task complexity to the extent needed to feed Earned Value Management and financial reporting to the degree that governance requirements are adequately satisfied. It stands to reason that a PMIS emplacement facilitating Earned Value Management from individual tasks, as does a financial management system from individual transactions, can empower the ability to manage projects for compliancy and transparency for good governance. A PMIS can also be used to escalate financial risk to executive management, who are in fact the stakeholders for project governance and success.

2.14 **Methodology, Agility and Risk: A Horizontal Dimension**

The previous discussions focused on the vertical or process improvement dimension, as illustrated in Annexure A. The essential idea was that productivity and quality could be improved upon by installing CMMi processes which would work to remove risk, rework, and waste.

2.14.1 Overview

This next section aims to investigate project methodology, agility, and risk as a horizontal dimension. In essence, per Annexure A, Traditional Project Management (TPM) is situated on one side of a horizontal continuum and Agile Project Management (APM) on the other side. The assumption is that there should be more risk associated with agile project methods than with traditional project methods.

2.14.2 Introduction

The researcher, who works full time in project management, is often told that the PMBOK is traditional project management utilising a waterfall approach, yet this is incorrect. In fact, the latest version of the PMBOK embraces Traditional, Agile, Iterative and Adaptive environments. The PMBOK is primarily a Body of Knowledge. However, it is also a method that can accommodate full project lifecycles or project phases, as its ‘Close Project or Phase’ (process #4.7, Annexure S) elucidates. In addition, PMBOK processes can be configured in size, complexity, and speed, and process flows can be made to iterate with outputs leading back to prior processes if needed (PMI 2017; Vargas 2017).

The Project Management Body of Knowledge (PMBOK) 6 but can also be used as a general approach or methodology for running projects. In addition, several other formal project management methodologies exist. These include, but are not limited to, PRINCE2 Agile (2015), formerly PRINCE2 (OGC 2009); Dynamic Systems Development Method (2014); and SCRUM (2016). Table 2.4 places the PMBOK, regarded as Traditional Project Management, in context with some of the well-known project methodologies, standards and frameworks. From the researcher’s perspective, the fact that agile is missing in Table 2.4 could be partly due to the relative newness of agile when compared to the project methodologies found in the table. The absence of agile, when Table 2.4 was created, could also be due to the fact that agile was not considered an adequate or proven method to run projects correctly from a good governance perspective, which could introduce risk in project management.

Figure O.3 in Annexure O clearly shows that the Scrum agile method has 19 processes, and Risk Management is handled within the Scrum Aspects area. These inclusions, if Scrum projects are run at CM L2 by its project processes and the CM L3 risk management processes, would sufficiently qualify Scrum for inclusion into Table 2.4. Regardless, the following discussion and analysis of agile methods may help to shed more light on this subject and place the different project methodologies into context.

2.14.3 Agile

In February 2001, at the Snowbird Ski Resort in Utah, 17 influential members of the software development society met to formulate the Manifesto for Agile Software Development (Sutherland, Shwaber, Highsmith, Cockburn, *et al.,* 2001). In summary, the manifesto stated that it valued individuals and interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation, and responding to change over following a plan. In essence, it was agreed that while there is value in the items on the right, they valued the items on the left more.

The following principles of agile were also identified at Snowbird:

1. Customer satisfaction by early and continuous delivery of valuable software,
2. Welcome changing requirements, even in late development,
3. Working software is delivered frequently (weeks rather than months),
4. Close, daily cooperation between businesspeople and developers,
5. Projects are built around motivated individuals, who should be trusted,
6. Face-to-face conversation is the best form of communication (co-location),
7. Working software is the principal measure of progress,
8. Continuous attention to technical excellence and good design,
9. Simplicity - the art of maximizing the amount of work not done - is essential,
10. Best architectures, requirements, and designs emerge from self-organizing teams, and
11. Regularly, the team needs to reflect on how to become more effective and adjust accordingly.

The Project Management Institute also has an agile certification called the Agile Certified Practitioner (PMI-ACP). In addition, the PMBOK® 6 Guide also includes a detailed section dedicated to the use of what it refers to as Iterative and Adaptive Environments.

Figure 2.13 below gives an overview of the typical agile environments and shows their relationship with Kanban and Lean.

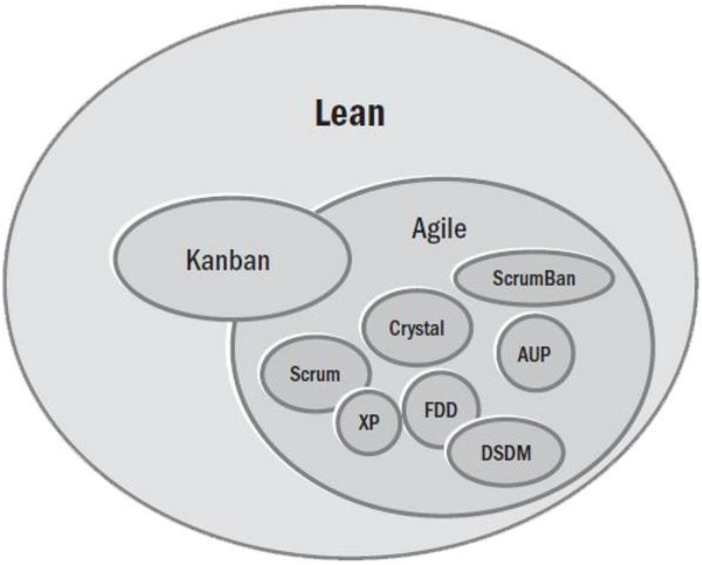


Figure 2.14: Agile Environments

Source: Project Management Institute (2017:819)

Figure 2.13 places the various agile approaches within an agile grouping. Kanban, of which ScrumBan is a blend of Kanban and Scrum, is a Lean approach that aims to visualise all work undertaken within a Kanban board. Kanban was developed by Taiichi Ohno for Toyota Automotive in Japan as a simple planning system whose aim was optimal control of managed work and inventory at every stage of production. Lean is a production method derived from Toyota's 1930 operating model, "The Toyota Way." Lean aims to maximize customer value while minimizing waste. Kanban does not follow a plan but aims to complete prioritized work before pulling additional work from a backlog. According to the Agile Certified Practitioner certification, Kanban aims to limit work in progress and speed up the release of features across the production process. In this way, capacity is created, compliant with the findings of the Goldratt & Cox, 2004 theory of constraints.

Considering Annexure A, agile environments are seen to exist on a horizontal continuum of more or less agility, moving away from traditional project constraints. Figure 2.14 considers this as a move away from a predictive environment in terms of the dimensions of Frequency of Delivery and Degree of Change.

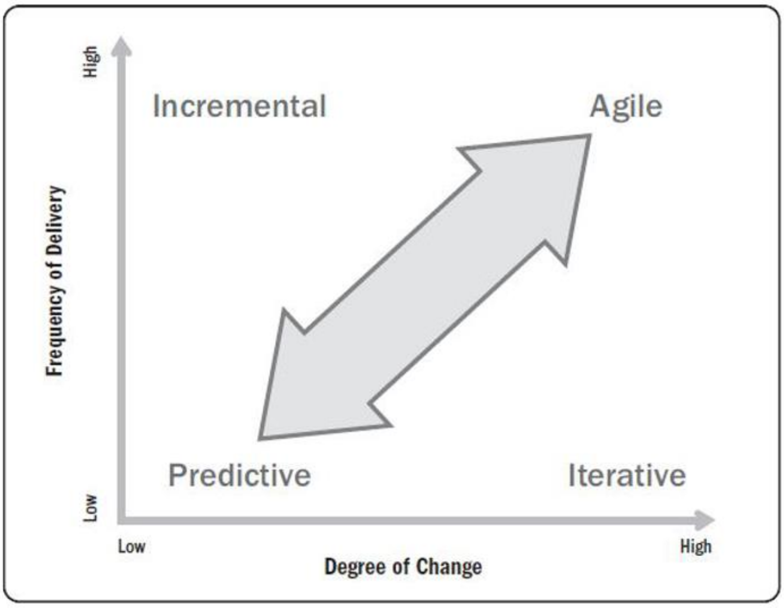


Figure 2.15: Degrees and Frequency of Change

Source: Project Management Institute (2017:827)

Figure 2.14 shows that Agile Project Management operates iteratively as opposed to Traditional Project Management which, based on long-range planning, required a more predictive environment. Delivery in Agile needs to be incremental, involving a product that offers more features over time. In Traditional Project Management long-range planning would focus on a single predicted requirement released at the end of the project or phase.

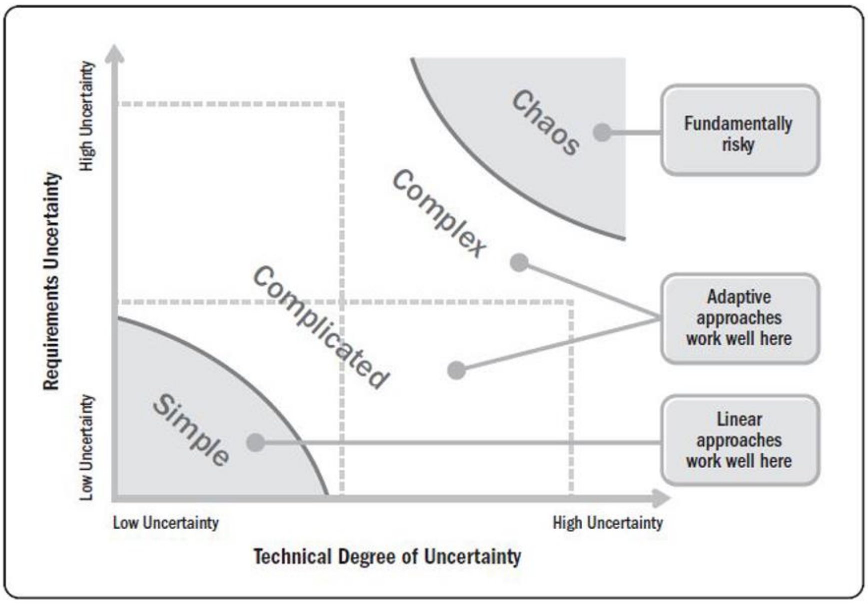


Figure 2.16: Degrees of Uncertainty and Risk

Source: Project Management Institute (2017:822)

Figure 2.15 shows that higher levels of risk are typically associated with higher uncertainty. When requirements are simple or predictive, then linear approaches, understood as planned approaches, such as Traditional Project Management, ought to work best. However, as uncertainty increases, this will bring with it an inability to undertake long-range planning and consequently higher levels of risk.

Based on the previous discussion, Annexure A illustrates graphically why methodology and agility together are seen as a horizontal dimension with traditional projects on or towards the left and projects with more agility are situated more towards the right on the left to right continuum. Risk, as shown in Annexure A, and along the horizontal dimension, is understood as having a direct relationship with a lack of plan detail. The assumption is that projects that operate towards the left have access to more detailed requirements, and therefore if a plan is used, these projects should have a better understanding of the way forward, which should improve project quality. Owing to this perception, especially if projects are complex, projects on the left of the continuum should be run with more control, and therefore less risk, as opposed to projects on the right, as Figure 2.16 illustrates.

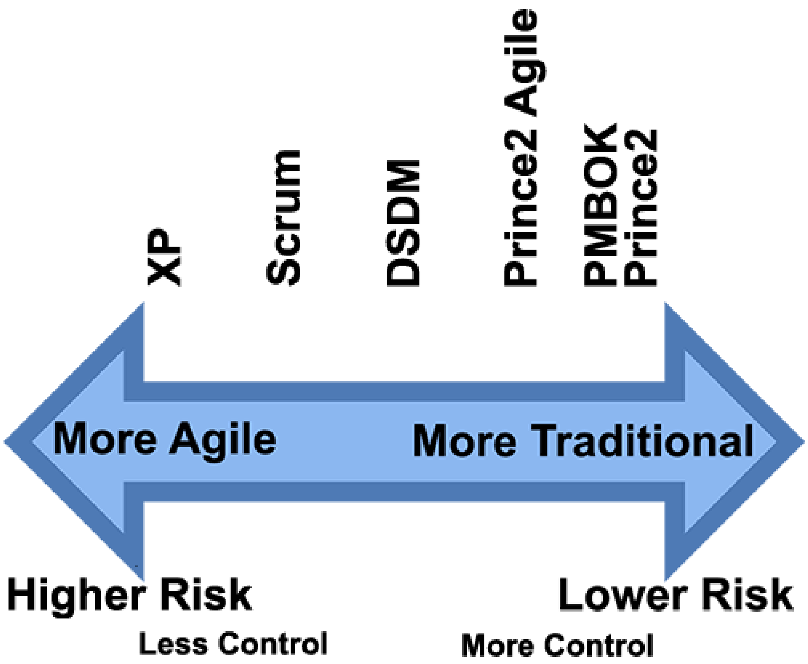


Figure 2.17: Traditional Agile Continuum

Source: Synthesized by the researcher

Figure 2.16 shows that with more agility should come less control and higher risk. With traditional project methods, which involve more linear approaches, there should be an increase in control, leading to reduced risk. On the other hand, if agile is run well according to its method and processes at CM L2+, at each iteration the development team could introduce better versions of need satisfying software, limiting risk.

In the scrum method, a principle called empirical process control is used to ensure that the agile team capitalises on observation of actual outputs, rather than on reliance on planning. If each agile product delivered in an iteration is carefully analysed, failings could be rapidly improved upon with the goal to produce quality working software by quickly adjusting the developmental approach. In this regard, empirical process control utilises the concept of failing fast in order to overcome problems, reduce risk, and move rapidly towards tested solutions that have value. The researcher suggests that the PMIS could be a way to better install and run agile processes for tighter monitoring and control. The PMIS would be a good way to leverage the benefits of empirical process control. This is due to the tight coordination of the software development team working off the developed schedule, supported by the testers and business stakeholders and users. Tasks on the PMIS can be set up to require their approval to fully sign off that acceptable value has been delivered. With EVM incorporated into the PMIS emplacement, a project is transparent, and project expenditure that does not rapidly produce value could be managed more tightly or the project terminated.

Agile is not without criticism. In a recent talk at the Goto 2015 conference, Thomas (2015), one of the creators of the Agile Manifesto, rejected the commercialization of Agile. His reason was that Agile, with a capital letter A, now often refers to a brand name, with originators of new types of Agile hoping to sell more of their product. Instead, agile should be used with a lowercase letter and be seen as an adjective. This could facilitate a return to what was originally intended by the Agile Manifesto. Barry Boehm (2002) told us to get ready for agile methods with care. Steven Rakitin (2001) voiced his concern that in his experience, the items on the right of the Agile Manifesto are essential and by following those on the left, the Agile Manifesto served only as easy excuses not to do the difficult work and with “hacker interpretations” there would be no understanding of, or adherence to, generally accepted and essential engineering disciplines. According to Rakitin the essence of following the items on the left means “Great! Now I have a reason to avoid planning and to just code up whatever comes next". Schach (2007:61) believes that agile will be unable to handle complex software projects. He is quoted to have said that anyone can successfully hammer together a few planks to build a doghouse, but it would be foolhardy to build a three-bedroom home without detailed plans.

A possible yardstick that may be available to measure agile methods could be the concepts of Big Documentation Up Front (BDUF), Enough Documentation Up Front (EDUF), and No Documentation Up Front (NDUF).

With Traditional Project Management it is understood that to be to the left of the horizontal continuum of Annexure A utilises detailed documentation and design upfront, agile with Scrum and Extreme programming (XP) to the right of the continuum, use limited documentation, preferring to build smaller solutions across multiple iterations. Iterative development focuses on the agile principle of empiricism where the Scrum process (#16) Demonstrate and Validate Sprint (Annexure O, Table O.1) means obtaining signoff from a Product Owner at the end of each sprint that they validate what the team has demonstrated.

The Dynamic Systems Development Method (DSDM) towards the middle of the horizontal continuum in Annexure A, being a compromise between Traditional and Agile, utilises as a part of its method what could be considered by some as the nebulous concept of just enough detail.

A challenge with the traditional approach is that projects can be delayed as teams try to create the big design. In this regard, the territory is often unclear until the project is underway. On the other hand, the challenge with agile is that if the required detail for a complex project is not available, then time expended on the "big design" can cause delays, end up being unproductive, and can also introduce tunnel vision and risk. Concerning risk, a project budget commitment established upfront, without fully understanding the detail, which is pre-allocated, could be inflexible due to a rigid managerial mindset if additional budget is required. According to Clinning & Marnewick (2017:11) if an economy and businesses within it have scarce resources, the budget to sustain project development may not be possible or available.

Cunha, Moura & Vasconcellos, (2016) found that often approaches for project management are not sufficient for a flexible and uncertain product, such as software. They recognise that project success could depend on how software project managers deal with the problems they face and how they make decisions. A problem with Agile, or possibly a benefit, depending on the viewpoint, was that the Project Management role was reduced to Servant Leader and often agile teams, who insisted that the team decide the way forward, could not make crucial decisions within this diminished leadership vacuum. Mcavoy (2009), in a longitudinal study, found that empowered, cohesive teams can exhibit problems, such as groupthink or the Abilene Paradox, where a group of people collectively decide on a course of action that is counter to the preferences of many or all of the individuals in the group.

The Wysocki (2014:59) project type selection matrix in Figure 2.17 advocates upfront selection of the best method for the type of project being run as an essential ingredient for project success.

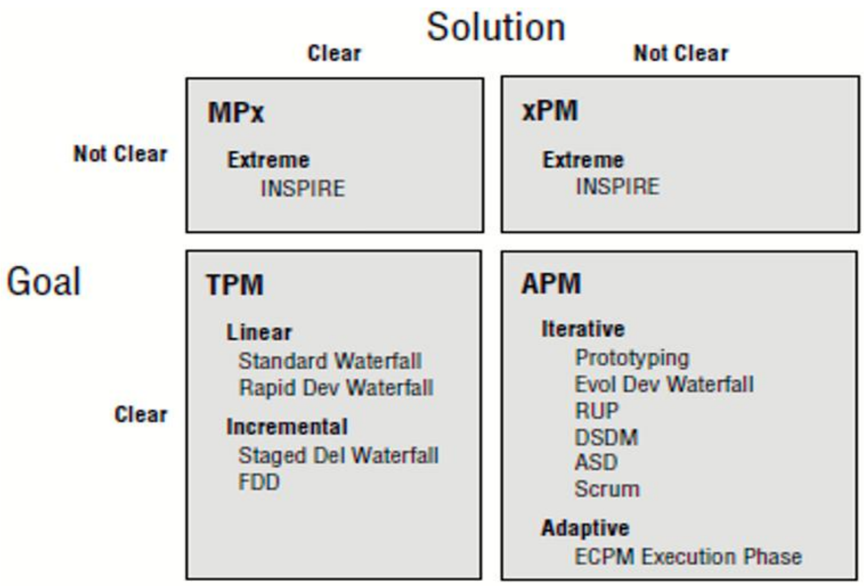


Figure 2.18: The Project Type Selection Matrix

Source: Wysocki (2019:424)

Figure 2.17 shows that if requirements and scope are unclear or undefined, any project type falling into the quadrants other than Traditional Project Management (TPM) should, at an essential level, be out of control and at risk. To run these types of projects, the project manager or agile team may need to rely heavily on tight risk and issue management as they navigate forward. According to Wysocki, this is due primarily to the classic constraints and controls no longer being actively applied, as shown in Figure 2.18.

Figure 2.19: Scope Triangle

Source: Wysocki (2019:40)

Figure 2.18, Wysocki's Scope Triangle, is different from the traditional Iron Triangle of Project Management. The Iron Triangle approaches time, cost, and scope as points on the triangle, which are quantifiable, based on the Traditional Project Management Plan. The Scope Triangle, on the other hand, focuses on time, cost, and resource availability as lengths of the sides of the triangle, which are continuously exposed to risk, primarily due to a lack of time spent planning. According to Wysocki, time spent on planning in the top left, top right and bottom right quadrants of Figure 2.17 above would be counter-productive and fruitless. These quadrants and the different agile methods rely on not knowing the future, but through agility could inspect and adapt, opting rather to capitalise rapidly on opportunities presented at the end of each short timeboxed iteration, or in the case of the Scrum method - sprint.

Boehm & Turner (2002:2) summarised and rationalised the shared value propositions embodied in the Manifesto for agile Software Development as follows: "responding to change over following a plan is roughly great! Now I have a reason to avoid planning and to just code up whatever comes next".

Jakobsen, Ruseng, Sutherland, (2009) commented that using the CM Model and Scrum together can facilitate improved performance while at the same time maintain CM Model compliance. Boehm & Turner, (2004:2) conceded that there were viable mappings between CM Model specific goals and agile practices, and there was a benefit in assessing agile software development using CM Model, which could bring about agile-based improvement efforts.

Pikkarainen & Mantyneimi (2006) found that agile as a methodology is dependent on the ability of team members to fully understand and appreciate their environment and organisational capabilities. With this in place, they could more easily identify and closely collaborate with the project stakeholders. Risk analysis in this context is used to define and address a set of defined risks that are associated with agile and plan-driven methods.

Conversely, Hansen & Baggesen, (2009) observed that CM Model process focus and especially the need to achieve certification at set CMMi levels isolated the teams from each other and could create disruption and distrust. They concede that a move to agile and the use of Scrum did engage people in much more collaboration and visualisation of many hidden problems, which allowed them to deal with these more rapidly. Hansen's experience was that teams that worked the problems and focused on improvement via the regular use of retrospectives were able to create more trust and understanding between each other. These activities could change the focus from instead of having the processes as a goal to having the process as a tool to deliver maximum business value, with high productivity and quality output as a goal.

Stocks (2013) believed that as agile approaches are becoming more widespread, the majority of agile projects were smaller and more focused and applied to relatively simple solutions with fewer integration points and co-located teams. Accordingly, the challenge for larger, more complex projects would be to apply agile thinking and work practices more successfully to enterprise-wide solutions with large budgets where more stringent investment approval mechanisms and governance requirements existed. Stocks added that additional controls, such as oversight of project execution, would need to form part of the governance requirements. The researcher notes that the PMIS emplacement set up correctly by following the CM Improvement Framework in Chapter 4 could be used to facilitate, deliver, and assist to control these requirements.

2.14.4 Agile, a silver bullet at Capability Maturity Level 1?

The researcher observes from his experience that agile is often offered as a solution to replace the inability of Traditional Project Management to deliver solutions. However, agile undertaken at CM L1, without adequate attention to the rigorous processes that comprise agile methods, could be regarded as a silver bullet. To this end, without the agile methodology and processes being tightly managed, this may remove essential controls required to run a project successfully. The undone agile or scrum team (Annexure O, Figure O.3) that fails to deliver value sprint after sprint must surely introduce risk, rework, and waste.

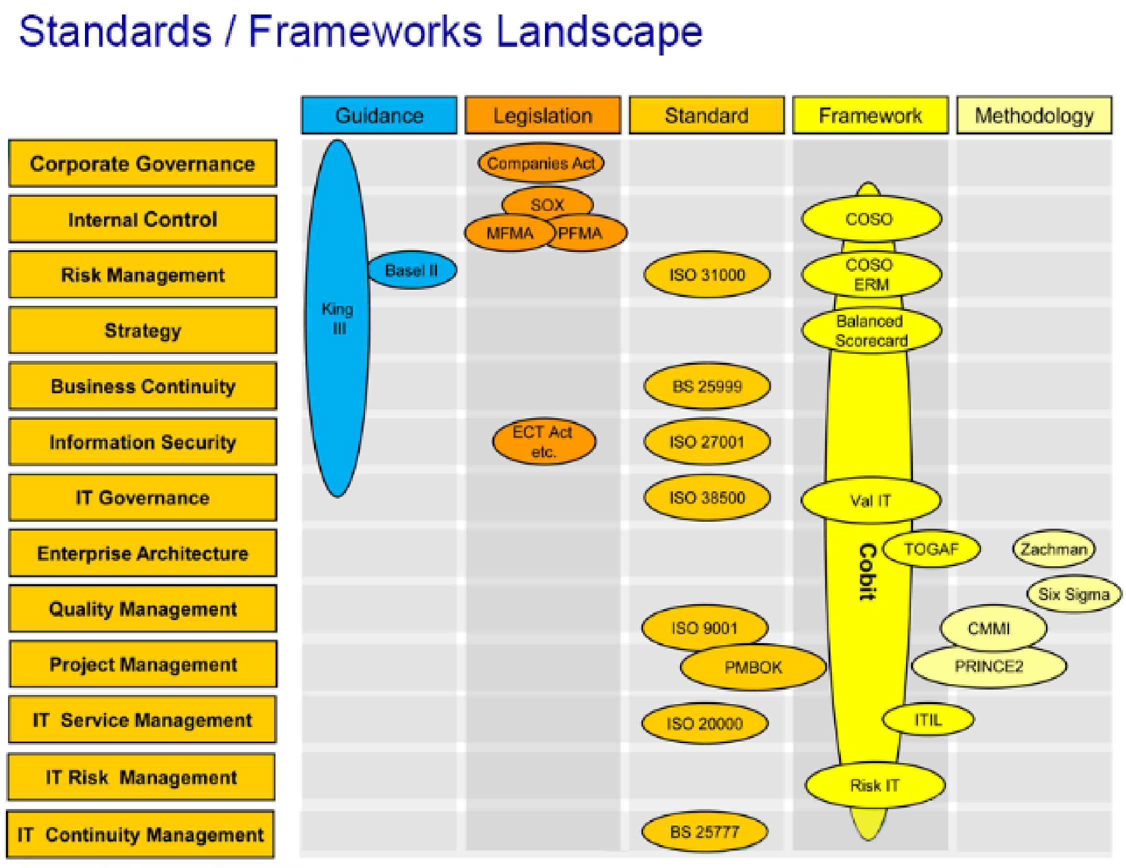
With Traditional Project Methodology appearing to be criticized by the agile community, the researcher has personally witnessed very expensive failures of agile applied within the Public Sector, and other sectors, from audit reports written after the fact. From the researcher's perspective, the inability to produce value was primarily linked to the incorrect use or lack of attention to detail when using agile processes. Ultimately, it may be that agile is best suited to smaller software projects (or modules) rather than complex programming systems products, as discussed in Figure 2.1 at the beginning of this section.

Sohi, Hertogh, Bosch-Rekveldt, *et al.*, (2016) found that lean and agile significantly correlate to either reducing complexity or managing complexity. Dikert, Paasivaara & Lassenius, 2016 researched large scale agile transformations and found the most salient success factor categories to be management support, choosing and customising the agile model, training and coaching, and mindset and alignment. Lowry & Wilson, (2016) looked at the role of IT in creating agile organisations and found that modern organisations were becoming increasingly dependent on IT to remain agile and competitive in a rapidly changing market. However, there remained gaps in understanding on exactly how IT resources support IT agility. The researcher suggests that a PMIS emplacement used correctly according to the PMIS CM Implementation Framework presented in Chapter 4 could focus and reinforce CMMi software engineering process application for best practice for any project method or type, be it traditional or agile. This could address the gaps between IT and the business and may facilitate how agile could better handle software project complexity.

2.15 Standards / Frameworks Landscape

Many other methodologies, frameworks and standards exist. Table 2.4 gives an overview of some of the more well-known of these and the way they interrelate.

Table 2.4: Standards Frameworks Landscape



Source: State Information Technology Agency EPMO 2016

Table 2.4 offers a means by which business can orientate itself about the standards and frameworks landscape. At the top of the Table are arrayed the headings for Guidance Mechanisms, Legislation, Standards, Frameworks, and Methodology. A range of focus areas is listed down the left-hand side. An example of how the table is used is to associate Project Management with the PMBOK as a Standard while COBIT and ITIL are Frameworks that could be considered, and PRINCE2 is a possible Methodology that can be applied. King III, from a Guidance perspective, could be considered for Corporate Governance, Internal Controls, Risk Management, Strategy, Business Continuity, and IT Governance.

Table 2.4 may not fully comprehensive and does not include agile, however the researcher notes that a degree of fusion and overlap is apparent between the processes found in the CMMi and those found elsewhere in many of the standards, frameworks and methodologies mentioned. It also appears that over time, many of these are moving towards a common approach, which is mirrored in the ranking of the processes found in the CMMi schedule of processes in Annexure D.

The researcher observes that over time some methodologies will be promoted for a while as being the best of the breed, and these may soon be replaced by others that purport to be better. If a project methodology is used by an organisation that is operating at a CM level below what is required for tight management of the methodology and its processes and the method underperforms, this may cause a sound methodology to be unfairly dismissed. The PMBOK 5 appears to have been criticised during the arrival of several new agile methodologies as having limited agility. In response to this, in a relatively short time, the PMBOK 6 rallied and the latest manual of 900+ pages now fully incorporate much of the agile mindset. In addition, the PMI has introduced an official Agile Certified Practitioner exam, which focuses fully on the agile approach. When this example is considered, the researcher contends that it may not be as much about the methodology but rather about the ability to operate the methodology correctly, applying its processes correctly at CM L2.

If CM L1 means that essential processes are not applied or inadequately applied, then this could point to the source of the problem rather than incorrectly attributing the problem to the methodology itself. The correct installation and use of the PMIS emplacement, according to the PMIS CM framework provided in Chapter 4, could help to refocus effort back onto the project task being used correctly within a transparent and accessible developed schedule #6.5, irrespective of the project methodology. This should enable the use of project metrics to correctly point to the reasons for project failure to correct and improve, rather than looking elsewhere for the next silver bullet.

ITIL version 3 and 4, DevOps and COBIT 5 will now be briefly addressed in sections 2.15.1 to 2.15.3 to further the discussion.

2.15.1 ITIL version 3 & 4

The IT Infrastructure Library (ITIL) is a set of processes and best practices for IT Service Management (ITSM). ITIL aims to align IT services to best satisfy the needs of the organisation. ITIL version 3 is currently published as a set of five volumes, each covering a stage in the IT Service Management lifecycle. Figure 2.19 shows the 5-stage ITIL Lifecycle that includes Service Strategy, Service Design, Service Transition, Service Operation, and Continual Service Improvement. ITIL 4, released at the end of

2020, has published a foundation manual along with four supporting manuals entitled ITIL 4 Create, Deliver and Support, ITIL 4 Direct, Plan and Improve, ITIL 4 Drive Stakeholder Value, and ITIL 4 High-velocity IT. ITIL 4 focuses directly on pinning down value streams and then managing these for optimisation within the service value system (SVS). The following discussion will focus only on ITIL version 3 which is summarised in an overview in Figure 2.19 below.

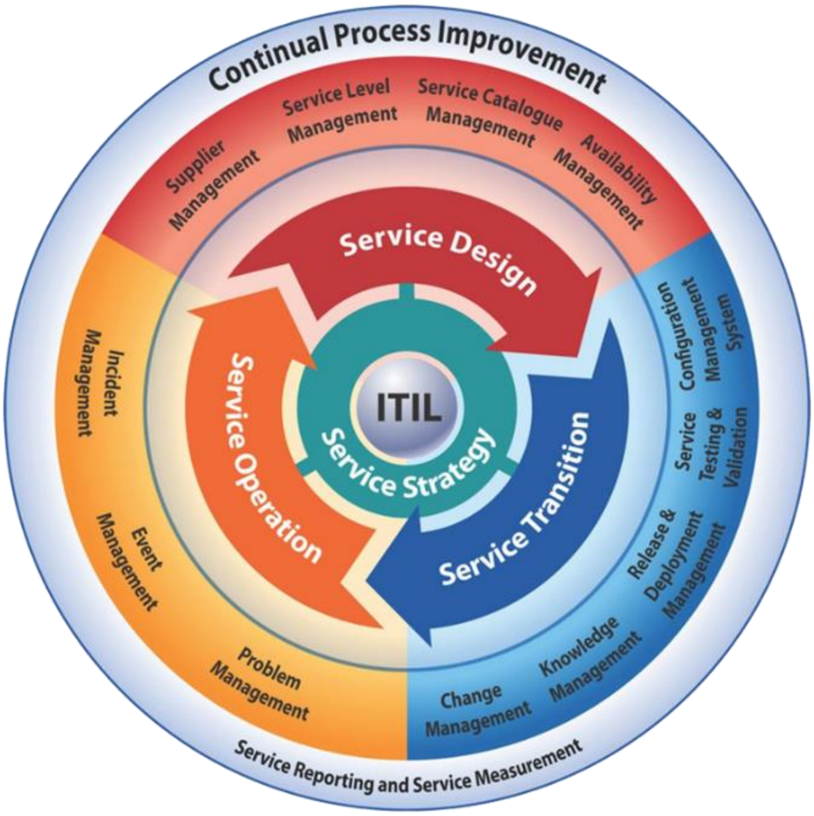


Figure 2.20: ITIL Lifecycle

Source: Krishna Kaiser (2018:50)

Figure 2.19 explains the five Service Lifecycle phases, including the Service Management processes within them and their interaction with each other. ITIL comprises 26 processes, incorporating much of ISO/IEC 20000. This standard was previously known as BS 15000. Since July 2013, ITIL has resided under the auspices of AXELOS, which is a joint venture between Capita and the UK Cabinet Office. ITIL is of crucial importance to software project management because the ability to manage value-adding software solutions throughout its lifecycle is essential for business success.

Table E.1 in Annexure E has been synthesised by the researcher to illustrate how the Continued Service Improvement Register of ITIL can be understood to comprise “pain points” or “improvement areas” within IT systems in the organisation. Annexure E illustrates how these are then absorbed by ITIL Service Strategy for prioritisation via the Service Catalogue. In turn, Service Strategy, if the change is complex, may require that a project be created to house and manage the change initiative. Thereafter, if the business case is approved and following project management principles, the change moves through Service Design and Service Transition and finally ends up in Service Operation, where it is consumed as a value-adding service.

Continual Service Improvement (CSI) is, as the name implies, ongoing and cyclical, focusing on systems improvement from one stable system baseline to the next. It is of interest that the CSI Register, which feeds into the Service Catalogue, is similar in many ways to the Agile or Scrum Product Backlog. Annexure O has more detail on the Scrum method, discussing, in particular, the Product Backlog and how it is a primary driver for the success of the Scrum method. In this regard, ITIL can incorporate the agility of Scrum, with both deriving benefits from each other towards the continual improvement of value streams.

2.15.2 DevOps

DevOps is the tighter collaboration of operations engineers and developers working together across the entire service lifecycle, from system strategy to systems design, to the transition of systems to operations. Continual Service Improvement, as Figure 2.20 shows, involves devising strategies and projects to address improvement initiatives.

Figure 2.20 shows how DevOps and the ITIL Lifecycle Stages fit together.

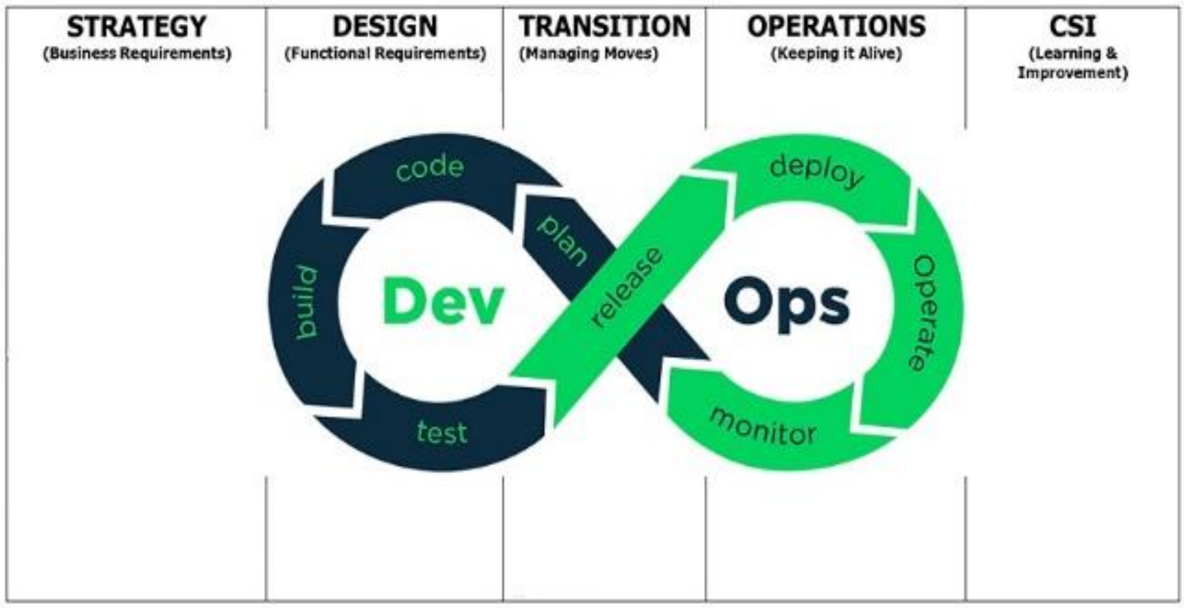


Figure 2.21: DevOps Overlaying ITIL Lifecycle Phases

Source: Synthesized by the researcher

Figure 2.20 superimposes DevOps on top of the ITIL lifecycle phases. The detailed processes that are found in ITIL (found in Table E.1 in Annexure E) are not listed above, but it is understood that they will be used correctly and with agility. Essentially, DevOps aims to create tighter connections between software development and IT operations to serve business needs better. The figure above explains that from a DevOps perspective, this will include planning, coding, building, testing releasing, deployment, operations, and monitoring. The fact that Dev and Ops are situated within an infinity symbol reinforces the Continual Service Improvement of ITIL.

The introduction to this chapter discussed the depth or spiralling dimension found in Annexure A. The Continual Service Improvement phase of ITIL 3 is similar to level 5 of the CMMi model. It intends to ensure that improvements are introduced, which are then stabilised at a higher level of quality than that which existed before.

There are numerous techniques available in DevOps, ranging from the use of source control, testing methods, encapsulation of code for release via automated technologies, like Docker (an encapsulation engine that bundles code to facilitate rapid release of working software), and others. According to What is DevOps (2019), the approach does not differentiate between different disciplines. “Ops” can refer to all types of staff, including systems administrators and engineers, operations staff, release engineers, DBAs, network engineers, security professionals, and many other titles. “Dev”, as opposed to “Ops”, refers to developers, but in practice means all the people involved in developing, testing, and releasing new software.

DevOps is tightly connected with the approaches of Agile and Lean. A misconception appears to exist with the view that “Dev” were the “makers” and “Ops” were the staff who handle “operations” and the perception that these two were treated as silos. Therefore, DevOps is understood as growing out of an agile mindset. Gene Kim, the author of DevOps Handbook, says that DevOps being in opposition to ITIL is a misnomer. “Even releasing 10,000+ deployments/day requires processes”. Kim’s example explains that solving software engineering complexity is at the heart of DevOps, and as ITIL is process-based, with some 34 processes (described as practices) in ITIL, the assumption could be that these processes should be practised if DevOps excellence is the goal.

According to Kaiser (2018:294), the activity that typically works to undermine DevOps objectives for speed and agility will always be the approval from business. Kaiser believes that Product Owners, a scrum project method role, could be re-aligned to become the new release managers. If the release management team may have been made redundant by faster automation via DevOps practices, release management still needs an owner for the entire release management process. This can be a single Product Owner and Release Manager role that cuts across both development and operations. Product Owners are an adequate choice, mainly because of their closeness to the business and the development and operations teams. However, the researcher’s experience has found that often the Product Owner will confuse their Scrum role of user story creation and prioritisation for business, who must sign off the demonstration that these are working at the end of a Scrum sprint, to a new role of development manager. In Scrum, the Scrum Master and Developers decide what user stories they will focus on, which are then pulled into development by the Developers, not enforced upon them by the Product Owner.

Table E.1 in Annexure E explains how the Continual Service Improvement Register naturally feeds into ITIL Service Strategy to become the prioritised list of business requirements for focus in the next version of software. This prioritised list in the Scrum Project Method is called the Product Backlog and is ultimately the responsibility of the Product Owner, as discussed in Annexure O. However, from a CMMi Project Planning (PP) and Project Monitoring and Control (PMC) perspective, having business via the

Product Owner force delivery may not be the best approach as quality and stability of next versions of software are the goals.

Therefore, awareness of the PMBOK, Agile, Scrum, CMMi, and ITIL processes and how best to apply these cohesively could better facilitate DevOps' success. Attempting to install DevOps off a base that is operating ineffectually at CM L1 may be unsuccessful, owing to the lack of awareness or disregard of many interacting and facilitating principles, practices, and processes that are required to work in unison for overall success.

2.15.3 COBIT 5

COBIT has been created and is maintained by ISACA for IT management and IT governance. COBIT provides a series of implementation steps and controls over IT, forming a framework of IT-related processes and enablers.

COBIT (Control Objectives for Information and Related Technologies) is a best-practice framework which can supply IT controls that range across Corporate Governance, Internal Controls, Risk Management, Strategy, Business Continuity, Information Security, IT Governance, and other areas.

COBIT was first released in 1996 as a series of control objectives to assist with financial audits relating to IT. COBIT incorporates ISO/IEC 38500 and is used to address IT-related business processes and responsibilities in value creation (Val IT) and risk management (Risk IT).

Figure 2.21 shows the seven steps that comprise the COBIT lifecycle.

Figure 2.22: COBIT Lifecycle

Source: ICASA (2011:37)

The figure above shows how COBIT is used to bring about a controlled and manageable change in strategic direction in Information Technology. The figure asks seven questions which are then answered through a program of work, as illustrated by the COBIT model. The Deming Plan Do Check and Act, Section 1.7, operates similarly with the assumption that the achievement of success without a clear plan will be difficult. As COBIT is a complex method, more detail can be found in Annexure I.

The researcher appreciates that COBIT will be a challenge to implement. In addition, as soon as governance and controls are introduced through any method or framework, this will introduce complexity and slow down agility. The use of the PMIS could assist with the implementation of COBIT as many of the controls that need to be implemented can be facilitated by the system.

Section 2.16 focuses specifically on the PMIS. It offers more insight into how it could work and the benefits that can be derived.

2.16 The Project Management Information System

It is appreciated that there are several PMIS in the market. This discussion focuses on the general use of the PMIS to facilitate the successful delivery of projects with agility and good governance.

In a case study conducted by the Project Management Institute (Telecom & Technologies 2006), it was found that the company Du Telecom credited the success of a 32% growth rate in its first four years to making a PMIS available to their project management teams. The PMIS emplacement enables several key features that can boost productivity while facilitating higher levels of project coordination and control for project teams. These include improvements in project scheduling, estimating, collaborative work via portals and dashboards, the ability to better manage resources and tighter management, and control of project documents and data. According to (Kashyap 2019), the improvements offered by a PMIS can be focused on three key areas of benefit, which include the dimensions of visibility, accountability, and organisation.

The visibility dimension facilitates an ability to monitor and control progress across numerous projects, identify projects at risk, monitor timelines, and the capability to share project status in real-time. According to Braglia & Frosolini, (2014:1) the goal of utilising a PMIS ought to increase efficiency by making the development cycle more visible. Bočková, Sláviková & Gabrhel, 2015:715 explain that game theory can be used to position each project as a game to “maximise gains and minimise losses,” focusing the team towards the achievement of success.

The accountability dimension allows project updates to be available to team members and stakeholders in real-time. This facilitates verification and validation against actual data, which promotes a better perception and ability to respond to a project’s status. Ultimately, the PMIS enables higher confidence for the project team and stakeholders, which in turn promotes improved levels of project accountability. These improvements offer ease-of-access via summary views of the project to avoid overdue tasks and/or missing deadlines, and ultimately, the reduction of confusion.

The organisational dimension offers an ability to keep project workflows centralised in a single system. Not only does it facilitate project control from a management perspective, but also ease-of-access by team members to manage project detail and to maintain project updates. The benefit to the organisation is that all team members and stakeholders can buy into the project method, which processes are centralised, understood, accepted, consistently applied, and credible.

Nguyen & Nguyen, 2016, in a study on the ability of the PMIS to assist the project team to achieve success, found that user satisfaction while working on a system was influenced by the user ease-of-use, system quality, functional and informational quality, and support and service quality.

Meredith & Mantel (2009:462) discuss how a PMIS is beneficial where project complexity is a challenge. They recommend choosing an application that offers "functions of friendliness." User Experience (UX), in this regard, could include calendars, budgets, graphics, charts, reports, and so forth. Ultimately, the PMIS can greatly assist practitioners to detect latent issues before they occur, achieve project milestones, and collaborate more easily and to a greater extent. It would appear that with the PMIS emplacement, challenges faced with difficult projects can be easier to overcome, making the project goals more achievable.

A PMIS emplacement can also collect and distribute information and report on key performance indicators (KPIs) by utilizing variances on the baseline plan against actuals. The PMIS emplacement also offers summations of complex project status via dashboards using many different visual aids, e.g. Gantt charts, project progress graphs, and task status traffic lights, also called RAG (Red Amber Green) indicators. Figure 2.22 illustrates that the PMIS Dashboard can build up from the project tasks and consolidate across PMBOK Knowledge Areas.

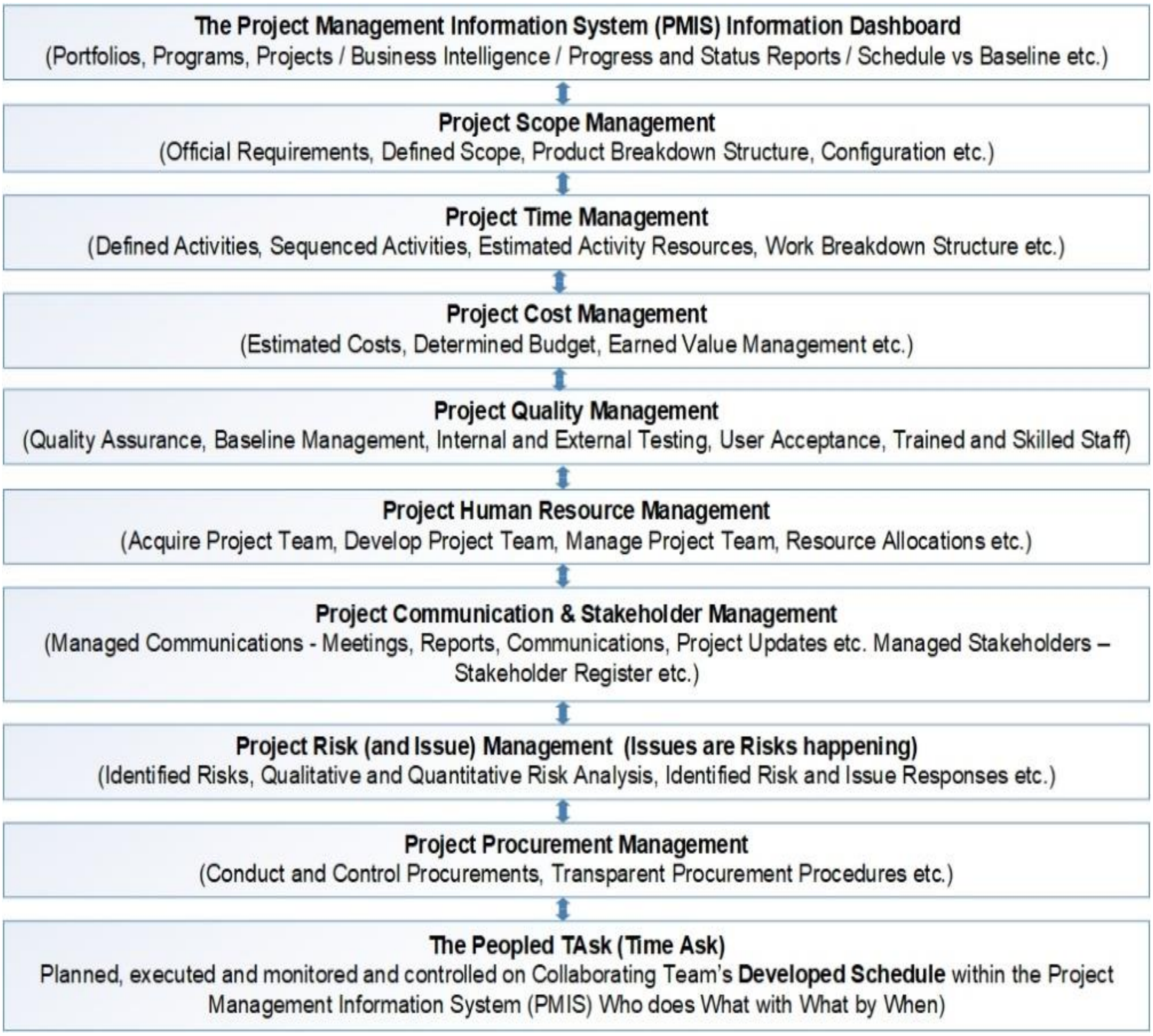


Figure 2.23: A PMIS Information Stack

Source: Synthesized by the researcher

Figure 2.22 shows that the peopled task on a PMIS is an essential input. This task, called a time ask, is a period agreed upon by a resource on the PMIS to deliver value. The task, when placed within context, can be used by the PMIS to populate the PMBOK Knowledge areas. Feeding relevant data into the Knowledge Areas in the PMIS stack means it can be converted into information, knowledge, and wisdom. An example of this could be two hours that are set aside to test newly developed software. This task is agreed upon by a member of the project team and has an impact on project Scope, Time and Cost, and also the Human Resources Knowledge Area. If the task is completed on time with no problems, then the fact is communicated to team members and stakeholders via the PMIS Dashboard. If a problem is found, this can feed the Risk Management Knowledge Area.

Caniëls & Bakens, (2012:9) expected that a project manager's satisfaction with a PMIS would be indirectly related to the quality of decision making. However, in their study, they found "a positive effect between the two". About the potential for PMIS information quality and the potential for information overload, they found that "project overload, as well as information overload, are positively, albeit weakly, related to PMIS information quality".

Ultimately, team visibility via a PMIS off the developed schedule (process #6.5 baselined and moved into Execution) implies that all share a jointly agreed playing field where teams and individuals can accept the challenge and deliver results. Successes against the project constraints set up in the developed schedule of the PMIS means that individuals and teams can prove their worth. Because the peopled tasks are visible on the PMIS, it is possible also to provide additional motivation in the form of financial rewards. Sustaining successes can be used to create role models and fame, which in turn can be used to further motivate the current team and other teams and promote knowledge sharing.

2.16.1 The PMIS and Resource Management

A key benefit of a PMIS is its ability to support better resource allocation. According to Tromp & Homan, (2015:213) "the main problem in a multi-project environment is the lack of dedicated teams on one project". Employees allocated over many projects and whose effort and focus are demanded simultaneously on these projects may well lead to negative consequences for the projects. The Theory of Constraints explains how to isolate the most important limiting factor (or constraint) that stands in the way of achieving a goal. After identifying the constraint, the idea is to improve it until it is no longer the limiting factor. According to Goldratt (1997), challenges common to all projects are bad multitasking, students syndrome, Parkinson's Law, and dependencies between steps. These can all result in budget and time overruns, ultimately compromising the ICT value proposition. Points 2, 6 and 12 from the PMIS CM Improvement Framework in Section 4.3 Table 4.1 deals specifically with resource allocation (or resource loading) onto the developed schedule.

The fact that tasks cannot be created simultaneously on different project schedules in a PMIS emplacement means that resource allocation challenges are remedied. This, the researcher notes, may not sit well with management who run projects at CM L1 where achievement and success are driven through exhortations to achieve unrealistic targets (Annexure G – point 10), which are not linked necessarily to the estimates on the project plan given by the staff that need to achieve them. Engwall & Jerbrant, (2003) suggest that the resource allocation syndrome of multi-project management is not an issue in itself; “rather it is an expression of many other, more profound, organisational problems of the multi-project setting.” Management insistence on producing more and faster with less in terms of resourcing will simply exacerbate the problem of project non-delivery.

From the Deming perspective in Annexures F and G, which thinking underpins much of the Capability Maturity Model integrated, a project task should be correctly allocated, fully resourced and achievable as far as possible. In turn, the task should also be accepted by the task owner, which means that they agree that it is achievable within the constraints of the current plan, phase, or iteration. This activity should occur before the Preliminary Schedule is converted to a final developed schedule and before the task is baselined and moved into the Execution Process Group, as Directed and Managed Project Work in Progress (PMBOK 6 process #4.3). Essentially, operating this way means the difference between starting work prematurely or immaturely on a Developing or Preliminary Schedule in the Planning Process Group; or proceeding correctly and maturely off a Developed and Baselined Schedule in the Execution Process Group. This mindset is illustrated by the arrows in Annexure S, which can also show how the PMBOK matrix can be further improved. In essence, operating thus a project, via its planning schedule (the Scheduling Knowledge Area of the PMBOK), is thought about, made ready, and correctly planned and aligned first before the team rush into project execution.

Where agile tasks are considered, where the outcome is not firmly planned because agile assumes that too much planning is not efficient or effective, team activity to achieve goals is time-boxed into a next iteration or sprint, which means that activity is still being monitored and controlled.

2.16.2 PMIS empowering team creativity

Borštnar & Pucihar, (2015) conducted a study about the installation of a PMIS into an R&D company. In the study, it was found that creative and innovative R&D staff could not be constrained by strict project rules set by the organisation, such as time plans, documentation, and reporting. Findings supported the need to maintain a balance between the unstructured and structured. Essentially also between the creative processes, which are typically rule-free, with structured processes to monitor and control them. What was found to be a driver for success for the PMIS in the multi-project management environment was that the system allowed employees to support themselves. Borštnar & Pucihar, (2015:20) found that benefits consisted of "processes of work planning and reporting, transparent work evaluation, and open communication". According to Borštnar & Pucihar, (2015:21), this provided a degree of transparency into project activity at all times, accountability and transparency via reporting of work done, time spent, and other relevant data about the project. In a capable and mature environment where skilled resources can control their own tasks, estimates of work involved should be respected. Ultimately, the PMIS emplacement could assist resources as a team striving to achieve the mutually understood and agreed goals. Team visibility and control of the developed schedule as their playing field should stimulate and empower creative solutions and success.

2.16.3 EPMO, PMOs, the PMIS Emplacement and Team Productivity

This dissertation aims to prove that the PMIS emplacement can be used to increase productivity within projects and groups of projects. The PMI (2012:7) found that the number of Project Management Offices (PMO's) was on the increase. A PMO with a PMIS can automate various components of the project management process, including automating the schedule, work authorization, resource management, risks and issues management, and team collaboration. In addition, the PMIS (built on a database or, more generally, a data warehouse) can capture and disseminate information and report on key performance indicators (KPIs), baselined against actuals and many other metrics as telemetry. The first dimension of the PMIS Improvement Framework prescribes that an Enterprise Project Management Office be driven off a PMIS is shown in Table 4.1.

Raymond & Bergeron, (2008:219) conclude that the Project Management Information System makes a significant contribution to project success and should continue to be the object of project management research. The use of qualitative cues, like graphics and charts in PMIS, instead of natural language text may work well in South Africa's multilingual and culturally rich society.

Izmailov, Korneva, Kozhemiakin (2016:97) identify several common questions for all project types that suggest focus areas to ensure projects are finished on time. These include the following: they take longer than planned, there are permanent budget overruns, payments are not received in time, there are too many amendments and alterations, there is too much overtime and all too often resources are not available in time (even if promised), the necessary documents are not available in time (information, specifications, materials, design), there is a constant change of priorities, a lot of effort is spent to achieve the interim results, and superiors are required to increase the number of projects in work.

Goldratt (1997) identifies four factors that appear to cause the negative effects listed above; bad multitasking, student's syndrome, Parkinson's Law, and dependencies between steps. Goldratt (1997:92) acknowledges that the student's syndrome involves delaying work until the last minute and Parkinson's Law is the observation that work will expand to fill the time available. He explains that management needs to "find their weakest link" (or constraint) and "strengthen it." The two ways to strengthen a bottleneck is to "add more capacity" or to "squeeze the maximum from the capacity" that exists. According to Izmailov, Korneva, Kozhemiakin (2016) the Theory of Constraints further attempts to overcome the problems by reducing the number of jobs in the pipeline and prioritizing effort into an immediate window of opportunity, freezing work (and even competing projects) that fall outside this window. Izmailov, Korneva & Kozhemiakin tie managing the project's buffer time to higher levels of productivity. When the project's buffer time consumes more rapidly than the work is performed on the critical chain, the buffer is in the red state, and the project runs the risk of being late. When the buffer and critical chain, moving at the same speed, the buffer is yellow, which is a positive status. When the work is done at a faster pace than the buffer is consumed – the project proceeds, and the buffer is green. "At a glance, the project manager can understand which of his projects are going well, which are in danger, and decide where and when to intervene." Figure 2.23 shows how the PMIS emplacement can compile project buffers for use to manage the critical path as part of the critical chain.

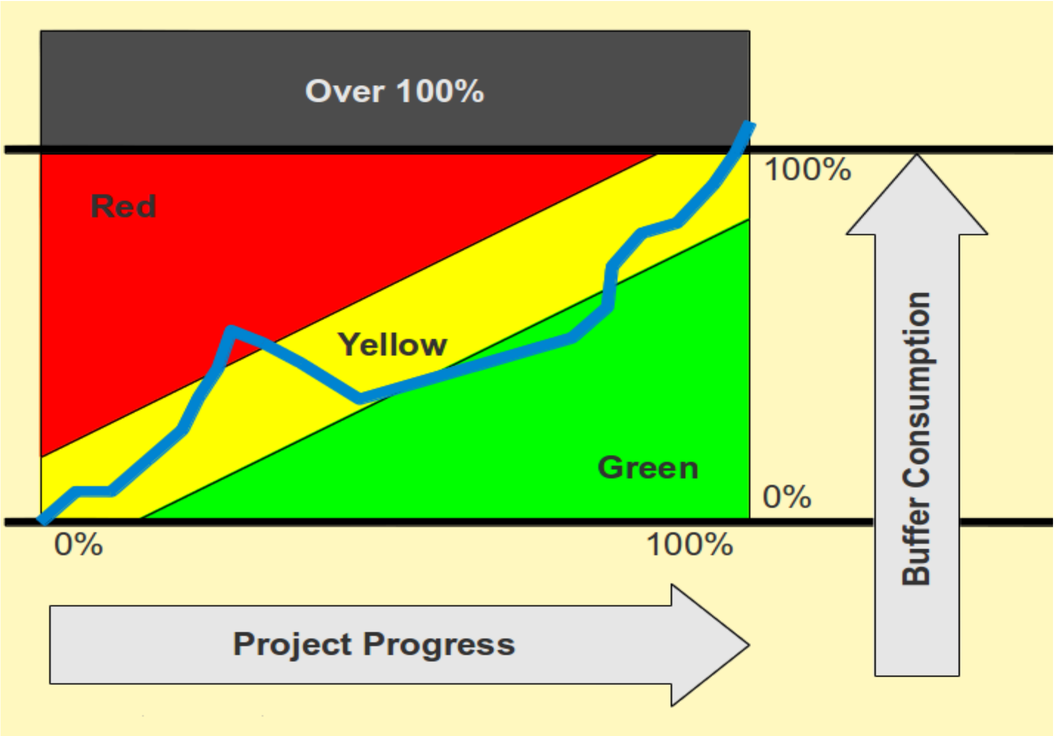


Figure 2.24: Project Buffers and the Critical Path as Critical Chain

Source: Tendon (2012)

Figure 2.23 shows how buffer consumption can be managed on a PMIS. If the project buffer moves into the red, then it is a warning that management is required to return the buffers back downwards into the yellow or green areas. The PMIS can use these graphs to create visibility of the project from a buffer management perspective and allows the team to ensure that projects are run optimally.

According to Walton & Heeks, (2011) a process-based approach to facilitate ICT is well established. Apart from the ability to better manage the core CM L2 processes of Requirements Management (REQM), Project Planning (PP) and Project Monitoring and Control (PMC), the CM L2 process Measurement Analysis (MA) is also derived from the PMIS. The correct installation and use of the PMIS are guided by the PMIS CM Improvement Framework Table 4.1, which means that the CM L2 process Product and Process Quality Assurance (PPQA) is also installed and streamlined over time.

2.16.4 PMIS Emplacement for Governance and Risk Management

Documentation saved in the PMIS addresses many of the Project Quality dimensions for good governance. Kostalova, Tetrevova, Svedik*, (*2015:100) report that the PMIS can contribute in the areas of “requirements management, time management, cost management, risk management, and reporting." This dissertation addresses these areas in points 4, 5 and 11 of the PMIS CM Improvement Framework in Table 4.1. Quality Gates and using the PMIS emplacement for version control in project documents can enable smart approaches to project management. Risk and Issue Management, as part of the Risk Management Knowledge Areas of the PMBOK and also the CMMi process (RSKM), is seamless via the PMIS to team members. Project challenges are avoided, quickly solved, or escalated to the executive, with all capturing and routing enabled via the Risk and Issue management module of a PMIS emplacement.

2.17 The Spiralling Up or Down Dimension

This dimension is focused primarily on continuous improvement. Figure V.1 Annexure V explains the concept. Essentially the depth, spiralling or continual improvement dimension is a third axis Z linked to the horizontal axis X and the vertical axis Y, as explained in Annexure A. The idea is that by focusing on improvements in axes X and Y, via techniques such as Total Quality Management, Continual Service Improvement and PDCA, etc., allow the axis Z to move (soar) upwards to a new level of Productivity and Quality. If the new level is maintained, then further improvements can be considered and those also maintained. All improvements must be driven off Production and Quality statistics which is the essence of CMMi Level 4 and CMMi Level 5. The PMIS emplacement correctly installed per the PMIS CM Improvement Framework in Chapter 4 is a proposed approach to achieve these improvements in order to achieve, maintain and sustain improvement gains.

2.18 Summary

When starting on the CMMi journey, being able to operate at CM L2 means that project processes are understood, applied and their inputs, transformation and outputs are monitored and controlled for delivery and success. This CM L2 mindset and behaviour must be an essential starting point for professional project execution if productivity and quality, and not risk, rework, and waste, are the goals stakeholders are aiming for. Agile, well-intentioned as a panacea, if undertaken at CM L1, could be fallacious. Ultimately, it could be said that any project methodology undertaken at CM L1, without process focus, will be problematic.

Being able to pay careful attention to the management of the traditional or agile project plan before and during project execution, from a Project Planning process (PP) and Project Monitoring and Control process (PMC) perspective, is a vital requirement to move out of the chaos reported on in the Chaos Reports (2014), surely indicating operations are at CM L1. Task management and avoiding the slipping task on a developed schedule in real-time, as illustrated in Figure 1.2 in Chapter 1, could be the best place to start to ensure success at CM L2. Successful management of Project Planning process (PP) and the Project Monitoring and Control process (PMC) at task level while the project is in Execution, due to the complexity required for CM L2, should require the automation and assistance that, arguably, only a PMIS emplacement can provide.

Therefore, the strategic use of a PMIS emplacement in a Software Project Management methodology to achieve CM L2 maturity holds much promise to alleviate the said challenges and increase the ICT value proposition. After the installation of CM L2, process improvement projects can be used to install all the Capability Maturity processes up to Level 5.

2.19 Conclusion

The Literature Analysis above focused on several areas to understand and answer the research questions and objectives found in Sections 1.9 and 1.10 in Chapter 1.

The Research Dimensions Model Thesis Statement below, from the Research Dimensions Model in Annexure A, aims to summarise the work done in Chapter 2 into the statement below.

Research Dimensions Model Statement:

A systematic way to achieve CM L2

(to escape the gravitational pull of chaos and entropy from no process focus at CM L1)

is via TAsking (Time Asking) on developed schedules on a PMIS.

TAsk Control is like Transaction Control in a Financial System.

If CM processes PP and PMC bolster Project Execution, then they will enable Productivity

and Quality improvements needed to make and sustain the move to Stability and Productivity at L2.

Further, if initial processes REQM, CM, MA, PPQA, SAM can be installed, all processes for L2 are in place.

And if initial process VER, VAL & RKSM, then essential CM L3 processes will be installed.

Else if the above processes are not in place and managed, project management will remain at

risk and wasteful operating at CM L1.

It would appear from the literature analysis undertaken in Chapter 2 that the thesis statement above has merit. In this regard, installing the PMIS emplacement correctly by following the 12 points that make up the PMIS CM Improvement Framework in Table 4.1 could benefit the well running of software projects for compliance at CM L2. Further benefits can be gained when CM L2 as a stable base is installed on a PMIS, allowing the project team to aspire to achieve higher CM Levels considering the additional software engineering and supporting processes found in Annexure D.

The PMIS CM Improvement Framework will also have a benefit if analysed from a people-process-technology triad perspective. To this end, the PMIS (a technology), as a means of installing project management and other important processes (the 'glue'), can facilitate the process whereby complex software project teams (empowered people) can navigate their way more carefully and successfully towards success, aware of and avoiding the potential dangers to be found in the tar pit.

Finally, working harder is not always the best solution if the effort expended is at CM L1. Low capability and maturity and a lack of command and control in project management must account for many of the problems identified by the Standish Chaos Reports. Plausibly, if one cannot readily measure something, one cannot improve it. Arguably, the best way to improve anything must be to start from a known baseline.

The scope and range of topics of the articles above in relation to the proposed thesis statement of this research are broad. Much has not been included due to space considerations. Ultimately, the researcher believes that the field of study in respect of this research holds much promise.

Chapter 3 deals with the reasons and assumptions for the research and also the research design and methodology.

CHAPTER 3:
RESEARCH DESIGN AND METHODOLOGY

3.1 Overview

Chapter 2 focused on a literature review that was applied across several dimensions in software engineering and software project management. The dimensions were derived from a Theoretical Research Dimensions Model compiled by the researcher which is found at the beginning of Chapter 2 (Figure 2.1) and is also reproduced in Annexure A. The PMIS emplacement is seen as a possible way to assist with the achievement of attainment and compliancy of project and software engineering processes at a stable and product level above CM L1.

Chapter 3 looks at research methods in general to select a suitable research approach for this work. Once the research method has been selected it is considered in more detail from the perspective of the reasons for the research, assumptions, and limitations, as well as any ethical considerations that may apply.

3.2 Introduction

According to Oates (2012:16-21), the following are reasons for doing research:

- Research can be used to solve a problem or to come up with a better way.
- Research can be used to add to a body of knowledge and find evidence to inform practice.
- Research can be used to assist in improvements when it comes to predicting, planning, and control.
- Research can be used to develop a greater understanding of people and their world.
- Research can be used to contribute to people's wellbeing or add to personal needs.
- Research can be used to test or disprove a theory, to understand another person's point of view, or to create more interest in the writer.

The researcher has organised the above list in descending order of importance for this work.

3.3 Research Design Methods

The research design is approached using the method in Figure 3.1. This is based on the Research Onion by Saunders, Lewis & Thornhill, (2016).

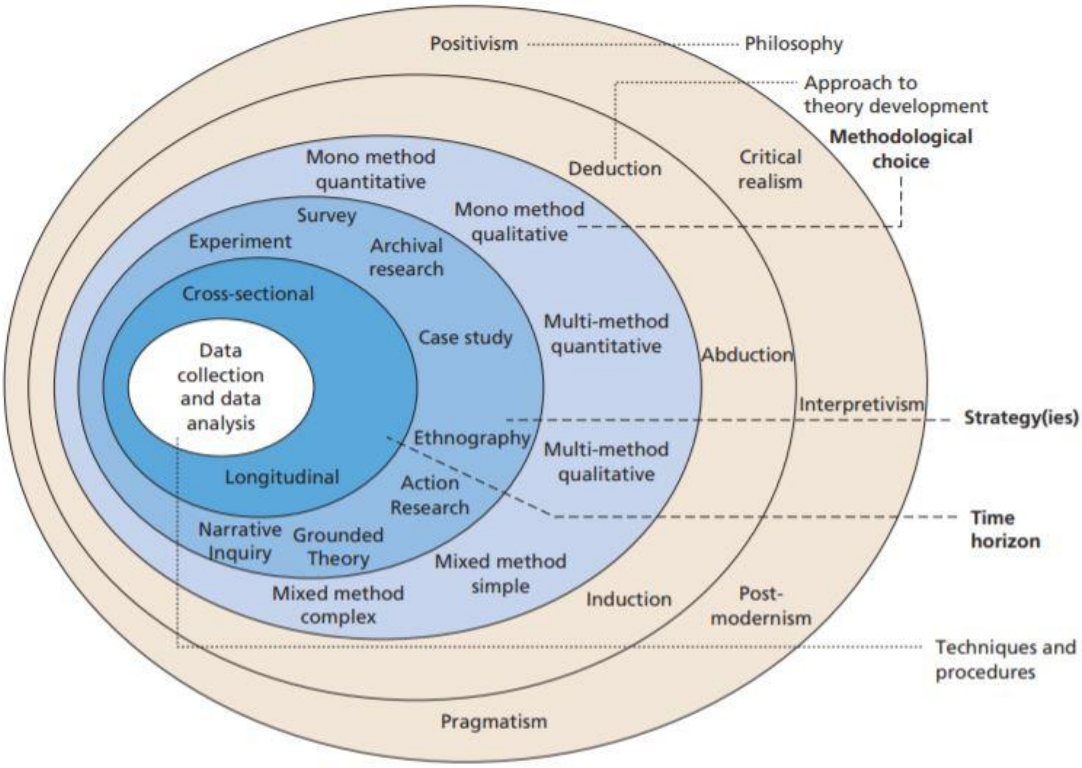


Figure 3.1: Research Onion:

Source: Saunders, Lewis & Thornhill (2016:164)

Using the Research Onion as a model, the outermost layer requires a research philosophy to be followed. This layer offers a range of choices from the Pragmatist approach at the bottom of Figure 3.1 to the Positivist approach towards the top. After consideration, the researcher settled on two possibilities of either undertaking a survey or producing a case study to test the PMIS CM Improvement Framework.

However, after the researcher reached out to a number of potentials to ask if they would be interested in running this research as a survey or case study, it was found that all potentials were not prepared to open up their project management approaches to outside scrutiny, even if the findings were to be kept confidential. Due to this, and after lengthy discussion with his professor, the researcher decided to conduct a theoretical case study rather than conduct an actual case study or run a survey.

Based on the fact that a theoretical case study would be the basis of the research, it was then decided to select the Interpretivist paradigm from the Research Onion model, using the Qualitative Methodology. This also appeared to be a suitable compromise between the Constructivist and Interpretivist paradigms.

The second layer of the onion was considered next. As deductive and inductive approaches are both possible for this work, it was decided that the middle ground in the Saunders, Lewis and Thornhill model, which referred to abduction, would be applied to this research. According to Mitchell (2018:12), when faced with continuous cycles of inductive and deductive reasoning, sometimes “abductive reasoning produces useful knowledge and serves as a rationale for rigorous research.” In addition, “mixed methods were found to combine numerical and cognitive reasoning that led to a 'best answer' to data that otherwise could not be adequately explained.” When referring to Subjectivism, Saunders, Lewis & Thornhill, (2007:136) contend “that social phenomena are created from the perceptions and consequent actions of those social actors concerned with their existence”. Railton (1995), discussing subjectivism as phenomenological research, stated that it was concerned with the study of personal experiences from the perspective of an individual, and the importance of personal perspectives and interpretations could not be underestimated.

Focusing on the third and fourth layers of the Research Onion, the mono method qualitative option appeared to be best suited for the proposed research approach, based on the use of a Theoretical Case Study. The case study in the fifth layer of the Research Onion exists in a time horizon between the extremes of cross-sectional and longitudinal research. This was one possibility from a range of methodological choices, including the experiment, survey, archival research, ethnography, action research, grounded theory and narrative inquiry.

Finally, from a techniques and procedures perspective, in the innermost circle of Saunders, Lewis & Thornhill Research Onion, while data can be collected, for reasons mentioned, the researcher will not be collecting data during this research. Instead, he will apply the PMIS CM Improvement Framework to a theoretical case study in an attempt to understand theoretically how the proposed framework could be used to improve the application of project management and other processes above Capability Maturity Level 1. While a Case Study and Survey are not ruled out, and the researcher has already compiled a comprehensive set of Survey questions which are ready to be used, this will need to form part of a future approach in this research. In this regard, even if the answer came back that organisations were not going to allow a Case Study or Survey this response, in itself, could be of value.

3.4 Ethical Considerations

The researcher is working under the guidance of the University of South Africa. Due to this fact, all ethical considerations applicable to the School of Computing (SOC) in the College of Science, Engineering & Technology (CSET) at UNISA will apply. The necessary ethical clearance for this research has been obtained, and the ethics clearance certificate is available in Annexure V.

3.5 Summary

Chapter 3 has presented the research method that will be used. It may be possible to run a Case Study or Survey in future, as part of a Quantitative Research approach, however it was decided, due to the complexity of the subject and possible reluctance to participate in such a survey, that a theoretical case study approach was best for this particular type of research. In this regard, this dissertation is seen as an initial step in discovery, while a Case Study or Survey could be a next step forward in the research.

3.6 Conclusion

Chapter 2 and Chapter 3 have set the stage to present the PMIS CM Improvement Framework in Chapter 4.

Chapter 4 will go into detail about the PMIS CM Improvement Framework, with its 12 improvement steps. A short discussion on game theory is also included in Chapter 4, as well as a discussion on how this can be used to improve the use of the PMIS emplacement. Quantitative Research will not be used at this stage of the research. However, a points system is nevertheless put forward in Table 4.2, which can be used in future quantitative research as and when it is needed.

CHAPTER 4:
PMIS CM IMPROVEMENT FRAMEWORK

4.1 Overview

Chapter 3 presented the proposed research approach for this work. The research proposes that a case study be used to research the viability of the Project Management Information System (PMIS) Capability Maturity (CM) Improvement Framework in Table 4.1. The PMIS emplacement and its potential use as a means to improve project management process application at Capability Maturity Level 2 and above, were discussed at length in Chapters 1 and 2.

Chapter 4 deals in more depth with the research offering and how it can be utilised and ways it can be further researched. As with Chapter 2, part of this chapter appears as a research article (Corrigan et al., 2018) in Annexure R.

4.2 Introduction

This chapter discusses the research offering presented in Table 4.1. A primary goal for the research is to be able to successfully implement and utilise the PMIS emplacement for CM L2 process compliance. CM L3 processes, including Risk Management (RSKM) and some software engineering processes, such as Verification (VER) and Validation (VAL), are also available to those able to install and use all of the twelve steps of the PMIS CM Improvement Framework successfully.

Using the PMIS CM Improvement Framework to create operational stability for the PMIS emplacement could facilitate a systematic move up and out of the chaos mentioned in the Standish Chaos Reports (2014). When the PMIS emplacement is operating well at CM L2, due to the stable base that this provides, additional Capability Maturity integrated processes at higher CM levels could also be more easily and successfully installed, as and when they are required. This should further improve the project team’s ability to achieve higher levels of productivity and quality by releasing value more rapidly to the business through software project management.

It is worth noting that the PMIS CM Improvement Framework presented below has focused on some functionality that already exists in commercial PMIS systems. However, other functionality presented as desirable may not yet exist. And functionality may be missing in one system but may exist in another system that the team are not using. Ultimately, the PMIS CM Improvement Framework attempts to list essential functionality that should exist in a PMIS to facilitate the successful move up and out of CM L1 behaviour to stability at project process application at CM L2. This move up and out of CM L1 is illustrated graphically as 12 improvement steps that are inserted into the PMBOK 6 dashboard, as illustrated in Annexure S.

The 12 steps in the point previously mentioned correlate with the 12 improvement dimensions (also improvement steps) of the PMIS CM Improvement Framework in Table 4.1. In essence, the 12 steps aim to bridge a possible improvement gap between CM L1 to CM L2 compliancy. In this regard, the right-facing arrowheads in Annexure S were added to identify the PMIS CM Improvement Framework scope, which is located between process #6.5 (the finalization of project planning) up to process #4.3 (the beginning of project execution, after schedule resource loading and baseline, 'minding the gap Figure 2.10'). However, all processes within the PMBOK matrix (and other project methodologies, including Agile) are considered by the PMIS CM Improvement Framework when running a project from the start of the Initiation Process Group (#4.1) to the end of a Project or Project Stage (#4.7). This insertion into the PMBOK 6 dashboard in Annexure S could also be considered as a possible improvement to the PMBOK 6 dashboard and a necessary requirement that could be considered to achieve PMBOK 6 project process application at CM L2 and above.

The essential idea of this dissertation is that the PMIS CM Improvement Framework Table 4.1 facilitates the creation of a developed schedule (resource loaded and ready for baseline) as an output of the Develop Schedule process #6.5 on the PMIS emplacement so that the project team can say, "Now We Are Ready To Do It." This essential step, often ignored in projects undertaken at CM L1, empowers CM Level 2 compliance at the beginning of the Project Execution Process Group process #4.3. If an agile method is used, then tasks can be built collaboratively, on the fly, by resources, adding their contribution to the schedule on the PMIS as the project moves into process #4.3 as a next agile iteration or time box.

The PMIS CM Improvement Framework Table 4.1 is presented on the next page.

Table 4.1: PMIS Capability Maturity Improvement Framework

PMIS CM Improvement Framework

12.	The PMIS Resource Pool is a focus of organizational attention towards skills improvement for excellence.	No	Yes (1)
11.	Risk and Issue management module on the PMIS is available to all to protect the projects (schedules).	No	Yes (1)
10.	Earned Value Management (and other) project status graphics are available on the PMIS.	No	Yes (1)
9.	PMIS projects are driven by the EPMO financial reporting cycle.	No	Yes (1)
8.	PMIS offers variance tracking of scheduled tasks as actuals against schedule baseline.	No	Yes (1)
7.	PMIS project schedules viewed as Gantt charts with task slippage indication.	No	Yes (1)
6.	PMIS facilitates "Virtual Team Collaboration" from anywhere connected to the Internet.	No	Yes (1)
5.	Stage Gate capability is available to Project Manager on the PMIS.	No	Yes (1)
4.	PMIS is secure repository for all project artifacts.	No	Yes (1)
3.	Project Managers authorised by EPMO to run PMIS Projects.	No	Yes (1)
2.	Project resourcing driven off PMIS Resource Module.	No	Yes (1)
1.	EPMO driven off PMIS Emplacement.	No	Yes (1)
➡	CM Level 1		
➡	PMIS for CM Level 2+ (12 Improvement Dimensions)		

How to use the PMIS CM Improvement Framework:

1. Enter from the bottom.
2. Award 1 point for each Yes if Improvement Dimension is in place and operating well.
3. As you proceed a No answer may not appear underneath a step you have answered with a Yes.

Source: Developed by the researcher

Table 4.1 has two entry points on the bottom left, indicated by the black right-facing arrows. Of these two, the bottom entry point is used if improvements are to be made, moving from CM L1 to CM L2. The numbers starting at 1 and continuing up to 12 on the left-hand side are the 12 improvements that will be installed by using the PMIS CM Improvement Framework. These 12 improvements are seen as contiguous steps that must be climbed in order, from 1 to finally arrive at 12. The 12 steps represent the essential PMIS functionality required in order to put into place a PMIS that will facilitate a move from CM L1 to CM L2. At each step, if the PMIC CM Improvement objective for that step is installed successfully, a “Yes” acknowledges the achievement and for this acknowledgement, a single point (1 out of 12 possible points) is awarded. The top entry point (2nd black arrow above the 1st black arrow) assumes that no improvement from CM L1 to CM L2 is possible. As such points cannot be awarded for any of the 12 steps.

Section 4.3 below unpacks how the framework is used in more detail.

4.3 More detail on how to use the PMIS CM Framework in Table 4.1

The 12 steps of the PMIS CM Improvement Framework in Table 4.1 are only accessible from the bottom left of the table. When answering the questions at each step in the table, the steps are completed in order from the bottom, one step after another. This is per design and facilitates stable operation of the PMIS emplacement as each next step depends on previous steps, i.e. the PMIS CM Improvements are fully installed in order.

The steps in Table 4.1 work bottom-up in the same way that the CMMi processes work bottom-up, as is illustrated in Annexure C. The step-by-step progression from 1 to 12 is reiterated in Annexure S, where the steps allow systematic improvement as one moves across the improvement gap between the project planning and the project execution process groups.

For each step in the framework, a **Yes** or **No** answer is required to answer the questions asked in Table 4.1:

- **Yes:** Functionality is available on a PMIS emplacement and is implemented and is operating well.
- **No:** A PMIS is not implemented within the EPMO, and the functionality is not available.

Per the design of Table 4.1, No-answers cannot appear under Yes-answers, and all higher steps are dependent on the agreement that a lower step is installed and operating nominally. If functionality required for the step is not available, or if the operation is substandard, “No” is answered. As soon as “No” is answered, then the CM Improvement level is achieved, and questioning via the PMIC CM Improvement Framework will terminate at that point. A No-answer is an indication of the next focus area for attention in order to implement the PMIS emplacement successfully at CM L2. As soon as the No-answer for a particular step is converted into a “Yes”, further progress and improvement can be made, continuing up the PMIS CM Improvement Framework.

An overview of each step is presented next.

4.4 Discussion of the PMIS Dimensions

The points below offer more detail on each step found in the PMIS CM Improvement Framework in Table 4.1.

4.4.1 Step 1: EPMO driven off PMIS Emplacement

An Enterprise Project Management Office (EPMO) should be in place. The EPMO must also be running on a Project Management Information System (PMIS) emplacement in order to qualify that step 1 is in place.

The Enterprise Project Management Office (EPMO) operates with a similar imperative as an Accounting Department, which is underpinned by a financial accounting system. Irrespective of how many, the number of projects being run on the PMIS are all unique accounting entities; each is run within its own separate project instance on the PMIS. In addition, the project instances each have separate project accounts managed by the EPMO, consolidated and reconciled into the Financial Department, per Section 2.13.1. The EPMO has the final say on which projects are created within the PMIS.

Projects can also be grouped into programs and portfolios. Depending on how the projects, programs and portfolios are set up on the PMIS, these groupings will be available as selectable views from within the system. Not only can groups be viewed on the PMIS, as previously mentioned, but also by province. In addition, Project Management Offices within provinces on the PMIS will all consolidate and report into the EPMO.

When the researcher worked in the Public Sector, the PMIS that was used in the PMO was set up to run in this manner. In this regard, projects in a province were grouped into either a project or groups of projects by programme or portfolio and this facilitated decentralisation from the central EPMO. In this regard, a province had full control of their PMO and human resources allocated therein but was accountable to the EPMO for delivery and ongoing project performance. Based on this structure on the PMIS, it is clear that rogue projects were not allowed. A rogue project is a project that does not fit into the PMO or EPMO strategic plan. In this regard, Figure 2.11, the Meril-De Programme and Project Initiation Processes refer.

4.4.2 Step 2: PMIS resourcing driven off Resource Module

The PMIS emplacement has a central resource module wherein all project human resource details are loaded. This module contains personal details of all project human resources, including their skills and other information. To work correctly, the resource module stores all relevant certifications for resources that appear in the module. For example, if a human resource is employed as a Microsoft Certified Systems Engineer, this certificate is secured within the module, transparent and available to all working on the PMIS system.

Project schedules on the PMIS can only be resourced (from a human resource perspective) from the PMIS Resource Module. Per Step 3, an appointed project manager is permitted to run a project by the EPMO, with allocated skilled human resources to the project from the Resource Module.

4.4.3 Step 3: EPMO authorises Project Manager for PMIS Projects

Each project that is created in the PMIS is allocated a project manager. In this regard, the project manager can be seen as the owner of the project. The project manager is accountable for good governance and professional project management process application for their project, accountable for project performance to their PMO but ultimately to the EPMO. As Agile projects see the project manager role as a servant-leader role, this should not undermine the intention of Step 3, which is to have a single point of contact who is authorised to run the project on the PMIS.

With regard to Step 2, the project manager is responsible for allocating resources from their project resource pool, made available by the central EPMO resource module, to tasks on the project schedule.

4.4.4 Step 4: PMIS is a secure repository for all project artifacts

All project artifacts, including documents, diagrams, audio, video, etc., with changed versions, should be saved to and secured within the PMIS document management system.

The document system, or repository, can be extended but is not restricted to include, for example, a copy of the latest source code and the Definitive Media Library (DML), which forms part of the continually integrated code base. The DML is an output from the ITIL processes discussed in Chapter 2 Section 2.15.1. Being able to release working software correctly and fast, and with due consideration and respect for the governance surrounding these processes is essential for the well running of Traditional and Agile Project Management and ITIL, COBIT, DevOps, and so forth. The processes pertaining to ITIL (Chapter 2 Section 2.15.1), DevOps (Chapter 2 Section 2.15.2) and COBIT (Chapter 2 Section 2.15.3) refer.

The project manager has control over the relevant project artefacts, and it is up to the project manager to ensure these are uploaded into the project space (repository) in the PMIS. However, the EPMO will also need to have control over project artefacts from a governance perspective. To this end, versions of a document are kept centrally on the PMIS. This allows the project manager (or other project resources, such as Business Analysts and others) to check out a document, make changes and save these as a latest version. In this way, every time a project document is saved to the document repository and approved by the project manager, only the latest version of it will be accessible. Any changes will increment the document version and a saved document can only be deleted by those with appropriate permissions. From an EPMP and good governance perspective, permissions would need to reside centrally.

If the PMBOK. PRINCE2 or Scrum Body of Knowledge methodologies are used, then outputs of each process will typically indicate what artifacts are required for completion of that process. This means that as the project moves through its lifecycle, key artefacts that need to be produced, stored, and secured will be known, potentially forming the basis of an audit trail by the EPMO or other auditing body. From a good governance perspective, it follows that a project cannot be created by the EPMO unless the project has been officially authorised (PMBOK processes 4.1 and 13.1 Annexure S), which typically will require an approved Project Charter document. As the PMBOK Knowledge areas point to project artefacts that are required, so do the PRINCE2 or Scrum and other Methodologies point to their specific project process requirements at each step of a project lifecycle.

For an Agile project utilizing the Scrum Methodology, approved User Stories before the Sprint starts and a Sprint report, Product Owner approved Demo Deck, and a Retrospective report at the end of the Sprint are essential project artefacts that should be saved to the project repository regularly. Typically, this would need to be Sprint by Sprint. More information on how Scrum functions and its associated processes as discussed in Annexure O.

4.4.5 Step 5: Stage Gate capability available on the PMIS.

Stage Gates or Quality Gates are available on the PMIS for use by the project manager. Typically, the project manager can configure the PMIS database to have Stage or Quality Gate selectors, which can be applied when required, but typically after delivery of an output from a finalizing project process.

Example 4.1 below illustrates how Stage or Quality Gates can be used.

Example 4.1

A project remains in the Initiate Process Group until the Project Charter with Stakeholder Register (approved with signatures) is archived in the PMIS document repository. Only then will the project manager open the stage or quality gate and allow the project to move into the Planning Process Group. Only the project manager for this project has the rights on the PMIS to close and open the gate. It may appear that quality and stage gates go against agility, but this is not the case. Often projects run into trouble simply because they were not fully approved or correctly initiated. In essence, the ability to correctly apply the processes in the Initiate Process Group is the difference between a lack of State or Quality Gate at CM L1 and one operating well at CM L2.

An example of behaviour at CM L1 in the Planning Process Group could be that numerous project artefacts are required to complete the Project Plan. However, though several requirements are yet to be supplied at the required quality, or approved, these are ignored. Owing to urgency, typically from management, the project is rushed into Execution before sub plans for Scope, Cost, Time, Stakeholder Management or other Knowledge Areas are finalised, or a completed Project Plan (PMBOK process #4.2) or Project Schedule (PMBOK process #6.5) is available.

A project manager operating at CM L2 would not allow the Stage or Quality Gate to be opened, consequently not allowing the project to move into the Execution Process Group until all process outputs for the Planning Process Group are completed in good order.

4.4.6 Step 6: PMIS facilitates Virtual Team Collaboration

Any resource in the EPMO resource, allocated to a project resource pool by the EPMO, has to be configured with appropriate rights to work on a project and collaborate with other team members using the system. Authorized access can be via a local area network, wide area network, or from anywhere accessible globally via the internet. As the PMIS emplacement needs to be secured from threats in the environment (viruses, hackers, etc.), security rights must also allow users to connect to the system when working remotely.

Example 4.2 gives an example of virtual team collaboration.

Example 4.2

A real-world example of virtual collaboration on a PMIS is evidenced by the fact that the researcher is currently working on this dissertation but is also logged onto his work virtual private network via the internet from an uncapped high-speed router at his home. In the last two minutes, he has updated information in a user story in a project where he is the Scrum Master and has changed its status from Blocked to Do. The researcher, who is currently working from home in a suburb in Johannesburg, has a work office based in Sandton in Johannesburg that he works at, an hour away by car. The PMIS database is currently being backed up as part of a backup plan into the cloud, which data could be located anywhere in the world. In addition, resources who are working on the same project are accessing it in real-time from India late in the afternoon, while the researcher is accessing the project in the morning. Finally, a stage gate is imposed in the project where only the Scrum Master can close out completed work, which will be done on receipt of successful test results which are due from a tester in India.

4.4.7 Step 7: PMIS schedule as Gantt chart with task slippage

The Project Schedule can be presented as a Gantt chart with task slippage indicator capability. Figure 1.2 illustrates how it will look on a PMIS emplacement. Task slippage is typically a setting that is put on or off on a project schedule in the PMIS. Scrum boards typically do not have this functionality because the depth of detail (Figure 1.1 - The Triple Constraint) may not be regarded as important in order to be agile. The time constraint, specifically for all user stories on a scrum board, is managed within a timebox. However, if a task is to contain sufficient information to facilitate project monitoring and control, and if a baseline is also used, then slip lines are a natural extension of PMIS functionality, showing that a task's estimated due date for completion is within nominal limits or has been exceeded. Because a project becomes late a day at a time, being able to see what tasks are slipping facilitates rapid resolution and project health. It is the researcher's experience that electronic Scum boards do not (arguably cannot) offer this level of detail and functionality.

4.4.8 Step 8: PMIS task variance against schedule baseline

If a project schedule is baselined, it will be possible to compare actuals against baseline during the next project phase or over its lifecycle. The project schedule is not the project plan. However, a saved baseline of the schedule could be regarded as the project plan stamped or saved to compare actuals against the planned schedule in the future. As the schedule is worked on, and tasks are completed or uncompleted and moved forward to new dates, this, compared to the baseline schedule, will give an overview of project activity and status. Having a baseline in place against the planned and developed project schedule facilitates implementation on the PMIS emplacement of Earned Value Management metrics discussed in Section 2.8.

4.4.9 Step 9: PMIS projects driven off EPMO financial cycle.

The EPMO supported by the PMIS emplacement should work off a regular reporting cycle with regard to project financials. Financial reporting ties the actual work completed by projects back to the achievement of project plans and utilization of approved budget allocations for those plans.

An example of how this works on a PMIS at task level is given in Example 4.3.

Example 4.3

An example of how this works at task level could be development work, which plans for and takes a developer 3 hours to produce a software module. If the developer rate is R1000 per hour, then the cost of the completed task for the project is R3000. When the task is completed and signed off after testing, the module can be paid for; the cost is debited against the project's budget. Cumulating totals for all projects at the EPMO level will reconcile into financial entries in the Accounts Department. If the task takes 6 hours to complete, then the same software module cost would double, and the project budget will be eroded more rapidly and at a higher rate than originally expected.

From an accounting perspective, being able to tie task completion on a schedule to the cost of the developer and the individual project budget allocation on the PMIS back to the overall EPMO project's budget is desirable. In essence, this means that a budget allocation can be utilised by a project as long as the project is producing value. Extra budget will need to be motivated should time estimates prove to be inadequate.

Step 10 explains more about the measurement of value and how the PMIS emplacement can be used to monitor and control planned delivery against approved delivery.

4.4.10 Step 10: Earned Value Management off the PMIS

Earned Value Management Analysis (and other) project status metrics, graphics, and reports can be made available on the PMIS if a baseline is in place. Figure 2.7, the Earned Value Management (EVM) graph, consolidates many of the nine popular project formulas that are available for a PMIS emplacement if the system has been set up correctly. Figure 2.7 is a standard graph used by Project Management Professionals to ascertain how to run a project successfully from a budgeting and cost perspective to ensure that value is produced within the project constraints.

The "To Complete Performance Index" formula, one of the nine popular project formulas mentioned, is a single number for project health throughout a project's lifecycle.

However, the EVM graph in Figure 2.7 and the standard project formulas are often unavailable to a project team because they are unfamiliar with their use, or the project is not empowered and set up correctly to produce the information required to facilitate the management of earned value.

The utilisation of Earned Value Management, built from tasks and their completion status, can be an essential PMIS reporting tool to facilitate the goal of using the project schedule as a playing field. This capability unlocks the benefits of game theory mentioned by Bočková et al. (2015:715).

4.4.11 Step 11: Risk and Issue management off the PMIS

The use of the Risks and Issues Module on the PMIS means any member of the team can be involved in and can manage project Risks and Issues. In addition, issues can also be allocated to any member of the team as tasks on the project schedule for resolution. Risks and Issues are also visible to and can be allocated to the executive or project stakeholders so long as they are loaded onto the Resource Module (Step 2).

Per Deming, Annexure F and Annexure G, this allows a project team to hold their executive to account, ensuring they are fully involved in the project and can be called upon to assist should the project run into trouble. Executive support to continue with the project, or not, can be based primarily on accurate project status derived from the formulas in Step 10. Step 11, based on the full implementation and utilization of steps 1 to 11, the researcher contends, could be a single and compelling reason why the PMIS emplacement can assist to combat the chaos that is explained in the Standish Chaos reports (2014).

4.4.12 Step 12: PMIS Resource Pool to focus skills improvement

The PMIS Resource Pool is a focus of organisational attention towards people and the improvement of their skills for excellence. People excellence is dealt with in the People Capability Maturity Model in Annexure I. An important goal in project management should be the creation of Empowered Teams, which is a People CMM L4 process. To achieve PCMM L4 may require sourcing or qualified individuals or expenditure on training for key roles in the team. Based on the other PCMM improvement dimensions found in Annexure I, it is clear that the PMIS emplacement can facilitate numerous benefits with regard to People Capability Maturity. Professional teams can be visualised and then created within the Resource Module of the PMIS emplacement, thus facilitating the benefits of game theory.

Example 4.4

A highly skilled developer is needed for the team. This developer is expensive. Using the Resource Pool (a module within the PMIS), motivation is put forward to recruit someone with the required skills. The People Capability Maturity model is very clear; it does not make sense to lose quality staff. Therefore, every effort is made to keep the team member satisfied and working productively on the team. In the researcher's experience, companies he has worked for that do not understand this concept find that they lose unhappy or frustrated staff who walk out the door with much hard-gained knowledge as intellectual property (potential competitive advantage) that the company has invested in heavily.

4.5 PMIS developed schedule and Game Theory

The performance and betterment of professionals and teams can be enhanced by monitoring production statistics. If performance is monitored over time, statistics that improve can indicate health, power, and affluence, while those that decline may indicate problems that need to be addressed. Performance that has slipped from normal operation down to emergency and then into danger may, if not handled, end up as a non-existence condition. If the condition of non-existence is understood as a contra survival state, this is something that ought to be avoided.

Example 4.5

The American National Football Association (NFL) professional teams are a good example of team excellence and also of a high level of capability and maturity in operation. The national football culture in the United States ensures that schools and universities nationally feed the NFL annually with highly competent footballers, all with the desire to be headhunted for skills they have mastered up to that point. Extensive training and financial rewards within the NFL imply that team members are supported by their teams to become the very best professionals that they can be. It is the opinion of the researcher that much of the success of the NFL is created primarily by their playing field, which looks like a ruler. This 100-yard (or 91,44 meter) long field allows measurement of activity in yards (meters), down to 1-inch (or 2,54cm) precision, as is shown in Figure 4.1. Often only a few inches will decide who gets the advantage at the end of a series of downs; this specialised measurement is undertaken by the "chain gang." Because the NFL is fastidious about performance statistics, these are often quoted as record holders for the longest throw or fastest drive to touchdown from their own 10 yards, and so on.

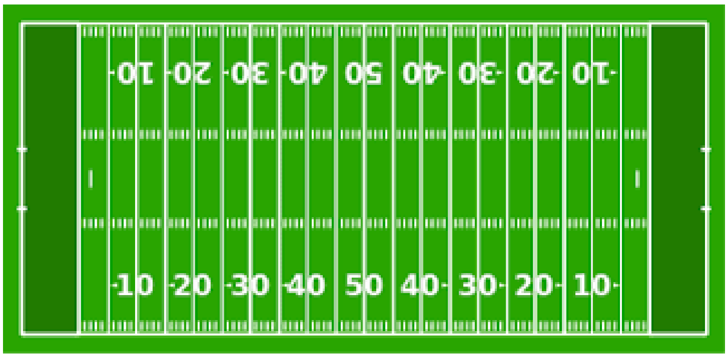


Figure 4.1: American Football Field (Like a Ruler)

Source: https://en.wikipedia.org/wiki/American_football_field

Figure 4.1 is a graphical view of an American Football field. Clearly this looks like a ruler. The length of the field is 100 yards from end to end, made up in 10-yard increments. Within each 10-yard span, this is broken down into clear inch marks. When a game is in play, all activity is closely related to gains or losses of a team within these grid marks. The fact that the playing field is created to look like a ruler and to give rapid feedback in the form of measurements supports the truth attributed to Lord Kelvin, “If you can’t measure it, you can’t improve [on] it.”

In a similar way to the American Football field in Figure 4.1, the developed schedule on the PMIS emplacement is also a playing field that project resources can play on, using performance stats to improve their game. Figure 1.2 clearly shows how the developed schedule, with slip lines, can be used to facilitate productivity improvements.

Using this NFL analogy, if a project can be regarded as a game, and the best project teams need to be good enough to make it to the Super Bowl, then highly trained professionals is an essential part of the attainment of excellence. The management of the complex software project resource pool for consistent delivery of valuable software by the most experienced people should be self-evident. If staff are not a correct fit, or their behaviour is sub-optimal or tardy, then, as with American Football, they will not last long in a winning team. This essential feature of continuous improvement is handled by point 12 of the PMIS CM Improvement Framework.

4.6 **Using the Framework for Quantitative Analysis.**

Chapter 3 decided not to focus on Quantitative Analysis and Surveys at this stage of the research. However, as 12 points are allocated for the attainment of improvements across each of the 12 steps in the Capability Maturity Improvement Framework model in Table 4.1, this can offer the possibility of an improvement path that can be quantitatively managed, as illustrated in Table 4.2.

Table 4.2: PMIS CM Point System

POINTS	PMIS CM LEVEL	CAPABILITY MATURITY
10 - 12 Points	PMIS CM Level 2	Expert
7 - 9 Points	PMIS CM Level 2	Advanced
4 - 6 Points	PMIS CM Level 2	Intermediate
1 - 3 Points	PMIS CM Level 2	Beginner
0 Points	PMIS CM Level 1	Unaware

Source: Synthesised by the researcher

If the steps in Table 4.1 are used in conjunction with Table 4.2, this can be used for quantitative research. Example 4.6 below explains how this could work.

Example 4.6

Starting at the entry point at the bottom of Table 4.1:

- Step 1: The EPMO has PMIS in Place: Yes [√] = 1 point.
- Step 2: The Project Resourcing is being driven off a Resource Module in PMIS: Yes [√] = 1 point.
- Step 3: Project managers are put into place by the EPMO: Yes [√] = 1 point.
- Step 4: The PMIS is a secure repository for project artifacts: No = 0 points.

The total from the above exercise is 3 points. Using Table 4.2 above, with 3 points awarded, this indicates that a Beginner PMIS CM Improvement Framework Level has been achieved.

Chapter 3 decided that the researcher would not undertake quantitative research. It will be recommended in Chapter 6 that future work could consider the possibility of using the above approach to rank CM Levels of respondent feedback in a survey.

4.7 **Summary**

Chapter 4 presented the PMIS CM Improvement Framework. The PMIS CM Improvement Framework is essentially a guide for the executive and other stakeholders on how to source, install, implement and use the PMIS emplacement in order to move software project processes to stable operation at CM L2. Each of the 12 improvement steps was explained, and details were given on how each could be implemented. It was further mentioned that each step, from 1 to 12, should be implemented to ensure that lower improvement objectives were installed and operational first; before looking at next steps. As soon as it was discovered that a step was not installed and operational, this would form the focus for areas for improvement. Ultimately, the goal would be to implement all the steps in order, from step 1 to 12, with the output, a fully functional PMIS emplacement.

Game Theory was discussed towards the end of the chapter. In this regard, the developed schedule on the PMIS emplacement was viewed as a playing field where team action could be measured for improvement based on performance metrics. For this to work, the developed schedule on the PMIS emplacement would need to be seen as the software project team’s playing field. Finally, as this research is based on a Theoretical Case Study, an additional table has been added to the end of the chapter that illustrates a points allocation system to implemented steps in the PMIS CM Framework. The idea would be to use this for future quantitative research, possibly to elicit responses utilising a questionnaire.

4.8 Conclusion

The PMIS CM Improvement Framework has now been developed and is ready to be validated. Chapter 5 applies the PMIS CM Improvement Framework to a Theoretical Case study. During the next chapter, if the researcher finds during the Theoretical Case Study that he needs to adjust the PMIS CM Improvement Framework from this chapter to include improvements, this will be done as and when required.

CHAPTER 5:
CASE STUDY

5.1 **Overview**

Chapter 4 discussed the PMIS CM Improvement Framework and its 12 steps in detail. The PMIS CM Improvement Framework was presented as a guide for the executive and other stakeholders on how to source, install, implement, and use the PMIS emplacement in order to move software project processes to stable operation at CM L2. Each of the 12 improvement steps was then unpacked, and details were given on how each could be implemented. The concept of Game Theory was addressed at the end of the chapter, and an additional table that tied a points allocation system to implemented steps in the PMIS CM Framework was added. As indicated, this table could be used in future work to facilitate quantitative analysis.

This chapter will make use of the PMIS CM Improvement Framework and apply it to a Case Study.

5.2 **Introduction**

A theoretical case study on which the framework will be validated is discussed in section 5.3 below. The case study is presented in italics to differentiate it from the rest of this dissertation.

Thereafter, Section 5.4 will be used to apply the 12 PMIC CM Improvement Framework steps in Chapter 4 to test the framework against the case. As the case study is tested, findings may result in an amendment of the PMIS CM Improvement Framework in Table 4.1. Any improvements noted are indicated towards the end of the chapter. It is also captured in Annexure U.

5.3 **Theoretical Case Study**

A national agency in the public sector of a Southern African Country provides Information Technology Services to Government Departments and Municipalities. The services encompass the provision and maintenance of software, hardware, and infrastructure, including ad hoc services, such as procurement, consulting, and project management. The goal of the Agency is to keep costs low by centralising and therefore *maximising government buying power through economies of scale. The agency is structured to implement IT policy, procedures and controls centrally from its head office, and while regional offices cater to clients in the regions, they are ultimately accountable for performance to head office.*

The agency decided that procurements must be centralised in order to enforce transparency and good governance. Procurements processes have also been brought under the auspices of the agency's Enterprise Project Management Office (EPMO). This is due to the fact that procurements are seen as one of the PMBOK knowledge areas, and the handling of procurements within EPMO projects should increase transparency while helping to reduce tender fraud, which is a problem for the agency. It is also the intention of the agency to implement the Capability Maturity Model integrated (CMMi) from Carnegie Mellon University, which supports the three constellations of CMMi processes for improvements in Development, Procurement, and Services. These, the agency understands, will greatly assist them to deliver improved services to their clients. In addition to the three constellations mentioned, the People Capability Maturity model is also targeted for implementation to assist the agency with the recruitment of skilled staff, and for it to become an employer of choice.

As the government is currently implementing several ICT standards, it has been agreed that all client engagement with the agency must follow the best practice standard for IT Service Management encapsulated in the ITIL (IT Infrastructure Library) version 3. Accordingly, Service Level Agreements (SLA) must be put into place and approved for each Municipality or Government Department before additional budget for IT services will be approved. In addition, wherever possible, all new requirements must be incorporated within projects managed by the EPMO. This is in line with ITIL, which will create a project within the Service Strategy lifecycle phase, should the requirement be part of a strategic initiative aimed at improving existing valuable offerings to clients or developing new value stream. A national help desk with a toll-free call centre is available to agency clients to log all service requests, and for the management and resolution of service problems.

Feedback from clients, including management statistics from the help desk, indicates that hardware and infrastructure, and even the provision of "commercial off-the-shelf" (COTS) software solutions, like the Microsoft Office suite, appear to be working well for the agency and that these are easily managed within the ITIL service management *model and SLAs. However, the agency appears to run into difficulties when new value streams driven through projects need to be managed well. As all projects types appear problematic when managed within the agency EPMO, it appears that the agency is specifically challenged in producing valuable and need-satisfying software solutions, which are urgently required by clients. This is exacerbated by the need to provide customised software programs that are developed in house to meet specialised needs of the government and municipalities. These specialised systems often fall under national legislative constraints applicable to State Security or the Justice Departments, and due to confidentiality reasons, software requirements cannot be outsourced to private companies.*

To cater for these needs the agency has recently created a specialised software projects management unit that operates mainly from the Head Office but with some staff contributing from the regional branches. Following the ITIL method, Continual Service Improvements for core systems produce a backlog of requirements in ITIL Service Strategy, indicating the need for the creation of system-specific software projects. ITIL Service Design and ITIL Service Transition ensure that the next versions of software are designed, developed and tested before being transitioned into a User Acceptance Testing environment, typically held at the Municipal or Governmental office. From there, after UAT sign off, the new or next version of the software is released into live at the Municipality or Government office. In certain cases, production machines may be housed centrally, or these can be decentralised into the regions, depending on what is the most cost-effective or apt option. If a new IT stack is required, then the software project will facilitate the installation and new software system deployment onto these handling procurements of the new stack within the project.

Unfortunately, as there is an SAP system to manage finances within the agency, it appears that it is unable to manage project costs, except at a high level, as a budget allocation with project costs debited at some point after they are incurred. To this end, SAP project budgets are often not in line with project performance, and it is difficult to manage the delivery of project value produced against project spend. In addition, more often than not, budgets are exhausted long before the project can be deployed and signed off by the client as completed. And when software is finally released into User Acceptance Testing, the client is often irate because the functionality they were *assured would be provided is either missing or the software does not work as expected.*

It was thought that Traditional Project Management methods were outdated and that by implementing agile methods, the value would be more rapidly deployed, and client frustration would be reduced. Yet, even after installing the latest Scrum Agile methods and providing agile boards, stickies and pens, it appears that the ability to produce successful software was no better than it was before. In fact, the problems seem to have escalated because by moving over to agile, many of the traditional controls were no longer maintained, and consequently, more software projects were failing mandatory government audits. There is also a serious resource allocation problem as software developers are uniquely skilled, over-allocated and in short supply. Due to the pressure of work, often developers are allocated to work on a particular project. However, due to pressure to deliver, they are also needed on other projects that have run late. Many software developers have left the agency due to increasing workload and being held accountable for failures they feel are not their responsibility due to insufficient time to do the work properly and over-allocation.

The problem, in the form of continued and vociferous complaints from frustrated clients, has now been escalated to the office of the Minister in charge of the agency. Being concerned, the Minister demands a clear plan of action to solve the problems going forward and requires monthly status reports, and ultimately, a solution to the problem within 24 months.

The EPMO head has found a Project Management Information System (PMIS) Capability Maturity (CM) Improvement Framework in a research paper and has decided to use it to implement a PMIS emplacement while simultaneously striving to install CMMi for Development version 3.1 and People Capability Maturity Model for process compliancy improvements in software engineering and towards becoming an "employer of choice." It is hoped that the Framework can also speed up the ability to implement crucially needed parts of CMMi for Procurement and Services as these are used with CMMi for Development. To this end, the EPMO head has also put in place a goal to pass a "CMMi SCAMPI" audit for CMMi for Development processes version 3.1 at CM L2 within 12 months. This goal, the EPMO head believes, should focus primarily on the software development teams within the agency to get them working *successfully off the PMIS emplacement at CM L2, and, if the goal is successfully achieved, this should go a long way towards addressing the ministerial concerns.*

End of Theoretical Case Study:

Source: Synthesised by the researcher

Section 5.4 below will take the above case study and apply the PMIS CM Improvement to it.

5.4 **Applying PMIS CM Improvement Framework to the Case Study**

The intention will be to start at the bottom of the framework at step 1 and then move through each of the steps or improvement levels, describing the application of each to the case study as we proceed step by step up the 12 steps in the framework.

5.4.1 Step 1: EPMO driven off PMIS Emplacement

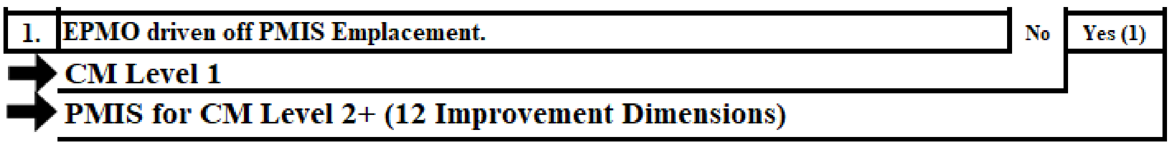


Figure 5.1: PMIS CM Improvement Framework Step 1

Source: Synthesized by the researcher

Step 1.1 EPMO head Assumes Key Stakeholder Responsibility

The Agency Head of the Enterprise Project Management Office (EPMO), schooled in the project methodologies of PMBOK, PRINCE2, PRINCE2 Agile, DSDM, SCRUM, and also in CMMi, ITIL and COBIT, and with an understanding of Deming's wisdom found in Annexure F and Annexure G, appreciates that the role of Agency Head would have to be personally involved in the CM L2 process improvement project for it to be successfully implemented and operational within a year. The EPMO head knows from research available that the installation of all the CMMi processes from CM L2 to L5 (per Annexure D) should take between 3 to 5 years to complete.

Step 1 (Figure 5.1) has revealed a shortcoming in the framework. The importance of the EPMO head as a key stakeholder to follow the PMIS CM Improvement Framework and use it to drive the PMIS emplacement into existence was found to be a strategic driver for success. The revised framework, which has been amended to include this finding, is presented at the end of this Chapter in Section 5.4 and also in Annexure U.

Project Processes Prioritized in Phase 1

To reduce the potential for overwhelm when trying to achieve too much too quickly, phase 1 of the project would be to set up the PMIS emplacement to empower mainly CM L2 processes, with only a few CM L3 processes. Therefore, the researcher suggests the installation of a PMIS emplacement for the following CMMi for Development processes only, to be achieved in the following order: PP, PMC, REQM, CM, MA, PPQA at CM L2 and RSKM, VER and VAL at CM L3.

Stakeholder Engagement

To simplify the project for stakeholder understanding, the CM model in Annexure C was used. This model, in the form of a table listing each of the CM processes in order by CM level, would be easier to comprehend than the schedule of all CM processes that is Annexure D. Figure 2.5 was also used to explain to stakeholders the problems and inherent risks associated with operating heroically at CM L1. Annexure S was used to explain how the PMBOK matrix works with regard to Knowledge Areas and Process Inputs, Tools and Techniques, and Outputs. The need for the EPMO to drive towards process excellence, especially in CMMi, ITIL, PRINCE2, and PRINCE2 agile was stated, as well as emphasising the use of the PMIS emplacement to facilitate the well running of these processes within the EPMO.

Project Duration and Focus

The Project estimates for phase 1:

- 24 months to implement the PMIS (12 months safety buffer has been included).
- PMIS is in place and ready to handle CM Level 2 processes + RSKM, Ver and Val at CM Level 3.

Discussions held pertaining to project duration and focus

Stakeholders were informed that the first three processes to be installed should be PP, PMC and REQM. These core processes would be essential in order to move away from CM L1 behaviour. If installed correctly on the PMIS emplacement, these core processes could assist to move the EPMO over the line towards institutionalising and entrenching CM L2 behaviour.

The project was estimated to take 3 to 5 years if it was to address the full installation and stable operation of all processes at CM L2, L3, L4 and L5. However, the big bang approach was not recommended. It was decided to reduce the scope and only handle CM L2 processes and three processes from CM L3 as a phase 1.

Additional processes in CM L3, L4, and L5 would receive focus after the successful implementation of phase 1. As the CMMi constellations in Figure 2.4 were interlinked, achievements in CMMi Development could be more easily extended to CMMi Services and Procurements if it was demonstrated that the PMIS emplacement was installed correctly and working optimally. Again, the decision to run additional phases would depend primarily on the ability to demonstrate the success of phase 1. It was also appreciated that there might not be a need to install additional processes if those in phase 1 sufficed.

Using the PMIS CM Improvement Framework as a Phase 1 Checklist

To achieve success in phase 1, the EPMO head decided to follow the PMIS CM Implementation Framework steps using Table 4.1 as a checklist, which is its intended use. The EPMO head appreciated that with the PMIS CM Implementation Framework, after successful implementation of the 12th step, the core processes for CM L2 should be installed on the PMIS emplacement, including RSKM, VER and VAL at CM L3. To this end, by following the PMIS CM Implementation Framework steps, especially if these were driven into place under tight direction and management of the EPMO head, it was estimated that there was a good possibility to complete phase 1 of the project within 12 months. To be on the safe side, the EPMO head negotiated an emergency buffer of an additional 12 months, making the planned duration of phase 1 to be 24 months.

Step 1.1: PMIS Procurement, Installation and Administration

Under the auspices of the EPMO head, with the full support of the Minister, the Agency rapidly obtained authorization and then procured and installed the latest version of a COTS PMIS Solution. Selection of the PMIS solution vendor was facilitated by the PMIS CM Implementation Framework Table 4.1 as core features that were required for PMIS emplacement success could be easily identified upfront.

PMIS Administration

For Step 1, the EPMO head recruited two certified PMIS Administrators to set up the PMIS emplacement within the EPMO. One, with over a decade of working experience, was appointed the lead administrator, and the other was appointed in a supporting role. The support administrator would be trained by the lead administrator to support the PMIS, should the lead administrator need to go on leave or not be available. The EPMO head was given full administration rights and assumed the third administrator role alongside the PMIS lead administrator and support administrator. All governance on the system started and ended with the EPMO head, who was ultimately responsible for its stability and well running in Service Operations (ITIL Lifecycle phase 2.15.1).

After 1 Month of the PMIS Installed

To this end, the PMIS system, as an emplacement, was fully installed and was made available to all EPMO staff nationwide in one month.

PMIS Sandbox

The ability to create projects in a live environment was restricted initially, except in a "sandbox" environment, where project managers could only play with the system and create dummy projects. They could add other resources to test these projects if they were added to the resource module (Step 2) and could log onto the project space in either the Project Manager or Developer role to test the system functionality. The sandbox was an exact mirror of the live environment and would always remain in place for training purposes.

The application of Step 1 of the PMIS CM Improvement Framework facilitated project progress, apart from the shortcomings identified and rectified in the enhanced framework found at the end of this Chapter in **Section 5.5** and **Annexure U**.

5.4.2 Step 2: PMIS resourcing driven off Resource Module

2.	Project resourcing driven off PMIS Resource Module.	No	Yes (1)

Figure 5.2: PMIS CM Improvement Framework Step 2

Source: Synthesized by the researcher

As the PMIS emplacement comes with a resource module, the EPMO head tasked that all resources within the agency (within the EPMO) must be entered into this module. Specific requirements from the EPMO head was a high level of security needed for resource module administration. To this end, the lead and support PMIS administrator would assume the role of resource module administrators. Both would back up each other, and between them, they would control the quality and functionality of the resource module within the PMIS emplacement. The EPMO head was again given full administration rights and assumed the third administrator role alongside the PMIS lead administrator and support administrator, offering input from a quality assurance perspective.

Add EPMO resources to Resource Module

The resources to be entered would include EPMO and PMO executives and secretarial staff, project managers, project administrators, and software developers and testers. When adding resources, their personal details would need to be captured. To facilitate this, from a completeness perspective, a template was set up, which mirrored the input form within the resource module of the PMIS emplacement. In addition, each resource would also require information captured about their job role (project manager, Developer, Tester, EPMO administrator, etc.). The template supported the input field form, and mandatory fields had to be completed before the form could be saved to the PMIS resource module database, ensuring completeness. All other staff added to the Resource Module could view the Resource Module, but they could not add, modify or delete records in the module.

Step 2.2: Add project managers and Developers certifications

From a qualifications and certifications perspective, the first goal was to ensure that only project managers with valid certifications were added. To this end, the methods PRINCE2 and PRINCE2 agile were the preferred project management standard for the agency. The reason for using the PRINCE2 method as a standard was due to the success of PRINCE2 and PRINCE2 agile in governments in the United Kingdom and Europe. Only certified Project Management staff with valid certifications could be added to the resource module in the Project Management role, which ensured quality human resourcing. The intention was that at any time the resource module was audited, the auditor would be able to obtain a list of certified project managers and also view the latest PRINCE2 or PRINCE2 agile certificates, which were scanned into and secured in the PMIS document share. A certification expiry date was also entered into the system to facilitate six months' prior notice of certificate expiry to enable the project manager to prepare themselves to update their certification.

Add Certified Developers

After all National project managers were entered into the PMIS resource module, all other developer staff on a national basis were added. The EPMO was aware that project managers and developers who resided outside the national boundaries could be added, which was facilitated by the PMIS emplacement. Developers is regarded as an all-encompassing term that included software developers, database developers and designers, testers, business analysis, technical architects, and others. The job was completed in two weeks after the installation of the PMIS emplacement within the EPMO.

After 1 ½ months resource module fully installed and functional

After one and a half months (six weeks), the PMIS emplacement was ready for use in a production environment.

Step 2, which involved the installation of the PMIS by the PMIS administrators, including the PMIS resource module, prepared the environment as a solid base to support many of the next steps in the PMIS CM Improvement Framework. These were implemented systematically, as will be discussed next.

5.4.3 Step 3: EPMO authorises Project Manager for PMIS Projects

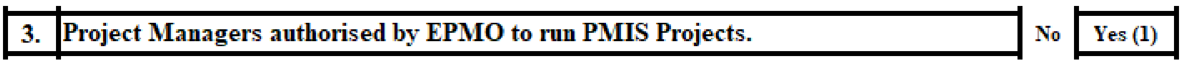


Figure 5.3: PMIS CM Improvement Framework Step 3

Source: Synthesized by the researcher

Only certified project managers added to the PMIS emplacement could create projects on the PMIS emplacement. In addition, projects could only be created when they were required according to the ITIL process in Service Strategy, where this was an approved requirement with a signed off Project Charter and Business Case (typically authorised by the PRINCE2 Executive Figure K.2).

The Business Case would typically be for the initiation of a new project for new software or a next version of existing software. Annexure E shows, within the Service Strategy phase, that the trigger for a new project could only occur after an approved

Project Charter and Business Case. Figure 2.7 explains how the processes in the Strategy Planning Wall can be used to support ITIL Service Strategy, especially with regard to bringing together the PMIS, CMMi and EVM. The Meril-De model (Olivier, 2015) in Figure 2.12 shows how strategic planning can create multiple project initiators, which can be grouped by strategic direction and also utilizing single projects, projects in a program, or projects in a portfolio.

Any approved Project Charter and Business Case would be scanned in, saved, and secured in the document repository of the PMIS. As documents were scanned in high resolution only, signatures on the document would be legible. These project documents would contain sufficient detail to pass audit, including the stakeholder matrix and the high-level project approach from a Knowledge Area and Planning Process Group perspective, signalling the initiation of the project as an output from the Initiation Process Group of the PMBOK was ready in the PMIS emplacement. Numerous projects will be tasked, and each approval for a new project would also install an accountable project team within a new project instance created on the PMIS emplacement under the auspices, control, and authority of the appointed project manager.

Projects on PMIS restricted by resource availability

The PMIS emplacement, with a database that can be scaled in size and capacity, is able to accommodate a large number of projects. The number of projects would be limited to the number of certified project managers and teams that are brought to bear. This was an important feature as it ensured that projects were self-contained entities, which should be self-sustaining; important from a Theory of Constraints perspective as discussed in Section 2.16.2.

PMIS manages resource over allocation

The PMIS emplacement, with a database that can be scaled in size and capacity, is able to accommodate a large number of projects. The number of projects would be limited to the number of certified project managers and teams that are brought to bear. This was an important feature as it ensured that projects were self-contained entities, which should be self-sustaining; important from a Theory of Constraints perspective as discussed in Section 2.16.2**.**

If the EPMO would want to use developers with unique skills on multiple projects, this could be allowed, as long as their time was negotiated, approved, and then scheduled by the project managers responsible. To this end, it is worth noting that the PMIS emplacement works on the basis that any resource can schedule their own time in hours against any project they are allowed to transact on. However, final approval of the time allocation, including time allocated for a task that is being extended, ultimately fell to the project manager. A project manager, who is on top of their project and who knows that a human resource is focused on an important task and must not be distracted, will negotiate in the best interest of their project and its ultimate success.

EPMO authorizes projects and resources

The creation of new projects on the PMIS emplacement would ultimately be managed centrally by the EPMO head. The EPMO would be in a position at any time to draw reports on all the projects that were created on the PMIS emplacement and view their progress from a PRINCE2 method (or other methods, PRINCE2 Agile, Scrum, PMBOK, etc., if needed), and by project phase or other dimensions.

Correct access rights on the PMIS were carefully set up to ensure that the certified project manager had the rights and associated functionality to run their project within the PMIS emplacement. Templates available to the PMIS administrator allow them to define the project manager role with the correct rights needed on the PMIS emplacement to do their work. This template took a few seconds to apply. Developer rights were similarly set up and applied, as were other groups with associated and appropriate rights that were needed. An example could be a business analysis group that would have different access rights to the PMIS emplacement to those of the project manager or developer group.

When the project manager allocates project resources to their project within the PMIS, this activity will be done by following either PMBOK, PRINCE2, or other method project processes. Using the PMBOK as a standard of which processes were typically needed, meant that regardless of the project method, clear steps for a typical project was always available. For the project manager, it was a simple exercise of checking across the PMBOK matrix from process #4.1 to #4.7 from a project process group perspective and also from process 4.1 down to 13.6, if Knowledge Areas needed to be considered, applied to process groups, as is illustrated in Annexure S. Naturally, if the project manager followed a particular method (PRINCE2 Agile, etc.), these processes would take precedence and be followed, but always keeping in mind the overall project management process required for a well-run project or phase as illustrated in the PMBOK matrix, per Annexure S.

Resources were allocated electronically from the resource module into the project within the PMIS so that a preliminary schedule was converted into a developed schedule, ready for movement into execution, #6.5 moving into #4.3 of Annexure S. In this way, if a developer added to a project, the project manager would see the project in their PMIS program on their personal computer either networked on a local area network or via a browser view to an instance of the PMIS emplacement in the cloud. Logged onto the system and having access to a project, they could open up the project and join the team. As the project manager was ultimately in charge of their project, the actual method followed when working through the Project Process Groups from Initiate to Plan to Execute was left up to them. Ultimately, when the project manager was ready, the project would be baselined, the preliminary schedule would be promoted to a final schedule, and then moved into execution (Executing Process Group). As Agile operates differently than Traditional Project Management, it follows pretty much the same process groups as can be seen from Annexure O Table O.1.

It is evident that Steps 1 to 3 created a controlled environment within the PMIS emplacement that facilitated tighter project management process implementation and control. This allowed project managers and project teams to be more successful in the projects they run as opposed to trying to run projects without the support of a PMIS emplacement.

As project management is also about good governance, step 4 facilitated the PMIS emplacement used as a project artefacts repository wherein process outputs could be stored as evidence. From the researcher’s experience, being able to pass a project audit is very important, as this check from outside the project could confirm the project is on track and in good health. If the audit is based on sound auditing procedures, this can give the project manager and team assurance that their actions are compliant with auditing controls and the project is being well-managed.

5.4.4 Step 4: PMIS is a secure repository for all project artifacts

4.	PMIS is secure repository for all project artifacts.	No	Yes (1)

Figure 5.4: PMIS CM Improvement Framework Step 4

Source: Synthesized by the researcher

Secure PMIS project artifacts repository

The resource module needs to be secured and have qualified staff working on it. Correspondingly, the PMIS Document module will need similar controls. For the PMIS emplacement, the document repository would be controlled by the EPMO from an access and administration perspective. As the agency was fastidious about governance, the document repository extended beyond the EPMO and encompassed the agency as a whole. However, as the document repository was integral to the PMIS emplacement, each project had its own dedicated repository, which was ringfenced from other projects by folder structure managed via access control.

Document persistence & version control

Document versioning controls were set up centrally so that any document added to the document repository needed to be approved by the project manager. In addition, once approved, the document could only be signed out and changed as a copy of the original version. This way, an audit trail would be maintained for all items added to the document repository (by whom and when) and edited (by whom and when). Documents no longer required would be archived, becoming invisible to users, but still available if required from within the system. The decision to archive a document or not was made by the project manager for each project.

The ability to only work on the edited version of project documents meant the EPMO always had control over project artifacts from the perspective of oversight, accountability, and good governance.

Code persistence & Versioning

The PMIS would also track code changes and code updates using a code versioning system. These systems allowed code to be checked out by a programmer (to a branch) and then checked in later with the changed code. Ultimately, on an ongoing basis, but before a program was finally compiled, all code branches would need to be checked in and the trunk would need to be compiled into the latest version of software, which would be secured in the definitive media library, also under version control. Updated code could be reverted to a previous version of the code, should this be required.

V-Model for Software Engineering (to mind the gap)

If the V-Model in Figure 2.10 is considered, being able to handle Requirements Management (REQM), Verification (VER), Configuration Management (CM), Bidirectional traceability of requirements, and scope lockdown (REQM), etc. professionally, it becomes evident that a PMIS emplacement with the capability to store artifacts for project management purposes as project process outputs and audit purposes was of crucial importance. By the same token, being able to test against requirements meant that testing and Validation (VAL) by the client were also enabled by the PMIS emplacement. All of the above means that CMMi's Product and Process Quality Assurance (PPQA) process was now achievable. Being able to deploy software, releasing it to development for internal testing initially, then to User Acceptance for User Acceptance Testing and finally into Production could all be facilitated by the PMIS emplacement, ultimately under the control of the project manager. This ensured that moving into and down and then up and out of the V-Model, systematic next steps of software could be carefully managed from a quality perspective. Bidirectional traceability or requirements and Configuration Management meant tight control of what enhancements would be added to the code base, when this would be done and how. This, carefully managed on the PMIS by the project manager and team, would ensure safe passage through the "tar pit."

Project Management Method and Processes

If the PMBOK, or PRINCE2, or PRINCE2 agile, or even Scrum Body of Knowledge methods were used, then outputs of each project process in the PMBOK Matrix Annexure S would clearly state what artifacts were required for that process. This meant that, as the project moved through its lifecycle, key artifacts to be produced could be stored and secured. Using the PMBOK matrix and detail in the PMBOK empowered the project manager to know what should be produced irrespective of the project method.

The PMBOK Matrix, Annexure S, shows typical project artifacts for each Process Group, specific artifacts for PRINCE2 or PRINCE2 Agile or Scrum can also be tracked, often producing a similar artifact. The Project Charter and the Business Case previously mentioned are examples of such a document. From a good governance perspective, projects cannot be created by the EPMO unless the project has been officially authorised by its stakeholders (PMBOK processes 4.1 and 13.1), which typically would require an approved Project Charter document or Business Case. Also, a schedule ought not to be worked on if a project plan (PMBOK process #4.2) has not been approved with artifacts stored in the document repository. Using the Knowledge Areas and Process Groups of the PMBOK ensures that a budget is allocated for the project, staff are appointed and allocated to the project schedule before it can be moved into Execution, and other auditable project controls are carefully and well managed. If traditional project management is often criticised because it follows a process and it is slow, possibly going slowly but surely is a good approach, especially as complex software systems may be the focus of work.

Agile Methodology

For an Agile project, utilising the PRINCE2 Agile or Scrum Methodology approved User Stories before the sprint started and a Sprint report and Retrospective report at the end of each iteration should comprise core project artifacts to be saved to the project repository regularly (sprint by sprint). As Scrum projects typically run with limited planning, it would be important that working software is produced at the end of each sprint. Approvals for the next versions of working software can be signed off, and these should be secured in the project repository of the PMIS. These version-controlled software products would need to be tracked from a Continuous Integration and Continuous Deployment perspective, from Development to Testing, User Acceptance Testing, and finally to Production. This should facilitate the installation of ITIL and DevOps per Sections 2.10.1 and 2.10.2, as well as COBIT processes.

Step 4 enabled the ability on the PMIS emplacement to manage the storage of project artifacts. This evidence could be used to support delivery milestones as auditable outputs from project processes. The ability to use project processes at CM L2, and especially the focus on a process as a production engine to turn inputs via transformation into measurable and storable outputs, meant that by following any project method and focusing on its process's outputs, these could be managed to improve productivity and also assist with the delivery of value and quality. Outputs that can be measured can be improved. This means that a project manager and team who are able to focus on the creation of artifacts to store as evidence can do so, improving incrementally, thereby increasing productivity and quality outputs more and more over time.

5.4.5 Step 5: Stage Gate capability available on the PMIS

5.	Stage Gate capability is available to Project Manager on the PMIS.	No	Yes (1)

Figure 5.5: PMIS CM Improvement Framework Step 5

Source: Synthesized by the researcher

Quality Control via Stage Gates

The project manager would be able to configure the Project Schedule on the PMIS emplacement to have Stage or Quality Gate selectors. This means that only when all outputs for the Initiation Process Groups are in place and saved to the project repository can the Project Schedule be moved into the Planning Process Group. In addition, only the project manager for each project would have the ability to open the next gate along the project lifecycle. The Stage or Quality Gate selector would be managed by the project manager and opened to the next Process Group, only if it is suitable to do this. The example of an approved Charter document would be sufficient for the project manager to allow passage into the next process group or process. This works in reverse as well because the project manager, expecting sign off of a Charter can monitor the length of time taken to deliver the output, and if delayed, can start to escalate Risks and Issues on the PMIS which, via collaboration features, means the EPMO head and other stakeholders who have access to the system are immediately notified, and if they must do work to approve the output, they can also be tasked.

While it may appear that quality gates work against the ability to be agile, this is not the case. Often projects run into challenges simply because they were not fully approved or initiated but are already burning budget in the Execution Process Group. The same may apply to the Planning Process Group, for which numerous project artifacts may be needed for Requirements Definition. These would include the sub plans for Scope, Cost, Time, Stakeholder Management and others, before a Project Plan (PMBOK process #4.2) or Project Schedule (PMBOK process #6.5) can be moved into Execution.

User Acceptance Testing

When the project is moving out of the V-Model, Figure 2.10, the requirement to test and have the client sign off new software features in a new software version in UAT means that development is not completed until it is found to deliver value and complies with the client's requirements. The need for these signed-off documents to be secured in the project document repository is self-evident.

Often in agile, planning is very limited, or no plan is available, and the team is operating on the basis of simply coding "what comes next". If this is the case, the ability to develop correct software that can be inspected and adapted would be very important from a delivery perspective and risk management perspective. In this case, the quality gate is the approval of the demonstration by the Product Owner as a means to obtain further funding for the agile project.

Step 5 showed that quality or stage gates could ensure that projects delivered what they were set up to do and within timeframes that were cost-effective from an Earned Value Management perspective. Sign-off by business meant that value was being delivered. Negative variances produced by the PMIS emplacement, where a Stage Gate was not opened when it should have been due to delays, could be a way to focus on areas of the project that were delayed or needed to be more efficient. Using quality and stage gates tactically, the project manager and team were able to better manage project delivery, systematically producing and storing quality outputs.

5.4.6 Step 6: PMIS facilitates Virtual Team Collaboration

6.	PMIS facilitates "Virtual Team Collaboration" from anywhere connected to the Internet.	No	Yes (1)

Figure 5.6: PMIS CM Improvement Framework Step 6

Source: Synthesized by the researcher

Collaboration through Connectivity

Any resource in the EPMO resource pool, allocated to a project by the project manager, will be configured with appropriate rights to work on the project schedule and other PMIS modules, either via a local area network, wide area network or from anywhere accessible globally via the internet.

The fact that all resources are attached to and collaborating on the project schedule, wherever they are globally, means that project tasks can be managed in real-time. Tasks can even be transferred between resources. This facilitates game theory because tasks, like a rugby ball, can be moved tactically towards goal lines.

Step 6 assisted the project manager and team, as tasks needing attention and tasks that started to fall behind due to failure to deliver against original estimations could be addressed in real-time as soon as the team became aware of non-delivery. Because tasking on a schedule on the PMIS emplacement could also be handled from anywhere connected to the internet, tasks could be based on traditional or agile project methodologies. The need to get task approval on the PMIS emplacement, as completed and approved by the project manager, resulted in a sharpened focus on productivity and quality to ensure the agreed scope arrived on time and within budget, regardless of the project method used.

5.4.7 Step 7: PMIS schedule as Gantt chart with task slippage

7.	**PMIS project schedules viewed as Gantt charts with task slippage indication.**	**No**	**Yes (1)**

Figure 5.7: PMIS CM Improvement Framework Step 7

Source: Synthesized by the researcher

Gantt Charts with Task Slippage Indication

The Project Schedule is presented as a Gantt chart with Task slippage indicator capability, which is offered by many of the PMIS. Figure 1.2 shows this functionality. Task slippage is typically a setting that will be switched on in the PMIS. The developed schedule is a more detailed representation of tasked work than the agile board. This would be a preferred method, as the agile board will not be able to handle project constraints often referred to as the dimensions that make up the Iron Triangle, Figure 1.1. The slippage functionality that is found in the Gantt chart is not readily available on an agile board. Slippage of tasks in the developed schedule would be relatively easy for a project manager or team to observe, owing to the graphical nature of the slipping task when compared to its position against initial dates. With the ability to view the problem, the team can rapidly respond and keep the project on track.

Step 7 resulted in project teams using the Gantt chart with slipping feature to achieve far tighter control over their schedule. This, in turn, resulted in higher production, with projects being completed quicker and with less drift away from the original plans, as was previously found to be the case without the PMIS emplacement.

5.4.8 Step 8: PMIS task variance against schedule baseline

8.	PMIS offers variance tracking of scheduled tasks as actuals against schedule baseline.	No	Yes (1)

Figure 5.8: PMIS CM Improvement Framework Step 8

Source: Synthesized by the researcher

Baseline vs Actual

Variance tracking of actuals is available on the PMIS at task level from the PMIS emplacement, derived against the saved projects estimated schedule. The baseline was saved by the project manager when the project schedule was moved from the Planning Process Group into the Execution Process Group. This is illustrated graphically as a series of slip lines that can be viewed by any member of the project team when they utilise this view. The information as variances between actual and baseline enables several indicators as telemetry indicating project status and health.

Visual representations and feedback for Step 7 and Step 8 facilitated the use of the project schedule as a playing field from a game theory perspective, Figure 2.8. This resulted in projects being managed far more tightly, with many able to deliver results that were closer to originally planned time scales, closer to original budgets and planned scope. With less variance between the plan or project baseline and the actual delivery of the tasks on the developed schedule, the team and project manager felt they were able to deliver value to stakeholders. This ability to deliver and win against the original game plan caused team energy and focus to increase. In time, this ability to produce could be enhanced as the team found better and smarter ways to work together.

5.4.9 Step 9: PMIS projects driven off EPMO financial cycle

9.	PMIS projects are driven by the EPMO financial reporting cycle.	No	Yes (1)

Figure 5.9: PMIS CM Improvement Framework Step 9

Source: Synthesized by the researcher

EPMO Financial Reporting Cycle

The EPMO and PMIS emplacement both work off a regular project reporting cycle. Typically, this is tied to the actual versus plan across the dimensions of the Iron Triangle, Figure 1.1, of which project budget allocated and its use to produce value during the project lifecycle is a crucial metric. Project reporting cycles ought to be the same as financial reporting cycles. The reporting cycle tie in the start of the project with the start of the budget allocation against the project, from the perspective of the agency's accounting department. Regular reporting cycles are then used to display project status from an Iron Triangle, Figure 1.1, perspective, and this can be regularly monitored and controlled for good governance. If a reporting cycle is maintained, status reports can be produced regularly to show project status and health against the original plan and budget. These are available in real-time on the PMIS emplacement, and if produced to meet the requirements of the EPMO, they will form the basis of an audit of project health imposed by the EPMO on the project manager and team. It stands to reason that usage of the project budget should bring about value earned for the budget used, and tight management is essential to ensure that projects deliver value on this basis. Without the PMIS emplacement, it is doubtful that a project team will be able to manage the complexity of this while also working on achieving the project tasks.

Project Reporting Transparency

Top-level EPMO summary reports are available from the PMIS by a program (groups of projects on the PMIS) or by a single project with drill-down capability down to task level. Reports are typically made available to project boards (a PRINCE2 requirement), key sponsors (and invited stakeholders). As projects become late "a day at a time", reports follow a weekly, week-by-week publication cycle.

Step 9 required a moving together of EPMO and agency finance departments to achieve synchronisation between agency accounting budgets and those on the PMIS emplacement. This resulted in budgets that could be applied with higher certainty and be managed more effectively, being the same in the EPMO and the agency finance department. The ability of the agency to quote for project work was also improved, as projects could be brought in on time and within budget more effectively and efficiently than before the use of the PMIS emplacement.

The application of Step 9 of the PMIS CM Improvement Framework indicated some shortcomings which required tighter executive coordination between the EPMO and the finance department. To this end, accountable executives from the EPMO (the EPMO head) and the finance department head were found to be strategic drivers for success. This was rectified in the enhanced framework found at the end of this Chapter in Section 5.5 and Annexure U.

5.4.10 Step 10: Earned Value Management off the PMIS

10.	**Earned Value Management (and other) project status graphics are available on the PMIS.**	**No**	**Yes (1)**

Figure 5.10: PMIS CM Improvement Framework Step 10

Source: Synthesized by the researcher

Earned Value Management Graphics

Earned Value Management Analysis (and other) project status graphics, reports, etc., are available on the PMIS. Figure 2.9 shows the Earned Value Management graph. This is a standard graph used by Project Management Professionals. From a PMBOK (or PRINCE2) perspective, Earned Value Management includes all project formulas and especially the "To Complete Performance Index" as one number for project health throughout its lifecycle. Earned Value Management, built out of tasks and their status, is an essential PMIS reporting product. This facilitates game theory by using the project schedule as a playing field in order for teams to focus on excellence, as illustrated by Figure 2.8.

While modification may be needed if the PMIS is to work more closely with a financial system, it is not difficult, as all PMIS schedules can easily be configured to monitor financial information at task level. In this regard, Step 10 implemented well on the PMIS emplacement resulted in the project manager and team being able to monitor and control their projects far better than without the use of the PMIS to support EVM metrics.

Step 10 elicits a shortcoming in the framework with respect to a financial system, yet the enhanced version in Table 7, **Section 5.5**, and in **Annexure U** remedied the situation.

5.4.11 Step 11: Risk and Issue management off the PMIS

11.	**Risk and Issue management module on the PMIS is available to all to protect the projects (schedules).**	**No**	**Yes (1)**

Figure 5.11: PMIS CM Improvement Framework Step 11

Source: Synthesized by the researcher

Risk and Issue management

Issues are risks happening. Serious issues, which a project team is unable to solve, need to be escalated to the Project Board Executive (PRINCE2) or project sponsor who are also members of the PMIS resource pool. As all in the project team are aware of the risks and issues raised, team collaboration on the project includes visibility of risk and issues in the system. In addition, the project manager, or appropriate member of the project team, can pick up the risk or issue and schedule actions to resolve it as soon as possible, thus avoiding problems for the project. It is the researcher's experience, from over two decades of project management, that a reason for project failure is the inability of the executive to step in and mitigate risks and issues when the team is unable to do so. To this end, if a project gets stuck and requires assistance, the team, as workers at the worker level, is unable to solve the problem, the solution to unblock the project resides with the executive.

Therefore, using Step 11 accountability for risks or issues that the team cannot solve must be moved up to the level of the executive where it should reside to unblock the project. Or the executive, utilising performance metrics on the project from the PMIS, may decide not to mitigate the risk and rather to terminate the project; this decision lies in their domain.

Step 11 facilitated the use of the PMIS emplacement to raise and promptly manage risks and issues. If the risks and issues could not be solved by the team, they became the problem of the executive, which ultimately was where they would need to remain and be resolved.

5.4.12 Step 12: PMIS Resource Pool to focus skills improvement

12.	The PMIS Resource Pool is a focus of organizational attention towards skills improvement for excellence.	No	Yes (1)

Figure 5.12: PMIS CM Improvement Framework Step 12

Source: Synthesized by the researcher

People Capability Maturity Model

Under the auspices of the EPMO head, the PMIS Resource Pool is a focus of organisational attention towards people and the improvement of their skills for excellence. People excellence is dealt with in People Capability Maturity Model in Annexure H.

The goal in project management should be the creation of Empowered Teams, which is a People CMM L4 process. To achieve such a goal requires an expenditure on training for key roles in the team. As per Example 4.4 (American football case) from Chapter 4, Section 4.5, only the best team members can ensure a Super Bowl win. Anyone on the team who is not pulling their weight will not remain a valued member of the team for long.

The ability of Step 12 to coordinate and develop professional human resources allowed project managers to form teams that were capable of high productivity and were able to deliver quality. The application of Step 12 of the PMIS CM Improvement Framework showed that a close relationship between the EPMO and Human Resources Department was required. This finding identified shortcomings in the original PMIS CM Improvement Framework, which were identified and rectified in the enhanced framework found at the end of this Chapter in Section 5.5 and Annexure U.

5.5 **Amended PMIS CM Improvement Framework**

After working through the PMIC CM Improvement Framework (Table 4.1) steps, the researcher has applied improvements to the original table in a revised Table 5.1 . The table is reproduced in large size in Annexure U as Table U.1.

Table 5.1: Revised PMIC CM Improvement Framework

PMIS CM Improvement Framework

12.	b) The PMIS Resource Pool is a focus of organizational attention towards skills improvement for excellence. a) The Human Resources Head accepts key stakeholder role to ensure PMIS emplacement Resource Pool is focus of organizational attention towards skills improvement for excellence.	No	Yes (1/2) Yes (1/2)
11.	Risk and Issue management module on the PMIS is available to all to protect the projects (schedules).	No	Yes (1)
10.	b) Earned Value Management (and other) project status graphics are available on the PMIS. a) Finance Department Head accepts key stakeholder role to ensure PMIS emplacement reflects accurate financial position in respect of projects financials.	No	Yes (1)
9.	b) PMIS projects are driven by the EPMO financial reporting cycle. a) Finance Department Head accepts key stakeholder role to ensure PMIS emplacement reflects accurate financial position in respect of projects financials.	No	Yes (1/2) Yes (1/2)
8.	PMIS offers variance tracking of scheduled tasks as actuals against schedule baseline.	No	Yes (1)
7.	PMIS project schedules viewed as Gantt charts with task slippage indication.	No	Yes (1)
6.	PMIS facilitates "Virtual Team Collaboration" from anywhere connected to the Internet.	No	Yes (1)
5.	Stage Gate capability is available to Project Manager on the PMIS.	No	Yes (1)
4.	PMIS is secure repository for all project artifacts.	No	Yes (1)
3.	Project Managers authorised by EPMO to run PMIS Projects.	No	Yes (1)
2.	Project resourcing driven off PMIS Resource Module.	No	Yes (1)
1.	b) EPMO driven off PMIS Emplacement. a) EPMO Head accepts PMIS emplacement project key stakeholder role.	No	Yes (1/2) Yes (1/2)
➔	CM Level 1		
➔	PMIS for CM Level 2+ (12 Improvement Dimensions)		

How to use the PMIS CM Improvement Framework:

1. Enter from the bottom.
2. Award 1 point for each Yes if Improvement Dimension is in place and operating well.
3. As you proceed a No answer may not appear underneath a step you have answered with a Yes.

Source: Compiled by the researcher

Table 5.1 has been improved to rectify some of the problems and feedback that were found during the research process.

Executive Involvement, Oversight and Accountability:

During the case study, it was found that dedicated executive involvement in phase 1 of the project was important for its success. This executive role, which is understood to be a project key stakeholder and project champion role, means that the executive works closely with the team to identify and unblock any problems the team may have during the project. The Deming quote mentioned in Chapter 2.11 explains that "only management can change the system, and it is the system that must be fixed and not the workers."

In this table, step 1 elicited the EPMO head accepting the PMIS emplacement project in a key stakeholder and project champion role. Steps 9 and 10 identified the head of finance to be a key stakeholder and project champion role. For step 12, the head of Human Resources was appointed in a key stakeholder and project champion role.

Shortcomings in the PMIS CM Improvement Framework were therefore linked directly to a lack of project champions as key stakeholders in the areas of the EPMO head, head of Finance and head of Human Resources.

Code world:

Another area that came to light was the feedback received from developers. Many developers the researcher spoke to indicated that they were simply not interested in project method or governance. All they wanted to do was write software and solve problems in what many referred to as "code world." Feedback also indicated that while they understood that their pay check was conditional on them following project process (making sure they work on clear business requirements, attend project meetings or scrum standuppers etc.) this was often viewed as an irritation that dragged them away from where they needed to be – in code world writing code. One top developer on a project the researcher was managing explained that he saw code world as a triangle with a point on top and with the base of the triangle being code world, where software engineers and developers worked. Project governance, project management, executive concern was viewed as residing in the base of another (inverted) triangle above the developer triangle with points touching. I.e. the base of the upper triangle was where business and the executives operated while the base of the lower triangle was the developer and software engineer's domain and concern.

The researcher, noting again the concern in Chapter 2 from Steven Rakitin (2001), that the Agile Manifesto core principles served only as easy excuses not to do the difficult work and with "hacker interpretations" there would be no understanding of, or adherence to essential engineering and programming disciplines. And according to Rakitin, the essence of following the items on the left of the Agile Manifesto means "Great! Now I have a reason to avoid planning and to just code up whatever comes next." Also, it is the researchers experience that developers often do not want transparency and would prefer to operate with impunity in the bottom triangle, essentially fearful that transparency will bring unwanted attention from above down upon them, which was an irritation, side tracking them and getting in their way. In scrum over the past 5 years the researcher has often become aware that quality in Feature and User Story creation is lacking from a CM L2+ perspective, simply because developers would rather put in a high level "placeholder" description to placate the burndown and then do their own thing, and go their own way.

With the above in mind, and also noting that CM L2+ behaviour must be more favourable than operating at CM L1, it would appear that tighter control is required from business (the executive) to make sure that the projects aim to deliver value as quickly as possible is met. As buy in from the developers is critical to ensure transparency and value delivery, a process-based solution at CM L2+ in scrum would mean that a User Story is clearly Verified before a sprint starts by the Product Owner. Thereafter developers (with clear visibility of the Product Owners Acceptance Criteria on the User Story) can be allowed to say **How** and **How Much** will be allowed onto their Sprint Backlog for the next sprint iteration. The Product Owner, working off their Product Backlog, would only be allowed to say **What** is the most important next requirement (User Story) and **By When** they would like this handled by the development team. This essential segregation between Product Owner (Business) and Developers (Information Technology) is explained by Cohn (2004:172 and 27) who understood that user stories can be used to facilitate software which must be "developed opportunistically" and "if either side dominates these communications, the project loses." And finally, where the scrum process Demonstrate and Validate requires written approval of Done stories, these are approved and signed off by the Product Owner at the end of each sprint in a formal manner. In this regard a signed off demo deck as core scrum artifact produced at the end of each sprint could be considered.

The revised table 5.1 has been reengineered by the researcher to ensure that the above new observations are catered for.

5.6 Case Study Conclusion

Phase 1 of the project focused specifically on installing the 12 steps of the PMIS CM Improvement Framework. The goal of phase 1 was to have all 12 steps installed and approved. Having phase 1 of the project completed should put the agency at CM L2+, with some processes from L3 also installed. This means that processes available for

CM L2 are available and operating stably. In addition, some L3 processes are also installed, and these are also operating stably. In this regard, the following processes are installed by the PMIS CM Improvement Framework: PP, PMC, REQM, PPQA, MA, SAM, RSKM, VER and VAL.

The next project phases, not defined as yet, would involve how the agency would move itself along the CMMi path from CM Level 2 up to CM Level 5. As has been mentioned, this was estimated to take between three to five years to achieve. However, if the agency was able to install the PMIS emplacement nationally along the lines stipulated by the PMIS CM Improvement Framework, and this was completed successfully in 12 months, without having to use the additional 12-month buffer, there was a good possibility that CMMi processes could be installed fully in less than five years. In this regard, the agency has exhibited agility and maturity at installing the necessary capabilities required for CMMi compliance. While a SCAMPI accreditation from the CMMi was available for the agency for phase 1, if they were interested, it was decided that SCAMPI was not the goal.

Utilising the 12 PMIS CM improvement steps, the agency was able to successfully install the PMIS emplacement within a year. The system was found to have been set up in such a manner that empowered teams to run projects correctly and successfully from a CM L2 perspective. Owing to the existence and functionality offered, the PMIS system was relatively easy to audit. In addition, knowing that an audit would be required, it was found that projects were well run by teams within the EPMO as they were aware of the audit requirements and ensured they were compliant. The way the PMIS emplacement was set up ensured that full accountability of the PMIS emplacement project was owned by the EPMO head, yet operational and managerial control was delegated to the project managers and project teams. Since the PMIS emplacement empowered everyone in the EPMO to achieve their goals, from a game’s theory perspective, all staff appreciated what was required in order to be successful and responsive; they ensured the initiative was a success.

Ultimately, if a CMMi SCAMPI was required by the EPMO, this accreditation was achievable because the PMIS emplacement set up according to the PMIS CM Improvement Framework facilitated process compliancy at the standards demanded by the CMMi for CM L2 SCAMPI compliancy. The Head of the EPMO decided to put the agency forward, and they were awarded International CMMi SCAMPI certification status within four months. The Ministry was satisfied with the quality improvements that had been achieved, and this was reinforced with positive feedback from clients. The EPMO goal for the following year would be to move more fully into the installation of Software Engineering, requiring the inclusion and stabilising of remaining processes at CM L3.

5.7 **Summary**

The aim of Chapter 5 was to apply the PMIS CM Improvement Framework to a case study. This was undertaken by applying each step systematically and in order from the start of the Framework at step 1, moving up to step 12. It does appear that the PMIS CM Improvement Framework assisted the agency in its goal to install CM L2 in 12 months. In addition, it appears that by applying the steps of the PMIS CM Improvement Framework in order and ensuring that earlier steps were completed satisfactorily, before starting on next steps, implementation success was assured.

From step 1, which established the PMIS emplacement firmly within the control of the EPMO, resources, with emphasis on the project manager and project team, were empowered in steps 2 and 3 to run project processes correctly using the PMIS developed schedule as their agreed playing field. In step 4, the project artifacts were secured within the emplacement, allowing the project teams to start building value and show that this was achieved via proof in the form of saved and auditable artifacts. Steps 5, 6, 7 and 8 facilitated the use of stage gates and baseline versus actuals in tasking to see slippage and other variances graphically. This allowed collaborating teams to focus on tasks on project schedules as a field of play in a game, where proficiency and winning was the focus, based on Measurement and Analysis (CM L2 process MA). Steps 9 and 10 allowed for closer liaison with projects finance through the implementation of the Earned Value Management graphs, which would allow teams to take winning to a new level of expertise, allowing projects to be brought in on time and within budget. Risks and issues were navigated in step 11, and professionalism in human resourcing was facilitated in step 12. If the EPMO were so inclined, with the 12 steps installed, they could start to run projects on the basis of financial reward for compliance or achievement within the plan. Financial rewards for excellence could create highly motivated professional teams focused on rapid value creation.

5.8 Conclusion

The PMIS CM Improvement Framework, if followed, should offer a systematic approach towards the installation of a PMIS emplacement so that it supports and empowers process focus and application at CM L2. In addition to CM L2 processes, which are fundamentally project management processes, the PMIS CM Improvement Framework facilitated the installation of some processes at CM L3, which included verification, validation, and risk management. With the stability offered by the PMIS emplacement operating at CM L2, it should be a simple matter to continue installing other CM processes with projects that are designed for and aimed at the attainment of processes at CM L3, L4 and finally L5.

In a similar approach that has been taken to install CMMi for Development 1.3 processes, so too can the other CMMi constellations for Procurement and Services with their processes also be installed easily. Gains can be lost if they are not constantly upheld. However, having a PMIS emplacement stable and operating should assist in firming up gains, thus creating a stable base to achieve higher CM levels; if this is a goal. Ultimately, the essential or core processes presented in this work should be more than adequate to suit the needs of most entities, be they agencies operating in the Public Sector or businesses operating in the Private Sector.

Chapter 6 aims to answer the research questions presented in Section 1.10 and to address the research objectives in Section 1.11. based on the observations of the Literature Review in Chapter 2 as well as the findings from the case study in this chapter. Chapter 6 will conclude with final observations as well as ideas for future work.

CHAPTER 6:
CONCLUSIONS AND FUTURE WORK

6.1 Introduction

In the preceding chapters, evidence has been gathered to answer the research questions in Section 1.9. The research objectives in Section 1.10 can now also be addressed. In addition, observations from the application of the PMIS CM Improvement Framework in the case study in Chapter 5 can also be used as findings.

This chapter will now investigate each of the research questions and the research objectives to propose answers for consideration. Section 6.2 addresses each of the Research Questions (RQs) from Section 1.9, and each Research Objectives (ROs) from Section 1.10 in turn, and will consider the extent to which the objective has been met (RAs) in the dissertation. Chapter 6 will conclude with ideas for future work.

6.2 Answering Research Questions

Research Question (RQ)1: What is the status quo with respect to Software Project Management in Agile?

Research Answer (RA)1: Due to the complexity of the medium, software projects are often difficult to manage and run successfully. The discussion in Chapter 2 focused upon the problem of successfully managing software projects without becoming mired in the "tar pit." To this end, the finding of this dissertation is that a step-by-step or stepwise manner, moving carefully from a controlled baseline of the latest version of valuable software to the next version of more valuable software could be a good way to proceed. This approach is followed when working through the PMIS CM Improvement Frameworks steps as a next step cannot be taken until a current step qualifies. The theoretical case study and application of the framework showed by example how the framework is intended to work.

The ITIL processes find that it is best practice to produce a version-controlled consolidation of the latest code into a Definitive Media Library (DML) as a definitive step in Continuous Integration (CI). Extreme Programming (XP), the Agile method for software development, also asks for this, in what is referred to as the 10-minute build. Once code is consolidated into a version-controlled DML, which can then be deployed into “next” environments along a deployment path, which typically consist of a testing and Quality Assurance (AQ) environment, a User Acceptance Testing (UAT) environment, and after signing off, software can be deployed into production. Deployed correctly with supporting production metrics means the other side of Continuous Integration, Continuous Deployment, or even Continuous Delivery production improvement goals are achievable.

The fact that Agile is designed to work in short incremental iterations means that long term planning can be avoided. If the way forward is not clear, this means that Agile well-run should be flexible and able to learn quickly, changing direction towards the goal of customer or Product Owner-approved working software that delivers more value than could have been produced with a more traditional project management approach.

The researcher’s Dimensions Model of Software Project Management in Annexure A explains that with Agile there could be risks associated with insufficient planning. Risk, the model surmises, is inversely proportional to the amount of planning undertaken. According to the model, risk increases if planning decreases. Alternatively, if a project methodology is followed that reduces the amount of planning in relation to the complexity of the software project goal, more and tighter risk management will be required. This finding is borne out by Wysocki’s scope triangle in Figure 2.19, and the effect appears primarily due to the fact that the classic project constraints and controls are no longer being actively applied.

It would appear from the findings of this dissertation that a smart Agile team should be fully aware that operating with agility could increase risk. In addition, the smart Agile team who are mature and capable of operating above CM L1, and who can apply Agile processes consistently and correctly, and if they are empowered by a PMIS emplacement that facilitates agility in tasking along the lines recommended by the PMIS CM Improvement Framework, should be successful in producing valuable software. The utilisation of a PMIS emplacement can also assist the Agile team to manage their risk and issues faster and more effectively, which should assist their successful and safe journey across the “tar pit.” If Agile is undertaken without due appreciation or consideration of the above dangers, especially if software is complex, an incapable and immature Agile team, not using agile processes well or at all, could experience difficulties. The situation could worsen, culminating over time, if their cost of operation increases above the approved value that is being delivered.

It appears that a Software Development Team who are adhering to the project management processes for their chosen project methodology, applying these correctly at capability maturity level 2, should be more successful than those operating at capability maturity level 1 and do not follow these procedures. This observation that Agile or Traditional Project Management processes must be followed should apply equally if the team is an Agile team following the Scrum methodology or a team that follows more traditional project management processes, in more rigorous and tightly controlled methodologies like, PRINCE2 or PMBOK. If Agile is undertaken without full appreciation of what it takes to deliver quality software at capability maturity level 2, then project failures and criticism may await Agile Project Management, as was found and levied against Traditional Project Management.

The theoretical case study in Chapter 5 illustrates that a PMIS emplacement facilitates the well running of both Agile and Traditional Project Management at CM L2. Correct use of the PMIS Improvement Framework could create the necessary awareness and controls to ensure project management is undertaken with process focus at CM L2 and above, whatever the chosen methodology.

RQ2: To what extent can Agile be successful below Capability Maturity Level 2?

RA2: From the previous discussion, with specific reference to Standish Group chaos reports (2014), it appears that Agile run at CM L1 would not facilitate a return to the basics of project process management, which is urgently required if complex software projects are to be run safely in order to deliver and to produce value.

There appears insufficient research available to support the success of Agile fully. In the researcher's personal experience and observation, supported in some cases by evidence from failed audits undertaken on Agile projects he was involved in during his career, these appear to illustrate that Agile run heroically at CM L1 was found wanting for complex software projects. The researcher concedes that Agile could be very successful with smaller module-based software improvements if requirements are well understood and stable, and the team are developing solutions from a known and version-controlled software baseline. In the researcher's experience, continuous, capable and mature application of Agile processes is not easy to achieve, and the Agile team's ability to sustain quality production, sprint after sprint, may not be assured without full appreciation of the Scrum Master, whose primary role it is to drive in quality from an Agile project process perspective. In addition, the researcher has found that Agile teams at CM L1 are quick to undermine the importance of the Servant Leadership role of the Scrum master and when this happens, team productivity and quality decrease. Agile projects run at CM L1 also appear to create higher levels of risk, rework, and waste, and this could contribute towards the creation of the chaos mentioned in the Standish reports rather than help to remove it.

With recent excitement surrounding Agile methodologies, it does appear that an ability to criticise the Agile approach constructively is limited if run at CM L1. The Dunning Kruger effect shown in Figure 2.5 may help to clarify this problem. In addition, since many project budgets appear to be grouped into annual departmental budgets, it can be difficult to determine on a project-by-project basis how well Agile projects perform and what their actual contribution to Earned Value is when compared to their running costs.

Agile explains that short iterations are used to reduce complexity and rapidly produce value if the Agile method is used correctly. Scrum methodology, as an example of a popular Agile Methodology, with its process #16 (Demonstrate and Validate sprint), clearly explains that value must be demonstrated to and validated and approved by the Product Owner at the end of each sprint. However, it would appear from the researcher's experience that it is rare for Product Owners to agree to sign off that value has been added from the sprint. This essential Product Owner's approval role must continue to guide the Agile team to sustainable levels of higher quality and productivity on an ongoing basis sprint after sprint. Therefore, if the Product Owner does not drive in quality and productivity, it could signify immature application of scrum processes and CM L1 behaviour. This often occurs because user stories for scrum are not compiled or used correctly, which talks to the quality of some planning and estimating processes that need to be managed well, as is illustrated in Annexure O Table O.1. In addition, the delay in moving new software into production means there is often a long delay between software being demonstrated and finally being released and used.

The pressure on Agile teams to deliver value while they are not supported fully from an ITIL or DevOps perspective can slow down official approvals by the Product Owner, who is often reluctant to trust the remainder of the deployment process. Without the Product Owner closely monitoring delivery at the end of the sprint, with formal approvals for correctly done software, the necessary environment needed for the delivery of valuable software may not exist. Therefore, the application of process #16 is a crucial process that must be applied well and rigorously in Scrum to ensure that User Stories created in the Plan and Estimate Phase (processes #7 to #12 Annexure O) are correctly delivered. The research suggests that the ability to approve and release working software rapidly is a crucial metric for success in agile, which is essentially an application of the core CM L2 processes of project planning and project monitoring and control. It is the ability to apply the Scrum principle of Empirical Process Control in the demonstration of value in the form of working software via inspect and adapt processes that truly empowers agility.

Another challenge is that agile software is often built on a developer's laptop, and there are often delays before software is fully deployed into production via a process of continuous integration and continuous deployment. This could undermine the notion of working software moving rapidly into production after each sprint as the risk is that software is not available for use by stakeholders and cannot be validated. If the developer's laptop is damaged, the risk is that software could be lost if it has not been recently checked into a software versioning system. The same applies to the definitive media library; if this is not maintained and version-controlled, developer code can be lost if the development team is not fastidious.

While many other challenges exist where agile projects are concerned, many of which have been identified in the research, it does appear to the researcher that it will be very hard to run agile projects successfully at CM L1. And while some agile projects may be run with some project method processes being utilised, focusing on some while ignoring others will not fully empower CM L2 process application and success. To this end, the example of the scrum team who diligently hold daily standuppers yet rarely produce working software most probably implies they are an undone scrum team operating at CM L1 and not producing at CM L2.

Through the PMIS emplacement, adequate controls can be put into place to ensure that project processes are followed, and project tasks are completed on a developed schedule, which enables successful release management. These tasks, when approved, will lower the risk of non-delivery while helping to secure valuable software into the definitive media library. A PMIS emplacement would also facilitate sign off by the Product Owner, which would occur automatically when working software has been successfully released and signed off by clients as part of User Acceptance Testing.

The Case Study in Chapter 5 illustrates that a PMIS emplacement facilitates the well running of both Agile and Traditional Project Management at CM L2. This should reduce and might fully remove, the potential for chaos that could exist with limited or lack of application of core processes at CM L1.

The ancient fable of Aesop, of the tortoise and the hare, implies that it may be wise to reject the notion of more haste and less control at CM L1. Possibly a slower, steady, and systematic application of correct processes and approaches at CM L2, supported by the power of the PMIS emplacement, could be a better and safer way to traverse the challenges of the 'tar pit.' The V-Model , Figure 2.10, with core CM L2 processes, REQM, PP, PMC, and CM, and VER, VAL and RISK at CM L3 managed tightly on a PMIS emplacement appears to be where effort ought to be focused for an increase in productivity and quality while reducing risk, rework, and waste if agile is to be relied upon to deliver value in complex software projects.

RQ3: To what extent may the use of a Project Management Information System facilitate Software Project Management at Capability Maturity Level 2, thereby enhancing the ICT value proposition for software projects?

RA3: If the Project Management Information System (PMIS) is compared to an accounting or financial system, then the challenge of managing tasks can be compared to the challenge of managing individual financial entries in a double-entry system. If business has evolved to the point where accounting systems are regarded as essential for good governance, then similarly, the PMIS emplacement should also be regarded as essential for project health and good governance throughout a project lifecycle. To this end, if a PMIS emplacement creates transparency by capturing project task inputs and measuring these as progress towards clearly defined project goals, business should become “less obsessed with outcomes of the processes but rather begin to manage the processes for quality.” (Knowles 2011:11).

Complexity in project management occurs when an EPMO needs to manage numerous projects, each with its own resources, budgets and delivery timelines. When considering the fact that Earned Value Management, especially the “To Complete Performance Index,” can be derived in real-time at any point during the project lifecycle from a PMIS emplacement that is set up correctly, it could facilitate Project Management delivery at CM L2, enhancing the ICT value proposition for software projects.

Due to the complexity of managing a software project at task level, it appears that it will be difficult to achieve productivity and quality gains while ensuring stable operation at CM L2 without a PMIS emplacement. Again, complexity could increase exponentially when groups of projects need to be managed as programs of work. If the complexity of running projects correctly against the processes and Knowledge Areas of the PMBOK matrix in Annexure S is considered carefully, the challenge of being able to operate correctly at CM L2 could be challenging if undertaken manually and without a PMIS emplacement. Conversely, if a PMIS emplacement is installed correctly and project processes are being managed at task level on developed schedules by the system, the ability to monitor project progress from an Earned Value Management perspective could be easier.

Graphs of project progress based on Figure 2.9 are the standard approach for good project management according to the PMBOK® 6. This is an urgent requirement from a budgeting and Cost Management perspective. As the planned value and actual value produced is tightly associated with the time and cost to produce value, it stands to reason that tight management using Earned Value Management graphs are essential for project health and success. While a range of metrics is available using Earned Value Management graphs, the “To Complete Performance Index” (TCPI) is a single health metric produced at any time during a project lifecycle. The TCPI metric cannot be produced unless a developed schedule is being used. If projects and project schedules are complex, then it is unlikely that Earned Value metrics can be produced without the PMIS emplacement. Projects that are not producing value according to their plans may need to be stopped, or financial loss could result. Without the Earned

Value Management metrics applied rigorously to measure the actual value being produced, it may not be possible to deliver satisfactory results.

The PMIS emplacement set up correctly according to the PMIS CM Improvement Framework in Chapter 4 must ensure that projects deliver higher productivity and quality with less risk, rework, and waste. The case study in Chapter 5 illustrated the application of the Capability Maturity Model processes, empowered by the power of the PMIS emplacement. The Case study also illustrated, per Deming's assertions in Annexure F and Annexure G, that the executive is ultimately responsible for a project's success. Leaving the responsibility to a project team on the shop floor could be the executive abdicating their responsibility, which may not be done if CM L2 is the goal. If the PMIS CM Improvement Framework is utilised, the executive will be empowered with more control over projects for the delivery of earned value management at CM L2 and above. If the executive accepts that they are ultimately accountable for project success, the PMIS emplacement, with its Risk and Issue module, can facilitate rapid solutions from the tier of management that can solve problems. To this end, the PMIS makes the executive accountable, ensures they are always in the loop and can be tasked to make the tough decisions required to keep projects on track, viable, and healthy.

The ability of the organisation to improve higher levels of productivity, quality, and value should improve as the organisation reaches for higher levels of Capability Maturity, above CM L2. Ultimately, with the continual improvement processes found in CM levels, it may be possible to reach levels of productivity, quality, and excellence that were not available previously. Ultimately, if experience curves, Figure 2.6, are appreciated, those able to implement the PMIS emplacement rapidly should have an excellent advantage over competitors who are unable to meet the challenge.

The essential need for the PMIS emplacement to facilitate process control at CM L2 while a project is in execution is stated previously. CM L2 processes, including Configuration Management (CM), Measurement Analysis (MA), Process and Product Quality Assurance (PPQA), Risk Management (RSKM), Verification (VER), Validation (VAL), and others, can all be run more efficiently, providing data in real-time on a PMIS emplacement. If one can measure something, it is a first step towards improving it. The ability to install, use and benefit from the PMBOK and CMMi processes (including ITIL, COBIT, and others) are facilitated by the PMIS emplacement.

The 12 dimensions in the PMIS CM Improvement Framework in Table 4.1 indicate the areas in ICT that can be enhanced. Some of these include managed resource pools of trained staff, the virtual collaboration of team globally, developed schedules that are managed based on earned value, task slippage and many other information management indicators, a repository for project documentation, quality gates enforced by the system, risk and issue management, etc. Ultimately, the fact that resources can collaborate globally, in real-time and from anywhere connected to the internet offers the project team incredible opportunities to excel at their game.

Owing to project management complexities typically found in software engineering (and other) project types, the main thrust of this research is that without a Project Management Information System (PMIS emplacement), the Project team's ability to adequately monitor and control project management processes for Capability Maturity success at Level 2 (and above) at the task level on a project schedule would be challenging. The Agile (Scrum) whiteboard with post-it notes could be equally constrained. Consequently, project performance, despite good intentions from a talented team, would, without the PMIS emplacement, naturally degrade to what the researcher denotes as a heroic effort at Capability Maturity Level 1.

The value proposition promised is pivotal. This may be achieved is through process-driven ITC better managed on developed schedules (playing fields) on the PMIS emplacement. To this end, the use of a PMIS emplacement to enhance process maturity in software project management is precisely to promote process-driven ICT.

Agile, often viewed as a silver bullet solution (Brooks 1986), if undertaken at CM L1 on complex software projects, would often not be able to produce satisfactory results. Therefore, for the attainment of CM L2 (Managed), a PMIS emplacement is not simply a recommended tool of the 'Direct and Manage Project Work' process #4.3, Annexure S, as is stated by the Project Management Institute. Instead, as argued in this work, it is essential to facilitate project management success for earned value.

Ultimately, within a gradient scale of improvement, a direct relationship appears to exist between the installation (or not) of a PMIS emplacement, the way the PMIS emplacement is installed within an organisation, and an organisation's ability to achieve stability at CM L2 and above.

RO1: Align the Project Management Information System (PMIS) Capability Maturity (CM) Framework with respect to the current status quo regarding Software Project Management in agile.

Based on this dissertation and the application of the theoretical case study, it is the opinion of the researcher that the PMIS CM Improvement Framework is fully aligned with respect to the current status quo regarding Software Project Management in Agile.

In recent years, the Project Management Institute (PMI) has produced its latest PMBOK version 6, which identifies both a project or a project phase (process # 4.6 of the PMBOK dashboard), and both require the application of project processes for success. In this regard, the project phase can also be a 2-week Scrum sprint. The PMI has also produced an Agile Certified Practitioner (ACP) certification, which focuses specifically on Scrum as a preferred Agile Project Management methodology. In support of Scrum and Software Engineering, the ACP focuses specifically on Extreme Programming as the selected approach for software developers who want to implement agile thinking and behaviour. With the PMIS emplacement, the focus is not as much at the level of the project or phase and its duration. Instead, resources commit to complete project tasks to focus on delivering solutions to satisfy user requirements on the developed schedule in the Execution Process Group. In this regard, while methodology and process are still very important, the PMIS offers the project team the ability to task themselves in a way that focuses on delivery as it unfolds, play-by-play, on the playing field. This advantage should bring about real agility no matter the project methodology chosen.

RO2: Determine whether a PMIS CM Improvement Framework can be used below Capability Maturity Level 2.

The researcher has designed the PMIS CM Improvement Framework to move the user rapidly up and out of CM L1. The PMIS CM Improvement Framework stipulates 12 improvement steps that are required to move up into CM L2 behaviour.

In a similar manner that the discipline of accounting processes will require certain actions on behalf of bookkeepers to ensure that their books of account on financial accounting software are kept compliant and in good shape, so too will the PMIS emplacement require project teams to operate in a manner that befits the goals and objectives of quality project management at CM L2. With a firm CM L2 foundation in place, the project team will be able to use the PMIS CM Improvement Framework to install CM L3, 4 and 5 processes, as and when they are required.

RO3: Determine the position at which the PMIS should facilitate Software Project Management at Capability Maturity Levels above 2, thereby enhancing the ICT value proposition for software projects.

The researcher has designed the PMIS CM Improvement Framework to create a firm base of CM L2 processes to be able to aspire towards and install CM L3, 4, and 5 processes as and when they are required. While the PMIS CM Improvement Framework stipulates improvement steps that are required to move up into CM L2 behaviour, some of the uppermost of the 12 steps include CM L3 processes.

For example, the Project Management Information System implemented by the PMIS CM Improvement Framework will facilitate Risk Management. Risk management (RSKM) is found at CM L3. The PMIS emplacement Risk and Issue Management module installed, being able to task the executive to solve problems that the project team are unable to solve, should do much for project success and health. Unlike in projects managed without the PMIS emplacement, which could not avail themselves to this assistance, the executive will be brought into play rapidly to assist if the PMIS emplacement is compliant with the PMIS CM Improvement Framework and fully operational.

Using the PMIS CM Improvement framework to implement a PMIS system that can handle the 12 improvement steps will facilitate Software project management using core CMMi processes that span across the three constellations as described in Section 2.4 and Figure 2.3. The PMIS emplacement set up correctly and fully operational can provide an improvement path to cover software development, acquisition, and services processes that could include all processes found in Annexure D, including the PCMM process found in Annexure H.

RO4: Propose a new diagram to be embedded into the PMBOK Process Groups and Knowledge Areas dashboard.

The new diagram produced by the researcher is found in Annexure A. This diagram called a Research Dimensions Model, Figure A.1, was compiled by the researcher to pin down the various dimensions that are envisaged to be at play regarding this research. Annexure A explains the Research Dimensions model and how it works.

6.3 Reflections and Future Work

The dissertation proposed a PMIS CM Improvement Framework to facilitate the implementation of a PMIS emplacement. The PMIS emplacement, if set up correctly according to the framework and fully operational, should be able to facilitate project management at CM L2 and above. It could be a challenge for an EPMO to install a PMIS emplacement correctly. It would also be a challenge to run projects run on a PMIS emplacement that were compliant with CM L2 processes and above. Using the PMIS CM Improvement Framework, the challenges could be understood and overcome, as was shown by the case study in Chapter 5.

As the Capability Maturity Model is all about improving processes for increased productivity and quality across five improvement levels, the PMIS emplacement, correctly installed and providing a firm and stable base at CM L2, should facilitate improvement while acting also to limit behaviour at CM L1. This continual process improvement focus using the PMIS emplacement is what the PMIS CM Improvement Framework offers, and should productivity and quality decrease for any reason, dropping below CM L2, the PMIS CM Improvement Framework can be applied again in remediation.

6.4 Research Assumptions (Case Study or Survey)

When the researcher had initially intended to undertake a survey for this research, he compiled a detailed questionnaire comprised of 30 questions to be administered to practitioners in industry. However, the researcher and his professor decided that a survey may not be well accepted in pro-Agile environments in South Africa if these were operated at CM L1. Therefore, we opted for a theoretical case study instead. This decision resulted in a change of focus, from deductive and mono method quantitative research more towards multi-method qualitative research.

6.5 Future Work

The validation of the PMIS CM Improvement Framework presented in this dissertation was conducted through a hypothetical case study, synthesised from the experiences of the researcher. Future work in this area could involve a quantitative-based industry survey utilising the points system developed in Section 4.6. Furthermore, the framework could be exercised among project management companies in industry by the actual application thereof on their processes. It is anticipated that it would facilitate the buy-in of Enterprise Project Management Offices within these industries that are looking to improve their project management processes, productivity, and quality.

6.6 Conclusion

This section concludes the research. The PMIS CM Improvement Framework appears to hold much promise as a systematic and stepwise way to install a PMIS emplacement. The PMIS CM Improvement Framework is also mindful of much of the surrounding methodologies and frameworks that exist. In this regard, it can further assist a willing executive to achieve a firm platform from which many improvements can be managed for productivity and quality improvements. These include improvements in CMMi, ITIL, DevOps, and COBIT, including areas alluded to in Annexure B, to name a few. The benefit of being able to tap into the People Capability Maturity levels towards the Empowered Workgroup is of particular interest because, ultimately, empowered people striving to do better is a key ingredient to success.

References used in this work, and the relevant Annexures appear below.

REFERENCES

Alawad, D., Panta, M., Zibran, M. & Islam, M.R. 2018. An empirical study of the relationships between code readability and software complexity. *27th International Conference on Software Engineering and Data Engineering, SEDE 2018*. 122–127.

Babar, S., Thaheem, M.J. & Ayub, B. 2017. Estimated cost at completion: Integrating risk into earned value management. *Journal of Construction Engineering and Management*. 143(3).

Beer, M., Eisenstat, R.A., University., H., Administration., G.S. of B. & Research., D. of. 2000. *Overcoming the "silent killers" to strategy implementation and organizational learning*. [Boston]: Division of Research, Harvard Business School.

Bočková, K.H., Sláviková, G. & Gabrhel, J. 2015. Game Theory as a Tool of Project Management. *Procedia - Social and Behavioral Sciences*. 213:709–715.

Boehm, B. 2002. Get ready for agile methods, with care. *Computer*. 35(1):64–69.

Boehm, B. & Turner, R. 2004. Balancing agility and discipline: evaluating and integrating agile and plan-driven methods. *Proceedings. 26th International Conference on Software Engineering*. 2–3.

Borštnar, M.K. & Pucihar, A. 2015. Impacts of the Implementation of a Project Management Information System – a Case Study of a Small R&D Company. *Organizacija*. 47(1):14–23.

Boyle, A. 2016. *Gradatim Ferociter! Jeff Bezos explains Blue Origin's motto, logo … and the boots*. [Online], Available: https://www.geekwire.com/2016/jeff-bezos-blue-origin-motto-logo-boots/#:~:text="Gradatim Ferociter" is Latin for,can't cut any corners. [2020, July 10].

Braglia, M. & Frosolini, M. 2014. An integrated approach to implement Project Management Information Systems within the Extended Enterprise. *International Journal of Project Management*. 32(1):18–29.

Brooks, F. 1995a. *The Mythical Man-Month : essays on software engineering. Anniversary Edition*.

Brooks, F.P. 1975. *The Mythical Man-Month : essays on software engineering*. Addison-Wesley Pub. Co.

Brooks, F.P. 1986. No silver bullet-essence and accidents of software engineering. *Proceedings of the IFIP Tenth World Computing Conference*. 1069–1076.

Brooks, F.P. 1987. No silver bullet: Essence and accidences of software engineering. *Computer*. 20(4):10–19.

Brooks, F.P. 1995b. *The Mythical Man - Month. Essays on Software Engineering Anniversary Edition*.

Caniëls, M.C.J. & Bakens, R.J.J.M. 2012. The effects of Project Management Information Systems on decision making in a multi project environment. *International Journal of Project Management*. 30(2):162–175.

Chrissis, M.B., Konrad, M., Shrum, S., Chrisses, M.B., Konrad, M. & Shrum, S. 2011. *CMMI for development.* 3rd ed. Upper Saddle River, NJ: Addison-Wesley.

Chrissis, M.B., Konrad, M. & Shrum, S. 2011. *CMMI for development.* 1.3 ed. Upper Saddle River, NJ: Addison-Wesley.

Clinning, G. & Marnewick, C. 2017. Incorporating sustainability into IT project management. *South African Computer Journal*. 29(1):1–26.

CMII research Institute. 2010. How CMII Can Further Enhance ISO 9001 : 2008.

Cohn, M. 2004. *User Stories Applied: For Agile Software Development*. Pearson Education, Inc.

Corrigan, M., Mtsweni, S.E. & van der Poll, J.A. 2018. The Project Management Information System as Enabler for ICT4D Achievement at Capability Maturity Level 2 and Above. *Krauss K., Turpin M., Naude F. (eds) Locally Relevant ICT Research. IDIA 2018. Communications in Computer and Information Science, vol 933. Springer*. Communicat(993):296–310. [Online], Available: https://doi.org/10.1007/978-3-030-11235-6_19.

Cunha, J.A.O.G., Moura, H.P. & Vasconcellos, F.J.S. 2016. Decision-making in Software Project Management: A Systematic Literature Review. *Procedia Computer Science*. 100:947–954.

Curtis, B. & Hefley, W.E.. 2001. SEI-95-MM-02 People Capability Maturity Model. 0.

Dear Agile, I'm Tired of Pretending. 2019. [Online], Available: https://medium.com/columbus-egg/dear-agile-im-tired-of-pretending-d39ab6a12003.

Deming, W.E. 1986. *Out of the crisis*. Cambridge, Mass.: Massachusetts Institute of Technology, Center for Advanced Engineering Study.

Diaz-Ley, M., Garcia, F. & Piattini, M. 2010. MIS-PyME software measurement capability maturity model - Supporting the definition of software measurement programs and capability determination. *ADES Advances in Engineering Software*. 41(10):1223–1237.

Dikert, K., Paasivaara, M. & Lassenius, C. 2016. Challenges and success factors for large-scale agile transformations: A systematic literature review. *Journal of Systems and Software*. 119:87–108.

Dooley, J.F. 2017. *Software development, design and coding: With patterns, debugging, unit testing, and refactoring second edition*.

Dora. 2017. *New research from DORA: What sets top-performing DevOps teams apart*. [Online], Available: https://inthecloud.withgoogle.com/state-of-devops-18/dl-cd.html [2020, July 10].

DSDM Consortium. 2014. *Agile PM. Agile Project Management Handbook Version 2.* 2nd ed. D. Consortium (ed.). DSDM Consortium.

Dudley, M. 2019. *Ladies and gentlemen, here comes the Dunning-Kruger Effect*. [Online], Available: https://medium.com/@mike.dudkey78/ladies-and-gentlemen-here-comes-the-dunning-kruger-effect-f31b09d52573 [2018, August 02].

Engwall, M. & Jerbrant, A. 2003. The resource allocation syndrome: the prime challenge of multi-project management? *JPMA International Journal of Project Management*. 21(6):403–409.

Entinex. 2016. *CMMIFAQ - brutally honest, totally hip, no nonsense answers (a public service of Entinex)*. [Online], Available: http://www.cmmifaq.info/#16 [2020, July 11].

Eveleens, L.J. & Verhoef, C. 2010. project management The Rise and Fall of the Chaos Report Figures. *IEEE Software*. 27(36):6. [Online], Available: www. c omp u t e r . o r g / s o f tw a r e.

Extreme Programming Web Site. 2019. [Online], Available: http://www.extremeprogramming.org/ [2018, April 09].

Fleming, Q. & Koppelman, J. 2010. *Earned Value Project Management*. Vol. 16.

Fraser D. Steven, Brooks P. Frederick, Fowler Martin, Lopez Ricardo, Nomioka Aki, Northrop Linda, Parnas David Lorge & Thomas David. 2007. “No Silver Bullet” Reloaded - A Retrospective on “Essence and Accidents of Software Engineering”. *Publication of: Conference, OOPSLA 07*. 1026–1030.

Fraser D. Steven, Brooks P. Frederick, Fowler Martin, Lopez Ricardo, Namioka Aki, Northrop Linda, Parnas David Lorge & Thomas David. 2008. *No Silver Bullet Reloaded Retrospective OOPSLA Panel Summary*. [Online], Available: https://www.infoq.com/articles/No-Silver-Bullet-Summary [2016, October 11].

Glass, R.L. 2005. IT Failure Rates--70% or 10-15%? *Software, IEEE*. 22:111–112.

Goldratt, E.M. 1997. *Critical chain*. Great Barrington, MA: North River Press.

Goldratt, E.M. & Cox, J. 2004. *The Goal*. 3rd ed. The North River Press.

Gonçalves, R. 2012. Supporting Time Planning Aligned with CMMI-DEV and PMBOK. *ICSEA 2012, The Seventh ….* (c):352–360. [Online], Available: http://www.thinkmind.org/index.php?view=article&articleid=icsea_2012_13_10_10161.

Guess, V.C. 2006. *CMII for Business Process Infrastructure.* 2nd ed. CMII Research Instutite.

Guess, V.C. 2012. *The CMII Model*. 1st ed. CMMi Instutite (ed.). CMII research Institutee.

Hailpern, B. & Padmanabhan, S.“. 2002. Software debugging, testing, and verification. in I.S. J. (ed.). 4–12.

Hansen, M.T. & Baggesen, H. 2009. From CMMI and isolation to scrum, agile, lean and collaboration. *Proceedings - 2009 Agile Conference, AGILE 2009*. (July):283–288.

Haverkort, B.R. & Zimmermann, A. 2017. Smart Industry: How ICT Will Change the Game! *IEEE Internet Computing*. 21(1):8–10.

Hirsch, B.J. a. 2012. PMI's Pulse of the Profession. *2012 PMI Pulse of the Profession*. March(3):1–9.

Hong Zhu. 2005. Software Design Methodology - From Principles to Architectural Styles.

Hughes, B. & Cotterell, M. 2009. *Software project management*. London: McGraw-Hill.

ICASA. 2011a. *COBIT 5 A Business Framework for the Governance and Management of IT*. ISACA. [Online], Available: http://www.isaca.org/COBIT/Pages/COBITFramework.aspx.

ICASA. 2011b. *COBIT 5 A Business Framework for the Governance and Management of IT*. ISACA.

Institute of Directors. 2002. *King Committee on Corporate Governance*.

Issa, A., Hatiboglu, B., Bildstein, A. & Bauernhansl, T. 2018. Industrie 4.0 roadmap: Framework for digital transformation based on the concepts of capability maturity and alignment. *Procedia CIRP*. 72:973–978.

Izmailov, A., Korneva, D. & Kozhemiakin, A. 2016. Effective Project Management with Theory of Constraints. *Procedia - Social and Behavioral Sciences*. 229(March):96–103.

Jakobsen, Ruseng, C. & Sutherland, J. 2009. Scrum and CMMI Going from Good to Great. 333–337.

Johansen, B. 2017. *The New Leadership Literacies*. Barrett-Koehler Publishers, Inc.

Johnson, P. & Ekstedt, M. 2016. The Tarpit - A general theory of software engineering. *Information and Software Technology*. 70:181–203.

Kabir, P. 2015. *Capability Maturity Model integrated: A short overview*. [Online], Available: https://www.slideshare.net/AHMPervejKabir/cmmi-55277312.

Kashyap, V. 2019. *12 Amazing Features to Look for in a Project Management System*. [Online], Available: https://www.proofhub.com/articles/project-management-system [2020, February 12].

Khurshid, N., Bannerman, P.L. & Staples, M. 2009. Overcoming the First Hurdle: Why Organizations Do Not Adopt CMMI. 15504:38–49.

Kirschenbaum, I. 2020. [Online], Available: https://www.slideshare.net/ilankirschenbaum/practical-scrum-course-day-1.

Knowles, G. 2011. *Quality Management*. graeme knowles and bookboon.com.

Kostalova, J., Tetrevova, L. & Svedik, J. 2015. Support of Project Management Methods by Project Management Information System. *Procedia - Social and Behavioral Sciences*. 210:96–104.

Kotter, J.P. 1996. *Leading Change*. Harvard University Press.

Krishna Kaiser, A. 2018. *Reinventing ITIL® in the Age of DevOps*. Publisher: Apress, Year: 2019.

Kruger, J. & Dunning, D. 1999. Unskilled and Unaware of It: How Difficulties in Recognizing One's Own Incompetence Lead to Inflated Self-Assessments. *Journal of Personality and Social Psychology*. 77(6):1121–1134. [Online], Available: http://arxiv.org/abs/1409.8351%5Cnhttp://link.springer.com/10.1007/s11412-014-9201-z%5Cnhttp://pss.sagepub.com/content/25/8/1608.long%5Cnhttp://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=4060654&tool=pmcentrez&rendertype=abstract%5Cnhttp://www.loc.

Lanjewar, S. n.d. *CMMi Maturity Levels*. [Online], Available: https://www.umsl.edu/~sauterv/analysis/Fall2013Papers/Lanjewar/CMMI.html [2020, October 22].

Light, M., Rosser, B. & Hayward, S. 2005. Realizing the benefits of projects and portfolio management. *Gartner Research ID G00125673*. 31.

List of IFRS Standards. 2020. [Online], Available: https://www.ifrs.org/issued-standards/list-of-standards/ [2020, July 12].

LogicalModel.net. 2016. *Parallels in Delivering Organisational Change as a Bowl Of Soup*. [Online], Available: http://www.logicalmodel.net/pm_ngt/soop-prince2-and-pmbok-as-a-bowl-of-soup/ [2016, October 13].

Lowry, P.B. & Wilson, D. 2016. Creating agile organizations through IT: The influence of internal IT service perceptions on IT service quality and IT agility. *Journal of Strategic Information Systems*. 25(3):211–226.

Luthans, F. 2012. *Organizational Behaviour An Evidence - Based Approach*.

Markovic, I. 2018. *What is Agile Scrum and why we use it*. [Online], Available: https://tms-outsource.com/blog/posts/scrum/ [2020, March 18].

Mcavoy, J. & Butler, T. 2009. The role of project management in ineffective decision making within Agile software development projects. *European Journal of Information Systems*. 18:383.

Meedeniya, D.A., Rubasinghe, I.D. & Perera, I. 2019. Software artefacts consistency management towards continuous integration: A roadmap. *International Journal of Advanced Computer Science and Applications*. 10(4):100–110.

Meredith, J.R. & Mantel, S.J.J. 2009. *Project Management A Managerial Approach*.

Meredith, J.R. & Mantel, S.J.J. 2012. *Project Management A Managerial Approach*. 8th ed. Hoboken, N.J: John Wiley.

Mitchell, A. 2018. A Review of Mixed Methods, Pragmatism and Abduction Techniques" has now been published in The Electronic Journal of Business Research Methods, Volume 16 Issue 3. *Electronic Journal on Business Research Methods*. 16:103–116.

Monitoring, P. 2010. Have PMBOK , Will Take CMMI Maturity Level 2. (Cmmi):5. [Online], Available: http://www.calibresys.com/documents/news/Have PMBOK_take CMMI Maturity Level 2.pdf.

Musawir, A. ul, Serra, C.E.M., Zwikael, O. & Ali, I. 2017. Project governance, benefit management, and project success: Towards a framework for supporting organizational strategy implementation. *International Journal of Project Management*. 35(8):1658–1672.

National Treasury. 2010. Public Finance Management Act, No. 1 of 1999. *Government Gazette*. (33059):1–82.

Nguyen, D. & Nguyen, T. 2016. Information Systems Success: The Project Management Information System for ERP Projects. *Lecture Notes of the Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, LNICST*. 165(February).

Oates, B.J. 2012. *Researching information systems and computing*. 2nd ed. London: Sage Publications.

Ockerman, S. & Reindl, S. 2019. *Mastering Professional Scrum*. Addison-Wesley.

Office of Government Commerce. 2009. *Managing successful projects with PRINCE2.* 5th ed. London: TSO.

Olivier, A. 2015. *The Nine Vital Components for Successful Strategy Execution in the Public Sector*. Johannesburg: PMI AFRICA CONFERENCE 2015.

Oncken, W. & Wass, D.L. 1999. Who ' s Got the Monkey ? Who ' s Got the Monkey ? *Harvard Business Review*. (December):6.

Ozkaya, I. 2019. Are DevOps and Automation Our Next Silver Bullet? *IEEE Software*. 36(4):3–95.

Pane, E.S. & Sarno, R. 2015. Capability Maturity Model Integration (CMMI) for Optimizing Object-Oriented Analysis and Design (OOAD). *Procedia Computer Science*. 72(Cmmi):40–48.

Parnas, D.L. & Weiss, D.M. 1987. Active Design Reviews: Principles and Practices. *Proceedings of the 8th International conference on Software engineering*. 265(7):132–136.

PDCA Cycle: The 4 Gears of Continual Service Improvement. n.d. [Online], Available: https://blog.masterofproject.com/pdca-cycle/ [2021, July 31].

People Capability Maturity Model (P-CMM). 2009. 2nd ed. [Online], Available: http://oai.dtic.mil/oai/oai?&verb=getRecord&metadataPrefix=html&identifier=ADA512354.

Pienaar Muell, R. & Cronje, S. 2008. Strategy implementation : overcoming the challenge. *mantod Management Today*. 24(2):12–15.

Pikkarainen, M. & Mäntyniemi, a. 2006. An approach for using CMMI in agile software development assessments: experiences from three case studies. *SPICE 2006 conference, Luxemburg*. (May):1–11. [Online], Available: http://agile.vtt.fi/docs/publications/2006/2006_agile_cmmi_camera_ready.pdf.

PMI. 2007. *Practice Standard for Scheduling*.

PMI. 2019. *EVM*. [Online], Available: https://www.pmi.org/pmbok-guide-standards/framework/earned-value-management-2nd [2019, September 23].

Pollack, J., Helm, J. & Adler, D. 2018. What is the Iron Triangle, and how has it changed? *International Journal of Managing Projects in Business*. 11(2):527–547.

Ponsteen, A. & Kusters, R.J. 2015. Classification of Human- and Automated Resource Allocation Approaches in Multi-Project Management. *Procedia - Social and Behavioral Sciences*. 194(October 2014):165–173.

Project Management Institute. 2011. *Practice Standard for Earned Value Management, 2nd Edition*.

Project Management Institute. 2013. *A guide to the project management body of knowledge (PMBOK ® guide)*. 5th ed. Atlanta: Project Management Institute, inc.

Project Management Institute. 2017a. *A Guide to the Management Body of Knowledge*. 6th ed. Project Management Institute (ed.). Atlanta: Project Management Institute, inc.

Project Management Institute. 2017b. Success Rates Rise: Transforming the high cost of low performance. *PMI's Pulse of the Profession - 9th Global Project Management Survey*. 1–32.

Project Management Institute. 2019. *PMI Standards*. [Online], Available: https://www.pmi.org/pmbok-guide-standards/foundational [2019, June 26].

Rahmani, H., Sami, A. & Khalili, A. 2016. CIP-UQIM: A unified model for quality improvement in software SME's based on CMMI level 2 and 3. *Information and Software Technology*. 71:27–57.

Rahmati, V. 2016. Software Development's Total Cost Evaluation by Applying Complexity Function of Different Orders. *International Journal of Modern Engineering Technologies*. 1(1):1–5. [Online], Available: http://www.icdst.org/OAJ/index.php/IJMET/article/view/5.

Railton, P. 1995. *Objective and subjective research perspectives*. Vol. 8.

Rakitin, S. 2001. Manifesto elicits cynicism. *IEEE Computer*. 34(12):4–7. [Online], Available: http://sunset.usc.edu/events/2002/arr/letters.pdf.

Raymond, L. & Bergeron, F. 2008a. Project management information systems: An empirical study of their impact on project managers and project success. *International Journal of Project Management*. 26(2):213–220.

Raymond, L. & Bergeron, F. 2008b. Project management information systems: An empirical study of their impact on project managers and project success. *International Journal of Project Management*. 26(2):213–220.

Rensburg, M.J. Van & Davis, A. 2010. Strategic Management Guide (UNISA STRBESC).

Richards, K. & Cooper, L. 2015. *PRINCE2 Agile*. Norwich: The Stationery Office. [Online], Available: www.

Robinson, C.G. 1982. Experience curves as a planning tool. Part 2. Pyrrhic victories and practical problems. *South African Journal of Business Management*. 13(4):159–168.

Saunders, M., Lewis, P. & Thornhill, A. 2007. *Research Methods for Business Students*. 4th ed.

Saunders, M., Lewis, P. & Thornhill, A. 2016. *Research Methods for Business Students*. Pearson Education.

Schach, S. 2007. *Object Orientated and Classical Software Engineering. Eighth Edition*.

Schach, S.R. 2011. *Object-oriented and classical software engineering*. 8th ed. New York: McGraw-Hill.

Schubert, L., Tsitsipas, A. & Jeffery, K. 2018. Establishing a basis for new software engineering principles. *Internet of Things*. 3–4:187–195.

Schwaber, K. & Sutherland, J. 2017. The Scrum Guide: The Definitive The Rules of the Game. *Scrum.Org and ScrumInc*. (November):19.

Scott, J.C. 1990. [Online], Available: http://site.ebrary.com/id/10579306.

SDLC - V-Model. n.d. [Online], Available: https://www.tutorialspoint.com/sdlc/sdlc_v_model.htm [2020, October 27].

Selleri Silva, F., Soares, F.S.F., Peres, A.L., Azevedo, I.M. De, Vasconcelos, A.P.L.F., Kamei, F.K. & Meira, S.R.D.L. 2015. Using CMMI together with agile software development: A systematic review. *Information and Software Technology*. 58:20–43.

Setpathy, T. 2016. *Scrum Body of Knowledge*. SCRUMstudy.

Shango, D. 2016. King 4: An outcomes-based corporate governance code fit for a changing world. *Pwc*. (November):1–16. [Online], Available: www.pwc.com/structurefor.

Sohi, A.J., Hertogh, M., Bosch-Rekveldt, M. & Blom, R. 2016. Does Lean & Agile Project Management Help Coping with Project Complexity? *Procedia - Social and Behavioral Sciences*. 226(October 2015):252–259.

Staples, M., Niazi, M., Jeffery, R., Abrahams, A., Byatt, P. & Murphy, R. 2007. An exploratory study of why organizations do not adopt CMMI. *Journal of Systems and Software*. 80(6):883–895.

Stocks, G. 2013. *Ensure governance for agile projects*. [Online], Available: http://www.business-it.co.za/ensure-governance-for-agile-projects/.

STS Sauter Training & Simulation S.A & STS Sauter. 2013. Comparing PMBOK ® Guide 4 th Edition , PMBOK ® Guide 5 th Edition and ISO 21500. *STS SA*. 2–27. [Online], Available:

www.sts.ch.

Sun, Y. & Liu, X. (Frank). 2010. Business-oriented software process improvement based on CMMI using QFD. *INFSOF Information and Software Technology*. 52(1):79–91.

Sutherland, J., Schwaber, K., Highsmith, J., Cockburn, A., Martin, R.C., Beck, K., Beedle, M., Cunningham, W., et al. 2001. *Manifesto for Agile Software Development*. [Online], Available: http://agilemanifesto.org/ [2018, April 09].

Tarnovski, M. 2014. *CMMI — Capability Maturity Model Integration*. [Online], Available: https://www.plays-in-business.com/cmmi-capability-maturity-model-integration/ [2020, March 08].

Telecom, O. & Technologies, H. 2006. Case Study: Enabling a Customer-Centric Experience through Project Man a gement. [Online], Available: https://www.pmi.org/-/media/pmi/documents/public/pdf/case-study/enabling-customer-centric-experience.pdf.

Tendon, S. 2012. *Project Buffers and the Critical Path*. [Online], Available: https://tameflow.com/blog/2012-10-04/buffer-management-and-risk-management-in-TOC/ [2020, March 12].

The Standish Group. 1995. The standish group report. *Chaos*. 49:1–8.

The Standish Group. 2014. The Standish group: the chaos report. *Project Smart*. 16.

Thomas, D. 2015. *Agile is Dead*. [Online], Available: https://www.youtube.com/watch?v=a-BOSpxYJ9M.

Tromp, J.W. & Homan, T. 2015. How Unplanned Changes Emerge while Implementing a Project Management Information System (PMIS) in a Complex Multi Project R&D Environment. *Procedia - Social and Behavioral Sciences*. 194(October 2014):211–220.

Turley, F. 2010. *The PRINCE2 Training Manual-Practitioner Level*. 1st ed. TAG (ed.). London: TAG.

Vargas, R. 2017. *Elaboration of the PMBOK 6 process flow*. [Online], Available: https://www.youtube.com/watch?v=GC7pN8Mjot8 [2018, August 03].

Verwijs, C. 2018. *Why Scrum requires completely done software every sprint*. [Online], Available: https://medium.com/the-liberators/why-scrum-requires-completely-done-software-every-sprint-f7fa3ca33286 [2020, March 18].

Wagner, J. 2017. *The People, Process, Technology Triangle is dead, and has evolved into a diamond!* [Online], Available: https://www.linkedin.com/pulse/people-process-technology-triangle-dead-has-evolved-diamond-wagner.

Walton, M. & Heeks, R. 2011. *Development Informatics Improve ICT4D Project*. Development Informatics Group. [Online], Available: http://sacj.cs.uct.ac.za/index.php/sacj/article/view/398.

What is DevOps. 2019. [Online], Available: https://theagileadmin.com/what-is-devops/.

Wideman, R.M. 2001. *Project Management Simply Explained A Logical Framework to Help Your Understanding 1*.

Wysocki, R.K. 2014. *Effective Complext Project Management*.

Wysocki, R.K. 2019. *Effective Project Management: Traditional, Agile, Extreme, Hybrid*. 8th ed. John Wiley & Sons.

Zaman, U., Jabbar, Z., Nawaz, S. & Abbas, M. 2019. Understanding the soft side of software projects: An empirical study on the interactive effects of social skills and political skills on complexity – performance relationship. *International Journal of Project Management*. 37(3):444–460.

Annexure A: Research Dimensions Model

This Research Dimensions model, Figure A.1, was compiled by the researcher to pin down the various dimensions that are envisaged to be at play regarding this research. The research hypotheses are found in the circle at the top right of Figure A.1.

Figure A.1 shows that any improvement in Production and Quality must surely occur along a vertical axis with High Risk, Rework, and Waste at the bottom and High Productivity and Quality at the top of the figure. From a Risk, Rework, Waste, Productivity, and Quality perspective, these support the model in Annexure D.

A horizontal axis in the graphic explains that Risk, Rework, and Waste will increase or decrease along the horizontal axis in direct relation to the project method used and the time spent on Planning and Monitoring and Control. Based on observation, it appears that agile projects could be riskier to run than traditionally managed projects. Furthermore, project processes adherence needed per CM L2 processes could decrease the more one moves away from Traditional Project Management.

The vertical axis from Deming's Plan, Do, Check and Act perspective illustrates a spiralling upwards when improvements occur, a holding pattern if status remains unchanged, or a spiralling downwards if conditions deteriorate. From a game theory perspective, being able to plot production statistics while a project is in flight, made available on a PMIS emplacement producing Earned Value Management statistics, means project flight plans can be more tightly managed for higher levels of productivity, quality, safety and reliability.

Annexure A asserts the hypothesis for this work: The only way to achieve CM L2 is via tasking on a developed schedule within a PMIS. Ultimately, the implementation of processes for Productivity and Quality improvements must start at CM L2, if the gravitational pull of chaos at CM L1 is to be escaped. The initial CM L2 processes that have to be installed first are Project Requirements (REQM), Project Planning (PP) and Project Monitoring and Control (PMC).

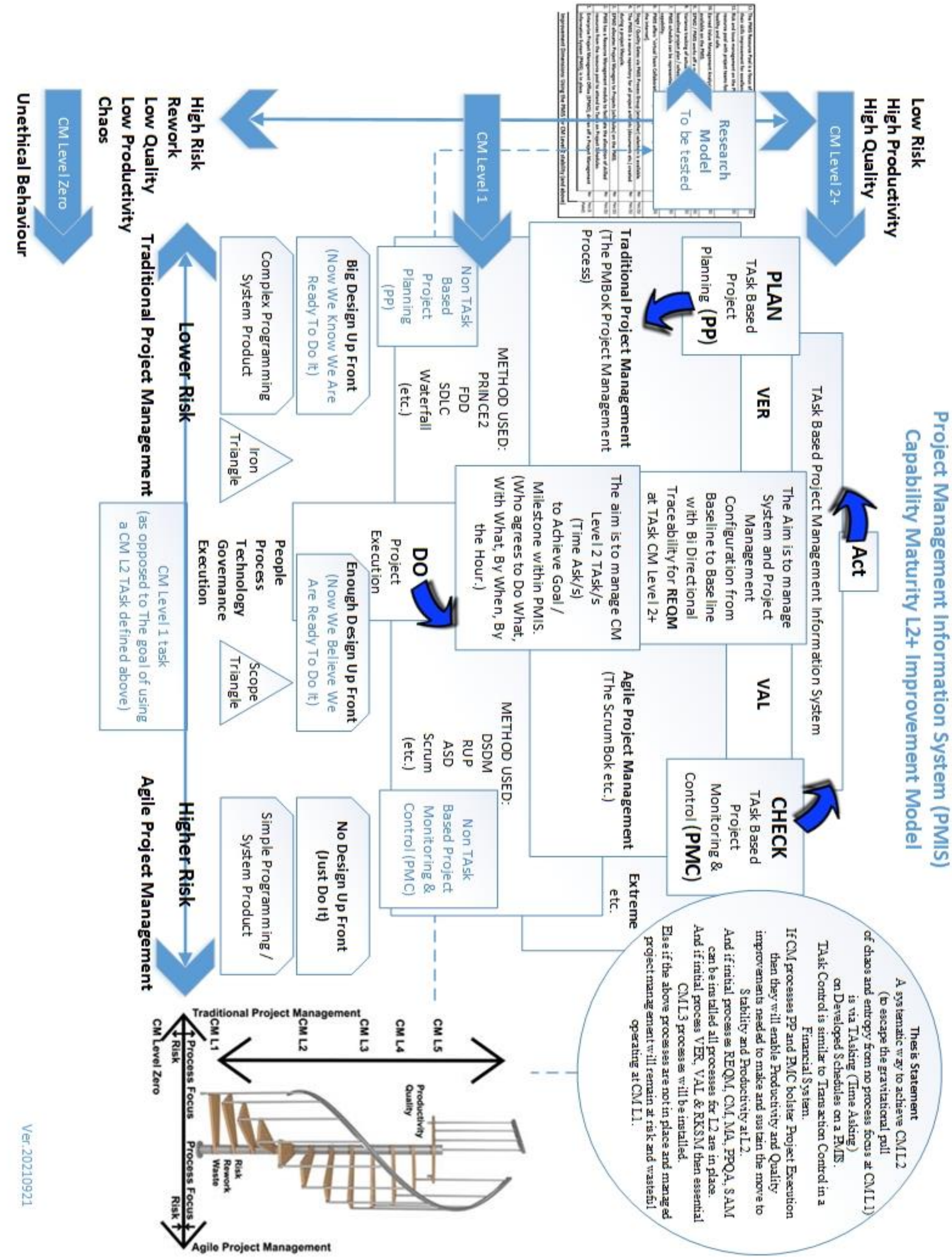


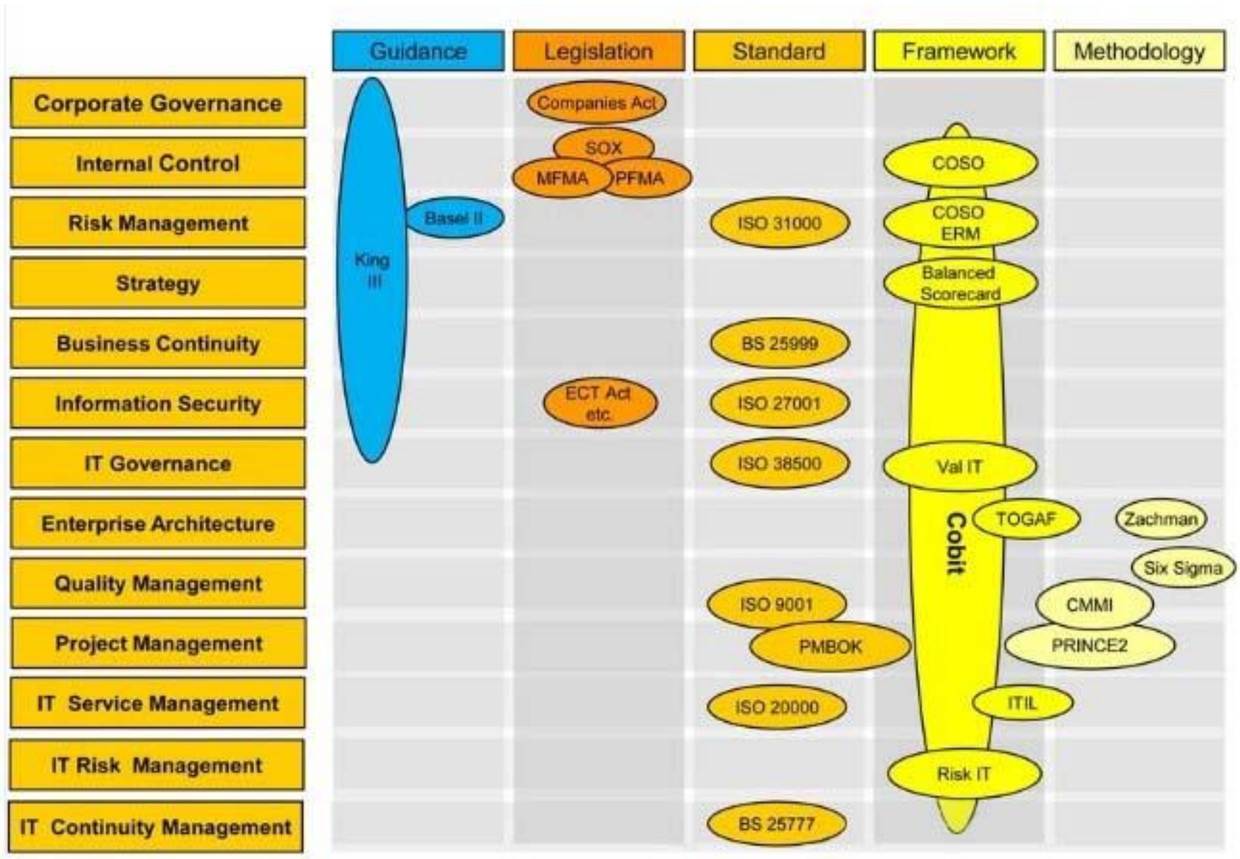
Figure A.1: Research Dimensions Model

Source: Compiled by the researcher

Annexure B: Standards / Frameworks Landscape

This Standards / Frameworks Landscape was made available to the writer during his work in the Public Sector at the State Information Technology Agency. The table was used within the EPMO. From the researcher's experience over many years and having qualified and certified in, and read up on all of the areas that comprise this table, it is included here for the succinct overview it provides.

Table B.1: Standards / Frameworks Landscape



Source: Synthesized by the researcher from State Information Technology Agency 2016

Annexure C: Capability Maturity Model

Table C.1 is a high-level summary of the processes that comprise the Capability Maturity Model integrated version 1.3 from the Carnegie Mellon University in the United States of America. The table is included in this work as it clearly explains that Risk & Waste is found at CM L1, which is also the realm of heroic behaviour. Productivity and Quality improvements are obtained by installing the processes mentioned. This table is easier to process than the schedule of processes found in Annexure D. In addition, the writer has indicated that the process focus and management of CM L2 processes are all about Project Management processes. These processes dovetail with the processes found in the PMBOK matrix in Annexure S, and while these only contain the process names, Annexure S provides far more detail for each project process, including its Inputs, Tools and Techniques, and Outputs. To that end, Table C.1 connects the detail found in Annexure D to the detail found in Annexure S in a basic way and shows that Productivity and Quality gains are possible using processes application. However, if process application is ignored, then clearly Risk & Rework is the logical outcome, which appears to be what the Standish Reports are concerned about (2014).

Table C.1: Capability Maturity Model integrated

Level			Capability	Result
5	Optim-izing	Continuous Process Improvement	Organizational Innovation & Deployment Causal Analysis & Resolution	Productivity & Quality
4	Quanti-tatively Managed	Quantitative Management	Quantitative Process Management Software Quality Management	
3	Defined	Process Standardization	Requirements Development Technical Solution Product Integration Verification Validation Organizational Process Focus Organizational Process Definition Organizational Training Integrated Product Management Risk Management Integrated Teaming Integrated Supplier Management Decision Analysis & Resolution Organizational Environment for Integration	
2	Managed	Project Management	Requirements Management Project Planning Project Monitoring & Control Supplier Agreement Management Measurement & Analysis Product & Process Quality Assurance Configuration Management	
1	Initial	Heroic Efforts	Design Develop Integrate Test	Risk & Waste

Source: Synthesized by the researcher from Chrissis *et al.* (2011) & Kabir (2015)

Annexure D: CMMi for Development Version V1.3

Table D.1 is a detailed listing of the Capability Maturity integrated processes from the Carnegie Mellon University CMMi model. The table is separated into four sections, Process Management, Project Management, Engineering, and Support Processes. Each section is on a separate page.

The following notes explain the headings at the top of the table, and how to use the table.

- At the top left of the table is the heading Category. Category refers to a logical grouping of Process Areas, which include Process Management, Project Management, Engineering, and Support.
- At the right of Category is Process Area. Process Area refers to a set of activities, methods, practices, and transformations that are used to develop and maintain systems and associated products. There are 22 Process Areas in Capability Maturity Model integrated for Development V1.3. Essentially, a Process Area is a cluster of related practices in an area that, when performed collectively, satisfy a set of goals considered important for making significant improvement in that area.
- To the right of the Process Area is the Process Area (PA) shortcode. This is a shortened code that could be easier to use than a longer Process Area name. An example is Project Planning, which has a short code PP and the short code for Project Monitoring and Control is PMC.

  Process Areas are common to both Continuous and Staged representations. In the Staged Representation approach, process areas are organised by maturity levels first. Processes for level 2, for example, must be institutionalised and completed fully before starting on level 3. In the Continuous Representation approach, selected processes can be improved in any order that suits the organisation. An example of this could be focusing on Project Planning (PP) and Project Monitoring and Control (PMC) at level 2 first, ignoring all the other processes for level 2 and moving directly to Risk Management (RSKM) at level 3.

- To the right of the Process Area is a short code for the Maturity Level or ML. The Maturity Level indicates the level of the process in a range from Level 1 to Level 5. PP, for example, is ML 2, while RSKM is ML 3.
- To the right of Maturity Level (ML) is the purpose of the process. This is a short description of the process and what it aims to achieve.
- Within each Process Area, Specific Goals and Specific Practices are stated. Specific Goals address the unique characteristics that describe what must be implemented to satisfy the Process Area. Specific Goals are required model components and are used in appraisals to help determine whether a process area is satisfied.
- Specific Practices are activities considered important in achieving the associated Specific Goal. The Specific Practices describe the activities expected to result in the achievement of the Specific Goals of a Process Area. Specific practices are expected model components.

Table D.1: CMMi Processes by Category

CMMI® for Development, V1.3

Category		Process Area	PA	ML	Purpose
Process Management		**Organizational Process Focus**	OPF	3	The purpose of Organizational Process Focus (OPF) is to plan, implement, and deploy organizational process improvements based on a thorough understanding of the current strengths and weaknesses of the organization's processes and process assets.
	SG 1	**Determine Process Improvement Opportunities**			
	SP 1.1	Establish Organizational Process Needs			
	SP 1.2	Appraise the Organization's Processes			
	SP 1.3	Identify the Organization's Process Improvements			
	SG 2	**Plan and Implement Process Actions**			
	SP 2.1	Establish Process Action Plans			
	SP 2.2	Implement Process Action Plans			
	SG 3	**Deploy Organizational Process Assets and Incorporate Experiences**			
	SP 3.1	Deploy Organizational Process Assets			
	SP 3.2	Deploy Standard Processes			
	SP 3.3	Monitor the Implementation			
	SP 3.4	Incorporate Experiences into the Organizational Process Assets			
		Organizational Process Definition	OPD	3	The purpose of Organizational Process Definition (OPD) is to establish and maintain a usable set of organizational process assets, work environment standards, and rules and guidelines for teams.
	SG 1	**Establish Organizational Process Assets**			
	SP 1.1	Establish Standard Processes			
	SP 1.2	Establish Lifecycle Model Descriptions			
	SP 1.3	Establish Tailoring Criteria and Guidelines			
	SP 1.4	Establish the Organization's Measurement Repository			
	SP 1.5	Establish the Organization's Process Asset Library			
	SP 1.6	Establish Work Environment Standards			
	SP 1.7	Establish Rules and Guidelines for Teams			
		Organizational Training	OT	3	The purpose of Organizational Training (OT) is to develop the skills and knowledge of people so they can perform their roles effectively and efficiently.
	SG 1	**Establish an Organizational Training Capability**			
	SP 1.1	Establish Strategic Training Needs			
	SP 1.2	Determine Which Training Needs Are the Responsibility of the Organization			
	SP 1.3	Establish an Organizational Training Tactical Plan			
	SP 1.4	Establish a Training Capability			
	SG 2	**Provide Training**			
	SP 2.1	Deliver Training			
	SP 2.2	Establish Training Records			
	SP 2.3	Assess Training Effectiveness			
		Organizational Process Performance	OPP	4	The purpose of Organizational Process Performance (OPP) is to establish and maintain a quantitative understanding of the performance of selected processes in the organization's set of standard processes in support of achieving quality and process performance objectives, and to provide the process performance data, baselines, and models to quantitatively manage the organization's projects.
	SG 1	**Establish Performance Baselines and Models**			
	SP 1.1	Establish Quality and Process Performance Objectives			
	SP 1.2	Select Processes			
	SP 1.3	Establish Process Performance Measures			
	SP 1.4	Analyze Process Performance and Establish Process Performance Baselines			
	SP 1.5	Establish Process Performance Models			
		Organizational Performance Management	OPM	5	The purpose of Organizational Performance Management (OPM) is to proactively manage the organization's performance to meet its business objectives.
	SG 1	**Manage Business Performance**			
	SP 1.1	Maintain Business Objectives			
	SP 1.2	Analyze Process Performance Data			
	SP 1.3	Identify Potential Areas for Improvement			
	SG 2	**Select Improvements**			
	SP 2.1	Elicit Suggested Improvements			
	SP 2.2	Analyze Suggested Improvements			
	SP 2.3	Validate Improvements			
	SP 2.4	Select and Implement Improvements for Deployment			
	SG 3	**Deploy Improvements**			
	SP 3.1	Plan the Deployment			
	SP 3.2	Manage the Deployment			
	SP 3.3	Evaluate Improvement Effects			

Category		Process Area	PA	ML	Purpose
Project Management		**Project Planning**	PP	2	The purpose of Project Planning (PP) is to establish and maintain plans that define project activities.
	SG 1	**Establish Estimates**			
	SP 1.1	Estimate the Scope of the Project			
	SP 1.2	Establish Estimates of Work Product and Task Attributes			
	SP 1.3	Define Project Lifecycle Phases			
	SP 1.4	Estimate Effort and Cost			
	SG 2	**Develop a Project Plan**			
	SP 2.1	Establish the Budget and Schedule			
	SP 2.2	Identify Project Risks			
	SP 2.3	Plan for Data Management			
	SP 2.4	Plan for Project Resources			
	SP 2.5	Plan for Needed Knowledge and Skills			
	SP 2.6	Plan Stakeholder Involvement			
	SP 2.7	Establish the Project Plan			
	SG 3	**Obtain Commitment to the Plan**			
	SP 3.1	Review Plans That Affect the Project			
	SP 3.2	Reconcile Work and Resource Levels			
	SP 3.3	Obtain Plan Commitment			
		Project Monitoring and Control	PMC	2	The purpose of Project Monitoring and Control (PMC) is to provide an understanding of the project's progress so that appropriate corrective actions can be taken when the project's performance deviates significantly from the plan.
	SG 1	**Monitor Project Against Plan**			
	SP 1.1	Monitor Project Planning Parameters			
	SP 1.2	Monitor Commitments			
	SP 1.3	Monitor Project Risks			
	SP 1.4	Monitor Data Management			
	SP 1.5	Monitor Stakeholder Involvement			
	SP 1.6	Conduct Progress Reviews			
	SP 1.7	Conduct Milestone Reviews			
	SG 2	**Manage Corrective Action to Closure**			
	SP 2.1	Analyze Issues			
	SP 2.2	Take Corrective Action			
	SP 2.3	Manage Corrective Action			
		Supplier Agreement Management	SAM	2	The purpose of Supplier Agreement Management (SAM) is to manage the acquisition of products from suppliers.
	SG 1	**Establish Supplier Agreements**			
	SP 1.1	Determine Acquisition Type			
	SP 1.2	Select Suppliers			
	SP 1.3	Establish Supplier Agreements			
	SG 2	**Satisfy Supplier Agreements**			
	SP 2.1	Execute the Supplier Agreement			
	SP 2.2	Accept the Acquired Product			
	SP 2.3	Ensure Transition of Products			
		Integrated Project Management	IPM	3	The purpose of Integrated Project Management (IPM) is to establish and manage the project and the involvement of the relevant stakeholders according to an integrated and defined process that is tailored from the organization's set of standard processes. For IPPD, Integrated Project Management also covers the establishment of a shared vision for the project and the establishment of integrated teams that will carry out objectives of the project.
	SG 1	**Use the Project's Defined Process**			
	SP 1.1	Establish the Project's Defined Process			
	SP 1.2	Use Organizational Process Assets for Planning Project Activities			
	SP 1.3	Establish the Project's Work Environment			
	SP 1.4	Integrate Plans			
	SP 1.5	Manage the Project Using the Integrated Plans			
	SP 1.6	Establish Teams			
	SP 1.7	Contribute to the Organizational Process Assets			
	SG 2	**Coordinate and Collaborate with Relevant Stakeholders**			
	SP 2.1	Manage Stakeholder Involvement			
	SP 2.2	Manage Dependencies			
	SP 2.3	Resolve Coordination Issues			
		Risk Management	RSKM	3	The purpose of Risk Management (RSKM) is to identify potential problems before they occur so that risk-handling activities can be planned and invoked as needed across the life of the product or project to mitigate adverse impacts on achieving objectives.
	SG 1	**Prepare for Risk Management**			
	SP 1.1	Determine Risk Sources and Categories			
	SP 1.2	Define Risk Parameters			
	SP 1.3	Establish a Risk Management Strategy			
	SG 2	**Identify and Analyze Risks**			
	SP 2.1	Identify Risks			
	SP 2.2	Evaluate, Categorize, and Prioritize Risks			
	SG 3	**Mitigate Risks**			
	SP 3.1	Develop Risk Mitigation Plans			
	SP 3.2	Implement Risk Mitigation Plans			
		Quantitative Project Management	QPM	4	The purpose of Quantitative Project Management (QPM) is to quantitatively manage the project's defined process to achieve the project's established quality and process-performance objectives.
	SG 1	**Prepare for Quantitative Management**			
	SP 1.1	Establish the Project's Objectives			
	SP 1.2	Compose the Defined Process			
	SP 1.3	Select the Subprocesses and Attributes			
	SP 1.4	Prepare for Quantitative Management			
	SG 2	**Quantitatively Manage the Project**			
	SP 2.1	Monitor the Performance of Selected Subprocesses			
	SP 2.2	Manage Project Performance			
	SP 2.3	Perform Root Cause Analysis			

Category	Process Area	PA	ML	Purpose
Engineering	**Requirements Management**	REQM	2	The purpose of Requirements Management (REQM) is to manage the requirements of the project's products and product components and to identify inconsistencies between those requirements and the project's plans and work products.
	SG 1 Manage Requirements			
	SP 1.1 Understand Requirements			
	SP 1.2 Obtain Commitment to Requirements			
	SP 1.3 Manage Requirements Changes			
	SP 1.4 Maintain Bidirectional Traceability of Requirements			
	SP 1.5 Ensure Alignment Between Project Work and Requirements			
	Requirements Development	RD	3	The purpose of Requirements Development (RD) is to produce and analyze customer, product, and product component requirements.
	SG 1 Develop Customer Requirements			
	SP 1.1 Elicit Needs			
	SP 1.2 Transform Stakeholder Needs into Customer Requirements			
	SG 2 Develop Product Requirements			
	SP 2.1 Establish Product and Product Component Requirements			
	SP 2.2 Allocate Product Component Requirements			
	SP 2.3 Identify Interface Requirements			
	SG 3 Analyze and Validate Requirements			
	SP 3.1 Establish Operational Concepts and Scenarios			
	SP 3.2 Establish a Definition of Required Functionality and Quality Attributes			
	SP 3.3 Analyze Requirements			
	SP 3.4 Analyze Requirements to Achieve Balance			
	SP 3.5 Validate Requirements			
	Technical Solution	TS	3	The purpose of Technical Solution (TS) is to design, develop, and implement solutions to requirements. Solutions, designs, and implementations encompass products, product components, and product-related lifecycle processes either singly or in combination as appropriate.
	SG 1 Select Product Component Solutions			
	SP 1.1 Develop Alternative Solutions and Selection Criteria			
	SP 1.2 Select Product Component Solutions			
	SG 2 Develop the Design			
	SP 2.1 Design the Product or Product Component			
	SP 2.2 Establish a Technical Data Package			
	SP 2.3 Design Interfaces Using Criteria			
	SP 2.4 Perform Make, Buy, or Reuse Analyses			
	SG 3 Implement the Product Design			
	SP 3.1 Implement the Design			
	SP 3.2 Develop Product Support Documentation			
	Product Integration	PI	3	The purpose of Product Integration (PI) is to assemble the product from the product components, ensure that the product, as integrated, functions properly, and deliver the product.
	SG 1 Prepare for Product Integration			
	SP 1.1 Establish an Integration Strategy			
	SP 1.2 Establish the Product Integration Environment			
	SP 1.3 Establish Product Integration Procedures and Criteria			
	SG 2 Ensure Interface Compatibility			
	SP 2.1 Review Interface Descriptions for Completeness			
	SP 2.2 Manage Interfaces			
	SG 3 Assemble Product Components and Deliver the Product			
	SP 3.1 Confirm Readiness of Product Components for Integration			
	SP 3.2 Assemble Product Components			
	SP 3.3 Evaluate Assembled Product Components			
	SP 3.4 Package and Deliver the Product or Product Component			
	Verification	VER	3	The purpose of Verification (VER) is to ensure that selected work products meet their specified requirements.
	SG 1 Prepare for Verification			
	SP 1.1 Select Work Products for Verification			
	SP 1.2 Establish the Verification Environment			
	SP 1.3 Establish Verification Procedures and Criteria			
	SG 2 Perform Peer Reviews			
	SP 2.1 Prepare for Peer Reviews			
	SP 2.2 Conduct Peer Reviews			
	SP 2.3 Analyze Peer Review Data			
	SG 3 Verify Selected Work Products			
	SP 3.1 Perform Verification			
	SP 3.2 Analyze Verification Results			
	Validation	VAL	3	The purpose of Validation (VAL) is to demonstrate that a product or product component fulfills its intended use when placed in its intended environment.
	SG 1 Prepare for Validation			
	SP 1.1 Select Products for Validation			
	SP 1.2 Establish the Validation Environment			
	SP 1.3 Establish Validation Procedures and Criteria			
	SG 2 Validate Product or Product Components			
	SP 2.1 Perform Validation			
	SP 2.2 Analyze Validation Results			

Category		Process Area	PA	ML	Purpose
Support		**Configuration Management**	CM	2	The purpose of Configuration Management (CM) is to establish and maintain the integrity of work products using configuration identification, configuration control, configuration status accounting, and configuration audits.
	SG 1	**Establish Baselines**			
	SP 1.1	Identify Configuration Items			
	SP 1.2	Establish a Configuration Management System			
	SP 1.3	Create or Release Baselines			
	SG 2	**Track and Control Changes**			
	SP 2.1	Track Change Requests			
	SP 2.2	Control Configuration Items			
	SG 3	**Establish Integrity**			
	SP 3.1	Establish Configuration Management Records			
	SP 3.2	Perform Configuration Audits			
		Process and Product Quality Assurance	PPQA	2	The purpose of Process and Product Quality Assurance (PPQA) is to provide staff and management with objective insight into processes and associated work products.
	SG 1	**Objectively Evaluate Processes and Work Products**			
	SP 1.1	Objectively Evaluate Processes			
	SP 1.2	Objectively Evaluate Work Products			
	SG 2	**Provide Objective Insight**			
	SP 2.1	Communicate and Ensure Resolution of Noncompliance Issues			
	SP 2.2	Establish Records			
		Measurement and Analysis	MA	2	The purpose of Measurement and Analysis (MA) is to develop and sustain a measurement capability that is used to support management information needs.
	SG 1	**Align Measurement and Analysis Activities**			
	SP 1.1	Establish Measurement Objectives			
	SP 1.2	Specify Measures			
	SP 1.3	Specify Data Collection and Storage Procedures			
	SP 1.4	Specify Analysis Procedures			
	SG 2	**Provide Measurement Results**			
	SP 2.1	Collect Measurement Data			
	SP 2.2	Analyze Measurement Data			
	SP 2.3	Store Data and Results			
	SP 2.4	Communicate Results			
		Decision Analysis and Resolution	DAR	3	The purpose of Decision Analysis and Resolution (DAR) is to analyze possible decisions using a formal evaluation process that evaluates identified alternatives against established criteria.
	SG 1	**Evaluate Alternatives**			
	SP 1.1	Establish Guidelines for Decision Analysis			
	SP 1.2	Establish Evaluation Criteria			
	SP 1.3	Identify Alternative Solutions			
	SP 1.4	Select Evaluation Methods			
	SP 1.5	Evaluate Alternatives Solutions			
	SP 1.6	Select Solutions			
		Causal Analysis and Resolution	CAR	5	The purpose of Causal Analysis and Resolution (CAR) is to identify causes of selected outcomes and take action to improve process performance.
	SG 1	**Determine Causes of Selected Outcomes**			
	SP 1.1	Select Outcomes for Analysis			
	SP 1.2	Analyze Causes			
	SG 2	**Address Causes of Selected Outcomes**			
	SP 2.1	Implement the Action Proposals			
	SP 2.2	Evaluate the Effect of Implemented Actions			
	SP 2.3	Record Causal Analysis Data			

Source: https://resources.sei.cmu.edu/library/asset-view.cfm?assetid=9661

Annexure E: ITIL Processes

ITIL or the IT Infrastructure Library is an IT Service Management best practice suite comprising five core manuals with a large amount of supporting material. ITIL is currently transitioning from ITIL version 3 to ITIL version 4. At the beginning of 2020, only an ITIL Foundation Manual has been released to the public. As such, the discussion on ITIL will focus on ITIL version 3.

Table E.1 has been compiled by the writer and illustrates that the IT Infrastructure Library has 26 processes. The processes have been compiled over previous iterations or versions of ITIL and are compiled or constructed into five process groups or lifecycle phases, based on industry experience and best practice.

The first ITIL process group, Service Strategy, aims to produce a Service Strategy plan defining the best way forward for an IT service providing area. Service Strategy receives inputs from the Continual Service Improvement (CSI) process group in the form of a CSI register, which is a continually groomed list of system requirements with the most valuable at the top. Service Design processes follow Service Strategy to ensure that valuable software is correctly designed and developed. Service Transition processes focus on the development team being able to release the next versions of valuable software along a quality assurance path towards final sign off by the business user in a User Acceptance Testing (UAT) environment that closely mirrors that of the Production System. Service Operation processes accept approved versions of the software into production to ensure that technology and software remain available, effective, and continue to operate efficiently. Finally, the Continual Service Improvement (CSI) processes, which work in much the same way as the Plan Do Check Act Cycle, are sent along to the next Service Strategy iteration for consideration as a potential value in future builds of software.

It is of interest to note that while there is an opinion that DevOps has eclipsed ITIL, this is not the case. The founder of DevOps and author of the DevOps Handbook, Gene Kim, says that DevOps opposing ITIL is a misnomer. “Even releasing 10,000+ deployments per day requires processes, but what goes against DevOps objectives are the approvals.” (Krishna Kaiser, 2018:74)

A clear objective for software development that has ramifications for game theory must be the ability to rapidly release valuable software in line with ITIL best practices, and if possible, following the more rapid DevOps processes. Organisations that can do this well will outshine and outlast those that cannot.

Table E.1: ITIL Processes by Service Area

STRATEGY	DESIGN	TRANSITION	OPERATIONS	CSI
(Business Requirements) *[NB There are 26 ITIL Processes. Their names are underlined in italic, and between square brackets per the list in the SS Manual pg 28]* CSI REGISTER **SS PROCESSES:** 1. [*Strategy management for IT services*] • Strategy assessment • Strategy generation • Strategy execution 2. [*Service portfolio management*] • Pipeline • Service Catalogue • Retired 3. [*Financial management for IT services*] • Accounting • Budgeting • Charging **BUSINESS CASE (Charter kicks off the project management processes)** 4. [*Demand management*] **PBAs:** Patterns of Business Activity Assist us to understand how customers (users) utilize the service for allocation of the right capacity • Classification • Attributes • Requirements 5. [*Business relationship management*] • Identifying Strategic Requirements of customers **UTILITY = Fit for Purpose** **Warranty = Fit for Use** **U + W = Value = Customer** **Service Assets: Capabilities + Resources** **4 Ps** • Perspectives • Patterns • Positions • Plans	**(Functional Requirements)** Setting Policies **5 Aspects of SD** Service Solutions, Processes, Measurement & Metrics, Architectures, Management Info & tools **SD PROCESSES:** 1. [*Design coordination*] Guideline on How to design 2. [*Service catalogue management*] Menu of services: Consists of Business + Technical 3. [*Service Level management*] (SLM) Interface between IT & Business: SLR, SLA, OLA, SLAM, SIP 4. [*Availability management*] Ensure uptime service levels meet or exceed agreed Business needs 5. [*Capacity management*] Performance of Service • Business • Service • Component 6. [*IT service continuity management*] What is needed in a disaster 7. [*Information security management*] Security Policy, CIA (Confidentiality, Integrity, Availability) 8. [*Supplier management*] Manage 3rd Parties – SCD (Supplier & Contract Database) UC **4 Ps-** **P**roducts, **P**eople, **P**artners **P**rocesses	**(Managing Moves)** - Reduce performance variations - Set customer expectations **ST PROCESSES:** 1. [*Transition planning and support*] Offering complete Planning for Service Transition and to manage resources 2. [*Change management*] Control, Assess, Review, Authorise, Manage all changes 3. [*Service asset & configuration management*] Accurate Information Configuration Management System (CMS) about services & Configuration Item (CI) relationships Definite Media Library (DML) 4. [*Release & deployment management*] • Plan • Build and Test • Deploy • Review and Close 5. [*Service validation and testing*] 6. [*Change evaluation*] 7. [*Knowledge management*] Manages Information and Knowledge **– DIKW** (**D**ata, **I**nformation, **K**nowledge, **W**isdom) **There is NO tool to measure Wisdom**	**(Keeping it Alive)** Day to day operations of services **SO PROCESSES:** 1.[*Event management*] Automate routine communication (Informational, Warning, Exception) 2. [*Incident Management*] Restore service with minimum impact on business 3. [*Request fulfilment*] Deals with requests from users (Pre-approved changes / Access requests) 4. [*Problem management*] Root Cause, Known Error Database (KEDB), Workaround 5. [*Access management*] **(AIRDS)** Access, Identity, Rights, Directory service, Service Groups **FUNCTIONS IN SO** **Service Desk:** Restores service to users in the shortest possible time **Types of SD:** Local, Centralized, Virtual, Follow the sun, Specialized **Technical Management** Support the ongoing operations of the IT Infrastructure lifecycle. **Application Management** Managing applications throughout the lifecycle. **Operations Management** - Operations Control (Operators / Monitoring) - Facilities Management (Physical IT environment: Data Centres & Computer Rooms)	**(Learning & Improvement)** (Pain Points / Ideas for Improvement) Provides input to all phases of the lifecycle **CSI REGISTER** Keeps records of all improvement opportunities To Strategy as Input **CSI Model / Approach** Model is based on **6** questions on "how to improve": 1. What is the vision? 2. Where are we now? 3. Where do we want to be? 4. How do we get there? 5. Did we get there? 6. How do we keep the momentum going? **Deming Cycle** Consists of **4** key phases called **PDCA** for quality control and consolidation (**P**lan, **D**o, **C**heck, **A**ct) **Measurement & Metrics There are 3 types of metrics**: Service, Process, Technology **Baseline:** starting point Used for later comparison **SSi PROCESS:** **1.Process [*Seven-step improvement process*]** Identify, Define, Gather, Process, Analyse, Present, Improvements

Synthesized by the researcher from ITIL version 3 manuals (SS, SD, ST, SO & CSI)

Annexure F: Deming's Red Bead Experiment

Red Bead Experiment Overview

In 1982, Dr Edwards W. Deming created a teaching tool for his seminars around the world. These seminars, to teach his famous 14 Obligations of Management, were called the 'The Red Bead Experiment' or 'Red Bead Game'.

The red bead, in this context, is a metaphor for a process in the workplace that is operating sub-optimally. Red beads can also be understood as the lack of a process that needs to be implemented in order to ensure a workplace is operating more optimally.

Deming starts the experiment with a container filled up with a large number of white beads. He then adds a small number of red beads into the container and mixes them all. Then he elects members from his audience to come up onto the stage and assume various managerial and worker roles. When the game is played, each player is asked to use a specially designed metal paddle to scoop out the beads, but the goal is to remove as many of the red beads as possible. Each scoop constitutes a day's work, and statistics of each are visible on a board. Each time the paddle is used, it draws 50 beads from the bowl, some white and some red. The white beads represent good things experienced in a day (good processes), while the red beads represent bad experiences (problematic processes). Understandably, there are always different mixes of red and white beads every time a draw occurs. Deming illustrates to the audience that a manager can make much fuss about the amounts of red and white beads found in each draw by berating bad draws (negative performance indicators) or rewarding good draws (positive performance indicators). However, the essential point and wisdom of the experiment are that the statistical focus on each draw is fallacious. I.e. regardless of the management fuss surrounding the drawing process, the drawer (worker) has no control over the draw. Only the manager (executive) can change the outcome and this is simply done by removing the red beads from the container (system).

Red Bead Experiment Observations

1. All the variations came from the process. There was no evidence that any worker was better than another.

2. The workers who were not allowed to remove the red beads manually, except with the paddle, could not do any better. The best people doing their best does not matter. Therefore, managers rushing to blame employees is counterproductive. Ultimately, as managers are the only people who can make a difference to the scenario, they must strive to improve the processes and make them so robust that it produces acceptable statistics (products), no matter who runs it. So, when a problem with a process occurs, the goal should be first to investigate what went wrong with the process. If the process is found to be in order, then the manager can begin to determine if there was an operator error.
3. Pay for performance in an environment that is operated as the red bead experiment can be futile. The performance of the workers is governed by the process which is not operating optimally.
4. Inspection after the process is complete does not improve quality but merely catches defects before they leave the plant. The quality inspectors in the red bead experiment were not adding value to the process. They are just there to make sure defective products did not reach the customer. Since no inspection process is perfect, one can assume that even with two quality inspectors, some defective products will still make it to the customer. As managers, one must instil quality efforts at all stages of the process to detect defects as soon as they are made, rather than discovering them after having performed more valued added activities to them. The beads may have been defective when they were received from a supplier, but with 'end-of-the-line' inspection, these defective beads are discovered only after having wasted a lot of time and effort working on them.
5. Clear instructions to workers will only increase the probability that the process will behave as intended. Clear instructions will not improve a process that is fundamentally flawed or out of control. As the red bead experiment illustrated, statistically, there were wild variations in production performance from day to day.
6. Intimidation creates fear, which does nothing to improve a process.

7. Praise will encourage a person to perform the process as they have learned to perform it. It will not improve the process.
8. Banners and Slogans raise the awareness of quality as an issue to be concerned with but also tells people that management believes that a reminder is required to produce a quality product, thus creating an environment of mistrust.
9. Incentives will not improve a process and have a short effect on employee morale.
10. The process has natural variation. Data must be collected about the process to understand the range and variance of the variation.
11. To satisfy the customer consistently, the process must be capable of meeting customer requirements. If the customer's requirements are tighter than can be produced consistently, one will only produce acceptable products by accident.
12. Only management can change the system.

Adapted by the writer from https://deming.org/explore/red-bead-experiment.

Annexure G: Deming’s 14 Observations for Management

These observations explain the mindset required to move towards higher levels of productivity and quality. The essential lesson is that management must create the change as generally on the shop floor, workers will be unable to do this.

1. Create constancy of purpose toward improvement of product and service, with the aim to become competitive, stay in business, and provide jobs.
2. Adopt the new philosophy. We are in a new economic age. Western management must awaken to the challenge, learn its responsibilities, and take on leadership for change.
3. Cease dependence on inspection to achieve quality. Eliminate the need for inspection on a mass basis by building quality into the product in the first place.
4. End the practice of awarding business based on the price tag. Instead, minimize total cost. Move toward a single supplier for any one item on a long-term relationship of loyalty and trust.
5. Improve the system of production and service constantly and forever to improve quality and productivity, and thus constantly decrease costs.
6. Institute training on the job.
7. Institute leadership (see Point 12). The aim of supervision should be to help people, machines, and gadgets to do a better job. Supervision of management needs an overhaul, as well as supervision of production workers.
8. Drive out fear so that everyone may work effectively for the company.
9. Break down barriers between departments. People in research, design, sales, and production must work as a team to foresee problems of production and in use that may be encountered with the product or service.
10. Eliminate slogans, exhortations, and targets for the workforce asking for zero defects and new levels of productivity. Such exhortations only create adversarial relationships, as the bulk of the causes of low quality and low

productivity belong to the system and thus lie beyond the power of the workforce.

11. Eliminate work standards (quotas) on the factory floor. Substitute leadership.

    Eliminate management by objective. Eliminate management by numbers, numerical goals. Substitute leadership.

12. Remove barriers that rob the hourly worker of their right to pride in workmanship. The responsibility of supervisors must be changed from sheer numbers to quality.

    Remove barriers that rob people in management and engineering of their right to pride in workmanship. This means, inter alia, abolishment of the annual or merit rating and management by objective.

13. Institute a vigorous program of education and self-improvement.

14. Put everybody in the company to work to accomplish the transformation. The transformation is everybody's job.

Adapted by the writer from https://deming.org/explore/fourteen-points.

Annexure H: People Capability Maturity Model

The People Capability Maturity Model processes are shown in Table H.1.

The Software Engineering Institute from Carnegie Mellon University developed the People Capability Maturity Model. The P-CMM is a maturity framework patterned after the structure of the CMM so that it focuses on continuously improving the management and development of the human assets of a software or information systems organisation. The P-CMM provides guidance on how to continuously improve the ability of software organisations to attract, develop, motivate, organise, and retain the talent needed to steadily improve their software development capability (Curtis & Hefley 2001).

Of specific interest to this work is the Empowered Workgroup, which is found at CM L4. Many processes need to be handled before staff are more fully empowered, as illustrated in Table H.1.

Table H.1: People Capability Maturity Model

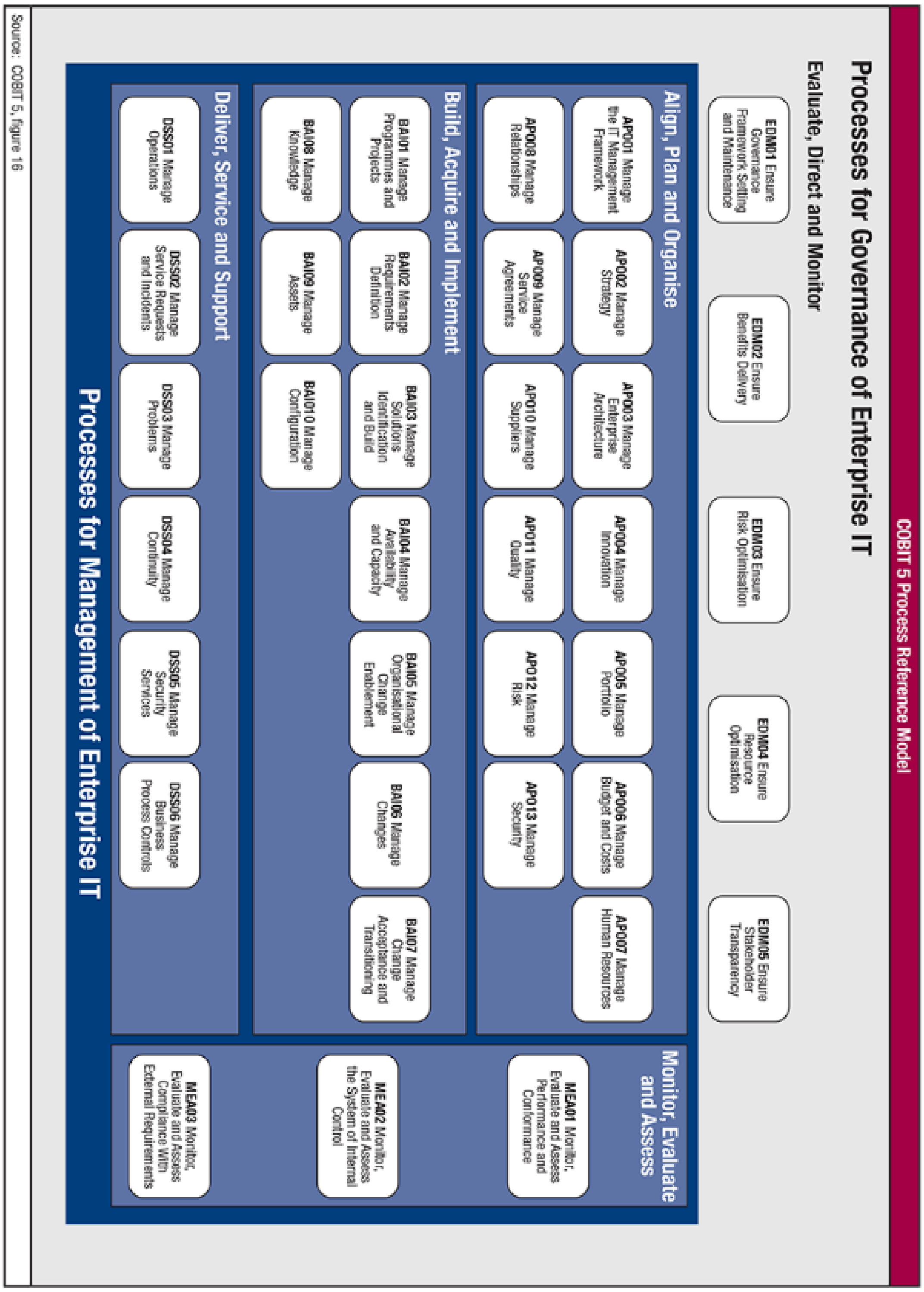
Maturity Levels	P-CMM Objectives - a roadmap towards Employer of Choice			
	Developing Individual Capability	Building Workgroups & Culture	Motivating & Managing Performance	Shaping the Workforce
5 Optimizing	Continuous Capability Improvement		Organizational Performance Alignment	Continuous Workforce Innovation
4 Predictable	Mentoring Competency Based Assets	Competency Integration Empowered Workgroups	Quantitative Performance Management	Organizational Capability Management
3 Defined	Competency Development Competency Analysis	Workgroup Development Participatory Culture	Competency Based Practices Career Development	Workforce Planning
2 Managed	Training & Development	Communication & Coordination	Compensation Performance Management Work Environment	Staffing

Synthesized by the researcher from People Capability Maturity Model (2009) & Curtis & Hefley (2001)

Annexure I: COBIT 5 Processes

Table I.1 is a summary of the COBIT processes. COBIT is a model for control of the IT environment. In developing COBIT, standards from different sources have been used, each covering a part of the information. COBIT supports IT governance by providing a comprehensive description of the control objectives for IT processes and by offering the possibility of examining the maturity of these processes. It helps in understanding, assessing and managing the risks together with the benefits associated with information and related IT. COBIT provides an IT governance instrument that allows managers to bridge the gap with respect to control requirements, information systems (IS) & information technology (IT) issues, and business risks, in order to communicate that level of control to stakeholders. It enables the development of clear policy and good practice for the control of IT throughout organisations. (ICASA, 2011).

Table I.1: COBIT 5 Processes



Source ICASA, 2011:33

Annexure J: PMI & PMBOK - More Detail

The Project Management Body of Knowledge Guide (PMBOK®) Version 5 (2013) & Version 6 (2017) is compiled by the Project Management Institute (PMI). The Project Management Institute was incorporated in Pennsylvania in 1969. While several other publications were also produced by the PMI, the PMBOK is the institute's single most important publication. The PMBOK, as an ANSI standard, is primarily used as a Body of Knowledge. It is also used by students to prepare for the Certified Associate in Project Management (CAPM) and Project Management Professional (PMP) exams. The first PMBOK was released in 1983. The latest PMBOK Guide is now in its 6th Edition, released in 2017. The new PMBOK® Guide 6th edition contains 978 pages, including the Agile Practice Guide (186 pages). The PMBOK 6th edition is regarded as an important (major) update. Table J.1 illustrates the PMBOK journey since 1996.

Table J.1: PMBOK Editions from 1996 to 2017

Year	Edition	Project Groups	Knowledge Areas	Processes	Pages
1996	1996	5	9	37	~ 180
2000	2000	5	9	39	~ 210
2004	Third	5	9	44	~ 400
2008	Fourth	5	9	42	~ 460
2013	Fifth	5	10	47	~ 620
2017	Sixth + Agile	5	10	49	~ 980

Source: Project Management Institute (2017)

Table J.2 illustrates how the Develop Schedule process, one of the 49 processes that make up the PMBOK 6, works from an Inputs, recommended Tools and Techniques, and Outputs perspective.

Table J.2: PMBOK 6 Process 6.5 ITTO

Schedule Planning Process – Develop Schedule #6.5				
INITIATION	**PLANNING**	**EXECUTION**	**MONITOR & CONTROL**	**CLOSING**
Process name	**Inputs**	**Tools and techniques**	**Outputs**	**Knowledge Area**
Develop Schedule	.1 Project management plan • Schedule management plan • Scope baseline	.1 Schedule network analysis	.1 Schedule baseline	Project Schedule Management
	.2 Project documents • Activity attributes • Activity list • Assumption log • Basis of estimates • Duration estimates • Lessons learned register • Milestone list • Project schedule network diagrams • Project team assignments • Resource calendars • Resource requirements • Risk register	.2 Critical path method	.2 Project schedule	
	.3 Agreements	.3 Resource optimization	.3 Schedule data	
	.4 Enterprise environmental factors	.4 Data analysis • What-if scenario analysis • Simulation	.4 Project calendars	
	.5 Organizational process assets	.5 Leads and lags	.5 Change requests	
		.6 Schedule compression	.6 Project management plan updates • Schedule management plan • Cost baseline	
		.7 Project management information system	.7 Project document updates • Activity attributes • Assumption log • Duration estimates • Lessons learned register • Resource requirements • Risk register	
		.8 Agile release planning		

Source: Project Management Institute, 2017

It is noteworthy that the ITTO (Inputs, Tools and Techniques, and Outputs) above clearly stipulates that a Project Management Information System is an important Tool and Technique that must receive consideration. In the PMBOK 5 guide, only process 4.3 indicated the need for a Project Management Information System. In PMBOK 6, twelve processes now refer to the fact that a Project Management Information System is important. It is also clear from the outputs from process #6.5 that schedule and resource baselines are required in order to proceed into the Execution Process Group. With these baselines in place, the Preliminary project Schedule process (#6.5) is now elevated to a developed schedule ready for application in the Execution Process Group.

The PMBOK is one of several publications compiled by the PMI. For some of the important Knowledge Areas, of which there are ten, including Project Integration Management, some have Practice Management Standards available. The Practice Standard for Scheduling 2nd Edition (2007) and the Practice Standard for Earned Value Management 2nd Edition (2011) have been used in this work. Regular publications, conferences, research, and new information produced by its members means the PMI is coordinating the effort of its members to improve the body of knowledge.

Annexure K: PRINCE2 & PRINCE2 Agile - More Detail

PRINCE2 (Projects IN Controlled Environments: Office of Government Commerce 2009) is a structured project management method endorsed by the United Kingdom as the project management standard for projects that need to be run in the Public Sector. The Central Computer and Telecommunications Agency (CCTA) first published PRINCE2 in 1989. PRINCE2 is derived from a combination of PROMPTII and the original PRINCE project management method. The goal was a UK Government standard for Information Technology (IT) project management.

The version of PRINCE2 released in 1996 formed a more fine-tuned and generic project management method. PRINCE2 has become increasingly popular and is currently the official standard for project management in the UK. Over time, the CCTA became the Office of Government Commerce (OGC). Recently the OGC handed the PRINCE2 methodology over to AXELOS as a joint venture in 2014 to develop, manage, and operate qualifications in best practice in methodologies. AXELOS currently offers several certifications, which include Projects IN Controlled Environments (PRINCE2), Managing Successful Programmes (MSP), Management of Risk (M_o_R), Portfolio Management (MoP), Value Management (MoV), Information Technology Infrastructure Library (ITIL), Portfolio, Programme and Project Offices (P3O), P3M3, RESILIA, and PRINCE2 Agile.

The PRINCE2™ method has seven principles, seven themes, and seven processes. The seven principles include continued business justification, learning from experience, defined roles and responsibilities, managing by stages, managing by exception, focusing on the creation of products, and tailoring to suit the project environment. The seven themes include the business case (why), the organisation (who), quality (what), plans (how, how much, when), risk (what if), change (what is the impact), and progress (on target, tolerance). The seven processes include starting up a project (SU), initiating a project (IP), directing a project (DP), managing stage boundaries (MSB) and controlling a stage (CS), managing product delivery (MP), and closing a project (CP).

Figure K.1 is an overview of the process interactions that occur within PRINCE2.

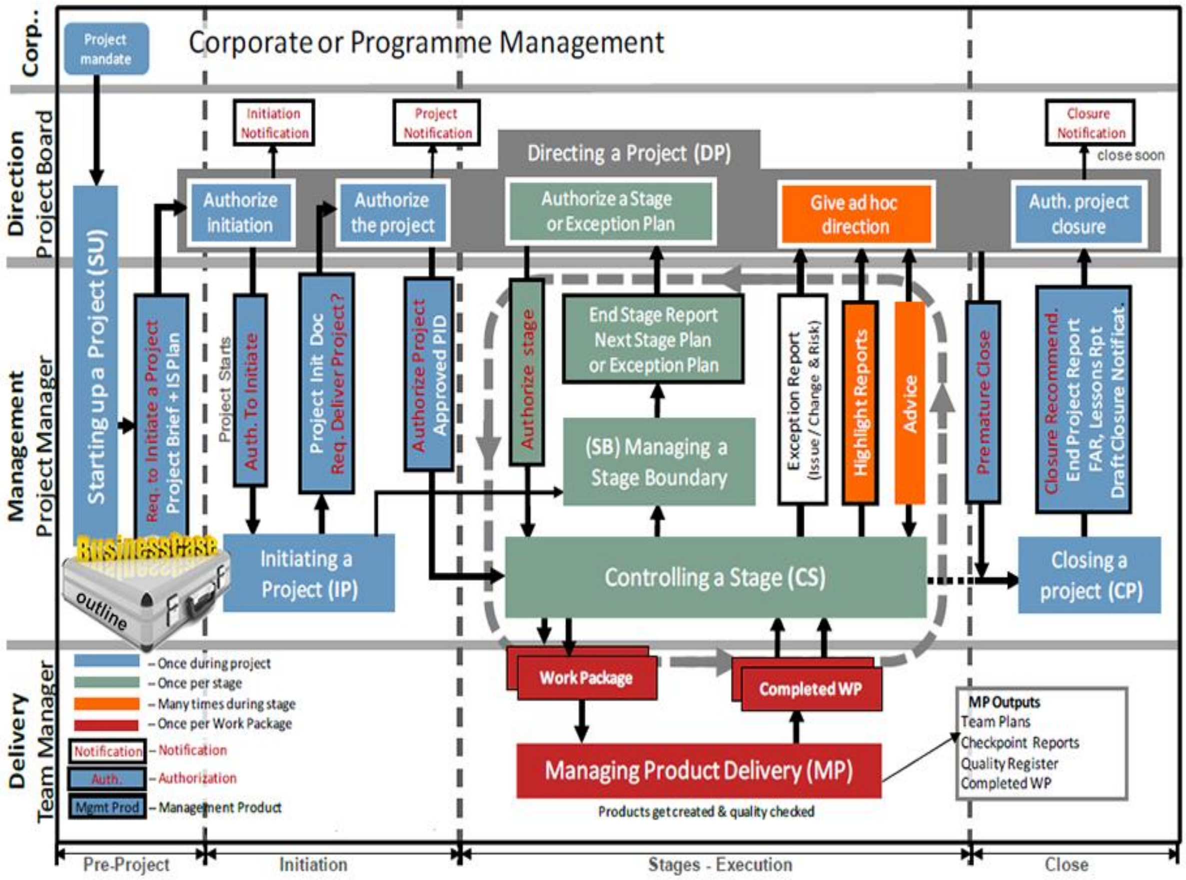


Figure K.1: PRINCE2 Process Interactions

Source: Turley (2010:9)

One of the many strengths of PRINCE2 is its Directing Process (DP). Project roles and responsibilities, and the project mandate is derived through the Starting Up a Project (SU) process that requires, as its initiation process, the constitution of the PRINCE2 Project Management Board. This occurs upfront before the project starts.

The Board, headed by the Executive, has a Senior User role to represent the business requirement and a Senior Supplier role to represent the supply of services needed to achieve delivery of the requirement. The Project Management and Delivery processes are at lower levels, reporting into and obtaining Direction from the Board as shown in Figure K.2.

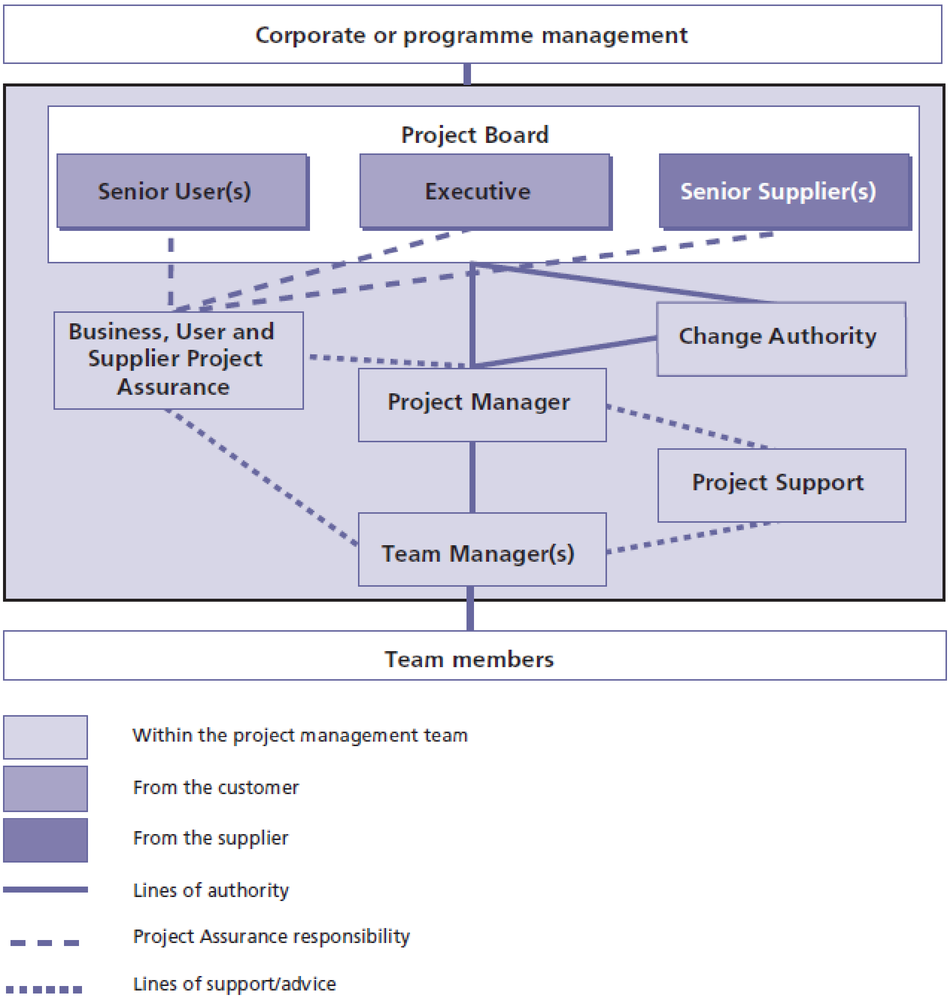


Figure K.2: PRINCE2 Board

Source: Office of Government Commerce (2009:33)

Since a PRINCE2 project should only be run if it produces value, and as PRINCE2 benefits are reviewed for their value add at a future time after a project is concluded (per the PRINCE2 'Benefit Review Plan'), the Board is understood to be the correct entity to manage this. "A Benefits Review Plan is used to define how and when a measurement of the achievement of the project's benefits, expected by the Senior User, can be made." (Office of Government Commerce, 2009:235)

PRINCE2 Agile (2015) is a recent version of PRINCE2, which cater to the perceived need for agility in its customer base.

Figure K.3 succinctly shows how PRINCE2 Agile differs from PRINCE2. The Team Manager uses agile within the lowest delivery layer of the PRINCE2 method to support the project manager (middle management layer), who reports on progress to the

Project Board (top, Directing layer). PRINCE2 project process and Governance is exerted downwards by the Board. Responding to this, execution processes and techniques, facilitated by the PRINCE2 Agile approach, push up from the delivery level. These two forces produce a blend of Traditional Project Management (TPM) in a controlled environment, and Agile Project Management (APM), after the project products are agreed, facilitate the production of value by the methodology.

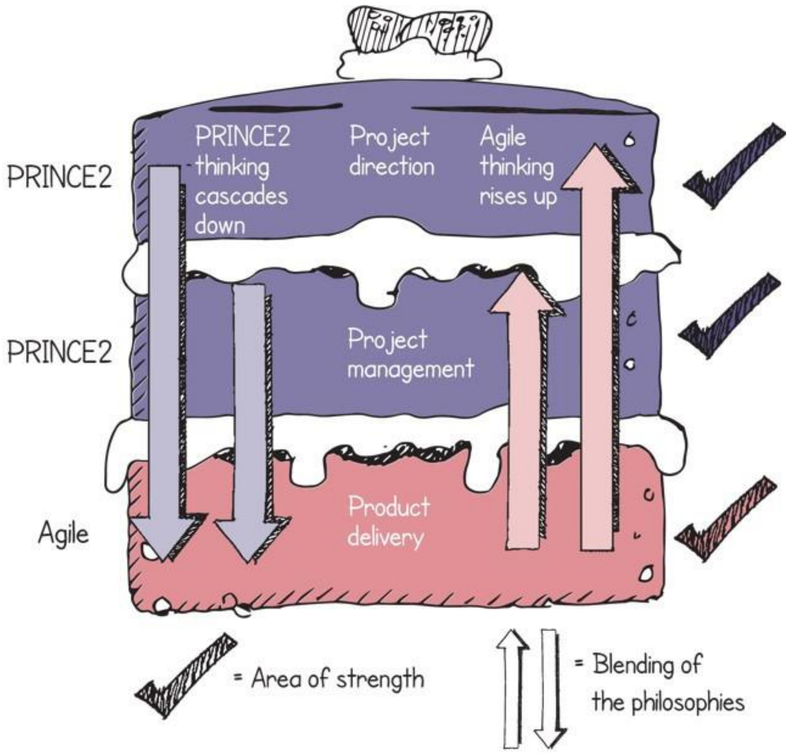


Figure K.3: PRINCE2 Agile

Source: Richards & Cooper (2015:17)

PRINCE2 Agile, as a methodology, includes information on many agile techniques. These include Kanban, Lean, and others.

PRINCE2 Agile has received criticism from the agile community, stating that the method was simply traditional masquerading as agile. The researcher successfully ran a software project for the Presidency in the South African Government over a period of two years that was run on a combination of PRINCE2 and PRINCE2 Agile. At the time, the researcher was faced with stakeholders who were insisting on an agile approach, yet tight governance was also a requirement of the Public Finance Management Act. Running the project, the researcher appreciated the wisdom from the PRINCE2 Agile manual (2015:22), which asks the reader to beware of prejudice.

The advice continues to state that one would think that bringing more control and governance into the agile domain could prove counterproductive. However, PRINCE2 Agile represents a marriage that is based on the opposite view that control and governance allow agile to be used in more situations, such as those involving multiple teams or complex environments. From the manual: A fighter aircraft is built with a deliberately unstable airframe. This instability gives it agility and allows it to change direction easily and adapt quickly to situations. However, to do this still requires control and governance.

Annexure L: Combining the PMBOK & PRINCE2

It is possible to combine methodologies in order to produce an amalgam, which is more powerful than its parts. A combined approach using PMBOK and PRINCE2 illustrates how this could work. PRINCE2 is a powerful and elegant model for project control. PMBOK is a structured and detailed collection of project planning and tracking processes with clearly defined Inputs, Tools and Techniques, and Outputs. The PMBOK does not say exactly how to wrap project controls around a project. PRINCE2 explains exactly the method that must be followed. “Together PRINCE2™ and PMBOK® could be what is needed in the ‘boiler-room’ of corporate change!” (LogicalModel.net 2016) Using PRINCE2 as the recipe and PMBOK for the ingredients.

Annexure M: Carnegie Mellon SEI & CMMi

The Capability Maturity Model integrated (CMMi) process improvement suite from Carnegie Mellon University Software Engineering Institute (SEI) offers a process improvement path to all key areas of the organisation. Currently, there are three 'flavours' of CMMi called constellations. Each constellation has content that targets improvements in particular areas attuned to the organisation focus:

- Capability Maturity Model integrated for Development – I.e. 'DEV' (Develops products and complex services),
- Capability Maturity Model integrated for Acquisition – I.e. 'ACQ' (The acquisition of goods and services from others)
- Capability Maturity Model integrated for Services – I.e. 'SVC' (The provision and delivery of services)

People Capability Maturity Model (P-CMM), still on a previous version when compared to the constellations above, which are version 3, focuses on the human resources of a business charged with delivery of Capability Maturity Model integrated processes.

The assumption with the Capability Maturity Model integrated is that the organisation has its own standards, processes, and procedures by which they get things done. The content of the CMMi is to improve the performance of those standards, processes, and procedures – not to define exactly how a company must use them. CMMi cannot tell an organisation what is or is not important to them. CMMi, however, can provide a path for an organisation to achieve its performance improvement goals.

Annexure N: DSDM - More Detail

The Dynamic Systems Development Method (DSDM) is one of the oldest agile methods and has been around since 1991. The DSDM method is defined in the Agile Project Management Handbook Version 2 (2014), which is maintained by the DSDM Consortium.

Of the many agile methods available, DSDM adopts the standpoint of Enough Documentation Up Front (EDUF) to deliver against requirements within an agreed timeframe. Requirements are prioritized using the MoSCoW method. The MoSCoW method decides Must Have, Should Have, Could Have, Won't Have.

A high-level overview of DSDM, created by the writer from the DSDM manual, is found in Figure N.1.

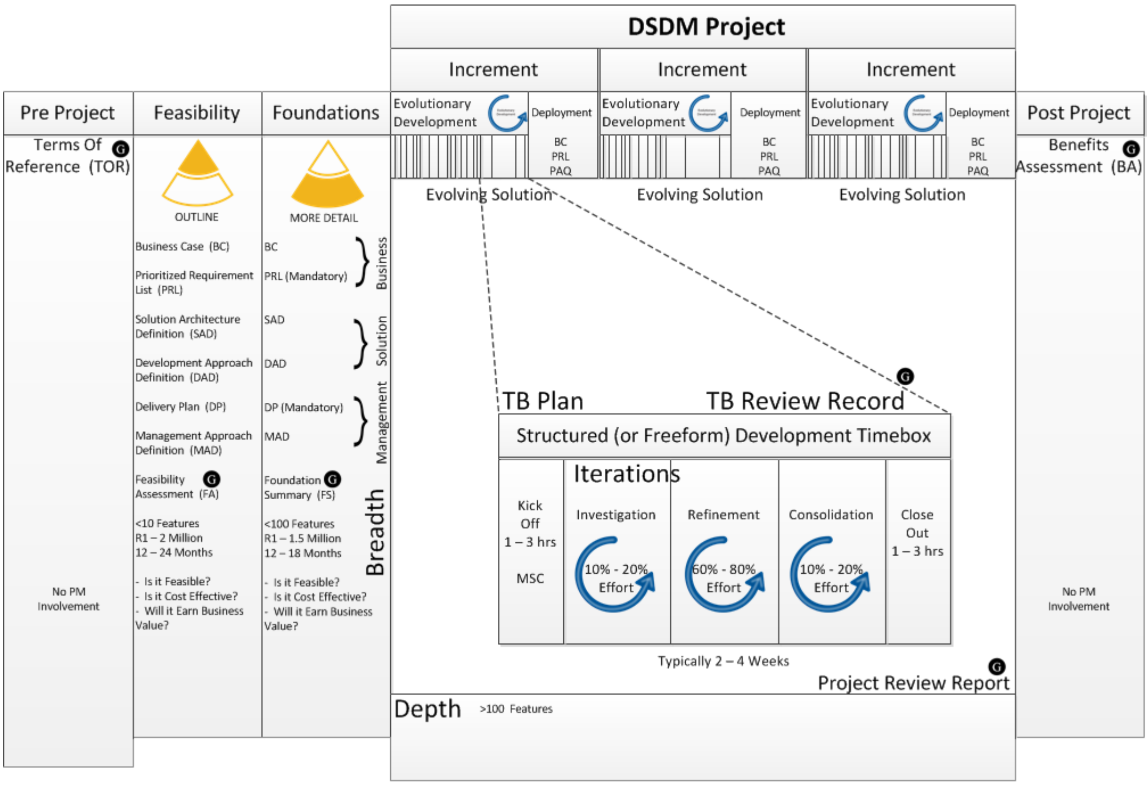


Figure N.1: DSDM Process Model

Source: Synthesized by the researcher from DSDM Manual (2014:16)

As with other agile teams, the DSDM team is understood to be a high functioning, capable, and empowered team with all the necessary skills residing within the team. The DSDM team model, referred to as the DSDM Snowman (as it looks like a snowman), operates at the project level, the solution development team level, and a support team level.

The eight principles of DSDM (2014:16) support a philosophy that maintains “Best business value emerges when projects are aligned to clear business goals, deliver frequently, and involve the collaboration of motivated and empowered people.” The eight principles include focusing on the business need, delivering on time, collaboration, never compromising on quality, building incrementally from firm foundations, developing iteratively, communicating continuously and clearly, and demonstrating control.

Annexure O: SCRUM - More Detail

Scrum is an agile project management methodology used primarily for software development projects but can be used for any type of project.

Scrum principles are regarded as mandatory and must be used in all Scrum projects. The six Scrum principles include empirical process control, self-organisation, collaboration, value-based prioritisation, time-boxing, and iterative development.

The goal of the Scrum team is to deliver new software capability every one to six weeks. As with DSDM, Scrum utilises time-boxing to focus effort, which is called sprints. The analogy of Scrum and its methods to the game of Rugby is intentional.

According to the 12th Annual State of Agile report, it is estimated that 70% of software teams make use of the Scrum method or a Scrum in a hybrid form. Several organisations maintain that they are best placed to house and publish Scrum Knowledge. The researcher, who was scrum certified through SCRUMStudy, notes that this organisation has compiled a detailed Body of Knowledge, which, in its 3rd edition and 403 pages in length, is regarded by him as an invaluable resource on the Scrum method. This Scrum Body of Knowledge contains detail on Scrum Principles, Aspects, Roles, and the Scrum Processes, including Inputs, Tools, and Outputs (Setpathy 2016). On the other hand, founders of Scrum Ken Schwaber and Jeff Sutherland's Scrum Guide (2017) is compact and easy to assimilate, being only 19 pages in length.

The Scrum team is typically composed of no more than nine members. The core roles in Scrum are the Scrum Product Owner, the Scrum Master, and the Scrum Developers. The Product Owner usually represents an internal or external customer. There is only one Product Owner who must convey the overall mission and vision for the Scrum team. The Product Owner is ultimately accountable for managing the Product backlog, a prioritized list of all the requirements that have been identified.

The Scrum team's focus is on the Scrum backlog, which is a list of project products selected from a backlog list and compiled in User Story format by the Product Owner. The User Stories as products on the Scrum backlog must be delivered at the end of a sprint, if possible. While the Scrum Master is a servant leader to the team, whose role it is to run daily stand-up meetings and to assist the team, it is ultimately the team who decides how to organise themselves to deliver results.

The Scrum Master as a facilitator works closely with the Product Owner, Development Team, and the Organisation. The Scrum Master ensures that the team adheres to Scrum theory, best practices, and principles and ensures that Scrum rules, as stipulated in the Scrum Guide or Scrum Body of Knowledge, are followed. The Scrum Master protects the team and assists them to perform optimally. This can include removing impediments, helping the Product Owner groom the backlog, and facilitating regular meetings. The relationship between the Product Owner, The Scrum Master, and the Scrum Team is illustrated in Figure O.1.

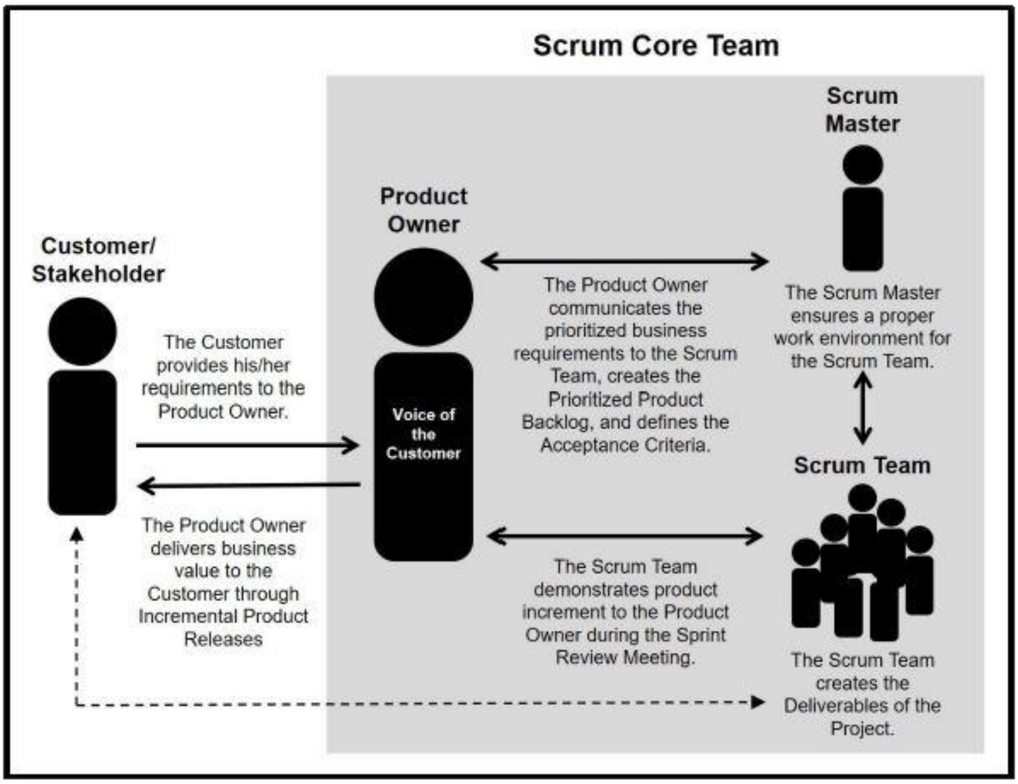


Figure O.1: Scum Roles and Goal

Source: Setpathy (2016:63)

According to Scrum Process #16, Demonstrate and Validate Sprint, this is a crucial process as it ensures that the Scrum team demonstrate working software to the Product Owner at the end of each sprint. This occurs during the Sprint Review meeting. A Scrum Retrospective meeting is also held at the end of each sprint where the team discuss the previous sprint and how it can be improved upon moving into future sprints.

What is noteworthy about Scrum, and especially the latest Scrum Body of Knowledge 3rd release, is that Scrum Processes and Inputs, Tools, and Outputs have been defined in detail. Table O.1 illustrates that Scrum has similar Initiating, Planning and Monitoring and Controlling Processes to those found in the PMBOK 6 Guide. Therefore, for the purposes of this research, Scrum undertaken without due consideration to Project Planning (PP) and Project Monitoring and Control (PMC) from a CMMi perspective must be run sub-optimally at CM L1.

Table O.1: Scrum Process

Chapter	Phase	Fundamental Scrum Processes
8	Initiate	1. Create Project Vision 2. Identify Scrum Master and Stakeholder(s) 3. Form Scrum Team 4. Develop Epic(s) 5. Create Prioritized Product Backlog 6. Conduct Release Planning
9	Plan and Estimate	7. Create User Stories 8. Estimate User Stories 9. Commit User Stories 10. Identify Tasks 11. Estimate Tasks 12. Create Sprint Backlog
10	Implement	13. Create Deliverables 14. Conduct Daily Standup 15. Groom Prioritized Product Backlog
11	Review and Retrospect	16. Demonstrate and Validate Sprint 17. Retrospect Sprint
12	Release	18. Ship Deliverables 19. Retrospect Project

Source: Setpathy (2016:36)

The Agile Scrum processes in Table O.1 operate according to the flow found in Figure O.2.

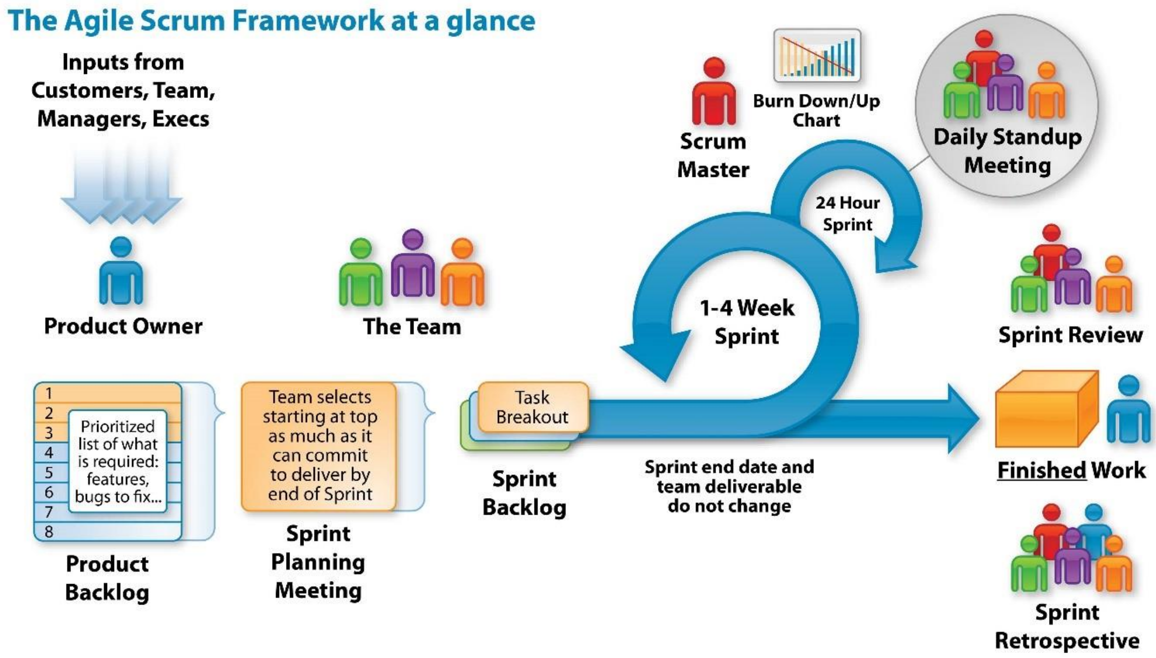


Figure O.2: Agile Scrum Framework and Process Flow

Source: Markovic (2018)

In a recent publication, Mastering Professional Scrum (Ockerman & Reindl 2019), the authors focused on the Scrum Principle of Empirical Process Control as an area of delinquency that should be urgently attended to and improved. According to the authors, seven types of dysfunctional scrum teams existed, which included the Un-Done Scrum, the Mechanical or (Zombie) Scrum, the Dogmatic Scrum, One-Size-Fits-All Scrum, Water-Scrum-Fall, Good Enough Scrum, and Snowflake Scrum. Figure O.3 shows clearly why the Un-Done Scrum dysfunction will introduce risk if work remains undone and not ready for release into production at the end of a sprint.

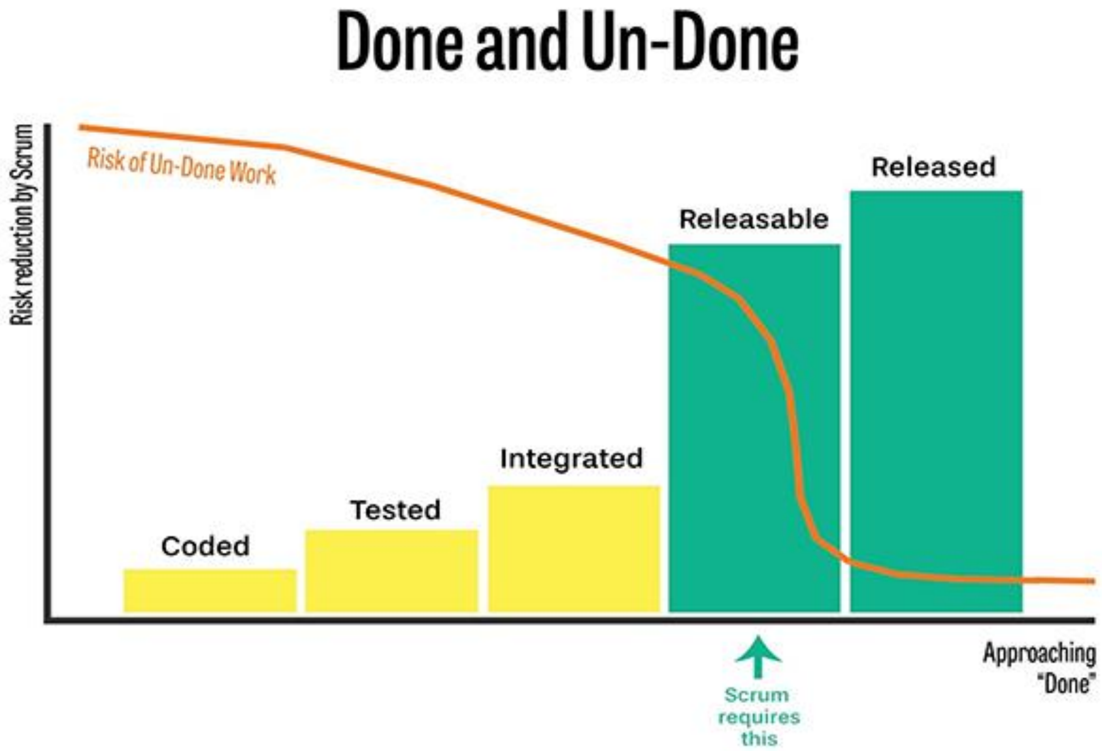


Figure O.3: Done and Un-Done Scrum

Source: Verwijs (2018)

The goal of the Scrum Principle Empirical Process Control is to be able to focus on work that is completed in the sprint to inspect it and then adapt a next approach around this work during the next sprint if required. This inspect and adapt approach is crucial as Scrum (like other Agile methods) does not spend too much time in detailed planning. At the end of a sprint, if the software being built is unavailable to the Product Owner and team to inspect in a format ready for release, then no value can be said to have been produced by the sprint. This problem and associated risk can become worse if the situation continues over many sprints, as is illustrated in Figure O.4.

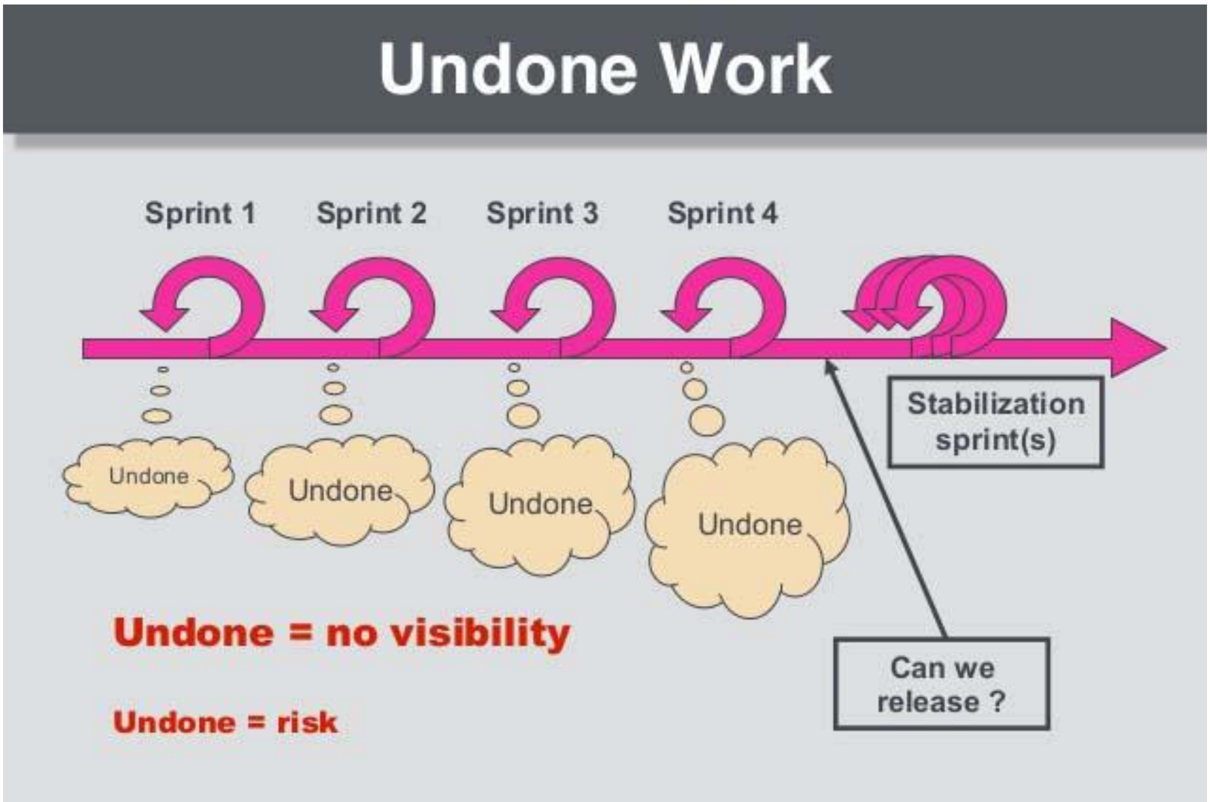


Figure O.4: Undone Work Carried Over

Source: Kirschenbaum (2020)

It stands to reason that if Requirements Management (REQM), a CMMi L2 process in Agile, is being managed based on No Documentation Up Front, this could introduce risk.

Annexure P: Extreme Programming (XP) - More Detail

Extreme Programming (XP) gained traction in the 1990s. XP originated from the Chrysler Corporation.

Extreme Programming (XP) requires highly competent team members, each with excellent knowledge of software engineering, software programming, and development practices in general. There is a high focus on verbal communication within XP teams, which are small and focused. The team goal focuses on close collaboration. Ever-evolving design is the goal using incremental development, flexible scheduling, and automated test codes. XP teams are motivated at all times to give rapid feedback, keep it simple, and have courage. The different roles in the approach include customer, developer, tracker, and coach. The XP method, well-documented via its website ("Extreme Programming Web Site" 2019), has been widely adopted due to its well-defined engineering practices.

Annexure Q: The Researcher's PMWay website

To assist with his dissertation, the researcher built a Project and Process Management website as a sandbox to investigate a number of the concepts presented in this work. As the site grew, the cost of hosting became unaffordable, and it was transferred from public to home hosting at the end of 2020. As such, the website is only available when the researcher brings it online, typically from 7 AM to 5 PM, Monday to Friday.

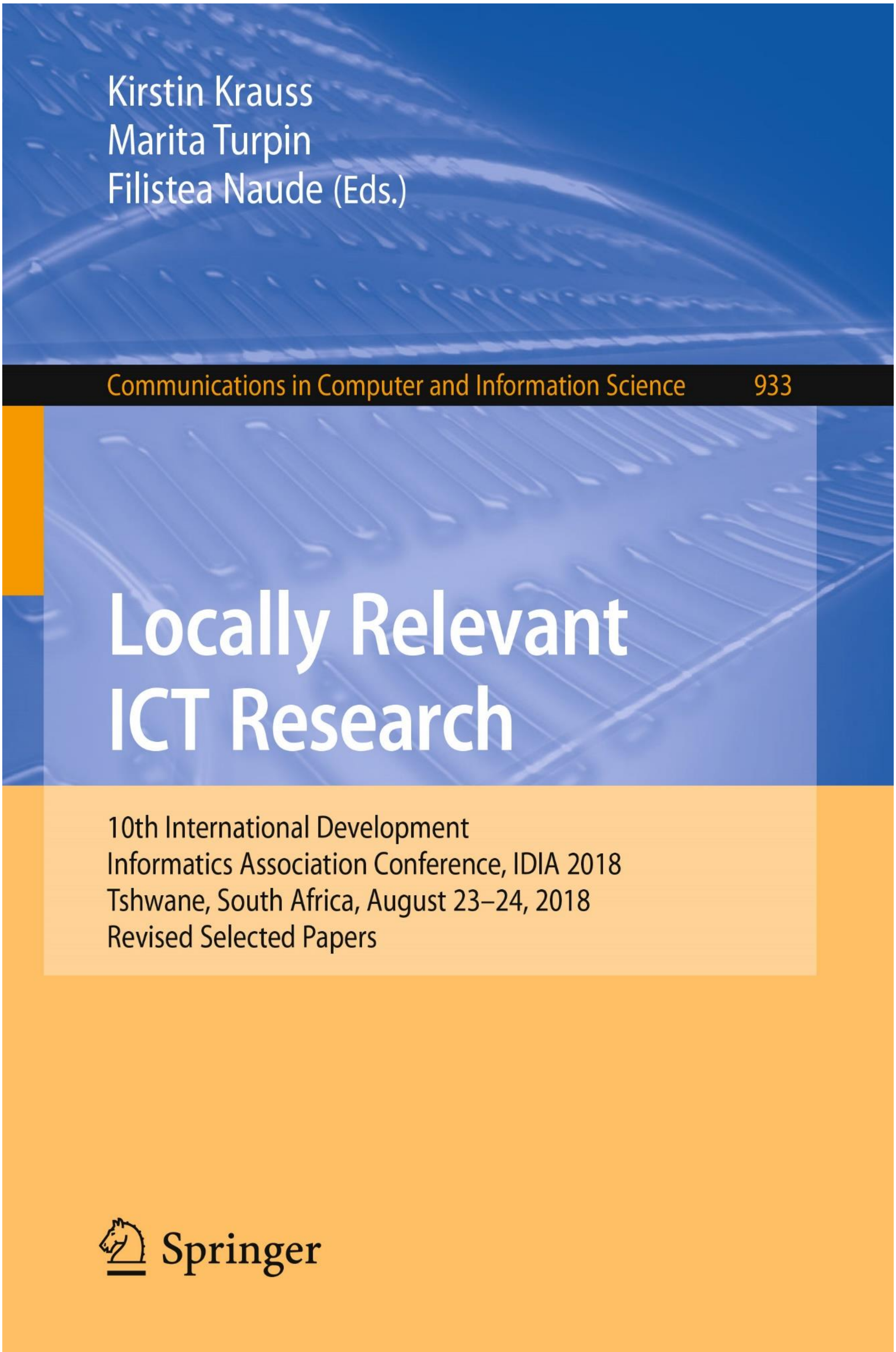
If access is required, the website can be made available to a reader via a request directed to the researcher's private email: **markjc@mweb.co.za** or his UNISA email: **4151690@mylife.unisa.ac.za**. The internet address for the website is **https://pmway.hopto.org.**

It is worth noting that the researcher was awarded Professional Development Units (PDU's) by the PMI towards his Project Management Professional certification for the website when it was hosted publicly. The high-level slideshow, accessible at the bottom of the home page (without needing to log into the website), is an attempt to summarise the findings from the website. These, in turn, where appropriate, have found their way into this dissertation.

Annexure R: Journal Article Published

The following publication in Appendix R emanated from this research:

Corrigan M.J., van der Poll J.A., Mtsweni E.S. (2019) The Project Management Information System as Enabler for ICT4D Achievement at Capability Maturity Level 2 and Above. In: Krauss K., Turpin M., Naude F. (eds) Locally Relevant ICT Research. IDIA 2018. *Communications in Computer and Information Science*, vol 933. Springer, Cham. https://doi.org/10.1007/978-3-030-11235-6_19


Kirstin Krauss
Marita Turpin
Filistea Naude (Eds.)
Communications in Computer and Information Science 933
Locally Relevant
ICT Research
10th International Development
Informatics Association Conference, IDIA 2018
Tshwane, South Africa, August 23–24, 2018
Revised Selected Papers
Springer

The Project Management Information System as Enabler for ICT4D Achievement at Capability Maturity Level 2 and Above

Mark Jonathan Corrigan[1(✉)], John Andrew van der Poll[2], and Emmanuel Samuel Mtsweni[3]

[1] School of Computing, University of South Africa, Phapha Building, 2nd Floor Science Campus, Cnr Christiaan de Wet & Pioneer Streets Florida, Johannesburg, South Africa
4151690@mylife.unisa.ac.za

[2] Graduate School of Business Leadership, University of South Africa, Cnr Janadel and Alexandra Avenues, Midrand, South Africa
vdpolja@unisa.ac.za

[3] School of Computing, University of South Africa, G J Grewel Building, 3rd Floor Science Campus, Cnr Christiaan de Wet & Pioneer Streets Florida, Johannesburg, South Africa
mtswees@unisa.ac.za

Abstract. Current trends in project management promote Agile as a solution for challenges around project delivery, focusing more on individuals and interactions than on processes and tools. This paper postulates that any project method, especially agile, if undertaken with limited process focus below Capability Maturity (CM) Level 2, may well result in project delivery failures, despite best intentions from the project team. With dynamic Internet growth in developing countries, and advancements in virtual collaboration, projects ought to involve tailoring jointly-agreed project management processes at Capability Maturity level 2 and above. In addition, the utilisation of a Project Management Information System (PMIS) emplacement is essential for project management success. The use of a PMIS is in line with a process-driven approach to ICT4D, and will facilitate the ICT value proposition for a developing economy. Following a comprehensive literature review, the paper asserts that Capability Maturity Level 2 will remain relatively unattainable, unless the PMIS emplacement becomes a strategic driver for success. To this end, we propose a PMIS CM Improvement Framework to the PMBOK Process Group and Knowledge Area matrix, and establish the value proposition for ICT4D.

Keywords: ICT4D · Agile · Capability maturity levels · Chaos Productivity · Theory of Constraints · Project management Process management · Project Management Information System Software engineering · Earned value management · Value proposition

K. Krauss et al. (Eds.): IDIA 2018, CCIS 933, pp. 295–310, 2019.
https://doi.org/10.1007/978-3-030-11235-6_19

1 Introduction

The recent dynamic growth of the Internet in developing countries unlocks the numerous benefits of a Project Management Information System (PMIS) emplacement, accessible in real time by a collaborating team. To this end, ICTs have the potential to initiate new developments (cf. ICT4D) in project management professionalism and expertise. Coupled with these, the well-intentioned use of Agile undertaken at CM Level 1 could result in inferior project products from a software engineering quality perspective. Consequently, our premise is that both Traditional and Agile Project Management methods may benefit from a PMIS emplacement, thereby enhancing ICT4D for South Africa as a developing country.

Unlike Business as Usual, the sixth (6^{th}) edition of the Project Management Body of Knowledge Guide (PMBOK®) defines a project as "a temporary endeavour undertaken to create a unique product, service, or result" [1].

The 49 processes of the PMBOK sixth edition are collated in a matrix that combines five Process Groups across ten Knowledge Areas (including one Integration area) down the side. Supporting each process are Inputs; Tools and Techniques; and Outputs. The PMBOK Process Group and Knowledge Area matrix in Fig. 5, below, forms an overview of a project management lifecycle and its constraints [1] (p. 25). This matrix is often simplified to the concept of the Iron Triangle which focuses on Time (Schedule), Cost and Scope processes as the essential managed constraints for the achievement of a project management goal.

The Standish Chaos Reports [2] and Agile Manifesto [3] question the value of processes, tools and, especially, the Iron Triangle, stating that Traditional Project Management with its Waterfall Approach has no agility, and will continue to fail to deliver successful projects. Proponents of these ideas advocate an agile approach as the way forward. Yet, as argued in this paper, the use of a PMIS may go a long way in facilitating increased ICT capability, enhancing a more mature ICT4D approach and ultimately improving the value proposition for developing economies.

Edwards W. Deming, prime mover of the Capability Maturity model estimated that in his experience, "most troubles and most possibilities for improvement add up to the proportions something like this: 94% belongs to the system (responsibility of management) and 6% special" [4] (p. 248); Therefore, once the trouble has hit the floor, the worker level, the workers will have little ability and control to improve anything. Consequently, managerial insistence on a "Just Do IT" agility, if expended at CM Level 1, would imply that effort expended by underpowered or underskilled project personnel, despite their best efforts, would not have the desired effect, and may well lead to project failure. This is especially relevant in the South African context, often suffering from a lack of resources and skilled personnel.

The PMIS emplacement, utilising the strengths of virtual team collaboration, and facilitating focus on the "peopled task" in an environment based on game theory (the Developed Schedule as playing field within the PMIS), could be the solution needed to galvanise project management improvements in capability, maturity and professionalism, for success. In this context, the concept of the "peopled task, project tasking or time asking" is essentially who on the project team agrees to do what, with what and by when. The project playing field, typically a Gantt chart, in planning is populated by the project team tasks comprising professional estimates for project goal achievement. Measured in hours of work, in project execution, task slippage, illustrated graphically in the chart with a slip line, or as a variance against an agreed baseline, would be cause for team concern.

The layout of the rest of the paper is as follows: Our research questions are given in Sect. 1.1. An overview of project management aspects follows in Sect. 2, 3, 4, 5, and 6 These include Project Management being challenged, the Methodology and Agility dimension, the Capability Maturity dimension, the People-Process-Technology triad dimension and the role of the PMIS, especially in the local (SA) context. The main contribution of our work – the development of a PMIS CM Improvement Framework, is presented in Sect. 7. Conclusions and future work, followed by a list of references, conclude the paper.

1.1 Research Questions

Our above discussion leads to three research questions:

- What is the interplay between Agile and traditional project management methodologies (**RQ1**)?
- To what extent may the use of a PMIS (Project Management Information System) facilitate Capability Maturity (CM) Level 2 development, thereby enhancing the ICT value proposition for a developing economy (**RQ2**)?
- How may the PMBOK Process Group and Knowledge Area matrix be enhanced to cater for sustainable Capability and Maturity improvements (**RQ3**)?

2 Literature Review

2.1 Project Management Being Challenged

Since the release of the original Chaos Reports, these have received global attention. Some criticise the reports on statistical inaccuracies and a lack of research process transparency [5]. Most of the reports, however, illustrate a fundamental and deep concern at the inability to control high project costs, often failing to deliver the (ICT) value that has been promised to stakeholders and the community.

The original Standish Group Chaos Report [2] (p. 9) identifies that software development companies often experience numerous project challenges and impaired factors which include, among others, limited support from the Executive Management, a lack of clarity of objectives, an inability to plan correctly, and unrealistic expectations.

Schach identifies managerial challenges as a main result of the Chaos Reports findings. They found that only 35% of projects were successfully completed, and 19% were either cancelled before completion, were over budget, late, had fewer features and functionality than initially specified, or were never implemented. In fact, "just over one in three software development projects was successful and almost half the projects displayed one or more symptoms of the software crisis" [6] (p. 5).

According to Guess [7] (p. 25), the reasons why software development projects failed or where challenged, are attributed to poor user inputs, vague requirements, poor cost and schedule estimates, failure to plan, and late or ignored failure warning signals.

Despite criticisms, the Chaos Reports have caused many organisations to question their approach to project management. There appears to be agreement that renewed focus has caused project management (PM) maturity to rise in certain areas [8] (p. 6), but Capability Maturity (CM) Level 2 ought to be a clear starting point if productivity and quality, and not risk, rework and waste, are the goals stakeholders are aiming for. Agile, while well intentioned as a panacea, if undertaken at (CM) Level 1, could be fallacious; therefore, the strategic use of a PMIS in a software project management (SPM) methodology to achieve Level 2 maturity, holds much promise to alleviate the said challenges and increase the ICT value proposition.

3 Methodology and Agility as a Horizontal Dimension

The PMBOK is often disregarded as 'Traditional Project Management' utilising the 'Waterfall' approach, yet this is not correct. In fact, the latest version of the PMBOK embraces Traditional, *Agile*, *Iterative* and *Adaptive* environments. While the PMBOK is primarily a Body of Knowledge, it is also a method that can accommodate full project lifecycles or project phases (Process Groups) as its 'Close Project or Phase' (process #4.7 – Fig. 5) elucidates. In addition, PMBOK processes can be configured in size, complexity and speed; and process flows can be made to iterate with outputs leading back to prior processes, if needed [1, 9].

Numerous other project management methodologies exist. In this paper, PRINCE2 Agile [10], formerly PRINCE2 [11], Dynamic Systems Development Method (DSDM) [12], SCRUM [13] and Extreme programming (XP) [14] are considered.

Project methodologies are often assessed in terms of their agility. The **PR**ojects **IN** **C**ontrolled **E**nvironments method (**PRINCE**) [11] is tightly process-driven, while its latest edition, PRINCE2 Agile [10], fully embraces Agile, albeit within the Board control and tight governance of the original method. The Dynamic Systems Development (DSDM) [12], SCRUM [13] and Extreme programming (XP) [14] methods conform more closely to pure Agile as defined by the Agile Consortium Manifesto [3].

The Agile Manifesto embodies the prioritisation of individuals and interactions on the left over processes and tools on the right, working software over comprehensive documentation, customer collaboration over contract negotiation, and responding to change over following a plan [3]. Boehm [15] tells us to "get ready for agile methods, with care." Steven Rakitin [16] voiced his concern that "in his experience, the items on the right are essential," and by following those on the left, the Agile Manifesto "served only as easy excuses not to do the difficult work" and with "hacker interpretations" there would be no understanding of, or adherence to, generally accepted and essential engineering disciplines. According to Rakitin, the essence of following the items on the left means, "Great! Now I have a reason to avoid planning and to just code up whatever comes next." Schach believes that Agile will be unable to handle complex software projects – "Anyone can successfully hammer together a few planks to build a doghouse, but it would be foolhardy to build a three-bedroom home without detailed plans." [17] (p. 61).

The concepts of Big Design Up Front (BDUF), Enough Design (EDUF) and No Design (NDUF) can further assist with an understanding of Agile. It is possible to map BDUF, EDUF and NDUF in a right-to-left continuum against the four core values of the Agile Manifesto. While Traditional Project Management on the right utilises detailed documentation and design up front, Agile, with SCRUM and Extreme programming (XP) to the left of the continuum, uses limited to no detail. The Dynamic Systems Development Method (DSDM) towards the middle of the continuum, being a compromise between the two, utilises what could be considered by some as the nebulous concept of just enough detail before moving a project into execution. A challenge with the traditional approach is that projects can be delayed while creating the big design – the territory is often unclear until the project is under way. The challenge with Agile is that if the required detail for a complex project is not available, this can be a serious risk. In addition, a project budget commitment is often established up front without fully understanding the detail, and, once pre-allocated, a rigid managerial mindset may not be flexible if additional budget is required. In a resource-scarce economy, an additional budget to sustain project development may simply not be available [18] (p. 11).

Wysocki's [19] (p. 59) project type selection matrix in Fig. 1 advocates that 'up front selection' of the best method for the type of project being run is essential for project success:

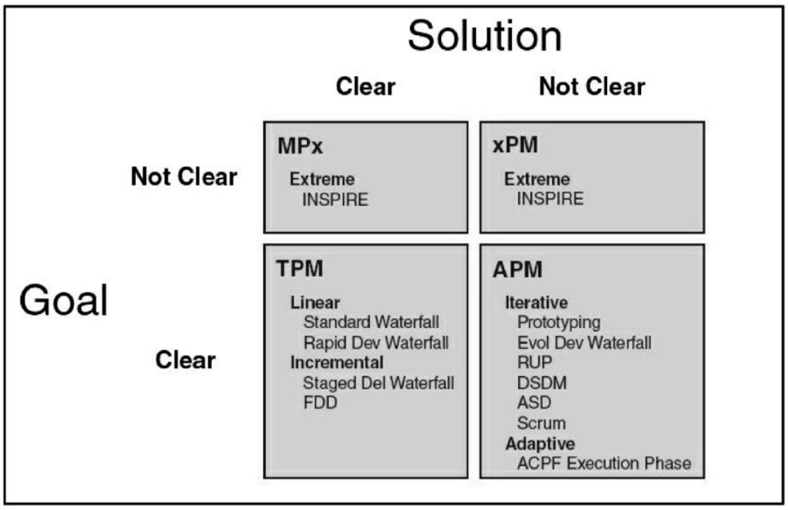


Fig. 1. The project type selection matrix [19]

While project methods constantly improve, a unique approach could be a combination of methods. For example, PRINCE 2 Agile method could be followed as a recipe under Board Supervision (Board Executive as Head Chef), while PMBOK could be followed to select the best project processes as 'ingredients to be added' [20]. The PMIS, implementing stage gates, can assist to reinforce project quality throughout a project lifecycle. If agility is required without the planning overhead, task slippage on the PMIS Schedule (baselined in the PMBOK Execution Process Group in Fig. 5, below), will quickly show whether agile activity is delivering value for the business and its stakeholders, or not.

4 Capability Maturity Levels as a Vertical Dimension

The Capability Maturity Model integrated (CMMi) [21] (p. 52) depicted in Fig. 2, below, is a recognised process improvement model from the Software Engineering Institute at Carnegie Mellon University in the U.S. As it is difficult to achieve Capability Maturity (CM) Level 2, companies need to address and conquer process improvement challenges to do this. Hirsch [8], discussing the PMI (Project Management Institute's) Pulse of the profession found a positive correlation between higher maturity levels, and on time and on budget delivery of projects. Naturally, the ability to run projects that deliver value on time and on budget can have significant benefits for resource-strained industries in the South African economy.

Capability Maturity Model – Integrated

Level	Focus	Process Areas	Result
5 Optimizing	*Continuous process improvement*	Organizational Innovation & Deployment Causal Analysis and Resolution	Productivity & Quality
4 Quantitatively Managed	*Quantitative management*	Organizational Process Performance Quantitative Project Management	
3 Defined	*Process standardization*	Requirements Development Technical Solution Product Integration Verification Validation Organizational Process Focus Organizational Process Definition Organizational Training Integrated Project Management Risk Management Decision Analysis and Resolution	
2 Managed	*Basic project management*	Requirements Management Project Planning Project Monitoring & Control Supplier Agreement Management Measurement and Analysis Process & Product Quality Assurance Configuration Management	
1 Initial	*Competent people and heroics*		Risk & Waste

Fig. 2. Capability Maturity Model integrated (adapted from [21])

A starting point for CM Level 2 is the Project Planning (PP) and Project Monitoring and Control (PMC) processes. PP and PMC processes also reside on either side of the PMBOK Execute Process group, to strengthen it. The crucial importance of PMBOK planning is emphasised by the fact that just under half the total of 49 PMBOK processes fall within the planning process group. Consequently, it can be argued that with limited or no planning, and a lack of good monitoring and control processes in place, projects are operating at CM Level 1 (initial phase) and would exhibit high levels of risk, rework and waste.

Deming's 14 Observations for Management, Red Bead Experiment, and Plan, Do, Check, Act (PDCA) cycle, among other contributions, clearly showed that without planning, and monitoring and control in place to guide and improve execution, quality delivery via streamlined production processes will remain the elusive silver bullet [22], hence ICT4D, with respect to SPM value delivery, if undertaken heroically at CM Level 1, would be ineffective and would remain a pipe dream.

In support of the above, the Dunning and Kruger effect [23] (p. 1) states that "people tend to hold overly favourable views of their abilities in many social and intellectual domains." The Dunning-Kruger effect has implications for the project management profession, as illustrated in Fig. 3, below. Project process stability at CM Level 2, for both traditional and agile project management, can be viewed as the start of a move up the slope of enlightenment, towards the goal sustainability of ICT projects.

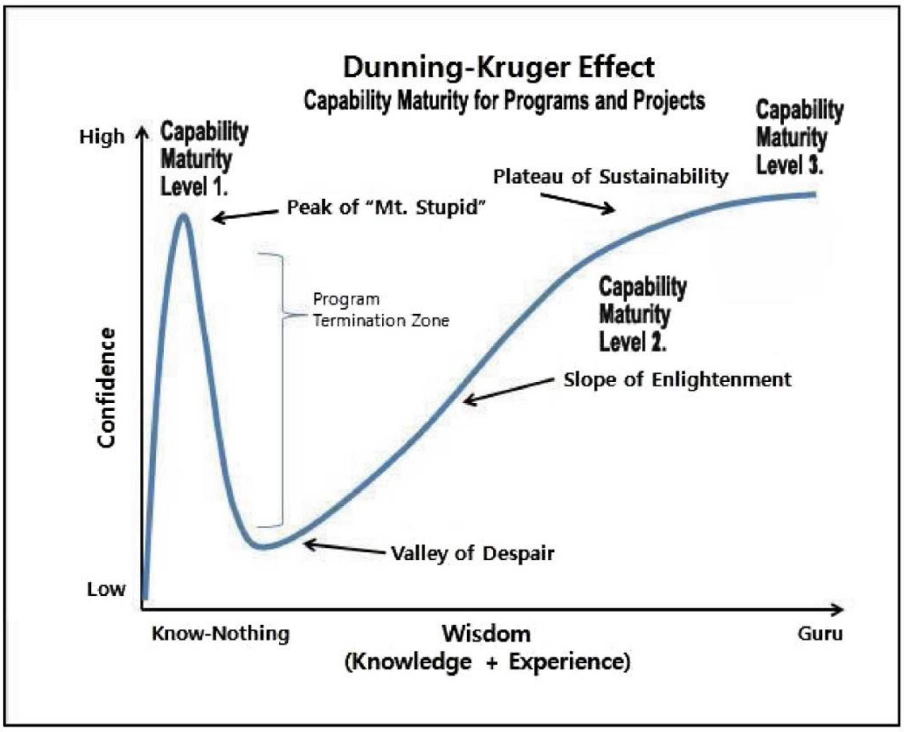


Fig. 3. Dunning-Kruger effect on projects and programs (adapted from [24])

The Dunning-Kruger effect, when applied to complex projects, implies a lack of experience and unrealistic expectations, and may set a project up for failure. CM Level 2, which is essentially a base project management process set, ought to be the starting point to facilitate project success and stakeholder satisfaction. Consequently, a complex, Agile-driven project at CM Level 1 runs the risk of being severely challenged, thereby compromising good ICT4D intentions.

5 The People-Process-Technology Triad Dimension

According to Deming, "the supposition is prevalent the world over that there would be no problems in production or service if only our production workers would do their jobs in the way that they were taught. Pleasant dreams. The workers are handicapped by the system, and the system belongs to the management." [4] (p. 121). Also, "it is not enough that management commit themselves to quality and productivity, they must know what it is they must do. Such a responsibility cannot be delegated" [4] (p. 8); in addition, "people are already doing their best. The problem is with the system. Only management can change the system." [4] (p. 8). Chrissis *et al*., on the Capability Maturity Model integrated (CMMi), further elucidates Deming's wisdom: "while process is often described as a node of the people-process-technology triad, it can also be considered the 'glue' that ties the triad together" [21] (p. 4). Essentially, while everyone realises the importance of having a quality and motivated workforce, even the most talented staff might not perform at their best if the process (system) is not clearly understood, or is operating non-optimally (cf. slow or inoperative systems, compromising the value ICT brings to a developing economy).

In addition, software design and development, and deployment of technology, are complex undertakings and, according to Brooks, "can be regarded as being one of the most difficult undertakings" [26, 27] (p. 5). In Brooks' Anniversary Edition [27], some 20 years after the original publication of the *Essays on Software Engineering*, he again emphasised project complexity and why software development projects to create 'programming systems products' can, if not carefully managed, often become mired in ever-increasing difficulties he referred to as the "tar pit" of failed software projects. Brooks' description is of ancient lumbering beasts of old ensnared, yet "the fiercer the struggle, the more entangling the tar, and no beast is so strong or so skillful but that he ultimately sinks. Large-system programming has, over the past decade, been such a tar pit, and many great and powerful beasts have thrashed violently in it" [27] (p. 4).

In terms of answering the research questions, when projects are complex, especially if software engineering projects are involved, many CM model processes can be used to make the arduous journey safer. The correct balance of people-process-technology within the triad should always be an important consideration for business success.

Conclusion: The preceding discussions provide an answer to our 1st research question, **RQ1,** in the sense that a CM Level 2 is much needed, and may not necessarily be provided by following pure Agile, if naively undertaken at Level 1. Note that some groundwork towards ICT4D addressed in **RQ2** has also been established.

6 The Project Management Information System (PMIS)

Bočková [28] (p. 712) explains that game theory can be used to position each project as a game to “maximize gains and minimize losses,” focusing the team towards the achievement of success. Team visibility via a PMIS of the Developed Schedule (process #6.5 baselined and moved into Execution) implies that all share a jointly-agreed playing field where teams and individuals can excel. The goal of utilising a PMIS ought to be, as Braglia and Frosolini [29] (p. 1) put it “to boost efficiency by making the development cycle more visible.” As mentioned before, a task with a slip line would be cause for concern; however, if the Deming wisdom is understood, then this project task should also be correctly allocated, fully resourced and agreed by the task owner that it is achievable at worker level, before a commitment baseline is installed on the Developed Schedule.

The PMI [8] (p. 7) found that the number of Project Management Offices (PMOs) was on the increase. A PMO with a PMIS can automate various components of the project management process, including automating the schedule, work authorisation, resource management, risks and issues management, and team collaboration. In addition, the PMIS (built on a database or, more generally, a data warehouse) can capture and disseminate information and report on key performance indicators (KPIs), baselined against actuals and many other metrics. Meredith and Mantel [30] (p. 462) discuss how a PMIS is beneficial where project complexity is an issue. They recommend choosing an application that offers “functions of friendliness” (User Experience – UX) which includes calendars, budgets, graphics, charts, reports, etc. Ultimately, the PMIS can greatly assist practitioners to detect latent issues before they occur, achieve project milestones, and collaborate more easily and to a greater extent. They conclude that “PMIS makes a significant contribution to project success and should continue to be the object of project management research” [31] (p. 219). The use of qualitative cues, like graphics and charts in a PMIS, instead of natural language text, may work well in South Africa’s multilingual and culturally rich society.

The key benefit of a PMIS is in its ability to facilitate resource allocation. The fact that tasks cannot be created at the same time on different project schedules in a PMIS, means that resource allocation challenges are remedied. Engwall suggests that the resource allocation syndrome might be an effect of “opportunistic project management behaviour within the organizations,” and is a consequence of flawed organisational procedures, rather than poor project management practices [32] (p. 408). Addressing the root cause, Engwall advocates, is for the executive to solve. Ultimately, the resource allocation syndrome of multi-project management is not an issue in itself; “rather it is an expression of many other, more profound, organizational problems of the multi-project setting” [32] (p. 408). Management insistence on heroics, and producing faster with less, will simply exacerbate the problem if they do not grasp how to run projects at CM Level 2 and above.

According to Tromp [33] (p. 213), “the main problem in a multi-project environment is the lack of dedicated teams on one project.” Employees allocated over many projects and demanded simultaneously on these projects have negative consequences for the projects. The Theory of Constraints explains how to isolate the most important limiting factor (or constraint) that stands in the way of achieving a goal. The idea, after identifying the constraint, is to improve it until it is no longer the limiting factor. According to Goldratt [34], challenges common to all projects are bad multitasking; students' syndrome; Parkinson's Law; and dependencies between steps. These can result in budget and time overruns; and compromised content, compromising the ICT value proposition.

A PMIS can also collect and distribute information and report on key performance indicators (KPIs), utilising the baseline plan against actuals. The PMIS also offers summations of complex project status via dashboards using many different visual aids, e.g. Gantt charts, project progress graphs, and task status traffic lights. As before, the SA multilingual society will benefit from these visual cues (Fig. 4).

Fig. 4. The PMIS information stack (synthesized by researchers)

A *process*-based approach to facilitate ICT4D is well established [35]. Apart from the ability to better manage the core CM Level 2 *processes* of Project Planning (PP) and Project Monitoring and Control (PMC), the CM Level 2 process Measurement Analysis (MA) is also derived from the PMIS. The correct installation and use of the PMIS is guided by the PMIS CM Framework in Fig. 6, below, meaning that the CM Level 2 *process* Product and Process Quality Assurance (PPQA) is also installed and streamlined over time.

Documentation saved in the PMIS addresses many of the Project Quality dimensions for good governance. Requirements Management (REQM) and Configuration Management (CM) are pinned down with quality gates on the PMIS, implying that the team can proceed when they are Ready To Do It (cf. acquired sufficient resources, competent skill set, requirements, etc.) and not because of pressure to be heroic and Just Do It. Essentially, this is the difference between going off "half-cocked" on a Developing Schedule in the Planning Process Group, and proceeding correctly off a Developed and Baselined Schedule in the Execution Process Group – as is illustrated in Fig. 5. Risk and Issue Management (RSKM), a CM Level 3 process, is seamlessly accessible in the PMIS to team members. Project challenges are avoided or quickly solved, or escalated to the executive. Ultimately, process improvement projects can be used to install all the Capability Maturity processes up to Level 5. Achieving such maturity would significantly benefit a developing economy often constrained by a lack of resources and a developing workforce. Concerning the people-process-technology triad dimension in Section 5 above, the PMIS (a technology), as a means of installing project management and other important processes (the 'glue'), can facilitate the process whereby project teams (empowered people) can navigate through the tar pit. This should ensure that resources are used optimally, and much needed user-satisfying technology is successfully created.

Finally, working harder is not always the best solution if effort expended is at CM Level 1. Low capability and maturity, and a lack of command and control in project management, must account for many of the problems identified by the Standish Chaos reports. Plausibly, if you cannot readily measure something, you cannot improve it. Arguably, the best way to improve anything must be to start from a known baseline.

Conclusion: The preceding discussions answer our 2nd research question, **RQ2**, namely, how the use of a PMIS may facilitate CM Level 2 development, thereby enhancing the ICT value proposition for a developing economy (**RQ2**).

7 PMIS CM Improvement Framework

The PMIS CM Improvement framework could be a starting point to ensure that the need for the PMIS is more fully understood. It can also be used to guide the installation of PMIS CM Improvement dimensions in a phased and systematic manner. The right-facing arrowheads in Fig. 5, below, have been added by the researchers to identify the PMIS CM Improvement framework scope which has been designed to operate from process #6.5 (the end of project planning) up to process #4.3 (the beginning, after schedule baseline, of project execution). The framework aims to lift CM dimensions from Level 1 to Level 2 compliancy, eliciting the importance of processes to facilitate project success, and contribute to ICTs for development [35].

The PMIS CM Improvement Framework, below, facilitates the creation of a Developed Schedule (resourced and ready for baseline) as an output of the Develop Schedule process #6.5 on the PMIS so that "Now We Are Ready To Do It." This empowers CM Level 2 compliancy at the beginning of the Project Execution Process

Knowledge Areas	Initiating Process Group	Planning Process Group	Executing Process Group	Monitoring and Controlling Process Group	Closing Process Group
	Project Management Process Groups				
4. Project Integration Management	4.1 Develop Project Charter	4.2 Develop Project Management Plan	4.3 Direct and Manage Project Work 4.4 Manage Project Knowledge	4.5 Monitor and Control Project Work 4.6 Perform Integrated Change Control	4.7 Close Project or Phase
5. Project Scope Management		5.1 Plan Scope Management 5.2 Collect Requirements 5.3 Define Scope 5.4 Create WBS		5.5 Validate Scope 5.6 Control Scope	
6. Project Schedule Management		6.1 Plan Schedule Management 6.2 Define Activities 6.3 Sequence Activities 6.4 Estimate Activity Durations 6.5 Develop Schedule		6.6 Control Schedule	
7. Project Cost Management		7.1 Plan Cost Management 7.2 Estimate Costs 7.3 Determine Budget		7.4 Control Costs	
8. Project Quality Management		8.1 Plan Quality Management	8.2 Manage Quality	8.3 Control Quality	
9. Project Resource Management		9.1 Plan Resource Management 9.2 Estimate Activity Resources	9.3 Acquire Resources 9.4 Develop Team 9.5 Manage Team	9.6 Control Resources	
10. Project Communications Management		10.1 Plan Communications Management	10.2 Manage Communications	10.3 Monitor Communications	
11. Project Risk Management		11.1 Plan Risk Management 11.2 Identify Risks 11.3 Perform Qualitative Risk Analysis 11.4 Perform Quantitative Risk Analysis 11.5 Plan Risk Responses	11.6 Implement Risk Responses	11.7 Monitor Risks	
12. Project Procurement Management		12.1 Plan Procurement Management	12.2 Conduct Procurements	12.3 Control Procurements	
13. Project Stakeholder Management	13.1 Identify Stakeholders	13.2 Plan Stakeholder Engagement	13.3 Manage Stakeholder Engagement	13.4 Monitor Stakeholder Engagement	

Fig. 5. PMBOK 6 Process Group & Knowledge Area matrix (adapted from [1] (p. 25))

Group at process #4.3. If an Agile method is used, tasks are built collaboratively (on the fly) by resources, adding their contribution to the schedule on the PMIS, while the project moves into process #4.3 as a next Agile iteration or time box. The PMIS CM Improvement Framework is presented in Fig. 6 below:

12.	The PMIS Resource Pool is a focus of organizational attention towards people and their skills improvement for excellence.	No	Yes (1)
11.	Risk and Issue management on the PMIS is available to any member of the PMIS resource pool with project teams focused on keeping the projects (schedules) healthy and safe.	No	Yes (1)
10.	Earned Value Management Analysis (and other) project status graphics are available on the PMIS.	No	Yes (1)
9.	EPMO / PMIS works off a regular project reporting (actual vs plan / budget) cycle.	No	Yes (1)
8.	Variance tracking of actuals on the PMIS at Task level is available against the baselined project plan / schedule.	No	Yes (1)
7.	PMIS schedule can be represented as a Gantt chart with Task slippage indicator capability.	No	Yes (1)
6.	PMIS offers "virtual Team Collaboration" capability [from anywhere connected to the internet].	No	Yes (1)
5.	Stage / Quality Gates via PMIS Process Group (and other) selectors is available.	No	Yes (1)
4.	The PMIS is a secure repository for all project artifacts (documents etc.) created during a project lifecycle.	No	Yes (1)
3.	EPMO allocates Project Managers to Projects (schedules) on the PMIS.	No	Yes (1)
2.	PMIS has a Resource Management module to facilitate the allocation of skilled resources from the resource pool to attend to Tasks on Project Schedules.	No	Yes (1)
1.	Enterprise Project Management Office (EPMO), driven off a Project Management Information System (PMIS), is in place.	No	Yes (1 Point)

Improvement Dimensions: Using the PMIS for CM Level 2 stability (and above)

Fig. 6. PMIS CM Improvement Framework developed by the researchers

Notes on Fig. 6:

1. The EPMO with PMIS operates with a similar imperative as an Accounting Department underpinned by a financial accounting system.
2. The Resource Allocation Syndrome is prohibited on the PMIS [32].
3. The EPMO has central control of all projects (programs) within the PMIS; outsourcing Project Management control.
4. Project documents/System Configuration Management documents, etc. are version controlled and accessible on the PMIS by the EPMO/sponsors, managers and teams.
5. Ever mindful of agility; the Process Group selector ensures projects are only moved to a next Process Group when appropriate – i.e. from the perspective of good governance, quality tested, etc.
6. Virtual collaboration is the ability to collaborate with all team members on schedules on the PMIS from anywhere via the Internet – i.e. a secure environment must exist on the PMIS (network/Internet/virtual private network, etc.).
7. Task slip lines typically can be switched on or off. This visual representation facilitates using the project schedule as a playing field from a game theory perspective.
8. Top-level summaries should be available by program (groups of projects on the PMIS) or by a single project and with drill-down capability to task level.

9. Reports typically to Project Boards/Sponsors (and identified stakeholders). As projects become late "a day at a time," reports follow a weekly, week-by-week publication cycle. Reports focus on the Plan, Do, Check, Act cycle.
10. From a PMBOK perspective, EVM includes all project formulas, and especially the "To Complete Performance Index" as "one number" for project health throughout its lifecycle. EVM, built of tasks, is an essential PMIS reporting product facilitating the goal of this paper to use the project schedule as a playing field.
11. Issues are Risks happening. Serious Issues, which a project team are unable to solve, are escalated up to the Project Board Executive/Sponsor who are also members of the PMIS resource pool. Per Deming, "such responsibility cannot be delegated." [4].
12. People excellence is dealt with in People CMM [36]. Excellence can also be measured by CM Focus Area (Development [21], Services, Acquisitions) focusing on process improvement by Category, Process Area and Maturity Level. In addition, the field of Human Resources has much to contribute.

Conclusion: The preceding discussions provide an answer to our 3rd and last research question, namely, how the PMBOK Process Group and Knowledge Area matrix may be enhanced to cater for sustainable CM improvements (**RQ3**). Note the link between the PMBOK *processes*, enabled using a PMIS, and *process*-driven ICT4D. The use of a PMIS as an emplacement and firm base will be a strategic value-generation driver by "focusing effort" in a resource-constrained developing economy.

8 Conclusions and Future Work

Due to project management complexities typically found in software engineering (and other) project types, the main thrust of this research is that without a PMIS, the project team's ability to adequately monitor and control project management processes for CM success at Level 2 (and above) at the task level on a project schedule, would be challenged. The Agile (Scrum) whiteboard with "stickies" will be equally constrained. Consequently, project performance, despite good intentions from a talented team, would, without the PMIS, naturally degrade to Heroic effort at CM Level 1.

The value proposition promised by innovative ICTs for a developing economy – the ICT4D – is pivotal. The way this may be achieved is through process-driven ICT4D, better managed on Developed Schedules (playing fields) on the PMIS. To this end, the use of a PMIS to enhance process maturity in software project management is precisely to promote process-driven ICT4D. Improving process excellence by CM levels promises to optimise scarce resources and improve on a developing skill set such as the South African workforce.

Agile, often viewed as a silver bullet solution [22], if undertaken at CM Level 1 on complex software projects, would often not be able to produce satisfactory results. Therefore, for the attainment of CM Level 2 (Managed), a PMIS is not simply a recommended tool of the 'Direct and Manage Project Work' process #4.3 in Fig. 5, as is stated by the Project Management Institute. Rather, as argued in this paper, it is essential to facilitate project management success for earned value.

Ultimately, within a gradient scale of improvement, a direct relationship appears to exist between the installation (or not) of a PMIS, the way the PMIS is installed within an organisation, and an organisation's ability to achieve stability at CM Level 2 and above.

Future work is this area will be directed at a validation phase of the CM Improvement Framework developed in Fig. 6. This will be undertaken by targeted surveys in selected industries. Such instrument has already been developed, and will be administered in due course, having obtained the necessary ethical clearance and company permission.

Further value to be added by our PMIS in the ICT4D arena needs to be investigated.

References

1. Project Management Institute: A Guide to the Management Body of Knowledge. Project Management Institute, Inc., Atlanta (2017)
2. The Standish Group: The Standish group: the chaos report. Project Smart 16 (2014)
3. Sutherland, J., et al.: Manifesto for Agile Software Development. http://agilemanifesto.org/
4. Deming, W.E.: Out of the crisis. Massachusetts Institute of Technology, Center for Advanced Engineering Study, Cambridge (1986)
5. Eveleens, L.J., Verhoef, C.: The rise and fall of the chaos report figures. IEEE Softw. **27**(1), 30–36(2010). https://doi.org/10.1109/MS.2009.154, ISSN 0740-7459
6. Schach, S.R.: Object-Oriented and Classical Software Engineering. McGraw-Hill, New York (2011)
7. Guess, V.C.: CMII for Business Process Infrastructure. CMII Research Instutite (2006)
8. Hirsch, B.J.a.: PMI's Pulse of the Profession. 2012 PMI Pulse Prof., 1–9 March 2012
9. Vargas, R.: Elaboration of the PMBOK 6 process flow. https://www.youtube.com/watch?v=GC7pN8Mjot8
10. Richards, K., Cooper, L.: PRINCE2 Agile. The Stationery Office, Norwich (2015)
11. Office of Government Commerce: Managing Successful Projects with PRINCE2. TSO, London (2009)
12. DSDM Consortium: Agile PM. Agile Project Management Handbook Version 2. DSDM Consortium (2014)
13. Setpathy, T.: Scrum Body of Knowledge. SCRUMstudy (2016)
14. Extreme Programming Web Site. http://www.extremeprogramming.org/
15. Boehm, B.: Get ready for agile methods, with care. Comput. (Long. Beach. Calif.) **35**, 64–69 (2002)
16. Rakitin, S.: Manifesto elicits cynicism. IEEE Comput. **34**, 4–7 (2001)
17. Schach, S.: Object Orientated and Classical Software Engineering, 8th edn. (2007)
18. Clinning, G., Marnewick, C.: Incorporating sustainability into IT project management. S. Afr. Comput. J. **29**, 1–26 (2017)
19. Wysocki, R.K.: Effective Complext Project Management (2014)
20. LogicalModel.net: Parallels in Delivering Organisational Change as a Bowl Of Soup. http://www.logicalmodel.net/pm_ngt/soop-prince2-and-pmbok-as-a-bowl-of-soup/
21. Chrissis, M.B., Konrad, M., Shrum, S.: CMMI for Development. Addison-Wesley, Upper Saddle River (2011)

22. Brooks, F.P.J.: No silver bullet-essence and accidents of software engineering. In: Proceedings of the IFIP Tenth World Computing Conference, pp. 1069–1076 (1986)
23. Kruger, J., Dunning, D.: Unskilled and unaware of it: how difficulties in recognizing one's own incompetence lead to inflated self-assessments. J. Pers. Soc. Psychol. **77**, 1121–1134 (1999)
24. Dudley, M.: Ladies and gentlemen, here comes the Dunning-Kruger Effect. https://medium.com/@mike.dudkey78/ladies-and-gentlemen-here-comes-the-dunning-kruger-effect-f31b09d52573
25. Scaled Agile: SAFe ® 4.5 Introduction Overview of the Scaled Agile Framework ® for Lean Enterprises. A Scaled Agil. Inc. White Pap. (2017)
26. Brooks, F.P.: No silver bullet: essence and accidences of software engineering. Comput. (Long. Beach. Calif.) **20**, 10–19 (1987)
27. Brooks, F.P.: The Mythical Man - Month. Essays on Software Engineering, Anniversary Edition (1995)
28. Bočková, K.H., Sláviková, G., Gabrhel, J.: Game theory as a tool of project management. Procedia Soc. Behav. Sci. **213**, 709–715 (2015)
29. Braglia, M., Frosolini, M.: An integrated approach to implement Project Management Information Systems within the extended enterprise. Int. J. Proj. Manag. **32**, 18–29 (2014)
30. Meredith, J.R., Mantel, S.J.J.: Project Management: A Managerial Approach (2009)
31. Raymond, L., Bergeron, F.: Project management information systems: an empirical study of their impact on project managers and project success. Int. J. Proj. Manag. **26**, 213–220 (2008)
32. Engwall, M., Jerbrant, A.: The resource allocation syndrome: The prime challenge of multi-project management? Int. J. Proj. Manag. JPMA **21**, 403–409 (2003)
33. Tromp, J.W., Homan, T.: How unplanned changes emerge while implementing a Project Management Information System (PMIS) in a complex multi project R&D environment. Procedia Soc. Behav. Sci. **194**, 211–220 (2015)
34. Goldratt, E.M.: Critical Chain. North River Press, Great Barrington (1997)
35. Walton, M., Heeks, R.: Development Informatics Improve ICT4D Project Success? Development Informatics Group (2011)
36. People Capability Maturity Model (P-CMM) (2009)

Annexure S: PMBOK ® 6 Process Matrix with Improvement Gap

The PMBOK Matrix on the following page has been split down the middle, and the PMIS CM Improvement Framework “gap” has been included in the middle of the matrix. This has been done to explain where the PMIS CM Improvement steps occur and how they bridge CM improvements up to Level 2 between process #6.5 and process #4.3.

While all processes in the matrix are always relevant, the gap is primarily where the PMIS emplacement is situated and where it will be applied. As most of this work focuses upon Table S.1, no further discussion on this will be included here.

Table S.1: PMBOK Matrix PMIS Improvement Gap

Knowledge Areas	Project Management Process Groups					
	Initiating Process Group	Planning Process Group	PMIS CM IMPROVEMENT FRAMEWORK	Executing Process Group	Monitoring and Controlling Process Group	Closing Process Group
4. Project Integration Management	4.1 Develop Project Charter	4.2 Develop Project Management Plan	12. 11. 10.	4.3 Direct and Manage Project Work 4.4 Manage Project Knowledge	4.5 Monitor and Control Project Work 4.6 Perform Integrated Change Control	4.7 Close Project or Phase
5. Project Scope Management		5.1 Plan Scope Management 5.2 Collect Requirements 5.3 Define Scope 5.4 Create WBS	9. 8. 7.		5.5 Validate Scope 5.6 Control Scope	
6. Project Schedule Management		6.1 Plan Schedule Management 6.2 Define Activities 6.3 Sequence Activities 6.4 Estimate Activity Durations 6.5 Develop Schedule	6. 5. 4. 3. 2. 1.		6.6 Control Schedule	
7. Project Cost Management		7.1 Plan Cost Management 7.2 Estimate Costs 7.3 Determine Budget			7.4 Control Costs	
8. Project Quality Management		8.1 Plan Quality Management		8.2 Manage Quality	8.3 Control Quality	
9. Project Resource Management		9.1 Plan Resource Management 9.2 Estimate Activity Resources		9.3 Acquire Resources 9.4 Develop Team 9.5 Manage Team	9.6 Control Resources	
10. Project Communications Management		10.1 Plan Communications Management		10.2 Manage Communications	10.3 Monitor Communications	
11. Project Risk Management		11.1 Plan Risk Management 11.2 Identify Risks 11.3 Perform Qualitative Risk Analysis 11.4 Perform Quantitative Risk Analysis 11.5 Plan Risk Responses	THE 12 STEP IMPROVEMENT GAP IN THE PMBOK 6 PROCESS GROUP & KNOWLEDGE AREA	11.6 Implement Risk Responses	11.7 Monitor Risks	
12. Project Procurement Management		12.1 Plan Procurement Management	MATRIX THAT IS IDENTIFIED FOR RESEARCH	12.2 Conduct Procurements	12.3 Control Procurements	
13. Project Stakeholder Management	13.1 Identify Stakeholders	13.2 Plan Stakeholder Engagement		13.3 Manage Stakeholder Engagement	13.4 Monitor Stakeholder Engagement	

Synthesized by the researcher from Project Management Institute (2017:25 & 556)

Annexure T: Proposed PMIS CM Improvement Framework

The Proposed PMIS CM Improvement Framework below was applied to the case study found in Chapter 5**.**

Table T.1: Proposed PMIS CM Improvement Framework

PMIS CM Improvement Framework

12.	The PMIS Resource Pool is a focus of organizational attention towards skills improvement for excellence.	No	Yes (1)
11.	Risk and Issue management module on the PMIS is available to all to protect the projects (schedules).	No	Yes (1)
10.	Earned Value Management (and other) project status graphics are available on the PMIS.	No	Yes (1)
9.	PMIS projects are driven by the EPMO financial reporting cycle.	No	Yes (1)
8.	PMIS offers variance tracking of scheduled tasks as actuals against schedule baseline.	No	Yes (1)
7.	PMIS project schedules viewed as Gantt charts with task slippage indication.	No	Yes (1)
6.	PMIS facilitates "Virtual Team Collaboration" from anywhere connected to the Internet.	No	Yes (1)
5.	Stage Gate capability is available to Project Manager on the PMIS.	No	Yes (1)
4.	PMIS is secure repository for all project artifacts.	No	Yes (1)
3.	Project Managers authorised by EPMO to run PMIS Projects.	No	Yes (1)
2.	Project resourcing driven off PMIS Resource Module.	No	Yes (1)
1.	EPMO driven off PMIS Emplacement.	No	Yes (1)
➡	CM Level 1		
➡	PMIS for CM Level 2+ (12 Improvement Dimensions)		

How to use the PMIS CM Improvement Framework:

1. Enter from the bottom.
2. Award 1 point for each Yes if Improvement Dimension is in place and operating well.
3. As you proceed a No answer may not appear underneath a step you have answered with a Yes.

Source: Compiled by the researcher

Annexure U: Revised PMIS CM Improvement Framework

The Revised PMIS CM Improvement Framework in Table U.1 was slightly amended to include findings after it was applied in its original form in Annexure T to the case study found in Chapter 5.

Table U.1: Revised PMIS CM Improvement Framework

PMIS CM Improvement Framework

12.	b) The PMIS Resource Pool is a focus of organizational attention towards skills improvement for excellence. a) The Human Resources Head accepts key stakeholder role to ensure PMIS emplacement Resource Pool is focus of organizational attention towards skills improvement for excellence.	No	Yes (1/2) Yes (1/2)
11.	Risk and Issue management module on the PMIS is available to all to protect the projects (schedules).	No	Yes (1)
10.	b) Earned Value Management (and other) project status graphics are available on the PMIS. a) Finance Department Head accepts key stakeholder role to ensure PMIS emplacement reflects accurate financial position in respect of projects financials.	No	Yes (1)
9.	b) PMIS projects are driven by the EPMO financial reporting cycle. a) Finance Department Head accepts key stakeholder role to ensure PMIS emplacement reflects accurate financial position in respect of projects financials.	No	Yes (1/2) Yes (1/2)
8.	PMIS offers variance tracking of scheduled tasks as actuals against schedule baseline.	No	Yes (1)
7.	PMIS project schedules viewed as Gantt charts with task slippage indication.	No	Yes (1)
6.	PMIS facilitates "Virtual Team Collaboration" from anywhere connected to the Internet.	No	Yes (1)
5.	Stage Gate capability is available to Project Manager on the PMIS.	No	Yes (1)
4.	PMIS is secure repository for all project artifacts.	No	Yes (1)
3.	Project Managers authorised by EPMO to run PMIS Projects.	No	Yes (1)
2.	Project resourcing driven off PMIS Resource Module.	No	Yes (1)
1.	b) EPMO driven off PMIS Emplacement. a) EPMO Head accepts PMIS emplacement project key stakeholder role.	No	Yes (1/2) Yes (1/2)

➡ CM Level 1

➡ PMIS for CM Level 2+ (12 Improvement Dimensions)

How to use the PMIS CM Improvement Framework:

1. Enter from the bottom.
2. Award 1 point for each Yes if Improvement Dimension is in place and operating well.
3. As you proceed a No answer may not appear underneath a step you have answered with a Yes.

Source: Compiled by the researcher

Annexure V: Continuous Improvement / TQM

The website Masters of Project Management refer to the Plan Do Check Act Cycle as the four gears of continual service improvement.

While there is much that can be said on the subject of PDCA, Continuous Service Improvement, Total Quality Management, etc., Figure V.1 explains the concept extremely well.

Section 1.6 dealt with the concept of PDCA in some detail. In ITIL 3, Continual Service Improvement is an ITIL phase all of its own. The output of CSI is a CSI Register, and it is via this register that the ITIL Service Strategy (Service Catalogue) is planned and system improvement projects envisaged. The researcher has mentioned the similarity between the CSI register and the Scrum Backlog. And finally, CMMi Levels 4 and 5 are all about continual improvement. The PMIS emplacement, set up correctly and operating optimally, will facilitate the CM Level 2 processes Project Planning (PP), Project Monitoring and Control (PMC) and Measurement Analysis (MA), which are the essential processes needed to spiral up and out of CM Level 1 “initial” behaviour.

Figure V.1 below succinctly explains the concept of Continual Improvement and how Plan Do Check Act is used.

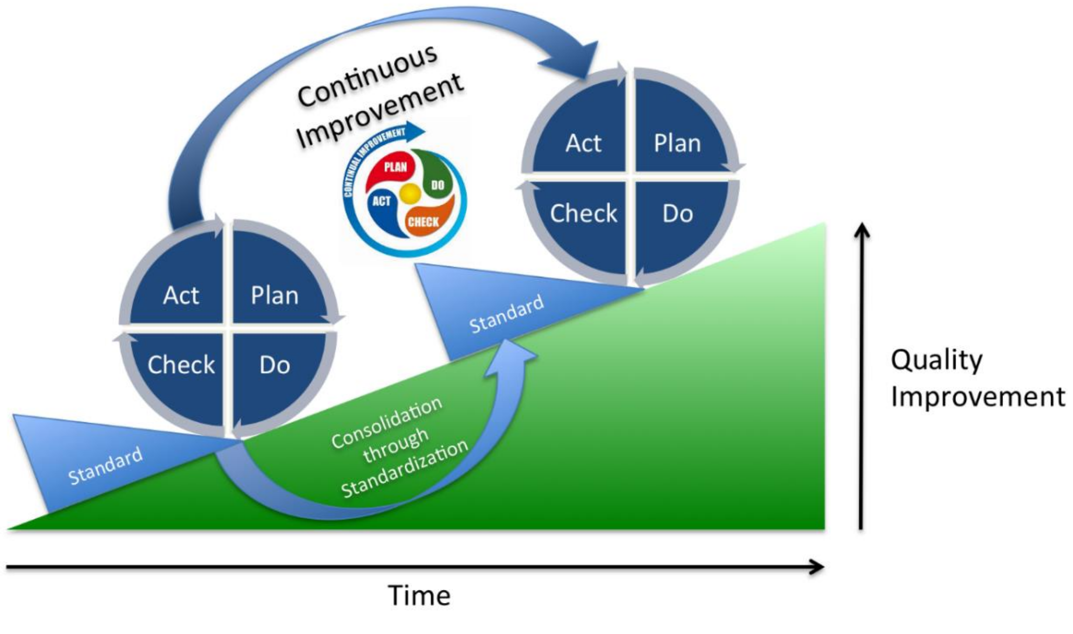


Figure V.1: PDCA & Quality Improvement

Source: PDCA Cycle: The 4 Gears of Continual Service Improvement

Annexure W: Certificate of Approval for Research

UNISA university of south africa

UNISA COLLEGE OF SCIENCE, ENGINEERING AND TECHNOLOGY'S (CSET) ETHICS REVIEW COMMITTEE

15 August 2020

Dear Mr MJ Corrigan

ERC Reference #: 2020/CSET/SOC/024
Name: Mark Jonathan Corrigan
Student #: 4151690
Staff #:

Decision: Ethics Approval from 15 August 2020 to 14 August 2023 (No humans involved)

Researcher: Mark Jonathan Corrigan
4151690@mylife.unisa.ac.za, 082-803-2304

Supervisors: Prof. JA van der Poll
Unisa SBL, vdpolja@unisa.ac.za, 011-652-0316
Mr ES Mtsweni
Unisa School of Computing, mtswees@@unisa.ac.za, 011-471-3019

Working title of research:

An Information Systems Project Maturity Framework for Level 2 Compliance

Qualification: MSc in Computing

Thank you for the application for research ethics clearance by the Unisa College of Science, Engineering and Technology's (CSET) Ethics Review Committee for the above mentioned research. Ethics approval is granted for 3 years.

*The **negligible risk application** was expedited by the College of Science, Engineering and Technology's (CSET) Ethics Review Committee on 15 August 2020 in compliance with the Unisa Policy on Research Ethics and the Standard Operating Procedure on Research Ethics Risk Assessment. The decision will be tabled at the next Committee meeting for ratification.*

The proposed research may now commence with the provisions that:

1. The researcher will ensure that the research project adheres to the relevant guidelines set out in the Unisa COVID-19 position statement on research ethics attached.

University of South Africa
Preller Street, Muckleneuk Ridge, City of Tshwane
PO Box 392 UNISA 0003 South Africa
Telephone: +27 12 429 3111 Facsimile: +27 12 429 4150
www.unisa.ac.za

2. The researcher(s) will ensure that the research project adheres to the values and principles expressed in the UNISA Policy on Research Ethics.
3. Any adverse circumstance arising in the undertaking of the research project that is relevant to the ethicality of the study should be communicated in writing to the College of Science, Engineering and Technology's (CSET) Ethics Review Committee.
4. The researcher(s) will conduct the study according to the methods and procedures set out in the approved application.
5. Any changes that can affect the study-related risks for the research participants, particularly in terms of assurances made with regards to the protection of participants' privacy and the confidentiality of the data, should be reported to the Committee in writing, accompanied by a progress report.
6. The researcher will ensure that the research project adheres to any applicable national legislation, professional codes of conduct, institutional guidelines and scientific standards relevant to the specific field of study. Adherence to the following South African legislation is important, if applicable: Protection of Personal Information Act, no 4 of 2013; Children's act no 38 of 2005 and the National Health Act, no 61 of 2003.
7. Only de-identified research data may be used for secondary research purposes in future on condition that the research objectives are similar to those of the original research. Secondary use of identifiable human research data require additional ethics clearance.
8. No field work activities may continue after the expiry date 14 August 2023. Submission of a completed research ethics progress report will constitute an application for renewal of Ethics Research Committee approval.

Note

The reference number 2020/CSET/SOC/024 should be clearly indicated on all forms of communication with the intended research participants, as well as with the Committee.

Yours sincerely,

Mr C Pilkington
Chair of School of Computing Ethics Review Subcommittee
College of Science, Engineering and Technology (CSET)
E-mail: pilkicl@unisa.ac.za
Tel: (011) 471-2130

University of South Africa
Preller Street, Muckleneuk Ridge, City of Tshwane
PO Box 392 UNISA 0003 South Africa
Telephone: +27 12 429 3111 Facsimile: +27 12 429 4150
www.unisa.ac.za

Prof. E Mnkandla
Director: School of Computing
College of Science Engineering and Technology (CSET)
E-mail: mnkane@unisa.ac.za
Tel: (011) 670 9104

Prof. B Mamba
Executive Dean
College of Science Engineering and Technology (CSET)
E-mail: mambabb@unisa.ac.za
Tel: (011) 670 9230

URERC 25.04.17 - Decision template (V2) - Approve

University of South Africa
Preller Street, Muckleneuk Ridge, City of Tshwane
PO Box 392 UNISA 0003 South Africa
Telephone: +27 12 429 3111 Facsimile: +27 12 429 4150
www.unisa.ac.za

Figure W.1: Certificate of Approval for Research

Source: University of South Africa

Annexure X: Certificate of Editing

Certificate of Editing

This is to certify that the dissertation

AN INFORMATION SYSTEMS PROJECT MATURITY FRAMEWORK FOR LEVEL 2 COMPLIANCE

by

MARK J CORRIGAN

has been proofread and edited for English language usage.

Date: 28 August 2021

LHugo

Lianne Hugo

Language Practitioner
B.A. (HMS)
PGCE

Email: liannehugo79@gmail.com

Figure X.1: Certificate of Editing

Source: Lianne Hugo

Annexure Y: Turnitin Report

The Turnitin report and notes is found below.

Highest percentage similarity:

The top item on the report refers to the Springer Journal Article mentioned in Annexure R. This previous research was submitted by the researcher and two of his Professors. This published paper, which amounts to 5% on the similarity report, forms the basis for most of this new research.

Other similarities less than one percentage:

On investigation it was found that many of the other similarities are in the report because they are standard text in various methodologies or bodies of knowledge or manuals on subjects being discussed. An example of this is the Agile Manifesto or a Project Methodology like the Project Management Body of Knowledge or a manual for COBIT 5. The principles in the Manifesto or Project Methodology or COBIT 5 must be reproduced as they are in order not to subvert the generally accepted definition and meaning intended. In all cases the source texts are fully referenced in the section REFERENCES found after Chapter 6 and proceeding Annexure A.

2/19/22, 8:10 PM Turnitin

Document Viewer

Turnitin Originality Report

Processed on: 19-Feb-2022 16:07 SAST
ID: 1766116196
Word Count: 62029
Submitted: 1

AN INFORMATION SYSTEMS PROJECT MATURITY FRAME... By Mark Corrigan

Similarity Index	20%

Similarity by Source	
Internet Sources:	18%
Publications:	13%
Student Papers:	10%

exclude quoted exclude bibliography exclude small matches mode: quickview (classic) report Change mode print download

5% match (Internet from 07-May-2020)
https://link.springer.com/chapter/10.1007%2F978-3-030-11235-6_19

1% match (student papers from 21-Jan-2018)
Submitted to Online Education Services on 2018-01-21

<1% match (Internet from 22-Jul-2019)
https://link.springer.com/content/pdf/10.1007/978-1-4842-3976-6.pdf

<1% match (Internet from 20-May-2019)
https://link.springer.com/content/pdf/10.1007%2F978-3-642-55035-5.pdf

<1% match (Internet from 15-Jan-2020)
https://link.springer.com/content/pdf/10.1007%2F978-3-319-98845-0.pdf

<1% match (Internet from 19-Jul-2019)
https://link.springer.com/content/pdf/10.1007%2F978-3-642-12442-6.pdf

<1% match (Internet from 30-May-2020)
https://link.springer.com/chapter/10.1007%2F978-3-030-11235-6_2

<1% match (Internet from 23-Oct-2020)
https://link.springer.com/chapter/10.1007%2F978-1-4842-3976-6_3

<1% match (Internet from 31-Jan-2022)
https://link.springer.com/content/pdf/10.1007/978-3-030-13431-0.pdf?code=05776 5ed-a674-42ab-a655-ce22b36c76a7&error=cookies_not_supported

<1% match (Internet from 27-Nov-2017)
http://www.pmway.co.za

<1% match (Internet from 21-Jul-2020)
https://www.pmway.co.za/scrumrca

<1% match (publications)
"Locally Relevant ICT Research", Springer Science and Business Media LLC, 2019

<1% match (Internet from 19-Apr-2021)
http://uir.unisa.ac.za

<1% match (Internet from 16-Oct-2021)
https://uir.unisa.ac.za/bitstream/handle/10500/28030/dissertation_makama_nr.pdf?isAllowed=y&sequence=1

<1% match (Internet from 02-Nov-2020)
http://uir.unisa.ac.za

<1% match (Internet from 12-Jul-2019)
http://uir.unisa.ac.za

https://www.turnitin.com/newreport_classic.asp?lang=en_us&oid=1766116196&ft=1&bypass_cv=1 1/59

Figure Y.1: First page of the Turnitin Report

Source: Turnitin.com