



Systems Engineering and Project Management (SEPM) Joint Working Group

Guide to SE and P3M Processes

Issue 1.0

Dec 2016

Summary

The aim of this document is to collate and compare process frameworks from within the PM and SE disciplines and form a foundation upon which key information on processes within SE and PM environments can be captured and expanded upon. The information in this document does not form an exhaustive review of the frameworks, processes and procedures in both disciplines, but aims to highlight the main areas of overlap, where they complement each other and where there are potential tensions. This document is intended to be read in conjunction with the accompanying *Guide to Life Cycles and Life Cycle Models* and *Integrated Life Cycle Representation*.

A list of various SE and PM process frameworks is presented to provide practitioners in each discipline with an increased awareness of the scope of the processes identified with each discipline. Using the International Standards Organisation (ISO) frameworks for Systems Engineering and Project Management it is possible to provide a definition of the areas of common practice, but only based on those ISO frameworks. Other process frameworks differ in their purpose and approach thus making similar mappings difficult. However by using an integrated life cycle representation (the “SEPM Life Cycle”), it is possible to demonstrate further areas of common and complementary processes.

Because of the different ways in which the SE processes can interact with PM processes, a taxonomy has been devised that describes both the ‘Nature of Interaction’ as well as the ‘Scale of Interaction’. The Scale of Interaction is a spectrum that runs from Gaps through Touchpoints and Overlaps, to Fusions. The Nature of Interaction can take the form of Boosts, Handshakes and Tensions.

The document then uses this taxonomy to identify potential key SE and PM process Fusion areas, which will be developed in later documents, as well as considering certain tensions (and their sources) between the two disciplines.

Authors:

Andrew Gray, Ken Richardson, Kate Rooke, Tony Thornburn

Copyright © 2016 A.Gray, K.Richardson, K.Rooke, T. Thornburn. Published and used by INCOSE UK Ltd., INCOSE and the Association for Project Management with permission.

The following document represents the thoughts and conclusions of the Systems Thinking SIG and not necessarily the views of the APM or INCOSE UK. It is intended to assist Project, Programme and Portfolio Management and Systems Engineering practitioners wishing to explore concepts and ideas around Systems Thinking in P3M and to stimulate discussion on the subject. Feedback on the contents of this paper should be sent to the Systems Thinking SIG (SystemsThinkingSIG@apm.org.uk). It therefore does not constitute any formal position (or liability arising) on the part of the International Council for Systems Engineering (INCOSE), INCOSE UK Ltd. or the Association for Project Management (APM), nor should any formal endorsement by these bodies be inferred.



Contents

1. Introduction	1
Purpose and Scope	1
Document Structure	2
Terminology	3
Acknowledgements and references	3
2. Definition of a process	4
Reference definitions	4
3. SE process frameworks	5
ISO15288:2015	5
INCOSE Systems Engineering Handbook	5
4. P3M process frameworks	6
APM Body of Knowledge	6
PMI® PMBOK®	7
PMI® Standard for Program Management	8
ISO Standards	8
Axelos Global Best Practice suite	9
5. A comparison of SE and P3M processes	12
The ISO comparison for systems engineering and project management	12
Touchpoints and overlaps in programme and portfolio activities	12
An integrated suite of process products	13
A taxonomy for the range of interactions	14
An example of gaps, touchpoints, overlaps and fusions	17
6. Examples of SE and P3M process fusion	20
Complementary processes or frameworks	20
Governance	21
Product based planning	21
A programme as a system of systems	21
Portfolio and programme architectural modelling	22
Verification and validation in benefits management	22
Agile requirements satisfaction: Clear thinking in the agile environment	22
Interdependency management	23
Soft systems methodologies in stakeholder management	23
Integrated supply chain management	23
Requirements definition in contracts	23
Transition definition and management	23
7. Areas of process friction or tensions	25
Change is not straightforward - tensions exist	25
Is it too much bureaucracy or robust governance?	25
When does defining a solution rigorously become gold plating?	26
Do you achieve technical maturity or meet the programme timescales?	26
Making and maintaining a justifiable business case	26
The four horsemen: Ignorance, perception, interpretation and misapplication	27
Styles and behaviours	27



Figures

Figure 1:	The Different Perspectives on Change	2
Figure 2:	ISO15288:2008 System life cycle processes	5
Figure 3:	An overview of the elements within the APM Body of Knowledge.....	6
Figure 4:	PMI® PMBOK® Guide process groups and knowledge areas.....	7
Figure 5:	PMI® Program Management processes	8
Figure 6:	ISO21500 process groups and subject areas	9
Figure 7:	PRINCE2® processes and themes	10
Figure 8:	MSP® themes and transformational flow processes	11
Figure 9:	MoP® definition and delivery practices	11
Figure 10:	A mapping between processes defined within ISO15288:2008 and ISO21500:2012	12
Figure 11:	The Integrated SEPM Vee-model.....	13
Figure 12:	Representation of the range of interactions between SE and P3M disciplines	15
Figure 13:	The steps involved in project planning – a project perspective	17
Figure 14:	A gap between SE and PM practitioners when undertaking project planning	17
Figure 15:	Touchpoint between SE and PM when undertaking project planning	18
Figure 16:	Overlaps between Se and PM when undertaking project planning	18
Figure 17:	SE and PM fusion when undertaking project planning.....	19
Figure 18:	Project/programme tensions field	25

Tables

Table 1:	Workstreams within the APM/INCOSE Joint Working Group	1
Table 2:	A set of SEPM process products through the programme/project life cycle	14
Table 3:	A taxonomy for the range of Interactions between SE and PM	16

1. Introduction

Purpose and Scope

This is a simple guide collating and comparing process frameworks in Systems Engineering (SE) and Portfolio, Programme and Project Management (P3M). The purpose of this document is to form a foundation upon which key messages and information can be captured related to Processes within the P3M and SE environments. This therefore means that information can be provided in support of the other Workstreams shown in Table 1, as well as providing the basis for the final Workstream 8 output.

Table 1: Workstreams within the APM/INCOSE Joint Working Group¹

What are the benefits?	
WS 1	Compelling value proposition
How to deliver the benefits?	
WS 8	Processes and lifecycles
WS 4	Roles and responsibilities
WS 6	Competency framework
WS 7	Education and training
How to promote the benefits?	
WS 2	Communication
WS 3	Guidance material
WS 5	Case studies

There are 3 main Objectives of Workstream 8 “Processes and Life Cycles”:²

1. To identify where SE and P3M models, approaches and ways of working overlap and are complementary, and identify the nature of the relationships between the two disciplines.
2. To develop (where appropriate) a set of unified processes and lifecycle models (or look to utilise existing unified models and processes).
3. To communicate, review and exploit these processes and lifecycle models amongst the PM and SE communities.

This document addresses Objectives 1 and 3 from the process viewpoint. Its purpose is not to give an exhaustive review of the frameworks, processes and procedures in both disciplines but to highlight the key

¹ APM/INCOSE JWG on SE/PM Integration, “Aims and Objectives”, Version 1

² APM/INCOSE JWG on SE/PM Integration, “Workstream 8 Project Brief”, Version 1, September 2013

areas of overlap, where they work synonymously, and potential tensions. It is important to remember that tailoring of frameworks, processes and tools to the project is essential for success. The document merely provides a gateway to detailed definitions of processes that are maintained within published frameworks – it does not seek to duplicate these definitions or to redefine them. The document does however provide an initial foundation to explore the synergies between the disciplines – it is the intention to extend and deepen this foundation in later releases of this (and related) documents.

This guide is not intended to compare and review individual life cycles and life cycle representations. This will be undertaken in an accompanying publication “**Guide to Life Cycles and Life Cycle Models**”. Developments in integrating life cycle representations identified in this document can be found in the third document in this series - “**Integrated Life Cycle Representation**”.

Systems Engineering and Project, Programme and Portfolio Management share common values in thinking about complex problems, delivering enduring change (or transformation), and bringing together disparate disciplines. In effect they present different perspectives on change, as illustrated in **Error! Reference source not found.** This document looks to explore these different perspectives in terms of the processes involved in each discipline, and how the two perspectives interact.

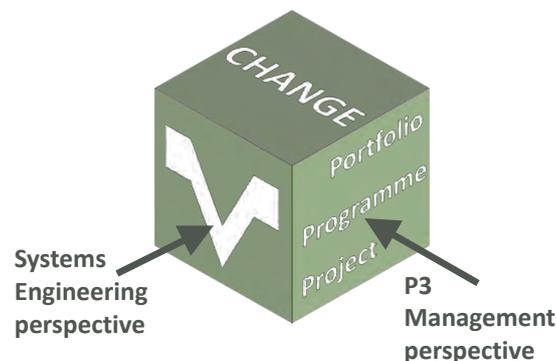


Figure 1: The Different Perspectives on Change

Document Structure

This document explores the relationships between processes in Portfolio, Programme and Project Management and Systems Engineering through 7 sections. In addition to this introduction, these sections cover:

<i>What is a process?</i>	Definition of a process. A comparison of various definitions of a process and whether any conflicts arise from differences in definitions.
<i>What SE Processes are there?</i>	SE process models. A review of Systems Engineering process frameworks. In practice this is dominated by those defined within ISO15288.
<i>What P3M Processes are there?</i>	P3M process models. A review of certain P3M process frameworks, covering those within the APM Body of Knowledge, the PMI Body of Knowledge, ISO standards and the Global Best Practice Suite from Axelos.
<i>Where do they overlap or interact?</i>	A comparison of SE and P3M processes. Identification of the overlap of processes included within SE and P3M frameworks, using the ISO standard for a back-to-back comparison.
<i>Where can they be combined positively?</i>	Examples of SE and P3M process fusions. Identification of key areas where SE and P3M processes can combine or interact to provide an enhanced performance or output.
<i>Where do they cause frictions?</i>	Areas of process friction or tension. A review of areas where tensions and frictions can arise between SE and P3M processes.



Terminology

The term P3 is used to denote Project, Programmes and Portfolios (as defined within the APM Body of Knowledge), and P3M denotes Project, Programme and Portfolio Management. Where the discussion only refers to Projects and Programmes, the term PPM will be used. SE is used to denote Systems Engineering approaches, models, processes and ways of thinking.

Acknowledgements and references

Unless otherwise referenced, information is taken from the INCOSE *System Engineering Handbook* (SEHBK) v4 (2015)³ or the APM *Body of Knowledge* (PMBOK) 6th Edition (2012)⁴. Key references will also include information from the ISO Standard ISO15288:2015 *Systems and software engineering – System life cycle processes* and the ISO standard (ISO21500:2012) *Guidance on Project Management*. Use has also been made of the information contained within the *Guide to the SE Body of Knowledge* (currently at v1.3)⁵.

Figure 2 is defined using information from ISO15288:2015 ©ISO

Figure 4 is defined using information from A Guide to the Project Management Body of Knowledge 5th Ed., Pennsylvania: PMI®

Figure 5 is defined using information from A Standard for Program Management 3rd Ed., Pennsylvania: PMI®

Figure 7 is defined using information from Managing Successful Projects with PRINCE2, 2009 Ed., London:TSO.

Figure 8 is defined using information from Managing Successful Programmes, 2011 Ed., London:TSO.

Figure 9 is defined using information from Management of Portfolios, 2011 Ed., London:TSO.

PRINCE2®, P3M3®, MoP® and MSP® are Registered Trade Marks of AXELOS Limited

PMI® and PMBOK® are Registered Trade Marks of Project Management Institute Inc.

³ INCOSE-TP-2003-002-04-2015, available at INCOSE. 2015. *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*, version 4.0. Hoboken, NJ, USA: John Wiley and Sons, Inc, ISBN: 978-1-118-99940-0

⁴ Available to download for APM members at https://www.apm.org.uk/sites/default/files/APM%20Body%20of%20Knowledge%206th%20edition_secure.pdf. Alternatively an online version is available for all at knowledge.apm.org.uk

⁵ *Guide to the System Engineering Body of Knowledge*, v1.3, available at <http://www.sebokwiki.org/>

2. Definition of a process

In this chapter we consider various definitions of a process to determine if there are any issues arising from them.

Reference definitions

Within the various key reference sources, there are a number of definitions of a process.

Collins Dictionary⁶ defines a process as

A series of actions which are carried out in order to achieve a particular result.

ISO15288:2015 defines a process as

A set of interrelated or interacting activities, which transforms inputs into outputs.

ISO 21500:2012 defines a process as⁷

A set of interrelated activities... Project management processes ...determine how the activities selected for the project are managed.

There are no discernible issues arising from the basic definitions of processes. There are however situations or frameworks where an activity, e.g. Risk Management, may be termed differently. For example, in PRINCE2 Risk Management is a *Theme* (or *Component* in past versions), in MSP it is a *Governance Theme* and in Management of Portfolios it is a *Delivery Practice*. For the purposes of this document these terms will be considered as synonymous – they all include the basic processes within risk management (albeit each is focused on the different perspective of each framework). Refer to the individual frameworks for further information and definition of the nuances behind the different terms.

The term ‘stage’ is used in the INCOSE SE Handbook to denote an element of a life cycle. The term ‘phase’ is used in P3M literature, for example the APM Body of Knowledge. Whilst these are generally interchangeable, the term ‘stage’ has specific connotations (for example in PRINCE2) and so in this document the term ‘phase’ will be used to denote an element of a life cycle (except where a specific reference uses an alternative term).

⁶ Collins Learner Dictionary, online version, <http://www.collinsdictionary.com/dictionary/english-cobuild-learners/process> [accessed 30th December 2014]

⁷ Quoted in ISO15288 from ISO9000:2005

3. SE process frameworks

In this chapter we identify frameworks that describe or define processes used within Systems Engineering.

ISO15288:2015

ISO15288:2015 defines four activity groups to support SE, as shown in Figure 2, each with a number of identified processes. The standard defines each of these processes.

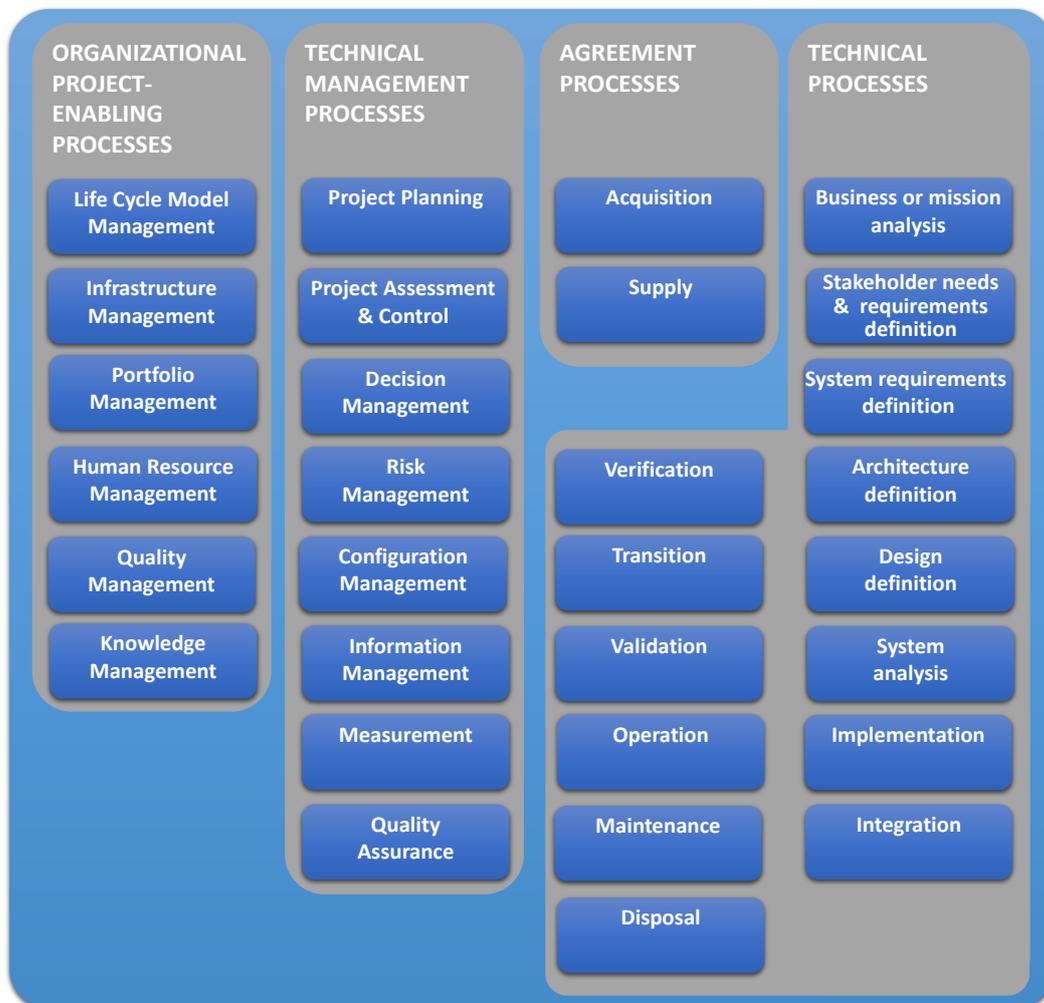


Figure 2: An overview of ISO15288:2015 System life cycle processes

INCOSE Systems Engineering Handbook

The processes, and process groups, defined in the SEHBK are those of ISO15288:2015 shown in Figure 2. Note that the SEHBK also includes a mapping of inputs and outputs between processes.

4. P3M process frameworks

In this chapter we identify various frameworks that describe or define processes used within Project, Programme and Portfolio Management.

APM Body of Knowledge

The APM Body of Knowledge (APMBOK) is a framework and taxonomy for communicating and describing the various elements that make up or support P3M. It does not proscribe actual processes for project, programme and project management.

It does describes a list of activities that fall into four main sections, each of which contains sub-sections that group key related elements together, as shown in Figure 3.



Figure 3: An overview of the elements within the APM Body of Knowledge

PMI® PMBOK®

The Project Management Institute Guide to the Project Management Body of Knowledge (PMBOK® Guide) groups individual processes according to the process group in the prescribed PMBOK® life cycle⁸ and the various knowledge areas, as shown in Figure 4.

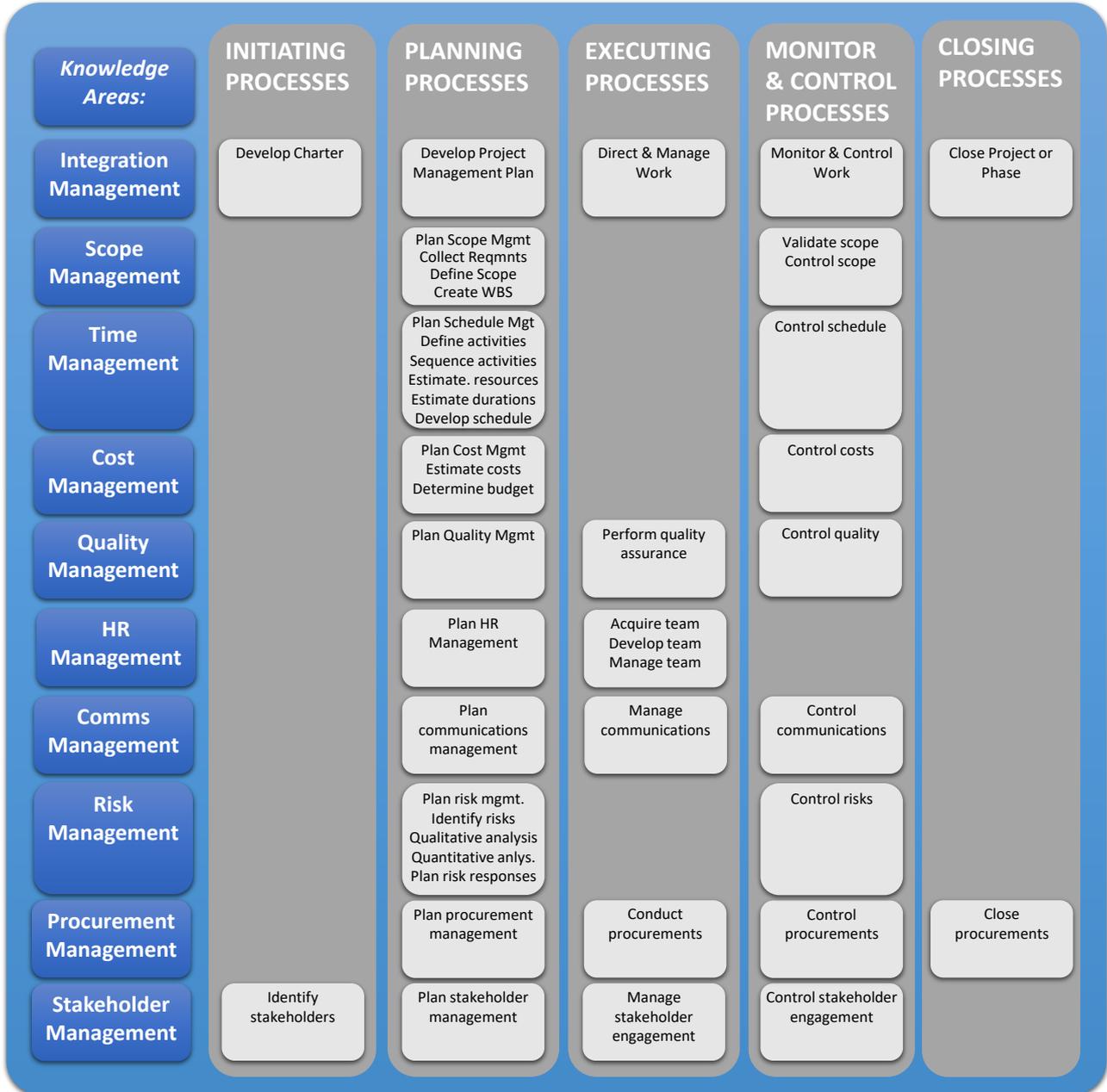


Figure 4: A summary of PMI® PMBOK® Guide process groups and knowledge areas

⁸ For further information on this life cycle see *Guide to Lifecycles and Life Cycle Models*

PMI® Standard for Program Management

The PMI Standard for Program Management framework is composed of a set of *Performance Domains* with a series of *Supporting Processes*. The *Supporting Processes* are broken down into nine Knowledge Areas, each consisting of a number of individual processes. The *Performance Domains* and *Supporting Processes* are shown below in Figure 5.



Figure 5: An overview of PMI® Program Management processes

ISO Standards

ISO21500:2012 Guidance on Project Management contains a similar breakdown of processes, shown in Figure 6, to that of the PMBOK®, although there are some differences in the framework and terminology used.

At the time of writing, the ISO Guidance on Programme Management (ISO21503) is still in development, and ISO21504:2015 Guidance on Portfolio Management has just been published (July 2015). The latter does not contain process definitions in the same manner as ISO21500:2012, but provides descriptions of aspects of Portfolio Management aimed at the understanding of portfolio management within senior management teams.

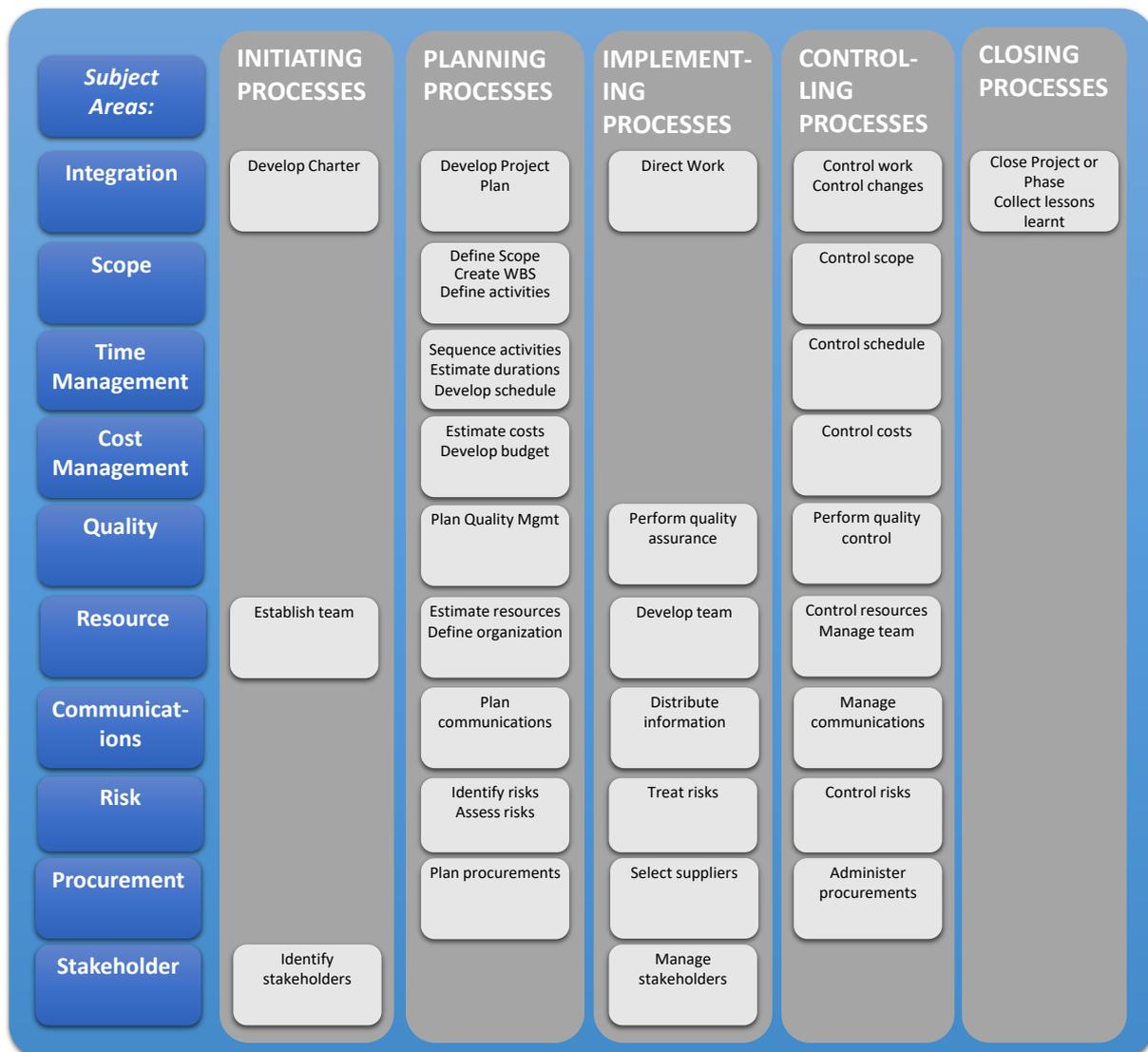


Figure 6: An overview of ISO21500:2012 process groups and subject areas

Axelos Global Best Practice suite

The following frameworks for Project, Programme and Portfolio Management were originally developed through the UK Office of Government Commerce (OGC) (and preceding organisations), but are now managed through Axelos Limited, a joint venture between The Cabinet Office and Capita⁹.

Project Management – PRINCE2®

PRINCE2® is a structured method for Project Management, originally launched in 1996 and used by the UK Government and other organisations. It has transformed through various editions, and is currently at the 2009 Edition.

⁹ For further details, refer to www.axelos.com.

It is expressly designed to be flexible to the project environment and context, and able to be tailored to the project scale. It contains definitions of Project Management *Principles*, *Themes* and *Processes* (and a few techniques).

The *Processes* describe the steps through the project life cycle (covering the direction, management and delivery of the project), and the *Themes* describe the aspects of project management that have to be undertaken throughout these steps, as shown in Figure 7. They can therefore be said to form a combined suite in a similar manner to the *Processes* and *Knowledge/Subject Areas* in PMBOK® and ISO21500).



Figure 7: An overview of PRINCE2® processes and themes

Programme Management – MSP®

Managing Successful Programmes® (or MSP®) provides a framework for the management of programmes, bringing together *Principles*, *Themes* and *Transformational Flow Processes* in a similar manner to PRINCE2® for projects. It was launched in 1999, and its use is extending across a wide range of public and private organisations who are undertaking changes of different forms within businesses or societies.

Figure 8 shows the themes and processes defined within MSP®.



Figure 8: An overview of MSP® themes and transformational flow processes

Portfolio Management – MoP®

Management of Portfolios® (or MoP®) is a guide to the *Principles* and *Practices* of Portfolio Management. It is based around a model of portfolio management that consists of a portfolio definition cycle and a portfolio delivery cycle that provide a framework for the *Practices*, all underpinned by the *Principles*, as shown in Figure 9



Figure 9: An overview of MoP® definition and delivery practices

5. A comparison of SE and P3M processes

In this chapter we explore where and how the SE and PM processes interact according to various models, and develop a taxonomy to describe the nature and scale of this interaction.

The ISO comparison for systems engineering and project management

Figure 10 shows a comparison between SE (ISO15288:2008) and Project Management (ISO21500:2012) processes, showing where the process definitions overlap or map to each other. Because of the manner in which the standards are defined, they allow for a consistent comparison between the process definitions.

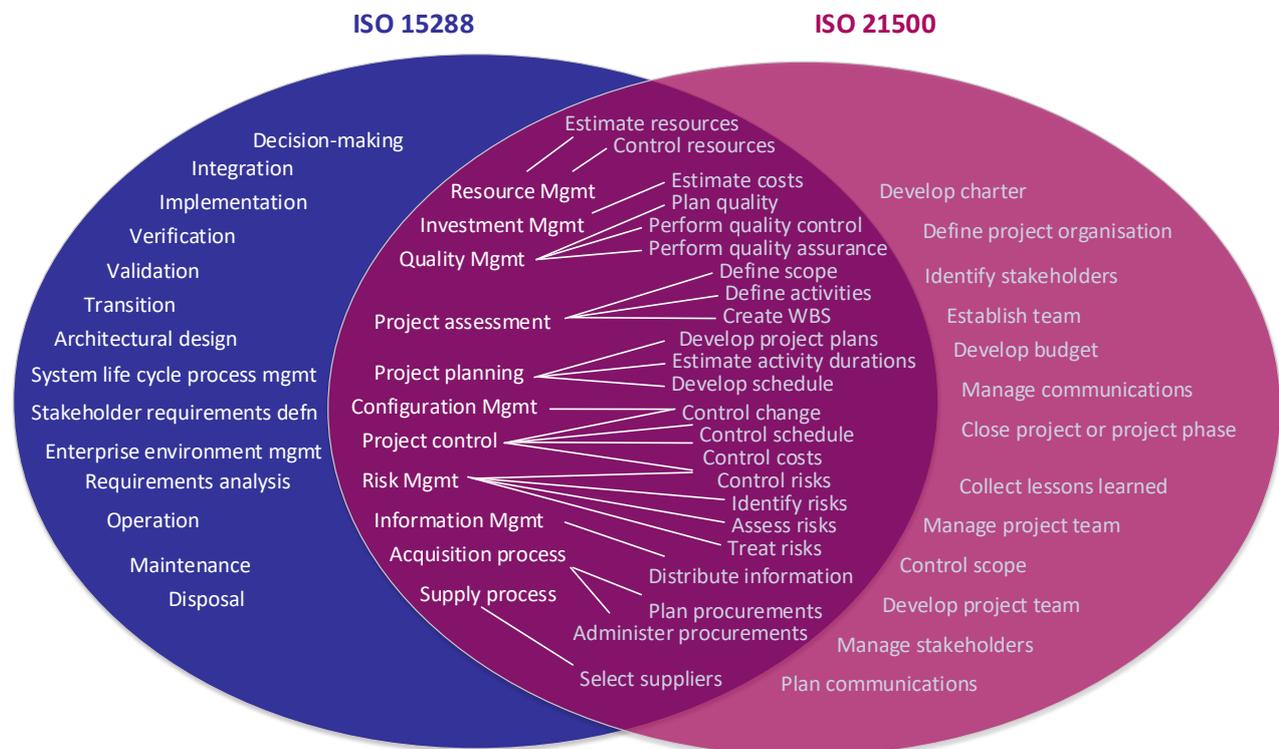


Figure 10: A mapping between processes defined within ISO15288:2015 and ISO21500:2012

Touchpoints and overlaps in programme and portfolio activities

The ISO standard on Guidance on Programme Management is intended to be ISO 21503, which is currently in development. The corresponding Guidance on Portfolio Management is ISO 21504:2015. However this is not defined in the same manner as ISO 21500:2012 (i.e. process based) and is geared towards written guidance for senior leadership. It is therefore not practical at this time to undertake a comparison in the same manner as for ISO 21500:2012.

However, it is worth considering and comparing the underlying framework principles between programme management and systems engineering, particularly in the initial Concept and final Delivery / Production phases of the life cycle.

In the Concept phase (being the term used by both the APMBOK and the SEHBK), both PM and SE processes are focused on defining the actual need and viable options for potentially satisfying that need. Both disciplines are utilising stakeholder engagement processes, as well as architectural approaches to defining a blueprint for the change activity to follow. There is also considerable synergy between the elicitation and definition of user requirements, and the identification and definition of intended end benefits.

Synergy also exists within the Delivery (APMBOK) or Production (SEHBK) phases in the transitions that occur from the system under development into new (or modified) capabilities within the operational environment. Furthermore the validation of the capabilities provided by the new system should involve the realisation of benefits and thus confirmation of the driving vision behind the programme.

The accompanying document in this series “**Integrated Life Cycle Representation**” describes the definition of an integrated SEPM life cycle, based upon the Vee-Model representation, shown in Figure 11, which in particular brings together the relationships between programme management and systems engineering / systems architecture activities.

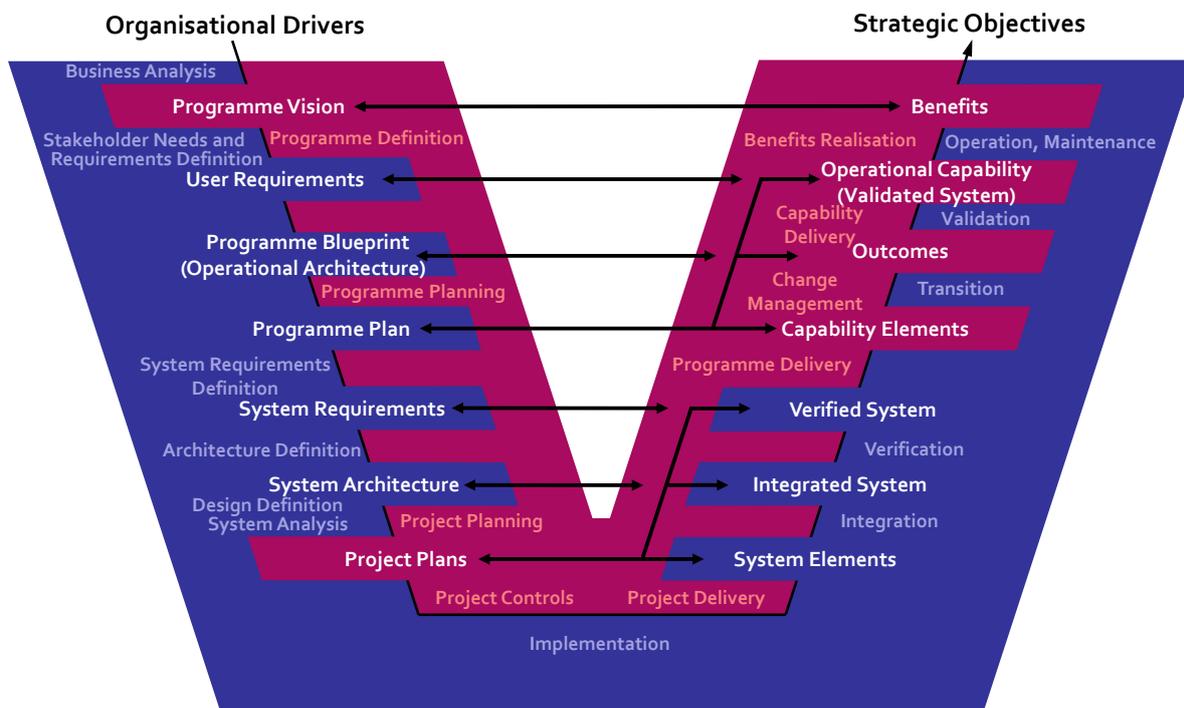


Figure 11: The Integrated SEPM Vee-model

An integrated suite of process products

As a further comparison in addition to the integrated Vee model shown in Figure 11,



Table 2 below illustrates the combination and overlaps arising from some of the products associated with both the SE and PM processes during the programme/project life cycle.

Table 2: A set of SEPM process products through the programme/project life cycle

Process Category <i>Key Processes</i>	P3M Key Products	SE Key Products
Programme Identification	Vision Statement Programme Mandate Programme Brief Initial Business Case Initial Risk Register Benefits Map Benefits Realisation Plan	Define the Business or Mission Analysis process to define the business or mission problem or opportunity Characterize the solution space Determine potential solution class(es) that could address problem or take advantage of opportunity Initiates the lifecycle of the System of Interest by defining the problem domain Identify: major stakeholders, environmental conditions & constraints. Develop preliminary life cycle concepts for through life considerations Define business validation criteria Proposed/Initial Concept of Employment (Context)
Programme Definition <i>Stakeholder Requirements Definition</i>	Programme Blueprint Programme Definition Document Programme Plan Output Descriptions Projects Dossier	Stakeholder needs & requirements definition System requirements definition Integration Test & Evaluation Plan Stakeholder Use Cases System Boundary External ICDs Concept Documents Enterprise-level Architecture Verification and Validation Plan Maintenance & Disposal Plan Transition Plan
Programme Execution <i>Transition, Validation, Operation, Maintenance, Benefits Realisation</i>	Benefits Review Report(s)	Verification Report Transition Report Realised Capability Validation Test Report
Project Definition <i>Requirements Analysis Architecture Design</i>	Project Mandates Project Briefs Project Initiation Documents Project Plan Product Descriptions	Stakeholder Requirements System Requirements System Use Cases System Architecture Verification & Validation Plan Integration Test & Evaluation Plan System Design Specification Internal ICDs
Project Execution <i>Implementation Integration Verification</i>	End Project Report Benefits Review Plan	System Elements Verification Reports Integration Test & Valuation Report Realised System Validation Report

A taxonomy for the range of interactions

The interaction between the disciplines of Systems Engineering (SE) and Portfolio, Programme and Project Management (P3M) can occur in many different forms, and these can bring about either positive or negative outcomes. If this range of interactions is broken down into the amount or 'scale' of the interaction and the nature of the interaction then it can be represented as shown in Figure 12.

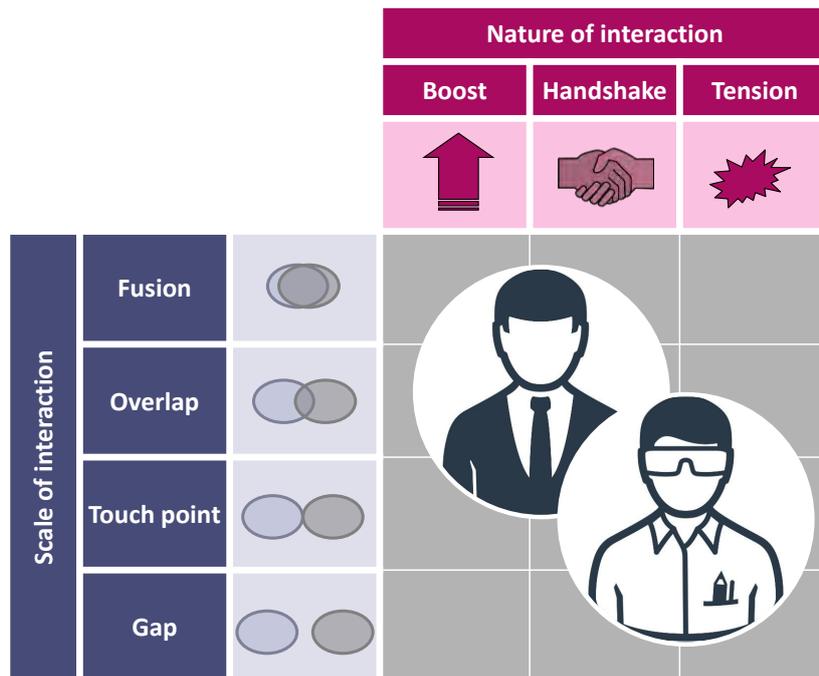


Figure 12: Representation of the range of interactions between SE and P3M disciplines

The ‘Scale of Interaction’ can be regarded as a spectrum running from a gap between the two disciplines up to a degree of fusion between the disciplines. The ‘Nature of Interaction’ is a consideration of the resultant outcome of the interaction – a ‘boost’ being seen generally as a positive beneficial outcome (where the combination of the two disciplines is indeed greater than the sum of the individual elements, or $1+1 \gg 2$), a neutral acceptance and facilitation of the interaction (or $1+1=2$) or a tension where the disciplines (and/or the practitioners) are in conflict (or $1+1 < 2$).

Table 3 below describes each of the elements of this range of interactions.

Table 3: A taxonomy for the range of Interactions between SE and PM

SCALE OF INTERACTION	NATURE OF INTERACTION		
	Boost	Handshake	Tension
<p>FUSION</p> <p>Both disciplines are working in an integrated approach, merging their respective perspectives to form (new) ways of thinking to address problems and challenges. Differences between the disciplines are blurred.</p>	<p>A fusion is a boost when the two disciplines merge and can use differing perspectives as a creative force.</p>	<p>A fusion is a handshake when responsibilities are understood and exploited successfully whilst separate identities are maintained</p>	<p>A fusion is a point of tension if the representatives do not share a common vision and are culturally divergent in shared activities. This pulling in different directions leads to inefficient and ineffective overall performance.</p>
<p>OVERLAP</p> <p>Responsibility for this work is part of both SE and PM disciplines, though possibly with different terminology.</p>	<p>An Overlap is an energy Boost when the responsibilities and methods being worked together lead to an overall increase in performance and a new level of joint understanding and maturity for the endeavour.</p>	<p>An Overlap is a Handshake if the responsibilities and methods for work transfer are clear and agreed, and the performance is monitored by both sides, working to agreed standards and terminology</p>	<p>An Overlap is a tension if the responsibilities and methods for work transfer are not clear and agreed, leading to work duplication, conflicting terminology or standards, and the performance is not monitored by both sides, leading to delays or errors</p>
<p>TOUCHPOINT</p> <p>There is a clear natural distinction between SE and PM disciplines – work is passed to and fro with no confusion. Both perspectives are looking to achieve the same objective, but may use different concepts or terminology.</p>	<p>A Touchpoint is an Energy Boost if the information being passed to and fro leads to improved performance and understanding on one or both sides. The two disciplines leverage off each other.</p>	<p>A Touchpoint is a Handshake if the responsibilities and methods for work transfer are clear and agreed, and the performance is monitored by both sides</p>	<p>A Touchpoint is a tension if the responsibilities and methods for work transfer are not clear and agreed, and the performance is not monitored by both sides, leading to delays or errors</p>
<p>GAP</p> <p>Responsibility for this work may be part of either/both SE and PM disciplines, though possibly with different terminology, and is not being addressed by either discipline or it is being undertaken by a separate (business) function with minimal interaction between the disciplines.</p>	<p>A Gap can be an energy Boost if the gap is recognised as the responsibility of a third discipline and that discipline is bought into an integrated way of working (e.g within an Integrated Project Team or undertaken at Portfolio or business level), or if the Gap is recognised and closed by SE or PM disciplines, such that overall team performance is attained.</p>	<p>A Gap is only a Handshake if the need for this work is considered and it is agreed that it is genuinely not required by both disciplines, or the responsibility is agreed and accepted through a third party.</p>	<p>A Gap is a tension if the responsibilities and methods for work transfer are not clear and agreed, or the SE and PM disciplines are heading in separate directions, leading to required work not being covered, and the performance is not monitored by both sides, leading to delays or errors</p>

An example of gaps, touchpoints, overlaps and fusions

An example of the different scale of interactions can be seen when considering the steps in devising a project plan based on a breakdown of an output system (in PM terms, a Product Breakdown Structure or PBS). Typical steps involved in such a process are shown in Figure 13, which is taken from the *Product Based Planning technique* of PRINCE2®.

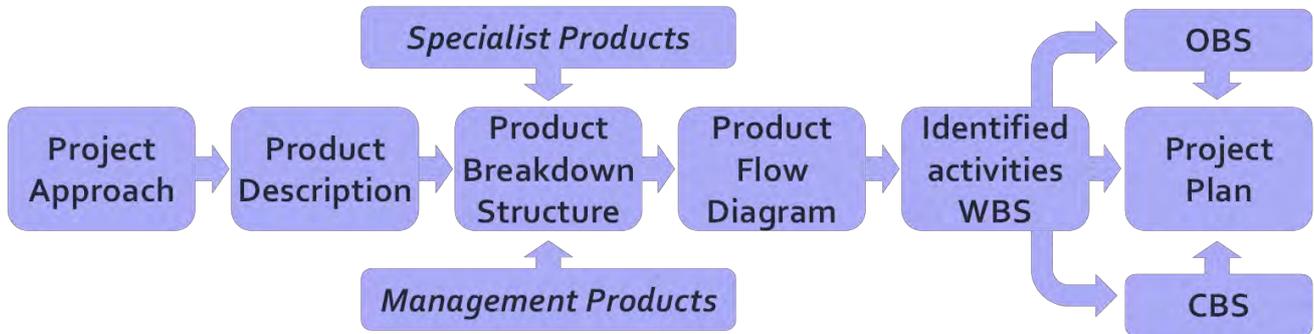


Figure 13: The steps involved in project planning – a project perspective

In Figure 13 the *Specialist Products* are the final system outputs, the *Management Products* are items created during the course of the activity to manage the development of the system, the *WBS* is the *Work Breakdown Structure* (breakdown of activities or tasks within the project) the *OBS* is the *Organisational Breakdown Structure* (the team/resource structure to undertake the tasks) and the *CBS* is the *Cost Breakdown Structure* (the project financial breakdown based on the tasks and the products created).

Gap

In the Gap situation shown in Figure 14 the System Engineer (SEng) is defining the detailed product system breakdown and this may be their prime deliverable. They may be working within an Engineering Function with little regard for the project environment. At the same time, the Project Manager (PMgr) is defining a plan based on what they believe the top-level system elements are, not the detailed information provided by the SEng. This can arise when the type of new output product is consistent in general terms with past activities and outputs (for example in a manufacturing environment), and so there is past evidence and metrics upon which such a plan can be devised (a standard project approach).

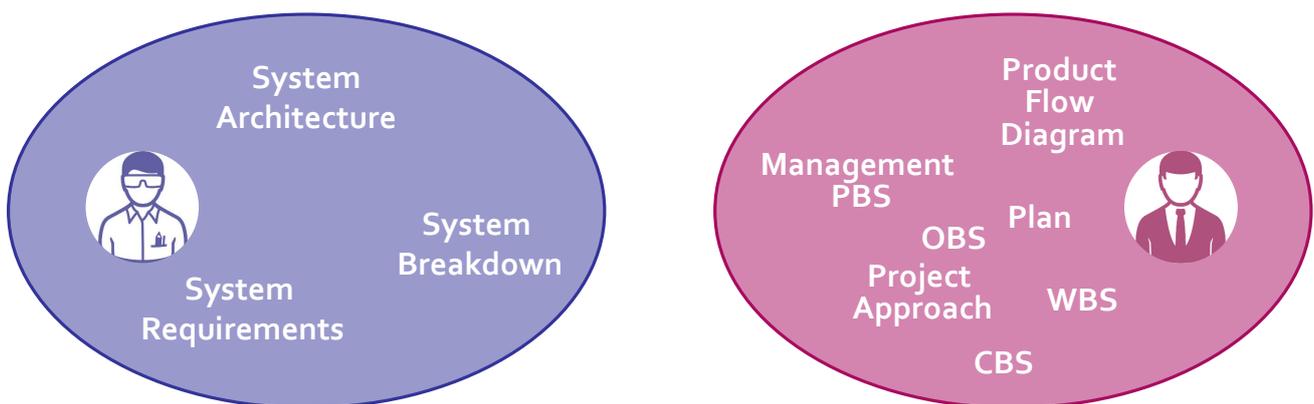


Figure 14: A gap between SE and PM practitioners when undertaking project planning

Touchpoint

Figure 15 now shows an example of a Touchpoint. Here a combined product breakdown structure is built of two halves – the SEng passes over the product system definition, and the PMgr adds the elements that make up the management system. But this is a one-directional flow of information – the PMgr then proceeds to

define the remainder of the project information based on the information passed to them, possibly in ignorance of the value that resides in the wider SEng activity.

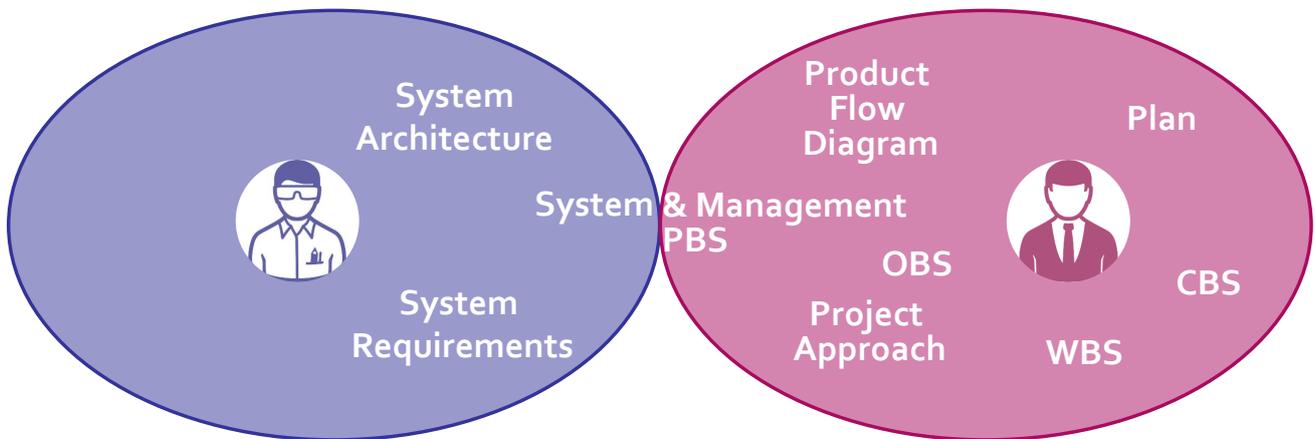


Figure 15: Touchpoint between SE and PM when undertaking project planning

Overlap

In the overlap situation (Figure 16), the SEng and PMgr are working together on combined elements – the SEng is defining the system and knows what is required to produce it, how it should be produced and who should be doing that work. The PMgr therefore leverages off that knowledge to produce the overall final plan. But it is still perhaps a ‘them and us’ mentality between the disciplines.

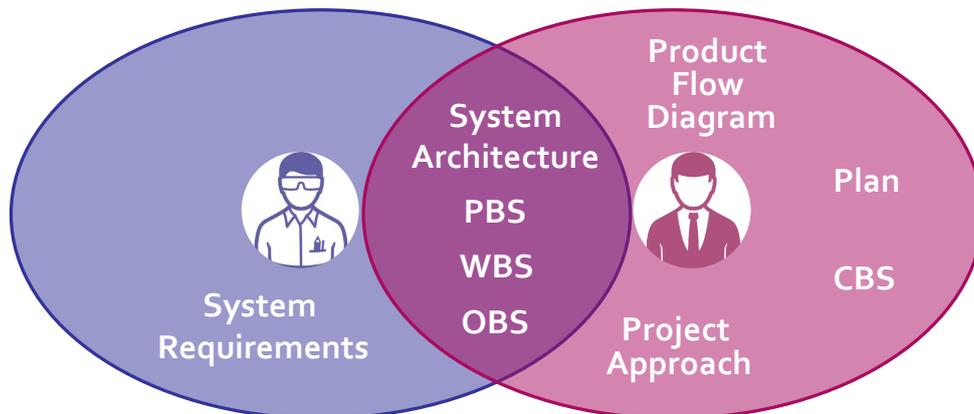


Figure 16: Overlaps between Se and PM when undertaking project planning

Fusion

Finally, in the Fusion scenario illustrated in Figure 17 the two disciplines are seamlessly complimenting each other throughout the value chain that makes up the product based planning process. In this situation it is important to have defined roles and responsibilities (otherwise tensions will form) but those roles work hand in hand, with the SEng involved from the start of the planning process in helping to define the overall project approach.

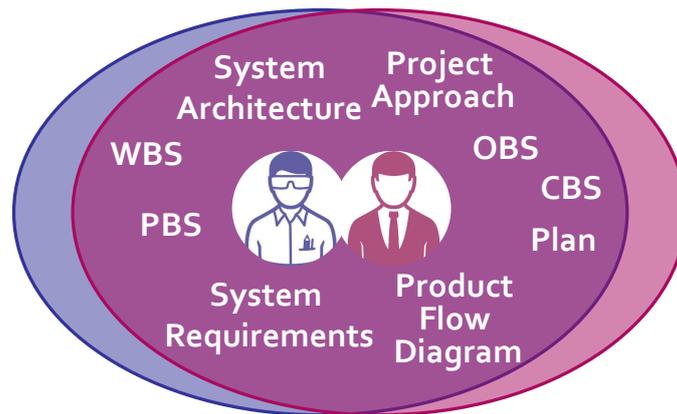


Figure 17: SE and PM fusion when undertaking project planning

A dose of reality

The examples above are simplified in black and white terms to illustrate the main concepts. Actual boundaries and scopes will be fuzzy and grey. But this simplified breakdown helps to articulate where both disciplines (and their different perspectives) could work together – and also to help identify whether the outcome of the interaction is a benefit or a penalty. Increased benefits can be identified when ‘moving up the scale of interaction’ from gaps through touchpoints and overlaps to fusions, but it is important to ensure that these benefits can be obtained rather than creating tensions or being restricted by tensions.

6. Examples of SE and P3M process fusion

In this chapter we identify key areas where SE and P3M processes can combine or interact to provide an enhanced performance or output.

Complementary processes or frameworks

As an overall observation regarding fusion points, it is worth noting a view¹⁰ of system engineering values as

- a. A way of thinking about complex problems;
- b. A way of delivering transformation / enduring change
- c. A way of bringing together disparate disciplines.

This is also a way of defining the values of P3M. With such a common set of values, it is not surprising that there are a number of areas where SE and P3M processes can come together for mutual benefit. This section considers examples where systems engineering and P3M processes and approaches can be used in support of each other, including:

- Governance
- Product based planning
- A programme as a system of systems
- Portfolio and programme architectural modelling
- Verification and validation in benefits management
- Agile requirements satisfaction: Clear thinking in the agile environment
- Interdependency management
- Soft systems methodologies in stakeholder management
- Integrated supply chain management
- Requirements definition in contracts
- Transition definition and management

The following sections describe each of these fusion areas in a summarised form. Further details will be found in a series of individual definitions being produced by the Joint Working Group.

¹⁰ Dr Michael Wilkinson, speaking at the APM / INCOSE UK Joint Workshop on Systems Engineering and Project Management, January 2013



Governance

Project, programme and portfolio governance can be strengthened through incorporating key SE approaches, techniques and tools (and how these elements and processes interact), particularly in dealing with areas of higher complexity. Principles such as multidisciplinary approaches, integration, open modular design and integrated product teams can all support robust planning and controls leading to improved governance.

Product based planning

Product-Based Planning is a technique described in the PRINCE2® framework which becomes much more powerful when integrated with the Systems Engineering techniques used to develop the Product Breakdown Structure (PBS) and Work Breakdown Structure (WBS).

A PBS is a hierarchical decomposition of the products that are required in order to deliver the proposed solution (including project management products – eg plans - as well as project deliverables). This is then translated into the activity-orientated product flow diagram or WBS to define what needs to be done.

Systems Engineering techniques and methods provide the robust means to address key elements of the product-based planning process (items in brackets are PRINCE2® processes):

- identifying options, constraints and how the project is going to be addressed (defining the Project Approach);
- defining acceptance criteria, confirmation of success, change control, selection of appropriate quality assurance methods (Planning Quality);
- identifying products, sequencing and decision points (Defining & Analysing Products);
- identifying external constraints, overlapping activities, dependencies (Identifying Activities and Dependencies).

This joint approach, which continues through the life cycle, provides a complete description of the work to be undertaken, the interdependencies and acceptance criteria, which leads to robust planning and scheduling and hence a greater likelihood of project success.

A programme as a system of systems

A system of systems approach considers “any technology-assisted human enterprise as a designed arrangement or organisation of a number of systems”, where the wider system has properties over and above those of the individual sub-systems (“emergent properties”) and where individual sub-systems that work together are interoperable and compatible (“coherent”). The MOD System of Systems Approach, JSP 906, establishes common ways of working, common languages and common products. This concept goes beyond the definition within the SE Handbook, which is primarily focused on ‘operational’ or ‘product-based’ system of systems.

A programme (according to the MSP® framework) is “a temporary flexible organisation create to co-ordinate, direct and oversee the implementation of a set of related projects and activities in order to deliver outcomes and benefits related to the organization’s strategic objectives”. In abstract terms, if a project can be considered as a system to deliver specific products, then the programme is focused on how these systems interact with each other, how to deliver an overall coherent capability, managing the external boundaries with business-as-usual operations and dealing with stakeholders. Programme management is appropriate where it delivers value over and above those from individual separate projects. There are strong parallels with the challenges for developing system of systems: system elements could operate independently, have different life cycles, initial requirements are likely to be ambiguous, complexity is a major issue,

management can overshadow engineering, fuzzy boundaries cause confusion and system of systems management continues through life. These also are the challenges facing programmes, so where a systems engineer addresses these challenges in a system of systems environment, they will be directly supporting the work of programme management.

Portfolio and programme architectural modelling

Enterprise architecture principles (or model-based systems engineering) are used in defining operational models for current and future states (Target Operating Models). These are also synonymous with the architectural principles behind programme blueprints (especially as defined within the Managing Successful Programmes® framework) as both disciplines are aimed at defining a holistic representation of all the elements that make up organisational states and what has to change. But the principles behind architectural frameworks such as MODAF and TOGAF can be extended to portfolio and programme definition to define change and its relationship to the wider organisation.

Verification and validation in benefits management

Benefits realisation planning and delivery is focused on the definition of benefits, their dependencies and how (and when) these benefits are expected to be confirmed as realised. As such, benefits realisation planning demonstrates the means of verifying the benefit realisation in terms of the relationship with capabilities generated, outcomes achieved and transitions undertaken. Validation of the benefits comes with ensuring that the end benefits actually realised align with those that underpin the change vision (and detailed through the user requirements and blueprint).

This activity can therefore be seen as a continuation of the integrated test, evaluation and acceptance planning and execution that form a key part of a systems engineering approach. The means by which robust user requirements are defined with stakeholders (eg the Validation process within ISO15288) is also the means by which benefits are identified and quantified, and benefits (intermediate or end) provide justification for user requirements. The discipline of the requirements processes and the Integrated Test, Evaluation and Acceptance Plan (ITEAP) should be repeated (or indeed continued) into the Benefits Realisation Plan, and thus the relative skills with the PM and SE communities should support each other.

Agile requirements satisfaction: Clear thinking in the agile environment

When using any of the agile methodologies it is important to identify when requirements may be addressed. Some requirements may be addressed after a small number of sprints, others may take longer. These two areas have been called Mode 1 & Mode 2 ways of working. However, it may be useful to consider the different requirement types as linear and dynamic where linear requirements are those that are delivered to support the agile process and dynamic requirements are those that are delivered using the agile methodology. Both types of requirements are pertinent to agile delivery, though linear requirements become more important when an agile methodology is used at scale, e.g. across a number of development teams working on the same project or programme.

In general, linear requirements may be thought of as those that either need a longer term stable delivery environment, or those that need a momentum to be maintained for their delivery, e.g. ones that need human interaction and agreement. Examples of requirements that may be considered to be linear are: the delivery of an infrastructure that the software being delivered in an agile manner runs on, or data sets that are required to be used by software being developed using an agile methodology.

Dynamic requirements are those that will be subjected to definition changes during their delivery, and the agile method is well suited to deliver these. Generally, they are ones that are user-facing and delivered using software.

Both types of requirements need to be defined to an appropriate level of granularity to allow testing to take place and need to be subjected to governance during their lifetime. However, it may be appropriate to use different tools or methods for the two requirement types. E.g. creating a URD & SRD may not be appropriate for a set of dynamic requirements.

Interdependency management

A key element of programme management, and a source of its potential complexity, is the recognition and management of interdependencies between projects (sub-systems), and between the projects and programmes and those elements outside the programme (system) boundary. These interdependencies represent information or physical items that pass across interfaces between different project or programme elements. In the SE environment, “focus is placed on controlling the interfaces between system elements and external systems” in order to help address ambiguity, fuzziness and complexity. Therefore a robust systems engineering environment, with its attendant focus and discipline around interfaces, should be fully integrated with the work of the project, programme or portfolio definition and execution.

Soft systems methodologies in stakeholder management

Two of the biggest challenges for Project or Programme Managers managing a complex ‘soft’ problem are the ability to properly understand and scope the problem, and the difficulties around managing a group of disparate stakeholders. Soft problems can often be emotive and so logic does not always prevail and stakeholders need to feel involved in the decision if the Programme is to have any hope of achieving milestones. Problems that involve human behaviour are particularly complex, and many projects fail because the ‘human element’ is not properly accounted for. Soft Systems Methodology (SSM) can be integrated with PM principles to effectively understand the complexity of a project, and the influence of the stakeholder environment on that complexity. SSM techniques can be used to achieve consensus, especially where there are differing opinions within the stakeholder community.

Integrated supply chain management

The whole-life holistic perspective of the SE disciplines, and their supporting processes, provides a firm platform for understanding how a change activity, and its subsequent introduction into normal steady-state operations, is to be achieved within the context of a supply chain. Links between the supply chain elements, and their relative importance at different stages of the life cycle, can be captured, modelled and implemented through systems engineering practices to arrive at a robust sustainable strategy for supply chain management for ongoing definition, production, implementation and through-life logistics support.

Requirements definition in contracts

Strong contracts require a robust specification of the work to be undertaken (in particular outputs) and the relationships with other areas of the project. Specifications that are fully aligned with (and derive from) user and system requirements will therefore be integrated with the wider project. The test and acceptance criteria that will not only provide contractual confirmation for payment, but will also be aligned with project schedules and the vertical verification activity. Passing requirements information through the contract may also provide a wider contextual background that would help suppliers make robust decisions affecting delivery.

Transition definition and management

According to the definition of the ISO15288:2008 Transition Process, its purpose is “to establish a capability to provide services specified by stakeholder requirements in the operational environment”. It defines how to plan and perform the transition, and how to generate the requirements for a successful transition,



installation procedures and constraints. It therefore fully supports the Realizing the Benefits process within the Managing Successful Programmes® framework, which also describes the transition to business as usual operations for a change programme. By fusing the two disciplines, a robust transition can be defined, particularly if the system of interest encompasses both the outputs/outcomes as well as the wider business and operational environment.

7. Areas of process friction or tensions

In this chapter we review areas where tensions and frictions can arise between SE and P3M processes.

Change is not straightforward - tensions exist

Tensions can arise not only from actual SE/PM perspective differences but also from preconceptions and mis-communications. Terminology clashes, over-elaboration in both requirement setting and project planning, overlaps or gaps in responsibilities and a failure to articulate the value of SE or PM processes to leaders or teams all contribute to tensions, underpinned by a lack of mutual understanding and respect¹¹.

In addition, any process issues are exacerbated by the ‘tensions field’ shown in Figure 18 that operates within project and programme environments due to the differing demands and objectives that are present. Both SE and PM practitioners must recognise and understand how their perspectives and actions both affect, and are affected by, these tensions.

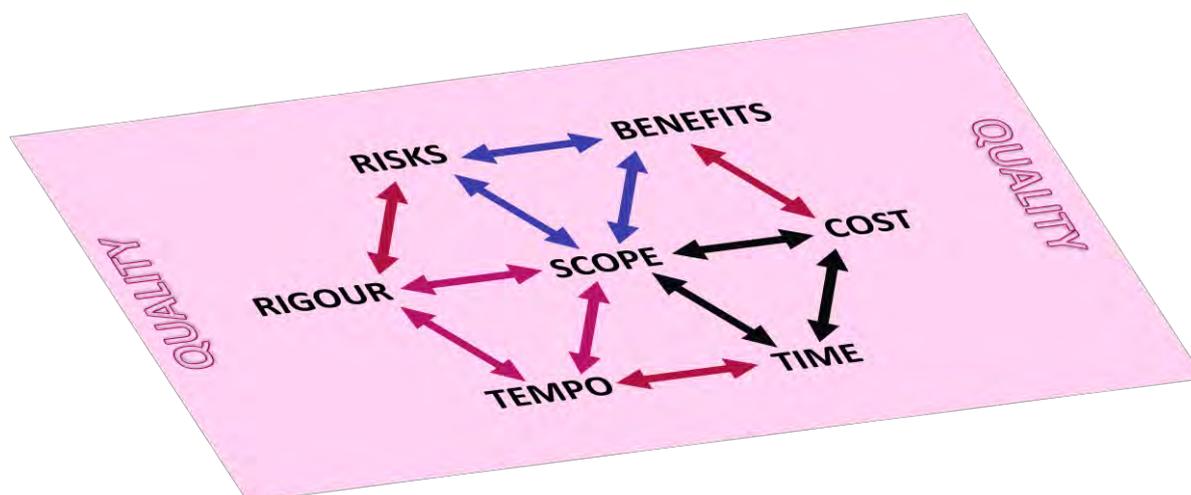


Figure 18: Project/programme tensions field

Figure 18 includes the classic ‘Iron Triangle’ of project tensions between scope, cost and time, as well as the tensions in agile environments between tempo, scope and rigour and the programme tensions between risk, benefits and scope. However all of these elements interact with each other, for example continually justifying the change through cost vs benefit, or time to market limiting the number of design iterations.

Quality is another parameter that is typically included: however quality is normally a set requirement – or at least a minimum quality level is required. In this representation it is the ‘field’ itself – various quality requirements will affect the strength of all of the different tensions.

Is it too much bureaucracy or robust governance?

“Too much bureaucracy” is a charge often levelled at project environments where there seems to be (from a certain perspective) excessive paperwork or ‘hoops to jump’ before progress can be made to the next phase

¹¹ See “Integrating programme management and systems engineering: A mutualistic approach”, Mark Fielding-Smith, INCOSE UK ASEC 2011 [available through http://incoseonline.org.uk/Program_Files/Calendar/Show_Event_Details.aspx?CatID=Events&EventID=178 : accessed 4th May 2015

of work. The time taken to obtain approvals through stages/tranches/business cases can be seen as delays or lost time, leading to frustration or a drop in momentum.

Whilst there will certainly be cases of stifling bureaucracy and waste, there also needs to be a level of governance commensurate with the scale of the project and the investment being placed in it. There has to be a continuous demonstration of the justification for the project¹² and the expenditure required. Both the PM and SE practitioners have to work together to understand the overall needs of the project and to find the optimum balance between the level of governance (and its requirements) and effort placed in solution generation.

When does defining a solution rigorously become gold plating?

When an emphasis is being placed on obtaining as complete (or 'best') a solution definition as possible, at the expense of management constraints (such as market entry conditions or maintaining shareholder confidence) then tensions can arise between SE and PM practitioners. The excessive refinement ('gold plating') of solution requirements can occur beyond the point at which the solution specification or definition is sufficient for work to proceed with a commensurate level of risk. In dynamic and agile environments such attention will be counterproductive, but in risk adverse or heavily regulated environments the attention to detail is important (at the relevant key points in the project).

Stakeholders will also want reassurances that their needs will be satisfied, or concerns addressed, particularly in the case of users or beneficiaries. Therefore the appropriate focus has to be placed in working to produce the necessary information, through combinations of benefits planning and user requirement setting. Governance and investment stakeholders will also want to see early progress.

Do you achieve technical maturity or meet the programme timescales?

The potential for over-elaboration in system definition also sets up a tension with the need to maintain timescales. Where the leadership comes from a PM perspective then the focus will likely be on attaining milestones according to the overall schedule. This may set up behaviours whereby the technical maturity at these milestones is compromised for the sake of schedule adherence – milestones may be passed 'with actions'¹³ – rather than accepting that the solution is not at a sufficiently mature stage. This would build up problems for the later stages of the programmes, just at a point where changes are more expensive to undertake.

However an inherent danger in a technically-led programme is that the focus on maturity occurs at the expense of time – and 'good' products end up losing market ground to technically inferior competitors. Organisations have looked to balance these tensions (not always successfully) and examples include large programmes where the initial work is solution focused and led by the technical leadership, but then passed to a programme leadership at an appropriate juncture to drive the system production, verification and validation.

Making and maintaining a justifiable business case

As stated above project justification has to be maintained throughout the life cycle, and the business case is critical to ensuring that the project rationale is defined and continually reviewed. The business case can be

¹² This does not infer that speculative or entrepreneurial projects that may ultimately end in the product not being completed should not be undertaken – the potential in developments (for example in the pharmaceutical industry) is a form of justification.

¹³ That is, the milestone is deemed 'passed' for timescale adherence but necessary changes and further developments are passed into the next phase.

perceived by SE practitioners solely as a PM product where the justification is made using economic terms, and ‘something the PM practitioners need to do to’. However a good business case contains different elements, as in the example of the UK Treasury guidelines¹⁴ (‘Green Book’) where the business case comprises:

- A **strategic** case – the rationale for why you need to undertake the programme/project
- An **economic** case – the cost/benefit analysis of the available options
- A **commercial** case – the viability of any procurement approach
- A **financial** case – the affordability of the overall programme/project
- A **management** case – the achievability of the programme/project (in terms of its execution)

As systems engineering inputs are important in support of all of these individual elements, when the business case is generated without the direct involvement of SE practitioners then there is a danger that the whole case is not made, or that there is sufficient robustness in the underlying data and the business case is not intrinsic to the whole undertaking.

The four horsemen: Ignorance, perception, interpretation and misapplication

Even with a wealth of guidance, advice, books and training courses on the theories associated with the separate disciplines, tensions will arise when individuals

- Are not aware of the needs of the different perspectives;
- Let their own negative experiences and those of others affect their perception and behaviours;
- Interpret incorrectly information they have been given and circumstances they are placed in, and
- Incorrectly apply theories and guidance.

This applies equally to both the SE and PM perspectives.

Styles and behaviours

Tensions will also arise from the culture and behavioural sets that surround each of the SE and PM communities. These can be influenced by the organisational culture in the workplace, market sector behaviours or the communities of practice and professional bodies. Where silos exist (functional or physical) then the barriers will be hard to overcome, and it will be difficult to determine if any process tensions that exist are caused by the process definition itself or by the underlying culture.

¹⁴ In particular refer to ‘Public sector business cases using the 5 case model (HM Treasury Green book supplementary guidance), available from <https://www.gov.uk/government/publications/the-green-book-appraisal-and-evaluation-in-central-government> [sic]