
ISBN for SETE2018: 978-1-925627-15-2

# The case for investment in Systems Engineering in the early stages of Projects and Programs

Stephen Cook
Shoal Engineering Pty Ltd &
The University of Adelaide
stephen.cook@shoalgroup.com

Shaun Wilson
Shoal Engineering Pty Ltd
shaun.wilson@shoalgroup.com

## ABSTRACT

*The evidence for the value of early investment in systems engineering has been accumulating for decades. There are many published papers and reports citing quantitative data and there is good agreement that an expenditure of no less than 5% of the total project cost on systems engineering is needed to achieve project success. Indeed, the evidence also indicates that SE expenditure of three times that amount is ideal because increasing SE expenditure continues to reduce project risk but does not increase the overall project expenditure; rather certain necessary activities are simply moved forward. Much of this data relates to projects that were conducted a fair time ago. This paper reviews contemporary materials and compares the new findings to the classical material. The conclusion drawn is that the recommended levels of expenditure on systems engineering are unchanged but now there is guidance to indicate the sorts of activities that provide the greatest payoff in the early stages of the lifecycle. Foremost among these is mission definition with project scoping, systems architecting, technical analysis, and system design also featuring prominently. Requirements engineering was also mentioned but not in terms of volume of effort but in terms of the need to use good systems engineering and analysis to translate the high-level requirements (that rarely change) into detailed technical requirements.*

## INTRODUCTION

Those of us who are accustomed to applying Systems Engineering (SE) principles and practices to large, complex, technical projects find it hard to envisage tackling a substantial project without a well-established SE framework and associated set of processes, methods, tools, techniques and artefact archetypes. We see the value of systems engineering to be self-evident and the case for its use axiomatic. There are, however, an increasing number of project staff who have not had the benefit of a formation encompassing engineering design and development, and these people need to be convinced of the value of using a rigorous approach and its concomitant investment to produce a task-appropriate set of SE process products (models, architectures, plans, design artefacts, documents, etc.), especially for smaller projects. There are also some who embrace the use of SE by contractors to support the design, production, and sustainment systems, but are less convinced of the need for SE in the definition of the systems to be acquired.

This paper seeks to build a business case based on published evidence to inform all project professionals that expenditure in quality systems engineering at the early stages of complex projects, regardless of their size, is a valuable investment that reduces project risks and improves project outcomes.

Cook (2000) reviewed the seminal material on systems engineering lessons learned to that date. This paper opens by recalling the systems engineering context used for that paper and the key findings presented. Next, we establish the contemporary systems engineering context for this paper which is an expansion of that used in the earlier work. A review follows that examines the literature published from 1999 onwards that forms the major contribution of the paper. The business case follows from this and draws on both bodies of work.

## HISTORICAL CONTEXT

The following description is abstracted from Cook (2000) to provide the context for the findings reported in that work. That paper described the formation of SE as a discipline following the Second World War in response to the increasing cost and technical complexity of the development and acquisition programs in the 1950s and 1960s. He states that by the 1970s, SE was seen as a successful methodology for handling the technical and managerial complexity of these large projects, and that some of this recognition was no doubt due to large program failures that could have been avoided, or at least mitigated through the use of systems engineering (M'Pherson, 1980; DSMC, 1983). He concludes the historical perspective by stating that in the 1990s, systems engineering received increased attention as ways were sought to reverse the trend of increasing project failure, particularly in large software-based systems.

SE had its origins in the telecommunications, defence and aerospace sectors, particularly in the United States of America and the United Kingdom. Cook (2000) commented on the increasing trend towards applying SE to a wider range of industries. The cancellation of many US military standards in the mid-1990s increased the rate of evolution of SE and began the process of producing domain-independent standards that were more strongly driven by industry concepts of best practice.

Cook (2000) invoked the definition of SE from the contemporary INCOSE SE Handbook (Robertson, 1998):

> *'Systems engineering is an interdisciplinary approach and means to enable the realization of successful systems'.*

Unusual for the time, he pointed out that SE can also be considered as a (human activity) system in its own right: one that creates, modifies and supports, large, complex, technical systems. He also stated that SE should not be considered merely as a set of procedures to be followed; instead, it should be thought of as a way of thinking that encompasses a set of competencies and a set of processes that can be invoked as needed, each of which can be achieved through a range of methods. He went on to say that an important aspect of SE practice is the selection and tailoring of the processes to suit each project. He finished the SE context description by discussing the rise of capability maturity models (Gabb, 1999; Sheard and Lake, 1998) and the advent of the latest standard ANSI/EIA-632, which was released in 1999 that listed 13 processes needed to engineer a system.

This was a period when, in the eyes of many, good SE equated to working within a framework embodying high levels of process maturity. Nonetheless, Cook recognized in his introduction that SE does not exist in isolation to the culture of the organisation and that project environments exist within enterprise environment as described by ANSI/EIA-632.

## FINDINGS FROM EARLY LESSONS LEARNED

NASA (1989) identifies a broad range of mistakes that have been made repeatedly in complex technical projects, some of the more salient ones being as follows:

1. **Lack of clear definition of requirements early in system design phase including:**
   a. Lack of early and thorough requirements analyses prior to the start of a design;
   b. Vague and incomplete requirements;
   c. Starting designs before the requirements were known; and
   d. Design from the bottom-up rather than the top-down.
2. **Poorly defined technical interfaces between sub-systems including:**
   a. Failure to understand interface requirements;
   b. Too narrow a focus within the discipline/subsystem level, omission of a systems viewpoint or systems engineering awareness by the discipline engineer; and
   c. Incomplete interface definition and failure to update interface documentation.
3. **Inadequate test planning during the system design phase, including:**
   a. Lack of early test planning;
   b. Inadequate consideration early in the systems design phase of how large amounts of data will be reduced, analysed, and reported;
   c. Not seeking out expert advice early to develop verification plans; and
   d. Inadequate consideration of testability to assure that the design can be tested to demonstrate specification compliance.
4. **Failure to think the design through to completion of integration including:**
   a. Inadequate consideration of accessibility/maintainability early in the design; and
   b. Inadequate consideration of handling and transportation requirements.
5. **Insufficient attention paid to the effect of mission operations on design requirements.**

The NASA study also notes that each of these could have been ameliorated by following traditional systems engineering principles and practice as espoused from the early 1980s onward (DSMC, 1983).

An important lesson from NASA at that time is that if the expenditure on the system design phases (up to and including the preliminary design phase) is <u>less than 5%</u> of the estimated cost of the project, very significant cost and schedule overruns can be expected; see Figure 1 (over) from NASA (1995).

This percentage appeared elsewhere in the systems engineering literature (M'Pherson 1990, DSMC, 1983). Furthermore, one of the tenets of the Downey procedures, employed by the UK Ministry of Defence since the late 1960s, is that '*up to 15% of the estimated total development cost may need to be spent on project definition in order to reduce risk*' (DERA, 1996).

Perhaps not surprisingly, the same source notes that such investment rarely occurs in practice and cites this as one reason for disappointing project outcomes.

Furthermore, money must be spent on actually conducting and completing the associated tasks (see say Blanchard and Fabrycky, 1998) to properly inform decision-making, not simply creating process products to satisfy bureaucratic needs not based on a solid foundation of system design and analysis.

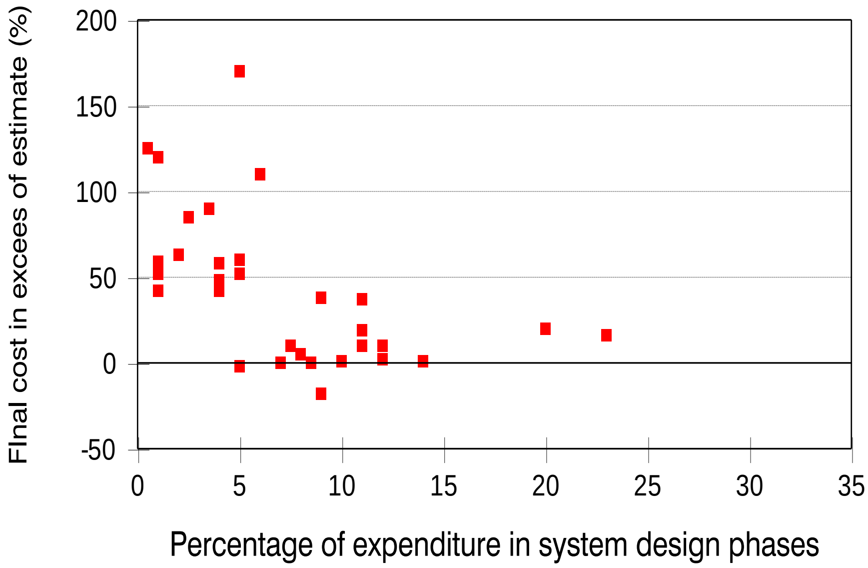


**Figure 1. Cost overrun versus expenditure in systems design phase from NASA (1995).**

The UK National Audit Office 1997 Review showed that the top 25 major equipment programs had an average slippage of 35-40 months and an annual cost growth of 7-9%. Projects missed, on average, 40% of program milestones and 10% of projects failed to meet key technical requirements. The *Learning From Experience* initiative identified the major shortcomings as being:

- Ineffective co-operation between stakeholders,
- Over-optimism in predicting time and cost,
- Insufficient appreciation of technical and commercial risk,
- Ineffective incentivisation of suppliers, and
- Insufficient investment at the project initiation stage.

The *Smart Procurement* policy that built upon the application of Downey principles was introduced to adopt a through-life systems approach to procurement, allied to improved commercial practices and measures related to personnel, training, and information systems with:

- A formal *Project Initialisation Phase* to replace the *Pre-Feasibility Phase* of the Downey cycle, the intention being to engage stakeholders and establish an applied research program;
- Co-operative stakeholder involvement with a focus on elimination of organisational and non-constructive interfaces between stakeholders;
- Concurrency and enlightened approval strategies;
- Improved requirements management termed *Smart Requirements* that saw the evolution of the *User Requirements Document* in parallel with *System Requirements Document*;
- Improved estimating and predicting; and
- Incremental acquisition.

Cook then proceeded to discuss military and commercial software lessons from the period. Jackson, (1997) was cited as an example of the poor performance of UK information technology investments across 14,000 organisations. It found that:

- 80% to 90% of systems did not meet their goals,
- About 80% were delivered late and over-budget,
- Around 40% of developments failed or were abandoned,
- Under 40% addressed training and skills requirements,
- Less than 25% fully integrated business and technology objectives, and
- Only 10% to 20% met their success criteria.

While it is hardly comforting to know that the civil sector also had great difficulty realising successful software projects, this data helped appreciate that the defence sector was in fact performing better at the time.

Attention then turned to The Standish Group (1995) research that undertook a similar study in the USA. This substantial study received responses from 365 organisations covering some 8,380 applications in banking, securities, manufacturing, retail, wholesale, health care, insurance, services and local, state and federal organisations. These were categorised by outcome per Table 1.

**Table 1. Project outcomes from The Standish Group (1995).**

| Outcome | Description | Percentage of projects |
|---|---|---|
| **Type 1**<br>Project success | Project completed on time, on budget, with all features and functions as initially specified. | 16.2 % |
| **Type 2**<br>Project challenged | Project completed but over budget, over time, with fewer features and functions than initially specified. | 52.7 % |
| **Type 3**<br>Project impaired | The project was cancelled at some point during the development cycle. | 31.1 % |

Usefully, the Standish report also offered some analysis on the reasons for projects failing and used this to produce a *Success Criteria Metric* that can be used to establish probable project success; see Table 2. The score shown in the points column of Table 2 was derived directly from survey responses and relates to the percentage of projects where this criterion determined the fate of the project.

**Table 2. Standish project success potential metric (The Standish Group, 1995).**

| Index | Success criteria | Points |
|---|---|---|
| 1 | User involvement | 19 |
| 2 | Executive management support | 16 |
| 3 | Clear statement of requirements | 15 |

| Index | Success criteria | Points |
|---|---|---|
| 4 | Proper planning | 11 |
| 5 | Realistic expectations | 10 |
| 6 | Smaller project milestones | 9 |
| 7 | Competent staff | 8 |
| 8 | Ownership | 6 |
| 9 | Clear vision & objectives | 3 |
| 10 | Hard-working, focussed staff | 3 |
| **TOTAL** | | **100** |

In addition to the survey reported above, The Standish Group conducted interviews with various managers to shed further light on why projects are beset with problems. While most comments echoed the findings of the survey, the problem of competing priorities emerged as an important issue, particularly when this stemmed from an un-resourced re-organisation.

Cook concludes his data collection with the description of the work by Moody et al. (1997) that examined aircraft developments to produce the graph in Figure 2 that shows that project performance (based on technical, cost and schedule performance) is positively correlated with systems engineering fundamentals (based on the adequacy of a number of well-established systems engineering activities, good requirements engineering, sound architectural design, thorough design reviews, rigorous systems analysis, and trade-offs).

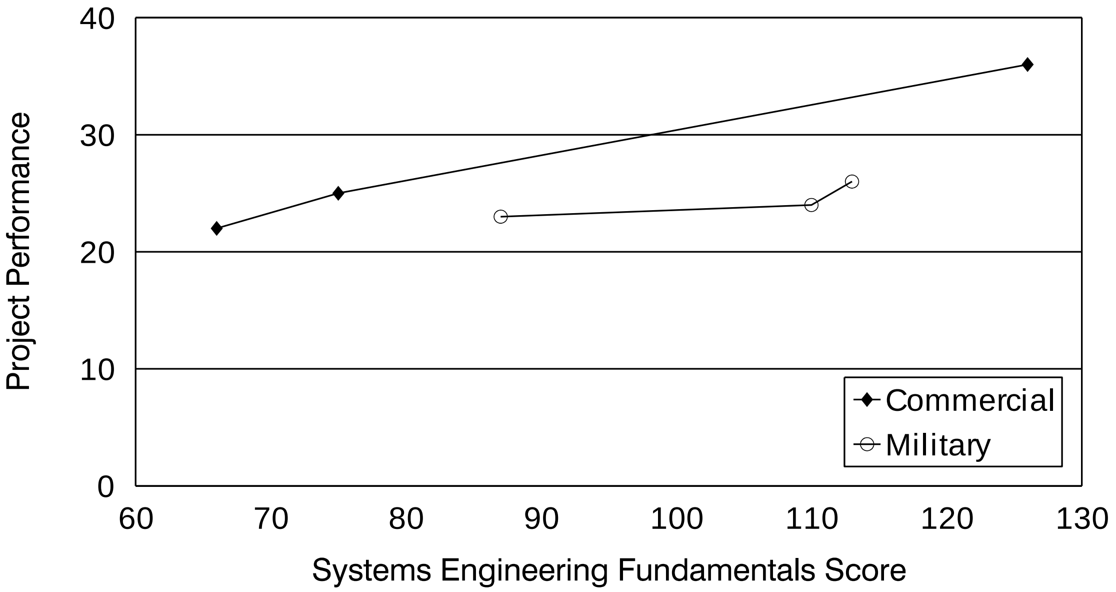


**Figure 2. Project performance score versus systems engineering fundamentals (from Moody *et al.*, 1997).**


## CONTEMPORARY CONTEXT

Sillitoe (2018) provides a new definition of SE arrived at by the INCOSE Fellows:

> *'Systems engineering is a transdisciplinary approach that applies to systems principles and concepts to enable the successful realization and use of engineered systems and whole-system solutions.'*

He then explains that this move to the new definition has four major drivers:

1. The need for a paradigm shift in the governing assumption about SE to move away from '*SE takes charge*' towards '*SE facilitates effective collaboration*'.
2. The shift in focus from the upfront definition of '*controlled*' systems operating in deterministic scenarios, towards '*learning and evolving*' systems (which might be autonomous) operating in changing and non-deterministic environments, hence the emphasis on '*purpose and success criteria*', before '*needs and functionality*'.
3. The need for SE to allow for market driven as well as customer driven development, so SE may be working with "anticipated" rather than "actual" customer needs and functionality.
4. Recognition that SE should be transdisciplinary (Rousseau *et al*, 2018) rather than merely interdisciplinary.

Not surprisingly, many of these drivers for change arise from newer practices such as System of Systems Engineering (SoSE) which operate on systems challenges that do not adhere to the criteria describe above for when traditional SE works well. Indeed, the drivers for change are well aligned with the principles that inform successful SoSE practice (Pratt and Cook, 2017).

## CONTEMPORARY FINDINGS

Defence (2003) reflected the need for considerable early expenditure: '*In our view, complex projects may require that up to 10% to 15% of project funds be spent before approval to proceed to tender.*'

This established wisdom has been reinforced in more recent years by substantial research; a useful collection of which can be found in Cook and Wilson (2015).

Perhaps the most influential work on systems engineering return on investment is by Honour (2013) that reports on research performed over a 15-year period. Figure 3 (over) illustrates his findings on SE effort versus cost overrun from a survey of 43 projects of varying sizes in the USA, Israel, and Australia.

There are three points worth noting from this research:

1. The return on investment can be determined from the slope of the graph. If levels of investment in SE are low, this return is more than ten times whereas in the worst case where there is already substantial investment in SE there is no additional return on investment but also no loss either since this amounts to bringing work forward.
2. Simply spending money on SE is not sufficient; the quality of SE work is crucial and hence Honour has corrected the curve to reflect this, because not surprisingly, experienced specialists can be expected to do a much better job for a given sum than generalists taking on a new challenge such as requirements engineering or architectural design.
3. Project costs relate to the cost up until the acceptance into service of the first production item not to the fleet costs. Hence for something like a single satellite communication system or radar, the SE cost is a much higher proportion of the project cost than for a fleet of ships or vehicles.

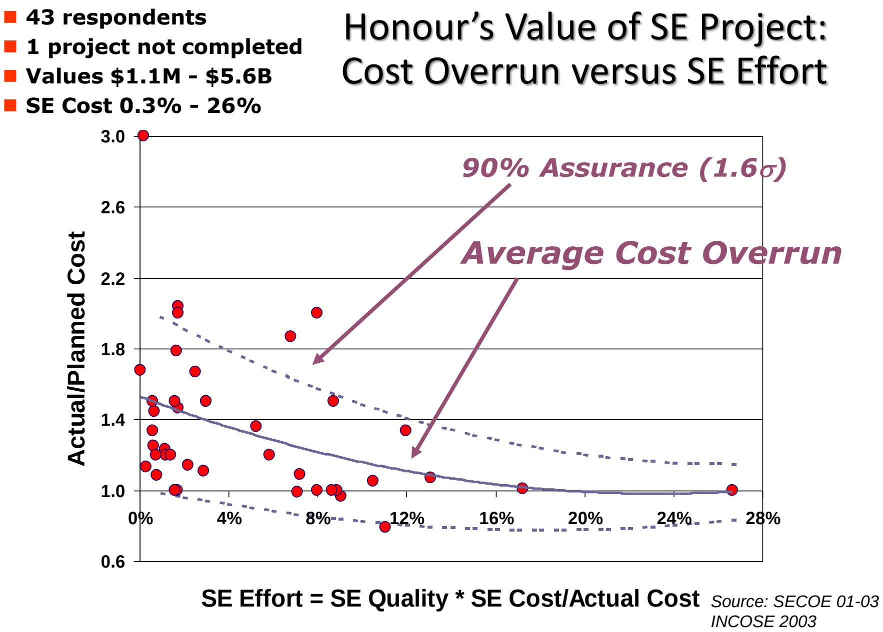


**Figure 3. Correlation of cost overrun versus SE effort (Honour, 2013).**

Systems engineering is practiced by both acquiring organisations and supplying organisations, so it is important to understand where an acquiring organisation should invest its resources in the early lifecycle stages. Honour's work once again provides valuable insights from a second data set of 53 projects collected for this purpose, the findings for which are given in Figure 4 and in Table 3 (over).

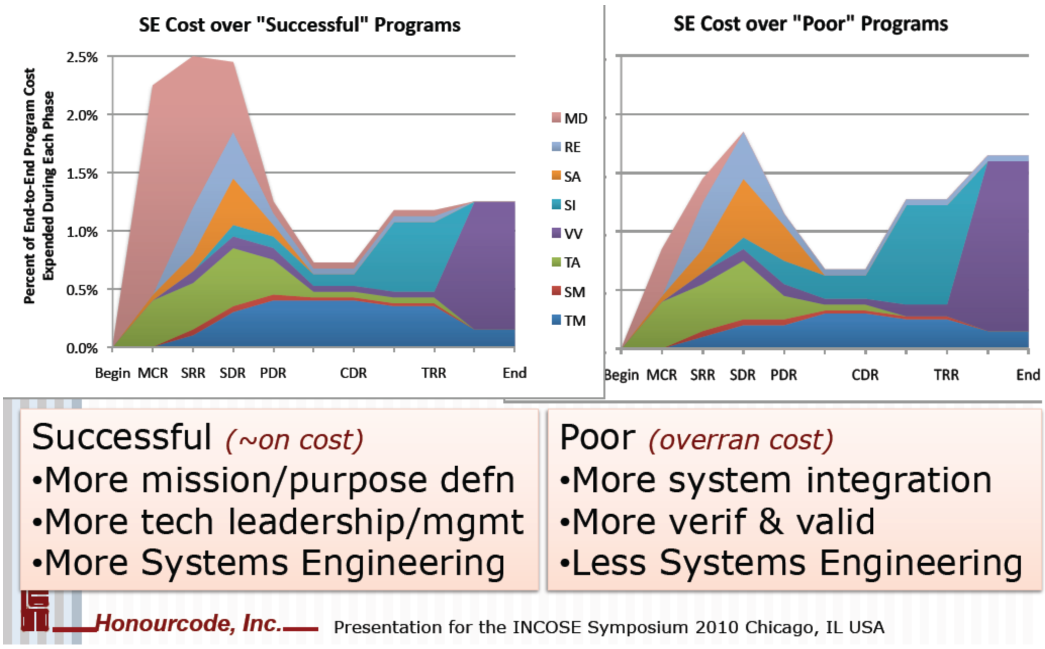


**Figure 4. SE spending profile over time for successful and poor programs (Honour, 2010).**


**Table 3. Optimum spread of SE expenditure (Honour, 2013).**

| SE activity | Optimum spread |
|---|---|
| Mission Definition | 1% |
| Requirements Engineering | 2.5% |
| System Architecting | 2.5% |
| System Implementation | 3% |
| Technical Analysis | 4% |
| Technical Management | 4% |
| Scope Management | 1% |
| Verification and Validation | 7% |
| **TOTAL** | **25%** |

This analysis shows that successful projects spend a much higher proportion of their budget on mission definition (*Why do we need it? How will we use it?*) and requirements engineering than poor projects. The latter spend far more on fixing the subsequent problems with disproportionately large expenditure on extended systems integration. This result is not surprising because it has long been known that the cost to extract project defects increases dramatically as the project progresses as shown in Figure 5 extracted from the *INCOSE Systems Engineering Handbook* that was sourced from a US Defence Acquisition University publication in 1993.

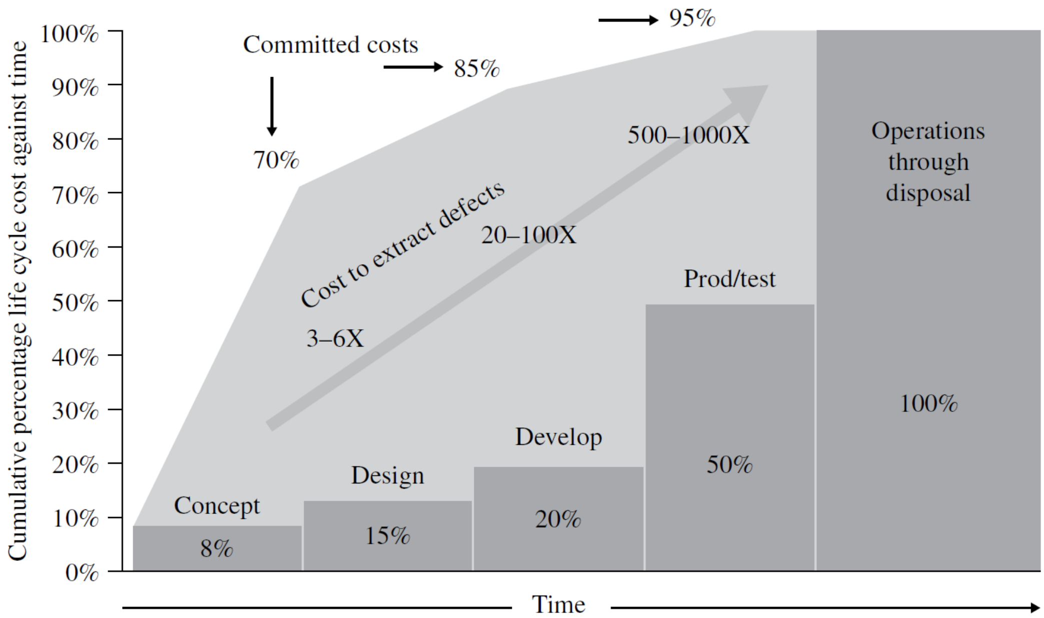


**Figure 5. Committed life cycle cost against time that illustrates increasing cost of extracting defects with time (INCOSE, 2014, from DAU, 1993).**


The current version of the *Defence Acquisition Guidebook* (DAU, 2017) cites recent research from the US Government Accountability Office (GAO, 2016) that reinforces the research cited above:

> *'Systems engineering is the primary means for determining whether and how the challenge posed by a program's requirements can be met with available resources. It is a disciplined learning process that translates capability requirements into specific design features and thus identifies key risks to be resolved. Our prior best practices work has indicated that if detailed systems engineering is done before the start of product development, the program can resolve these risks through trade-offs and additional investments …'*

Other GAO reports provide corroborating advice and explicitly reinforce that the use of model-based approaches to SE is highly correlated with good project outcomes:

> *'Positive acquisition outcomes require the use of a knowledge-based approach to product development that demonstrates high levels of knowledge before significant commitments are made. In essence, knowledge supplants risk over time.' GAO (2012)*

Finally, the GAO (2015) analysed 78 projects to form the view that cost, schedule and performance issues in projects stem not from requirement creep *per se*, because the high-level requirements rarely changed, but from not enough system engineering before acquisition to properly understand the capability needs and translate these into detailed technical requirements.

The various research results emphasising the benefits of early-stage investment in SE have been distilled into practice guidance by, for example, the US National Research Council (2008) which published a comprehensive checklist for early-stage SE activities covering:

- Concept development,
- Key Performance Parameters (KPP),
- Concepts of Operations (CONOPS),
- Cost and schedule scoping,
- Performance assessment,
- Architecture development,
- Risk assessment, and
- Implementation strategy.

Furthermore, the latest iteration of ISO/IEC/IEEE 15288 (2015) introduced the *Business or Mission Analysis Process* that is intended to precede the *Stakeholder Needs and Requirements Definition Process*. ISO/IEC/IEEE 15288 (2015) states that the purpose of the new process is to …

> *'define the business or mission problem or opportunity, characterise the solution space, and determine potential solution class(es) that could address a problem or take advantage of an opportunity.'*

Thus, the ISO standard recognises that SE needs to encompass activities from soft system methodologies to work with stakeholders to achieve an understanding of how to improve a problem situation. In such paradigms, a statement of objectives and a high-level user needs document are the output of systemic practice, not a starting point.

Having now established a strong rationale for early-stage SE, what SE should be conducted and how, the question then turns to what the level investment should be and when should it be scheduled. The work on the design of constructive costs models pioneered by the University of Southern California for software engineering (CoCoMo) and systems engineering (CoSysMo) have started to shed light on these questions. For example, Figure 6 extracted from Boehm *et al* (2000) indicates that the percentage of schedule that should be allocated to initial architecture and risk reduction is a strong function of project size. Similarly, the CoSysMo model has many calibration factors related to such aspects as project complexity and requirement volatility.

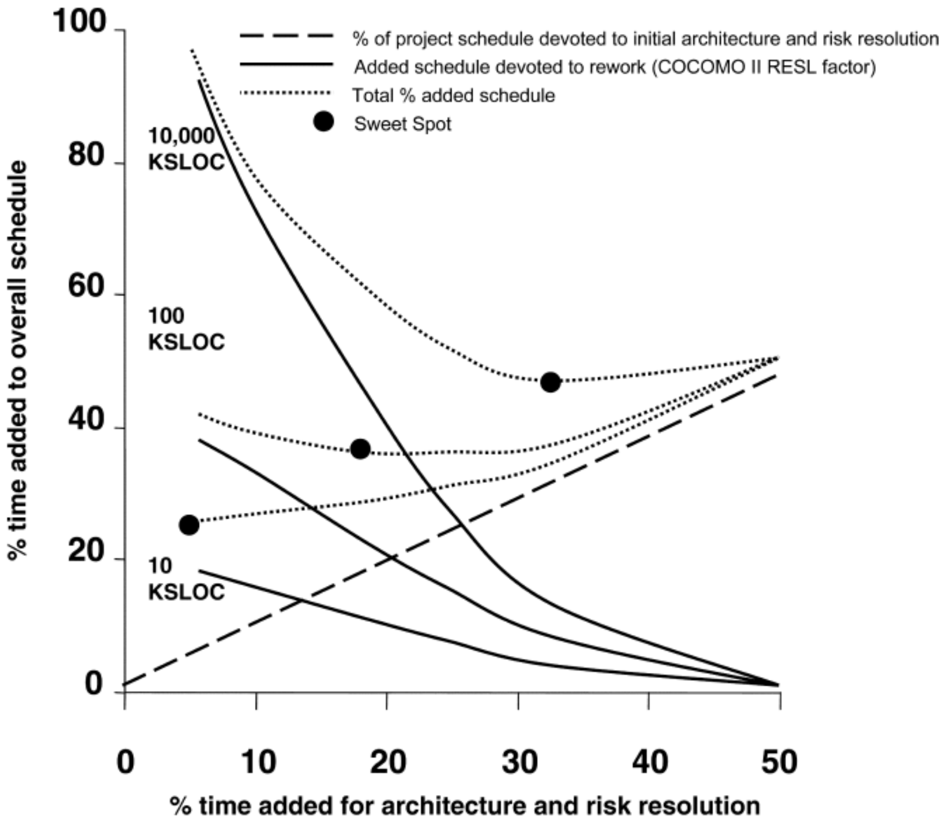


**Figure 6. Schedule sweet spots for early investment versus software size (Boehm et al. 2008).**

## CONSISTENCY OF FINDINGS OVER TIME

Findings from Cook (2000) salient to practice improvement are as follows:

1. Adherence to systems engineering principles and processes across an organisation will save money. Many troubled projects cite reasons for their difficulties that should have been avoidable through the application of systems engineering. Thus, systems and software engineering are a good investment and the organisation should develop an environment whereby their use becomes part of the work culture.
2. Insufficient investment in the early design phases (5% to 15%) is likely to lead to project cost overruns of between 50% and several hundred percent.

3. It is imperative to understand what is needed and to gain user and other stakeholder engagement before embarking on a project and to retain it throughout the life of the project.
4. The user focus must be captured in a clear, traceable, and testable set of requirements with an adequate process to manage inevitable evolution and change.
5. A whole-of-life perspective is important both to create and implement appropriate system engineering methods and to identify and measure the benefits of the approach.
6. System Engineering processes and the application of standards must be tailored to the task in hand.
7. Select suppliers with a sound enterprise environment honed on projects of a similar scale and complexity; failure to heed this advice has been the cause of many project difficulties.
8. Attention to interface definition and management is vital for project success.
9. Planning for system verification and validation (test and evaluation) must be undertaken in the conceptual design phase to ensure that what the users consider to be important to the success of the system is actually evaluated.

A key theme of Cook (2000) was that while SE had its origins in functionalist (or hard systems) approaches that traditionally focus on the provision of engineered systems, even by the year 2000 it had been clear for some time that SE would benefit from incorporating interpretive or soft systems approaches. In traditional SE, the methods of science, engineering and mathematical analysis underpin the problem-solving approach and it is known to work well when the following conditions are met (Cook *et al*, 1998):

1. Systems objectives can be defined at the very beginning of the project.
2. These objectives are shared among stakeholders.
3. All stakeholders can envisage the expected system solution.
4. The process can be summarised as moving the system from an initial state to an end state.
5. The environment (technology, organisational, social policy) is stable.

When some of these conditions are not met the need for supplementary approaches is indicated. Some of these are extrapolations of traditional practice while others are based on the alternative methods of reasoning found in the soft sciences. These can be useful for the following reasons:

1. Their primary interest is in changing organisational culture and gaining commitment from participants to a course of action.
2. They recognise the importance of values, beliefs and philosophies.
3. They use an interpretive approach to tackle systems problems.

Soft systems methodologies do not replace SE methods but can augment them to better achieve certain goals such as improved requirements elicitation, stakeholder support, user acceptance, and management effectiveness. Interested readers are referred to Checkland and Holwell (1998) for a comprehensive coverage of the applicability of these methods and the difference between the hard and soft traditions. Cook (2000) concluded that about half of the difficulties experienced in technical projects could be best be dealt with by the introduction of interpretive, culturally-aware soft systems methodologies.

The more recent evidence completely supports the first three findings from the earlier work of Cook (2000). The GAO reports reflect that SE practice is now becoming increasingly model-based and state that the use of model-based systems engineering approaches is highly correlated with good project outcomes. Of importance is that this finding explicitly applies to the acquisition community, not the supplier community.

The fourth finding by Cook (2000) is also directly reinforced by the GAO (2015) finding that project cost, schedule, and performance issues can be traced back to inadequate SE effort before acquisition to translate the capability needs into detailed technical requirements. It is much more cost-effective to do this work in the early stages than after the project becomes a '*Project of Concern*'.

Contemporary systems engineering thinking in the US Department of Defence is to adopt Digital Engineering where the entire project lifecycle is supported through model-based SE techniques. Together with the model-based UK Systems of Systems (SoS) approach (Coffield, 2016) that seeks to ensure that all projects will deliver capabilities that fully integrate into the evolving UK force, it is clear that whole-of-life perspectives are becoming mainstream.

The sixth point that relates to the need to tailor SE approaches is now a key principle extolled by the new Capability Life Cycle Manual (Defence, 2017) and other acquisition regulations and is well-accepted in SE circles.

The eighth point on interface management is expanded into mainstream approaches such as mission engineering in the US (Moreland and Baron, 2015), the UK SoSA (Coffield, 2016), and the integration and interoperability assessments in the Australian Department of Defence.

The last point on early T&E planning has been entrenched for some time in most SE practices and remains as valid today.

Philosophically, perhaps the most important changes over the years have been the increasing awareness that SE starts well before the specification; it should be involved with identifying and categorising systems issues and proposing ways to address them. The proposed new definition of SE with its concomitant worldview of problem solving and facilitation rather than functionalist execution of processes indicates calls for more early-stage systems engineering. Newer forms such as SoSE embody these concepts and put the emphasis on clearly understanding the purpose of projects and keeping the high-level objectives at front of mind not the detailed process products.

## CONCLUSIONS

Recent research reinforces the findings from the earlier paper that investment in early-stage SE is highly correlated with good project outcomes. In financial terms, it also provides a demonstrable return on investment. There are many non-financial benefits as well such as higher likelihood of delivering a capability that is really needed, better integration of the new capability into the operating environment, improved understanding of the evolution of the capability over time, and superior overall outcomes.

## REFERENCES

ANSI/EIA-632, *Processes for Engineering a System*, Electronic Industry Alliance, 1999.

Blanchard B. S., & Fabrycky W. J., *Systems Engineering and Analysis*, 3rd Ed., Prentice Hall, 1998.

Boehm, B., Valerdi, R. and Honour, E., *The ROI of systems engineering: Some quantitative results for software-intensive systems*, Journal of Systems Engineering, 11(3), 2008, pp.221-234.

Coffield, B., *The System of Systems Approach (SOSA)*, INCOSE INSIGHT, 2016.


Cook S. C., Sydenham P. H., Harris D. D. and Harris M. B., Lecture Notes for *Principles of Systems Engineering*, University of South Australia. December 1998.

Cook S.C, *What the Lessons Learned from Large, Complex, Technical Projects Tell Us about the Art of Systems Engineering*, Proceedings of the INCOSE International Symposium 2000, pp680-687.

Cook S.C. & Wilson S.A., *Determining Return on Investment for MBSE, SE & SoSE*, Proceedings of the Australian Systems Engineering Workshop, Systems Engineering Society of Australia, 2015.

Checkland, P., and Howell S., *Information, Systems and Information Systems*, Wiley, Chichester, ISBN 0-471-95820-4, 1998.

DAU, *Defence Acquisition Guidebook*, https://www.dau.mil, Defense Acquisition University, 2017.

Defence, *Interim Capability Life Cycle Manual*, Commonwealth of Australia, 2017.

DERA, *The MoD's Downey Procedures*, DRA/LS(CS)/SYS_ENG/TG0/CMR/96/1, 1996.

DSMC, *Systems Engineering Management Guide*, Defence Systems Management College, Fort Belvoir, Virginia, USA, 1993.

Jackson M.C., *Critical Systems Thinking and Information Systems Development*, Proceedings of the 8th Australasian Conference on Information Systems, 1997.

Gabb A.P., *System Engineering & Software Engineering Standards and Model*, [On line accessed 10 March 1998], URL: www1.tpgi.com.au/users/agabb/

GAO, *Assessment of selected weapons programs*, GAO-12-400SP, US Government Accountability Office, 2012.

GAO, *Military Service Chiefs' concerns reflect need to better define requirements before programs start*, GAO-15-469, US Government Accountability Office, 2015.

GAO, *Detailed systems engineering prior to product development positions programs for success*, GAO-17-77, US Government Accountability Office, 2016.

Honour, *Systems engineering return on investment*, Proceedings of the INCOSE International Symposium, Chicago, 2010.

Honour, *Systems engineering return on investment*, PhD thesis, University of South Australia, 2013.

ISO/IEC/IEEE 15288, *Systems and software engineering – System life cycle processes*. ISO, IEC and IEEE, 2015.

M'Pherson P. K., "Systems Engineering: an approach to whole system design," *The Radio and Electronic Engineer*, 1980, Vol. 50, No 11/12, p545-558.

MoD, *Strategic Defence Review Study 2F3/7 Smart Procurement*, PPB/P(98)2, UK Ministry of Defence, 1998.

MoD, *The Acquisition Handbook – A Guide to Smart Procurement*, 2nd edition, August 1999. [On line accessed 23 February 2000], URL: www.mod.uk/policy/spi/handbook/front.htm

MoD, *Interoperability management for IPTs,* UK Ministry of Defence, 2008.

Moody J.A., Chapman W.L., Van Voorhees F.D. & Bahill A.T., *Metrics and Case Studies for Evaluating Engineering Designs*, Prentice Hall PTR, Upper Saddle River, NJ, 1997.


Moreland J.D. and Baron N.T., *Integration and Interoperability Mission Engineering* published in *Leading Edge*, January 2015.

NASA, *NASA Systems Engineering Handbook*, SP-6105, June 1995.

NASA, *Space Engineering Lessons Learned*, Engineering Directorate, 1989, [Online accessed 25 Aug 1998], URL: http://home.erols.com/ee/sysengll.htm

National Research Council, *Pre-Milestone A and Early-Phase Systems Engineering: A Retrospective Review and Benefits for Future Air Force Systems Acquisition*. Washington, DC: The National Academies Press, 2008.

Pratt, J. M., & Cook, S. C., *A Principles Framework to Inform Defence SoSE Methodologies* pubished in *Disciplinary Convergence in Systems Engineering Research* (pp. 197-213). Springer, Cham, 2018.

Robertson T.C. (Ed), *Systems Engineering Handbook*, INCOSE, 1998.

Sheard S.A. and Lake J.G., *Systems Engineering Standards and Models Compared*, Proceeding of the INCOSE Annual Symposium, 1998.

The Standish Group, *Chaos*, The Standish Group, 1995, [Online accessed 24 March 1999], URL: www.standishgroup.com/chaos.html

## BIOGRAPHIES

**Stephen Cook** is a Systems Engineering Advisor with Shoal Engineering Pty Ltd where he applies his knowledge to a range of systems engineering management and research challenges. He is also the Professor of Defence Systems at the University of Adelaide where he works in the Entrepreneurship, Commercialisation and Innovation Centre undertaking research and teaching in system of systems engineering and complex project management. Until June 2014 he was the Professor of Systems Engineering at the University of South Australia where he led a number of research concentrations for over 15 years. Preceding this he accumulated 20 years of industrial R&D and SE experience spanning aerospace and defence communications systems. Prof Cook, PhD, is an INCOSE Fellow, a Fellow of Engineers Australia, and a Member of the Omega Alpha Association. He is a Past President of the Systems Engineering Society of Australia.

**Shaun Wilson** is the Chief Executive Officer of systems engineering house, Shoal Engineering Pty Ltd. He is a practising systems engineer with particular expertise in aerospace modelling and simulation and in conceptual design of complex systems. His experience spans from aerospace and defence to mining and leisure sports. Shaun sits on a range of company boards, holds multiple degrees, and is published in several technical fields. He is a Past President of the Systems Engineering Society of Australia.

**ISBN for SETE2018: 978-1-925627-15-2**