Towards a Grand Unified Theory of Systems Engineering

Joseph Kasser
Temasek Defence Systems Institute
National University of Singapore,
Block E1, #05-05
1 Engineering Drive 2
Singapore 117576
tdskj@nus.edu.sg

Yang-Yang Zhao
Norwegian Institute of System Engineering
Buskerud and Vestfold University College
Frogs vei 41,
3611 Kongsberg, Norway
yangyang.zhao@hbv.no

Abstract. The need for research into the nature of systems engineering was articulated in 1994. This paper summarizes some of that subsequent research focusing on the papers published on the unification of systems engineering within the last 20 years and presents a number of frameworks that can unify the various views of systems engineering, explain the overlaps between systems engineering and management, manage complexity and overcome some of the deficiencies in the current systems engineering terminology and graphics.

1. INTRODUCTION

Section 2 covers the need for research into a grand unified theory of systems engineering (GUTSE) (Friedman, 2006), articulated during the closing session of the National Council on Systems Engineering (NCOSE) 1994 symposium and summarizes research into the nature of systems engineering by providing a list of publications. Section 3 demonstrates the use of holistic thinking by using the perspectives perimeter to summarize perceptions of systems engineering from the Holistic Thinking Perspectives (HTP). Section 4 brings together and summarizes the following seven frameworks, six of which have been previously published in per-reviewed journals, which could be used to construct a GUTSE.

1. Holistic Thinking.
2. Hitchins-Kasser-Massie framework (HKMF) framework for understanding systems engineering.
3. Differentiating between Systems Engineering the Role (SETR) and Systems Engineering the Activity (SETA).
4. Pure Systems Engineering, Applied Systems Engineering and Domain Systems Engineering.
5. A Problem Classification Matrix.
6. A Systems Engineering Competency Maturity Model Framework.
7. The Nine-Systems Model.

If the GUTSE is considered as a system, the frameworks are considered as subsystems. Six of these frameworks have been published in peer-reviewed journals. Section 5 summarizes the paper and contains the conclusions.

2. THE NEED FOR A GUTSE

George Friedman called for the development of a GUTSE (Friedman, 2006) echoing (Hill and Warfield, 1972) who wrote "*development of a theory of systems engineering that will be broadly accepted is much to be desired*." Friedman wrote about the stovepiping of systems engineering, and the use of inappropriate graphics and terminology, in, and since 1994 when he had called for research into the nature of systems engineering. There is an additional need for a GUTSE because since the early 1990's, systems engineers have had a problem, not only explaining what they do, to other people but also defining it amongst themselves.

The baseline for this research into the nature of systems engineering began at the 1994 symposium of the NCOSE, where presenter after presenter opened their presentation with a definition of systems engineering and each definition was different. However, when each presenter continued by describing the functions performed by systems engineers, they talked about planning, organizing, directing and controlling; the traditional functions of management (Fayol, 1949) page 8). When asked what systems engineers did, their answers were also different. These observations in 1994 triggered a research program into the nature of systems engineering and its overlap with project management and problem-solving. The initial research showed that there seemed to be no unique body of knowledge to systems engineering and that all of the activities performed by systems engineers, apart from possibly requirements and interface management, were also performed by other types of engineers (Kasser, 1996). The paper concluded with "*systems engineering is a discipline created to compensate for the lack of strategic technical knowledge and experience by middle and project managers in organizations functioning according to Taylor's "Principles of Scientific Management*". Subsequent research into the nature of systems engineering included a literature review of text books published between 1959 and 2011 starting with (Goode and Machol, 1959) as well as the proceedings of all the international symposia of the INCOSE since 1991. Further research into the nature of systems engineering, systems engineering tools, process improvement, the overlap between systems engineering and project management and problem-solving and systems engineering's attempts to manage complexity was documented in (Kasser and Hitchins, 2012; Kasser, 2012b; Kasser and Hitchins, 2011; 2010; Kasser and Zhao, 2014; Kasser, 2014; Kasser, et al., 2014; Kasser and Hitchins, 2013; Kasser, 2010d; 2007a; 2013b; 2000a; c; b; Kasser and Shoshany, 2000; Cook, et al., 2001; Kasser, 2001; Kasser and Shoshany, 2001; Kasser, 2002c; b; f; d; a; g; e; Kasser and Cook, 2002; Kasser, et al., 2002; Cook, et al., 2003; Kasser, 2003; Kasser and Cook, 2003; Kasser, et al., 2003; Kasser, 2004b; a; c; 2005a; b; c; Kasser and Mirchandani, 2005; Kasser and Palmer, 2005; Kasser, et al., 2005a; Tran and Kasser, 2005; Hari, et al., 2006; Kasser, 2006; Kasser, et al., 2006; Scott, et al., 2006; Kasser, 2007d; b; Kasser, et al., 2007; Kasser, 2009a; b; Kasser and Hitchins, 2009; Kasser, et al., 2009; Kasser, 2010b; a; c; Kasser and Frank, 2010; Kasser, et al., 2010; Kasser, 2011b; a).

3. PERCEPTIONS OF SYSTEMS ENGINEERING

Systems engineering has been defined as "*the science of designing complex systems in their totality to ensure that the component subsystems making up the system are designed, fitted together, checked and operated in the most efficient way*" (Jenkins, 1969). However, in the ensuring 45 years, instead of performing systems engineering as defined by Jenkins, systems engineers seem to have been busy creating more and more complex models and processes. This observation can be mapped into the holistic problem-solving process shown in Figure 5 as the undesirable situation, where:

- ***The undesirable situation*** is the failure of systems engineering to manage the complexity of the systems development environment.
- ***The future desirable situation*** is systems engineering successfully managing the complexity of the systems development environment.
- ***The solution*** is a theory of how to manage complexity and a set of tools for managing complexity based on the theory.
- ***The problems are:***
    1. To understand the reasons why systems engineering has not been able to manage the complexity of the systems development environment in a repeatable manner.
    2. How to develop a theory for managing complexity and the tools for managing complexity based on the theory.

When faced with a problem it is always useful to find out if anyone has faced the same or a similar problem and understand their approach to remedy their problem. Mendeleev, when faced with the problem of making sense of the relationships between chemical elements and their properties, sorted the elements into a table. His contribution was to create a framework, the Periodic Table of Elements, and populate it with the known elements, leaving gaps which represented unknown elements. Using a similar approach, the perceptions of systems engineering from the perspectives perimeter identified in the research were sorted and grouped in the HTPs. A summary of these perceptions include:

3.1. **Big Picture perspective**

Perceptions from the *Big Picture* perspective included:

- Successes and failures.
- Inadequate systems engineering is repeatedly cited as a major contributor to failed projects particularly in the National Aeronautical and Space Administration and the United States of America Department of Defense (DOD) (Evans, 1989; Leveson, 2004; Welby, 2010; Wynne and Schaeffer, 2005).
- Systems engineering:
  - Is practiced in many domains.
  - Is practiced in different phases of the system lifecycle.
  - Overlaps with other disciplines.
  - Interfaces to other disciplines.
- What academia was teaching.
- Systems engineering seemed to be poorly taught in general.

3.2. **Operational perspective**

Perceptions from the *Operational* perspective included:

- What systems engineers produce.
- What systems engineers do; the role of the systems engineer.
- The roles of the systems engineer in 1969 (Jenkins, 1969).
- The roles of the systems engineer in 1996 (Sheard, 1996).
- There was a focus on the systems engineering process.
- "*A single process, standardizing the scope, purpose and a set of development actions, has been traditionally associated with systems engineering*" (Arnold, 2000).
- Systems engineering is performed in the context of three streams of work between milestones (Kasser, 1995). Hence the three streams contain systems engineering and non-systems engineering activities.
- Systems engineers perform systems engineering in various scenarios which included:
  - Conceptual design.
  - Requirements management.
  - Architecting.
  - Interface management.
  - Testing.
  - Integrating.
  - Verification and validation.
  - Engineering management.

3.3. **Functional Perspective**

Perceptions from the *Functional* perspective identified the following functions performed in the operational scenarios:

- Problem-solving.
- Systems thinking.
- Holistic thinking, going beyond systems thinking.
- Analysis.
- Synthesis.
- Decision-making.
- Communicating.

3.4. **The Structural Perspective**

Perceptions from the *Structural* perspective included:

- The many definitions of systems engineering; at least forty.
- Systems engineers were people who used tools.
- The tools of systems engineering in the 1960s (Alexander and Bailey, 1962; Wilson, 1965; Chestnut, 1965).
- The tools of systems engineering in 2005 (Jenkins, 2005).
- Systems engineering met Kline's requirements for a discipline (Kline, 1995).

3.5. **The Generic Perspective**

Perceptions from the *Generic* perspective included:

- The similarity between systems engineering and mathematics; two enabling disciplines providing tools used to solve problems in other disciplines.
- Systems engineering was demonstrating the symptoms of a discipline in its early stages. For example, the state of:
    - Electrical engineering before the formulation of Ohms' law in 1827 and Maxwell's equations in 1873.
    - Chemistry before Mendeleev framed the Periodic Table of Elements in 1869.
- The focus on process improvement is not unique to systems engineering For example, Peter Drucker wrote "*Throughout management science--in the literature as well as in the work in progress--the emphasis is on techniques rather than principles, on mechanics rather than decisions, on tools rather than on results, and, above all, on efficiency of the part rather than on performance of the whole*" (Drucker, 1973) page 509).

3.6. **The Continuum Perspective**

Perceptions from the *Continuum* perspective included:

- Dichotomies in the literature on systems of systems.
- Dichotomies in the literature on complexity.
- The Distinction between subjective and objective complexity.
- Two types of [real] complexity; real-world and artificial (Kasser and Palmer, 2005).
- The Seven camps in systems engineering (Kasser and Hitchins, 2012).
- While there was some commonality in the text books on systems engineering each book had a different focus and contained unique material.
- There was no single systems engineering process. All the process descriptions were different.
- The systems engineering process overlapped the problem-solving process.
- The two paradigms of systems engineering, the 'A' paradigm and the 'B' paradigm (Kasser, 2012b).
- Differentiating between Systems Engineering - the Role (SETR) and Systems Engineering - the Activity (SETA) (Kasser and Hitchins, 2013).

3.7. **The Quantitative Perspective**

Perceptions from the *Quantitative* perspective included:

- A literature review reveals that many of the works on improving systems engineering have focused on improving and developing new systems engineering processes, and tend to ignore people, see (Swarz and DeRosa, 2006; Goldberg and Assaraf, 2010) for typical examples.
- The literature on excellence focused on people, not process (Rodgers, et al., 1993; Harrington, 1995; Peters and Austin, 1985; Peters and Waterman, 1982).
- The top ten reasons for project success and failure cited in the oft quoted Standish report made no mention of process (CHAOS, 1995).

- There were five distinct types of systems engineers (Kasser, et al., 2009). These observations were matched to factors conducive to innovation in the innovation domain literature (Gordon G. et al., 1974).
- The relative number of systems engineers and other disciplines needed in the different phases of the system lifecycle (JAXA, 2007).
- While requirements were considered critical to systems engineering, there was no metric for the goodness of a requirement.

3.8. **The Temporal Perspective**

Perceptions from the *Temporal* perspective included:

- The history and origin of systems engineering.
- The Standards used in systems engineering ignore the early stage systems engineering activities.
- The degree of micromanagement in the Standards increased exponentially over time from the AFCM 365-5 in 1967 (Gelbwaks, 1967) to the DOD Architecture Framework (DODAF) in 2004 (DoDAF, 2004).
- The evolution of systems engineering from a process that began with the development of a common vision of the solution, to a product that began with the elicitation and elucidation of requirements.
- DOD 5000.1 required the use of systems engineering.
- DOD 5000.2 emphasized the use of systems engineering
- DOD 5000.2-R assigned the early stage systems engineering activities to Cost as an Independent Variable (CAIV) to be performed by Integrated Product and Process Development (IPPD) (DOD, 1998; DOD 5000.2-R, 2002).
- Systems engineering evolved into two paradigms, the ‘A’ paradigm and the ‘B’ paradigm (Kasser, 2012b). The ‘A’ paradigm is the original systems engineering of the 1960s starting with the early stage activities and the B paradigm follows the DOD approach of starting with requirements elicitation and elucidation.
- Three attempts to manage complexity in the INCOSE literature (Martin, 2004; Adcock, 2005; Paul and Owunwanne, 2006).

3.9. **The Scientific Perspective**

Insights from the *Scientific* perspective included:

- The different camps of systems engineering were views of systems engineering from a single perspective.
- The distinction between the SETR and SETA paradigms clears up the overlaps and conflicts in the literature between project management, systems engineering and Engineering, and provides a view of the relationships between the activities performed to realize a system (Kasser and Hitchins, 2013).
- The development of the Hitchins-Kasser-Massie Framework (HKMF) for understanding systems engineering shown in Figure 2 (Kasser and Massie, 2001; Kasser, 2007c; d) and summarized in Section 4.2[1].
- The different roles performed by systems engineers and the different types of products developed in different phases of the system lifecycle identified from the HKMF are two reasons why systems engineers, using different single viewpoints, could not agree on the nature of systems engineering. This is a similar situation to the fable of the blind men studying the elephant and drawing different conclusions.
- Systems engineering could be partitioned into pure and applied systems engineering in a similar manner to partitioning of mathematics into pure and applied maths where:

---

[1] The HKMF is summarized in Section 4.2 to keep the seven frameworks together in Section 4.

  - Applied systems engineering covers the activities performed in the operational scenarios in which systems engineering is performed (Section 3.2).
  - Pure systems engineering cover the functions performed in the activities (Section 3.3).

- Understanding why there were different versions of the systems engineering process. Since "*the systems engineer creates a unique process for his or her particular development effort*" (Biemer and Sage, 2009), consider each published version of the process as the unique process created for their particular development effort by someone or some group at some point in time, at some point in the system lifecycle.
- The 'B' paradigm is inherently flawed. Recognition that the 'B' requirements paradigm is inherently flawed is not a new observation. For example:

  - Sutcliffe et al. proposed reducing human error in producing requirements by analysing requirements using an approach of creating scenarios as threads of behaviour through a *use case*, and adopting an object-oriented approach; namely they proposed a return to the 'A' paradigm (Sutcliffe, et al., 1999).
  - Daniels et al. point out that standalone requirements make it difficult for people to understand the context and dependencies among the requirements, especially for large systems and suggest using use cases to define scenarios (Daniels, et al., 2005).
  - One of the two underlying concepts of Model Based Systems Engineering (MBSE) is to develop a model of the system to allow various stakeholders to gain a better understanding of how well the conceptual system being modelled could remedy the problem, before starting to write the requirements. MBSE with its roots in the process camp of systems engineering and the 'B' paradigm has rediscovered the CONOPS and is trying to return to the 'A' paradigm.
- The confusion between the "systems engineering process" and the problem solving process can be resolved by recognizing:

  1. IEEE 1220 stated that "*the systems engineering process is a generic problem-solving process*" (IEEE 1220, 1998) Section 4.1). IEEE 1220's replacement of the term "the problem-solving process" by the term "the systems engineering process" seems to have led to today's focus on process; specifically the [solution] system development process. Had the standard instead stated that 'systems engineers apply the generic problem-solving process' the focus of systems engineering might have remained as the original focus on managing complex problems (Jenkins, 1969).
  2. When the problem is:

     - ***Small or non-complex*** the sequence of activities in the remedial action is known as the "problem-solving process".
     - ***Large or complex***, the sequence of activities in the remedial action is known as the "systems engineering process" instead of the system development process (SDP).

4. FRAMEWORKS FOR A GUTSE

Making sense of systems engineering is a complex problem, and once the information perceived from the Holistic Thinking Perspectives (HTP) was examined, inferences from the *Scientific* perspective suggested that the following seven interdependent frameworks together with a number of to be determined others, could eventually lead to a GUTSE:

1. Holistic Thinking.
2. The HKMF.
3. Differentiating between SETR and SETA.
4. Pure systems engineering, applied systems engineering and domain systems engineering.
5. A Problem Classification Matrix.
6. A Systems Engineering Competency Maturity Model Framework.
7. The Nine-Systems Model.

The reasons for this assertion include:

1. Holistic thinking is a very powerful thinking tool for dealing with complex problems.
2. The HKMF can be used to clarify the role of the systems engineer and their activities in the three streams of work.
3. The SETR-SETA paradigm shows that while the role of the systems engineer and the role of the project manager may overlap when viewing systems engineers and project managers in different organizations, the activities known as systems engineering and project management do not overlap.
4. The Nine-System Model, being based on the holistic problem-solving process, manages complexity more systemically and systematically than the three previous approaches found in the INCOSE symposium literature (Martin, 2004; Adcock, 2005; Paul and Owunwanne, 2006).
5. The combination of the frameworks provide a way to state the value of systems engineering as being "a part of the application of a systemic and systematic holistic approach to remedying complex problems".

Each of these frameworks except for "Pure Systems Engineering, Applied Systems Engineering and Domain Systems Engineering" has been previously published in peer-reviewed journals and consequently is only summarized in this section.

4.1. **Holistic Thinking**

Holistic thinking (Kasser, 2013a) goes beyond systems thinking by not only thinking about a system as a whole but also by doing the thinking in a systemic and systematic manner. It does this by perceiving issues from the perspectives perimeter using the HTP coupled with active brainstorming (Kasser, 2009a) to think in a systemic and systematic manner about a system (ideation)[2], coupled with critical thinking (ideation and idea evaluation) as shown in Figure 1. The elements of holistic thinking include:

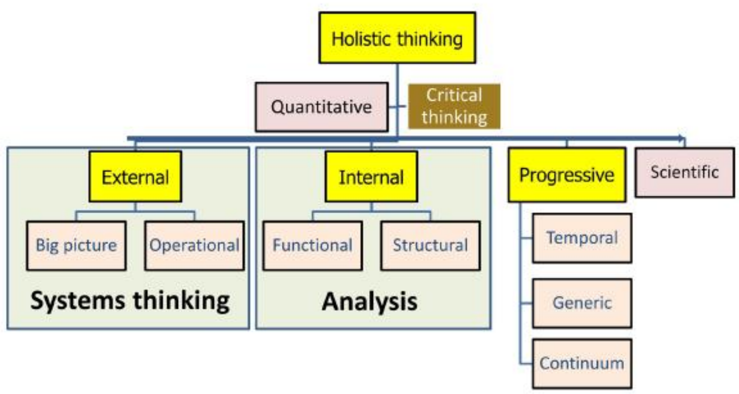

Figure 1 Holistic Thinking (Structural perspective)

- Viewing an issue/system/problem from each of the HTPs.
- Documenting perceptions of an issue/system/problem in the HTPs in the manner of Section 3. This is a powerful way of organizing information when viewing a messy situation.
- Triggering ideas about remedying undesirable situations, and identifying problems using active brainstorming (Kasser, 2009a).

4.2. **The HKMF for understanding systems engineering**

Two dimensions of the HKMF for understanding systems engineering (Kasser, 2007c; d; Kasser and Massie, 2001) plot the product complexity and process (lifecycle) phase on different axes as discussed herein.

---

[2] Using the internal, external, progressive, Quantitative and Scientific perspectives from nine different viewpoints on the perspectives perimeter.

4.2.1. **The vertical dimension**

The vertical or product dimension is based on the work of (Hitchins, 2000) who proposed the following five-layer model for systems engineering:

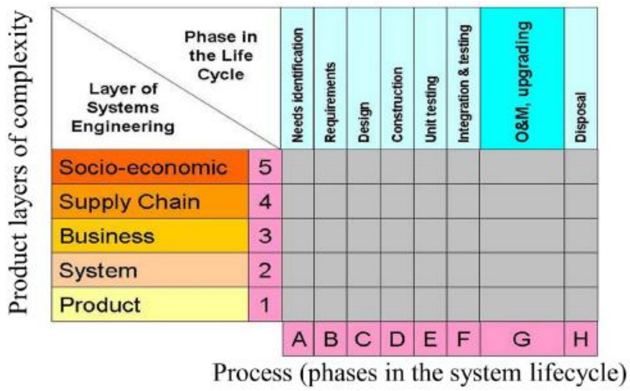

Figure 2 The HKMF

- **Layer 5 -** Socioeconomic, the stuff of regulation and government control.
- **Layer 4 -** Industrial Systems Engineering, or engineering of complete supply chains/circles. Many industries make a socio-economic system. A global wealth creation philosophy. Japan seems to operate most effectively at this level.
- **Layer 3 -** Business Systems Engineering - many businesses make an industry. At this level, systems engineering seeks to optimize performance somewhat independent of other businesses
- **Layer 2-** Project or System Level. Many projects make a Business. Western engineer-managers operate at this level, principally making complex artefacts.
- **Layer 1-** Product Level. Many products make a system. The tangible artefact level. Many [systems] engineers and their institutions consider this to be the only "real" systems engineering.

4.2.2. **The horizontal dimension**

The horizontal or process dimension covers the phases of the systems life cycle. The phases have been stated in various ways in various standards, conference papers and books, but for this framework they are defined in generic terms as:

1. Identifying the need.
2. Requirements analysis.
3. Design of the system.
4. Construction of the system.
5. Testing of the system components.
6. Integration and testing of the system.
7. Operations, maintenance and upgrading the system.
8. Disposal of the system.

The resulting two-dimensional framework is shown in Figure 2.

4.2.3. **Initial perceptions from the HKMF**

In its early days, the HKMF has (Kasser, 2007c):

- Provided an understanding of why systems engineers can't agree on their roles and activities (Kasser and Palmer, 2005).
- Provided an understanding of the reasons for the overlap between systems engineering and management from the origin of the situation discussed by (Johnson, 1997).
- Shown a relationship between Enterprise Systems Engineering and the Engineering of Systems.

4.3. **Differentiating between SETR and SETA**

Systems engineering means different things to different people. This section summarizes the differences between SETR - the role or job description of the systems engineer and SETA - a set of activities known as systems engineering (Kasser and Hitchins, 2009; Kasser, et al., 2009):

4.3.1. **SETR**

SETR is a subjective definition from the *Operational* perspective of systems engineering. It can be a:

- Job title such as network systems engineering, control system engineering, communications systems engineering, etc. In many instances the type of system is dropped from the title. The on-the-job activities performed in such a role include, systems engineering, design, engineering, project management, testing, etc. SETR is performed in many domains, generally associated with technology and is often process-centric.
- Philosophy and a way of life (Hitchins, 1998) which Kasser interpreted as the application of holistic thinking to problem-solving (Kasser, 2013a).

4.3.2. **SETA**

SETA is an objective definition of an activity based on the following criterion (Kasser and Hitchins, 2009; Kasser, et al., 2009):

- If the activity *deals with parts and their interactions as a whole*, then it is an activity within the set of activities to be known as SETA.
- If the activity *deals with a part in isolation*, then the activity is not an activity within the set of activities to be known as SETA but is part of another set of activities ('something else'), e.g., Engineering, project management, software engineering, etc.

SETA is a return to Hall's definition of "*systems engineering as a function not what a group does*" (Hall, 1962) page 11).

The Relationships between the Activities

The relationships between the systems engineering, project management and engineering activities performed to realize a solution system are shown in Figure 3 (Kasser and Hitchins, 2013) which is based on the holistic problem-solving process. An entity associated with the undesirable situation kicks off the SDP which consists of a set of activities performed in series and in parallel (a process) which produces a solution system which is designed to remedy the undesirable situation. The activities in the SDP can be divided into project or engineering management, systems engineering and engineering where:

- ***Domain knowledge*** is the underpinning information used by holistic thinking in the performance of the activities performed in a SDP which require knowledge of the problem, solution and implementation domains.
- ***Holistic thinking*** is the use of the thinking tools that use the domain knowledge to identify and remedy problems in undesirable situations in the activities known as project management, engineering and risk management.
- ***Risk management*** is the set of activities that anticipate, prevent and mitigate risks in the problem, solution and implementation (product and process) domains used in the activities known as project management, systems engineering and engineering.
- ***Project management*** is the set of activities known as planning, organizing, directing and controlling (Fayol, 1949) page 8). Project management incorporates risk management to manage process risks. Some of these activities are also currently known as systems engineering management.
- ***Systems engineering*** (SETA) is the set of activities defined in Section 4.3.2 and incorporates risk management to manage system level product risks when designing and integrating the overall system/interacting subsystems.
- ***Engineering*** is the set of non-SETA engineering activities that create subsystems incorporating risk management to manage product risks.

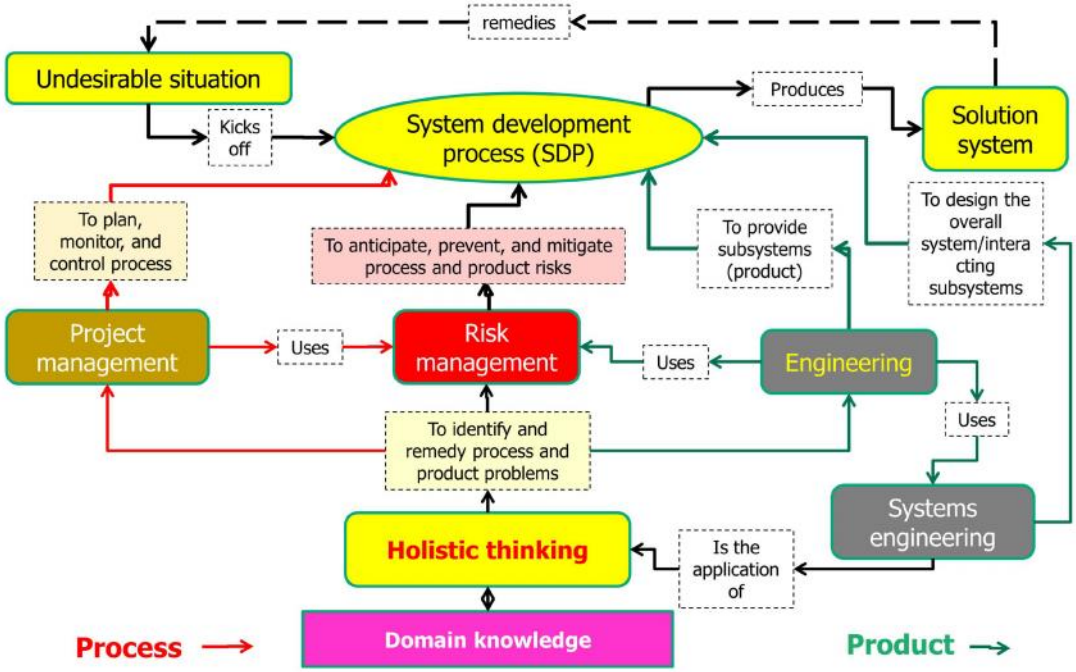

Figure 3. The relationships between the activities in the SDP in the SETA paradigm

The concept map shown in Figure 3 shows a non-overlapping relationship between systems engineering, project management and engineering. In Figure 3 engineering activities may 'provide' by creating, by building, by purchasing commercial-off-the-shelf (COTS) products, by changing a process, by reorganizing a human activity system or by a combination of all or some of the above.

4.3.3. **Mapping SETR and SETA**

As discussed in Section 4.3.1, the role or job of the system engineer, engineer and project manager is to perform an appropriate mixture of the activities known as systems engineering, engineering and project management as well as any other pertinent activities to the project. Each person's role in SETR will be different because the mixture of activities depends on the organizational situation and is different. This is why, project management, systems engineering and engineering have been perceived as being overlapping.

Due to the various ways in which SETA and non-SETA have been allocated to personnel[3] performing SETR and non-SETR, in any specific organisation at any specific time, roles and activities do not overlap 100%. Thus a person with the role or job title of systems engineer will perform a number of activities that include systems engineering, project or engineering management, and engineering. And an engineer might perform a mixture of engineering and systems engineering.

4.4. **Pure systems engineering, applied systems engineering and domain systems engineering**

This framework was created from the need to sort information discovered during a benchmarking study of the stated content[4] of the required courses in 11 systems engineering Master's degrees in 2013 (Kasser and Arnold, 2014). Before creating the framework it was difficult to see any patterns in the data. Once sorted by the three types of systems engineering it was easy to infer information in-

[3] The word 'personnel' is used to avoid the semantically loaded terms engineers, systems engineers, project manager, etc.

[4] Recognizing that what was actually taught may not be what was stated on the institution's' Web site.

cluding:

- Different courses had different mixtures of pure systems engineering, applied systems engineering and domain systems engineering.
- Textbooks on systems engineering focused on different aspects of pure systems engineering, applied systems engineering and domain systems engineering.
- The problem of creating a Systems Engineering Body of Knowledge (SEBOK) was simplified. The knowledge could be structured as pure and applied. The domain knowledge was not necessary, it being the province of the domain, so the problem of providing domain knowledge for SETR in the SEBOK was dissolved.

4.5. **A Problem Classification Matrix**

A major difficulty in managing complexity has been the formulation of the problem. This framework (Kasser, 2012a) is based on distinguishing between subjective and objective complexity and classifying problems by:

- Level of difficulty of the problem.
- Structure of the problem.
- Complexity of the problem.

4.5.1. **Level of difficulty of the problem**

Classifying problems by level of difficulty is difficult in itself because difficulty is subjective since one person's easy problem may be another person's medium, ugly or hard problem. Ford introduced four categories of increasing order of difficulty for [well-structured] mathematics and science problems: easy, medium, ugly, and hard (Ford, 2010). These categories may be generalized and defined as follows:

- ***Easy problems*** are problems that can be solved in a short time with very little thought.
- ***Medium problems*** can be solved after some thought, may take a few more steps to solve than an easy problem and can probably be solved without too much difficulty, perhaps after some practice.
- ***Ugly problems*** are ones that will take a while to solve. Solving them involves a lot of thought, many steps and may require the use of several different concepts.
- ***Hard problems*** usually involve dealing with one or more unknowns. Solving them involves a lot of thought and some research and may also require iteration through the problem solving process as learning takes place (knowledge that was previously unknown becomes known).

4.5.2. **Structure of the problem**

Perceived from the Continuum perspective, problems lie on a continuum which ranges from 'well-structured' to 'ill-structured', where:

- ***Well-structured problems*** are problems where the existing undesirable situation and the Feasible Conceptual Future Desirable situation (FCFDS) are clearly identified. These problems may have a single solution or sometimes more than one acceptable solution. Well-structured problems with single solutions tend to be posed as closed questions, while well-structured problems with multiple solutions tend to be posed as open questions.
- ***Ill-structured problems***, sometimes called 'ill-defined' problems or 'messy'[5] problems are problems where either or both the existing undesirable situation and the FCFDS are unclear (Jonassen, 1997).
- ***Wicked problems*** are extremely ill-structured problems[6] first stated in the context of social policy planning (Rittel and Webber, 1973).

---

[5] When complex.

[6] Technically there is no problem since while the stakeholders may agree that the situation is undesirable, they cannot agree on the problem.

4.5.3. **Complexity of the problem**

The complexity the problem is determined by the number of issues, functions, or variables involved in the problem; the degree of connectivity among those variables; the type of functional relationships among those properties; and the stability among the properties of the problem over time.

4.5.4. **The Problem Classification Matrix**

The structure of the problem and the level of difficulty of problems are combined in the two-dimensional problem classification matrix shown in Figure 4. The two dimensions are:

- ***Problem structure*** which is objective and ranges from non-complex through complex well-structured problems to complex ill-structured problems and wicked problems.
- ***Level of difficulty*** which is subjective and ranges from easy to hard; where simple problems can be easy and medium while complicated problems are those that are ugly and hard.

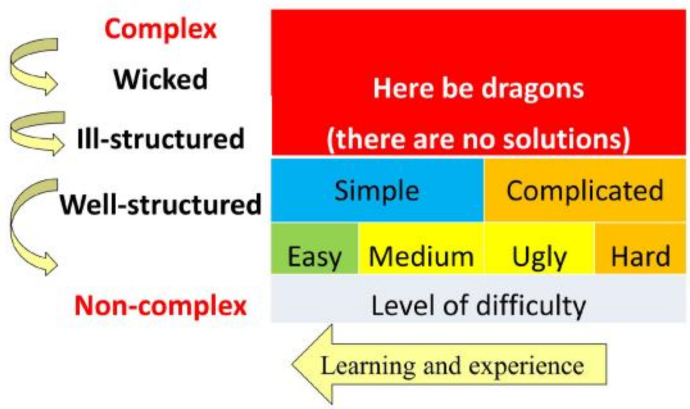

Figure 4 Problem classification matrix

Different people may position the same problem in different places in the matrix. This is because as knowledge is gained from research, education and experience a person can reclassify the subjective difficulty of a problem down the continuum from 'hard' towards 'easy'. There are no solutions to ill-structured and wicked problems; they must be converted to well-structured problems before the main problem solving process can begin.

4.6. **A Systems Engineering Competency Maturity Model Framework**

Current approaches for constructing and using competency models are based on observations of what systems engineers do in organisations but there is no way to directly compare competency models or verify if indeed they are fit for purpose.

The Systems Engineering Competency Maturity Model Framework (CMMF) (Kasser, et al., 2013) shown in Table 1 combines Holistic Thinking, the HKMF, differentiating between SETA and SETR, the pure, applied and domain structure of systems engineering, and the Five Types of systems engineers to create a framework that can be used to assess existing competency models and can also be used as a framework on its own. The framework is used to assess competencies in three categories of knowledge:

1. Applied and domain systems engineering.
2. Pure systems engineering (cognitive skills).
3. Individual traits.

Knowledge of the HTPs is assessed as declarative, procedural and conditional (Woolfolk, 1998). Assessment of knowledge, cognitive skills and individual traits is made in ways already practiced in the psychology domain and do not need to reinvented by systems engineers. Where knowledge is required at the conditional level, it includes procedural and declarative. Similarly, where knowledge is required at the procedural level, it includes declarative knowledge.

Table 1 A Systems Engineering Competency Maturity Model Framework

	Type I	Type II	Type III	Type IV	Type V
Category 1:	Knowledge areas				
Systems engineering	Declarative	Procedural	Conditional	Conditional	Conditional
Problem domain	Declarative	Declarative	Conditional	Conditional	Conditional
Solution domain	Declarative	Declarative	Conditional	Conditional	Conditional
Implementation domain	Declarative	Declarative	Conditional	Conditional	Conditional
Category 2:	Cognitive characteristics (Holistic Thinking)				
Systems Thinking					
Descriptive (8)	Declarative	Procedural	Conditional	Conditional	Conditional
Prescriptive (1)	No	No	Procedural	No	Conditional
Critical Thinking	Confused fact finder	Perpetual analyser	Pragmatic performer	Pragmatic performer	Strategic re-visioner
Category 3:	Individual traits (sample)				
Communications	Needed	Needed	Needed	Needed	Needed
Management	Not needed	Needed	Needed	Needed	Needed
Leadership	Not needed	Not needed	Needed	Needed	Needed
Others (specific to situation)	Organization specific	Organization specific	Organization specific	Organization specific	Organization specific

A literature review showed that the problem of assessing the degree of critical thinking in students seemed to have already been solved (Eichhorn, 2002; Wolcott and Gray, 2003; Allen, 2004; Paul and Elder, 2006). Wolcott and Gray aggregated lists of critical thinking abilities by defining five levels of critical thinking (Wolcott and Gray, 2003). In addition, Wolcott's method for assessing a critical thinking level was very similar to that used by Biggs for assessing deep learning in the education domain (Biggs, 1999). Since a tailored version of the Biggs criteria had been used successfully at the University of South Australia for assessing student's work in postgraduate classes on systems engineering (Kasser, et al., 2005b), Wolcott's method was adopted for the CMMF. Wolcott's five levels (from lowest to highest) are:

4. Confused fact finder.
5. Biased jumper.
6. Perpetual analyzer.
7. Pragmatic performer.
8. Strategic re-visioner.

The horizontal dimension provides a way to assess the competence of a person in each broad area of the vertical dimension against the levels of increasing ability.

Competencies can be developed for specific roles using workflow analysis of the activities performed in the appropriate area of the HKMF.

4.7. **The Nine-System Model**

The Nine-System Model (Kasser and Zhao, 2014):

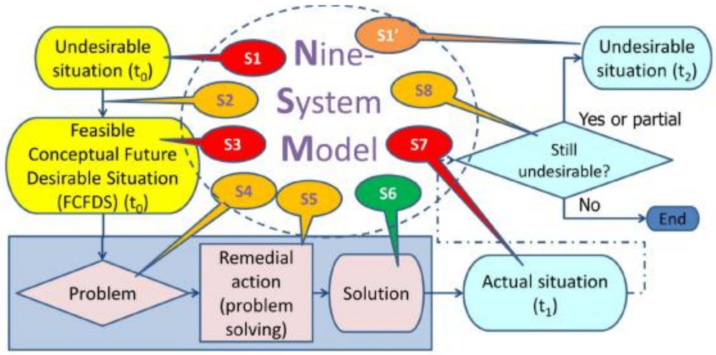

Figure 5 Mapping the nine systems to the holistic problem solving process

- Is based on the problem-solving approach to systems engineering in accordance with IEEE 1220 which stated that "*the systems engineering process is a generic problem-solving process*" (IEEE 1220, 1998) Section 4.1).
- Maps into the holistic problem-solving process as shown in Figure 5.
- Manages complexity by abstracting out all information about the System of Interest (SOI) that is not pertinent to the issue at hand.
- Is an application of the theory that complexity can be managed (but not reduced) by applying a set of rules for grouping/aggregation/synthesis.
- Is a self-similar framework model usable in any level of the hierarchy.
- Encompasses aspects of the Seven Samurai (Martin, 2004), BPR, Checkland's Soft Systems Methodology (Checkland and Scholes, 1990), Hitchins' approach to systems engineering (Hitchins, 2007) and the SIMILAR process (Bahill and Gissing, 1998)
- Incorporates much of the content of the MIL-STD-499 (MIL-STD-499A, 1974), EIA 632 (EIA 632, 1994) and IEEE 1220 (IEEE 1220, 1998) Standards.
- Incorporates the seven principles for systems engineered solution systems (Kasser and Hitchins, 2011).
- Provides a template incorporating built-in best practices that conform to the 'A' paradigm of systems engineering (Kasser, 2012b).
- Is a conceptual model since as the *Temporal* perspective shows, all the systems do not coexist at the same time.
- Comprises the following situations, processes and socio-technical systems in a clearly defined interdependent manner:

  S1. The undesirable or problematic ***situation.***
  S2. The ***process*** to create the FCFDS.
  S3. The FCFDS that remedies the undesirable ***situation***.
  S4. The ***process*** to plan the transition from the *undesirable or problematic situation* (S1) to the FCFDS (S3).
  S5. The ***process*** to perform the transition from the *undesirable or problematic situation* (S1) to the *FCFDS* (S3) by providing the *solution system* (S6) according to the plan developed in the *planning process* (S4). S5 could be the SDP or an acquisition process if a suitable COTS system is available.
  S6. The solution ***system*** that will operate within FCFDS[7].

[7] The adjacent and supporting systems are not separate systems in this model because they are considered as subsystems or adjacent systems of the *solution system* (S6). If they are:

S7. The actual or created ***situation***.

S8. The process to determine that the realized solution remedies the evolved *undesirable situation*.

S9. The ***organization***(s) containing the processes and providing the resources for the operation and maintenance of the processes. S9 is also often known as the Enterprise.

Each of the nine systems must be viewed from the appropriate descriptive HTP. In summary, the Nine-System Model:

1. Is founded on a theory based on aspects of problem solving and system engineering.
2. Links into the existing problem-solving and process paradigms.
3. Builds best practices into systems engineering.
4. Discourages the current reductionist and isolationist views of a system by means of the built-in metasystem (S7).
5. Encourages testing of solution system (S6) in context of the created situation (S7).
6. Abstracts out complexity and consequently opposes today's tendency to make things more complex.
7. Contains clear boundaries and lines of demarcation between the nine systems.
8. Shows that Development Test and Evaluation (DT&E) takes place as one of the streams of work in S5 and Operational Test and Evaluation (OT&E) takes place in S8. Hence by definition adoption of the Nine-System Model incorporates those activities as best practice.
9. Includes aspects that tend to be ignored in current paradigm, such as:
   a. Planning the realization process.
   b. The concept that the top level system is something else's subsystem as in an airfield is part of an Air Defence System.
10. Can be used to clarify the confusion arising from different perspectives of systems engineering in the literature by showing how the Standards, the SIMILAR process, problem solving and Hitchins' version of systems engineering relate to each other.

5. SUMMARY AND CONCLUSIONS

This paper has summarized 20 years of research into the nature of systems engineering. Section 2 covered the need for research into a GUTSE (Friedman, 2006), articulated during the closing session of the NCOSE 1994 symposium and summarized research into the nature of systems engineering by providing a list of publications. Section 3 used the perspectives perimeter to summarize perceptions of systems engineering from the HTPs. Section 4 summarized seven frameworks that could be used to construct a framework of a GUTSE. Six of the frameworks, and the insights and benefits of the frameworks have already been published in detail in the cited peer review publications. All of the frameworks:

- Have provided insight into the nature of systems engineering and tools to use in the application systems engineering.
- Are interdependent.

The conclusions from the research are:

---

1. ***Subsystems:*** they are purview of the systems engineer of *solution system* (S6) in the same manner as any other subsystem and can be seen in the *Structural* and *Functional* HTPs of the *solution system* (S6).
2. ***Adjacent systems:*** they show up in the *Big Picture* perspective of the *solution system* (S6); their operational interactions and interfaces are seen in the *Operational* perspective of the *solution system* (S6). However, since S6 and the adjacent systems are subsystems of the metasystem operating in S7, the specification of the nature of the adjacent systems are the purview of the system engineer of that metasystem in the same way as the specification of the nature of the subsystems of S6 is the purview of the system engineer of the *solution system* (S6).

1. Even if the seven frameworks do not constitute subsystems of a GUTSE, they have been shown to be useful tools.
2. The combination of the frameworks provide a way to state the value of systems engineering as “being a part of the application of a systemic and systematic holistic approach to remedying complex problems”.

Further research is continuing.

6. BIOGRAPHIES

Joseph Kasser has been a practicing systems engineer for more than 40 years and an academic for about 16 years. He is a Fellow of the Institution of Engineering and Technology (IET), an INCOSE Fellow, the author of *“Holistic thinking: creating innovative solutions to complex problems”,* “*A Framework for Understanding Systems Engineering*” and “*Applying Total Quality Management to Systems Engineering”*, and many INCOSE symposia papers. He is a recipient of NASA’s Manned Space Flight Awareness Award (Silver Snoopy) for quality and technical excellence for performing and directing systems engineering and other awards. He holds a Doctor of Science in Engineering Management from The George Washington University. He is a Certified Manager and holds a Certified Membership of the Association for Learning Technology. He also started and served as the inaugural president of INCOSE Australia and served as a Region VI Representative to the INCOSE Member Board. He has performed and directed systems engineering in the UK, USA, Israel and Australia. He gave up his positions as a Deputy Director and DSTO Associate Research Professor at the Systems Engineering and Evaluation Centre at the University of South Australia in early 2007 to move to the UK to develop the world’s first immersion course in systems engineering as a Leverhulme Visiting Professor at Cranfield University. He is currently a Visiting Associate Professor at the National University of Singapore

Yang-Yang Zhao is an Associate Professor in the Norwegian Institute of System Engineering at Buskerud and Vestfold University College. She received her Ph.D. degree from the National University of Singapore in June 2013, having previously earned a master’s degree in System Design and Management there in July 2009. She received her first master’s degree in Engineering Management from the City University of Hong Kong then worked as a research assistant at its Department of Manufacturing Engineering and Engineering Management. Her research interests include system engineering, technology management and innovation strategy, and her work has appeared in the Journal of System Engineering, the Journal of Technology and Engineering Management, Total Quality Management & Business Excellence Journal and refereed international conferences.

7. REFERENCES

Adcock, R. D., "Tailoring Systems Engineering Lifecycle Processes to meet the challenges of Project and Programme applications," *The 15th International Symposium of the International Council on Systems Engineering (INCOSE)*, Rochester, NY., 2005.

Alexander, J. E. and Bailey, J. M., *Systems Engineering Mathematics*, Prentice-Hall, Inc., Englewood Cliff, NJ., 1962.

Allen, M., *Smart thinking: skills for critical understanding and writing*, Oxford University Press, 2004.

Arnold, S., *Systems Engineering: From Process towards Profession*, proceedings of The 10th Annual Symposium of the INCOSE, Minneapolis, MN, 2000.

Bahill, A. T. and Gissing, B., *Re-evaluating systems engineering concepts using systems thinking*, IEEE Transaction on Systems, Man and Cybernetics, Part C: Applications and Reviews 28 (1998), no. 4, 516-527.

Biemer, S. M. and Sage, A. P., "Systems Engineering: Basic Concepts and Life Cycle," *Agent-Directed Simulation and Systems Engineering*, L. Yilmaz and T. Oren (Editors), Wiley-VCH, Weinheim, 2009.

Biggs, J., *Teaching for Quality Learning in University*, Society for Research into Higher Education and Open University Press, 1999.

CHAOS, The Chaos study, The Standish Group, 1995, http://www.standishgroup.com/chaos.html, accessed on March 19, 1998.

Checkland, P. and Scholes, J., *Soft Systems Methodology in Action*, John Wiley & Sons, 1990.

Chestnut, H., *Systems Engineering Tools*, John Wiley & Sons, Inc., New York, 1965.

Cook, S. C., Kasser, J. E. and Asenstorfer, J., *A Frame-Based Approach to Requirements Engineering*, proceedings of 11th International Symposium of the INCOSE, Melbourne, Australia, 2001.

Cook, S. C., Kasser, J. E. and Ferris, T. L. J., *Elements of a Framework for the Engineering of Complex Systems*, proceedings of the 9th ANZSYS Conference, Melbourne, 2003.

Daniels, J., Bahill, T. and Botta, R., "A Hybrid Requirements Capture Process," *the 15th annual International Symposium of the INCOSE*, Rochester, NY., 2005.

DOD 5000.2-R, Mandatory Procedures for Major Defense Acquisition Programs (MDAPS) and Major Automated Information System (MAIS) Acquisition Programs, US Department of Defense, 2002.

DOD, "DoD Integrated Product and Process Development Handbook," O. o. t. U. o. D. A. a. Technology) (Editor), Department of Defence, Washinton DC, 1998.

DoDAF, *DoD Architecture Framework Version 1.0, 9 February 2004*, 2004.

Drucker, P. F., *Management: Tasks, Responsibilities, Practices*, Harper & Roe, New York, 1973.

EIA 632, "EIA 632 Standard: Processes for engineering a system," 1994.

Eichhorn, R., Developing thinking skills: critical thinking at the army management staff college, 2002, http://www.amsc.belvoir.army.mil/roy.html, accessed on April 11, 2008.

Evans, D. C., Systems Engineering Lessons Learned (SELL), 1989, http://evansopticalengineering.com/Page00/sysengll.htm, accessed on 6 September 2011.

Fayol, H., *General and Industrial Management*, Sir Isaac Pitman and Sons, Ltd., London, 1949.

Ford, W., Learning and teaching math, 2010, http://mathmaine.wordpress.com/2010/01/09/problems-fall-into-four-categories/, accessed on 17 November 2011.

Friedman, G., *On the Unification of Systems Engineering*, INSIGHT 8 (2006), no. 2, 16-17.

Gelbwaks, N. L., *AFSCM 375-5 as a Methodology for System Engineering*, Systems Science and Cybernetics, IEEE Transactions on 3 (1967), no. 1, 6-10.

Goldberg, R. and Assaraf, S., *System Engineering Process Improvement using the CMMI in Large Space Programs*, proceedings of 13thAnnual NDIA System Engineering Conference 25-28 Oct., San Diego CA, 2010.

Goode, H. H. and Machol, R. E., *Systems Engineering*, McGraw-Hill, 1959.

Gordon G. et al., *A Contingency Model for the Design of Problem Solving Research Pro-gram*, Milbank Memorial Fund Quarterly (1974), 184-220.

Hall, A. D., *A Methodology for Systems Engineering*, D. Van Nostrand Company Inc., Princeton, NJ, 1962.

Hari, A., Kasser, J. E. and Weiss, M. P., *Lessons Learnt From the Applications of QFD to the Definition of Complex Systems*, proceedings of The 16th International Symposium of the International Council on Systems Engineering (INCOSE), Orlando, FL., 2006.

Harrington, H. J., *Total Improvement Management the next generation in performance improvement*, McGraw-Hill, 1995.

Hill, J. D. and Warfield, J. N., *Unified Program Planning*, IEEE Transactions on Systems, Man, and Cybernetics SMC-2 (1972), no. 5, 610-621.

Hitchins, D. K., *Systems Engineering…In Search of the Elusive Optimum*, proceedings of Fourth Annual Symposium of the INCOSE-UK, 1998.

---, World Class Systems Engineering - the five layer Model, 2000, http://www.hitchins.net/5layer.html, accessed on 3 November 2006.

Hitchins, D. K., *Systems Engineering. A 21st Century Systems Methodology*, John Wiley & Sons Ltd., Chichester, England, 2007.

IEEE 1220, "Standard 1220 IEEE Standard for Application and Management of the Systems Engineering Process," 1998.

JAXA, Basics of Systems Engineering (draft) , Version 1B, Japan Aerospace Exploration Agency, 2007.

Jenkins, G. M., "The Systems Approach," *Systems Behaviour*, J. Beishon and G. Peters (Editors), Harper and Row, London, 1969, p. 82.

Jenkins, S., *A Future for Systems Engineering Tools*, proceedings of PDE 2005, The 7th NASA-ESA Workshop on Product Data Exchange (PDE), 2005.

Johnson, S. B., *Three Approaches to Big Technology: Operations Research, Systems Engineering, and Project Management*, Technology and Culture (1997), 891-919.

Jonassen, D. H., *Instructional design model for well-structured and ill-structured problem-solving learning outcomes*, Educational Technology: Research and Development 45 (1997), no. 1, 65-95.

Kasser, J. E., *Applying Total Quality Management to Systems Engineering*, Artech House, Boston, 1995.

---, *Systems Engineering: Myth or Reality*, proceedings of The 6th International Symposium of the INCOSE, Boston, MA., 1996.

---, *The Certified Systems Engineer - It's About Time!*, proceedings of the 10th Annual Symposium of the INCOSE, Minneapolis, MN, 2000a.

---, *Enhancing the Role of Test and Evaluation in the Acquisition Process to Increase the Probability of the Delivery of Equipment that Meets the Needs of the Users*, proceedings of the Systems Engineering Test and Evaluation Conference (SETE 2000), Brisbane, Australia, 2000b.

---, *A Framework for Requirements Engineering in a Digital Integrated Environment (FREDIE)*, proceedings of the Systems Engineering, Test and Evaluation Conference (SETE), Brisbane, Australia, 2000c.

---, *Writing Requirements for Flexible Systems*, proceedings of INCOSE-UK Spring Symposium, 2001.

---, "The acquisition of a System of Systems is just a simple multi-phased parallel-processing paradigm," *The International Engineering Management Conference*, Cambridge, UK, 2002a.

---, "The Cataract Methodology for Systems and Software Acquisition," *SETE 2002*, Sydney Australia, 2002b.

---, *Configuration Management: The silver bullet for cost and schedule control*, proceedings of The IEEE International Engineering Management Conference (IEMC 2002), Cambridge, UK, 2002c.

---, *Does Object-Oriented System Engineering Eliminate the Need for Requirements?*, proceedings of the 12th International Symposium of the International Council on Systems Engineering, Las Vegas, NV., 2002d.

---, *Isn't the acquisition of a System of Systems just a simple multi-phased time-ordered parallel-processing process?*, proceedings of 11th International Symposium of the INCOSE, Las Vegas, NV., 2002e.

---, *A Prototype Tool for Improving the Wording of Requirements*, proceedings of the 12th International Symposium of the INCOSE, Las Vegas, NV, 2002f.

---, *Systems Engineering: An Alternative Management Paradigm*, proceedings of The Systems Engineering Test and Evaluation Conference, Sydney Australia, 2002g.

---, *Object-Oriented Requirements Engineering and Management*, proceedings of The Systems Engineering Test and Evaluation (SETE) Conference, Canberra, 2003.

---, *The First Requirements Elucidator Demonstration (FRED) Tool*, Systems Engineering: The Journal of the International Council on Systems Engineering, 7 (2004a), no. 3.

---, *Researching the Properties of Object-Oriented Requirements*, proceedings of the Conference on Systems Engineering Research (CSER), Los Angeles, CA, 2004b.

---, *Researching the properties of object-oriented requirements to increase the probability of delivering a product the customer really needs as well as improving the effectiveness of the production process*, proceedings of INCOSE-Taiwan Chapter Conference: System Engineering on Industrial Applications, Taiwan, 2004c.

---, *Introducing the Role of Process Architecting*, proceedings of The 15th International Symposium of the International Council on Systems Engineering (INCOSE), Rochester, New York, 2005a.

---, *An Object-Oriented Approach to Requirements Engineering and Management*, proceedings of Rockwell's Consultants Seminar Series 2005 in Automation on the Move '05, Adelaide, 2005b.

---, *Viewing Systems Engineering from Outside the Box*, proceedings of the Third Annual Israeli National Conference on Systems Engineering, 2005c.

---, *Reducing the cost of doing work by an order of magnitude (by applying systems thinking to systems engineering)*, proceedings of the 21st Centre of Excellence Workshop: Challenges for life-based systems development, Tokyo, Japan, 2006.

---, *Eight deadly defects in systems engineering and how to fix them*, proceedings of the 17th International Symposium of the INCOSE, San Diego, CA, 2007a.

---, *A Framework for Understanding Systems Engineering*, Booksurge Ltd, 2007b.

---, *The Hitchins-Kasser-Massie (HKM) Framework for Systems Engineering*, proceedings of the 17th International Symposium of the INCOSE, San Diego, CA., 2007c.

---, *A Proposed Framework for a Systems Engineering Discipline*, proceedings of The Conference on Systems Engineering Research, Hoboken, NJ, 2007d.

---, *Active Brainstorming: - A systemic and systematic approach for idea generation*, proceedings of the 19th International Symposium of the International Council on Systems Engineering, Singapore, 2009a.

---, *The forthcoming Seldon crisis in systems engineering*, proceedings of presentation to the INCOSE Singapore Chapter on 6 April, Singapore, 2009b.

---, *Functions: The teaching language for systems engineering*, proceedings of the 4th Asia-Pacific Conference on Systems Engineering (APCOSE 2010), Keelung, Taiwan, 2010a.

---, *Holistic Thinking and How It Can Produce Innovative Solutions to Difficult Problems*, proceedings of the 7th bi-annual European Systems Engineering Conference (EuSEC), Stockholm, Sweden, 2010b.

---, *SEMP, TEMP and SHMEMP! It's time to stop the Mishigas*, proceedings of Researches and

Development Directions in Systems Engineering, The Gordon Center, Technion, Haifa, Israel, 2010c.

---, *Seven systems engineering myths and the corresponding realities*, proceedings of the Systems Engineering Test and Evaluation Conference, Adelaide, Australia, 2010d.

---, *An application of Checkland's soft systems methodology in the context of systems thinking*, proceedings of the 5th Asia-Pacific Conference on Systems Engineering (APCOSE 2011), Seoul, Korea, 2011a.

---, *Applying Holistic Thinking to Improving Your Sex Life*, proceedings of the Sixth Israeli Conference on Systems Engineering, Hertzlia, 2011b.

---, "Complex solutions for complex problems," *Third International Engineering Systems Symposium (CESUN)*, Delft, Holland, 2012a.

---, "Getting the Right Requirements Right," *the 22nd Annual International Symposium of the International Council on Systems Engineering*, Rome, Italy, 2012b.

---, *Holistic Thinking: creating innovative solutions to complex problems*, Createspace Ltd., 2013a.

---, *Model-Based Systems Engineering: Back to the future?*, proceedings of Asia-Pacific Council on Systems Engineering Conference (APCOSEC), Yokohama, Japan, 2013b.

---, "Improving the practice of systems engineering by adjusting the terminology," *submitted to the INCOSE International Symposium*, Las Vegas, NV, 2014.

Kasser, J. E. and Arnold, E., "Are postgraduate courses in systems engineering teaching the right things?," *the 24th International Symposium of the International Council on Systems Engineering (INCOSE)*, Las Vegas, 2014.

Kasser, J. E. and Cook, S. C., *The Communications Requirements Evaluation & Assessment Prototype (CREAP)*, proceedings of the 12th international symposiunm of the INCOSE, Las Vegas, NV., 2002.

---, *Using a Rapid Incremental Solution Construction Approach to Maximise the Completeness and Correctness of a Set of Requirements for a System*, proceedings of the 13th International Symposium of the International Council on Systems Engineering (INCOSE), Washington DC, 2003.

Kasser, J. E., Cook, S. C., Scott, W., Clothier, J. and Chen, P., *Introducing a Next Generation Computer Enhanced Systems Engineering Tool: The Operations Concept Harbinger*, proceedings of SETE 2002, Sydney Australia, 2002.

Kasser, J. E. and Frank, M., *A Maturity Model for the Competency of Systems Engineers*, proceedings of the 20th International Symposium of the INCOSE, Chicago, IL., 2010.

Kasser, J. E., Frank, M. and Zhao, Y. Y., *Assessing the competencies of systems engineers*, proceedings of the 7th bi-annual European Systems Engineering Conference (EuSEC), Stockholm, Sweden, 2010.

Kasser, J. E. and Hitchins, D. K., A framework for a systems engineering body of knowledge, 0.6, Report to the Fellows Committee, International Symposium of the International Council on Systems Engineering, 2009.

Kasser, J. E. and Hitchins, D. K., "Unifying the different systems engineering processes," *Conference on Systems Engineering Research*, Hoboken, NJ., 2010.

---, "Unifying systems engineering: Seven principles for systems engineered solution systems," *the 21st International Symposium of the INCOSE*, Denver, CO., 2011.

---, "Yes systems engineering, you are a discipline," *the 22nd Annual International Symposium of the International Council on Systems Engineering*, Rome, Italy, 2012.

---, *Clarifying the Relationships between Systems Engineering, Project Management, Engineering and*

Problem Solving, proceedings of Asia-Pacific Council on Systems Engineering Conference (APCOSEC), Yokohama, Japan, 2013.

Kasser, J. E., Hitchins, D. K., Frank, M. and Zhao, Y. Y., *A framework for benchmarking competency assessment models*, The Journal of the International Council on Systems Engineering (INCOSE) 16 (2013), no. 1.

Kasser, J. E., Hitchins, D. K. and Huynh, T. V., *Reengineering Systems Engineering*, proceedings of the 3rd Annual Asia-Pacific Conference on Systems Engineering (APCOSE), Singapore, 2009.

Kasser, J. E., Kaffle, S. and Saha, P., *Applying FRAT to improve systems engineering courseware: Project Review*, proceedings of SEEC Research Group, SESA-South Australia and INCOSE-Australia joint meeting, Adelaide, 2007.

Kasser, J. E. and Massie, A., *A Framework for a Systems Engineering Body of Knowledge*, proceedings of 11th International Symposium of the INCOSE, INCOSE, Melbourne, Australia, 2001.

Kasser, J. E. and Mirchandani, C. J., *The MSOCC Data Switch Replacement: A Case Study in Elicitating and Elucidating Requirements*, proceedings of the 15th International Symposium of the International Council on Systems Engineering (INCOSE), Rochester, NY, 2005.

Kasser, J. E. and Palmer, K., *Reducing and Managing Complexity by Changing the Boundaries of the System*, proceedings of the Conference on Systems Engineering Research, Hoboken NJ, 2005.

Kasser, J. E., Scott, W., Tran, X.-L. and Nesterov, S., *Improving the measurement of the understandability of requirements*, proceedings of 1st Colloquium for Information Systems Security Education - Asia Pacific, Adelaide, Australia, 2005a.

---, *A Proposed Research Programme for Determining a Metric for a Good Requirement*, proceedings of Conference on Systems Engineering Research, Los Angeles, CA., 2006.

Kasser, J. E. and Shoshany, S., *Systems Engineers are from Mars, Software Engineers are from Venus*, proceedings of the Thirteenth International Conference on Software & Systems Engineering & their Applications, Paris, 2000.

---, *Bridging the Communications Gap between Systems and Software Engineering*, proceedings of INCOSE-UK Spring Symposium, 2001.

Kasser, J. E., Sitnikova, E., Tran, X.-L. and Yates, G., *Optimising the Content and Delivery of Postgraduate Education in Engineering Management for Government and Industry*, proceedings of the International Engineering Management Conference (IEMC), St. John's, Newfoundland, Canada, 2005b.

Kasser, J. E., Tran, X.-L. and Matisons, S., *Prototype Educational Tools for Systems and Software (PETS) Engineering*, proceedings of Proceedings of the AAEE Conference, 2003.

Kasser, J. E. and Zhao, Y.-Y., *Managing Complexity via the Nine Systems in Systems Engineering*, proceedings of the 24th International Symposium of the International Council on Systems Engineering (INCOSE), Las Vegas, NV, 2014.

Kasser, J. E., Zhao, Y.-Y. and Mirchandani, C. J., *Simplifying Managing Stakeholder Expectations using the Nine-System Model and the Holistic Thinking Perspectives*, proceedings of the 24th International Symposium of the International Council on Systems Engineering (INCOSE), Las Vegas, NV, 2014.

Kline, S. J., *Conceptual Foundations for Multidisciplinary Thinking*, Stanford University Press, Stanford, 1995.

Leveson, N., *The Role of Software in Spacecraft Accidents*, AIAA Journal of Spacecraft and Rockets 41 (2004), no. 4.

Martin, J. N., *The Seven Samurai of Systems Engineering: Dealing with the Complexity of 7*

Interrelated Systems, proceedings of 14th Annual Symposium of the International Council on Systems Engineering, Toulouse, France, 2004.

MIL-STD-499A, *Mil-STD-499A Engineering Management*, United States Department of Defense (USAF), 1974.

Paul, A. S. and Owunwanne, C., "The Systems Project: Life Cycle Development/Management of as Many as Four Interrelated Systems," *16th Annual Symposium of the International Council on Systems Engineering*, Orlando, FL., 2006.

Paul, R. and Elder, L., *Critical Thinking: learn the tools the best thinkers use - concise ed.*, Pearson Prentice Hall, 2006.

Peters, T. and Austin, N., *A Passion for Excellence*, Warner Books, 1985.

Peters, T. J. and Waterman, H. R., *In Search of EXCELLENCE*, Harper and Row, 1982.

Rittel, H. W. and Webber, M. M., *Dilemmas in a General Theory of Planning*, Policy Sciences 4 (1973), 155-169.

Rodgers, T. J., Taylor, W. and Foreman, R., *No-excuses Management*, Doubleday, 1993.

Scott, W., Kasser, J. E. and Tran, X.-L., *Improving the Structure and Content of the Requirement Statement*, proceedings of The 16th International Symposium of the International Council on Systems Engineering (INCOSE), Orlando, FL., 2006.

Sheard, S. A., *Twelve Systems Engineering Roles*, proceedings of The 6th Annual International Symposium of the NCOSE, Boston, MA., 1996.

Sutcliffe, A., Galliers, J. and Minocha, S., *Human Errors and System Requirements*, proceedings of IEEE International Symposium on Requirements Engineering, Limerick, Ireland, 1999.

Swarz, R. S. and DeRosa, J. K., *A Framework for Enterprise Systems Engineering Processes*, proceedings of International Conference on Computer Science, Engineering and Applications (ICSSEA), 2006.

Tran, X.-L. and Kasser, J. E., *Improving the recognition and correction of poorly written requirements*, proceedings of The Systems Engineering Test and Evaluation (SETE) Conference, Brisbane, Australia, 2005.

Welby, S., *DoD Systems Engineering Update*, proceedings of 36th Air Armament Symposium, 2010.

Wilson, W. E., *Concepts of Engineering System Design*, McGraw-Hill Book Company, 1965.

Wolcott, S. K. and Gray, C. J., Assessing and Developing Critical Thinking Skills, 2003, http://www.wolcottlynch.com/Downloadable_Files/IUPUI%20Handout_031029.pdf, accessed on 21 May 2013.

Woolfolk, A. E., "Chapter 7 Cognitive views of learning," *Educational Psychology*, Allyn and Bacon, Boston, 1998, pp. 244-283.

Wynne, M. W. and Schaeffer, M. D., *Revitalization of Systems Engineering in DoD*, Defense AT&L (2005), 14-17.