2009

Reengineering Systems Engineering

Kasser, Joseph; Hitchins, Derek; Huynh, Thomas V.

Reengineering Systems Engineering

Joseph Kasser
Visiting Associate Professor
Temasek Defence Systems Institute
National University of Singapore
Block E1, #05-05, 1 Engineering Drive 2, Singapore 117576
joseph.kasser@incose.org

Derek Hitchins
ASTEM
Consultant Systems Architect

Thomas V. Huynh
Associate Professor
Department of Systems Engineering
Naval Postgraduate School
Monterey, CA, 94393
thuynh@nps.edu

Abstract. This paper documents (1) the early phases of using systems engineering to develop a conceptual system – the system being developed is a systems engineering body of knowledge (SEBoK) and (2) the findings and opportunities generated in those early phases. The approach was based on identifying activities specific to systems engineering, as opposed to the broad raft of activities that systems engineers might undertake, according to their role. An activity-based definition of systems engineering vs. non-systems engineering role-based definition was developed.

The second part of the paper identifies five types of systems engineers, discusses the evolution of systems engineering in terms of those five types, and hypothesizes that a major cause of the failure of systems engineering is the allocation of inappropriate types of systems engineers to early lifecycle phase systems engineering activities. The paper concludes with some insights and recommendations for further study.

Evolution of the role of systems engineering

Descriptions of systems engineering currently comprise different interpretations of the activities known as systems engineering and the broad raft of activities that systems engineers might undertake according to their role in the workplace. This quagmire has developed because different users of the term 'systems engineering' for almost 50 years have chosen or perceived different meanings. For example, one comment from 1960 was "*Despite the difficulties of finding a universally accepted definition of systems engineering*[1]*, it is fair to say that the systems engineer is the man who is generally responsible for the over-all planning, design, testing, and production of today's automatic and semi-automatic systems*" (Chapanis, 1960) page 357). (Jenkins, 1969) page 164) expanded that comment into the following 12 roles of the systems engineer:

1. He tries to distinguish the wood from the trees – what's it all about?
2. He stimulates discussion about objectives – obtains agreement about objectives.
3. He communicates the finally agreed objectives to all concerned so that their co-operation can be relied upon.
4. He always takes an overall view of the project and sees that techniques are used sensibly.
5. By his overall approach, he ties together the various specializations needed for model building.
6. He decides carefully when an activity stops.

[1] Fifty years later, nothing has changed in that respect.

7. He asks for more work to be done in areas which are sensitive to cost.
8. He challenges the assumptions on which the optimization is based.
9. He sees that the project is planned to a schedule, that priorities are decided, tasks allocated, and above all that the project is finished on time.
10. He takes great pains to explain carefully what the systems project has achieved, and presents a well-argued and well-documented case for implementation.
11. He ensures that the users of the operational system are properly briefed and well trained.
12. He makes a thorough retrospective analysis of systems performance.

Seven of these roles of the systems engineer (activities performed by a person with the title systems engineer) overlap the role of the project manager (activities performed by a person with the title project manager). Research into the reason for the overlapping of the disciplines turned up information as to how the overlap originated in the form of the following statement. *"Driven by cold war pressures to develop new military systems rapidly, operations research, systems engineering, and project management resulted from a growing recognition by scientists, engineers and managers that technological systems had grown too complex for traditional methods of management and development"* (Johnson, 1997) Thus systems engineering, project management and operations research can be seen as three solutions to the problems posed by complex systems in the Cold War by three different communities of practice (Johnson, 1997) that have continued to evolve and overlap. Some of the evolution in systems engineering can be seen in the very little overlap between the 12 roles documented by (Jenkins, 1969) and the following 12 systems engineering roles documented by (Sheard, 1996):

1. **Requirements Owner (RO) Role.** Requirements Owner/requirements manager, allocator, and maintainer/specifications writer or owner/developer of functional architecture/developer of system and subsystem requirements from customer needs.
2. **System Designer (SD) Role.** System Designer/owner of "system" product/chief engineer/system architect/developer of design architecture/specialty engineer (some, such as human-computer interface designers)/"keepers of the holy vision" (Boehm, 1994).
3. **System Analyst (SA) Role.** System Analyst/performance modeler/keeper of technical budgets/system modeler and simulator/risk modeler/specialty engineer (some, such as electromagnetic compatibility analysts).
4. **Validation and Verification (VV) Role**. Validation and Verification engineer/test planner/owner of system test program/system selloff engineer. VV engineers plan and implement the system
5. **Logistics and Operations (LO) Role.** Logistics, Operations, maintenance, and disposal engineer/developer of users' manuals and operator training materials.
6. **Glue (G) Role.** Owner of "Glue" among subsystems/system integrator/owner of internal interfaces/seeker of issues that fall "in the cracks"/risk identifier/"technical conscience of the program".
7. **Customer Interface (CI) Role.** Customer Interface/customer advocate/customer surrogate/customer contact.

8. **Technical Manager (TM) Role.** Technical Manager/planner, scheduler, and tracker of technical tasks/ owner of risk management plan/product manager/product engineer.
9. **Information Manager (IM) Role.** Information Manager (including configuration management, data management, and metrics).
10. **Process Engineer (PE) Role.** Process engineer/business process reengineer/business analyst/owner of the systems engineering process.
11. **Coordinator (CO) Role.** Coordinator of the disciplines/tiger team head/head of integrated product teams (IPTs)/system issue resolver.
12. **"Classified Ads Systems Engineering" (CA) Role.** This role was added to the first eleven in response to frustration encountered when scanning the classified ads, looking for the INCOSE-type of systems engineering jobs.

Jenkins' roles relate to conceiving and planning the solution system while almost 30 years later, few of Sheard's roles address the original systems engineering approach to conceiving and planning the solution system. Sheard's set of roles relate to interpersonal relationships between the practitioners of disparate skills and disciplines implementing the solution system. Furthermore, according to both Jenkins and Sheard the role of the systems engineer (the activities performed by a person with the title systems engineer) overlaps activities performed (the roles) by people from other professions[2]; see (Brekka, et al., 1994; Roe, 1995; DSMC, 1996; Kasser, 1996; Sheard, 1996; Johnson, 1997; Watts and Mar, 1997; Bottomly, et al., 1998; Kasser, 2002b) and Figure 1 for just a few examples of the different overlaps between systems engineering and project management. Note the Defense Systems Management College definition of systems engineering as "*The management function which controls the total system development effort for the purpose of achieving an optimum balance of all system elements. It is a process which transforms an operational need into a description of system parameters and integrates those parameters to optimise the overall system effectiveness*" (DSMC, 1996). Notice the use of the term "*management function*"! In addition, see (Emes, et al., 2005) for overlaps between systems engineering and other disciplines and (Hari, et al., 2004) for an example of

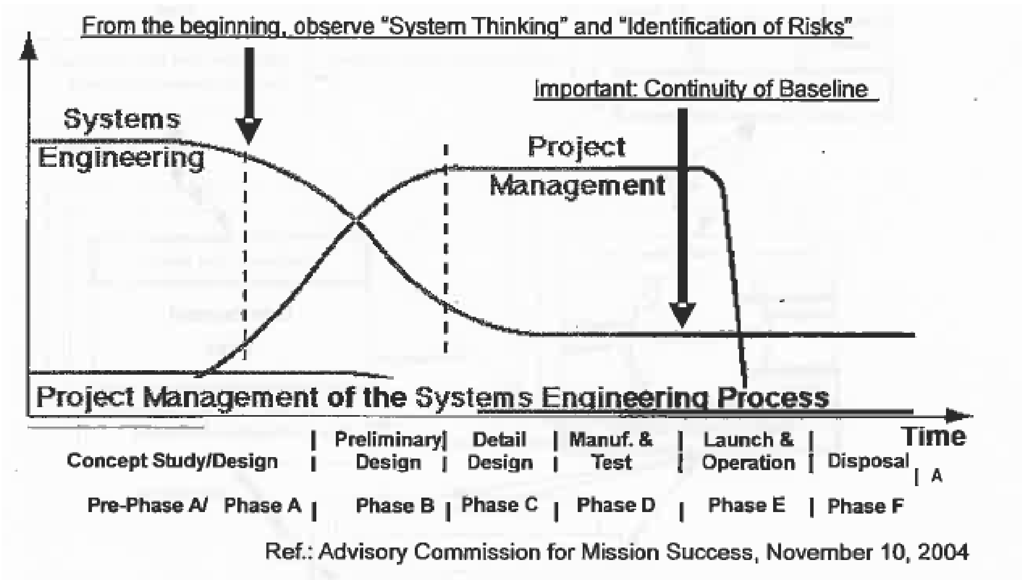

Figure 1 JAXA Project management and systems engineering (JAXA, 2007)

[2] A different set, as seen across the years.

the activities performed in new product design that overlap systems engineering. In addition, the activities performed by the systems engineer in one organisation are different to those performed by a systems engineer in another organization and so are the knowledge requirements for their activities. Consequently, defining a body of knowledge for systems engineering poses a major challenge.

Defining a body of knowledge based on the role of a systems engineer will be difficult if not impossible because the role of the systems engineer has evolved over time so that it is different in practically every organisation. As such, the solution to the problem of defining a body of knowledge for systems engineering is to dissolve the problem by making a change in the paradigm. This approach, which redesigns the system containing the problem or changes the perspective from which the problem is viewed to produce an innovative solution is one of the four ways to tackle a problem suggested by (Ackoff, 1999) page 115). The paradigm change is made by making a distinction between a set of activities known as systems engineering and the role of the systems engineer which is the sum of the systems engineering and non-systems engineering activities systems engineers perform in the workplace (Kasser and Palmer, 2005). The focus on activities is a return to Hall's definition of "systems engineering as a function not what a group does" (Hall, 1962) page 11) and means that the knowledge needed by systems engineers in their roles will be more than the activities to be defined as 'systems engineering' (see below), and that knowledge can be separated into 'systems engineering' and 'everything else'. The 'systems engineering' knowledge will be placed in the systems engineering body of knowledge (SEBoK), and the subset of knowledge of 'everything else' that will be needed will be out of scope of the SEBoK but will be referenced appropriately.

Separating out the systems engineering knowledge

In the activity paradigm, various people in various disciplines at various times perform a set of activities from the time a problem is being defined, though the conceptualisation, design, construction and operation of the system that solves, resolves or dissolves the problem to the time that the system has been taken out of service and disposed. This set of activities may be partitioned into subsets in various ways such as by professional discipline (project/engineering management, systems engineering, engineering, new product design, etc.) and by time (the phases in the system lifecycle). Various systems engineers and non-systems engineers perform different subsets of systems engineering activities and different subsets of non-systems engineering activities. As discussed above, the mapping of the role of the systems engineer to activities is different in different organisations, hence the aforementioned difference in their descriptions when systems engineers get together and discuss their roles.

Looking at the structure of organisations from the temporal perspective, in general the structure of organisations is still based on the work of F.W. Taylor who systems engineered his mining organisation and split the work into two streams of activities which have become known as 'management' and 'labour'(Taylor, 1911). However, since that time, the structure of companies and the nature of work have changed. Organizational structures have become flatter, decision making has become decentralized, information is widely shared, workers form project teams, even across organizations, and work arrangements are flexible (Microsoft, 2008). Taylor's split is no longer applicable. Consequently, we now propose to reengineer (in the sense of the word as used by (Hammer and Champy, 1993) Taylor's split for organisations developing systems by splitting work into two different streams, systems engineering and non-systems engineering and further partitioning the non-systems engineering streams as described below.

The INCOSE Fellows definition of systems engineering was considered as a starting point for determining what went into the systems engineering stream. The definition is "*Systems Engineering is an engineering discipline whose responsibility is creating and executing an interdisciplinary process to ensure that the customer and stakeholder's needs are satisfied in a high quality, trustworthy, cost efficient and schedule compliant manner throughout a system's entire life cycle*" (INCOSE Fellows, 2009). However, if the words '*engineering discipline'* are replaced by the words '*project management*' many project managers would consider the definition to apply to project management. This definition may be understood as applying to both the role of the systems engineer and the role of the project manager since the roles overlap both in space and time as discussed above. As such, an alternative definition is needed.

Further research to determine what went into the systems engineering stream showed that the approved Standards used in systems engineering do not seem to actually apply to systems engineering – they cover systems engineering management and the processes for engineering a system! Thus:

- Mil-STD-499 covers systems engineering management (MIL-STD-499, 1969).
- Mil-STD-499A covers engineering management (MIL-STD-499A, 1974) dropping the word 'systems' from the title.
- The draft (MIL-STD-499B, 1993) and MIL-STD-499C (Pennell and Knight, 2005) Standards contain the words "systems engineering" in their titles but the Standards were never approved and these Standards also (as did 499 and 499A) generally ignore most of the problem identification, whole solution conceptualisation and solution implementation planning activities that take place in Phase A of **Figure 1**.
- ANSI/EIA-632 covers processes for engineering a system (ANSI/EIA-632, 1999).
- The IEEE 1220 Standard is for the application and management of the systems engineering process (IEEE 1220, 1998).
- The ISO/IEC 15288 Standard lists processes performed by systems engineers (Arnold, 2002) and hence may be considered as being applicable to the role of the systems engineer rather than to the activities known as systems engineering.

The phases in providing a whole complete solution to a problem can be considered as a set of activities performed by various people in various disciplines at various times. Some of those activities are systems engineering, and some are not systems engineering. The next approach was to develop a list of activities that could be described as systems engineering. Research found several sources of lists of activities including:

- (Eisner, 1988) who lists a general set of 28 tasks and activities that were normally performed within the overall context of large-scale systems engineering. He calls the range of activities 'specialty skills' because some people spend their careers working in these specialties. Thus according to Eisner [the role of[3]] systems engineering overlaps at least 28 engineering specialties.

---

[3] Author's interpretation.

- (Hyer, 1997) provides a list of nine activities for systems integration but which do not necessarily take place during the systems integration phase.
- (Eisner, 1997) page 156) expands (Eisner, 1988) and discusses 30 tasks that form the central core of systems engineering. The whole area of systems engineering management is covered in just one of the tasks. Eisner states that "*not only must a Chief Systems Engineer understand all 30 tasks; he or she must also understand the relationships between them, which is an enormously challenging undertaking that requires both a broad and deep commitment to this discipline as well as the supporting knowledge base*".

Should the research continue in this direction, the resulting list would be long, subjective and open to never ending discussion. Looking outside the box, lessons learned from psychology indicate that long lists are not the way to proceed. At one point of time in the development of theories of motivation, Henry A. Murray identified separate kinds of behaviour and developed an exhaustive list of psychogenic or social needs (Murray, 1938). However, the list is so long that there is almost a separate need for each kind of behaviour that people demonstrate (Hall and Lindzey, 1957). While Murray's list of 39 kinds of behaviours has been very influential in the field of psychology, it has not been applied directly to the study of motivation in organizations because the length of the list makes it impractical to use. On the other hand, Maslow's hierarchical classification of needs (Maslow, 1966, 1968, 1970) has been by far the most widely used classification system in the study of motivation in organizations. Maslow differs from Murray in two important ways; his list is:

- **Arranged in a hierarchy** -commonly drawn as a pyramid, and contains a set of hypotheses about the satisfaction of these needs.
- **Short** -- Only five categories.

The eventual approach chosen to determine what is and what is not a systems engineering activity was to dissolve the problem by developing a criterion for what constitutes an activity to be defined as systems engineering rather than trying to resolve the problem by a developing a list of activities. The following criterion was used to determine if an activity does or does not belong in the set of activities to be known as systems engineering:

- 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 systems engineering.
- 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 systems engineering but is part of 'something else', e.g., engineering management, software engineering, etc.

The activities of systems engineering have focused on both analysis and systems thinking. Analysis which has three steps (Ackoff, 1991) can be performed as 'reductionism' or 'decomposition' – reducing the parts to ever decreasing components *in isolation*, but should be performed by the systems engineer as 'elaboration' (Hitchins, 2003) pages 93-95) – which examines the parts in increasing detail *without losing track of the part's relationship to the overall system*. Systems thinking, on the other hand also has three steps (Ackoff, 1991) but they are slightly different. Comparing analysis and systems thinking in the manner shown in **Table 1**, one can see that the focus of analysis is to look inwards while the focus of systems thinking is to look outwards. Both analysis (in the form of 'elaboration') and systems thinking have their place in the activities performed in developing an understanding of a

Table 1 Analysis and Systems Thinking

Analysis (Machine Age)	**Systems Thinking (Systems Age)**
1. Take apart the thing to be understood	1. A thing to be understood is conceptualized as a part of one or more larger wholes, not as a whole to be taken apart;
2. Try to understand how these parts worked	2. An understanding of the larger system is sought;
3. Assemble an understanding of the parts into an understanding of the whole.	3. The system to be understood is explained in terms of its role or function in the containing system.

system (Hitchins, 1992) page 14) and are but two of the systems thinking perspectives (Kasser and Mackley, 2008).

Since the activities forming the 'something else's' are part of the context of systems engineering and are often performed by systems engineers, it is recognised that systems engineers need the knowledge to perform or understand many of the activities defined as 'something else' but that knowledge *per se* is out of the scope of the SEBoK and will be identified accordingly. The 'something else' activities were further partitioned into the following sets of non-systems engineering activities:

- Engineering.
- Management.
- Other.

The proposed activity paradigm definitions of the systems engineering sets of activities and the non-systems engineering sets of activities are as follows:

- **Systems engineering** is the set of activities involved with *dealing with parts and their interactions as a whole.*
- **Engineering** is the set of activities *dealing with a part in isolation.* If the part is not a technological product, for example if the part is such as a human element, then use of language is such that the activity is not called engineering but something else, such as training or exercising.
- **Management** is the set of activities known as planning organising, directing, staffing and controlling activities for and in the production of the *part in isolation*.
- **Other** is the remaining set of activities not included in the previous definitions.

Combining these definitions it can be seen that in the activity paradigm:

- **Systems engineering management** is the set of activities known as planning organising, directing, staffing and controlling *systems engineering* activities *in isolation from the other sets of management activities*.
- **Engineering management** is the set of activities known as planning organising, directing, staffing and controlling *engineering* activities *in isolation from the other sets of management activities.*

Lastly for the sake of completing the set of definitions, a **task** is an activity performed within a specific period of time and a **project** consists of a temporary endeavor [set of tasks] undertaken to create a unique product, service or result (PMI, 2004). It follows that:

- **Project management** is the set of activities known as planning organising, directing, staffing and controlling a temporary set of tasks undertaken to create a unique product, service or result, *in isolation from the other projects*.

The next phase in determining the SEBoK will be to identify the activities performed in each phase of the system lifecycle developing a concept of operations of the work being performed and then using the simple activity-based criterion to determine which of the activities are and which are not systems engineering. Each activity will be defined in such a manner as to terminate with the production of a tangible product or products which is/are transferred to the start of the subsequent activity in accordance with (Kasser, 1997). The activities have been grouped by the phases in the first and second systems engineering processes[4] in the system lifecycle for a system that is developed from conception to disposal[5] using the Hitchins-Kasser-Massie Framework (HKMF) for understanding systems engineering (Kasser, 2007) shown in **Figure 2**. Each area of the HKMF can potentially contain all sets of (systems engineering and non-systems engineering) activities – some more than others. **Figure 1** also provides an indication of the relative ratios between the sets of activities known as systems engineering and the sets of activities known as project management over the system life cycle.

The HKMF has also identified one reason for debates in the meaning of terminology used by systems engineers. Words such as ‘capability’ and ‘system design’ have different meanings in different areas of the HKMF. The confusion in the use of the term ‘operations concept’ and ‘concept of operations’ can be similarly be clarified when one realizes that the terms refer to products produced in different columns of the

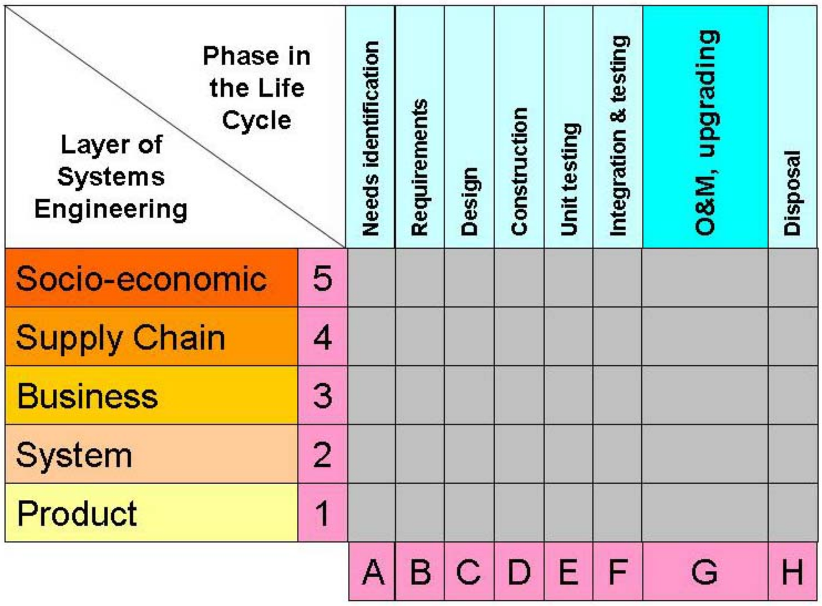

Figure 2 The HKM Framework for understanding systems engineering

[4] The first systems engineering process deals with identifying the real problem and a number of alternative conceptual solutions followed by the choice of an optimal conceptual solution to the whole problem, The second systems engineering process follows the first and deals with the creation, operation and disposal of an optimal physical implementation of the conceptual solution to the problem generated by the first systems engineering process.
[5] Other lifecycles do exist.

HKMF. In addition the vocabulary for describing concepts in Layer 2 for single system development in isolation is different to the vocabulary used in Layer 3 to express the same concepts in business processing reengineering.

The vertical dimension of the HKMF contains the five-layers of systems engineering (Hitchins, 2000). The horizontal dimension of the framework is organized as sequential phases in providing a whole complete solution to a problem as an overall, end-to-end process which consists of conceiving a whole solution to solve a problem and making that whole "come to life" for the development of a single system in isolation. The phases have been stated in various ways in various Standards, conference papers and books, but in the HKMF they are defined in generic terms as:

A. **Identifying the need**. This is the phase where the bulk of the set of activities known as systems engineering is performed. Yet in the Type II systems engineering educational paradigm it tends to be glossed over (see below for an explanation of the five types of systems engineers and why this phase is glossed over). Phase A is based on (Hall, 1962), (Gelbwaks, 1967), (Hitchins, 1992) and the summary in (Brill, 1998) and contains the first 'systems engineering' process addressing the conceptual solution. Phase A comprises the following sub-phases:

    1. This sub-phase contains the set of activities that explore/scope the problem, leading directly to Phase A.2. The activities performed in this phase produce a definitive statement of the problem-in-context.
    2. This sub-phase contains the set of activities that conceive the whole solution system (which 'emerges' from/"complements" the problem) and produces the concept of operations (CONOPS) that describes how the solution system will operate in its future environment.
    3. This sub-phase contains the set of activities that design the whole solution system, identify the environment, other interacting systems, the subsystems, parts, interactions, functional architecture, physical architecture, etc., etc., - but still all of the whole.

B. Requirements analysis. This phase is the first phase of the second 'systems engineering' process addressing the physical solution and its implementation and contains the set of activities that specify the solution system as a full set of specifications for the whole and for the parts and their infrastructure, including the environment/Weltanschauung or paradigm that justifies them. If the specifications are in the form of text mode requirements, the output of this phase tends to be at the 'A' specification level (MIL-STD-490A, 1985). Unfortunately, many systems engineers have been educated to consider this phase as the first phase of a single systems engineering process. For example, (1) according to (Martin, 1997) page 95), (Eisner, 1997) page 9), (Wasson, 2006) page 60) and (DOD 5000.2-R, 2002), pages 83-84) requirements are one of the inputs to the 'systems engineering process'; and (2) in one postgraduate class at University of Maryland University College the instructor stated that systems engineering began for him when he received a requirements specification (Todaro, 1988). While (DOD 5000.2-R, 2002) pages 73-74) does call out the 'analysis of possible alternatives' subset of activities in Phase A2 of the HKMF, those activities are called out as part of the separate seemingly independent Cost as an Independent Variable (CAIV) process which (1) is a way of complicating just a part of the concept of designing budget tolerant systems (Denzler and Kasser, 1995) using the

Cataract approach (Kasser, 2002a) and (2) takes place **before** the DOD 5000.2-R 'systems engineering process' begins.

C. **Design.** This phase contains the set of activities that creates a more detailed design of the whole solution system through a combination of people, doctrine, parts, subsystems, interactions, etc., including configuration, architecture and implementation criteria. The output of this phase tends to be at the 'B' specification level (MIL-STD-490A, 1985).
D. **Construction.** This phase contains the set of activities that create the individual parts, subsystems, interactions, etc. *in isolation.* Consequently the set of activities are mainly engineering, training, etc., not systems engineering. This situation is indicated in **Figure 1** by the down slope in the line showing the amount of systems engineering at this phase.
E. **Unit Testing.** This phase contains the set of activities that validate the performance of the individual parts, subsystems, interactions, etc. *in isolation* against their requirements. Consequently the set of activities are mainly engineering, not systems engineering.
F. **Integration and testing of the system.** This phase contains the set of activities that (1) combines the parts, subsystems, interactions, etc., to constitute the solution system, and (2) establishes, under test conditions, the performance of the whole solution system, with optimum effectiveness, in its operational context.
G. **Operations, maintenance and upgrading of the system.** This phase contains the set of systems engineering and non-systems engineering activities that actively provide a solution to the problem for which the whole system was created. This phase includes operating the system, support to maintain operations; improvements to the whole to enhance effectiveness, and to accommodate changes in the nature of the problem over time. These changes iterate phases A to F (call them Ga .. Gf), ideally without rendering the operating solution system materially inoperative for an unacceptable period of time.
H. **Disposal of the system.** This phase contains the set of activities that dispose of the system. This phase is rendered necessary where either where the problem no longer exists, or the solution system is no longer capable of solving the problem effectively or economically. If the disposal method has not been predetermined, this phase may also iterate phases A to F (call them Ha .. Hf).

This approach to determining the contents of the SEBoK is also domain independent but recognises that systems engineers do need domain knowledge (as well as systems thinking, communications and interpersonal skills). A serendipitous outcome of this approach which needs more research, would truly reengineer the work of (Taylor, 1911). For example,

- The potential exists to redraw role boundaries to align with the activity boundaries and remove much of the role overlap and inefficiency in organisations.
- A systems engineering approach can be used to determine the systems and non-systems engineering activities performed in any row and column of the HKMF based on the operations performed in that area of the framework. The activities can be grouped in various ways into specific roles (job positions) and the knowledge requirements for those roles can be developed. These

requirements would provide the knowledge component requirement for the person or persons to be assigned to perform the activities. The competency requirement for the person would be determined separately.

The five types of systems engineers

The human side of systems engineering is the systems engineers who perform the roles known as systems engineering. These roles perform the conceiving and creating the solution system systems engineering activities, the project management activities, engineering and other speciality engineering activities in various mixes depending on the phase in the system lifecycle and the organisation in which the systems engineer works. Optimal performance of each of the activities requires different characteristics in the systems engineer. Previous attempts to identify characteristics of systems engineers have been based on the traits attributable to systems engineers e.g. (Hall, 1962; Frank, 2006) and the INCOSE UK Systems Engineering Competencies Framework (Hudson, 2006) The list of desirable traits is increasing steadily. However, the lessons learned from psychology discussed above suggest lists are not the way to proceed and that an alternate approach be found. Hence, instead of using lists of traits, an alternative approach[6] of characterising systems engineers into the following five types is proposed based on their ability to deal with problems and solutions.

- **Type I**. This type is an “apprentice who can be told “how” to implement the solution and can then implement it.
- **Type II**. This type is the most common type of systems engineer. Type II’s have the ability to use the systems engineering process to figure out how to implement a physical solution once told what conceptual solution to implement.
- **Type III**. Once given a statement of the problem, this type has the necessary know-how to conceptualize the solution and to plan the implementation of the solution.
- **Type IV.** This type has the ability to examine the situation and define the problem (Wymore, 1993) page 2).
- **Type V.** This type combines the abilities of the Types III and IV, namely has the ability to examine the situation, define the problem, conceptualise the solution and

Table 2 Failure data from GAO Report 06-368, 2006

Cost and Schedule Outcomes Sorted by Percent of Product Development Remaining

Programs	Percent cost growth[a]	Schedule growth, in months	Percent of development remaining
Aerial Common Sensor	45%	24	85%
Future Combat System	48%	48	78%
Joint Strike Fighter	30%	23	60%
Expeditionary Fighting Vehicle	61%	48	49%
C-130 Avionics Modernization Program	122%	Delays anticipated	Undetermined
Global Hawk (RQ-4B)	166%	Delays anticipated	Undetermined

Sources: DOD (data); GAO (analysis and presentation).

[a]Cost growth is expressed as the percent change in program development cost estimates in 2005 base year dollars.

[6] Based on years of observations by the authors.

plan the implementation of the physical solution.

Types I to III are levels through which a person grows with education and experience. The debate on 'nature' or 'nurture' comes into play at Levels IV and V. However, irrespective of the debate, it is important to identify people with the potential to become Type IV's and V's as early as possible in their careers and then to provide them with fast track training to enable their organization to obtain the best use of their capabilities in the future.

The new approach to characterizing systems engineers provides a hypothesis for a reason for the failure of systems engineering in the early stages of large projects (Hiremath, 2008) and other examples of poor systems engineering implementation (GAO, 2006). For example, the cost and schedule overruns in the Joint Strike Fighter (JSF) development project shown in **Table 2** were predicted in (Kasser, 2001) and hence probably preventable. Had Type V systems engineers been working on the phases of the JSF project in column A of the HKMF, the factors identified as potential causes of cost and schedule overruns leading to the prediction in (Kasser, 2001) would have probably been identified as risks. Appropriate risk management techniques would then have been recommended and if these risk management techniques had been implemented[7], the ensuring cost and schedule overruns would have been reduced.

Research seems to show that the early systems engineers of the 1950's and 1960's tended to focus on identifying the problem (Wymore, 1993) and finding an optimal solution (Goode and Machol, 1959; Hall, 1962). These early systems engineers were of Type III, IV, and V, while the systems engineers who came later tended to focus on processes (Type II)'s. Back in the "good old days" of systems engineering Type III, IV and V systems engineers solved/resolved/dissolved the problem in the first 'systems engineering' process addressing the conceptual solution, then initiated the implementation of the solution, and moved on to the next contract, leaving the Type II's to continue assisting the development of the solution system in the second systems engineering process. There then came a time when there was a lack of new projects and so many of the Type III, IV and V's were laid off and lost to the discipline. When the need for systems engineers picked up again, in general only the Type II systems engineers were left and they took over systems engineering. They had seen a successful process for developing systems and so their focus was on the second systems engineering process. They wrote the standards used in systems engineering (MIL-STD-499, 1969; MIL-STD-499A, 1974; EIA 632, 1994; IEEE 1220, 1998) for other Type II systems engineers to follow. These Standards in turn became the foundation for educating systems engineers. The 499, 499A, 632, 1220, and 15288 Standards cover the systems engineering process and engineering management because there is actually very little systems engineering (the activity not the role) in the subsystem design, construction, and unit testing phases (HKMF Columns C, D and E) of the systems lifecycle for a single system in isolation. Activities pertaining to subsystems and units in isolation are engineering of systems not systems engineering activities according to the criterion defined above. The mantra became 'follow the process and all will be well'. The term GIGO - garbage in, garbage out, was acknowledged but ignored. In this paradigm:

[7] A big "if" since political considerations in the Type II process paradigm would probably have precluded the risk mitigation activities.

Table 3 Focus of Standards – Chronological Order

	MIL-STD 499C	ANSI EIA 632	IEE 1220	CMMI	ISO 15288
Conceptualizing problem and alternative solutions	**No**	**No**	**No**	**No**	**No**
Mission Purpose/Definition	**No**	**No**	Yes	Yes	Yes
Requirements engineering	Yes	Yes	Yes	Yes	Yes
System architecting	Yes	Yes	Yes	Yes	Yes
System implementation	**No**	Yes	**No**	Yes	Yes
Technical analysis	Yes	Yes	Yes	Yes	Yes
Verification & validation	Yes	Yes	Yes	Yes	Yes

- While the subset of the systems engineering profession focuses on processes (Type II systems engineers), the literature on "excellence" focuses on people (Type V systems engineers) e.g. (Peters and Waterman, 1982; Peters and Austin, 1985; Rodgers, et al., 1993).).
- The focus is on process and not on providing an understanding of the context and the ability to tailor the process (as was called out in (MIL-STD-499, 1969). This is seen in systems engineering courses where the students are taught about the process but not about the context.
- Processes seen to work in one culture or organization have been copied verbatim by other organizations, with dismal results. Examples can be found in the lessons learned in (O'Toole, 2004) and (Angel and Froelich, 2008)'s reasons for a claimed Six Sigma initiative 60% failure rate.
- The systems engineering process has a high degree of correlation to the problem solving process because that was the process documented in the Standards. See (GDRC, 2009) and (OVAE, 2005) for typical examples of the problem solving process.
- The Standards commonly used/taught in systems engineering (MIL-STD-499, 1969; MIL-STD-499A, 1974) and (DOD 5000.2-R, 2002) pages 83-84) ignore most of the activities allocated to Phase A in **Figure 1** and Phase A of the HKMF resulting in the critical first systems engineering process addressing the conceptual solution being out of mainstream Type II systems engineering. **Table 3** contains data extracted from Table 5 in (Honour and Valerdi, 2006) and rearranged in chronological order[8] showing the lack of coverage of the mission purpose/definition activities in MIL-STD 499 and ANSI EIA 632. The top row in **Table 3** has been added in this paper to show that MIL-STD 499 and ANSI EIA 632 do not cover the conceptual activities in the first systems engineering process and while the Systems Engineering Capability Maturity Model (CMM), the draft MIL-STD-499C Standard and ISO 15288 do address the mission/purpose definition activities to some extent they also do not cover the conceptual activities in the first systems engineering process. This situation (addressing mission/purpose definition activities to some extent while failing to cover the first systems engineering process) also appeared in a survey of current systems engineering processes in (Bruno and Mar, 1997) and in (Fisher, 1996)'s list of the engineering and systems engineering activities assigned to the systems

[8] Based on the issue date of MIL-STD-499, not the draft MIL-STD-499C since the contents of MIL-STD 499A and MIL-STD-499B don't differ from MIL-STD 499C in this respect.

Table 4 Types of Knowledge (Woolfolk, 1998)

Declarative knowledge	Knowledge that can be declared in some manner. It is "knowing that" something is the case. Describing a process is declarative knowledge.
Procedural knowledge	Knowing how to do something. It must be demonstrated; performing the process demonstrates procedural knowledge.
Conditional knowledge	Knowing when and why to apply the declarative and procedural knowledge.

engineering organization/team based on the (MIL-STD-499B, 1993)/(EIA 632, 1994) Standards.

A benchmark of systems engineering postgraduate degree syllabi

A benchmark of systems engineering postgraduate degree syllabi seems to indicate that:

- Much of systems engineering is now taught as declarative and procedural knowledge (See Table 4) (Woolfolk, 1998) describing the second systems engineering process. To be fair, this is not unique to systems engineering (Microsoft, 2008). 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.) Today's academic institutions seem to be producing Type II systems engineers and managers (engineer leaders); but they should be producing or at least identifying personnel with Type V characteristics by teaching conditional knowledge.
- Some academic institutions teaching systems engineering are leaving out the critical first systems engineering process of HKMF Column A. For example, a proposed reference curriculum for systems engineering (Jain and Verma, 2007) begins in Column B of the HKMF. This reference curriculum complies with (Martin, 1997) page 95), (Eisner, 1997) page 9), (Wasson, 2006) page 60) and (DOD 5000.2-R, 2002), pages 83-84) which consider requirements as one input to the systems engineering process as mentioned above. This failure to teach the critical first systems engineering process has resulted in (1) at least one generation of "systems engineers" who are unfamiliar with the critical activities in Column A of the HKMF and (2) the terms CONOPS and 'operations concept' being used interchangeably by some systems engineers who do not have an appropriate frame of reference to understand the difference between the two documents when old timers try to explain it to them.

Hypothesis for a reason for the failure of systems engineering

Based on a combination of the five types of systems engineers and the history of systems engineering paraphrased in terms of those five types, the hypothesis is that a current cause of failures in systems engineering is the assignment of Type II systems engineers or higher types trained in a Type II process thinking paradigm to tasks that need the problem/solution characteristics of the Type III, IV and V systems engineers. The associated prediction to test the hypothesis is that the cost and schedule overruns and other failures will continue in spite of all the funding being allocated to systems engineering education if the education of engineer leaders remains in the Type II paradigm and starts with the activities in column B of the HKMF. Type II systems engineers are and should be doing the engineering of systems (following the process designed by the Type V systems engineers). Type V systems engineers should be doing systems engineering in Columns A, B, F and G of the HKMF.

Recommendations

The following recommendations are made to improve systems engineering based on the research so far:

1. Continue with the development of the SEBoK creating a concept of operations for the product producing activities in each rectangle of the HKMF using the simple activity-based criterion to determine which of the activities are and which are not systems engineering and then defining the knowledge requirements for the activities known as systems engineering.
2. Investigate the potential of redrawing role boundaries to align with the activity boundaries and remove much of the role overlap and inefficiency in organisations. This approach, which needs more research, would truly reengineer the work of F. W. Taylor (Taylor, 1911).
3. Once the activities performed by systems engineers in each area of the HKMF have been identified, an appropriate level of competence for the activity should be made and optimal systems engineering teams could then be designed.
4. Work with psychologists to identify characteristics of the five types of engineer leaders so that the Type V's may be identified early in their career and put through fast track training to increase their value to their organizations.
5. Modify the curriculum for teaching systems engineering to include activities enabling the early identification of potential Type V's.
6. Modify the curriculum for teaching systems engineering to include the system engineering activities performed in Column A of the HKMF.
7. Develop a good set of educational materials for use with the modified curriculum.
8. Identify the activities performed in the non-systems engineering streams in each column of the HKMF. Then determine the knowledge and type of engineer leader needed to make optimal decisions and quantify the risks associated with decision making with specific levels of imperfect knowledge and using the wrong type of engineer leader. This model should inform customers concerning the prediction of the probability of future project failure at any point in any column of the HKMF by comparing the situation in a real project with the data in the model.

Summary

This paper documented the early phases of using systems engineering to develop a SEBoK and some of the findings. The first part of the paper discussed the nature of the problem and dissolved the problem by applying an out-of-the-box approach. The second part of the paper identified five types of systems engineers, discussed the evolution of systems engineering in terms of those five types, hypothesised that a major cause of the failure of systems engineering is the allocation of inappropriate types of systems engineers to systems engineering activities and identifed a critcal gap in systems engineering education. The paper concluded with some insights from the out-of-the-box approach and recommendations for further study.

Conclusion

The out-of the-box approach to developing the SEBoK seems to be achievable and has produced some interesting insights.

Biographies

Joseph Kasser has been a practicing systems engineer for nearly 40 years and an academic for about 10 years. He is an INCOSE Fellow, the author of "*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, is a Certified Manager and holds a Certified Membership of the Association for Learning Technology. He has also served as the initial president of INCOSE Australia and Region VI Representative to the INCOSE Member Board. 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 principal at The Right Requirement Ltd. in the UK and a Visiting Associate Professor at the National University of Singapore.

Derek Hitchins retired from full time academic work in 1994 on medical grounds, and is now a part-time consultant, teacher, visiting professor and international lecturer. Formerly, he held the British Aerospace Chairs in Systems Science and in Command and Control, Cranfield University at RMCS Shrivenham. Prior to that, he held the Chair in Engineering Management at City University, London. Derek started as a Cranwell apprentice and retired as a wing commander from the Royal Air Force after 22 years, to join industry. His first industry appointments were as the System Design Manager of the Tornado F3 Avionics, Technical Co-ordinator for UKAIR CCIS, and UK Technical Director for the NATO Air Command and Control System (ACCS) project in Brussels. He subsequently held posts in two leading systems engineering companies as Marketing Director, Business Development Director and Technical Director before becoming an academic in 1988. His current research is into system thinking, system requirements, social psychology & anthropology, Egyptology, command & control, system design and world-class systems engineering. He has published three systems engineering books: "Putting Systems to Work", John Wiley & Sons, in 1992; "Advanced Systems Thinking, Engineering and Management," Artech House, 2003; and, "Systems Engineering: A 21st Century Systems Methodology," John Wiley & Sons in 2007/2008. He inaugurated the IEE's PG M5 — Systems Engineering. He also started the UK Chapter of INCOSE and was its inaugural president. He is an INCOSE Fellow, an INCOSE "Pioneer" and a Charter Member of the Omega Alpha Association.

Thomas V. Huynh is an associate professor of systems engineering at the Naval Postgraduate School in Monterey, CA. His research interests include complex systems and complexity theory, system-of-systems engineering methodology, and systems architecting. Prior to joining the Naval Postgraduate School, Dr. Huynh was a Fellow at the Lockheed Martin Advanced Technology Center, Palo Alto, CA, where he engaged in research in computer network performance, computer timing control, bandwidth allocation, heuristic algorithms, nonlinear estimation, perturbation theory, differential equations, and optimization. He was also a lecturer in the departments of Physics and Mathematics at San Jose State University. Dr. Huynh obtained simultaneously a B.S. in Chemical Engineering and a B.A. in Applied Mathematics from UC Berkeley and an M.S. and a Ph.D. in Physics from UCLA. He is a member of INCOSE.

References

Ackoff, R. L., "The Future of Operational Research is Past," Journal of the Operational Research Society, Volume 30, 1979, in *Critical Systems Thinking Directed Readings*, R. L. Flood and M. C. Jackson (Editors), 1991.

Ackoff, R. L., *Ackoff's Best. His Classic Writings on Management*, John Wiley & Sons, Inc., New York, 1999.

Angel, C. D. and Froelich, J., Six Sigma: What Went Wrong?, 2008, http://www.destinationcrm.com/Articles/Columns-Departments/The-Tipping-Point/Six-Sigma-What-Went-Wrong-51394.aspx, last accessed 26 April 2009

ANSI/EIA-632, *Processes for Engineering a System*, American National Standards Institute and Electronics Industries Association, 1999.

Arnold, S. (Editor), *ISO 15288 Systems engineering — System life cycle processes*, International Standards Organisation, 2002.

Boehm, B., "*Integrating Software Engineering and Systems Engineering*", Systems Engineering, Vol. 1 (1994), no. 1.

Bottomly, P. C., Brook, P., Morris, P. W. and Stevens, R., "*A Study of the Relationship of Systems Engineering to Project Management*", proceedings of Fourth Annual Symposium of the INCOSE-UK, 1998.

Brekka, L. T., Picardal, C. and Vlay, G. J., "*Integrated Application of Risk Management and Cost of Quality*", proceedings of The 4th Annual International Symposium of the NCOSE, 1994.

Brill, J. H., "*Systems Engineering ---A Retrospective View*", Systems Engineering, Vol. 1 (1998), no. 4, 258-266.

Bruno, M. E. and Mar, B. W., "*Management of the Systems Engineering Discipline*", proceedings of the 7th Annual Symposium of the International Council on Systems Engineering, Los Angeles, CA., 1997.

Chapanis, A., "Human Engineering," in *Operations Research and Systems Engineering*, C. D. Flagle, W. H. Huggins and R. H. Roy (Editors), Johns Hopkins Press, Baltimore, 1960.

Denzler, D. and Kasser, J. E., "*Designing Budget Tolerant Systems*", proceedings of The 5th Annual International Symposium of the NCOSE, St Louis, MO., 1995.

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

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

DSMC, *Systems Engineering Management Guide*, vol. May 1996, Defence Systems Management College, 1996.

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

Eisner, H., *Computer Aided Systems Engineering*, Prentice Hall, 1988.

Eisner, H., *Essentials of Project and Systems Engineering Management*, John Wiley & Sons, Inc., 1997.

Emes, M., Smith, A. and Cowper, D., "*Confronting an identity crisis - How to brand systems engineering"*, Systems Engineering, Vol. 8 (2005), no. 2, 164-186.

Fisher, J., "*Defining the Roles and Responsibilities of the Systems Engineering Organization/Team"*, proceedings of the 6th Annual Symposium of the International Council on Systems Engineering, Boston, MA, 1996.

Frank, M., "*Knowledge, Abilities, Cognitive Characteristics and Behavioral Competences of Engineers with High Capacity for Engineering Systems Thinking (CEST)"*, The INCOSE Journal of Systems Engineering, Vol. 9 (2006), no. 2, 91-103.

GAO, "DEFENSE ACQUISITIONS Major Weapon Systems Continue to Experience Cost and Schedule Problems under DOD's Revised Policy," GAO, 2006.

GDRC, The Problem Solving Process, 2009, * http://www.gdrc.org/decision/problem-solve.html, last accessed 11 Jan 2009

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

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

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

Hall, C. S. and Lindzey, G., *Theories of Personality*, John Wiley & Sons, 1957.

Hammer, M. and Champy, J., *Reengineering the Corporation*, HarperCollins, New York, 1993.

Hari, A., Weiss, M. and Zonnenshain, A., "*ICDM - An Integrated Methodology for the Conceptual Design of New Systems"*, proceedings of System Engineering Test and Evaluation Conference SETE 2004, Adelaide, Australia, 2004.

Hiremath, M., "*Systems Engineering in Acquisition Strategy: Change Needed,"* INCOSE Insight, Vol. 11 (2008), no. 5, 32-33.

Hitchins, D. K., *Putting Systems to Work*, John Wiley & Sons, Chichester, England, 1992.

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

Hitchins, D. K., *Advanced Systems Thinking, Engineering and Management*, Artech House, 2003.

Honour, E. C. and Valerdi, R., "*Advancing an Ontology for Systems Engineering to Allow Consistent Measurement"*, proceedings of Conference on Systems Engineering Research, Los Angeles, CA., 2006.

Hudson, S. (Editor), *Systems Engineering Competencies Framework*, INCOSE UK, 2006.

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

INCOSE Fellows, A Consensus of the INCOSE Fellows, 2009, http://www.incose.org/practice/fellowsconsensus.aspx, last accessed 18 March 2009

Jain, R. and Verma, D., "*Proposing a Framework for a Reference Curriculum for a Graduate Program in Systems Engineering"*, INSIGHT, Vol. 10 (2007), no. 3, 9-11.

JAXA, *Basics of Systems Engineering (draft) , Version 1B*, 2007.

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

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

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

Kasser, J. E., "*Transition via Transactions: First steps in creating a customer driven organization"*, proceedings of First World Customer Service Congress, Tyson's Corner, VA, 1997.

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

Kasser, J. E., "*The Cataract Methodology for Systems and Software Acquisition"*, proceedings of SETE 2002, Sydney Australia, 2002a.

Kasser, J. E., "*Systems Engineering: An Alternative Management Paradigm"*, proceedings of The Systems Engineering Test and Evaluation Conference, Sydney Australia, 2002b.

Kasser, J. E., *A Framework for Understanding Systems Engineering*, Booksurge Ltd, 2007.

Kasser, J. E. and Mackley, T., "*Applying systems thinking and aligning it to systems engineering"*, proceedings of the 18th INCOSE International Symposium, Utrecht, Holland, 2008.

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.

Martin, J. N., *Systems Engineering Guidebook: A process for developing systems and products*, CRC Press, 1997.

Maslow, A. H., *The Psychology of Science*, Harper and Row, 1966.

Maslow, A. H., *Toward a Psychology of Being*, Van Nostrand, 1968.

Maslow, A. H., *Motivation and Personality*, Harper & Row, 1970.

Microsoft, Transforming Education: Assessing and Teaching 21st Century Skills,, 2008, http://download.microsoft.com/download/6/E/9/6E9A7CA7-0DC4-4823-993E-A54D18C19F2E/Transformative%20Assessment.pdf, last accessed 20 March 2009

MIL-STD-490A, *Specification Practices*, United States Department of Defense, 1985.

MIL-STD-499, *Mil-STD-499 Systems Engineering Management*, United States Department of Defense (USAF), 1969.

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

MIL-STD-499B, "Draft MIL-STD-499B Systems Engineering," United States Department of Defense, 1993.

Murray, H. A., *Explorations in Personality*, Oxford University Press,, 1938.

O'Toole, P., "*Do's and Don'ts of Process Improvement"*, proceedings of 2005 U.S. Census Bureau SEPG Conference, Suitland. MD, 2004.

OVAE, Problem-Solving Process, 2005, http://www.c-pal.net/course/module3/pdf/Week3_Lesson21.pdf, last accessed 11 Jan 2009

Pennell, L. W. and Knight, F. L., "Draft MIL-STD 499C Systems Engineering," The Aerospace Corporation, 2005.

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.

PMI, *A Guide to the Project Management Body of Knowledge (PMBOK Guide)*, Project Management Institute, 2004.

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

Roe, C. L., "*The Role of the Project Manager in Systems Engineering"*, proceedings of The 5th Annual International Symposium of the NCOSE, 1995.

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

Taylor, F. W., *The Principles of Scientific Management*, Harper & Brothers Publishers, 1911.

Todaro, R. C., "Lecture Handout, ENEE 648R," in, University of Maryland University College, 1988.

Wasson, C. S., *System Analysis, Design, and Development concepts, principles and practices*, Wiley-Interscience, Hoboken, New JErsey, 2006.

Watts, J. G. and Mar, B. W., "*Important Skills and Knowledge to Include in Corporate Systems Engineering Training Programs"*, proceedings of The 7th Annual International Symposium of the INCOSE, Los Angeles, CA, 1997.

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

Wymore, A. W., *Model-Based Systems Engineering*, CRC Press, Boca Raton, 1993.