

# Unifying systems engineering: Seven principles for systems engineered solution systems

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

Derek K. Hitchins  
ASTEM  
Consultant Systems Architect  
England

Copyright © 2011 by the Authors. Published and used by INCOSE with permission.

**Abstract.** Systems engineering is presently demonstrating the characteristics of being in the emerging stages of a discipline. A discipline generally matures when an overriding axiom is presented and accepted by the majority of practitioners. This paper presents one such high level underpinning axiom for systems engineering that has the potential to unite the disparate camps within systems engineering and enable the practice of systems engineering in all application domains to achieve successes similar to those it achieved in the National Aeronautical and Aerospace (NASA) environment in the 1960's and 1970's. The axiom does this by focusing on the solution system rather than on systems engineering.

## Introduction

Although systems engineering has been in existence since the 1940's, it is still demonstrating the characteristics of being in the emerging stage of a discipline. The characteristics of a discipline in this stage include application successes and failures as well as debates based on subjective opinions by participants in different camps talking past each other and a general lack of listening. The current somewhat overlapping camps in systems engineering include:

- Life cycle phase based
- Process-problem paradigms
- Discipline-enabler paradigms
- Systems thinking and non-systems thinking

Consider each of these camps:

**Life cycle phase.** Some systems engineers seem to have an understanding of the early stage systems engineering activities that take place in the concept definition stage of a solution system acquisition<sup>1</sup>. The majority however has no idea that the concept definition phase even exists, they don't understand what happens in that phase and they think that systems engineering in the acquisition domain begins with the requirements analysis phase. The early stage campers tend to be the old timers; while the others tend to be those systems engineers educated in the last 20-30 years in the paradigm based on the United States Department of Defence (DOD) where the whole set of activities performed in early stage systems engineering were removed from "sys-

---

<sup>1</sup> The solution system is acquired to remedy a problem.

tems engineering”<sup>2</sup>. In this paradigm requirements are one of the inputs to the ‘systems engineering process’, see (Martin, 1997) page 95; Eisner, 1997; Wasson, 2006) page 60; DOD 5000.2-R, 2002), pages 83-84) for typical examples.

**Process-problem paradigms.** Some systems engineers are process-focused: others focus on identifying the correct problem and realizing the best solution achievable within the constraints that exist at the time. The process-focused systems engineers’ mantra is to apply the systems engineering process and all will be well. These are the campers who tend to insist that organisations must modify themselves to follow a particular process standard in accordance with the (United States Department of Defense 5000 Guidebook 4.1.1), which states that *the successful implementation of proven, disciplined systems engineering processes results in a total system solution that is robust to changing technical, production, and operating environments; adaptive to the needs of the user; and balanced among the multiple requirements, design considerations, design constraints, and program budgets*. For example, (Arnold 2000) quoted (MIL-STD-499B 1993) and (IEEE 1220 1998) stating “*a single process, standardizing the scope, purpose and a set of development actions, has been traditionally associated with systems engineering*”. However, these campers can’t seem to see the big picture and don’t seem to realise that there is no single widely agreed upon systems engineering process since over the years, the systems engineering process has been stated in many different and sometimes contradictory ways, including:

- The (MIL-STD-499B, 1993), (EIA 632 1994) and (IEEE 1220 1998) processes;
- The lists of processes in ISO/IEC 15288 (Arnold 2002);
- The waterfall process (Royce 1970);
- The V model version of the process;
- The spiral, incremental and evolutionary models;
- System Lifecycle functions (Blanchard and Fabrycky 1981) ;
- State, Investigate, Model, Integrate, Launch, Assess and Re-evaluate (SIMILAR) (Bahill and Gissing 1998);
- The basic core concepts accepted by most systems engineers (Mar 2009);
- A systems engineering approach to addressing a problem (Hitchins 2007).

These campers also ignore the literature on excellence which focuses on people and ignores process; see (Peters, 1982; Peters, 1985; Rodgers, 1993) for typical examples.

The other camp focuses on the problem, identifying the best solution available given the constraints at the time, developing a process tailored to the occasion; see (Hitchins, 2007) for an example. Some of these campers also address carrying out that process to realize the solution system; see (Bahill and Gissing 1998) for an example.

**Discipline-enabler paradigms.** These camps are related to the process-problem gap. Systems engineering meets the requirement for a discipline proposed by (Kline, 1995, page 3) who wrote “a discipline possesses a specific area of study, a literature, and a working community of paid scholars and/or paid practitioners”. However, all these elements overlap those of project management and other disciplines. See (Jenkins, G. M., 1969; Brekka, et al., 1994; Roe, 1995; DSMC, 1996; Sheard, 1996; Johnson, 1997; Watts and Mar, 1997; Bottomly, et al., 1998) for

---

<sup>2</sup> . This removal was documented in 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.

just a few examples of the different overlaps between systems engineering and project management. Moreover, according to (Hall, 1962, page 20), (Goode and Machol 1957) make no distinction between ‘systems engineering’ and ‘engineering design’ or even ‘design’ and use the terms interchangeably.

The discipline camp preaches that systems engineering needs to widen its span to take over other disciplines. For example, the latest systems engineering standard ISO/IEC 15288 (Arnold 2002) contains the word ‘management’ in many of its processed. In the enabler camp, systems engineering is just a thinking tool that can be used in all disciplines for tackling certain types of problems; see “[*systems engineering*] is a philosophy and a way of life” (Hitchins, 1998) which can be considered as the application of holistic thinking in the workplace.

**Systems thinking and non-systems thinking.** The systems thinking camp tend to be systems engineers who can view an issue from a systems (several) perspective while the non-systems thinkers tend to have a single viewpoint. The non-systems thinkers also generally exhibit the ‘biased jumper’ level of critical thinking (Wolcott and Gray, 2003) and also tend to belong in the process paradigm. The systems thinkers generally also belong in the enabler paradigm.

## Towards unification

One reason for these debates and the camps is that systems engineers have different opinions on the nature of systems engineering. This is because:

- of the devolution of systems engineering over the past 60 years from a holistic paradigm to a stove-piped paradigm;
- systems engineering is so broad that systems engineers working on one part of the “systems engineering process” face different problems and perform different activities to those working in another part;
- they work in different application domains; and
- They often can’t see the big picture perspective of systems engineering

These different opinions of systems engineering can be represented by the situation portrayed in the parable about the blind men feeling different parts of an elephant and deducing different animals. The parable<sup>3</sup> which is told in verse in an Indian setting in (Saxe, 1873) pages 77 and 78) as quoted by (Yen, 2008) ends with the following stanzas.

“And so these men of Indostan  
Disputed loud and long,  
Each in his own opinion  
Exceeding stiff and strong,  
Though each was partly in the right,  
And all were in the wrong!

MORAL.

So oft in theologic wars,  
The disputants, I ween,  
Rail on in utter ignorance  
Of what each other mean,

---

<sup>3</sup> The parable is said to have originated in China sometime during the Han dynasty (202 B.C. – 220 A.D.).

And prate about an Elephant  
Not one of them has seen!”

Systems engineering will not and cannot re-emerge as a unified discipline until the majority of the practitioners understand the situation, realise the big picture, and progress past these debates. From the temporal perspective, a discipline matures when one or more underpinning axioms has been hypothesized, presented to, and eventually accepted by the community. This acceptance occurs either when the axiom can represent the views of all or nearly all of participants in the debates, or, when one view becomes the dominant paradigm over the course of time. This paper presents one such high level underpinning axiom for systems engineering; namely “seven principles for systems engineered solution systems”. While the axiom applies to systems engineering, the principles apply to the finished product irrespective of the systems engineering camp producing the solution system.

### **Seven principles for systems engineered solution systems**

The success of systems engineering in the NASA environment in the 1960’s and 1970’s was attributed to a set of eight principles summarized in (Hitchins, 2007) page 85). This paper states a hypothesis that systems engineers working in different domains using various tools, techniques and methodologies, can meet the objective of systems engineering by applying the following set of principles to the solution system they are realizing:

1. There shall be a clear, singular objective or goal.
2. There shall be a concept of operations (CONOPS) from start to finish of the mission describing the normal and contingency mission functions as well as the normal and contingency support functions performed by the solution system that remedies the problem.
3. The solution system shall be designed to perform the complete set of remedial mission and support functions for the operational life of the system.
4. The solution system design may be partitioned into complementary, interacting subsystems.
5. Each subsystem is a system in its own right, and shall have its own clear CONOPS, derived from, and compatible with, the CONOPS for the whole.
6. Each subsystem may be developed independently and in parallel with the other subsystems provided that fit, form, function and interfaces are maintained throughout.
7. Upon successful integration of the subsystems, the whole solution system shall be subject to appropriate tests and trials, real and simulated, that expose it to extremes of environment and hazards such as might be experienced during the mission.

Consider each of these principles.

#### **1. There shall be a clear, singular objective or goal.**

The task of the systems engineer shall have a clear singular objective goal. In the concept definition stage of a systems acquisition, this goal may be to identify the underlying problem or root cause of a situation, and to conceive one or more potential solutions. In the later phases of the solution system development lifecycle (SDLC)<sup>4</sup> the goal is generally to realize a solution system

---

<sup>4</sup> The notion that the solution system is generally a technological system that needs to be developed, and hence the name system development lifecycle, seems to be US Department of Defense inspired. Essentially, there need not be

that remedies the problem. For example, in the 1960's the NASA goal was to put a man on the Moon and return him safely to earth by the end of the decade. Similarly in the LuZ SEGS-1 system the goal was to provide a system that would convert solar energy to electrical power (Kasser, 2008).

**2. There shall be a clear concept of operations (CONOPS) from start to finish of the mission describing the normal and contingency mission functions as well as the normal and contingency support functions performed by the solution system that remedies the problem.**

The CONOPS documents the normal and contingency<sup>5</sup> performance of the overall solution system. The CONOPS is the foundation document<sup>6</sup> for the rest of the system realization activities since the remaining work in the SDLC realizes the solution system by converting the mission and support functions described in the CONOPS into a real system. Application of this principle leads to a holistic system development approach ensuring that all pertinent mission and support functions, such as operational availability, logistics, human operations, threat neutralizations, etc. are included in the system up-front in an integrated holistic manner and not as a bolt-on after the fact. A clear vision of the solution system anticipates subsequent activities that try to clarify the original customer's problem.

As an example, the CONOPS for the command and control system in Luz SEGS-1 (Kasser, 2008) was to generate electrical power using solar energy as the fuel. This mission was to be accomplished by deploying a field of parabolic trough reflector mirrors each morning, following the movement of the sun during the day to keep the mirrors focused on the sun and then stowing the mirrors in the evening when the sun set below the horizon. The support functions were to keep the mirrors clean, and to repair and maintain the elements of the system.

The CONOPS can also serve as a model of the solution and be incorporated in a simulation to allow various stakeholders to gain a better understanding of the problem space and determine if, and how well, the conceptual system being modelled could remedy the problem should that conceptual solution system be realized.

A CONOPS also facilitates elucidating requirements since the stakeholders, by agreeing on the CONOPS agree on the mission and support functions that the solution system will perform. The CONOPS can even minimize the number of requirements needed for certain types of systems such as LuZ's SEGS-1 (Kasser, 2008) because the CONOPS communicates the functionality and performance that must be developed and proven.

**3. The solution system shall be designed to perform the complete set of remedial mission and support functions for the operational life of the system.**

The application of this principle produces a solution system that performs the mission and support functions described in the CONOPS. The solution system does not have to be technological or even a new acquisition. The solution system lies somewhere along a continuum that stretches from 'fully automatic technological' to 'manual with no technology'; and may be a modification of an existing system, a change to an existing process, tactics, doctrine, policy, or training or

---

any (technological) development; instead, solution systems can be synthesized by bringing together existing systems to create a new unitary whole.

<sup>5</sup> The repeated use of "normal and contingency mission and support" is to emphasize the holistic approach.

<sup>6</sup> The word 'document' is used herein to represent information, not a paper document.

some combination. However, when applied to technological solution systems, this principle helps to ensure that the effects of component obsolescence, Diminishing Manufacturing Sources and Material Shortages (DMSMS), logistics, reliability, maintainability, the human element and other pertinent factors currently considered somewhat independently are considered interdependently in a holistic interdisciplinary manner from conception. Further, if the solution system is designed to perform in a hazardous or threatening context, then the solution system shall incorporate support functions to counter threats and to manage risks.

This principle takes into account changes in/to the need/problem at any point in the SDLC. For example, in the Apollo program, the need (and hence the requirements) did not change during the SDLC, and the operational life of each iteration of the manned element of the system was short; measurable in days. Each Apollo Lunar Surface Experiments Package (ALSEP)<sup>7</sup> however had a much longer life span.

Other early successful projects such as the transcontinental [United States of America] television microwave relay system (Hall, 1962) were also not subject to changing needs. However, today's solution system creation process must be able to cope with changes in the needs before the solution system is delivered, and the solution system itself needs be realized in such a manner that upgrades reflecting changing needs during its operational phase can be incorporated without major perturbations.

According to this principle, the cost-effectiveness of the solution system is not a design criterion – just effectiveness. Apollo was more concerned with doing the job at all, than doing it efficiently – money was not an issue (at least meeting the goal of placing a man on the moon by the end of the 1960's). When the systems engineer designs the solution system options, cost and schedule must not be an issue. Cost and schedule considerations may be used as selection criteria for choosing the desired solution system *after* the solution system options have been designed. In addition, systems engineers should be involved in any adjustments to the scope of the solution system realization project to fit the constraints of cost and schedule.

#### **4. The solution system design may be partitioned into complementary, interacting subsystems.**

The systems engineers design the solution system so that the desired functionality emerges from the complete design. For example, the performance of NASA's Apollo Moon Mission was emergent, coming as it did from the cooperation and coordination of the Saturn V launcher, the command module, the mission crew, the lunar excursion module, the telecommunications subsystem, mission control subsystem, etc. Performance is emergent because these various subsystems of the whole are of *dissimilar nature*, yet cooperate and coordinate their different functions and actions. So, you cannot point to any one subsystem and say – 'performance was down to that one'. All contributed, all cooperated and coordinated their actions.

#### **5. Each subsystem is a system in its own right, and shall have its own clear CONOPS, derived from, and compatible with, the CONOPS for the whole.**

This principle reflects the observation that systems exist within containing systems and incorporates the hierarchical perspective into the system. The principle has often been stated as "one person's system is another person's subsystem". Hierarchies are fundamental to nature. "All

---

<sup>7</sup> A set of scientific instruments deployed at the landing site of Apollo 12 to 17 designed to operate for a year. Each ALSEP contained the same central station and a slightly different set of scientific instruments.

*complex structures and processes of a relatively stable character display hierarchical organisation regardless of whether we consider galactic systems, living organisms and their activities or social organisations*"(Koestler, 1978) page 31). (Koestler, 1978) page 32) further quotes (Needham, 1945) as "*once we adopt the general picture of the universe as a series of levels of organisation and complexity, each level having unique properties of structure and behaviour, which, though depending on the properties of the constituent elements, appear only when those are combined into the higher whole, we see that there are qualitatively different laws holding good at each level.*" This concept of hierarchy is not a new concept, for example the following nursery rhyme<sup>8</sup> is closely based on lines by Jonathan Swift from his long satirical poem "On Poetry: a Rhapsody" (1733),

Big fleas have little fleas,  
 Upon their backs to bite 'em,  
 And little fleas have lesser fleas,  
 and so, ad infinitum.  
 And the great fleas, themselves, in turn  
 Have greater fleas to go on;  
 While these again have greater still,  
 And greater still, and so on.

As an example consider an allied naval convoy crossing the North Atlantic Ocean in 1942. The convoy is a system. Each ship in the convoy can be considered as both a subsystem of the convoy, or as a system<sup>9</sup>. There was a CONOPS for the convoy. There were separate CONOPS for the naval escort ships and the merchant vessels describing the actions and interactions of these subsystems of the convoy in various scenarios.

**6. Each subsystem may be developed independently and in parallel with the other subsystems provided that fit, form, function and interfaces are maintained throughout.**

Each subsystem, being a system, needs its own systems engineers who conceive, design and develop their [sub]system as an interacting part of the containing system. These [sub]system systems engineers face in two directions – upwards and outwards into the containing system, to ensure on-going compatibility with the containing system and its CONOPS, including all of the other interacting subsystems at the same level in the hierarchy; and downwards, into the interacting sub-subsystems within their own [sub]system. The downwards task of developing the subsystems (function) can be considered as engineering when the focus is on the [sub]system as an independent entity.

During the phases of the SDLC when the subsystems are being developed in parallel, the systems engineering activities are those that focus on the subsystem as a part of the system and ensure that fit, form and interfaces are maintained. If the SDLC takes a long time, the effect of changes in the need on the subsystem realization has to be taken into account. Experience has shown that subsystem designs and development may be subject to “creep”. Consequently, it is necessary to have budgets for the whole system, as well as budgets for each of the subsystems — for instance

---

<sup>8</sup> The Siphonaptera, Wikipedia, accessed 14/2/2011

<sup>9</sup> Alternatively, the naval ships could be one subsystem and the merchant marine ships a second subsystem of the convoy. Each ship is then a subsystem within the naval or civilian subsystem of the convoy. If there are ships from the navies of more than one allied country in the convoy, then the ships of each country could constitute a subsystem within the naval subsystem. As stated in the text, one person's system is another person's subsystem.

the weight budget was important to Apollo, as was a failure rate budget. It would not have done for the failure rate for one subsystem – say the capsule – to go off the scale! This is what is meant, in part, by conceiving, designing and developing the subsystem independently but within the context of the whole and the other interacting subsystems.

7. **Upon successful integration of the subsystems, the whole solution system shall be subject to appropriate tests and trials, real and simulated, that expose it to extremes of environment and hazards such as might be experienced during the mission.**

This principle minimizes situations in which solution systems are delivered that are not fit for purpose and do not provide a solution in the intended environment.

## Discussion

Poor systems engineering has been blamed for system acquisition failures<sup>10</sup> according to many sources including (Wynne, 2004). An objective view might suggest that budget and time overruns smack of either poor estimating of cost and schedules or understating the real estimates for reasons that appeared valid at the time. However, in all fairness, poor early stage systems engineering does seem to have been a contributor to some of those failures resulting from producing solutions systems that do not remedy the need when deployed. Attempts to mitigate the effects of poor early stage systems engineering in the early stages of the system have resulted in system development becoming increasingly technologically focused, excessively complicated and stove-piped into independent streams of activities including:

- Systems Engineering
- Project Management
- Life Cycle Costing or Total Ownership Cost
- Performance Based Logistics
- Integrated Logistics Support
- Maintenance Management
- Supply Chain Management
- Technical Training Management
- Technical Data Management
- Configuration Management
- Risk Management
- Independent Verification and Validation
- Human Systems Integration

These are but some examples of the independent streams of activities in the various specialties in the SDLC. Not only is this stove-piping against the holistic concept of systems engineering, stove-piping produces overlapping activities, confusion, and unnecessary expense and also provides a breeding ground for turf wars in organizations. The documentation overhead is increasingly becoming expensive and documents that should be interdependent are independent being produced because of legislation rather than as a result of actual need. Today's DOD systems engineering paradigm has added so many bolt-on's to compensate for having removed the front end of systems engineering that it has become expensive and unworkable (Costello, 1988). Reversion to the original holistic *weltanschauung* (world view or paradigm) is long overdue since in the 20

---

<sup>10</sup> Defined herein as cost and schedule overruns, cancellations and delivered systems that are not fit for purpose.

years since the Costello Report was published, the situation has worsened.

## Summary

Systems engineering is presently demonstrating the characteristics of being in the emerging stages of a discipline. A discipline generally matures when an overriding axiom is presented and accepted by the majority of practitioners. This paper presents one such underpinning axiom for systems engineering. The principles within the axiom apply to the solution system, production of which is the common goal of all the camps within systems engineering. As a consequence, the axiom has the potential to unite the disparate camps within systems engineering by allowing them to agree on the principles applying to the solution system which will then enable the practice of systems engineering to repeat the successes it achieved in the NASA environment in the 1960's and 1970's in all current and future application domains.

## Conclusion

The principles presented in this paper apply to the solution system being systems engineered rather than to systems engineering. As such the axiom has the potential to unite the disparate camps within systems engineering by allowing them to agree on the principles. Applying these principles to the solution system will then enable the practice of systems engineering to repeat the successes it achieved in the NASA environment in the 1960's and 1970's in all current and future application domains.

## References

- Arnold, S., "Systems Engineering: From Process towards Profession", proceedings of the 10th Annual Symposium of the INCOSE, Minneapolis, MN, 2000.
- Arnold, S., Ed. (2002). *ISO 15288 Systems engineering — System life cycle processes*, International Standards Organisation.
- Bahill, A. T. and B. Gissing, "Re-evaluating systems engineering concepts using systems thinking." *IEEE Transaction on Systems, Man and Cybernetics, Part C: Applications and Reviews*, Vol. 28 No.4, pages 516-527, 1998.
- Blanchard, B. B. and W. Fabrycky., *Systems Engineering and Analysis*, Prentice Hall, 1981.
- 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.
- Bottomly, P. C., Brook, P., Morris, P. W. and Stevens, R., "A Study of the Relationship of Systems Engineering to Project Management", proceedings of the Fourth Annual Symposium of the INCOSE-UK, 1998.
- Costello, R. B., "Bolstering Defense Industrial Competitiveness: Preserving Our Heritage, the Industrial Base Securing Our Future," Office of the Undersecretary of Defence (Acquisition), Washington DC, 1988.
- 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.
- DSMC, *Systems Engineering Management Guide*, Defence Systems Management College, 1996.
- EIA 632, EIA 632 Standard: Processes for engineering a system, 1994.

- Eisner, H., *Essentials of Project and Systems Engineering Management*, John Wiley & Sons, Inc., 1997.
- Goode, H.H. and Machol, R.E., *System Engineering-an Introduction to the Design of Large-Scale Systems*, McGraw-Hill Book Co., Inc., New York, 1957.
- Hall, A. D., *A Methodology for Systems Engineering*, D. Van Nostrand Company Inc., Princeton, N.J., 1962.
- Hitchins, D. K., "Systems Engineering...In Search of the Elusive Optimum", proceedings of Fourth Annual Symposium of the INCOSE-UK, 1998.
- 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.
- Jenkins, G. M., "The Systems Approach," in *Systems Behaviour*, J. Beishon and G. Peters (Editors), Harper and Row, London, page 82, 1969.
- Johnson, S. B., "Three Approaches to Big Technology: Operations Research, Systems Engineering, and Project Management", *Technology and Culture*, Vol. pages 891-919, 1997.
- Kasser, J. E., "Luz: From Light to Darkness: Lessons learned from the solar system", proceedings of the 18th INCOSE International Symposium, Utrecht, Holland, 2008.
- Kline, S. J., *Conceptual Foundations for Multidisciplinary Thinking*, Stanford University Press, Stanford, 1995.
- Koestler, A., *JANUS A Summing Up*, Random House, New York, 1978.
- Lake, J. G., "Axioms for Systems Engineering", *The Journal of NCOSE*, Volume 1 Number 1, July-September 1994.
- Mar, B., Commentary on the Consensus of INCOSE Fellows, INCOSE, 2009.
- Martin, J. N., *Systems Engineering Guidebook: A process for developing systems and products*, CRC Press, 1997.
- MIL-STD-499B, Draft MIL-STD-499B Engineering Management, United States Department of Defense, 1993.
- Peters, T. and N. Austin. *A Passion for Excellence*, Warner Books, 1985.
- Peters, T. J. and H. R. Waterman., *In Search of EXCELLENCE*, Harper and Row, 1982.
- Rodgers, T. J., W. Taylor, et al., *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.
- Royce, W. W. *Managing the Development of Large Software Systems*. Proceedings of IEEE WESCON, 1970.
- Saxe, J. G., *The Poems of John Godfrey Saxe, Complete edition*, James R. Osgood and Company, Boston, 1873.
- 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.
- Wolcott, S. K. and Gray, C. J., *Assessing and Developing Critical Thinking Skills*, 2003, [http://www.wolcottlynch.com/Downloadable\\_Files/IUPUI%20Handout\\_031029.pdf](http://www.wolcottlynch.com/Downloadable_Files/IUPUI%20Handout_031029.pdf).
- Wynne, M. W., "Policy for Systems Engineering in DoD," Memo by Undersecretary of Defense for Acquisition, Technology, and Logistics, Washington, DC, 2004.

Yen, D. H., *The Blind Men and the Elephant*, 2008,  
[http://www.noogenesis.com/pineapple/blind\\_men\\_elephant.html](http://www.noogenesis.com/pineapple/blind_men_elephant.html), last accessed 26 October 2010

## Biographies

**Joseph Kasser** has been a practicing systems engineer for 40 years and an academic for about 10 years. He is a Fellow of the Institution of Engineering and Technology (IET), 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. He is a Certified Manager and holds a Certified Membership of the Association for Learning Technology. He has also served as the inaugural president of INCOSE Australia and as 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 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.