The Voice of the Systems

The Journal Of The Israeli Systems Engineers

ISSUE NO.22 | June 2018

- Software & Systems Engineering Interplay and the SEMAT Kernel (pp. 6-20)
  Marcel Jacques Simonette and Edison Spina

- Lateral Thinking: A Creative Method Example (pp. 21-34)
  Avner Engel

- Four Layers Approach for Developing System Thinking Assessment Tool (pp. 35-46)
  Moti Frank and Sigal Koral Kordova

- The Digital Transformation and its Impact on the Working Models of the Systems Engineers and Analysts [Abstract] (pp. 47-48)
  Raz Heiferman

www.iltam.org
# Table of Content

**The Voice of the System | Issue No.22 | June 2018**

## INTRODUCTION

<table>
<thead>
<tr>
<th>The Voice of the Editor</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amir Tomer</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>The Voice of INCOSE_IL President</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ram Oron</td>
<td></td>
</tr>
</tbody>
</table>

## ARTICLES

<table>
<thead>
<tr>
<th>Software &amp; Systems Engineering Interplay and the SEMAT Kernel</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Marcel Jacques Simonette and Edison Spina</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Lateral Thinking: A Creative Method Example</th>
<th>21</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avner Engel</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Four Layers Approach for Developing System Thinking Assessment Tool</th>
<th>35</th>
</tr>
</thead>
<tbody>
<tr>
<td>Moti Frank and Sigal Koral Kordova</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Raz Heiferman</td>
<td></td>
</tr>
</tbody>
</table>

## JOINING AND INFORMATION

<table>
<thead>
<tr>
<th>INCOSE_IL 2019 Conference - Save the Date</th>
<th>49</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>The program of the Systems Engineering Seminar at the Gordon Center, 19/6/2018</th>
<th>50</th>
</tr>
</thead>
</table>

The Hebrew section is on the opposite side of the journal

חלקי העבר phải על העמוד השני מצא מדיו השמיני
The Gordon Center for Systems Engineering Seminar, which is conducted concurrently with the publication of this, the 22nd issue, is dedicated this time to System Thinking and what Systems Engineering can learn from other disciplines. Accordingly, the articles in the issue are concerned with the same topic.

During recent years, a big group of people, led by Ivar Jacobson, are working on a holistic definition of Software Engineering, which will be able to host every existing and future approach and methodology for software development. The approach, known as The Essence of Software Engineering, or SEMAT (Software Engineering Method and Technology) has also been recognized as an OMG standard and is currently undergoing the edition of a text book as well as documentation of case studies and examples. In the meantime, it was recognized that the approach is very applicable to other engineering disciplines, and mainly Systems Engineering, and is currently being extended. The article by Simonette and Spina presents the way SEMAT may be employed as an important bridge between Software and Systems Engineering.

A new book, dealing with creativity and innovation in Systems Engineering, has been recently published by Wiley. The author of this book, Dr. Avner Engel from Tel-Aviv University, presents in his article, which is based on parts of his book, the concept of Lateral Thinking and its importance for Systems Engineers.

Beyond the recognition of its importance, the concept of System Thinking is still intensively researched, in order to understand its ingredients, and to find out what is the set of characteristics whose collection may define a human as having System Thinking. This issue has crucial importance in the recruitment and employment of Industrial and Systems Engineers. Frank and Kordova present in their article a four-layer tool for the assessment of System Thinking for such engineers.

Raz Heiferman, a very experienced expert in the field of information systems, has been researching for several years the phenomenon of the Digital Revolution and its impact of Organizations and Processes. His article in this issue addresses the digital maturity of an organization, i.e. its readiness to deploy a digital transformation in its structure and behavior.

Finally, please allow me to express my deepest thanks to the presidencies of INCOSE_IL and ILTAM for electing me as the chair of INCOSE_IL 2019 conference, which will be conducted next March. Later in the issue you will find a personal invitation from me to save the date and mark it in your calendars.

I wish you pleasant reading.

Prof. Amir Tomer, CSEP
Kinneret College
The Editor
amir@amirtomer.com
Dear Systems Engineering Community,

One of our focus subjects in the last year was system engineering leadership. Traditional systems engineering training programs are generally focused on technical and professional aspects, whereas leadership training is saved for managerial programs. Today, there is an increased need for system engineers that can take end-to-end ownership of activities and processes, while leading teams, even without clear authority. This need is pronounced in projects with only a few system engineers. One could say that leadership capabilities are critical for systems engineers’ success and development.

Last March we held, for the first time, a 3-day Systems Engineering Leadership Workshop. We had more than twenty participants from about ten different companies, along with mentoring by leading system engineers. As a mentor that closely followed the workshop, I was happy with the great atmosphere of sharing and learning. Also, we received excellent feedbacks from the participants. Following the success, we are planning another workshop later this year.

Soon we shall hold (again, for the first time), a Human Factors Engineering for Systems Engineers workshop. This subject was initiated by community members, inspired by INCOSE’s HSI concept, evolved into a working group with academia and industry members, and would now be more widely exposed in a workshop. This is a good example of our opportunities to initiate and create new connections for our community.

We have initiated our work towards the 10th International Israeli System Engineering Conference, which will take place in Herzlia, in March 2019. Per our request, Prof. Amir Tomer, who is also the editor of our journal in the last years, has agreed to serve as the Chair of the conference. I am confident that under Prof. Tomer’s professional leadership, and with our full support, we shall hold a high quality conference that reflects the high level of systems engineering in Israel. Those who are interested in overseas conferences, can also participate in the INCOSE International Symposium that will take place in Washington in July, and the European EMEASEC 2018 Conference in Berlin in November.

Sincerely,

Ram Oron, Ph.D.
President, INCOSE IL
ram.oron@orbotech.com
Software & Systems Engineering Interplay and the SEMAT Kernel

Marcel Jacques Simontette and Edison Spina

Universidade de São Paulo, São Paulo, Brazil

marceljs@usp.br • spina@usp.br

This article is reprinted from:

Copyright © 2013, by WSEAS Press - reprinted by permission

Abstract

Software systems development teams, using agile practices, have a bias to focus on simpler solutions in early iterations and increments of a software system development process. Usually, requirements related to the system as a whole, as dependability, are considered requirements that are treated in the last iterations cycles. Without Systems Engineering concepts, these teams discovery too late that there is not refactoring to make up the initials choices. The interplay between Software Engineering and Systems Engineering must be increasingly present in organizations that develop software systems. System Engineering approach helps the software development team to coordinate their work processes and steer the team to the development of software systems with quality and value to the business that demands the system. This paper is about the Endeavor dimension of the SEMAT Essence kernel, and how we can work with it to reach the interplay between Systems and Software Engineering.

Keywords: Software engineering • System engineering • Essence kernel • Software process • Socio-technical, complexity
1. Introduction

There are several organizations in which the development of software systems is not the primary business; nonetheless, these companies have a department or area that works with Information Technology (IT). When the IT department of these organizations receives a demand from some other organization business area, a team is formed to perform the software system development (SSD) endeavor. However, the IT department of this kind of organizations was not created to innovate, to create innovative software systems for each business demand; they were created to be efficient [1]. Moreover, it is a commonplace the fact that these IT departments also have to deal with demands that are related to others software systems that are already in operation, and also with business pressures to reduce costs and delivery times.

When the IT department defines a team to the SSD endeavor, it is necessary to define a way of work that allows the team members integration and cooperation. A way of work that steers the understanding of what must be implemented, the implementation itself, and the validation of the system that is delivered. The team coordination must be done with a focus on the delivery of a software system that achieves both the customer needs and expectations. This kind of mantra of SSD management can be reached by way of work that promotes a unique understanding of what must be delivered, considering both the functional and non-functional aspects of the business demand. Furthermore, the way of work must have a quick and effective reaction to the changes that are imposed to the software systems requirements; changes originated by market forces that act upon the organization business.

Some of the factors that characterize the differences between organizations of the same business sector are the qualities of their software systems. This characteristic requires that IT departments seek innovation and flexibility of their processes and software systems, which brings complexity to both the software system as the development process itself.

Systems Engineering offer practices that can enable the integration and collaboration among IT team members, which enable the development of a
single vision of the different parts of a system to be developed [2]. It is the vision of the software system as a whole that promotes the identification of the system dependability issues [3], [4] at the beginning of iterations of the work of the SSD endeavor.

Work, Team, and Way of Work are present in one of the three areas of concern that were identified in the SEMAT kernel. They take part of the things that we always deal with in Software Engineering. They are the alphas of the SEMAT kernel Endeavor context [5].

SEMAT kernel is the first answer to a “Call for Action” [6], a little step in the process of redefining the Software Engineering. This paper is about Systems Engineering through the life cycle of software systems, supporting Software Engineering practices with a socio-technical approach that promotes the integration of the people that take part of an SSD endeavor, looking for a knowledge construction about the system as a whole, since the early cycles of system integration. Barry Boehm has also highlighted the importance of Systems and Software Engineering interplay in several articles, as [7], [8] & [9]. Boehm argues that without up-front Systems Engineering and teambuilding, two common failure modes occur: a leadership scarcity of the agile team, and the bias to focus on easiest-first solutions in the early iterations; an approach to treat dependability issues, like scalability and security, as features to be incorporated in later iterations, when no amount of refactoring will compensate the early choices.

This approach contributes to enabling IT team to develop software systems that add value to the organizations, contributing to organization performance and efficiency.
2. SEMAT Kernel - The Essence

An Opportunity identified by a business area of an organization is a search for a solution to the IT department of this organization. A solution in which the people that work in the SSD process make use of Software Engineering and System Engineering practices to understand what must be done, validate this understanding, and implementing the solution. Although there are several practices to conduct this process, there is a set of essentials elements that are universal and are present in all the endeavors to develop high-quality software systems. These elements - called alphas in SEMAT kernel - deal with “things we always work with” and “things we always do” in an SSD process. These elements provide an integrated set that enables the predictability of the delivery, and the continuous improvement in the way of work, communication, and understanding of what, and how, the team is working in a moment, and what they must to do in the following project increments [5].

SEMAT kernel does not compete with existing SSD methods. It is not a new methodology to SSD. It is an answer to the need of redefining Software Engineering [10]. Essence is a kernel of universal elements that are both present in several SSD methods and that enable the measurement of the project progress and health. The kernel was first published in the SEMAT submission to OMG call: Foundation for Agile Creation and Enactment of Software Engineering [11]. Figure 1 and figure 2 represent the “things we always work with” and “things we always do” in an SSD process. More than a conceptual model, the kernel provides:

I. A framework for SSD teams to think about the endeavor progress and the state of their efforts.

II. A common basis to discuss, improve, compare, and share Software Engineering best practices and methods.

III. A framework to SSD teams assemble practices from different origins, continuously improving the way of work.

IV. A framework for defining metrics that are independent of practices, to evaluate both the quality of developed software and the methods used to develop it.
V. And the most important, a way to help SSD teams to understand where they are, what they should do next, and where they need to improve.

2.1. Endeavor alphas

Alphas are representations of the essential elements that must be monitored to guide the success of SSD process. Each alpha has a set of states, and each state has a checklist that identifies whether the state has been met. Alphas must be seen together, not in an approach that individually considers each alpha [5].

The kernel alphas that belong to the Endeavor area of concern are Work, Way of Work, and Team. Observing the Endeavor kernel dimension at Figure 1, Team performs and plan the Work, and that the Team applies Way of Work, which guides the Work (Fig. 3).

2.2. Endeavor kernel dimension and the inherent software complexity

IT team members are professionals that are under pressure both to be efficient and to produce software systems that meet stakeholders need; they must address the identified Opportunity. There are different ways of work that these professionals can use to fulfill their mission successfully. However, in this endeavor, they must not forget that the Work to be performed depends on understanding the requirements of the problematic situation that appears as an Opportunity. Whichever the Way of Work applied by the Team, there is an issue present in all SSD endeavors: the identification of the problematic situation, the Opportunity, and the Requirements that such problematic situation brings. This identification process is primarily an activity related to people talking to people [12], [13], [14].
Figure 1. “Things we always work with” in SSD process.

Figure 2. “Things we always do” in an SSD process.
Team member's deal with information and knowledge obtained through interactions with Stakeholders to understand the Opportunity and Requirements. In this process, and especially in the implementation of the software system, these team members need to work in a concentrated way to deal with the underlying complexity of the SSD process [15], [16]. This complexity has also been highlighted by Frederick Brooks in his classic 1987 article “No Silver Bullet: Essences and Accidents of Software Engineering” [17], in which he points out that due to a large number of distinct elements that are present in the SSD process, the development of this type of system is an activity that is more complex than any other type of man-made construction. As evidence of the number of elements that exist in the SSD process, Clarke & O’Connor [18] argue that there are 40 factors and 170 sub-factors that are present in SSD context.

![Figure 3: The alphas of the kernel Endeavor area of concern and their interrelationships.](image)

**Figure 3:** The alphas of the kernel Endeavor area of concern and their interrelationships.

### 2.3. Endeavor dimension and its relationships

**Figure 4** shows all the relationships of the SEMAT kernel alphas that belongs to the Endeavor area of concern:

**I.** Work has relationships with Requirements - Opportunity - Software System.

**II.** Team has relationships with Software System - Stakeholder.

**III.** Way of work has relationships only with alphas of the Endeavor dimension.
Nevertheless, it is an essential link between Team and Work and is under the direct influence of the factors mentioned by Clarke & O’Connor [18].

Despite the apparent simplicity of the relations between the alphas shown in Figure 4, there is a challenge in these relations. A challenge related to the cooperation among Team members, and between these people and stakeholders. This relationship is essential to IT team realize a work that adds value to the organization.

3. Systems Engineering

The Cartesian way of dealing with a problem is to divide it into smaller and simpler parts, as much as possible; it is the most successful technique used by Engineering, and this approach enables humanity to reach the current state of technology. However, even with the successes of this approach, there are problems in which this approach cannot succeed; System Engineering uses System Thinking to deal with these problems. Hitchins [2] argues that System Thinking caught the attention of engineers when they realized that the Cartesian approach has difficulties to deal with systems that include people. According to Peter Senge [19], Systems Thinking is a discipline to see a system in its totality, a kind of framework to view the system components interrelationship more than to view the components, to view the system patterns of change more than to view system static images.

Hitchins [2] states that Systems Engineering uses Systems Thinking to understand the nature of the problems. This approach looks for practical experiences and interactions with the problem, trying to understand it, and proposes solutions that may not solve it, but that can provide a response that improves the problem understanding, allowing the development of a solution which is the best at the moment.
Figure 4 - Alphas of kernel Endeavor dimension and their relationship with other kernel alphas.

Using this Systems Engineering approach, that is more qualitative than quantitative, and that make use of experience, IT team members can develop a Way of Work that enables them to deal with SSD problems. To implement this approach IT team can use the Lean Startup cycle [20] in the contrary course:
The construction of a minimal version of the software system is the implementation of a response to an Opportunity that is the best possible solution at the moment. The relationship between the Team and the Stakeholders to build this minimal version of the software system leads to an evolution of the knowledge about the Opportunity, a learning experience that allows the evaluation and evolution of the software system, and a new version of the software system can be developed. This approach gives rise to an evolutionary spiral of knowledge and systems in which each turn adds more value to the organization.

The complexity of people interactions in these cycles of Learn (knowledge) -> Measure (feedback) -> Build (System Construction), and the technology used by this people, make up a kind of patchwork of technology and people that give rise to a Sociotechnical System [21] that kernel Endeavor dimension must take into account.

Usually, the Team applies a Way of Work that focuses on developing the technical part of the software system, concentrating on the software issue, and the Team assumes that the requirements reflect the business need. This technical approach results in a software system that works as the Team thinks.
that it must work, and that do not take into account the stakeholders culture and experience.

4. The Lean Startup Cycle on the Contrary Course

When following an approach like the evolutionary spiral of knowledge developed by doing the Lean Startup cycle on the contrary course (Fig. 5), Team members have the opportunity to use System Thinking framework of System Engineering to learn through the knowledge and experience of the stakeholders since the early iterations. This approach makes it possible to have, in the sequence of iterations, software system versions more effective and resilient. At each cycle, since the early ones, the Team can incorporate more treatments for dependability requirements, which arises from the interrelationships that exist not only among the technical components of the software system but also between these components and the people that interact with the software system.

5. Conclusion and Future Work

It is impossible to predict the dependability issues that a software system will demand. It is not possible to leave to the last iterations cycles the evaluation of these issues; when the Team technical debt [22] about dependability issues is already at a point in which the refactoring is not possible to be done to meet the identified need [7].

The different fashions of Software Engineering methods and practices do not deal effectively with the inherent complexity of SSD because they are based on code development, without considering the inter-relationship between:

I. Technical components (hardware & software).

II. Issues related to the organization or the business process.

III. Issues related to the relationship between the team members.
IV. Issues related to the relationship between team members and stakeholders.

V. The real intentions of the users using the software system.

The effective interplay between Systems Engineering and Software Engineering is a critical condition to enable IT department to add value to the organization business, and no longer be seen as a cost of operation for organizations in which the software development is not the core business. The System Engineering systemic approach enables the IT team to develop an understanding of what has to be developed, considering the various inter-relationships, and inspire a socio-technical environment in which the IT team objective is more than build a software system, is to add value to the organization business.

The authors of this paper are conducting researches about extensions both to SEMAT kernel alphas and to the alpha states checklists, to offer an answer that considers the inherent complexity of SSD process, in which dependability has a strong presence. ■
References:


About the authors

**Marcel J. Simonette** received the B.S. degree in 1991, the M.Sc. degree in 2010, and the Ph.D. degree in 2015. All titles were obtained in Escola Politécnica (Engineering School) of University of São Paulo (EP-USP), São Paulo, Brazil. Currently, he is a researcher at Knowledge Engineering Laboratory (KNOMA) - Department of Computer and Digital Systems Engineering (PCS) of EP-USP. Also, he is a collaborator on Study Center of Society and Technology (CEST- Centro de Estudos Sociedade e Tecnologia in Portuguese). He has experience with European Commission Projects experience as a team member in BELIEF (FP6), BELIEF 2 (FP7), and VertbrALCUE (Alfa3). Dr. Simonette research areas are Software Engineering, Requirements Engineering, System Engineering, and Sociotechnical Systems.

**Edison Spina** received the Ph.D. degree in Electronic Engineering at Escola Politécnica (Engineering School) of University of São Paulo (EP-USP). M.Sc. degree in 1990 and B.S. degree in 1981, both from EP-USP. He is Assistant Professor - Department of Computer Engineering and Digital Systems at EP-USP - since 1988. He is Freelancer Advisor in Electromagnetic Interference and Compatibility since 1987 and Auditor Leader ISO 9000 since 1995. From 1986 to 2000 he was Director of ANTARES ELETRÔNICA. Also, he was Project Manager at the Foundation for Engineering Technological Development - FDTE, and Electronic Systems Safety Analyst at LCA and GAS labs of the Department of Computer Engineering and Digital Systems at EP-USP. He has European Commission Projects experience as Task Leader in INSTINC Project (FP6), and as Brazilian team coordinator for BELIEF (FP6), BELIEF 2 (FP7), VertbrALCUE (Alfa3), eMundus, and SMART2 (Erasmus Mundus).

Prof. Dr. Spina is member of the KNOMA - Laboratory of Knowledge Engineering, Academic Coordinator of the Study Center Society and Technology of USP, Member of the Committee for International Relations of the EP-USP, and Council Member of the Committee of Sciences and Technology of OAB (Brazilian Bar Association).
Lateral Thinking: A Creative Method Example
Avner Engel, Systems Engineering Research Institute -
Tel Aviv University (TAU-SERI)
✉ aengel1000@gmail.com

Abstract
The purpose of this paper is to acquaint systems engineers and engineers in general, with the subject of Lateral Thinking (LT). This is done by way of a short example comparing conventional thinking with six lateral thinking techniques. LT is the brainchild of Edward de Bono offering a powerful way to think outside the box in a creative way.

In general, LT employs six strategies as follows: (1) Mastering the problem at hand, (2) Asking fundamental questions, (3) Recognizing and controlling assumptions, (4) Breaking the rules, (5) Broadening one's perspectives, and (6) Using provocative ideas.

This paper was adopted from the authors' 2018 book: Practical Creativity and Innovation in Systems Engineering, published by Wiley, as part of Wiley's Systems Engineering and Management Series.

1. Introduction
Education at the school systems, from the elementary to the university levels, instructs children and adults to think vertically. That is, to seek logical solution to problems on a step-by-step basis, identify a particular solution to a problem, and then explore the ramifications of that solution. A good example of vertical thinking is embodied in the way people play chess. At any time, one knows explicitly the positions of pieces involved in the game as well as the exact rules governing the game. On this basis
one projects his or her future moves against any potential moves of the opponent. However, this is seldom the case in real life, and students are rarely taught other, overarching strategies to solve problems in this chaotic, unstructured world.

Lateral thinking (LT), the brainchild of Edward de Bono (1967), is a method for solving problems by way of an indirect and creative approach, using reasoning that is not immediately obvious. By and large, LT involves the generation of "horizontal" imagery, encouraging the generation of multiple solutions while ignoring, at least initially, their detailed implementation. LT exposes one’s hidden assumptions and inner boundary conditions, bringing new insights into the problem-solution process. It penetrates through the confines of obsolete ideas, ushering creativity by generating new patterns. In short, LT is based on rearrangement of one’s mental information. This, in turn, enables one to escape from the rigid brain-pattern established throughout one’s life. The result of lateral thinking is an extensive set of alternative solutions, each of which can then be subjected to further vertical thinking.

Of the four thinking tool clusters defined by Edward de Bono (i.e. idea-generating tools, focus tools, harvest tools, and treatment tools), only the first one, idea-generating tool cluster, intended to break routine thinking patterns, is discussed in the book. Readers interested in the remaining thinking tools are encouraged to turn to the references at the end of the part of the book. We will use a brain-teasing example of how to cross a rickety bridge to examine the six-strategies methodology that follows.

2. Example: Four People on a Rickety Bridge

For ease of understanding, an example with a conventional solution is provided ahead of the lateral thinking method description.

Four people plan to cross a rickety bridge at night. They have one flashlight, and the bridge is too dangerous to cross without one. The bridge can
support up to two persons at a time (Figure 1). The individuals crossing the bridge exhibit the following characteristics: person A requires 1 minute, person B requires 2 minutes, person C requires 7 minutes, and person D requires 10 minutes. What is the shortest time needed for all four persons to cross the bridge?

Figure 1 - A rickety bridge example

The “correct,” or conventional, solution is shown here. However, after reviewing the LT strategies, the reader may come up with some creative ideas, either individually or in a group.

Step 1. A and B cross the bridge  ➔  Total time so far: 2 minutes
Step 2. B returns with the flashlight  ➔  Total time so far: 4 minutes
Step 3. C and D cross the bridge  ➔  Total time so far: 14 minutes
Step 4. A returns with the flashlight  ➔  Total time so far: 15 minutes
Step 5. A and B cross the bridge  ➔  Total time so far: 17 minutes
3. Lateral Thinking Method

Lateral thinking may be implemented in various ways. The book suggests carrying out the following six LT strategies:

<table>
<thead>
<tr>
<th>Strategy 1: Master the problem at hand.</th>
<th>Strategy 4: Break the rules.</th>
</tr>
</thead>
</table>

**STRATEGY 1 Mastering the Problem at Hand**

The relevant maxim for this lateral thinking strategy is: “Look before you leap, think before you act.” It advises that one should look deeply upon an issue at hand and think how it is to be done before beginning to act. It also hints of potential vulnerabilities for the ones who do not heed this advice. Mastering the issue entails (1) clarifying the problem and then (2) analyzing it. Clarifying the problem calls for identifying the known information about the problem. This should include but not be limited to the relevant facts, inferences, speculations, and opinions as well as identifying what is not known about the problem. Typically, analyzing the problem calls for answering relevant questions, such as:

- What is the problem?
- Why does the problem exist?
- Who is causing the problem?
- When did the problem first occur?
• What is the impact of the problem?

During this analysis, one can be fully aware of the problem and so avoid all kinds of pitfalls on the way to resolving it.

For the bridge example, one could analyze the problem and ask elemental questions like: “Why do these people need to cross the bridge? Could they take another route?”

---

**STRATEGY 2  Asking Fundamental Questions**

The next LT strategy is to ask a lot of nonthreatening, seemingly obvious questions that might bring about new insights into the true nature of the problem. The goal is to be able to challenge the basic perception of the problem and the issues around it. Hopefully, this will lead to fresh new ideas. Typical fundamental lateral thinking questions might be:

- Is it possible to restate the problem?
- Are we dealing with the real important problem?
- Is the problem caused by our operational procedures/strategy?

Asking fundamental questions provides mechanism to investigate the necessity, validity, and uniqueness of the problem at hand. As such, it can illuminate certain aspects of the problem that would otherwise be overlooked.

Regarding the rickety bridge example, one could ask seemingly unrelated questions like: Is the objective (i.e. minimize the overall bridge crossing time) truly important? Are there other, more important issues, to deal with (e.g. safety of the people involved or future travelers)?

---

**STRATEGY 3  Recognizing and Controlling Assumptions**

The origin of the word assumption is the Latin word assumptionem, meaning “a taking or receiving.” An assumption may be defined as “a hypothesis that
is taken for granted,” Assumptions play an important role in our lives and are created as a result of past experiences, societal culture and people and events that shape our lives. Furthermore, people are seldom aware of the effect of their inner assumptions on their actions. Assumptions often distort our reality by causing our brain to ignore information that does not correspond with what we have chosen to believe. Analyzing our assumptions identifies why we think and act in a certain way, while controlling assumptions enable us to set aside previous convictions and view problems in a more objective way.

Individuals and members of a group need to be cognizant of their own assumptions in order to be able to assess all the relevant issues pertinent to a given problem. In particular, psychologists observe a common phenomenon called the assumption of competence. In essence, most people perceive themselves as more competent and infallible than they really are. In other word, our brain tends to assume we know more than we actually do. So what can be done about this?

• Adopt some measure of doubt. First, one must recognize that each and every person has ingrained assumptions about every situation. Healthy doubt allows one to consider he may be wrong and still feel comfortable with it. Doubt must be applied in a balanced way. Too much doubt leads us to inaction, while overconfidence can easily lead us astray.

• Ask questions and listen. Asking plenty of basic questions is probably the best mechanism to fill in the blanks and weed out fact from fiction. Listening to a person talking, being truly present, is a necessary ingredient in the process of controlling one’s assumptions. Also, true listening consists of accepting the integrity of the other person and waiting patiently, without interruption, for all of the evidence to be presented.

• List the assumptions. When confronted with a problem, it is often advisable to write out the assumptions inherent to the problem at hand, as well as the proposed solution. This process allows us to capture our understanding in a definitive way, thus gaining overt insight into the situation.

Regarding the rickety bridge example, one could ask a question about our assumptions: The stated problem was: "What is the shortest time needed
for all four persons to cross the bridge?”. We may assume the bridge must be crossed immediately but this is not necessarily the case. Can the bridge crossing operation be postponed to the daytime? In this case, two persons (A and B) will cross the bridge and then the other two persons (C and D) will cross the bridge, requiring a total time of only 12 minutes.

---

**STRATEGY 4 Breaking the Rules**

Some teachers love students who unquestioningly follow the rules and do not disturb the class. Many managers value employees who eagerly accept the dogma of the company, do not challenge authority or their assigned tasks and ask no questions. It is a human trait to resist change. We are often most comfortable in a steady and predictable environment. However, creativity and innovation are invariably related to changes in the status quo. If people never question the norm or challenge conventions, there would seldom be any advancement in this world.

According to Sternberg (1999), there are three major categories of creative contributors: ones that accept current paradigms, ones that reject current paradigms, and ones that attempt to integrate multiple current paradigms into a new one. Creative contributions that reject current paradigms are often based on breaking the rules and are the least likely to be accepted. Thus, these individuals may repeatedly be considered troublemakers. However, ideas created by them are often the most creative and innovative ones.

Within engineering, rules are put in place to establish standards, mostly aimed at ensuring the achievement of stated goals. So, based on our training, experience, and natural psychological tendencies, following rules is something most of us engineers are quite comfortable with. In summary, breaking rules can be tricky due to the risk involved, but the flip side of this outlook is that often people break “rules” that, in fact, are simply a set of conventions. Here are some suggestions for an enlightened rule-breaking process:

- Rules identification. A good starting point in problem solving is to list as
many relevant rules as possible. The list should distinguish among laws of the land, explicit rules and standards, unwritten rules, implicit rules as well as day-to-day common practices.

• Concept viability evaluation. Deep evaluation must be undertaken regarding the rules to be broken and the proposed creative ideas. One should study carefully the nature of the said rules, their origin, raison d’etre and the level of resistance this approach may induce. Similarly, the pros and cons regarding the proposed creative ideas should be carefully analyzed vis-à-vis all stakeholders (colleagues, managers, customers, society at large, etc.).

• Risks and consequences evaluation. An engineer must be thoroughly familiar with the industry’s laws and regulations before he proceeds. Being careless when it comes to industry’s requirements can bear a risky outcome, sometimes with devastating effects on the public and the company as well as on the person’s career. Therefore, one should carefully weigh the potential downside of breaking established rules and consider whether one’s is willing to take responsibility for whatever may happen.

• Ethics and value evaluation. An engineer contemplating breaking established rules is advised to proceed only if he feels, true to himself, no ethical or moral objection. In other words, he should put firstly his obligations to colleges, managers, clients, his company as well as his profession and others.

Regarding the rickety bridge example, one could break a rule and ask: “Is the stated bridge load limit (i.e. two persons at a time) well-founded? Maybe three persons could cross the bridge safely at a time?”

STRATEGY 5 Broadening Perspectives

Perspective, in the context of the book is “the faculty of seeing all the relevant data in a meaningful relationship.” In other words, perspective is the ability
to step back from an existing position and gain a broader understanding of the situation. This process of “reframing” the situation expands one’s ability to generate new creative ideas. What can be done about this?

• The obvious. Broadening one’s perspectives starts with the obvious: Read extensively, ask questions and listen, spend time with new people, and generally, open up to the world (e.g. surfing the internet, watching TV, visiting museums, theaters and concert halls, taking trips etc.). Finally, stay humble so all this deluge can actually penetrate one’s mind and heart.

• Practicing different perspectives. Practice how other stakeholders regard a given situation (e.g. individual colleagues and managers, customers, suppliers, etc.).

Regarding the rickety bridge example, one could consider the perspective of other travelers trying to cross the bridge. They may come at night, not knowing the poor state of the bridge; they may not know the maximum allowable bridge load, etc. It may be more important to fix the bridge or install a clear warning sign rather than the mission presented in the original problem.

---

**STRATEGY 6 Utilizing Provocative Ideas**

In everyday language, provocation is an action or statement that is intended to make someone angry or distraught. However, in the context of the book, provocative ideas, radical and unrealistic as they may be, intend to forcibly cause one’s mind to move out of its comfort zone, allowing one to consider potentially surprising solutions to a problem at hand.

Provocative ideas are intentionally quite absurd and often seem nonsensical. Such ideas are often different from ideas raised during brainstorming, where participants offer solutions without active judgment but those that are feasible within the problem domain. There are several provocation creative techniques like wishful thinking, exaggeration, reality distortion, and the like.
Provocative ideas could, in principle, solve the relevant problem but, by and large, are not realistic. Therefore, the reciprocal step needed is to undertake a mental transformation to convert the provocative ideas into sensible solutions. There are several ways to do this – for instance, performing transformation via principle. Here, one extracts key principles from the provocative ideas that are critical in solving the problem. Then one may devise a realistic solution based on those principles.

In summary, implementing creative solution based on provocative ideas entails the following:

- Create one or more provocative ideas that principally solve the problem at hand.
- Identify the principles that underlie the provocation.
- Move to a realistic solution based on the identified principles.

Another essential component of this technique is to seek more than a single provocative idea and underlying principle. The best strategy is to keep looking for reasonable number of sets containing provocative ideas as well as their reciprocal principles. It is reasonable to expect that several such sets will enable one to propose a better solution.

Regarding the rickety bridge example, one could propose a provocative idea involving building a large catapult that will hurl all four persons onto the other side of the bridge in one fell swoop. The principle that underlies this idea is the possibility of all four persons crossing the bridge simultaneously at great speed. One solution utilizing this principle is to fix or enlarge the bridge so it will be safe for four persons to cross it together at full running speed or, even, using a fast vehicle.
4. Conclusions and remarks

The paper reflects the potential benefits available to systems engineers and engineers at large utilizing an established creative method like lateral thinking in their daily work. Readers interested in expanding their creative skills and innovative competency are advised to study the authors' 2018 book: Practical Creativity and Innovation in Systems Engineering, published by Wiley as part of Wiley’s Systems Engineering and Management Series (Figure 2).

The basic premise of the book is that creative abilities of human beings are not fixed, inborn traits but, rather, are changing over their lifetime. For example, researchers show that children exhibit remarkable abilities to look at problems and come up with new, different, and creative solutions. However, as they grow to adulthood, these abilities diminish substantially. Fortunately, creative skills can also be learned. Many studies show that well-designed training programs enhance creativity across different domains and criteria. Hopefully, engineers adopting some of the creative methods discussed in the book will achieve improved creative skills as well.

Another premise of the book is that many creative engineers are stalled in their innovative efforts by organizations that claim to promote innovation but that, in fact, crush such efforts. Indeed, it is the author's impression (as well as other researchers) that, beyond boasting, the vast majority of companies and other organizations are creativity-averse. Naturally, creative engineers working for such organizations are frustrated and discouraged. Not less important are the accumulated losses for the organizations themselves as well as to society at large from neglecting many creative ideas without due consideration. The book attempts to explore this phenomenon and offer

---

2 Book’s front cover: Image is part of a dry rock garden consisting of gravel and massive boulders placed by Hoichi Kurisu. Photo courtesy of Frederik Meijer Gardens & Sculpture Park, Grand Rapids, Michigan, USA.
practical advice to organizations as well as to the multitudes of demoralized engineers. In particular, engineers are advised to expand their professional and intellectual horizons, seek to reduce risks inherent in their new ideas, and learn to obtain colleagues’ support as well as deal with reactionary management. In short, adopt a more entrepreneurial attitude.

The book is organized in five parts: Part I: Introduction, contains material about the principles of the book and its content. Part II: Systems Engineering, describes basic systems engineering concepts and a partial and abbreviated summary of Standard 15288 systems’ life cycle processes. In addition, this part includes a recommended set of creative methods for each life cycle process. Finally, this part provides some philosophical thoughts about engineering. Part III: Creative Methods, the heart of the book, provides an extensive repertoire of practical creative methods engineers may use. Part IV: Promoting Innovative Culture, deals with ways and means to enhance innovative culture within organizations. Finally, Part V: Creative and Innovative Case Study, presents an exemplary creative and innovative research and implementation undertaking.

Fundamentally, the book is written with two categories of audience in mind. The first category is composed of practicing engineers in general and system engineers in particular as well as first- and second-line technical managers. These people may be employed by various development and manufacturing industries (e.g. aerospace, automobile, communication, healthcare equipment, etc.), by various civilian agencies (e.g. NASA, ESA, etc.) or with the military (e.g. Air Force, Navy, Army, etc.). The second category is composed of faculties and students within universities and colleges who are involved in Systems, Electrical, Aerospace, Mechanical, and Industrial Engineering. The book may be used as a supplemental graduate level textbook in creativity and innovation courses related to systems engineering. Selected portions of the book may be covered in one or two semesters.

Finally, readers should note that the book does not pursue new theories or theses with regards to creativity and innovation. To the contrary, the author
seeks to acquaint systems engineers with well-established facets of creativity and innovation. In order to achieve this objective, the author drew upon his engineering experience, communicated with many people, and collected information from many sources, books, articles, internet blogs, and the like (giving credit where credit’s due). Bibliographies at the end of each part of the book identify invaluable sources for deeper understanding of the various subject matters discussed in the book. The author gained much knowledge from these resources and is indebted to the individuals, researchers, and experts who created them.
Bibliography


About the author

Dr. Avner Engel, holds a B.Sc. in Electrical Engineering from the University of Maryland, an M.Sc. in Computer Systems from the University of New York and a Ph.D. from Tel-Aviv University. He has 45 years’ experience in areas of software programming, systems and software engineering, technical management within large firms in the United States and Israel, academic teaching, and scientific research. He is the author of Verification, Validation and Testing of Engineered Systems, published by Wiley as part of Wiley’s Systems Engineering and Management Series.
FOUR LAYERS APPROACH FOR DEVELOPING A TOOL FOR ASSESSING SYSTEMS THINKING

Moti Frank  
Faculty of Management of Technology, HIT-Holon Institute of Technology  
✉️ moti.mf@gmail.com

Sigal Kordova  
Faculty of Management of Technology, HIT-Holon Institute of Technology  
✉️ sigalkord@gmail.com

This article was also published in Industrial Engineering & Management, 4: 178.  
doi:10.4172/2169-0316.1000178

Abstract

To perform successfully systems engineering tasks, systems engineers need a systems view, in other words, a high capacity for engineering systems thinking (CEST). A tool for assessing systems thinking of engineers, once validated, may be used for systems engineering workforce selection and development, developing systems engineering curriculum, education, and training programs, as well as a standard tool for assessing systems engineers’ competencies. Since there is no known way of directly ‘measuring’ systems thinking in general and CEST in particular, an indirect method is needed. This paper proposes an idea for developing an indirect means, i.e. a questionnaire for assessing the CEST of systems engineers. The idea is composed of four logic layers.

Keywords: Systems Engineering, Systems Thinking, Engineering Systems Thinking, Capacity for Engineering Systems Thinking (CEST), Interest Inventory.

INTRODUCTION

Systems engineering is an interdisciplinary field of engineering that focuses on how to design and manage complex engineering projects over their life cycles. Systems thinking is what makes systems engineering different from
other kinds of engineering and is the underpinning skill required to do systems engineering (Beasley & Partridge, 2011). Engineering Systems Thinking is a major high-order thinking skill that enables individuals to perform systems engineering tasks (Frank, 2000; 2002).

To perform successfully systems engineering tasks, systems engineers need a systems view, in other words, a high capacity for engineering systems thinking (CEST). The main characteristic of systems engineers possessing high CEST is the ability to see the whole picture and identify the system emergent properties, capabilities, behaviors, and functions without looking at the detail. It is assumed that CEST is a measurable and consistent quality of personality, and it can be used to distinguish among individual engineers.

Systems thinking seeks to address and solve complex problems by understanding the system parts and their interactions within the context of the whole system, rather than in isolation (Hitchins, 2003). Systems thinking is a discipline for seeing the whole. It is a framework for seeing interrelationships and repeated events rather than things, patterns of change rather than static ‘snapshots’. Systems thinking is the perception of the ‘constructs’ underlying complex problems (Senge, 1994).

Systems thinking is considered a high order thinking skill. The simplest thinking skills are learning facts and recall. Higher order skills include critical thinking, creative thinking, analysis, problem solving, and systems thinking.

The battery for assessing CEST of individuals comprises a set of paper-and-pencil, field, and lab tests. This paper discusses one of the paper-and-pencil tests – an interest inventory – considering rationale, principles, uses, and stages of development, and presents some examples and results of studies aimed at checking its reliability and validity.
How to Assess Capacity for Engineering Systems Thinking (CEST) by a paper-and-pencil Questionnaire: the idea

The Assessing Need

A tool for assessing systems thinking of engineers, once validated, may be used for systems engineering workforce selection and development, developing systems engineering curriculum, education, and training programs, as well as a standard tool for assessing systems engineers’ competencies. Such a tool may be used for selection, filtering, and screening of candidates for systems engineering job positions, and for placing the ‘right person in the right job’. The tool may be used also as an instrument for evaluating the effectiveness of systems engineering curriculum and training programs.

Since there is no known way of directly ‘measuring’ systems thinking in general and CEST in particular, an indirect method is needed. For example, IQ tests are paper-and-pencil indirect tests for ‘measuring’ the intelligence of individuals. This paper proposes an idea for developing an indirect means, i.e. a questionnaire for assessing the CEST of systems engineers. The idea is composed of four logic layers.

The First Layer

As stated above, engineering systems thinking is a major high-order thinking skill that enables individuals to perform systems engineering tasks (Frank, 2000; 2002). We are dealing here with a thinking skill, so it is relevant to discuss the common methods of ‘measuring’ other thinking skills such as intelligence and creativity.

An intelligence quotient, or IQ, is a score derived from one of several standardized tests designed to assess intelligence. One of the common ways for assessing intelligence is by separately assessing abilities such as arithmetic, spatial imagery, reading, vocabulary, memory, and general knowledge and then weighting the individual scores to one general index (the IQ). For example, The WAIS-III version of the Wechsler Adult Intelligence Scale was released in 1997. It provided scores for Verbal IQ, Performance IQ, and Full Scale IQ, along with
The current version of the test, the WAIS-IV, which was released in 2008, is composed of 10 core subtests and five supplemental subtests, with the 10 core subtests comprising the Full Scale IQ. With the new WAIS-IV, the verbal/performance subscales from previous versions were removed and replaced by the index scores. The General Ability Index (GAI) was included, which consists of the Similarities, Vocabulary, and Information subtests from the Verbal Comprehension Index and the Block Design, Matrix Reasoning, and Visual Puzzles subtests from the Perceptual Reasoning Index. The GAI is clinically useful because it can be used as a measure of cognitive abilities that are less vulnerable to impairments of processing and working memory (Pearson, 2008).

There are four index scores representing major components of intelligence: Verbal Comprehension Index (VCI), Perceptual Reasoning Index (PRI), Working Memory Index (WMI), and Processing Speed Index (PSI). Two broad scores are also generated, which can be used to summarize general intellectual abilities: Full Scale IQ (FSIQ), based on the total combined performance of the VCI, PRI, and WMI, and PSI General Ability Index (GAI), based only on the six subtests that the VCI and PRI comprise.

A similar approach can be found in creativity tests. For example, Torrance (1984), building on Guilford's work, created the Torrance Tests of Creative Thinking (TTCT) based on four scales: Fluency – the total number of interpretable, meaningful, and relevant ideas generated in response to the
stimulus; Flexibility – the number of different categories of relevant responses; Originality – the statistical rarity of the responses; and Elaboration – the amount of detail in the responses. The third edition of the TTCT in 1984 eliminated the Flexibility scale from the figural test, but added Resistance to Premature Closure and Abstractness of Titles as two new criterion-referenced scores on the figural. To the five norm-referenced measures that he now had (fluency, originality, abstractness of titles, elaboration and resistance to premature closure), Torrance added 13 criterion-referenced measures: emotional expressiveness, story-telling articulateness, movement or actions, expressiveness of titles, syntheses of incomplete figures, synthesis of lines and circles, unusual visualization, extending or breaking boundaries, humor, richness of imagery, colorfulness of imagery, and fantasy (Cramond et al., 2005).

The Second Layer

From IQ and creativity assessing approaches, we have learned that the first stage of developing a tool for assessing systems thinking in engineers should be decomposing the ‘engineering systems thinking’ to factors that can be separately assessed. But how is this done? In order to answer such a question, let us consider one of the approaches taken in Leadership research.

There are a vast number of books, articles, and studies available which attempt to identify the competencies and qualities of effective leaders. For example, Morrison (2000) presents a global leadership model, while Jokinen (2005) reviews and discusses the main findings of previous leadership studies aimed at identifying characteristics and competencies of effective and successful leaders.

Thus, we can see that a basic approach for understanding the ‘leadership phenomenon’ is to identify the competencies of successful leaders. It is suggested here that, in the same manner, in order to understand the ‘engineering systems thinking phenomenon’, we should try to identify the competencies and characteristics of engineers who are systems thinkers.
But how can we know which engineers can be considered as systems thinkers as the tool for assessing CEST has not been developed yet?

The Third Layer

As mentioned above, to perform successfully systems engineering tasks, systems engineers need a systems view, in other words, a high capacity for engineering systems thinking (CEST). Thus, for the sake of this study, we may hypothesize that successful systems engineers are characterized by high Capacity for Engineering Systems Thinking (CEST). That is to say, successful systems engineers are systems thinkers. This hypothesis should be reconsidered when the development of the systems thinking assessment tool has been completed. In order to understand the ‘engineering systems thinking phenomenon’, let us try to identify therefore the competencies and characteristics of successful systems engineers.

In actual fact, this has already been done. Frank (2006) aimed at identifying the characteristics of successful systems engineers. The study included observations of systems engineers who were evaluated – by at least three peers and two supervisors – as being ‘successful’. Eighty-three, which later aggregated to 34 competencies, of successful systems engineers were identified in this study. These 34 competencies were classified into:

Ten cognitive characteristics – understanding the big picture, interconnections, systems synergy, multiple perspectives, systems without getting stuck on details, implications of proposed change, a new system or concept immediately upon presentation, analogies and parallelism between systems, limits to growth and thinking creatively. All these 10 characteristics are related to systems thinking.

Eleven abilities – analyzing the need, analyzing/developing the concept of operations, requirements analysis, conceptualizing the solution, generating the logical solution (functional analysis), generating the physical solution (architecture synthesis), ‘seeing’ the future, using simulations and systems engineering tools, optimizing, using systems design considerations, leading
trade studies.

Ten individual traits – management skills, team leader, building and controlling the work plan, defining boundaries, taking into consideration non-engineering factors, good human relations, team player, good communication skills, good interpersonal skills, autonomous and independent learner, strong learning skills, willing to deal with systems, curious, innovator, initiator, promoter, originator, asks good questions.

Three dealing with multidisciplinary knowledge and experience.

Thus, the content validity of the proposed tool can be achieved by basing its items on the finding in the latter study (Frank, 2006). Each competency of successful systems engineers, found in that study, may be separately assessed by a single item or several items of the proposed tool and then the individual scores weighted to one general index.

**The Fourth Layer**

One of the findings by Frank (2006) is that in order to be a successful systems engineer, one must have both a will and an interest in being a systems engineer. In addition, as mentioned, successful systems engineers possess a high capacity for engineering systems thinking (CEST).

Thus, the three components discussed here – success in a systems engineering position, an interest in systems engineering positions and CEST – are all interrelated. The will and interest to be a systems engineer basically means the desire and interest to be involved in job positions that require CEST. In other words, we may hypothesize that there is a high positive correlation between the engineering systems thinking extent (CEST) of an individual and his/her interest in what is required from successful systems engineers. Figure 1 is a simple concept map that depicts the relationships between these three components:
Figure 1. The relationships between the desire, successful SE and CEST

If this hypothesis is supported, then it enables developing a method for assessing the extent of CEST of individuals. This is because interests may be assessed by an interest inventory which is very common and frequently used to help people choose a profession, and as a selection tool (to determine whether a certain individual is suitable for a certain role) in the recruitment process (Anastazi, 1988).

assessing the interest for engineering positions’ required capacity for engineering systems thinking (CEST)

Frank (2010) introduces a tool for assessing interest for systems engineering positions and other engineering positions’ required capacity for engineering systems thinking (CEST). Usually, the items in interest inventories deal with preferences, specifically likes and dislikes regarding a diverse group of activities, jobs, professions or personality types. Likewise, the items included in the tool discussed in this chapter refer to ranges of likes and dislikes regarding systems engineering activities, various disciplines and knowledge required from systems engineers, systems engineering activities and types of people involved in projects.

In its present version, the tool consists of 40 pairs of statements. For each pair, the examinee has to choose between the two statements according to his/her preference. The examinee checks answer “A” if he/she prefers the first statement or answer “B” if he/she prefers the second statement. In order to improve the questionnaire’s reliability, questionnaire items were reorganized, so in some cases “A” represented the systems thinking answer and in other cases “B” represented the systems thinking answer. Each “A” answer receives 2.5 points, while each “B” answer receives no point. Thus, the range of the scores is 0-100.

VALIDATING THE TOOL – RESULTS OF RECENT STUDIES

Four types of validity have already been checked in a series of studies of pilot studies – content validity, contrasted groups validity, concurrent validity and
construct validity (Frank, 2010). Here are results of additional study. Koral Kordova, Ribnikov & Frank (2015) identified the factors that influence the development of systems thinking among systems engineers. In this study, contrasted group validity was determined by comparing the grades of two contrasted groups - systems engineers and domain engineers such as software and hardware engineers. It was found that there is a significant difference between the average scores of CEST among systems engineers and domain engineers (Sig=0.000).

Moreover, a significant correlation was found between supervisors’ ranking in relation to these engineers’ systems thinking capabilities and the average score they achieved on Frank’s questionnaire (Sig=0.000, r =0.855). This finding is determined the concurrent validity. Construct validity was checked by factor analyses. In addition, while they were filling out the questionnaire, the engineers themselves were asked to evaluate their desire to engage in systems-related projects; a significant correlation was found between this evaluation and the results of Frank’ questionnaire (Sig=0.000, r=0.763).

CONCLUSIONS

Every enterprise strives to fill positions in the organization with employees who have the best chance to succeed. Employees are also interested in entering positions that fulfill their aspirations. Selection and screening processes can help match the interests of both parties, thus contributing both to the organization and the individual. The selection process for systems engineering positions should reliably predict those employees who can succeed and reject those who are likely to fail. Out of the employees who can succeed as systems engineers, it is necessary to choose those who have the highest chance of succeeding.

From the organization’s point of view, rejection of candidates who might have succeeded in systems engineering positions can be critical, especially under conditions of an ever-increasing shortage of systems engineers. Likewise, placing engineers who later fail in systems engineering positions is also an expensive error, taking into consideration the necessary training which will
be invested and the subsequent damage which might be caused to the projects in which they are involved. The tool presented in this paper may be used for selection, filtering, screening of candidates for systems engineering job positions, and for placing the ‘right person in the right job’.

REFERENCES


About the authors

Prof. Moti Frank obtained his three degrees from the Technion, Israel. After accomplishing B.Sc. in electrical engineering he served around 20 years in the IAF as a communication systems engineer. His last position was the commander of the aerial communication and navigation division. After his military service he returned to the academy and accomplished his M.Sc. in science education and Ph.D. in science education combined with industrial engineering and management. Concurrently he worked for several years at the Technion and later joined the Holon Institute of Technology (HIT), where he held various positions, the latter of which was VP for Academic. Moti is currently the VP of Ramat Gan Academic College.

Dr. Sigal Kordova is an industrial and management engineer, with B.Sc., M.Sc. and Ph.D. from the Technion. She was the head of the industrial engineering and management track in the school of practical engineering and was an active partner in the development and design of the industrial engineering and management programs in the education system. Currently Sigal is heading the elective section of Project Management in the faculty of technological management at the Holon Institute of Technology (HIT) and is active in academic research in the areas of systems engineering and project management.
The Digital Transformation and its Impact on the Working Models of the Systems Engineers and Analysts Presentation [Abstract]

Raz Heiferman, Co-CEO, Be Digital

This is the abstract of the author's presentation at the 19/6/2018 Systems Engineering seminar at Gordon Center – Technion, Israel. An article by the author, on the digital maturity of the organization is in the Hebrew side of the journal.

We are living and working in a new era – the Digital Era. This era brings profound changes in the working models of the Systems Engineer and Analysts. This presentation provides a short introduction to this new era and the Digital Transformation of organizations and highlights the major changes in the working models of the System Analysts.

The Digital Transformation is defined as the process of rethinking the business models and strategies through the adoption of the new and incredible potential of the digital technologies in order to compete and improve the organizational performance. Strategy has changed from the view of long term and sustainable competitive advantage to the concept of transient competitive advantage and the need for agile organizations – quick go-to-market, organizational structures, innovative business models data driven culture.

All those changes have a great impact on the working models of the new Systems Engineers and Analysts. They need to adopt methodologies that embed agility as a major theme. Although the new systems operate in a very complex environment that is very different from the monolithic architectures we knew in the past, the constant pressure of the organizations for quick development cycles and go-to-market strategies with intelligent products that will provide a superior customer experience. Those pressures pushed the development
of methodologies like Design Thinking, Dev/Ops, Agile Development, Open Source, Continuous Deployment, Fail Fast, MVP (Minimal Viable Products), Cloud Computing, Cyber by Design and more.

We will browse shortly through those concepts that the new System Engineers and Analysts must learn and adopt.
The 10th Israeli International Conference on Systems Engineering

26-27 March 2019, Daniel Hotel, Herzliya

Call for Papers, abstracts and presentations
www.iltam.org/incose_il2019

The international conference of INCOSE_IL, which is conducted biennially, is the main event of the Israeli Systems Engineering community. The conference attracts hundreds of participants from Israel and abroad, under the greeting of INCOSE – the International Council for Systems Engineering.

As is expected from a 10th conference, an especially rich program will be established, including guest lectures, research reports, applied practices, tools and methods, and more, in a variety of parallel sessions. The second day will be dedicated to a full-day seminars by a world-known Systems Engineering expert.

Since you wouldn't want to miss such important event, mark it in your calendar right now and follow the details published on the Conference website. Shortly a call for papers and presentations submission will be published.

We look forward to meeting you at the conference.
Prof. Amir Tomer, INCOSE_IL 2019 chair

For Submission and Registration

TEL. 03-6889214  FAX. 03-6889216  EMAIL. incose_il@iltam.org
The Voice of the Systems
The Journal Of The Israeli Systems Engineers

ISSUE NO.22 | June 2018

INCOSE_IL - THE ISRAELI SOCIETY FOR SYSTEMS ENGINEERING
ILTAM - Users' Association Of Advanced Development Of Complex Systems And SoS

Address: 29 Hamered st. Tel-Aviv, Israel 6812511
Tel: +972-3-6889220 • Fax: +972-3-6889216
Email: iltam@iltam.org • Site: www.iltam.org
FOLLOW US on Facebook: facebook.com/iltamorg

GORDON CENTER FOR SYSTEMS ENGINEERING
TECHNION - ISRAEL INSTITUTE OF TECHNOLOGY
www.gordon-se.technion.ac.il

Graphic design: Ezra Siton