PART III

THE INTERVIEWS

3.1

DEVELOPMENTS IN A COMPLEX, TECHNOLOGICAL WORLD – THE AVIATION AND SPACE INDUSTRIES

3.1.1 "STRUCTURED, MULTIDISCIPLINARY METHODS OF RESOLVING LATERAL PROBLEMS"

An Interview with Norman Augustine

The government of the United States engages in diverse and widespread activities at the South Pole. According to the American perception, the southern continent of Antarctica should not be under the jurisdiction of any particular country; it belongs to the world. True to this position, the United States established a research station at the South Pole, making no territorial claims and has been cooperating with various countries in conducting research studies of the continent. These studies range from investigations into the origin of the universe to the impact of global warming on sea-level rise.

In addition to the station's research objectives, its location also bears geo-political significance: the government of the United States bases its viewpoint on the assumption that if they were to leave such a strategic location, other countries might seek control over the area – possibly even leading to military conflict. A number of nations have in fact made claims of ownership of (sometimes overlapping) parts of Antarctica, nearly all of which include the South Pole area. Therefore, in order to advance science and prevent such a situation with its sensitive implications from occurring,

Managing and Engineering Complex Technological Systems, First Edition.
Avigdor Zonnenshain and Shuki Stauber.

the government of the United States has been operating and financing the research station and its support stations for decades. These stations located in Antarctica are provided logistics from locations beyond the boundaries of the continent.

The US activity was assessed during the second half of the 1990s, particularly due to the fact that operational costs were extremely high – a natural result of long supply lines and severe weather conditions. The examination was conducted by a committee of renowned experts who represented a variety of relevant disciplines. Mr. Norman Augustine, an expert in Systems Engineering and Aeronautical Engineering, headed the committee, and at the same time, served as president and CEO of the large aerospace manufacturer, Lockheed Martin. The committee published its conclusions in 1997, in principle: to continue the research operations and activities and to find ways to improve efficiency and reduce costs. This resulted in the construction of a new station at the South Pole to replace the then-existing station that was deemed no longer safe for habitation.

More than a decade passed, and in 2011, another committee was appointed to examine ways for reducing operational costs. Norman Augustine headed this committee as well, even though he had already retired and had been devoting his professional skills to diverse public service activities.

The conversation with Augustine focused on two primary subjects: the working and behavioral patterns implemented by the committee while attempting to make the United States government's operations in Antarctica more efficient – working patterns drawn from the content world of systems engineering; and his perception regarding the essence of systems engineering.

Examination of Operations in Antarctica

Augustine believes that he was not chosen to head the examination process because of his experience as head of the committee in 1997, which had conducted a broader examination and had reviewed implications and policies. In contrast, the current committee was being summoned to examine performance: the task was to examine ways for reducing logistical costs of the research studies being conducted in Antarctica.

Augustine: "This task is more related to my skills as a systems engineer – my experience as someone who has managed wide-ranging projects, which necessitate the formation of groups from different fields of activity and coordinating work efforts.

The following is an example of a dilemma that most people would not consider, whereas a systems engineer probably would. What would happen if the length of the shower that each member of the research station team at the South Pole took could be shortened by 30 seconds? Why would this be significant? Because there is plenty of ice there, but not liquid water. The ice, of course, must be heated and melted into water – a process that consumes large amounts of energy. And it is not only the cost of the fuel used for heating, but the distance it must travel in order to reach the station that is also of significance. Fuel is brought to the station by ship that requires the service of icebreakers and then by cargo aircraft. People at the station understandably might not give much thought to the 30 seconds. They simply want to take showers.

We wanted to evaluate the fully burdened cost of a shower per minute. This is an example of a systems engineering analysis. In addition to the technical, engineering side, there are also human aspects. How will people react to the idea of taking shorter showers? If they are told that the reasons are simply for saving on expenses, they might object. But, if they are told that the money saved will enable them to conduct additional research, they might be motivated, as scientists, to agree. Psychology also enters the picture, as does economics, and the wider perspective of systems engineering is required. But the problem of linking showers to icebreakers to aircraft is further complicated in that some energy must be consumed to generate electric power to provide light and heat for the station – and if that process produces excess ("waste") energy that can be used to help melt ice, then the equation changes altogether."

Forming the Staff. The committee operated within the framework of the "National Science Foundation and the White House office of Technology Policy" that also provided the members of the committee with a supporting, administrative team. The first step in selecting suitable committee members was the preparation of a list of some 25 subjects that, in Augustine's estimation and in accordance with the opinions of White House and Foundation personnel, were the central issues to be addressed by the committee. For example: how to employ icebreakers in order to reach the South Pole; how to provide for proper health-care services; how to prevent fire in an environment with lots of water – but all in the form of ice. (This past year a Brazilian station was destroyed by fire.)

The next stage involved the characterization of the skills required in order to cope with the issues being raised. For example, the ability to conduct a financial analysis, an understanding of ice breaker operations, and the input of someone who had endured weather conditions of 80° below 0 °F.

Augustine: "We reached the stage of identifying those with the necessary qualifications. We noted the names of people we were acquainted with, as well as the names of those most likely to know people endowed with the skills that we were looking for. We also made use of the Internet. We contacted 'The National Academy' and reviewed the names of scientists and engineers. We inquired into who excels in this field and who in that one, who was willing to devote the considerable effort needed, including travel to the South Pole. We received recommendation for the most qualified individuals and contacted them to see if they might be willing to take part in the mission." A matrix of project needs/individual capabilities was the result.

The search and detection team did not review the candidates merely by their expertise and professional reputation: "Personal qualities were also important and the qualities of the recommended candidates were examined. We did not want to appoint individuals who might be 'zealots,' with decisive and firm positions that were not subject to change even in the face of new information. From our point of view, this was crucial.

The second important quality that we looked for was the ability to work with other people. This kind of framework calls for numerous meetings and discussions and lots of hours together on ships, on airplanes and in conference rooms. When you have one individual who wants to do all the talking, the system breaks down. We were looking for people who would want to listen at least as much as they would want to talk. On the other hand, you don't want a room full of people who just sit there and don't say a word. You want those who know how to be contributors."

Members of the committee were expected to do their jobs voluntarily as a public service. (The committee was unpaid.) In Augustine's estimation, it would be a matter of 8–9 months of part-time work. The willingness among the members to participate in the committee's activities stemmed from a variety of reasons: the desire to take part in an undertaking of national importance (most committee members were involved in numerous such activities;) out of interest and the challenges involved; the desire to be part of a unique experience; and the desire to make a contribution.

Why wasn't an undertaking of this type imposed upon an international consulting firm like Mackenzie?

Augustine: "A consulting firm would not have been able to put together a group of professionals of the caliber of our group. For example, if Mackenzie had contacted me, I don't think I would have accepted. I am not looking for traditional, paid work at this stage in my life. But, when the folks from the White House asked – I willingly accepted,[1] considering doing so part of one's duty as a citizen."

Planning. After the committee members were chosen, a working framework was formulate. Augustine: "I made an outline of the final report and we began allocating responsibilities to each member of the committee. By and large, those were executive decisions, since we needed to balance the use of the particular talents of the 12 of us. As the work process progressed, we made a number of revisions when we found that some 'chapters' overlapped or that new issues arose. We could generally deal with this by combining task groups or adding new ones.

As we got further into our undertaking we realized that we had focused so much of our attention on activities in Antarctica itself and its first-tier supply points (New Zealand and Chile) that we were not adequately examining the "base" activities back in the United States. New task groups were created to specifically focus on the latter. These were composed of members designated by the chairman based on their individual skill sets."

Gathering Information. The next stage was the gathering of information in order to characterize and decompose the Antarctic support system and the ways in which it functioned. This included visits to the various research stations and their supporting

[1]The resulting committee of 12 included the recently retired head of the US Coast Guard; a recently retired "four-star" Air Force General; the recently retired chairman of the US House of Representatives Committee on Science and Energy; the recently retired head of logistics for one of the world's largest international corporations; a retired Navy Admiral; and a number of highly respected scientists and engineers.

bases as well as meetings with the people involved in the research activities at the South Pole and other sites – some housing over 1000 people in small villages and others consisting of a tent or two.

Since members of the committee reside at different locations throughout the United States, as well as a representative of the French Antarctic Program, we periodically got together for 2-day meetings (there have been six such meetings to date), not including our extended travel time together. We generally met in the evening of day one and continued discussions throughout the following 2 days. During these meetings, experts and professionals with relevant information appeared before the committee. For example: an expert who addressed us was able to shed light on the function of traverse vehicles under extremely cold climate conditions.

The committee considered both specialization matters as well as more complex and comprehensive systemic subjects. One of the more important issues was the matter of shipping, handling, and the transportation of goods.

Augustine: "Large amounts of supplies reach the South Pole by ski-equipped, four-engine turboprop aircraft. Special vehicles (that move on caterpillar tracks) can also be used to reach the South Pole, but it takes quite a long time, 15 days for a round trip from the coast. This is a complex choice with many different aspects. It is the analysis of a system, which is the focus of systems engineering. In order to decide which manner of transportation to choose, both the advantages and the disadvantages of each option need to be evaluated, as well as unintended consequences of making changes, thus involving everything from weather analysis to human safety and the impact of operations on the natural environment. Therefore, all of the committee members take part in these discussions. On the other hand, the matter of which type of aircraft is more appropriate is a specialization issue that does not require the input of the icebreaker expert. For the same reason, not everyone goes to visit all of the different sites. Sometimes, only a group of three to four committee members makes the trip to a particular location. The committee members were, as a whole, already quite familiar with the issues at hand having collectively made 82 trips to Antarctica, of which 16 were to the South Pole."

What Is the Connection Between the Project and Systems Engineering?

One could certainly ask the question: Why should a description of a project of this kind be included in a book that deals with systems engineering? It appears to be a classic example of organizational consultation, including the gathering and analysis of data, the pinpointing of problems and recommendations for improvement. It is not the design for a new airplane.

Augustine: "It is the characterization and analysis of a very large and complex operation comprised of numerous components that interact with one another and require a process of trades-offs, similar to the example mentioned earlier regarding methods of transportation. Another example: perhaps it would be wiser to integrate robotic sensors and a broadband communications system instead of flying some of the researchers to Antarctica at all. This way, some of the scientists could remain at home and fewer people would be needed for the on-site collection of data which is a major driver of cost. In fact, 90% of the person-days spent in Antarctica as part of the US effort there are involved in logistic support of the 10% who are actually performing the research."

In other words, the thought processes of a systems engineer were required in order to achieve appropriate results. According to Augustine, even though most of committee members do not consider themselves to be systems engineers, the vast majority of them are endowed with the ability to broaden their thought processes and think by the rules of systems engineering in dealing with problematic and diverse situations.

About Systems Engineering

What Is Systems Engineering? A system is anything evolved from elements that need to work together and that affect one another. Systems engineering is the art and science of assembling numerous components together (including people) in order to perform useful functions. For example: an air logistics system includes air traffic control radars, airplanes, pilots, passengers, runways, communications, meteorologists, and much more.

- It is not merely the assembly of components, but rather the assembly of components in a coordinated fashion. The concept is not to assemble components and then await the outcome; rather, it is to assemble components in a logical, efficient and economical way that accomplishes a desired function (without interfering with the performance of the functions of other systems).
- Systems engineering is a way of dealing with wide-ranging problems in an organized and disciplined manner.
- Engineers generally solve engineering problems, as opposed to systems engineers who deal with problems that involve many diverse factors ranging from people to economics and from politics to science.
- Systems engineers deal with problems by synthesis, analysis, and trade-offs.
- One of the most important areas of activity for a systems engineer concerns trade-offs: determining the appropriate "dose" of each component in the system.
- Systems engineering is a managerial tool. But, it is more than engineering management – it also deals with matters that are not "physical." In many aspects, it is not a technical skill. This can be frustrating for engineers who do not like dealing with human issues and their ambiguities.

Not Just Engineers

- Systems engineering thus differs from traditional engineering in that it also takes non-engineering components into consideration. For example: economics. Economics is a central component in systems engineering.

Systems engineering has grown from within engineering because both fields of study are based on similar principles, such as the assembly of components, their analysis, trade-off, and modeling. These tools are also useful for solving problems in areas beyond the fields of hardware and software, such as in industrial organization design, health-care management, and financial investing.

– The word engineering has become a generic term related to problem solving. We are familiar with terms like organizational engineering.
– Engineers generally know how to deal with the laws of nature: if you do things exactly the same way in the same environment, then whatever works today will usually work tomorrow. On the other hand, if people are included in the system, the result might be very different. The identification of systems engineering with the world of engineering does not necessarily stem from the fact that engineers are required to undertake much of the work, rather from the fact that the discipline has grown from within the field of engineering. For example: the input of economists and financial analysts has been most significant in many of the projects in which I have been involved in my career. Today, the combination of engineers and physicians is having an enormous impact on health care.
– While at the Pentagon in 1965, a Colonel with whom I worked conducted analyses of the likely contribution to national security of funds assigned to ballistic missiles as compared with submarines, civil defense, airplanes, or training. He was criticized for this "audacity," but, in essence, we make exactly these trade-offs every year when we prepare the defense budget. But we do it in our heads, often without disciplined methodology or analyses. Systems engineering can provide discipline in building a budget: how much should be allocated to health, to national security, and to roadways?
– Human Resources Managers often apply systems engineering tools, but for traditional reasons they are not defined as systems engineers; system engineering is usually classified among the physical fields. The thought processes of systems engineering are taught in depth in engineering courses, but not in most other fields of study.
– To some, systems engineering is not considered a specific discipline, rather simply as a means for dealing with complex problems.

Skills of a Systems Engineer

– In order to be a good systems engineer, one needs first to specialize in a single professional field and then broaden oneself to become a systems engineer. This is because systems engineering touches upon problems that are extraordinarily complex. Solving these problems requires a variety of skills – more than one person alone is capable of developing. But almost all good systems engineers previously had established themselves working in a single discipline.
– A good systems engineer needs to combine understanding of fundamental phenomena, experience with a wide range of fields, with a deep and comprehensive understanding of one professional field. Many different people have

worked for me in my career: lawyers, financial personnel, chemists, engineers, people in marketing and advertising, as well as those involved in human resources. Knowing how to put these pieces together is simply an example of systems engineering in a different context. For example, at Lockheed Martin, there were 180,000 of us working together to accomplish challenging goals.

Leadership and Systems Engineering. Many believe that a good systems engineer often needs to display leadership qualities, particularly when in charge of goal-oriented teams that are working with a limited budget and a tight schedule. Norman Augustine supports this belief. He writes and gives lectures about leadership quite frequently, while emphasizing the elements that touch upon the work of systems engineers:

– Leaders cause the right things to be done. They seek out the good, the just and the correct result. Therefore, Hitler and Lenin were not leaders, even though they motivated – or intimated – people to act. That is not sufficient. Some of the greatest criminals in the world had leadership qualities, but they lacked integrity – true leaders motivate people to achieve inspirational and constructive goals and do so because they want to do so.

– While conducting my research study on great leaders, including those that I have been fortunate enough to know, I discovered that in spite of the numerous differences in their individual personalities and styles, they share a number of similar qualities. These qualities were revealed particularly during times of crisis. As the saying goes, during times of calm, every ship has a good captain. I found 12 common traits among the finest leaders I knew:

  a. *Character* – if people do not trust you, they will not follow you. True leaders have high moral and ethical standards. They do the proper thing, no matter what the price.
  b. *Vision* – leaders know what they are aspiring to achieve … in the words of GE's Jack Welch, they can see around corners.
  c. *Competence* – a person cannot lead the way without being knowledgeable in his or her field of activity. Among the many important skills required for leadership is the ability to evaluate people and be able to decide who is more suitable in a certain situation. This is related to my philosophy of management: find good people, tell them what you want and step aside.
  d. *Energy*: leaders are people committed to their goals. They never run out of energy.
  e. *Courage* – a leader needs to instill confidence in others. He or she must have the ability and the daring to take calculated risks.
  f. *Perseverance* – a leader needs to display determination, resolve and persistence. Never give up … unless it becomes clear that the goal being sought is no longer appropriate.

g. *The ability to motivate others* – a leader motivates people to do things above and beyond what they thought they were capable of doing. In this world of ours, which is very achievement-oriented, the ability to make people invest that additional effort is what makes the difference. A leader motivates by setting an example. In addition, a leader has to show people that he or she believes in them, and enable them to express their creativity. This is an essential condition for building a team.

h. *Lacking egotism* – Great leaders are team players. They give credit to others. This is how they achieve great accomplishments: by being goal-oriented rather than being motivated by personal interests. Ironically, the best way to get ahead is not to try to get ahead.

i. *Decisiveness* – a leader needs to be capable of making the tough decisions … sometimes without all the facts they would like.

j. *Judgment* – decisiveness needs to be combined with good judgment. Leadership includes intuition and a kind of control mechanism that prevents one from making impulsive decisions. This quality is reinforced by the ability to take criticism and to listen to others and to learn. According to the dictionary, judgment is "a process of formulating an opinion or an assessment by means of diagnosis or comparison." Good judgment is the ability to recognize the best of possible approaches to a challenge … and the ability to let go of an outdated one.

k. *Mentoring* – great leaders create other great leaders. One of the more outstanding ways to characterize leaders is to examine the people that develop under their mentorship. Leaders give encouragement, inspiration, and advice.

l. *Listening* – Peter Drucker said that 60% of all managerial-administrative failures stem from a lack of communication. A leader must know how to listen. Warren Buffet told me that leaders need someone next them to remind them that "the Emperor has no clothes."

3.1.2 "PLANNING SYSTEMS THAT FIT THE NEEDS OF BOTH CLIENTS AND USERS"

An Interview with Yossi Ackerman

A question that still has not been answered clearly is whether systems engineering is, first and foremost, the concern of engineers, who see it as a tool for handling complex systems; or a systemic management field that affects management patterns and non-technological organizations as well.

This chapter explores the positions of the president of a major technological corporation on this issue. An engineer by training, Yossi Ackerman has never filled the position of a systems engineer, but nonetheless sees himself as one – an engineer who thinks systemic.

Background

When most of his friends would go to study at the local high school of his hometown in the Galilee, Yossi Ackerman would get up two hours earlier to get to Haifa, to Bosmat Electronics Tech School. His parents had decided that the boy had to learn a trade and become an electronics technician. Back then, in the 1960s, Bosmat was a reputable practical and theoretical school. The Technion, for instance, would receive its graduates with no entrance examinations.

Yossi Ackerman: “We studied advanced theoretical electronics and mathematics. Alongside that, we worked in workshops and learned mechanical and microchip processing. The school’s teaching philosophy was multidisciplinary. We learned to use a lathe to produce accurately shaped and sized pieces; we learned how to document our work, so that we could continue from the same point the next day; we learned about mechanical drawings, but we had bible studies too – because an electronics engineer has to know more than just electronics.”

Upon graduating, Yossi Ackerman was accepted into the IDF’s Academic Reserve program and studied Aeronautical Engineering at the Technion. He describes it as a systemic discipline, an aggregate of mechanical engineering, control, and other fields.

During his military service in a flight testing unit in the Air Force, Yossi Ackerman’s career in systems did not halt its progress: “The good thing about that unit was that we learned new things all the time. They would tell me ‘we want to receive a new electronic warfare system,’ and I had to learn all about it, approve it for use by the air force, train the team of users and integrate it into the air force. At that point, I knew nothing about the area and had to start studying it from scratch. Later, there would be a new bearing that they wanted to use on the plane. So I went from one field to the next, and discovered that everything I had learned in Bosmat and thought to be irrelevant was in fact very relevant, because they had, in effect, taught me not to fear anything new. So today, whenever I stand before a new subject, I know I can handle it, or, rather, that I can learn to handle it. And if, after that, there is still something I do not know, I bring in an expert who does.”

After being discharged from the military, he returned home, to manage the family farmstead. For 7 years, his main occupation was raising turkeys. Once again, he had to learn new things, and once again he employed the approach he had adapted for himself – what he did not know, he either learned or used the services of experts: “I did everything myself. I was a welder; I built turkey coops, installed air conditioning systems, engaged in biology, food engineering, economics, and the management of expenses and income.”

Yossi Ackerman had obviously left an impression on the friends he had made during his high school, university and military years, because in 1982, after 7 years, in which he had not practiced engineering, he was approached by Elbit Systems and recruited – not as a junior engineer but as the head of the Lavi Project (for a more detailed discussion on the Lavi Project, see interview with Ovadia Harari), in which Elbit was in charge of the plane’s avionics (e.g., the navigation and piloting systems, or the weapon arming systems). After a period of hesitation, he accepted and, at first, started working for the company in a half-time position.

The first month was, unsurprisingly, traumatic, and, towards the end of it, Ackerman asked to be released from the mission. He describes the conversation that followed with his superior rather graphically, seeing as it had been a constituting event in his professional life: "After three weeks, I came to my boss, Yossi Tidhar, and said: 'We both made fatal errors, you in recruiting me, and me in accepting. Let's erase everything; I'm going back to the farm. The people here are much smarter than me, they know everything, and I don't understand what they want from me.'

He said to me: 'Let's sit down. Tell me what your difficulties are and we will resolve them. Let's take care of each problem separately.'

I said: 'First of all, I don't understand all these letters: SOW, WBS, MFD.'

He replied: 'When you are sitting with other people and you hear something you do not understand, write it down on a piece of paper, then later come to me and I will explain what it is. There aren't thousands of these words, maybe thirty. What is the next difficulty?'

I said: 'In three days, I have negotiations in Tel Aviv with the CEO of a certain company Elbit has been in conflict with for a long time, and I do not even know what the argument is about, not to mention Elbit's position on the issue. How can I be sent there?'

And he answered: 'Let's break this one down, too. Who do you think can handle the problem?

I said: 'This person knows the economic issue.'

He said: 'Excellent, he will join you at the meeting. Who else knows anything?'

I said: 'That person knows the technical part, and the one who headed the project before me knows the history of the conflict.'

He said: 'Both of them will join you, then. Prepare for the meeting with the three of them and go together, If anything else that you are not familiar with comes up at the meeting, tell them you need time to think and will get back to them later with an answer.'

And that it exactly what happened. The negotiations went smoothly, and the end result was excellent. I have carried this lesson with me ever since. This is how I resolve crises. Whenever there is a difficulty, we break it down into components and address them one by one."

Yossi Ackerman had been the manager of the Lavi Project until it was shut down in 1985. That year, he replaced his superior as the manager in charge of the aviation projects in the company's air division (which included production systems and other elements as well). As time passed, he moved up through the ranks, until, in 1996, he was promoted to the position of Company President, which he filled until the year 2013.

He sees himself as a systems engineer, although that had never been his job title: "I think like a systems engineer. Even as a farmer, I was a systems engineer. Back then, I did not know I was practicing systems engineering, because the term did not exist. Anyone with a can-do attitude is a systems engineer. This is systems thinking, and if you add the word 'engineer,' you just narrow it down to the technological field. All managers' work is essentially systemic."

Systems Engineering at Elbit

As a large technological company, Elbit does everything that falls within its areas of activity. Therefore, Yossi Ackerman says: "We encouraged people to handle areas outside their specializations and lead projects Elbit had never before attempted. For example: we heard that the Air Force wanted to privatize its flight school, or the Firefighters Squadron, or the UAV project (unmanned aerial vehicle – the authors). At first, we did not even know what those were. But someone took it upon themselves to handle it. They would study the subject, gather materials and read the literature. This is how we won the UAV tender, and the Firefighters Squadron Operation tender. Compared to us, a niche company with no systems engineers would say: 'This, we do not do,' because they would not have the person who would say: 'I'm a systems engineer, and I know how to learn new subjects.' In that company, the systems engineer would be the CEO."

He brings the example of a niche company Elbit had acquired: "Tadiran Kesher made only basic radios that connected five or six modules – a small, focused niche. The company had maybe one systems engineer. This was not a disadvantage – one can make a lot of money focusing on a specific field."

He adds: "This approach is relevant to any area. For example, the more specialized a doctor is, the more he charges, and the less he knows. He must also say that he knows less. He even has to admit he has no understanding in other areas, so that he can focus his expertise. On the other hand, a general practitioner, who has to provide answers to any problem, is a systems engineer. He either knows the answer, or says: 'I will read up on this and get back to you with an answer,' or sends for an expert. A general practitioner is a profession, much the same way a systems engineer is."

All along the road, we find a significant overlap between the job of a manager and that of a systems engineer. It is, therefore, not surprising that in many companies, including Elbit, there are such job titles as "Technical Manager." The boundaries between the various position holders are unclear, and Yossi Ackerman sees this as a good thing. He argues that there is no benefit in setting clear boundaries, if constraints that arise force the boundaries into place anyway: "In Elbit, those assigned to fill the position of systems engineers do not necessarily have degrees in that discipline, but they do possess the relevant skills to the project. There are projects with dozens of systems engineers, but not all of them bear that title. It is, however, likely that as we rise through the system hierarchies, there is a higher chance that those who bear the title 'systems engineer' will also have the words 'systems engineer' written in their formal job description.

There is need for flexibility. For example, if Elbit employs one hundred systems engineers, but at a certain point in time, only has enough systems engineering work for sixty of them, the remaining engineers will perform other tasks, like those of technical or marketing managers.

The same principle holds for other jobs. For instance, when the head of a division has to choose a manager for an electronics project, he can simply choose the best electronics engineer available. But, if the project includes a range of disciplines, he might want to choose someone who is not as good at electronics, but is the best at systems thinking. If that engineer is good at electronics, but unable to make the necessary compromises when faced with various constrains, then he is not suited for project management."

Yossi Ackerman finds that whether or not someone is a systems engineer throughout their professional life, even if they do not bear the title, is a question of character. To demonstrate, he brings the example of Professor Ovadia Harari (see interview with him in this book), who, even as the Vice President of IAI, remained a systems engineer: "Of course, as vice president, he was a super-systems engineer, because in a position like this, one can only delve so far into the specifics, professionally.

Defining a systems engineer is fuzzy logic. It is not something that can be defined. Nor, in fact should it. It needs to be given space, and then defined in accordance with the given situation. A manager who looks at the whole technical and technological picture can be called a systems engineer."

He agrees with the statement that the higher the rank of the engineer, the more management elements his job entails.

More on systems engineering in Elbit

On the effects of the matrix structure: "The Lavi Project had employed hundreds of people from various disciplines. But, because of Elbit's matrix system (see also the interview with Mimi Timnat), those who worked with me were not subjected to me, but allocated to me, so that at any point, their superiors could decide to pull them away into another project. A systems engineer needs to know how to make people want to work for him. In addition to the technical part, he must have an understanding of psychology – of people. Some knowledge of economics and law is recommended as well."

On the training of systems engineers: "In the 90s, hundreds of people in Elbit were already practicing systems engineering. Not all of them were called 'systems engineers,' but in time, they, too, received the title. Then, we saw that their knowledge of physics or electronics was not enough; that they had to be taught systems engineering. And so, we began to devise additional training programs for them.

A good systems engineer can manage without furthering his studies. This is true for every field: there are exceptional teachers who have never studied pedagogy, and there are, of course, those who have. There are excellent systems engineers who never attended formal training frameworks. They possess the right qualities and are self-taught.

Nonetheless, continuing education programs have an added value. They have put things in order. They poured meaning into what systems engineers were doing. These programs also create a common denominator among Elbit's systems engineers, each of whom had arrived from a different unit. As time passed, because of the growing importance of the systems engineer's role in each project, we wanted to institutionalize the field and asked the Technion to create better-founded educational frameworks that awarded Master's degrees (see interview with Professor Aviv Rosen)."

On a systems engineer's professional background: "A systems engineer needs to have a technical background. He does not have to be an engineer; he can be a technician or a practical engineer. He can also be someone who has accumulated technical experience as a technical officer at the air force, because he understands the client's needs. But these situations are rare. In most cases, a systems engineer in Elbit has basic engineering training."

The Evolution of Systems Engineering

Yossi Ackerman first began to recognize the importance of systems engineering after the cancellation of the Lavi Project, in the 1980s: "The Lavi was invented by engineers. They did not ask the pilots what would suit them best. They said: 'We know which engine is best.' The client was silent, and that is what brought this project down. So we realized we had to become customer-oriented, rather than engineering-oriented. We had to give the client what he needed. We also understood that providing the client with a worthy solution required more than one discipline."

He describes the background of the emergence of systems engineering:

"In the past, there were washing machines that required two technicians to operate. Gradually, people understood that in order for everyone to be able to use a product, human engineering was needed too. If the machine were made by an engineer alone, its design would not require systems engineering. He would create an excellent motor, but the machine would rust, break down every 2 days, and cost four times as much. To make a good washing machine, one needs to know about water, to prevent it from being covered with scale after a month of use; one needs motors; stainless steel, to keep it from rusting; one needs knowledge of electricity, economics, marketing.

Over the years, companies began to understand that more than one discipline is needed to advance a project. Kodak, for example, is among the companies that had failed to grasp this. After all, a carpenter, who failed to realize that he had to learn a thing or two about metal, lost his livelihood. Today, architects study psychology, because one needs to know more than one discipline."

On the development of the patterns of working with clients: "For many years, our American clients would give us blueprints, and we would execute them – this was called the 'Build to Print' method. Later, the method changed to 'Build to Spec' – we were given a specification and asked to make a card that had all the functions listed in the specification. This method was also used widely for quite a few years. I introduced a different method, called 'Build to Need.' It means, understanding what the client needs, not what he wants. Some clients think they are very clever, and they will tell you exactly what is needed. In these cases, we do not have much left to do, and we lose our creativity. We would receive a specification: a thick book, a thousand pages long. We did not need to prove that our product was good, just that it met the specification requirements.

As time passed, clients understood that they needed to make their books thinner, so they explain what is needed, not how to do it. They can tell us that if a rocket is launched at us from the Gaza Strip, they want our system to intercept it within ten seconds, at a given distance from its target. This is fine, but they should not tell us whether the launching vehicle has to have eight wheels, or six wheels. Telling us that the operating unit has to number eighteen soldiers is also legitimate, but they should not plan everything."

On the direction systems engineering is headed: "One of the main problems is the unwillingness of experts to talk to others from outside their discipline. A doctor who specializes in internal medicine will not consult an orthopedist. If a patient hospitalized in the internal medicine ward has an orthopedic problem, he will be taken to the orthopedics ward, because the orthopedist will not come to the internal medicine ward. If, god forbid, the patient dies because of an internal problem, then, as far as the orthopedist is concerned, he died healthy, because 'his part' was fine. This is the very same reason economists failed to predict the economic crises, and political scientists failed to predict the Arab Spring. Had they all thought together, in interdisciplinary teams that included economists, political scientists and psychologists, they would have managed to predict these events.

But all this is about to change. Thanks to the internet, now everyone understands, and reads, and knows everything. Nobody trusts the experts anymore. Some of the writers in the world's leading journals today are architects, psychologists, not just scientists. It is beginning to sink in."

More Insights on Systems Engineering

On the relationship between systems engineer and project manager

- "The difference between a program manager (project manager in Elbit-speak) and a systems engineer is insubstantial. A systems engineer sees the entire project from a technical-operational standpoint. The program manager also sees the technical-operational aspect, as well as other things, such as the economic and legal aspects. For all that, there is considerable of overlap between them. A systems engineer does many things the program manager does. In small projects, one person fills both positions.
  The relations between the program manager and the systems engineer vary, according to the nature of each project and the people working on it. If the program manager is very economically inclined, the systems engineer will enter the field of economics as well, because he knows that if he creates a system that cannot be afforded, it will not be a good system. There is no structured framework defining the activity areas of each one."

On measuring a systems engineer's success

- "A systems engineer is measured by the success of his project. Uzia Galil (founder of Elron – the authors) once said: 'A great engineer is one who plans something useful that someone is willing to pay for.' If a systems engineer has made something no one needs, he is not a very good systems engineer. A good systems engineer must also understand the constraints. He needs to know that if he makes a technologically impressive system that requires 1,500 soldiers to operate it, that is not a good system."

3.1.3 "SEEING BEYOND TECHNOLOGY – UNDERSTANDING THE MISSION"

An Interview with John Thomas

Like the majority of professionals we interviewed, John Thomas, INCOSE President during 2012–2013, had already adopted the thought patterns of a systems engineer from the onset of his career as a young engineer. Like the others, he had no inkling at the time that he had done so. During the 1970s, he worked as an electronic engineer with the development teams of the US Air Force and there, he adopted the work patterns of a systems engineer – naturally – as someone who sees things not only from a technological perspective, but from a wider, systematic one.

After his work with the Air Force, John Thomas became a member of the Department of Defense industrial complex. During the 1990s, after completing his position as Head Systems Engineer within four separate programs, he joined the ranks of the large, international consulting firm, "Booz Allen Hamilton." Thomas climbed the company ladder and was promoted to serve as a Senior Vice President and Head Systems Engineer. He left the company in 2012 and founded his own consulting firm.

We spoke with him about the development of his career as a systems engineer and his viewpoints on the consolidation and nature of the discipline.

Personal Background

John Thomas is a clear example of an engineer who was not instructed or trained in the discipline of systems engineering. Rather, the systematic approach was an inherent and integral part of his basic thought processes. This approach assisted him to excel from the very start of his professional career.

He began his career as an electronic technician with the American Air Force. Until that time, the start of the 1970s, the electronic systems in these airplanes were mostly analog electronics. Technicians knew how to open damaged devices and repair the damage, or, if needed, to replace the damaged parts. This trend began to change with the development in computer capacity. The complex digital devices that had been designed were much more difficult to repair and, could not be repaired by the technician, but had to be shipped to intermediate depot locations for repair. The approach calling for the replacement of digital devices, opposed to repairing them, began to take hold as standard operating procedures for those technicians working on the aircraft.

John Thomas: "This change led to frustration among many technicians who saw their role being transformed right in front of their very eyes. They had been trained to examine the source of the electronic problem. Now they had to know how to decipher computer codes – which is very different from rendering an electro-mechanical solution."

Example:

Within the Air Force Wing in which John Thomas was employed, a problem was discovered in the inertial navigation subsystem of the FB-111 aircraft. When asked to deal with the defective inertial navigation subsystem, he did not limit his examination to a technological angle of that subsystem alone. He studied the specifications of the equipment and the ways in which the navigation subsystem had been integrated to interact with other subsystems within the aircraft. Thomas discovered that a constantly recurring problem of the navigation subsystem stemmed from the flow of signals between the navigation subsystem and other avionics subsystems related to it. He then studied the specifications of the equipment in these other avionics subsystems (e.g., the avionic flight instrumentation subsystem, and found the root cause of the problem. Flight compass signals were at times out of tolerance due to aircraft placement on the runway. These out-of-tolerance compass signals generated a failure during the start-up sequence of the inertial navigation subsystem). Thus, he was able to successfully tend to the problem, where others had failed to do so.

John Thomas: "It was not the professional knowledge, in and of itself, rather, it was my mindset. My mindset provided a means for system-level thinking that made the difference in fixing this problem. I made use of analytical tools and displayed curiosity regarding the interactions among the subsystems of the aircraft. I tried to understand how the algorithm of the larger system addressed the specific subsystem that I was working on. Even though I was an electronic technician, I did not limit my examination to the specifications of the electronic circuits. I also reviewed the sequence of activities that lead to the implementation of the task and the ways in which the different parts are integrated into it."

Later on, John Thomas studied electronic engineering and returned to work with the Air Force – this time as an engineer and an officer: "I continued working in the same manner. It was my job to solve systematic problems. If a system didn't work properly, it would be taken apart and its components would be examined in order to determine what was wrong. The algorithm would be examined and then it would be reassembled."

When did you realize that in essence you are working as a systems engineer?

John Thomas: "I did not know what a systems engineer was until the mid 1980's. We were working on certain project when members of the Department of Defense came to visit my unit in the Air Force. They presented the field on which they were working and it was quite evident that I understood the mission and technology as well as they did, and from a systems perspective, even better than they did. They recruited me to serve as the Head Systems Engineer of the project. That's how I formally changed from being someone who thinks systematically into a head systems engineer."

Systematic Thinking and the Human Perspective

John Thomas agrees with the distinction between a "pure" engineer and a systems engineer, claiming that an engineer sees the parts, whereas a systems engineer sees not only the parts, but also the connections between them that create a process. This process generates a new and valuable behavior that no one part by itself could provide. In his opinion, this type of perception is not solely a gift of nature – it can also be learned.

To clarify this position, he presents the psychological model developed by the research duo, Isabel Myers and Katherine Briggs.[2] Their highly acclaimed model, exposed in the 1970s, divides personality types into four categories of courses of action. These four categories of action are mixed and matched to form 16 combinations of personality profiles. These profiles reflect different behavioral patterns. Individual behavior can be characterized according to one of these 16 profiles. But, Myers and Briggs claim that this does not mean that a person will behave solely in accordance with the behavioral patterns that make up his personality profile. Rather, the personality profile reflects the mode of operation an individual feels most comfortable operating within. Individuals can and do operate with different personality profiles when required to do so for the success of their career. Operating outside their "preferred" personality profile can cause them stress until they master and become comfortable with the expanded set of skills required to be successful – but they can be taught these skills. Thomas feels strongly that this claim by Myers and Briggs is indeed true. He has gone through multiple training programs to make him more comfortable in operating outside his preferred personality profile. The result has been an expanded set of both soft and hard skills that have benefited Thomas's career.

In relation to systems engineering: Even though not everyone is born with the characteristics of a successful systems engineer, a person can train himself to be one. But, he might be a bit uncomfortable playing the role: "There are some people with systematic perception who notice the connections and sequences, but do not display any interest or enjoyment in doing so."

(Why would a person want to serve in a function that involves activities in which he feels "less comfortable"? Apparently, for a variety of reasons, for example: one could assume that a successful engineer, who values the importance of being promoted along the administrative hierarchy, might want to serve as a systems engineer, during an intermediary stage, in order to prove his capabilities in fulfilling this role on his way to being appointed to an executive position – the authors).

Thomas, who joined the ranks of the consulting firm "Booz Allen Hamilton" in 1991, said that if a systems engineer wants to be influential, "he has to be able to pinpoint the central track that influences the process."

He gives the following example: "We were consulting a client who was having fundamental problems with his internet protocol. We noted that the personnel spoke only in technological terms, about volume and velocity, whenever referring to the problem. I did not limit my attention to the technological problems, but also observed the body language of the client's representatives, who came from technological backgrounds. I noticed how they talked with their colleagues about the problems, both confirmed and unconfirmed. There was clearly much more at hand than mere technological problems. It might be very easy to adhere to the claim that the entire problem is technological, but that is never really the case. I asked questions and

[2]The four categories of the Myers and Briggs model: (i) The way a person processes the energy around him (sequences of introversion–extroversion); (ii) The way a person thinks (intuitive–analytical thinking); (iii) The way a person gathers data; (iv) The way a person makes decisions.
Each profile includes the four categories, in varying amounts, thus creating 16 different profiles (4×4).

noted that the responses were very dramatic. Also, everyone seemed to be addressing the problem with different terminology. It seemed as if they were talking about the same thing, but using different words. One of them called a certain phenomenon 'rust' while another called it 'brown'. Why were they behaving this way? It was just confusing matters. The explanation: each one viewed things from a completely different angle based on the different histories of their careers."

John Thomas sees himself as a systematic thinker: "A systems engineer needs the basic skills of critical thinking as well as engineering terminology. Today I see the complexity of the system more than the solution to the technical problem. Today, contrary to the past, I cannot offer a quality evaluation of an electronic panel. However, I am highly capable of technical designs. I have the basic knowledge to ask the right questions. I can tell if the person who responds to them knows what he is talking about. I have the ability *to see beyond the technology and to understand the problem*. I also have an understanding of the mission or business challenges that the technological system has to solve."

He emphasizes the importance of understanding the human component: "A good systems engineer sees not only the technical parts, but notes the human components, as well. As a systems engineer you acquire the skills to synthesize the data, to examine the mutual relations between them and to understand the considerations of the individuals within the group.

This ability enables a systems engineer to speak in terms of 'brown' with one group and in terms of 'rust' with the other. In this case, the systems engineer becomes a highly intensified interpreter. In the previous case, people were speaking a different language and they were frustrated. They came from different units within the company and were at odds with each other. We were the ones that clarified to them that they were actually talking about the same problems and were all concerned about the same things."

This incident illustrates one of the primary challenges that systems engineers are faced with – to be able to recognize the real problem in need of a solution, while in most cases it is not the problem declared by the client.

Thomas cites another example: "About ten years ago I worked with a group of people who defined themselves as systems engineers. We had been asked by the management of the organization to strengthen the abilities of their systems engineers. When we met with the group, it became apparent that the problem was not the lack of abilities among the engineers, rather the lack of understanding and agreement at the managerial level regarding the implementation of the tools available to systems engineers for solving problems. There had been a council of systems engineering, but it was abolished because senior management did not understand its relevance.

As consultants, we were asked to improve the skills of their systems engineers. In reality, the level of understanding among members of the senior management and the ways for incorporating the current staff of systems engineers were the matters in need of improvement. Once this happened, they were able to use the systems engineering base that they already had, which was actually quite good, to continue to develop. But, the next part of the assignment was not so simple. After the real problem had been identified, the group best suited to solve the problem had to be selected. The existing staff of system engineers, on whom the improvements were to be focused, was comprised of frustrated personnel who had felt unappreciated for quite some time. The problem was that they were not the right people to teach the members of the senior management how to make use of systems engineering, evidenced by the fact that they had not been able to do so during their years of employment there.

This group felt the lack of appreciation and suffered from demoralization. Therefore, the central task was to change the attitude of the group. We had to show them that their attempts to influence the management had not been effective because they had relied on only one set of tools in order to do so.

The members of the group had to realize that since they had not been successful in achieving their goals by using their current tools, it would be senseless to continue along the same pathway, doing the same things over and over again. They had to approach the problem in a different fashion. We sold them on approaching the problem in a different fashion by appealing to their greatest skill. As system engineers their great skill was to think systematically about a problem. They soon saw that they had a socio-technical problem. Not simply a technical problem. Once their viewpoint of the problem was changed by this expanded view of the system they quickly embraced different approaches to attack the problem from which they were accustomed."

If the problem is behavioral and not technical, isn't that a classic example of organizational consultation?

John Thomas: "Yes, but for a group of systems engineers. Telling them to go and check to see what the demands are, which is a clever way of saying: 'go and check to see what the real problem is,' is an approach promoted by systems engineering. It is not a technical process. In my understanding, it is the curiosity of the individual to view interconnectedness as part of the task. All along the way, the systems engineer comes into contact with human beings, in various aspects. The job of the systems engineer in these situations is to ask simple questions and to listen. This enables him to raise additional points and to examine the interconnections.

In many instances, organizational consultants are equipped with the skills of asking and listening. But, not many organizational consultants have the skills of a systems engineer of interpreting the problem from a systemic view point. In many cases, organizational consultants view these kinds of problems primarily as being problems of interpersonal dynamics."

According to John Thomas, a great amount of additional work is needed for technological organizations to understand that technological systems cannot be managed separately from human systems. The two are intertwined.

He recalls: "We discussed this during a visit to one of our clients. They said to me: 'you hit the point. We have many good engineers and many good systems engineers. However, some of our great engineers do not have the social skills required to form coalitions and work across multidisciplinary (technical and nontechnical) teams. This creates monumental problems.' We had been developing their technological abilities, but had never implemented the concept that they need to serve as facilitators by means of integrating human talent.

Wherever I go throughout the United States, and when I meet with a company vice president or a senior vice president, I find that if the discussions begin with talks about technical problems, they end up with talks about systems engineering – but in a different way – as if by hinting at the subject. Interpersonal dynamics are not even mentioned. Only the technical problems are voiced. You won't hear a word about the fact that they don't know how to work together. When asked what their fundamental problems are, they often tell you: 'My engineers are struggling to resolve a specific technical problem'– you won't hear – 'My engineers create social conflict. They don't know how to navigate through conflict to arrive at a more effective solution.'"

About Leadership and Systems Engineering

The understanding of the centrality of the human factor in the world of systems engineers led to placing an emphasis on the importance of leadership – the leadership qualities of people – as an important component of the discipline. This issue is also addressed by others interviewed for our book (see the interview with Norman Augustine, for example). John Thomas is an enthusiastic supporter of this position. He was quite alarmed when we asked him if he, in his position as Head Systems Engineer, is actually a manager.

John Thomas: "No, certainly not. I am not a manager. I am a leader. I allocate managerial responsibilities to others. Managers see people as production units. 'How many hours did you put in?' 'I did a performance evaluation on you.' In contrast, leaders see people as human beings. Leadership comes as a result of a world-view. It is not 'you did fine here' and 'you didn't meet timetables there.' A man is not a machine. Once I was that kind of manager. I had left the wounded behind and was on the verge of being fired. It was a crisis that brought about a fundamental change in me.

A leader is able to have people follow him because they want to be part of what he is leading toward. The team gives him power over them – through his authenticity and transparency of purpose. A leader sets the standards and the manner of behavior he expects to be conducted and then demonstrates by living to those standards himself. By doing so, he provides others with the fundamentals of systems engineering."

Leadership is extremely important for systems engineers because the variety of skills and actions required for developing and building a complex system is so great that only a group of individuals working together can do so. Therefore, leadership is required in order to ensure that their actions and efforts are successfully coordinated and integrated. In his lecture, John Thomas states that "technological leadership is not an option for a systems engineer, it is an absolute necessity."

The Rational and Intelligent Use of a Variety of Qualities

It can be deduced from Thomas's words that a systems engineer should be endowed with a variety of qualities – qualities that sometimes contradict each other. As such, he needs to maneuver between them. Increased awareness enables him to do so, even if the role is less comfortable for him (according to the aforementioned Myers and Briggs model).

We illustrate our meaning by listing and describing a set of rational and intelligent qualities that a good systems engineer should be endowed with, according to Thomas:

In his opinion, a systems engineer should be able *to move from domain to domain* by means of a wider perspective of the overall system. The desire to do so stems from natural curiosity and the ability to recognize situations in which it is possible to do so. In order to get into the thick of things, a systems engineer has to know how *to ask the right questions* and to delve deeper into the matter (a quality that sometimes requires a display of assertiveness and the ability to insist on receiving a fitting response). On the other hand, a systems engineer also needs to adopt a humble and modest approach, certainly during his entry into a new field, and to refrain from behaving as if he is Mr. Know-It-All. *He should acknowledge the fact that he does not know enough and, at times, admit that there are things he does not understand – and then ask for additional clarification*. (Assertive people are not fond of finding themselves in such situations and are reluctant to admit their inferiority – even it is temporary).

Isn't this similar to the situation in which the manager of an organization in a certain industry is transferred to manage an organization in another industry? Isn't it like those who claim that management is management, under any conditions? So, similarly, systems engineering is systems engineering, regardless of the situation.

John Thomas: "Whoever thinks like that is making a big mistake. Even in management it is not the same. A manager who does not understand the field in which he operates, the tasks and the products, and who thinks that management is primarily about making financial decisions, will lead the organization to disaster. If the Head Financial Manager becomes the General Manager of the company and thinks that it is simply a matter of profit and loss, he will destroy the company. On the other hand, if he invests time and energy into processes and into the development of new lines of company products, he could have a very powerful influence on the company.

You can't move from domain to domain if you lack the humility to understand that the basis of your managerial knowledge is insufficient. It will suffice only after you have studied the field in which you are to operate. A systems engineer knows how to begin to build the box, but he cannot make decisions within that box until he has collected data and learned from others. In these situations, he has to surround himself with people who will help him to absorb the basics of knowledge required to turn him into an expert. A systems engineer has professional knowledge, but he must remain humble and be aware of what he does not know. This way, he can learn a great deal and he can learn quickly.

I have been involved in multiple domains during the course of my career. I have been successful in them because, when moving from one domain to another, it took at least a year to reach the previous level of problem solving. At the start of my involvement in each domain I lacked an overall perspective of the phenomenology of the task."

Another important quality of a systems engineer is the *ability to simplify complex matters* (see the interview with Kobi Reiner):

John Thomas: "The ability to simplify things enables a systems engineer to serve as a problem solver. I am able to get to the root of the problem within a short period of time, even if I am not proficient in the field. And, I always have an expert in the field nearby. It is a combination of methodology and intuition. If you rely solely on methodology, you will not find a solution. You need a process to help you direct your intuitions."

In his lecture, John Thomas defines the six basic skills of a systems engineer as follows:

1. Craftsman: knowledgeable in the processes, work methods, and technical work tools.
2. Functional: combines the ability to delve deeply into matters with a wider perspective.
3. Programmatic Understanding: this includes the abilities of contract planning and communications, planning expenses, and the ability to assess timetables.
4. Leader: the ability to function under pressure; to manage conflicts; to make decisions; to know how to communicate, form, and lead work teams; and to be a man of vision.
5. Problem Solver: the abilities of critical thinking, systematic thinking, and associative thinking, and the ability to simplify matters.
6. The ability to understand one's surroundings: familiar with the technology and the fields, building a suitable life circle, and understanding the tasks at hand.

The Training of a Systems Engineer

Academic training programs for systems engineers are usually at the Master's Degree level of study. The widespread opinion is that only an engineer with experience in an industrial field can really learn and internalize the fundamentals of the discipline. One of the problems that the field is faced with is that basic engineering studies focus on the technological plain. We asked John Thomas, who agrees that this indeed is the situation, while he feels that engineering schools follow this line of focus.

He responded: "When learning about circuits during my studies of electronic engineering, I learned how to design an oscillator. But, I did not learn how the oscillator takes part in a broader system or how it should be situated onto an electronic panel. I did not learn how a number of such panels fulfill a much broader function. No one ever explained to me how to integrate my professional skills into a broader context. The same can be said today. When I talk about this with the Deans of Engineering Departments, their response to me is that they do not have enough time to integrate all of the engineering fields that should be taught into their academic programs. I tell them that I am not asking them to add more courses, rather to instruct the lecturers to present the context of how course work fits into a system. So that their engineers, be they chemical, mechanical or mathematical engineers, who have a natural tendency for the subject, will say: ‘great, now I know how to do this technically, but I also understand how it fits into the broader picture.’

I tell them that there is no need for an additional courses on system engineering or on leadership at the under graduate level. However, I am concerned about the fact that they come into contact with so many engineers and never mention that powerful design (mechanical, chemical, electrical, …) comes from a system perspective and that the power to influence decisions comes through leadership skill. They don’t display enough systematic thought in the classroom. The important point is whether someone who finishes school is curious about the significance of his work.”

3.1.4 “SIMPLIFICATION CAPABILITIES IN A COMPLEX ENVIRONMENT”

An Interview with Dr. Kobi Reiner

We have already described Rafael’s activities as the developer of the Surface-to-Air Missile Defense System – “Iron Dome.” This chapter will discuss the systems engineering characteristics as being expressed in one of the earlier, “air-to-surface” missile projects, developed during the late 1990s and early 2000s, while placing an emphasis on a systems engineer’s professional development within the project.

Before the Project

Risk Reduction. Rafael has two sources for its knowledge creating processes. The first is “top down,” achieved by financing the business directorates that perceive future operational needs. The second is “bottom up,” it stems from the engineering teams, where the engineers try to improve operational response in two ways: by presenting new technical and/or algorithmic capabilities, which are then revealed in professional magazines and conferences; and by optimizing the efficiency of the processes that improve the product’s performance and increase its reliability. The funding the “bottom up” activities receive is negligible, seeing as most of the initiatives that originate from the lower levels stem from the engineers’ need to challenge their own intellect in an attempt to bridge their professional knowledge gaps, and the need to make development processes (which are not funded by the projects) more efficient and reliable.

In the professional jargon, these are known as “risk reduction” processes. They are called that because, when an order comes in for the development of a relevant project, where the knowledge created at the “lower levels” can be made use of, it helps minimize the technological and process-related risks that otherwise might hamper its success.

Development initiatives that originate at the lower levels occur partly due to encouragement from the organization itself, and partly due to the engineers' natural professional curiosity. For instance, an engineer can return from a professional conference, having discovered gaps in his knowledge. He feels he must now fill these gaps and go in search of solutions. Or, another engineer can find a certain technological capability possessed by competitors while browsing through professional publications, and try to duplicate it for Rafael. These initiatives are part of any engineer's professional development, and pose substantial professional and operational challenges. In many cases, such initiatives form collaborations within Rafael, or evolve within the framework of existing professional workgroups.

"Bottom up" development initiatives that receive no funding from the business administration units are known throughout Rafael as "Black R&D". Kobi Reiner, a chief systems engineer in the R&D and Engineering Directorate at the Missiles and Network Centric Warfare Division, says that a substantial part of the early "Black R&D" initiatives do indeed find their uses.

He explains and illustrates: "The general risk factor is comprised of a quantification of three risk components: a technical component, failure to adhere to the schedule, and the possibility of the risk being realized. For example, the United States has very accurate navigation systems, and we want to achieve this level of accuracy as well. During the project's launch phase, there are, as yet, no systems in Israel with that capability, but work is being done to improve the performance of existing systems. Since it is impossible to know for certain that the mission will be a success, this constitutes a risk assessment – an assessment of the company's ability to achieve the desired performance level.

The risk level is, of course, derived from the complexity of the task. If, for example, somewhere in the world, a certain new technology emerges in the field of missile guidance systems that use image matching, the task of integrating this capability into our missiles is considered relatively simple, thus making it a 'low risk' project. If, however, we want to develop an entire missile that includes the new navigation system, a new engine, new wings, a new homing head and other new elements, and then integrate all these as well, the task will be considered a 'high risk' project."

Learning of Capabilities and Creating the Technical Specification. IAF operations had defined the special operational needs for an air-to-surface missile, and presented them before those responsible for developing the armaments for IAF. This group initiated preliminary talks with representatives of the Rafael Air-to-Surface Directorate – the professional unit tasked with the development of the missile. The purpose of these talks was to assess Rafael's ability to meet the operational requirements. IAF representatives met with Rafael experts as needed (some were "risk reducers," whose developments were deemed relevant for the project) to learn about the options for each of the missile's subsystems. The output of these meetings was the definition of the missile's technical specification.

Kobi Reiner: “Risk reduction does not mean risk elimination. In this project, before the FSD (Full Scale Development – performed in accordance with the previously formulated specification), we spent a year and a half working to reduce avionic risks (the missile’s avionic capabilities, such as accuracy and the ability to successfully home on the target) and aero-mechanical risks (the missile’s engineering capabilities, such as the ability to perform and structurally withstand maneuvers). Nonetheless, it is important to stress that designated risk-reduction activities allow the project to enter the FSD phase with reduced, if not fully eliminated, risk factors.”

The technical specification document accounts for the findings of the risk reduction phase in two ways: the capabilities demonstrated at the risk reduction phase are integrated into the document as requirements, while high-risk capabilities (for which there is no certainty of accomplishment) are omitted from it, or labeled “optional.”

These pre-launch phases entail no commitment on the part of the potential buyer. However, there are cases when the Air Force chooses to invest in the project financially, even in this early stage of assessing Rafael’s existing capabilities.

Kobi Reiner: “At this point, before the project is underway, the systems engineer needs to be a professional as well as an entrepreneur, intimately familiar with the professional environment, because his main goal is to define technological ability, and assess whether the project can be realized at all. This is why, in the early, risk reduction stages, it is not necessary to dote on the small details of the realization. Rather, it is important to concentrate on proving the company’s technological realization capabilities.

This process can begin with the creative, slightly disorganized individual. The further into the project one gets, the more one requires a management-oriented mind. The nature of the ideal systems engineer changes as the project advances. The ability to put the right systems engineer in the right place makes the difference between success and failure.”

The Project

Phase One: A Disciplinary Systems Engineer

The Organizational Structure of the Project. The new missile’s development included two interwoven projects: the avionics project, and the aero-mechanics project. Both project chiefs reported to the Head of the Program. Each project included disciplinary systems engineers, each responsible for the development of all the elements in the subsystems in his area.

Kobi Reiner: “Today, the components of a system are becoming more complex and more convoluted on all the technical levels; so, understanding them requires familiarity with more engineering elements. Consequently, the systems engineer’s work becomes more complicated. The multitude of engineering elements makes integration difficult.”

Part of Kobi Reiner’s job as a disciplinary systems engineer on the aeronautics project was managing three “leaders” at Raphael, and heading an external subcontractor’s designated project. Each of the leaders served as a development engineer and was in charge of developing a component in a subsystem, for which Kobi was responsible. The "leaders" are called thus, because they are fully responsible for leading the development of the component under their charge. Each leader, in turn, manages professional development personnel: the developers.

Kobi Reiner: "The 'leaders' are not systems engineers by title, but in order to realize the component under their charge, they, too, must, in effect, do systems engineering and project management work. The integration of two or more areas is, in itself, a task for a systems engineer. For this reason, in small projects, one person can fill the position of both leader and systems engineer."

To summarize, the organizational structure of the new missile's development had five levels: the head of the program, the heads of the two projects, the disciplinary systems engineers, the leaders, and, finally, the developers.

One of the complexities of a large project is the question of organizational reporting. The further down the organizational structure one looks, the higher the chances that the employees he finds will not be an integral part of the project; rather, they will simultaneously serve the needs of other projects, and have other tasks, related to their expertise. In this case, the three most senior ranks, including the disciplinary systems engineers, were exclusive to the project. The leaders and developers, however, acted in service of the project, from within their respective professional departments.

Kobi Reiner: "I set the technical objectives for the leaders, but those who managed them were their superiors within the engineering array. In our case, in the early stages of the development phase, which demanded engineering-oriented focus from the team, all of the leaders' time was dedicated to the project, but some of the developers beneath them were assigned to work on other projects as well. A situation like that might create management constraints, caused by the developers not being sufficiently available. The leaders were not always able to resolve these problems, and so, when they believed I would be better suited to deal with the superior of the developer in question, they asked me to intervene."

On the disciplinary engineer's work methods:

Kobi Reiner: "During the initial processes of formulating the work patterns for the FSD phase, the systems engineer must be very well-organized. This is both so people can have a clear, well-defined framework of the work ahead of them, and so that the engineer himself can write the system requirements specification along with them. This activity took the form of joint work meetings. I support the team meetings approach, because it minimizes the risk of things falling between the cracks. I formulated the overarching requirements specification for the system under my charge. The three leaders then derived the requirements for the components under their charge from this specification. For this, a leader would derive the requirements for his component from the overarching specification, and then construct yet another requirements specification, which would, in turn, serve the developers working under him."

Although a disciplinary systems engineer directs a sizeable group of people, the bulk of his work is systems engineering related, mostly because said people are not directly subjected to him.

Kobi Reiner: "Most of my work was engineering and coordinating between the different persons that performed the work for me at different (physical) locations.

One person's output was another's input – for one leader to be able to move forward with his tasks, another had to complete his own task and provide the first leader with its results. My job, as a systems engineer, was to coordinate and synchronize these actions.

I was also required to coordinate laterally. Thus, for instance, from time to time, I had to check my own systems, for which I needed testing equipment, which was the responsibility of another engineer who worked on the project. He did not always find it convenient to provide me with the equipment whenever I wanted it. This and other dependency situations between the inputs and outputs of the engineers in the project are examples of the constraints I was faced with. Of course, any problem can be brought before the project manager, so that he can use his authority to resolve it. But a good systems engineer who owns his actions wants to solve his own problems. He will only turn to the project manager for these kinds of solutions as a last resort. His first duty is to try and overcome the hurdles himself."

Phase Two: Leading the Process of Obtaining the Avionic Certifications

Taking Responsibility. Kobi Reiner's advancement within the project hierarchy was not planned. Rather, it was a direct result of his work and the initiatives he had led. When a task requires a lot of coordination, as is typical of systems engineering, many things fall between the cracks. Those who bother to "pick them up" stand out for the better.

The head of the program asked Kobi Reiner to take up another, relatively small task, that did not justify assigning a full-time disciplinary systems engineer. The task was to supervise and coordinate the work of a subcontractor – IAI. In this assignment, he no longer represented his own, avionics division, but the entire development program.

The project kept moving forward, and entered the phase of the first system trial. The trial was comprised of rendering digital simulations of the future missile's performance. This required synchronization between the testing equipment, computers, and simulators.

Explanation: the approval processes of a system trial include the important stage of going over the hybrid simulation (a simulation that combines operational hardware and software – the authors). In order for the tests to be realized in the hybrid environment, there needed to be full synchronization between the electronic software specialists who had constructed the special equipment used for the hybrid simulation, the leaders of the various operational units, who had to bring the products of their work into the hybrid approval environment, the software specialists, and the algorithm specialists.

Kobi Reiner: "At this stage of creating the integration, I was more dominant than the other disciplinary systems engineers. The nature of integration is such, that you try to put things together, and they refuse to work. Not everybody has the ability to manage the big picture and deal with the small things at the same time. Some are better at planning and writing, others are better at making the connections. A systems engineer can be creative, he can have a talent for designing the architecture, but unable to lead the development of the final product. I took the responsibility because I wanted the system to succeed. I got into it naturally, without being officially asked to. I found myself in meetings with many leaders who were not assigned to me, and I managed the convergence process. I became a sort of lead systems engineer for the avionics project. Actually, I managed the technological integration."

Did that not disturb the other disciplinary engineers?

Kobi Reiner: "It was done in good spirit, because there were people there, who did not place their egos before everything else. They recognized that they were unable to do what was needed to complete the approval processes, and I thought they were glad that someone else came in and prevented their systems from failing. Of course, had someone objected, I would not have gone near his people – if he did not want help, he would get no help. But that did not happen, because the head of the program had been wise enough to select people who were both skilled professionals and amiable individuals."

In time, Kobi Reiner's actions caused him to be formally responsible for leading the avionic approvals. Here, too, he based some of his work on joint meetings, such as he had in the project's early stages: "When people talk among themselves, they fill in the gaps. Had I spoken to each of them separately, I would have risked leaving some of those gaps unfilled. On the other hand, in crowded meetings, there is always the chance of what we call 'junk requirements' coming up – someone has a less-than-relevant idea, and he believes it's important that we look into it. Then, he needs to be told, politely, that his demand is off-topic.

A systems engineer has to be able to cut, to know when he is spreading himself too thin. Engineers tend to complicate things, and it is his job to stop them, because today, the possibilities are endless. One of the key traits a successful systems engineer must possess is the ability to simplify, when the atmosphere is one of complexity and complication. A good systems engineer prevents complication from emerging."

Phase Three: Chief Systems Engineer

Three years after its launch, the structure of the program was reorganized: avionics and aeromechanics were merged into a single project, to improve both projects' integrative response. Kobi Reiner was appointed chief systems engineer of the extended project, and, effectively, of the entire program.

In this new position, did the weight of his management work increase at the expense of his engineering practice?

Kobi Reiner: "As chief systems engineer, my job was still, first and foremost, to solve problems. What did change was how far down into the system I could descend. This was not just because of the position, but also because most of the engineering problems are found in the early stages of the project. The further along the project was, the shallower the engineering issues became. At that point, we dealt less with innovations, and more with approvals and tests. But, even as Chief Systems Engineer, the bulk of my work was still engineering related. I did, however, sense that I had much more responsibility, and of course, I met with higher ranking representatives of the client. Another part of my job was coordinating the processes of structuring the new missile's support and maintenance systems. This included writing the technical documents and establishing the training frameworks."

Problem Solving

Solving the problems that come up in the course of the work is one of the systems engineer's cardinal tasks in every stage of the project. As the development advances, changes bear heavier consequences. The most difficult problems arise during the convergence processes before testing. These processes include, among other things, the planning of the right combinations and integrations for the performance of additional tests during the convergence, intended to reduce the chances of failure during the main trial. Some areas, such as flight control, are impossible to test in the early phases and must wait until after the missile has been "launched."

Kobi Reiner describes the process of handling a problem that had come up in the later stages of the project, very close to the day of the trial: "There are times when the interim tests fail to detect a faulty system, and the problem is only discovered during the final test, right before the scheduled time of the trial. Keeping up with deadlines is very important, as is meeting performance requirements.

Often, in these situations, the problem can be located and a solution presented quickly, but in some cases, the leader and his people announce that they cannot resolve the problem in time. What then? Do we enter the process of studying the problem and fixing it, knowing that the analysis alone should last three weeks, not to mention the repair process, which might take another two months?

The answer is to find a bypass. We do not resolve the problem, but search for another solution that would circumvent it for the purpose of the trial, allowing us to meet its objectives. After the trial, the full-scale solution is applied, to be tested in the next system trial. Sometimes, in these situations, there is no choice but to sacrifice one of the trial's objectives (this is, of course, done with the buyer's approval).

This example illustrates the fact that a chief systems engineer needs to be constantly making decisions. One who is unable to make decisions cannot serve as a project's systems engineer.

We had quite a few such dilemmas in our own development program. For example, we had to converge the work of three leaders for an integrative software approval (even if each leader separately approves the software developed by him, it does not guarantee that the three will play together), and one of them turned out to be one week late. As chief systems engineer, I was faced with the following dilemma: either to delay the whole process by one week, or improvise a solution that would move the process forward and minimize any damages caused by the delay. I solved the problem by asking the leader who was off-schedule to write a program that simulated his computer, in one day. That way, we could keep going without his software, 'pretending' his computer was there."

Constraints such as these can also create management problems, such as those that stem from the project's organizational structure. For instance, the systems engineer puts pressure on the leaders to keep up with the schedule. If a delay, like the one in the aforementioned example, occurs, and is not resolved quickly and efficiently enough, a situation might form, where the other two teams are idle. The leader of one of those teams might complain to his professional superior at the engineering array, saying that he had been stressed for nothing, and had nothing to do. In more severe cases, the team that accomplished its mission and is waiting for the late team to catch up is given work on another project in the meantime. It then becomes much more difficult to bring that team back into the fold of the delayed project.

Kobi Reiner: "Even back when I was a disciplinary systems engineer, one of my main goals was to get to the end of a project with minimal stomach-aches on the developers' side. It is important to hear what is on their minds; it gives them a good feeling. I never gave the developers instructions without going through the leader, but I approached them to hear the goings-on."

The Completion of the Project and Further Insights

Kobi Reiner had left his position as chief systems engineer, before the development of the missile was completed, and was appointed another high-ranking position at Rafael: chief systems engineer of the engineering array. Next are some of his other insights on systems engineering.

On Being the Chief Systems Engineer of a Project, and Being the Chief Systems Engineer of the Entire Engineering Array

In Rafael, chief systems engineers are assigned not only to projects, but to the engineering array as a whole as well. The company's engineering array includes specialized professional departments: an electronics department, a software department, an electro-optics department; each with its own chief systems engineer. The chief systems engineer of the engineering array is not a chief systems engineer on the operational level. His occupation is more focused on forward thinking – on improving technologies and optimizing development processes. Project schedules tend to be extremely demanding. A project's chief systems engineer is therefore more of an operational engineering manager. The chief systems engineer of the engineering division, on the other hand, is more focused on the professional, process-related aspects.

Kobi Reiner: "I am an aeronautical engineer by trade, but my job is to be the chief systems engineer of a division that includes many engineering departments, other than aeronautics. My business card says 'Senior Systems Engineer.'"

On the Essence and Evolution of Systems Engineering

Systems engineering is connecting components together, in order to receive new functionalities. You take a body, an engine, a navigation system, a homing head – things that have no relation to one another – and you stitch them together to make them function as a missile.

Systems engineering includes management elements within it. This is why, for a systems engineer, the transition to management is easy. If a systems engineer tends to manage anyway, the transition will be natural for him. At Rafael, it is a common course of development.

There is a range of different systems engineers. Some are architecture gurus, who do not go into the integrations that bring the system to flight condition. They make only conceptual stitches, coming up with new algorithms or new technological concepts. Other systems engineers define processes and combine abilities up to the trial phases. On the project level, too, there are chief systems engineers who are nearer to management, and others who feel more at home with engineering.

Today, everyone wants to complete many projects on a short schedule and slim budget, but without sacrificing quality. The multitudes of projects, on the one hand, and the limited pool of systems engineers, on the other, create a situation where the systems engineering in a project is "thin."

One of the main problems is that not enough effort is invested into planning ahead. The heavy operational loads leave no time for people to read the materials. These are the situations where systems engineering needs to understand the benefits of advance planning, and use it to prevent failures.

On the Uniqueness of Systems Engineering in Aeronautics

Aeronautics studies are unique, because they teach many different subjects, but perceive the airborne platform as a super-system. It is, essentially, a multidisciplinary field.

There is a kind of systems engineering within aeronautics that handles constrains other engineering disciplines are unable to meet. An aircraft needs to be safe and, at the same time, able to fly – these are two opposites. Make a perfectly safe aircraft, and it will never fly. Make an aircraft that flies, but is not safe enough; and you might not land. For this reason, the aeronautics field has long since handled systems engineering problems, that other engineering disciplines have taken long years to resolve. Aeronautics experts have these solution methods built into their thought processes, while other disciplines are still struggling with ingraining them into their methodologies.

Nevertheless, in the software field, the dominant approach is the exact opposite. Software specialists are normally not interested in the ultimate goal. They have always said: "Define the algorithm for me. I don't care whether it makes the wing move or not." They see everything through the software; physical issues do not concern them.

Some algorithm experts are systems engineers, and some are not (because work on the architecture starts even as early as at this point). A flight control specialist writes the algorithm, but he should also care which hardware it will run on. When he writes the algorithm, he should perceive the required performance and account for the environment, in which the algorithm will be realized.

Many systems engineers come from the control field. This is also due to the nature of the discipline. In aeronautics, control is more systemic than in other fields. Integration is, by definition, what it does; whether you call it that or not, this is systems engineering.

3.1.5 “COMPLEX MEGA-SYSTEMS THAT CANNOT BE SUPERVISED”

An Interview with Hillary Sillitto

One of the main forces that drove the development of systems engineering was the development of technological abilities that, with the help of software, allowed for the creation of more and more complex technological systems. This created a need for a technological professional responsible for integrating the subsystems that form the complex system.

Technological systems are an integral part of the modern world. They provide a variety of services and integrate themselves into larger systems, some of which include non-technological components as well. Thus, super-systems are created, which cannot be controlled because they are intertwined with human systems.

The development and consequences of this phenomenon, and the place of systems engineering within it, were the subject of our conversation with a systems engineer, who is also a senior executive in a multinational technological company.

Personal Background

After graduating university with a degree in physics in the 1970s, Hillary Sillitto was hired to work for Ferranti in Edinburgh, a company that developed systems for the aviation industry. He had decided to study physics because he thought that field would provide him with the best foundation for his future career. According to him, physics lays down the foundation for seeing the commonalities between different kinds of systems and different types of problems.

His first position at the company, where he continued to work for 18 years, was as an optics engineer in a laser systems group.

Hillary Sillitto: “At the time, lasers were a relatively new technology. I joined the company about 15 years after these systems had first been implemented by the Americans in Vietnam. The IRF ordered laser systems from the company. A new department was founded, and young engineers, I among them, were recruited into it.”

How does a physicist find himself among engineers?

Hillary Sillitto: “Most engineering disciplines rely on physics. At its core, optical engineering, lasers included, is physics. An optical engineer must understand the physics of optical systems. We had engineers from different disciplines at the department. Electronics engineers and mechanical engineers handled the structures of detectors and receivers, while physicists worked as optical and laser engineers.”

He adds: “Not all physicists are theoretical. There is a spectrum ranging from the theoretical to the applied. The gap between applied physics and engineering is small, so, in effect, we were optical engineers or laser engineers. For instance, my job was to design the optical parts of the system, like the telescope that had to produce the structure of the beam.”

While he continued on his career path at the company, Hillary Sillitto also studied for a master’s degree in applied optics. He worked on a number of projects in the area, as an optical engineer on some, and was responsible for a wider scope of systems on others. Even as he performed these tasks, he continued being an optical engineer, but, in hindsight, these tasks possessed clear systems engineering characteristics. Gradually, he became a systems engineer in effect, a definition he embraces to this day. As he sees it, this way of development is a product of his natural tendency to provide system wide solutions to the needs of different types of clients.

At the time, which was during the 1980s, Ferranty saw systems engineering as a collection of work methods, rather than a profession or a job title on its own. It took years of working for the company before the word "systems" found its way into Sillitto's job description, when he was appointed Chief System Concept Manager.

In 1993, Hillary Sillitto changed jobs and began working for Thales Electronics – UK, an international company specializing in the aviation defense and security field, and employing roughly 70,000 employees worldwide (with its headquarters located in France).

Sillitto: "Recruited mostly due to my knowledge of systems, I joined the electronics division. At the time, the company wanted to provide a proper response to the market changes. Up until then, it had been a supplier of optical systems. However, there was a lot of competition in that area, so the management wanted to rise to a higher place in the supply chain, by being able to provide our clients with more complex systems (systems that include several integrated subsystems). These systems had higher added value. Aircraft had become more and more complex, packed with more electronics. We saw that clients were looking for ways to integrate different products, and the company was preparing to respond that need."

After 5 years of working for Thales, his job description was formally changed to systems engineer. He was appointed chief systems engineer and became the manager of a group of multidisciplinary experts that provided support for a number of the company's projects. At the same time, he was also part of a team that developed systems engineering methodologies for the company.

Around the time of the interview, in the year 2013, Sillitto was promoted to Thales Fellow (a senior position in the professional hierarchy) – a sort of organizational consultant who travels around the world and helps the managers of different important company projects to adopt more systematic ways of thinking. Up until that time, he had been the head of systems engineering for Thales UK.

Hillary Sillitto: "In that position, I saw myself as a systems engineer in a management position. Alongside my managerial work, I also performed technical surveys of projects, so I was responsible for tools and processes as well. I tried to allocate as much time as I could to the projects, to professional work, to thinking about critical problems and ways of solving them. It was a combination of management and engineering. Like sometimes telling people not to run too fast until they fully understood the issues they're dealing with."

The Increasing Complexity and Open-ness of Systems

Hillary Sillitto points out a phenomenon that has a significant effect on how today's works is run. It is a broad view that goes beyond the technological world, although technology greatly impacts the occurrence of the phenomenon itself. It is multi-systemic, the ever-growing, ever-accelerating emergence of interdependent super-systems.

In this chapter, we will present the main principles of this phenomenon, as perceived by Hillary Sillitto and based on his experience.

Hillary Sillitto: "Most problems that are found in complex systems are caused by people, rather than technical failures. We understand technology far better than we understand the way people behave. We understand the engineering aspects, but not the socio-technical aspects, because at no point in history have we continually adopted so many new technologies so quickly." He agrees with the statement that systems are becoming more and more complex, among other reasons, simply because they can be developed (not because they are needed, but that is a discussion on a different topic, which our book does not address – the authors).

Hillary Sillitto: "We connect systems together, creating mega-systems, because they allow us to do things we could not do otherwise, to solve problems that cannot be solved otherwise. They allow us to do things better, or to develop new business models and create new opportunities, like the internet has."

There is a "super problem" that stems from the formation of such enormous systems: the large number of risks these systems entail. When designing complex systems, the thinkers and planners see the opportunities and chances, but are not always able to assess the risks and try to minimize them early, at the planning and design phase.

Hillary Sillitto demonstrates: "One of the main reasons for the financial crisis of 2008 was the ability to transfer funds across the world very quickly, or, one of the biggest problems of the internet is the terrorists' ability to use it to do harm. So, the more complex systems become, the more the connections between them multiply, the higher the chances that something will go wrong, be it on purpose or due to plain stupidity. Thus, the importance of the need to balance opportunities and risks cannot be stressed enough."

This development creates new areas of knowledge like "system science," which seek to understand the behavior of complex systems. In the past, such systems operated independently, but they grew larger and larger, and connections began to form between them. The interdependency is increasing and can no longer be fully controlled.

Hillary Sillitto: "Complex systems cannot be controlled, but only influenced (on this subject, see interview with Olivier De Weck). For instance, a connection is formed between two supersystems, where there had never been a connection before: energy and transportation. The connection between the scope of the transportation and the energy consumption affects world climate. Our increased consumption of the world's resources creates the need to seriously examine alternative ways to improve our quality of life. For example, mobile phones have had a positive effect on the quality of life in Africa. They also eliminate the need for frequent traveling. The same benefits can be obtained using communication systems. Or, take the question of water: desalination requires a lot of energy, so we must find alternative ways of doing it."

Nevertheless, how do we eliminate, or at least minimize, the risks entailed in the creation of such complex systems?

Hillary Sillitto: "In some cases, it is necessary to determine (even on legislative and regulatory levels) that a system does not exist independently, but has to contribute and integrate itself into the larger system above it. Awareness of the price of a system's existence needs to be raised, not only in financial terms, but in terms of natural resources, personal safety and more.

This means we can no longer remain satisfied with just technological personnel – engineers, mathematicians and physicists – for the planning and design of super-systems. We probably need to include other professionals, like psychologists, for instance, to help bridge the socio-technical gap."

On Systems Engineering

The aforementioned suggests that as we rise through the hierarchy of systems, at a certain point, systems engineers are no longer enough to lead the inter-systemic integration. Of course, we have yet to reach this point, as today, the field is headed by the approach that a systems engineer is, first and foremost, a technological expert, usually himself an engineer, with "systemic skills."

Hillary Sillitto: "the implementation of systems thinking goes beyond systems engineering. Systems engineering focuses on the engineering aspects, while systems thinking also touches the management and social aspects that lead the system towards success."

He presents a model that includes three levels of systems engineering:

The first level: "closed systems" – the system has clear boundaries and its people do not care for what happens outside of it;

The second level: "open systems in an open environment" – an open system, connected to its environment, which can be regarded as infinite in extent;

The third level: "open systems in a closed planet": super systems that include other systems, which, at their widest expression, include the entire world: and the environment these systems exist in is not infinite but has a hard boundary, the boundary of our planet. (For example, fishermen used to treat the sea as an infinite source of fish and an infinite sink for waste. Now they have to recognize that fish stocks are limited and pollution accumulates in the seas).

As aforesaid, the apparent question is: how relevant is systems engineering, when it comes to the third level?

In this context, Hillary Sillitto agrees that the development of complex mega-systems makes systems engineering more relevant than ever, because it includes the fundamental tools that allow us to deal with such immense complexity.

Hillary Sillitto presents another model that distinguishes between two different classifications of systems engineering. This is a model by software systems engineering expert, Prof. Dave Stupples of UCL:

The first level: Systems engineering within a discipline, like optics or software.

The second level: Interdisciplinary systems engineering working across a combination of disciplines, like mechanics, electronics, or software.

The third level: Socio-technical systems engineering – the combination of technology and the human and societal factor.

The first two levels are clearly in the systems engineer's playground. But who should be responsible for the ever growing interface between technology and society?

Hillary Sillitto admits that he does not know. He agrees with the possibility that this increasing need may bring about a new profession: "maybe there will be people who will offer a different approach and start examining pragmatic aspects. Maybe they will be called 'systemists.' This level is taking systems engineering principles beyond engineering."

He distinguishes between process-oriented systems engineers and systems engineers with systemic thinking, which transcends process-oriented thinking, and involves understanding the "whole system" and how it interacts with its environment to create the desired emergent properties.

Most engineers say: 'tell me what the problem is and I will find the solution for you'; and when they find the solution, they are satisfied. They don't care whether they've solved the right problem. This is why we need people who will define the relevance of problems. People who will examine whether the systems are still relevant as the needs of the clients change. Systems engineers must be that kind of people, but there is no uniform pattern to systems engineering.

In some companies, many people who are not systems engineers contribute to the systems work on a project. On the other hand, there are people whose job title is 'systems engineer,' but who do not think systemically at all.

We asked Hillary Sillitto, how, in his opinion, processes and systemic thinking can be combined.

He says that the major factors of the product's purpose and life cycle must be identified and focused on. For this purpose, it is important to understand the design review sequence. Understanding the purpose and life cycle of a product allows us to define what is important in each phase, which decisions to make and in what order.

At the same time, he says: "The right performance indices must be determined. It is important to decide what really makes the difference between success and failure, and knowing how to organize different types of activities in order to support them. In addition, it is important to know how to combine the knowledge of which processes to initiate and the response to the question of why we need to initiate those processes. We need to understand the details but to avoid falling into their traps (on this subject, see interview with Kobi Reiner)."

On systems engineering and intercultural differences:

Intercultural differences, whether they belong to different peoples or organizations, impact people's behavior patterns, including those of systems engineers. Hillary Sillitto finds significant differences in the implementation patterns of systems engineering. He demonstrates, using the early influences that facilitated the creation of the differences between the United Kingdom and the United States as an example: "It is customary to assume that systems engineers are technology oriented, but it is different from one country to another. In the UK, there is more awareness of the soft parts of a system, of the fact that a system is a combination of people, processes, technology and information. The US, on the other hand, is more technologically oriented and pays less attention to the human factor.

One of the reasons for this is the background for the field's development. System engineering in the UK received a push in the 1930s, due to the need for developing complex defense systems, including the radar, to defend the UK during World War II. The US, experienced a similar 'push' a decade later from the Manhattan Project (The United States' project for developing nuclear weapon during World War II – the authors). The Manhattan project was mostly technological, while the air defense battle management system that defended the British civilians, which had laid down the foundations for systems-oriented operational analysis in the UK, was a complex socio-technical system, designed to get the right information to the human decision makers who were the heart of the system."

He gives another example of the differences between countries, this time from a different angle, namely, culturally biased system considerations: "Some countries, like France, the UK, US, Israel, purchase military systems to really make use of them, to provide a real military capability, while other countries mainly want the industrial advantages they can obtain through the purchase. Meaning, the chances of them actually using an aircraft for military purposes are small. There is a much greater chance that they are interested in its technology to enhance their industrial base.

Intercultural differences are expressed, among other places, in the defense industries, the cradle of systems engineering, where the main clients' (mostly governments) general approach has changed.

In the UK, up until 20 years ago, the main risk in ordering a project lay on the government (for further discussion of the switch from 'cost plus' to 'fixed price,' see interview with Ovadia Hararri). During the 90s, the country performed an integration of defense organizations and its relationship with its suppliers underwent a significant reform. The government began transferring the responsibility to the prime contractors, who had to take some of the risks upon themselves."

This interculturality meant that in order to handle the new rules of the game, the suppliers had to formulate new work patterns. They had to deepen the collaborations between them (and bridge inter-cultural gaps, not only between countries, but between business organizations – the authors), because many complex systems involve a number of subcontractors, and if one of them fails to fulfill his responsibilities, the others (and, of course, the client) might be negatively affected as well.

A solution to this problem requires systemic thinking – but it is a business, rather than a technological issue.

According to Hillary Sillitto, a senior executive in a military technology company, such cases require the following: "The companies involved in the project need to formalize an agreement that divides both the risk share and gain share between them. Such a situation motivates the companies to work together to achieve their common goal, so that if anyone has a problem, everyone has to work as a team to solve it and achieve the goal in the most efficient way possible."

He raises another problem that stems from intercultural differences, an inside problem, typical of a multinational company: "Here in Thales, we must figure out how to bridge the cultural gaps between the different countries where we operate. In many cases, the integrator in each country offers a solution for his clients, composed of systems manufactured by us in different countries." (For further details on this subject, see interview with Gilli Fortuna).