3.4

SYSTEMS ENGINEERING AND ACADEMIA

3.4.1 "APPLYING HOLISTIC THINKING"

An interview with Prof. Joseph Kasser

As a field in its infancy that has yet to complete its evolution and is in the process of defining itself, it is only natural for systems engineering to be perceived differently by different people. Even now, some see systems engineering as a profession, at the base of which lies the development and implementation of engineering processes, and thus, only those who have training as engineers can practice it. The other end of the spectrum is occupied by those who do not see systems engineering as a profession at all, but rather as a collection of tools that helps the formulation of thought processes for more efficient problem solving (problems which are not necessarily related to engineering). As such, systems engineering can potentially help anyone, not just engineers.

One of the loudest voices siding with this approach is that of Prof. Joseph Kasser. An electronics engineer by training, he turned to systems engineering later in his career. Today, he is a prominent researcher and lecturer on systematic and holistic thinking.

We have conversed with him about his perception of the field.

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

The General Approach

Prof. Joe Kasser's words suggest that his perception is comprised of three concentric circles: the largest circle is *holistic thinking*. It is a way of thinking that sees the system from various perspectives: not just from within, but from without as well. This approach allows one to see the system from unusual perspectives. This way, things that, at first glance, may appear as difficult problems become the foundations for creative solutions. Holistic thinking can serve anyone who has a problem to solve.

The second circle is *systematic thinking*, which is, in fact, part of holistic thinking. It examines the system, its components, and the relations between them. It also examines the processes at work within the system. Systematic thinking is relevant for all systems, not just for technological, engineering-oriented ones. For instance, an organizational unit is a system too, and managing employees requires systematic thinking.

The third circle is *systems engineering*. Systems engineering is the application of holistic thinking, often, but not always, implemented in systems that are technological at their core. This is why there are many who claim that a systems engineer must have training in engineering.

Background

Systematic thinking has accompanied Prof. Kasser from days as a teenager: "Even back then, I understood that I saw things differently than others. My father, and my teacher, taught me to see things from other people's points of view."

All his life, he acted as a holistic thinker, even as an organizational specialist, whose job description did not include the word "system".

Joe Kasser: "During the 70s, I worked as a systems engineer for Bendix Aerospace – then, a subcontractor for NASA, engaged in building the science package the astronauts took to the moon on the Apollo 15, 16, and 17 missions. I worked in accordance with traditional systems engineering work patterns, and I still managed to find creative solutions for the problems I faced."

During the 80s, Prof. Kasser worked in Israel for Luz Industries, a company that developed and established solar fields. He joined the company as head of electronic control and monitoring and was also the lead systems engineer of the solar field.

He says: "All the odds said Luz was supposed to fail as a startup, because of the combination of distance, communication, and language constraints. But it succeeded anyway. It took me 15 years to figure out why – because we used holistic thinking in our systems engineering. That is what made the difference between success and failure."

Prof. Kasser demonstrates the use of holistic thinking in a company: "We needed to control a field of mirrors using computers. The computing costs were supposed to reach nearly a million dollars, a very large expense. We looked for a solution and, by thinking outside the box, found a way to use concepts from satellite communications. We saw the similarities between the satellite network and the mirror controller network.

This is an example of looking at things from a different perspective. Not only did we lower the computing costs to $2,000 per control system from $300,000, we were also able to simplify the software."

His last job as a systems engineer was at Ford Aerospace. This was when he began his doctorate studies, as part of the company's employee training program. In 1997, Prof. Kasser founded his own consultant firm, graduated, and became an academic researcher and lecturer.

Prof. Kasser does not only lecture and write about looking at systems differently to find creative solutions; it is also evident that he uses this approach in his work as a researcher and lecturer.

Thus, for instance, his dissertation, done in the engineering faculty, examined systems engineering from various angles, not just those that pertain to engineering: "it was a combination of systems engineering and business administration. It was based on the assessment that a group of small businesses, each responsible for different tasks, can, in effect, function like a large company."

He adds and demonstrates: "When I began reading systems engineering textbooks, I discovered that they were full of contradictions. I particularly remember a symposium, where a number of lecturers had presented their perception of the field. Each of them had defined systems engineering differently.

What stood out particularly was that I could not find anything that systems engineering did that was unique to systems engineering. Everything that was defined as part of systems engineering was also done by someone else: financing, testing, design, architecture … all those were performed by other people in other fields. A year later, I published an article entitled 'Systems Engineering – Myth or Reality'."

Today, he has just completed writing a book that combines systematic thinking and holistic thinking, entitled "Holistic Thinking Creating – Innovative Solutions to Complex Problems."

In the second half of this chapter, we will present Prof. Joe Kasser's perceptions on systems engineering in its wider context.

Beyond Systems Engineering

On holistic thinking: "Most people examine something from a single perspective. If they like it, they adopt it. Their minds don't examine any further aspects, they don't look around. I call it 'the perspective parameter' – how your mind looks at the problem from different perspectives. A problem needs to be examined from all sides. If we do that, we will see more solutions than problems.

Holistic thinking allows people to see solutions where others see only problems. Holistic thinking is important for understanding things related to creative solutions.

Holistic thinking deals with more than the internal and external perspectives and how they relate to one another. It examines the way systems work, how they are used, their liability and more.

In order to get a good system and find solutions to problems, it is necessary to go to the roots of the system and compare it to other systems. This way, it is possible to discern common qualities, positive and negative ones, to see the differences and figure out the reasons for them.

It's not enough to look at things from the inside, at the structure and functionality of a system. To find solutions, one needs to look at things from more perspectives.

Holistic thinking for problem solving is suitable for any situation, from control systems development to tracking a person on the moon to properly holding a glass of water – if you put your pinky underneath the glass of water you drink from, it's because you understand that the older a person gets, the less steady his hands become. By placing the pinky underneath the glass, you minimize the chance of the glass falling. This is an example of risk management viewed systematically.

Anyone who wishes to solve a problem needs to think holistically. The way to create the solution is systems engineering, an application of holistic thinking."

On systematic thinking: "Thinking systematically means looking at the different parts of a system and how they relate to one another.

Systematic thinking means two things: the first, thinking about the system and examining the relations between its components. The second, thinking systematically.

The question is: where do we start defining the system? Defining the boundaries of the system is critical. For one person, the system is the car; for another, it is the car and its passengers; for a third person, all the cars on the road are the system. The engine is also a system. This is where systematic thinking is needed."

On Systems Engineering

On the essence of systems engineering: "Systems engineering is a way of thinking for solving problems. It includes a set of tools, solutions and alternatives, processes and people. This problem solving process is the process of systems engineering.

It is important to make the distinction between the role of systems engineering or the job title and the activity of systems engineering.

Systems engineering is, in essence, the application of holistic thinking. It's called systems engineering because we tend to use the wrong words. People come with a concept and don't know there already is a word for it, so they invent a new one. The word engineering means making something happen. Some see systems engineering as processes. Some see it as a way of solving problems. Others see it as a discipline full of procedures.

Systems engineering is both a profession and a discipline.

Systems engineering is a way of thinking, and so, it is unaffected by cultural differences. Cultural differences only affect how things are expressed. In different cultures, systems engineering is expressed in different ways."

On qualities and abilities: "Good systems engineers see connections that other people miss. This is why they also see solutions.

Every engineer has to be a systems engineer some of the time. He or she won't be using this skill all the time, but when he has to solve a problem, he will need it. Once a solution is found, the rest is engineering.

The good people in the field of systems engineering hail from the world of physics or electronics, because they understand the real world. It is an important component in their training. It is important for a systems engineer to possess training as an engineer, to understand exact sciences. He needs to be one who understands that things cannot be changed simply because we want to change them. Engineers use systems engineering on the technological level, in the construction of airplanes, ships or tanks.

Systems engineers are engineers, because that is where it helps them solve problems. As a rule, one does not have to be an engineer in order to be a systems engineer. One has to be an engineer, if one works in an engineering environment. If a systems engineer practices aerospace systems engineering, he has to be familiar with aerospace. But if the systems engineering pertains to marketing, then he needs to be well versed in marketing, and he doesn't have to be an engineer. Systems engineering is the application of holistic thinking, you can be a marketing systems engineer, if you implement systematic thinking.

A systems engineer needs to know the industry he works with, otherwise he will reach the wrong conclusions and his decisions do senseless things. For example, if he knows nothing about TVs and sees smoke rising from a broken TV, he might assume that TVs are normally filled with smoke and that, therefore, has stopped working because the smoke has leaked out, so the smoke needs to be put back in for it to work again.

A project leader does not have to have systems engineering background. What's important is for him to be able to think holistically."

On the effective application of systems engineering: "People don't usually stop to think what a systems engineer needs to do. If we look only at what they do, without knowing what they need to do, we will never know if we've missed something important.

In effect, today most systems engineers follow processes. Only a small minority are problem solvers."

On Systems Engineering and Management

Joe Kasser: "Systems engineering is a problem solving mechanism that includes many managerial elements, because it involves human components and processes. It is a mechanism adopted mostly by engineers, because they found it useful for solving systemic technological problems.

There is an overlap between systems engineering and management, because systems engineering designs the processes that managers later supervise. There are a lot of professionals who use systems engineering tools – detectives, for instance. Also, cooks, who have to cook a meal where all the ingredients are combined in the right order, the right way, and at the right time.

Managers don't normally use systems engineering tools, because they are not taught to do so. Systems engineering methodologies should be taught in administration schools, not just engineering schools.

Systems engineering is the management tool of the 21st century. It is a different management method that includes tools and techniques suited for each case."

3.4.2 "A POWERFUL NATURAL CURIOSITY AND AN ABILITY TO TRULY LIKE PEOPLE"

An interview with Dr. Cecilia Haskins

Despite the fact that systems engineering is a young discipline, a sizeable number of people already hold job titles as "systems engineers" and many higher education facilities now offer programs that grant advanced degrees in this field.

Dr. Cecilia Haskins, an American teaching systems engineering in Norway, is among those who believe that a systems engineer does not have to be an engineer, and the appearance of the word "engineering" in the name might in itself be a mistake. The really important word is "systems" – relating to the ability to see the big picture and act systematically. In this chapter, Cecilia Haskins describes her perception of systems engineering.

Who is a Systems Engineer?

Cecilia Haskins has always been a systems engineer: "I was born a systems engineer. From a very young age, I have been very systematic in the way I approached everything. I had the ability to organize other people's work as well as my own. When my father, who also had a systematic mind, would return from work and talk about his day, I would listen and ask questions."

At the young age of 15, Cecilia already knew she wanted to pursue a career in Electronic Data Processing (EDP), but at the time, the United States did not yet have a university that offered education in this new field: "And then I got some great advice from my father – advice I now pass on to others – to study natural sciences, especially chemistry, the perfect background for systems engineers. The basic principle of chemistry is that atomic elements are very small, and everything else is made from these. This is why it is the perfect mental model for systems engineering. In my 40 years of work, systems engineers who started out as chemists were the ones I enjoyed working with the most."

It is evident from Cecilia's words that the ability to be a successful systems engineer is, first and foremost, inborn. Cecilia specifies further: "A systems engineer needs to have two key traits and both are critical for good performance: the first is *natural curiosity*, namely, never getting tired of asking questions; the second is truly and deeply *liking people* – no matter who they are or what your impression of them is – having the desire to do things with them, and learn from them."

She gives an example from her work as a student advisor: "Today I am involved in a project, where graduate students manufacture a car. It is a long process, currently in its third year. Three years ago they realized that a project of this scale needed to employ systems engineering, and I was charged with teaching them these skills. The first student who volunteered for the project was a born systems engineer. Nobody even came near his level of ability. I hardly had to advise him at all. He instinctively knew what to do to help his team succeed. All I had to do was to refer him to the literature so that he could write his dissertation. He did things naturally, and when, later, he was exposed to the literature he saw how the things he had done were formally defined. The other students who came after him were different. They studied first and then said: 'oh, we should do something about those requirements.' They used the literature, what we had taught them, and they did it very well, but it didn't come naturally to them like it had for the first student."

Another "amusing" (in her words) example of the two key traits required of a systems engineer is taken from the beginning stages of Cecilia Haskins' career: "I started my career with a summer job as a programmer analyst in the early 70s. In my first year, my boss asked me to develop a program that included certain components. As usual, I started to ask him a million questions he was completely unable to answer. And then he said to me: 'if you want answers to all these questions, go ask the client yourself.' Thus, at the age of 17, my curiosity and interpersonal skills had qualified me to sit with a client with no one else present. These two central traits are essential to be a systems engineer."

There are many reasons Cecilia believes that a systems engineer does not have to be an engineer: "We have many systems on our planet that do not require knowledge of advanced calculus. Moreover, stereotypical engineers love their work but have minimal inter-personal skills. Often, people are offered the opportunity to become systems engineers and fail because they lack natural curiosity (beyond the specific task they are charged with) or because working with other people is not high on their priority list and conflicts arise."

The Essence of Systems Engineering

Cecilia Haskins believes systems engineering to be a combination of discipline, worldview, and profession that offers various ways of problem solving: "Systems engineering offers a structured discovery process for solving problems. The goal of systems engineering is to always see the big picture and do what needs to be done systematically, so as not to miss anything important.

Conway's Law suggests that the way we organize our company is the way we intend to solve our problems. This is what makes the company successful in the long term. However, organizations today tend to be hybrids: they have a hierarchy, but at the same time, they empower their employees. Over the years, people have learned that if they do things in certain ways, they get better results. Systems engineering thrives as long as people are convinced that it is important to understand, validate, and implement requirements, and to think about solutions in terms of architecture."

Cecilia Haskins created a simplified five-stage systems engineering model as part of her PhD research. This would not be sufficient for complex or critical systems, but is sufficient for many everyday problems.

The five stages of the model are as follows:

S – Stakeholders: Identifying the people who are impacted by or can impact the problem;

P – Problem Formulation: Making sure you are treating the problem, not a symptom;

A – Analysis: Defining solution options, creating alternatives etc.;

D – Decision making: Deciding which of the alternatives to employ and in what order (what to do first, what to do later);

E – Evaluation: Continuously evaluating what we think we know.

On Systems Engineering and Project Management

Cecilia Haskins believes systems engineering has a symbiotic relationship with project management: "Systems engineering and project management are united from the start. One of the major problems for managers managing a project is that you don't always see the big picture. You spend a lot of time dealing with budgets, schedules, and deadlines. I have observed project managers take nonoptimal decisions just so they can say 'mission accomplished.' Systems engineering means helping projects succeed. To me this word is key – succeed; to make a project successful.

In the past, no separation existed between these two fields. The managerial and technological components were handled together intuitively. But today, we get to a level of specialization so high that everyone is immersed in their own field and people become disjointed. Project management and systems engineering are like yin and yang (complementary opposites – a term taken from ancient Chinese philosophy – the authors) – one cannot succeed without the other."

The Development of Systems Engineering in Norway

Cecilia Haskins, who lives and works in Norway, tells of the development of systems engineering in that country: "In Norway, the term was introduced in the mid-90s, but systems engineer job positions have only begun to emerge in recent years. Slowly, companies are beginning to understand that this tool is vital for projects to succeed, for good products to be manufactured, for successful systems to be created. But these are early days; there is still a long way to go. Managers are also beginning to understand the important symbiosis between project management and systems engineering."

The oil and gas industry is the most prominent in Norway and the first to recognize the importance of systems engineering after the Norwegian defense.

Cecilia Haskins: "The oil and gas industry is already proficient in many engineering processes related to extraction and construction. In time, they realized that the construction industry was not organized well enough, and often inefficient. For products located in extreme environmental conditions there are many challenges, both technological and physical. They have begun to recognize the fact that systems engineering can help find solutions to some of those problems."

For example: "I had a graduate student who was investigating a company performing a project for the oil and gas industry. The company worked in two organizational groups. One group would talk to the customers before the product was delivered, at the design phase; while the other handled the product's useful life period of 25 years.

The surprising discovery was that these two groups rarely interacted or made use of formal channels for sharing information. My student asked them whether they had thought to look at their product in terms of a system life cycle, so that in future projects, each group could learn from the other group's experience. They said 'great idea!' and accepted his suggestion."

The willingness to embrace systems engineering stems, in Haskins' opinion, among other things, from the change of generations of executives in the industry: "Today there is a senior management layer in the process of retirement. These are people who come from the traditional backgrounds, very different from the background of those arriving to replace them. In addition, engineering schools today pay a lot more attention to the interdisciplinary approach, even if they don't call it systems engineering. Students today know that talking to the other engineers working on a project may be the right thing to do. The new generation is more open to systems engineering, it makes more sense to them.

There is a company in Norway that has recruited many of my former students. The great thing about this company is that they did it slowly, gradually. For instance, former students of mine had to work in three different positions before joining the corps of systems engineers. The company understood that students straight from college do not start out as brilliant systems engineers. It was a number of years before they put the words 'systems engineer' in a job title, and that is how it should be."

3.4.3 "EXPANDING THE BOUNDARIES OF THE SYSTEM"

An interview with Prof. Olivier de Weck

After roughly 10 years of education and training, during which he worked as an engineer and a systems engineer in the aviation industry, while completing his bachelor's degree studies in engineering and his master's degree studies in systems engineering, Olivier de Weck became a prominent researcher and lecturer in the systems engineering field.

We spoke with him about the nature of systems, the connection between systems engineering and the academy, and the developments of the systems engineering profession, as it is integrated into various industries.

Personal Background

The environment Olivier de Weck had grown up in provided a fertile ground for his development as a systems engineer. He was born and raised in Fribourg, Switzerland; a city where half the citizens are French-speaking and the other half German-speaking. But his mother, a translator, and his father, an immunologist, spoke English to him at home.

Olivier de Weck: "Even as a child, my environment was multilingual. Nothing was uniform. I had connections with different countries, different occupations, different cultures, and different languages. The environment I had grown up in was fertile ground for learning about life in a culturally and linguistically complex environment."

But it was not just the environment, but his heart that had led him to systems engineering. Attracted to the aviation world from a very young age, he joined the air force after being drafted into the military (Switzerland has a mandatory militia conscription system similar to Israel) and, after a training period, became an airplane technician.

At first, he was responsible for the F-5, a two-engine plane, considered relatively "simple" to maintain.

Olivier de Weck: "My passion for systems engineering emerged when I started troubleshooting. Pilots would return from a flight and complain of abnormal phenomena, like vibrations during the flight or low oil pressure. They would fill out a flight report with a description of the malfunction, which would then be passed on to our crew, and it was up to us to solve the problem. Today, planes are a lot more complex than that, with an automatic error code displayed for each malfunction. Today they tell you: 'code 538' and you know exactly what the problem is. But back then, nearly 30 years ago, I had to dig through the books and try to figure out the source of the problem. We knew the symptom was the problem, but we did not know the reason behind it, so we had to troubleshoot.

I didn't specialize in a specific area. I had to know enough about all the plane's subsystems to point out the source of any problem. It's a lot like solving a crime: you raise a hypothesis and then you put it to the test. It was an intellectual challenge. I had to absorb a lot of information very quickly. I loved it."

As part of his military service, Olivier studied mechanical engineering at the Swiss Federal Institute of Technology (ETH) in Zurich. He was discharged in the late 90s as an officer, in charge of the maintenance of an entire squadron.

Olivier de Weck agrees that that was the point when he began adopting systems engineering work patterns, but it was only toward the end of his engineering studies that he discovered the discipline, after reading a book written about several authors, based on a fundamental systems engineering book written by Ralph Hall in 1962.[1]

Having completed his military service, he moved to the United States, where he spent four years working at the aircraft manufacturer McDonnell-Douglas in St. Louis. His last position was engineering program manager for the Swiss FA-18 aircraft. He received his master's degree in systems engineering from MIT, where he proceeded to more advanced studies and became an associate professor, researcher, and lecturer.

Olivier de Weck sees himself as both a practicing systems engineer and a researcher of the field. According to him, despite practicing systems engineering in effect, he is able to "zoom out" and see the bigger picture of the system engineering

[1] Reference to the German textbook on Systems Engineering: http://www.amazon.de/Systems-Engineering-Grundlagen-Reinhard-Haberfellner/dp/328004068X/ref=sr_1_1?s=books&ie=UTF8&qid=1366637658&sr=1-1&keywords=Haberfellner.

field. He believes those who have never practiced systems engineering cannot always research it.

On the Nature of Systems

The system's boundaries:

One of the central dilemmas for anyone who practices systems engineering, including the systems engineer himself, is that of the system's boundaries. Thus, for instance, one can decide that the boundaries of the system are the client's technological requirements for the project. Or, one can define the boundaries to include the technological system's impact on the environment. Of course, such decisions can radically change the system's characteristics and design.

Olivier de Weck says that: "Traditional systems engineering had always been inward focused. It made sure all the system's components (the components, the processes, the subsystems) worked together to produce the system's end products and satisfy the requirements set forth by the client. Systems engineering never gave much importance to what lay outside the system.

The problem is that many systems become composed of other systems (see interview with Hillary Sillitto for a discussion on this subject). They become very large and more and more complex. We start building systems like we have never designed before; connecting systems that were not designed to work together. We do this because we think to derive some benefit from it."

He demonstrates: "Drivers texting behind the wheel is the most common reason for traffic accidents in the United States (in the past that reason used to be drunk drivers). This has to do with systems engineering, because if you analyze the problem, the traditional transportation system is now joined with communication systems in unexpected ways and means of communication, with human behavior and motivation at the center of it all."

How the boundaries of a system are defined affects the design of the solution to the problem: "If we look at a system as purely technological, we may be able to limit the device's communication abilities when it is in motion. But if we expand the definition to include the entire transportation system, the solution can also be preventive legislation."

All this, de Weck sums: "has to do with systems engineering because when there is a problem with the system, a system-wide solution is needed." (Of course, technology is at the core of the combined transportation system, so the issue still lies in the systems engineer's playground – the authors).

Expanding the boundaries of a system also allows us to see connections we cannot always see from within the system itself. Olivier de Weck: "We design a system with certain boundaries and see no correlation between A and B, but if we expand the boundaries, we can suddenly see a correlation (or synchronization) outside the original system. Seeing this correlation has a profound effect on how the inside of the system is designed. For this, the concept must be modeled in greater detail."

Between short-lived and long-lived systems:

The differences between systems with a limited lifespan (such as development projects) and those designed to operate over long periods of time, with no expiration date (such as toll roads or the Internet) have already been addressed elsewhere in this book.

While "short-lived" systems, according to Olivier de Weck, are artificial frameworks designed to achieve concrete objectives, "long-lived" systems can evolve like living organisms and are more difficult to control.

He says: "these systems are partly planned, partly evolved. Some parts of them are designed as artificial systems by systems engineers, while other parts need to be addressed in a way that simulates biological thinking, with the concept of evolution in mind. Evolution cannot really be controlled, but one can plant good genes in the system."

To the question whether the maintenance of a "long-lived" system (maintenance is usually less complex than development – the authors) requires the use of system engineering tools, Olivier de Weck says that it depends on the complexity of the system. According to him, a simple system, like a road, does not require systems engineering (from here we can conclude that a toll road, which includes collection systems, electronic monitoring systems, rescue crews etc., may indeed require the use of such tools – the authors), but a more complex system, like a refinery, does need them.

He agrees with the statement that in certain cases, a system's maintenance can be seen as an ongoing project that requires constant troubleshooting: "If we want the system to continue operating at a certain level, we need to upgrade as well. And then it becomes a sort of gradual development process."

Systems Engineering and the Academy

When discussing the question whether systems engineering is just a collection of disciplines, a job, or a profession, Olivier says that systems engineering is indeed a profession, albeit a young one (we called it "profession coming into being" – the authors), that has only existed for about 20 years.

He describes the evolution of the profession within the academic framework: "Traditional systems engineering is processes oriented. 'Follow these steps and these instructions and meet the standards.' Before World War II, engineering had been a list of specified actions, of empirical rules of thumb following an empirical recipe. During and after the war, larger, more complex projects, like the Manhattan project, began to emerge, in which most of the leaders were not engineers, but physicists. At the time, a very influential article had been published, which stated that systems engineering could no longer base itself on empirical rules of thumb, and had to rely on the laws of physics. The academy was forced to reexamine itself. Engineers began switching to a more analytical direction, aiming to understand the physical and chemical bases. Suddenly, the difference between what had been taught to chemistry students and chemical engineering students was not so great anymore. Engineering moved toward science, more analytic than synthetic (on this subject, see interview with Prof. Aviv Rosen).

Only in the last 20 years, with the birth of systems engineering as a research field, has the academy begun to understand that synthesis is also necessary in development. When I discovered systems engineering, it was already about more than just problem solving, but implementing systematic thinking in the technological world. The direction it was heading towards was how to pick an amorphous situation, model it at the right level of abstraction and carefully dissect it into its smallest components that could be solved, then reintegrate into a working solution."

On the same subject, he had this to add: "Systems engineering is a new discipline based initially on trial and error, empiricism. There was a study that suggested that if less than 12% of a project's budget is invested in systems engineering, the project is bound to encounter problems, because important things will be missed. In this situation, the project leaders are sure to face unpleasant surprises. In contrast, investing more than 18% of the budget in systems engineering would cause the project to become too cumbersome, making systems engineering a burden and too bureaucratic. These findings are also based on empirical observations, on experience – these are not proven theories yet, they have no scientific basis.

There is a research field called 'systems science.' Many of those who practice it are experimental physicists and mathematicians. They model systems in a very abstract fashion. Often systems are represented as binary networks using graph theory. Those who work on projects in the field don't know what to do with that. They say: 'this is interesting, but it does nothing to help me'."

Olivier de Weck claims that there is a significant gap between the theoretical models of a system and practical systems engineering, which relies on processes and hands-on experience. This gap will take some time to bridge, but progress is being made in this direction.

The Human Aspect of Systems Engineering

Olivier de Weck is of the opinion that the human factor in systems engineering is underestimated. He agrees with Technion Prof. Aviv Rosen's claim that one of the reasons for this is the fact that it cannot easily be described in mathematical terms (an environment engineers find convenient to function in) and is difficult to oversee and control.

He says that one of the most important issues in systems engineering is the decision concerning a system's level of autonomy, namely, the act of differentiating between actions that should be performed by man and those that should be performed by machine.

Olivier de Weck: "For this purpose, it is important to understand when people function at their best. There is a curve that nicely describes the relationship between people's effectiveness and how busy they are. The optimal level is achieved at around 85% cognitive loading. This leaves people with another 15% of their time to handle unexpected things and emergencies. Beyond this level of cognitive loading, people become overloaded and make mistakes out of pressure. In addition, such an environment causes people to become unhappy and worn out.

On the other side of the spectrum, some studies suggest that when automation levels are very high, people have very little to do and get bored. Examples are operators of power stations. They become distracted and fail to pay attention when needed, which decreases their effectiveness. For this reason, when systems engineers design a system, they must place a heavier emphasis on human limitations. We have only recently begun to realize that the right way to design systems may revolve around human ability."

He gives an example of a system that failed, because it did not consider the complexity of human behavior: "To resolve its air pollution problem, Mexico City decided on a new policy that would decrease the traffic in the city by half. They designed a new transportation directive that directed anyone with an odd license plate number to only be able to drive on odd days, while those with even numbers could only drive on even days. The result was unexpected: not only did the air pollution levels not decrease, they rose. The reason was that a large number of people simply purchased a second car with a license plate that allowed them to drive during the remaining days of the week. To minimize the extra expenses, people tended to buy old cars, which tend to pollute more. The legislation did not take into account people's ability to circumvent it and was eventually abolished."

Systems Engineering in Different Industries

It is apparent that systems engineering mostly aims to find solutions for complex systems, those that need organized work processes so that possible malfunctions can be prevented and handled successfully when they do occur. From here, the differences in the use of systems engineering in different industries are derived.

Olivier de Weck finds that systems engineering is especially developed in the aeronautics industry, as well as in the submarine and nuclear reactor industries, because these are industries that require very efficient or very safe engineering.

Olivier de Weck: "In these systems, you cannot rely on what you see and say 'it looks OK to me,' you have to be very accurate. One such industry that failed to adopt this approach properly is the oil and gas industry. A lot of offshore oil drilling takes place in shallows, but major incidents, like the BP Oil Spill (British Petroleum, one of the world's largest energy companies – the authors) in the Gulf of Mexico, happen in deep waters. These drilling projects are complex systems that have to handle extreme conditions, not unlike those of space exploration, namely, working with robots under high pressure, at high temperatures, and at distant locations. In spite of all that, when asked about systems engineering, the people of this industry usually respond by asking what that is. This needs to change.

The first signs of the implementation of systems engineering are beginning to emerge in refineries founded today, but things are still done sloppily, and the dangers are many. When the system operates at low temperatures and pressure levels and there is a leak, the leak is repaired and the problem is resolved. But when the pressure and temperatures are high, the same leak becomes a serious problem. This industry has only now begun to understand that it cannot go on this way." (see discussion about this industry in the interview with Cecilia Haskins).

Olivier de Weck sees a significant difference between the business sector and the public sector, in terms of the sector's willingness to adopt systems engineering work patterns.

He says: "Systems engineering in the public sector, in government or defense projects (which usually are also government projects – the authors), is integrated into the system, an inseparable part of the requirements specification. The business sector, on the other hand, is focused on immediate or short-term benefits, and so, only uses systems engineering methodologies if it has added value, namely, financial profitability.

The problem with systems engineering in the business world is that its short-term benefits are somewhat hidden. It is mainly viewed as an extra business expense. Even if great efforts were invested into systems engineering, the benefits will only emerge after a period of time which could be several months or years. When a complex system lasts many years, people will talk about what an impressive job the systems engineers had done on it, and how they should be thanked and appreciated for it. But after so many years, those systems engineers will not receive the recognition they deserve, because by then they will have retired or moved away. The gap between cause and effect here is very wide. There is a need for more research on quantifying the value of systems engineering."

In this regard, Olivier de Weck agrees that systems engineering is akin to preventive medicine.

More Insights On Systems Engineering

On the qualities of a systems engineer

- Olivier de Weck's words suggest that not anyone can be an effective systems engineer. Alongside his understanding of the technological, systematic aspect, a systems engineer must also have very good interpersonal skills. He must understand the fields he is engaged in and, at the same time, be open to collaboration, because "by their very nature, systems engineers intervene in other people's business. They look under your carpet, to see what you swept under it. This is why systems engineers must be not only knowledgeable, but also possess the right character for the job. Systems engineers who are less knowledgeable and more bureaucratic, those who strictly follow procedures, are not always popular people. Such people are seen as a nuisance and a bureaucratic overhead."
- In de Weck's opinion, a systems engineer has to have the basic education and training of an engineer, because he needs to be trained in planning, designing, and operating technological systems. Oftentimes, he or she has a home discipline like controls, structures and materials, or software engineering, to name a few.Good home disciplines for systems engineers are those that tend to touch all or most of a system. This is true in cases when technology lies at the core of a system – be it an engine or a database.

On the essence of the job

- One of the major questions concerning the definition of the profession is whether a systems engineer is a specialist or a generalist. Olivier de Weck believes that the most common assumption is that a systems engineer is a generalist, but a fundamental problem is apparent: a systems engineer does not start his career as one, but rather becomes a systems engineer after gaining experience as an engineer. “Companies recruit systems engineers as engineers first; then, as part of the organization’s career paths and rotation, they go through a number of positions and eventually become systems engineers, having learned a little bit of everything. In my opinion, that is not the best way. Systems engineering is and should be a specialization in itself, like software engineering or thermodynamics. Systems Engineers are specialists in formulating proper requirements, deriving verification plans, managing complexity and interfaces etc … this is not a general skill, but an abstract set of skills that require specialists.”

On a systems engineer’s professional background

– Another important point already raised in other chapters of this book (see, e.g., the interview with Mimi Timnat) is whether a lead systems engineer in a project needs to be an engineer hailing from the project’s core discipline. For instance, should a project that is, at its core, mechanical be led by a systems engineer with background in mechanical engineering?
  Olivier de Weck: “You can’t be a good systems engineer if you only know a little about each discipline utilized in your project. There are fields, in which one must be more knowledgeable, while in other fields, one can get by with less knowledge. It is important to be well versed in at least some fields; otherwise it is impossible to see nuances that are vital for understanding the system. A lead systems engineer who only knows a little about each field will be perceived as overly naïve.
  If there is a project, the professional core of which is outside the areas the lead systems engineer is well versed in, but does relate to them, then that seems alright to me. This has to come alongside a deep understanding of the fundamental processes of systems engineering, such as requirement definition, interface management, system examination, and system assessment. These are all areas in which the systems engineer must have in-depth knowledge and understanding.”
– For this reason, Olivier de Weck believes a systems engineer cannot easily move from one industry to another. “The knowledge a systems engineer accumulates in a certain industry may not be as relevant in others. Switching between industries like that is what creates the situations where an engineer’s knowledge in most of the fields the new industry’s project concerns itself with is minimal, and he has to start educating himself on a variety of subjects.”
  (On this subject, see discussion with Mimi Timnat)
  “However”, he adds, “systems engineers are quick learners and many approaches and methodologies can be applied to more than one industry. Of course, not all industries adopt them, and when they do, they often do it differently.”

(Among other reasons, this is because there are no uniform rules yet for working with the various engineering fields, and systems engineering standards are only now being formulated – the authors).

3.4.4 "A PROFESSION MEANT TO SERVE THE NEEDS OF THE INDUSTRY"

An interview with Prof. Aviv Rosen

Prof. Aviv Rosen from the Faculty of Aerospace Engineering at the Israel Institute of Technology (the Technion) is also the head of The Gordon Center for Systems Engineering, home of the Technion's systems engineering graduate program. In our interview with him, we discussed his perception of the discipline's evolution, its character, and its importance; the connection between aeronautics and systems engineering; and the training of systems engineers.

The Two Overarching Domains of the Engineering World

Prof. Aviv Rosen finds that, by way of generalization, the world of engineering can be divided into two domains – analysis and synthesis: "Analysis is associated with the world of research, and its products are usually models for understanding various phenomena. Innovations often begin with analysis. Conversely, synthesis is the ability to bring components together and produce an engineering product. This is usually done by the industry. Synthesis is considered to be of a more routine nature, and was therefore perceived as inferior to analysis by academia for many years. Research was thought of as a more lucrative practice, as it offered the possibility of discovering new things and publishing one's findings in scientific magazines.

But times have changed, and the importance of synthesis has slowly increased in many engineering fields. Consequently, the rate of appearance of major technological innovations in these fields has diminished. Take, for example, the jet engine: it conquered the market after the end of World War II, and still remains the most common means of airborne propulsion, sixty years later. Today, there are a lot of similarities between different types of aircraft, and success is no longer determined by the level of technology (as it does not vary much between the various manufacturers), but by how to better integrate the system as a whole, that is – by synthesis.

While, in the past, the market wanted the best technology and paid little mind to the cost, today the orientation is also – and in many cases, mainly – economical. Technologically, the industry has everything it needs. As far as it is concerned, today's added value is in the benefits of higher profitability. Competition is no longer over breakthroughs, but over who has the best systems engineering."

Systems Engineering is More Than Just Engineering

Prof. Rosen's first direct, unmediated encounter with systems engineering happened when, as the Dean of the Technion's Aerospace Engineering Faculty, he was presented with the industry heads' request to found a training program for systems engineers (more details on this next). But, according to him, the turning point in his personal views on the discipline was when he first began to understand the importance of the 'soft sciences': "I come from the world of exact sciences – mathematics, formulas, and computer programs – and I have learned that, in many cases, the so-called 'soft' sciences are no less important than the technology. If the technology is superb, but the economics are poor, the product will not sell. If you have created a technological wonder that does not suit the market, you have failed. At the end of the road, there is always a client, and his psychology needs to be taken into account; products must sell.

Systems engineering knows how to take these things into account and integrate them into the engineering process. The engineering and the needs of the client have to go together, and this combination is what systems engineering is about. For example, one of the discipline's most salient terms is 'requirements management.' In the past, engineers did not give requirements the weight they deserved; they just developed as well as their professional abilities allowed them to. Today, it is clear that system requirements cannot be ignored, and must be a pivotal factor throughout the product's life cycle. This dramatic change in perception was instilled by systems engineering, and the new approach has now become part of the industry's methodology."

Systems Engineering and its Affinity with Aeronautics

Other chapters of our book have addressed the claim that systems engineering has evolved hand in hand with aerospace engineering. Prof. Aviv Rosen expands on this: "An aeronautical engineer's training is very similar to that of a systems engineer. Even during the 60s, back when I was a student in the Technion's aerospace engineering faculty, there was a conflict between the need to specialize in one of the various subdisciplines of aeronautics, and the need to train a 'well balanced engineer,' with background knowledge in all areas. The Technion had, and still has, faculties, where the students choose the course of their professional development within the faculty. For instance, the mechanical engineering faculty offered specializations in energy, production, and other areas. Aeronautics, however, argued that the uniqueness of the field is its multidisciplinary nature, an airplane being an interdisciplinary and multidisciplinary craft. An aeronautical engineer is a better engineer, if he sees the whole picture, rather than just the aerodynamics, structure or control systems.

I, for instance, specialized in structures in my second and third degree studies, and only years later returned to the study of aerodynamics. I would not have taken that path if, during my first degree studies, I had specialized only in structures, without first being acquainted with the other areas. It is this exposure that creates within the student the openness to learning new things.

There are considerably fewer aeronautical engineers than there are mechanical or electrical engineers, and still, people ask me 'how come there are so few of you, and yet we see you everywhere we go?' The answer is that aeronautical engineers receive a multidisciplinary education, and are unafraid to enter unfamiliar territories later in their careers."

As aforesaid, this approach stems from the unique nature of an aircraft.

Aviv Rosen: "Airborne vehicles are technological systems that undergo optimization even as early as at the planning stages. They would not have been able to fly, otherwise. To illustrate this point, we can use an analogy with another discipline, such as construction. A construction engineer builds the structure; after him, the plumber comes and performs his work, and even if the two never coordinate, things will work out – the building will stand. Even if more weight is added or a groove is made, it will not make much of a difference. In comparison, on an aircraft, all the activity is coordinated and integrated, from beginning to end."

Systems Engineering Training in Israel

Aviv Rosen: "During the 90s, Israeli industry badly needed systems engineers, mainly in the technologically and systemically complex defense and aviation industries. The people in those industries felt that systems engineers who acted as such on their own accord were not enough; they needed to be given tools and methodologies to help them bring order to their applications of systems engineering. The recognition of the importance of systems engineering had already set-in abroad, mostly in the United States, and the Israeli industry wished to adopt similar patterns. Companies like IAI and Rafael began training systems engineers on their own, using internal training frameworks.

In the mid-nineties, representatives of these companies approached the Technion and asked it to establish an academic program that would grant its graduates an academic degree in systems engineering. The special affinity between aeronautics and systems engineering (and the fact that the request came from the aviation industries) led the petitioners to the then dean of the aerospace faculty, Prof. Aviv Rosen."

Why were they unsatisfied with their own, internal training programs?

Prof. Aviv Rosen: "It was a trend that started abroad, when more and more large systems failed. They were either technologically or economically unsuccessful, or simply failed to meet their deadlines. People saw that things were not working and found the problem to be *in the connections*. Consumers, such as the US Air Force, were the ones who put their finger on the problem: a lot of the time, the industry 'did not care' – it employed the 'cost plus' strategy.

The industry in Israel was not free of irregularities and mismatches either. One particularly notable phenomenon was the unexpected rise of new needs, which required changes to be made in mid-project. Technologically, the industry's output was impressive, but there was a sense of something not being right.

With the increasing academization of systems engineering overseas, the Israeli industries also wished to adopt the trend. This was not due to a lack of knowledge (they had very serious training programs that included hundreds of hours of study), but because they wanted an academic program that provided its graduates with 'qualifications' in the new field; a respectable program that added a different point of view and contributed to the field's advancement."

According to him, *systems engineering is a profession, meant, first and foremost, to serve the needs of the industry.* A profession born mostly out of practical needs in the field, not an area of study that the academicians wished to explore of their own volition.

What, then, was the Technion's interest in developing such a unique program, and agreeing to the special format of including industry representatives in the structuring and direction of the program, no less? After all, the academic steering committee was based on more than just academicians, and the approval of such bold initiatives is not an everyday occurrence in a respectable academic institution, the natural inclination of which, as these establishments go, is toward conservatism.

Prof. Aviv Rosen explains that parallel institutions abroad have similar attitudes: "The great impact of engineering institutions like MIT resides in the training of engineers. The training happens in research-oriented academic institutions, because an institution that performs research can train engineers better than one that occupies itself only with training."

And, in the context of Israel and the Technion, he adds: "The founding fathers of the Technion had established that one of its goals was to contribute to the needs of the State of Israel; training a systems engineer, who would benefit Israeli industry, is one way of realizing this vision."

The Technion's program is, as aforesaid, meant to train people from the industry, who have some experience in engineering work. Applicants are required to have at least three years of hands-on experience. The framework of the studies is also unusual, as it takes into account the limitations of students who are older than average and usually have families and work full-time. The students have one, concentrated weekly day of studies. Spread across two and a half years, the program's curriculum includes approximately 500 hours of classes (and many more hours of work outside the lecture rooms), and it places a heavy emphasis on practical assignments, done in teams. This is, of course, to foster one of the most basic elements of systems engineering – teamwork. It is also why the final project of the systems engineering training program, which, in a regular academic framework, would tend to be personal, is a group project (of course, each group member is graded separately).

The first class of the Technion's training program graduated in 2001. As of 2012, the program has trained approximately 900 systems engineers.

On the students:

At first, students who entered the program came mostly from the companies that had taken part in the initiative of founding it: Rafael, IAI, and the IDF's technological forces.

In the years that followed, the circle of organizations and industries that provided the program with students slowly expanded.

Prof. Aviv Rosen: "Today, those who come to study here are mostly mechanical engineers, aeronautical engineers, and electronics engineers, but some *computer science graduates* can be found among them as well. The education of the latter is, in most cases, focused almost exclusively on the world of computing, and they are not always interested in expanding their knowledge, because the market has a demand for them without it as well. But a software engineer who can 'speak other languages' has an advantage over software engineers who focus only on software. Software and systems engineering share many similarities, because computers are the glue that holds the various systems' components together (see also interview with Sharon Shoshany Tavory).

We also have some *physicists* among our students. Physicists make good systems engineers, because they have very good simplification skills, which they acquire thanks to the nature of physics as a discipline. The ability to simplify often allows a better understanding of complex systems."

Further Insights on Systems Engineering

Systems Engineering and Control

– Many systems engineers start out in the control field. This is due to the nature of the profession. Control is, by definition, integration, and so, it follows systems engineering principles. In aeronautics, control is often more systemic than in other fields.

Systems Engineering and Professionalization

– Systems engineering tends toward a broader perspective, rather than toward specialization. Engineering needs both specialized experts and people who can and want to see things from a wider angle. The "want" part is important, because not everyone is willing to give up specialization. Many people are very comfortable as experts; they enjoy being perceived as such and having people come to seek their advice. There are, however, others, who get "bored" after a few years of specialization and begin searching for something new. Engineers receive a lot of systemic stimuli. If they focus on specialization, they will remain specialized. But, if they have an open mind, a broad perspective, and instinctive leadership ability, they will naturally tend toward systems engineering.

On the Integration of Systems Engineering Culture in Organizations

– Changing an organizational culture is a difficult task, even if the CEO is convinced that systems engineering is very important. I asked one such CEO: 'If you have examples from past projects that prove how very beneficial systems engineering is, why don't you decree that no project in your organization goes without systems engineering?' The answer I received was: 'It doesn't work like that. If our man is not convinced, every time there is a glitch, he will tell me (the CEO): 'This is because you dropped all this systems engineering on me. It disrupts things, instead of helping.' People need to be convinced that the change is worthwhile. This is one of the advantages of our academic program: there are 900 graduates out there, in the field, right now, talking about it. And when they see an organization without systems engineering, they ask: 'Why no systems engineering?' This has an impact, but it will take time.

On the Qualities of a Systems Engineer

- If one is not personally inclined toward systems thinking, he will not be a good systems engineer. This trait is more essential than even training or experience.
- The process of developing a system (the V model, see page … end of part one) often begins with the creative 'disorganized' systems engineer. As it progresses, it requires a mind less 'wild' and more administrative. If, during the execution stage, as the project ascends the V model graph, creativity levels climb too high, the project will become problematic. The ideal systems engineer changes with each stage of a project's advancement. The ability to put the right systems engineer in the right place makes the difference between failure and success. (See chapters on The Lavi and The Iron Dome projects).
- I am not a systems engineer, because, as I see it, a systems engineer is one who actively practices the discipline in the industry, while my career is in academia.

A Systems Engineer's Work Method

- A good systems engineer needs to know how to ask questions. He needs to tell the specialist underneath him: "I am not the expert, but your explanation is not convincing. Come up with a better founded explanation, or I will keep asking questions."