3.2

DEVELOPMENTS IN INDUSTRY AND COMMERCE AND IN COMPLEX CIVILIAN SYSTEMS

3.2.1 "THE ABILITY TO IDENTIFY BOTTLENECKS AND ELIMINATE THEM"

An interview with Dr. Gilead (Gillie) Fortuna

If we try to define systems engineering in one, cohesive sentence, it would be the ability to apply systems thinking and multidisciplinary perspective to the management of technological systems. This suggests that systems thinking is an important element of systems engineering and one of a systems engineer's main skills.

We interviewed a high-ranking manager in Israeli industry, who had begun his career as a chemical engineer and later advanced to management positions in several different industries. This pivotal trait is the focus of this chapter; a trait important not only to systems engineers, but also to anyone interested in management.

This chapter comprised of two parts. Part one focuses on Gilead Fortuna's time as researcher and senior manager in a defense company; the defense industry being one of the first to develop systems engineering and one known for good systems thinking. Part two tells of his time in the chemicals, pharmaceuticals, and dietary supplements industries, where systems engineering is given less weight.

Managing and Engineering Complex Technological Systems, First Edition.
Avigdor Zonnenshain and Shuki Stauber.
© 2015 John Wiley & Sons, Inc. Published 2015 by John Wiley & Sons, Inc.

Part one: The Experience in the Defense Industry

Gillie Fortuna studied chemical engineering in the 60s and began working for Rafael as a student in the IDF (Israeli Defense Forces) academic reserve program. As a chemical engineer, he developed propellants – the substances that generate the thrust within the missile engine system, causing it to accelerate. Being naturally curious and an inherent systems thinker, he began venturing outside the fields of chemistry and chemical engineering very early in his professional career. He did not have to go far, for Rafael's focus is not on chemistry. It did not take long for him to begin looking into missile engines – the systems where the propellants he had a part in developing were being put to use. He soon moved on to developing and managing all the systems of the missile, as well as other, innovative systems.

But why should a chemical engineer delve into the workings of a missile engine? Why did the missile developers not hand him a specification of requirements for the chemical composition of the propellants, so he could simply develop them and supply them to the developers?

Gillie Fortuna: "Nobody hands you a ready-made specification. You create it together, piece by piece. When the missile developers construct a system, they have to constantly make trade-off; we made those tradeoffs together. The mission profile requires the engine and thrusters to suit the mission parameters, some of those parameters being performance, mechanical stresses, and reliability. For example, if the engine allows the missile to maneuver close to the target, the warhead can be small, and if the engine is weaker, a larger warhead is needed.

I quickly discovered that what I was really interested in was communication with the missile developers. I did not necessarily focus on the materials (by the way, understanding the composition of the materials that serve the missile is, for all intents and purposes, also an expression of systems thinking), but on having to meet all the mission requirements."

Gillie Fortuna's testimony implies that even at that early stage in his professional progression, he had essentially stepped into the role of a systems engineer. He himself does not see it that way, though. At the beginning of the interview, he claimed to have never been a systems engineer. It is true that he was never formally defined as a systems engineer, but in practice, he behaved as one, at least some of the time. It was not long before he was promoted to management positions in Rafael. His last position there, before he left the company in the early 90s, had been VP of Marketing and Business Development. By the end of the interview, he became convinced that he had indeed been a systems engineer, mostly in his position as the head of Rafael's aeromechanics division and VP of Operations; a position where, among other things, he was placed in charge of the follow-up and coordination of most of Rafael's major projects, including such areas as resource allocation and working with the buyers.

Supportive Organizational Culture

According to Gillie Fortuna, formal, academic studies provide an infrastructure of knowledge, which is important in and of itself; but the real professionalism comes from gaining experience: "I stopped being just a chemical engineer after my first year of work, I became a technologist fairly quickly. I took part in the design of the missile engine, which meant that I had to understand mechanics and computing. I was interested in how others worked, and not confined to my specific area. Gaining experience is what made me into an aeronautics expert. In the long run, it does not matter what you studied at university anymore – what matters is what you did. Rafael's organizational culture was conducive to such phenomena, and so they occurred fairly quickly."

He explains: "Rafael placed a lot of weight on people's abilities and interests. If people were willing and able, they were allowed to move forward. Some professionals want to evolve in their own professional area, while others want to look ahead (on this subject, see also the chapter on Indigo). My nature does not allow me to narrow my point of view and not see things from a broader perspective. Rafael encouraged systemic perspectives even back in the 60s. Such organizational culture has a major effect on the development of this ability in people."

Gillie Fortuna suggests that the development of his systems thinking, one of his fundamental character traits, was enhanced by an organizational culture that encouraged the use of systemic approaches.

It should be noted that he does not claim that people who wish to continue specializing exclusively in their own professional area necessarily lack systemic perspective; only that their systemic perspective finds its expression within the framework of their chosen occupation. He gives an example about a rocket propulsion expert (by the name of M.), to illustrate this point: "M. chose to remain a propulsion expert, and yet he speaks 'systemic.' He holds a systemic dialogue with those around him: he is a consultant who knows how to speak the language of the people he consults, and is highly motivated to deepen his expertise in propulsion and contribute to complex systems by using his ability to provide professional solutions that rely on his in-depth knowledge of his field. What drove me was the desire to make a difference, and constantly do entirely new things. To an extent, it was the desire to constantly shift my point of view and see different sides of the system, while improving my ability to affect it as a whole. Fortunately, I also possessed the management skills I needed to accomplish this."

The Distinction Between Systems Thinking and Systems Management – The Question of Dependency

Gillie Fortuna distinguishes between systems thinking and systems management: "Systems thinking has been with me all my life, from a very young age. Systemic management, however, is a different matter: I only reached it in 1981, when I headed the aeromechanics division, which employed approximately 1,500 people. When we were given the task of developing Israel's first satellite, I was put in charge of an entire, complex system for the first time. It is true that in my previous position, as head of department, I also managed a system, but not one so complex and innovative.

Naturally, I appointed an excellent project manager, who later became an excellent expert in space related fields, and remained one for many years."

He explains: "The subsystems within a system are interdependent. A subsystem in a product, such as a missile, is fairly well-defined, and the interdependency between its components is under your control. As head of department, almost 90% of all resources were in my hands. The system was, therefore, not a complex one."

If follows that in a very large, complex system, the dependency on external factors is far greater. Consequently, the manager's level of control is lessened, and he has to face more constraints, being part of a larger system.

Gillie Fortuna presents two examples of the systemic dependency issue, from two different periods of his time in Rafael:

The first example refers to his time as head of research department, when he handled the field of rocket materials (in the 70s): "We developed a system of deception rockets for the Israeli Navy. The navy had successfully tested the system, and was satisfied with the results. The development of the system and combat doctrine ended in 1971. At the time, Rafael had been only a developer, not a producer; we outsourced the production to Soltam. Two years later, the 1973 Arab–Israeli War broke out, the deception rocket system proved its effectiveness, and the navy decided it needed more such systems. They approached Soltam, and discovered that they had many unusable systems lying around in storage. It turned out Rafael had set a very tight tolerance range (the maximum permitted deviation from the desired dimensions – the authors). Soltam was having trouble meeting the high standard, and failed to keep up the scheduled production pace. The company never bothered to ask Rafael to increase the tolerances. The disqualified systems just lay about in their warehouses, waiting for solutions.

We were called in to address the problem, and because, in times of war, no one has a mind to 'play around with tolerances,' within days, we had turned hundreds of systems usable and made a significant contribution to wartime production.

The systemic lesson here is that the developer cannot provide efficient production specifications, if it has never encountered production processes. Therefore, bringing production and development closer together provides vital advantages.

This story also helped lead Rafael to decide to start producing its own products. It changed the perception that the distinction between organizations in the defense industry should not be between developers and producers. The new approach stated that the differences between organizations are in the types of systems they specialize in, regardless of whether they develop or produce them, or both."

The second example is taken from his time as the head of the aeromechanics division (during the 80s): "As part of the development of a powerful laser, able to hit flying targets, Rafael was tasked with the development of the laser beam, while another industry was put in charge of developing the beam's control array. The control system was very important because for the beam to be effective, it had to be focused on the target long enough to generate the heat that would damage it.

Rafael successfully produced the beam, but the project was discontinued by us, because the other industry was unable to make the control system reliable enough.

This is an example of a system, where you do your part well, but your systemic perspective leads you to decide to invest no more of your time in the project, because it cannot be completed. In retrospect, it is clear that the project had been ahead of its time."

Identifying bottlenecks:

According to Gillie Fortuna, one of the most important skills for those who see the world systemically is the ability to identify and address bottlenecks: "It is an art, the importance of which cannot be overstated, because one cannot keep investing in system upgrades indefinitely – it is too expensive. So, instead of upgrading the system, you upgrade the bottlenecks. The ability to identify them and find ways to open them up is 'the name of the game' in today's global business competition."

Part two: The Experience in the Pharmaceuticals and Chemicals Industries

After leaving Rafael, Gillie Fortuna filled senior management positions in two companies in the chemical industry – his original field of specialization. From 1995 to 2001, he served as the CEO of TAMI, the R&D institute of a large group of companies. Later, from 2001 to 2009, he served as VP of Business Development and Operations in Teva.

He relates a systemic problem he had come across in TAMI: "TAMI developed some very special things, but very few of them were actually implemented in the companies of the group. This was because the development engineers in TAMI were not well-liked by the other companies, who perceived them as pretentious. TAMI's personnel, on their part, disdained the executives of companies who were unwisely refusing to put the wonderful products they had developed to use. Moreover, the transition from development to production was too vague. Yet another problem was that TAMI was managed by engineers and chemists who focused on the success of their development projects – their measurement of success was in their laboratories, or at most, in the semi-industrial facility, and the client company's ability to implement the development was not accounted for in their judgment of their own achievements.

When I had arrived at the company, we changed that attitude. Thus, for instance, when we offered the CEO of Rotem Deshanim (one of the companies in the group) to use a certain development of ours, he said he had a different need and asked that we propose a solution that would allow him to turn a contaminated acid they had produced large amounts of into a food quality acid, regardless of the fact that we had already developed a process for producing food quality acid directly from raw phosphate rocks. To resolve this, Rotem and TAMI engineers sat down and developed a solution together, collaborating from the very beginning of the research and development process. Changing the measurement of success from the success of a development project to its successful implementation by the client company was, to me, an application of systemic perspective."

Systems engineering in the chemicals industry:

Many of the systems engineers we have encountered in this book were integrated into development projects. TAMI, however, had no systems engineers, despite being a research institute. Gillie Fortuna explains: "Systems engineering must not have been needed. Systems engineering is important in a system with numerous components that require tradeoffs to be made. In the chemicals industry, most of the systemic view stems from the need for optimization between several products, some of which are beneficial, while others are attached as part of the process. There are not as many alternatives as there are in aeronautical systems. It is possible to examine alternatives, considering the purity of the material and the cost of the product, but it does not compare to the complexity and high level of the alternative examination process required to launch complex airborne systems into the air. In the chemicals industry, the final test is the application of the development to competitive, economic production, which integrates innovation into the chemistry, the process, the product, and the way the product fits in with the needs of the clients. In the end, this, too, necessitates a systemic view of all the development and economic production capabilities, but at a different and at a lower level of complexity."

Systems engineering in the pharmaceuticals industry.

Gillie Fortuna: "In the field of pharmaceuticals, the development of the active ingredient that facilitates the correct treatment of the disorder is in itself a highly complex process, because it requires the developers to understand the workings of the human body, identify the cause of the disorder, and then derive the treatment options that use the active chemical ingredient. Once the concept's medical potential and feasibility are proven, the system that delivers the drug into the body needs to be developed as well: the active ingredient needs to get to the right place, at the optimal dosage. The next step is guaranteeing minimum side effects and risks to other systems in the patient's body.

I had the unique honor of taking part in a beautiful joining of Rafael's systems and the world of pharmaceuticals, thanks to a systems engineer at Rafael. In 1991, as Chairman of The Civilian Business Development Committee, which we had founded, I approved the first 100 thousand dollar investment in the attempt to realize his idea and develop feasibility for a guided pill, used for diagnosing disorders in the small intestine, a project which later became a highly successful commercial system, (a company by the name of Given Imaging). In this industry, the development period from the concept stage to its full realization is even longer than that of airborne systems, and some of the processes along the way necessitate the application of a systemic approach, in order to reach successful business realization."

Systems engineering in the dietary supplements industry.

Gillie Fortuna: "After I had left Teva, I worked part time as Senior VP in two dietary supplement companies (Gadot Biochemical Industries and Frutarom), and today I am a board member in a very smart dietary supplement company (Enzymotec) that has also expanded into the pharmaceuticals market. In these industries, tradeoffs are constantly made between the effects of the supplement's ingredients, the studies that try to understand the contribution of the active ingredients to the consumers' health, and the attempts to hold clinical trials at various levels of complexity. In actuality, it seems that the more importance is given to the required approval and the potential contribution, the more complex the process of proving it becomes, though it never reaches the complexity of pharmaceutical development."

The Question of Dependency and Risk Management

The previously discussed question of dependency, in the context of controlling resources in complex projects, rises to a new strategic level when brought up in the context of Teva Pharmaceutical Industries – a global company that operates in different countries, both as producer and as marketer of its own products.

Gillie Fortuna: "In Teva, the operations in each country are treated as a separate business unit. Financial and specialization related considerations have led Teva's main management to the conclusion that it is best to concentrate the global production of each product in one place. This means that, for instance, the US marketer has to sell a lot of products produced outside the US, by another business unit. This structure creates high levels of interdependency. The US marketer needs to supply products at very tight schedules and at short notices, but he depends on the goods being shipped from Israel, Hungary or India. This means that the managers of Teva make commitments without having complete control over whether or not they can be fulfilled (unlike the division head at Rafael, who commits, knowing that he has complete control of the production resources).

Complex systems, like that of Teva, are characterized by a lot of co-dependency between subsystems. Factors beyond the control of the sellers, for instance, depend on priority orders, when production resources are limited. Specifically, the representation in China, under my leadership, sold a drug to the Chinese, local market, but the factory that produced in Hungary prioritized the US market, because it was four times as profitable, rendering us unable to realize the potential we had created by marketing our product in China.

The ability to manage client commitments without controlling the resources entails a lot of risk management that only a fine-tuned cooperation between all the involved factors can achieve."

He gives another example from his time in Enzymotec that demonstrates the part risk management plays in a company's systemic perspective: "Enzymotec's business strategy shows advanced systems thinking. It focuses on lipids and develops three families of products: mother's milk, special dietary supplements and medical food. Consequently, difficulties in one market are not enough to cause significant damage to the company. It is rare to find a company that develops three lines of products, all based on the same technology – that is smart risk management."

Further Insights on Systems Engineering

On Strategic Planning

– Other chapters in this book have already addressed the planning component of systems engineering and systems thinking. In the case of Gillie Fortuna, it is a central ingredient in his approach to management: "Wherever I went, I used strategic planning, which really is just an organized thought process. It is defining the strengths and weaknesses, threats and opportunities, the vision – where you want to go – choosing paths of action and managing the risks they entail. Strategic planning is a lot like systems engineering, you use it to see the whole of the system."

On Project Managers as Systems Engineers

– "A project manager in an organization like Rafael has to have an engineering background. A project lead who is not an administrator hires one to help him manage the budget, but the systemic decisions are ones he must understand all the way through. He has to be a technological expert; even heading a division requires in-depth technological skills. The project manager is therefore the true chief systems engineer. The person defined as the project's chief systems engineer is a kind of deputy of the project manager, who also dedicates some of his time to administrative work."
– Gillie Fortuna further adds: "In my experience, I agree with the (authors') statement that, effectively, there are two chief systems engineers in a project, and one of them, as the deputy, dedicates part of his time to administration."

On the Duality of the Systemic Perspective

– A good systems engineer has to see both ways. He must understand the subsystems and their limitations; this is a critical component of systems engineering and of a systemic view. This is why it is not enough to see things from the top down, you have to be able to see bottom up as well. As the CEO of US company, Dow Chemical once told me: "A manager must maneuver between the need to focus and make operative decisions on the single system level (*zoom in*), and the ability to adopt a broad perspective of the consequences of the realization of the vision, and the path that leads to it (*zoom out*)."

3.2.2 "WELL-ORGANIZED WORK IS ALWAYS NEEDED; THE PROBLEM IS PEOPLE DON'T ALWAYS WANT TO MAKE THE EFFORT"

An interview with Boaz Dovrin

It is often said that in its early days, systems engineering evolved at an accelerated pace in the defense, aviation, and space industries. In these large and complex systems, the need to think systemically and facilitate an educated integration of technological systems was especially urgent; more so, because in addition to the technological factors, the systems had to take into account economic considerations and the human needs of large teams. In time, the recognition of the necessity of systems engineering found its way into other technologically oriented industries as well.

This chapter describes the professional development of a senior systems engineer, who had started out in the defense industry, and then transitioned into an industry, where systems engineering was less advanced – the medical equipment industry. It also addresses Boaz Dovrin's views and standpoints on systems engineering as a whole and about the systems engineer profession in particular.

Professional Development as a Systems Engineer

Defense, Military, and Civilian Background. After graduating from his studies of aeronautics at the Technion, in 1989, as part of the IDF's academic reserve program, Boaz Dovrin was enlisted into the IDF and served in the Intelligence Corps' SUAV unit. As a technical officer, he began to act as a systems engineer, unawares. Only later, in 1996, when he was discharged and joined Elbit, did he knowingly become a systems engineer, for this time, the term was in his job title.

Boaz Dovrin: "A SUAV is a system that includes not only the hull of the craft, the systems installed on it and their replacement parts; but also the training of its operators and the management of the committee of inquiry that investigates the reason a SUAV crashed. I was never trained as a systems engineer. Some systems engineers are born this way, systemic by nature. I am one such person. I look at things from more than just one angle. Still, training and experience have helped me hone my skills."

Elbit wanted to recruit a systems engineer to work in the communications field. The wanted ad listed aeronautical engineering training and a military background among the desired skills. Dovrin applied for the job and got it. From the outset, in his very first position, he had to face issues of a markedly systemic nature that concerned the human–technology interface: "A system was developed for transmitting telegrams onto airplanes. This was a collaborative project between three different companies and the Israeli Air Force. The system was not working as it should, and all three companies – Elbit, Elta and Tadiran – were casting the blame at one another. A few months after I had joined the development teams, the system began to work.

This had a number of causes, the first of which was the work method. It is important to work methodically, and not lock on to the easiest solution right away. If a problem is searched for in an orderly fashion, chances are it will be found. Even if something does not work out, it is important to keep going and check everything all the way through. Never compromise and say 'oh well, these are the circumstances … '

The second cause, I think, is the fact that I had arrived from the outside, and had not been involved in the development up to that point. When people are in a predicament, they become irrational, their emotions run high, and they tend to blame those around them for past failures. Joint meetings gathered to deal with the problem can easily blow up, and unexpectedly stray from the technical level. Another helpful fact was my balanced, impartial perspective. I did not immediately assume Elbit was in the right. Rather, I arrived with the desire to solve the problem."

During his 11 years in Elbit, Dovrin had evolved into a senior systems engineer. In 2007, he left the company and joined the Israeli subsidiary of 'Philips Medical Systems' – a company engaged in the field of medical equipment. Philips' Israeli branch develops and produces CT scanners.[1]

[1]The CT scanner is an advanced imaging device, where the X-ray tube and detectors are spun around, allowing the body of the subject to be scanned from all sides, generating a 360° image.

Systems Engineering at Philips Medical Systems

Systems Engineering and Organizational Culture. Based in Cleveland, United States, Philips Medical Systems employs approximately 500 people, over 100 of which are stationed at the Haifa development center. Even as he interviewed for the position at Philips, the major differences between the industry he had left and the industry he was about to join, when concerned with systems engineering, soon became painfully obvious: “I understood from the questions they had asked me (in the job interview) that they did not know what systems engineering was. They were basic questions, completely out of place for someone who had arrived from Elbit or Rafael. Questions like ‘what’s a design review?’ or ‘where does one submit requirements documents, and what is to be done with them?’

Up until 15 years ago, some of the world’s industries were not even aware systems engineering existed. The head of the electronics division at Philips at the time was himself a systems engineer who also arrived from Elbit, and he understood that Philips needed systems engineering. Nevertheless, it took him several years to convince everyone that the discipline was necessary for the company. There was no systems engineering in the organization, and it did not miss it. For instance, the need for systems engineers never came up in the executive training courses.

The gaps between Phillips and Elbit were so large that I could not understand how their projects worked; how they were able to manage multiple projects without synchronizing their resources. I told my boss in Cleveland that I thought we should work differently. We should prepare an organized work scheme that included all the activities we needed to perform, and the length of time each one would require. I offered to create a plan in an Excel spreadsheet, with data, like the number of people needed, the estimated time it would take to perform each task and so on. He was excited and said ‘Do this for all our people in Haifa.’

It was difficult, like tilting at windmills, because most people said ‘nobody is forcing you to act this way, so why invest all this time and effort? They argued that in many cases, the investment was wasted, ‘because after months of work, my project is suddenly changed,’ or ‘the contents of the project will change ten times and there is no point in bothering with Excel tables,’ or ‘I’m about to move to another position, and when I do – all the preliminary work I have done will become irrelevant.’

I disagreed. To me, it would always be relevant: ‘it will serve the one who replaces you. It may be difficult now, when you do it for the first time, but later on, it will make things easier for you.’ In the end, everyone needs order; the problem is that nobody wants to make the effort to achieve it.”

But, if this seemed like the right way to the management, why were the employees not simply, instructed to adopt “systemic” work patterns? “It was not ingrained in the company’s organizational culture, and so it was not enough for a certain manager to believe in systems engineering. It does not work like that. There was an engineering manager in the company, who saw systems engineering as ‘his baby.’ Every conversation with him began with how important systems engineering was. However, not everybody agreed with him.

We invested a lot of energy, as systems engineers and as managers, into selling the idea that systems engineering was needed. And still, there were those, who had been with the company for 30 years, and they told us: 'when I am going to plan my next DMS (the detector array of the CT scanner, see below – the authors), I won't need a systems engineer, because I have an engineer who has been working with me for the past 20 years, and he will write the requirements. I don't need some outsider, who came here only 4 years ago. What does he know?' Changing an organization takes time. It doesn't happen in a day.

Another difficulty in convincing people to accept the change stems from the fact that, perhaps, at the level of the organizational unit, the use of systems engineering really is not always feasible. But there is no doubt it benefits the organization as a whole, because, when you work in chaos, you never know what your people are doing. This is especially true for an organization spread across several places around the world. Sometimes, some of those places are seemingly working on the same things, but effectively, each one is developing something else entirely, without you knowing about it – several teams work on the same task and there is no synchronization whatsoever between them.

Philips used a system, where if a major problem came up, they would recruit a large team to resolve it. But this method created other problems that often went unnoticed at the time, because the consequences of bringing all those people together to solve the current problem were not thought through. Nobody could say what the meaning of such a move would be.

They lacked the maturity that comes with decades of understanding; the ability to realize that systems engineering is the way to go. Even if you try to instill systemic work methods in the organization, every problem, even a minor one, can be disruptive. For instance, you bring in a systems engineer who is not good enough, and when people see his mediocre performance, they come to you and say 'there you go – systems engineering is not worth much'."

Nevertheless, Philips Medical Systems is a successful company, and its product is among the leaders of the global market. So, perhaps it is possible to succeed without adopting systems engineering work patterns.

Boaz Dovrin: "Philips Haifa (originally Elscint) was the company that originally developed the Slice CT Scanner (an imaging technology based on scanning 'slices' of tissue in the subject's body – the authors). Today, it is ranked fourth, and might soon descend to the fifth place. It still employs very talented people, and even if you were to tie up each one's arm and foot, they would still be talented and do great things.

That might be part of the problem. Extraordinarily talented people often resist methodologies. They feel hindered by them; they need to be creative, and systems engineering is, by nature, methodical. So perhaps those geniuses, the physicists of Philips, should not have to work with the methodology.

Philips will continue to be successful, because it has good people, but I fear its market share will continue to decrease in size, because the lack of order creates a surplus of manpower, and that hurts the company's competitiveness."

The management of Philips recognized this risk and brought in new managers, who are now trying to dramatically change the company's conduct. Some of those planned changes are not suitable for a company like Philips and are actually hurting it. On the other hand, some changes match the methodologies that existed in Elbit and include such practices as preparing detailed Excel spreadsheets of all the resources needed for each project; or viewing the company's manpower holistically, thus allowing the managers to made educated decisions regarding employee mobilization and planning ahead; or preparing high-level design discussions and not moving on to the realization phase until the detailed design is complete. Additionally, the betterment of the systems engineering team and the recruitment of engineers with methodological experience have allowed the company to finally begin closing the gaps. Eventually, the time and effort paid off, and Philips Medical Systems began to be convinced in the necessity of systems engineering.

Boaz Dovrin elaborates: "When I had first come to the company, there were 4–5 systems engineers in Haifa, and another 3 in Cleveland. Today, the company employs about 30 systems engineers. The field is definitely growing. Changes in organizational culture are also evident. In the past, systems engineering was perceived as something bad. If someone was offered to become a systems engineer, he would recoil. Today, people, even physicists (a physicist is considered to be the most desirable development position in Philips), are lining up to become systems engineers. They are beginning to understand the importance of it; it just takes time. It isn't easy to persuade an entire organization to switch to different work methods."

A Problem and Its Solution. When Dovrin first came to Philips, he worked on a development project in the DMS array (Digital Measurement System) – the CT scanner's detector system. The system was being developed by teams in Israel and in the United States. The DMS is a complex system (it contains 365 thousand detectors) that requires a very high level of accuracy to operate. Even the smallest signal disruptions can distort the resulting image. For this reason, the system has to be virtually noise-free.

Dovrin tells us of a problem that had come up in this context and of the way it was eventually resolved: "It is very important for the detectors in the scanner to operate at a constant temperature, otherwise the readings are distorted. In the new scanner, the temperature within the DMS would fluctuate; its regulation mechanism did not work. In previous scanners, this problem had been resolved by placing a small heater inside the DMS, with fans next to it, to disperse the air. This way, a constant temperature was maintained. In the new scanner, a new solution was implemented, which, as aforesaid, did not work.

After repeated testing, we found an air leak in the ventilation duct. In the new scanner, the ventilation method had been changed. The small fans that served different parts of the system (the power supply, the X-ray tube, and the detector array) were replaced with a single, more powerful fan, which was meant to serve all the installed systems at once. This was an example of the phenomenon referred to as 'let's reinvent the wheel' – a result of wanting innovation. Add the fact that the development was worked on by teams based in different countries (Israel and the United States) with a major time difference and competition between them, and coordination problems are a reasonable outcome. The questions that need to be asked at the beginning are not always asked.

Historically, there were two companies, each developing its own CT system: one in Haifa and another in Cleveland. After the two were merged, residual friction prevented full cooperation, creating competition and more than a few disruptions in the work process. Each team was developing a different part, thinking that the parts would interface perfectly; all because there was nobody who could see the whole of the system.

The amount of air that was to reach to DMS had been defined during the system planning phase, but the actual amount did not meet that requirement. The Cleveland-based team responsible for air flow had been presented with the requirement concerning the necessary level of ventilation, and sent their acknowledgement to the team in Haifa. This, however, was where the dialogue had ended, for no one made sure the requirement was met as promised, and, as we later learned, it had, indeed, not been."

Dovrin led the project toward the resolution of the problem, using the systems engineering toolset (which was presented earlier, pertaining to Elbit Systems): good interpersonal conduct and a systematic approach to technological failures: "First and foremost, it was important that I talk to everyone. I avoided the question of who was to blame for the problem, and focused on brainstorming for a solution. This required everyone to pitch in. We went over the system methodically, step by step. For instance, we noticed that we were having difficulties examining a rapidly rotating system, so we detached it and put it on a table. We decided we would not move forward until we made it work on the table.

Once the leak and air supply problems were resolved, the technical failure was fixed, and the system began to work at a constant temperature; we even improved the temperature control algorithm, to boot."

Managing a Group of Systems Engineers. As he rose through the company hierarchy, Dovrin was appointed manager of the Haifa systems engineering group and then promoted yet again – to global manager of the company's system engineering. In this demanding position, he not only had to run the systems engineering, but also had to continue to ingrain the understanding that it was a necessary field: "Seeing as Philips had no systems engineering culture, some of its engineers knew nothing of work methods, and had to be taught how to work correctly. For example, they needed to be told how to properly write the requirements, and how to present them to the people developing the products – the software and electronics engineers. I had to review the work of those who wrote the requirements, and inspect the way the document was submitted.

Aside from that, there was no organized methodology for integration. In Elbit, the person in charge of integration was a systems engineer. Not so in Philips: integration was carried out by testing groups, usually comprised of electronics engineers and practical engineers, because they were more familiar with the system and knew how to examine the hardware during the integration process. But the electronics engineers mostly looked at how the hardware functioned. Once they heard there was to be software as well, they stepped aside. Organizational changes have since been implemented: the hardware testers' team was made part of a systems engineering department, and today, the tests are performed from a more holistic perspective. To support this process, a 'test plan' document is prepared, listing the subject of each test, the person or persons responsible for each stage of the testing, the expected results, and timetables."

The Do It Yourself (DIY) House Project

After about 4 years at Philips, Dovrin left to found his own startup company, convinced that his systems engineering skills would help him on his new venture, just as they had once helped him build his own house: "I decided to manage the construction of my own house, not just because of financial considerations, but because I found the idea interesting. This kind of curiosity is one of the defining character traits of a systems engineer. I began the project by teaching myself about the subject, and formulating a requirements document. My wife and I defined what we wanted, then sat down and reviewed the document with an architect. Next, we presented each of the tradesmen we hired with a copy of the document. Many couples who take on projects like this start with a humble home of 4 rooms, and end up with a palace of 8. They go into a store to choose ceramic tiles, and leave with thousands of shekels worth of marble. In our case, the construction was completed one month ahead of schedule, and the cost was lower than we had planned. This was a highly irregular outcome for a project of this sort."

Insights on Systems Engineering

On the Qualities of a Systems Engineer

– To be a good systems engineer, one needs to understand both the system and the engineering. Many people have a systemic perspective, but trade-offs and other actions that characterize systems engineering are not enough. The engineering part, the ability to closely examine the smallest of engineering details, even those outside one's original field of study, is just as important. If one employs systemic sensibility, but does not design a control system, because he does not know how, or because he finds it uninteresting, then that person is not a systems engineer. If an engineer wants to design a system but is unwilling, or does not know how to prepare a requirements document, then he is not a systems engineer.
– Generally speaking, systems engineering is a work method that allows the integration of different disciplines, for the purpose of producing a complex output. But the fact that somebody studied these methods does not make that person a systems engineer. One needs to have the right qualities and the basic perception, as they are more important than the knowledge. Take someone who knows nothing about the discipline, but possesses a natural curiosity and systems thinking ability; it will take time for him to learn, but when he does, he will get results.

- Suppose I interview someone for a job, who really likes working on printed circuit boards. It is the area where he gained most of his experience, and it is what he does for a living; that is fine, but he cannot be a systems engineer. If, however, he is versed in many different areas, that means he was curious enough about them to go and study them; meaning he has the fundamental traits of a systems engineer.
- As a systems engineer, I can easily enter any field. In Elbit, for example, I was once asked to clarify the company's new pension arrangement to other people. Engineers were complaining about receiving pension reports and not knowing what they meant. My boss thought I could prepare clear explanatory sheets, after I met with the pension company, and gain an in-depth understanding of the subject. So, I did. For three months, I attended regular meetings with the insurance company representatives, and together, we wrote clear explanatory sheets, which the insurance company could use to communicate not only with Elbit, but also with other companies as well.
- One of the most prominent traits of a good systems engineer is the ability to *visualize the end result from the start*; knowing where to go, and aim for that direction from the very beginning of the project. A systems engineer who works this way usually accomplishes his missions quickly and efficiently.

On Systems Engineering and Creativity

- In principle, methodical work can resolve problems, but working methodically without taking shortcuts is only possible in an ideal world. The reality of it is that we live in a world where time and money are limited resources, so one cannot try everything. Therefore, one has to be creative. If you take the road you believe in, you will get results faster. Of course, this way, you might miss out on some even better results, because you have not tried all the options – a "good enough" solution is not necessarily the best solution.

Planning and Organizational Culture

- Organizations should adopt a planning-oriented organizational culture. If a business is disorganized, it is only a matter of time before it fails. The larger the organization, the more crucial this principle; it is less important for smaller organizations.

3.2.3 "MANAGEMENT-ORIENTED SYSTEMS ENGINEERS ALSO SEE THE BUSINESS ASPECTS"

An interview with Alon Gazit, Erez Heisdorf, and Benjie Rom

Is "systems engineer" a profession or a position? In many cases, a profession accompanies a person throughout his life; it becomes an inseparable part of his identity, the very spinal column of his career. A position, on the other hand, is just one of several stages in the progression of one's career, which lasts a definite amount of time.

We have examined this question through the standpoints and experience of three managers in a high-tech company that operates in the printing industry. All three have begun their careers as engineers; served, for a time, as systems engineers; and then moved on the take up managerial positions. As managers, they are in charge of two types of systems engineers: those who see systems engineering as a profession, and those who see it as a position.

From Mechanical Engineer to Systems Engineer

Indigo Digital Press was founded in the late 70s as a startup company in the field of printing technologies. Its founders developed a unique, liquid type of ink (before then, all printing inks were powder based). The company made a strategic decision to develop digital color printers for the commercial and industrial sectors (rather than standard photocopiers). This meant developing a complex product, for a professional market with high entry barriers.

The first steps in this area were taken in the mid-80s. It began with a newly formed team of developers, one of the members of which was Alon Gazit, a mechanical engineer by education and, currently, the company's VP of R&D.

A developer who joins a startup company must adopt a systemic view. At this stage in a company's life cycle, there are relatively few employees, who must each perform a range of different tasks. So, Alon Gazit may have joined the company as a mechanical designer, but in effect, he says: "I did more than just mechanical design. My work included several areas, such as physics, electricity, chemistry and mechanics. Interfacing with all these disciplines had helped me develop a systemic understanding and systemic abilities. Only later, in retrospect, did I realize I had, in fact, been practicing systems engineering as well."

However, even looking back, he does not believe he had been an actual systems engineer, then. This perception extends to his next position in the company, as the leader of a team of engineers: "Management did not make the difference. The team dealt with aspects of mechanical engineering, and did not have an all inclusive view of all the different disciplines. When I was a team leader, I was not familiar with the complexities of other fields. It was far beyond me."

The next phase took place in the early 90s, when Alon Gazit served as a head of a department. The team he had led had taken part in the creation of a comprehensive system for a client, and only then was he acquainted with other disciplines for the first time; a situation that led him to become an actual systems engineer.

Alon Gazit: "Our product encompasses many disciplines, all of which eventually coalesce into ink dots on a sheet of paper: electro-optics, physics, chemistry, software, to name just a few. When we began the integration of the different fields, to make the machine work, I began to familiarize myself with them, to understand them, and, most importantly, to contribute to the development of the product as a whole."

A few more years had passed before Indigo first learned the term "systems engineer," in the mid-90s. At that time, the company was no longer a startup. New employees were recruited, some of them as systems engineers.

Alon Gazit was a systems engineer for a certain period of his professional life, but he no longer sees himself as one: "I am a development manager, who encounters many systemic dilemmas in his day-to-day work. But they are different from the dilemmas of a project's systems engineer. Today, I no longer deal with specifications that need to meet client requirements."

For example, one of the phenomena that embody the systemic perspective – seeing beyond the project level – is "commonality," a situation where a certain project's infrastructure can assist with the development of other products. Another example is when a certain project manager at Indigo, who worked on "product series 4," is required to also make use of products from "series 3." This meant that the systemic view included business considerations, budget constraints, and manpower needs as well. It was about looking not only beyond the system that was the project, but also beyond the system that was the series of products (each product being a project in its own right). According to Alon Gazit, this is a systemic perspective, but it is not systems engineering – it is management.

This approach suggests that in cases like his, a systems engineer is not a profession, but just one job title on the way up to a management position. However, as previously stated, this is not the only career path a systems engineer can take. Another path exists alongside it, and for those who take it, systems engineering becomes a profession.

The Two Development Paths of a Systems Engineer

Our interview with Alon Gazit, for which we were joined by two other senior project managers at Indigo, Erez Heisdorf and Benjie Rom, has helped clear up the distinction between the two types of systems engineers.

We shall set them apart by calling those who see systems engineering as a position "Management-Oriented Systems Engineers" and those who see it as a profession "Professional Systems Engineers." The three interviewees of this chapter are, naturally, of the former category. Today, as managers, they operate systems engineers of both types.

Erez Heisdorf: "Some systems engineers want to manage other people, to organize; they have the personalities of leaders. In contrast, there are those who wish to delve deeper into their fields and mature as professionals. They are not interested in managing people; they wish to focus on the technology."

Benjie Rom: "Management-oriented systems engineers have the ability to take a broad view of the situation. Compared to them, professional systems engineers see less of the edges. They certainly possess a broad perspective, as they must, but they do not see all the business angles. They focus more on the technical side, while management-oriented systems engineers are able to see the product not only through their own eyes, but also through the client's as well."

Alon Gazit: "Management-oriented systems engineers are not afraid to confront 'neighboring' factors, and this sometimes leads to conflicts. Factors like marketing, service, operations, and even matrix bodies [note: matrix bodies are the Indigo term for professional units that support the project (see, for comparison, the engineering units at Rafael, mentioned in the interview with Kobi Reiner)]. They have leadership abilities and communication skills, and they are more willing to compromise – they are potential managers. Professional systems engineers are more solid, more perfectionists. They enjoy dealing with technology, but not the small, technological details – technology in a wider sense. This is why most of them are not inventors. These are mostly found in the matrix bodies."

Alon Gazit believes that the system engineer position is a critical step in the development of anyone who wishes to eventually manage multidisciplinary systems. When asked about the abilities a management-oriented systems engineer needs, he replies with an example: "Suppose there is a multidisciplinary problem that needs solving, and it is not yet possible to even define where it originates, and, consequently, who should be handling it. This is the type of problem a good systems engineer needs to be able to solve, by combining leadership and analytical skills. In a case like this, a professional systems engineer would find the problem more difficult, because, although he has the necessary analytical skills, he will be limited by his lacking leadership ability."

A Systems Engineer at Indigo

The way systems engineering develops within an organization is influenced by the nature of the organization, its organizational culture, and the fields it is engaged in, among other things.

Having joined Indigo as a project manager, after serving as a systems engineer in Elop, an Elbit Systems company (for more on systems engineering at Elbit, see interviews with Mimi Timnat and Boaz Dovrin), Benjie Rom relates his first-hand experience of the differences between systems engineering in Indigo and systems engineering in other organizations. According to him, there is a vast difference between the companies, in terms of their systems engineers' responsibilities: "Systems engineering at Elop is a very clearly defined discipline. The main difference is that the systems engineers there are responsible for designing parts of the system, while here, at Indigo, they have no part in the design. Here, a systems engineer can share his experience with the designers, or take the group in a certain direction, because he has the necessary experience, but he does not actually design; that is done by the matrix bodies. The main reason for this is the complexity of Indigo's products, which necessitates the placement of a systems engineer in each technological group, thus reducing the need for the systems engineers to deal with the project's more technological components."

Alon Gazit provides another comparison: this time, with other technological bodies within Indigo's parent company, HP (which acquired Indigo in 2002): "In the print field, there are similarities between job titles here and at HP. But, in most cases, a systems engineer at HP has less room for growth. The reason for this is Indigo's organizational structure and technology, which force good systems engineers to interface with many more disciplines than the average systems engineer at HP."

He demonstrates: "In the late 80s, HP developed the inkjet technology, which included two main areas of development: the ink and printheads, and the printer itself. Each field was assigned to a different organization within HP, which had its own separate systems engineers. In our company, the technology that facilitates the interaction between the ink and the machine is much more complex, and forces the systems engineer to deal with a much wider range of disciplines. So, in principle, there are similarities, but the levels of complexity are different."

Erez Heisdorf expands on the subject of complexity, by demonstrating: "One of the subsystems of a printer is the feeder, which allows both sides of the sheet to be printed on, and then proceeds to stack the sheets, after the printing process is complete. The group tasked with the development of this subsystem has its own, internal, systems engineer, because the system is a combination of mechanics, electronics and software, and someone needs to manage its systemic aspects. The machine has five or six such subsystems, and they, in turn, must be synchronized with each other. We call the systems engineer in charge of this subsystem a 'functional systems engineer'."

At Indigo, the tasks a systems engineer receives match his abilities and traits. It follows that there is more than one type of systems engineer in this company.

Alon Gazit: "To me, a project's systems engineer is the highest technical authority. I can give an example of such a person. He was the right hand of the program manager. He was also in charge of the distribution of the specification's components between the different units, the distribution of the budget to the various disciplines, in addition to integration and validation [validation is a process that makes sure the system meets the specified requirements]. Alongside that, there were other systems engineers who worked underneath him, and dealt with the details.

The matrix bodies also have systems engineers. For instance, the software body has a systems engineer who manages the software field, and underneath him, there are systems engineers who have to do their best to define the software before it is written, and coordinate between the different software teams that write it. So, there indeed are different types of systems engineers."

The Growth and Development of a Systems Engineer at Indigo

During the first years of Indigo's existence, systems engineers emerged from within the ranks of its employees. Some of them, like Alon Gazit, have since become the executive spine of the R&D division. The source of this phenomenon is discussed in the beginning of this chapter: in a small organization, there is a high chance for systems engineers to handle a number of different areas and be intimately familiar with the company's range of activities. Hence, a large part of the systems engineers who evolved at Indigo belong to the "management-oriented" type."

But, as the company grew, so did the difficulty of developing new systems engineers out of existing employees, and the need arose for recruiting systems engineers who had grown and gained experience elsewhere. A considerable part of these systems engineers could be classified as "professional systems engineers."

Alon Gazit is of the opinion that a major cause of this is the company's size. Size necessitates specialization, making the employees of a company focus on narrower, more specific fields, and reducing the chance of their learning a wide range of technologies. One possibility of bringing about such development, besides recruiting systems engineers, who have already gained experience in other organizations, is sending employees to be trained outside of Indigo. Yet, these courses mostly help broaden their attendants' horizons and foster new thought patterns. Naturally, they are unable to meet a company's concrete needs. These needs can be met in on-job training, as well as internal training courses.

Other chapters in this book have already addressed the question of which base discipline systems engineers grow out of. Some have argued that the answer depends on the core disciplines the organization deals with. It appears that this is true for Indigo, where a considerable part of the systems engineers began their careers as mechanical engineers.

Alon Gazit: "Traditionally, in our company, systems engineers start out as mechanical engineers. This is a direct result of how Indigo was established. The company had set out with many mechanical engineers, because the machine has a lot of 'metal.' The integration process is also carried out by 'metal.' But systems engineering is more than just hardware; almost every field of mechanical engineering is present here, from thermodynamics to optics, to materials engineering."

The people of Indigo believe that systems engineers should, ideally, be engineers by trade, although they can also be physicists. In their experience, the engineering background of systems engineers is usually in either mechanical or electronics engineering (they perceive aeronautical engineering as a subdiscipline of mechanical engineering).

And what about software engineers? Alon Gazit: "There are systems engineers with that background, too, but they usually prefer to take up handling the systemic aspects of software."

Further Insights on Systems Engineering

On the fundamental traits of a good systems engineer

Benjie Rom:

- Curiosity and the desire to learn different fields, rather than specialize in one discipline.
- The ability to make decisions and distinguish the essential from the nonessential. When employing a broad perspective, this becomes an everyday need, because there is never much room to maneuver, as there are not only technological constraints to work within, but also a demanding schedule and limited resources.
- The ability to delve into the details and analyze. This is true for both professional and management-oriented systems engineers, with one difference: depth. Still, both must have an analytical mind.

Erez Heisdorf:

- Good systems engineers need to be able to work on two depth levels simultaneously. They must be able to go into the small details when they meet with the work teams, and be able to see things from very high up at the same time. People cannot always "live" on these two levels; it is a unique ability.
- Common sense.
- The ability to make decisions.

3.2.4 "OPTIMIZATION BY THE TOP RANKS"

An interview with Dr. Amir Ziv-Av

Systems engineering is a new discipline that still seeks to define its place and interrelations with other disciplines and fields that exist alongside it or overlap with its areas of activity. One such area is optimization, the purpose of which is to find an optimal value for functions, under given constraints.

We have stated repeatedly that systems engineering emerged as a result of the increasing complexity of technological systems; a complexity enhanced by the development of the software field and its integration in technological systems. But technology, complex though it may be, is usually only one part of larger, more intricate systems that assist modern human society in its day-to-day conduct. One of the more obvious examples of such complexity can be found in transportation systems.

This chapter deals with the combination of these two areas, namely, optimization and complex technological projects, and their affinity toward systems engineering. It revolves around an interview with Dr. Amir Ziv-Av, Chief Scientist at The Ministry of Transport, who also wrote his doctoral dissertation on systems optimization.

Personal Background

Having graduated high school with a practical engineering diploma in mechanical engineering, Amir Ziv-Av's enlistment with the IDF Ordnance Corps was only natural. After being discharged from his military service with the rank of an officer, he decided to further his study of mechanical engineering and applied to Tel Aviv University. Even as a student, he worked in his area of expertise, providing planning services to private companies and to the development units of the IDF's Combat Engineering Corps. After receiving his Bachelor's degree, and having simultaneously completed part of his Master's degree studies, he received an offer to return to his old unit. As a young Major, he headed the Department of Mechanical Development, with a team of 12 engineers under his command. About two years later (in 1979), after completing a (short) command and headquarters course, he was transferred to the headquarters of the Engineering Corps, to found the corps' own development branch. At the same time, he was also busy with projects related to the Merkava Main Battle Tank.

Amir Ziv-Av: "During that time, I began effectively practicing systems engineering, as I formulated the methodology for the analysis and development of the process of breaking through minefields. I used the Air Force's operations research branch, because The Engineering Corps still spoke the traditional language of "mines per frontline meter," while I wanted to speak in terms of optimizing the resources used by an attacking force attempting to overpower a defending force in a barricaded position. Clearly, I was leaving the boundaries of my area of specialization – mechanical engineering. My work in the development branch enhanced my systemic perspective, particularly the ability to define and analyze a system. It was my first practice of systems engineering. When you are the head of an entire corps' development branch, you do not actually do any developing, rather, you define and accompany the development process. The means at the disposal of any corps are multidisciplinary in nature."

In the early 80s, after approximately six years of standing military service, Amir Ziv-Av left the IDF and completed his Master's degree in mechanical engineering. During the 80s, he served as Head of Opto-Mechanical Development in Optrotech, a company that developed automated, optical systems for inspecting printed circuit boards. He did not arrive at the position as a systems engineer, for the term was not widely known at the time. Nonetheless, his career path was paved by "systemic" traits.

Amir Ziv-Av: "I had not come from the hi-tech industry, nor was my education of great relevance to the position. For the most part, Optrotech hired me because of my systemic perspective. Most of the people who reviewed my candidacy were physicists.

Mechanical engineering is not the ideal background for a systems engineer, and the ability to practice systemic subjects depends more on the person than on the discipline he specialized in. A systems engineer does not always have to be an engineer, and the higher you climb to look at the system from a bird's eye view (and the further away you get from technological issues), the less necessary is it for you to be an engineer.

I am also not afraid of venturing into unfamiliar areas, because when you look at everything from up high, you apply the same principles to all fields. I learned the principles of optics, asked questions and received all the help I needed in my search for optimization. The questions that arise in optics are no different than in other fields: 'which trade-off should I choose?,' 'what is the level of mechanical accuracy needed here?,' 'what extent of control or accuracy do I need?,' or 'what is the risk level posed by this new concept?'."

Systems Engineering and Optimization

Amir Ziv-Av sees a close connection between optimization and systems engineering, because one of systems engineering's main goals is optimization. He believes that one of the core characteristics of systems engineering – a holistic, all-inclusive perspective – stands at the heart of what optimization really means: "viewing the system as a whole, not as one discipline or another, but as an ensemble of economic, operational and technological components. The essence of the connection between optimization and systems engineering is the development of optimal concepts for correct integration of all the technologies at our disposal. A 'product' is an answer to a collection of differently weighed objectives, and at the heart of its development process stands the task of maximizing the target function. In the end, to win the competition over the heart of the client, one must have a relative advantage, and how does one obtain that? By doing more with fewer resources; that is the bottom line."

Which brings us to the importance of intuition: "One shouldn't force quantification on everything. Some complex situations include so many parameters, that weighing each one is impossible; other times, there is missing data. This is when intuition comes into play."

A well-devised strategy can greatly enhance the ability to make the right decisions, even under conditions of uncertainty: "The product concept is its strategy, and the product tactic is its details. If the strategy is good, mistakes can be corrected, even if the tactics are wrong. Take, for example, development. Being an iterative process (a process that repeats itself in order to examine a range of different situations. If errors are discovered, they are fixed, and the process begins again – the authors), a good development strategy keeps the developers on the right track. A bad strategy, on the other hand, is very difficult to correct, no matter how good the tactics."

One of the best examples of this, according to him, is the 2006 Lebanon War, where an incorrect strategy led to failure and to the eventual appointment of a committee of inquiry. This happened despite the impressive heroics seen in the field and the good field rank performance at the battalion and company levels (to this we, the authors, add that had the strategy been better, the soldiers and commanders in the field may have had fewer heroic acts forced upon them): "They were unsuccessful in changing the general direction. The same holds for product development. When the general direction is the right one, corrections can be made en route; but, when the project is heading the wrong way, there is nothing to be done."

After his time in Optrotech, Amir Ziv-Av transferred to Keter Plastic, where he served as Head of Development and management member. His experience there taught him, once again, of the importance of good strategy: "The CEO and owner of the company had a clear strategy in mind. He defined the direction the company was headed. At the time, the company's scope of sales had been roughly 50 million dollars a year; today, it is close to one billion. Everything he had said he was about to do, he did: he decided what to produce and what not to produce, who to compete with and who to cooperate with, where to establish local factories, who to ship to, when to launch which product and what risks to take in the process, who to copy from and when to take initiative, and at which point in the factory's growth it needed to be split in two.

The internal workings of the company were another matter. There were various conflicts and differences, such as competition between two internal elements in production. But when the strategy is so good, these little things are not enough to hinder it."

The striving toward optimization entails two main principles: robustness and simplicity.

Amir Ziv-Av: "When you set off on the road, especially if it is a long or tricky one, there is a lot of uncertainty. This is why *robustness* is such an important component in optimization – it is the ability to aim for a system that is insensitive to changes in the operating point. This is achieved with wide design margins (referring to reserves, not to marginal situations) and modularity, so that when something new comes up, it can be integrated into the system with no difficulty. This way, even if, ten years from now, the system is not in its prime, it will be in the right environment. And if there is a problem, it will either be easy to fix, or, its consequences will be possible live with.”

Amir Ziv-Av then talks about the Trans-Israel Highway (also known as “Highway 6”) as a good example of the robustness principle. When the highway was being designed, the traffic load and number of lanes needed were impossible to predict. So, only two lanes were paved in each direction. After the highway proved to be surprisingly popular among commuters, it was decided to add another lane. Seeing as this was taken into account during the planning stage, the addition necessitated no changes to existing infrastructure: “We did not have to touch a single interchange connecting the highway to the lateral roads, not one bridge, street light or drainage system; even the guard rails were left almost entirely untouched. It was a very robust solution.”

Simplicity means action is not always necessary. Sometimes, the right thing to do is to do nothing, making inaction the optimal solution.

For example, the changes in Dizengoff Square: “Dizengoff Square was a beautiful, safe plaza with good air circulation – a pleasant urban environment on all accounts. They took a good thing, invested tens of millions of shekels in it, and turned it into a bad thing. Now, the streets are congested, and the atmosphere is unpleasant.”

Amir Ziv-Av gives two examples of strategic failures in areas nearing his current occupation: the aviation and automotive industries.

On unnecessary developments in aviation, the failure of which, could have been avoided beforehand:“The Concorde airliner was a technological masterpiece and, at the same time, an economic failure. The fact that the investment would not be returned could have been predicted. It was a known fact that the price of a seat on a passenger airline was derived from the cost, and ranged around 300 thousand dollars, independent of the airline model. A seat on the Concorde cost around one million dollars. No airline ticket can justify such a price.

To return the investment entailed in the development of a passenger airliner (and we know that the investment in the Concorde was not far off that of the Jumbo jet), one needs to sell between 200 and 300 units. The Concorde sold only 15 – a massive economic failure, while the Jumbo sold over 2000 – a huge success.

The Concorde’s advantage of speed was only relevant for the time spent flying across The Atlantic, seeing as the extra time spent by the passengers at the airport before they boarded the plane and after they got off it remained unchanged. The downside – three hours spent on a crowded plane – did not justify the difference in ticket prices.

A similar phenomenon occurred with the Airbus A380, a giant jetliner that required adjustments to be made in many airports to accommodate it. Why did the developers assume the airport authorities would agree to change their runways and terminals just for them? Moreover, the process of shipping the enormous parts of the body of the plane from the subcontractors to the assembly factory was extremely complicated. The fact that so far, only a few dozen places were supplied is a testimony to the project's failure. Each year, the cash-flow is delayed by many more billions of dollars – a financial disaster."

On redundant mergers and acquisitions in the automotive industry: "Some of the mergers that took place in the automotive industry were unnecessary, and ended in major losses. Mercedes, a profitable, stable, focused company, joined with disorganized and inefficient Kreisler. The endeavor ended with the two companies splitting up and suffering billions in damages in the process. Ford bought and then sold Jaguar and Volvo, losing money on both deals. GM began producing the utter nonsense called the Hammer, only to sell it later at a loss."

To the question of how, in spite of everything, these fiascos come about, Amir Ziv-Av responds with one sentence: "The megalomania of executives makes them lose focus, and focus is almost always a condition for success."

Systems Engineering, Optimization, and Transportation Systems

One of the most prominent expressions of the mutual affinity between systems and larger systems can be found in the content worlds Amir Ziv-Av inhabits today. As the Chief Scientist at the Israel Ministry of Transport, National Infrastructure and Road Safety, he is in charge of formulating the strategy of transportation systems.

For instance: when planning the layout of a road, the considerations include such parameters as road length, which needs to be minimized, so that the road takes up as little area as possible; and the radii of turns, which should be maximized to raise safety levels. The planners strive for making the road as inherently safe as it can be, to lessen the need for additional safety means, such as guard rails. Other considerations include minimizing incident sensitivity (among other reasons, to avoid a situation where an accident causes a roadblock; a major problem even for short lengths of time). All these parameters then need to be considered from a systemic perspective, so as to reach the optimal result.

The rate of mileage increase in Israel is three times the rate of increase in road area. The authorities are attempting to close this ever-increasing gap in two ways: the first, lowering mileage by transitioning to available and inviting public transit, and the second, increasing the efficiency of land infrastructures. Another helpful means would be changing behavior patterns by encouraging people to start their workdays at different hours of the day or working from home, but it lies outside the jurisdiction of the Ministry of Transport.

Infrastructure efficiency necessitates optimization, which relies upon systems engineering. This is similar to aerial transport, where the planner can, for instance, decide that in three months, on a certain day, at a certain time, a certain flight will pass over Greece. But such accurate control is impossible in land transportation. So, according to Amir Ziv-Av, the authorities settle for dealing only with the national transportation network and targeting congested areas: "We cannot manage the single car traveling from Be'er Sheva to Arad, but we can manage heavily congested regions like the Tel Aviv, Haifa, and Jerusalem Metropolitan Areas. We need to plan the system so that, for example, a train does not leave five minutes before the bus arrives at the train station. To prevent that, we need to manage traffic signal timing or interchanging lanes, when most people travel in a certain direction; all while giving priority to public transit and emergency service vehicles, as needed."

Today, traffic data is available to any driver, so they can do their own, individual optimization. The massive use of smartphone devices allows this level of data accessibility. The pedestrian or driver will simply state his destination, and the device will present him with the optimal route, suited for his individual needs. Technology will also work against such nuisances as queues forming at the entrances and exits to and from parking lots, by using automatic billing, a technology that already exists and is used today. An automatic billing system, like the one in use on Highway 6, can be installed at parking lot entrances or exits.

An example concerning the use of cameras as a means of enforcement.

Amir Ziv-av: "Transportation in Israel is considered safe by international standards. One of the main reasons for this is the substantial improvement of infrastructures (grade separation, guard rails, and roundabouts). The use of road safety cameras for speed limit, red light, and center line enforcement is another important factor. Studies have shown that placing cameras on road systems can significantly lower accident rates. In France, adding 3,600 cameras to support enforcement has lowered the traffic related death rate by 50% in ten years. An equivalent effort in Israel would require 1,500 cameras. At present, however, there are a mere few dozens. Expert opinions and relevant authorities (like the traffic police and the Road Safety Authority) believe this investment to be worthwhile, and human benefits aside, it is also feasible economically. Nevertheless, implementation has been very slow."

He gives another example, this time, of non-systems thinking: "One of the problems with implementing traffic control programs is that the technological systems are not integrated. It is a 'Tower of Babel' type of situation. The traffic light control center of the Tel Aviv municipality does not serve nearby cities, whereas the traffic itself belongs to a single, metropolitan urban entity. The decision to use a certain technology is mostly driven by the results of tenders, each of which took place at a different time, under different circumstances. On a national level, there is no optimal, all-inclusive, systemic-technological perspective. In other words, there is no systems engineering. For example, the Tel Aviv Municipal Control Center may find it easier to manage all fifteen cities of the Tel Aviv Metropolitan Area, than to wear itself out trying to interface with them, as it does today."

But Amir Ziv-Av is optimistic: "I am currently leading the ministry in this very direction – toward comprehensive systemic optimization, on a national level. All in all, people want things to change for the better, so long as the change does not come at their expense. Still, they recognize the need for integration and coordination on the road towards finding a systemic solution that overcomes constraints; for example, by establishing a metropolitan authority with representatives from each of the member municipalities."

Further Insights on Systems Engineering

On the evolution of systems engineering

– "Today, most products are interdisciplinary, and so the need for systems engineering is on the rise. 30 years ago, for instance, cars were, for the most part, mechanical products; so were jumbo jets. Today, cars are equipped with computers and communication systems, even the mechanism that opens the window has some small processor in it – everything is mixed together. The Smartphone is at the forefront of technological, software and communications knowledge. But if it can fall on the floor, survive a shock of 500 Gs and not break, it means its mechanical engineering is also cutting edge."

On the essence of systems engineering and its affinity for optimization

– "Systems engineering means zooming out. Not dealing with the molecules, the 'micro,' but with the macro. If you zoom in, you will see the molecules. A systems engineer does the opposite – he zooms out, decides on the various disciplines that, together, form the solution, and defines the interfaces between them. Systems engineering is optimization of the highest order. It places a lot of weight on operations research and entails many legal, as well as economic considerations."

On a systems engineer's professional background

– "A systems engineer doesn't necessarily have to be an engineer. The central trait of a systems engineer is a comprehensive view, an ability based on the systems engineer's personal skills, rather than his area of study. Of course, this depends on which level of the system the engineer is stationed in – the higher the level, the less engineering skill is needed. The more you zoom out, the less important an engineering background becomes."

On engineering and systems engineering studies

– "Decades ago, engineering studies were four years long. They are still four years long, today. How can this be? Knowledge has increased a thousand fold – they say it doubles every few years. The answer is that luckily, base disciplines – physics and mathematics – change at a much slower pace. These are the subjects that develop skills. Those who study them receive the tools for learning and understanding the other disciplines. Mechanics, for example, includes several fundamental physical principles, and the rest is mathematical developments. If you studied mathematics and physics, but know nothing about heat transfer; when you open a book on heat transfer, you will be able to understand it, even if you are not an energy engineer. Perhaps, in the distant past, students learned a large part of the mechanical knowledge in existence during their first degree studies; while today, they learn only a tiny bit of it. But if the engineer has the ability to learn, that does not matter."