1.2

A MULTIDISCIPLINARY, SYSTEMIC VIEW

In the previous chapter, we discussed the trends that led to the increase in the need for systems engineering, namely: *the ever-growing complexity of technological systems*, alongside the ever-increasing demand for *appropriate solutions for the needs of the clients* – the buyers and users of the systems (the two are not always the same). This combination compels engineering teams charged with the development of technological products to account for nontechnological constraints, related to finances and management. Therefore, the systems engineer who manages these teams should be willing to engage in areas beyond his formal engineering training, in the desire to meet the clients' needs, by exposing himself to broader technological fields, while handling managerial and organizational issues.

On the increasing complexity of systems:

The continuous accumulation of knowledge allows for the creation of advanced systems that are, naturally, also very complex. It is a well-established fact that the more complex the system, the higher the risk of it being prone to faults and difficult to operate. It follows that one of the main challenges is creating a product that is as technologically advanced and, at the same time, as simple as it can be. This increases the need for *strong simplification capabilities* alongside *efficient examination of alternatives*.

The Iron Dome developers attested to this: "We could have gotten a more complex 'servo' in one round of development. To reach a 'servo' that simple required more thought and more talent. Finding a complicated solution is fairly easy. To find a simple solution, you need to start thinking."

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

On the increasing importance attributed to clients' needs:

The last few decades are characterized by the increasing power of clients everywhere: from fashion retail to education and healthcare services. This trend did not skip the technological world, where, often, the client's representatives are no less competent than the developers they meet with. Clients' control systems for complex technological products have grown tremendously, and their involvement in all stages of product development has increased. The need to account for the clients' needs and demands has become paramount. Among other factors, this trend is the result of the changes in budgeting methods, as clients are now much less lenient when it comes to deviating from predetermined financial frameworks, and so grew the demand for engineers who knew *how to handle themselves with the clients' representatives*, who were able to negotiate with the clients and speak their language.

In many cases, clients see themselves as the ones making the demands, and the developers as the ones tasked with meeting them. This pattern of conduct tends to have a negative impact on the work rate and even on the quality of the final product. Today, the world is beginning to realize that this fine weave of relations has to be handled wisely and with care, which brings us back to The Iron Dome Project, whose developers stated that in their case, "The client almost merged with the project. This does not go without saying, and there are those who even criticize it, saying that perhaps it is best for the client to keep his distance, so that he may represent the other side, and maintain his ability to provide objective feedback. In The Iron Dome Project, however, it worked very well, because of the client's representatives' ability to successfully maintain their independent thought."

1.2.1 THE BOUNDARIES OF A SYSTEM

One of the major dilemmas encountered by all who practice systems engineering, including the systems engineers themselves, is the question of defining a system's boundaries. Thus, for instance, one might decide that the boundaries of a system are the client's technological requirements of the project. But one might also decide to include the technological system's impact on the environment. Naturally, such decisions can radically change the design and character of the system.

Expanding the boundaries of a system also reveals context that cannot always be seen from within the system's original boundaries.

Prof. 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."

Prof. Joe Kasser stresses this as well: "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."

1.2.2 SYSTEMS OF SYSTEMS

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. Today, many systems are starting to turn into ‘**systems of systems**’. They become very large and more and more complex. We start connecting systems that were not designed to work together.”

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

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

Olivier De Weck gives an example from the field of transportation: “Drivers texting behind the wheel is currently the most common cause of traffic accidents in the United States. 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.”

Sillitto says: “The more complex systems become, the more the connections between them multiply, the higher the chances that something will go wrong, be it on purpose or due to plain stupidity. Thus, the importance of the need to balance opportunities and risks cannot be stressed enough.”

The multileveled nature and complexity of systems raise the importance of *risk management* in all systems, whether they pertain to engineering or not. Dr. Gillie Fortuna gives an example from a system that constitutes the organizational structure of a major corporation, spread out around the globe – Teva Pharmaceutical Industries. Efficiency considerations have led Teva to place its marketing and production arrays under different managements. This created “a lot of interdependency between subsystems. 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.”

Dr. Amir Ziv-Av also raises the importance of *optimization*, which he defines as: “Viewing the system as a whole, in its ensemble of economic, operational and technological components.” According to him, “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, which is attained by doing more with fewer resources."

1.2.3 MANAGING THE HUMAN FACTOR

The increasing complexity of technological systems has, in turn, impacted the *complexity of organizational systems*, where systems engineering operates. Developing a complex technological system requires the skills of many people, hailing from many different fields. Therefore, *the systems engineer's integration work does not end with the technological context; today, more than ever, it is required in the human context as well*. More and more experts believe that the importance of the ability to lead multidisciplinary teams outweighs even that of the systems engineer's technological competence.

This need has grown even further due to globalization, which played a key role in the increase in the power and importance of multinational corporations. These companies wish to utilize their advantages of size and global deployment by running multinational, multidisciplinary work teams, spread out around the globe. Of course, this approach makes the task of directing them even more difficult.

1.2.4 TRAITS DERIVED FROM AN INTERDISCIPLINARY, SYSTEMIC VIEW

The aforementioned shows just how crucial it is for a systems engineer to possess *an all-inclusive, systemic view*.

Boaz Dovrin is of the opinion that a key condition for a systems engineer's success is his ability to visualize the end of the project from day one. Prof. Ovadia Harari explains the necessity of this trait from the opposite direction as well: "a systems engineer must see the whole picture and use common sense to filter out the less important details, otherwise, the endless dive into the small details will disrupt his work processes."

An all-inclusive, systemic view cannot exist without a second important ability: *multidisciplinarity*. In the words of Prof. Aviv Rosen: "systems engineers are people who are willing to delve into areas outside their natural habitat."

Ovadia Harari illustrates this statement: "a systems engineer should take the budget issue into account. He must understand that money is a vital parameter. If a systems engineer is focused solely on technology, he will not have the correct balance, required of a good systems engineer." He also adds that "a systems engineer should be able to *talk to a variety of experts in a clear and simple language*."

The ability to simplify is important, not only for dialogue with different experts, but also to provide answers to technological needs. John Thomas believes that the ability to simplify things, "to see beyond technology and understand the problems," allows a systems engineer to fulfill one of his most important missions – *problem* *solving*. John Thomas also uses the word “audacity” to describe another trait required of systems engineers. A systems engineer must be audacious, to be able to strive for the accomplishment of his tasks, while breaking through the obstacles in his path and dealing with limitations and difficulties.

Other experts also see *systems thinking as a way to solve problems*. Sandy Friedenthal, for instance, says he applies systems thinking to everything he does: “it’s a way of thinking, an approach to problem solving. The focus of this approach is to understand different stakeholder perspectives and concerns, and define a problem first before jumping to a solution. Then establish value from the perspective of the stakeholders, determine alternative approaches to address the problem, evaluate the alternative solutions, and validate the solution addresses the need. System thinking provides a way to think about how the pieces of the solution fit together to address the problem.”

Another important trait all systems engineers should possess *is the ability to choose between alternatives*. This is important, because “today’s infinite (technological) possibilities” create a wide range of options on the one hand and a large number of constraints on the other hand – a situation that forces one to choose wisely.

Norman Augustine comments on this: “compared to engineers, who solve engineering problems, *systems engineers face problems that have intrinsic conflicts and numerous components*. They do this by analyzing and by making trade-offs: this is *a process of balancing out different considerations that also affects the determination of the appropriate* ‘*dose*’ *of each component in the system*. This is one of systems engineering’s most important areas of activity.”

In the same context, Dr. Kobi Reiner adds: “a systems engineer has to be able to cut. Engineers tend to complicate things, and it is his job to stop them, because today, the possibilities are endless. One of the key traits a successful systems engineer must possess is *the ability to simplify, when the atmosphere is one of complexity and complication*. A good systems engineer prevents complication from emerging.”

These traits allow the systems engineer to shape the craft of *coordination and integration* between the subsystems that make up the system he is in charge of.

For a systems engineer to have these abilities, required for successfully doing his job, he must first possess a row of fundamental traits. Ovadia Harari presents three such traits:

The first is a mix of *openness, curiosity, and refusal to accept things as they are*: “systems engineers are dynamic individuals, involved in many different areas … they are open-minded; they ask questions and have a dialogue with you (as a manager). If you want something from them, they don’t just go and do it, they ask you what you need it for. They sit with you and examine whether a different solution is also acceptable … ”

The second trait is *learning ability*: “a systems engineer needs to be able to learn, to expand his mind, even at the age of 45. For example, the client raises a demand for an electric flight control system. You cannot tell him: ‘no. I will revert to a mechanic flight control system, because that is what we know.’ You open up the books, you take some courses, and you begin to understand.”

The third trait is methodicalness: “a systems engineer is a methodological person. He has a method and the tools to use it. He must adopt process-oriented thinking.”

Dr. Gillie Fortuna adds: “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 … a manager must maneuver between the need to focus and make operative decisions, and the ability to adopt a broad perspective of the consequences of the realization of the vision, and the path that leads to it.”

The larger and more complex the system is, the more important the management mechanisms that control it. This effect reaches so far down the hierarchy that today, more than ever before, systems engineers are required to possess management skills as well.

Prof. Joe Kasser finds that the two areas, management and systems engineering, are intertwined: “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. 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.”

The next chapter is dedicated to this issue precisely. In it, we will discuss one of the most vital overarching traits a systems engineer must possess – leadership skills and management abilities.