1

SCIENCE, ENGINEERING AND SYSTEMS ENGINEERING

A number of recognized scholars and historians have written extensively on science and engineering, from its meaning and origin to its current status [Leonard, 1989; van Doren, 1991; Checkland, 1993; McClellan and Dorn, 1999; Gribbin, 2002; Bryson, 2004; Priestley, 2014]. It is not the purpose of this chapter to present a thorough review of science and engineering, but just to set a basic framework for the better understanding of the advent of systems engineering as a discipline in the second half of the 20th century.

1.1. Science

Long gone are the days when our ancestors would look terrified at a thunderstorm, at an erupting volcano or at so many other natural phenomena. In their inability to come up with an explanation they resorted to gods and mythology to explain what in one way or another caught their attention or shocked them.

Science is one of the major activities of the human mind. It can be defined as the knowledge gained from the physical or material world through systematic study, observation and experimentation. The word 'science' comes from the Latin word *scientia*, which means knowledge. Science as we know it originated about 2500 years ago in the ancient Greece. Thales was a pre-Socratic Greek philosopher who lived in Miletus

(Asia Minor) circa 624 to 546 B.C.E. He is known as the first true scientist as, according to available records, he was the first person to try to explain natural phenomena without reference to gods and mythology. Thales tried to develop testable and verifiable explanations about the universe and the behavior of things. The Greeks succeeded first of all by discovering nature, willing as they were to consider the world as a natural system, governed by natural laws, thus leaving the gods out. The Greeks were interested primarily in knowledge and understanding, and only secondarily in practical usefulness. This focus on knowledge per se is what we would call today basic research.

Science involves structured and unbiased observations, from which understanding of the observed phenomena is gained. The conducted observations enable the formulation of general laws or rules; further observations will result in those laws either being re-confirmed, rejected, or in need of amendments. That is the process of the so-called scientific method, which has been described by a number of authors [Carey, 2010; Gauch, 2012]. The scientific method is the body of techniques for investigating phenomena and gaining new knowledge, as well as for validating or amending what we already know. Knowledge is gained through the recording and analysis of observable, measurable, empirical and reproducible evidence, subject to the laws of reasoning. The scientific method is thus the procedure by which knowledge is generated in empirical studies. The main advantage of the scientific method is that it is unprejudiced as theories are accepted based on the result of tests and experiments that anyone can reproduce. The scientific method is divided into six steps, as depicted in figure 1.1. The first step is the making of observations and the formulation of questions. This first step is conducted somehow informally and not necessarily in a structured manner. Second, a hypothesis is constructed and it will allow the prediction of future observations. Third, an experiment is designed to test the validity of the hypothesis. Fourth, the experiment is run and data are collected. Fifth, based on the gathered evidence the hypothesis is approved or rejected in total or in part. If the hypothesis is not approved then it is necessary to go back to the second step, and to reformulate the hypothesis and/or redesign the experiment, as needed. In any case, the sixth and final step is to report the results. This last step is essential because it enables the needed repeatability of the experiments, which is the cornerstone of the scientific method.

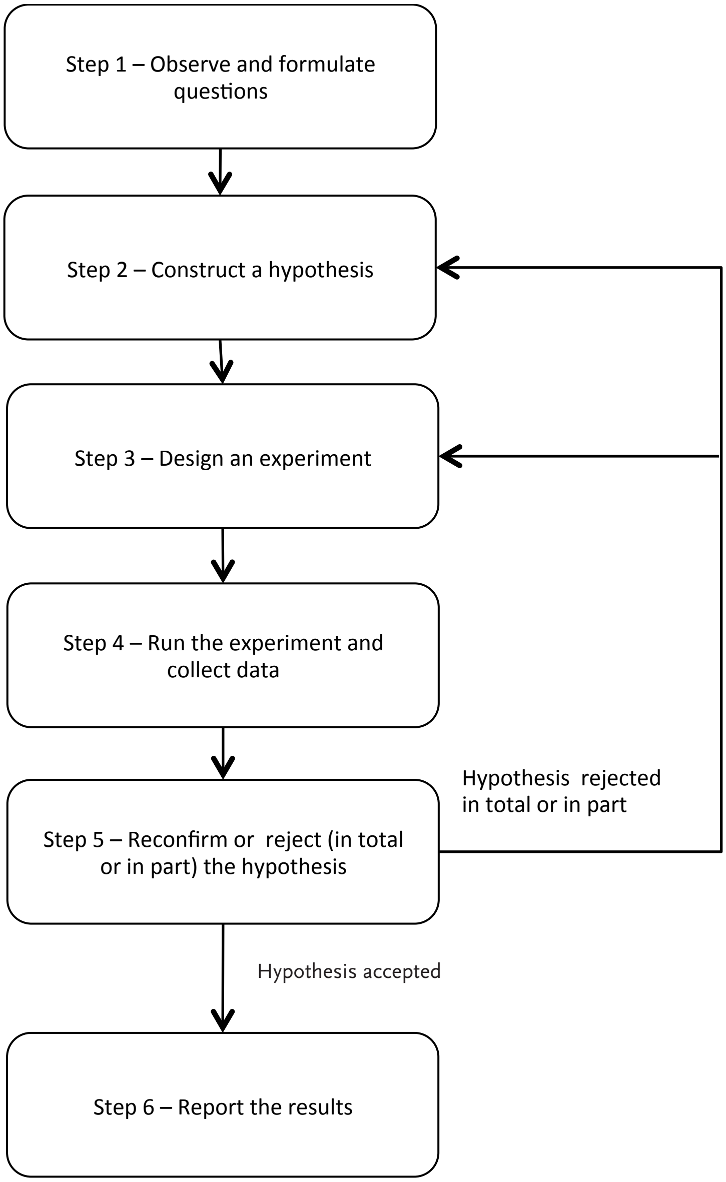

Figure 1.1. The scientific method.

Research enables the advancement of science. In the words of Hungarian-born and Nobel Prize winner Albert Szent-Györgyi, *to do research is to see what everybody has seen and to think what nobody has thought*. Research is frequently divided into two categories: basic research and applied research;

the former is performed without thought of practical ends, seeking to widen the understanding of the phenomena of a certain field, whereas the latter refers to scientific studies and activities that seek to solve practical problems or to achieve specific goals. Basic research is the true pacemaker of technological progress. If basic research aims at extending the boundaries of fundamental understanding, applied research is always directed toward some identified societal problem or need.

1.2. Engineering

Benjamin Franklin, a noted polymath, used the term *homo faber*, Latin expression for *man the creator*, to refer to the human being as the designer and developer of artificial objects, tools in particular, to achieve specific goals. Human beings have the faculty to create artificial objects, in particular tools to make other tools, in order to achieve specific goals and to ultimately control and alter the environment. Engineering is the application of the possessed knowledge to the design and development of equipment, machines, artifacts and any other devices. The term *engineering* derives from the Latin *ingeniare* and is related to the word engineer, which means designer or constructor of engines.

The discipline of engineering is extremely broad and encompasses a range of more specialized fields or branches of engineering, such as marine engineering, aeronautical engineering, civil engineering, telecommunications engineering, electrical engineering, chemical engineering and so on, each one with a more specific emphasis on a particular area of technology and with specific applications. The fundamental difference between science and engineering is that the former is the urge to know and understand things, whereas the latter is the urge to do things and to achieve practical ends. The word engineer conveys the meaning of inventor or designer of machines and artifacts. As the Hungarian engineer and physicist Theodore von Kármán put it, *scientists study the world as it is, engineers create the world that has never been*.

One of the problems experienced by humankind with the diversification of engineering branches or disciplines was communication among specialists. Although such diversification allowed putting into practice in many fields of the knowledge acquired in scientific endeavors, thus facilitating the design and development of many complex devices that in general improved the quality of life for human beings, it had the drawback of complicating communication among specialists in different areas. A deeper view into any one area normally implied lack of holistic perception, given that problems and opportunities are multifaceted, requiring a very broad perspective to fully understand their nature. That lack of global view and communication among specialists prevented the design and development of truly effective and efficient solutions. Furthermore, it implied little cross-fertilization across fields and consequently an excessive redundancy of efforts, which frequently meant an inefficient use of scarce resources.

1.3. Systems

Many definitions have been given for the word *system*, which comes from the Latin *systema*, derived itself from the Greek *σύστημα*, or a whole made up of of several parts or members. A system is a group of interacting, interrelated, or interdependent components that constitute a whole and that has been designed to fulfill an identified need. Systems can be classified as natural or human-made, and this book is about the latter. Generically, those things designed and developed by human beings to achieve stated goals or satisfy perceived needs or opportunities are called systems. Etymologically, the word system comes from the Greek *systema*, a set of inter-related elements that work together toward the achievement of a common objective. That is a crucial concept, as not any collection of elements constitute a system. There have to be relationships among the elements, and collectively they have to exhibit behavior receiving inputs, processing them and yielding outputs, for those elements to constitute a system. Human-made systems are designed and developed to cope with identified needs or perceived opportunities. The process of designing the right system requires a true understanding of the need or opportunity at stake. Translating the need or opportunity into the requirements that drive the design and development effort is frequently a formidable challenge. Customers' needs and wishes are always legitimate, but customers are not necessarily the best prepared to express them.

Engineered systems are human-made artifacts that solve societal needs. They are characterized by four essential attributes. First, they are reasonably predictable; this means that the system works in foreseeable ways transforming given inputs into expected outputs or performance. Second, they are reliable; this means that they perform satisfactorily during a certain period of time (in the broadest sense of the term) in a given environment. Third, they are transparent; this means that the structure of the system (its integrating components and their inter-relationships) is known and can be stated explicitly. Fourth and final, they are controllable; this means that the system can be run or governed according to a set of defined instructions.

Systems have three elements: the elements that integrate the system, their attributes, and the relationships between these elements. Frequently systems are mistaken with the prime equipment. The solution to a need or opportunity is a system that is integrated by a prime equipment and by all its necessary logistics support elements. These elements ensure that the system maintains a certain degree of effectiveness throughout its operational life. Boundaries may not always be clear, as part of those elements of the enabling and logistics support structure may be shared with other systems. Let us consider for example the need to facilitate transportation of people between two cities. A potential solution is to connect both cities with a train service. The rolling stock (the train, as we normally would call it) is just the prime equipment; the entire system or solution requires other elements such tracks, stations, signaling systems, personnel and maintenance facilities, to name a few. Some of those elements (like the tracks, the stations or the maintenance facilities) may be shared with other systems (other trains). Therefore, systems are to be seen in a broad sense, thinking not only about their main or prime equipment but also about their associated enabling and logistics support elements. A solution may actually be comprised of an array of systems [Martin, 2004], of which the main ones are:

a) The intervention system. It is the intended solution to the problem.
b) The realization system. For the intervention system to be commissioned, some resources will be required. These resources will likely include people, facilities, services, materials, tools, processes, knowledge, and the like; all together they comprise the realization system.
c) The deployed system. It is the system as fielded and commissioned. Desirably, it will be the same as the intervention system, but it could be that the system has changed in its transition into operation. The changes may be in the form of configuration and/or performance degradation as a consequence of the interaction of the system with its environment.
d) The sustainment system. It provides the deployed system with all the necessary support to stay operational and to exhibit the level of effectiveness required by the customer.

Figure 1.2 shows the mentioned systems and their interrelationships.

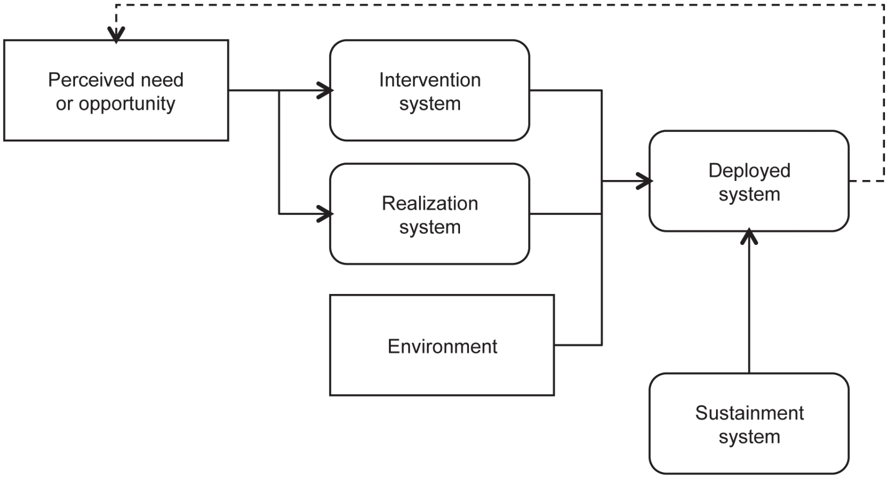

Figure 1.2. The intervention, realization, deployed and sustainment systems.

1.4. Systems Engineering: the Origins

Systems engineering can be defined, in a colloquial manner, as the structured approach to solving identified needs or business opportunities by designing systems that fulfill them. There are many definitions of systems engineering, although all have in common the transformation of the analysis of a need or opportunity into requirements, the holistic view, the integration of engineering and non-engineering disciplines, the consideration of the entire life cycle and the need for the system to effectively and efficiently fulfill its goals throughout. As defined by the International Council on Systems Engineering (INCOSE), systems engineering is an engineering discipline whose responsibility is to create and execute an interdisciplinary process to ensure that the customer and stakeholder's needs are satisfied in a high-quality, trustworthy, cost and schedule efficient manner throughout a system's entire life cycle. Systems engineering therefore bridges the traditional engineering disciplines as well as non-engineering disciplines, focusing on the system as a whole. Although the definitions vary, all have some common threads. Systems engineering is an interdiscipline with some special characteristics such as the global view, the top-down approach, the life-cycle perspective and the emphasis on the initial definition of requirements that define the problem or opportunity being tackled.

There is no single moment or milestone that marks the birth or foundation of systems engineering. A number of initiatives, mainly between the late forties and the late eighties, represented a substantial change in the way systems were conceived and dealt with. World War II gave a tremendous thrust to the development of technologies required to design and develop complex systems that would yield a military advantage to one side over the other. The development of systems such as the atomic bomb, the radar, the V1 and V2 missiles, advanced aircraft, life support devices and many others implied the need to combine and integrate multiple technical disciplines, with associated engineering challenges far beyond the ones faced so far. Furthermore, the compressed and demanding time schedules required new approaches in tasks identification and planning, coordination, risk assessment and program management. Systems engineering, as it is known today, was developed to meet these formidable challenges. In the wake of World War II other gigantic programs such as the Apollo were undertaken and a number of textbooks were published that identified systems engineering as a distinct discipline, quickly gaining recognition due to its capability for dealing effectively and efficiently with the design and development of complex systems.

Looking at the evolution of the texts on systems engineering from the first ones published in the late fifties, it can be noted that the vast majority addressed a common set of topics. These topics were usually related to the identification of user needs, to translating needs to requirements, to the trade study process, to system analysis methods, to integration and test, and to verification and validation. As systems engineering progressed as a discipline more topics were added in the new books that were published. These new topics included design for topics such as reliability, maintainability, human factors, manufacturability, supportability, life-cycle cost and system phase-out.

The following is a list of outstanding individuals (or institutions), by no means meant to be comprehensive, who contributed with their work and vision to the development of the discipline known today as systems engineering.

a) Bell Telephone Laboratories. In the 1940s Bell Laboratories was the first to use the term ‘systems engineering’. The study in 1945 of anti-aircraft guided missile systems was regarded as a milestone in systems engineering precisely because it was comprehensive enough to address a whole system. Systems engineering was born with analyses of ends and definition of objectives as integral parts of the engineering effort.

b) Norbert Wiener. A monthly series of discussion meetings on the scientific method organized by Arturo Rosenblueth (of the Harvard Medical School) and Norbert Wiener (of the Massachusetts Institute of Technology) ended up with the publication by the latter in 1948 of his seminal book *Cybernetics: or Control and Communication in the Animal and the Machine*. As he noted, science had become increasingly the task of specialists, getting deeper into fields that progressively grew narrower, strongly hindering communication among them [Wiener, 1965]. As stated by Wiener, *if a physiologist who knows no mathematics works together with a mathematician who knows no physiology, the one will be unable to state his problem in terms that the other can manipulate, and the second will be unable to put the answers in any form that the first can understand*. Communication among specialists was certainly needed. Furthermore, Wiener concluded that in any activity a very important form of communication was feedback that can be defined as the return of a portion of the output of a process or system to the input, especially when used to maintain or control the performance of the process or system. Being aware of the unity of the problems related with communication and control in the machine as well as in living tissue, Wiener and Rosenblueth decided to call cybernetics the field of control and communication theory. Cybernetics is derived from the Greek term κυβερνητικός (*kubernētikos*, good at steering) and was chosen to recognize the first significant paper on feedback, published in 1868 by Clerk Maxwell, who talked about governors, a Latin corruption for κυβερνητικός.
c) G. W. Gilman. Probably the first formal attempt to teach systems engineering was made in 1950 at the Massachusetts Institute of Technology by G. W. Gilman, then Director of Systems Engineering at Bell Laboratories, Inc.
d) Society for General Systems Research. This organization was founded in 1956 by biologist and philosopher von Bertalanffy, the economist Boulding, the mathematician and biologist Rapoport, the psychologist Miller and the anthropologist Mead. The founders of that society, currently called International Society for the System Sciences, felt strongly that the global or holistic approach was being consistently overlooked and neglected by the conventional disciplines, which stressed and emphasized excessive specialization and a reductionist approach to science. Among the major aims of the general systems theory were the integration of the various sciences

(natural and social), the development of unifying principles that run vertically through all the individual sciences, and a much-needed integration in scientific education.

e) Harry H. Goode and Robert E. Machol. These two University of Michigan educators and researchers published in 1957 *Systems Engineering: An Introduction to the Design of Large-Scale Systems*. Their book was one of the first authoritative texts in systems engineering, addressing its philosophy and methodology. As Goode and Machol then observed, engineers and administrators were witnessing the emergence of a broadening approach to the problem of designing equipment, a phenomenon that had been so far poorly understood and loosely described. That phenomenon had been called systems design, systems analysis, and even the systems approach. In 1964 Machol became the chairman of the newly formed Department of Systems Engineering at the University of Illinois.
f) Jay W. Forrester. MIT's professor Jay Forrester is the founder of the methodology of analysis known as systems dynamics, which deals with the interactions of the variables of a dynamic system. Forrester published in 1961 *Industrial Dynamics*, a method he developed to study the behavior of industrial systems to show that growth and stability are influenced by policies, decisions, structure and delays. The causal models built help explain the behavior of the system as a result of the interaction between its parts; frequently the interactions are more important than the parts taken separately. The methodology that Forrester developed was initially known as industrial dynamics, later called urban dynamics and finally called system dynamics. It advocates for the modeling of systems with a true global or holistic approach. In 1971 he published *Principles of Systems*, another landmark book.
g) University of Arizona. In 1961 the University of Arizona established the first department of systems engineering with graduate programs.
h) Arthur D. Hall. In 1962 he set forth in his book *A Methodology for Systems Engineering* his concept of systems engineering, encompassing some essential elements such as the awareness of the customer's needs, the consideration of systems engineering as a multi-faceted discipline, and the distinction of three different perspectives in the environment: the technical, the business or economic, and the social.
i) Harold Chestnut. He was an electrical engineer and helped establish the fields of control theory and systems engineering. He wrote in

1965 his textbook *Systems Engineering Tools* and in 1967 *Systems Engineering Methods*, as part of a series of books on systems engineering and analysis for which he acted as editor.

j) United States Air Force (USAF). USAF was the first organization to publish a comprehensive systems engineering document. In 1966, Handbook 375-5 *Systems Engineering Management Procedures* was published, describing in great detail a systems engineering framework. The handbook was later replaced with MIL-STD-499 *Systems Engineering Management*.

k) Ludvig von Bertalanffy. In 1969 the Austrian biologist and philosopher von Bertalanffy published the seminal book *General System Theory*. In it he noted that modern science was characterized by an ever-increasing specialization, which makes communication among specialists a true challenge. At the same time, he noted that similar problems and conceptions emerged in very different fields. He thus saw the need for a unified approach and general system theory was born as a doctrine of principles applying to all systems.

k) Russell L. Ackoff. He was a pioneer in the fields of operations research, systems thinking and management science. He has authored over 30 books, including *On Purposeful Systems: An Interdisciplinary Analysis of Individual and Social Behavior as a System of Purposeful Events* (together with Frederick E. Emery) in 1972 and *Redesigning the Future: Systems Approach to Societal Problems* in 1974.

m) Benjamin S. Blanchard and Wolter J. Fabrycky. These two professors at Virginia Tech stressed in the early eighties the need to design systems with a true life-cycle perspective, stating that the life cycle commenced with the identification of the need or opportunity and ended with the system phase-out. Blanchard and Fabrycky have chaired for many years a prestigious program in systems engineering at Virginia Tech and have co-authored a number of landmark books in systems engineering and related disciplines, among them *Systems Engineering and Analysis* (1981) and *Life-Cycle Cost and Economic Analysis* (1991).

n) INCOSE. The year 1990 was particularly noteworthy because it marked the formation of the first professional organization solely dedicated to systems engineering. A group of 30 individuals representing government, industry, and academia met in August 1990 in Seattle to discuss forming an organization with a systems engineering focus. The participants formed the National Council on Systems Engineering (NCOSE). Later and in recognition of increasing overseas membership, the organization was renamed International Council On Systems Engineering (INCOSE).

o) Andrew P. Sage. Sage has authored and/or co-authored a number of influential books in the field, including *Systems Engineering* (1992), *Introduction to Systems Engineering* (2000) and the *Handbook of Systems Engineering and Management* (2009).

1.5. Systems Engineering: the Concept

In essence, systems engineering is about solving needs or fulfilling opportunities in a broad sense. This means that not only the users or end customers are to be considered. For any identified need or opportunity there will be many other stakeholders to be taken into account, whose specific needs regarding the system to be designed and developed will have to be considered too. This is the global view that characterizes the systems approach. The current environment is characterized by a number of challenges, as depicted in figure 1.3. Stakeholders in general, and customers in particular, are becoming more and more demanding as they know better what they can ask for, although this does not mean that they can always formulate well the requirements of their perceived need. As the needs or opportunities increase in complexity, so do the systems engineered to fulfill them. Normally this comes with long acquisition cycles, which on the one hand facilitate the better understanding of the problem at stake, but on the other hand pose the drawback of allowing more time and latitude for the introduction of changes in the requirements. These changes can take a serious toll on the design and development effort, as they may conflict with the rest of the requirements and/or may imply a total or partial redesign of what has been done so far. New technologies are continuously being developed and progress at formidable paces, reaching rapidly higher levels of maturity. This means that many systems have elements or even subsystems that are obsolete, either from a functional or a market perspective, even before the design is frozen, the system is produced, or it is delivered to the customer or the market. To further complicate things there is an ever-increasing industrial competition, which forces companies to accelerate the formulation of requirements and the design process at the expense of incurring more risks. Even more challenging, supply chains are dynamic. Companies merge, disappear, discontinue production or support of certain products, and so on. This means that the needed supply support may not always be readily available and that obsolescence is, more frequently than not, the name of the game. Although all systems are designed with intended operational lives, it is common for users to keep them in service for longer-than-initially-planned periods of time, if the systems still show a reasonable effectiveness and/or if the users lack the financial resources to replace them. This results in extended operational lives that further complicate some of the previously indicated issues.

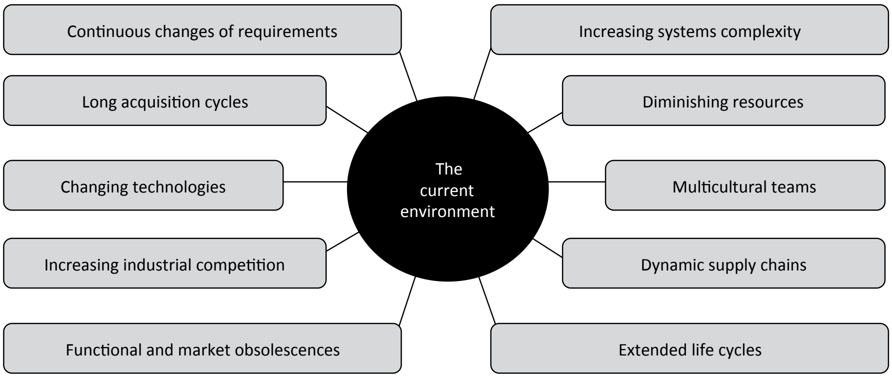

Figure 1.3. The challenges in the current environment.

An identified need or a perceived opportunity will trigger the launching of the initiative of designing and developing a system that will fulfill it. Such effort will normally be undertaken in a series of steps, each one called a project. Although there are differences in their definitions, it is widely accepted that a program is an initiative of large complexity and that requires its decomposition into individual, smaller parts, each one called a project. A project is an activity with a clear and defined goal, a beginning and an end. A portfolio is the set or collection of projects run and managed by a company or institution at a certain point in time, projects not necessarily being inter-related. Figure 1.4 shows the decomposition of a program into projects, run by different companies, each of them having their own projects portfolio.

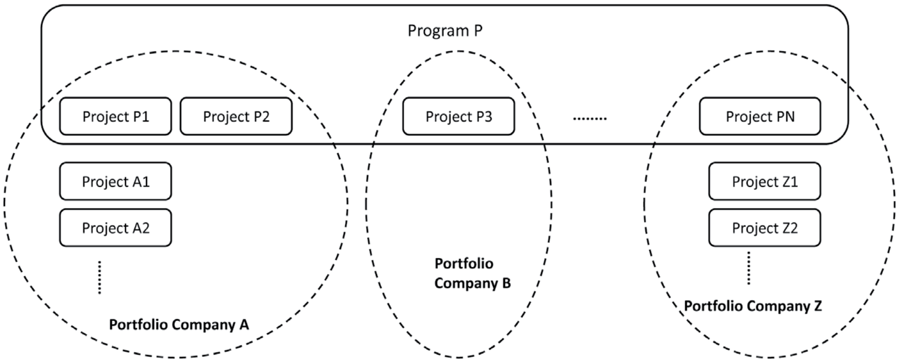

Figure 1.4. Programs, projects and portfolios.

Let us consider the example of the Eurofighter Program. In the 80s the world was still immersed in the Cold War. There was the Soviet Union, the Warsaw Pact, the Federal Republic of Germany and the Democratic Republic of Germany, the latter part of the communist block and hosting Mig-29s that could reach the main West-European cities within hours. That threat motivated four nations (the then Federal Republic of Germany, Italy, Spain and the United Kingdom) to join efforts in the design and development of a state-of-the-art fighter aircraft that would give their air forces superiority in the air. Such a gigantic program, involving four customers (the air forces of the four nations), two industrial consortia that gathered eight major industrial partners (the Eurofighter consortium, grouping one cell company per nation, and the Eurojet consortium, grouping one engine company per nation) and one NATO agency (NATO Eurofighter Management Agency, NEFMA), was split in many projects. The projects were linked to different life-cycle phases (like Production Investment, for example), to the design of different parts of the aircraft itself (like the Head-Up display, for example), or to different elements of the overall system to be designed and developed (like the Logistics Information Systems that customers and contractors had to build in accordance with standard AECMA 2000M), to name just a few.

Systems are designed to fulfil actual or perceived needs. The identification of a need, be it a functional or an operational deficiency, triggers the definition of the need in the form of stakeholder requirements, enabling the identification of potential design concepts, which results in the selection of a preferred solution. The definition of a selected design concept is accomplished through system requirements, and subsequent design and development of the system that satisfies those requirements effectively and efficiently throughout the intended in-service life. The design and development process can easily span a number of years. This may result in evolved or deprecated requirements by the time that the systems is commissioned and enters into service, and the system may no longer satisfy in full the current need. Even if the system fully satisfies the user needs at the time of deployment, the operational life of the system will inevitably result in a degradation of its performance. No matter the effort deployed in maintenance programs and even in reliability growth programs, eventually time and use (whether system has been in active use or even in a stand-by mode) take a toll on system performance. Long gone are the days in which failure rates were assumed to be constant; increasing failure rates imply, among other things, less operational availability and thus lower performance. Additionally, during the system life, the needs of the user may continue to evolve into more advanced and challenging requirements. Exceptions will arise, but it is a common occurrence that systems are replaced with new generations because of the growing capability gap, which eventually renders the system of insufficient value to its customer. This gap prompts the system replacement with a new and improved system that better meets the current need of the customer. The most obvious example of this is the personal computer industry. If the performance gap did not drive upgrades, most personal computer owners would still be using an 8086 based chipset. The delta between these two trends (additional requirements and degraded performance) represents a capability gap, as depicted in figure 1.5. The fact that some design and development efforts are long and can easily span over many months and even a number of years means that, by the time the system is fielded and commences its operational life, the needs of the customers will have evolved and now the system may no longer truly fulfil the customer's need. Many systems have elements or subassemblies that are obsolete even before the design of the entire system is frozen, given the pace at which technology changes, and by a similar token the evolution of the customer's need means in many instances that the fielded system already presents a capability gap from the moment it enters its operational life, as portrayed in the mentioned figure.

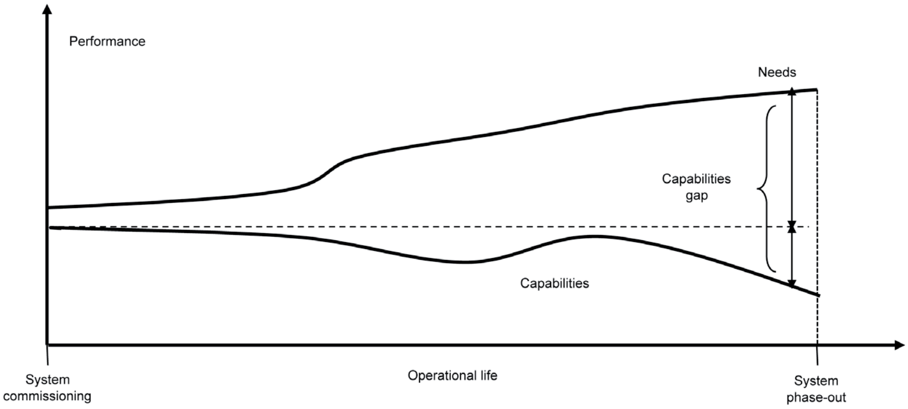

Figure 1.5. The capabilities gap during the operational life.

When systems are designed, it is possible that some requirements are initially fulfilled in excess. That may be done on purpose, anticipating a future increase in a need, if that excess in fulfilment is cheaper and/or technically easier to achieve initially than it would be later on. The additional capability exhibited regarding a given requirement is called margin. Should a requirement initially not be met it would be a deficit or gap and it would have to be accepted by the customer, although most likely prompting the necessary corrective actions. The figure above refers to the overall capability of a system, which is a combination of the extent to which all the requirements are met. This means that perhaps there is margin for a few requirements and yet the overall capability of the system is considered unsatisfactory to the end user, who thus experiences the capabilities gap.

Many complex systems in large industrial sectors are designed to have very long operational lives. To mention just a few, the transportation sector, and more specifically the rail sector, lay track with the expectation of it being there for decades; the same can be observed for the trains and cars that travel those tracks. Within the naval systems sector, it is common knowledge that planned operational lives of 40 years or even more is the norm, rather than the exception. Commercial and military aircraft remain in operation for many years, with some cases of extremely high longevities, such as the B-52 bombers that in 2005 completed 50 years of service, with an operational life foreseen to double that figure. Nuclear power plants are operated for many decades to amortize the huge investments required to build them. The list of sectors would be endless. It is not unusual that the scarcity of resources faced by most customers or institutions results in extending the already-long operational lives of the systems, as they lack the financial means to replace them. This extension in the operational lives further complicates the problem, and the capabilities gap increases, mainly due to the greater difficulty of sustaining systems in satisfactory operational condition as time elapses.

There are several reasons that explain the capabilities gap experienced by many systems users. One may be the inefficiencies in the management of the required logistics support, in spite of the higher attention paid to it and the significant improvements made in the last decades. If the logistics support resources required by the systems were perfectly managed, the system would retain throughout its useful life its performance characteristics as delivered the first day of operation. However, management is far from perfect and the inevitable consequence is a loss in system availability and degradation in the way in which many missions or operational profiles are accomplished. Moreover, all functional and market obsolescence problems that occur at any of the suppliers of the system integrator will eventually hit the system integrator and the end user, unless they are rapidly identified and satisfactorily managed. Another reason may be that new needs or requirements from users in general will not be satisfied by the system in the current configuration. The technological obsolescence suffered by the system will further enlarge the capabilities gap. The decomposition of the gap into its main drivers and the suggested mitigation strategies are depicted in figure 1.6.

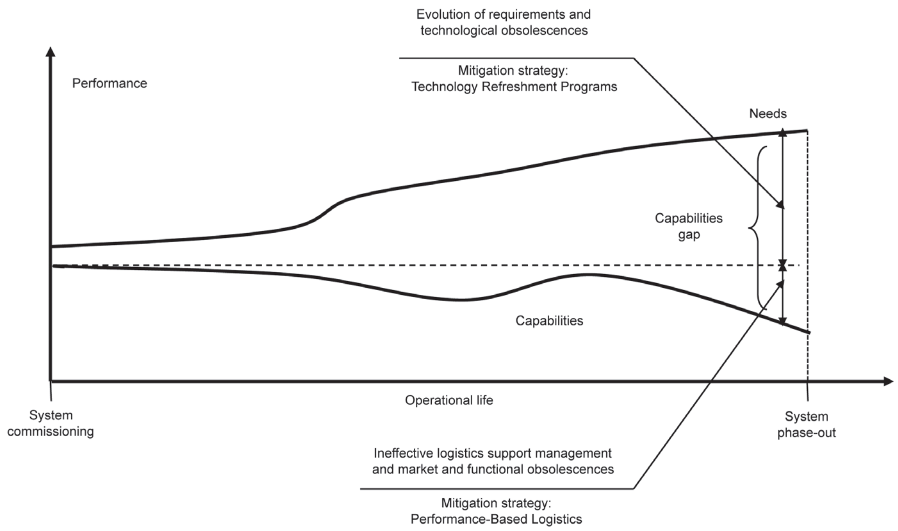

Figure 1.6. Reasons for the capabilities gap and corresponding mitigation strategies.

Aware of this gap, many users are starting to specify that the delivered systems must perform throughout the operational lives, in relative terms, as when they were first put into service. The likelihood of this becoming a reality will depend, to a large degree, on actions and decisions taken during the early phases of the life cycle, envisioning the need for measures that will eventually diminish this capabilities gap to the extent technically and economically feasible. The first step towards solving an unsatisfactory condition or situation is an appropriate diagnosis. Only after the root cause of a problem has been identified can the adequate corrective actions be defined and implemented. The 'lower' part of the performance capability gap, which represents the breach between actual performance and the one the system had when it entered into operation, is mainly due to the following reasons:

1) An ineffective and inefficient logistics support of the system. Although great progress has been made in the logistics area since the advent of the integrated logistics support philosophy, many systems still suffer operational deficiencies that stem from problems with the logistics support elements, whether lack of spares when and where needed, documentation of poor quality or that does not reflect actual system configuration, lack of skilled and trained users and/or maintainers, unreliable test equipment, facilities not duly equipped, and the like.
2) Market obsolescence. The supply chain of most system integrators are comprised of a huge number of companies and the operational lives of many systems are very long. More often than desired, those companies in the supply chain face situations such as mergers, decisions to abandon a line of production, planned obsolescence, bankruptcy, etc., which eventually imply lack of support resources that surface at the system integrator level, hurting the end customer. Market obsolescence may take the form of spares no longer supplied, repair actions no longer possible (or if done, at least at a much higher than usual turn-around-time and cost), lack of qualified personnel to process engineering change proposals in a timely manner, etc.

The 'upper' part of the performance capability gap, which represents the breach between the performance the system had when it entered into operation and the one that would be currently needed, is mainly due to the following reasons:

a) Technological obsolescence. While some elements or equipment continue to function (and can still be supported), the advent of new technologies may render them functionally obsolete. One example is the storage device for computer systems. In the 80's tapes were widely used to store data, to be replaced in the 90's by external hard drives. In less than 15 years we have gone from an expensive, external hard disk of 1 megabyte of the size of a washing machine, to a flash drive with no moving parts which fits in a small pocket and can store data in the order of many gigabytes. The new capability results in far less weight, far less volume, far more reliability, and at far less cost. Continuing to use certain devices or technologies, although still operational and reasonably supportable, would render the system functionally obsolete in relative terms.

b) New requirements. The long design and development cycles usually implies that by the time the system is commissioned and enters into service, the need that drove the systems engineering framework has evolved. Even if, when commissioned, the system still met the primary requirements and fully satisfied the initial need, over the years the need will continue to evolve. Continuing the example from above, while the initial capability may have been to store single frame pictures, the new requirement is for storing video. The old storage device is not capable of storing that much data, but modern storage devices can satisfy the new requirement using less volume while providing higher reliability and at a lower cost. Thus, to the extent that the system stays as initially designed, those new requirements into which the evolving need materializes will not be met and satisfied, the user will thus suffer a performance capability lack.

Different ailments call for different remedies. After identifying the potential causes of the undesired performance capability gap experienced by many systems throughout their operational lives, remedial actions can be put into action. Even better, knowledge of the nature of these potential situations allows for actions in the early stages of the life cycle. There are two main mitigation strategies that help users cope effectively and efficiently with the mentioned performance capability gap, reducing it throughout the system operational life to acceptable levels. These two strategies are performance-based logistics (PBL) and technology refreshment programs (TRP), both of which will be addressed in depth in chapter 8.

Systems are designed and developed to deliver to the users the capabilities they need during their operational lives. To sustain a system in operation a number of logistics support resources are required. Logistics is as old as mankind. Since the age of the hunter-gatherers, human beings have manufactured and maintained tools and artifacts with which to attain their goals, like collecting edible plants and fruits, and hunting animals. The life style of hunter-gatherers conveyed also some kind of transport and storage of food and belongings. The term logistics comes from the Greek *logistikos,* used to denote someone skilled in calculations. The problem with the old approach to logistics is that artifacts and equipment were, till relatively recently, first produced and only after production the logistics issues were considered, as best as they could. The advent of the systems approach in the middle of the 20th century rapidly permeated all fields and disciplines. In particular, the consideration in the logistics field of the global or holistic view that characterizes the systems approach meant the coining of the philosophy known as Integrated Logistics Support (ILS). As defined by [DSMC, 1986], ILS is the disciplined, unified and iterative approach to the technical and management activities necessary to: (1) integrate support considerations in the design of the system; (2) develop support requirements that are derived from the performance goals of the system; (3) acquire the necessary logistics support elements; and (4) provide the required support throughout the useful life, at the lowest possible cost. This means that when addressing a need or opportunity it is mandatory to analyze it also from the logistics perspective, so logistics considerations and requirements are an integral part of the design and development effort from the very beginning. When the need or opportunity is also translated into the right set of logistics support requirements, the design of the system and of the associated and needed logistic support elements will be optimized, increasing the likelihood of customer satisfaction with the delivered solution throughout the entire operational life of the system. Logistics plays a most relevant role in the systems engineering approach [Blanchard, 1974; Jones, 1987; Green 1991; Langford, 1995]. The logistics support elements are portrayed in figure 1.7.

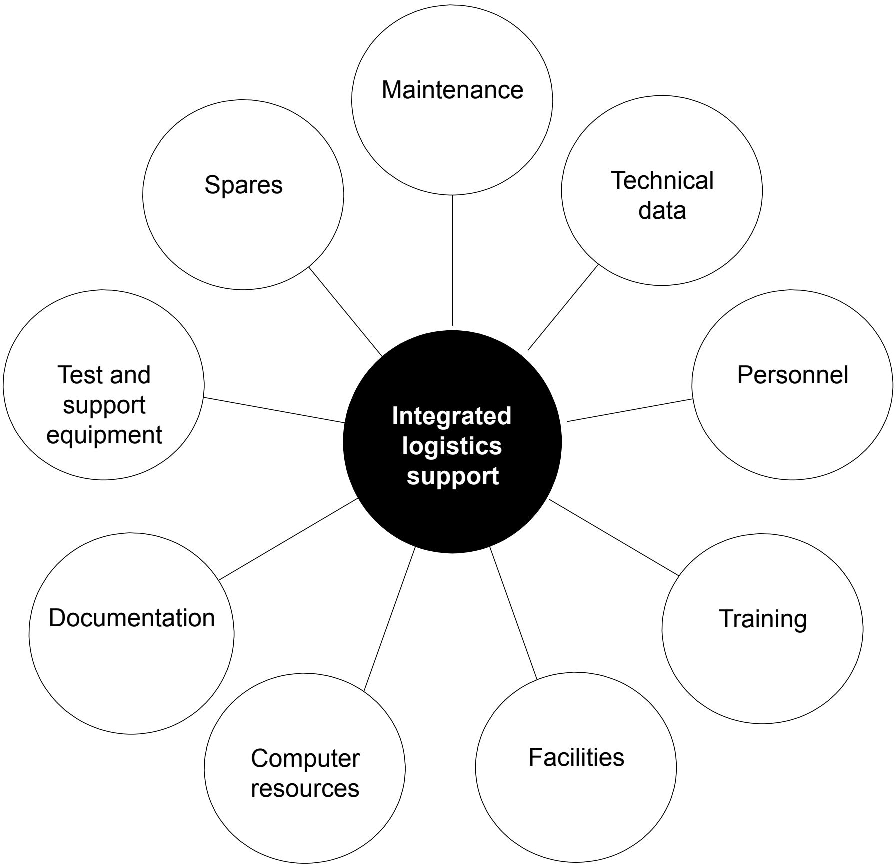

Figure 1.7. The logistics support elements.

The Italian engineer and economist Vilfredo Pareto gained recognition by stating his famous 20/80 rule. When referring to the distribution of wealth he stated that 80 percent of it was in the hands of 20 percent of the population. That idea of complementary percentages has been applied to many other situations and environments: 20 percent of the requirements representing 80 percent of the budget in a project, 20 percent of the tasks in a project implying 80 percent of the risks, and so on. In the book *The Fermi Solution - Essays on Science* [von Baeyer, 1993] the importance of discriminating in any problem or decision-making process the significant or relevant few aspects or criteria is highlighted. The systems engineer has to reconcile the global view with the capability of telling which are, out of the many, the vital few. This can apply to identifying the few stakeholders that are most significant, or the few requirements that are truly decisive in terms of the technological challenge that they pose or the cost implications that they will have. The strength of the systems approach is with this balanced capability of seeing the whole picture while being capable of discriminating those few, relevant aspects that make the difference.

1.6. Systems Engineering: the Framework

As defined in The Free Dictionary, a process is a series of actions, changes or functions or steps bringing about a result; a framework is a set of assumptions, concepts, values and practices that constitute a way of viewing reality. A process conveys the idea of linearity in the execution of a series of steps, somehow like a recipe that offers little latitude or flexibility for its implementation. On the other hand, a framework is much more agile, flexible and versatile, as it offers a solid foundation while leaving the practitioner room to adapt it to the circumstances and to the nature of the problem at stake. That is why it is called the systems engineering framework what allows the transformation of identified needs or opportunities into solutions, and not the systems engineering process. As a framework, it reflects the iteration that will be present in it (thus, not a linear and fixed sequence of steps) as well as the fact that it is based on some assumptions and basic concepts and allows the systems engineer some degree of freedom in this customization and implementation, always seeking the satisfaction of a perceived need or opportunity in the most effective and efficient manner. The systems engineering framework is triggered by the identification of a need or by the perception of a market opportunity, and ends when the resulting system is phased out. Between that beginning and that end, a number of steps are taken, with feedback among them as appropriate. The framework is depicted in figure 1.8 and will be described in detail in chapter 3. Once the problem has been formulated, the next step is the identification of the stakeholders, greatly assisted through the performance of the so-called Concept of Operations (CONOPS). Requirements are then elicited from the stakeholders. The set of requirements has to be validated, that is, it is necessary to confirm that they are a complete and correct translation of the identified problem, without any prejudice or bias as to what the potential solution could be. After that hurdle has been cleared the next step is the identification of as many design concepts as possible, of ways to deliver the needed functionalities; out of the many choices, the preferred design concept is chosen. Till this moment, activities belong in the problem domain and from now on they will pertain to the solution domain; an essential characteristic of the systems approach is the distinction between the problem domain and the solution domain, which are separate and do not overlap, as portrayed in figure 1.9. For the selected design concept the stakeholder requirements will be translated into system requirements, together with their corresponding verification methods. Those requirements will lead, through the performance of the functional analysis, to the definition of the system architecture and to the identification of the characteristics of the system elements. After preliminary verification at the end of the design phase that the system requirements have been met, the next step is the production of the system. The availability of the physical system enables a more accurate verification of the requirements and also the validation of the solution, in this case through the performance of the system acceptance tests. The system will then be commissioned and will enter its operational life during which logistics support activities will be carried out (including outsourcing under the performance-based logistics schema, in which results are purchased and not resources) and reliability growth and technology refreshment programs will be conducted. Unless the system is consumed or destroyed with its use (like a missile, or a rocket that launches satellites), or if it simply can not be retrieved (like a space probe sent to collect data on parts of the universe), at the end of its operational life the system will have to be phased out.

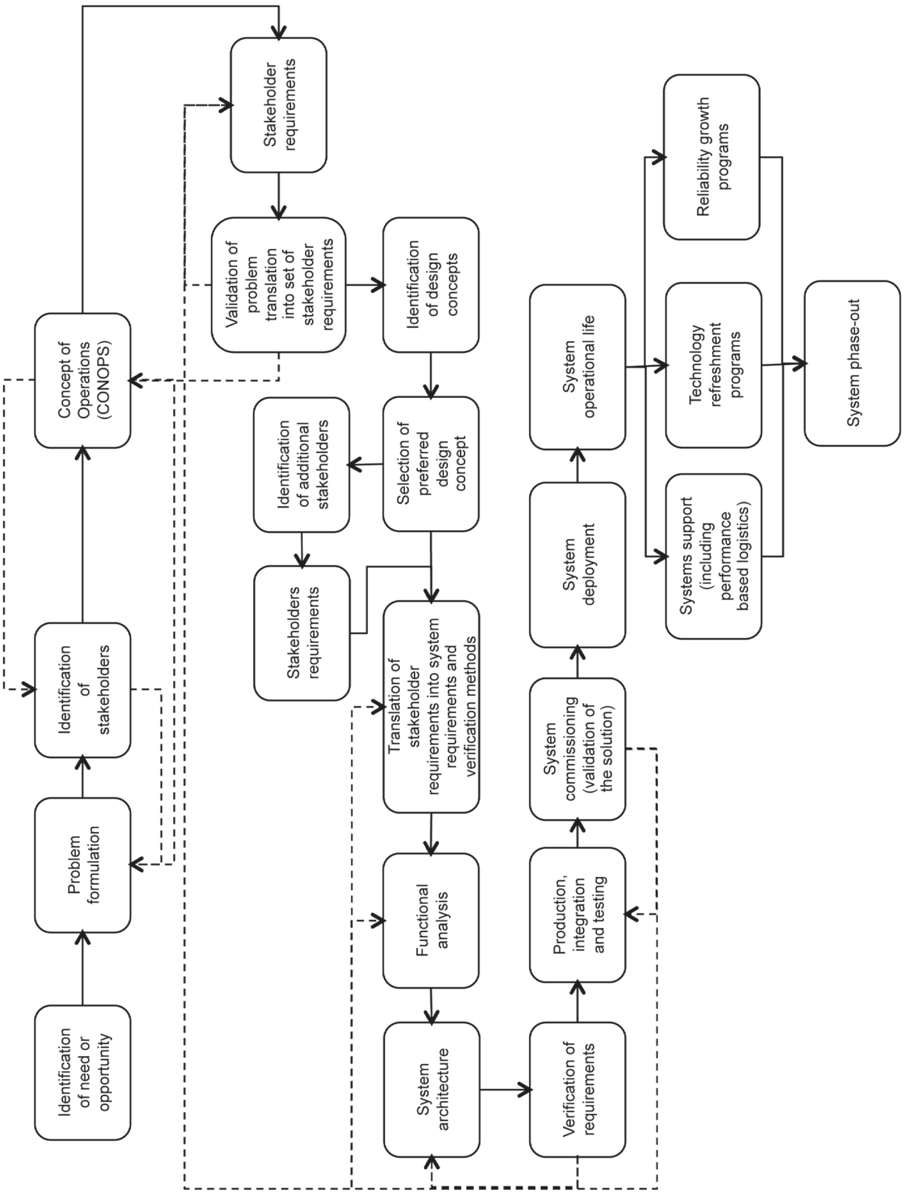

Figure 1.8. The systems engineering framework.

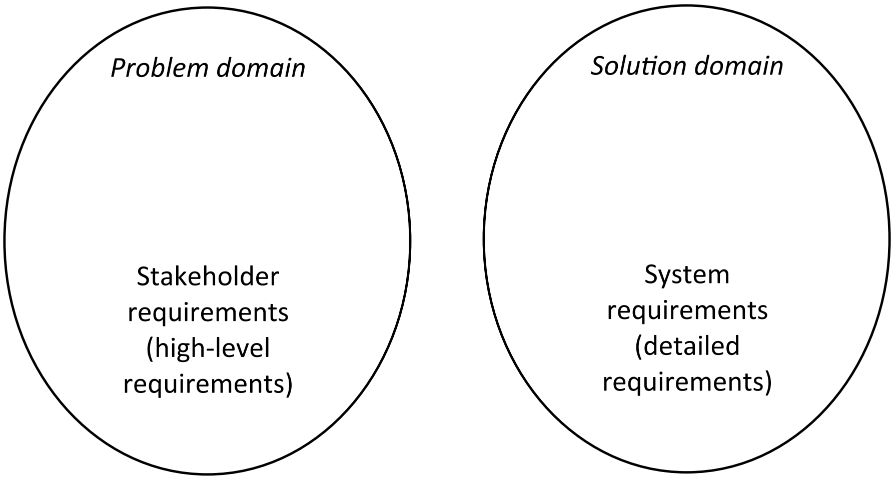

Figure 1.9. The problem domain and the solution domain.

It is important to stress that it is the systems engineer who has to own the systems engineering framework, and not the other way around. Following a framework blindly and strictly by the book, without adapting and tailoring it as needed in each particular case, only reflects lack of true understanding and command. The systems engineer has to have profound knowledge of the framework, knowing when and exactly how to use it so that it delivers its full potential. A framework is not just a series of steps always to be carried out in exactly the same manner. The global view that the systems engineer must have begins with a deep understanding of the framework as a whole, because it is a tool and not an end in itself. Knowing what has to be done and why is understanding the framework; being capable of customizing its application in light of the specific characteristics of each project. This reflects the real command of the subject and that is what the systems engineer has to strive for.

1.7. Systems Engineering: the System of Interest

The systems engineering framework depicted and succinctly described in the previous section applies not only to the entire system that is developed to meet the identified need or opportunity, it can be applied to any part of it. The part to which the framework is applied is known as the system of interest. The application of the framework is therefore done iteratively, starting at the entire system level and repeating it as many times as necessary for the design of the different parts of the system (subsystems, assemblies, and so on). Let us consider the need for establishing a link between two cities lying at opposite sides of a strait so that their inhabitants can cross from one side to the other. One of the many potential design concepts or solutions is a ferry, as other solutions could be a bridge or a tunnel, to identify a few. Assuming that the ferry has been selected, at some stage in the design process it will be necessary to decide the architecture of its propulsion plant, for which many alternatives exist. The propulsion could be achieved through a diesel engine coupled to a propeller through a reduction gear and a shaft, or through electrical pods, or by using, for example, the so-called Voith-Schneider propellers. The systems engineering framework would then be applied to solve the problem of identifying the most appropriate propulsion plant for the ferry, the propulsion plant being the system of interest. If what is selected is the diesel engine coupled to a propeller, the next step could be the design of the propeller itself; it could have more or fewer blades, which could be fixed or have controllable pitch. In this case the system of interest would be the propeller. This concept of the system of interest is of extreme importance because the systems engineering framework is not only applied at top level and for the design of the entire system. The framework can and should be applied to whatever part of the problem has to be solved, addressing such part as the system of interest. By focusing on the right part of the system at every stage of the design and development process, the idea of system of interest acts as an engine for the design of the complete system. The concept of system of interest and its role as the engine of the systems engineering framework is described in detail in [NASA, 2007]. Figure 1.10 shows, for the mentioned example, what the system of interest would be in each of the stages of the design of the ferry.

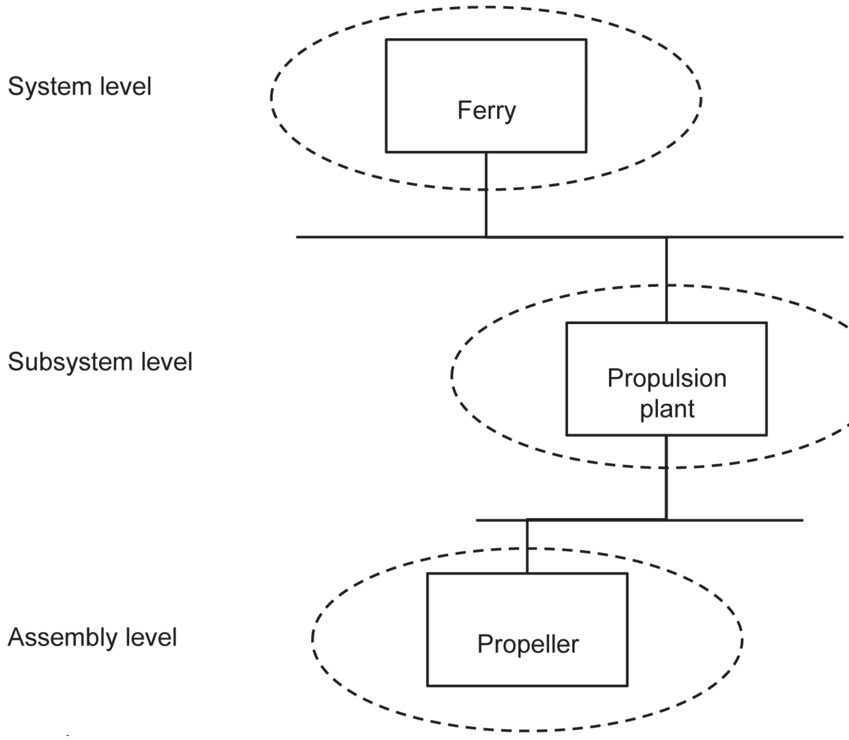

Figure 1.10. The system of interest at different levels in a project.

The identified need or perceived opportunity is expressed and defined through the corresponding stakeholder requirements, later translated into system requirements once a design concept has been chosen. Those system requirements are initially formulated at system level and, as the functional analysis transform requirements into system architecture, the system-level requirements will be allocated to the lower levels of the system, starting with the subsystems, proceeding then to assemblies, subassemblies, and so on. At each stage in the framework the system of interest receives a set of requirements from the higher level (the stakeholders, at top system level) that allows the corresponding design. The requirements are allocated and flow down the structure of the system in a top-down manner, while detailed design of elements and their integration is performed bottom-up. The inputs and outputs of the system of interest, from the higher level and to the lower one respectively, are depicted in figure 1.11.

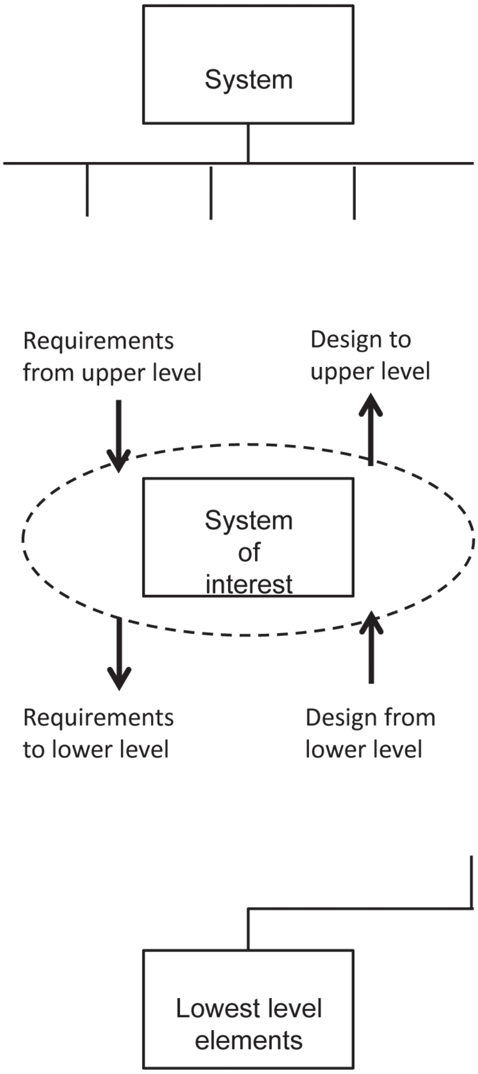

Figure 1.11. Inputs to, and outputs from, the system of interest.

1.8. Systems Engineering: Soft Problems

This book describes the systems engineering framework as it is applied to deal with problems or opportunities that can be defined well and in an objective manner. These are the so-called hard problems. In many instances the perceived problem or opportunity is such that it is next to impossible to express it in the form of a set of requirements, both because the rapidly changing nature of the need or opportunity and because of the difficulty in stating objectively the goals to be achieved. These are the so-called soft problems. A derivative of the systems method, known as soft systems methodology, was developed to deal with these cases [Checkland, 1990; Wilson, 2001]. These references provide ample details on the methodology and its application. Nevertheless, it is important that the systems engineer is aware of the existence of these two types of problems, although in reality problems cannot be classified in a binary way, as if they were either hard or soft. Problems may be anywhere in the spectrum between those two extremes. If the 'soft' content of a project is relatively low, it will then be treated as a 'hard' case and the methodology presented in this book will apply. To the extent that the soft part increases, it will be necessary to exercise special caution in the definition of the project goal and scope, revisiting it as necessary and always being aware that the designed solution may rapidly loose effectiveness due to the inevitable changes in the nature of the problem.

1.9. Systems Engineering: the Human Dimension

This book is about the systems approach; it describes the systems engineering framework. Nevertheless, it is essential to keep in mind that the framework is a tool that will be applied by human beings and therefore it is appropriate to state the desirable characteristics or traits of a systems engineer. Any framework is only as strong as its weakest link and with the global view that characterizes the systems approach, it can be considered that the systems engineer is one more link or element, part of the overall systems engineering framework. The systems engineer should not be the weakest link in it, but the polar opposite; the systems engineer is to have a profound understanding of the framework in order to properly apply it in the most diligent manner.

It could be said that a good systems engineer has to excel in everything, but that would be too easy to say and, in a way, not too helpful, as that would simply be impossible. It is necessary to identify and distinguish the abilities or skills that a professional has to have to be considered a true systems engineer. These are the main traits of a good systems engineer, synthesized in figure 1.12 (showing the most important one in the center):

a) Excellent communication skills. Above all, systems are designed by human beings and for human beings. That means that the systems engineer has to communicate with other people, whether they are the customers, other stakeholders, team members, experts in different fields, or any other individuals with whom he has to interact. The first and foremost skill that a systems engineer is to have is outstanding communication skills, which means both receiving and transmitting. The former is the capacity to listen and understand, whereas the latter is the capacity to explain things in a way that the other party can understand. If one is unable to understand what is the functionality desired by a customer, no amount of technical expertise will be of any help. Without the understanding of the nature of the problem, no solution that is developed will be truly effective. Two-way communication is thus essential and it is a skill that, like any other one, can be developed through practice. Many professionals invest time and effort into developing their technical skills and competences, while ignoring their communication skills.
b) Capable of having a global view and of simultaneously discriminating the vital few. One of the cornerstones of the systems approach is the global or holistic view. Yet, it should not be said that all aspects are always equally important. In any situation it is desirable to have a global view and, at the same time, to be able to distinguish what are the specific factors or elements that stand above the rest and tell the difference. Otherwise the trees may prevent the systems engineer from seeing the forest. These two apparently polar opposite qualities can actually be held simultaneously and are extremely powerful when combined. If the capability of seeing the big picture denotes the maturity and broad vision of a systems engineer, being able to discriminate at the same time the few aspects or elements that stand above the rest shows an even deeper capability; identifying the vital few aspects out of the big picture reflects the deep understanding that a systems engineer has of the problem at stake.
c) Open-minded and with relentless curiosity. A true systems engineer does not believe that he is in possession of the ultimate truth; he is always open to listening to others and to considering new perspectives, new possibilities and new alternatives. As important as being confident is being humble, good ideas emerge from having many ideas and the best ones may easily come from others. A systems engineer is always keen to explore new ways of doing things, not necessarily replicating old solutions; that is the true driver of innovation. 'What else?' is the most important question that the systems engineer can ask himself.

The eagerness to explore, understand and learn is what drives and motivates a systems engineer.

d) Comfortable with change and uncertainty. Change is the only sure thing to happen; it is inevitable. Moreover, uncertainty is always part of the picture, one way or another. Therefore the systems engineer has to know how to live and deal with change and uncertainty, without feeling uncomfortable or stressed.

e) Combination of a broad academic education with ample industry experience. A good and solid theoretical foundation is indeed a must and it should be complemented by expertise in at least one field. It facilitates communication with other specialists and the coordination of their efforts and contributions towards a common goal. The academic foundation is to be further complemented with abundant and broad industry experience, as putting things into practice allows to consolidate knowledge from a theoretical perspective and reinforces it. The graduation ceremony at universities is frequently called Commencement to reflect that it is at that moment that the real learning process begins. Learning is a race without a finish line; a systems engineer is permanently learning through continued practice, reflection, formal courses, publications, and so on.

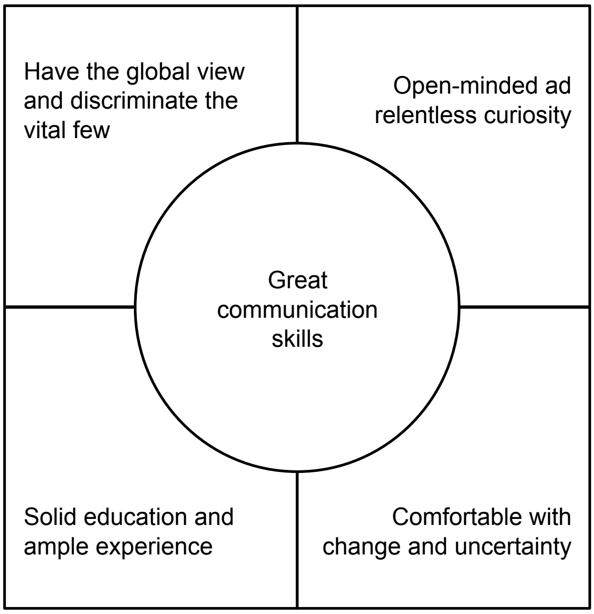

Figure 1.12. The most desirable traits of a true systems engineer.

1.10. Systems Engineering: Applied in Companies and Institutions

This book describes the essence of systems engineering. It reviews its origin, addresses its elements, summarizes the characteristics of the systems approach and describes the steps and elements in the systems engineering framework. Yet, applying systems engineering in a given company or institution will require certain customization depending on its culture and on the nature of the projects undertaken. The purpose of this book is to describe the systems engineering framework so that it can be of assistance to practitioners. Moreover, it will hopefully be a guide for companies or institutions that wish to develop specific procedures for the application of systems engineering. It is not the purpose of this book to address the cultural challenges that will be faced. For example, it is not the same to think of a large company such as Airbus designing and developing a system, like the A-350 aircraft, out of own initiative, as to think of the same company designing and developing another system like the A-400M military transport aircraft under a contract with a pool of customers, or to think of any of the smaller companies that will be part of its huge supply chain in any of those projects. The essence of systems engineering is not altered based on company or project size, yet the way it is actually implemented will have to reflect the peculiarities of both. What is essential in such customization process is not to lose sight of what the end goal is and what the differential traits of the systems approach are. Companies and institutions end up developing their own set of procedures for the implementation of the systems approach; those procedures typically cover the areas of requirements engineering and management, design, life-cycle costing, integrated logistics support, and so on.

REFERENCES

Blanchard, B. S. *Logistics Engineering and Management*. Prentice Hall, Inc., 1974.

Baeyer, H. C. von *The Fermi Solution - Essays on Science*. Courier Dover Publications, 1993.

Bryson, B. *A Short History of Nearly Everything*. Broadway Books, 2004.

Carey, S. *A Beginner's Guide to Scientific Method.* Cengage Learning, 2010.

Checkland, P.; Scholes, J. *Soft Systems Methodology in Action*. John Wiley & Sons, Ltd., 1990.

Checkland, P. *Systems Thinking, Systems Practice*. John Wiley & Sons Ltd., 1993.

DOREN, C. van *History of Knowledge*. Carol Publishing Group, 1991.

Defense Systems Management College. *Integrated Logistics Support Guide*. Defense Systems Management College, 1986.

JONES, J. V. *Integrated Logistics Support Handbook*. TAB Books, Inc., 1987.

LANGFORD, J. W. *Logistics - Principles and Applications*. McGraw Hill, Inc., 1995.

LEONARD, B. *The Landmarks of Science*. Facts on File, 1989.

GAUCH, H. G. Jr. *Scientific Method in Brief*. Cambridge University Press, 2012.

GREEN, L. L. *Logistics Engineering*. John Wiley & Sons, Inc., 1991.

GRIBBIN, J. *Science. A History*, *1543-2001*. Allen Lane, 2002.

MARTIN, J. N. *The Seven Samurai of Systems Engineering - Dealing with the Complexity of Seven Interrelated Systems*. Presented at the 2004 Symposium of the International Council on Systems Engineering (INCOSE).

MCCLELLAN, J. E.; DORN, H. *Science and Technology in World History: an Introduction*. The Johns Hopkins University Press, 1999.

PRIESTELY, J. *History of Science*. Kindle, 2014.

NASA/SP-2007-6105 Rev 1. *NASA Systems Engineering Handbook*. NASA, December 2007.

WIENER, N. *Cybernetics: or Control and Communication in the Animal and the Machine*. The MIT Press, 1965.

WILSON, B. *Soft Systems Methodology: Conceptual Model Building and its Contribution*. Wiley, 2001.