Guidelines for creating systems

Joseph Kasser

Temasek Defence Systems Institute

National University of Singapore

Block E1, #05-05

1 Engineering Drive 2, Singapore 117576
tdskj@nus.edu.sg

Abstract. A review of the INCOSE literature on systems engineering has shown that publications in systems engineering especially in systems engineering education tend to assume the system exists and then go on from there. Hence there is a need for guidance on conceptualizing and creating the system. This paper, written for both educators and practitioners provides some of that guidance by introducing and describing an iterative holistic process for conceptualizing and creating a system and/or situation, and determining which elements of the situation belong inside the system boundary and which elements remain outside, namely the paper describes the S2 process in the Nine-System Model (Kasser and Zhao, 2014).

Keywords. Systems, Nine-System Model, creating systems, complexity

1. Introduction

The common denominator in 22 listed definitions of a system (Kasser, 2013) pages 178 - 179), short for system of interest (SOI) or 'Sysrep'[1] (Kline, 1995) shown in Figure 1 represent a part of a situation where anything:

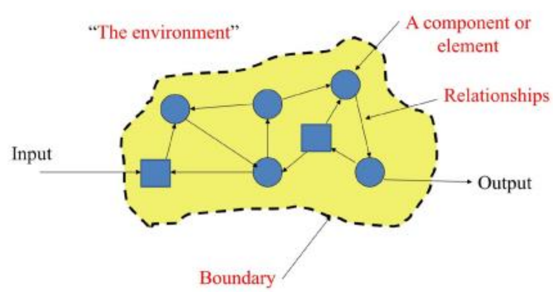

Figure 1 The elements of a system (Flood and Jackson, 1991)

- ***Inside the boundary*** is a part of the SOI and is partitioned into subsystems or components which may be people, technology, processes, doctrine, etc.
- ***Outside the boundary*** comprises the context, metasystem or environment and although not shown in the figure is also partitioned into adjacent systems.

The words 'the environment' in Figure 1 abstracts or masks out the complexity in the environment to allow the system engineer to focus on the SOI, The environment and the SOI together constitute a situation, the metasystem commonly known as the context.

A common system myth is that the universe is made up of systems such as physical, mechanical, natural, biological and socio-economic. The reality is that the observer creates the boundaries of the SOI (Beer, 1994; Churchman, 1979) page 91), as well as the classifications of systems such as physical, mechanical, natural, biological and socio-economic. It is the act of drawing the boundary that creates the SOI. Moreover if a prior observer has created a SOI

[1] Kline's term never went into general use

that is appropriate to the specific undesirable situation, the observer tends to use a version of the created SOI with or without an appropriate degree of tailoring.

This paper discusses the magic that happens to create the SOI and the adjacent systems in the situation; namely the S2 process in the functional view of the Nine-System Model (Kasser and Zhao, 2014) shown in Figure 2[2]. The S2 process contains the activities that:

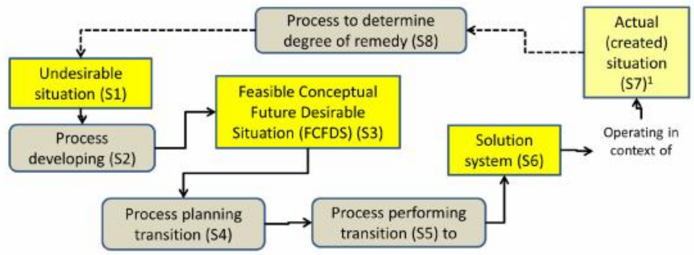

Figure 2 The Nine-System Model (*Functional* perspective)

1. Examine the undesirable situation (S1).
2. Conceptualize the Feasible Conceptual Future Desirable Situation (FCFDS) (S3). The FCFDS will eventually become the metasystem, situation or environment (S7) that will contain the SOI (S6) and its adjacent systems once each has been created by their own separate individual system development process (S5) in accordance with the specific plans produced by the planning process (S4).

2. The process for creating a system

A literature search only found the following two approaches for creating systems.

1. Athey's systemic systems approach (Athey, 1982) page 13).
2. O'Connor and McDermott's set of guidelines (O'Connor and McDermott, 1997).

2.1. Athey's systemic systems approach

Thomas Athey drew the boundary of a system such that:

- The set of components which can be directly influenced or controlled in a system design are included in the system.
- The factors which have an influence on the effectiveness of the system, but which are not controllable, are part of the environment, namely are outside the system.

2.2. O'Connor and McDermott's set of guidelines

O'Connor and McDermott introduced the following set of guidelines for drawing systems (O'Connor and McDermott, 1997) page 166):

1. Draw from your experience and viewpoint.
2. Draw with a goal in mind.
3. Start wherever you want.
4. Include events.
5. Define system boundaries.
6. Include time span and people involved.
7. Only include elements that can change when influenced by another element.

Unfortunately, while O'Connor and McDermott's guidelines are interesting and useful, they can lead to unnecessary complexity and errors in the creation of the system. For example:

[2] S9 is the organizations in which the processes take place and as such S9 does not show up in (is abstracted out of) the functional view.

- "*1. Draw from your experience and viewpoint*" ignores the wealth of experience offered by others and leads to the Not Invented Here (NIH) syndrome.
- "*7. Only include elements that can change when influenced by another element*" ignores elements that influence the system but do not change. For example a closed systems view of the pendulum clock ignores the effect of gravity because while it is there, it remains constant in a specific location, and may be ignored. However, if the clock is moved into a different gravitational field, the mass on the end of the pendulum will need to be adjusted or replaced to compensate. As a possible second example, the Lunar Surface Gravimeter experiment flown to the Moon in the Apollo 17 mission did not perform as expected on the Moon (Giganti, et al., 1971) and may have suffered from the lack of compensation for the difference between Terrestrial and Lunar gravity.

3. The Nine-System Model approach for creating a system

This paper suggests a process for creating systems based on the problem-solving paradigm with reference to the Nine-System Model (Kasser and Zhao, 2014). The process for creating a SOI contains the following activities:

1. Examine the undesirable situation (S1) from several different perspectives.
2. Develop an understanding of the situation (S1).
3. Create the FCFDS containing the SOI (S3).
4. Use the principle of hierarchies.
5. Abstract out the parts of the situation (S1 and S3) that are not pertinent to the problem.
6. Partition the FCFDS (S3) into the SOI (S6) and adjacent systems.
7. Optimize the interfaces.
8. Partition the SOI into subsystems.

Note:

1. The activities should be performed in an iterative sequential parallel manner not in a sequential manner.
2. The FCFDS (S3) will evolve to the actual or created situation (S7) during the time taken to plan the system development process (S4) as well as the time taken to perform the system development process (S5).

Consider each of these activities as follows.

3.1. Examine the undesirable situation from several perspectives

Traditional systems enquiry creates dynamic views of the behaviour of a SOI using tools such as causal loops (Senge, 1990), system dynamics (Clark, 1998; Wolstenholme, 1990), queueing theory, linear programming and other tools used in Operations Research. Other approaches include building models or applying sets of equations suitable to the class of situation. However, while modelling the behaviour of a SOI does provide a wealth of information, using this single behavioural perspective does not provide a full understanding of the SOI and may even lead to a misunderstanding, identification of the wrong cause of the undesirability and a definition of the wrong problem. Thus use of these traditional tools must be considered as only a part of the process of examining the situation to gain an understanding of the situation.

3.1.1. The need for multiple perspectives

The concept that a single perspective may lead to errors in understanding what is being viewed has been known for centuries if not longer and is best illustrated by the parable of the blind men perceiving a part of an elephant and inferring what animal they are perceiving (Yen, 2008). Since each man perceives a different part of the elephant, they each infer that they perceive a different animal. It takes a combination of the perceptions to understand the nature of the animal being felt.

The concept of using multiple views and models of a system has long been known in systems and software engineering, and several approaches have been introduced, including:

- The models used in Paul Ward and Stephen Mellor's version of structured systems analysis (Ward and Mellor, 1985).
- The models used by Derek Hately and Imtiaz Pirbhai in specifying, respectively, the requirements for and the design structure of software-based systems which grew up around real-time embedded systems (Hately and Pirbhai, 1987).
- The views in the United States Department of Defense Architecture Framework (DoDAF, 2004).
- The Holistic Thinking Perspectives (HTP) (Kasser, 2013) pages 90-110).

One holistic approach to implementing the process for examining the situation from several perspectives is to use Active Brainstorming which poses the questions 'who', 'what', 'where', 'when', 'why' and 'how' (Kipling, 1912) from the HTPs (Kasser, 2013) page 150).

3.1.2. The Holistic Thinking Perspectives

"*Systems thinking looks at the whole, and the parts, and the connections between the parts, studying the whole in order to understand the parts*" (O'Connor and McDermott, 1997) page 2). The HTPs combine systems thinking and critical thinking in a systemic and systematic manner to provide an understanding of the situation, and also go beyond systems thinking to infer the nature of the solution system that should remedy the undesirable situation. For example, assuming an undesirable or problematic situation in which the government of a nation feels that they need to upgrade their Air Defence System (ADS), the situation can be perceived from the HTPs as follows:

- ***The Big Picture perspective*** or helicopter view provides a *static* perspective of the situation. In many instances, the system and its adjacent systems may be predefined for the analyst by virtue of the history that lead up to the undesirable situation. For example, if the Government already had a Department of Defence, the ADS is then a part (subsystem) of the National Defence Force (NDF). The situation can be aggregated into the ADS and its adjacent systems.
- ***The Operational perspective*** is an external, open system or 'black box' perspective which hides or abstracts out everything inside the system and provides a *dynamic* perspective of:
    - What the system does (in the big picture), namely providing the air defence for the nation in conjunction with the adjacent systems.
    - The missions/operations performed by the system, expressed in scenarios or use cases. These mission and support scenarios might show how the ADS responds to different types of threats, e.g., intruder detection (manned and unmanned aerial vehicles) and how preventative maintenance is performed.
    - The desired and undesired inputs and outputs, e.g., trained pilots, ordnance, and spares.
    - The interactions with the adjacent systems. One example might be a systems dynamics view of the rate of replenishment of ordnance and spare parts during the different scenarios.

Table 1 Scenario to Function mapping

	Scenario A	Scenario B	Scenario C	Scenario D	Scenario E
Function 1	Used	Not Used	Used	Not Used	Not Used
Function 2	Used	Used	Not Used	Used	Not Used
Function 3	Not Used	Used	Not Used	Not Used	Used
Function 4	Used	Not Used	Used	Not Used	Not Used
Function 5	Not Used	Used	Used	Used	Not Used
Function 6	Not Used	Not Used	Not Used	Used	Used

- ***The Functional perspective*** is an internal, closed system or 'white box' perspective which abstracts out or hides everything outside the system boundary and provides a *dynamic* perspective of the mission and support functions performed by the system during the operational scenarios. The relationship between the functions performed by a system and the scenarios which use those functions can be represented as shown in Table 1[3].
- ***The Structural perspective*** is an internal perspective which provides a *static* perspective of the structure of system, including the system architecture, the physical components, the technology, people, organisation and subsystem boundaries. For example, the buildings, technology, ordnance and people in the ADS are in the *Structural* perspective.
- ***The Temporal perspective*** is a perspective of the evolution of the situation/system in the past and a projection of its future. This perspective:
  - Provides reflection on the past and lessons learned.
  - Identifies patterns of behaviour which can lead to future prevention of defects or errors, maintenance concepts, logistics issues, the need to deal with obsolescence.
- ***The Generic perspective*** provides an external perspective of the similarities to other systems and considers the system as an instance of a class/type of system. For example, a cargo ship is a surface ship; a car is a land vehicle, a surface to air missile is a missile as well as an unmanned aerial vehicle (UAV), etc. This perspective can identify patterns of behaviour and produces the concept of inheritance. For example:
  - the system inherits from …,
  - the system behaves like …,
  - the system looks like …
- ***The Continuum perspective*** provides an external perspective pertaining to the differences between the system and other systems, such as the difference between the ADS and the ADS of a similar nation. This perception:
  - Focuses on differences between the system and other systems, such as behaviour, colour, shape, size, etc.
  - Is sometimes known as divergent thinking.
  - Leads to a range of solutions rather than a single solution.
  - Helps visualise things in shades of grey rather than in black and white.
  - Indicates that either/or solutions are only two points on a continuum of potential solutions.
- ***The Quantitative perspective*** provides the:

---

[3] The words "Not Used" are inserted into the table to avoid and error of omission in the table since a blank entry could either mean not used or be an error.

  - Numbers associated with the other perspectives.
  - Insights as to the significance of information such as in the 80-20 rule sometimes known as the Pareto principle named after Vilfredo Pareto (1848-1923).

- ***The Scientific perspective*** is the outcome of the analysis often in the form of a hypothesis/guess as to:
  - An understanding of situation.
  - A statement of problem/issue.
  - An idea of what needs to change to remove the undesirable attribute of the situation.
  - A vision of the FCFDS (S3) containing the solution system (S6) that once fielded into service should turn into the reality of an actual situation (S7).

3.1.3. The linkage between the perspectives

Perceptions from each perspective provide information about part of the situation. For example, consider a car as the system in the context of home family life. When the car is perceived from the HTPs, the perceptions might include:

- ***Big picture*** - road network, traffic congestion at various times of the day, etc.
- ***Operational*** - going shopping, taking the children to school, etc.
- ***Functional*** – parking, parked, starting, traveling from place to place, stopping, etc.
- ***Structural*** – car with four doors, chassis, engine, transmission, six wheels and boot/trunk.
- ***Generic*** - (4-wheeled land vehicle) trucks, vans, etc.
- ***Continuum*** – different types of engines and vehicles (land and non-land), etc. e.g. internal combustion, external combustion, steam, electric engines
- ***Temporal*** – transportation before the motor car, different types of propulsion systems in the early days of the horseless carriage, steam, internal combustion, external combustion, etc. and future concepts in cars. e.g., Stanley steamer, Ford Model T, Ford Edsel, hybrid cars etc.
- ***Quantitative*** – mph, engine power, number of passengers, cost, price, etc.
- ***Scientific*** – depends on what is undesirable about the car.

3.1.4. Storing information in the HTPs

When analyzing a situation, in general, with respect to the system and situation, descriptions of:

- ***"Who …"*** belong in the:
  - *Big Picture* perspective if pertinent to an adjacent system or systems.
  - *Operational* perspective if pertinent to who is performing in a scenario, vignette or use case.

- ***"What …"*** belong in the:
  - *Big Picture* perspective if it is pertinent to the purpose of the system.
  - *Operational* perspective if pertinent to a scenario, vignette or use case.
  - *Structural* perspective if pertinent to a physical element of the situation.

- ***"Where …"*** belong in the *Big Picture* perspective or the *Structural* perspective.
- ***"When …"*** belong in the:
  - *Operational* perspective if pertinent to a scenario, vignette or use case.

- *Temporal* perspective if pertinent to the timeline in the story leading up to the situation.

- ***"Why …"*** belong in the *Big Picture* perspective except when it is explaining something in another perspective.
- ***"How …"*** belong in the:
  - *Operational* perspective; how it is used which tends to be expressed in scenarios.
  - *Functional* perspective or the *Structural* perspective; how it works which tends to be expressed in functions or mechanics.

In addition:

- Numeric information belongs in the *Quantitative* perspective.
- The cause or reason for the undesirable situation and what to do to remedy as well as how to do it, it is then inferred and stated from the *Scientific* perspective.
- If the system goes through different states and there are major differences in its attributes in each state as time passed, then each state should be considered as a different system and a different set of HTPs for each state will be needed

3.2. Develop an understanding of the situation

After examining the situation from the eight descriptive HTPs, the systems engineer should develop an understanding of the situation. For example:

- The entities involved in the situation should have been identified. These entities include those directly involved and the indirect stakeholders. See Kasser, Zhao and Mirchandani for an example of using the Nine-System Model to manage stakeholder expectations (Kasser, et al., 2014).
- The behaviour of the SOI can be understood from the information obtained from the relationships in the *Operational* and *Functional* perspectives. This information is often used to build a behavioural model.
- The undesirable aspects tend to show up in the *Structural*, *Operational* and *Functional* perspectives and should have been identified by discussions with the stakeholder and by analysis.
- The cause or causes of the undesirability and a conceptual approach to remedying the undesirability should have been inferred (*Scientific* perspective) from the eight descriptive HTPs.

3.3. Create the Feasible Conceptual Future Desirable Situation

The FCFDS (S3) is a modified existing situation (S1). Even in in situations where the stakeholders cannot agree on the causes of the undesirability, they should be able to agree on the nature of the undesirability and a situation in which the desirability is no longer present. As such, the initial version of the FCFDS is the existing situation with the undesirability removed, and often with suggested improvements added.

The FCFDS will contain a number of elements coupled together as shown in Figure 3.

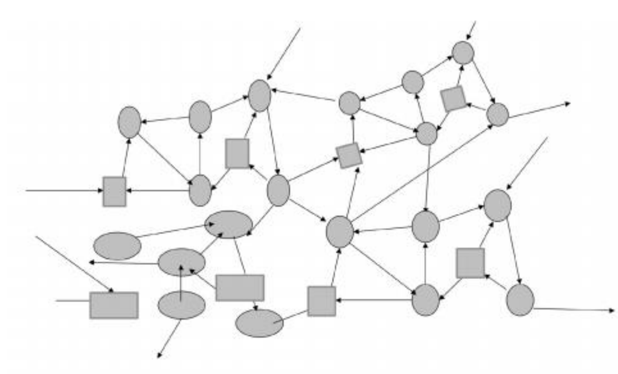

Figure 3 The FCFDS

3.4. Use the principle of hierarchies

The principle of hierarchies in systems (Spencer, 1862) cited by (Wilson, 2002) is one of the ways humanity has managed complexity for most of its recorded history. It includes:

- Keeping the systems and subsystems at the same level in the hierarchy of systems.
- Abstracting out or hiding the internal components of systems and subsystems. For example, Maier and Rechtin recommend that the way to deal with high levels of complexity is to abstract the system at a high a level as possible and then progressively reduce the level of abstraction (Maier and Rechtin, 2000) page 6).
- The concept that one systems engineer's subsystem is another systems engineer's SOI. For example:
    - The ADS is the SOI as far as the ADS systems engineer is concerned but it is a subsystem within the NDF.
    - A missile battery is a subsystem of the ADS, but is the SOI as far as the missile battery systems engineer is concerned.
    - A missile is a subsystem of the missile battery, but is the SOI as far as the missile systems engineer is concerned.
    - The radar is a subsystem of the missile battery, but is the SOI as far as the radar systems engineer is concerned.

A situation is a system which contains a number of systems. Each system in turn may contain a number of subsystems. Each subsystem may be further elaborated into a number of components (subsystems of the subsystem). This concept is often shown in the traditional hierarchical structure such as in organisation charts, work breakdown structures and product breakdown structures.

3.5. Abstract out the parts of the situation that are not pertinent to the problem

Dealing with issues in any specific situation will probably only need a subset of the information perceived from the different perspectives. For example, consider the problem of docking a resupply vehicle such as the US Space Transportation System (Space Shuttle) to the International Space Station (ISS). Each is a complex system in itself, yet when solving the problem of docking a Shuttle to the ISS, all the underlying complexity that is not relevant to the docking problem is abstracted out. Thus, we construct a closed system view to simplify the problem by abstracting out (filtering out) everything other than information pertinent to the:

- Relative positions of the spacecraft.
- Relative velocity of the spacecraft.
- Relative orientation in X, Y and Z axes of rotation.

So, it is the systems engineer's role to determine which elements are pertinent to the problem and abstract out the remainder[4].

3.6. Partition the FCFDS into the SOI and adjacent systems

It is the act of drawing the system boundary that creates the system (Beer, 1994; Churchman, 1979) page 91). When the undesirable situation already contains a SOI, such as in an upgrade or replacement situation, then the existing SOI tends to be the starting point for creating a

[4] When dealing with existing systems or systems that have already been realized in other places, this information will be generally be available using the *Generic* perspective. When dealing with unprecedented systems, good systems engineers will immerse themselves in the situation to identify which elements are important, the underlying assumptions that may cause problems, etc.

new SOI. However, the systems engineer should not assume that the boundaries of the existing and new (replacement) SOIs are identical and keep in mind that the boundaries of the SOI may need to change to remedy the undesirable situation as described below.

The entities in the FCFDS should be aggregated into the SOI and adjacent systems by some common denominator such as function, mission or physical commonality according to the rules for performing the aggregation described below.

3.6.1. Rules for performing the aggregation

When performing the aggregation, the rules to follow are:

1. ***Keep number of subsystems at any level to less than 7±2*** in accordance with Miller's rule to facilitate human understanding of the SOI (Miller, 1956).
2. ***Configure each subsystem for the maximum degree of homeostasis***. This rule:
    - which is widely used in human systems as well as in technological systems provides risk management and interface simplification since a subsystem configured according to this rule:
    - Ensures that the subsystem can continue to operate if the command and control link is lost.
    - Often requires a simple interface that passes relatively low-speed high-level commands and status information rather than high-speed real-time control commands.
3. ***Maximize the cohesion of the individual subsystems and minimize the coupling between subsystems*** (Ward and Mellor, 1985).

3.6.1.1. Cohesion and coupling

There are various types of cohesion and coupling.

3.6.1.1.1. Continuum of coupling

When perceiving coupling and cohesion from the *Continuum* perspective, the degree of coupling and cohesion can be seen as lying on a continuum as follows:

- ***Independent:*** the end of the continuum where the elements are not coupled at all.
- ***Interdependent:*** the continuum where the coupling of the elements ranges from loosely-coupled to tightly-coupled.
- ***Inseparable:*** the other end of the continuum where the elements are so tightly coupled that they cannot be separated.

3.6.1.1.2. Relating or joining the elements together

Cohesion and coupling also define how the elements relate or join together, where:

- ***Cohesion*** is the term used with respect to the view of <u>a single</u> system or subsystem such as the elements shown in Figure 1.
- ***Coupling*** is the term used with respect to a view of <u>more than a single</u> subsystem or subsystem.

Ian Sommerville provided the following list of types of cohesion in the software domain (Sommerville, 1998):

1. ***Coincidental****:* the elements have no relationship.
2. ***Logical:*** the elements are performing similar functions.
3. ***Temporal:*** the ***elements*** that are activated at a single (the same) time.
4. ***Procedural:*** the ***elements*** make up a single control sequence.

5. ***Communicational:*** the elements that operate on the same input data or produce the same output data.
6. ***Sequential:*** the output from one element in the component serves as input for some other element.
7. ***Functional:*** each element is necessary for the execution of a single higher level function.

Other types of coupling from the software domain include:

- ***Content coupling (high):*** one element modifies or relies on the internal workings of another element, e.g. accessing local data of another element.
- ***Common coupling:*** two elements share the same global data, e.g. a global variable.
- ***External coupling:*** two elements share an externally imposed data format, communication protocol, or device interface.
- ***Control coupling:*** one element controls the logic of another, by passing it information on what to do, e.g. passing a what-to-do flag.
- ***Stamp coupling (Data-structured coupling):*** the elements share a composite data structure and use only a part of it, possibly a different part, e.g. passing a whole record to a function which only needs one field.
- ***Data coupling:*** the elements share data, e.g., through parameters.
- ***Message coupling (low):*** the elements are not dependent on each other; instead they use a public interface to exchange parameter-less messages.
- ***No coupling:*** the elements do not communicate with one another.

In the physical realm, one can add other forms of coupling including:

- ***Mechanical coupling:*** the elements are coupled together by mechanical means, e.g. rivets, nuts and bolts, nails, joints, glue, welds, hook and loop fasteners, etc.
- ***Gravitic coupling:*** the elements are coupled together by gravity, e.g. one element rests on top of another. This type of coupling is common on planetary surfaces.
- ***Magnetic coupling:*** the elements are coupled together by magnetic means, e.g. intruder alarms, magnetic locks and items on refrigerator doors.
- ***Electrostatic coupling:*** the elements are coupled together by electrostatic charges.

Each type of coupling has advantages and disadvantages. The role of the systems engineer is to examine the different ways components can be aggregated into subsystems and use a design approach that maximizes cohesion and minimizes coupling which contributes to optimizing the interaction between the interfaces of the subsystems. A useful tool to perform this activity is the N^2 chart (Lano, 1977) or the Design Structure Matrix (Eppinger and Browning, 2012).

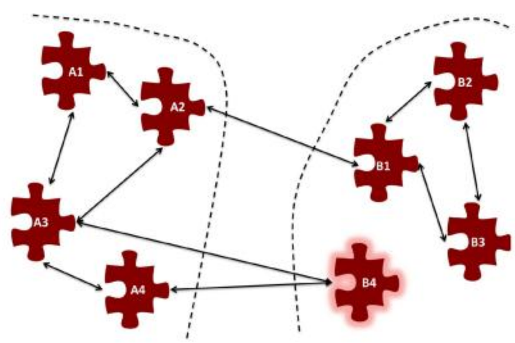

Figure 4 Cohesion and coupling

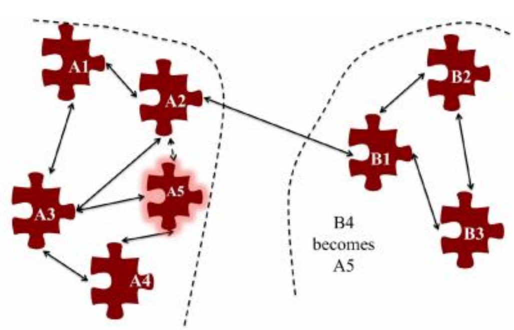

Figure 5 Better coupling and cohesions

3.6.1.2. Variations on a theme

However, maximizing cohesion and minimizing coupling is not always the rule in systems engineering. Consider the two subsystems 'A' and 'B' shown in Figure 4. There are three interfaces between the two subsystems. Note that element B4 in subsystem 'B' does have any connection with the remaining elements in subsystem 'B'. From the software perspective the coupling is coincidental and if the rules are followed, element B4 should be moved to subsystem 'A' to become element A5 and reduce the number of interfaces to a single interface as shown in Figure 5. The systems engineering rules are slightly different and depend on the situation. For example:

- If subsystem 'A' is the flight subsystem of an aerial reconnaissance system and subsystem 'B' is the ground control subsystem, then B4 may be located in the ground subsystem because it may consume too much power or be too heavy to fly. In such a situation, it is the role of the systems engineer to monitor the rate of change in technology to determine that in the future, should the system be upgraded, element B4 is a candidate for replacement with a different technology that would allow it to be moved to subsystem A4.
- Element B4 could also represent a function performed by the operator in the ground subsystem in the initial release of the operational software. This approach allows for an incremental software delivery approach where the function is intended to be migrated to the flight subsystem in subsequent software upgrades.

3.6.2. Creating the SOI and adjacent systems

It is the act of drawing the system boundary that creates the system (Beer, 1994; Churchman, 1979) page 91). The FCFDS is partitioned into the SOI and adjacent systems using the rules for performing aggregation discussed in Section 3.6.1.

3.7. Optimize the interfaces

Optimizing complex systems represents a challenge for reasons that include:

1. There will usually be different viewpoints on what should be optimized.
2. Traditional approaches to complex systems development either ignore the issue or optimize subsystems.

Addressing the second challenge, Wymore stated "*Conventional systems engineering wisdom has it that if subsystems are optimized, then the system cannot be optimum.*" (Wymore, 1997) and then used a mathematical approach to show that conventional wisdom was mistaken and how it was possible for systems engineers to ensure that optimum design of the subsystems can result in an optimum design of the system. The principle of hierarchies also indicates that conventional wisdom is wrong but in a graphical manner and without providing an optimal design *since system optimization at one level is always a subsystem optimization of the metasystem*.

If any system is a subsystem of the containing or metasystem, then where does the optimization take place? The answer is that system optimization at any level optimizes the interactions between the subsystems at that system level within the constraints imposed by the systems engineer of the metasystem, via:

1. The "*the proper allocation of the system requirements to the subsystems*" (Wymore, 1997).
2. The rules for performing the aggregation discussed in Section 3.6.1.

3.8. Partition the SOI into subsystems

Once the FCFDS has been partitioned into its subsystems, the SOI and adjacent systems, by the metasystem systems engineer, the SOI systems engineer then partitions the SOI into subsystems using the same process for creating a system, namely by going back to Section 3.1 and working on the SOI. This is in accordance with the concept that one systems engineer's subsystem is another systems engineer's system in the hierarchy of systems.

The internal subsystem partitioning within each adjacent system are the province of the particular adjacent system systems engineer just like the internal details of the SOI are the province of the SOI systems engineer. Note:

1. In some cases the system boundaries may need to change over time, such as when an organization is reorganized and as discussed in cohesion and coupling above.
2. The metasystem systems engineer may occasionally override the SOI subsystem partitioning to meet metasystem requirements as discussed in Section 3.6.1.2.

4. The recursive perspective

As may be noted from Section 3.8, the Nine-System Model process discussed in Section 3 is recursive. The first time through the process, the SOI is entire undesirable situation (S1) which is partitioned into the SOI and adjacent systems. The second time through the process, the undesirable situation (S1) is the need to partition the SOI into its subsystems. The adjacent systems are the province of their own systems engineers. The third time through the process the undesirable situation may be the need to realize a subsystem of the SOI (Kasser and Zhao, 2014).

5. HTP contribution to the system requirements

If the SOI is going to be created, then the system development process (S5) includes the production of a matched set of specifications for the SOI (S6) and each of its subsystems. In general, the:

- ***Big picture*** perspective contributes to the interface requirements.
- ***Operational*** perspective contributes to the performance requirements.
- ***Functional*** perspective contributes to the functional requirements.
- ***Structural*** perspective contributes to the technology, physical and 'ility' requirements, e.g., reliability, maintainability, survivability etc.
- ***Generic*** perspective contributes requirements than can be inherited from that class of system.
- ***Continuum*** perspective contributes to identifying differences between the SOI and similar systems that affect the requirements. For example, some of the requirements may not be as stringent, or may be more stringent than those of a particular similar system.
- ***Temporal*** perspective contributes to the requirements for adoption of new technology, managing obsolescence and flexibility to adapt to future situations.
- ***Quantitative*** perspective provides the numbers and tolerances for the functional and performance requirements.

6. Summary

This paper, written for both educators and practitioners filled a gap in the literature by introducing and describing an iterative holistic process for conceptualizing and creating a system and/or situation, namely the paper described the S2 process in the Nine-System Model (Kasser and Zhao, 2014).

7. Biography

Joseph Kasser has been a practicing systems engineer for more than 40 years and an academic for about 16 years. He is a Fellow of the Institution of Engineering and Technology (IET), an INCOSE Fellow, the author of *"Holistic thinking: creating innovative solutions to complex problems"*, "*A Framework for Understanding Systems Engineering*" and "*Applying Total Quality Management to Systems Engineering"*, and many INCOSE symposia papers. He is a recipient of NASA's Manned Space Flight Awareness Award (Silver Snoopy) for quality and technical excellence for performing and directing systems engineering and other awards. He holds a Doctor of Science in Engineering Management from The George Washington University. He is a Certified Manager and holds a Certified Membership of the Association for Learning Technology. He also started and served as the inaugural president of INCOSE Australia and served as a Region VI Representative to the INCOSE Member Board. He has performed and directed systems engineering in the UK, USA, Israel and Australia. He gave up his positions as a Deputy Director and DSTO Associate Research Professor at the Systems Engineering and Evaluation Centre at the University of South Australia in early 2007 to move to the UK to develop the world's first immersion course in systems engineering as a Leverhulme Visiting Professor at Cranfield University. He is currently a Visiting Associate Professor at the National University of Singapore.

8. References

Athey, T. H., *Systematic Systems Approach. An Integrated Method for Solving Systems Problems*, Prentice-Hall, Englewood Cliffs, NJ., 1982.

Beer, S., *The Heart of Enterprise*, John Wiley & Sons, Stafford Beer Classic Edition, Chichester, 1994.

Churchman, C. W., *The Systems Approach and its Enemies*, Basic Books, Inc., New York, 1979.

Clark, R., *System Dynamics and Modeling*, Operations Research Society of America, 1998.

DoDAF, *DoD Architecture Framework Version 1.0, 9 February 2004*, 2004.

Eppinger, S. D. and Browning, T. R., *Design Structure Matrix Methods and Applications*, The MIT Press, Cambridge, MA., 2012.

Flood, R. L. and Jackson, M. C., *Creative Problem Solving*, Wiley, 1991.

Giganti, J. J., Larson, J. V., Richard, J. P. and Weber, J., "12 Lunar Surface Gravimeter Experiment," *APOLLO 17 Preliminary Science Report* E. M. Jones and K. Glover (Editors), NASA, 1971.

Hately, D. J. and Pirbhai, I. A., *Strategies for Real-time systems Specification*, Dorset House Publishing, New York, NY, 1987.

Kasser, J. E., *Holistic Thinking: creating innovative solutions to complex problems*, Createspace Ltd., 2013.

Kasser, J. E. and Zhao, Y.-Y., *Managing Complexity via the Nine Systems in Systems Engineering*, proceedings of the 24th International Symposium of the International Council on Systems Engineering (INCOSE), Las Vegas, NV, 2014.

Kasser, J. E., Zhao, Y.-Y. and Mirchandani, C. J., *Simplifying Managing Stakeholder Expectations using the Nine-System Model and the Holistic Thinking Perspectives*, proceedings of the 24th International Symposium of the International Council on Systems Engineering (INCOSE), Las Vegas, NV, 2014.

Kipling, J. R., "The Elephant's Child," *The Just So Stories*, The Country Life Press, Garden City, NY, 1912.

Kline, S. J., *Conceptual Foundations for Multidisciplinary Thinking*, Stanford University Press, Stanford, 1995.

Lano, R., The N2 Chart, 1977.

Maier, M. K. and Rechtin, E., *The Art of Systems Architecting*, CRC Press, 2000.

Miller, G., *The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information.*, The Psychological Review 63 (1956), 81-97.

O'Connor, J. and McDermott, I., *The Art of Systems Thinking*, Thorsons, San Francisco, 1997.

Senge, P. M., *The Fifth Discipline: The Art & Practice of the Learning Organization*, Doubleday, New York, 1990.

Sommerville, I., *Software Engineering*, Addison-Wesley, 1998.

Spencer, H., "First principles," *A System of Synthetic Philosophy*, Williams and Norgate (Editor), vol. 1, London, 1862.

Ward, P. T. and Mellor, S. J., *Structured Development for Real-Time Systems*, Yourdon Press, 1985.

Wilson, T. D., *Philosophical foundations and research relevance: issues for information research (Keynote address)*, proceedings of Fourth International Conference on Conceptions of Library and Information Science: Emerging Frameworks and Method, University of Washington, Seattle, USA, 2002.

Wolstenholme, E. F., *System Enquiry a system dynamics approach*, John Wiley & Sons, 1990.

Wymore, A. W., Subsystem Optimization Implies System Suboptimization: Not!, 1997, http://www.sie.arizona.edu/sysengr/wymore/optimal.html, accessed on 30 September 2013.

Yen, D. H., The Blind Men and the Elephant, 2008, http://www.noogenesis.com/pineapple/blind_men_elephant.html, accessed on 26 October 2010.