Teaching Technical Leadership in a Networked World

Scott Workinger, Ph.D.

Abstract

Networked computing is driving a new paradigm in systems engineering, characterized by size, complexity, different techniques and new types of working relationships. The challenges of educating systems engineers in the new paradigm are explored and a particular approach is outlined. The approach focuses upon an explicit presentation of the paradigm structure to give students an analytical framework for understanding the strengths and weaknesses of the new paradigm as it evolves. A comprehensive introduction to the new paradigm, engineering examples, the presentation of practical tools, and hands-on experience are viewed as crucial to helping make the new paradigm a reality to the students.

Introduction

A systems engineer's role is to exercise technical leadership in the analysis, architecture, implementation, and management of systems that transcend the scope of a single specialized engineering discipline. The development of networked systems has had a profound impact upon engineered systems. The largest man-made systems are many times as large as the largest systems that existed before networked computing became a ubiquitous part of most work environments in the 1990's. This has created unprecedented challenges, which systems engineers have begun to address. Beyond the new scope, there are new types of objects that are used in systems, new architectures, new techniques, new ways of working together for engineering professionals, and in some cases, new values. In the aggregate, there is so much new substance pertaining to this leap forward in professional practice that some systems engineers have begun to speak of a new paradigm.

As always, with new paradigms, there tend to be variations regarding terminology. Various individuals apply terms such as "Systems of Systems," "Capability Engineering" and "Complex Systems" in the context of the new paradigm.[1] These concepts and associated techniques are being applied in practical ways by working systems engineers. Even so, it is likely that terminology will continue to evolve as professionals develop new techniques, validate them in practical application, and consider new perspectives on the practical significance of the new developments. What seems well established, at this point, is that there are significant new developments and that it's time for those developments to be shared, so that they can 1) bring benefit to the wider community of systems engineering professionals, 2) generate dialog about the details of the new developments, and 3) stimulate further development of insight and technique.

Educating students in a new paradigm for professional practice can present serious instructional challenges. Most teaching is properly geared toward educating students in an existing paradigm. In these situations, the educational process tends to be routine 1) because the content is well known, 2) expectations tend to be well set, and 3) the path through the material is well established.

Instructional Stance

New paradigms change the rules not only for the content, but also for the students' expectations and for the instructional path. Hence, there is typically significant variation in the basic view, the scope of the material, and the strategy of instruction. There are several significant efforts to teach aspects of the new paradigm. Notably, Sarah Sheard is offering a course that addresses important complexity issues entitled, "What You Need to Know About Chaos, Complexity, and Complex Adaptive Systems to Do Systems Engineering Well in the 21st Century."

This paper describes a somewhat different approach that was used to develop a course entitled, "Systems of Systems, Technical Leadership in a Networked World." This particular approach is distinguished by

---

[1] This paper will not enter the debate upon whether these concepts individually or in the aggregate constitute a valid new paradigm for the profession of systems engineering. That issue is being addressed by other authors. Indeed, one expects this particular debate to continue for some time. Such argument is an important and necessary aspect of establishing and validating a new paradigm and both sides of such a debate make important contributions. This paper has a more modest objective, the discussion of how a new paradigm can be taught, using the new body of systems engineering techniques as an example.

1) its view, which attempts to address the new paradigm explicitly, 2) its scope, which attempts to present an overview of the whole new paradigm, which includes complexity issues, but includes other aspects of the new paradigm, as well, and 3) it's strategy of instruction, which is to focus upon engineering applications, including exercises. This is ambitious. Hence the intent is not to say everything that could be said about the new paradigm, but to simply attempt to make a useful contribution, while stimulating an active dialog and listening alertly to the experiences of students as they engage in class dialog. It is believed that there is power in addressing the paradigm, as a whole, because only addressing parts of the paradigm will create implicit (or explicit) objections in the minds of students.

Much of the research and discussion on complex systems theory uses examples from natural processes, such as biology and meteorology. These are important examples; however, engineers tend to avoid creating analytically indeterminate systems, wherever possible. Not surprisingly, many engineers are skeptical of examples from outside the engineering domain. In order to make the point that avoiding complexity isn't always possible and to make that point compellingly for an engineering audience, one needs engineering examples. Hence, the examples used in the course emphasize large, complex, engineered systems such as supply chains, logistics systems, electronic music distribution, military forces, airport operations, and the Internet.

One interesting aspect of complex systems theory is that complexity theorists cannot seem to agree on a definition of complexity. Most people's intuitive understanding of complexity is that it pertains to systems with so many components and relationships that it is difficult to analyze and predict system behavior. Many complexity theorists see complexity as a characteristic of systems with analytically indeterminate dynamics, such as a pendulum that is pushed outside its linear range of motion. But there are many competing definitions. Whimsically, one might suggest that a complex system is a system studied by complexity theorists. There may be more than a little truth in this.

Systems that were once thought complex become simple with the proper insight. So, complexity, in some sense describes systems whose behavior is currently not yielding to insight. The assertion that the behavior of any specific engineering system will always be beyond our insight, may be practically unprovable and pragmatically uninteresting, particularly since engineered systems can be re-engineered. What is clear from experience is that large, complex systems exhibit behavior that surprises their architects with alarming regularity. This has practical consequences that must be addressed by responsible systems engineers. The new paradigm includes strategies for addressing this issue.

In this paper, the author outlines an approach to teaching a new professional paradigm using the new systems engineering paradigm as the key example. To avoid argument about any particular concept among the new and rapidly evolving body of knowledge, the new paradigm is referred to simply as, "Technical Leadership in a Networked World."

The Structure of Paradigms

In his book, The Structure of Scientific Revolutions, Thomas Kuhn extensively discussed the nature of a paradigm as an organizing framework for scientific inquiry, focusing on 1) what is to be observed, 2) the kinds of questions that are posed, 3) how the questions are to be structured, and 4) how the results are to be interpreted. This seminal work has evoked application and extension of Kuhn's concepts in many other fields, including engineering. Systems engineering as it was formulated in the later half of the 20th century clearly applies a different paradigm than those of the specialized engineering disciplines.

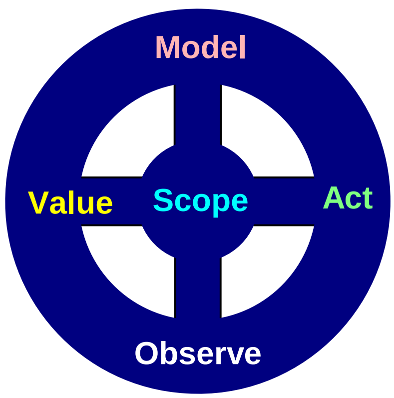

Figure 1: Generic Process Metaphor.

Due to the integrative nature of the discipline, one of the most important skills of mature systems engineers is the ability to assess whether or not he has enough information and knowledge to solve a problem. So, it is important, not simply to know, but to know what one knows (and doesn't know). Therefore, having an explicit structural understanding of the paradigm a systems engineer is applying can help him to understand crucial gaps in his current understanding or potentially help him to see deficiencies in the paradigm, itself. Such deficiencies are the rule, rather than the exception with new paradigms. First, there is a significant paradigm shift, a revolutionary process, then the "holes" in the new paradigm are plugged, an evolutionary process. This is the motivation for presenting the structure of paradigms to students when they are studying a new paradigm.

Figure 1 shows a wheel as a generic process metaphor. There are five components in this process metaphor:

- Scope – This action defines the space in which the process occurs. The space may be defined as a physical space or as an abstract space, such as a namespace. The other four components operate upon objects that exist in some sense in a defined space (or spaces).[2]
- Observe – This action binds[3] to objects in the defined space.
- Value – This action assess the qualities of bound objects in the defined space.
- Model – This action makes logical distinctions to build a model of objects in the defined space.
- Act – This action manipulates objects in the defined space. This may be action in a physical space, in the process of building a system or action in a cognitive space, such as an action in constructing an architectural model.

The process metaphor is iterative. Action leads to observation, and the process continues. This process metaphor is useful in describing physical processes, computing processes, and cognitive processes.

Figure 2 shows the structure of a paradigm. The five types of aggregates in a paradigm are:

- Spaces – The spaces that are defined as relevant. These may be physical spaces, analytically constructed spaces, abstract spaces.
- Objects – These are the objects that are considered relevant to the paradigm. An object may be a system, one of its components, or a relevant part of a system's environment.
- Valuations – These are the values that lead a particular object or feature to be assessed as positive, negative, or neutral in the context of the paradigm.
- Logical rules – These are the basis for discerning relationships between objects when establishing a model.[4]
- Actions – These are the techniques that are in some sense applicable for manipulating objects within the paradigm.

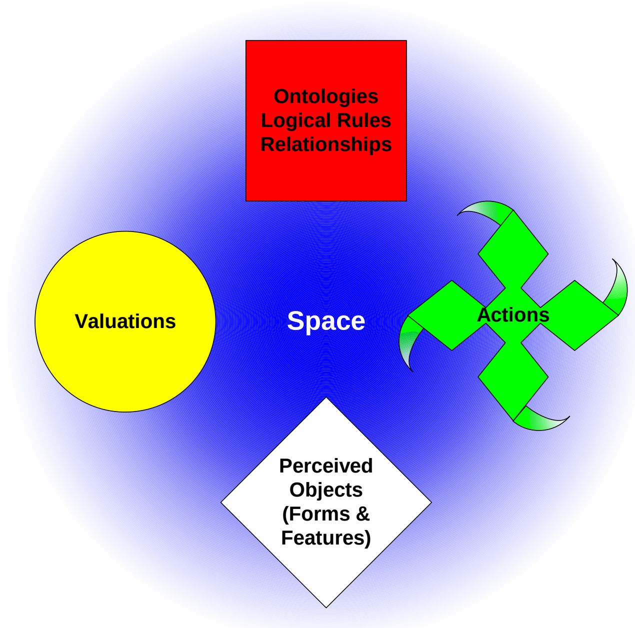

Figure 2: Generic Paradigm Structure

By itself, the general process is valuable for its wide applicability, but lacks strength. It gets its strength by being bound to a paradigm.

Figure 3 shows process and paradigm together. Each action and aggregate has its role. Objects are observed, valued, analyzed, and manipulated within the context of what is possible, according to the paradigm. To quote Edward Feigenbaum, "In the knowledge, is the power." Or, in this context, one might say that a process derives its power from the content of the paradigm it applies. This motivates the search for new paradigms.

---

[2] The ability to nimbly switch contexts (defined spaces) in a way appropriate to the problem at hand is an important systems engineering skill.

[3] At its simplest, a binding is a selection of "this object." More subtly, it implies the reification of the object that is bound.

[4] Thomas Gruber formally defined an ontology in a computational sense as "a specification of a conceptualization," coining the modern usage of the term. See http://www-ksl.stanford.edu/kst/what-is-an-ontology.html. This is distinct from the classical definition of the word ontology, used by philosophers.

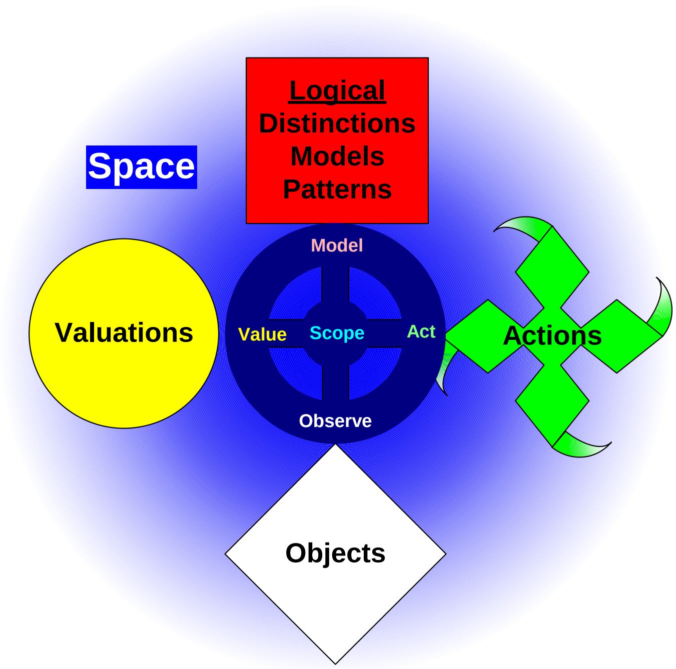

Figure 3: Process and Paradigm

Content of the New Paradigm

Some of the changes include:

- Scope - The scope of systems engineering effort has changed. Systems are many times larger and this has created scaling issues. There has been a corresponding increase in complexity. In some sense, the spaces are more complex.
- Engineered Objects – New types of objects are present in our engineering environments. Such objects are as small as interface components like network protocols and as large as the Internet, itself. Other new kinds of objects include systems of networked legacy systems, embedded and networked computers.
- Values – Although values are often subtle and in some sense, emergent, it is clear that new values are emerging. Attitudes toward complexity are changing. At one time, many systems engineers believed that every aspect of a well-engineered system's performance could be designed in advance. That belief is changing. Attitudes toward working relationships are changing, also. Twenty years ago, engineered systems were owned and systems engineers were employed by systems' owners. Today, important engineered systems, such as the Internet are not owned by any single entity. This is often the case with Systems of Systems. Systems engineers must get comfortable with other kinds of relationships, such as "coopetition." In many situations with systems of systems, an engineer will not have complete control of an architecture and must settle for influence. Twenty years ago, the values of senior architects would have led them to consider such a situation as unacceptable.
- Architectural Patterns - New systems architectural patterns are being used and in many cases, these patterns are based upon new distinctions. Examples include systems of systems, networked computation, the layered interface, the interface constitution, loose coupling, and the stateless server.
- Processes - New processes are being used both inside systems and to design the systems. There are four major categories of processes that are discussed at length in the course: Architecture, integration, testing and evaluation, and leadership and collaboration. These will be discussed in the next section.

In the aggregate, then, when we look at these factors, it is natural to talk about a new professional paradigm for Systems Engineering practice.

Four Types of Processes

Four types of processes are discussed within the new paradigm.

Architecture

Traditionally, architecture has been regarded as a "top-down" activity. While this remains important with large systems, in a systems of systems engineering context certain aspects are emphasized:

- Capability Engineering - A distinctive aspect of systems of systems is that they are often developed by connecting existing, legacy systems. This has led to a new set of procedures known as capability engineering as part of the new paradigm.
- Analysis of Legacy Systems - One aspect of capability engineering that was less important with systems architected in a "green field" context, is the critical nature of the analysis of existing legacy systems that are modified to support the new capability.
- Architectural Frameworks – Simply viewing large, complex systems in ways that support systems engineering tasks is challenging. Architectural frameworks are discussed as a tool for this purpose with DoDAF given as a significant example.

- Dynamic Optimization – The practical complexity of the new systems is such that it is difficult to optimize their initial implementations. Strategies for improving performance over time are discussed, with examples given, such as the history of supply chain management.
- Architectural Patterns – Some of our largest systems would be impossible without the development of new architectural patterns. One prominent example from the course is the Internet, and what makes it work. The Internet could not have been developed without the use of new architectural patterns. Since networked computation is, in some sense, driving the new paradigm, the details of the new architectural patterns are discussed with some care, particularly those that support massive scaling and integration.

Integration

Eberhardt Rechtin observed that, "The greatest leverage in architecting is at the interfaces." Networked computation is driving the new paradigm. An examination of networked computation shows the importance of interfaces and how they are used. Key issues include:

- Characterizing interfaces – Interfaces have characteristics that need to be well understood.
- Interface abstraction – The strategy of designing interfaces using layers of abstraction is examined using the most successful example of this approach, the TCP/IP protocol suite.
- Collaboration role – Well defined interfaces have a key role in facilitating collaboration inside an engineering organization. This includes designing for emergence.
- Interface standards and Interface Constitutions – Standardizing interfaces is examined as a strategy.
- COTS – The success of Commercial Off the Shelf Systems is significantly influenced by interface design. Successful strategies are examined.
- Coupling – Loose versus tight coupling is too narrow a characterization for most systems engineering situations. The dimensions of coupling are examined.
- Legacy Systems – Success strategies for integrating legacy systems are examined, including configuration management issues and design for integration.

Testing & Evaluation

The potential for emergent behavior is magnified by the size and complexity of systems developed, today. At the same time, testing is more difficult. In many cases, it is impossible to fully test systems. It may even be impossible to fully simulate a large system. Strategies for uncovering emergent behavior, testing, simulation and evaluation are examined.

Leadership & Collaboration

Leadership and collaboration have changed dramatically. For many systems of systems, an "SoS team" is a myth. Teams are not geographically co-located. Systems engineers are often "collaborating" with people that haven't entered the workforce concerning projects that haven't even been visualized, as yet. This occurs because much of the value for certain well-designed systems of systems is emergent. The Internet, which performed well beyond what its original designers visualized, is an example of this.

Making it Real

In some sense, a new paradigm isn't "real" to students until they have internalized it. This can be addressed by using realistic teaching examples; however, for a new paradigm, that usually won't be sufficient. The new paradigm won't be real until the students have hands-on experience with it. In the class project, students are divided into four groups. Student groups are given responsibility for a robot with a working design and an initial exercise that puts each group in competition with the others. Then the students are shown collective emergent behavior for the system of systems with all four robots and given a new goal that requires cooperative behavior.[5] The new goal is to channel the emergent behavior and create better performance for the new capability. This includes creating an attractor cycle for the system of systems. Students gain hands-on experience with systems engineering for a complex system and the interpersonal experience of a situation with both competitive and cooperative aspects.

[5] The dynamics are "complex" in a complexity theoretic sense.

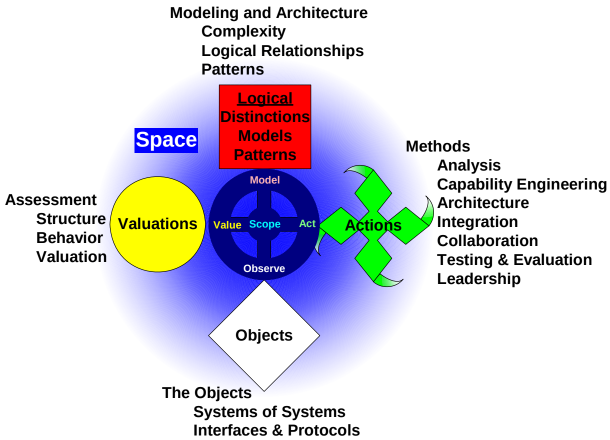

Figure 4: Technical Leadership in a Networked World

Summary

Education in the new paradigm presents significant challenges. One strategy for addressing these is to be explicit about the paradigm to give students an analytical framework for understanding the strengths and weaknesses of the new paradigm as it evolves. A comprehensive introduction to the new paradigm, engineering examples, the presentation of practical tools, and hands-on experience are viewed as crucial to helping make the new paradigm a reality to the students. Given the dynamics of globalization, the new paradigm is expected to evolve rapidly, and ongoing dialog with students and rapid adaptation is viewed as essential.

Acknowledgements

I am indebted to my colleagues Eric Honour, Dave Walden, and John Pratchios for their collaborative efforts in developing the course described in this article and for their efforts to educate their colleague. I am also indebted to Honourcode for seeing the value in this initiative and sponsoring the development of the course. The privilege of working with excellent colleagues is the icing on the cake of a professional career.