Homework Assignment #1

Comparison of Systems Thinking Methodologies

- Review Section 2.9.2 of SEH v4 which describes two specific methods of systems thinking, and some associated common traits.
- Describe (<= 2 pages):
  - Two of the methods
  - The differences between the two methods selected
  - A rationale for choosing one method over the other
- This assignment requires you to read Systems Engineering Handbook v4 (SEH v4) section 2.9.2 which describes methods of systems thinking, to select two of the methods to describe and contrast, and finally to provide a rationale for selecting one over the other. NOTE, there are two methods specifically described in subsections 2.9.2.1 (System Dynamics), and 2.9.2.2 (Soft Systems and Action Research). Subsections 2.9.2.3 (Discovering Patterns), and 2.9.2.4 (Habits of a Systems Thinker) provides additional attributes of systems thinking and/or a systems thinker, which actually applies to all methods, but does **not** describe additional specific methods.

Due 6pm ET, 19 Aug

p. 5

Table 2.4 SE return on investment

From Honour (2013). Reprinted with permission from Eric Honour.

Current SE effort (% of program cost)	Average cost overrun (%)	ROI for additional SE effort (cost reduction $ per $ SE added)
0	53	7.0
5	24	4.6
7.2 (median of all programs)	15	3.5
10	7	2.1
15	3	−0.3
20	10	−2.8

2.9 SYSTEMS SCIENCE AND SYSTEMS THINKING

This section summarizes the nature of systems science and systems thinking. It also describes how they relate to SE.

2.9.1 Systems Science

Systems science brings together research into all aspects of systems with the goal of identifying, exploring, and understanding patterns of complexity that cross disciplinary fields and areas of application. It seeks to develop interdisciplinary foundations that can form the basis of theories applicable to all types of systems (e.g., in nature, society, and engineering) independent of element type or application.

p. 6

Additionally, systems science can help to provide a common language and intellectual foundation for SE and make practical system concepts, principles, patterns, and tools accessible to practitioners of the "systems approach." An integrated systems approach for solving complex problems needs to combine elements of systems science, systems thinking, and SE. As such, systems science can serve as the foundation for a metadiscipline that unifies the traditional scientific specializations.

The information in this section is extracted from the systems science article in Part 2 of the SEBoK (SEBoK, 2014). Figure 2.8 illustrates the relationships between systems science, systems thinking, and general systems approach as applied to engineered systems.

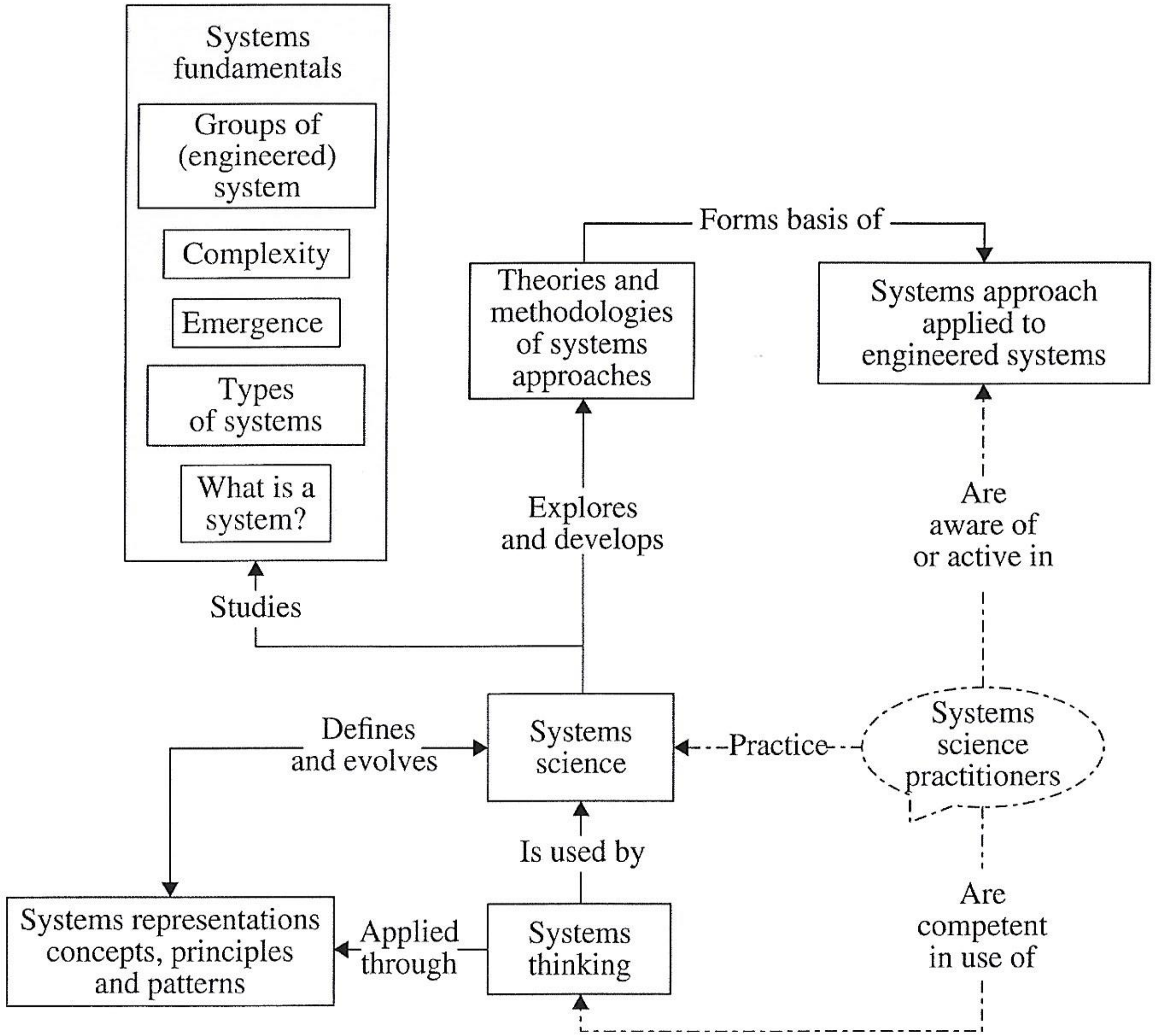


Figure 2.8 Systems science in context.

From SEBoK (2014). Reprinted with permission from the BKCASE Editorial Board.

p. 7

Systems science is an integrative discipline that brings together ideas from a wide range of sources sharing in a common systems theme. Some fundamental concepts now used in systems science have been present in other disciplines for many centuries, while equally fundamental concepts have independently emerged as recently as 40 years ago (Flood and

Carson, 1993).

Systems science is both the "science of systems" and the "systems approach to science," covering theories and methods that contrast with those of other sciences, which are generally reductionist in nature. Where it is appropriate, the reductionist approach has been very successful in using the methods of separating and isolating in search of simplicity. However, where those methods are not appropriate, systems science relies on connecting and contextualizing to identify patterns of organized complexity.

Questions about the nature of systems, organization, and complexity are not specific to the modern age. As John Warfield (2006) put it,

> Virtually every important concept that backs up the key ideas emergent in systems literature is found in ancient literature and in the centuries that follow.

p. 8

It was not until the middle of the twentieth century, however, that there was a growing sense of a need for, and possibility of, a scientific approach to the problems of organization and complexity in a "science of systems" per se.

Biologist Ludwig von Bertalanffy was one of the first to argue for and develop a broadly applicable scientific research approach based on *open system theory* (Bertalanffy, 1950). He explained the scientific need for systems research in terms of the limitations of analytical procedures in science. These limitations are based on the idea that an entity can be resolved into and reconstituted from its parts, either materially or conceptually:

> This is the basic principle of "classical" science, which can be circumscribed in different ways: resolution into isolable causal trains or seeking for "atomic" units in the various fields of science, etc.

p. 18

Research in systems science attempts to compensate for the inherent limitations of classical science, most notably the lack of ways to deal with emergence. Systems science has developed—and continues to develop—hand in hand with practice, each maturing and learning from the other. Various efforts have taken on complementary or overlapping issues of the new "systems approach" as progress has been made over time:

- Cybernetics (Ashby, 1956; Wiener, 1948)
- Open system and general system theory (Bertalanffy, 1950, 1968; Flood, 1999)
- Operations research (Churchman et al., 1950)
- Hard and soft systems thinking (Checkland, 1998; Lewin, 1958)
- Organizational cybernetics (Beer, 1959; Flood, 1999)
- Critical systems thinking (Jackson, 1989)

- System dynamics (Forrester, 1961; Senge, 1990)
- SE (Hall, 1962)
- System analysis (Ryan, 2008)
- Service science and service SE (Katzan, 2008)

A broader discussion of contrasts between analytical procedures and integrative system concepts is provided in *Model-Oriented Systems Engineering Science* (Hybertson, 2009) from the perspective of a traditional versus the complex SE views of systems.

2.9.2 Systems Thinking

As a systems engineer, it is vital to develop knowledge and skills that can be utilized in performing a deep analysis of problem or opportunity situations for which system responses are required. As noted earlier, systems science has contributed to the development of such knowledge. However, during the twentieth century, a number of approaches to performing deep analysis have arisen under the title of "systems thinking." While it is difficult to put a precise boundary around systems thinking and differentiate it from systems science, many systems thinking methods and tools have become popular and have been successfully utilized in multidisciplinary contexts. p. 19

2.9.2.1 System Dynamics

Jay Forrester of MIT developed the DYNAMO simulation language and observed that a common means of analyzing complex systems could be used in multiple disciplines (1961). Several students of Forrester refined these ideas into methodologies and tools that have provided a useful basis for analysis. Peter Senge, with his popular book *The Fifth Discipline* (1990), established systems thinking as a discipline. He also developed the link, loop, and delay language as a means of graphically representing system dynamics. Based upon two primary loops (growth and limit), a number of so-called archetypes have been developed to describe a variety of situations. Another student, Barry Richmond, further developed archetypes by adding flow mechanisms in the simulation languages STELLA and iTHINK, which are commercially available.

2.9.2.2 Soft Systems and Action Research

Peter Checkland (1975) observed that the use of classical engineering approaches to complex problems falls down since there are many soft factors (attitudes, practices, procedures, etc.) that affect systems. He also observed that the path to improvement must come through the development and analysis of alternative models. Based upon analysis, discussion, and dialogue, a course of action is planned and executed, and the results p. 9

observed as feedback for further analysis. John Boardman has been inspired by Checkland's work and has developed a version of the soft systems approach supported by a tool called Systemigrams (Boardman and Sauser, 2008).

2.9.2.3 Discovering Patterns

Central to systems thinking is the discovery of patterns. A pattern is a representation of similarities in a set of problems, solutions, or systems. Systems thinking captures and exploits what is common in a set of problems and corresponding solutions in the form of patterns of various types, as noted in the archetypes described previously.

Systems engineers use the general information provided by patterns to understand a specific system problem and to develop a specific system solution. Examples include, but are not limited to, the following:

- Software design patterns, such as adapter
- System architecture patterns, such as layered architecture and single sign-on
- Community or urban design patterns, such as ring road, pedestrian street, and food court
- Security and safety patterns, such as fault-tolerant design and role-based access control
- Interaction patterns, such as publish–subscribe
- Domain-specific patterns, such as the suspension bridge pattern

Pattern categories include domain taxonomies, standards, templates, architecture styles, reference architectures, product lines, abstract data types, and classes in class hierarchies.

Another important category of patterns is associated with complex, counterintuitive systems. An example is "shifting the burden." When a problem appears in a complex system, the intuitive response is to apply a quick short-term fix, rather than take the time to develop a long-term solution. The problem with that approach is that there is often a trade-off between the short-term fix and the long-term solution, such that the initial success of the quick fix reduces the chances the real solution will be developed. This pattern appears in the typical short-term/long-term priority struggle between the systems engineer, project manager, discipline engineering communities, and other stakeholders within a project. Shifting the burden is one of a set of "system archetypes," a class of patterns (sometimes called patterns of failure or antipatterns) that illustrate the underlying systems pathology behind many of the barriers to effective SE. Awareness and understanding of these patterns are important for people trying to embed the systems approach in their organization and take positive action to counter these challenges.

Additional examples and references are given in the article "Patterns of Systems Thinking" in (SEBoK, 2014).

2.9.2.4 Habits of a Systems Thinker

While there are several additional aspects of systems thinking introduced by multiple contributors, the following list summarizes essential properties of a systems thinker (Waters Foundation, 2013):

p. 15

- Seeks to understand the big picture
- Observes how elements within the system change over time, generating patterns and trends
- Recognizes that a systems' structure (elements and their interactions) generates behavior
- Identifies the circular nature of complex cause-and-effect relationships
- Surfaces and tests assumptions
- Changes perspective to increase understanding
- Considers an issue fully and resists the urge to come to a quick conclusion
- Considers how mental models affect current reality and the future
- Uses understanding of system structure to identify possible leverage actions
- Considers both short- and long-term consequences of actions
- Finds where unintended consequences emerge
- Recognizes the impact of time delays when exploring cause-and-effect relationships
- Checks results and changes actions if needed: "successive approximation"

People who think and act in a "systems way" are essential to the success of both research and practice (Lawson, 2010). Successful systems research will not only apply systems thinking to the topic being researched but should also consider a systems thinking approach to the way the research is planned and conducted. It would also be of benefit to have people involved in research who have, at a minimum, an awareness of systems practice and ideally are involved in practical applications of the theories they develop.

For a more thorough description of the "discipline" of systems thinking, refer to Part 2 of SEBoK (2014) as well as the popular website *Systems Thinking World* (Bellinger, 2013).

2.9.3 Considerations for Systems Engineers

p. 10

Sillitto (2012) provides a useful digest of concepts from systems science and systems thinking, organized so as to be immediately useful to the SE practitioner. Properties that are generally true of the sort of systems that systems engineers find themselves involved with are as follows:

1. A system exists within a wider "context" or environment.
    - The context includes an "operational environment," a "threat environment," and a "resource environment" (Hitchens, 2003).
    - The context may also contain collaborating and competing systems.
2. A system is made up of parts that interact with each other and the wider context.
    - The parts may be any or all of hardware, software, information, services, people, organizations, processes, services, etc.
    - Interactions may include exchange of information, energy, and resources.
3. A system has system-level properties ("emergent properties") that are properties of the whole system not attributable to individual parts.
    - Emergent properties depend on the structure (parts and relationships between them) of the whole system and on its interactions with the environment.
    - This structure determines the interactions between functions, behavior, and performance of the parts and interaction of the system with the environment—in ways both intended and unintended.
4. A system has the following:
    - A life cycle
    - Function, which can be characterized following Hitchens as "operate – maintain viability – manage resources" or as "observe – orient – decide – act" (Hitchens, 2003)
    - Structure, including the following:
        – A boundary, which may be static or dynamic and physical or conceptual

        – A set of parts

        – The set of relationships and potential interactions between the parts of the system and across the boundary (interfaces)
    - Behavior, including state change and exchange of information, energy, and resources
    - Performance characteristics associated with function and behavior in given

- Performance characteristics associated with function and behavior in given environmental conditions and system states

5. A system both changes and adapts to its environment when it is deployed (inserted into its environment).
6. Systems contain multiple feedback loops with variable time constants, so that cause-and-effect relationships may not be immediately obvious or easy to determine.

Additional properties that are "sometimes true" that systems engineers are likely to encounter are as follows:

1. A system may exist independent of human intentionality.
2. A system may be part of one or several wider "containing systems."
3. A system may be self-sustaining, self-organizing, and dynamically evolving (such systems include "complex adaptive systems").
4. A system may offer "affordances"—features that provide the potential for interaction by "affording the ability to do something" (Norman, 1990):
   - Affordances will lead to interactions whether planned or not. For example, the affordance of a runway to let planes land and take off also leads to a possibly unintended affordance to drive vehicles across it, which may get in the way of planes, leading to undesirable emergent whole-system behavior.
5. A system may be:
   - Clearly bounded and distinct from its context (the solar system, Earth, planes, trains, automobiles, ships, people)
   - Closely coupled with or embedded in its context (a bridge, a town, a runway, the human cardiovascular system, the Internet)
   - Of fluid and dynamic makeup (a club, team, social group, ecosystem, flock of geese, and again the Internet)
6. A system may be technical (requiring one or multiple disciplines to design), social, ecological, environmental, or a compound of any or all of these.

These lists help explain the need for systems engineers to think about the SOI in its wider context, so as to ensure they understand both the properties important to the system's purpose and those that might give rise to undesirable unintended consequences.

2.10 SYSTEMS ENGINEERING LEADERSHIP

Many of the processes in this handbook rightly discuss management (e.g., decision management, risk management, portfolio management, knowledge management), and these are all important aspects of the SE process. However, leadership is an equally important topic to systems engineers. In a paper entitled "What Leaders Really Do," J. P. Kotter (2001) states that "leadership is different from management, but not for the reason most people think." Kotter defines the key differences between leaders and managers as:

p. 11

- Coping with change versus coping with complexity
- Setting a direction versus planning and budgeting
- Aligning people versus organizing and staffing
- Motivating people versus controlling and problem solving

A quote often attributed to Peter Drucker is: "Managers do things right. Leaders do the right things." Compare this to the informal definitions of the SE verification and validation processes: "Verification ensures you built the system right. Validation ensures you built the right system." Both verification and validation are important for the development of systems. Likewise, both management and leadership are important for systems engineers and their teams. Different phases of a project demand emphasis on different aspects of leadership.

Aspects of leadership that are particularly relevant for systems engineers include:

- Thinking strategically and looking at the long-term implications of decisions and actions to set vision and course
- Seeing the "big picture"
- Casting or capturing the vision for the organization and communicating it (the systems engineer may be working in support of the identified leader, or sometimes, it isn't the leader's prerogative to "cast" the vision)
- Defining the journey from the "as is" of today to the "to be" of tomorrow
- Turning ambiguous problem statements into clear, precise solution challenges for the team
- Working with the stakeholders (including customers), representing their points of view to the team and the team's point of view to them
- Maximizing customer value by ensuring a direct tie of all engineering effort to the customer business or mission needs
- Establishing an environment for harmonious teams while working to leverage the potential benefits of diversity (including bridging cultural and communication differences in multidisciplinary teams)

p. 12 p. 22

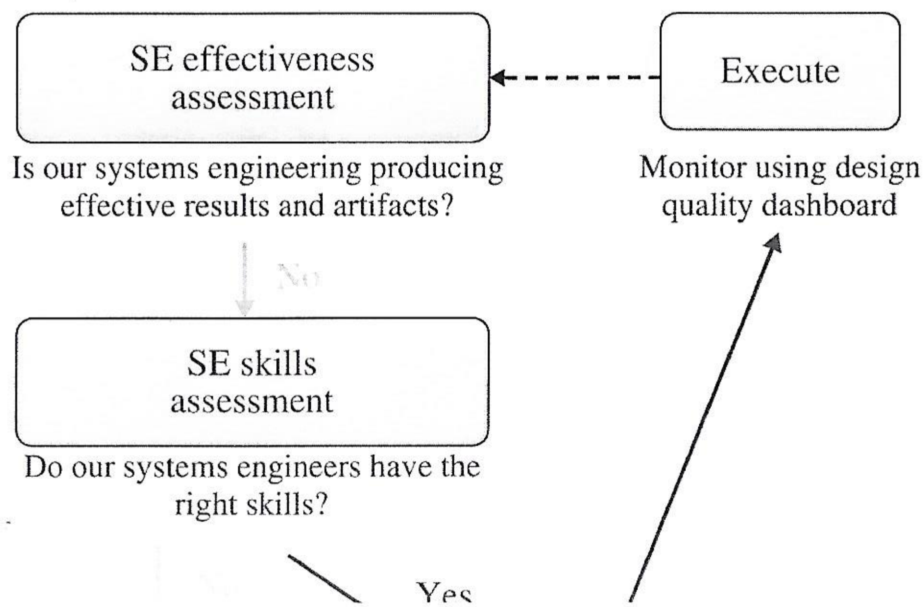
- Challenging conventional wisdom at all levels
- Managing conflicts and facilitating healthy conflict around ideas and alternatives
- Facilitating decision making
- Demanding and enabling excellence

p. 13

Leadership is both an opportunity and a critical responsibility of the systems engineer. The SE leader must have a systems view that takes into account the context, boundaries, interrelationships, and scope. They drive better solutions through the holistic understanding of the problem and its context and environment. SE leaders highlight the risks of unintended consequences in a proactive manner. Many times, SE leaders need to move the conversation from "price and cost" to "value and ROI." SE leaders need to serve as a model for the adaptability, agility, and resilience that is sought in both the systems and the teams that develop them. After all, SE leaders have the "best seat in the house" for seeing the broader systems view (Long, 2013).

2.11 SYSTEMS ENGINEERING PROFESSIONAL DEVELOPMENT

To efficiently and cost-effectively deliver differentiated products to the market, an organization needs to know what gaps exist in their overall capability. An individual needs to know what skills would enable them to be more effective, to develop those skills, and to have a standard to demonstrate and communicate their skill levels. The overall system for optimizing SE delivery is shown in Figure 2.9.



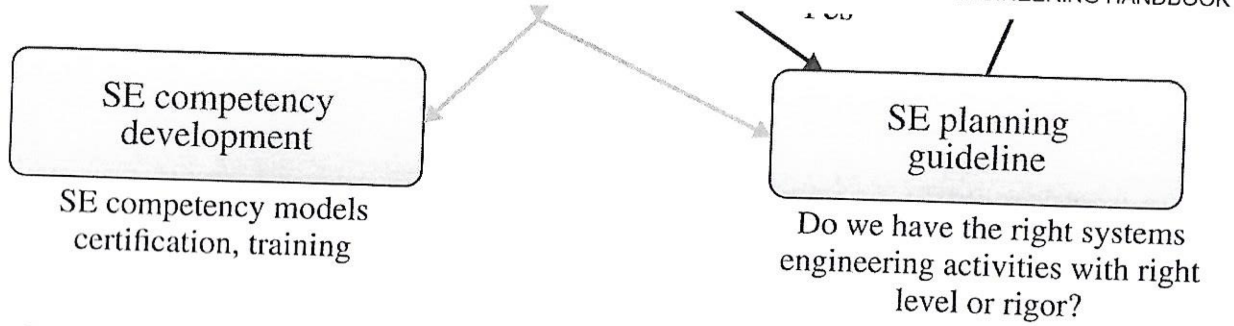


Figure 2.9 SE optimization system.

Most development, but especially for SE, is achieved through experience and on-the-job training. Typically, 70% of development is achieved through experience, 20% through mentoring, and only 10% through training (Lombardo and Eichinger, 1996). Training creates an

p. 23