Shaping Socio-technical System Innovation Strategies using a Five Aspects Taxonomy

Donna H. Rhodes and Adam M. Ross
Massachusetts Institute of Technology
Systems Engineering Advancement Research Initiative
77 Massachusetts Avenue E38-572
Cambridge, Massachusetts 02139 USA
seari@mit.edu

Abstract. This paper introduces and describes a *five aspect taxonomy* for the engineering of complex systems, including structural, behavioral, contextual, temporal and perceptual aspects. The taxonomy has proven useful for (1) characterizing methods to ensure coverage of essential aspects of engineering complex systems; (2) providing a focusing framework to develop and select systems engineering innovation strategies; and (3) providing an organizing structure for classifying methodological research projects. Each of the five aspects is described, and the taxonomy is used to discuss recent and ongoing research on innovation strategies within and across the five aspects.

Introduction and Motivations

Modern socio-technical systems exhibit complexities in multiple dimensions, driving the need for innovation strategies beyond those used in non-complex product engineering. Innovation in the systems context involves a broad perspective and new ways of thinking beyond traditional systems engineering practice (Rhodes and Hastings 2004). Innovation in complex systems may occur at multiple levels of the system (component, subsystems, systems and system of systems levels). As such, any innovation at the component level will likely impact system behavior at the broad system level in some manner. Analysis of localized impact must be carefully assessed for secondary impact on the entire system and its environment. In complex systems the technological system and the enterprise system are intimately interconnected, and therefore innovation in one will certainly impact the other. For example, a new organizational structure in a company developing a product system is likely to influence innovations in the product system architecture itself. Another important consideration in regard to complex systems is that innovation at the interfaces is equally or more important than at the component level.

In developing socio-technical systems, it is important to realize that decisions should be made in a manner that will set up the *possibility* for innovation in future. Any decision must be carefully made to enable options and possibilities such as downstream changes or insertion of new technologies. Innovators need to think across multiple dimensions with sensitivity to time, context, and the diverse preferences of large sets of system stakeholders. As complexity increases, the difficulty of identifying innovation strategies grows but there is a corresponding increase in the potential value of individual innovations as these can impact the larger whole.

Wheatley (1999) describes innovation as being “fostered by information gathered from new connections; from insights gained by journeys into other disciplines or places; and from active, collegial network and fluid, open boundaries.” She notes that innovation arises from ongoing circles of exchange, where information is not just accumulated or stored, but created. The result is that “knowledge is generated anew from connections that weren't there before.” The potential for innovation in socio-technical engineering systems is significant at the intersection of opportunities, capabilities, and strategies. Opportunities arise from several avenues such as new challenges, new markets, new societal needs, new threats, and other dynamic factors. Capabilities are enabled by availability of new technologies or new knowledge. Strategies are means to address new opportunities through new capabilities, which may involve new integrations, new knowledge or information synergies, new processes, or new methods. Systems engineering performs important work at these intersections of opportunity, capability, and strategy, as enabled by socio-technical innovation strategies.

In the context of this paper, *socio-technical innovation strategies* refer to systems engineering methodological approaches (constructs, enablers, supporting guidance and training) that can be applied to the design and development of complex engineering systems and associated enterprises. Complex systems and enterprises inherently have many dimensions and, accordingly, necessitate multi-dimensional methodological approaches. The five aspects taxonomy framework is presented as a useful organizing construct to reflect on research outcomes, as well as possibilities for combining methods and achieving research synergies.

Research Landscape

The ability to undertake research on the development of socio-technical innovation strategies requires an appropriate research landscape to promote transdisciplinary and cross-domain inquiry. The field of *engineering systems* serves as a highly suitable landscape for socio-technical engineering research (Rhodes and Hastings 2004), and is characterized by:

1. A broad interdisciplinary perspective: technology, policy, management, social science.
2. Intensified incorporation of system lifecycle properties, or "ilities", such as sustainability, safety, survivability and flexibility.
3. An emphasis on an enterprise perspective, acknowledging the interconnectedness of the "product system" with the enterprise system that develops and sustains it.
4. A complex synthesis of stakeholder perspectives, of which there may be conflicting and competing needs that must be resolved to serve the highest order system need.

Given the significant number of innovations possible within a socio-technical system, a focusing framework is a useful construct for inquiry, for challenges of both technological systems and enterprises (Rhodes, Ross and Nightingale 2009).

Five Aspects Taxonomy

Taxonomies are useful for categorizing information in order to share knowledge with others. The five aspects taxonomy discussed in this paper originated in an effort to characterize engineering complexity, and subsequently the authors found the classification to a useful lense for organizing a collection of past and ongoing research on socio-technical innovation strategies. The five aspects are: structural, behavioral contextual, temporal and perceptual. The framework itself is not intended to be a novel contribution to the systems engineering field, but rather is a basic frame to understand facets of innovation strategies and communicate research on emerging methods. Historically, while all the aspects have been long received some level of attention in systems engineering methods (for example, Hall 1989), the structural and behavioral aspects have dominated. More recently, architecture frameworks (UK MOD 2008, US DoD 2009) have reached beyond technical system considerations to broader enterprise considerations.

In initial experiences with the taxonomy as a classification tool, research outcomes were categorized by aspect. As experience has grown, it was observed that socio-technical innovation strategies were more likely at the intersection of two or more aspects. For example, model-based systems engineering can be placed at the intersection of structural and behavioral aspects. Rhodes and Ross (2010) discuss a brief history of combining these aspects over the past two decades, as illustrated in Table 1 below. The descriptive model of Karas and Rhodes (1987) included functional and physical (structural) and operational (behavioral) views. A decade later, prescriptive approaches appeared in the literature, for example, Oliver et al. (1997) discussed the importance of both structure and behavior in systems development, providing foundational work for the rapid evolution in model-based systems engineering methods that have followed. Another decade later, INCOSE (2007) published a survey of six leading model-based systems engineering methods and Oliver et al. (2009) publish a chapter in a systems engineering handbook.

Table 1: Example Initiatives in Evolution of Model-Based Systems Engineering

Emergence of Model-Based Systems Engineering (examples initiatives)	
1987	Descriptive method with function and physical (structural) and operational (behavioral) views, implemented in early computer based environment L. Karas, and D.H. Rhodes, Systems Engineering Technique, *Design, Development and Testing of Complex Avionics Systems: Conference Proceedings*, 1987
1997	Prescriptive approach for engineering complex systems using structural and behavioral system models D. Oliver, T. Kelliher, and J. Keegan,, *Engineering Complex Systems with Objects and Models*, NY: McGraw Hill, 1997
2007	Initial publication of INCOSE Survey of six leading MBSE methodologies with enabling toolset environment INCOSE TD-2007-003-01, Survey of Model-Based Systems Engineering Methodologies, 10 June 2008

The structural and behavioral aspects are evident in basic systems engineering analysis and synthesis practice, but additional aspects (including contextual, temporal and perceptual) appear to be increasingly important due to the complexities of modern systems and enterprises, as enumerated and defined in Table 2 below.

Table 2: Five Aspects Defined and State of Art/Practice

STRUCTURAL *related to the form of system components and their interrelationships*	Fundamental to the current “State of the Practice” for systems architecting and design, and model-based systems engineering
BEHAVIORAL *related to performance, operations, and reactions to stimuli*	
CONTEXTUAL *related to circumstances in which the system exists*	Sources for new constructs and methods for advancing “state of art”. Examples include: *Epoch Modeling* *Multi-Epoch Analysis* *Epoch-Era Analysis* *Multi-Stakeholder Negotiations* *Visualization of Complex Data Sets*
TEMPORAL *related to dimensions/ properties of systems over time*	
PERCEPTUAL *related to stakeholder preferences, perceptions and cognitive biases*	

While the set of aspects may not be complete, the taxonomy is meant to be holistic and to enable cross-aspect inquiry. Each of the five aspects includes a rich set of constructs and considerations. Table 3, while not comprehensive, provides examples.

Table 3: Example Constructs and Considerations within the Aspects

aspect	**Examples of constructs and considerations**
STRUCTURAL	• heterogeneous/homogenous components/constituents • elaborate networks, loose and tight couplings • layers, vertical/horizontal structures, multiplicity of scales
BEHAVIORAL	• variance in response to stimuli • unpredictable behavior of technological connections • emergent social network behavior
CONTEXTUAL	• many complexities and uncertainties in system context • political, economic, environmental, threat, market factors • stakeholder needs profile and overall worldview
TEMPORAL	• decoupled acquisition phases and context shifts • systems with long lifespan and changing characteristics • time-based system properties (flexibility, survivability, etc.)
PERCEPTUAL	• many stakeholder preferences to consider • perception of value shifts changes with context shifts • cognitive constraints and biases

In the sections that follow, each of the five aspects is discussed in further detail, with illustrative examples of recent methodological outcomes.

Structural Aspect

The *structural aspect* in the taxonomy relates to the physical and logical components of the system, and their interrelationships. Complex socio-technical systems are comprised of various structures, and contain elaborate networks. These networks may have loose and tight couplings between the network nodes. These networks may be physical system networks, information system networks, human networks, and any combination of these. Structural aspects also relate to architectural forms beyond traditional systems hierarchical structures, for example layers and vertical/horizontally integration. Multiplicity of scales is also a prevalent characteristic of complex socio-technical systems.

Innovation related to the structural aspect of systems has traditionally involved new materials and technologies, developed through planned evolutionary approaches or sometimes through disruptive means. As systems become more complex, the designer has opportunities for innovation that stem from the leverage at the interfaces, clever ways to combining existing components and features, and cost effective ways to add components and features to leverage existing ones.

Discovery of Latent Value through Attribute Classes. Ross (2006) described *attribute classes* to assist the system designer in understanding perceived value in context of an overall value spectrum, ranging from articulated value to multiple dimension of unarticulated value. Using attribute classification, the desired system attributes are characterized using several value classes including: articulated value, free latent value, combinatorial latent value, accessible value and inaccessible value. The classification scheme enables a deep exploration of system attributes as logical structures of the system, and is designed to uncover latent value during the conceptual system design phase. Attribute classification results in logical structures that help to ensure that system designers account for future changed value perceptions by thinking about these attributes according to ease by which the system can display them. Since attributes can be on function (logical structure) or form (physical structure), to "display" an attribute means that the system "does" or "exhibits" that attribute. For example, an attribute can be the color of the system, or the spatial resolution of generated images. The cost to display these attributes is how much it takes to either have or change the color, or to have or change an image spatial resolution. The attribute class spectrum from least to most costly ranges from articulated "designed for", class 0 attributes, to inaccessible value, class 4 attributes as shown in Table 4 below.

Table 4: Attribute Classes

Class	Attribute Class Name	Property of Class	"Cost" to Display
0	Articulated Value	Exist and assessed in initial design	0
1	Free Latent Value	Exist, not assessed in initial design	0
2	Combinatorial Latent Value	Can exist through recombining class 0 and 1 attributes	Small
3	Accessible Value	Can be added through changing the system design variable set (scale or modify initial system design)	Small→large
4	Inaccessible Value	Cannot be feasibly added through changing the design variable set	Large→infinite

Once elaborated through using the attribute classes, the designer can look to create "combinatorial value" to takes advantage of design form and function already in the system. Ross and Rhodes (2008c) describes an illustrative example for a cell phone, where this attribute classification can be used by designers to assist in generating innovations through deriving new attributes through combinations of attributes in the initial cell phone design. In an example, combining class 0 and class 1 attributes for the cell phone led to the ability to provide custom ringtones by uncovering latent value (class 2). To access new value (the case of class 3), the design can be feasibly and affordably modified to leverage accessible value as shown in an example where a "child tracking" cell phone service was made possible through modified phone design to leverage navigation technology.

Behavioral Aspect

The behavioral aspects of socio-technical systems relate to the performance and operations of the technological system, and the enterprise that develops and operates it. Behavioral aspects relate to the response that a system has to internal and external stimuli, and the variances related to that response. In complex systems with large numbers of interconnections, the likelihood of unpredictable behavior is significant. Model-based systems engineering, employing executable models, is allowing engineering teams to better understand such behavior while systems are still in the design phase.

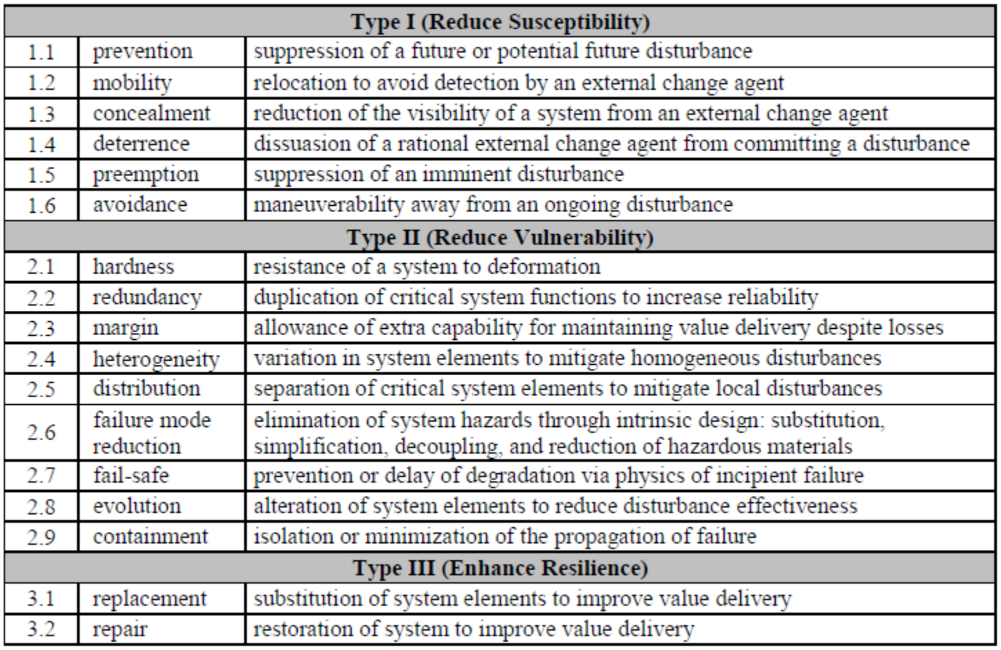
As humans and organizations are an integral part of engineering systems, there are emergent social network behaviors that occur. Management innovations and interventions can be used to foster more effective systems engineering behavior in the enterprise to influence resulting technological system outcomes. For example, a six year research effort has developed systems engineering leading indicators, or measures for predictability of effective systems engineering performance, used to monitor and take corrective action (Rhodes et al. 2007). In other recent research, success factors for collaborative systems thinking have been identified through empirical research (Lamb and Rhodes 2008, Lamb and Rhodes 2009), which can be leveraged in managing engineering design teams to foster systems behavior.

In regard to the technological system design, an interesting area of inquiry related to the behavioral aspect involves behavioral system properties, sometimes called “ilities” (Ross, Rhodes and Hastings 2009). As an example, survivability is a desired “ility” in many types of systems facing finite disturbances that impact delivery of value to stakeholders. Empirical research on the property of survivability, described below, resulted in architectural principles that can be applied to design systems that exhibit active and/or passive survivability behavior. Further, the approach has been used to foster concept generation in development of survivable systems.

Principles for Architecting for Survivability. Richards (2009) defines survivability as “the ability of systems to minimize the impact of finite- duration disturbances on value delivery.” This research empirically derived and tested a set of seventeen survivability design principles (see Table 5 below), spanning susceptibility reduction, vulnerability reduction, and resilience enhancement strategies.

The work has resulted in a process for applying the survivability design principles to the concept generation phase of multi-attribute tradespace exploration. Applying the design principles augments the creativity of system designers by ensuring consideration of a broad tradespace of design alternatives, and also helps designers to quickly screen a large number of candidate designs prior to concept evaluation (Richards et al. 2009).

Table 5: Survivability Design Principles (source: Richards 2009)



Type I (Reduce Susceptibility)		
1.1	prevention	suppression of a future or potential future disturbance
1.2	mobility	relocation to avoid detection by an external change agent
1.3	concealment	reduction of the visibility of a system from an external change agent
1.4	deterrence	dissuasion of a rational external change agent from committing a disturbance
1.5	preemption	suppression of an imminent disturbance
1.6	avoidance	maneuverability away from an ongoing disturbance
Type II (Reduce Vulnerability)		
2.1	hardness	resistance of a system to deformation
2.2	redundancy	duplication of critical system functions to increase reliability
2.3	margin	allowance of extra capability for maintaining value delivery despite losses
2.4	heterogeneity	variation in system elements to mitigate homogeneous disturbances
2.5	distribution	separation of critical system elements to mitigate local disturbances
2.6	failure mode reduction	elimination of system hazards through intrinsic design: substitution, simplification, decoupling, and reduction of hazardous materials
2.7	fail-safe	prevention or delay of degradation via physics of incipient failure
2.8	evolution	alteration of system elements to reduce disturbance effectiveness
2.9	containment	isolation or minimization of the propagation of failure
Type III (Enhance Resilience)		
3.1	replacement	substitution of system elements to improve value delivery
3.2	repair	restoration of system to improve value delivery

Contextual Aspect

The contextual aspect of socio-technical systems pertains to the environmental circumstances in which the technological system, and its associated enterprise, exists. Socio-technical systems typically face many uncertainties over their lifespan, related to political, economic, cultural, threat, and market factors. Researchers across a wide variety of fields have begun to give more attention to context and contextual drivers. In the field of behavioral science, Malmsjo and Ovelius (2003) conducted an empirical investigation, including more than 150 interviews, that resulted in identification of external factors for change: type of work performed, working climate, time at disposal, competition, technology, laws and regulation, economy, feedback, environmental demands and expectations, culture, and conflicts. Another example is Woods and Rozanski (2009), who describe their realization, in light of experience, that a prior six viewpoint software architectural framework was incomplete, and resulted in the addition of a "contextual view."

Systems engineering has traditionally considered "context" as a fixed external environment in which a system exists, imposing constraints on it. In recent times, contextual factors have been viewed more as uncertainties that may sometimes be leveraged or changed, rather than considered purely constraints. In systems of systems, the relationship of constituent system contexts and resulting SoS context is particularly complex (Shah, Rhodes and Hastings 2007). Research related to the contextual aspect has included approaches to modelling context and incorporating this in tradespace exploration (Ross 2006), and specific approaches such as real options to address uncertainties (Mikaelian et al. 2008, Mikaelian 2009). Mechanisms to alter the impact of undesired contextual factors have also been developed, as discussed below.

Insulating a System from its Context using the System Shelter. The concept of system shell provides a value robust construct for mitigating effects of changes in context and expectations by decoupling the system from the external sources of change for that system. It is not always a viable or affordable approach to make changes to the fundamental system structure or function as stakeholder value expectations shift. As shown in the figure below, the system shell is comprised of two layers: the inner shell, or "shelter," and the outer shell, or "mask," each serving a unique purpose in insulating a system from changes in its context. The system shelter changes how the system "sees" its external context and stakeholders. The system mask changes how the system "is seen" by the external context and stakeholders.

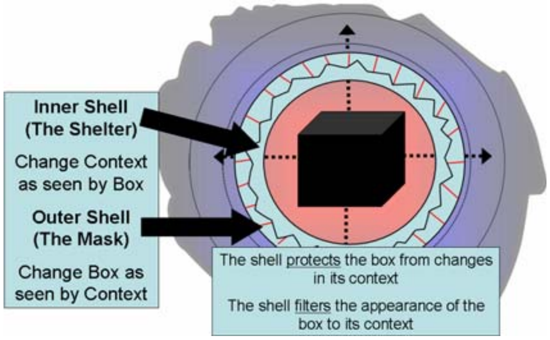


Figure 1: System Shell Construct (Ross 2006)

The shelter concept provides an alternate design approach to deliver value by making the system insensitive to external changes, instead of directly adapting the fundamental system for the external change. The shelter prevents the system from experiencing changes in its context, and requires a protective mechanism. Using the system shelter construct, the designer can investigate innovation strategies that make the system insensitive to context changes. Ross and Rhodes (2007) cite illustrative examples: the use of use of protective films on building window glass to reduce solar penetration into internal rooms; radiation shielding for people and equipment isolating people and equipment from the radiation; and firewalls serving to isolate computers from malicious streams of data on networks.

Temporal Aspect

Temporal analysis relates to the dimensions and properties of systems over time. Accordingly, addressing socio-technical system challenges involves responsiveness to changing needs and operational environments in a dynamic world. The need for anticipatory analysis is essential, and innovations in both methods and enabling environments are important for performing advanced systems engineering (Rhodes and Ross, 2009). Insights from other fields inform the development of temporally-based methods, for example: scenario analysis (Kazman et al. 2000), environmental scanning (Choo 2001); morphological analysis (Ritchey 1997), and scenario planning (Schwartz 1991). Traditional systems engineering has fallen short in addressing the temporal aspect, with limited analysis methods for understanding how systems perform over time. Two decades ago Hall (1989) described the challenge of anticipatory temporal analysis: “a forecast is daunting because it encompasses a comprehensive description of the environment from before the time of conception of a new system, through every period of its lifecycle, to its ultimate demise”.

Epoch-Era Analysis for Anticipating Future System Timelines. Epoch-Era Analysis is a time-based analysis method used for conceptualizing system timelines using natural value-centric timescales wherein the context itself defines the timescales (Ross and Rhodes 2008a). Whereas system lifecycle phases shift on a determined schedule, epoch shifts occur naturally and may be anticipated or unexpected. An epoch shift may be due to such factors as political change, availability of new technologies, policy change, and others. Epoch-Era Analysis has been applied to the evaluation of technological systems as a visualization and communication approach and also more rigorously in tradespace exploration, as shown in Figure 2 below. The approach has the potential to enable system designers to think in a more continuous and anticipatory manner in a world that demands a system to match the cadence of a changing environment. In this method, each epoch is defined as a period of fixed context and stakeholder needs. The set of possible epochs for a system can be elaborated, and multi-epoch analysis performed. From this set, particular time-ordered series of epochs can be constructed to form “eras” representing possible sequences of contexts and needs facing a system across its lifespan (Roberts et al 2009). More recently, epoch-based thinking has been applied to enterprises (Rhodes, Ross and Nightingale 2009).

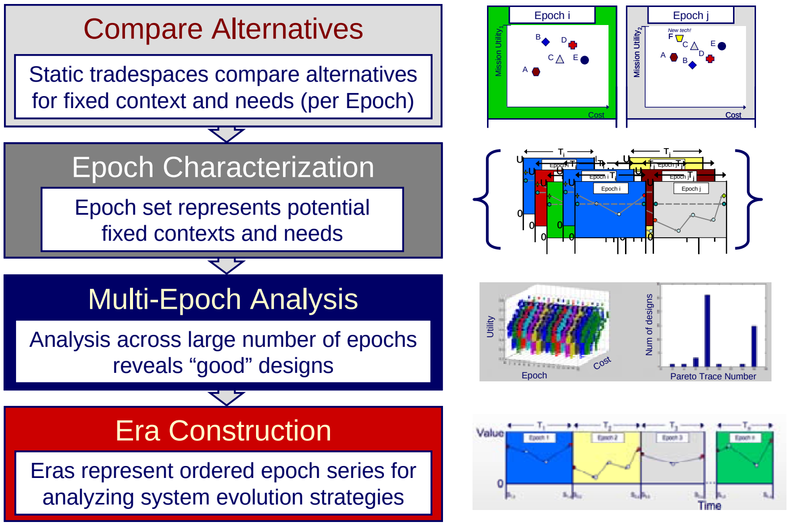


Figure 2: Epoch-Era Analysis (Ross 2006)

Perceptual Aspect

The perceptual aspect relates to stakeholder expectations and their perception of the world as limited by cognitive biases. Given that complex systems have many stakeholders, there are typically varied preference sets. Each individual stakeholder’s preferences may shift over the system lifespan as they experience the system, as well as when system contexts shift. Cognitive constraints in human decision making are one consideration, for example, where constructs are needed to effectively communicate complex data sets for appropriate, rapid processing and interpreting for decision-making.

Perceptual aspects also come into play when selecting and implementing innovation strategies, where cultural or demographic-related biases come into play. For example, Ogawa and Rhodes (2009) examined the impacts of culture on adoption of effective innovation strategies across organizations. The research found that the highly effective concept design methods and tools used in a western culture could not be easily adopted in an eastern culture due to subtle perceptual-based differences related to how they approached decisions, setting baselines, and making design changes.

Altering the Perception of a System using a System Mask. As described earlier in the contextual aspect section, the system shell is a construct for mitigating effects of changes in context and expectations by decoupling the system from the external sources of change. The shell construct may include the shelter, or the mask. The mask changes the system as seen by its context, that is, it “masks” the true system thereby preventing the need to change the system to meet external changing perceptions of stakeholders. As described in Ross and Rhodes (2007), the mask is often used in consumer products as a strategy to address the need for satisfying diversity of stakeholder stylistic preferences. Simple examples, as cited in the earlier paper, include product designs with many variations of faceplates such as a Swatch™ watch or Nokia™ phone, where an underlying core architecture is used in a family of products which appear to the customer to be different products. The use of “perceptual filters” can change the stakeholder’s experience with a system without truly changing the system. As systems become increasingly complex, such innovation strategies are likely to become increasingly important to meet the needs of diverse stakeholder sets that may shift over time.

Innovations through Combination and Synthesis

Examining the innovation strategies applicable to each of the five aspects has been valuable in assessing an overall research portfolio. However, it has been observed by the authors that innovation strategies can be generated through considering the combination of two or more aspects, and synthesis across the five aspects.

Responsive Systems Comparison for Developing Dynamically Relevant Systems. The Responsive Systems Comparison (RSC) method is a seven process method for system concept generation, evaluation, and selection, resulting from a decade of research that has been combined and evolved. The seven processes are: (1) Value-driving Context Definition; (2) Value Driven Design Formulation; (3) Epoch Characterization; (4) Design Tradespace Exploration; (5) Multi-Epoch Analysis; (6) Era Construction; and (7) Lifecycle Path Analysis. The method has been developed using recently evolved constructs and methods, and is described in detail by Ross, McManus et al. (2008, 2009). The seven processes of RSC are used to conduct a tradespace exploration study to assess a large number of system design alternatives on a common value-centric basis across changing contexts and needs. The goal of RSC is to generate knowledge about tradeoffs, compromises, and risks to a system development project, and identify system concepts that are actively and/or passively value robust (using either changeable and/or versatile strategies). RSC is applied to identify system design concepts that provide desired performance over time at less cost. The strength of the method is that it enables dialogue and knowledge building among system designers and stakeholders. RSC resulted from two foundational research areas addressing different aspects.

The first area of foundational work resulted in the tradespace construct and an approach for populating and exploring tradespaces. This work evolved into a method of dynamic *Multi-Attribute Tradespace Exploration* (MATE), used to explore tradespaces over multiple time periods (Ross, 2006). Dynamic MATE method involves structural and behavioral aspects, as well as the perceptual aspect as it reveals patterns in stakeholder perceived system value.

A second major area of research, *Epoch-Era Analysis*, addresses the need to consider systems (and how they provide value to stakeholders) in context of a dynamic world (Ross 2006, Ross and Rhodes 2008). In Epoch-Era Analysis, the system lifespan is divided into a series of epochs, defined as 'naturally occurring' time periods where significant stakeholder needs and the system context are fixed. The method aids the understanding of how the various design concepts perform within an epoch, modelled using a set of epoch variables. Analysis can be performed to understand how the system concepts perform in the set of possible epochs the system may experience in its lifetime. Multiple sequential epochs can be strung together to create an era (or scenario) in order to analyze possible system evolution paths.

RSC uses a value-based approach to elicit stakeholder needs, and these are used by system designers to define design vectors of possible system concepts satisfying these needs. Together, the needs (quantified through multi-attribute utility) and design vectors are used to computationally generate tradespaces for exploring thousands of design alternatives through a model-based approach. The method can be applied to a single epoch, but is even more powerful when used to quantitatively perform temporal analysis of the versatility and changeability of designs across multiple possible futures. These anticipated epochs can be sequenced into a "system era" through scenario analysis approaches in order to gain insight into how the design alternatives perform across shifts in contexts and needs over time. The seven processes of RSC are shown in Table 6 below.

Table 6: Responsive Systems Comparison (RSC) Processes

RSC Method – Description of the Seven Processes (Ross et al. 2009)	
1. Value-Driving Context Definition	The purpose of the Value-Driving Context Definition process is to identify, understand, and capture the overall system concept(s) value propositions, key stakeholders, and their interactions.
2. Value-Driven Design Formulation	The purpose of the Value-Driven Design Formulation process is to characterize specific stakeholder needs and formulate possible system concept solutions.
3. Epoch Characterization	The purpose of the Epoch Characterization process is to parameterize the range of contextual uncertainties in the form of Epochs.
4. Design Tradespace Evaluation	The purpose of the Design Tradespace Evaluation process is to gain an understanding, via modeling and simulation, of how key systems trades and concepts fulfill the overall value-space (attributes) in response to contextual uncertainties (Epochs).
5. Multi-Epoch Analysis	The purpose of the Multi-Epoch Analysis process is to identify value robust system designs.
6. Era-Construction	The purpose of the Era Construction process is to develop Era timelines from the underlying Epochs.
7. Lifecycle Path Analysis	The purpose of the Lifecycle Path Analysis process is to develop near- and long-term system value delivery strategies in response to time-dependent contextual uncertainties.

Figure 3 illustrates the concept of an era, with five sequential epochs of varying time periods. Each tradespace frame in the sequence is for a fixed epoch of indicated duration. Any individual candidate design concept (represented as a point in the tradespace plot) is likely to shift in performance (y axis) for cost (x axis) as the context and needs (as modeled using epoch variables) change. Using this analysis, designers can identify versatile designs that do well across all epochs, as well as designs that can be affordably changed to accommodate the epoch shift. In the figure it can be seen that "design 3435" moves about the tradespace when considered across a series of changing epochs, remaining a design of interest (high value for cost).

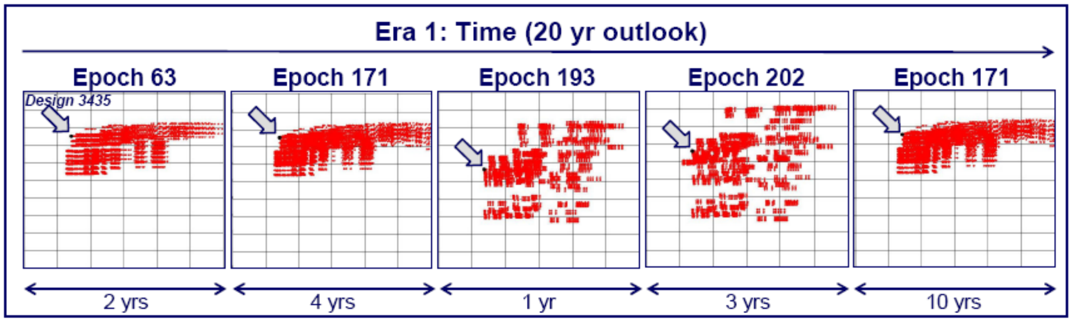


Figure 3: Using RSC Methods to Assess Design Performance Across Epoch Shifts

The RSC method serves to help identify innovative system design concepts that perform well across time, or can do well if changed appropriately in response to the epoch shift. There are several metrics that have been developed to measure this passive value robustness (versatility of designs) and active value robustness (changeability of designs). These metrics include:

- **Normalized Pareto Trace (NPT)** for a particular system design is a metric for the passive value robustness of that design in a given set of epochs. To determine the Normalized Pareto Trace number, the Pareto efficient set of designs for each epoch is calculated, and the relative frequency of occurrence of the designs in the superset constructed across all the Pareto sets is determined. A high relative frequency (high Normalized Pareto Trace), indicates that a design is value robust over many changes in the system context. Described in (Ross, Rhodes, and Hastings 2009), other forms of this metric include the Fuzzy Pareto Trace (allowing for the inclusion of slightly "suboptimal," off-Pareto designs) and the Normalized Fuzzy Pareto Trace.
- **Filtered Outdegree (FOD)** is a changeability metric for a particular system design, and represents the number of possible change paths from that design, given cost and time constraints. Highly changeable designs (having a high filtered outdegree) indicates a potentially actively value robust design. A list of highly changeable designs can be obtained using the FOD metric (Ross, 2006). A further evolution of this metric combines Pareto Trace with Filtered Outdegree to identify "valuable flexible designs" across epoch pairs (Viscito et al, 2009).

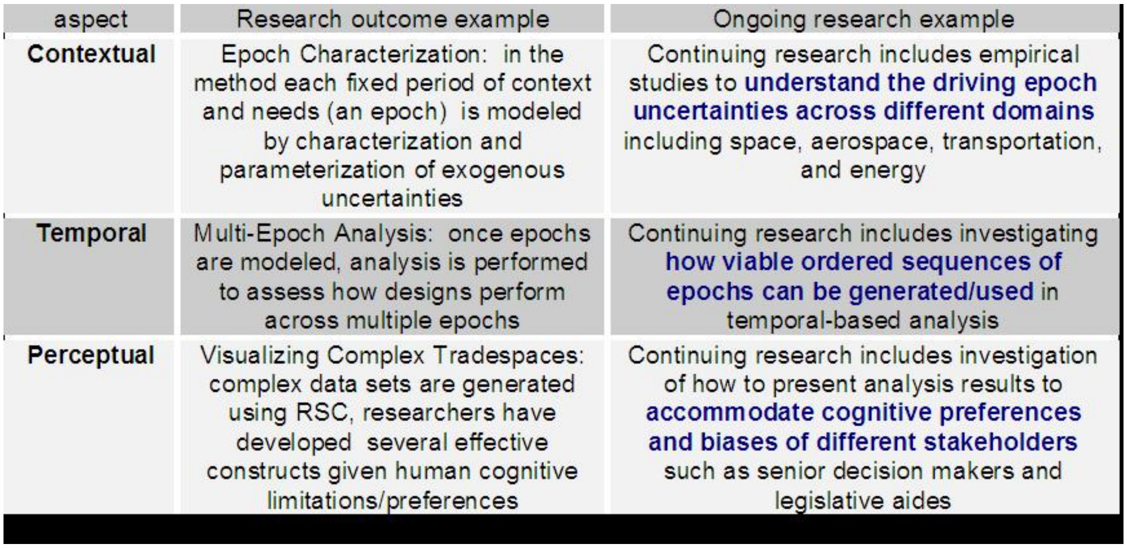
Development of useful metrics remains an open area of research for RSC. Lifecycle path analysis (process 7) is the most active area of methodological research in development of the RSC method. Contemporary challenges drive the critical importance of understanding value delivery strategies in response to time-dependent contextual uncertainties. In evolving this process, the RSC research team is examining the usefulness of previously developed innovation strategies across the five aspects, as well as investigating work by other researchers.

Future Directions and Conclusions

The five aspects taxonomy has been a useful framework for examining the contributions of individual innovation strategies in assessing an overall research program. More importantly, it has motivated an approach to look for innovation strategies at the intersection of two or more aspects. The RSC method has been described as an example of developing a significant innovation strategy through synthesis of multiple research outcomes across the five aspects, which has been supplemented with prescriptive guidance materials and training materials.

Table 7 shows examples of RSC research outcomes with respect to the contextual, temporal, and perceptual aspects, along with examples of how this method is being extended with new research.

Table 7: Continuing Research on RSC Method



aspect	Research outcome example	Ongoing research example
Contextual	Epoch Characterization: in the method each fixed period of context and needs (an epoch) is modeled by characterization and parameterization of exogenous uncertainties	Continuing research includes empirical studies to **understand the driving epoch uncertainties across different domains** including space, aerospace, transportation, and energy
Temporal	Multi-Epoch Analysis: once epochs are modeled, analysis is performed to assess how designs perform across multiple epochs	Continuing research includes investigating **how viable ordered sequences of epochs can be generated/used** in temporal-based analysis
Perceptual	Visualizing Complex Tradespaces: complex data sets are generated using RSC, researchers have developed several effective constructs given human cognitive limitations/preferences	Continuing research includes investigation of how to present analysis results to **accommodate cognitive preferences and biases of different stakeholders** such as senior decision makers and legislative aides

Taxonomies are descriptive tools that can enable research dialogue, collaboration and knowledge sharing of research outcomes across multiple research communities. In this paper five aspects of socio-technical systems have been identified for classifying and evolving socio-technical systems innovation strategies. The further use of the taxonomy as a thinking construct and as an organizing framework for methodological research outcomes may lead to the discovery of additional important aspects, as well as new research directions.

Acknowledgement

The authors gratefully acknowledge funding for this research provided through MIT Systems Engineering Advancement Research Initiative (SEAri, http://seari.mit.edu) and its sponsors including the US Government and the Singapore Defence Science & Technology Agency.

References

Choo, C. 2001. Environmental scanning as information seeking and organizational learning", *Information Research*, Vol. 7 No. 1, Oct.

Hall, A.D. *Metasystems Methodology*, Oxford, England, Pergamon Press, 1989.

Karas, L. and Rhodes, D. 1987. Systems engineering technique. Design, development and testing of complex avionics systems: Conference Proceedings.

INCOSE. 2003. INCOSE-TD-2007-003-01. Survey of Model-based Systems Engineering Methods.

Kazman, J., Carriere, S. and Woods, S. 2000. Toward a discipline of scenario-based architectural engineering, *Annals of Software Engineering.* Volume 9 Issue 1-4.

Lamb. C.T. and Rhodes. D.H. 2008. Systems thinking as an emergent team property: ongoing research into the enablers and barriers to team-level systems thinking. 2nd Annual IEEE Systems Conference. Montreal. Canada.

———2009. Collaborative systems thinking: uncovering the rules of team-level systems thinking. 3rd Annual IEEE Systems Conference. Vancouver. Canada.

Malmasjo, A. and Ovelius, E., 2003, Factors that induce change in information systems, *Systems Research and Behavioral Science*, 20, pp 243-253.

Mikaelian.T. 2009. *Integrated real options framework for model-based identification and valuation of options under uncertainty*. Doctor of Philosophy Dissertation. Aeronautics and Astronautics MIT.

Mikaelian. T. Rhodes. D.H. Nightingale. D.J. and Hastings. D.E. 2008. Managing uncertainty in socio-technical enterprises using a real options framework. 6th Conference on Systems Engineering Research. Los Angeles. CA.

NY Times. Feb 29 2008. Hoping to make phone buyers flip.

Ogawa. A. and Rhodes. D.H. 2009. Culture: a key factor for implementing te concurrent engineering approach. INCOSE International Symposium, Singapore.

Oliver, D., Andary, J. and Frisch, H. 2009. Model-Based systems engineering, in A. Sage and W. Rouse, *Handbook of Systems Engineering and Management*, Edition 2, New York, NY: John Wiley & Sons, pp. 1361-1400.

Oliver, D. Kelliher. T. Keegan. J. 1997. *Engineering complex systems with models and objects*. New York. N.Y. McGraw Hill.

Rhodes, D.H. and Hastings, D. 2004. The Case for Evolving Systems Engineering as a Field within Engineering Systems. MIT Engineering Systems Symposium.

Rhodes, D.H. and Ross, A.M., 2010, Five aspects of complex systems: enabling constructs and methods, 5th Annual IEEE Systems Conference, San Diego, CA.

______2009. Anticipatory capacity: leveraging model-based approaches to design systems for dynamic futures. 2nd Annual Conference on Model-based Systems. Haifa. Israel.

Rhodes. D.H. Ross. A.M. and Nightingale. D.J. 2009. Architecting the system of systems enterprise: enabling constructs and methods from the field of engineering systems. 3rd Annual IEEE Systems Conference. Vancouver. Canada.

Rhodes. D.H. Valerdi. R. and Roedler. G.J. 2009. Systems engineering leading indicators for assessing program and technical effectiveness," *Systems Engineering*, Vol. 12, No. 1, pp. 21-35.

Richards. M.G. 2009. *Multi-attribute tradespace exploration for survivability*. Doctor of Philosophy Dissertation. Engineering Systems Division MIT.

Richards. M.G. Ross. A.M. Hastings. D.E. and Rhodes. D.H. 2009. Survivability design principles for enhanced concept generation and evaluation. INCOSE International Symposium. Singapore.

Ritchey, T. 1997. Scenario development and risk management using morphological field analysis: research in progress. Proceedings of the 5th European Conference on Information Systems, Volume III, pp. 1053-1059.

Roberts, C.J. Richards. M.G. Ross, A.M. Rhodes, D.H. and Hastings, D.E. 2009. Scenario planning in dynamic multi-attribute tradespace exploration. 3rd Annual IEEE Systems Conference. Vancouver. Canada.

Ross. A.M. 2006. *Managing unarticulated value: changeability in multi-attribute tradespace exploration,* Doctor of Philosophy Dissertation, MIT, June.

Ross. A.M. and Rhodes. D.H. 2008a. Using natural value-centric time scales for conceptualizing system timelines through epoch-era analysis. INCOSE International Symposium 2008. Utrecht. The Netherlands.

———2008b. Architecting systems for value robustness: research motivations and progress. 2nd Annual IEEE Systems Conference. Montreal. Canada.

———2008c. Using attribute classes to uncover latent value during conceptual systems design. 2nd Annual IEEE Systems Conference. Montreal.

——— 2007. The system shell as a construct for mitigating the impact of changing contexts by creating opportunities for value robustness. 1st Annual IEEE Systems Conference. Honolulu. HI.

Ross. A.M. Rhodes. D.H. and Hastings. D.E. 2008. Defining changeability: reconciling flexibility. adaptability. scalability. modifiability. and robustness for maintaining lifecycle value. *Systems Engineering*. Vol. 11. No. 3. 246-262.

———2009a. Using pareto trace to determine system passive value robustness. 3rd Annual IEEE Systems Conference, Vancouver, Canada.

———2009b. Using pareto trace to determine system passive value robustness. 3rd Annual IEEE Systems Conference. Vancouver. Canada.

Ross. A.M. McManus. H.L. Long. A. Richards. M.G. Rhodes. D.H. and Hastings. D.E. 2008. Responsive systems comparison method: case study in assessing future designs in the presence of change. AIAA Space 2008. San Diego CA.

Ross. A.M. McManus. H.L. Rhodes. D.H. Hastings. D. and Long. A.M. 2009. Responsive systems comparison method: dynamic insights into designing a satellite radar system, AIAA Space 2009. Pasadena CA.

Shah, N.B., Hastings, D.E., and Rhodes, D.H. 2007. Systems of systems and emergent system context. 5th Conference on Systems Engineering Research Hoboken, NJ, March.

Schwartz, P. 1991. The Art of the Long View, New York, NY: Doubleday.

US DoD. 2009. Department of Defense Architecture Framework. Version 2.0. Vol 1. May

UK MoD, 2008. Ministry of Defence Architecture Framework. Versions 1.2. September.

Viscito. L. Chattopadhyay. D. and Ross. A.M. 2009. Combining pareto trace with filtered outdegree for identifying valuably flexible systems. 7th Conference on Systems Engineering Research, Loughborough University, UK.

Wheatley, M. 1999. *Leadership and the new science.* San Francisco: Berrett-Koehler Publishers.

Woods, E. and Rozanski, N., 2009, The System Context Architectural View, *IEEE/IFIP WICSA/ECSA, pp 333-336.*

Biography

Donna H. Rhodes, is a senior lecturer and principal research scientist in the MIT Engineering Systems Division. She is co-founder and director of MIT's Systems Engineering Advancement Initiative (SEAri), a research group focused on advancing the theories, methods, and effective practice of systems engineering applied to complex socio-technical systems through collaborative research with industry and government. Dr. Rhodes is a Past President and Fellow of INCOSE, and presently is the director of INCOSE's doctoral student network. She is a recipient of the INCOSE Founders Award and several INCOSE Distinguished Service Awards, and an associate editor for the INCOSE journal Systems Engineering. Prior to coming to MIT, Dr. Rhodes had 20 years of professional and management experience in the aerospace and defense, systems integration, and commercial products industries. She received her Ph.D. in Systems Science from the T.J. Watson School of Engineering at SUNY Binghamton.

Adam M. Ross is a research scientist in the Engineering Systems Division at MIT, and co-founder and lead research scientist for MIT's Systems Engineering Advancement Initiative (SEAri). Dr. Ross has professional experience working with government, industry, and academia including NASA Goddard, JPL, Smithsonian Astrophysical Observatory, Boeing Satellite Systems, Draper Laboratories, US Air Force, selected US government agencies, Harvard University, and Florida State University. Previously he was a postdoctoral associate with the MIT Center for Technology Policy and Industrial Development (CTPID) and a research assistant with the MIT Lean Aerospace Initiative. He received his Ph.D. in Engineering Systems from MIT, two masters in Aeronautics & Astronautics and Technology & Policy from MIT, and dual BS from Harvard University.