# Triply Articulated Modelling in Complex Systems

**Professor Bernard Cohen**
**School of Informatics, City University**
**London, EC1V 0HB, UK**

**and**

**Philip Boxer**
**Boxer Research Ltd**
**London, UK**

## ABSTRACT

We argue that traditional systems engineering (TSE) is inadequate for engineering complex systems. This inadequacy becomes particularly problematic when the system in question is a System of Systems (SoS) whose behaviour depends on how human users of the component systems, anticipating the consequences of their own behaviour, interact with those systems and with each other. Considering its users to be part of the SoS renders such a system anticipative, and therefore necessarily complex.

To be effective, the designer of an SoS must incorporate an understanding of its users' models of their contexts of use. However, the process of design then becomes reflexive, since it must include within itself models of its users' anticipations of its relation to the SoS's environment, including those of its designers.

As a result, complex systems engineering (CSE) has to become a distinct discipline, involving explicit processes that occur at the time of use of an SoS, not only for negotiating shared meaning between component systems within peer-to-peer relationships, but also for considering the dynamic effects of differing uses of component systems on the behaviour of the SoS as a whole.

By modeling users' relations to demand as well as to the behaviour and organisation of component systems, triply articulated modeling supports the negotiation of shared meaning and accommodates the reflexivity required of a CSE regimen.

**Keywords**: Systems-of-systems, asymmetric demand, enterprise modeling, triple articulation, power-to-the-edge, anticipative systems, third-order systems, complex systems engineering, granularity, stratification, simplicial complex, value ladders.

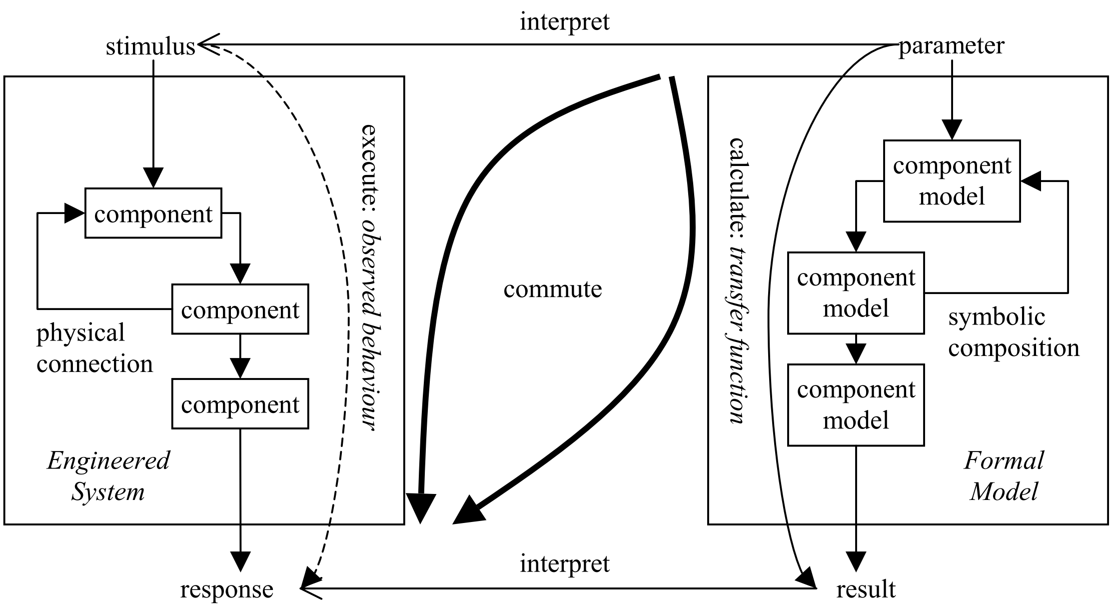


**Figure 1: The Modeling Relation: Engineered Systems**

## 1. MODELING SYSTEMS

Systems engineering involves the construction and manipulation of analytical models of systems. Such a model is a symbolic expression, each 'state' of the system being represented by a set of values attributed to the model's variables. By applying a suitable calculus to the model, one may infer what state transitions the system may undergo. A model is adequate for the purposes of a systems engineer if it explicates and predicts the system's behaviour with sufficient accuracy. For this reason, systems engineers usually choose to design their systems using components whose models provide computationally effective composition calculi (see fig. 1). A model is said to be 'locally stable' if transitions from any two states whose values differ by a small amount lead to states that are similarly close together. Thus a system is said to be 'chaotic' if its model, although adequate, is locally unstable with respect to measurable differences in value of its state variables. The model of the terrestrial atmosphere used in weather forecasting is of this type. In contrast, a system is said to be 'complex' if, even though its observed behaviour seems to exhibit enough local stability to make prediction possible, no adequate model of it could be formulated in advance of such observations. This is typically the case in a 'system of systems' (SoS), a class that hes been defined as follows [5]:

**Operational Independence of the Elements**: If the system-of-systems is disassembled into its component systems the component systems must be able to usefully operate independently. The system-of-systems is composed of systems which are independent and useful in their own right.

**Managerial Independence of the Elements**: The component systems not only can operate independently, they do operate independently. The component systems are separately acquired and integrated but maintain a continuing operational existence independent of the system-of- systems.

**Evolutionary Development**: The system-of-systems does not appear fully formed. Its development and existence is evolutionary with functions and purposes added, removed, and modified with experience.

**Emergent Behaviour**: The system performs functions and carries out purposes that do not reside in any component system. These behaviours are emergent properties of the entire system-of-systems and cannot be localized to any component system. The principal purposes of the systems-of-systems are fulfilled by these behaviours.

**Geographic Distribution**: The geographic extent of the component systems is large. Large is a nebulous and relative concept as communication capabilities increase, but at a minimum it means that the components can readily exchange only information and not substantial quantities of mass or energy.

Although adequate models may exist for all of the component systems of an SoS, not all of the environmental variables that affect its composite behaviour can be known in advance.

Traditional Systems Engineering (TSE) does not fare well in these circumstances [1]. Its inadequacy becomes particularly problematic when the behaviour of the SoS depends on how human users of the component systems, anticipating the consequences of their behaviour, interact with those systems and with each other. Considering its users to be part of the SoS renders such systems anticipative, and therefore necessarily complex.

## 2. VALUING SERVICES

The clients for the services of systems are *actors*: anticipatory systems [2] who derive their demands for services from their formulation of themselves as context to their use of those services. If the service delivered to a client does not satisfy the client's demand when deployed in its context-of-use, then the client will experience a *value deficit* (see fig.2).

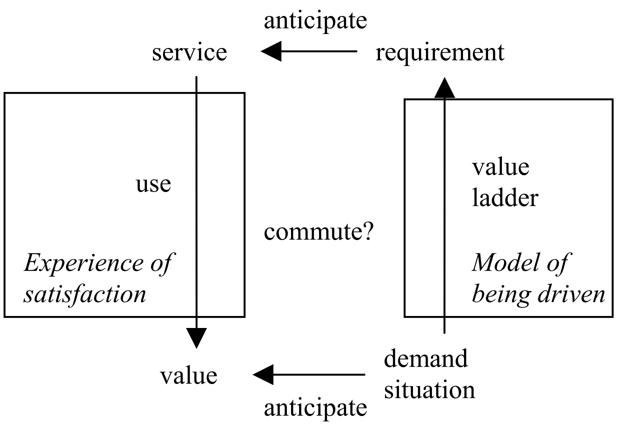


**Figure 2: The Modelling Relation: Anticipatory (Embodied) Systems**

The more control that can be exerted over the behaviour of users, the more tractable, intellectually and computationally, becomes the design of an SoS because the requisite predictability can be imposed. Under these circumstances, some tools can model the expected behaviour. But whatever the level of predictability that can be imposed on the component systems of an SoS, engineering their composition still remains highly problematic [3]. For example, in military uses of SoS, attempts to derive shared situational awareness are inherently problematic because they do not take into the account the effects of complexity on how a common operational picture may be built [4]. Inconsistencies arise not only at the level of data fusion, leading to unintended consequences, but also at the level of synchronisation between its users, where differing situated interpretations of command intent cannot be assumed to be mutually consistent.

To be effective, the designer of an SoS must therefore incorporate an understanding of its users' models of their contexts of use. However, the process of designing such a system then becomes reflexive, since it must include within itself models of its users' anticipations of its relation to the SoS's environment, including those of its designers.

## 3. COLLABORATIVE COMPOSITION

As a result, CSE has to become a distinct discipline, or regimen [1], involving explicit processes that occur at the time of use of an SoS, not only for negotiating shared meaning between component systems within peer-to-peer relationships, but also for considering the dynamic effects of differing uses of component systems on the behaviour of the SoS as a whole. In practice, this means that CSE becomes like a procurement process, but one that goes beyond 'smart acquisition' through the continuously collaborative nature of its relationships with both its suppliers and its operational users.

In TSE, the composition of components is directed by the system designer, who assumes that the system requirement has been fully specified and adequately covers the contexts in wich the system's services will, eventually, be used. In CSE, this assumption is no longer valid. The component systems in a SoS must be composable, in a timely manner, by (or, at least, in close collaboration with) the actors who will use, and value, the services of the composite system in a wide variety of contexts-of-use. In order to achieve this effect, it is necessary to distinguish (see fig. 3) the models of:

**how-it-works**: the nature of the services themselves, more or less over-determined by the causal nature of the processes from which they are constructed; from

**use-in-context**: the nature of the demand itself, which introduces constraints relating to the client's particular context-of-use.

In the left column, there is a single model of how the service works. This model may be built for a particular use-in-context (bottom-left), or it may be parameterised so that it can be adapted to multiple forms of use (top-left). Either way, what the service actually does remains unchanged. On the right, however, there is no single model, but instead multiple components, each with its own model, and each one able to be composed with other models. Composition defines a SoS. In the single use-in-context, a designer can impose a *directed* integration on these components (bottom-right), in which the component systems maintain an ability to operate independently, but their normal operational mode is subordinated to the requirements of the single use-in-context. This directed approach is the one normally used for making large complex SoS projects work, and works through imposing composition externally.

In all three instances so far, however, there is a presumption of a *symmetrical* relation to demand arising from the *a priori* construction of the service. This

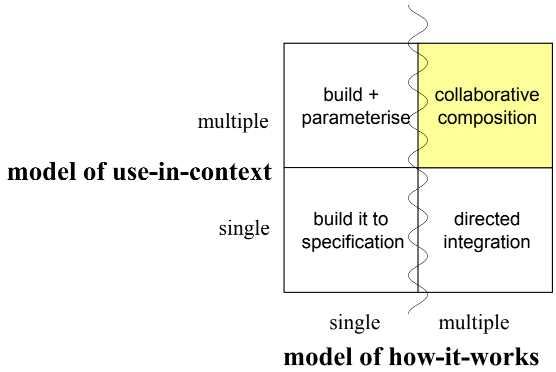


**Figure 3: Collaborative Composition**

approach cannot be used top-right, whether because the composition needed to satisfy a particular use-in-context cannot be predicted by the designer, or because there is too great a variety of such uses to be able to predict which one a given user will want until the time of use. Either way, composition has to be a dynamic response to the user's demand.

Under these circumstances, it is necessary to define a *granularity* and *stratification* of components that can support the variety of uses-in-context.

**Granularity** refers to the level of detail (or abstraction) at which component services are defined.

**Stratification** refers to the way that different layers of composition of component services ultimately relate to the end use.

The approach to composition must be *collaborative* since component systems must be composed at the time of use in collaboration with the user's demand. The more asymmetric the demand, the more granularity and stratification have to be elicited from the nature of the context-of-use in which the demand arises, and the more composition has to be collaborative. The act of determining which components get to participate in the collaborative process of composition is called *Orchestration* (see fig. 4).

The user here is an embodied actor, possessing a model of her own organisation of demand in relation to which her own and others' behaviour is oriented, through specifying services that might satisfy her demands and anticipating, possibly erroneously, their satisfaction by those services. Thus the user's model of demand has itself to be included in the process of service composition. This anticipatory property is characteristic of *third-order systems*, which exhibit what Rosen [1] referred to as 'closed loops of entailment' and for which there exist no computationally effective analytical procedures.

By modeling users' relations to demand as well as to the behaviour and organisation of component systems, **triply articulated modeling** supports the negotiation of shared meaning and enables the identification of appropriate granularity and stratification, thereby supporting the orchestration of SoS and and accommodating the reflexivity required of a CSE regimen.

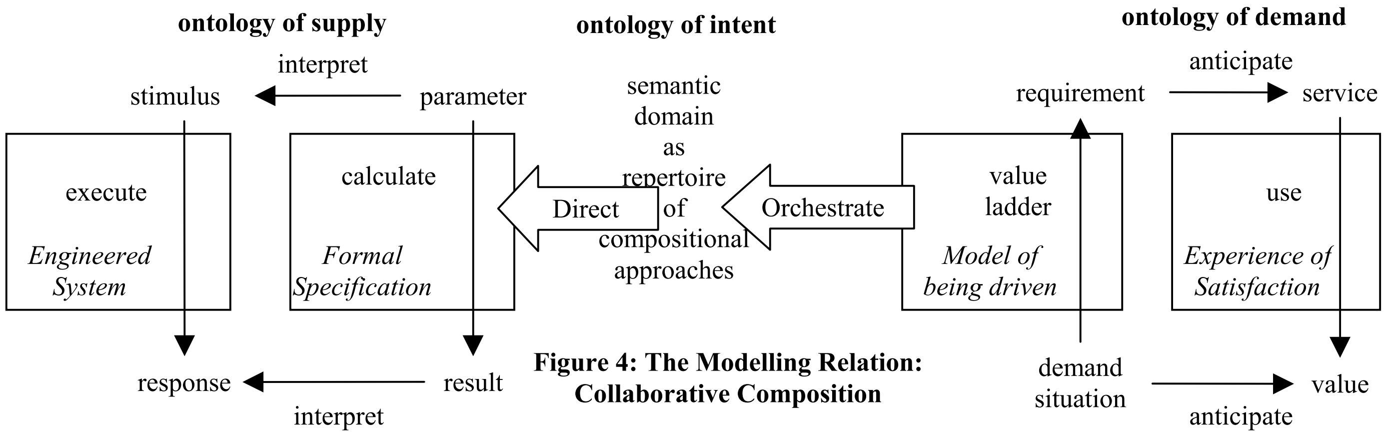


**Figure 4: The Modelling Relation: Collaborative Composition**

### 4. THE THREE ARTICULATIONS

Triply articulated modelling furnishes three different types of model that are inherent to the way the actor-observer constructs her world. These models are elicited indirectly from the actor-observer, using graphical media to facilitate the representation of value chains and of organisational structure/function, synchronisation and hierarchy, which are transcribed into a user-specific 'knowledge base' using the meta-language of PAN, a computational tool implemented in Prolog.
Each articulation in the PAN representation is a relational structure of linked directed graphs (see fig. 5).

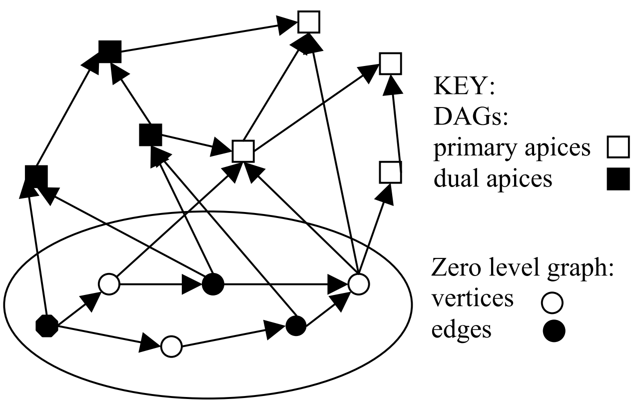


**Figure 5: The Structure of Articulations**

The **existential** articulation models the actor's knowledge of how her world behaves in terms of:
○ *processes*, which are (objectively) observable by the changes that they induce in the (public) state-of-affairs;
● *events*, which are states-of-affairs that are observed to pertain after the occurrence of a process;
□ *coordinations*, which are collections of processes (and/or coordinations, recursively) that are observed to occur together in some purposeful way; and
■ *views*, that record the occurrence of past events.
This model gives a formal account of the world observed by the actor, explicating its material behaviour in terms of processes that act in it. In this kind of model, which when elaborated becomes a scientific theory, one event logically entails another if there is a process that is enabled by the first and terminates in the second. This formal model is related to the material world by an interpretation that takes each event to a state-of-affairs and each process to a 'material cause', in such a way that every chain of logical entailments commutes with an observably causal effect. The 'lower' one goes down the recursive existential articulation, the 'deeper' become the 'formal causes' in terms of which the material behaviour of the world is explicated. Of course, like all formal theories, the existential articulation may be inconsistent or erroneous, subject to alteration as the actor learns more about her world. To be complete, it would have to include a 'theory of everything', which is a paradoxical concept.
In the existential articulation, the actor models her world from an *exo* [6] perspective. Standing 'outside' of it, she can envisage its behaviour with its causality reversed, tracing effects back to their (formal) causes. But she can account for the effects on herself of processes in her world only by constructing a model of her world from the *endo* perspective (with herself in it) where, although she experiences as irreversible the occurrence of states-of-affairs, she may nevertheless know that the concurrency of certain of these occurrences may be reliably controlled.
The **deontic** articulation models the actor's knowledge of this controllability in terms of:
○ *outcomes*, which are observable states-of-affairs;
● *transformations*, which denote, but do not necessarily give an account of, mechanisms that alter outcomes;
□ *synchronisations* of outcomes that may be made to occur together; and
■ *compositions* of transformations that may be considered as atomic.
The existential and deontic articulations may be composed by asserting mappings among their respective components that induce the implication of certain existential events in certain deontic outcomes. As with the existential, this **composed existential-deontic** articulation may be inconsistent or erroneous. In addition, although the existential models each process as a 'closed system', that interacts only through the events to which it

is directly related, not all of these interactions may be mapped into the deontic. Indeed, the existential's incompleteness leaves open the possibility of behavioural interactions that are not represented at all. These 'open system' phenomena, such as 'feature interaction' and 'emergent behaviour', are therefore necessarily lacking from the composed articulation.

This model defines the repertoire of all behaviour paths known to the actor. It is the space in which the actor, as 'efficient cause', can act by constructing and executing plans that make certain behaviour paths, and hence certain states-of-affairs, come to pass. Which plans the actor chooses will depend on how she values their implicated behaviour paths and outcomes. To give an account of her choosing, we must construct a model, not of the material world, but of the embodied individual as 'final cause'. This (*exo*) account is of the actor's anticipation of the effects in herself, as a particular 'other', of experiencing certain behaviour paths, and of how her choice among these behaviour paths is entailed by that anticipation. This reversal of the direction of entailment (from 'efficient' to 'final' cause) characterises the 'anticipatory system' [2].

The **referential** articulation models the actor's knowledge of herself as an anticipatory system in terms of:

○ *demand situations*, which are states-of-affairs whose coming to pass is anticipated to be satisfied with respect to certain drivers;

● *drivers*, which attribute value to the actor's experience (by being, more or less, 'satisfied' by paths-of-behaviour);

□ *customer situations*, in which the experience of situations (recursively) common to certain demand situations is anticipated to be satisfied with respect to certain drivers common to those demand situations; and

■ sets of *drivers* that, collectively, drive demand situations.

Direction is reversed in the referential articulation, so that the 'higher' one goes, the more of the actor's being is implicated in the demand situations, and the more complex becomes the actor's valuation of her experience of them in terms of drivers. Equally, the 'lower' one goes (away from the demand situations), the 'simpler' become the patterns of behaviour entailed by the actor's anticipation of their experience, until the actor becomes indifferent to the way the situation is experienced *per se* (referred to as *requirements*, being context-independent customer situations).

As in the other two articulations, the referential may be erroneous or inconsistent and, since the actor's anticipation of her world always 'leaves something to be desired', is necessarily incomplete. In particular, since this fundamental human limitation restricts the actor's ability to express her experience of being driven and the extent of contexts in which she anticipates satisfaction, she cannot directly associate drivers with demand situations. The zero-level graph of the **elicited referential articulation** is therefore degenerate in that its source and target functions are empty (that is, its vertices and edges are completely disconnected).

This form of the referential articulation, as elicited from the actor, does not provide the structure or content required for composition with the other two articulations. It is possible, however, to ***compute*** from the elicited data a ***hypothetical*** **constructed referential articulation** which provides the minimal required structure and content and is consistent with the elicited data. Its components are:

○ requirements;

● drivers;

□ customer situations, each being a context of use in which the subtended requirements arise; and

■ value profiles, each being an anticipation of satisfaction identified with the subtended set of drivers.

The actor may now assert mappings among the objects of the three articulations, which may induce inconsistencies among them. The PAN toolset exposes these symptoms of invalidity to the actor, and her consequent repair of them, by appropriately updating her models, accelerates her learning and facilitate her strategic development.

## 5. THE TRIPLE ARTICULATION

The resulting composite structure represents an actor's model of the enterprise in which she is engaged, rather that being a model of the enterprise itself. PAN can transform this structure into any of six different **composite triple articulations**, depending on which articulation is to be *privileged*, and in which *sequence* the articulations are selected (see fig. 6).

Key:: **E**xistential; **D**eontic; **R**eferential

**D** × **E** → role hierarchy
  × **R** → what don't we know how to satisfy?
**D** × **R** → supply-side goal stratification
  × **E** → what can't we do from the supply-side?
**R** × **D** → demand-side goal stratification
  × **E** → what can't we do from the demand-side?
**E** × **D** → capability hierarchy
  × **R** → what are we unable to satisfy?
**E** × **R** → supply-side value stratification
  × **R** → what are we unable to orchestrate?
**R** × **E** → demand-side value stratification
  × **D** → what don't we know how to orchestrate?

**Figure 6: The six pruning sequences**

The privileged articulation defines the domain in relation to which the composite articulation is formed, with each of the objects that is not mapped to some object in this domain being deemed to be absent from the world denoted by the composite and 'pruned' away. The resultant ***composite knowledge base*** (CKB) is internally consistent, defined in terms of objects that together comprise an ontology embedded in the semantic formation of a particular actor-observer.

*Note that we differ here from those knowledge representation approaches that adopt the 'closed world assumption' of a Universal Ontology.*

PAN can project this CKB into a number of matrices that are *stratified* in relation to each other. The simplest stratification (see fig. 7) arises because of the way the articulations are organised with respect to the referential, showing the particular relation of the enterprise to its *pragmatic* context. Thus the lower strata represent products and services that are incorporated into the products and services of the higher-level strata, that finally arrive at their use within contexts-of-use. Crucial in this analysis is the way in which the ontology of the lower level strata is aligned to the upper level strata by the particular form of their relation to the context-of-use.

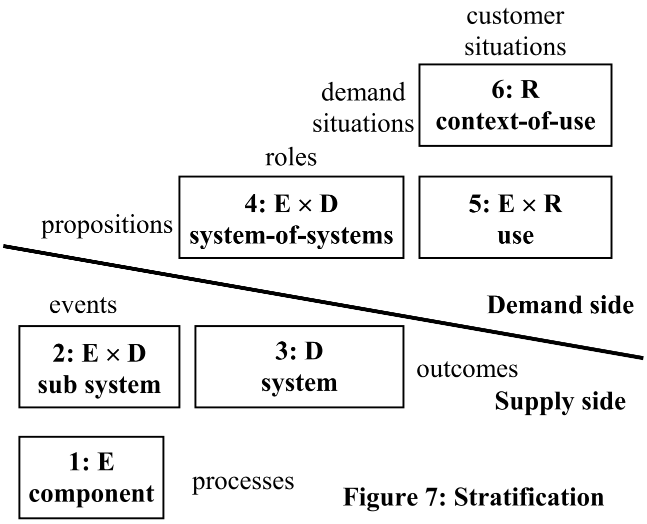


**Figure 7: Stratification**

There is of course no limit to how many tiers can be separated out but they can always be projected into six distinct strata. For any projection, each stratum can be construed as a binary relation, or *simplicial complex*, and subjected to *landscape analysis*, an extension of Q-analysis [7] which produces 3D histograms in which structural gaps are made graphically visible. These gaps signify the absence of direct relationships between objects within each stratum, reflecting the particular ways in which they are *unable* to interoperate.

## 6. THE THREE RISKS

In considering the implementation of any complex system, we can distinguish three kinds of risk:

*performance risk*: the possibility that a component product or service might not work as specified by the supplier;

*composition risk*: the possibility that component, and hitherto independent, products or services might be not be able to interoperate in such a way as to provide an intended product or service as a whole; and

*implementation risk*: the possibility that the intended product or service might not satisfy the client's demand as expected when used in its actual context-of-use, even though it meets the product or service specification agreed with the supplier.

Traditional analysis, simulation and testing regimes deal with performance risk and, to some extent, composition risk. But implementation risk is peculiar to the relational response to asymmetric demand (the positional response being to ignore the value deficit associated with it) and is beyond the purview of Traditional Systems Engineering.

Triply Articulated Modeling has been designed to operate in this arena. Implementation risks are readily identified as 'holes' in the landscapes of strata 5 and 6 and as measurable structural differences between them, while composition risks are similarly evident at strata 3 and 4. And although the semantics of the models at strata 1, 2 and 3 do not directly admit behavioural analysis or simulation, their structural holes and differences do suggest where further traditional analyses should be concentrated.

## 7. CONCLUSIONS

Triply Articulated Modeling and its computational tools allow users to define what they need of a complex system in terms of what they are doing as users, by referrring to the geometry and agility of the system as well as to its functionality. This enables greater transparency in the management of risk at multiple strata by supporting reasoned argument about the 'trade spaces' where decisions must be made. It also supports the procurement of systems-of-systems that are to be deployed in social contexts, such as defence and healthcare, where 'power' over how systems are to be composed must be moved 'to the edge' where it can produce its effects collaboratively.

We are currently investigating the algebraic structure of articulations so as to refine the analytical treatment of composition and projection and to increase the analytical power of the toolset.

## 8. REFERENCES

[1] D. Norman, **Engineering a Complex System - a study of Air & Space Operations Centre** ICCS 2004.

[2] R. Rosen, **Life Itself**, Columbia University Press, 1991.

[3] See the five 'challenges identified by IBM Rational **Software Enterprise Transformation** at www-306.ibm.com/software/rational/solutions/etrans/web services.html

[4] Y. Bar-Yam, **Report to Chief of Naval Operations Strategic Studies Group,** 2001

[5] M. W. Maier**, Architecting Principles for Systems of Systems**, www.infoed.com/Open/PAPERS/systems.htm

[6] H. Atmanspacher and E. Ruhnau (eds), **Time, Temporality, Now**, Springer Verlag, Berlin, 1997.

[7] R.H. Atkin**, Mathematical Structure in Human Affairs**, Heinemann 1974.