Complex Adaptive Systems, Volume 1
Cihan H. Dagli, Editor in Chief
Conference Organized by Missouri University of Science and Technology
2011- Chicago, IL

Innovation as emergence:
Hybrid agent enablers for evolutionary competence

William D. Schindel *

ICTT System Sciences, 378 South Airport Street, Terre Haute, IN 47803 USA

Abstract

Human-performed innovation processes can be viewed as complex adaptive systems in their own right. The competitive effectiveness of such innovation processes is of not only academic interest, but also in commercial and military domains, in which levels of competency in innovation spell the difference between success and failure for the competitors.

This paper describes an epigenetic view of innovation as an interaction of innovation agents, domain systems, and models of those systems. As contrasted from a genetic algorithms framework, we instead examine the case in which multiple localized innovation agents that include humans or hybrid human-machine agents are in interaction with evolving models of target systems.

Key enablers of this framework include (1) system patterns (emergent, configurable, evolvable models) and (2) innovation agent views. The resulting framework can improve understanding of human-performed innovation processes, as well as point out enablers of increased competency for the overall system of innovation.

Keywords: Innovation; Emergence; Evolution; Competence

1. Introduction: Human-performed innovation

There is an extensive literature describing system development processes—a subset of all human-performed innovation processes, illustrated in Schindel et al [1]. These descriptions of development processes frequently describe functions performed across departments of an enterprise, as summarized by ISO/IEC [2] and further detailed by INCOSE [3] and NASA [4]. It less clear that these descriptions address what actually occurs in the cognitive processes of skilled individual engineers performing innovation processes, per Ring [5]. This paper summarizes a framework describing human-performed innovation processes in terms of smaller units of work, performed by interacting agents, from which emerges a model of an innovated system of interest.

_____
* Tel.: 812-232-2062; Fax: 812-232-2998
E-mail address: **schindel@ictt.com**

Human-performed innovation is a special case of innovation processes in general. This paper is limited to consideration of that special case—different models apply in other types of innovation. This paper also connects to traditional engineering views of these human-performed innovation processes, while offering some different ways to understand and even perform them. Our starting point is the view that "All Innovation Is Innovation of Systems" as described in Schindel [1]. By this we mean that, whether the innovation itself is systemic or of a more "point" nature, there is always a way to view the innovated thing in a systems context. In fact, a starting point for this perspective is the traditional "Vee" diagram of systems engineering, depicted in Figure 1, as in INCOSE [3].

We are particularly interested here in the <u>iteration</u> process that traditionally is intuitively pictured as traveling up and down the Vee, to eventually yield a newly innovated system. We begin by constructing a more formal view of the nature of this process, when performed in a certain way— by a particular Model-Based Systems Engineering (MBSE) method.
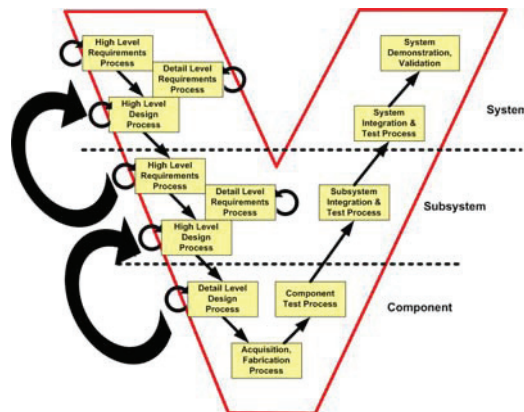

Figure 1: The Traditional "Vee Diagram" of Systems Engineering

2. Models; Model-Based Systems Engineering (MBSE)

This paper is based on the use of Model-Based Systems Engineering (MBSE), and in particular on model information satisfying the S* Metamodel, summarized in Figure 2. Both MBSE and the S* Metamodel are discussed further in Schindel [6] and [7]. There is no assumption of a particular modeling language, and most contemporary system modeling languages can express information compliant with the S* Metamodel. <u>Models</u> here are assumed to be data structures that described systems engineering information including stakeholder values, system requirements, high level design information, failure modes and effects, and other systems engineering information. S* Models conform to the S* Metamodel of Figure 2.
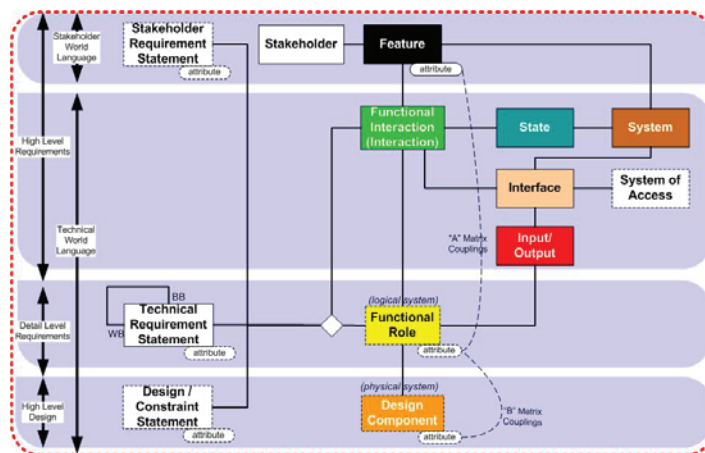

Figure 2: The S* Metamodel

3. Emergence: Local model gaps, views, and agents

The traditional Vee Diagram of Figure 1 provides an intuitive "big picture" of some hierarchical aspects of the nature of the systems development process. For purposes of this paper, we will limit our view to the left side of the Vee diagram. (That is, we are not including end system integration, verification, validation, customer selection, and in-service life cycles--all of which play additional important roles in the innovation process, beyond the scope of this paper.)   Figure 3 provides a more detailed view of a set of logical information processing processes and model information that could implement the left side of the Figure 1.
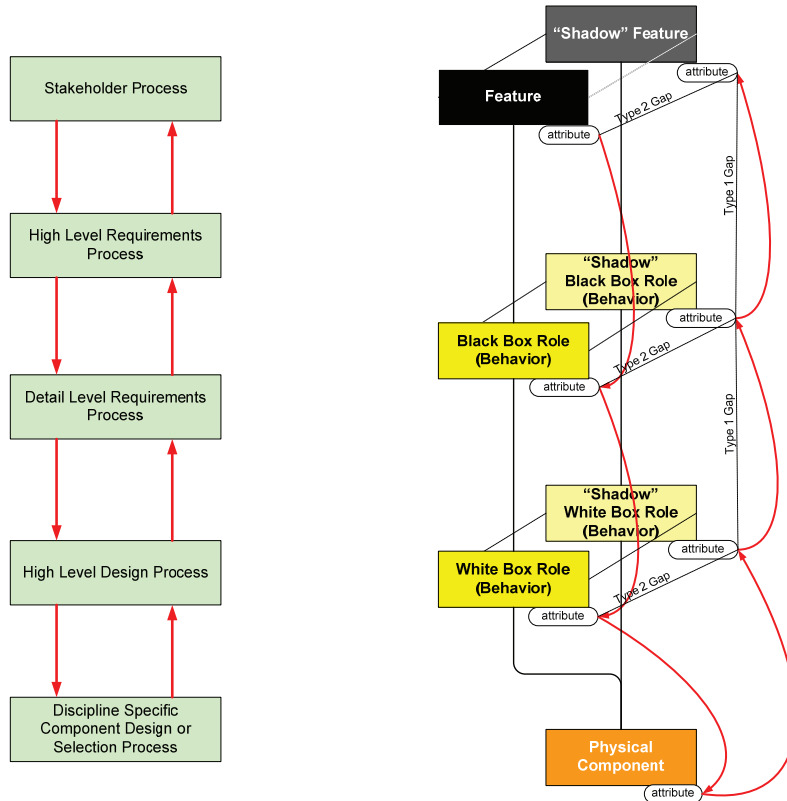

Figure 3: A Hierarchical Innovation Process Model

Figure 3 illustrates the idea that, for a given physical design, we are interested in two sets of information: a set of requirements or intentions, and a matching "shadow" set of capabilities. The former represent the requirements implied by stakeholder needs. The latter represent the capabilities of the real selected design components. As shown in Figure 3, the capability values propagate upward from a selected design, whereas the required values propagate downward from the stakeholder needs. The difference between these value pairs is shown in Figure 3 as "Type 2 Gaps". Type 2 Gaps reflect the difference between what is wanted versus real capability. At the Feature level, they reflect that difference in stakeholder terms. These attributes (and gaps) typically include expressions of uncertainty, and at the Feature level expressions of risk (weighted impact, usually in terms of risk to performance, cost, and schedule). By contrast, Type 1 Gaps (in Figure 3) represent inconsistencies of descriptions in the different levels of the system hierarchy. Type 1 Gap checking is closer to the "tracing" that occurs in traditional systems engineering.

It is important to understand in Figure 3 that all the capabilities of the candidate physical component are represented by the attributes of its allocated White Box Role, which represents all behavior. The attributes of the Physical Component are limited to simple non-behavioral identity attributes, such as manufacturer, part number, material, or dimension. Figure 3 shows only a single physical component class with no shadow, because it is about actual candidate components, not desired ideal behavior.

By further decomposing the processes of Figure 3 into those shown in Figure 4, we introduce the idea of Local Innovation Agents, which for purposes of this paper may be considered to be humans. As shown in Figure 4, they interact with other Information System Agents through Local Innovation Views.

Figure 4 represents the idea of agents interacting with information as external models (information artifacts as objects outside the mind of the human agents), as discussed further in van Fraassen [8].
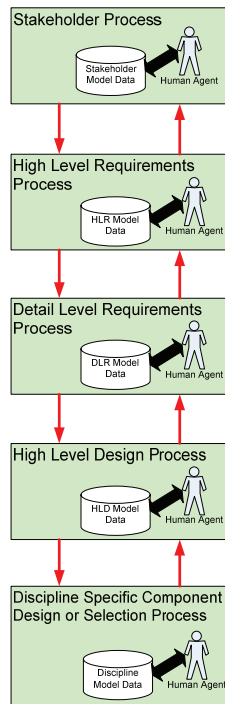
Figure 4: An Interacting Agents View of Innovation

Agents shown in Figure 4 are charged with reducing (or eliminating) the Type 1 and Type 2 gaps of Figure 3. Note that these gaps are "local" within the model. An important point here is the decomposition of the complex innovation process to a collection of (relatively) simpler interactions of a more localized nature. By "localized" we don't just mean performed by a local agent: We also mean that only a limited part of the information model of Figure 2 is considered by each agent. Even though none of these agents considers "the whole model", a whole model of the innovated system emerges over a series of interactions by these agents with their local parts of the model. Emergence is further discussed in Holland [9].

The tasks performed by these agents might be considered "local", but they are not trivial, and may involve considerable expertise. The Innovation Competencies involved are discussed further in Schindel et al [1].

4. Pattern-Based Systems Engineering (PBSE)

One way to boost the performance of the emergence process, or to reduce the level of expertise required of the agents, is the use of Pattern-Based Systems Engineering (PBSE) in this process, as in Schindel [10], Bradley, et al [11] and Schindel and Smith [12].   Patterns are Models (in the sense of Section 2) that are configurable and re-usable. The Patterns we are discussing in this paper are S* Patterns, which are configurable, re-usable S* Models. Such Patterns describe configurable families of systems, and may represent product lines, market segment applications, or other system family ideas.

The introduction of Patterns changes Figure 4 into Figure 5, in which the Innovation Agents have the benefit of known Pattern data to enhance their configuration of the Models, and in which additional Pattern Management Agents look after a parallel process of accumulating learning in the form of new or improved Patterns. This creates an additional emergence process over a longer time frame, in which System Patterns emerge that incorporate what has been learned. The feedback process that assesses pattern gaps and closes them with improved patterns is shown in Figure 6. This loop-closing learning process in some cases is used to capture the patterns of natural phenomena, and is related to the scientific method, as summarized by Williams [13].
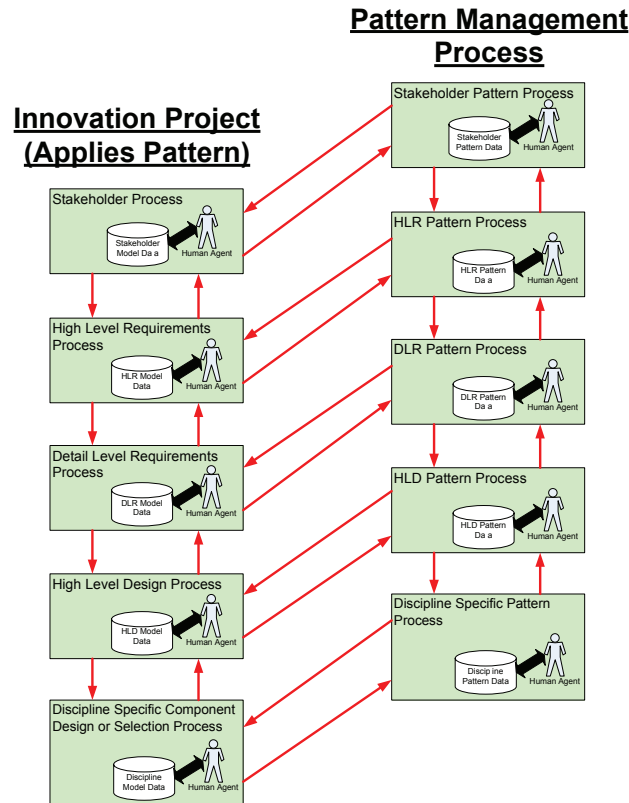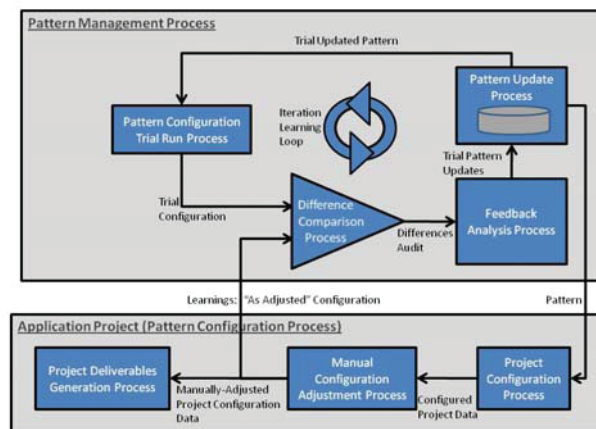
Figure 5: Introduction of Patterns

Figure 6: Pattern Management Feedback Loop

5. Conclusions and future questions

Traditional descriptions of human-performed innovation processes emphasize various information deliverables as primary, and the relationships between them (as in tracing requirements to design) as secondary. This paper describes a different perspective, in which the information relationships are primary, and the efforts by specialized agents to close consistency gaps in those relationships leads to emergence of the information deliverables. A global fabric of information emerges from these local efforts. This might be viewed, for example, as using Design Review to <u>create</u> Design and Requirements, instead of only to <u>assess</u> them.

The Innovation Process has been modeled in more detail than the summaries of Figures 3-6. Modeled systems can have performance models. This suggests the following opportunities for further investigation, involving Risk, Cost, and Time:

1. Simulation of performance of the modeled systems of innovation.
2. Calibration of the simulations with real experiences.
3. Integration of this hybrid model with more automated models, such as genetic algorithm approaches.
4. Addition of the right side of the Vee model.
5. Addition of the in-service part of the system life cycle model.

6. References

1. W. Schindel, S. Peffers, J. Hanson, J. Ahmed, W. Kline, "All Innovation is Innovation of Systems : An Integrated 3-D Model of Innovation Competencies ", Proc. of ASEE 2011 Conference (2011).
2. ISO/IEC 15288: Systems Engineering—System Life Cycle Processes. International Standards Organization (2008).
3. *INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*, Version 3.2, International Council on Systems Engineering (2010).
4. *NASA Systems Engineering Handbook,* NASA/SP-2007-6105, Rev 1, U.S. National Aeronautics and Space Administration (2007).
5. J. Ring, "Capability Maturity Models vs. Process Standards vs. The Real World", *INCOSE Insight*, July, (2001) 23.
6. W. Schindel, "Requirements statements are transfer functions: An insight from model-based systems engineering", Proceedings of INCOSE 2005 International Symposium, (2005).
7. W. Schindel, "What Is the Smallest Model of a System?", Proc. of the INCOSE 2011 International Symposium, International Council on Systems Engineering (2011).
8. B. Van Fraassen, *Scientific Representation: Paradoxes of Perspective*, Oxford U Press, 2008.
9. J. Holland, *Emergence: From Chaos to Order*, Perseus Books (1998).
10. W. Schindel, "Pattern-Based Systems Engineering: An Extension of Model-Based SE", INCOSE IS2005 Tutorial TIES 4, (2005).
11. J. Bradley, M. Hughes, and W. Schindel, "Optimizing Delivery of Global Pharmaceutical Packaging Solutions, Using Systems Engineering Patterns" Proceedings of the INCOSE 2010 International Symposium (2010).
12. W. Schindel, and V. Smith, "Results of applying a families-of-systems approach to systems engineering of product line families", SAE International, Technical Report 2002-01-3086 (2002).
13. D. Williams, "How Concepts of Self-Regulation Explain Human Knowledge", *The Bent of Tau Beta Pi*, Winter (2011) 16-21.