Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 28 (2014) 848 – 856
Procedia Computer Science

Conference on Systems Engineering Research (CSER 2014)

Eds.: Azad M. Madni, University of Southern California; Barry Boehm, University of Southern California;
Michael Sievers, Jet Propulsion Laboratory, Marilee Wheaton, The Aerospace Corporation
Redondo Beach, CA, March 21-22, 2014

# Toward an Experiential Design Language: Augmenting Model-Based Systems Engineering with Technical Storytelling in Virtual Worlds

Azad M. Madni[a]*, Marcus Nance[b], Michael Richey[b],
William Hubbard[b], Leroy Hanneman[b]

[a]*University of Southern California, 854 Downey Way., Los Angeles, CA 90089, USA*
[b]*The Boeing Company, USA*

**Abstract**

Systems engineering has made significant strides over the last decade with the advent of Model Based Systems Engineering (MBSE) and the Systems Engineering Markup Language (SysML). These advances have made it possible for collaborative engineering teams to communicate using a common language and share information in the form of digital models rather than hard-to-maintain paper documents. One major advance that needs to occur next is enhancing systems engineering languages to address the needs of all stakeholders especially non-engineers. In this paper, we advance the proposition that recent advances in systems modeling, virtual world building, and technical storytelling are the key enablers of next generation systems engineering and systems engineering languages. To this end, this paper presents an innovative approach for developing an experiential design language that augments existing systems engineering language (i.e., SysML) with new perspectives informed by exploration and storytelling in virtual worlds. This language is intended to allow all stakeholders to understand and collaborate on system designs without having to learn engineering notation, and bring to bear their unique perspectives during collaborative system design. Furthermore, immersive experiences made possible by storytelling in the virtual world can potentially illuminate key system

* Corresponding author. Tel.: 213-740-3442; fax: 213-740-1120.
*E-mail address:* azad.madni@usc.edu


Selection and peer-review under responsibility of the University of Southern California.
doi:10.1016/j.procs.2014.03.101

interactions and behaviors, allowing all stakeholders to make meaningful contributions in upfront systems engineering. The benefits and payoffs of the experiential design language are: increased participation of all stakeholders in upfront systems engineering; superior collaboration among stakeholders by allowing them to focus on different system levels and perspectives; and elegant system design by getting the architecture right the first time with minimum structural complexity.


Selection and peer-review under responsibility of the University of Southern California.

*Keywords*: experiential design, systems modeling, virtual world, model based systems engineering, technical storytelling, model based gaming, stakeholders, business metrics

## 1. Introduction

Systems engineering today is being fueled by advances in Model Based Systems Engineering (MBSE), with SysML[1] being one of the primary language. OPM[2] is another language that is being used to operationalize MBSE. These advances have made it possible for collaborative engineering teams to communicate using a common language and share information in the form of digital models rather than hard-to-maintain paper documents. However, these system modeling languages employ highly technical notation that prevents stakeholders who are not engineers from contributing to system design and validation in upfront engineering. Consequently, the resultant designs fail to take into account key considerations that are not only important to stakeholders but in fact could potentially be "design drivers." Not surprisingly, there tends to be an increase in Engineering Change Requests (ECRs) and design iterations. These inevitably result in schedule delays and cost over-runs. In light of the foregoing, the challenge today is to extend systems engineering languages in a manner that makes them comprehensible to all stakeholders. Successfully addressing this challenge is key to allowing all stakeholders to provide valuable and timely inputs into upfront systems engineering when change is both possible and affordable.[3]

In this paper, we advance the proposition that recent advances in systems modeling, experiential design, and technical storytelling in virtual worlds are the key enablers of next generation systems engineering and systems engineering languages. To this end, this paper proposes the development of an experiential design language which builds on existing systems engineering language (i.e., SysML) by adding a new experiential perspective within the MBSE rubric. The experiential perspective is rooted in storytelling and "what-if" exploration in virtual worlds.[4] An experiential-based language is intended to allow all stakeholders to understand system designs without having to learn engineering notation, and bring to bear their unique perspectives during collaborative system design. Within this construct, immersive experiences in the virtual world are expected to illuminate system interactions and behaviors, thereby allowing all stakeholders (especially non-engineers) to make meaningful contributions to upfront systems engineering.

This paper is organized as follows. Section 2 presents a methodological framework for incorporating an experiential design perspective into MBSE to produce elegant systems. Section 3 presents the key elements of a closed loop approach to developing an experiential design language. Section 4 summarizes the implications of this advance, and outlook for the future.

## 2. Incorporating Experiential Design into MBSE

Experiential design involves perspectives rooted in intuitively appealing abstractions[5,6] and portrayal of physical reality that non-engineers can understand, relate to, interact with, and contribute to. In particular, an experiential perspective can potentially extend the representation of systems to include perspectives enabled by exploration and storytelling in virtual worlds. An experiential perspective promotes stakeholders' understanding and fosters effective communication between non-technical stakeholders and system engineers[3]. . Experiential perspectives also engage stakeholders in meaningful ways by placing the design within its operational usage context defined by the scenario/story which unfolds in the virtual world. To create experiential perspectives, the role of the system designer/engineer is expanded to include that of a creator of experiences that address stakeholder concerns[3].

Stakeholder experiences can be shaped not only by the multi-sensory environment but also by the use of the right metaphors:[7] structural; orientational; and ontological. The *structural metaphor* comes into play when one concept is structured in terms of another (e.g., argument is structured as war). The *orientational metaphor* pertains to structuring experiences in terms of spatial directions (e.g., down direction is negative, and up direction is positive). The *ontological metaphor* helps structure experiences of abstract phenomena in terms of concrete objects and forces (e.g., risk level in terms of colors; proximity in terms of symbol size; interaction strength in terms of color and link thickness). From a systems modeling and engineering perspective, this means that system designers need to also design conditions that contribute to stakeholders' experiences during system modeling and designing. These insights provide the motivation for developing an Experiential Design Language (EDL).

From a functional perspective, an EDL adds an experiential perspective informed by storytelling in virtual worlds to the traditional systems modeling views associated with MBSE. The technical story comprises the system being developed, actors that interact with the system, events timeline, environmental entities that participate in the story, and the locations and behaviors of all entities. Figure 1 presents a methodological framework that shows the relationships between system models, experiential perspective, and storytelling in virtual worlds. It also shows the mappings/transformations between the system design space and the storytelling space. These include: a) developing system models in SysML from customer-supplied operational scenarios and use cases provided in SysML; b) mapping specific entities and relationships from use cases and activity sequence diagrams (system models) to virtual world entities and relationships needed to support storytelling; c) rendering virtual world entities and their behaviors in the context of stories that unfold in the virtual world; and d) offering multiple role-specific and function-specific lenses for different stakeholders during technical storytelling. The overall approach is presented within the context of a "diverted aircraft scenario," which is used as an illustrative example.[8]

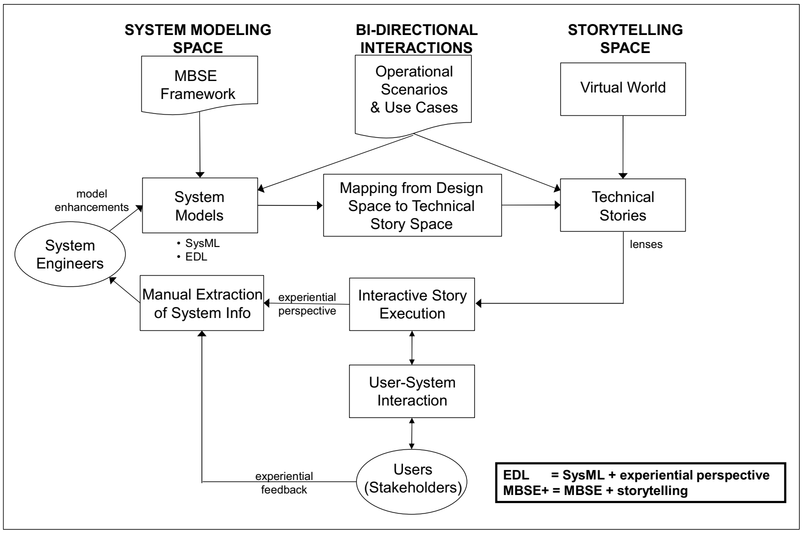


Figure 1: MBSE+: Augmenting MBSE with Storytelling to Create Elegant Design

From a stakeholder perspective, the role of EDL goes beyond communication and comprehension to being a valuable source of experience and insights for stakeholders. For example, the EDL is intended to capture how stakeholders talk about their experiences with the system in the operational environment provided by the virtual world. As shown in the figure, user interactions with the system in the virtual world are also a source of experience. When traditional system views are augmented with experiential perspectives, the resultant system architecture based on requirements and concerns of all stakeholders can be expected to be elegant. In other words, the combination of MBSE perspectives and the experiential perspective assures wider stakeholder participation early and often in the design process. The latter is a key enabler of elegant system design[9,10,11] Madni[10,11] defines an elegant system as "an engaging and affordable system that offers requisite functionality and quality attributes with minimum structural complexity." This perspective provided the motivation for EDL development.

## 3. Key Elements of the Approach

In the following paragraphs, the key elements of the EDL development approach are discussed using the "aircraft divert scenario" as an illustrative example.

### *3.1. Illustrative Scenario for Technical Storytelling*

The illustrative scenario used to demonstrate the development of technical stories is the "aircraft divert scenario." This scenario, which comprises both sunny day and rainy day versions, revolves around an airplane that takes off, and experiences a malfunction, that requires making a decision about whether or not to divert the aircraft. The scenario employs an airplane that in one case is equipped with an Aircraft Health Management (AHM) system, and in the other case is not. With the AHM, the airplane is expected to have fewer diverts because of up-to-the-minute information about aircraft health and availability of potential landing sites. However, today the full value of the AHM is not always realized because many stakeholders are not conversant with AHM usage. A key purpose of the EDL is to enhance the understanding of AHM on the part of all stakeholders by allowing them to "experience" working with the system within scenario-based stories that unfold in virtual worlds. Stakeholders that experience the design in this fashion are also able to collaborate effectively with each other, and provide useful and timely feedback on system design to systems engineers during upfront system engineering.

### *3.2. System modeling*

Operational scenarios/use case diagrams and operational business process/activity diagrams, represented in SysML, are the starting point for developing system models. The mapping of use case scenarios to system models typically exposes gaps in both. These gaps are filled by subject matter experts (SMEs). Thereafter, the system model is checked for completeness, consistency, and traceability to source documents/SMEs. The system model is then used by downstream activities such as storytelling in virtual worlds. For the "aircraft divert scenario" example, the operational scenarios/use cases comprise both rainy day and sunny day conditions.

A prerequisite to systems modeling is defining what constitutes the system and what constitutes the environment. To this end, considerations such as system purpose, dependencies, and interaction density were used to define the system. For the illustrative example, the system is composed of the pilot/flight crew, the AHM system, and the maintenance controller. The rest is the environment which is a source of inputs to the system and where the effects of system actions and other scenario events are felt and manifested. Figures 2 through 4 represent exemplar SysML representations of the AHM system model. Figure 2 shows the system actors, while Figure 3 shows an overview of use cases. Figure 4 is an activity diagram of the "Prepare for System Use" use case. The SysML models contribute to the creation of the virtual world within which stories unfold. To this end, a few key questions need to be answered such as:

1. What are we trying to explore in the virtual world and what questions do we want to answer?
2. Who are the stakeholders and what are their perspectives that need to be represented in the virtual world?
3. Who are the characters in the technical story that interact with the system? What are their personality characteristics (i.e., risk tolerance, skill levels, preferred communication methods)?
4. What story vignettes need to be incorporated in the overall storyline to answer questions of interest to the stakeholders?

For example, in Figure 4, the first step in the activity diagram is to "survey the site." In a top-level behavioral diagram, this activity is usually quite clear. However, when creating a virtual world representation, we need more information such as what are we looking for at the site, what is at that site that is of interest to us, and what obstacles (i.e., things that might get in the way) might be at that site. We also want to know how we plan to survey the site – do we walk around and look at things and make check marks in a book, or are we going to bring up a computerized inventory in which case we might define surprise twists in the technical storyline such as when the inventory doesn't match the physical site. The storyline can also include constraints such as maximum time to complete the survey, injection of human errors to produce storyline twists, and possibly relationships between trusted and untrusted surveyors that can impact survey reliability. With respect to Figure 2, it is important to note that only a subset of characters that participate in the storyline need to be represented. The rest can become part of the external environment.

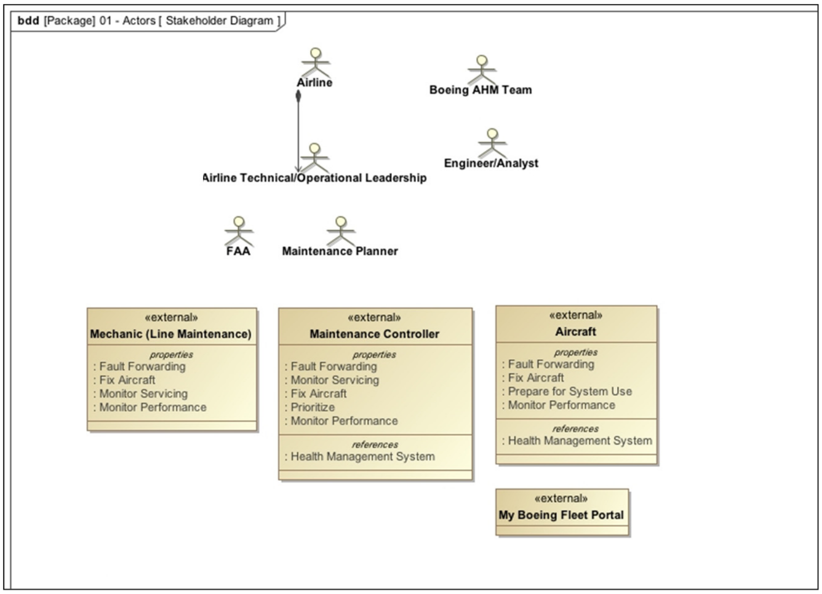


Figure 2. Stakeholder Diagram (Actors)

Figures 3 and 4 are the SysML behavioral representation of the AHM. Figure 3 shows a collection of use cases for the "aircraft divert scenario." For the illustrative example, only a subset of these use cases is needed. These use cases are represented in specific story vignettes. Each vignette has initiating condition, key events, and terminating condition. As noted earlier, the system under consideration comprises the pilot/flight crew, the AHM system and the maintenance controller. The AHM performs several functions including fault forwarding, fixing the aircraft, providing prognostics, monitoring servicing, prioritizing faults, monitoring performance, and prepare for system use. These capabilities of the AHM are depicted in Figure 3. The AHM interacts with the aircraft, the airline technical and operational leadership, and the AHM team.

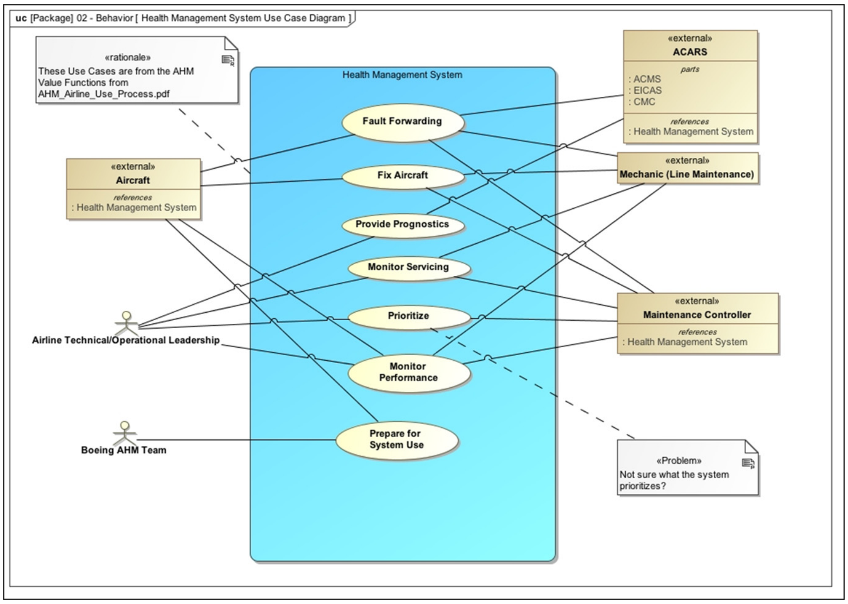


Figure 3. Health Management System Use Case Diagram (Behavior Diagram)

Figure 4 presents "prepare for system use" activity diagrams. This view, in swim lane format, presents both the activity sequence and the interfaces and interactions among the swim lane elements.

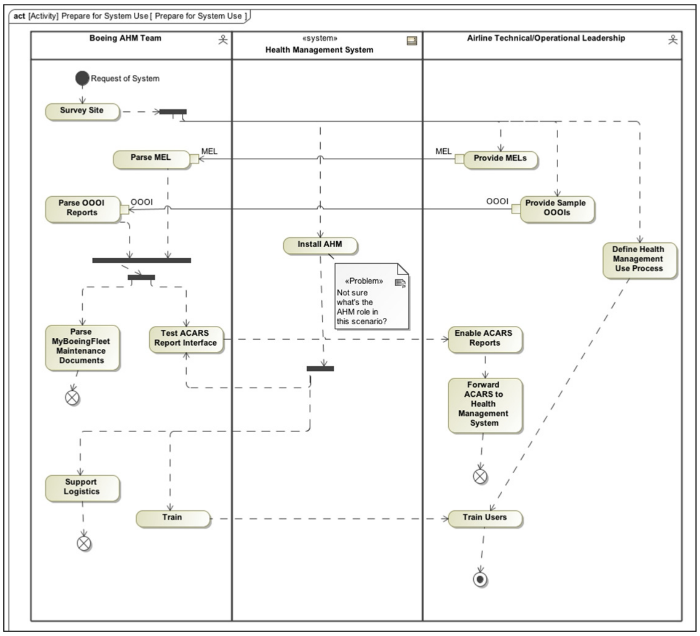


Figure 4. Prepare for System Use (Activity Diagram)

## *3.3. Mapping System Models to Storytelling Space*

A common practice in defining system goals and stakeholder needs is to begin by creating representative operational scenarios (use cases) that systems engineers (SEs) use to collaborate with subject matter experts (SMEs) and non-technical stakeholders. It is common that operational scenarios are documented primarily in prose form in operational scenario documents. These documents tend to be subject to inconsistencies, ambiguities, and gaps. Not surprisingly, systems engineers prefer formal, structured diagrams such as use cases and activity diagrams. However, most SMEs and non-technical stakeholders tend to be unfamiliar with systems engineering methods and formal modeling notations. Thus, effective collaboration between systems engineers, SMEs, and other non-technical stakeholders is a major challenge. This recognition provides the impetus for adding an experiential perspective to the perspectives currently provided in MBSE. The addition of experiential perspectives, derived from storytelling in virtual worlds, to MBSE is what we call MBSE+.

A story enacted within virtual worlds, can dramatically enhance collaboration among all stakeholders, while circumventing potential confusion that typically results when free-text is used to describe operational scenarios and system goals. A story can alo be represented using a formal language. Then, a relatively straightforward bi-directional mapping can be constructed between system models and the story space (Table 1).

Table 1. Mapping Between Design Space and Storytelling Space is not one-to-one[3]

| SYSTEM MODEL (Design Space) | VIRTUAL WORLD (Storytelling Space) |
|---|---|
| **Structural and Behavioral Views** | **Structure and Behavior** (system behavior emerges as subsystems interact with each other, and system interacts with story characters; collective interactions alter technical story trajectory and potentially mission outcomes) |
| **Use Case(s) and Activity Sequence Diagrams** | **Technical Story** (how system is used, by whom, and under what conditions; contextualizes use case(s) and sequence diagrams; allows stakeholders to collaborate and provide feedback on needed system changes) |
| **Humans and Roles** | **Technical Story Characters** (story participants, or story narrator, SMEs; interact with system as story unfolds; have personalities and preferences that influence their interaction with system |
| **Viewpoints** | **Stakeholder Lenses** (author-defined; eventually constructed on demand) |
| **Basic Flow of Events** | **Main Storyline** (defines events, characters, and interactions between and among them) |
| **Alternate Flow of Events** | **Vignettes** (multiple clusters of interactions among specific characters who interact with the system as story unfolds) |
| **Exception Flow of Events** | **Surprise Vignettes** (represent twists in technical storyline and/or unexpected or surprising ending resulting from what-if changes to system properties and resulting system behavior; illuminate system resilience to unanticipated/emergent conditions) |
| **User-System Interactions** | **Character interactions** (define which characters interact with the system, and how under different contexts) |
| **Task Prerequisites/ Dependencies** | **Character Task Performance Requirements** (information that system needs to provide to characters to perform tasks) |
| **System Composition and Human Interaction** | **Extended Family Tree** (defines how subsystems relate to each other and to the characters in the story) |

It is important to note that the virtual world is a domain-limited yet verifiable representation of the system of interest. Verification of the system representation is performed collaboratively by all stakeholders in terms of the ability of the virtual world representation to support the different use cases of interest and to answer questions posed by stakeholders from their respective perspectives. For example, a SME can interact with the virtual world through a simulation interface to investigate how the system behaves in a variety of situations in the virtual world, especially at the boundaries.

It is important to note that only a subset of system model elements will map to entities in the virtual world. The remaining entities that populate the virtual world are derived from the technical stories, which are informed by use case scenarios, system models, and SME knowledge. Story vignettes also provide the context that facilitates the mapping of system model artifacts to virtual world entities. The mapping of system elements, events, actors, parameters, and relationships into engaging technical narratives can be partially automated. In sum, the technical storytelling environment allows all stakeholders to explore system usage and behavior from their own perspective for a variety of conditions and with different assumptions.

### *3.4. Story Execution and Data Collection in Virtual Worlds*

Stories and story vignettes are developed using a combination of: operational scenario use cases, system models, and expert knowledge. Story vignettes have embedded conditions in the operational scenarios that help illuminate system behaviors and create insightful experiences in virtual worlds. The technical storyline and each vignette have specific entry conditions (i.e., what is true when the story/vignette begins), triggers for each vignette, and exit conditions (i.e., the terminating conditions for each vignette/story). It is important to note that the storyline is nonlinear and depends on the interactions between the system and characters/entities in the simulated environment. By being able to interact and explore system behaviors in the virtual world, stakeholders can: increase their understanding of system behaviors; explore and critique system behaviors under a variety of "what-if" circumstances; and suggest changes to system design from their respective perspectives (i.e., lenses). The lenses can reflect the perspective of a particular role/function or an impact factor that the stakeholder is interested in. An example of a role/function related factor is "communication." Example of impact factors for this illustrative example are cost, schedule, risk, airline delay, maintenance turnaround time, airplane diverts, cancelled flights, and customer satisfaction. Stakeholder feedback to system engineers/modelers are in the form of recommended changes to system behaviors and other key properties (e.g., appearance, interactions with other entities). Stakeholders can also offer opinions on the usability and overall utility of the system. Some of the specifics of story execution are discussed next.

*Role-specific and function-specific lenses.* Role-specific and function-specific lenses are a means to make a complex system comprehensible to all stakeholders and allow them to provide meaningful inputs about system behavior from their respective perspectives. Role-specific lenses allow users to view the interactions within and outside the system from the perspective of a particular role. In the aircraft divert scenario, examples of roles are: pilot; flight crew; maintenance controller; ground crew/line maintenance; and airline operations center. Examples of function-specific lenses are: fault history; communications; fault status; prioritized faults; and forwarded faults.

*User-system interaction prototyping.* The purpose of user-system interaction prototyping[12] (i.e., horizontal prototyping) in a virtual world is to: a) create technical story-driven system experiences in the virtual world for all stakeholders to interact with and provide feedback that can be delivered to system modelers; b) partially populate the virtual world with entities and relationships contained in and addressed by the horizontal prototype.[12] The technical story provides contextual information such as location, environment entities, terrain, weather, and visibility. Collectively, these information items help create an powerful experiential environment for stakeholders to interact with and develop an understanding of the system.

*Simulated story vignettes execution*. The execution of story vignettes is an "interactive step through" the technical story in which time-stamped scenes are presented to stakeholders in a partially ordered sequence to create semi-immersive experiences. The behavior of the system becomes apparent to stakeholders as the technical story unfolds. Eventually, the human step-through can be replaced with automated story sequencing. The Unity3D game engine can also be used to accomplish the latter.

*Data Collection and System Model Refinement.* As the technical story executes in the real world, tacit knowledge and dependencies gradually become explicit as various characters interact with the system and with each other. This information is collected and used to refine the initial set of system models, especially use cases and activity sequence diagrams (Figure 1). Also, user-system interaction data is used to refine the system concepts of operation (CONOPS), user-system interactions, and the user interface.

## 4. Implications and Outlook for the Future

The intent of the Experiential Design Language (EDL) is to engage all stakeholders early and often in complex systems engineering. The business motivation for EDL stems from the need to reduce time and cost of system development while enhancing stakeholder/user acceptance. For a company such as Boeing that makes commercial aircraft for the airline industry, the business motivation goes farther in that it extends to the business impact on its customers. Flight cancellations, aircraft delays, and diverted flights have enormous cost impacts on the airline industry, not to mention customer defections to competing airlines. Reduction in diverted aircraft through full utilization of AHM system-supplied information can dramatically reduce airline costs while also ensuring customer retention. Full utilization of AHM-provided information can only occur with full understanding of AHM capabilities on the part of all stakeholders. This understanding, in large part, is made possible by the EDL which provides an experiential perspective to MBSE. The MBSE+ framework, which employs principles of elegant design, adds the experiential perspective supplied by EDL to existing MBSE perspectives, thereby enhancing the participation of and collaboration among all stakeholders. Future work includes the development of an automated interface for bi-directional information exchange between the system models and the technical storytelling environment.

## Acknowledgement

This project is being conducted in cooperation with Professor Alex McDowell of the University of Southern California's School of Cinematic Arts. Douglas Orellana, Ph.D. candidate in USC's Systems Architecting and Engineering Program, performed the systems modeling for the illustrative example. Dr. Michael Sievers of Caltech's Jet Propulsion Laboratory provided helpful comments that contributed to improving the overall quality of the paper. Mr. David Kinney of Boeing served as subject matter expert for the aircraft divert mission scenario.

## References

1. Friedenthal, S., Moore, A., and Steiner, R. (2012) *A Practical Guide to SysML, Second Edition: The Systems Modeling Language* (The MK/OMG Press), Elsevier, Inc.
2. Dori, D. (2012) *Object-Process Methodology – A Holistic Systems Paradigm*, Springer Verlag, New York.
3. Madni, A.M. "Expanding Stakeholder Participation in Upfront System Engineering Through Storytelling in Virtual Worlds, submitted to *Systems Engineering* in August 2013.
4. Jenkins, H. Transmedia Storytelling, MIT Technology Review, January 2003.
5. Woo, H.R. A Holistic Experiential Approach to Design Innovation, in *Advances in Human Computer Interaction* (Ed. Shane Pindler), October 2008.
6. Dalsgaard, P. Experiential Design: Findings from Designing Engaging Interactive Environments, in *Advances in Human Computer Interaction* (Ed: Shane Pindler), InTech, October 2008.
7. Lakoff, G. and Johnson, M. (1980) *Metaphors We Live By*, Chicago: Chicago University Press.
8. Hubbard, W. Experiential Design Language: Case Study of Diverted Flight Scenario, 2013.
9. Griffin, M.D. (2010) How do we Fix Systems Engineering? IAC-10.D1.5.4, *61st International Astronautical Congress*, Prague, Czech Republic, 27 September – 1 October, 2010.
10. Madni, A.M. "Elegant Systems Design: Creative Fusion of Simplicity and Power, *Systems Engineering*, Vol. 15, No. 3, 2012.
11. Madni, A.M., Lecture on Elegant Design, SAE 549, Systems Architecting, 2012 to present
12. Madni, A.M. (1988) The Role of Human Factors in Expert Systems Design and Acceptance, *Human Factors Journal*, August 1988, 30(4), 395-414.