

Extending MBSE Methodology and SysML Formalism to Integrate Human Considerations

Camille Raymond, Daniel Prun

▶ **To cite this version:**

Camille Raymond, Daniel Prun. Extending MBSE Methodology and SysML Formalism to Integrate Human Considerations. HCI-Aero 2016, International Conference on Human-Computer Interaction in Aerospace, Sep 2016, Paris, France. hal-01367016

HAL Id: hal-01367016
https://enac.hal.science/hal-01367016

Submitted on 1 Mar 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Extending MBSE methodology and SysML formalism to integrate human considerations

Camille RAYMOND
Altran / ENAC
4, avenue Didier Daurat
31700 Blagnac
camille.raymond@altran.com

Daniel PRUN
ENAC, Université Fédérale de Toulouse
7 avenue Édouard-Belin
31055 Toulouse, Cedex 4
daniel.prun@enac.fr

ABSTRACT
In response to the challenges related to integrate human considerations within complex interactive systems, the Human-System Integration (HSI) initiative has been proposed.

Nevertheless, the HSI approach is difficult to fit efficiently into the system engineering processes because of the lack of shared methodologies, modeling semantics and tools.

This paper focuses on the needs for a new methodology and an adapted formalism to improve the collaboration between the system engineers and HSI practitioners. It concludes with the presentation of our methodology based on the Model-Based System Engineering (MBSE) and the SysML modelling language.

Keywords
System Engineering, Human-System Integration, SysML, MBSE

INTRODUCTION
Integrating human aspects within complex systems raised many issues in system design. Usually, industry tends to refer to work procedures to supervise operators during their activities. These measures showed some limits as they considered human as a limitation factor in a system. Indeed, in seeking to restrict human influence, they consequently limit human capacities of reflection and adaptation.

Thus, the Human-System Integration (HSI) approach has been proposed in favor of improving the considerations of the human characteristics in interactive systems design. Nevertheless, this alternative is difficult to fit efficiently into system engineering processes due to the inadequate methodologies, formalisms and tools allowing an efficient inter-integration of both disciplines [1].

This paper focuses on the need for a methodology to enhance the collaboration between system engineers and HSI practitioners through MBSE methodologies and SysML formalism. It presents our contribution lines through an aeronautical maintenance study case.

SYSTEM ENGINEERING AND MBSE
The system engineering is defined as “an engineering discipline whose responsibility is creating and executing an interdisciplinary process to ensure that the customer and stakeholder’s needs are satisfied in a high quality, trustworthy, cost efficient and schedule compliant manner throughout a system’s entire life cycle” [2]. The ISO/IEC 15288 [3] defines the processes involved in a project life cycle including: requirements analysis, architectural design, implementation, integration, verification and validation.

A system is expressed by a set of concerns (technical, organizational, financial, safety, human considerations etc.) and implies various stakeholders during the system life cycle (client, engineers, human factors practitioners, developers, testers, etc.). It is necessary to have standards, methodologies and tools to carry the system engineering activities. They allow to express and evaluate different system architectures tradeoffs, and elect afterthought the most appropriate solution satisfying all the concerns according to the resources available.

Thus, the system engineering community proposes the Model-Based System Engineering (MBSE) methodologies to support the system engineering processes. MBSE is defined as the “formalized application of modelling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase and continuing throughout development and later life cycles phases” [4]. MBSE methodologies depict the common system architecture breakdowns: operational layers, functional layers, logical layers and physical layers.

The MBSE methodologies substitute the traditional document-centric approach to a model-centric approach. The document-centric approach tends to produce a large amount of documents with various types of artifacts, potentially used in an inconsistent manner and ambiguous due to the use of natural language. The MBSE methodologies rely on centralized and up to date models, shared with all the stakeholders. They provide greater system design consistency and completeness, help to manage architecture complexity and improve the communication between stakeholders [5].

Contrary to the document-centric approach, models provide the advantage to be dynamically manipulated and

transformed. Through the viewpoint mechanism, MBSE allows the representation of models from the perspective of a related set of concerns [6]. Applying a viewpoint reduces the scope of an entire model to only present the interesting parts from a particular perspective. Besides, the model transformation mechanism produces new models from other models in input. For example, an analytical model for a simulation purpose can be produced from a descriptive model.

The MBSE traceability mechanism ensures the navigation through the layers. It allows system engineers to guarantee the consistency and the completeness of the models.

As presented, MBSE methodologies present many advantages to cover the system engineering processes. Thanks to the viewpoint, the model transformation and the traceability mechanisms, they allow various stakeholders to collaborate to the entire system architecture during the project life-cycle.

Nevertheless, studies [7] [8] highlight the fact that the human element is not adequately taken into consideration in systems design. They point the recurrent issues of the system engineering to usually address human considerations afterthought the system architecture has been already specified and designed. The ISO/IEC 15288 remains vague regarding human involvements within a system. Mavor and Pew [7] lists the shortcomings that lead to human integration problems:

- Human factors integration trends and standards are not captured
- Human performance metrics, targets, and limitations are not specified
- Human role design, job design, and organizational design are insufficiently captured
- Team activity and team requirements are insufficiently captured

In response to the challenges, the Human-System Integration (HSI) initiative has been proposed.

HUMAN-SYSTEM INTEGRATION

Human-System Integration (HSI) is defined as "a multidisciplinary approach of integrating human, technical and organizational aspects at a system level, with full considerations given to the human requirements" [1]. HSI frameworks give human factors practitioners the means to define human characteristics and constraints. Besides, they give systems engineers the tools to integrate these considerations into systems design. HSI extends tradeoff evaluations and agreements comparatively to the technical, organizational and human concerns.

HSI covers the following domains: human factors engineering, system safety, health, environment, habitability, survivability, training, personnel and manpower. These domains reflect how the human component impacts on the system performances and reciprocally, how a system influences the human component. HSI needs to address both human capabilities and system design within the system engineering processes.

Figure 1 illustrates the HSI model in terms of inputs and outputs [1]. HSI process is implied during different stages of the system development (*systems definition*, *systems development* and *systems deployment*). It also focuses on the user needs during all the system design steps (*highly concentrated user focus*) and exploits disciplines related to human studies (*human related technologies & disciplines*). The HSI process output is the system designed (*technology*), integrating human aspects (*people and organization*).

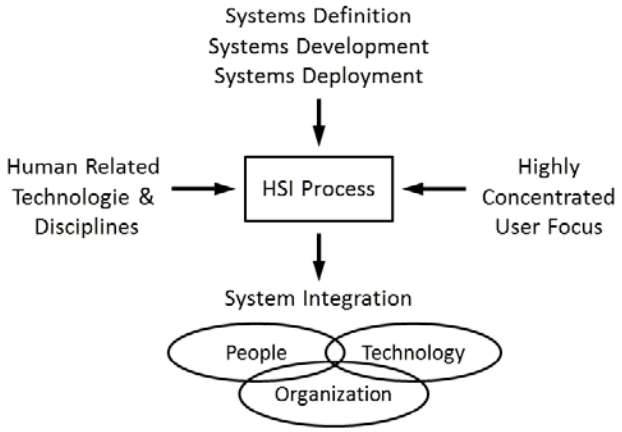


Figure 1 HSI Process [1]

Although relevant in many aspects, several studies highlighted major limitations to effectively integrate HSI in the system engineering processes.

Firstly, studies [7] [9] identified a lack of shared methods, tools and formalisms precluding a meaningful communication and collaboration across system engineers, HSI practitioners and other personnel implied in system engineering. Current HSI tools do not take sufficiently into account the system development process. Without these means, system design efforts will continue to be inconsistent, incomplete and redundant since these domains are not able to collaborate.

Secondly, [10] asserts that the integration of HSI within system engineering is not just a technical issue, but also cultural and organizational. New semantics are needed to enlarge the modeling semantics scope to the HSI semantics. This new semantics will allow human elements to be considered in the whole system perspective.

THE NEED FOR AN EXTENDED FORMALISM

As previously explained, one major limitation of HSI and system engineering disciplines to collaborate is the lack of formalism adapted.

In one hand, SysML modeling language is standardized and widely adopted in the system engineering field. In the other hand, HSI discipline deals with a large variety of non-standardized formalisms, each framework providing its own.

SysML provides the necessary means to encompass the HSI formalisms by extending its meta-model and including dedicated profiles. Several projects focus on the SysML extension to include HSI formalisms [10] [11] [12] [13].

One of the most interesting features of SysML is its ability to support the simulation of the designed system. The existing solutions to simulate a system regarding both technical and human aspects are currently restricted.

Traditionally, system engineers use simulation tools to analytically test the system behavior and properties according to the requirements and specifications. Besides, human factors engineers developed models [14] [15] [16] which simulate human behaviors according to different points of view: fatigue, stress, cognitive workload, performance etc.

The sticking point is that there are very few links between system engineering and human factors models for simulation. They remain fragmented on their objectives, semantics and tools. The possibilities of interfacing them remain poorly studied. This constitutes obstacles to fulfill the requirements and the quality of the system.

A great effort needs to be done to implement the model transformation mechanism, to transform descriptive models to analytical models allowing the simulation objectives.

In response to these challenges, the U.S. Army Research Lab developed the Improved Performance Research Integration Tool (IMPRINT) to model human-system tasks in order to analyze functional allocation and workload estimation [17]. The IMPRINT tool provides descriptive and analytical models. The descriptive models are comparable to the *Activity Diagram* SysML diagram since it defines the tasks process fall to the system and the actors. The analytical models run simulation based on the descriptive models to analysis the task allocation in terms of workload levels.

The IMPRINT approach is interesting in a MBSE context since it uses models to evaluate a design architecture in terms of technical and human factors points of view. Efforts must be done in this direction because it is not possible yet to fully integrate IMPRINT in a systemic models supported by MBSE.

THE NEED FOR AN EXTENDED METHODOLOGY

As Orellana [11] explained: “extending the modeling language is just a part of the extension needed. To extend MBSE to the specialty system engineering fields, not only do engineers need common semantics, but a way to integrate these semantics into current methods, processes, and tools being used by specialty system engineers”.

This assertion highlights the need of a methodology to support system management and system design integrating technical and human aspects.

New methodologies are needed to guide the collaboration enhancement between system engineering and HSI disciplines.

First of all, an organizational plan needs to be established for monitoring both system engineering and HSI disciplines to ensure an effective collaboration. Muralidhar [18] warns against the fact that one discipline prevails over the other one. Typically, system engineers tend to relegate aside human considerations aside because they do not know how to integrate these aspects into the system design. Developing a close relationship between system engineers and HSI practitioners is an important leverage to ensure the development of safe and effective systems. Dedicated supervisors should be allocated to manage the risk of disparities of concern in the design process.

Secondly, it is necessary to manage the systemic HSI process (Figure 1). The MBSE methodologies provide a favorable material to support it. They allow engineers to model the whole system, from the requirements analysis to the physical integration. Within this frame, HSI should focus on integrating their efforts on these models. The extended formalism and the traceability mechanism allow them to inspect the human considerations consistency and completeness through the entire system design.

Then, the methodology needs to be extended to guarantee that the Verification and Validation processes (V&V) consider human capabilities and constraints. The V&V activities are critical to ensure the overall quality of the system. Verification refers to the process of determining whether the system fulfill the specified requirements. Validation refers to the evaluation of the design applied complies with the functional and the performance requirements. Simulation, as described early, can assist system designers to perform these process.

CONTRIBUTIONS

The objective of this study is to highlight the lack of formalism and methodology precluding a meaningful collaboration between system engineering and HSI. In this context, it is difficult to design a system guaranteeing the mandatory criteria of performance and safety within modern complex interactive systems.

Our research is guided by the need of a system engineering methodology adapted fora the integration of human factors issues, paying particularly attention to take these aspects into account across all the system layers. We consider that MBSE provides a relevant frame to support this methodology. As presented, MBSE guides the modelling activities through all the system engineering process and ensures the completeness and the consistency of the system design.

Then, we regard SysML as an adapted formalism to support our future methodology. Already widespread in the system engineering community, SysML provides the means to integrate HSI modelling semantics thanks to the model transformation, profile and viewpoint mechanisms. An extended SysML will facilitate the involvement of human factors practitioners within system design.

Our research will be proved on an aeronautical maintenance study case. Maintenance is central to navigation safety since it prevents aircraft failures. This activity represents an important financial cost for airlines, and it is implied in flights delays, cancellation and also accidents. Maintenance activities involve a high level of cognitive tasks within a stressful environment. Operators are potentially subject to high stress and high work load levels which deteriorate their performance and reliability.

Our methodology proposes to:

- Carry on the extension of SysML with HSI to better integrate human aspects. In our case, a maintainer can be defined by its experience, its physical and cognitive tasks involved in its activity, its stress, its fatigue, its relation with its organization and so on.
- Trace these human aspects through all the MBSE layers. For example, the system used by the maintainers needs to be adapted to the current state of its user to ensure its performance. Then, the system could provide a dynamic functional allocation mechanism according to these aspects. Such an implementation impacts simultaneously all the system architecture layers (operational, functional, component and physical).
- Extend simulation models to include human aspects. As presented, the IMPRINT tool provides an interesting approach to evaluate human factors aspects within a system. Based on it, we will carry on the implementation of a model transformation mechanism to produce analytical models from descriptive models to simulate simultaneously technical and human aspects. This effort will allow to trade the system off to elicit the most adapted system design.
- Develop a methodology to supervise the HSI efforts within the system engineering processes.

REFERENCES

1. Booher, H.R. *Handbook of human systems integration*. John Wiley & Sons, 2003.
2. INCOSE. *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*. Wiley, 2015.
3. ISO/IEC/IEEE15288:2015. *Systems and software engineering - System life cycle processes*. Geneva, 2015.
4. INCOSE. *Systems Engineering Vision 2020*. 2007.
5. Estefan, J. Survey of model-based systems engineering (MBSE) methodologies. *Incose MBSE Focus Group*, 25, 8 (2007).
6. IEEE. IEEE Recommended Practice for Architectural Description of Software-Intensive Systems. *IEEE Std 1471-2000* (2000), i-23.
7. Mavor, A. S. and Pew, R. W. *Human-System Integration in the System Development Process: A New Look*. National Academies Press, 2007.
8. Hardman, N.S. *An Empirical Methodology for Engineering Human Systems Integration*. 2009.
9. Madni, A. M. Integrating humans with and within complex systems. *CrossTalk*, 5 (2011).
10. Orellana, D. W. and Madni, A. M. Human System Integration Ontology: Enhancing Model Based System Engineering to Evaluate Human-System Performance. *Procedia Computer Science*, 28 (2014), 19-25.
11. Orellana, D. W. and Madni, A. M. Extending Model-Based System Engineering for Human Machine Interaction Analysis and Fault Tolerant Design. *Infotech@ Aerospace 2012* (2012), 2537.
12. Bruseberg, A. Human views for MODAF as a bridge between human factors integration and systems engineering. *Journal of Cognitive Engineering and Decision Making*, 2, 3 (2008), 220-248.
13. Prun, D. Extending eFFBD formalism to task model. *INCOSE International Symposium*, 24 (2014), 67-81.
14. Paterno, F. *Model-based design and evaluation of interactive applications*. Springer Science & Business Media, 2012.
15. Vanderdonckt, J. Model-driven engineering of user interfaces: Promises, successes, failures, and challenges. *Proceedings of ROCHI*, 8 (2008).
16. Lebiere, C, Anderson, J. R., and Bothell, D. Multi-tasking and cognitive workload in an ACT-R model of a simplified air traffic control task. *Proceedings of the Tenth Conference on Computer Generated Forces and Behaviour Representation* (2001).
17. Allender, L. Modeling human performance: Impacting system design, performance, and cost. *Simulation Series*, 32, 3 (2000), 139--144.
18. Muralidhar, A. How human systems integration and systems engineering can work together. *Insight*, 11, 2 (2008), 11-14.
19. Hause, M. The Unified Profile for DoDAF/MODAF (UPDM) enabling systems of systems on many levels. In *Systems Conference, 2010 4th Annual IEEE* (2010), 426-431.