



**HAL**  
open science

## Towards model-based systems engineering (MBSE) patterns to efficiently reuse know-how

Quentin Wu, David Gouyon, Pascal Hubert, Eric Levrat

► **To cite this version:**

Quentin Wu, David Gouyon, Pascal Hubert, Eric Levrat. Towards model-based systems engineering (MBSE) patterns to efficiently reuse know-how. *INSIGHT - International Council on Systems Engineering (INCOSE)*, 2017, 20 (4), pp.31-33. 10.1002/inst.12178 . hal-01666623

**HAL Id: hal-01666623**

**<https://hal.science/hal-01666623>**

Submitted on 18 Dec 2017

**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.

## **Towards model-based systems engineering (MBSE) patterns to efficiently reuse know-how**

Quentin Wu, [quentin.wu@zodiacaerospace.com](mailto:quentin.wu@zodiacaerospace.com); David Gouyon, [david.gouyon@univ-lorraine.fr](mailto:david.gouyon@univ-lorraine.fr); Pascal Hubert, [pascal.hubert@zodiacaerospace.com](mailto:pascal.hubert@zodiacaerospace.com); Éric Levrat, [eric.levrat@univ-lorraine.fr](mailto:eric.levrat@univ-lorraine.fr)

Engineers from all domains have always been using their accumulated knowledge in order to develop new systems and reduce engineering phases during the lifecycle of a project. But, facing the increasing complexity of systems, difficulties emerge to formalize expert know-how, making it laborious to reuse (or at a great cost of energy, time and money). To deal with this complexity, MBSE has been introduced to shift the document-centric approach practiced by engineers towards a model-centric approach, that “integrates system requirements, design, analysis, and verification models to address multiple aspects of the system in a cohesive manner” (Friedenthal, Moore, and Steiner 2008). One consequence of this formalized application of modeling is opening the way to search new forms of expressible and reusable know-how to improve engineering activities. To face this reuse challenge, this short article is presenting three approaches: commercial off-the-shelf (COTS), set-based design and patterns.

### **State of the art of knowledge reuse**

These approaches belong to the process of "knowledge transfer" which consists of two sub-processes defined by (Majchrzak, Cooper, and Neece 2004). On one hand, the process by which an entity's knowledge is captured, called "knowledge sharing", and on the other hand, the process by which an entity is able to locate and to use knowledge, called "knowledge reuse". This paper focuses on the latter and presents three approaches that seem to be most likely to improve engineering activities.

The COTS approach can be seen as a "divide and conquer" design paradigm, as it introduces the notion of modularity at the software and hardware level of system. It consists in breaking down a problem into solvable sub-problems by already existing components. Consequently, (Hedman and Andersson 2014) pointed out that COTS may be selected and implemented for technical (less development time), business and organizational (reduce overall system development costs), and strategic reasons (access a technology that cannot be developed internally). However, in systemic thinking, the "whole" is greater than the sum of its "parts". For this reason, the advantages of COTS are accompanied by integration issues, early identified by (Boehm and Abts 1999) which are : functionality and performance (what it is expected to do), interoperability (no standard exist), product evolution (risk of no longer meeting the need) and vendor behavior (false promises). Beyond these concerns, it appears that the key to use COTS is the need of an efficient selection method, which understands companies' needs and situations to improve integration. This is a critical point when trying to start a MBSE project as stakes are high in terms of both time and budget. Thus COTS-based tools in the market does not provide the same usability, functionality, interoperability, technical support,... That is why (Friedland, Malone, and Herrold 2016) are customizing their own MBSE tools based on a COTS in order to answer their needs.

The "set-based design" approach consists in a paradigm shift for industries. Currently, (Kennedy, Sobek, and Kennedy 2014) describes a fumbling situation where companies spend a long time to build their products, and test it at the end of the development process. What often occurs is that rework is needed, which implies to spend time again to develop a new version of the product to test. (Sobek, Ward, and Liker, Jeffrey 1999) called this paradigm "point-based design", as the development team is moving from one "point" solution to

another one more mature and closer to the customer needs. (Kennedy, Sobek, and Kennedy 2014) emphasizes on the fact that an inversion of the paradigm is needed, and from their point of view, it is feasible by transferring the load toward the "front" phases of engineering activities. It implies for the development team to innovate on small tests that can help them define and challenge the limit of a technology in order to establish a set of possible designs. They can then build the system thanks to all the knowledge accumulated from these maturation activities. That is the reason why "point-based design" can be seen as a "document-based design" as requirements and specifications have been managed with documents and natural language, resulting in many iterations. As a result, "set-based design" and MBSE approaches aim at the same goals, which are reducing rework and improving decision-making. However, even if models support decision maker by presenting just the information needed, it still needs to be improved (Russell 2012). That is why this paradigm shift complements modeling approach on decision-making by systematically accumulating knowledge and defining a set of possible designs (by maturation) before making key decisions.

The third approach is based on the concept of "pattern", described for the first time in architecture by (Alexander, Ishikawa, and Silverstein 1977), then promoted in software engineering by (Gamma et al. 1993; Bushmann, Meunier, and Rohnert 1996). In the field of systems engineering, (Barter 1998) and also (Haskins 2003), were the first to introduce pattern and pattern language for capturing the engineering knowledge. (Pfister et al. 2012) adapted the concept to functional architecture design and describes patterns with enhanced function flow block diagram (eFFBD), which is translatable into SysML, allowing a bridge with MBSE. But other works are trying to link MBSE and patterns. (Schindel 2005) based his work on the hypothesis that systems engineering should be a merge of prose and diagrams in order to create a formal model. He sees "patterns as re-usable models" and applies it to requirements and design. This modelling framework has led to an INCOSE working group called MBSE Patterns (pattern-based systems engineering) and where patterns can be configured or specialized into product lines or into product systems. At a high-level, they constitute a generic system pattern model that can be customized for an enterprise's needs, configuration, use, so that engineers can benefit from the concepts of MBSE without being an expert of modeling methodologies.

In this section, three approaches have been presented, and possible links with MBSE have been considered.

## **Discussion**

From the three approaches presented in the previous section, arguments regarding "knowledge reuse" have not been discussed even if bridges with MBSE have been made explicit. As "knowledge reuse" is one key element when engineers seek to reduce rework and system development costs, these goals raise issues concerning knowledge management to perform efficient reuse. This discussion focuses on the pattern approach, as its configurability property and its capacity to combine with MBSE is more likely to help in the search for new forms of expressible and reusable knowledge.

The engineering artefact on which a pattern is applied may be the system of interest (SOI) or systems engineering activities (SEA) – such as requirement engineering, functional architecture design, and physical architecture design – whose aim is to produce a model of the SOI (Pfister et al. 2012). Both should be addressed in order to propose a broad approach for generic patterns based on MBSE methodology. But, to keep consistency between patterns – as they do not target the same objectives – they must not be mixed: one pattern address only one aspect (SOI or SEA). For example, COTS address a solution for the SOI, thus it may be possible to apply the same criteria in order to build SOI patterns, but a particular attention

must be paid concerning the pattern's configurability contrary to COTS. Another example would be to address a pattern concerning "set-based design" as a SEA pattern, but this would need to correctly address the principle of this approach in MBSE, which mean identify the level of abstraction or how to model it.

Beyond these aspects, a very important dimension appears when trying to practice "knowledge reuse" for innovation. Indeed, reuse cannot be applied like a copy and paste process as it also should facilitate innovation by allowing engineers to create new ideas from expressed knowledge. It is true that capitalizing on experiences happened when a solution is known and frequently used in the field. But, this would mean missing the separation of "knowledge reuse" detailed by (Majchrzak, Cooper, and Neece 2004) in two processes : one for replication, and the other for innovation. They identified a six stages process, which is more likely to foster innovation in the specific context of the Jet Propulsion Laboratory (JPL). But what drives JPL can possibly drive other institutions, thus it seems possible to build a SEA pattern concerning innovation. And as it appears that the process can be defined by use cases and scenarios (depending on the actors), and a functional flow, an MBSE approach can be considered.

To go further, a process of knowledge reuse combined with an MBSE approach that follows lean (McManus, Haggerty, and Murman 2005) or frugal (Weigl, Wang, and Sepahvand 2012) philosophies seems to make sense. Indeed, one issue that appears when applying MBSE concerns the level of abstraction a model is addressing. These ways of thinking can help engineers to find the level of modeling that is complete enough to provide the right information for the right stakeholders.

What emerges from this discussion, is the need to practice reuse in a smart way, in order to benefits from engineers feedback for improving engineering phases, while enabling innovation and bridges towards MBSE

## Conclusion

The three approaches presented in this article have raised many issues, which must strive stakeholders for developing efficient "knowledge reuse" in order to reduce costs, time, and facilitate innovation during engineering phases. The coupled approach between MBSE and "knowledge reuse" has shown promising outlook and especially concerning the concept of pattern which appears to be a possible part to the answer regarding the growing complexity of systems, as it is generic and does not follow a unique method.

## References

- Alexander, C, S Ishikawa, and M Silverstein. 1977. *A Pattern Language*. Ch. Alexander. doi:10.2307/1574526.
- Barter, Robert H. 1998. "A Systems Engineering Pattern Language." *In cose*, 350–53.
- Boehm, B., and C. Abts. 1999. "COTS Integration: Plug and Pray?" *Computer* 32 (1): 135–38. doi:10.1109/2.738311.
- Bushmann, F, R Meunier, and H Rohnert. 1996. "Pattern-Oriented Software Architecture: A System of Patterns." *John Wiley&Sons* 1: 476. doi:10.1192/bjp.108.452.101.
- Friedenthal, Sanford, Alan Moore, and Rick Steiner. 2008. *A Practical Guide to SysML: The Systems Modeling Language*. Edited by Elsevier. Morgan Kaufmann.
- Friedland, Brittany, Robert Malone, and John Herrold. 2016. "Systems Engineering a Model Based Systems Engineering Tool Suite : The Boeing Approach." In *INCOSE International Symposium*.
- Gamma, Erich, Richard Helm, Ralph Johnson, and John Vlissides. 1993. "Design Patterns: Abstraction and Reuse of Object-Oriented Design." *Lecture Notes in Computer Science* 707 (Mvc): 406–31. doi:10.1007/3-540-47910-4\_21.

- Haskins, Cecilia. 2003. "1.1.2 Using Patterns to Share Best Results - A Proposal to Codify the SEBOK." *INCOSE International Symposium* 13 (1): 15–23. doi:10.1002/j.2334-5837.2003.tb02596.x.
- Hedman, Jonas, and Bo Andersson. 2014. "Selection Method for COTS Systems." *Procedia Technology* 16. Elsevier B.V.: 301–9. doi:10.1016/j.protcy.2014.10.096.
- Kennedy, Brian, Durward Sobek, and Michael Kennedy. 2014. "Reducing Rework by Applying Set-Based Practices Early in the Systems Engineering Process." *Systems Engineering* 17 (3). doi:10.1002/sys21269.
- Majchrzak, Ann, Lynne P. Cooper, and Olivia E. Neece. 2004. "Knowledge Reuse for Innovation." *Management Science* 50 (2): 174–88. doi:10.1287/mnsc.1030.0116.
- McManus, Hugh L, Al Haggerty, and Earl M. Murman. 2005. "Lean Engineering : Doing the Right Thing Right." *1st International Conference on Innovation and Integration in Aerospace Sciences*, no. August: 1–10.
- Pfister, F, Vincent Chapurlat, M Huchard, C Nebut, and J.-L. Wippler. 2012. "A Proposed Meta-Model for Formalizing Systems Engineering Knowledge, Based on Functional Architecture Patterns." *Systems Engineering* 15 (3). doi:10.1002/sys.21204.
- Russell, Mike. 2012. "Procedia Computer Science Using MBSE to Enhance System Design Decision Making." *Procedia Computer Science* 8: 188–93. doi:10.1016/j.procs.2012.01.041.
- Schindel, William D. 2005. "Requirements Statements Are Transfer Functions: An Insight from Model-Based Systems Engineering." *INCOSE International Symposium* 15 (1): 1604–18. doi:10.1002/j.2334-5837.2005.tb00775.x.
- Sobek, Durward, Allen C Ward, and K Liker, Jeffrey. 1999. "Toyota S Principles of Set-Based Concurrent Engineering." *Sloan Management Review*, no. Winter: 67–83.
- Weigl, Joerg, Zizi Wang, and Helia Mohammadi Sepahvand. 2012. "Hydrogen Refueling Infrastructure Design for Personal Mobility Devices Using Frugal Engineering Approach." *Energy Procedia* 29: 668–75. doi:10.1016/j.egypro.2012.09.077.