

Analysis of Systems Structural Relationships Through a DSM-Based Approach

Yaroslav Menshenin¹, Yana Brovar¹, Edward Crawley², Clement Fortin¹

¹ Skolkovo Institute of Science and Technology

² Massachusetts Institute of Technology

Abstract: DSMs and related matrices are commonly used to represent system interfaces, which is closely associated with a need to decompose systems into their elements. However, besides the importance of managing the decomposed elements of systems, the systems engineer should also possess a capability to manage other systems' relationships. In this paper we are focusing on the DSM capability to represent three fundamental system entities and four structural relationships, as they are defined in the Object-Process Methodology. We argue that DSM is an appropriate tool to facilitate a systematic approach to digital transformation of objects and processes through the different stages of the design process. We use a coffee maker as a case study demonstrating how each entity flows from an abstract environment to a more detailed one following specific structural relationships.

Keywords: DSM, system relationships, conceptual design, digital transformation

1 Introduction

One of the core tasks of systems engineers is to identify and to manage the systems relationships. The Design Structure Matrix (DSM) developed by Steward (1981) and later advanced in many directions (Danilovic and Browning, 2004; Maurer, 2007; Lindemann, 2008) is an effective tool to manage a complex system, enabling the matrix models to capture the different DSM applications (Eppinger and Browning, 2012). DSM facilitates intelligent system decomposition and integration analysis (Browning, 2001).

DSM approaches are widely addressed in systems community. Rizzuti and Luigi De Napoli provided a perspective to integrate the DSM and Axiomatic Design (Suh, 1990) in product design (Rizzuti and Luigi De Napoli, 2014). Danilovic and Browning conducted a comparison of the DSM approach and the cross-domain DMM approach showing their complementary nature and mutual advantages (Danilovic and Browning, 2007; Danilovic and Browning, 2004). DSM has also been extended to the Multiple-Domain Matrix (MDM) (Maurer, 2007; Lindemann, 2008).

Eppinger and Browning distinguish two types of relationships, which are important in system modeling - hierarchical (vertical) and lateral (horizontal) ones (Eppinger and Browning, 2012). According to them, vertical relationships stem from the decomposition, while horizontal relationships stem from “interactions between elements, such as flows of material or information, at the same level” (Eppinger and Browning, 2012).

We have found that DSM is widely used to represent the decomposition relationships (those which are defined as “vertical” by Eppinger and Browning), as this leads to the sub-systems and their interactions. Specialization relationships (“horizontal” relationships) are used not so often, however, Menshenin and Crawley demonstrated that this type of relationships conveys an important information about the innovative concepts (Menshenin and Crawley, 2018).

In our work we applied the DSM-based methods to three fundamental system entities (object, process, and state) and four structural relationships (specialization, exhibition, decomposition, and instantiation), as they are defined in the Object-Process Methodology (OPM) (Dori, 2002). We have chosen OPM, as this modeling language documenting and modeling the core information about a system by means of simple constructs: objects are represented by rectangles; processes are denoted as ovals; and states are represented as rounded corner rectangles. OPM also includes a number of specific relationships between these objects, processes and states. The idea behind OPM is that the combination of these entities and their relationships allow a systems engineer to effectively represent a complex system of any nature, its function and behavior throughout design process.

In our paper, we aim at demonstrating the utility of the DSM-based approaches for managing an extended set of systems structural relationships, namely *specialization*, *exhibition*, *decomposition*, and *instantiation* as proposed by Dori (Dori, 2016; Dori, 2002). The objective of our paper is to develop and present a DSM-based framework that would enable a digital transformation of system entities through the design process by means of four structural relationships.

The specific objective of this work is to demonstrate a DSM-based framework, which contains the core entities and structural relationships and utilizes the DSM and DMM matrices. This will be illustrated in the example of a coffee maker case study to show how the approach could be used for relatively simple system. However, the design principles remain the same for more complex systems.

This paper is organized as follows. In Section 2 we provide the methodology to facilitate a digital transformation of systems entities through the structural relationships. In Section 3 we provide the DSM-based analysis of systems structural relationships. We explain how the DSM approach has been used to analyze the entities and relationships for a coffee maker case study. In Section 4 we discuss the results of the analysis performed in Section 3 and provide the conclusion, as well as the direction of future work.

2 Methodology

In this paper we have used the following methodology to meet the stated objectives of the work.

At the first step we encoded the core information about the case study – a coffee maker – into the OPcloud environment (Dori *et al.*, 2019). This information included the entities (objects, processes, and states related to the coffee maker), and their structural relationships encoded into the Object-Process Diagrams.

At the second step we encoded the information from the first step into the DSM-based matrices, which lead us both to DSM itself (entities representation) and DMM (entities to relationships) matrices.

It is important to mention that at both steps we included the information not only from an abstracted problem represented in a solution-specific environment, but also from a “...more fine-grained models” perspective (Maier *et al.*, 2016). Each of the analyzed examples of the coffee maker has been decomposed in order to reveal the functional allocations to components.

3 DSM-based analysis of systems structural relationships

Figure 1 illustrates a case study that is used throughout the paper – a coffee maker that is further specialized into an Espresso Machine and a French Press (see Figure 1). Looking at this Figure, one can identify a fundamental difference between these concepts in an informal way. The utility of the proposed approach in our work is that using it the systems engineer could demonstrate the conceptual difference in a more formal and systematic way. This would be enabled by the DSM-based approach allowing to identify the core entities and their transformation from abstract to a more detailed design; and the relationships allowing to facilitate such a transformation.



Figure 1. A case study: Espresso Machine (left) and French Press (right)

As the modeling environment, we exploit the capabilities of OPCloud that enables a modeling process for different design stages – not just conceptual, but also detailed design (Dori *et al.*, 2019). Figure 2 demonstrates the process of searching for alternative solutions for a coffee maker example. A highly abstracted function of a coffee maker is defined as “Extracting a Ground coffee”, as one can see from Figure 2. The process “Extracting” is specialized into two different processes: “Pressurizing” and “Steeping”. The ability to represent this step in a modeling way is important, as these are fundamentally different processes. Having this information, we can assign the design alternatives to each of these processes – “Espresso Machine” and “French Press”, correspondingly. Figure 2 contains three structural relationships: *specialization* (white triangle), *exhibition* (black triangle

inside white triangle), and *instantiation* (black circle inside white triangle). The *exhibition* link connects an object or a process with its corresponding attribute. For example, “Espresso Machine” has an attribute “Automatic”, while “French Press” has an attribute “Manual”. The *instantiation* link is used to represent class-instance relationship. For example, a “Coffee Maker” as a class represents the set of identical products, which are distinguished by serial numbers only, thus we see an instance “Coffee Maker Serial Number X” in Figure 2.

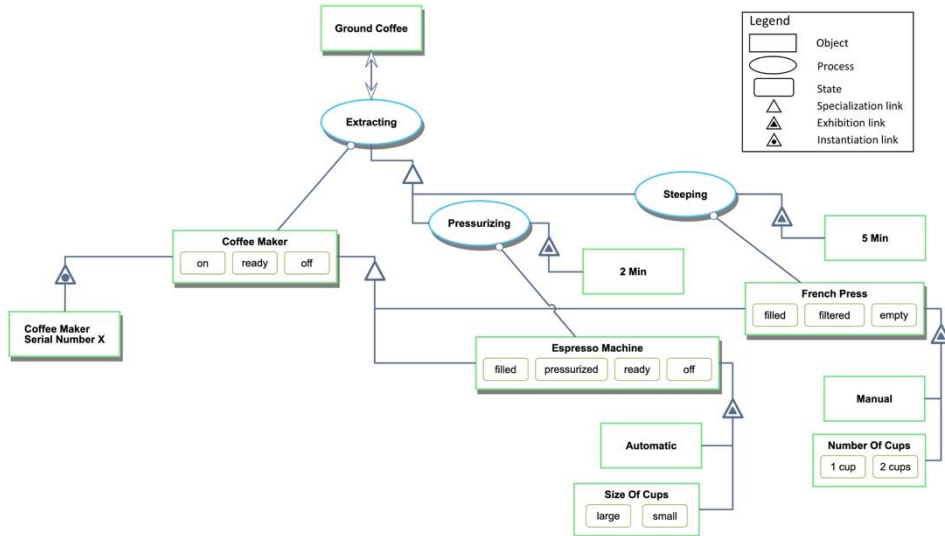


Figure 2. Searching for design alternatives for a Coffee Maker example

Figure 3 and Figure 4 demonstrate the fourth structural relationship - the *decomposition* (black triangle). This relationship appears at a later and more detailed stage of the design process: when a design alternative (“Espresso Machine” and “French Press”) is decomposed into its internal elements. It is also important to note that the *decomposition*, *specialization*, and *instantiation* relationships can only be applied between a single type of entity, namely relating *objects* or relating *processes*. On the other hand, the *exhibition* relationship also called *characterization* (Dori, 2016) is the only structural relationship which can link both objects and processes. It is therefore able to define a relationship between persistent (objects) and dynamic entities (processes) and thus between space and time. It is certainly of particular interest in a DSM matrix which can both include objects and processes. It is also transitive meaning that it can define a hierarchy, thus allowing the definition of more detailed entities and definition of detail design solutions.

As such, Figure 3 demonstrates the *decomposition* for an “Espresso Machine”, while Figure 4 – for a “French Press”. For example, we may notice that the “Espresso Machine” is decomposed into 5 sub-systems: “Portafilter”, “Heating element”, “Pump”, “Boiler”, and “Body”. Each one of these sub-systems performs its own function. For instance, the “Portafilter” is used for transferring a liquid coffee. The representations in Figure 3 and

Figure 4 encode the decomposed elements and associated processes. These representations correspond to functional allocations to components.

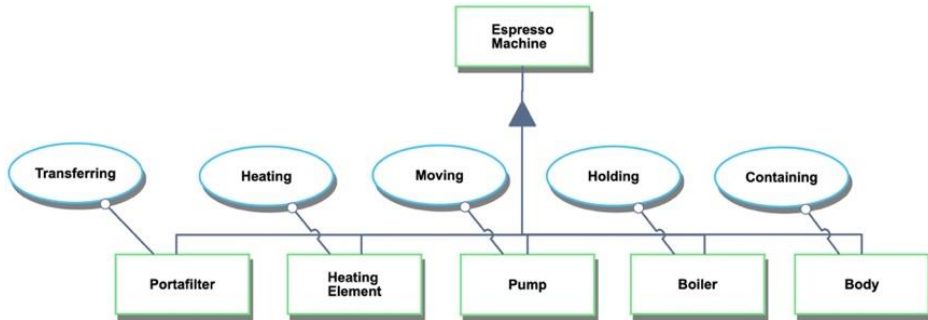


Figure 3. Espresso Machine Decomposition

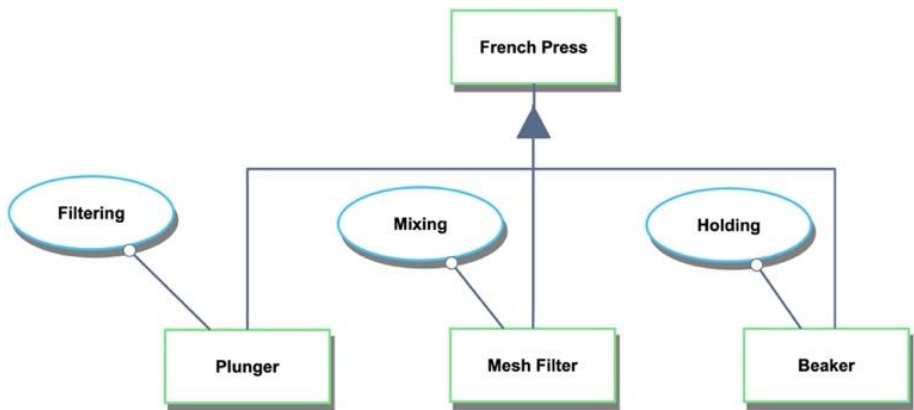


Figure 4. French Press Decomposition

The DSM-based representation of the same information as was demonstrated Figures 2 to 4 is shown in Figure 5. The left-hand side part of this matrix is a DSM representing the solution-specific environment (Menshenin and Crawley, 2020). This environment is responsible for the representation of an abstracted nature of the system. We notice that an object is a “Coffee Maker” that is used for the process “Extracting (Ground Coffee)”. This object can have three states - “on/ready/off” (see Figure 5).

The right-hand side part of the matrix is a DMM matrix. It logically extends the entities presented in the solution-specific environment through the application of structural relationships that transform these entities. This process corresponds to a movement from the abstract solution-specific environment to the detailed design. As such, the object “Coffee Maker” is specialized into the “Espresso Machine” and the “French Press”. This information is encoded into the subsequent cell at the intersection of a row “(1) Object” and a column “(4) Specialization”. It is also important to mention that this is in full correspondence with the modeling information contained in Figure 2. The process

“Extracting” present in the solution-specific environment in the DSM part of the table can also be specialized into the processes “Pressurizing” and “Steeping”. This information is also contained in Figure 2.

		OPM entity			Structural relationships						
		(1) Object	(2) Process	(3) State	(4) Specialization	(5) Exhibition	(6) Decomposition		(7) Instantiation		
OPM entity	(1) Object	Coffee maker			Espresso Machine	French Press	Espresso Machine Automatic Size of cups	French Press Manual Number of cups	Espresso Machine A: Portafilter B: Heating element C: Pump D: Boiler E: Body	French Press A: Plunger B: Mesh filter C: Beaker	Coffee Machine Serial Number X
	(2) Process		Extracting (Ground Coffee)		Pressurizing	Steeping	2 min	5 min	A: Transferring B: Heating C: Moving D: Holding E: Containing	A: Filtering B: Mixing C: Holding	
	(3) State			on/ready/off	filled/pressurized/ready/off	filled/filtered/empty	large/small	1 cup/2 cups			
		Solution-specific environment			Detailed design environment						

Figure 5. DSM/DMM representation of a Coffee Maker example and its relationships

The next relationship appearing in DMM part of the table of Figure 5 is the *exhibition* relationship. This relationship is also represented in a full correspondence with Figure 2: the objects “Espresso Machine” and “French Press” have their corresponding attributes, such as “Automatic” and “Size of cups” for the Espresso Machine; and “Manual” and “Number of cups” for the French Press. These objects and attributes are linked by the *exhibition* relationship. The processes “Pressurizing” and “Steeping” have the attributes of “2 min” and “5 min”, respectively. We also apply the *exhibition* relationship to represent information about the entity “states” which is an important characteristic that can be used to specify the product functions and its behavior. As such, the Espresso Machine has the states “large/small”, while the French Press has the states “1 cup/2 cups” (see Figure 5).

The *decomposition* relationship in a DMM part of the table of Figure 5 is related to the decomposition of an “Espresso Machine” and a “French Press”, correspondingly. We notice the 5 sub-systems for the Espresso Machine, and 3 sub-systems for the French Press. We may also notice the decomposition of the processes “Pressurizing” and “Steeping” into

its own sub-processes. This information for both entities – object and process – can be found in modeling environment in Figure 3 and Figure 4, correspondingly.

Both views – a DMM part of Figure 5 and the modeling representation of Figure 2 – also contain the *instantiation* relationships for object “Coffee Maker”, which is a “Coffee Machine Serial Number X”.

A core difference that can be observed from a Figure 5 comparing to the OPM representation of Figure 2 is that OPCloud does not represent such structural relationships as *specialization* for the entity “state”. In OPM the entity “state” is always part of an entity “object” and cannot be independent. The independence of states from objects appears in the DSM-based approach that is presented in Figure 5. For example, from the DSM we may notice a transformation of states “on/ready/off” as they appear in the solution-specific environment into the states “filled/pressurized/ready/off” for the Espresso Machine, or “filled/filtered/empty” for the French Press. We therefore propose that there is a value as represented in the DSM matrix to consider product/system “states” as entities of their own. Thus, the proposed DSM framework brings an additional capability for managing specifically systems states and structural relationships, which is critical for digital transformation and thus provide a powerful and synthesized mean of representation of product and systems at their early stage of development.

4 Discussion and Conclusion

In this paper, we demonstrated the utility of the DSM-based to facilitate a digital transformation of systems entities through the structural relationships. We presented a DSM-based framework allowing to transform the system entities (objects, processes, and states) through the design process by means of the structural relationships (*specialization, exhibition, decomposition, and instantiation*).

One of the outcomes of the paper is that we demonstrated that the DSM can be effectively used for all systems structural relationships, and not only *decomposition* and *specialization* as usual. We have demonstrated that DSM provides a flow from the abstract environment (the solution-specific environment) to a detailed design stage and when combined with DMM, provide a powerful mean of representation of transformation of complex systems entities from early stage of development to a more detailed one.

A specially interesting result appeared for the entity “state” in the DSM context, which brings additional capabilities for managing specifically systems states and structural relationships in combination with OPM. In particular, we demonstrated that this entity can also be transformed through the design process - from abstract to a more detailed representation, as it is demonstrated in Figure 5 for the coffee maker example. The entity “state” has an intrinsic role of connecting space (object) and time (process) and can be considered as powerful expressions of systems behavior and functions. One of the directions of future work would be to study the representation of structural and behavioral relationships in complex systems through the DSM approach. Another direction of future work would be to define a quantitative approach to represent various design alternatives meeting stakeholders needs.

Thus, the proposed DSM/DMM approach can support the systems engineer in digital transformation of the systems entities from an abstract design environment (the solution-specific environment) to a detailed design. In this capacity it could also contribute to the further development of OPM and the integration of the proposed approach into OPCLoud.

The limitation of our work is that we did not consider any other relationships that are present either in OPM (such as procedural links) or in other methodologies. This would also be the direction of future work.

References

- Browning, T.R., 2001. Applying the Design Structure Matrix to System Decomposition and Integration Problems: a Review and New Directions. *IEEE Transactions on Engineering Management*, 48, pp. 292-306.
- Crawley, E., Cameron, B., Selva, D., 2015. *System architecture: Strategy and product development for complex systems*. Prentice Hall Press.
- Danilovic, M. and Browning, T.R., 2007. Managing complex product development projects with design structure matrices and domain mapping matrices. *International Journal of Project Management*, 25(3), pp. 300-314.
- Danilovic, M. and Browning, T., 2004. A formal approach for Domain Mapping Matrices (DMM) to complement Design Structure Matrices (DSM). In: *Proceedings of the 6th Design Structure Matrix (DSM) International Workshop*, Cambridge. Cambridge, UK.
- Dori, D., Kohen, H., Jbara, A., Wengrowicz, N., Lavi, R., Soskin, N.L., Bernstein, K. and Shani, U., 2019. OPCLoud: An OPM Integrated Conceptual-Executable Modeling Environment for Industry 4.0. *Systems Engineering in the Fourth Industrial Revolution*, pp. 243-271.
- Dori, D., 2016. *Model-based systems engineering with OPM and SysML*. New York: Springer.
- Dori, D., 2002. *Object-Process Methodology: A Holistic System Paradigm*. Springer.
- Eppinger, S.D. and Browning, T.R., 2012. *Design structure matrix methods and applications*. MIT press, Cambridge.
- INCOSE webpage: Digital Engineering Information Exchange. Accessed 10.02.2020. <https://www.incose.org/incose-member-resources/working-groups/transformational/digital-engineering-information-exchange>
- Lindemann, U., Maurer, M. and Braun, T., 2008. *Structural complexity management: an approach for the field of product design*. Springer Science & Business Media.
- Maier, J.F., Eckert, C.M. and Clarkson, P.J., 2016. Model granularity and related concepts. In *Proceedings of the 14th International DESIGN Conference*, Dubrovnik.
- Maurer, M. S., 2007. *Structural awareness in complex product design*. PhD thesis, Technische Universität München, Munich, Germany.
- Menshenin, Y. and Crawley, E., 2020. A System Concept Representation Framework and its Testing on Patents, Urban Architectural Patterns, and Software Patterns, In *Systems Engineering Journal* <https://doi.org/10.1002/sys.21547>
- Menshenin, Y., Crawley, E., 2018. DSM-Based Methods to Represent Specialization Relationships in a Concept Framework. In *Proceeding of the 20th International DSM Conference*, pp. 151-157.
- Rizzuti, S., and De Napoli, L., 2014. The integration of DSM and Axiomatic Design in product design as part of a MDM process. In *Proceeding of the 16th International DSM conference: Risk and Change management in complex systems*, pp. 35-42.
- Steward, D.V., 1981. The design structure system: A method for managing the design of complex systems. *IEEE transactions on Engineering Management*, (3), pp. 71-74.

Menshenin, Yaroslav; Brovar, Yana; Crawley, Edward; Fortin, Clement

Contact: Y. Menshenin, Skolkovo Institute of Science and Technology, Space Center, Nobelya str. 3, 121205, Moscow, Russian Federation, +7 985 311 5294, y.menshenin@skoltech.ru, <https://faculty.skoltech.ru/people/yaroslavmenshenin>