KEYWORDS Engineering Design Process System Dynamics Design Structure Matrix System Behavior

Structure-based COMPILATION OF SYSTEM DYNAMICS MODELS for assessing ENGINEERING DESIGN PROCESS BEHAVIOR

Daniel Kasperek [1]
daniel.kasperek@pe.mw.tum.de

Sebastian Maisenbacher [1]
sebastian.maisenbacher@pe.mw.tum.de

Maik Maurer [1]
maik.maurer@pe.mw.tum.de

[1] Institute of Product Development, Technische Universität München, Munich, Germany

ABSTRACT

The dynamic behavior of complex systems is a well-known challenge within engineering. The paper presents a Multiple-Domain Matrix base model for the structure-based compilation of System Dynamics models for assessing engineering design process behavior. Classically, dependency modelling approaches are used to analyze the structure of a system such as the Design Structure Matrix (DSM), the Domain Mapping Matrix (DMM) or the Multiple-domain Matrix (Multiple-Domain Matrix) approaches. The major drawback of these approaches is that they depict a static view on the system and are therefore not suitable to model the dynamic behavior of complex systems. This paper suggests combining the dependency and dynamic modelling approaches. Previous results already show that on principle dynamic behavior can be deduced from structural models. Consequently, we use a dependency modeling approach as a basis for the compilation of a System Dynamics model to analyze the dynamic behavior of engineering design processes. The approach offers the possibility of design, flexibility and robustness analysis based on the underlying structure of engineering design processes.

1. Introduction and Background Information

The dynamic behavior of complex systems is a well-known challenge within engineering. Processes like engineering design processes are also complex systems, as a lot of factors, such as persons, resources, and iterations have to be taken into account *(Smith & Morrow, 1999)*. Soft factors like knowledge, motivation and skills of the involved persons also play an important role within the development of products. Additionally, external factors like market cycles, legislative cycles, or environmental factors influence the system. *(Kasperek, Maisenbacher, & Maurer, 2014; Smith & Morrow, 1999) (McGraw-Hill & Parker, 2002)* define dynamic behavior as "a description of how a system or an individual unit functions with respect to time".

Additionally to these external factors, internal influencing factors such as an important engineer gets sick, or an important tool collapses can occur and force the Engineering design process to react in one or another way. These reactions of the system are often undesired as the system may become unbalanced. This could i.e. result in a slowdown of the process. Consequently, it is beneficial to analyze systems for their behavior towards changes of influencing factors and optimize them for desired reactions if these factors occur. The general amount of potential factors is innumerable. The decision which particular factors have to be considered depends on the characteristics of the Engineering design process. Therefore no general proposition of influencing factors can be given.

On the other hand, not only influencing factors form the dynamic behavior of engineering design processes, but also the structure of the Engineering design process itself is essential for the reaction of the system. According to *(Maurer, 2007)* the structure can be understood as "the network formed by dependencies between system elements and thereby represents a basic attribute of each system".

Structure-based models allow of general analyses and deliver rather fundamental results. They are mostly used for early planning and system decomposition. *(Biedermann, Diepold, Lindemann, & Lohmann, 2012)*

According to *(Biedermann et al., 2012)* structure-based models require rather little data compared to dynamic-based models, while they claim to describe the engineering system completely *(Browning, 2001)*.

For engineering design processes the particular process steps, resources, engineers, or milestones can be seen as system elements and the allocation of resources to process steps or the achievement of milestones if process steps are finished represent the relations between these elements and thereby form the structure of the Engineering design process.

Often particular elements of the system structure can be used as actuating variables to react on changing influencing factors of Engineering design process, as the reaction of a system to influencing factors is mostly caused by its underlying structure *(Kasperek & Maurer, 2013)*. Typical actuating variables are: Size of work packages, adaption of work packages, order of tasks, organizational units and resources allocated to work packages.

Challenges for engineering design processes depending on the boundary conditions are illustrated in **Figure 1**:

- **Optimized Design:** If it is not likely that influencing factors

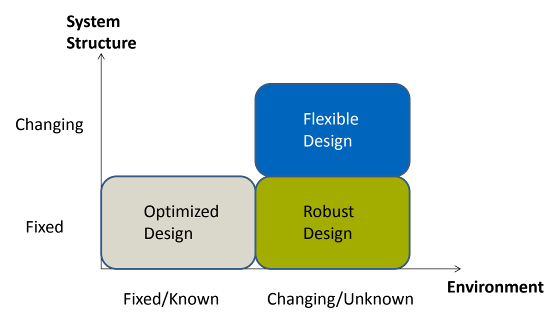

FIGURE 1. *Challenges for engineering design processes depending on boundary conditions (adapted from (Tsegaye, 2013))*

will force the system to react, the system structure might be optimized to increase the Engineering design process's performance. (E.g. to find the best allocation of existing resources to process steps)

- **Flexible Design:** If influencing factors are likely to occur and to affect the Engineering design process, as well as the inherent system structure might be subject to fluctuations (e.g. a new Engineering design process structure is tested), it might be beneficial to analyze the flexibility of the various system designs towards the expected influencing factors.
- **Robust Design:** If the system structure shall not be changed, but influencing factors are likely to occur and to affect the Engineering design process, it might be beneficial to analyze the robustness of the current system design towards the expected influencing factors.

This arises the questions how the structure of a system can be analyzed for its robustness and flexibility for certain influencing factors and how the structure of a system can be optimized for these factors?

The idea of analyzing the underlying structure of complex systems is not new. Classically, dependency modelling approaches are used to analyze the structure of a system such as the Design Structure Matrix *(DSM)*, the Domain Mapping Matrix *(DMM)* or the Multiple-domain Matrix *(Multiple-Domain Matrix)* approaches. The major drawback of these approaches is that they depict a static view on the system and are therefore not suitable to model the dynamic behavior of complex systems *(Diepold et al., 2010)*. Therefore other dynamic modeling approaches such as System Dynamics, agent-based modeling or discrete event simulation exist. Especially System Dynamics is used throughout the public and private sector for policy analysis and design. It is a method to analyze and simulate the dynamic behavior of systems on a high level of abstraction *(Meier & Boßlau, 2013)*. System Dynamics, though, does not offer the possibilities of dependency modeling as static aspects of systems cannot be described *(Meier & Boßlau, 2013)*. The other major drawback of dynamic-based models is the need for a lot of data when creating these models, hence, dynamic-based models usually describe only small parts or single effects of engineering systems *(Biedermann et al., 2012; Diepold et al., 2010)*. As high level management tool, it misses the ability to illustrate the underlying structure of the process. According to *(Biedermann et al., 2012)* structure-based *(e.g. design structure matrices – DSM)* and dynamic-based *(e.g. differential equations or fuzzy systems)* models are the two major modelling approaches for engineering systems.

This paper suggests combining the dependency and dynamic modelling approaches. Previous results already show that on principle dynamic behavior can be deduced from structural models *(Biedermann et al., 2012)*. Consequently, we use a dependency modeling approach as a basis for the compilation of a System Dynamics model to analyze the dynamic behavior of engineering design processes. Our approach serves the following aspects:

- Increase the understanding of the dynamic behavior of engineering design processes;
- Give decision support for the design of engineering design processes through System Dynamics simulation experiments;
- Allow for benchmarking of different structural designs of an engineering design process

Domain	Description
Task	The Task domain collects all entities that describe the execution of work done in the process; further terms are: Process step, action, activity, unit of behavior and work package.
Event	This domain addresses non-persistent occurrences in time that present a certain action, status or progress. Further terms are: Cycle, impact, influence.
Org. Unit	The organizational unit domain contains all human resources in their respective ordering; further terms are: Staff, responsibility, team, pool, lane, actors, roles, and committee.
Resource	The resource domain is intended for all non-human resources necessary to enable the process execution, such as IT-systems, equipment or knowledge. Further terms are: Attribute, mechanism, method, pool, and lane.
Time	The time domain addresses persistent time issues such as the start time of a task or milestones in the process. Further terms are: Attribute, duration, starting time, end time, average time, milestone, and phase.

FIGURE 2. *List and description of domains of the Multiple-Domain Matrix base model based on (Kreimeyer, 2009)*

	Task	Event	Org. Unit	Resource	Time
Task	• precedes temporally	• is influenced by	• is conducted by • is executed by • is required for • supported by	• requires • is supported by	• has duration of • is finished at • starts at
Event			• occurs in	• concerns	• happens at
Org. Unit				• uses	• is active during • is finished at • starts at
Resource					• is available at • is available during
Time					

FIGURE 3. *Multiple-Domain base model*

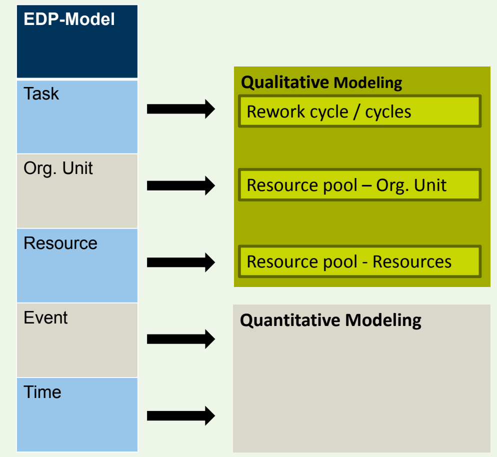

FIGURE 4. *Transformation of the Multiple-Domain Matrix domains into System Dynamics*

2. Multiple-Domain Matrix Base model for the System Dynamics modeling of Engineering design process

This paper proposes a Multiple-Domain Matrix base model for the System dynamics modeling of engineering design processes. This base model can be used to model the underlying structure of the engineering design process and serves as a basis for the compilation of System Dynamics models of engineering design processes. Based on a transformation concept by *(Kasperek & Maurer, 2013)* the authors propose to transform the Multiple-Domain Matrix base model into a qualitative System Dynamics Model. The Multiple-Domain Matrix base model is based on the "Meta-Multiple-Domain Matrix with domains and relationship types suited for most modeling and analysis purposes" suggested by *(Kreimeyer, 2009)* and serves the following purposes:

- To ease the construction of System Dynamics Models of Engineering design process
- To enable the analysis of the dynamic behavior of engineering design processes based on the underlying system structure
- To condense the core structure of System Dynamics Models into Multiple-Domain Matrix and thereby ease conclusions from System Dynamics Models

It offers the possibility of design, flexibility and robustness analysis of engineering design processes based on their underlying structure.

The meta-model developed by *(Kreimeyer, 2009)* is intended as a modeling scheme which is capable to describe relevant aspects of structural modeling and a goal-oriented process analysis. The meta-model provides orientation when modeling a structural process model. The meta-model systematizes and collects relevant domains and relationship types and puts these into a common framework. A structure consists of a particular pattern of nodes and edges in a graph, but a structure only has meaning if it is related to a certain semantic context. This context is provided by the meta-model of *(Kreimeyer, 2009)* that describes the types of nodes and edges concerning their meaning in an industrial application. The meta-model consists of two views on a structure: The domains, describing what types of entities are common to process modeling, and the relationship types, describing how the domains are commonly related. *(Kreimeyer, 2009)*

Based on a literature review, *(Kreimeyer, 2009)* defines domains that are most common and represent the usual domains found in process modeling. Based on this set, 5 domains are chosen for the Multiple-Domain Matrix base model. **Figure 2** lists the domains and a description of each domain, based on the definitions of *(Kreimeyer, 2009)*.

In the context of this work the Meta-Multiple-Domain Matrix is taken as a basis for the Multiple-Domain Matrix base model: While five of the six domains are adopted, the various possible relationship types are reduced for each possible relationship between the domains. With focus on the deduction of a System Dynamics Model of the dynamic behavior of the Engineering design process not all possible relationships are considered. Thus, only relationships that are important for the time dependent process flow are considered. As the base model can be adapted to different modeling situations, not only one, but more possible relationship types are given for some correlations. Relationship types can also be changed to other than the indicated ones if necessary. The preferred relationship types by the authors are indicated in bold letters. The Multiple-Domain Matrix base model is illustrated in **Figure 3**.

Taken into account the transformation concept between Multiple-Domain Matrix and System Dynamics by *(Kasperek & Maurer, 2013)*, the Multiple-Domain Matrix base model can be transformed into a System Dynamics Model. While the elements of the task, organizational unit und resource domains and their relationships can be directly transformed into elements of a qualitative System Dynamics model, the elements of the event and time domains and their relationships can be used for the quantitative System Dynamics modeling. Thereby the elements and relationships cannot be directly transferred into the System Dynamics model, but serve as a basis for the construction of the necessary differential equations for quantitative System Dynamics modeling. **Figure 4** shows the transformation of the Multiple-Domain Matrix domains into System Dynamics models. Relationships within the Multiple-Domain Matrix base model can be transformed into inter-phase relationships between rework cycles or into intra-phase relationships between rework cycles and resources, depending on the transformation of the involved domains.

3. Application of the Multiple-Domain Matrix Basis model

The Multiple-Domain Matrix base model can be used to build a system dynamics model of an engineering design process to examine its dynamic behavior. Compared to the other construction approaches, the structure-based compilation of the model offers the following benefits:

- Guideline of structural elements to be considered
- Guideline of relations to be considered
- More repeatable construction of the model
- Underlying modeling assumptions clearly represented in Multiple-Domain Matrix

To illustrate the applicability of the Multiple-Domain Matrix base model, a generic academic example is chosen: It represents a shortened version of a common engineering design process. The authors are aware of the fact that the evaluation does not prove the applicability of the procedure on an industrial level, but want to show the general idea behind it and therefore choose a shortened example.

		Task				Event	Org. Unit			Resource				Time		
		A	B	C	D	alpha	a	b	c	1	2	3	4	short	med.	long
Task	A		x				x									x
	B			x			x								x	
	C	x			x	x		x						x		
	D								x						x	
Event	alpha															
Org. Unit	a									x	x	x	x			
	b											x	x			
	c										x					
Resource	1															
	2															
	3															
	4															
Time	short															
	med.															
	long															

FIGURE 5. *Multiple-Domain Matrix model of the example, based on base model*

The Multiple-Domain Matrix base model as illustrated in **Figure 5** represents the structural model of the examined engineering design process. The represented dependencies are: Tasks precedes task temporally; Task is influenced by event; Task is conducted by org. unit; Org. unit uses resource; Task has duration.

Based on the Multiple-Domain Matrix model a qualitative System Dynamics model was generated. The qualitative System Dynamics model is illustrated in **Figure 6**. The domains of the matrix were transformed according to the transformation rules indicated in **Figure 4**. The dependencies were transformed as follows: The given explanations partly go beyond the level of qualitative modeling and already give insight to the quantitative implementation of the model.

- Tasks precedes task temporally: The workflow of task n+1 is dependent on the completion of the previous tasks n (N done)
- Task is influenced by event: event alpha influence task C, by triggering its rate Corrupt C. event alpha is implemented as a step function that increases the amount of remaining work c at one discrete time step (time of occurrence of the event)
- Task is conducted by org. unit: The organizational units are allocated to the workflow rates of the particular tasks. The rates are dependent on the potential work force of the organizational units.
- Org. unit uses resource: The resources are allocated to the organizational units. The more resources are allocated the higher the potential assumed work force of the organization units.
- Task has duration: This dependency cannot be seen in the qualitative model. The information is used as parameter for quantifying the original work to do for each task which is defined by definition n.

The variable finishing does not belong to the actual model of the dynamics of the engineering design process, but is introduced to sum up the current levels of completion for all tasks and thereby to measure the overall completion of the process.

The qualitative model was then quantified by the necessary information. As the example shall show the general applicability of the approach, the values were approximated by an expert focus group where necessary. **Figure 7** shows the results of the simulation of the overall System Dynamics model.

The overall process completion rate is indicated by the green color. It can be seen that the process is completed after 58 months and that the rate is subject to oscillation. This oscillation can be clearly attributed to the event alpha and the feedback loop from task C to task A. The occurrence of the event alpha is indicated by the red line. It can be seen the process completion rate decreases as soon as the event alpha occurs *(month 47)*. This can be explained by the additional work generated by the event which decreases the rate between already completed work packages and work still to do. Additionally it can be seen that the rework *(blue line)* of task A due to the feedback of task C also influences the process completion.

4. Discussion

Even though the presented application is simplified, the application of our approach of a compilation of System Dynamics models based on the underlying structure of engineering design processes shows the potential of dynamic structure modeling techniques for behavioral analysis. System Dynamics is a powerful tool which enables to further develop engineering design processes by assessing for example the influence of rework and events on the overall process duration. As this publication is a result of ongoing research activities, there is more research to conduct to reach a level of industrial applicability: Currently the base model is a substructure. Domains and relation types can be neglected or others can be added for the particular engineering design process, such as the domain degree of iteration to indicate the amount of iterations for each particular process step. The domains and relation types should be adapted in a way to fit best the necessary information to build the System Dynamics model. Depending on the particular behavioral pattern to be modeled, a variety of additional domains and relation types are possible. For the case that the behavior of the system towards changes of the environment shall be modeled, the event domain might need to be more detailed.

It might be necessary to take more domains and relation types into the base model and rework the current ones.

As for any model, the quality of the System Dynamics model is dependent on the underlying assumptions. The assumptions used within this model are mainly incorporated in the Multiple-domain matrix. If this information can be further improved the real system can be closer approximated.

5. Conclusion and outlook

The paper presents a Multiple-Domain Matrix base model for the structure-based compilation of System Dynamics models for assessing engineering design process behavior. The Multiple-Domain Matrix base model thereby serves the following purposes: To ease the construction of System Dynamics Models of Engineering design process, to enable the analysis of the dynamic behavior of engineering design processes based on the underlying system structure and to condense the core structure of System Dynamics Models into Multiple-Domain Matrix and thereby ease conclusions from System Dynamics models. The approach offers the possibility of design, flexibility and robustness analysis based on the underlying structure of engineering design processes. This base model can be further customized for the particular analysis purposes for each engineering design process. An exemplary case study shows the applicability of the base model and illustrates potential further use cases of the suggested approach. The next steps will be to model and analyze different engineering design processes from real life examples. Additionally, the transformation from structural model to System Dynamics model will be further refined. With an established System Dynamics model at hand, it is possible to analyze several scenarios of engineering design processes by simulation, which will help to further improve the planning and management of engineering design processes by providing a tool to handle the dynamic system behavior.

Acknowledgment

We thank the German Research Foundation *(Deutsche Forschungsgemeinschaft – DFG)* for funding this project as part of the collaborative research centre 'Sonderforschungsbereich 768 – Managing cycles in innovation processes – Integrated development of product-service-systems based on technical products'.

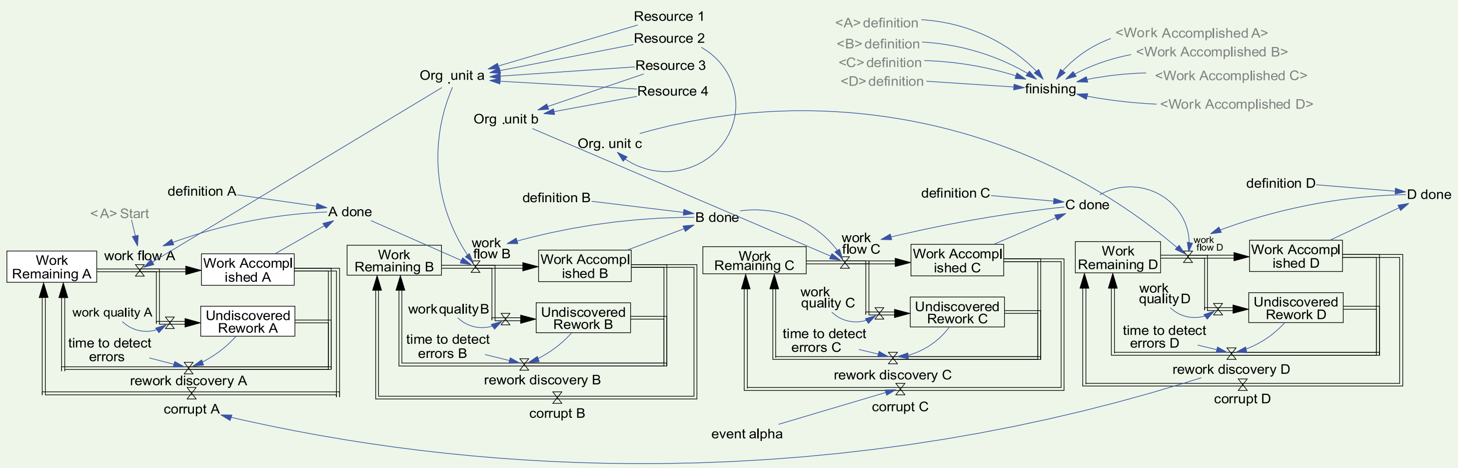

FIGURE 6. *Qualitative System Dynamics model of the engineering design process*

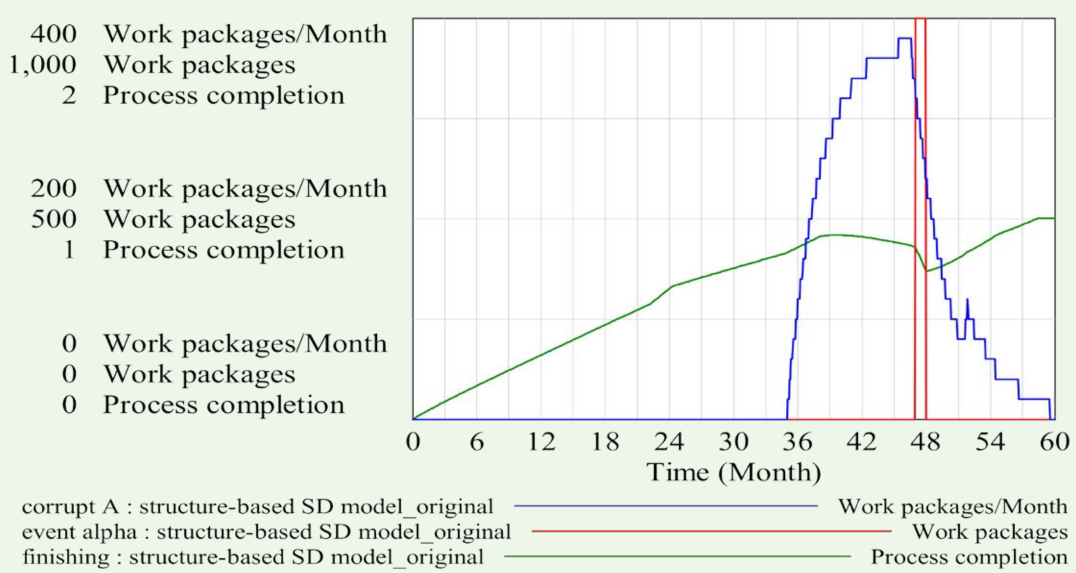

FIGURE 7. *Results of the simulation of the engineering design process*

authors

Daniel Kasperek holds a diploma in mechanical engineering (Dipl.-Ing. 2012) from Technische Universität München, Germany and is currently a research assistant at the Institute of Product Development at Technische Universität München. His research interests include all aspects of complexity management, but particulary the modelling of complex systems.

Sebastian Maisenbacher graduated in mechanical engineering at the Technische Universität München (TUM). He is now working as a research assistant at the Institute of Product Development at the TUM. His research includes structural modeling in the field of product development and value engineering.

Maik Maurer holds a PhD for his research in complexity management. He taught and leads research projects in complexity and knowledge management as well as Systems Engineering. He was awarded the ring of honor from the Association of German Engineers (VDI) and prizes for extraordinary lecturing. Maik Maurer co-chairs the German chapter of INCOSE and co-authored the book "Structural Complexity Management". He successfully founded companies in Germany and the US.

references

Biedermann, W., Diepold, K. J. K., Lindemann, U., & Lohmann, B. *(2012).* Delimiting Structural and Dynamical System Analysis in Engineering Management. In D. Marjanovic, M. Storga, N. Pavkovic, & N. Bojcetic *(Eds.)*, Proceedings of DESIGN 2012, the 12th International Design Conference *(pp. 1649–1658).* Dubrovnik, Croatia: Design Society.

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*(3)*, 292–306. doi:10.1109/17.946528

Diepold, K. J., Biedermann, W., Eben, K. G. M., Kortler, S., Lohmann, B., & Lindemann, U. *(2010).* Combining Structural Complexity Management and Hybrid Dynamical System Modelling. In Proceedings of DESIGN 2010, the 11th International Design Conference *(pp. 1045–1054).* Dubrovnik, Croatia.

Kasperek, D., Maisenbacher, S., & Maurer, M. *(2014).* Structure-based Analysis of Dynamic Engineering Process Behavior. In 2014 IEEE International Systems Conference *(SysCon 2014).* Ottawa, Canada.

Kasperek, D., & Maurer, M. *(2013).* Coupling Structural Complexity Management and System Dynamics to represent the dynamic behavior of product development processes. In 2013 IEEE International Systems Conference *(SysCon) (pp. 414–419).* Orlando, USA: IEEE. doi:10.1109/SysCon.2013.6549915

Kreimeyer, M. *(2009).* A Structural Measurement System for Engineering Design Processes. Dr. Hut, München.

Maurer, M. *(2007).* Structural Awareness in Complex Product Design. Munich, Germany: Dr. Hut.

McGraw-Hill, & Parker, S. P. *(2002).* McGraw-Hill Dictionary of Scientific and Technical Terms *(p. 2380).* McGraw-Hill Professional.

Meier, H., & Boßlau, M. *(2013).* Design and Engineering of Dynamic Business Models for Industrial Product-Service Systems. In Y. Shimomura & K. Kimita *(Eds.)*, The Philosopher's Stone for Sustainability *(pp. 179–184).* Berlin: Springer. doi:10.1007/978-3-642-32847-3_30

Smith, R. P., & Morrow, J. A. *(1999).* Product development process modeling. Design Studies, 20*(3)*, 237–261. doi:10.1016/S0142-694X*(98)*00018-0

Tsegaye, S. *(2013).* Flexible Urban Water Distribution Systems. University of South Florida.