INTERNATIONAL DESIGN CONFERENCE - DESIGN 2016
Dubrovnik - Croatia, May 16 - 19, 2016.

THEORY OF SOCIAL SYSTEMS ENGINEERING

T. Naumann

Keywords: design theory, sociotechnical systems, definitions, complexity

1. Introduction

This article refers to the paper "Meta-Model of Sociotechnical Systems" [Naumann and Koehler 2014]. It describes derivation, detailed structure, and content of a Meta-model of Socio-technical Systems (*MSTS*).
The design of the *MSTS* was initially carried out at the same time to the development of new methods and tools for the early phase of the automotive industry [Königs 2012], [Kallenborn 2013]. In doing so *MSTS'* derivation is heuristically based and aligned on system theoretical approaches from human sciences and social sciences [Luhmann 1985], [Willke 1991], [Luhmann 1996] as well as on *Ropohl's* System Theory of Technology [Ropohl 2009].
Initially developed as a phenomenological meta-model, the *MSTS* was constantly validated and refined as example for new Systems Engineering approaches [Königs 2012]. This paper now presents the extension of the meta-model to a "Theory of Social Systems Engineering" (*TSSE*).
It essentially serves as a first draft of such a theory and follows the fundamentals of every theory of science [Lossack 2006].
After describing elements of *TSSE* and a redefinition of terms of engineering design, application fields show conclusions from industrial practice. In this paper it is not possible to describe the application fields in detail. Especially here it should be referred to the related publications.
Based on communication analyses of collaborative engineering, findings were considered using the new approach [Naumann et al. 2011], e. g. the possibility of the identification of critical solution. In another usage of the new approach product development projects' ability to handle complexity as well as effectiveness and efficiency of their processes were analyzed [Kallenborn 2013].
Based on the analyzed data, the paper presents conclusions for principles of self-controlling and self-organization of sociotechnical systems. These conclusions act as concrete guidance and opens a field for further research in this area.

2. Research for tomorrow's product development

How do you envision product development's future? Will tomorrow's products be developed in the same way as today? Will you incrementally adapt existing tools and methods in the same linear manner? Technological development of human history presents many examples of product class development being interlaced with characteristics of artifacts. These characteristics are interlaced with artifact's usage [Popitz 1995]. It can therefore be expected, that increasing complexity of tomorrow's products need to be in accordance with an adequate complex product development.
Already today, complexity and intelligence of innovative products are almost exclusively defined over cross-disciplinary functions, software, as well as diversification, and interaction with system environment. *„DISTRONIC plus" is* a nice example for such aspects in cars and commercial vehicles.

This technical function comprises and couples cruise control, radar system, acceleration system, and brake system across complex control software. It is a highly-complex interaction of electrical, electronic and mechanical subsystems. In reality for drivers it is a new form of interaction with the system environment and a preliminary stage of autonomous driving. In order to achieve a given destination, the drivers have to conciliate so far complexity of system environment and technical system. They observe traffic, accelerate, steer, and have to brake. With "DISTRONIC plus" drivers still need to steer. With ongoing autonomous driving's potential, steering becomes superfluous. However, in order to fulfil the performance of autonomous driving, the car must become more complex itself. Accordingly, product development needs to be adapted. It should be technically, procedurally, organizationally, and communicational able to handle this complexity. But today's product development organizations reach limits.

The multidisciplinary functions of modular product architectures exceed their abilities to effectively and efficiently handle technical and procedural complexity. There are partially capacitive limits. But there are procedural limits too. They concern the ability to elaborate new know-how, to use, to document and to record effectively and timely. There are many limitations of today's methods and tools too. It is not possible to represent all the interconnection of technical subsystems. They are made for individual domains only, not for the overlapping functions of systems. Additionally, limitations need to be overcome which deal with the understanding of technical and procedural complexity - at least in industrial practice [Steward 1981], [Suh 1990], [Kannapan et al. 1993], [VDI-Richtlinie 2221 1993], [Eppinger et al. 1994], [Morelli et al. 1995], [Browning 1997], [Danilovic and Börjesson 2001], [Maurer et al. 2003], [Freeman 2004], [Batallas and Yassine 2006], [Maurer 2007], [Collins et al. 2009], [Lindemann et al. 2009], [Gokpinar et al. 2010]. It is one of the objective of this research to develop a universal systems approach which brings together domain knowledge and which enables interfaces between all socio-technical subsystems. Another objective is to develop systemically focused methods and tools.

3. Understanding of social systems engineering

Social system engineering is a specialization of system engineering. Many approaches deal with model based system engineering in order to integrate this approach in product development processes from specification to validation. Now a second concept is set aside for this approach. Simplified, it can be said model based system engineering deals with the effectiveness of the development of technical systems - doing the right things. While social systems engineering concerns an efficient development of technical systems through social systems - doing things right [Naumann and Koehler 2014].

A phenomenological meta-model of sociotechnical systems is on top of social system engineering. The model approach shows systematically systemic relations between social systems, which develop and use products, and technical systems on an abstract level. Moreover, the definitions of engineering objects and their relations - concerning individual applications like interaction and communication, systems specification, systems development, and process management - are defined in a new way. Finally, a basic system understanding of sociotechnical systems is established by the new model approach and the new understanding of definitions.

The new model approach furthermore provides a new basis for determination of complexity and the ability to handle complexity. With the assistance of new methods and tools - based on interaction and complexity analyses - existing product and process data can be interpreted better. Data can then be used in order to trace designing, decisions and changes during whole product development processes (*traceability*) [Storga et al. 2011]. For tomorrow's product development, outlined theories of social systems engineering allow for a new type of self-regulation and self-organization.

Most existing management methods are based on process management and project management. Their approaches consider activities and events as well as tasks and mile stones. Product complexity and procedural complexity are only abstractly included and are not directly measurable. Similarly, these methods are applied by project managers and process managers and not by engineers themselves. Usually engineers manage their individual approaches by fulfilling requirements through synthesizing and validating product characteristics on a very concrete level. Unfortunately, the maturity of the overall product, or the own part is often subjective, since an entire system is never in focus.

The *TSSE* enables raising objective measuring criteria and key figures of product and processes of an entire system. It's possible to bear them in relation and to make them available as control parameters for all members involved. Social system's self-regulation will be more objective. Product's and processes referentiality will be more concrete. Self-organization involves engineers into the steering and is more comprehensive than today.

4. Elements of TSSE

The theory of social systems engineering is a theory for the domain of product development. For its acceptance it has to fulfill some conditions, e.g. a theory's definitional demands. It also has to answer questions about its necessity, its relation to existing theories, and its own character. The comprehensive character of the Theory of Social Systems Engineering emerge by integration, pervasiveness, superstructure and comprehension.

Compared with existing approaches it integrates subdomains of technically and social systems based on General System Theory (= integration). All engineering processes of new products (*NP*) - from conception to start of production are supported regarding their pervasiveness. Existing methodologies can be integrated and their scope can be pointed out (= superstructure). Finally, the Theory of Social Systems Engineering should be simple to understand and practically applicable for users (= comprehension).

Theory implementors are domain-specific engineers as well as industrial engineers of new processes, methods and tools. Engineers think pragmatically in partial models (requirements, functions, active principles, forms) of their concrete (sub)product and their representations (specifications, virtual behavior and CAD models, physical prototypes). Using the theory as a paradigm, they can abstract gradually on the product development as a whole meta-system and can concretely understand the impact of sociotechnical relations on process dynamic and the effects of changes.

Industrial engineers and designers of new processes, methods and tools, scientifically searching for more efficient and effective approaches, acquire new entrances to product and process complexity of development. It extends from an abstract level to a concrete level - from the social function of technology as a progress' driver up to engineering objects interaction' on the level of parameters. By recognizing and analyzing self-similar characteristics of each socio-technical (sub)systems of product development, abstract questions can be answered concerning emergence, nature, principles and future of product development. Furthermore, it becomes concretely possible to specify a data model for new IT systems.

A theory is defined as a model of objectivity which allows descriptive and causal statements about reality. Based on this model understanding, it must therefore be possible to predict and derive recommendations for action. An essential part of a theory are explicit definitions and accordance about their understanding. Further definitional prerequisites are closure, consistency, and especially empirical confirmation. This is made possible through an operationalization from observed phenomena of reality [Thiel 1996], [Zima 2004].

According to this demand, the theory of social systems engineering contains and considers the following aspects:

1. Consideration of engineering's phenomena in different dimensions of complexity
2. Fundamentals of general system theory
3. Integration of existing theories and model approaches
4. Meta-model of socio-technical systems
5. Development of new SSE methods and tools
6. Understanding of Social System Engineering as part of System Engineering
7. Product and process specific methods and procedural models in relation to the *TSSE*
8. Super ordinate system of objectives.

As mentioned above, the *TSSE* model approach is mainly represented by the Meta-model of Socio-technical Systems, Figure 1 [Naumann and Koehler 2014].

The model concept of product and process, as well as their context-specific representations, have direct influence on how information and knowledge in engineering are structured, processed, and documented [Ostermayer 2001]. For the model approach therefore it was important to clearly represent a holistic structure and relations of each socio-technical system. Another reason is the normative meaning of simplifying model representations for communication. Comparable schematic models, e. g. the representation of the approach of VDI2221 [VDI-Richtlinie 2221 1993] or the "V-model" [Alpar et al. 2008], were important for many existing processes and applications.

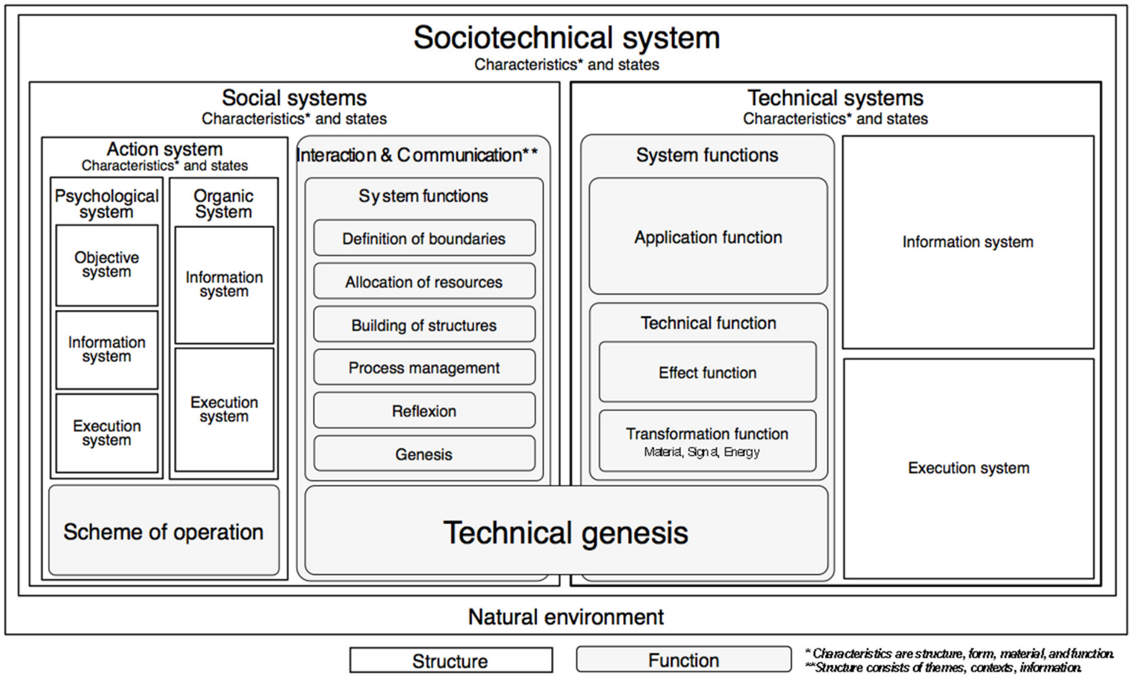

Figure 1. Sociotechnical system

The model approach expands on the fundamentals of General System Theory (2) and bears in relation to other existing theories. It integrates other existing model approaches and tries to surmount limitations of those methods and tools (3), in order to be validated permanently at observable phenomena (1). On this basis it should be possible to create new solutions (5) to reach the objectives of the super ordinate system of objectives (8). These should be concretely able to:

- make technical, social and procedural complexities understandable
- model and generate complex systems
- document all system-events
- measure complexities in real time
- provide recommendations for action
- finally support interaction and communication.

Initial empirical confirmation of this theory already took place through several methods of analysing complexity based on communication and maturities. Additionally, a new expert tool for system modelling was specified, implemented and validated [Königs 2012]. Hence new directions are shown for further methods and tool approaches and their development should be expressly encouraged here. The empirical basis of the *TSSE* is broadened by the operationalization of various production developments' complexity. Finally, the generic system performance of socio-technical systems can be analysed and systematised on fundamental principles.

5. Glossary and understanding of definitions

Using the *MSTS*' substantial definitions (*Begriffe*) in context of system understanding of sociotechnical systems will be (re)defined. An understanding of definitions (*Begriffsverständnis*) according to *VDI2330* conduces as a basis. A definition describes a perceptible and conceivable, however non-material object (*Gegenstand*). It represents a specific characteristic of the regarded object.

Definition of characteristics: Characteristics correspond exactly to the specific properties of an object, which conduce for conceptualization (Begriffsbildung) and distinction of definition (Begriffsabgrenzung) [DIN 2330 1996].

Each object can be described by properties (*general*) and characteristics (*specific*). But only characteristics transform items to specific objects. In context of the *MSTS*, objects are elements and functions of natural, social, and technical systems.

Each technical system questions whether a required or a defined characteristic is specific for this object, e. g. the number of four cylinders of an engine. With this validation, an engine becomes specifically the combustion engine *("four-cylinder")*. The principle of terminology implies that technical systems can be specified, developed, produced, and used only by the definition of characteristics.

In addition, technical systems have general properties, e. g. weight. Weight is a general property of each object which is suspended by force of gravity. General properties require no specification, development, or production. They develop virtually per se. However, each general property - like weight - can become a specific characteristic of a certain object, e. g. a lightweight component with particularly light weight.

Characteristics are distinguished in quality features (*Beschaffenheitsmerkmale*) and relation parameters (*Relationsmerkmale*). Quality features indicate characteristics of objects like: Shape, Dimensions, Material, Colour, Position, Time, and Structure.

In contrast, relation parameters describe the relation between two definitions. These include e. g.:

- Indicators of origin (inventor, manufacturer, place of origin)
- Comparative indicators (larger, smaller)
- Valuation criteria (cheap, sufficiently)
- Functional characteristics (= functions).

Quality features such as shape, material, structure, as well as the relation parameter „function" are therefore basic characteristics of technical systems. All system functions exist in relation to the quality features of a system. To draw a conclusion, quality features constitute relation parameters: functions.

All characteristics have due to structure and function among each other, diverse and variably often pronounced relations. Quality features of technical systems can reference to actions of social systems (= usability). They can be assigned to the application functions. The quality features of technical systems can be assigned to technical functions („features") if there is a system internal reference, a reference to systems environment, or not yet a reference to the actions of social systems.

The objective of technical genesis is the coincidence of action functions of social systems with application functions of technical systems, thus the quality features and the resulting relation parameters. Then, a technical system is functional for the user and respectively works.

Definition of function: A function represents a relation parameter of an object or a system, which exists by relationship with other objects and which comes into existence by quality features and / or other relation parameters of the object or the system.

Generally, a function is understood as material, energetic, or informational transformation of input parameters to output parameters, and respectively as reaction to events (e. g. signal of a speed sensor).

Even the fulfillment of requirements is designated ‚functionality' or just ‚usability': The function of an object is the task, which it has to fulfill. The function represents in addition to form, material, structure, etc. a substantial characteristic of each and every object, that in somehow form is used [Popitz 1995].

Due to the view of an object, a function is a characteristic of the object. A functional characteristic comes into existence from this perspective by quality features and relation parameters. In particular, a function is a relation characteristic of the object or system, which exists by relationship of other objects to the object or system.

A function cannot be thought without the consideration of such a relationship with another definition, since it exists as specific characteristic of a definition only in the relation to another object. These circumstances are valid for object and system, whereby relation determines the kind of a function: ‚The function of an object or a system is determined by their "behavior under certain conditions", whereby a further distinction between construction, use or service function is possible.'

Technical functions are therefore developed in relations from technical systems to other technical systems. Application functions of technical systems are accomplished in affiliation with a social system. This takes place via the application of an object / a technical system via a user / a social system (e. g. a

tool). In this use, a function is in the same manner a system function of a technical system as well as an action function of a social system.
An action function is fulfilled optimally by characteristics of an existing technical system - see equivalence principle of the technical genesis. If this is not the case, new characteristics are needed from the perspective of the action functions. These new characteristics must be synthesized only in the technical system by decomposition / composition and transformation. They are specified in that reason in form of requirements.
Definition of requirements: A requirement specifies an object (technical system) and corresponds to an analyzed, systematized, usually categorized condition between structures and / or functions of sociotechnical systems.
Requirements represent anticipated conditions, which are valid between the relation parameters and quality feature of social (and natural) systems and the relation parameters and quality feature of technical systems. Requirements are always relations between two objects. In this sense, they show a directed relationship with the product and give an answer regarding the 'why' or 'thus' of a condition. Example: Requirement C02 < 90g/m3 means the consumption of the car must be under 5,9l (gasoline), so that the current legislation is fulfilled.
As previously mentioned, the operationalization in socio-technical systems is made about the product with its characteristics. All characteristics which define a technical system have to be considered and proceeded during processes. Or the other way round: by the definition of all characteristics, the technical system is developed during processes.
Definition of 'processes' and 'events': Processes are real events of changing system characteristics - generated by the operationalization of system functions of social and technical systems.
In the context of model understanding of socio-technical systems, processes are formed through temporary defined and observable events. Each specification, definition or adjustment of a parameter, an element or a structure by a function represent an event. These events are produced as system functions of social systems, technical systems, and technical genesis.
Each system function stands, according to their understanding of definitions, in relation to other objects. Changes or reactions in form of further events are produced with all linked objects. These events, respectively the sequences of these events, are accessible over a new understanding of interaction and communication.
Understanding / definition of 'interaction' and 'communication': By interaction and communication, the interdependency between the system functions of all sociotechnical (sub) systems are to be understood.
Each socio-technical system, social subsystem and technical subsystem are structurally separated from one each other. The action system "human" exists as a unit and the technical system "car" as another. The occurrence as sociotechnical system comes into existence in the context of technical genesis via a structural coupling by characteristics of both subsystems. This affiliation results from the continuous processing from information to the alignment of the required / still needed characteristics of social system with existing / still to synthesized characteristics of technical system. It exists as long as both systems have interdependencies. That is as long as they communicate with each other.
The development of products implies for social systems to stand permanently in interdependency with technical systems, from specification of requirements until physical prototypes. All models of requirements, functions, active principles and geometry models represent thereby partial models of the product model, respectively they represented the product model.
Representations of product model can be considered again as feature spaces with systemic dimensions. Systemic dimensions of feature spaces of different product representations are on the one hand different by kind and number of their parameters which are set in direct system relationship - on the other hand by numbers of their transformation determined relations. Relations are thereby an essential part of the product model.

6. Application of the entire system model

The application of the *MSTS* depends on individual application context. Typical use cases when developing new methods and tools are [Königs 2012], [Kallenborn 2013]:

- Analysis of real sociotechnical systems

- Analysis of social and technical subsystems
- Analysis of processes and procedures
- Synthesis of procedural complexity for the control of socio-technical systems
- Synthesis of procedural and technical complexity for development of tools for system development.

Generally, it is recommended to proceed gradually, following described structures. A lot of information, which is represented abstractly in the meta-model, exist concretely in companies and projects. These are for example:

- Organigrams of social systems
- Decision minutes with tasks and decisions (communication)
- Product data models of technical systems (requirements, function descriptions, behavioural models, design data, product structures)
- Status lists and reports (maturities).

Figure 2 shows analyzed information of a product development project.

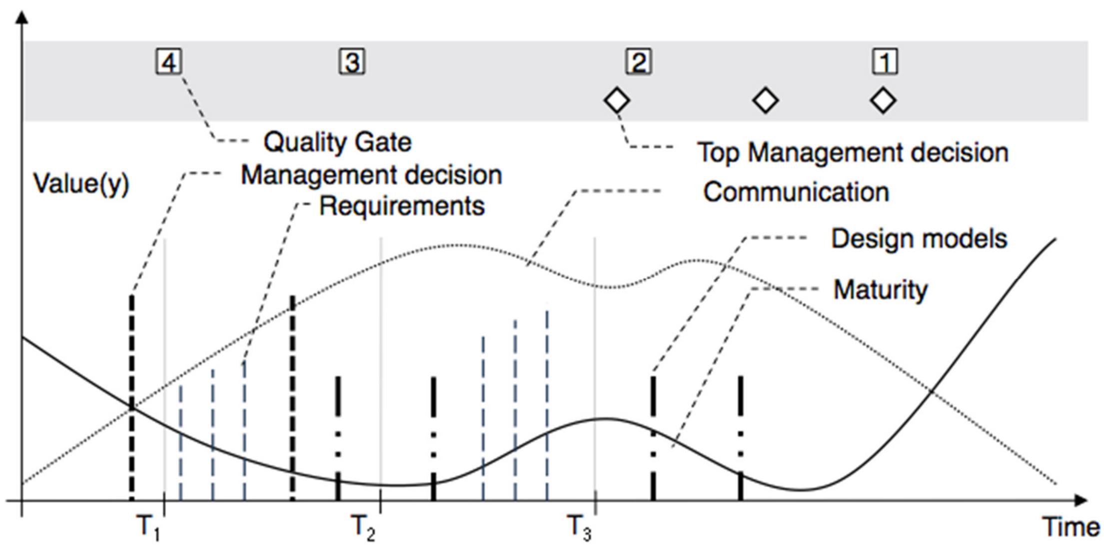

Figure 2. Existent product and process information

Access to system functions affords especially interaction and communication data, for example decision minutes, please see [Trier 2007], [Uflacker et al. 2009], [Dorta et al. 2011]. However, these have yet to be decoded according to the meta-model. Only then is it possible to determine which function has triggered an event in the meta-model and which structural elements are affected. Figure 3 shows schematically the decoding of a decision minutes' entry on elements of *MSTS*. The decoding is facilitated if a categorization of themes and sub-themes already exist. The always present timestamp makes it possible to analyze events and event sequences.

Tasks are assigned to people, which show formal and informal social networks.

The main goal of the *TSSE* remains the understanding of the stability relation between social and technical systems (the equivalence principle). This ensures reaching a sustainable development and reliable application for simple or complex as well as individual or distributed products. The fundamental relation is described by function of technical genesis and it arises in procedural complexity as well as in technical complexity of product development. Moreover, the procedural complexity is physically measurable with help of interaction and communication analyses. Technical complexity is defined by numbers of engineering objects (*EO*) and their engineering object relations (*EOR*).

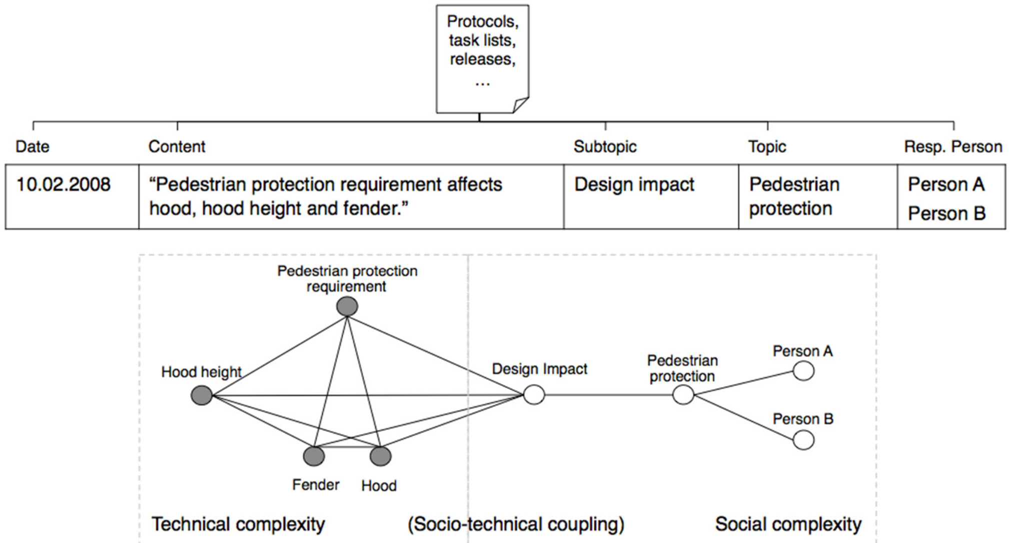

Figure 3. Decoding of information from decision minutes (pedestrian protection)

Objective complexity can be analysed by reference to system elements and their dependencies [Sosa et al. 2007], [Törlind et al. 2009]. Complexity increases with the number of elements and relations. In addition, complexity grows by transforming structures and functions. These changes over time are to be understood as *dynamic*. Based on these principles, the operational complexity of socio-technical systems is ascertained from the sum of the interacting system elements, Figure 4.

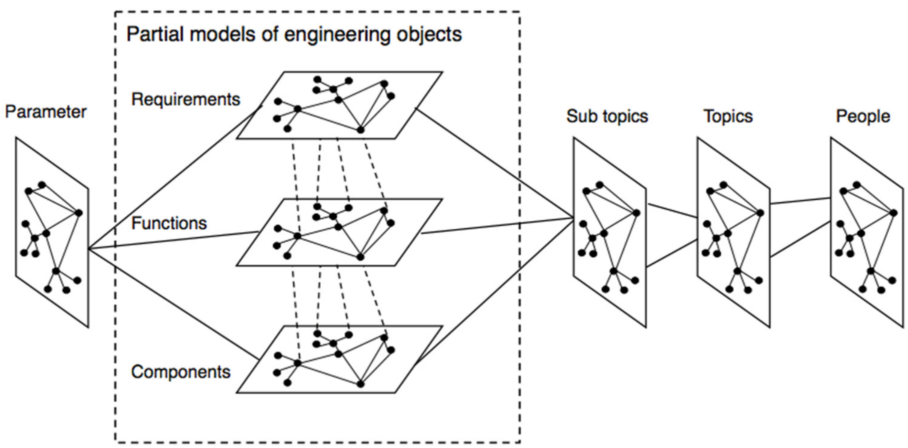

Figure 4. Principle of complexity analyses

Each system can only exist through effective and efficient processing of complexity. Technical complexity is processed parallel and iteratively in the product development's product development process. As previously mentioned, new methods are therefore needed. The *TSSE* enables determining and comparing the complexity of real product development projects methodically. Related to this, the answer exists to the question of how much complexity can be processed by a social system at all.

Additionally, the question can be answered *how* a social system handles complexity. By analyzing a “solution path” of real project data, it is possible to demonstrate effectiveness and efficiency [Rehmann-Sutter 1996], [O'Donnell and Duffy 2005]. The solution path thereby consists of all steps of synthesis of products. In product development processes, same steps of synthesis of products will always be applied parallel and iteratively. The sequences of these steps can be analyzed and evaluated as highly effective solution paths.

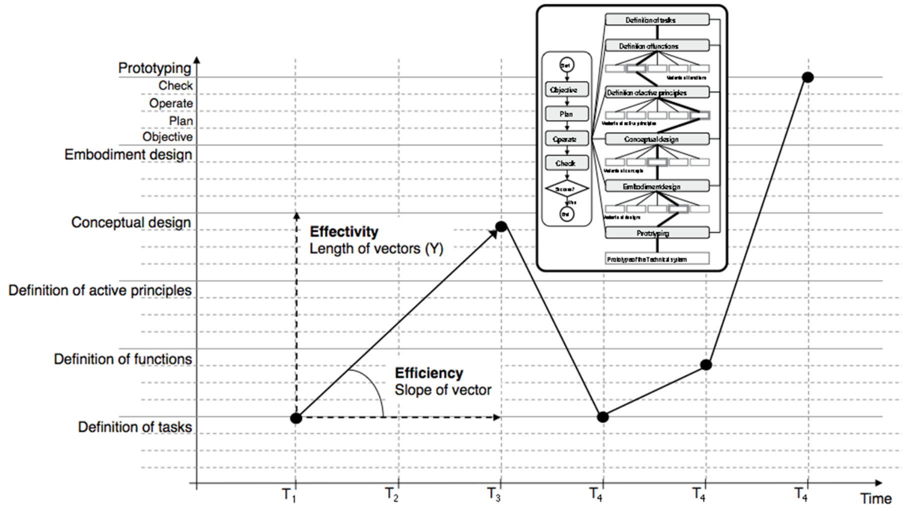

Figure 5. Principle of solution paths

In solution paths, effectiveness of procedures is determined by length of vector from one step to next step (= change of categories). Efficiency is determined by increase of the vector (= change over time). An analysis of solution paths illustrates how many iterations were applied until it succeeds. It is also shown how many recesses have been made, for example, due to a new input.

The analysis of complexity results in indicators for dealing with complexity:

- Ability to integrally process complexity
- Effectiveness and efficiency in processing complexity
- Ability to operate iteratively
- Reactions to new input.

Another objective of communication analysis is to identify potential problems and critical situations to allow for better comprehension. The ability to handle complexity is limited. When the system comes up against this limit, a critical situation occurs. The meaning of the underlying systemic understanding of communication is, that all events of action functions are marked by communication. As a result, critical situations of action functions follow critical situations in communication. Figure 6 shows a procedure for identifying critical situations by evaluating communication and complexity.

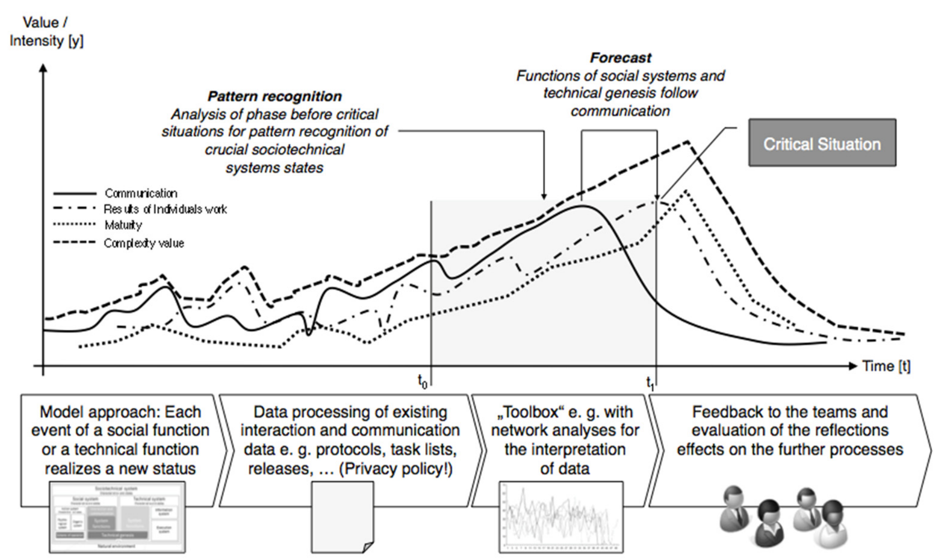

Figure 6. Analysis of critical situation

Resulting from the previous understanding of complexity, a critical situation in communication consists of many communication events and their relations. Practical experiences confirm this hypothesis. However other experiences show critical situations also arise due to lack of communication. With the model approach and communication understanding it is to be noted that it is also possible to demonstrate the potential of necessary communication. If this is not thoroughly performed, a critical situation can occur theoretically.
The development of new methods has been described up to now. But new methods also require new tools. Therefore, the possibility of developing new system tools based on meta-model and understanding of definitions of socio-technical systems will be described.

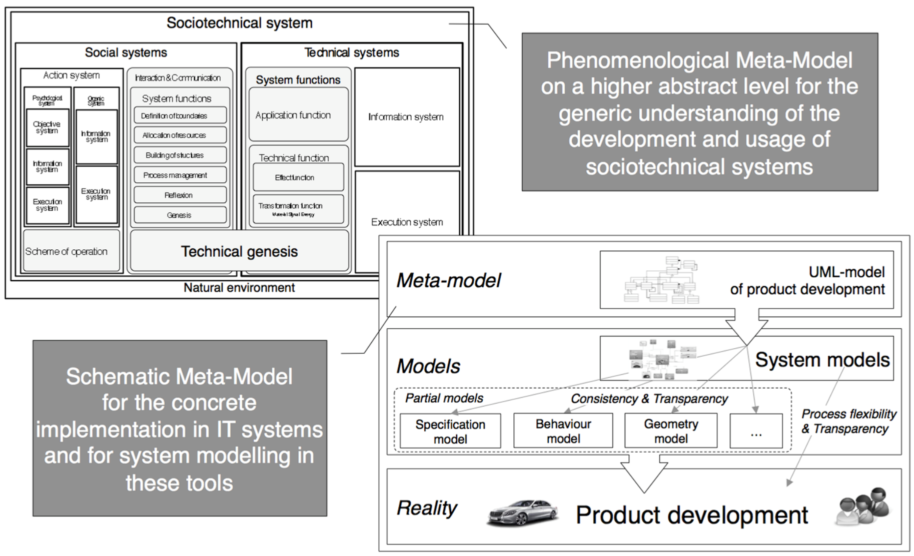

Figure 7. Model understanding for new methods and tools

Figure 7 shows the comprehensive model for understanding the development of new methods and tools. The *MSTS* as an abstract meta-model allows the analysis and systematization of phenomena of the real product development. Schematic meta-models can be developed for specific applications with help of this understanding, e. g. as *UML* notations. One application is the modelling of system models in context of systems engineering. A system model can consist of different representations of product model and process model. These must be consistent with each other and have to be related to each other.
New systems engineering tools support - based on the model of described model understanding - the developers in the systematic structuring, specification, synthesis and validation of new products. They lead to transparent processes and flexible changes; as systemic impacts can be handled better. In total, they lead to more effectiveness and efficiency.

7. Outlook: research questions and first conclusions considering product development's future

Which approaches help us develop methods and tools for tomorrow`s production development? An initial statement was made: Rising complexity of products can be realised only with an adequate production development. The genesis of technology and the systemic aspect clarify that technology and society can only be thought of together. Social needs and technical possibilities are always mutually dependent. The view of production development's future can take place from this common perspective only.
Trends show in which ways technology changes society and vice versa. They show among other things, a continuous networking of personal and shared data and things. Technologies are the "internet of things", for example smart things or cloud data. Analog and digital reality merge. But what can be identified with it?

Technology does not change the human being, but relieves the increasing complexity of an interlaced world. Technology supports striving for health and extends its existence. Technology is a driving force for progress. Progress democratises technology by its striving for efficiency. Each small improvement of function or quality of an object serves as a reference frame for other function's improvement or increasing quality of another object. Progress is a self-perpetuating driving force.
For methods and tool developers it is useful to analyse and understand trends and progress as social and technical driving forces. The theory of Social Systems Engineering should therefore help with schematic understanding of it's model approach and its operationalization. Conclusions from existing projects can be addressed to production development and their future:

- Complexity of interconnected products has already to be represented in organisation and processes of production development.
- The ability of social systems to process complexity should be enhanced by the ability to communicate all technical and social characteristics and relations.
- System engineering methods and tools should explicitly support networking of this characteristics and relations.
- Engineers should be enabled to specify complex requirements, define solution spaces, evaluate partial solutions from computers, adapt solution spaces, and document entire knowledge.
- Routine activities should be fulfilled by IT tools, for example management of information about changes, confirmation of changes, or adjustment of minor modifications.
- Effectiveness and efficiency of an entire system production development are determined on basis of interaction and communication's quality.
- Critical situations, which result from social and technical complexity, should become identified, classified and served for self-regulation of all participants.

Only if production development's future will become systemic, production development will become a system of the future.

References

Alpar, P., Grob, H. L., Weimann, P., Winter, R., "Anwendungsorientierte Wirtschaftsinformatik - Strategische Planung, Entwicklung und Nutzung von Informations- und Kommunikationssystemen", Vieweg + Teubner, Wiesbaden, 2008.

Batallas, D. A., Yassine, A. A., "Information Leaders in Product Development Organizational Networks: Social Network Analysis of the Design Structure Matrix", Engineering Management, IEEE Transactions on Engineering Management, Vol.53, No.4, 2006, pp. 570-582.

Berawi, M. A., Woodhead, R., "A Teleological Explanation of the Major Logic Path in Classic FAST", 44th Annual Conference of the Society of American Value Engineers International (SAVE International), Montreal, 2004.

Browning, T. R., "Exploring integrative mechanisms with a view toward design for integration, advances in concurrent engineering – CE97", Fourth ISPE International Conference on Concurrent Engineering: Research and Applications, August 20–22, 1997.

Collins, S. T., Yassine, A. A., Borgatti, S. P., "Evaluating product development systems using network analysis", Systems Engineering, Vol.12, No.1, 2009, pp. 55-68.

Danilovic, M., Börjesson, H., "Managing the multi-project environment", Proceedings of the 3rd international design structure matrix workshop, Cambridge, MA, USA, October 29–30, 2001.

DIN 2330, "Begriffe und Benennungen", Engeler, U. (Ed.), Basel, Weil am Rhein, Wien, 1996.

Dorta, T., Kalay, Y., Lesage, A., Pérez, E., "Elements of design conversation in the interconnected HIS", International Journal of Design Sciences and Technology, Vol.18, No.2, 2011, pp. 65-80.

Eppinger, S. D., Whitney, D. E., Smith, R. P., Gebala, D. A., "A model-based method for organizing tasks in product development", Research in Engineering Design, Vol.6, No.1, 1994, pp. 1-13.

Freeman, L. C., "The Development of Social Network Analysis: A Study in the Sociology of Science", Book-Surge, North Charleston, SC, USA, 2004.

Gokpinar, B., Hopp, W. J., Iravani, S. M., "The Impact of Misalignment of Organizational Structure and Product Architecture on Quality in Complex Product Development", Management Science, Vol.56, No.3, 2010, pp. 468-484.

Kallenborn, O., "New Product Development as Complex Adaptive Socio-technical Systems", EBS Business School, Wiesbaden, 2013.

Kannapan, S. M., Bell, D., Taylor, D., "Structuring information and coordinating teams in product development", Design theory and methodology, Vol.53, 1993, pp. 233–42.

Königs, S., "Conceptual Design and Implementation of a Method for Template-Based Systems Engineering", Technische Universität Berlin, Berlin, 2012.

Lindemann, U., Maurer, M., Braun, T., "Structural Complexity Management", Springer, Berlin, 2009.

Lossack, R.-S., "Wissenschaftstheoretische Grundlagen für die rechnergestützte Konstruktion", Springer, 2006.

Luhmann, N., "Social systems", Stanford University Press, 1996.

Luhmann, N., "Soziale Systeme", Suhrkamp, Frankfurt am Main, 1985.

Maurer, M., "Structural Awareness in Complex Product Design", Dissertation, Technische Universität München, Munich, 2007.

Maurer, M., Pulm, U., Lindeman, U., "Tendencies towards more and more flexibility", Transactions on Engineering Management, Proceedings of the 5th dependence structure matrix (DSM) international workshop, Cambridge, United Kingdom, October 22–23, 2003.

Morelli, M. D., Eppinger, S. D., Gulati, R. K., "Predicting technical communication in product development organizations", IEEE Transactions on Engineering Management, Vol.42, 1995, pp. 215–222.

Naumann, T., Koehler, N., "Meta-Model of Sociotechnical Systems: Derivation, Structure and Content", Proceedings of the tenth international symposium in tools and methods of competitive engineering (TMCE2014), Budapest, Hungary, 2014.

Naumann, T., Tuttass, I., Kallenborn, O., Königs, F., "Social Systems Engineering", Proceedings of the 18th International Conference on Engineering Design (ICED 11), Impacting Society through Engineering Design, Vol.4: Product and Systems Design, Lyngby/Copenhagen, 2011, pp. 357-368.

O'Donnell, F. J., Duffy, A. H. B., "Design performance", (illustrated ed.), Glasgow, Springer, 2005.

Ostermayer, R., "Pragmatisch-situative Wissensrepräsentation – ein Baustein für das Wissensmanage-ment", Shaker Verlag, 2001.

Popitz, H., "Der Aufbruch zur Artifiziellen Gesellschaft", J. C. B. Mohr, Tübingen, 1995.

Rehmann-Sutter, C., "Leben beschreiben: über Handlungszusammenhänge in der Biologie", Verlag Königshausen & Neumann, Würzburg, 1996.

Ropohl, G., "Allgemeine Technologie", Universitätsverlag Karlsruhe, 2009.

Shannon, C. E., "A mathematical theory of communication", Bell Systems Technical Journal, Vol.27, 1948, pp. 379–423.

Sosa, M. E., Eppinger, S. D., Rowles, C. M., "A Network Approach to Define Modularity of Components", Complex Products, In: Journal of Mechanical Design, Vol.129, No.1, 2007, pp. 1118-1129.

Steward, D. V., "The design structure system: a method for managing the design of complex systems", IEEE Transactions on Engineering Management, Vol.EM-28, No.3, 1981, pp. 71-74.

Storga, M., Marjanovic, D., Savsek, T., "Reference Model for Traceability Records Implementation in Engineering Design Environment", Proceedings of the International Conference on Engineering Design, Vol.6, Copenhagen, 2011, pp. 173-182.

Suh, N. P., "The Principles of Design", Oxford University Press, New York, NY, USA, 1990.

Thiel, C., "Theorie", In: Mittelstraß, J. (Ed.), Enzyklopädie Philosophie und Wissenschaftstheorie, Band 4, Stuttgart/Weimar, 1996, pp. 260-270.

Törlind, P., Sonalkar, N., Bergström, M., Blanco, E., Hicks, B., McAlpine, H., "Lessons learned and future challenges for design observatory research", ICED'09, Stanford, CA, USA, August 26, 2009.

Trier, M., "Virtual Knowledge Communities", VDM Verlag, Saarbrücken, 2007.

Uflacker, M., Skogstad, P., Zeier, A., Leifer, L., "Analysis of virtual design collaboration with team communication networks", ICED'09, Stanford, CA, USA, August 26, 2009.

VDI-Richtlinie 2221, "Methoden zum Entwickeln und Konstruieren technischer Systeme und Produkte", VDI-Gesellschaft Entwicklung Konstruktion Vertrieb, Düsseldorf, 1993.

Willke, H., "Systemtheorie I: Grundlagen", Gustav Fischer Verlag, Stuttgart, New York, 1991.

Zima, P., "Was ist Theorie? Theoriebegriff und Dialogische Theorie in den Kultur- und Sozialwis-senschaften", UTB 2589. Tübingen, 2004.

Thomas Naumann, Dr.-Ing.
Automotive, IT for Engineering
Weickmannstr. 14, 89077 Ulm, Germany
Email: thomas.t.naumann@daimler.com