No need to learn from each other? Potentials of Knowledge Modeling in Autonomous Vehicle Systems Engineering Towards new methods in multidisciplinary contexts

Ferdinand Schäfer, Reiner Kriesten, Daniela Chrenko, Alexandre Ravey, Franck Gechter

HAL Id: hal-02392710
https://hal.science/hal-02392710

No need to learn from each other? - Potentials of Knowledge Modeling in Autonomous Vehicle Systems Engineering

Towards new methods in multidisciplinary contexts

Ferdinand Schäfer, Reiner Kriesten

Institut für Energieeffiziente Mobilität

Hochschule Karlsruhe – Technik und Wirtschaft

Karlsruhe, Germany

Daniela Chrenko, Alexandre Ravey

Femto-ST,

Université Bourgogne Franche-Comté

Belfort, France

Franck Gechter

LE2I,

Université Bourgogne Franche-Comté

Belfort, France

Abstract—**Engineering autonomous driving functions has become a dramatic challenge in automotive engineering since it is now required to integrate knowledge from multi-disciplinary domains. In this context, the widespread engineering methods are showing their limit since they mainly integrate technological centered point of view. Thus, these new requirements lead naturally to the design of new method for engineering in automotive field. The goal of this paper is to sketch an overview of the possible improvements that Knowledge Modeling and ontologies can bring to Systems Engineering and especially in the case of Autonomous Driving functions.**

Keywords— Automated Driving (AD); Systems Engineering (SE); Requirements Engineering (RE); Model-Based Systems Engineering (MBSE); multidisciplinary engineering; interdisciplinary engineering; Knowledge Management (KMa); Knowledge Modeling (KMo); Ontology; Semantic Web (SW); Human Systems Integration (HSI)

I. INTRODUCTION

Engineers e.g. in the automotive software context are well-versed learning specialized programs and thus implicitly to deal with the meaning of meta-models behind the programs. Dealing with meaning and sharing knowledge in turn generally are perceived bothersome. However, the need to work with stakeholders from very different disciplinary contexts and to learn about their domain is crucial in engineering projects. Nevertheless, dealing with knowledge from different domains in engineering is often reduced to adapting the "wording" while avoiding explicitly approaching meaning.

In turn, some communities are practicing explicit approaches of meaning. For instance, the Artificial Intelligence (AI) community uses specialized models handling meaning called "ontologies" to develop intelligent systems and make them interoperable and the Semantic Web community aims at a global transfer and usage of usage via internet. Especially the approaches of Semantic Web bundles mature standards to explicitly handle meaning in terms of Knowledge Models.

Formal methods of Knowledge Models are commonly considered too oversized for application in engineering. This position paper thus addresses the question of principle, if Knowledge Models can positively contribute to Systems Engineering (SE). It presents potentials of usage of such Knowledge Models to support engineering in multidisciplinary contexts.

SE is a discipline, which gained importance in the aerospace sector and was standardized within EIA 632, IEEE 1220 and ISO/IEC/IEEE 15288:2015, Systems and software engineering — System life cycle processes, [1]. It intends to cope with the different types of complexities faced in engineering projects, from the engineered system up to the processes required during its life cycle. Fulfilling the requirements of the diverse parties involved thus is the major challenge to take on. According to INCOSE (International Council for Systems Engineering), [2], SE is an interdisciplinary approach[1] and aims the realization of successful (engineered) systems, [3]. Nowadays, Model Based System Engineering (MBSE) is commonly used for the development of Complex Systems. SysML (Systems Modeling Language) is the main established MBSE approach in industrial projects.

Automated Driving (AD) plays a leading role in automotive innovations as it opens new ways of auto-mobility. It more concretely represents a high economic value for new functions in passenger cars and commercial vehicles. From a historical

---

[1] Whereas Systems Engineering is an interdisciplinary approach, projects may in a first time be considered multidisciplinary, as no special interactions between disciplines may be guaranteed per se.

perspective, automation in cars was classically related to control tasks, e.g. to guaranty stability of maneuvers, like done with Anti-Lock Braking System. Modern automation furthermore relates to the command of the vehicle and thus concerns active decisions on accelerating and turning. This means replacing the human driver in the driving task. Developing such automation functions necessitates profound competencies of several specific engineering fields, such as environment perception and human machine interfaces. AD brings new dimensions of interdisciplinary engineering in the automotive field. SE can be seen as a promising approach for these development challenges.

AD as multidisciplinary innovation field is inherently linked to Knowledge Management (KMa). Engineering of AD systems is more knowledge intensive than other automotive projects since it includes a multitude of domains. Communicating and transferring knowledge between all involved stakeholders thus becomes decisive for the success of projects. In such multidisciplinary context, the elicitation and specification of requirements becomes a central issue. Knowledge and language used in the different domains concerning AD thereby can be considered as a starting point. Knowledge Modeling (KMo) can, in this context, be a method to bridge the gap from different domain knowledges to a common system description in such multidisciplinary contexts.

The paper presents a position that has matured in industrial innovation projects with the aim to setup an innovative Systems Engineering methodology in the context of Automated Driving. Main outlines of Automated Driving will therefore be given as a stereotype of a multidisciplinary engineering field. Moreover, a model-based methodology which has been evaluated in the automotive context will be referred. This is the base of the present paper for taking into consideration Knowledge Modelling within Systems Engineering. Knowledge Models and formal methods, which they belong to, are frequently considered too oversized for application in engineering. This position paper thus addresses the question of principle, if Knowledge Models can positively contribute to Systems Engineering. It proposes potentials of usage of such Knowledge Models to support engineering in multidisciplinary contexts.

The paper is structured as follows: section II describes the issues related to automated driving focusing on the variety of domains involved in this concept. Section III exposes the system engineering practices for developing automotive functions. Section IV introduces Knowledge Management and Knowledge Modeling as possible innovative methods for designing and conceiving autonomous vehicle systems. Finally, section V concludes the paper by giving key perspectives for future works.

II. THE AUTOMATED DRIVING ISSUES

A. An overview of the industrial point of view for current and future automated driving functions

AD concerns society in a degree never reached before by automation, as it integrates automated systems in areas with humans, through replacement of the human driver in the primary driving task. In the following, a socio-technical analysis of the topic is presented. AD does not evolve alone in these innovative times. Technological innovations facilitate new

forms of mobility and new functions for automotive industry. Electronics and more particularly software play a key role in this as they realize central parts of logics in systems. Even if the different domains of innovation are often perceived in an intermingled manner at the market side, they each play an own role in industry, where special know-how is required and specialized sectors work in corresponding value chains. Different categories of innovations can be identified and Fig. 1 gives a summary of main domains of automotive innovations according to the development of embedded systems. The Fig. summarizes innovation categories of VDA (Verband der Automobilindustrie), [4], and identifies the following domains with high importance for innovation in automotive Embedded Systems: AD, connected vehicles and propulsion and energy concepts.

The mentioned domains group technological evolutions and market relevant functions. On one hand, they each require specific knowledge and on the other hand can also be combined. An important example for combining innovation domains can be found in cooperative functions allowing for AD together with functions of connected vehicles through an increased exchange of information between traffic participants and their environment, as e.g. described in [5].

The desire of automation naturally is also present in mobility. In a cross-domain consideration, the core aim of automation is to replace humans in accomplishment of tasks. This can improve comfort and help in difficult situations. Besides, automation can allow for improving reliability, safety and precision. [6].

The evolution of automation for the driving task in passenger cars starts with the Cruise Control [7]. In 1958 Chrysler produced the first cruise control which allows for fulfillment of a reference speed [8]. Next important historical milestones are ABS (Anti-Lock Braking System) in 1978, Traction Control System (TCS) in 1987 and Electronic Stability Control (ESC) in 1995 [9].

The mentioned functions are not detailed here, but their evolution shows that driving comfort was a main motivation for the introduction of automation in the driving task, starting with Cruise Control. Beyond this, improvement of safety for a better control of the vehicle was the driving force for innovations like ABS, TCS or ESC. These automation functions are mostly related to the control of the vehicle: They help correctly realizing given commands of the human driver such as steering or accelerating.
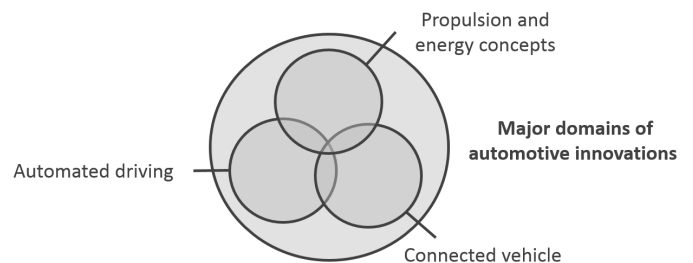

Fig. 1: Important domains of innovation in automotive Embedded Systems, according to [4]

Nowadays, expected advantages related to the development of higher levels of AD, go even further then safety and comfort. Beyond the simple control, functions currently concern the command of the vehicle. ERTRAC (European Road Transport Research Advisory Council) names additional advantages to safety and comfort, such as energy-efficiency, social inclusion of disabled persons in mobility and accessibility of zones with high density of population can be seen as main drivers for AD [10]. Ensuring automation of the driving task for complete use cases holds out the prospect of full relief for the human driver.

B. The different levels of automation

Nowadays numerous assistance functions are found in passenger cars and the ongoing development of more advanced functions of AD further augment the diversity. Superficial similarities often complicate a clear view on the specific characteristics of the different functions. Classifications allow for better overview in this confusion.

The field of AD can be structured in a meaningful way into levels of automation. Currently, 4 main references concerning levels of AD can be found: They are defined by the following organizations:

- BASt (Bundesanstalt für Straßenwesen) in Germany, [19]
- NHTSA (National Highway Traffic Safety Administration) in USA, [12]
- SAE (Society of Automotive Engineers) in USA, [13]
- VDA in Germany, [14]

These classifications partially differ in number of levels of automation, criterions chosen for differentiation and terminology used to describe them. In turn, they all distinguish levels by how control and monitoring is shared between driver and car. The Society of Automotive Engineers with the report SAE J3016, "Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems", [13], gives an orientation with a descriptive character for levels of driving automation. It is chosen as reference in this work due its fine granularity with six levels. Table 1 depicts the definition of SAE levels of driving automation according to SAE J3016.

The question approached here for the classification of AD at the same time represents a simple fundament for orientation and demonstrates the complexity of concerned functions especially for sharing the driving task between driver and system. For instance, if we consider the primary tasks as defined in [15], these can be decomposed as follows in relation with the SAE levels of automation (Fig. 2).

Table 1: Definition of SAE levels of driving automation according to SAE J3016, according to [13]

1: "driver assistance	"the driving mode-specific execution by a driver assistance system of either steering or acceleration/deceleration using information about the driving environment and with the expectation that the human driver perform all remaining aspects of the dynamic driving task"
2: "partial automation"	"the driving mode-specific execution by one or more driver assistance systems of both steering and acceleration/deceleration using information about the driving environment and with the expectation that the human driver perform all remaining aspects of the dynamic driving task"
3: "conditional automation"	"the driving mode-specific performance by an automated driving system of all aspects of the dynamic driving task with the expectation that the human driver will respond appropriately to a request to intervene"
4: "high automation	"the driving mode-specific performance by an automated driving system of all aspects of the dynamic driving task, even if a human driver does not respond appropriately to a request to intervene"
5: "Full automation"	"the full-time performance by an automated driving system of all aspects of the dynamic driving task under all roadway and environmental conditions that can be managed by a human driver"

It is then hard to explain what is made by the driver and what must be controlled by the system. Moreover, the transition from full automated system to full driver controlled system is not that simple and requires a specific training for the driver who must change quickly his(her) role from system's manager to regular driver. The challenge is then how to put the human driver at the center of the engineering process. Even if some methods are based on a human centric point of view [16], [17], the relationship between the system and the driver is generally only tied to the definition of an adapted Human Machine Interface (HMI).

The question of the Human-Machine link is central when talking about safety issues. This lead to the following questions:

- In what context does and can the driver activate an AD function?
- What is the functional content of the activated AD function to cover a mode for a certain period of time?
- Are the capabilities of the system sufficient to cover the mode as expected by the driver?
- Have the limitations of the system correctly been identified so that the human driver is requested in case the capabilities are overstrained?
- When the driver is requested, how is he informed about it and how long can he take to react?
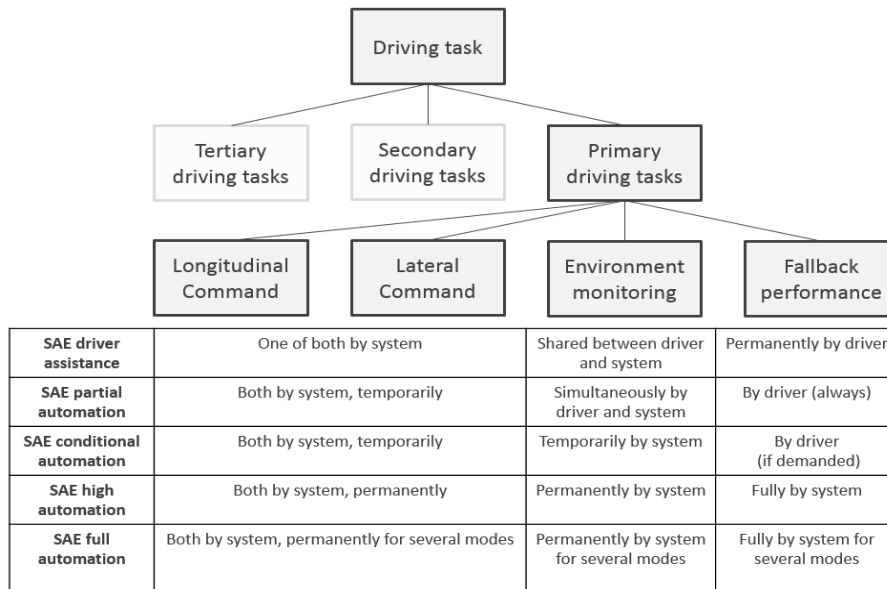
Fig. 2: Division of the driving task in AD and separation of the SAE J3016 levels "driver assistance", "partial automation", "conditional automation", "high automation" and "full automation", according to [13] and [15]

These Human related concerns are the center question of the development of AD and can be added to the other domains involved which involve various knowledge domain. The next section will expose a list of the domains generally encountered in AD engineering.

C. Domain related to AD

In the previous section, we exposed the socio-technical aspects of AD mainly related to the required interaction with the Human driver at several levels. Besides this human elated concerns, AD engineering is also tied to many other domains, related to specific knowledge, that must be considered. Among these specific knowledge, we can cite among many:

- Infrastructure concerns: the context of usage of AD.
- System behavior: the way system is performing primary and secondary tasks such as participating to the traffic flow or respecting traffic rules.
- Interaction with the human driver: this domain is related to technical points of view but also to ergonomic, psychology…
- Environment perception: the way the vehicle is perceiving its situation which implies the use of data fusion and detection and pattern recognition algorithms.
- V2X communication: in this domain concerns both technological/physical and security aspects.
- Control/command: the way the AD algorithm are controlling the hardware part of the vehicle.
- Standardization and Regulations: the law and rules context in which the autonomous vehicle is registered…

It is now clear that autonomous driving functions engineering is requiring a multi-disciplinary point of view and thus a system engineering methodology that takes this particularity into account.

III. AUTOMOTIVE SYSTEMS ENGINEERING

A. The three dimensionnal morphology of SE.

Systems Engineering (SE) covers the complete life cycle of systems with a specific relevance of the concept stage. A separation of problem space and solution space represents a useful basis for SE. Path-breaking works such as the "Three Dimensional Morphology of Systems Engineering" [18] underline its importance: Its second dimension is the "problem solving procedure". The first dimension in turn is the "time dimension" and is referred to as system life cycle in current standards like ISO/IEC/IEEE 15288.

Stages ("period within the life cycle of an entity that relates to the state of its description or realization", [19]) have to be distinguished from processes ("set of interrelated or interacting activities which transforms inputs into outputs". Processes concern the "problem solving procedure" (second dimension). ISO/IEC/IEEE 15288 defines different process groups and their processes, such as Stakeholder Requirements Definition Process or Architectural Design Process, for SE along the different life cycle stages. According to [20], the concept stage is executed to develop preliminary requirements and a design solution. In particular, it shall identify concepts, assess them and prepare baselines for stakeholder requirements and system requirements. The concept stage can thus in particular be seen as a preliminary iteration of technical processes [19].

B. Model Based System Engineering (MBSE).

The usage of models represents an accepted mean for SE nowadays. The advantages it brings are crucial for developments of new complex systems. INCOSE defines MBSE as "The formalized application of modeling to support system requirements, design, analysis, verification and validation activities beginning in the conceptual design phase

and continuing throughout development and later life cycle phases" [3]. It thus is used from the concept stage on. Significant advantages are reached through abstracting, documenting and verifying with models. The main representative of MBSE is Systems Modeling Language (SysML), [21]. In addition to SysML, some methodologies have been developed for the specific purpose of industrial innovation projects. For instance, a generic SE methodology named "Harmony for SE" has been evaluated for application in safety critical automotive projects. [22] and [23] present the methodology, which proposes a guideline for modeling with a subset of SysML in a domain-independent manner. It therefore starts with the so called "Requirements Analysis", hands over to the "System Functional Analysis" before passing to the "Design Synthesis". "Harmony for SE" thereby sequentially refers to the SysML Use Case Diagram, the Behavior Diagrams and the Structure Diagrams. "Harmony for SE" allows to demonstrate the benefit of a functional analysis of stakeholder requirements for description of a system´s behavior from the concept stage on but "Harmony for SE" doesn't propose an explicit approach to integrate stakeholders and their specific domain knowledge beyond the usage of MBSE with SysML.

C. Requirement Engineering (RE)

RE is "the subset of Systems Engineering concerned with discovering, developing, tracing, analyzing, qualifying, communicating and managing requirements that define the system at successive levels of abstraction". A requirement can then be understood as "a statement that identifies a product or process operational, functional, or design characteristic or constraint, which is unambiguous, testable or measurable, and necessary for product or process acceptability [24].

RE forms an own discipline and in ISO 15288 directly refers to the first both technical processes of: "Stakeholder Requirements Definition Process (Clause 6.4.1)" and "Requirements Analysis Process (Clause 6.4.2)". [24] presents a generic process for RE. It describes the transition from problem to solution space with technical solutions. This approach focuses at first step on the definition of the needs of the developed system. Indeed, it is obvious that, if the purpose of the system is not defined, it is hard to determine in fine if the developed system is satisfying the initial needs. Only then, stakeholder requirements (non-technical description, of what is being developed) and system requirements (technical summary of a development project) can be addressed.

The stakeholder requirements play the central role therefore. Needs in particular allow concretizing functional requirements, which are superordinate to technical aspects, which in turn describe, how the system will be realized. In terms of requirements for the concept stage, a top-level distinction shall be considered in the following way:

- Stakeholders needs: This concerns the purpose for the engineering of the system. It should describe as best as possible what different stakeholders expect from the system.
- Stakeholder functional requirements: A core description of the behavior of the System in the most non-technical way possible [25].

- Constraints of stakeholders: They control the way the capabilities of the system must be delivered.

D. How to integrate Stakeholders? Textual specification, natural language and semantics.

The MBSE model, as described previously, represents the central information artefact for SE, as it is used for handover for further productive steps like Software Engineering. In the information flow from all involved stakeholders towards the model, the textual stakeholder specification and system specification represent further key artefacts. Although the textual specifications, as opposed to the model, are pure documentation artefacts, they are representative for the developed systems and their quality.

The glossary is a central part of these documents and an indicator for their quality. Glossaries represent the central semantic artefacts in a textual specification. They are critical for understanding the specification and contain key information concerning the developed systems. The natural language is the common reference for communication between involved stakeholders. Domains and their experts thereby always refer to their specific terminologies which are key to their knowledge. Knowing that the source of information anyhow must have been the stakeholders, the focus shall be laid on their knowledge. The common reference for linguistics and knowledge are semantics.

In this section, the theoretical background for RE with focus on stakeholder requirements was presented. This builds the link to the specialized domains with their stakeholders as identified for AD. In SE, such an issue can be categorized as a matter of Human System Integration (HIS), which is defined as "An interdisciplinary technical and management process for integrating human considerations with and across all system elements, an essential enabler to systems engineering practice", [3]. With this, it seems usefull to integrate human concerns into SE processes. The main difficulty relies on the fact that requirements, that can be defined as logical, technical or textual elements are hard to be integrated due to their differences by nature.

IV. TOWARDS A METHOD BASED ON KNOWLEDGE MANAGEMENT AND MODELING FOR AUTONOMOUS VEHICLE SYSTEMS ENGINEERING

A. Knowledge Management

As previously said, the multi-disciplinary domains that are involved in AD must be integrated as best as possible in SE. Knowledge Management (KMa) offers a general frame for this [26]. KMa can be understood as the management of knowledge systems in general. It plays a fundamental role in company activities, especially when it comes to innovation projects. The most important challenges are to make knowledge transparent and to be able to transfer it between participants of projects. KMa concerns humans and the professional context in a broader sense. It includes different disciplines such as learning or communication. In this context, the company as socio-technical system moreover plays a superordinate role for measures of KMa.

The knowledge must be separated in its definition from the information. Even if both words may often not be delimitated properly. It is first easy to separate information and knowledge from data, which refers to coding such as numbers, language or pictures. The differentiation between information and knowledge in turn is more complicated. Both can be considered in a static (as packages) or a procedural consideration (as processes). Information in this way can be considered as the process of informing in a procedural consideration. Knowledge can also be considered in such a procedural consideration, where knowledge is a construction of reality that occurs through action. In a static consideration, knowledge can be superordinate to information as it represents information linked with knowledge-elements and integrated into a context of experience [26]. The main difficulty relies on the dimensions of expression of knowledge. In [26], knowledge is presented as an entity which can have projection on three basis dimensions psychological (procedural or declarative), accessibility (from individual to collective) and transferability (explicit vs. implicit).

Thus, in enabling a proper dimensional representation of knowledge, KMa offers a general frame to make tangible the specific knowledge of domains. Key principles of KMa are to make knowledge transparent and to be able to transfer it between individuals.

B. Knowledge Modeling

The remaining issue is then how to model knowledge. For this issue, we can take inspiration from the Artificial intelligence community which models knowledge using ontologies. Knowledge Modeling (KMo) has emerged in Artificial Intelligence with the development of Ontologies. [27] can be considered as a reference for a widespread definition of an ontology as an "explicit specification of conceptualization". Development of ontologies in the context of Artificial Intelligence focuses on making machines and programs intelligent. An ontology is a mean for the creation of knowledge-based systems. Thus, ontologies are widely used in Semantic Web (SW). The SW relies on semantic technology based on standards of W3C (WWW Consortium). Semantic technologies in general address challenges for processes with high information amounts, e.g. within a company [28]. The SW e.g. concerns the transfer and usage of explicit and machine usable forms of knowledge. Information shall be represented comprehensively for humans and machines and the cooperation between human and machines shall be sustained.

Beside the degrees of semantic richness, information artefacts for development of ontologies may be distinguished according to their purpose. Therefore, several types of ontologies are existing such as domain ontologies, metadata ontologies, generic and upper-level ontologies, representational ontologies, method or task ontologies…

C. KMa and KMo in SE

KMa serves as reference for the integration of the domains and their specific contribution for SE. Indirect knowledge transfer further underlines the importance of making knowledge explicit and collective. This in turn explains the value of the practice of KMo. Whereas the SW community makes KMo accessible, there is still a wide-ranging understanding of this practice. It is a fact, however, that it offers methods for building consensus, documenting semantics and improving information. SE requires clarification of the information, such as type and form, available in the domains to be integrated so that relevant, timely, complete and valid information can be provided. Whereas mind maps are used in nearly all contexts, more formal representations of knowledge only hardly succeed, not least because knowledge may be hardly tangible. The context, the purpose, the perspective of consideration, as well as the semantic richness developed in Knowledge Models appear crucial here. Terms and their semantics describe problems and resulting solutions for SE. This statement may be considered carefully in the context of AD especially because it involves many different domains. Even if, ontologies are sometimes used in SE, the diversity of ontologies can be emphasized. In [29], is presented a domain ontology for autonomous systems (OASys), which can serve for both communication and coordination in a research project concerning the domain of autonomous systems. Some other related works such as [30] and [31] are focusing on SE with a consideration of the human communication. Thus, in this way, KMo can contribute to SE. The crucial point is the identification of relevant knowledge and its particularities like context, purpose, validity, timing and perspective. The efforts for modeling shall carefully be adapted to goals. In the sense of KMa finally, KMo may contribute to mutual learning, if only the right knowledge is identified. To make a synthesis of these meanings, Fig. 3 shows a Venn-Diagram representing the relationships between SE/RE and KMa/KMo. Among other things, ontologies may also address issues such as:

- Ambiguity, inconsistency and incompleteness of requirements: Ontologies can help understanding, standardizing, error checking and conflict analysis of requirements.
- Lack of domain knowledge for guiding requirements elicitation: Ontologies can help representing domain knowledge for guiding requirements elicitation and hence produce better quality of specification.
- Communication of requirements: Ontologies may facilitate the communication between stakeholders in RE.

The challenge we need to tackle with is the integration of the knowledge of stakeholders. Besides, the more the constellation for a project is multidisciplinary the higher the integration must be.

V. CONCLUSION AND PERSPECTIVES

The goal of this paper was to sketch an overview of the possible improvements that can bring Knowledge Modeling and ontologies to SE and especially in the case of autonomous driving functions engineering which are tied to multi-disciplinary concerns.
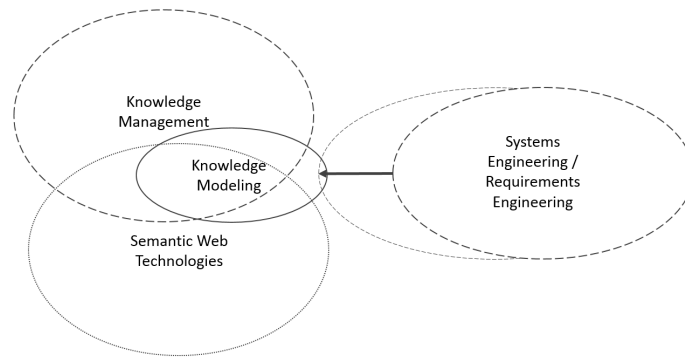
Fig. 3: Venn-Diagram for clarification of relations of KMo, KMa and SW Technologies, and the approach of the present work from RE to KMo

As explained in the previous sections, the quality of SE relies on the integration of all components and functions as required into a system. The groundbreaking contribution of MBSE has been to offer a model as basis for this integration as a step before the production from the concept stage on. The evaluated approach "Harmony for SE" underlines this, as its main strength is a procedure that firstly delivers an encapsulated behavior modeling, which is integrated into modeling of architecture afterwards. SysML delivers a solid foundation for this as a standard modeling language with tooling support.

The present work, with the analyzed references in SE and RE, has identified potentials for the use of Knowledge Models. The main potential for SE relies in the formalization of specific domain knowledge, which allows to summarize topics of high relevance for stakeholders enabling a better integration of their knowledge on specific domains. It is now clear that Knowledge Modeling can be a good candidate for organizing and formalizing the contribution of each domain of stakeholders in a systems engineering task. The next step is then naturally to setup an innovative SE methodology in the context of AD and to test it on well selected functions.

REFERENCES

[1] GfSE, Ed, INCOSE Systems Engineering Handbuch: Hanser, 2012.

[2] INCOSE. [Online] Available: http://www.incose.org. Accessed on: 2015.

[3] SERC, INCOSE, IEEE-CS, SEBoK. [Online] Available: http://sebokwiki.org/wiki/Guide_to_the_Systems_Engineering_Body_of_Knowledge_(SEBoK). Accessed on: Dec. 31 2015

[4] VDA. [Online] Available: https://www.vda.de. Accessed on: 2015.

[5] C. Stiller, Fernando Puente León, and Marco Kruse, "Information fusion for automotive applications – An overview," Information fusion, 2011.

[6] J. Lunze, Automatisierungstechnik: Methoden für die Überwachung und Steuerung kontinuierlicher und ereignisdiskreter Systeme. München: Oldenbourg Verlag, 2012.

[7] K. Lemmer, "Einführungsvortrag "autonomes Fahren"," Apr. 3 2014.

[8] K. Johan Aström, Richard M. Murray, Feedback Systems: Princeton University Press, 2008.

[9] K.-L. Haken, Grundlagen der Kraftfahrzeugtechnik. München: 2008.

[10] ERTRAC Task Force "Connectivity and Automated Driving", "Automated Driving Roadmap," ERTRAC, 2015.

[11] bast, "Rechtsfolgen zunehmender Fahrzeugautomatisierung," 2012.

[12] P. Rau, "Target Crash Population of Automated Vehicles," NHTSA, 2014.

[13] J3016: Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems - SAE International, 2014.

[14] VDA, Automatisiertes Fahren: Stufen des automatisierten Fahrens. [Online] Available: https://www.vda.de/de/themen/innovation-und-technik/automatisiertes-fahren/automatisiertes-fahren.html. Accessed on: Jan. 13 2016.

[15] M. Tonnis, V. Broy, and G. Klinker, "A Survey of Challenges Related to the Design of 3D User Interfaces for Car Drivers," Proc. of 3DUI 2006, IEEE, 2006, pp. 127–134.

[16] S. Vacek, Videogestützte Umfelderfassung zur Interpretation von Verkehrssituationen für kognitive Automobile. Karlsruhe: Univ.-Verl, 2009.

[17] E. POLLARD, P. MORIGNOT, F. NASHASHIBI, "An Ontology-based Model to Determine the Automation Level of an Automated Vehicle for Co-Driving," Istanbul, Jul. 10 2013.

[18] A. D. Hall, "Three Dimensional Morphology of Systems Engineering," IEEE Transactions on Systems Science and Cybernetics, 1969.

[19] SO/IEC/IEEE 15288:2008, Systems and software engineering — System life cycle processes.

[20] ISO/IEC JTC 1/SC 7/WG 7. Secretariat: Canada, "ISO/IEC CD 15288 FCD: Systems Engineering — System Life Cycle Processes," Jul. 2001.

[21] Object Management Group, "Systems Modeling Language: v1.4," Sep. 2015. [Online] Available: http://www.omg.org/spec/SysML/1.4/.

[22] H. P. Hoffmann, "Streamlining the development of complex systems through model-based systems engineering," in 2012 IEEE/AIAA 31st Digital Avionics Systems Conference (DASC), 6E6-1-6E6-8.

[23] H. P. Hoffmann, "IBM Rational Harmony Deskbook: Release 4.1," IBM, 2011.

[24] E. Hull, K. Jackson, and J. Dick, Requirements engineering, 3rd ed. London: Springer, 2011.

[25] G. Kotonya and I. Sommerville, Requeriments Engineering Processes and Techniques. New York: Wiley and Sons, 1998.

[26] E. Hartlieb, Wissenslogistik: Effektives und effizientes Management von Wissensressourcen. Wiesbaden: Deutscher Universitätsverlag, 2012.

[27] T. R. Gruber, "A translation approach to portable ontology specifications," Knowledge Acquisition, vol. 5, no. 2, pp. 199–220,

[28] T. Pellegrini and A. Blumauer, Semantic Web: Wege zur vernetzten Wissensgesellschaft, 1st ed. Berlin: Springer, 2006.

[29] J. Bermejo-Alonso, R. Sanz, M. Rodríguez, and C. Hernández, "Ontology-Based Engineering of Autonomous Systems," in 2010 Sixth International Conference on Autonomic and Autonomous Systems (ICAS), pp. 47–51.

[30] L. van Ruijven, "Ontology for Systems Engineering: Model-Based Systems Engineering," in 2012 European Modelling Symposium (EMS), pp. 371–376.

[31] D. Ernadote, "An ontology mindset for system engineering," in 2015 IEEE International Symposium on Systems Engineering (ISSE), pp. 454–46