Ontology based Scene Creation for the Development of Automated Vehicles

Gerrit Bagschik, Till Menzel and Markus Maurer

Institute of Control Engineering

Technische Universität Braunschweig

Braunschweig, Germany

Email: {bagschik, menzel, maurer}@ifr.ing.tu-bs.de

Abstract—**The introduction of automated vehicles without permanent human supervision demands a functional system description, including functional system boundaries and a comprehensive safety analysis. These inputs to the technical development can be identified and analyzed by a scenario-based approach. Furthermore, to establish an economical test and release process, a large number of scenarios must be identified to obtain meaningful test results. Experts are doing well to identify scenarios that are difficult to handle or unlikely to happen. However, experts are unlikely to identify all scenarios possible based on the knowledge they have on hand. Expert knowledge modeled for computer aided processing may help for the purpose of providing a wide range of scenarios. This contribution reviews ontologies as knowledge-based systems in the field of automated vehicles, and proposes a generation of traffic scenes in natural language as a basis for a scenario creation.**

I. INTRODUCTION

Safety assessment of automated driving functions is an emerging topic in the automotive industry. Several research and development projects show prototypes of automated vehicles in well-defined showcases. When it comes to series production, the ISO 26262 standard defines a state-of-the-art development process to ensure functional safety.

Automated vehicles will have to fulfill a safe driving task in a high number of operating scenarios. To comply with the hazard analysis and risk assessment demanded by the ISO 26262 standard, hazardous events "shall be determined systematically by using adequate techniques" [1, Part 3]. Therefore, operating scenarios, in which malfunctioning behavior of the item can be hazardous, have to be identified. Nowadays, these scenario catalogs are generated by experts during the development process. Expert-based scenario catalogs can be representative for critical situations. However, guarantees for completeness in terms of identifying all possible combinations cannot be given.

A main difference from assisted (level 1) or partial automated driving (level 2) to conditional and high automation (level 3 and 4) is the number of scenarios which have to be investigated and defined. All these scenarios are required to argue that the highly or fully automated driving function is ready for market introduction. Wachenfeld and Winner [2] deduce how many kilometers it takes in field tests for an automated

vehicle to be released. The high number of kilometers may result from the likelihood of encountering possible scenarios which might lead to an accident. However, they conclude that a scenario-based test process can replace time consuming field tests by shifting test case execution to simulation environments. Hence, the developers need more experts or more time for the experts to identify all relevant scenarios and combinations. Expert knowledge is reasonable for investigation of new items in development, but the process of scenario creation by experts is more creative than systematic. To get traceable and comparable scenarios, the experts need at least a consolidated vocabulary and the same understanding of how scenarios are organized.

Bergenheim et al. [3] further describe a semantic gap in safety validation processes. Since actual methods for defining requirements do not support the developers in arguing that the developed item is safe, the semantic gap rather occurs already during system development. The authors claim that the discussion on the *safety of the intended functionality* (SoTIF) can be resolved by a more detailed specification of functional safety requirements or a validation of the functional safety concept towards completeness. Thus, we propose that requirement specification and validation processes can be supported by systematically identified scenario catalogs in the concept phase of the development process.

Systematically identified scenarios based on expert knowledge can help to improve requirements engineering and safety analysis. Also, scenarios provide a basis for test case generation for simulation-based testing of automated vehicles. For this reason, we propose using knowledge-based systems to create scenarios for development of automated driving functions. Ontologies have successfully been used in the field of automated driving in recent years (cf. Section III). Many of these concepts show, how the use of knowledge-based systems can improve decision making and scene understanding in particular driving scenarios. Our concept includes knowledge about the design of traffic infrastructure, and the behavior and interaction between traffic participants. Furthermore, we propose how knowledge about traffic can be represented by a layered concept to divide it into smaller parts and model interactions between the layers.

This contribution is organized as follows: In the next two parts, the concept of ontologies and related work in the field of automated driving is summarized. Afterwards, our approach for traffic scene creation is presented and concluded.

II. ONTOLOGIES

Guarino et al. [4] give a summarized definition of *ontologies* as "a formal, explicit specification of a shared conceptualization." The conceptualization itself is described by Genesereth et al. as "an abstract, simplified view of the world that we wish to represent for some purpose. Every knowledge base, knowledge-based system, or knowledge-level agent is committed to some conceptualization, explicitly or implicitly" [5]. According to Studer et al. [6], ontologies can be divided by different levels of generality. These levels reach from general ontologies, which express domain independent knowledge, to domain and application ontologies, which contain true statements from particular domains like medical applications or electrical engineering or are only applicable to a certain application in a domain.

Fig. 1 shows the structure of ontologies with terminological and assertional boxes. Both types of boxes are combined to form the knowledge base expressed by an ontology.
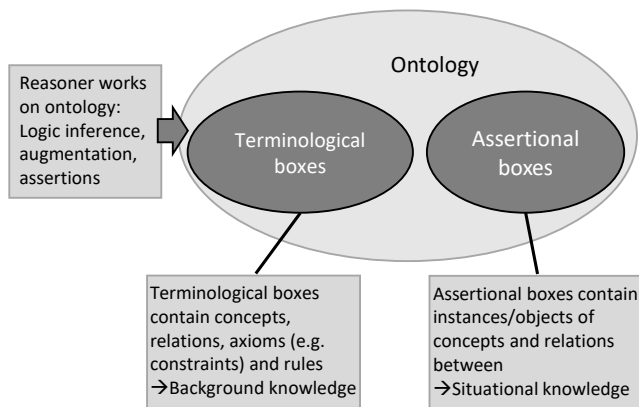

Fig. 1. Architecture of ontologies from Hülsen [7]

Terminological boxes describe the concepts of a domain. These concepts are expressed using hierarchical classes, axioms and properties. Assertional boxes represent instances of classes and observed facts of situative knowledge about the world. Following the definitions by Ulbrich et al. [8], assertional boxes describe scenes extracted from a given set of possible world entities and relations (defined by terminological boxes).

The concepts stored in an ontology have to be readable and understandable by human experts and computers. Humans can read the structure of an ontology, including the hierarchy and the axioms, which are expressed by natural language. For computers, this is accomplished by expressing ontologies in a description logic, which translates every concept of the domain into first order logic.

A common declarative language (which builds the semantic base for modeling) in the field of ontologies is the *Web Ontology Language* (OWL)[1], which is standardized and maintained by the World Wide Web Consortium (W3C). It is intentionally designed for applications on the semantic web, where pages can be connected with the meaning of the terms. For example, this enables better search results based on context and not a pure comparison of terms. OWL includes three variants of the language, namely OWL Lite, OWL DL (Description Logic), and OWL Full. The variants differ in the expressivity of the language, which increases from OWL Lite to OWL Full.

Ontologies are stored in first order logic, which allows reasoners to infer knowledge on terminological and assertional boxes. Reasoners are able to identify hierarchical missing concepts on terminological boxes, check for conflicts in the modeled concepts and check if similar concepts exist over the knowledge base. Assertional boxes can be checked for consistency and associated with the concepts from terminological boxes to infer situational knowledge, for example from sensor data. An overview of reasoners and a comparison study is given by Abburu [9].

III. RELATED WORK

Ontologies have been used for various applications in the field of automated driving. A major part of the contributions comes from work in recent years on situation assessment and behavior planning.

Armand et al. [10] describe how ontologies can be applied to model interactions based on spatio-temporal relations between traffic participants and infrastructure. They are applied to infer how classified object groups will behave and interact with the host vehicle in the future. Sensor data are used as assertional boxes for an ontology and to develop a human-like scene understanding. The understanding is based on object tracks, map data, and the dynamic state of the host vehicle. Therefore, the concept classes are divided into mobile entities, static entities, and context parameters which describe spatio-temporal relations. Relations between the classes are described by object properties for actual behavior like *goes towards* or necessary behavior like *has to decelerate* to maintain safe traffic. To integrate readings from vehicle sensors, the classes are annotated with data properties to map physical measures to the knowledge represented in the ontology. For the assessment and prediction step, behavior rules are stored in the *semantic web rule language* to infer knowledge from the terminological boxes to a given assertional box from the sensor data. The results are taken from an experimental car setup and show that ontology based knowledge can be used to assess traffic scenes in real time applications.

Similar scene understanding approaches are described by earlier contributions in Hülsen et al. [7] and Hummel et al. [11], as well as in Zhao et al. [12] [13] and Mohammad et al. [14].

---

[1]The abbreviation has switched letters due to a typo on a mailing list and the fact that OWL, like the bird, is easier to remember than WOL.

Ulbrich et al. [15] propose an environmental model derived from a knowledge base with hierarchical classes and relations between the entities. The ontology is implemented in C++ to provide an environmental model, which was updated by sensor data and used to make online decisions in the test vehicle *Leonie* of the project Stadtpilot [16].

Geyer et al. [17] formulate the need for a unified understanding of terms and definitions for automated driving to generate requirements expressed by use cases or scenarios. For this purpose, they propose an ontology to define and order the terms ego vehicle, scenery, scene, situation, scenario, driving mission, and route. Use cases and scenarios can be defined in a unified representation with the order derived from the ontology. The ontology is not technically implemented and shall form a basic understanding by reading the publication. After proposing the ontology, the authors emphasize two challenges in the generation of requirement catalogs: top-down development and the form of representation. On the one hand, information must be as complete as possible. On the other hand, they shall be understandable by humans involved in the development process. The main utilization of the proposed nomenclature and concept is to homogenize two projects in the field of cooperative vehicle guidance. A top-down approach for generating the catalogs was not described in the contribution. However, usage of guidelines for road design was mentioned as a starting point.

The ontology suggested by Geyer et al. was reviewed, unified with other contributions and further defined by Ulbrich et al. [8]. They defined differences and the coherence between the properties of each term. This contribution follows the definitions of Ulbrich et al. for the terms *scene, situation* and *scenario*.

Xiong [18] proposes a framework for *scenario orchestration with autonomous simulated vehicles* for simulation of test cases based on driving scenarios for automated vehicles. Therefore, the framework consists of an ontology for scenario orchestration (OSO), virtual driver(s), a collection of supporting modules in a scenario management module (SMM), and a scenario observer. One of the main concepts is a simulation supervision linked with the driver models, which calculate all interactions between the entities in the simulation and the necessary tools according to the defined scenario depending on the simulation framework. This framework has been evaluated in two simulation platforms by executing multiple predefined scenarios.

The ontology used in the framework describes concepts and relations between driving context, task representation, (simulative) actions, simulation monitoring, and temporal representations between entities in the simulation. The scenario representation around a vehicle (in this case) consists of road segments, intersections, and vehicles. Further, the orchestration framework is further able to determine which actions of vehicles are referred to which monitoring devices and measurements. This allows the framework to represent all of its tasks in relation to the actual simulated scenario. Nevertheless, scenarios for evaluation have been designed by experts and the

thesis has the focus on orchestration of all entities regarding simulation using the knowledge-based system.

To summarize the related work, we assume that ontologies provide a suitable framework for various applications in automated driving. This includes support of automated driving functions by inferring knowledge about traffic scenarios and enhancements in traffic safety by supporting a centralized traffic system. In regard of our focus on scene creation, we further investigated the concepts of Geyer et al. [17] and Xiong [18]. Ontologies from contributions discussing scene understanding are compared regarding the represented entities and hierarchies of classes to provide a basis for our knowledge modeling approach. In contrast to approaches for scene understanding from sensor data through ontologies, our approach for scene generation from an ontology needs to derive every possible assertional box from the modeled knowledge represented as terminological boxes. So far, we did not find contributions utilizing ontologies for initial scene creation.

IV. Ontology-based Scene Creation

As introduced in the beginning, automated vehicles will have to accomplish a large number of operating scenarios during their lifetime. Before deploying automated vehicles in public traffic, developers have to investigate these scenarios in the development process in order to ensure safe systems. Go and Carroll [19] describe this approach as scenario-based design paradigm for complex systems. The main idea of the concept is that multiple views of the same set of operating scenarios of a system by different stakeholders are needed to get well-engineered requirements. For automated vehicles, scenario-based design can help to analyze the system from multiple points of view. This includes defining the behavior of the system, human machine interaction, risk analysis, as well as the test process in simulations before going to expensive field tests.

According to Ulbrich et al. [8] "a scenario describes the temporal development between several scenes in a sequence of scenes. Every scenario starts with an initial scene." This contribution aims at proposing a creation process for (initial) scenes.

In a previous contribution [20], we propose a scene creation for a hazard analysis and risk assessment supported by a database at the example of an unmanned protective vehicle in the project aFAS [21]. Since the outcome of initial scenes will be identified by the experts in the project, we focus on the creation of scenes. A major disadvantage of combining database entries to generate scenes is that many of them were physically not possible or unreasonable. Databases contain no semantics which leads to unreasonable scenes. The project aFAS works, compared to other automated vehicles, in a limited use case where a protective vehicle shall operate unmanned and unsupervised on the hard shoulder of motorways. Our conclusion is that the process based on databases would not scale for wider use cases and more complex[2] scenes. The

[2]Related to quantity of participating elements

observation showed that knowledge is implemented implicitly by considering exceptions in the combination step in the database processing. For example, the knowledge about which operating mode can be executed in which area of the hard shoulder (normal or on- and off-ramps) was not explicitly designed in the database, but considered in the creation process. When it comes to arguing the safety of a system, assumptions and knowledge should be represented rather explicitly than implicitly to maintain traceability throughout the development process.

To get more accurate and useful scenes, we need a system that knows how data can be combined and what meaning data has in the real world. Ackoff [22] describes the hierarchy of data, information, knowledge and wisdom as shown in Fig. 2. To conclude the work of Ackoff, instead of data we
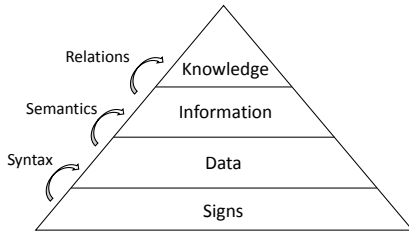
Fig. 2. Hierarchy of signs, data, information and knowledge adopted from Ackoff [22]

use information with semantics to combine information to meaningful scenes of real world traffic. Ontologies provide knowledge bases which are understandable for humans and computers. Based on the concepts of Ackoff, we propose a process for an ontology based scene creation as shown in Fig. 3. The following subsection will explain the knowledge acquisition, how knowledge is modeled, and the combination process.

A. Knowledge acquisition

Knowledge systems are divided into knowledge acquisition and knowledge representation. Guidelines for the creation of traffic infrastructure can be used for knowledge acquisition. These guidelines describe how infrastructure is organized, named, and how the relations can be represented. In Germany, guidelines for the creation of motorways, rural roads, inner city roads, crash infrastructure, markings, traffic lights, and signs exist and are periodically updated to new regulations. There are also various scenario catalogs from past projects or system descriptions, which contain knowledge about scenarios. These sources can also be used if the nomenclature of scenarios is translated properly. As Geyer et al. [17] describe, each project or catalog has its own nomenclature and concepts how to organize knowledge.

Guidelines apply to whole domains of traffic. The number of scenes, which can be created by combination of all entities of a domain, is very large. However, only a selection of scenes depending on the functional range is relevant for investigation. Therefore, the functional description of the system has
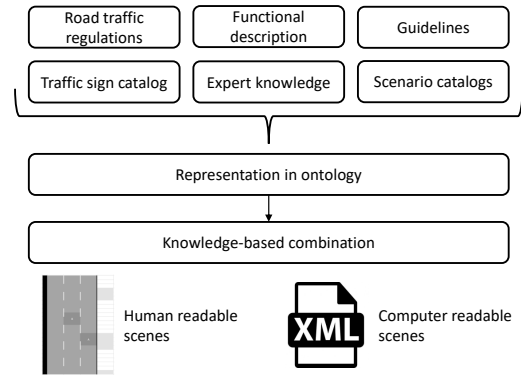
Fig. 3. Ontology-based process for scene creation

influence on the knowledge base. When a system is planned to operate automatically on the hard shoulder only, as the vehicle in the project aFAS, scenes in which no hard shoulder is present are not necessary for the evaluation. However, for the traceability of assumptions made in the analysis, it is important to explicitly represent decisions on which entities or relations from the knowledge are irrelevant for the system.

The third important aspect besides guidelines for traffic domains and the functional description is expert knowledge. On the one hand, every guideline defines only a part of traffic infrastructure and we need to represent how these parts can interact and which dependencies they have to each other. For example traffic rules regarding individual speed limits for each lane on motorways are bound to at least three lanes for each driving direction. On the other hand, we need to represent real world deviations from the guidelines and very unlikely entities or combinations of these. Reschka [23] describes safety-relevant events and how dilemma scenarios can occur from event chains in traffic. Knowledge about the causes of event chains is important to identify functional boundaries of the item and to derive meaningful safety concepts.

B. Layered model for knowledge representation

To organize all information in a knowledge base, we propose to use an adapted layered model for scene representation based on the work of Schuldt [24] as shown in Fig. 4. The first layer describes the layout of the road, including markings and topology. Schuldt [24] proposes to use basic elements like straights, curves and clothoides to define geometries. The parameters for the geometric models can be derived from guidelines in each domain of traffic (motorways, rural roads and urban roads). Further, the basic layer defines how the geometries are related to each other to build up the topology. For example, motorways in Germany follow defined layouts where markings, parameters and relations are given by standards. These guidelines can be used to model hierarchical concepts on how geometries representing lanes can be put together to build valid road snippets for traffic scenes.

The second layer adds traffic infrastructure to the road-level. Traffic infrastructure is part of the scenery, but not all sta-
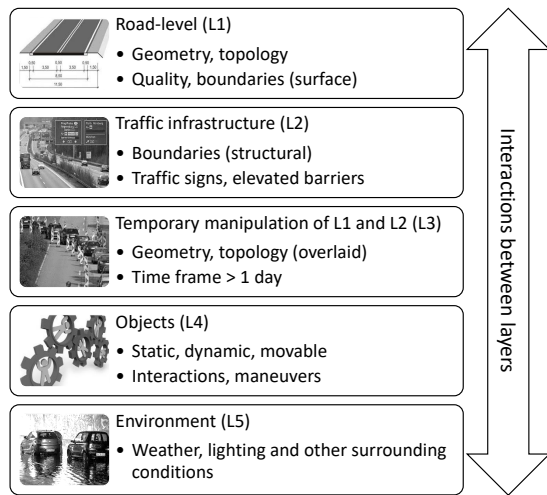
Fig. 4. Layer model for the representation of driving scenes adopted from Schuldt [24]

like time-to-collision or time-headway. If the rear vehicle of two following vehicles is driving fast in the same lane this maneuver can be categorized as approaching. These rules can be expressed by the semantic web rule language and extend the conceptualization of classes of traffic participants by behavior rules. Maneuvers can also have relations to other layers if a car is approaching some infrastructure. For our approach we extended the maneuvers with *fall back* as opposite to approach to describe a constellation where a vehicle behind another one has a lower speed.

In the fifth layer, Schuldt [24] describes environmental effects like weather but also the influences on infrastructure like aquaplaning. In our concepts, environmental effects will also consider influence on the interactions between traffic participants which results in different parameter ranges for relative parameters.

tionary objects belong to the traffic infrastructure. This differs from the four-layered concept of Schuldt [24] as we focus on separating semantic relations for an automated creation of scenes. For the creation process we propose to model traffic rules like speed limits or no passing zones as top-level classes. The sub-classes then represent instantiations of the traffic rules through traffic infrastructure like signs or road markings, which express the rules for the driver. Instantiations of traffic rules have several dependencies on each other which can be extracted from road traffic regulations. Due to dependencies on markings, the influences on the first layer have to be modeled according to the traffic rules.

Temporary manipulations of the first two layers are represented in the third layer. In this concept, the time frame for temporal manipulations was chosen to one day. Other objects which persist longer than one day can be modeled as infrastructure on the layers below. The layer contains classes which represent how construction sites have to be marked, routed and secured. The resulting changes to the original layout are marked as manipulation in the resulting traffic scenes.

All objects which do not necessarily belong to the traffic infrastructure are modeled in the fourth layer. Stationary and movable objects can be placed without extensive changes in the relations of the traffic infrastructure. Traffic participants are categorized into object classes like cars, trucks, cyclists, etc. To define interactions between the participants, we use atomic maneuvers which are disjoint to each other. Based on Tölle [25], Reschka [26] proposes to separate the whole driving task into nine maneuvers: drive up, follow, approach, pass, lane change, turn, turn back, and safe stop.

Each of these maneuvers is described by relational parameters to surrounding objects and semantic rules. For example the maneuver *approach* can be described by the distance to the approaching object, relative speed and derived parameters
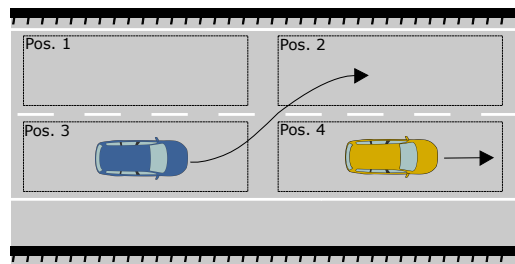

Fig. 5. Example scene

Fig. 5 shows a simple scene including:

- Layer 1: Layout RQ 31 with 2 lanes and a hard shoulder [27]
- Layer 2: Solid lines on the track limits, dashed line between lanes and solid crash barriers besides the road
- Layer 3: Nothing
- Layer 4: Blue vehicle starting a lane change to the left behind a yellow car which follows the right lane
- Layer 5: Normal weather and temperature conditions (not visible in the sketch)

C. Process for creation of traffic scenes

After the ontology for the given use case is modeled, the process for creation uses the knowledge to create valid traffic scenes. The ontology clusters entities in natural language and assigns a formal order through conceptualization. We propose to model parameter relations with data properties in the ontology. Each entity (represented by a word) in the ontology represents multiple relations to parameters in the physical state space. Lanes in the traffic network may include parameters like width, condition and friction coefficient. To represent interactions of layers, we annotate if an entity includes or influences a parameter. For example, rain in the weather layer has influence on the friction coefficient, which is included in the lanes in the first layer.

For the creation of all possible traffic scenes from a given ontology, we need to derive all possible assertional boxes. This stays in contrast to the contributions presented in Section III.

All approaches aim at inferring knowledge from terminological boxes to augment observed assertional boxes.

The traffic scene creation starts with layer 1 and 2 from the layered model. All concepts of possible road layouts according to German guidelines are stored in our ontology. Each layout is described by mandatory and optional elements. The RQ 31 layout from Fig. 5 at least consists of two lanes and a hard shoulder, and optional barrier and embankment elements as well as possible traffic rules like speed limitations. Our ontology uses statements like *consists_of*, *has_optional*, *enables*, and exact cardinalities for modeling possible road network layouts. For the automated creation process we use the java-based OWL-API[3] to access the elements and relations and build up every valid combination allowed by the modeled rules. Each layout then is stored in a separate assertional box with concrete instances where traffic scenes can take place. The layout from our example in Fig. 5 would then be expressed by Fig 6.
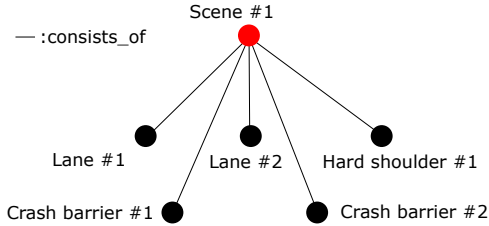

Fig. 6. First step of scene creation containing existential information

The generated infrastructure scenes only contain information which elements exists in a certain scene. To gain a useful scene description, the instances have to be arranged. This is the first step of logic reasoning in our process. Elements can be arranged *right*, *left*, *in_front_of*, and *behind* from each other. Thereby right and left as well as in_front_of and behind are reflexive properties. That means, that if one element is right of another, the second element automatically is left of the first one. Arrangement properties are only inferred for direct neighbors in the scene, which results in the properties shown in Fig. 7 for our example.
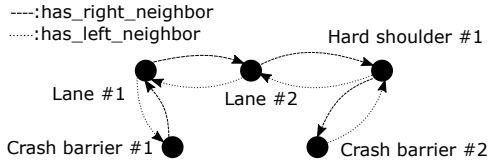

Fig. 7. Second step of scene creation containing arrangement information

The third layer of the layered model is not implemented in the first evaluation of our approach.

For the fourth layer we use a combination of permutation and logic reasoning to generate traffic participants and the possible behavior in regard of the given infrastructure and

[3]http://owlcs.github.io/owlapi/

traffic rules. Therefore, a configurable number of positions per lane is distributed on the infrastructure. Fig. 5 shows two possible positions on each lane of the scenery. The positions are generated based on a relation (*offers_position*) of the lane classes of the ontology. Each lane then gets the defined number of instances of positions (number 1 to 4 in our case). The arrangement of the position instances is inferred similar to ones before and based on the arrangements of the infrastructure. So far, only lanes offer positions, but for future adaptations of the approach the hard shoulder, for example, could offer positions for standing vehicles too. The second step then adds a defined number of traffic participants on the positions and permutes them to possible combinations. Fig. 8 shows the inferred scene for the example from Fig. 5.
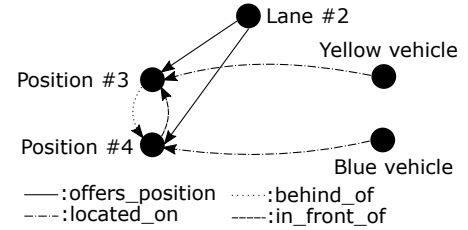

Fig. 8. Third step of scene creation containing traffic participants and arrangement information

This step considers elimination of reputations for semantic classes. In our example, it would make no difference if the blue vehicle is behind the yellow one or the other way around (semantic reputation). If the example would contain a truck and vehicle (passenger car) both combinations would make up another scene and would have to be generated. After the traffic participants are located relative to each other, driving maneuvers can be inferred from the positioning. For each maneuver a semantic web rule has to be implemented in the ontology. Semantic web rules can combine logic operators into rules. For example, a lane change to the left is possible if the car which shall conduct the maneuver has a neighboring lane on the left and has no car on the positions beside and in front of it in the neighboring lane. Based on the arrangement, every possible maneuver (follow, approach, fall back, lane change right/left, start from stand) for every traffic participant in the scene is inferred. In the last step, invalid or forbidden (expressed by semantic web rules) combinations are eliminated from the scene catalog. Forbidden maneuvers are based on the traffic rules from layer 2 (for example forbidden overtaking for trucks) and invalid combinations are given by the creation process. If potentially critical scenes shall be assessed, a scene in which two cars can change the lane towards the same position on the infrastructure could be generated. For normal (comfort) driving scenes, this combination would be invalid.

In the fifth layer predefined weather setups (sunny, rainy, cloud, etc.) are modeled and permuted at the end of the creation. For future investigations weather conditions will have recursive influence on allowed maneuvers and traffic rules (e.g. speed limits during wet road surface).

V. Preliminary Results and Discussion

To evaluate the concept, we modeled an ontology with 284 classes, 762 logical axioms and 75 semantic web rules to generate scenes for German motorways. Depending on the number of positions per lane in the infrastructure and the number of traffic participants we are able to automatically generate a large number of scenes. These created scenes in natural language can be used to investigate and define safe behavior of an automated vehicle in the concept phase or to derive test cases for simulation environments with a high range of varieties. Fig. 9 shows a part of a scene from the automated creation process.
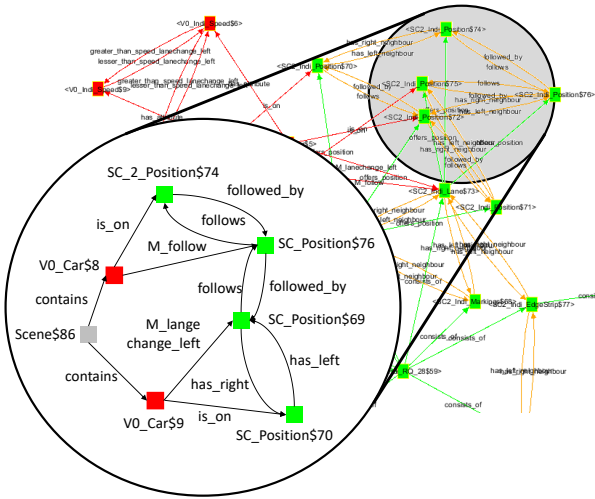
Fig. 9. Example scene graph. Red squares mark vehicles, green squares mark positions on the infrastructure, gray square marks the scene, arrows and captions describes relations and maneuvers.

Red squares mark vehicles, green squares mark positions on the infrastructure, gray square marks the scene, arrows and captions describes relations and maneuvers. For a first evaluation of this contribution we created a typical infrastructure setup on a three-lane motorway with three vehicles and positions per lane in good weather conditions. Therefore, we got 1016 traffic scenes which contain all combinations of three vehicles with possible maneuver combinations on the motorway. As introduced, we successfully removed semantic duplications, as the vehicle are permuted with repetition in the same semantic classes. Without this adjustment the number of possible scenes would be more than three times larger. Furthermore, we generated different combinations of Layer 1 and 2 for two-, three-, and four-lane motorways with different traffic rules. For these examples we got a total number of 652 infrastructure setups without combinations of traffic participants. These number indicate how large a full combination of all layers would be if three or more traffic participants are combined with 652 infrastructure setups. Since the complete description of scenario in natural language is still a large document, we also developed a simple HTML based visualization as shown in Fig. 10. This gives an user a quick overview of the infrastructure and the vehicles with maneuvers in the scene.
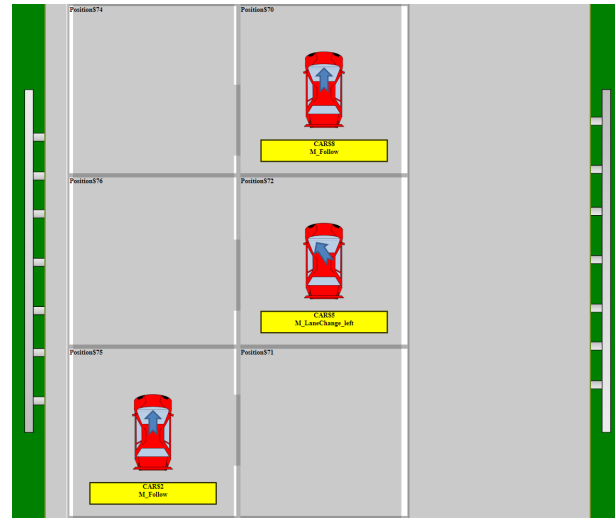
Fig. 10. Example HTML bases visualization of an automatically generated scene.

As a major advantage we see the reduced complexity in the knowledge base compared to the resulting scene catalog. To find missing elements on the layers, analysts do not need to investigate the scene catalog. Instead, it is possible to quickly access knowledge on the layers and check the modeled constraints and rules. If the elements are correctly modeled in the ontology, the elements will also be correctly combined with other layers in the resulting scene catalog.

Besides all combinatorial creation of traffic scenes, the approach lacks some disadvantages against real world scenarios. For now, the positioning in the lanes do not take a vehicles dimensions into account. This excludes, for example, two motorcycles riding next to a truck, which has multiple times a length of a motorcycle. There are also possible infrastructure setups which do not exactly apply to rules of guidelines. These cases can only be identified by real world data, which then may lead to new semantic rules and constraints in the knowledge base. To conclude our results, we propose to use our approach complementary to real world data analysis and as a base for creation of broadly variated traffic scenes for e.g. simulation based testing.

VI. Conclusion and Future work

In this contribution we proposed an approach for a knowledge based scene creation for automated vehicles. Ontologies are a representation of knowledge-based systems, which are widely applied in semantic web applications. Publications from the recent years show that ontologies also can represent traffic scenes. In contrast to scene understanding approaches for automated vehicles, we proposed a method to derive possible observations (assertional boxes) from modeled knowledge (terminological boxes).

A main advantage over creative expert based scene creation methods is the automated creation based on formalized knowledge. For the validation and verification, engineers do not have to review the resulting scene catalog but only the modeled knowledge base.

Based on the fully connected scene graph, we will further investigate the transformation of natural language based scenes to simulation data formats like OpenScenario and OpenDrive. To make the scene catalog handlebar for human experts, we need to identify valid abstractions concepts to conduct analysis steps on the scene catalog (for example in a hazard analysis and risk assessment).

ACKNOWLEDGMENT

REFERENCES

[1] *Road vehicles – Functional safety*, ISO Std. 26 262, 2016.

[2] W. Wachenfeld and H. Winner, "The Release of Autonomous Vehicles," in *Autonomous Driving*, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds. Berlin, Heidelberg, Germany: Springer Berlin Heidelberg, 2016, pp. 425–449.

[3] C. Bergenhem, R. Johansson, A. Söderberg, J. Nilsson, J. Tryggvesson, M. Törngren, and S. Ursing, "How to Reach Complete Safety Requirement Refinement for Autonomous Vehicles," in *CARS 2015-Critical Automotive applications: Robustness & Safety*, Paris, France, 2015.

[4] N. Guarino, D. Oberle, and S. Staab, "What is an Ontology?" in *Handbook on Ontologies*, S. Staab and R. Studer, Eds. Berlin, Heidelberg, Germany: Springer Berlin Heidelberg, 2009.

[5] M. R. Genesereth and N. J. Nilsson, *Logical Foundations of Artificial Intelligence*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1987.

[6] R. Studer, V. R. Benjamins, and D. Fensel, "Knowledge engineering: principles and methods," *Data & knowledge engineering*, vol. 25, no. 1-2, pp. 161–197, 1998.

[7] M. Hülsen, J. M. Zöllner, and C. Weiss, "Traffic intersection situation description ontology for advanced driver assistance," in *2011 IEEE Intelligent Vehicles Symposium (IV)*, Baden-Baden, Germany, 2011, pp. 993–999.

[8] S. Ulbrich, T. Menzel, A. Reschka, F. Schuldt, and M. Maurer, "Defining and substantiating the terms scene, situation, and scenario for automated driving," in *2015 IEEE International Conference on Intelligent Transportation Systems (ITSC)*, Las Palmas, Spain, 2015, pp. 982–988.

[9] S. Abburu, "A survey on ontology reasoners and comparison," *International Journal of Computer Applications*, vol. 57, no. 17, 2012.

[10] A. Armand, D. Filliat, and J. Ibañez Guzmán, "Ontology-based context awareness for driving assistance systems," in *2014 IEEE Intelligent Vehicles Symposium (IV)*, Dearborn, 2014, pp. 227–233.

[11] B. Hummel, W. Thiemann, and I. Lulcheva, "Scene understanding of urban road intersections with description logic," in *Logic and Probability for Scene Interpretation*, ser. Dagstuhl Seminar Proceedings, A. G. Cohn, D. C. Hogg, R. Möller, and B. Neumann, Eds., no. 08091. Dagstuhl: Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2008.

[12] L. Zhao, R. Ichise, T. Yoshikawa, T. Naito, T. Kakinami, and Y. Sasaki, "Ontology-based decision making on uncontrolled intersections and narrow roads," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, Seoul, Korea, 2015, pp. 83–88.

[13] L. Zhao, R. Ichise, Y. Sasaki, Z. Liu, and T. Yoshikawa, "Fast decision making using ontology-based knowledge base," in *2016 IEEE Intelligent Vehicles Symposium (IV)*, Gothenburg, Sweden, 2016, pp. 173–178.

[14] M. A. Mohammad, I. Kaloskampis, Y. Hicks, and R. Setchi, "Ontology-based Framework for Risk Assessment in Road Scenes Using Videos," *Procedia Computer Science*, vol. 60, pp. 1532–1541, 2015.

[15] S. Ulbrich, T. Nothdurft, M. Maurer, and P. Hecker, "Graph-based context representation, environment modeling and information aggregation for automated driving," in *2014 IEEE Intelligent Vehicles Symposium Proceedings (IV)*, Dearborn, MI, USA, 2014, pp. 541–547.

[16] T. Nothdurft, P. Hecker, S. Ohl, F. Saust, M. Maurer, A. Reschka, and J. Böhmer, "Stadtpilot: First fully autonomous test drives in urban traffic," in *2011 IEEE International Conference on Intelligent Transportation Systems (ITSC)*, Washington D.C., USA, 2011, pp. 919–924.

[17] S. Geyer, M. Kienle, B. Franz, H. Winner, K. Bengler, M. Baltzer, F. Flemisch, M. Kauer, T. Weißgerber, R. Bruder, S. Hakuli, and S. Meier, "Concept and development of a unified ontology for generating test and use-case catalogues for assisted and automated vehicle guidance," *IET Intelligent Transport Systems*, vol. 8, no. 3, pp. 183–189, 2014.

[18] Z. Xiong, "Creating a computing environment in a driving simulator to orchestrate scenarios with autonomous vehicles," Ph.D. dissertation, University of Leeds, Leeds, United Kingdom, 2013.

[19] K. Go and J. M. Carroll, "The Blind Men and the Elephant: Views of Scenario-based System Design," *interactions*, vol. 11, no. 6, pp. 44–53, 2004.

[20] G. Bagschik, A. Reschka, T. Stolte, and M. Maurer, "Identification of potential hazardous events for an Unmanned Protective Vehicle," in *2016 IEEE Intelligent Vehicles Symposium (IV)*, Gothenburg, Sweden, 2016, pp. 691–697.

[21] T. Stolte, A. Reschka, G. Bagschik, and M. Maurer, "Towards Automated Driving: Unmanned Protective Vehicle for Highway Hard Shoulder Road Works," in *2015 IEEE 18th International Conference on Intelligent Transportation Systems (ITSC)*, Las Palmas, Spain, 2015, pp. 672–677.

[22] R. L. Ackoff, "From data to wisdom," *Journal of applied systems analysis*, vol. 16, no. 1, pp. 3–9, 1989.

[23] A. Reschka, "Safety Concept for Autonomous Vehicles," in *Autonomous Driving*, M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, Eds. Berlin, Heidelberg, Germany: Springer Berlin Heidelberg, 2016, pp. 473–496.

[24] F. Schuldt, "Ein Beitrag für den methodischen Test von automatisierten Fahrfunktionen mit Hilfe von virtuellen Umgebungen - English title: Towards testing of automated driving functions in virtual driving environments," Ph.D. dissertation, Technische Universität Braunschweig, Braunschweig, 2017.

[25] W. Tölle, *Ein Fahrmanöverkonzept für einen maschinellen Kopiloten, english title: A concept of driving maneuvers for a machine co-pilot*, ser. Fortschritt-Berichte VDI. Düsseldorf: VDI-Verlag, 1996.

[26] A. Reschka, "Fertigkeiten- und Fähigkeitengraphen als Grundlage für den sicheren Betrieb von automatisierten Fahrzeugen in städtischer Umgebung - English title: Skill and ability graphs as basis for safe operation of automated vehicles in urban environments," Ph.D. dissertation, Technische Universität Braunschweig, Braunschweig, Germany, 2017.

[27] *Richtlinie für die Anlage von Autobahnen*, Forschungsgesellschaft für Straßen und Verkehrswesen Std., 2009.