

# An Adaptive Software and Systems Architecture for Driver Assistance Systems based on Service Orientation

M. Wagner, D. Zöbel, and A. Meroth

**Abstract**— State of the art automotive software systems presume a static configuration of electronic control units and software components. Assuming that future Driver Assistance Systems (DAS) will deploy dynamically changing software configurations, the authors propose an approach towards an adaptive software and systems architecture for DAS. The paper focuses on systems, which assist the driver while backing up truck and trailer combinations. These systems are characterized by a high degree of distribution as well as a high level of heterogeneity and changeability even at runtime. In order to meet these demands, the authors suggest the usage of service orientation combined with software agents. The functional units of these systems are encapsulated in services which are dynamically orchestrated by the agents whenever the system changes. This paper describes the ideas of this approach using a real world example. It also names and classifies the scenarios of re-configuration. The article concludes with a lookout on the future work within this project.

**Index Terms**— automotive software architecture, automotive systems architecture, driver assistance systems, re-configuration, service oriented architecture, software agents

## I. INTRODUCTION

### A. Driver Assistance Systems

Modern Driver Assistance Systems (DAS), often referred to as "Advanced Driver Assistance Systems" and Driver Information Systems support the driver during his main tasks (Table 1) of driving, operating the vehicle and operating devices in the vehicle [17]:

TABLE I. DRIVING TASKS [17]

Primary Conducting vehicle	Task: the	Stabilization, Route keeping, Navigation
Secondary Operating car devices	Task:	Gear change, Blinking, Wiping
Tertiary Ambient Control	Tasks:	Air Conditioning, Communication/Entertainment Systems

This is usually done by informing the driver through one or more information channels, or modalities (e.g. sound,

Manuscript received September 8, 2011. Marco Wagner was supported by a scholarship of the "Thomas Gessmann-Stiftung", Essen, Germany.

Marco Wagner and Ansgar Meroth are with the faculty of Mechanics and Electronics, Heilbronn University, Heilbronn, Germany.

(e-mail: Marco.Wagner@hs-heilbronn.de).

(e-mail: Ansgar.Meroth@hs-heilbronn.de).

Dieter Zöbel is with the faculty of Computer Science of the University of Koblenz-Landau, Koblenz, Germany (e-mail: Zoebel@uni-koblenz.de).

warning lights, vibration), sometimes by changing the classical semantics of the car's input devices, e.g. by using the steering wheel for triggering an overtaking assistant, or changing the force feedback of the wheel or the pedals to indicate whatever is required [18]. Moreover, they may also directly interact with the vehicle, e.g. by selectively braking or accelerating. For all those purposes, DAS need information about the car and its interaction with the environment. First, systems derived this information from sensors that measured the behavior of the car itself (e.g. sensors measure the acceleration or differences in wheel velocities in order to derive the friction of the street). Nowadays, information is collected with cameras, radars and lasers in order to directly obtain a picture of the environment.

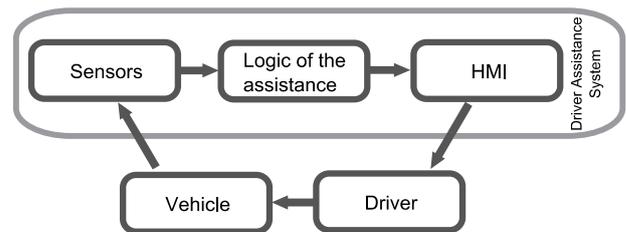


Fig. 1. The generic loop of informing driver assistance systems.

The functional reference architecture of an informing DAS consists of three components. Sensors are used to measure physical quantities characterizing the current state of the vehicle. The logic of the assistance uses these values to calculate some kind of information. This information is presented to the driver through the human machine interface (HMI) which constitutes the third component. Upon the provided information, the driver decides for and conducts an action (e.g. steering, braking) which consequently changes the state of the vehicle. Thus, the driver is part of a generic control loop which is shown in figure 1. There is of course a wide variety of DAS in modern cars. They support the driver in situations like lane changing, parking or stop-and-go traffic.

Current driver assistance systems are characterized by a static systems architecture since the number of electronic control units (ECUs) and the functional and electrical topology of the network connecting them rarely changes during their life cycle. Unlike this, future DAS will base on changing hardware configurations that might even alter at runtime. This may happen through connecting so called nomadic devices like personal digital assistants (PDAs) or Smartphones. Another scenario that is possible is future car-to-car or car-to-infrastructure communication systems where the car and its ECUs will be part of an ever-changing

network topology. A third category are assistants for truck and trailer combinations because of the possibility of changing parts of the vehicle and hereby components of the systems architecture at runtime.

### B. Driver Assistance Systems for Articulated Vehicles

Articulated vehicles have become a focus of the real time systems group at the University of Koblenz-Landau because their motion is not intuitive especially while backing up [14]. In order to assist the operator of a truck and trailer combination, the group developed a multitude of steering assistance systems. The sensors needed for this kind of assistance are a steering angle sensor and a device to measure the bending angle between the drawing vehicle and every trailer articulated. Using these quantities, the logic of the assistance computes an output presented by the HMI.

The type of assistance provided can vary between the acoustical, the visual or the tactile modality or can be given by altering the semantic of the steering wheel [1]. Using the tactile modality, the driver gets information about the current bending angle and its first derivation by adding an additional force to the steering wheel. This force is designed to enlarge the resistance of the steering wheel whenever the bending angle reaches a critical value. Another type of assistance can be offered by changing the semantic of the steering wheel. Instead of controlling the steering angle the driver is able to directly set up the bending angle by turning the steering wheel. In order to do so, the assistance logic calculates and adjusts the steering angle needed for the angle between truck and trailer given by the operator of the vehicle. The acoustical type of assistance generates sounds to help the driver backing up. Just like state of the art parking assistance systems that inform the driver about the situation of the car by emitting audio signals, the acoustical assistance for articulated vehicles warns the driver about critical situations related to the bending angle. A last type of assistance that has been investigated makes usage of the visual modality. Using a camera mounted on the back of the trailer and pointing backwards, the rear view of the vehicle is presented to the driver on a screen. The information content presented is enlarged by overlaying trajectories of the future path of the trailer and the overall vehicle using augmented reality techniques. This enables the driver to anticipate where the truck and trailer combination will be moving in the future and hereby allows him to adjust the steering wheel. The HMI of this assistance system is presented in figure 2.



Fig. 2. The visual assistance system for articulated vehicles [19].

These assistance systems for truck and trailer are highly heterogeneous since the dimensions of the problem are manifold (figure 3). An actual representation depends on:

- the geometry of the truck and trailer combination,
- the type of steering system built into the drawing vehicle,
- the HMI used,
- the distribution of the functional components,
- the kind of sensors used to measure the quantities needed,
- the market situation in the domain of commercial vehicles.

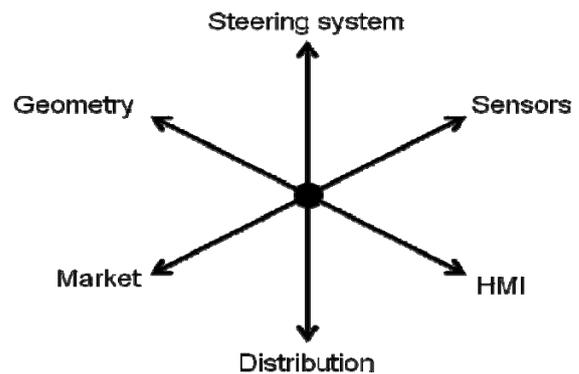


Fig. 3. The dimensions of driver assistance systems for truck and trailer.

The geometry of the vehicle is given by the number and the type of trailers connected. In principle there are two different kinds of trailers: the one-axle trailer and the two-axle trailer. The number of trailers articulated varies too.

The type of steering system is an important factor of the DAS for truck and trailer. There is a range of systems mounted in today's vehicles like simple mechanical steering, servo steering or active steering which allows overlaying an additional steering angle. In future cars steer-by-wire systems will be available as well and will increase the variety even more.

The HMI used is decisive for the modality used to inform the driver. The device presenting the assistance might be a built-in head unit as well as some kind of nomadic device connected temporarily only. Using the acoustical, the visual and the tactile modality the DAS can deploy a multitude of HMIs.

The software and hardware components can be distributed over all the different parts of the vehicle. This increases the complexity of the DAS since the number of possible topologies is big.

An essential part of every driver assistance system is the measurement of the current state of the vehicle. This is done by sensors varying in their principle of measurement as well as in their mounting position. For Driver Assistance Systems for truck and trailer, two quantities are highly important. These quantities are the steering angle of the drawing vehicle as well as the bending angle between truck and trailer. As the steering angle is needed for many other systems within an automobile, it is normally already measured and hereby available within the network system. This is not true for the bending angle since it has not been important to state of the

art systems so far. As there are no measurement systems available on the market, the real time systems group of the University of Koblenz-Landau and the Automotive Systems Engineering department of the Heilbronn University investigated several approaches. In order to ensure consumer acceptance these sensors should be either very cheap or the bending angle should be calculated using existing hardware or the hardware of the measurement system should provide additional functionality to the driver. Meeting the first demand simple angle sensors mounted to the coupler of the hitch and the pivot bearing of the front axle could be used. An alternative may be using the ultrasonic detectors of the parking assistance, mounted in many modern cars, to detect the angle between truck and trailer. If a system is used that is based on a camera mounted on the back of the drawing vehicle and analyzing the movement of the trailer behind, this device could offer a panorama view to the driver whenever no trailer is connected. All these approaches are very different in their principle of measurement and their mounting position which can even be distributed over the different parts of the vehicle. This fact sets up the demand to the architecture of being able to handle heterogeneous systems offering the same functionality.

Another influencing dimension is the market situation. As in the case of passenger vehicles, there is one manufacturer responsible for the whole system. Contrary to this situation, the market for trucks and trailers is much more heterogeneous since there is no single producer covering the overall system.

All these factors lead to a high complexity and variety in truck and trailer assistance systems. Our goal is to develop software and system architecture which manages the demands of this kind of DAS. The high degree of distribution and the multitude of variations will be managed by the principle of information hiding and the specification of well defined interfaces. Also, an adequate communication model has to be defined. In order to handle changes in the system at runtime, some kind of autonomy will be implemented as well. All the principles used will be suited to match the requirements of automotive software systems such as the characteristics of the available network systems and the fact that embedded devices with limited resources are deployed.

The remainder of the paper is organized as follows; Section 2 names approaches related to this topic. In the third section, our approach is described in detail using a real world example. Furthermore, scenarios in which a re-configuration has to take place are examined. The paper finishes with the forth section containing the conclusion and a lookout on future work within this project.

## II. RELATED WORK

In state of the art software and system architectures for the automotive domain changes during runtime are not considered. Even in current standardization projects like AUTOSAR, only static topologies are taken into account [2]. However there are some research projects that examine this new demand in the automotive sector.

Many of these projects are making use of the paradigms of service oriented architecture (SOA). "SOA is an approach to

build applications by using a set of loosely coupled black-box components (called services) orchestrated to deliver a well-defined functionality" [20].

Gacnik, Häger, Hannibal and Köster, for example, describe the possibility of using service oriented architectures to embed external software components into a DAS. As an example, they are using their approach to build up a travel assistance system that uses services from the internet to book railway tickets or order a taxi [3]. Another use of SOA in car-to-car communication is presented by Röckl, Gaczin and Schomerus in [5]. In this case the authors are deploying sensor fusion over more than one car. The usage of Web Services to couple functionalities of nomadic devices and the vehicle itself are proposed by Sonnenberg [16]. The idea is to deploy a central gateway, the so called "Vehicle Software Platform". This unit has two major tasks. First, it offers interfaces in the form of Web Services of all accessible functionalities of the car. Second, it converts the Web Services offered by external devices so that internal units of the vehicle are able to use them. All the systems described above are based on the assumption that only the situation around the vehicle changes while the internal architecture is static which is not true for truck and trailer assistance systems.

Another research project where SOA is used in the automotive domain is described in [6]. The authors Ermagan, Huang, Krüger, Meisinger, Menarini and Moorthy are describing the architecture of a central locking system based on the idea of services. This idea aims on increasing the reusability of software components but it doesn't consider any changes at runtime. An approach to deploy an adaptive configuration recognition, which is an important part of the re-configuration, has been published in [7]. The suggested system relies on a central ECU containing an extensive database. This fact prevents the approach from being used in the heterogeneous field of truck and trailer combinations. The capability of self-organization of biological systems is used by Hofman and Leboch [8]. The principles of these systems are transferred onto technical systems. However, the idea that components merge into static groups that offer their services only within the group environment limits the flexibility of this type of architecture. This limitation prevents this idea of being used for the re-configuration of a truck and trailer DAS.

Brokers are used to carry out runtime adaption in the approach presented in [4]. The authors propose to redirect any communication between the different parts of a system through such a broker. If any component of the system needs some particular functionality it calls the broker passing over the demands of the service call. These brokers hold a list of services along with their functionality and some further attributes. Basing on this database it decides on which service has to be called when getting a request. This approach is very promising but has two main disadvantages. Carrying out every communication using such a broker increases the risk of a complete system failure as well as the data volume to be send to call a service.

Another interesting approach towards adaptive software and system architectures is the use of software agents. "Intelligent agents are software entities that carry out some

set of operation on behalf of a user or another program with some degree of independence or autonomy, and in so doing, employ some knowledge or representation of the user's goals or desires" [15]. In [9] Schmidt, Görzig and Levi present an agent based software architecture for DAS. The main goal of their work is the integration of new components at design time. They do not consider changes at runtime.

The combination of SOA and software agents seems to be a very promising approach when it comes to heterogeneous systems that need to be self-reconfigurable. This combination of methods is already in use in other areas of application. Koutsoukos, Kushwaha, Amundson, Neema and Sztipanovits, for example, use SOA combined with a dynamic service discovery in the field of wireless sensor networks [10]. New approaches on the development of embedded systems based on service orientation are discussed by Tsai, Wei, Paul, Chung, Huang and Chen in [11]. Brogi, Popescu, Gutiérrez, López and Pimentel are applying SOA on embedded peer-to-peer systems like for example a network of temperature sensors [12]. In military environments these methods are used as well. Bridges and Mostashfi describe in a technique on how to dynamically orchestrate a sensor network in [13].

In the next section our approach towards an adaptive software and system architecture is described in detail. Using a real world example the workflow of the orchestration process is explained and scenarios of re-configuration are presented.

### III. TOWARDS AN ADAPTIVE ARCHITECTURE FOR DRIVER ASSISTANCE SYSTEMS

None of the approaches mentioned in section two entirely fulfills all the demands of such a heterogeneous, distributed and dynamically changing system like the DAS for truck and trailer. Some of them are able to handle the degree of distribution but are not designed to handle changes at runtime. Other ideas cover those changes at runtime but are too restricted or not developed to work within an automotive environment. This is why a new approach has to be found. We suggest a combined method that uses service orientation as well as software agents. The main idea is to split up the functionality into small compact units. These modules are implemented as services that offer well defined interfaces as well as mechanisms to allow service discovery. The main benefits of this representation of the software components are increasing reusability and the possibility of better distribution.

An important task in designing service oriented architecture is the so called orchestration<sup>1</sup> of the services. In the case of the DAS for truck and trailer the orchestration assembles the services available to build up an application capable of providing the desired assistance. The functional reference architecture given in figure 1 represents the general layout. The internal structure within the three components of the reference architecture is highly heterogeneous and depends on the services available in the current situation. In order to be able to handle the changes within the system, a dynamic orchestration of the services is needed. Contrary to applications where services are orchestrated in advance, a

dynamic orchestration allows changing the system at runtime. In the approach described here, this is done by one or more software agents. These agents can be situated anywhere in the system. In a first step, they are detecting all the currently available services. This step is followed by a determination of all possible configurations and types of assistance. Next, one configuration is selected and established through advertising the partners to each participating service. In the following subsection A, the re-configuration process is described in more detail using a concrete example. Subsection B contains the scenarios in which re-configuration is carried out.

#### A. Re-configuration in detail

An example of a combination of truck and trailer is given in figure 4. In order to obtain lucidity, the number of services in this particular example is kept low by using a high level of abstraction. In this situation the drawing unit offers a large set of possible human machine interfaces. Through the utilization of the head unit, acoustic signals as well as visual assistance can be output. Besides that, the in-built electric power steering can be used to allow haptic assistance. All these output media are implemented as services as well as the sensor for the steering angle. The trailer connected is a two-axle trailer. The type of sensors used to measure the bending angles between truck and trailer is not restricted in order to increase the level of flexibility. Due to this fact, more than one sensor to measure the same quantity may be available. In the configuration shown in figure 4, there are two sensors for each of the two angles to detect. First, the trailer offers two mechanical sensors placed in the coupler of the hitch and the pivot bearing of the front axle. In addition, a video based sensor consisting of a camera mounted on the truck and a reference object placed on the trailer is available, too. Another camera is mounted at the back of the trailer providing the rear view of the vehicle. The software modules containing the kinematic model of truck and trailer can be situated on any ECU within the vehicle.

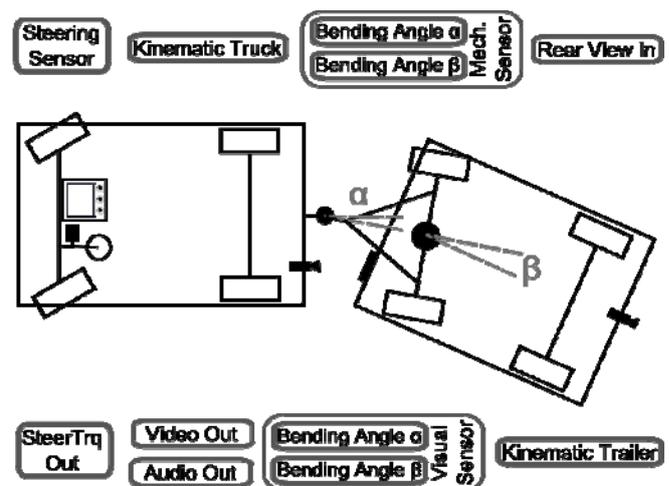


Fig. 4. The services within a typical configuration.

<sup>1</sup>In contrast with some authors in the SOA domain (e.g. [21]) we do not distinguish between orchestration and choreography as both terms describe the same functionality and are pursuing the same goal.

The orchestration starts with the service discovery. In the example given in figure 4, eleven services are detected:

- Services that are delivering input are the steering sensor, the rear view input, bending angle  $\alpha$  (visual sensor), bending angle  $\beta$  (visual sensor), bending angle  $\alpha$  (mechanical sensor) and the bending angle  $\beta$  (mechanical sensor).
- The logical services detected are the kinematic model of the trailer and the kinematic model of the truck.
- Detected services that can present some kind of output are video out, audio out and steering torque out.

In the next step, the possible types of assistance are determined. In this case a visual, acoustical or haptic assistance is possible since the components needed for these systems are available on the vehicle. As there is more than one possibility for the assistance to be provided a decision has to be made. This decision is based on the preferences of the driver which can be determined for example by a user input or a default value derived from prior user decisions. In this example visual steering assistance is selected.

This is followed by the selection of the services used in this configuration. In the given situation the steering sensor and the two kinematic models are chosen because they are unique in this case. Since visual assistance is executed, the video out service and the rear view input are selected as well. In the sensor domain, there is more than one alternative. Since both bending angles are measured by a visual sensor as well as by a mechanical sensor, the system has to autonomously choose the better option. This decision is based on a so called Quality of Service “QoS” parameter that is calculated for each service. The basis of such a calculation might for example be factors like the responding time of the service, the reliability or the accessibility depending on the network system which is used to connect to the service. In the given example, the mechanical sensors are selected.

After the determination of the services, they can be activated by announcing them the partners of the current configuration. For example, the kinematic model of the truck is set up with the address of the steering sensor, the kinematic model of the trailer and the video output. This is needed to allow a distributed processing where the services are exchanging their information directly amongst each other. Compared to a system relying on a central distribution unit, this approach increases the reliability and balances the network load within the bus systems.

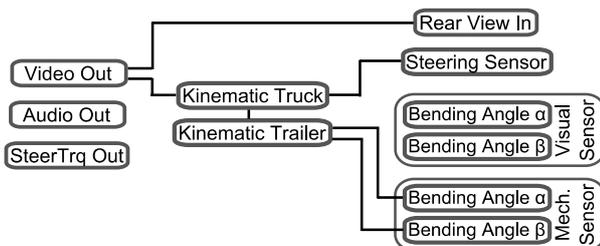


Fig. 5. The orchestration of services in the example given.

The result of the orchestration is presented in figure 5. The services activated for this configuration are connected

virtually to their partners. The newly established configuration is now ready to provide assistance to the user.

### B. Scenarios of re-configuration

A re-configuration might take place if for example a different trailer is connected. In this case, the whole process of orchestration as described before will be executed again. Within the example used earlier in this paper the two- axle trailer may be replaced by a one-axle trailer. This configuration can be seen in figure 6.

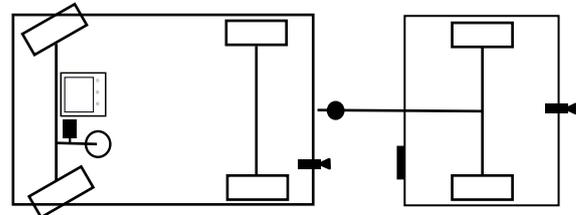


Fig. 6. Configuration with a one-axle trailer.

Another event that causes a re-configuration is a user input to change the type of assistance. In this situation, only the last part of the orchestration, the selection and activation of the services in use is being processed. With respect to the example used the system could switch to offer acoustic or haptic assistance.

Also a re-configuration has to be carried out in the case of the failure of one of the active services. In this situation, there are three options of action. When an essential service that is needed for each type of assistance fails and there is no replacement in the current configuration, no assistance can be provided. In the given example this might happen if e.g. the steering sensor fails.

Another event might be the failure of a service indispensable for the kind of assistance running at the moment. Getting back to the configuration discussed earlier this would mean that the rear view input or the monitor would go out of service. In case that this occurs, a change of the assistance modality would take place. For example, the electric power steering could be used for a haptic assistance.

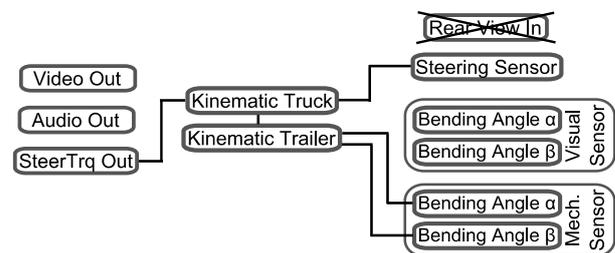


Fig. 7. The orchestration of services for haptic assistance.

Just as described in the situation of a user input, the system would have to select and activate the now utilized services. The result of this re-configuration is shown in figure 7. The system cuts off the rear view in and the video out service and connects the steering torque out service instead. In a third scenario, a service that is backed-up by one or more replacements fails. Here, one of the replacements would be activated and the partners of the service would be updated. In the case of the example configuration, this could be the

failure of one of the mechanical bending angle sensors which could be replaced by their visual equivalent.

#### IV. SUMMARY AND FUTURE WORK

In this paper a new approach to address dynamically changing automotive software systems has been presented. It uses the paradigms of service oriented architecture to enhance the flexibility and reusability and allows a high degree of distribution. The task of the orchestration of the services is executed by software agents that autonomously detect changes within the system and re-configures the DAS.

Since the fundamental concept of the project has been defined, future work will focus on the profound design of the components of this concept as well as on the implementation and validation.

One of the elements that will be further discussed is the Quality of Service parameter. This parameter needs to reflect the performance of each service affected by influences from inside and outside the component. This parameter should be easily computable and allow a comparison between services of the same function.

The character of the QoS parameter influences the operation of the system elementary. This is because a QoS parameter that depends, for example, on some kind of internal error counter or the current system utilization may change its value at any time. In such case, another scenario of re-configuration appears. As described in section 3, in case of a multitude of services handling one task the decision on which service is selected is based on the QoS parameters. In this situation, a change of the hierarchy might take place at any time without changing the systems architecture. In order to not influence the real time capabilities of the DAS a work flow has to be developed to manage such a scenario.

Another component to be specified more precisely is the communication model. In real world scenarios the services introduced might be connected by a mixture of state of the art automotive network systems. The goal is to design a communication system that offers a high flexibility and independence from the actual kind of network used. In addition to this, the overhead produced should be minimal. These often conflicting goals can only be achieved by taking into account the characteristics of the unique network systems used in state of the art vehicles.

The major element of the presented architecture is the service. Continuative work will lead to well-defined interfaces and service descriptions. Compared to web services as the main domain of usage of service oriented architectures, the hardware resources being available in embedded automotive systems are restricted. This fact has also to be considered when designing services for the truck and trailer assistance system. In addition, the granularity of the services has to be defined.

The micro architecture of the software agents used to orchestrate the services has not been defined so far as well. Their design will be driven by the type of adaption used. Concurrently, a mechanism of interaction between each other has to be developed.

An additional topic that will be discussed as the work in this project moves on is the integration of the mechanisms

into the AUTOSAR standard. Since AUTOSAR seems to be the most important standard in the automotive software domain for the near future, the presented methods should be seen within the context of it.

In a last step the truck and trailer assistance will be implemented using the presented architecture. This is done using hardware components that are close to those currently deployed in the automotive domain. The validation of this approach will be done on a full scale prototype to prove the mechanisms developed are working the way expected.

#### REFERENCES

- [1] Berg, Uwe; Zöbel, Dieter (2006), "Visual Steering Assistance for Backing-Up Vehicles with One-axle Trailer", in *Proc. of Vision in Vehicles II*, Dublin, Ireland.
- [2] Isabell Jahnich, Ina Podolski, and Aichm Rettberg, "Towards a middleware approach for a self-configurable automotive embedded system", Uwe Brinkschulte, Tony Givargis, and Stefano Russo, editors, *Software technologies for embedded and ubiquitous systems, Lecture Notes in Computer Science*, Volume 5287/2008, pages 55–65, October, 2008.
- [3] Jan Gacnik, Oliver Haeger, Marco Hannibal, and Frank Koester, "Service-oriented architecture for future driver assistance systems", in *Proc. of International Federation of Automotive Engineering Societies 2008*, Munich, 2008.
- [4] Howard Foster, Arun Mukhija, David S. Rosenblum, and Sebastian Uchitel, "A model-driven approach to dynamic and adaptive service brokering using modes", in *Proc. of International Conferences on Service-Oriented Computing 2008*, Sydney, 2008, pages 558–564.
- [5] Matthias Röckl, Jan Gacnik, and Jan Schomerus, "Integration of car-2-car communication as a virtual sensor in automotive sensor fusion for advanced driver assistance systems", in *Proc. of International Federation of Automotive Engineering Societies 2008*, Munich, 2008.
- [6] Vina Ermagan, To-Ju Huang, Ingolf Krüger, Michael Meisinger, Massimiliano Menarini, and Praveen Moorthy, "Towards tool support for service-oriented development of embedded automotive systems", in *Proc. of Modellbasierte Entwicklung eingebetteter Systeme III*, Braunschweig, Germany, 2007, pages 1–23.
- [7] Thomas Jäger and Mario Kaller. Patent: Verfahren zur adaptiven Konfigurationserkennung, EP 1 967 435 A2, 2008 (in German).
- [8] Peter Hofmann and Stefan Leboch, „Evolutionäre Elektronikarchitektur für Kraftfahrzeuge“, *it – Information Technology*, Volume 47, pages 212-219, 2005 (in German).
- [9] André Schmidt, Steffen Görzig, and Paul Levi, „Eine Software-Architektur für Fahrerassistenzsysteme“, *Informatik aktuell: Autonome Mobile Systeme 2003*, pages 44–55, 2003 (in German).
- [10] Xenofon Koutsoukos, Manish Kushwaha, Isaac Amundson, Sandeep Neema, and Janos Sztipanovits, "Oasis: A service-oriented architecture for ambient-aware sensor networks", in *Proceedings of the 13th Monterey conference on Composition of embedded systems: scientific and industrial issues*, Philadelphia, PA, USA, 2008, pages 125-149.
- [11] Wei-Tek Tsai, Xiao Wei, Ray Paul, Jen-Yao Chung, Qian Huang, and Yinong Chen, "Service-oriented system engineering (SOSE) and its applications to embedded system development", *Service Oriented Computing and Applications*, Volume 1/2007, pages 3–17, Springer Verlag, 2007.
- [12] Antonio Brogi, Razvan Poescu, Francisco Gutiérrez, Pablo López, and Ernesto Pimentel, "A service-oriented model for embedded peer-to-peer systems", *Electronic Notes on Theoretical Computer Science 2008*, pages 5–22, 2008.
- [13] D. Bridges and S. Mostashfi, "Dynamic orchestration of the sensor web", In *Proc. of the International Symposium on Collaborative Technologies and Systems 2008*, Irvine, CA, USA, 2008, pages 88–94.
- [14] Berg, Uwe; Zöbel, Dieter (2007), „Gestaltung der Mensch-Maschine-Interaktion von Lenkassistenzsystemen zur Unterstützung der Rückwärtsfahrt von Fahrzeugen mit Anhänger“, *Mechatronik 2007 - Innovative Produktentwicklung (VDI-Berichte)*, Vol. 1971. pp 575-588, 2007.
- [15] D Gilbert, M. Aparicio, B Atkinson, S. Brady, J. Ciccarino, B. Groszof, P. O'Connor, D. Osisek, S. Pritko, R. Spagna, and L. Wilson, "IBM intelligent agent strategy", Technical report, IBM Corporation, 1995.

- [16] J. Sonnenberg, "A distributed in-vehicle service architecture using dynamically created Web Services", in *Proceedings of the 14<sup>th</sup> International Symposium on Consumer Electronics*, Singapore, 2010, pp 1-5.
- [17] H. Bubb, „Der Fahrprozess – Informationsverarbeitung durch den Fahrer“, in *Proc. Of VDA Technischer Kongress*, Stuttgart, Germany, 2002
- [18] A. Meroth and B.Tolg, *Infotainmentsysteme im Kraftfahrzeug. Grundlagen, Komponenten, Systeme und Anwendungen* Wiesbaden: Vieweg Verlag,2007.
- [19] D. Zöbel. "Unfallvermeidung durch Rückfahrassistentz". Internet: [http://www.uni-koblenz-landau.de/koblenz/fb4/institute/IST/AGZobel/downloadbereich/press/OptischeAssistenz\\_PM.pdf/view](http://www.uni-koblenz-landau.de/koblenz/fb4/institute/IST/AGZobel/downloadbereich/press/OptischeAssistenz_PM.pdf/view) , Nov. 30, 2006 [Aug. 31, 2011].
- [20] C. Baroudi, *Service Oriented Architecture for Dummies*, New York, NY: Wiley Publishing, 2007, pp 340.
- [21] N. M. Josuttis, *SOA in practice*, Sebastopol, CA: O'Reilly Media, 2007, pp 298.



**Marco Wagner** is currently working as a scientific assistant at the faculty of mechanics and electronics of the Heilbronn University, Heilbronn, Germany where he is teaching in the field of automotive software systems and distributed systems. At the same time, he is enrolled as a PhD student at the faculty of Computer Science of the University of Koblenz-Landau, Koblenz, Germany. He received his Dipl.-Ing. (FH) degree from the Heilbronn University in 2008.

His research is focused on Software and Systems Architectures of dynamically changing Driver Assistance Systems. He also works on Driver Assistance Systems for articulated vehicles as well as on sensor systems for this class of systems.



**Dieter Zöbel** studied Computer Science and Mathematics at the Technical University Kaiserslautern, Germany, where he also received his PhD degree for a dissertation about resource management. Since 1991 he is professor for operating systems and real-time systems at the University Koblenz-Landau, Germany.

His research focus is on the domains of real-time systems as well as on autonomous and assisted driving.

With respect to assisted driving his research group has various projects on back-up assistance for truck and trailer combinations and holds several patents for respective sensor systems and human machine interfaces.



**Ansgar Meroth** is working as a professor at the faculty of mechanics and electronics of the Heilbronn University, Heilbronn, Germany where he is teaching computer science, information technology and systems engineering, mainly in the field automotive systems since 2003. At Heilbronn University he served as vice dean and vice president as well. Meroth received his Dipl.-Ing. in Electrical Engineering 1990 at the Karlsruhe Institute of Technology (KIT), former University of Karlsruhe, and his PhD in 1997 at the same university. From 1997 to 2003 he worked for Robert Bosch GmbH in various functions in systems engineering and organization development.

His research is focused on distributed driver assistance and information systems and human machine interfaces.