

On the Need of Data Management in Automotive Systems

Sandro Schulze,* Mario Pukall, Gunter Saake, Tobias Hoppe, and Jana Dittmann
School of Computer Science
University of Magdeburg, Germany
{sanschul, mario.pukall, saake, tobias.hoppe, jana.dittmann}@iti.cs.uni-magdeburg.de

Abstract: In the last decade, automotive systems changed from traditional mechanical or mechatronical systems towards software intensive systems, because more and more functionality has been implemented by software. Currently, this trend is still ongoing. Due to this increased use of software, more and more data accumulates and thus, has to be handled. Since it was no subject up to now to manage this data with software separately, we think that it is indispensable to establish a data management system in automotive systems. In this paper we point out the necessity of data management, supported by exemplary scenarios, in order to overcome disadvantages of current solutions. Further, we discuss main aspects of data management in automotive systems and how it could be realized with respect to the very special restrictions and requirements within such a system.

1 Introduction

An automotive system encompasses the hardware, i.e., sensors, actuators, and electronical control units (*ECU*), and the several bus systems used for their connection and communication amongst each other in a modern car. Furthermore, the software implemented to fulfill more and more functionality in a car, belongs to such a system. Especially the latter increased rapidly in the last decade, making an automotive system a software-intensive IT system. More precisely, it is estimated that in 2010 approximately 1 GB of software is installed in automotive systems [PBKS07]. This evolution is accompanied by an increasing amount of data [CRTM98]. Additionally, typically mechanical components are substituted by electronical ones, e.g., considering the *X-By-Wire* technology or driver assistance systems like *ESP (Electronic Stability Program)*. All these mentioned aspects lead to a highly complex system and in near future, the complexity is expected to increase further due to new technologies like *Car-To-Car (C2C)* or *Car-To-Infrastructure (C2I)* [ZS07, S. 387]. Hence, it is crucial to guarantee reliability and safety of an automotive system while providing an efficient and flexible management of data.

In fact, the data in such a system is managed ad hoc by each ECU on its own, using internal data structures. Subsequently, problems which may occur (e.g., inconsistencies or concurrency problems) are solved locally, using mechanisms implemented directly on the hardware. This, in turn, not only increases the already high complexity of the overall sys-

*The author is funded by the EU through the EFRE Programme under Contract No. C(2007)5254

tem. Moreover, this approach decreases the flexibility, extensibility and maintainability of the system. Since a change of the overall system configuration (e.g., new functionality resulting in new or altered data) probably entails necessary changes of one or more hardware implementations, this finally leads to increasing development costs. Thus, we conclude that a *data management system (DMS)* is inevitable to ensure the flexibility needed in automotive systems and beyond it, to decrease the development costs. Furthermore, such a DMS may be useful to ensure the reliability and safety of the system, its users and the respective environment.

In this paper, we introduce the idea of establishing a DMS in automotive systems and thus, to increase the already mentioned properties, e.g., efficiency, flexibility, or maintainability. Hence, we will discuss several motivating aspects for clarifying potential advantages of DMS in automotive systems. Furthermore, we examine the usefulness of integrating security mechanisms within such a DMS in order to address safety and reliability as well. Finally, we discuss how to address the particular conditions of an automotive system, regarding selected data management aspects.

2 Background

In the following we give a brief overview on automotive systems and the IT security in such systems.

2.1 Automotive Systems

An automotive system is a complex networked IT system, which is characterized by a frequent interaction of its components, in detail, dozens of ECUs, sensors and actuators. Each component can be seen as a self-contained embedded system, which leads to a highly heterogeneous character of the overall system. The heterogeneity is tightened, since usually different ECUs are provided by different manufacturers. The communication between these components takes place via bus systems, primarily over CAN, but also LIN or MoST are used [ZS07, S. 36 ff.]. An exemplary part of such an automotive system is depicted in Figure 1.

The different sensors (S_i) and actuators (A_i) are directly connected with the ECUs, which in turn, are connected via bus systems for communication. Furthermore, the ECUs are grouped into subbus systems according to their functionality. In our example, three subbus systems are depicted, namely *Comfort*, *Infotainment* and *Power Train* subbus system. The subbus systems differ slightly regarding conditions and constraints to be met. For instance, the power train system has hard real time constraints and thus, the transmission rate is higher than in other subbus systems. Apart from that, data can be exchanged between any arbitrary ECUs, regardless what subbus system they belong to. Furthermore, the exchanged data can be discrete as well as continuous, whereas the differences are due to the distribution of the data. The bus protocol distributes the continuous data in a time-triggered

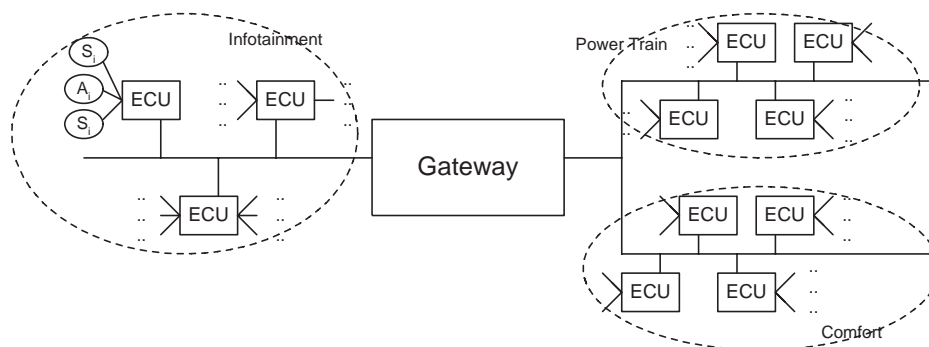


Figure 1: Exemplary Part of an Automotive System

way, i.e., the data is only valid for a certain time, while the discrete data is distributed in an event-triggered way (i.e., if new data is available, the ECUs are notified of this fact).

Usually, a micro controller commonly used in ECUs of current automotive systems (e.g., in off-the-shelf cars) has only a memory of 40-50 KB on average (distributed over RAM and EEPROM) and the computing power is less than 10 MHz. Additionally, the real time requirements within such a system are in dimensions of some milliseconds (with an upper limit of 10 ms), no matter which subbus system is considered. Thus, software development (in our case a DMS) is a challenging task for automotive systems.

2.2 IT Security in Automotive Systems

The IT security in automotive systems was neglected for a long time, but becomes more and more a focus of research due to the potential devastating consequences when it is violated. Per definition, security means reliability in terms of preserving security aspects of information (within a system) [HKLD07]. If all requirements are met to ensure these security aspects, namely *confidentiality*, *integrity*, *availability*, *non-repudiability* and *authenticity*, the system can be considered to be secure. In order to achieve a secure system, appropriate security mechanisms must be applied to the system.

An automotive system exhibits multiple opportunities for access from *outside* (e.g., by exploiting security vulnerabilities in wireless communication systems or by encompassing the use of manipulated media discs) or *inside* the car (e.g., directly by injected malicious code or additional devices which are physically attached either to explicit communication interfaces or implicitly to hooked-up bus wires). Thus, it is prone to malicious attacks, whereas the reasons can be diversified, e.g., tuning purposes or monetary interests. Examples for successful attacks can be found in [BD07, HD07, Paa08, HKD08b]. Although these attacks differ in the underlying approach as well as the target, they all have in common, that they aim at the manipulation of data. Subsequently, the mentioned security aspects have to be ensured for the *data* in automotive systems to keep the overall system secure. This data-centered character, and the fact, that violated security aspects can have

implications for the safety and reliability, make it worth to think about integrating security mechanisms into a data management system.

3 Exemplary Scenarios for Data Management in Automotive Systems

In this section we introduce exemplary scenarios for automotive systems, where a data management system is useful or even inevitable.

3.1 Uniform Data Structures

In current automotive systems, the data management is distributed as a hardware solution over the participating ECUs. Thus, each ECU is only responsible for the local data and its quality (e.g., consistency or availability) using internal data structures. This approach not only increases the already high complexity and heterogeneity of such systems, moreover, it leads to difficulties regarding the verification of the overall system. For instance, if data is distributed over several ECUs, each of them has to validate that the data is also the most recent one. Furthermore, the current decentralized hardware solution increases the I/O operations. Thus, a uniform data structure throughout the system, as provided by a data management system, is desirable, regarding an efficient data management. At first glance, this might be questionable because the current hardware solution offers a better performance (considering a single ECU) compared to a software solution like a DMS. However, the advantages of uniform data structures as provided by a DMS overcome the disadvantage of decreased performance. For instance, with uniform data structures, the data in an automotive system can be captured global, which facilitates validation (of data) or verification of the system. Furthermore, due to the structure, efficient data access can be achieved (e.g., using indexes) and thus, the performance can be increased. Moreover, uniform internal structures allows for uniform external representations of data as well.

3.2 Concurrency Control

Another reason for a DMS is the concurrency control, which ensures the integrity of the data and thus, the reliability of the system. In current automotive system the protocol of the bus system, mostly CAN, uses a kind of prioritization to control the (write) access rights for the bus, known as *bus arbitration* [ZS07, S.23 ff.]. In detail, an ECU needs the highest priority for a certain message (identified by a message ID) to put the data, contained in this message, on the bus. Additionally, on each ECU certain (local) mechanisms are implemented to ensure that the current data is the most recent one. In Figure 2 a simplified concurrency scenario is depicted for only two ECUs. The data x , available on the field bus, is read by the ECU_A at time t_s . After manipulating the data, the ECU_A writes the data x' at time t_{s+n} , so that x' is now the valid data. At the same time (or even a subsecond

before) another ECU (ECU_B) reads the "old" data x for further usage, e.g., computation of other data. If ECU_B does not notice that the data x is out of date, the further computed data is not valid and thus, may endanger the reliability of the system.

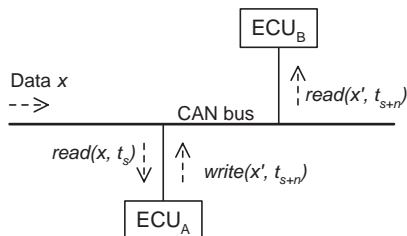


Figure 2: Example for Concurrent Access of Data

However, with the expected increase of complexity, due to additional electrical components or software, this approach is not appropriate anymore. Even today, it is difficult for a single ECU to ensure that the current data is up to date with its hard-coded mechanisms, since the data is scattered throughout the system at more than two ECUs, as supposed in our example.

Hence, a DMS may be useful to overcome this crucial situation by supporting a global view on the data. In the sense of concurrency control, *transaction management (TXM)* functionality could be used, to implement and coordinate simultaneous data access in automotive systems. By using respective strategies for assigning write access to ECUs, the efficiency and the reliability could be increased compared to the approach currently used.

3.3 Access Management

As already denoted in Section 2.2, a third person (e.g., an attacker) can participate in an automotive system by hooking up a device with the bus wire. Subsequently, the device can "listen" to the data, i.e., an unauthorized access has been established. Additionally, this external device can provide manipulated data as well as demand for certain data of another ECU, which is a serious problem regarding safety and reliability of the system. Since no solutions preventing such intrusion exist, *access management* functionality of a DMS could be useful to tackle this problem. If such a management knows of all participating ECUs (or at least the respective ones for a certain data), it can control the read and write access to the data (or the message it is contained in) and preserve the system from contamination through falsified data. Additionally, security mechanisms, e.g., digital signatures or public key infrastructures, could be integrated to hinder the access to the data.

Of course, the current technology (e.g., CAN) does not support this approach. But future communication technologies, e.g., C2C, C2I or any other wireless communication, provide the respective opportunities to establish such functionality. Thus, access control should be considered seriously as an further advantage of data management in automotive systems.

3.4 Trip Recorder for Insurance Purposes

This scenario points out which additional facilities a data management system can provide in automotive systems. It covers a common situation in the day-to-day transportation - an

accident. In the case that there are no witnesses involved in such an accident, it is difficult to reproduce how the incident occurred. Especially insurances are interested in the event reconstruction, even if this is only possible in parts. Thus, the information given by certain data could be employed for looking back at what happened exactly. Since not all data can contribute to the clarification, only those with a high degree of information should be considered. For instance, the data containing the speed or the steering angle provide useful information. Furthermore, the current state of the driver assistance systems (if present) or flags, if the light or the indicator signal was activated, are conceivable. However, if an accident occurs, an event has to be activated inducing the DMS to write the respective data into a protected partition of the persistent storage. After a further preparation of the data (e.g., information retrieval), it should be accessible to reproduce the accident.

4 Aspects of Data Management in Automotive Systems

In this section, we shed some light on implementation or design aspects of an automotive DMS. We identified three main aspects which have to be considered with care and we discuss each of them in the following.

4.1 Central vs. Distributed vs. Hybrid DMS

The first and maybe most important aspect, is the question where to establish the DMS within an automotive system physically. Three approaches are possible, namely *central*, *distributed* or *hybrid* DMS and we discuss their pros and cons in the following.

Central DMS. A central DMS means to implement the whole software system on a single ECU within the automotive system. Since the DMS has to communicate with all subbus systems and their respective ECUs, the central gateway is the only place where the central DMS can be enclosed. The major advantages of such a system are increased maintainability and extensibility. Updates or extensions (e.g., of functionality) only have to be provided for one piece of software which is situated at only one piece of hardware. Furthermore, strategies for concurrency control or other mechanisms can be handled internally since no other DMS exists in the system and thus, optimization should be easier. Nevertheless, such a central solution comes along with significant disadvantages as well. Firstly, it is always a bottleneck regarding availability or reliability. If the data traffic is very high, the DMS may slow down and thus, propagation of data is delayed or even aborted. Moreover, if the gateway exhibits a failure or is not accessible for one or more subbus systems, this may have devastating consequences for the whole automotive systems since data is not available or out-of-date. Subsequently, the efficiency and reliability of the automotive system is decreased by such a solution. Secondly, since an ECU (even a gateway) has limited resources, a central DMS would not cope with the requirements of an automotive system (e.g., hard real time, computing power) for processing the data. We conclude, that a central DMS may be not appropriate at all for an automotive system.

Distributed DMS. Since this approach proposes to implement only one DMS for the whole automotive system as well, it differs from the central DMS by the fact, that the software components of the DMS are (physically) distributed over the whole system. This can be done by using the *AUTOSAR* (*AUTomotive Open System ARchitecture*, www.autosar.org) standardization, which has been established for automotive systems. This standard proposes a runtime environment and standardized interfaces, so that software components can be distributed over different hardware units [H⁺06]. With this approach we can overcome some of the disadvantages of the central DMS approach. First, distributing the DMS over all ECUs tackles the bottleneck problem which occurs in case that an ECU fails. Furthermore, some software components could be replicated on different ECUs (i.e., different from the ECU which contains the original component) and in case of failure, the replicated component can substitute the original one. Second, due to the distribution, we are not limited to the resources of only one ECU, which is also a problem of the central DMS. Beside these improvements (regarding the central DMS approach), the distributed DMS exhibits the already mentioned advantages like maintainability or extensibility as well. But in spite of all these advantages, even this approach comes along with some critical disadvantages. Because of the distribution of software components, the communication between the ECUs increases. This, in turn, leads to higher data traffic which could be critical at a certain point. Furthermore, the coordination between the particular components is quite complex, which may be enforced in the case that one or more components (or their corresponding ECU respectively) fail. In summary, the distributed approach is yet more appropriate than the central DMS approach, although there is still room for optimizations.

Hybrid DMS. The last approach we want to introduce here is some kind of mixture of the previous two approaches. The idea is to establish one DMS for every subbus system. By this separation, we can adjust the several DMSs to the requirements, functionality and data traffic of the particular subbus system and thus, optimize the load balancing of the overall system. Within such a subbus system, the DMS can be distributed over an arbitrary number of ECUs and thus, benefit from the advantages of the distributed approach. Nevertheless, a single DMS has still a global view on the data of the whole automotive system regarding concurrency control or similar. For instance, if a certain piece of data is locked by the power train DMS, the DMS of another subbus system cannot write this data until it is unlocked. We think, that the hybrid approach can overcome all aforementioned disadvantages and thus, is the most appropriate solution for automotive systems.

4.2 Tailoring and Reusing Functionality

Once the decision for a certain kind of data management has been made, new problems regarding the implementation arise. As already mentioned, current solutions suffer from high development costs or poor maintainability. Furthermore, we have to consider the crucial requirements and constraints of an automotive system, e.g., limited memory, computing power or hard real-time requirements. To overcome these problems, the concepts of tailoring (software) and reusing (functionality) provide efficient solutions.

In the context of this paper, reusing functionality means that a DMS or parts of it can be reused instead of developing it from scratch. For instance, if we implement the hybrid DMS (cf. Section 4.1), it would be cumbersome and costly to develop the DMS for every particular subbus system from scratch. Rather, it would be eligible to develop one DMS, which could be adjusted to the requirements of the several subbus systems and thus, as much functionality as possible could be reused for every DMS. This not only leads to a noticeable decrease of development costs, but also improves the maintainability since changes to the DMS can be made at one central point (the original DMS). In addition, tailoring the DMS, i.e., the software contains only the functionality needed, can help to overcome the system requirements and constraints. For instance, if the DMS of the infotainment subbus system does not need SQL or index structures, this functionality is removed from the DMS. As a result, the DMS requires less system resources (e.g., memory, power consumption) and even the communication effort may be decreased. Further information about techniques and concepts for tailoring and reusing software can be found, e.g., in [K⁺90, CE00, CN06].

4.3 Integrating IT Security Mechanisms

The third aspect we consider, covers the goals of increased safety and reliability in automotive systems. A central point for achieving these goals is to ensure the security of the systems (cf. Section 2.2). Hence, it is useful to integrate respective security mechanisms in the data management, because this is where data are handled first, before sending or after receiving. In Section 3.3, we already denoted how security can be integrated in the access management in order to ensure the integrity of the data. However, for achieving the goals, further DMS components have to be enhanced with respective security mechanisms, so that a holistic protection can be provided. Thus, appropriate components have to be identified and security mechanisms have to be adjusted so that they fit the needs of the software as well as the requirements of the underlying automotive system. As a result, we envision a kind of security layer within an automotive data management system.

5 Related Work

A lot of work has been done in the several fields, unified in this work. In the following we confine ourself to mention only those which are highly related to this paper.

Automotive IT security: In the automotive domain, holistic concepts for IT security are a very young research topic. While such concepts are absent so far, common IT security mechanisms for protecting single components, based on encryption or digital signatures, can already be found in today's cars. Some examples are central locking, keyless entry and immobiliser systems [LSS06], but also memory contents like firmware updates are protected against unauthorised manipulations. Because such existing mechanisms are not conceived to provide a holistic protection for the entire system, in the recent past research

about holistic concepts has begun. For example, the application of Trusted Computing technology in future automotive IT systems is increasingly investigated (see [BEWW07]). There are also approaches to employ PKI infrastructure and certification of automotive components to verify aspects like integrity and authenticity at every start of the car [BZ08]. In previous work, also the extension of such future automotive IT security concepts by Intrusion Detection approaches [HKD08a] has been discussed.

Data management in embedded systems: Tailor-made data management for embedded systems has been widely proposed in the recent years. First, there exist several approaches, dealing with whole DMS as well as with certain parts of it, e.g., the storage manager, with respect to embedded systems in general [LAS05, SRS⁺07, R⁺08]. Amongst others, they focus on motivations for tailoring a DMS and new concepts, techniques or paradigms for achieving the goal with a minimized effort. Second, some work on data management for automotive systems has been done as well. For instance, Nyström et. al. discuss a component-based approach for an efficient data management as well as general data management issues in automotive (sub)systems [NTN⁺02, NTN^H04]. However, neither the former nor the latter addresses a holistic approach for data management in automotive systems, or even integrate IT security as important (non-functional) property.

6 Conclusions

Due to the high complexity in automotive systems, an efficient management of data is inevitable. In this paper, we suggested to establish a DMS for such systems to overcome the problems of current solutions. We presented different scenarios to point out the necessity of a DMS and beyond it, we proposed the idea of integrating IT security in such a DMS. Finally, we discussed three core aspects, which are crucial for the success of an automotive DMS and thus, have to be considered carefully during design and implementation.

References

- [BD07] A. Barisani and B. Daniele. Unusual Car Navigation Tricks: Injecting RDS-TMC Traffic Information Signals. In *Proc. of the CanSecWest Conf.*, 2007.
- [BEWW07] A. Bogdanov, T. Eisenbarth, M. Wolf, and T. Wollinger. Trusted Computing for Automotive Systems. In *Proc. of the VDI/VW Gemeinschaftstagung Automotive Security*, pages 227–237, 2007.
- [BZ08] D. Borchers and P.-M. Ziegler. Mit PKI gegen den Autoklau. Heise Newsticker, 2008. <http://www.heise.de/newsticker/meldung/104593>.
- [CE00] K. Czarnecki and U.W. Eisenecker. *Generative Programming: Methods, Tools, and Applications*. ACM Press/Addison-Wesley, 2000.
- [CN06] P. Clements and L. Northrop. *Software Product Lines: Practices and Patterns*. Addison Wesley, 2006.
- [CRTM98] L. Casparsson, A. Rajnak, K. Tindell, and P. Malmberg. Volcano - a Revolution in On-Board Communications. Technical report, Volvo Technologies, 1998.

- [H⁺06] Harald Heinecke et al. Achievements and exploitation of the AUTOSAR development partnership, 2006. http://www.autosar.org/download/AUTOSAR_Paper_Convergence_2006.pdf.
- [HD07] T. Hoppe and J. Dittmann. Sniffing/Replay Attacks on CAN Buses: A Simulated Attack on the Electric Window Lift Classified using an adapted CERT Taxonomy. In *Proc. of the Workshop on Embedded Systems Security (WESS) at EMSOFT 2007*, 2007.
- [HKD08a] T. Hoppe, S. Kiltz, and J. Dittmann. IDS als zukünftige Ergänzung automotiver IT-Sicherheit. In *Proc. of DACH Security*, 2008.
- [HKD08b] T. Hoppe, S. Kiltz, and J. Dittmann. Security threats to automotive CAN networks practical examples and selected short-term countermeasures. In *Proc. of the Int'l Conf. on Computer Safety, Reliability and Security (SAFECOMP)*, pages 235–248, 2008.
- [HKLD07] T. Hoppe, S. Kiltz, A. Lang, and J. Dittmann. Exemplary Automotive Attack Scenarios: Trojan horses for Electronic Throttle Control System (ETC) and replay attacks on the power window system. In *Proc. of the 23. VDI/VW Gemeinschaftstagung Automotive Security*, pages 165–183, 2007.
- [K⁺90] K.C. Kang et al. Feature-Oriented Domain Analysis (FODA) Feasibility Study. Technical Report CMU/SEI-90-TR-21, Software Engineering Institute, Carnegie Mellon University, 1990.
- [LAS05] T. Leich, S. Apel, and G. Saake. Using Step-Wise Refinement to Build a Flexible Lightweight Storage Manager. In *Proc. of the East-European Conf. on Advances in Databases and Information Systems (ADBIS)*, 2005.
- [LSS06] Kerstin Lemke, Ahmad-Reza Sadeghi, and Christian Stübke. Anti-theft Protection: Electronic Immobilizers. In *Embedded Security in Cars*, 2006.
- [NTN⁺02] D. Nyström, A. Tesanovic, C. Norström, J. Hansson, and N.-E. Bankestad. Data Management Issues in Vehicle Control Systems: A Case Study. In *Proc. of Euromicro Conf. on Real-Time Systems*, pages 249–256, 2002.
- [NTNH04] D. Nyström, A. Tesanovic, C. Norström, and J. Hansson. COMET: A Component-Based Real-Time Database for Automotive Systems. In *Proc. of the Workshop on Soft. Eng. for Automotive Systems at the ICSE*, 2004.
- [Paa08] Cristof Paar. Remote keyless entry system for cars and buildings is hacked, 2008. http://www.crypto.rub.de/imperia/md/content/projects/keeloq/keeloq_en.pdf.
- [PBKS07] A. Pretschner, M. Broy, I.H. Krüger, and Th. Stauner. Software Engineering for Automotive Systems: A Roadmap. In *Proc. of Future of Soft. Eng. (FOSE) at the ICSE*, pages 55–71, 2007.
- [R⁺08] M. Rosenmüller et al. FAME-DBMS: Tailor-made Data Management Solutions for Embedded Systems. In *Proc. of the Workshop on Soft. Eng. for Tailor-made Data Management (SETMDM)*, 2008.
- [SRS⁺07] G. Saake, M. Rosenmüller, N. Siegmund, C. Kästner, and Thomas Leich. Downsizing Data Management for Embedded Systems. In *Keynote of the Int'l Conf. on Information Technology*, November 2007.
- [ZS07] W. Zimmermann and R. Schmidgall. *Bussysteme in der Fahrzeugtechnik*. 3.Auflage. Vieweg + Teubner, 2007.