Complying with Law for RE in the Automotive Domain

Birgit Penzenstadler
penzenst@in.tum.de
Technische Universität München
Fakultät für Informatik
Software & Systems Engineering
85748 Garching, Germany

Jörg Leuser
joerg.leuser@daimler.com
Daimler AG
Group Research and Advanced Engineering
89081 Ulm, Germany

Abstract

The automotive industry is concerned with developing large and complex embedded systems. The original equipment manufacturers (OEMs) are responsible for the safety of their systems, enforced by law in terms of liability. At the same time, there is a number of laws, for example the Automobile Safety Act, that has to be obeyed by the specification.

We give insights into the current state of practice about how the automotive industry performs requirements engineering in order to comply with government laws and regulations. We analyse the challenges, and give ideas from research work in progress for tackling them.[1]

1 Motivation

The specification and development of automotive software involves the responsibility for the safety of the future users, i.e. the driver of the car and the passengers. Some of the responsibility is enforced by law, for example in terms of liability. In the first place, the original equipment manufacturer (OEM) is liable for his product, but guards against taking the whole responsibility for his suppliers' products by specifying in very exact terms within the subcontracts the guarantees that the suppliers have to provide.

Apart from liability, there are a number of laws for the automotive domain that have to be obeyed, the important ones are the automobile safety laws of the different countries. The OEMs need to make sure that their specifications comply to those laws and keep complying to them over time. The current state of practice comprises a long list of referenced documents within specification documents and different kinds of requirements from different stakeholders that have to be handled in compliance with the law. The challenge is to find an adequate solution to specify legal requirements and link to the source documents without requiring too much maintenance effort due to evolution of specifications and laws.

Related Work We have found three categories of related work: One is concerned with software engineering and law focused on risk management and the question of litigation [6, 7]. Another one is concerned with knowledge management for law enforcement in the legal domain [5, 11]. The third one is research on support for extracting rights and obligations from regulations [2].

None of them have yet addressed how to support compliance with law during requirements engineering in the automotive domain.

Contribution We analyse the challenges of complying to the law from an automotive systems development point of view and propose some simple and cost-efficient solutions to handle them.

Outline Sec. 2 gives insights into the state of practice. Sec. 3 sketches the ideas for solving the described problems. Sec. 4 wraps up and gives an outlook on future work.

2 State of Practice

Automotive software systems belong to the embedded systems domain. Many of them are safety-critical and at the same time automotive software is increasing in size and complexity.

[1] This work was partially funded by the German Federal Ministry of Education and Research (BMBF) in the framework of the REMsES project.

The specification for the development of the software for a single car is usually carried out in several documents, the so-called component specifications. The OEMs then assign the development of components to suppliers. Therefore they are confronted with the challenge of breaking down the legal requirements to fit the granularity of the component requirements specifications.

The example used to illustrate state of practice and solution ideas is the "dynamic window colouring" (DWC) system, a fictitious driver assistance system that enhances the vision of the driver by shading the car windows when the light ratios or environmental circumstances change.

2.1 Relevant Laws

As cars are developed in world-spanning product lines, laws of many different countries apply when specifying systems. Representative examples for relevant laws in the automotive domain are:

- Automobile Safety Act [3]
- Product Liability Law [8]
- Electromagnetic Compatibility [4]

An example from the DWC illustrates the demand for integration of different sources. The component specification contains a requirement concerning translucency which is derived from the standardized prescriptions for the approval of safety glass [12] that refines §22a of [3]:

> *The controller may not darken the front window below a translucency of 75% (see ECE R 43) [12]. For the Japanese version, the translucency may not be reduced below 70% (see MTO No. 67)* [9]

The relevant laws are extensive but often only a small percentage of their content is relevant for a specification. In our example the only part of the referenced normative constraint that is directly relevant to this requirement is Sec. 9.1.4 of Annex 3. It states how the translucency has to be measured and that the front window might not have a translucency below 75%. One of the reasons for the more general references is that to-the-spot references might change during updates of a normative constraint and require an update of the reference, even if the referenced content did not change.

Other laws, e.g. the product liability laws, have also indirect influence on the requirements. They require products to be developed using "state of the art" methods, for example standards from the ISO, IEEE or DIN (German Industry Standard). Although those standards are no laws, they have to be treated like laws in the specification process. The same is true for company standards. Therefore we will refer to laws and other documents that have to be treated like laws as normative constraints in the following.

> Def: *Normative Constraints are laws and their supplements, prescriptions, and standards, which are all treated equally during the specification process.*

Practitioners have found pragmatic ways in dealing with such normative constraints. This is mainly based on including references within the structure of their specifications (see Sec. 2.2). At the same time, this raises the need for traceability over time which means keeping the references and their content up to date (see Sec. 2.3).

2.2 Referencing normative constraints from specifications

Generally, the component specifications are documented using a document structure similar to the structure defined by the VDA template for component specifications [13]. One section of this structure is dedicated to referenced documents. The content of this section is in the simplest form only a list of references. As already indicated before, this list does not only contain laws but also other documents: for example the Automotive SPICE [1] as a domain specific form of ISO/IEC 15504, and the DIN 75220 [10]. Furthermore, there are also internal standards which might for example be a (translated) collection of laws or guidelines for development. These documents are generally referenced from within the specification by textual links using unique identifiers for each normative constraint.

The requirements of the final specification are often a collection of requirements written by different developers, e.g. area specialists, or groups of developers, e.g. a group dealing with standardized global requirements that are to be included in all specifications. Finally, there are requirements that are only part of one single component specification which are written by the team responsible for this specification. Each of these requirements may reference normative constraints and, more demanding, different parts of a specification may even reference different versions of such a document. A reason for such a conflict is that different components of a car are developed at different times and therefore

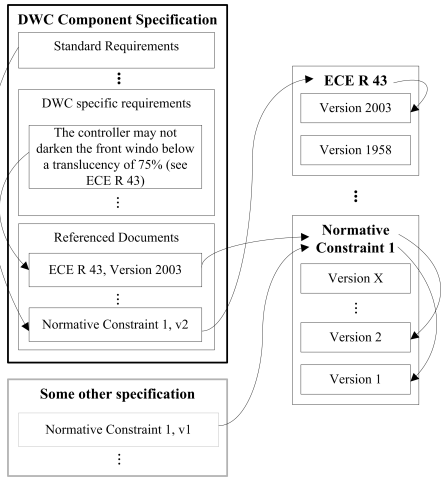

Figure 1. Tracing normative constraints in specifications

different versions of the same constraint were applicable. This can cause integration problems if other parts of the specification have to use a different version.

2.3 Current Challenges and Handling

As mentioned previously, automotive specifications merge content from different sources into a single artifact. As the different sources have quite specialized knowledge, it is not to be expected that the team generating the final component specification has extensive knowledge about all aspects covered by the different parts of the specification. This is also true for the relevant normative constraints. The handling of this distributed knowledge comprises the following challenges:

C1: Documentation of the relevant normative constraints and their derivations. One way of dealing with the large number of normative constraints is the use of a document management system. While keeping everything in one system would be preferable, currently multiple systems are in use which still do not contain all normative constraints. For documents in the systems, users get notified of changes. For multiply referenced documents like laws, the exact changes are even highlighted. But it is not possible yet to automatically point out the necessary changes to requirements.

C2: Traceability and retrieval of the relevant contents for delivery to suppliers. At the end of the specification process exists a list of referenced documents that the supplier needs. Documents that are not publicly available have to be shipped to the supplier in any case. Not publicly available documents are for example translated versions of foreign laws or a collection thereof. Shipping publicly available documents might seem unnecessary, as the supplier could get hold of them easily, but it fortifies their relevance. For contract reasons it is best practice to deliver all referenced documents on all accounts to be sure everyone is working with the exact same documents. However, as a safeguard, specifications of OEMs often contain parts giving the responsibility of an up-to-date- and compliance-check of the normative constraints to the supplier.

While the problem described so far was rather a legal one, a practical challenge is the collection of the different documents from all the repositories that they reside in. Although a standard set of documents might be readily available in a folder for transfer, the more specialized documents are often kept by the specialist that references them in "his" requirements. As the number of referenced documents easily reaches three figures it is a tedious job collecting all referenced documents in the correct version for transfer.

C3: Change management and evolution for keeping the documents, history and references up to date. Keeping the already referenced normative constraints up to date is a challenge as they are referenced by many specifications which often contain derived requirements. The approach taken in practice is a divide-and-conquer approach. Each developer is required to keep track of changes of relevant normative constraints concerning his scope. This also includes anticipating probable future changes. In practice, many specifications or parts thereof are evolutionary reused for multiple revisions of a car. Therefore changes in normative constraints also have to be reflected in "old" specifications. Instead of updating those immediately after a normative constraint changes, changes to requirements are possibly postponed until the specification is needed again. In case a normative constraint changes multiple times while the specification is not currently needed, on one hand this saves impact analyses, but on the other hand this approach poses the threat of forgetting the needed changes. As seen in Fig. 1, the job of keeping normative constraints and specifications synchronized is complicated by the fact that references to normative constrains frequently point to the whole normative constraint instead of the relevant spots within. This leads to expensive impact analyses.

3 Ideas for Solutions

How can we tackle the current problems? Practitioners want simple solutions that work with little effort. In this section we sketch a few ideas into that direction.

3.1 Content Management

General solutions from the research domain of knowledge management encompass various types of document management systems and information systems, but most of them are too extensive and cost-intensive to be helpful for our specific needs. Therefore we sketch four small-scale solutions that require small effort in set-up and maintenance but at the same time provide good support including bidirectional traceability and versioning.

1. List Currently, bundling legislative constraints in form of a simple list is the most obvious solution to summarize the necessary information as concise as possible. A simple ID from a database or a URL provides cheap traceability links. This solves C1 and C3 in a minimalistic way - not the best, but the cheapest. The drawback is that these lists and the corresponding references have to be maintained by hand and the search for relevant documents (C2) is also performed individually. This applies for all our proposed solutions, though at different costs, but when using lists the searching part is especially expensive.

2. Database Another economic solution is a bibliography or database for all relevant laws and standards (C1) that is updated regularly. It is desirable that such a reference includes traceability links not only from specification to law, but also in the other direction to enable straight forward evolution of the corresponding specifications automatically after a certain law or standard has changed (C3). On the other hand this can also be done when evolving from one specification release to the next version if there is no explicit need to update them in between. This depends on how fast the changed laws or standards require to be complied to, because usually there is a certain deadline from when on they have to be obeyed.

The traceability links provide easy access to the relevant documents for delivery to the subcontractor. Such a database could be set up to be available for the suppliers with partial access rights, thereby solving C2.

3. Document Management System A different solution with respect to availability would be a DMS, which solves C1 by storing the original source documents. In fact, this is a special form of the database solution. It is available for both the OEM and the suppliers, where the latter receives only partial access rights (C2). This way, a supplier can look up the normative constraints whenever he needs to without the need of big addendums to the specification.

Additionally, the OEM can update the normative constraints any time (C3) and automatically notify the supplier of the changes. The advantage in contrast to the database is that the solution is cost-efficient, because instead of transferring the information to a different system, the source documents are stored in their original form.

4. Wiki A rather informal way is capturing the information in a wiki. From the organizational point of view, rigorous access rights management is crucial for this solution. With respect to the confidentiality of the component specifications there should not occur any problems as the separated information about laws and standards without the corresponding specification is uncritical and may be accessible for every company member.

This effectively solves C1 and C3. As for C2, it is not desirable to give subcontractors access to the internal wiki of the OEM. Therefore, a mechanism has to be established to extract the relevant information for delivery to the subcontractor.

The advantage of the Wiki solution is its adequacy and support for geographically distributed cooperation.

Cost-efficiency There may be other solutions but the crucial point is the trade off between project complexity and costs. The solutions presented here require a small up-front investment compared to the easier handling of normative constraints. Additionally, they can be integrated with some common requirements management tools.

3.2 Process Support

As specifications in the automotive domain are so complex and extensive that no developer is familiar with *all* the relevant normative constraints at the same time or even over time (see Sec. 2.3), there is need for support via processes in at least two parts: system development and change management.

Development Process The lifecycle of the development process should include repeated checks of the

normative constraints, at least at three stages of each iteration of the development cycle:

1. During project setup, wherever possible, committing groups should get an understanding of which normative constraints will be used and therefore prevent conflicts.

2. Before the different specification parts are merged into one artifact, each group has to do a check of the referenced normative constraints.

3. Before the tender is started, a check for conflicts with normative constraints is needed.

Change Management As every development process also requires change management, there has to be defined process support for this task as well, either periodically, or triggered by notifications from the legal side, or triggered through defined checkpoints within the system development lifecycle.

One example for such a trigger with regard to the last category could be the decision to reuse a specification. The change management consists of three major steps:

1. Identification of changes in normative constraints,

2. an impact analysis on the requirements of affected specifications,

3. and finally the realization of the required changes to the specifications.

Cost-efficiency The process support requires the same trade off between system complexity and costs as the content management solution.

4 Conclusion

Ensuring compliance to normative constraints in the automotive domain is a challenge due to extensive specifications and many normative constraints. We have detailed the state of practice and sketched solutions with low costs, but there is still a lot of room for improvement that should be discussed during the workshop.

A next step will be a list of requirements for the documentation and process support for normative constraints and compliance. Another aspect is the mapping of normative constraints to types of requirements to ease evolution in the long run.

References

[1] Automotive SIG. Automotive SPICE, 2007.

[2] T. D. Breaux, M. W. Vail, and A. I. Anton. Towards regulatory compliance: Extracting rights and obligations to align requirements with regulations. In *RE'06: Proceedings of the 14th IEEE International Requirements Engineering Conference (RE'06)*, pages 49–58, Washington, DC, USA, September 2006. IEEE Society Press.

[3] Bundesregierung. Straßenverkehrs-Zulassungs-Ordnung. Bundesgesetzblatt Teil I, May 2008.

[4] Bundesrepublik Deutschland. EMV Gesetz. Bundesgesetzblatt, 2008.

[5] H. Chen, J. Schroeder, R. V. Hauck, L. Ridgeway, H. Atabakhsh, H. Gupta, C. Boarman, K. Rasmussen, and A. W. Clements. COPLINK Connect: information and knowledge management for law enforcement. *Decision Support Systems*, 34:271–285, 2003.

[6] J. Cosgrove. Software engineering and the law. *Software, IEEE*, 18(3):14–16, May 2001.

[7] A. Fuller and P. Croll. Why Don't We Teach Software Engineers about the Law. *Journal of Law and Information Science*, Vol 12:pp 115–118, 2001.

[8] Kullmann. *ProdHaftG*. Erich Schmidt Verlag, 5 edition, 2006.

[9] Ministry of Land, Infrastructure and Transport (Japan). Ministry of transport ordinance no. 67, 1951.

[10] Normenausschuss Automobiltechnik. *DIN 75220: Ageing of automotive components in solar simulation units*. DIN, 1992.

[11] A. Terrett. Knowledge management and the law firm. *Journal of Knowledge Management*, 2:67 – 76, 1998.

[12] United Nations Economic Comission for Europe. ECE R43, Revision 2, July 2003.

[13] VDA Verband der Automobilindustrie. Automotive VDA Standardstruktur Komponentenlastenheft, 2007.