A survey regarding the central role of the case base for maintenance in case–based reasoning

Ioannis Iglezakis[1] and Thomas Roth–Berghofer[2]

[1] DaimlerChrysler AG, Research & Technology, FT3/AD,
P.O. Box 2360, 89013 Ulm, Germany
ioannis.iglezakis@daimlerchrysler.com

[2] tec:inno GmbH,
Sauerwiesen 2, 67661 Kaiserslautern, Germany
thomas.roth-berghofer@tecinno.com

Abstract. Knowledge containers represent the structural elements of a case–based reasoning system, namely the *vocabulary*, the *similarity measures*, the solution transformations (*adaptation knowledge*), and the *case base*. Every knowledge container can be maintained separately, however, recent work in the area of case–based reasoning system maintenance concentrates on maintaining the case base. Based on this, four hypotheses are formulated: there is no maintenance without regarding the case base, cases are natural crystallization points for the knowledge in case–based reasoning systems, the case base triggers maintenance operations, and the case base contains knowledge for preventive maintenance. Furthermore, these hypotheses will be considered by measures, methods, and a framework for maintenance of case–based reasoning systems.

1 Introduction

The notion of knowledge containers [10] became the standard paradigm for the representation of structural elements in case–based reasoning (CBR) systems. A knowledge container is a collection of knowledge which is relevant to many tasks rather than to one subtask and each knowledge container gives structure to the knowledge. The knowledge containers are the *vocabulary* used, the *similarity measures*, the solution transformations (*adaptation*), and the *case base*.

Each knowledge container is able to hold almost all knowledge. For example, the similarity measure can be reduced to a simple comparison, if a reasonable number of cases are collected. If all possible cases are collected (which is of course an academic case) a pure interpreter approach can be built. Another example is that with a more sophisticated adaptation knowledge the number of cases can be reduced. In addition to this a better similarity measure can reduce the effort of finding a more sophisticated adaptation.

Because of performance reasons there is a compile step for knowledge containers. The knowledge for the vocabulary, similarity, and adaptation is compiled while the knowledge in the cases is used at run time. The compile time refers to the time after knowledge acquisition and before the actual problem solving. Furthermore, this means that these containers stay static and every change will need a new compilation. In contrast the acquisition of cases requires little or no additional effort if the cases are already present. Instead of compilation the cases are interpreted at run time and every change can be made immediately.

The compilation and interpretation of the knowledge containers affects their maintainability. Each container has its own maintenance approach and therefore can be described separately. Furthermore the increasing integration of CBR techniques in business processes shows that maintenance is required to keep the system in satisfactory working condition. By using knowledge containers maintenance can be applied separately on every container. However, looking at the recent work in the area of CBR system maintenance, there is one major stream of research — the concentration on maintaining the case base.

The purpose of this paper is first, to explain this concentration based on four hypotheses:

1. There is no maintenance without regarding the case base.
2. Cases are natural crystallization points for the knowledge in CBR Systems.
3. The case base triggers maintenance operations.
4. The case base contains knowledge for preventive maintenance.

Second, it will be shown that these hypotheses are supported by current research in maintenance of CBR systems.

The paper is structured as follows: Section 2 describes the scenario of how a commercial CBR system is commonly built. Section 3 develops the hypotheses mentioned above. Then, the support for the central role of the case base in current research is discussed (section 4). Finally, this paper concludes that the case base knowledge container must play the central role in maintenance in case–based reasoning (section 5).

2 Scenario

Companies routinely collect a vast amount of data through their business processes. This includes vast amounts of documents, and huge databases, customer support knowledge management systems, and planning systems. For example, sales companies have their product catalogs and customer data bases. Help–desk organizations collect manuals, magazines, frequently asked questions (FAQ), trouble shooting documents, and trouble tickets.

To build a CBR system all the four knowledge containers have to be filled from these data sources. This is outlined briefly in the following subsections.

2.1 Vocabulary

The vocabulary container includes all the information about the names, definitions, and structure of the CBR system. In business processes, keywords are first extracted from the data sources. Depending on the focus of the case base terms with no relevance (stopwords) are also collected. The stopwords can then be ignored in queries. An expert may add missing vocabulary if necessary, define an ontology and structure the keywords into hierarchies.

In terms of the product catalog example the names of the products and the order numbers are good candidates for the vocabulary knowledge container. From the product descriptions keywords along with stopwords can be extracted. An expert can add synonyms for the product names.

2.2 Similarity measures

The similarity container stores the measures which are necessary to retrieve a case. It characterizes part of the background knowledge of the particular domain, and together with an indexing function, the similarity measures establish the retrieve process. They can be extracted from the data sources and are entered by an expert. Similarity measures are somewhat indirect knowledge. Local similarity measures describe the similarity between single attributes of a query and a case. Global similarity measures combine local similarity values.

For example, local similarity measures compare product properties whereas global similarity measures compare the query and the cases on the whole. Local similarity measures can often be derived directly from the domain knowledge.

2.3 Adaptation knowledge

The adaptation container encloses the knowledge to transform the retrieved solutions meeting the needs of the given problem. Adaptation knowledge is dependent on the application domain and is usually harder to acquire than the similarity measures.

For example, products may be adapted to meet customer requirements but in help–desk scenarios there is often no chance to adapt a solution automatically to solve a given problem.

2.4 Case Base

The case base is developed as directly from the data sources as possible in commercial CBR applications. Therefore cases are easy to acquire in the construction and deployment phase of a case–based reasoning system. For example, product catalogs provide a good basis to build a case base whereas help–desk domains usually require more effort.

3 Hypotheses

While the development of case bases from companies' data sources is relatively well explored [5], the maintenance of commercial case–based reasoning systems is not. For the maintenance of commercial CBR systems the role of the case base knowledge container must be examined further. Therefore, four hypotheses[1] are presented: First, there is no

[1] Hypotheses are unproved assumptions, which may not always be directly proven as in mathematics.

maintenance without regarding the case base. Second, cases are natural crystallization points for the knowledge in CBR systems. Third, the case base triggers maintenance operations. And fourth, the case base contains knowledge for preventive maintenance. These hypotheses will be explained now in more detail.

3.1 There is no maintenance without regarding the case base

Maintenance can be done for every knowledge container separately but not without regard to the other knowledge containers. In particular, it cannot be done without regarding the case base. The vocabulary container is maintained by inserting new keywords from new cases within the case base, or by deleting or changing keywords. The similarity measures are updated through the contents of the case base. If there is adaptation knowledge, then the case base is used to refine this knowledge. The case base knowledge container itself can be maintained by the insertion of new cases into the case base or through the optimization of the contents of the case base.

3.2 Cases are natural crystallization points for the knowledge

Cases are discrete packets of knowledge in a CBR system, e.g. problem solution pairs in a help–desk system or products in a product catalog. Background knowledge can be grouped around these portions of knowledge. By putting additional knowledge into similarity measures a case becomes applicable in more situations. As a result of the addition of knowledge into the adaptation knowledge container, the applicability of the case is extended even further.

3.3 The case base triggers maintenance operations

The knowledge in the vocabulary, similarity measures, and adaptation containers is compiled knowledge. In contrast to the case base with its interpreted knowledge these three containers do not change as much as the case base. Because only (recognized) changes can be used to trigger maintenance operations the case base is the most appropriate choice.

Possible triggers are timers, counters, and quality measures [9]. A trigger can be based on single cases or on the whole case base. Timers and counters trigger the maintenance operations when they expire or (do not) reach a specific threshold. For example, *actuality* is a timer and *retrieval frequency* is a counter. Quality measures also can be derived from one or more cases. For example, *inconsistency* can occur in one single case (intra–case inconsistency) or between two ore more cases (inter–case inconsistency) [7].

3.4 The case base contains knowledge for preventive maintenance

The case base as the only interpreted knowledge container behaves dynamically with respect to its contents and size. Therefore, this ability of the case base knowledge container extends the maintenance with a preventive part. This preventive maintenance can be supported by trends or the detection of important and undeveloped or superfluous areas in the case base. For the prediction of trends data must be collected and meaningful measures must be used. This data can be collected from single cases and from the case base.

For example, in a help–desk environment a printer series is being replaced by a newer generation of printers piece by piece. The knowledge about this specific line of printers will become obsolete. The analysis of the access frequency of such cases shows that this knowledge tends to become superfluous and that this area of the case base should be kept under close observation. An analysis of unresolved queries would also show that there is an increasing interest in the newer printer generation. Visualization with the help of measures can support this essential maintenance task.

4 The role of the case base in current research

This section shows the role of the case base in current research with the help of the four hypotheses presented above.

4.1 There is no maintenance without regarding the case base

Heister and Wilke [6] describe how to handle changes in the vocabulary container through repair operations on the case base. They describe all possible operations on the vocabulary and the effects on the similarity measures and the case base. For example, the deletion of parts of the vocabulary can render similarity measures or cases useless. Repair scripts help the user to counteract vocabulary changes.

Aha [1] discusses four different case–based learning (CBL) algorithms. They reduce the costs of case retrieval through a case addition strategy. Noisy cases are eliminated through statistical prediction. Irrelevant features and similarity dependencies are detected and removed by adjusting the similarity weights.

Aha [2] and Aha and Breslow [3, 4] describe a maintenance approach which refines conversational case libraries (CCL). The basic idea of refining CCL is a transformation process which fulfills the constraints of a given design guidelines' knowledge. The input of the process is the CCL and some design guidelines. The output is a transformed CCL. First, a hierarchy inducer transforms the CCL into hierarchies. These hierarchies and the design guidelines are the input for a hierarchy editor. The results are hierarchies which meet the design guidelines. Finally, a case extractor transforms these back to cases of a CCL.

4.2 Cases are natural crystallization points for the knowledge

The improvement of adaptation knowledge is the topic of the work of Wilke, Vollrath and Bergmann [13]. This paper describes a *knowledge light* approach where adaptation knowledge is shifted from the other three knowledge containers. The main focus of this paper is to show how to transfer knowledge of the case base knowledge container into the adaptation knowledge container by using inductive learning methods. Adaptation knowledge is crystallized around the cases.

4.3 The case base triggers maintenance operations

The detection of inconsistencies and redundancy in semi–structured and unstructured case bases is the focus of the work of Racine and Yang [7, 8]. They propose measures to detect and eliminate intra–case inconsistencies, inter–case inconsistencies, and redundancies. Intra–case inconsistency is detected through constraint violations of attributes within a case. Inter–case inconsistency is discovered across two or more cases. Redundancy is identified if at least two cases are identical, which is difficult to determine in unstructured cases.

Reinartz, Iglezakis, and Roth–Berghofer [9] define several quality measures based on different case and case base properties. The quality measures describe specific characteristics of the case base such as correctness, consistency, uniqueness, minimality, and incoherence. These quality measures could be used to trigger maintenance operations.

Smyth and Keane [11] and Zhu and Yang [14] use the size of the case base as a trigger to preserve case base competence. Preserving competence means that the case base performance remains at a certain level while minimizing the size of the case base.

4.4 The case base contains knowledge for preventive maintenance

Smyth and McKenna [12] developed a system for visualization of competence groups and for the navigation through the space of competence group cases: the CASCADE (Case Authoring Support and Development Environment) system. This system stores the information about which competence group a case belongs to with the case.

5 Summary

This workshop paper looked at the role of the case base knowledge container for the maintenance of case–based reasoning systems. Based on a typical scenario for the development of a commercial case–based reasoning systems four hypotheses were presented:

1. There is no maintenance without regarding the case base.
2. Cases are natural crystallization points for the knowledge in CBR Systems.
3. The case base triggers maintenance operations.
4. The case base contains knowledge for preventive maintenance.

This paper showed also that these hypotheses are supported by current research. According to above observations, the case base knowledge container has the central role in maintenance of case–based reasoning systems.

Future research should concentrate on a framework for maintaining case–based reasoning systems by concentrating on maintaining the case base. Therefore, it is important to develop quality measures for case bases. These quality measures can then be used by methods for triggering maintenance operations and preventive maintenance. Also, methods are needed to implement quality improvement of the case base.

Acknowledgments

The authors are grateful to Thomas Reinartz (DaimlerChrysler, Research and Technology) for the fruitful discussions and helpful hints.

References

[1] David W. Aha. Case–based learning algorithms. In *Proceedings of the DARPA Case–Based Reasoning Workshop*, pages 147–158. Morgan Kaufmann, 1991.

[2] David W. Aha. A proposal for refining case libraries. In *German Workshop on Case–Based Reasoning (GWCBR)*, 1997.

[3] David W. Aha and Leonard A. Breslow. Learning to refine case libraries: Initial results. Technical Report AIC–97–003, Navy Center for Applied Research in AI, 1997.

[4] David W. Aha and Leonard A. Breslow. Refining conversational case libraries. In *Proceedings of the Second International Conference on Case–Based Reasoning*, pages 267–278, 1997.

[5] Ralph Bergmann, Sean Breen, Mehmet Göker, Michel Manago, and Stefan Wess. *Developing Industrial Case–Based Reasoning Applications: The INRECA Methodology*. Springer–Verlag, 1999.

[6] Frank Heister and Wolfgang Wilke. An architecture for maintaining case–based reasoning systems. In *Proceedings of the European Workshop on Case–Based Reasoning (EWCBR)*, pages 221–232, 1998.

[7] Kirsti Racine and Qiang Yang. On the consistency management of large case bases: the case for validation. In *Proceedings of the AAAI–96 Workshop on Knowledge Base Validation, American Association for Artificial Intelligence (AAAI)*, pages 84–90, 1996.

[8] Kirsti Racine and Qiang Yang. Maintaining unstructured case bases. In *Proceedings of the 2nd International Conference on Case–Based Reasoning (ICCBR)*, pages 553–564. Springer–Verlag, 1997.

[9] Thomas Reinartz, Ioannis Iglezakis, and Thomas Roth–Berghofer. On quality measures for case base maintenance. In *Proceedings of the 5th European Workshop on Case–Based Reasoning*, pages 247–259. Springer–Verlag, 2000.

[10] Michael M. Richter. The knowledge contained in similarity measures. Invited Talk at the 1st International Conference on Case–Based Reasoning (ICCBR), 1995.

[11] Barry Smyth and Mark T. Keane. Remembering to forget: A competence–preserving deletion policy for case–based reasoning systems. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 377–382, 1995.

[12] Barry Smyth and Elizabeth McKenna. A portrait of case competence: Modelling the competence of case–based reasoning systems. In *Proceedings of the 4th European Workshop on Case–Based Reasoning.*, pages 208–220. Springer–Verlag, 1998.

[13] Wolfgang Wilke, Ivo Vollrath, and Ralph Bergmann. Using knowledge containers to model a framework for learning adaptation knowledge. In Dietrich Wettschereck and David W. Aha, editors, *European Conference on Machine Learning (MLNet) Workshop Notes — Case-Based Learning: Beyond Classification of Feature Vectors*, pages 68–75, 1997.

[14] Jun Zhu and Qiang Yang. Remembering to add: Competence–preserving case addition policies for case base maintenance. In *Proceedings of the International Joint Conference in Artificial Intelligence (IJCAI)*, pages 234–239, 1999.