The Knowledge Engineering Review, Vol. 20:3, 305–310.
doi:10.1017/S0269888906000543 Printed in the United Kingdom

Knowledge management in case-based reasoning

KLAUS-DIETER ALTHOFF[1] and ROSINA O. WEBER[2]

[1]*Intelligent Information Systems, University of Hildesheim, 31141 Hildesheim, Germany; e-mail: althoff@iis.uni-hildesheim.de*
[2]*College of Information Science and Technology, Drexel University, Philadelphia, PA 19104, USA; e-mail: rosina.weber@drexel.edu*

Abstract

This commentary describes two core knowledge management approaches that applied case-based reasoning as a methodological foundation for organizational systems managing experience. These research projects illustrate the presence of knowledge management in case-based reasoning by focusing on the dualism between case-based reasoning and organizational approaches targeting knowledge management goals.

1 Introduction

Knowledge Management (KM) is an emerging discipline that focuses on efforts leading to the rational allocation of organizational knowledge assets. Typical KM solutions are described in terms of a knowledge cycle that entails knowledge tasks such as capture, distribution, and reuse. Knowledge cycles are strongly correlated with the case-based reasoning (CBR) cycle, which includes retrieve, reuse, revise, and retain steps (Aamodt & Plaza, 1994). The strong association between the CBR cycle and KM's knowledge cycles justify the consistent use of CBR to guide the design of KM systems (e.g., Kitano *et al*., 1993; Aamodt & Nygaard, 1995; Althoff *et al*., 1998b; Aha *et al*., 1999; Weber & Aha, 2003).

The affinity between KM and CBR goes beyond their cycles. At the research level, the KM literature recommends that effective KM solutions target people, processes, and technology (Abecker *et al*., 2000). From a CBR perspective, Aamodt and Nygaard (1995) have long ago suggested that CBR research has to consider practical applications and focus on optimizing not the CBR system alone but the combination of a CBR system and its user. This represented an important starting point for viewing CBR as an approach contributing to KM.

As a result, there have been many research activities on CBR and KM. The relationship between these fields is illustrated, for instance, by a number of CBR and KM-related events. In 1999, the *AAAI Workshop Exploring Synergies of Knowledge Management and Case-Based Reasoning* (Aha *et al*., 1999) focused on requirements for the effective contribution of CBR to KM. In 2000, the *AAAI Workshop on Intelligent Lessons learned Systems* (Aha & Weber, 2000) targeted a broader scope but its intelligent component relied mainly on CBR. In 2001, the program committee of the traditional *German Workshop on CBR* decided to change the name of their well-known annual CBR event to *German Workshop on Experience Management*.

The close ties between KM and CBR is also evidenced in books. Tautz (2000) describes how to customize experience management systems to organizational needs especially from a software engineering point of view. Bergmann (2002) represents an encompassing textbook on experience

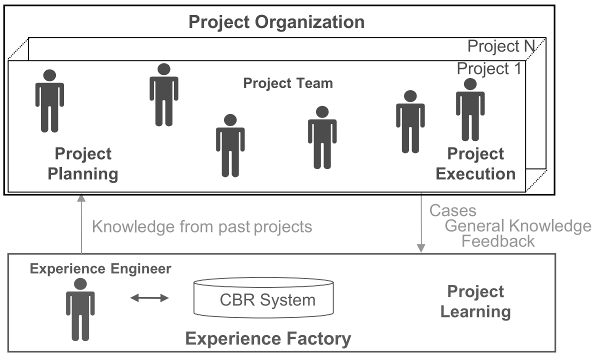

Figure 1 EF as an organizational model for running CBR systems

management, presenting all aspects of real-life CBR applications. Watson (2003) presents corporate memories from a CBR perspective.

2 Selected publications

Two core KM approaches were developed by applying the CBR methodology as a foundation for organizational systems that manage experience. These research projects illustrate the presence of KM in CBR by focusing on the dualism between CBR and organizational approaches targeting KM goals (Althoff *et al*., 1998a; Weber *et al*., 2001).

Althoff *et al*. (1998a) focused on the experience factory, which is an organizational framework to learning from experience, particularly tailored to software engineering. Weber *et al*. (2001) concentrated on lessons learned systems (LLSs), a common repository-based KM initiative (Stewart, 1997) for storing organizational experiences for future reuse. Both approaches used case-based reasoning as an underlying framework in knowledge management. We describe these two lines of research—EF/CBR and LLS/CBR—in the following sections.

3 Experience factory/case-based reasoning (EF/CBR)

Althoff and Tautz integrated CBR with the experience factory (EF) model (Basili *et al*., 1994a), an organizational approach for continuously learning from experience (and other kinds of knowledge) (Althoff & Wilke, 1997; Tautz & Althoff 1997; Althoff *et al*., 1998a). For the first time, the CBR methodology (Kolodner, 1993; Althoff, 2001; Watson, 2003) and technology (Bartsch-Spörl, 1987; Aamodt, 1989; Althoff & Wess, 1992) were seamlessly integrated within one conceptual framework (Tautz, 2000; Tautz & Gresse von Wangenheim, 1998).

Since late 1996 Althoff and Tautz (Tautz & Althoff, 1997, 1998) have developed a deep integration of the CBR approach and the EF approach. EF naturally introduces a form of 'experience management' (EM), which generalizes the concept of manipulating experiences. While CBR is an Artificial Intelligence technology for building knowledge-based systems, EF is an organizational approach for learning from experience that includes an experience base (EB) for storing these experiences. EF focused on the processes around the EB but *not* on how to implement an EB. The integration of CBR and EF led to four immediate positive consequences.

1. CBR became the obvious implementation technology for an EB (Henninger, 1995; Figure 1).
2. EF can be used as an organizational infrastructure (i.e. roles, responsibilities, processes, organizational implementation and management strategies, competence development strategies) for a CBR system (Althoff & Wilke, 1997). This includes using the quality improvement paradigm (QIP) underlying an EF for goal-oriented knowledge development for CBR systems (Figure 2).
3. EF techniques such as goal-oriented measurement and evaluation (Basili *et al*., 1994b) can be applied to CBR systems (Nick *et al*., 1999), representing an innovative contribution to the state of the art in CBR system evaluation (Althoff, 1997; Althoff & Nick, 2006).

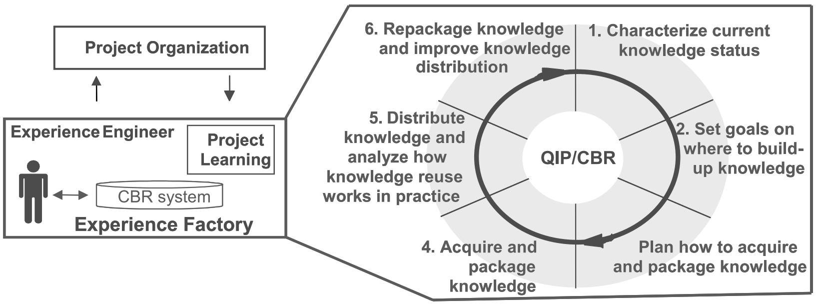

Figure 2 Goal-oriented knowledge development of CBR systems

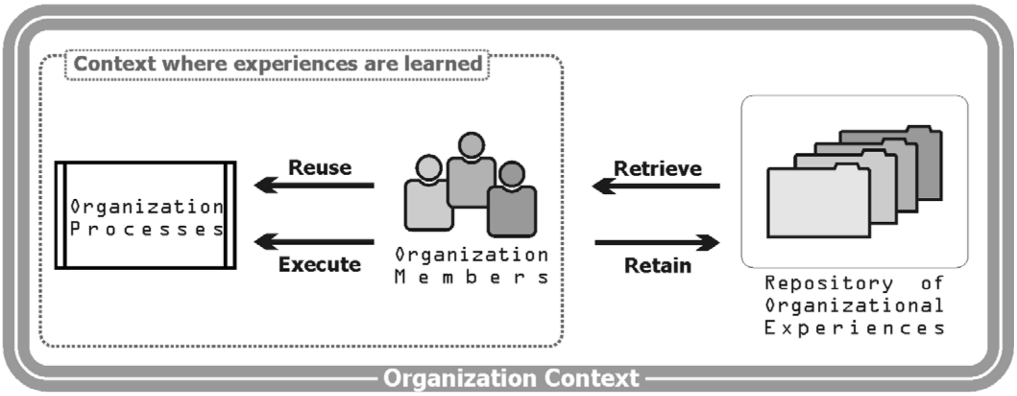

Figure 3 LLSs in the organization context

4. The CBR task-method decomposition model, a refinement of the CBR cycle by Aamodt and Plaza (1994), and its extension by Althoff (1997), can be used to describe EF processes in greater detail than was previously possible (Tautz & Althoff, 1997, 1998; Tautz, 2000; Nick, 2005).

Here the CBR cycle was viewed as 'human based', that is as a means to describe human problem solving and learning behavior, and not as an approach for systematically describing CBR systems, as it was originally thought by Aamodt & Plaza (1994). This was possible due to the work of Schank (1982) and Kolodner (1993) who (also) introduced CBR as a means for modeling human problem solving and learning behavior, which guaranteed the 'situatedness of CBR'. The outcome of the deep integration of EF and CBR was a methodology for building not only CBR systems but also experience management systems, sometimes also called 'experience-based information systems' (Nick, 2005; Althoff & Nick, 2006).

4 Lessons learned systems/case-based reasoning

Weber *et al*. (2001) have analyzed LLSs and categorized different methods for implementing the knowledge tasks in the lessons learned cycle. They have investigated the use of different intelligent techniques to support organizational knowledge sharing efforts. They concluded that CBR is, to a large extent, a suitable technology for implementing LLSs. Analogous to the EF, LLSs represent an organizational initiative that uses a repository to store knowledge learned from experience for future reuse (Figure 3).

Weber *et al*. (2001) describe and illustrate the potential positive consequences of adopting the CBR methodology. One potential advantage stems from using the representation of targeted processes (i.e. processes where lessons are applicable) to model cases. This strategy requires close integration of the lessons learned module to these targeted processes. It allows lessons to be retrieved based on their applicability to the targeted processes and in the context where they are delivered. This integration has the potential to alleviate most of the current problems found in traditional LLSs, where users have to use a standalone repository with poorly collected lessons that are not associated with their targeted processes. When knowledge is disseminated in the context of its reuse, it motivates more effective and efficient knowledge sharing. This article integrates ideas collected from the 2000 AAAI Workshop on Intelligent Lessons learned Systems.

For the practical adoption of CBR as the underlying framework for LLSs, Weber *et al.* (2001) propose a case representation for lessons learned. The case representation was later used in the monitored distribution (MD) approach for proactive distribution of lessons learned (Aha *et al.*, 2001). A description of lessons learned includes the organizational process that it targets. Therefore, MD can be integrated with organizational systems. MD addresses problems associated with other distribution methods that are divorced from targeted organizational processes and requires users to have the initiative and skills to access, manipulate, and interpret knowledge artifacts. MD motivates the reuse of a knowledge artifact by bringing to the attention of the user when and where it is applicable and by including a rationale for its reuse (Weber & Aha, 2003). The MD approach shifts the burden of knowledge dissemination from the user to the software through a CBR module that monitors when a lesson learned should be disseminated to the user by matching the lesson to the user's context. The benefit of the MD approach has been demonstrated in an experiment that simulated military operations planned with and without the reuse of lessons learned taken from the NLLS (Navy Lessons Learned System) repository (Aha *et al.*, 2001; Weber & Aha, 2003).

While the work on EF/CBR (Althoff *et al.*, 1998a,b) has evolved into the development of a design for experience-based information systems, the work on LLSs/CBR (Weber *et al.*, 2001) has taken a more technical approach. The most recent manifestation of LLSs/CBR is an automated learning module that can be integrated with another system (Weber & Wu, 2004). Differing from experience-based information systems that target humans as direct beneficiaries of managed experiences, the case-based knowledge management framework, described in (Weber & Wu, 2004), focuses on managing knowledge assets learned by intelligent systems. Another approach uses current tasks, a to-do list, role and a skill profile to proactively disseminate information items to individual knowledge workers (Holz, 2003).

5 Challenges

A challenging aspect of current knowledge sharing efforts concerns knowledge representation. Weber and Kaplan (2003) studied knowledge cycles in several implementations of knowledge-based methods for KM. One of their conclusions was that using different knowledge formalisms in each step of the cycle requires conversions that result in loss of knowledge. Knowledge can be available from different sources and formats, and knowledge conversion is a known challenge prone to errors. The adoption of a representation formalism, such as the case representation for lessons learned, that can be used throughout the entire knowledge cycle presents potential benefits with respect to knowledge permanence and precision.

The use of lessons learned modules as explained above also allows the representation of knowledge of different scales. For example, one lessons learned module can be composed of lessons that are applicable to a set of processes while another module uses a case base of a specific task. A generalization of the EF/CBR approach may be used to embed various types of knowledge-based systems in real-life environments (Decker & Althoff, 2004; Althoff *et al.*, 2005). Another result of the choice of formalism is its impact on knowledge acquisition (Weber & Kaplan, 2003). Knowledge acquisition is affected by the choice of representation. If acquisition is processed without the use of the target knowledge formalism, then it cannot capture the knowledge in its final form. Acquisition should not end until knowledge is acquired in the form it will be stored in and reused.

References

Aamodt, A, 1989, Towards robust expert systems that learn from experience—an architectural framework. In Boose, J, Gaines, B and Ganascia, J-G (eds) *Proceedings of EKAW-89, Third European Knowledge Acquisition for Knowledge-based Systems Workshop, Paris, July 1989*, pp. 311–326.

Aamodt, A and Nygaard, M, 1995, Different roles and mutual dependencies of data, information, and knowledge—an AI perspective on their integration. *Data and Knowledge Engineering* **16**, 191–222.

Aamodt, A and Plaza, E, 1994, Case-based reasoning: foundational issues, methodological variations, and system approaches. *AI Communication* **7**(1), 39–59.

Abecker, A, Decker, S and Maurer, F, 2000, Organizational memory and knowledge management. *Information Systems Frontiers* **2**(3–4), 251–252.

Aha, DW, Becerra-Fernandez, I, Maurer, F and Muñoz-Avila, H (eds), 1999, *Exploring Synergies of Knowledge Management and Case-Based Reasoning: Papers from the AAAI 1999 Workshop (Technical Report no. WS-99-10)*. Menlo Park, CA: AAAI Press.

Aha, DW, Weber, RO, Muñoz-Avila, H, Breslow, LA and Gupta, K, 2001, Bridging the lesson distribution gap. In *Proceedings of IJCAI'01*. San Mateo, CA: Morgan Kaufmann, pp. 987–992.

Aha, DW and Weber, RO (eds), 2000, *Intelligent Lessons Learned Systems: Papers from the AAAI 2000 Workshop (Technical Report no. WS-00-08)*. Menlo Park, CA: AAAI Press.

Althoff, K-D, 1997, Evaluating case-based reasoning systems: the INRECA case study, Habilitation thesis, Department of Computer Science, University of Kaiserslautern, Germany.

Althoff, K-D, 2001, Case-based reasoning. In Chang, SK (ed.) *Handbook on Software Engineering and Knowledge Engineering, vol. 1 Fundamentals*. World Scientific, pp. 549–588.

Althoff, K-D and Wilke, W, 1997, Potential uses of case-based reasoning in experience based construction of software systems and business process support. In Bergmann, R and Wilke, W (eds) *Proceedings of the 5th German Workshop on Case-Based Reasoning, LSA-97-01E, Centre for Learning Systems and Applications, University of Kaiserslautern, Germany*, pp. 31–38.

Althoff, K-D, Birk, A, Gresse von Wangenheim, C and Tautz, C, 1998a, Case-based reasoning for experimental software engineering. In Lenz, M, Bartsch-Spörl, B, Burkhard, H-D and Wess, S (eds) *Case-Based Reasoning Technology—From Foundations to Applications (Lecture Notes in Artificial Intelligence, 1400)*. Berlin: Springer, pp. 235–254.

Althoff, K-D, Bomarius, F and Tautz, C, 1998b, Using case-based reasoning for building learning software organizations. In *Proceedings European Conference on Artificial Intelligence 1998 Workshop on Building, Maintaining, and Using Organizational Memories (OM'98), Brighton, UK.*

Althoff, K-D, Mänz, J and Nick, M, 2005, Maintaining experiences to learn: case studies on case-based reasoning and experience factory: case studies and implications. In *Proceedings 6th Workshop Days of the German Computer Science Society (GI) on Learning, Knowledge, and Adaptivity (LWA 2006) Workshop on Machine Learning, Knowledge Discovery, and Data Mining, Saarland University, Germany*, October 2005.

Althoff, K-D and Nick, M, 2006, *How to Support Experience Management with Evaluation—Foundations, Evaluation Methods, and Examples for Case-Based Reasoning and Experience Factory (Lecture Notes of Computer Science/Artificial Intelligence)*. Springer.

Althoff, K-D and Wess, S, 1992, Case-based reasoning and expert system development. In Schmalhofer, F, Strube, G and Wetter (eds) *Contemporary Knowledge Engineering and Cognition*. Berlin: Springer, pp. 146–158.

Bartsch-Spörl, B, 1987, Approaches for processing case knowledge in expert systems (in German: ansätze zur behandlung von fallorientiertem erfahrungswissen in expertensystemen). *KI* **4**, 32–36.

Basili, VR, Caldiera, G and Rombach, D, 1994a, Experience factory. In Marciniak, JJ (ed.) *Encyclopedia of Software Engineering*, vol. 1. John Wiley and Sons, pp. 469–476.

Basili, VR, Caldiera, G and Rombach, D, 1994b, Goal-question-metric paradigm. In Marciniak, JJ (ed.) *Encyclopedia of Software Engineering*, vol. 1. John Wiley and Sons, pp. 528–532.

Bergmann, R, 2002, *Experience Management: Foundations, Development Methodology, and Internet-Based Applications*. Berlin: Springer.

Decker, B and Althoff, K-D, 2004, Process learning and experience management—results from the indigo project (in German: prozesslernen und erfahrungsmanagement—ergebnisse aus dem indiGo-Projekt). In *Proceedings 5th Workshop Days of the German Computer Science Society (GI) on Learning, Knowledge, and Adaptivity (LWA 2004) Workshop on Machine Learning, Knowledge Discovery, and Data Mining, Humboldt University Berlin, Germany*, October 2004.

Henninger, S, 1995, Developing domain knowledge through the reuse of project experiences. In *Proceedings Symposium on Software Reusability (SSR '95), Seattle, WA*, April 1995.

Holz, H, 2003, *Process-based Knowledge Management Support for Software Engineering*, PhD thesis, Department of Computer Science, University of Kaiserslautern, published at dissertation.de Online-Press.

Kitano, H, Shimazu, H and Shibata, A, 1993, Case-method: a methodology for building large-scale case-based systems. In *Proceedings of the National Conference on Artificial Intelligence (AAAI-93)*. Menlo Park, CA: AAAI Press.

Kolodner, JL, 1993, *Case-Based Reasoning*. Los Altos, CA: Morgan Kaufmann.

Nick, M, 2005, Experience maintenance through closed-loop feedback, PhD thesis, Department of Computer Science, University of Kaiserslautern, Germany.

Nick, M, Althoff. K-D and Tautz, C, 1999, Facilitating the practical evaluation of knowledge-based systems and organizational memories using the goal-question-metric technique. In *Proceedings Banff Workshop on Knowledge Acquisition, Modeling, and Management (KAW'99)*.

Schank, RC, 1982, *Dynamic Memory: A Theory of Learning in Computers and People*. Cambridge, MA: Cambridge University Press.

Stewart, TA, 1997, *Intellectual Capital: The New Wealth of Organizations*. New York: Currency Doubleday.

Tautz, C, 2000, Customizing software engineering experience management systems to organizational needs, PhD Thesis, Department of Computer Science, University of Kaiserslautern, Germany.

Tautz, C and Althoff, K-D, 1997, Using case-based reasoning for reusing software knowledge. In Leake, D and Plaza, E (eds) *Case-Based Reasoning Research and Development—Second International Conference (ICCBR97)*. Berlin: Springer.

Tautz, C and Althoff, K-D, 1998, Operationalizing comprehensive software knowledge reuse based on CBR methods. In Gierl, L and Lenz, M (eds) *Proceedings of the Sixth German Workshop on Case-Based Reasoning (GWCBR'98), Berlin*, Technical Report, University of Rostock, Germany.

Tautz, C and Gresse von Wangenheim, C, 1998, REFSENO: a representation formalism for software engineering ontologies, IESE-Report no. 015.98/E, http://www.iese.fraunhofer.de/pdf files/iese-015 98.pdf.

Watson, I, 2003, *Applying Knowledge Management: Techniques for Building Corporate Memories*. San Francisco, CA: Morgan Kaufmann.

Weber, R, Aha, DW and Becerra-Fernandez, I, 2001, Intelligent lessons learned systems. *International Journal of Expert Systems Research and Applications* **20**(1), 17–34.

Weber, R and Aha, DW, 2003, Intelligent delivery of military lessons learned. *Decision Support Systems* **34**(3), 287–304.

Weber, R and Kaplan, R, 2003, Knowledge-based knowledge management. In Jain, R, Abraham, A, Faucher, C and van der Zwaag, BJ (eds) *Innovations in Knowledge Engineering*. Advanced Knowledge International Pty Ltd.

Weber, R and Wu, D, 2004, Knowledge Management for Computational Intelligence Systems. In *Proceedings Eighth IEEE International Symposium on High Assurance Systems Engineering (HASE 2004)*. Los Alamitos, CA: IEEE Computer Society, pp. 116–125.