"Case-Based Reasoning Research and Development, Proceedings of the Third International Conference on Case-Based Reasoning ICCBR-99, Seeon Monastery, Germany, July 27-29, 1999", K.D. Althoff, R. Bergmann, L.K. Branting eds, LNAI 1650, pp. 132-146,.Berlin, Springer Verlag, 1999

Development and Utilization of a Case-Based Help-Desk Support System in a Corporate Environment

Mehmet Göker[1] & Thomas Roth-Berghofer[2]

[1] DaimlerChrysler Research and Technology 3, FT3/KL
P.O. Box 2360, D89013 Ulm, Germany
mehmet.goeker@daimlerchrysler.com
Phone: +49 731 5052856 Fax: +49 731 5054210

[2] tec:inno GmbH
Sauerwiesen 2, D67661 Kaiserslautern, Germany
roth@tecinno.com
Phone: +49 6031 606400 Fax: +49 6031 606409

Abstract: Current Case-Based Reasoning (CBR) process models present CBR as a low maintenance AI-technology and do not take the processes that have to be enacted during system development and utilization into account. Since a CBR system can only be useful if it is integrated into an organizational structure and used by more than one user, processes for continuous knowledge acquisition, -utilization and -maintenance have to be put in place. In this paper the short-comings of classical CBR process models are analyzed, and, based on the experiences made during the development of the case-based help-desk support system HOMER, the managerial, organizational and technical processes related to the development and utilization of CBR systems described.

1. Motivation

Case-Based Reasoning (CBR) has long been considered as an AI technology with comparably bw maintenance effort. However, with the advent of CBR systems in industrial environments, issues that have to do with the processes involved in putting a knowledge repository into operation in an organization arise. Especially the processes involved in initial and continuous knowledge acquisition, case-base and domain-model maintenance as well as the organizational impact of and impact of the organization on a CBR system have not been analyzed and understood completely. These aspects are currently neither covered in academic CBR models nor supported adequately in commercially available CBR systems.

On the following pages we describe the processes that had to be enacted during the development and utilization of the case-based help-desk support system HOMER [2]. After the processes had been derived from one specific application, they were verified, revised and re-used during several other CBR-projects by means of the

INRECA-II[i] methodology [1]. We believe that most of the results can be transferred to other domains and applications

2. Current Case-Based Reasoning Process Models

Several variations of the Case-Based Reasoning process model exist in literature [cf. 3,4,5,6]. The basic idea behind all approaches is to retrieve problem solving experience that has been stored as a case in a case-base, adapt and reuse it to solve new problems and, if not successful, learn from failures.

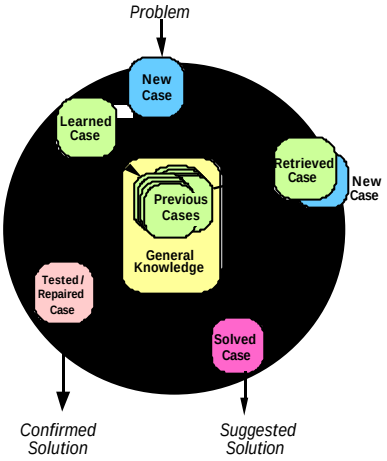

Fig.1: The Case-Based Reasoning Process Model according to Aamodt and Plaza [9]

On the abstract level the CBR process can be described to be comprised of four main tasks (Fig.1): *Retrieve*, *Reuse*, *Revise* and *Retain* [6].

During *Retrieval* the most similar case or cases in the case-base are determined based on the new problem description.

During *Reuse* the information and knowledge in the retrieved case(s) is used to solve the new problem. The new problem description is combined with the information contained in the old case to form a solved case.

During *Revision* the applicability of the proposed solution (solved case) is evaluated. If necessary and possible the proposed case is repaired.

If the case solution generated during the reuse phase is not correct and cannot be repaired, the case-base is updated with a new learned case or by modification of some existing cases in the *Retain* task.

3. Shortcomings of Current Case-Based Reasoning Process Models

3.1. Effects of User Groups

A CBR System is a means to store, share and re-use experience. If the experience stored in a CBR system is only used by the person who enters it, the use of the system will be rather limited. The goal of developing a CBR system, especially in a corporate environment, is to create a means to capture, cumulate and re-use corporate experience with all the benefits that are associated with such a venture [cf. 8].

It has been claimed that "Knowledge Acquisition for a case-based system is natural" [5], and that "CBR offers a significant benefit for knowledge maintenance: a user may be able to add missing cases to the case library without expert intervention"

[7]. While this may indeed be true for static domains with a very limited number of users of the system, we would like to be somewhat more cautious to this respect.

If a CBR System is not only used by one user but rather a group of users, the quality (in terms of representation and content) of the new cases that each user creates will vary. This will have a negative effect on the overall quality of the case-base (in terms of correctness, coverage and minimality) and reduce the effectiveness and efficiency of the system in general. Processes that ensure that the overall quality of the case-base does not deteriorate when new cases are entered have to be put in place. Depending on the user group that is going to utilize the system, the content of the case-base and the user interface of the system have to be adapted as well.

3.2. Effects of Time

Current CBR process models base their description on a static view of the domain. While this assumption is acceptable for academic purposes, it does not hold for real world applications.

Every real-world domain changes over time. Solutions that were applicable some time ago will become invalid. Indices that were suitable will become obsolete and similarities will change. Methods to ensure that the CBR system is up-to-date have to be developed and tasks that realize these methods have to be incorporated into the CBR process models.

3.3. Impact on/of the Organization

Both during the development and the utilization of a CBR System, changes in the way knowledge is handled take place within an organization. Personnel has to be dedicated to the task of acquiring and maintaining knowledge, the system has to be integrated into the daily operations and has to become part of the organizational culture. A CBR system can only be successful in the long run, if enough personnel to maintain, use and develop the system are available and set aside by management [cf. 9].

4. Case-Based Help-Desk Support Systems

Help-desks support end-users of complex technical equipment by providing information about the usage of the equipment and keep the systems operational by performing necessary maintenance tasks. Help desk operators are expected to be able to solve problems on very short notice, in a very short time, and to be knowledgeable in all areas that are related to the technical system at hand.

Help-desk operators use their own experiences to solve most of the problems that are relayed to them. However, as systems become more complex, the areas help-desk operators are experts in tend to diverge, i.e., problem solving experience is distributed among experts and the areas of expertise do not necessarily overlap. Nevertheless, when an end-user has a problem, he or she wants it solved as soon as possible. If that expert is not available, the user has to wait, which is annoying and not acceptable in a commercial environment. The problem-solving experience must be available to every help-desk operator at all times [2].

The goal of developing a case-based help-desk support system is to create a knowledge repository that contains problem-solving experiences for a complex technical domain that changes over time. This knowledge repository will be used in an organization, by a group of people with varying levels of expertise, in a time-critical operation. It is obvious that the development and use of such a system does not only involve technical processes, but also raises managerial and organizational issues. In the following sections, we describe the tasks that must be performed to develop a case-based help-desk support system and the processes that have to be put into place to make such a system operational.

5. Processes During Case-Based Help-Desk System Development and Utilization

5.1. Process Types

Table 1 lists the processes that have to be considered and /or performed during the development and utilization of a case-based help-desk system. We distinguish between organizational, technical and managerial processes [1].

Organizational processes cover those parts of the business process that need to be changed in order to make best use of a new software system. *Technical processes* transform and evolve product information from the problem description to the final (software) system. They address the development of the system and the required documentation itself. *Managerial processes* provide an environment and services for enacting the technical and the organizational processes.

		System Development	System Use
Managerial Processes		- Goal Definition - Awareness Creation - CBR-Tool Selection	- Progress Verification and Controlling
Organizational Processes		- Project Team Selection - Initial Domain Selection - Project Team Training - Knowledge Acquisition Process Development - Utilization Process Development	- End-User Training - Continuous Knowledge Acquisition - Utilization Process
Technical Processes	**General IT-System Related**	- System Specification - System Implementation - System Integration - System Verification	- Continuous System Maintenance
	Knowledge Repository Related	- Initial Knowledge Acquisition - Core Knowledge Acquisition	- Continuous Knowledge Acquisition and Maintenance

Tab. 1. Processes during case-based help-desk support system development and use.

5.2. Managerial Processes During System Development

Goal Definition. For a case-based help-desk support system project to be successful, precise goals must be determined at the outset. This enables management to fix the direction in which the project should develop and to measure the success of the project upon completion. Hard (quantitative) and soft (qualitative) success criteria should be identified [cf. 9]. Hard criteria are measurable quantities and cover aspects like:

- *problem solution quality* (first-call resolution rate, solution correctness, and consistency, average cost of proposed solution, and so on),
- *process quality* (average time needed to solve a problem, average number of escalations needed, quality of dynamic priority assignment, and so on),
- *organizational quality* (speedup in help-desk operator training, flexibility of staffing, cost per interaction, and so on).

Soft criteria, on the other hand, measure the subjective quality of the help-desk and cover aspects like:

- *end-user satisfaction* (availability of the help-desk, perceived competence, friendliness, and so on),
- *help-desk operator satisfaction* (workload, work atmosphere, repetitiveness of tasks, intellectual stimulation, and so on), and
- *corporate aspects* (preservation of knowledge, publicity, and so on.).

The goals must be communicated to the project team, and the team has to be motivated to achieve them.

When project goals are selected, it is important that these goals be realistic both in terms of their time frame and whether they can be achieved with an acceptable amount of resources.

Awareness Creation and Motivation. The case-based help-desk support system project targets the most precious asset of the employees: their experience. The project's goal is to collect the problem-solving experience of each relevant employee and make it available to whomever needs it in the organization.

Obviously the help-desk operators will have a motivational barrier to giving away their experience. Every employee knows that "knowledge is power." In help-desk environments or domains where experience is being used to solve problems having experience translates into being superior and indispensable, whereas giving away the knowledge can be perceived as becoming obsolete.

However, as soon as help-desk operators become part of a project team and understand that sharing knowledge means that they will get back much more than they invest, most barriers disappear. It has to be made clear that the user and beneficiary of the developed system is not going to be an anonymous "company," but they themselves. They will be able to access the experience of their colleagues and solve problems they could not solve before, as well as end situations in which colleagues constantly pester them for advice. The resulting help-desk system will enable them to work with increased efficiency and effectiveness.

Apart from the help-desk operators, management has to be motivated as well. CBR is perceived to be rather academic by most managers. While to them investing

resources into a database project seems to be no problem, investing into CBR is investing into a venture with an uncertain outcome. It has to be clarified that CBR is an established technology and by no means only an academic playground. The case-based help-desk support project must be seen as part of the long-term knowledge management strategy for the company. Since knowledge increases and evolves, the experience in a CBR system must be maintained continuously. System development is only the initial phase in any CBR project.

Without continuous management support and employees who are willing to fill and use the system, any CBR project is bound to fail.

CBR Tool Selection. Based on the project, domain, and user-group specifications, a suitable tool to develop the case-based help-desk support system must be selected. Criteria to be taken into account include:

- the operating environment in which the system is going to be used (hardware and software, network architecture, database type, and so on),
- the complexity of the technical domain (home appliances or networked workstations),
- the level of experience of both the end-users and the help-desk operators,
- the organization of the help-desk (number of levels, physical locations, and so on),
- the project goals that have been defined.

Since the case-based help-desk support system is going to serve as a (long-term) knowledge repository for the organization, this selection should be based not only on technical criteria, but also should take into account economic and organizational considerations, as well as strategic decisions of the company.

5.3. Organizational Processes During System Development

Project Team and Initial Domain Selection. The creation of a project team to serve as the "knowledge engineers" and the selection of a group to serve as initial test users of the system are the first organizational steps that must be taken.

Apart from the person implementing the case-based help-desk support system (*CBR consultant*), the project team should contain help-desk personnel who are very experienced in the relevant subdomain to be modeled and well respected by the help-desk operators outside the project group. Once selected, the members of the group should be kept constant, i.e., fluctuations should be avoided.

The group of initial users should comprise two types of help-desk personnel: One that is on a comparable level of expertise with the project team with respect to the selected subdomain (i.e., expert users) and help-desk personnel who are less familiar with the specific problem area (i.e., novice users). While the expert test-users can communicate to the project group in their language, the novice users will represent the target group for which the system is being implemented. Feedback from both types of users is required for a successful project. After a first "rapid prototype" has been implemented, the expert users can give hints regarding problems with the knowledge modeled in the system. The members of the novice user group, on the other hand, will serve as models of the help-desk operator who will use the system. The vocabulary in which the cases are being represented and the knowledge contained within them has to be adjusted to the novice user group

Which domain one selects for the initial knowledge acquisition is of utmost importance. The domain should be representative of the problems that are being handled at the help-desk, both in terms of complexity and frequency. It should also be a problem area that accounts for a considerable amount of the workload and about which the help-desk operators are interested in sharing (obtaining) knowledge.

Training the Project Team. Training the project team is an organizational process that has a major impact on the success of the help-desk project. At the beginning of the project, the project team is (most of the time) inexperienced with respect to CBR and knowledge acquisition. Since the project group will be responsible for system maintenance and continuous case acquisition after the development has finished, it is very important that they are trained in CBR, as well as in knowledge acquisition and modeling, during the initial knowledge acquisition.

While the project team should also get advanced training to be able to model, fill, and maintain the knowledge in the system, the test users only need to be trained in using the resulting case-based help-desk support system.

Development of the Knowledge Acquisition and Utilization Processes. The introduction and use of a case-based help-desk support system usually causes a re-evaluation and modification of the existing knowledge and information management processes in a help-desk environment. After the development of the case-based help-desk support system is complete, it will serve as the central source of information for the help-desk operators. To ensure a smooth flow of information, the knowledge sources and formats, as well as the qualification of the personnel that requires the knowledge, have to be analyzed, and processes that allow efficient and effective acquisition and use of knowledge have to be developed. One should keep in mind that while the group enacting the initial knowledge acquisition process is the project team and rather experienced, the users who use the system in the end (both in terms of knowledge retrieval and continuous acquisition) may be less qualified.

During the development of HOMER [2], we found it very useful to define three roles for the knowledge acquisition and utilization processes during the use of the help-desk system:

- the help-desk operator,
- the CBR author,
- the CBR administrator.

Help-desk operators are the users from the target group. Their duty is to use the implemented help-desk system in their daily work. If they cannot find an appropriate solution with the system, they will have to solve the problem on their own and generate a new case. Depending on the domain and on managerial decisions, this new case may or may not be made immediately available as an “unconfirmed” case to the other help-desk operators. For maintenance purposes, the operators are also encouraged to comment on the quality and applicability of the cases in the case base.

The unconfirmed, new cases have to be verified in terms of their correctness and suitability for the case base by the CBR author(s). The CBR author is a person with experience both in the domain and in using the CBR system. While the CBR author can decide on the quality and inclusion of a case in the case base, he or she is not

allowed to perform modifications on the vocabulary, the similarity, and the adaptation knowledge. These can only be performed by the CBR administrator.

The personnel enacting the roles of the CBR author(s) and the CBR administrator should be included in the project group from the start of the project. It should be noted that both these roles require a considerable amount of resources and should be performed by dedicated personnel. If the organization or the size of the help-desk does not permit dedicating more than one person to these tasks, the duties of the CBR author and CBR administrator should be performed by one person.

5.4. Technical Processes During System Development

General IT-System Development Related Processes. The development of a case-based help-desk support system is similar to any other IT project in most aspects. As usual, the system has to be specified, implemented, integrated, and verified in accordance with standard software engineering techniques. However, the user-interface and the connection to supporting programs (integration) are two features that require additional attention.

The user interface of the case-based help-desk support system has to be developed in accordance with the user group (i.e., second level, first level, or even end-user), the specific domain, and company policies (who is allowed to see what kind of data). It has to present the right data, at the right moment, and on the right level of abstraction.

A case-based help-desk support system cannot operate in isolation. While the CBR system will store experience, it will not contain data regarding device configurations, maintenance contracts, and users. Since this information is needed during problem solving, the system has to have interfaces to the databases containing this information.

Most help-desks use trouble-ticket tools in their daily operations; they record, manage, trace, escalate, and analyze the calls they receive. While these trouble-ticket tools are very useful in handling calls, they do not provide means to capture and reuse problem-solving experience. Depending on the environment, the case-based help-desk support system should also either be integrated into the user interface of the trouble-ticket tool or vice-versa. Data from the trouble-ticket system has to be transferred to the CBR system to initialize the attributes that relate to the data that has already been acquired. Except for very complex second-level applications, it is not feasible to have two points of entry to the problem-solving process.

Initial Knowledge Acquisition for the Case-Based Help-Desk Support System. A CBR system is useless without cases. When the case-based help-desk support system is handed over to the help-desk operators, it has to contain enough cases to cover at least part of the relevant problems at the help-desk. Initial knowledge acquisition serves three major goals:

- training the project team in knowledge acquisition,
- initializing the knowledge in the system,
- collecting enough help-desk cases to bootstrap the system.

During initial knowledge acquisition, the knowledge in the system can be distributed among the *domain model* (vocabulary), *similarity measure*, *adaptation knowledge*, and the *case base*. These *knowledge containers* [10] have to be created and filled. In principle, each container could be used to represent most of the knowledge. However,

this is obviously not very feasible, and the CBR consultant should carefully decide on the distribution of knowledge into the containers. After the initial knowledge acquisition is completed, this distribution is more or less fixed and should only be changed with caution.

The processes for the acquisition of knowledge for each container run in parallel and cannot be easily separated during the initial knowledge acquisition. Since the vocabulary lays ground for entering the cases and describing the similarity measures and adaptation knowledge, it has to be available first. However, to be able to create a domain model (i.e., the vocabulary), one has to understand how the domain is structured, and this can only be done by looking at the cases, the similarities, and the adaptation rules.

In our experience, the best way to approach this problem is to create and use standardized forms to acquire an initial amount of cases from the project team. The form should be developed in co-operation with the project team. A sample form that was developed for the initial case acquisition for the HOMER application is shown in Tab. 2.

The first thing that must be done is to ask the project team to fill out as many case acquisition forms as they can. By looking at the elements of the forms, the vocabulary (i.e., the phrases that have to be used and the domain structure) can be derived and a vocabulary that is capable of describing the cases that have been on the forms can be modeled.

By asking the project team what the range of possible values for each attribute on the forms is and inquiring what would have happened if one of the values on a form were different, a broad range of cases can be created and the vocabulary expanded in a short time. Discussions among the project team members raise the level of understanding of both the approach and the problems, and should be encouraged in this early phase. During initial knowledge acquisition, it is also advisable to have more cases on an "everyday" level rather than having a few extremely specific ones.

Homer Case Acquisition	
Problem Nr : 816	**Date:** 26.04.99
Author: S. Itani	**Verified by:** J. Fleisch
Problem Description (Failure)	Printer does print pages full of gibberish
Reason (Fault)	File is Postscript, Printer does not understand PS
Solution	Send File to Postscript Printer, delete file from queue
What did you check to find out what the problem was ?	
Printer Model	HP LJ 6L
File Type	Postscript
Other Notes:	The reverse of this problem did also happen, somebody sent a PCL file to a pure PS printer

Tab. 2. Sample form for initial case acquisition.

While the initial vocabulary is being created and value ranges fixed, questions regarding adaptation rules and similarities should be posed and the results entered into the system.

One of the major challenges one must face when creating a system to capture and represent the experience of domain experts, is determining the level of abstraction with which the domain and the knowledge will be modeled. If the model used is too simplistic, it will cause problems while the experience is being captured and will miss important details. If, however, the domain model is too specific, the user will get lost quickly in useless details, and knowledge acquisition will be very tedious and time consuming. Maintenance is very difficult for both a too-simplistic and a too-complex model.

The decision to use a structured domain model approach as opposed to a textual query-answer-based approach depends on the system's intended users. For inexperienced help-desk operators, a tool with which simple problems can be solved by answering a limited number of questions is of great value [18]. However, for experienced help-desk operators who would not bother to use a system for (subjectively) trivial problems, a structured domain model approach yields better results. The system will be able to present the not-so-obviously similar solutions that the help-desk operators could not find. Since knowledge contained in the domain model is used in similarity calculation, the retrieved solutions will be similar in a semantic and structural manner. The domain model allows the solutions in the case base to be applicable to a broader range of problems.

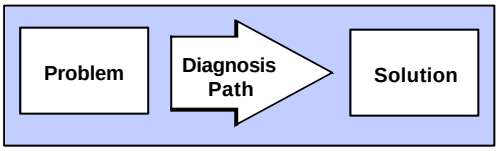

Fig. 2. Basic structure of a help-desk case.

The cases in the help-desk domain should be modeled in accordance with the approach the help-desk operators use in solving problems. We found the approach shown in Fig. 2 very suitable.

The *Problem Description* is the first information the help-desk operator gets from the end-user. This description is what the end-user subjectively perceives as the problem. It may or may not have to do with the actual cause of the failure.

The *Diagnosis Path* consists of the questions the help-desk operator must ask or the information he or she must obtain from various sources to arrive at a diagnosis. The diagnosis path contains the minimal amount of information that is necessary to diagnose the problem.

The *Solution* contains the fault, i.e., what caused the problem, and the remedy, i.e., how to solve the problem. Depending on how the system is implemented and what statistical information is needed for further evaluation, some additional, administrative data may also be added to the case description.

Each complete path from problem description to solution makes up one case.

Once the cases from the initial forms have been entered into the help-desk system, the system should be shown to the project group to verify the results it delivers.

Afterwards the initial knowledge acquisition can continue as more cases are entered from additional forms and the knowledge containers are incrementally updated.

Initial knowledge acquisition takes place in two steps. During the first, preliminary knowledge acquisition, the cases for the prototype of the case-based help-desk support system are collected. While the collected cases will help to initialize the knowledge containers and train the project team, the collection of the "core" cases for the system should be done in a second step, the core knowledge acquisition. Nevertheless, the approach that is used in both processes is similar.

6. Using the System

6.1. Managerial Processes During System Use

Project progress with respect to the qualitative and quantitative criteria selected as project goals must be monitored constantly during system development and use [cf. 9]. Regular project reviews should take place. Standard project planning and controlling techniques can and should be applied to case-based help-desk support projects.

Measuring the impact of the help-desk system on the efficiency and effectiveness of the target group (increase in first-call problem resolution, decrease in problem solution time, and so on) and making the results available to both the project and the target groups will motivate the help-desk operators to use the system and help uncover deficiencies.

6.2. Organizational Processes During System Use

Knowledge Utilization and Acquisition Process. The knowledge utilization and acquisition processes that have been defined during system development have to be enacted during system use. The use of the case-based help-desk support system contains the Application Cycle in which the system is used by the help-desk operator and the Maintenance Cycle in which the system is maintained by the CBR author and the CBR administrator (Fig. 3, section 6.3).

During the application cycle, the cases that are stored in the case-based help-desk support system are being used to solve problems. Even if no new cases are being acquired during this cycle, statistical data regarding the quality and usage of the cases (last retrieval time, last application date, success rate and so on) can be collected. This data can be used to determine the quality of the cases and for maintenance purposes.

Whenever a help-desk operator decides that the proposed solution is not appropriate, a new case has to be entered into the case base. However, since the quality of these cases varies according to the user entering them, they cannot be transferred to the case base without being verified by the CBR author. This is done in the maintenance cycle by the CBR author and the CBR administrator.

Training the Help-Desk Operators. Just as the test-users were trained during the project team training, the help-desk operators have to be introduced to the basics of CBR technology and the developed case-based help-desk support system. Since the

operators are going to participate in the continuous acquisition of knowledge, standards on how to store cases have to be introduced and taught. Feedback-channels also should be created and introduced during this training.

6.3. Technical Processes During System Use

Continuous Knowledge Acquisition and Maintenance. The knowledge contained in a case-based help-desk support system is an incomplete model of the domain in the real world. Whenever the real world changes, the model in the system has to be updated. The necessity for changes in the model may either arise from real changes in the world or from the learning effect associated with using the case-based help-desk support system. By learning, the system improves the model's coverage of the real world. Since the model is incomplete by definition, with growing knowledge, updates in the knowledge containers will be necessary.

While nobody would consider purchasing a database system with the assumption that it would continue to work without any maintenance at all, there seems to exist a misconception about knowledge-based systems in this respect. All concepts used for maintaining database systems are also applicable to knowledge-based systems. However, because of the semantics associated with the information in knowledge-based systems, additional maintenance operations are necessary. Learning and changes in the real world can make maintenance necessary for each knowledge container.

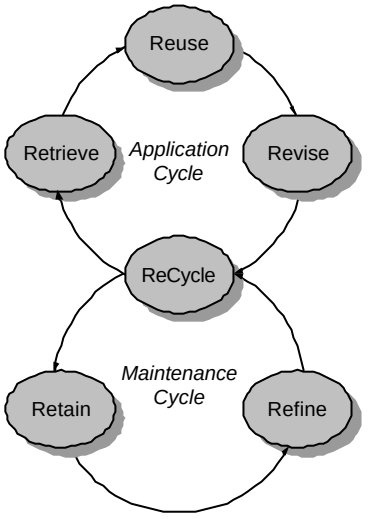

Fig. 3. Processed during the use of a case-based help-desk system

The utilization of a case-based help-desk support system comprises two linked process cycles: the *Application Cycle* and the *Maintenance Cycle* (see Fig. 3).

The *Application Cycle* takes place each time a user solves a problem with the case-based help-desk support system. During the application of the CBR system, the standard tasks *Retrieve*, *Reuse*, and *Revise* must be performed [6]. If the case solution generated during the reuse phase is not correct and cannot be repaired, a new solution has to be generated by the help-desk operator. The solution that has been retrieved by the system or created by the help-desk operator is put to use during the *Recycle* task. The *Application Cycle* is performed by the end-user of the system (help-desk operator).

Whenever a new solution is generated during system use, this case is stored in the case buffer, made available to all help-desk operators as an "unconfirmed case", and sent to the *Maintenance Cycle*. These steps as well as the maintenance cycle itself are not visible to the standard help-desk operator.

The *Maintenance Cycle* consists of the *Retain* and *Refine* tasks. While the *Application Cycle* is executed every time a help-desk operator uses the CBR system, the *Maintenance Cycle* can be executed less frequently, i.e., only when there is a need for maintaining the system or at regular intervals.

During the *Retain* task, the CBR author checks the quality of the new cases that were generated by the helpdesk operators and stored in the case buffer.

The CBR author verifies and approves the representation and content of each case. In terms of representation, the cases should

- contain the information that is necessary and sufficient to solve the problem,
- be described on an appropriate abstraction level.

The content is verified by checking whether the case is

- correct,
- (still) relevant, and
- applicable.

During the *Refine* phase, maintenance steps for the knowledge containers are performed by the CBR administrator. The case base, vocabulary, similarities, and adaptation knowledge have to be refined, and the potentially quality-decreasing effects of external changes in the domain, as well as the inclusion of new cases in the case base, have to be counteracted.

The goal of the *Refine* task with respect to the case base is to keep the case base correct, to have maximal coverage of the problem space, and to have no redundant cases. After each case has been validated in the retain task, their suitability for inclusion in the case base has to be determined.

Before a new case is taken into the case base, it must be checked to see

- whether it is a viable alternative that does not yet exist in the case base,
- whether it subsumes or can be subsumed by an existing case,
- whether it can be combined with another case to form a new one,
- whether the new case would cause an inconsistency, and
- whether there is a newer case already available in the case base.

The operations that have to be performed during case base maintenance vary depending on the application domain and the vocabulary that is used to represent the cases [cf. 12, 13, 14, 15, 16].

Both the inclusion of new cases and changes in the domain may have an effect on the validity and quality of the compiled knowledge containers (vocabulary, similarity, adaptation knowledge) as well. The maintenance of these containers is also performed in the *Refine* step. Since changes in the vocabulary can cause information in the cases to be no longer available or missing (e.g., attributes can be added and deleted, classes can be moved) maintenance of the vocabulary should be performed with utmost caution [cf. 17].

It should be noted that the refinement of the knowledge containers does not necessarily have to be triggered by external events but may also be performed through introspection. By analyzing the content of the knowledge containers, more efficient ways to structure the domain, adaptation rules, and similarities, as well as new cases, can be discovered or derived.

While maintenance operations for the case base can be performed by the CBR author, maintenance of the vocabulary, the similarity, and adaptation knowledge should only be performed by the CBR administrator.

General IT-System-Related Processes. Once the case-based help-desk support system has been put into operation, it has to be debugged, monitored, and updated continuously. The necessity for updates does not necessarily have to come from the help-desk system itself, but may also be initiated by changes in the (IT) environment. Since these processes are not CBR-specific but apply to IT systems in general, we refrain from going into their details here.

7. Summary

Current CBR-process models only cover the technological aspects of CBR system development. While the tasks given in these models suffice to develop systems that are used by a limited number of users in a static environment, problems that arise from larger user groups with differing levels of experience as well as dynamic domains are disregarded. Case-Based Reasoning in real world environments is not necessarily a low maintenance AI-technology and processes related to knowledge acquisition and maintenance play a very important role in the success of CBR projects in corporate environments.

In order to develop case-based help-desk systems that are being used in a dynamic, corporate environment by a large group of users, managerial, organizational and technical processes have to be taken into account. It has to be kept in mind, that once a CBR system is in place, continuous knowledge acquisition and maintenance is necessary. Processes for knowledge acquisition and maintenance have to be developed and put in place, and personnel has to be dedicated to perform these tasks.

8. References

1. Bergmann, R., Breen, S., Göker, M., Manago, M., Wess, S. "Developing Industrial Case Based Reasoning Applications: The INRECA Methodology.", Lecture Notes in Artificial Intelligence, 1612, Berlin, Springer Verlag, 1999
2. Göker M., Roth-Berghofer Th., Bergmann R., Pantleon T., Traphöner R., Wess S., Wilke W., *"The Development of HOMER: A Case-Based CAD/CAM Help-Desk Support Tool"*, Smyth B. & Cunningham P. eds., "Advances in Case-Based Reasoning, Proceedings of the Fourth European Workshop on Case-Based Reasoning EWCBR98 Dublin, September 23-25,1998", LNAI 1488, pp. 346-357, Berlin, Springer Verlag, 1998
3. Riesbeck C., Schank R., *"Inside Case-based Reasoning"*, Lawrence Erlbaum Associates, Publishers, Hillsdale 1989
4. Hammond K., *"Case-Based Planning-Viewing Planning as a Memory Task"*, Academic Press Inc, HBJ Publishers, San Diego, 1989
5. Kolodner J., "Case-based Reasoning", Morgan Kaufmann Publishers Inc, San Mateo, 1993
6. A. Aamodt, E. Plaza., *"Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches"*, AICOM Vol.7 Nr.1, pp.39-59, March 1994

7. Leake D., *"CBR in Context: The Present and the Future"* in Leake D. (ed.), "Case-Based Reasoning – Experiences, Lessons and Future Directions", pp. 3-30, AAAI press / MIT press, Menlo Park California, Cambridge Massachusetts, London, 1996
8. Kitano H., Shimazu H., *"The Experience Sharing Architecture: A Case Study in Corporate-Wide Case-Based Software Quality Control"* in Leake D. (ed.), "Case-Based Reasoning – Experiences, Lessons and Future Directions", pp. 235-268, AAAI press / MIT press, Menlo Park California, Cambridge Massachusetts, London, 1996
9. Stolpmann M. Wess S. *"Optimierung der Kundenbeziehungen mit CBR systemen-Intelligente Systeme für E-Commerce und Support"*, Addison Wesley Longmann (Business & Computing), Bonn, 1999
10. Richter M., *"Introduction"*, in Lenz M., Bartsch-Spörl B., Burkhardt H. D., Wess S. (Eds.), *"Case-Based Reasoning Technology, From Foundations to Applications"*, Lecture Notes in Artificial Intelligence Vol. 1400, pp.1-15, Springer-Verlag, Berlin, Heidelberg 1998. Also: Richter M., "The Knowledge Contained in Similarity Measures", Invited talk at ICCBR95, http://wwwagr.informatik.uni-kl.de/~lsa/CBR/Richtericcbr95remarks.html
11. Wilke W., Vollrath I., Bergmann R., "Using Knowledge Containers to Model a Framework for Learning Adaptation Knowledge", *ECML (European Conference on Machine Learning) Workshop, Prag, 1997*
12. Leake D., Wilson D., *"Categorizing Case-Base Maintenance: Dimensions and Directions"*, Smyth B. & Cunningham P. eds., "Advances in Case-Based Reasoning, Proceedings of the Fourth European Workshop on Case-Based Reasoning EWCBR98 Dublin, September 23-25,1998", LNAI 1488, pp. 196-207, Berlin, Springer Verlag, 1998
13. Smyth B., McKenna E., *"Modeling the Competence of Case-Bases"*, Smyth B. & Cunningham P. eds., "Advances in Case-Based Reasoning, Proceedings of the Fourth European Workshop on Case-Based Reasoning EWCBR98 Dublin, September 23-25,1998", LNAI 1488, pp. 208-220, Berlin, Springer Verlag, 1998
14. Surma J., Tyburcy J., *"A Study on Competence-Preserving Case Replacing Strategies in Case-Based Reasoning"*, Smyth B. & Cunningham P. eds., "Advances in Case-Based Reasoning, Proceedings of the Fourth European Workshop on Case-Based Reasoning EWCBR98 Dublin, September 23-25,1998", LNAI 1488, pp. 233-238, Berlin, Springer Verlag, 1998
15. Racine K., Yang Q., *"Maintaining Unstructured Case-Bases"*, Leake B. & Plaza E. eds., "Case-Based Reasoning Research and Development", Proceedings of the second International Conference on Case-Based Reasoning ICCBR-97 Rhode Island, July 1997, LNAI 1266, pp. 553-564, Berlin, Springer Verlag, 1997
16. Hüttemeister A., "Wartung einer Fallbasis", Diploma Thesis, University of Kaiserslautern, Department of Computer Science, February 1999
17. Heister F., Wilke W., "An Architecture for Maintaining Case-Based Reasoning Systems", Smyth B. & Cunningham P. eds., "Advances in Case-Based Reasoning, Proceedings of the Fourth European Workshop on Case-Based Reasoning EWCBR98 Dublin, September 23-25,1998", LNAI 1488, pp. 221-232, Berlin, Springer Verlag, 1998
18. Thomas, H., Foil R., Dacus, J. : "New Technology Bliss and Pain in a Large Customer Service Center", in: Case-Based Reasoning Research and Development, Proceedings of the ICCBR97, Leake, Plaza (eds.), , pp. 166-177, LNAI1266, Springer Verlag, Berlin, 1997

[i] Funding for this work has partly been provided by the Commission of the European Union (INRECA-II: Information and Knowledge Reengineering for Reasoning from Cases; Esprit Contract no. 22196). The partners of INRECA-II are: Acknosoft (prime contractor, France),

DaimlerChrysler (Germany), tecInno (Germany), Irish Multimedia Systems (Ireland), and the University of Kaiserslautern (Germany). http://www.inreca.org

Acknowledgements

The authors would like to thank Prof. M. Richter for triggering the ideas that led to this publication during the EWCBR 98 in Dublin.

We would also like to thank the reviewers of this paper for their encouraging and very helpful comments.