Industrial Application Development using Case-based Reasoning

Miroslav Sveda and Ondrej Rysavy

Brno University of Technology, Faculty of Information Technology, Bozetechova 2
CZ-612 66 Brno, Czech Republic

Abstract. Every design deserves decisions based on the application domain knowledge collected from previous similar implementations. The paper deals with stepwise development of a dedicated LAN-based industrial measurement application. The conception of this development stems from a knowledge preserving, graceful conversion of the original enterprise practice into a co-operative work supporting arrangement. The principal paradigm employed for this conversion is case-based reasoning augmented by rule-based support.

1 Introduction

Design decisions stem from the application domain knowledge that includes information about previous similar implementations. Such association with knowledge provokes to employ either an application domain expert or a dedicated knowledge-based system offering some support for the design of a novel application from scratch [1]. Another designing conception stems from the stepwise development of an original practice. In this case, the accumulated knowledge is maintained inside the budding system. Such approach can be successful when the subsequent changes are mild. On the contrary, a bigger development step requires utilizing some combination of both above mentioned conceptions [2]. This contribution presents a technique for reusing the amassed domain knowledge with an industrial measurement system whose architecture and service level were radically changed. Case-based reasoning represents its principal paradigm; besides, the overall strategy is augmented by rule-based reasoning and supporting mechanisms. Using this strategy, the knowledge-based subsystem can learn and follow the previously developed measurement processes. The original, manually controlled measurement process is carefully monitored by the subsystem in initial stages when the subsystem is actually taught by a practicing engineer. After that, the learned subsystem can continue almost autonomously, controlling not only the same but also the majority of similar measurements. The computer integrated measurement project for the test department of an electric motor manufacturer [4] offers an environment for the case study exemplifying this strategy.

Sveda M. and Rysavy O. (2007).
Industrial Application Development using Case-based Reasoning.
In *Proceedings of the 3rd International Workshop on Artificial Neural Networks and Intelligent Information Processing*, pages 76-82
DOI: 10.5220/0001615900760082

2 Application

The original measurement system for induction motors of various types and sizes consisted of testing stands equipped with control switchboards, load generators, and test sets comprising both analogous and digital devices. Testing engineers proceeded measurements for obtaining unloaded characteristics, load characteristics, short circuit characteristics, torque-speed curves, temperature rise tests, and lifetime tests; furthermore, these characteristics were evaluated on a mainframe, stored on magnetic tapes, and printed out in form of certificates. The task was to design a LAN-based system substituting the mainframe functions, propping the test department agenda, and enabling to implement an intended co-operative work support; nevertheless, all measurement procedures and documents had to retain compatibility with previously decreed status, and production measurement of motors could not be interrupted or delayed.

3 Target System Architecture

The system architecture can be briefly described in the following way--see Fig. 1. A test department backbone LAN interconnects test supervisor stations TSs (industrial PCs in the field site) and, in the office position, server S and test evaluator stations TEs (workstations). Moreover, it provides through a plant backbone various communications with other departments. Each test supervisor station TS can either manage its fieldbus connecting test controllers TCs of a testing stand, or perform gateway functions between the fieldbus and the LAN. Each test controller monitors or serves attached digital and/or analogous sensors and, conceivably, actuators.

All application software modules were considered to be implemented consecutively. At the beginning, while preserving original manually mastered measurement routines, the test controller tasks sample data from auxiliary outputs of the sensors and the sensed actuators and transmit these data to the test supervisor station, which only records the course of the measurement. Data records, which are augmented by time stamps, serve both for generating standard characteristics measurement outputs reassigned to a test evaluator station, and for collecting domain-specific knowledge in form of case histories. The auxiliary outputs become the main outputs--with possible replacement of some sensors and actuators--in the next stage of the development when the test supervisor task commands the course of each measurement test. In this case, the testing engineer inspects the tests from the test supervisor console. The final stage of the field operation development includes remote supervision of test series by a test evaluator station employing case-based reasoning and possible contingent field attendant interactions through a test supervisor console.

The development of the office-position application software consists in porting the original programs from the mainframe to the server, improving the graphical presentations and interactive behavior of these programs, implementing a new database of motor characteristics, and initiating a workgroup service support. That part of the project, implemented by a larger group of programmers, exceeds the scope of the presented paper.

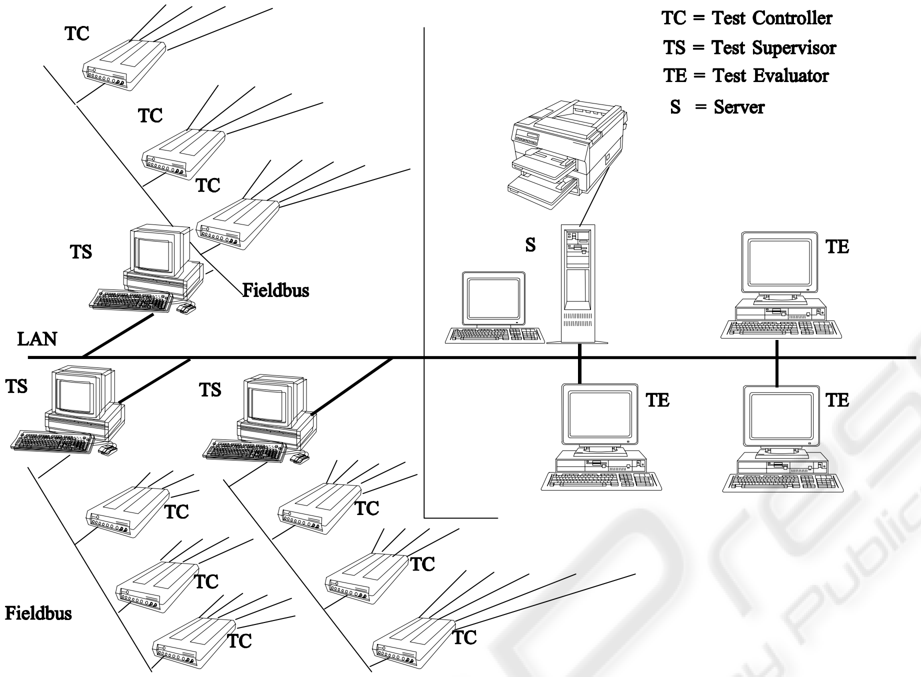

Fig. 1. Measurement system architecture.

4 Knowledge-Based Support

As mentioned earlier, the measurement process is controlled manually by experienced engineers in initial stages of the conversion. Concurrent learning process runs meanwhile collecting domain-specific knowledge about those manually controlled measurements. Knowledge-based subsystem traces expert's activities registering them with given tasks and inputs and received outputs. Thus, the learning process in initial stages facilitates a higher level control of the measurement process in later stages. So, experienced engineer can also be substituted by a novice then, without a significant worsening of the measurement quality. Case-based reasoning is the principal paradigm used for this purpose. Case-based reasoning appears to be a promising concept addressing successfully knowledge elicitation bottleneck of current knowledge-based systems [7]. Case-based reasoning differs from other rather traditional methods relying on case history. For a new problem, the case-based reasoning strives for a similar old solution. This old solution is chosen according to correspondence of the new problem to some old problem that was successfully solved by this solution. Hence, previous significant cases are gathered and saved in a case library. Case based reasoning is based on remembering the similar situation that worked in past. Elicitation means to collect those cases. Implementation represents identification of important features for the case description consisting of values of those features. A case-based reasoning system can only be as good as its case library [3]: only successful and sensibly selected old cases should be stored in the case library. The description of the

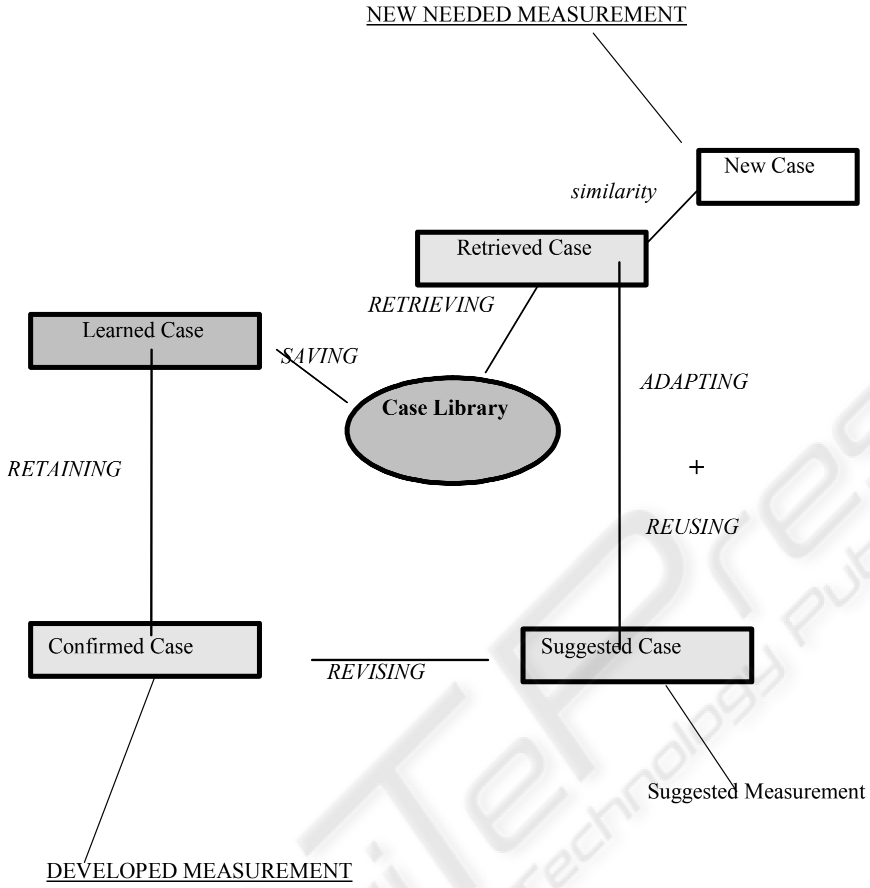

Fig. 2. Development of a new measurement procedure.

case should comprise the problem, solution of the problem, and any other information describing the context for which the solution can be reused. A feature-oriented approach is usually used for the case description.

Case library serves as a knowledge base of the case-based reasoning system. The system can learn by acquiring knowledge from the old cases. Learning is basically achieved in two ways: through accumulating new cases, and through the assignment of indexes. Solving a new case, the most similar old case is retrieved from the case library. The suggested solution of the new case is generated in conformity with this retrieved old case. Search for the similar old case from the case library represents important operation of case-based reasoning paradigm. Retrieval relies basically on two methods: nearest neighbor, and induction. The significant idea of the development of a measurement procedure is the following (Fig. 2):

- Reminding a user about earlier realized successful and verified measurement procedures in a similar context, i.e. retrieving the similar older measurement procedures from the case library;

- Facilitating reuse of the selected measurement process;
- Adaptation of the selected measurement to fit closer the new context; and
- Storing some selected measurement procedures for future reuse, after a positive evaluation process.

4.1 Case-based and Rule-based Reasoning

Case-based reasoning relies on the idea that situations are mostly repeating during the life cycle of an applied system. After some period, the most frequent situations can be identified and documented in the case library and, consequently, the case library can cover common situations. However, before the case-based paradigm can be adopted, it is necessary to control manually the measurement process for longer time, because case-based reasoning technique is not feasible at the very beginning with an empty case library.

When relying on the case-based reasoning exclusively, also the opposite problem can be encountered: after some period the case library can become huge and redundant. Registered cases represent clusters of very similar situations. Despite careful evaluation of cases before saving them in the case library, it is difficult to avoid that problem.

To compensate those insufficiencies, the case-base reasoning can be combined with rule-based support. This support has been gradually implemented. Rule-based reasoning should augment the case-based reasoning in the following situations:

- No suitable old solution can be found for a current situation in the case library and engineer hesitates about his own solution. So, rule-based module is activated. For a very restricted class of tasks, the rule-based module is capable to suggest its own solution. Once generated by this part of the framework, such a solution is then evaluated and tested more carefully. However, if the evaluation is positive, this case is later saved in the case library covering one of the gaps of the case-based module.
- Situations are similar but rarely identical. To fit closer the real situation, adaptation of the retrieved case is needed. The process of adaptation can be controlled by the rule-based paradigm, using adaptation procedures in the form of implication. Sensibly chosen meta-rules, i.e. knowledge about rules, can substantially improve the effectiveness of the system.

When relying on the case-based reasoning exclusively, also the opposite problem can be encountered: after some period the case library can become huge and redundant. Registered cases represent clusters of very similar situations. Despite careful evaluation of cases before saving them in the case library, it is difficult to avoid that problem.

4.2 Implementation Principles

For the discussed application, a case means a measurement task that consists of a sequence of measurements and its evaluation. Each measurement is described by a record containing the time stamp of the measurement and a set of attributes with their actual values sampled on inputs and outputs. There is a key parameter (either the time stamp or one item of the attributes) with prescribed values that initiate every measurement registering. The application domain knowledge consists in the definition of input/output attributes, key parameter including its measurement initiating values, output attribute values passage, and evaluation rules of input attribute value traces for a measurement task and a motor type. The evaluation usually takes a form of checking some relations among selected attributes.

5 Conclusions

Case-based reasoning, partially augmented by rule-based reasoning, has been adopted for measurement process support in the test department of an electric engine manufacturer. The chosen knowledge-based aid of the computer-based measurement system offers to preserve highly practical knowledge and experience about complex measurement procedures. Especially the case-based reasoning shows inspiring results. Role of rule-based reasoning is rather marginal. Some items of this comparison for the given application can be restated as follows:

- Case-based reasoning can work perfectly in the same or similar situation. Granularity of its knowledge base (i.e. case library) is rather coarse, with large chunks of knowledge. On the contrary, rule-based approach builds a solution gradually from a number of rather fine rules so that cases usually cover the solution entirely.
- Another advantage of the rule-based reasoning can be potentially optimal measurement procedure. On the other hand, case-based reasoning paradigm tends to rather suboptimal solutions. This paradigm relies on old and well-tried measurement procedures, updating an old solution only slightly for the purpose of adaptation.
- However, potentially rapid implementation is one of the most important advantages of the case-based reasoning. While implementation of a rule-oriented system can last years, the implementation of a case-oriented approach can be planned rather in months.
- Moreover, a needed involvement of an expert is restricted and easier for case-base reasoning paradigm. Main sources of knowledge are former measurement procedures. For this reason, substantial part of the knowledge acquisition can be automated.
- Expert cooperates on measurement description, selecting the most significant features for the description of this description and the consequent search. Expert also feels better discussing the concrete measurement procedures in the form cases, than developing general rules.

- Case library is more transparent than rule-oriented knowledge-base. Maintenance of this library is easier.

These and other factors suggest to emphasis the future development of the case-based paradigm for the similar engineering applications. Role of the rule-based support will remain restricted for the support improving the overall procedure, as mentioned above.

Acknowledgements

This paper is based on the work initially published in [5]. The knowledge-based support for that application was studied by Otakar Babka from University of Macau, Macau, Jana Freeburn from Fort Lewis College, Durango, CO, U.S.A., and the author, see [6]. The current paper restates lessons learned throughout that research while stressing more general experience utilizable for similar application domains. This work has been supported by the Czech Ministry of Education in the frame of Research Intention MSM 0021630528: Security-Oriented Research in Information Technology, and by the Grant Agency of the Czech Republic through the grants GACR 102/05/0723: A Framework for Formal Specifications and Prototyping of Information System's Network Applications and GACR 102/05/0467: Architectures of Embedded Systems Networks.

References

1. Alippi, C., Ferrero, A., Piuri, V.: Artificial Intelligence for Applications. IEEE Instrumentation and Measurement Magazine, Vol. 1, No. 2 (1998) 9-17
2. Frakes, W.B., Kang, K.: Software Reuse Research: Status and Future. IEEE Transactions on Software Engineering, Vol. 31, No.7 (2005) 529-536.
3. Kolodner, J, Case-based Reasoning. Morgan Kaufmann, San Mateo, CA, USA, (1993)
4. Machacek, J., Novotny, V., Opletal, V., Sveda, M.: Proposal of Hardware, Software, and Communication Interconnection of Automatic Measurement System for Asynchronous Motors (In Czech). Technical Report OZ 3342, VUES Brno (1991)
5. Sveda, M.: Design and Development of Industrial Measurement System--Architecture and Software. Microprocessing and Microprogramming, Vol. 40, No. 10-12 (1994) 887-890
6. Sveda, M., Babka, O., Freeburn, J.: Knowledge Preserving Development: A Case Study. IEEE Proceedings of International Conference and Workshop ECBS'97. Monterey, California, IEEE CS Press, Los Alamitos, California (1997) 347-352
7. Watson, I., Marir, F. Case-based Reasoning: A Review. The Knowledge Engineering Review, Vol. 9, No. 4. (1994)