An Architecture for Multiple Heterogeneous Case-Based Reasoning Employing Agent Technologies.

Elena I Teodorescu and Miltos Petridis[1]

Abstract. This paper presents an investigation into applying Case-Based Reasoning to Multiple Heterogeneous Case Bases using agents. The adaptive CBR process and the architecture of the system are presented. A case study is presented to illustrate and evaluate the approach. The process of creating and maintaining the dynamic data structures is discussed. The similarity metrics employed by the system are used to support the process of optimisation of the collaboration between the agents which is based on the use of a blackboard architecture. The blackboard architecture is shown to support the efficient collaboration between the agents to achieve an efficient overall CBR solution, while using case-based reasoning methods to allow the overall system to adapt and "learn" new collaborative strategies for achieving the aims of the overall CBR problem solving process.

1 Introduction

Case-based reasoning (CBR) is now an established artificial intelligence paradigm. Given a case-base of prior experiences, a CBR system solves new problems by retrieving cases from the case-base, and adapting their solutions to comply the new requirements[1].

Multiple Case Based Reasoning (MCBR) is used to retrieve solutions for a new problem from more than one case-base. Methods for managing sharing of standardized case bases have been studied in research on distributed CBR (e.g. [13]), as have methods for facilitating large-scale case distribution [10]. Leake and Sooriamuthhi propose a new strategy for MCBR - an agent selectively supplements its own case-base as needed, by dispatching problems to external case-bases and using cross-case-base adaptation to adjust their solutions for inter-case-base differences [4, 5, 6,13].

In many problems in modern organisations, the knowledge encapsulated by cases is contained in multiple case bases reflecting the fragmented way with which organisations capture and organise knowledge. The traditional approach is to merge all case bases into a central case base that can be used for the CBR process. However, this approach brings with it three challenges:

- Moving cases into a central case base potentially separates from its context and makes maintenance more difficult.
- Various case bases can use different semantics. There is therefore a need to maintain various ontologies and mappings across the case bases.
- The knowledge content "value" of individual cases can be related to its origination. This can be lost when merging into a central case base.

Keeping the cases distributed in the form of a Heterogeneous Multiple Case Based Reasoning system (HMCBR) may have a number of advantages such as increased maintainability and competence and the contextualisation of the cases. Past research at Greenwich [2][3] has shown the need to combine knowledge encoded in cases from various heterogeneous sources to achieve a competent, seamless CBR system.

Ontanon and Plaza [7] looked at a way to "improve the overall performance of the multiple case systems and of the individual CBR agents without compromising the agent's autonomy". They present [8] a framework for collaboration among agents that use CBR and strategies for case bartering (case trading by CBR agents). Nevertheless, they do not focus at the possibility of cases having different structures and what impact this will have on applying CBR to heterogeneous case bases. Leake [5] states that "An important issue beyond the scope of this paper is how to establish correspondences between case representations, if the representations used by different case-bases differ."

Given several case bases as the search domain, it is very likely that they have different structures. Ideally, accessing Multiple Case Bases should not require a change to their data structures. In order for an MCBR system to effectively use case-bases that may have been developed in different ways, for different tasks or task environments, methods are needed to adjust retrieved cases for local needs.

Leake and Sooriamurthi [4] proposed a theoretical "cross-case-base adaptation" which would adapt suggested solutions from one case base to apply to the needs of another. They are currently exploring sampling methods for comparing case-base characteristics in order to select appropriate cross-case-base adaptation strategies.

2 Adaptive CBR

In order to enable effective solution retrieval across autonomous case bases with differing structures, it is essential to have access and a good understanding of each of the different case base structures involved. This would make it possible to identify the commonalities, equivalences and specific characteristics of every case base associated with the system.

2.1 The process of adaptive CBR

Instead of trying to adapt the suggested solutions from one case base to the needs of another, the approach investigated in this study will be to create a "dynamic structure" of a general case. This dynamic structure would be modified every time a new case base with a new structure is added.

The process of adaptive CBR, within the architecture of the HMCBR System (Figure 1), will incorporate a number of steps.

Firstly, in order for the system to work with a particular case base, it will need to know the structure of that case base. Every newly added case base will therefore have to publish its structure to a Registry System. The published structures are required to have their own data dictionaries attached to enable the creation of a dynamic Data Dictionary.

[1] Department of Computing Science, University of Greenwich, Park Row, London SE10 9LS email:{ E.I.Teodorescu , M.Petridis}@gre.ac.uk

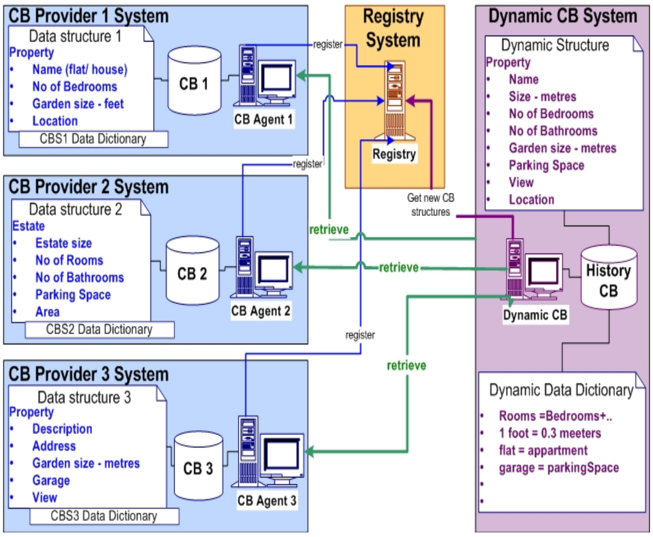


Fig. 1. The Architecture of the HMCBR System

The published structure will be retrieved by the Dynamic CB System and used to adapt the local dynamic structure to accommodate any new elements and map existing ones.

When the dynamic structure reflects all participating case bases, a case query can be submitted. The system would then reformulate the target case structure into each provider's case base structure. The target case structure will be a subset of the dynamic structure.

The reformulated cases are submitted to each provider and solution cases are retrieved using KNN techniques [1]. The structures of these solutions will be translated into the dynamic structure, thus creating a dynamic case base. Finally, the system will apply the classical CBR process to the dynamic case base.

The whole process is intended to provide a transparent view of the CBR process across the heterogeneous system.

2.2 Case Study

This case study requires searching for a property from three estate agencies without amalgamating their case bases structures.

Let us suppose that the estate agencies have different case base structures (figure 2).

A possible buyer should be able to search for a property and get all the suitable solutions from all three agencies. A search should retrieve the best matches from all case bases as if it was dealing with a single case base in a way transparent to the buyer.

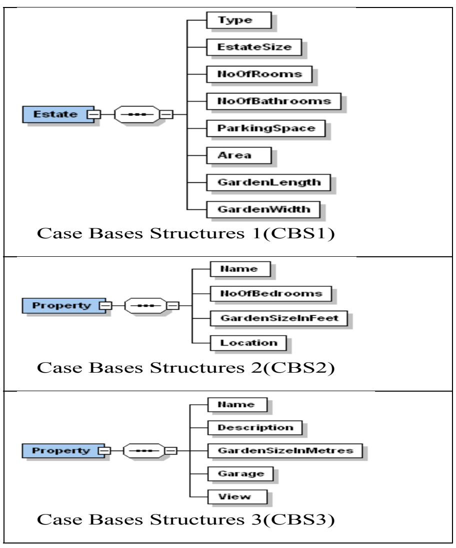


Fig. 2. Three different Case Base Structures

3 Creating the Dynamic Structure

Creating and maintaining a dynamic structure makes the self-adaptive multi case base reasoning system possible. By adding a new case base to the existing ones, new attributes are added to a global dynamic structure and new relations linked to these attributes are established.

CBS1. type / DCBS name	Apartment	Studio	Detached house
House	0	0	1
Flat	1	0.8	0

Fig. 3. Data Dictionary includes relations between some of the attributes.

A data dictionary is required to keep all the metadata for the dynamic structure. This data dictionary would have multiple functions: It records the location and the name of every attribute from the Case Base Structures (CBS) and how these are translated into the Dynamic Case Base Structure (DCBS). It also stores the type and any default value for every single attribute.

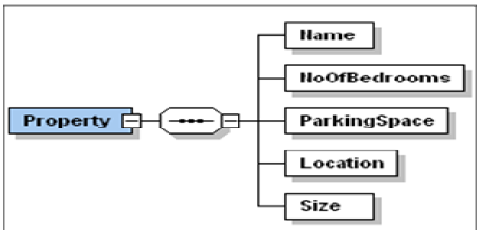
The Data Dictionary will reflect any relationships between the Dynamic Case Base Structure attributes. These relationships can be mathematical relationships or look-up tables (figure 3).

We will use the presented case study to show how a dynamic structure is created and how it is continuously changed by adding new case bases to the search domain.

Let us suppose that our general structure (the initial state of the Dynamic Structure containing few main attributes of a property) is already built (see figure 4). The structure has attached a basic Data Dictionary mainly containing the data types of the existing attributes.

We will show how this initial structure will be dynamically changed by consecutively adding the three agents to the search domain.

Adding the Case Base Structure 1 to the system implies mapping of the attributes ParkingSpace, Area and Type into the Dynamic Structure (these attributes are already existing in the initial structure) and also adding more attributes to it (i.e. NoOfRooms, NoOfBathrooms, GardenLength, GardenWidth)



Data Dictionary
Size: Double
NoOfBedrooms: Integer
Location: String
ParkingSpace: double

Name:	house	flat
house	1	0
flat	0	1

Fig. 4. Initial state of the Dynamic Structure and Data Dictionary

The Data Dictionary will reflect the mapping of attributes:

CBS1.ParkingSpace = DCBS.ParkingSpace;
CBS1.Area = DCBS.Location
CBS1.type= DCBS.name

The following attributes will be added to the dynamic data dictionary:

NoOfRooms: integer;
GardenLength: double; GardenWidth: double

Any other relevant relationships such as look-up tables for defining mappings between the values of attribute Type of CBS1 and the values of the attribute Name of the dynamic structure will be captured.

Case Base Structure 2 will add another attribute, GardenSize, to the Dynamic Structure and the data dictionary will record mapping of attributes:

CBS2.Name = DCBS.Name,
CBS2.Location = DCBS.Location ,
CBS2.NoOfBedrooms = DCBS. NoOfBedrooms;

The mathematical relationships are recorded:
DCBS.GardenSize = DCBS.GardenLength * DCBS.GardenWidth,

Functions can be applied, for example to keep the same metric system:

DCBS.GardenSize= CBS2.GardenSizeInFeet/$(3.281)^2$

The Data Dictionary would also include a look-up table showing the conversion of values of CBS2.Name to values of DCBS.Name.

Attention has to be paid to the meanings of the names of the attributes. For example, if the attribute "Type" in CBS1 and the attribute "Name" in CBS2 have the same meaning (they would be translated as "Name" in DCBS, with values found in a look-up table), the attribute "Name" from CBS3 has not the same meaning as the one from CBS2. It is actually translated into DCBS.Location (similar to CBS2.Location)

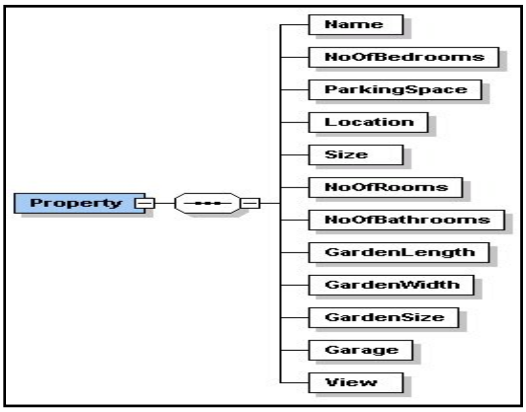


Fig. 5. Adapted Dynamic Structure after CBS3 was added

By adding the third estate agent case base to the search domain, the dynamic structure will grow even more (see figure 5) and the Data dictionary will reflect it by adding the attributes DSBS.Garage and DSBS.View.

The following attributes are mapped:

CBS3.Name = DCBS.Location
CBS3.Description = DCBS.Name
CBS3.GardenSizeInMeters = DCBS.GardenSize

Another look-up table can be created and added to the Data Dictionary to record the relationship between the Garage and ParkingSpace. Figure 6 shows the state of the Dynamic data Dictionary after CBS1, CBS2 and CBS3 are added.

Dynamic Data Dictionary
CBS1.Area = DCBS.Location
NoOfRooms: integer
CBS1.type= DCBS.name
DCBS.GardenSize: double
DCBS.GardenSize = CBS2.GardenSizeInFeet
DCBS.GardenSize = DCBS.GardenLenght *
DCBS.GardenWidth ...
CBS3.Name = DCBS.Location
CBS3.GardenSizeInMetres = DCBS.GardenSize

	Garage	ParkingSpace
Garage	1	0.7
ParkingSpace	0.7	1

Fig. 6. Adapted Dynamic Data Dictionary after CBS1, CBS2 and CBS3 are added

4 Optimising the agent collaboration process

In order to optimise the process of collaboration between the agents to achieve an efficient solution from the overall CBR process when applied across the heterogeneous case bases, an overall similarity metric is required. Additionally, an overall process to enable collaboration between the agents is necessary based on a flexible architecture to enable this collaboration.

4.1 Defining an overall similarity metric

The overall similarity metric between a target and a source Case can be defined as:

$$\sigma(C_T, C_s) = \sigma_{CBy}(C_T, C_s) * \omega_{CBy}(C_T) \qquad (1)$$

where:
σ: overall similarity
σ_{CBy}: similarity from case base provider CBy
C_T: target case
C_S: source case
$\omega_{CBy}(C_T)$: weighting for a case base provider y for case C_T

To allow for defining locally optimised similarity metrics for different providers, the following metric can be defined:

$$\sigma_{CBy}(C_T, C_s) = \sum_x \omega_{CBy}(x) * \sigma_{CBy}(C_T, C_s, x) \qquad (2)$$

where:
$\omega_{CBy}(x)$: the weighting from case base provider CBy for attribute x
$\sigma_{CBy}(C_T, C_s, x)$: the local similarity metric for provider CBy for attribute x.

This extended similarity metric takes into account the level of trust that the HMCBR system attributes to the competence of each case base provider. The level of trust is determined by applying CBR to the case-base of the history of queries. Additionally it allows to adjust the trust to particular providers to different "regions" in the case base allowing for case base providers to be "specialised" on particular types of domain knowledge. Finally, the extended metric allows for different ways of defining similarity based on possible particularities pertaining to individual case base providers.

Let us assume that in our case study the third estate agent is specialised in city apartments. After a few searches for country side houses with gardens, reasoning can be applied to the History case-base. Results will show that, for this particular query, the estate agent's level of trust is not high, i.e. there will be less solutions for this particular case base added to the Dynamic case-base.

A global level of trust of a provider's case-base can be calculating taking in consideration the results of all the previous enquiries for that provider.

4.2 An architecture and process to support effective collaboration between case base agents

The architecture of the HMCBR system shown in figure 1 contains the dynamic CB system, which incorporates a blackboard architecture. Blackboards have been used very effectively in the past for the construction of hybrid and agent based AI systems [11], [12].

The dynamic CB system is where the process for agent collaboration is controlled. It is based on a blackboard architecture incorporating the blackboard containing the target and retrieved cases from various providers together with similarity calculations and rankings. The blackboard also contains a log of the solution process and the reconciliation strategy followed, thus representing the state of the overall CBR solution process at any point in this process. Figure 7 shows the structure of the dynamic CB module incorporating the blackboard architecture.

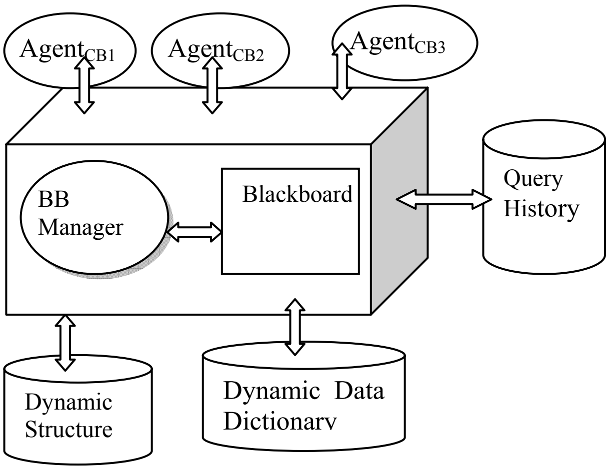


Fig. 7. The Dynamic CB system incorporating the blackboard architecture

The blackboard manager manages the overall solution process, communicates with and keeps track of the CB agents, selects and implements a solution strategy and monitors and evaluates the solutions achieved. Given a new target case, the blackboard manager decides on strategy for finding similar cases from the CB providers. The blackboard system decides which CB providers to use and the number of cases to retrieve from each one and other requirements, such as the requirement for diversity, similarity thresholds etc. The system then initialises the agents and assigns to them a mission. On return, the results (cases) are mapped using the dynamic data dictionary and written to the blackboard. A "global" CBR process is used to decide on the retrieved cases. The system then selects and presents the shortlisted cases after the reconciliation process and provides these to the user, together with links to their original forms for the user to explore. Finally, the system "reflects" on the process by updating the query history and confidence weights for each provider.

The system described here has been implemented and tested on a set of case bases from three different estate agent case bases, all using different structures. Experiments with the system have shown that the system can retrieve useful cases combining cases from all case bases to provide a more efficient overall solution when compared to using the case bases separately or mapping them to one central case base. Additionally, the system has shown that it can provide a more diverse retrieved case population in both cases. A full scale evaluation of the system, including using a different application domain is under way.

5 Conclusion

At a time of increasing web-based communication and sharing of knowledge between organisations and organisational units within enterprises, heterogeneous CBR applied to Multiple Case Bases seems to be the natural progression in this area of research.

The paper investigates an approach based on agents operating on different structures/views of the problem domain in a transparent and autonomous way. In this approach all data is kept locally by each case base provider in its native form. Agents can be dynamically added to the system, thus increasing the search domain and potentially the competence and vocabulary of the system.

This research proposes a new architecture for a self-adaptive MCBR system which involves the use of a dynamic structure based on the blackboard architecture. The Dynamic Structure reflects all participating case base provider structures. As new agents are added to the system, their case base structure is published and is used to adapt the Dynamic Structure accordingly.

The Dynamic Structure is used at runtime to translate search queries into the local structures of each agent. Each agent can then use the translated query to match it to its local cases and retrieve the best matches.

A Data Dictionary is created in order to manage the Dynamic Structure. This contains the metadata for the Dynamic Structure, such as mapping details of the case base provider's structures to the Dynamic Structure, type information and relationships between attributes of the dynamic structure.

The dynamic case base system manages the overall process, including controlling the agents, reconciling and optimising the retrieved cases and feeding back into its strategy by continuously adjusting weights representing confidence levels on individual case base providers. A prototype system to evaluate the efficiency of using a heterogeneous Multiple Case Based Reasoning system is currently being evaluated. Preliminary findings are encouraging.

Further work will concentrate into optimising the process of collaboration between the agents and methods and strategies for the reconciliation of retrieved cases.

References

[1] Kolodner, J.: Case-based Reasoning, Morgan Kaufmann, 1993

[2] Knight B, Petridis M, Mileman T: Maintenance of a Case-Base for the Retrieval of Rotationally Symmetric Shapes for the Design of Metal Castings LNAI 1998: Springer Verlag. Advances in Case-Based Reasoning, 5th European Workshop, Trento, Italy, Sept. 6-9.2000 pp 418-430

[3] Knight B, Petridis M, Mejasson P, Norman P: A Intelligent design assistant (IDA): a CBR System for Materials Design Journal of Materials and Design 22 pp 163-170, 2001

[4] David Leake & Raja Sooriamurthi: "Automatically Selecting Strategies for Multi-Case-Base Reasoning" - Proceedings of the Sixth European Conference on Case-Based (ECCBR) , Aberdeen, Scotland, September, 2002

[5] David Leake & Raja Sooriamurthi : "Managing Multiple Case Bases: Dimensions and Issues" (Proceedings of the Fifteenth Florida Artificial Intelligence Research Society (FLAIRS) Conference , Pensacola, Florida, May 2002),

[6] David Leake & Raja Sooriamurthi: "When Two Case Bases are Better Than One: Exploiting Multiple Case Bases" (Proceedings of the Fourth International Conference on Case-Based Reasoning (ICCBR) , Vancouver, Canada, July, 2001).

[7] Enric Plaza, Santiago Ontañón: Cooperative Multiagent Learning. Adaptive Agents and Multi-Agents Systems 2002

[8] Santiago Ontañón, Enric Plaza: A bartering approach to improve multiagent learning. AAMAS 2002

[9] Francisco J. Martín, Enric Plaza, Josep Lluís Arcos: Knowledge and Experience Reuse Through Communication Among Competent (Peer) Agents. International Journal of Software Engineering and Knowledge Engineering 1999

[10] Hayes, C., Doyle, M., Cunningham, P.: Distributed CBR Using XML, Proceedings of the Fourth European Conference on Case-Based (ECCBR) , Dublin, Ireland, June 1998

[11]R Engelmore, T Morgan : Blackboard Systems, Addison-Wesley, 1988

[12] Petridis M, Knight B, (2001): "A blackboard architecture for a hybrid CBR system for scientific software" in Proceedings of the Workshop Program at the 4th International Conference in Case-based Reasoning ICCBR01, Vancouver 2001, Eds R. Weber, G. von Wagenheim, pp 190-195, NCARAI, Naval Research Laboratory, Code 5510 Washington DC.

[13] Case Dispatching versus Case-Base Merging: when MCBR matters: David Leake and Raja Sooriamurthi. International Journal on Artificial Intelligence Tools: Architectures, Languages and Algorithms (IJAIT). Special issue on recent advances in techniques for intelligent systems. Vol 13, No 1, 2004