Perera, R.S., and Watson, I. (1996), "Multi-Agent Collaborative Case-Based Estimating and Design in NIRMANI: Organising a Multi-Perspective Case Memory", Kumar, B. (Ed.), Information Processing in Civil and Structural Engineering Design, pp. 53-64, Civil-Comp Press, Scotland, U.K.

Multi-Agent Collaborative Case-Based Estimating and Design in NIRMANI: Organising a Multi-Perspective Case Memory

Perera, R.S. & Watson, I.
AI-CBR, University of Salford
www.ai-cbr.org

Abstract

IT solutions to problems in construction design need to consider the perspectives of all the experts (or agents) involved; only then can IT provide a platform for integration. It is expensive to solve similar construction design problems from first principles. However, in practice designers and estimators reuse previous designs and estimates. The research described in this paper examines issues involved in the integration of many disciplines in construction with the use of Case-Based Reasoning techniques for the reuse of designs and estimates. It describes a perspective-based hierarchical case memory structure and a series of indexing methods for multi-stage retrieval. Construction projects are organised in to domain cases consisting of a hierarchy of meta, perspective and sub cases. Design and estimating cases selected for reuse are adapted with the use of sub-cases and domain specific adaptation rules. A prototype system NIRMANI (meaning designer in "Sinhalese") is currently being implemented encapsulating the case memory organisation and retrieval strategies described in this paper. This multi-agent system provides a common platform for a design team to support collaborative design.

1. Introduction :Finding IT solutions for Construction Problems

Design in construction involves many expert collaborators or agents (e.g., Architects, Quantity Surveyors, Structural Engineers, Service Engineers etc....). The product, the final design of a building is the result of the collaborative effort of these individuals. These experts have different perspectives of the product, the building [1]. Take for example the office area in a warehouse building. For an Architect, this is a functional space with aesthetic considerations whilst for a Quantity Surveyor it is an element of cost that is dependent on specification (quality and quantity). Similarly, in the eyes of the Structural Engineer, the same floor area is a structural element supported by beams, columns and foundation. Therefore, IT systems that provide solutions to design problems in construction should support multiple perspectives.

The construction industry is characterised with many problems related to the fragmentation of the industry. Some problems faced by the industry which this research addresses are given below.

1. The economic size and complexity of a building needs many specialists to create the final product.
2. This also causes difficulties in obtaining client requirements because the complexity causes the client difficulties in understanding the design by the client [2].
3. Cost control is also another problem that is caused by the complexity and the diverse interests of the design team [3].
4. Comparative design evaluation is expensive and results in sub optimal designs and premature bias to designs [4].
5. Construction problems are often caused by inadequate design [5].

Knowledge Based Systems (KBS) have been developed to provide solutions to construction problems. However, these have enabled automation of processes to a certain degree but have failed to address the key issue of fragmentation. IT solutions developed in the past have created *"islands of automation"* [6, 2] and have failed to achieve integration or collaboration in design. For example, ELSIE [6], and EMMY [7] have partially automated the cost estimating process. These provide an estimate without design. Similarly, rule-based and case-based systems such as HI-RISE [8], ARCHIE [9], CADRE [10], ADA [11], etc. support architectural design or structural design in isolation without considering the cost implications of the design. We believe that IT solutions to design problems must consider all the perspectives of the design problem.

Additionally, KBS development faces many problems such as ;

1. Knowledge elicitation is a very difficult process, often being referred to as the *"knowledge elicitation bottleneck"* ;
2. Implementing a KBS is a difficult process requiring special skills and often taking many years ;

3. Once implemented model-based KBS are often slow and are unable to access or manage large volumes of information;
4. Once implemented KBS are difficult to maintain ;
5. KBS systems cannot be developed where it is difficult to establish a domain model (e.g., in design).

Case-based reasoning (CBR) addresses all these issues. But, CBR has its own fair share of problems. Some of the problems with CBR that this research addresses are as follows:

1. Case memory management of large complex problems remains difficult. In complex domains such as in construction, it is difficult to accommodate multi-perspectives of the same data space. In such situations problems may arise in similarity assessment if cases are structured as large flat cases.
2. In complex domains such as construction cases are often *context* dependent; i.e. the semantic structural similarity of a case is not the only factor that must be considered. For example, a design case representing a warehouse in an urban area has a completely different context to a warehouse in a large industrial park. Therefore, an estimate for the latter cannot be directly derived from the former even if both the designs were exactly the same.
3. Where multi-agent, multitask case libraries are used, it is difficult to retrieve cases from such multiple cases since this type of retrieval may cause conflicts.
4. CBR systems often have problems of accessing data directly from existing databases (i.e. problems of data integration).
5. Case adaptation remains a problem with most CBR systems. Cases need adaptation that is highly knowledge intensive.

The research described in this paper tries to achieve collaborative design through a multi-agent case-based design and estimating system. This paper first briefly describes the CBR process and its application to estimating and design. It then proposes a conceptual model for collaborative design. Finally, it describes the organisational structure of a case memory for multi-agent case-based estimating and design and describes several retrieval strategies that facilitate collaborative design.

2. CBR And Its Application To Estimating and Design

2.1 An Overview of the CBR Process

CBR is a rich research paradigm within AI which addresses research agendas of philosophical issues and technological or practical issues. It has grown out of psychological models of episodic memory and the technological impetus of AI. CBR provides both a methodology for building systems and a cognitive model of how people solve problems [12, 13]. CBR has been defined as follows [14] :

"Case-Based Reasoning is the process of solving new problems by adapting solutions that were used to solve old problems."

At the highest level of generality CBR may be described as a process model with four processes [15] as in Figure 1 [16]

A CBR system retrieves a suitable case from the case library by matching indexes established for the new case (problem or target case). The information and knowledge in the retrieved case is then used to provide an initial solution to the problem posed. When it does not fully satisfy the problem specification the retrieved case is adapted using domain rules, heuristics and/or human intervention. The adapted case should then be evaluated and criticised to assess the suitability of the solution [17].

2.2 Case-Based Design And Case-Based Estimating

2.2.1 Case-Based Design (CBD)

Design is a task where an explicit domain model does not exist or is not yet adequately understood.

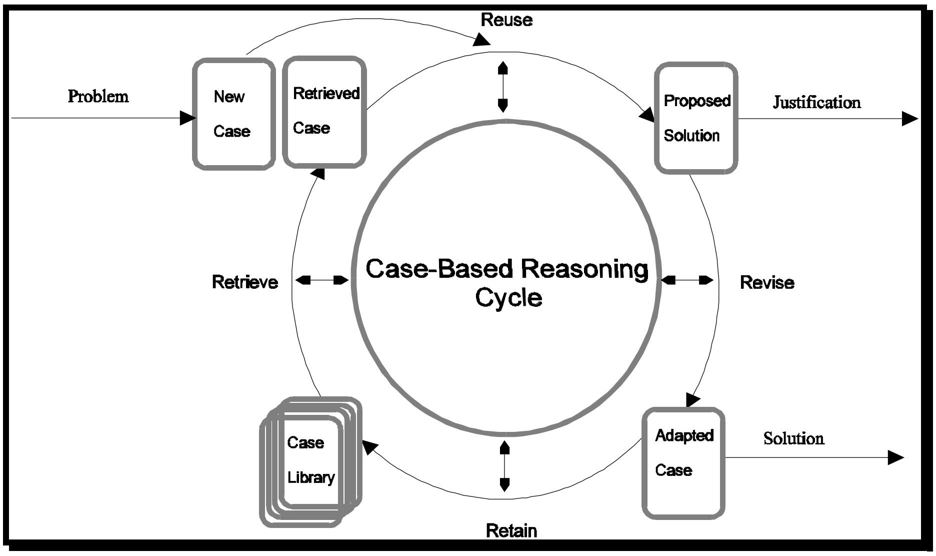

Figure 1 The CBR Cycle / Process Model

It is an ill-structured task [18] where design experience and heuristics play an important role [19]. Kolodner [13] describes design as a process of constraint satisfaction in creating an artefact that performs a certain function or fulfils a need. In most design problems these constraints may be under specified. Thus, there are usually many solutions to any given design problem. Sometimes all constraints may not be able to be satisfied which means a compromise is required. Moreover, constraints often cannot be considered in isolation.

During the design process Architects use parts or all of previous architectural [19, 20, 21]. In case-based design, the designer is offered previous solutions to a similar problem [21, 22], indicating how a previous combination of constraints were handled [13]. This process of using previous designs in the creation of new designs is called Case-Based Design (CBD). Maher & Balachandran [23] describes CBD as a hybrid approach as it uses specific design cases in conjunction with generalised or compiled knowledge.

Hence, CBD can be defined as ;

“The process of creating a new design solution by combining and/or adapting previous design solution(s).”

2.2.2 Case-Based Estimating (CBE)

Cost estimating is the prediction of the cost of an artefact, process or project either by using experience and/or a methodology. In the construction industry, there are several methods of estimating; for example, the unit method, storey enclosure method, elemental estimating. These methods are highly dependant on the experience of the estimator and cost data derived from previous construction projects.

Cost planning primarily uses the elemental estimating method for cost estimation [24]. This method uses similar cost analysis of projects (Elemental Cost Analysis - ECA) as the basis for estimation. The elemental rates are adjusted or adapted for quantity, quality and price (tender, market conditions, location etc.) [25, 26]. In other words, using a previous case (previous ECA) as the basis of the new estimate. Elemental estimating is, therefore, a classic example where CBR is applied by people to solve a problem.

Therefore, Case-Based Estimating (CBE) can be defined as:

“The process of estimating the cost of a new artefact, process or project using previous estimates or cost analyses.”

Surprisingly, there are very few examples where CBR has been used for estimation in any domain. FACE - Finding Analogies for Cost Estimation [27] is a system for estimating cost of software projects. It uses a case-base of previous estimates and a general global database for adaptation. Retrieved case are adapted using an analogy based algorithm to provide the new estimate. The system is implemented using CBR-Express™ & ART IM™. **Estor** [28, 29] is another system used for software effort estimation while CBR systems that help engineers to prepare bids in manufacturing have been reported [13, 30, 31, 32].

3. NIRMANI : A Multi-Agent Case-Based Collaborative Estimating & Design System

3.1 The Conceptual Model

The NIRMANI system (schematically described in Figure 2) generates a schematic design for Light Industrial Buildings (LIB) by retrieving previous design solutions that match the problem specification [16, 33]. The retrieved design will be adapted if required for architectural, structural and services requirements. Depending on the extent of adaptation of the design, costs for the chosen case (design) will be adapted providing a schematic design and a elemental cost plan for the building. The cost plan acts as a budgetary guide for further design development. The entire case-based estimating and design process is interactive giving the design team authority to guide the design process and to encourage creativity.

The stages of the case-based estimating and design process are briefly explained below. The NIRMANI system starts a session using an initial client brief with each criteria being weighted for importance (e.g., the number of occupants, total cost, shape, etc.) to retrieve a set of cases ranked according to its match to retrieval criteria. A multimedia case-base allows the designers and the client to visualise any preferred case as 2D & 3D CAD images, video images, scanned images (photographs), text etc. The design team then evaluates and confirms the selection of the most preferred case. The system maps all the design information to the new case creating a design brief (i.e., a building specification) for the new situation, based on the original case. The designers can modify and develop the brief to suit client requirements. The system compares the developed brief and the requirements with the original case and establishes design constraints for each decomposed design perspective or element.

NIRMANI uses an interactive adaptation process to satisfy design constraints and to revise the design using knowledge obtained from a knowledge base expressed as methods and rules. Case-based support is provided where appropriate. For example, for elemental cost adaptation, the user has the opportunity of using elemental cost sub-cases provided in the case base. (NB: sub cases are provided only when they can be used in a context *independent* manner.) Once constraints are satisfied (to a level acceptable to the designers) and the designers accept the new design, it is stored in the case library as a new case, thus completing the CBR cycle and learning from the new experience, i.e. the system conforms to a *dynamic memory model* [34, 35].

NIRMANI maintains the many perspectives of the design by the incorporation of *agents* representing each perspective. An agent is an object or a set of objects supported by an underlying case-base (these aspects are discussed in detail in the following section). The concept of multi-agents [36, 37] enables the system to achieve collaborative design.

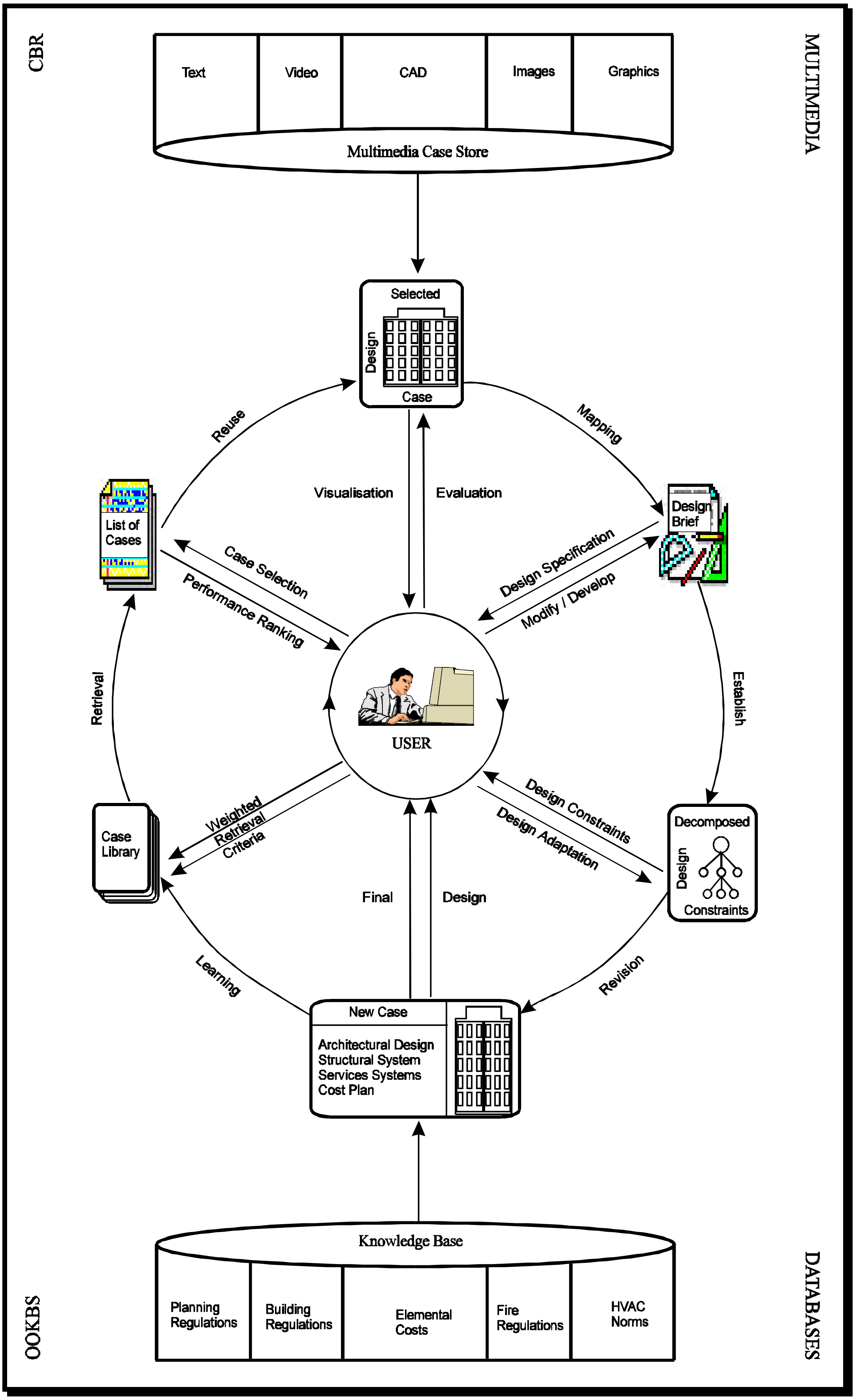

Figure 2 Case-Based Estimating & Design Process : An Overview

Table 1 provides a brief comparison of NIRMANI with other notable CBD systems. A critical and more detailed comparison of NIRMANI to other CBD systems can be found in Perera et.al. [38] and Watson & Perera, [33].

FEATURE	NIRMANI	OTHER SYSTEMS
Case Storage	Uses a multimedia storage of cases. An Object Oriented feature-based storage for indexing in hierarchical case-bases, CAD file storage, graphic file store (video & scanned images), database of costs, building regulations etc. The system uses a dynamic case-base derived from existing databases or other storage media.	Comparable to CASECAD but distinguished clearly from other systems such as SEED, CADRE etc. FABEL stores cases as CAD images and uses an OO representation and bitmaps where cases are design segments or elements.
Case Indexing	An Object Oriented perspective base structuring of cases is used with a flexible multi-level indexing schema. It has four main perspectives of Architectural, Structural, Services & Estimating. Case indexing is user directed and guided by the pre-set indexing approaches (described in detail in sections 5 & 6).	CASECAD & SEED use an object oriented indexing feature hidden from the user. Comparisons could be drawn to FABELš distance-based approach.
Case Retrieval	Provides retrieval of a set of cases ranked according to the degree of match to a set of weighted retrieval criteria (the design brief). Allows retrieval based on a set of *clues* input by the user. Therefore, high flexibility of retrieval. Uses a similarity metric to the analysis the degree of match.	Comparable with SEED, CASECAD & FABEL to some extent, but differentiated from CADRE which argues for an user selection of case. FABEL provides multiple retrieval methods.
Case Presentation	Uses a multimedia case presentation enabling greater visualisation and understanding of the design. Cases are linked to CAD drawings, video clips, photographs, related documents and WWW access.	CASECAD uses multimedia case presentation but only in the form of attribute - value pairs, 2D and 3D CAD images. FABEL uses CAD for case presentation.
Case Adaptation	Uses four main adaptation processors, i.e.: 1. Architectural Adaptation 2. Structural Adaptation 3. Services Adaptation 4. Cost Adaptation Rule based guidance is provided for case adaptation using a constraint satisfaction approach.	Most systems avoid adaptation and let the user adapt solutions. SEED uses an interactive adaptation (Architectural Design) procedure while CADRE & ACABAS uses dimensional topological adaptation.
Domain	NIRMANI operates on industrial buildings (warehouses), but could easily be used for other types of buildings. It deals with a comprehensive range of design aspects including Architectural design, Structural girding, Services System Selection and Cost Planning. Thus, it is the first system to deal with multiple domains in an integrated CBD environment.	ARCHIE, ARCHIE II, MEMORABILIA, CADRE & SEED operate in the domain of Architectural Design. CADSYN, CASECAD, STRUPLE deal with structural analysis and design. ACABAS caters for architectural design and structural griding. Most systems are developed for office buildings, court house, museums, apartment blocks etc. FABEL supports the detailed design of industrial buildings.
Implementation	Integration CBR within an OOKB development environment that facilities the use of CAD and databases. Runs in the MS Windows environment (PC based).	Most systems have been implemented in a UNIX environment (e.g., XWindows). Some use CBR development shells while others use common LISP, AutoLisp, or C.
Learning	NIRMANI learns by storing new designs generated by the system itself (a dynamic memory). It grows with the experience of the users forming a repository of co-operative experience of the design organisation. A case maintenance module allows the user to add, delete and edit cases. It keeps an account of the frequency of the usage of cases in order to prune the case-base as it grows.	Most systems rely on a considerable effort to build a memory of cases. Natural learning has been limited to a great extent. In FABEL, cases are generated from a more generic CAD based industrialised building system (ARMILLA & MIDI) using the DANCER user interface.

Table 1 Comparison of NIRMANI with other CBD Systems in Construction

4. Case Memory Organisation & Retrieval

Kolodner [13] defines a case as ;

"A contextualized piece of knowledge representing an experience that teaches a lesson fundamental to achieving the goals of the reasoner."

A case contains knowledge of the *lessons learnt* within a certain problem *context*. The content of a case may include the problem, its solution and/or the outcome. The important facet of CBR is that retrieval is not based on an *exact* match. Thus, it is assumed that a similar problem will have a similar solution.

The incorporation of multi-design tasks in NIRMANI has necessitated a *multi-perspective* or a *multi-task* case representation. As pointed out by Oxman [39], such a representation is useful for representing all possible similarities and dissimilarities that occur among different agent perspectives.

Bhat [40] states that there is no commonly accepted definition of agents; whilst Woolridge [41] proposes an agent in terms of an object or system that incorporates beliefs, actions and communications. However, an agent in NIRMANI is defined as an object or set of objects representing a perspective of a generic design function. This is supported by a perspective based hierarchical Case Base.

4.1 The Structure of a Case

This system represents a building as a *Domain case*. The *domain case* consists of a *meta-case* and a set of cases with each case having more sets of sub cases.

The meta-case representation schema are common to all perspective-based agents and are generally related to the context of the building. Cases represent each of the main perspective-based agents; e.g., Architects, Quantity Surveyors, Structural Engineers, Service Engineers. Sub cases contain specific categorisations of schema related to each perspective-based agent. For example, functional requirements, aesthetic requirements are two types of sub cases under the architectural perspective-based agent. Therefore, each domain-case consists of sets of hierarchically classified inter-referenced cases, as shown in Figure 3.

4.2 The Structure of the Case-Base

The case-base structure follows a similar structure to a *domain case* (Figure 3). It consists of a separate case-base for meta-cases, perspective cases and each categorisation of sub cases as shown below in Figure 4. In NIRMANI the meta case-base consists of Context cases that describes the context of a warehouse development project. It includes features common to all perspectives and describes the project as a whole along with the conditions in which the project was carried out. Perspective case-bases consists of cases that describe each perspective (I.e. the perspective of the architect, quantity surveyor or the structural engineer as an individual agent involved in the design development.). The sub case-bases at Level 1 consists of cases that sub divides each perspective in to important sets of features or aspects according to the needs of each perspective. For example the Architectural Perspective case-base have sub case-bases that describe each functional space (storage area sub case-base and office area sub case-base). Depending on the adaptation needs of a case any of these level 3 sub cases can be sub divided in to a set of level 4 sub cases. For example, a level 3 sub case representing the service core of a building can have stairs, toilets, service ducts as a set of level 4 sub cases. This sub division of cases can be carried out a level that is adequate to supply a set of adaptable sub cases. The implementation of this hierarchical case memory organisational structure and its use in case adaptation is explained in detail with examples in section 6 of this paper.

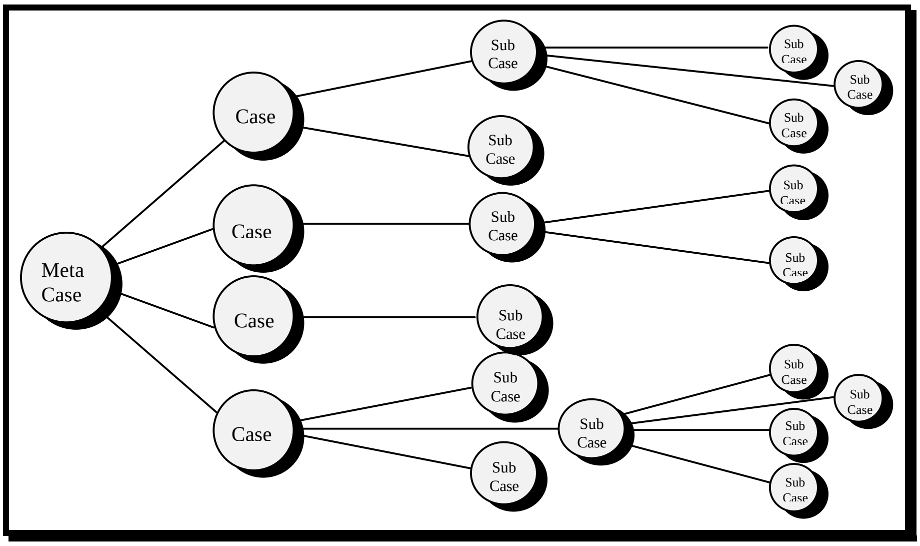

Figure 3 The Hierarchical Structure of a Domain case

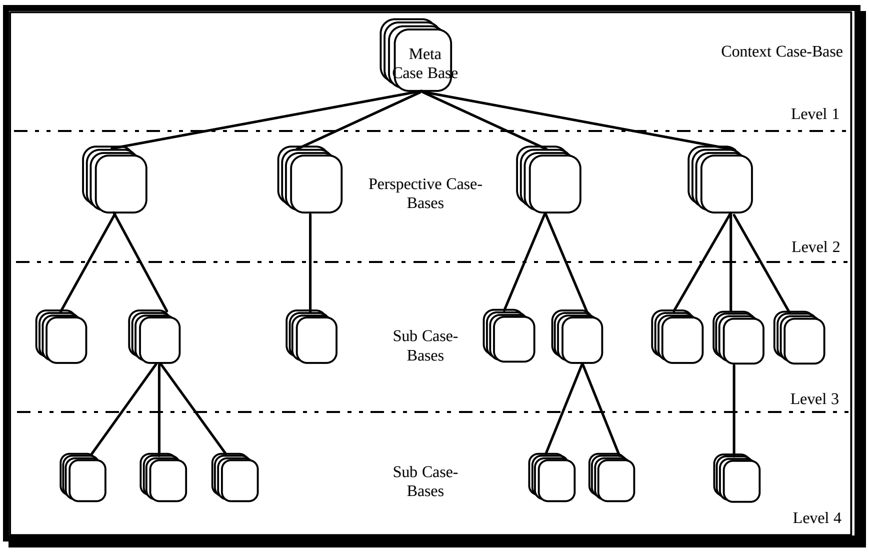

Figure 4 The Structure of Multi-Task / Multi-Perspective Case-Bases

4.3 Indexing & Retrieval of Cases

The multi-perspective case-base supports multi-task retrieval of cases. Cases and sub-cases of a *domain case* are referenced to each other and therefore, a retrieval of a case or sub-case from one case base can reference all the other family members of that case belonging to the same *domain case*. A similar case retrieval strategy is supported by ADA (Architectural Design Aid) [11]. However, whilst they use a similar hierarchical structure for cases they use a separate algorithm to classify cases according to the number of matching sub-cases for each retrieved case and they do not support multi-perspectives of the design.

Retrieval in NIRMANI consists of a weighting system for individual case attributes and also for case-bases. The case base weighting enables the indication of relative importance of a particular perspective or the importance of a particular sub-category. This lets the user retrieve cases that more closely match one perspective than another.

Three retrieval strategies have been defined for the system. The "Levels" in each of the retrieval strategies corresponds to the levels described in Figure 4.

4.3.1 Fixed Indexing

This is essentially standard nearest neighbour except that retrieval is from many case-bases. This uses a fixed index for *each* case-base and the search space includes *all* cases in *every* case-base. Cases with the highest similarity scores are selected as the final set of cases to retrieve. Case base weightings can still be used to indicate relative importance of a perspective, or a lower level case base. Figure 5 illustrates this retrieval strategy.

The entire retrieval process can be summarised as follows :

1. Cases are retrieved from each case-base independent of context or perspective.

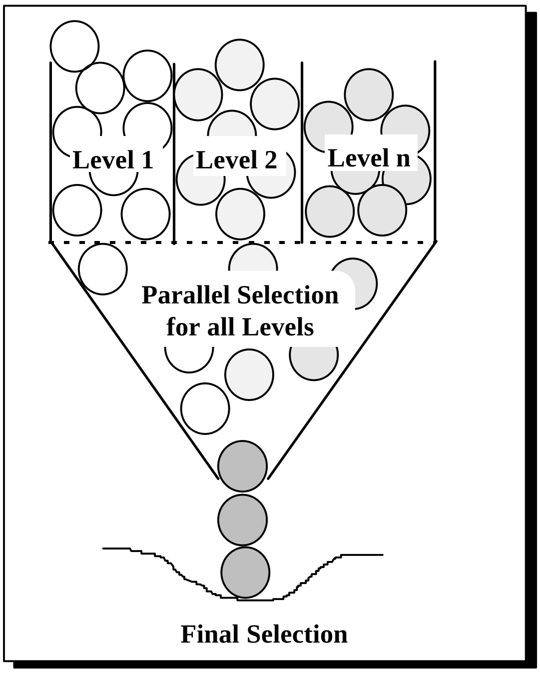

Figure 5 Fixed Indexing Method

1. Case-base weightings are used to indicate relative importance of each case-base.
2. Final selection indicates the scores gained by each case from each case-base.

The indexing is fixed and uses all cases in every case-base (i.e., context, perspective and lower levels).

This strategy is useful where cases need to be retrieved avoiding context dependency. For example, where an appropriate sub-case for case adaptation cannot be found according to the context of the *domain case* (i.e., the entire set of cases comprising a building), sub-cases can be retrieved directly without considering the context. However, a sub-case retrieved in this manner may be difficult to adapt because of interdependency and design constraints.

4.3.2 Narrowing Dynamic Indexing

This uses a *filtering* strategy to reduce or prune the search space. It uses the set of cases retrieved from the meta-case base (i.e., the context case base) as the perspective case-base for the retrieval of perspective cases. The set of cases retrieved by this second retrieval process form the third level case bases for all subsequent retrievals.

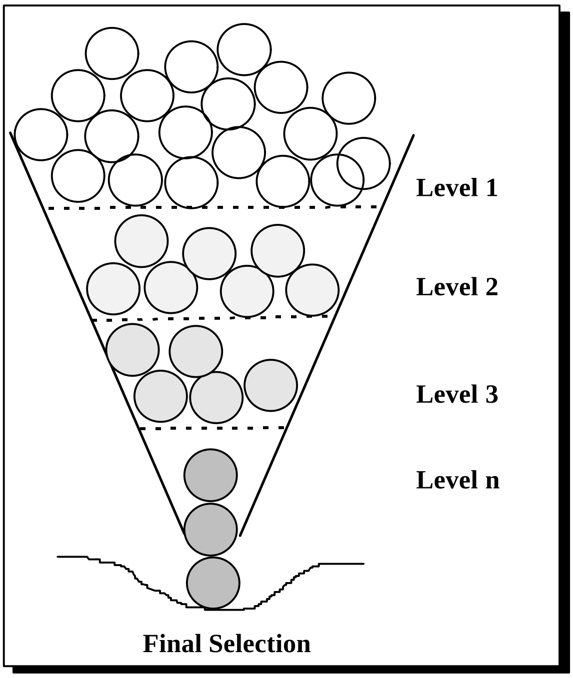

Figure 6 Narrowing Dynamic Indexing Approach

The entire retrieval process can be summarised as follows (Figure 6):

1. First, retrieve from the context case-base (Level 1) a set of matching context cases.
2. Use this set of cases to create an index for retrieving *contextually* relevant perspective-cases (Level 2).
3. Retrieve a set of matching cases from the perspective-cases of the context-cases retrieved in step i.
4. Use this set of cases to create an index for the retrieval of *perspectively* relevant cases from Level 3.
5. Repeat steps iii & iv until the system reaches the bottom level.
6. At this point the user can evaluate the similarity score each case has gained from each case-base and the overall weighted scores for the cases retrieved.

Since, the index for retrieval at each stage is created at the time of retrieval, the indexing process is dynamic. This type of retrieval strategy and indexing is useful for the initial retrieval of a *domain case*. Since in construction, design cases are highly context dependent this retrieval strategy ensures that *only* cases that are contextually relevant are retrieved.

4.3.3 Partial Dynamic Indexing

This combines the two previous methods. Here the cases are retrieved according to context, i.e. cases retrieved from the meta-case base (i.e., context case base) form the set of perspective and lower level cases to be used in all subsequent retrieval. This enables a fixed search of all these cases, but all cases retrieved will be from a similar context.

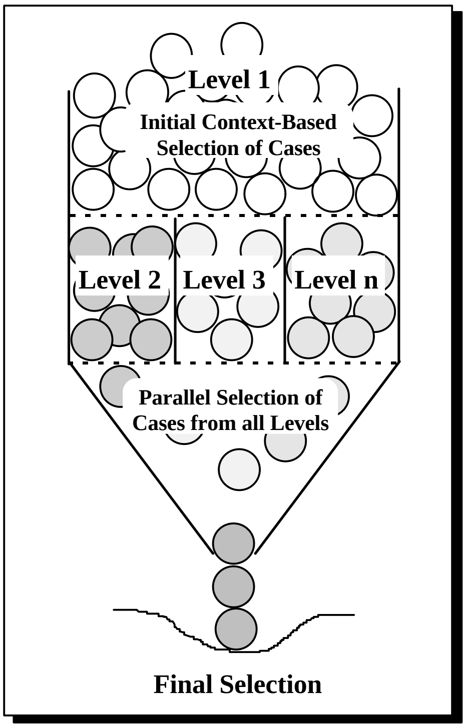

Figure 7 Partial Dynamic Indexing Approach

The retrieval process can be summarised as follows (Figure 7):

1. Cases are initially retrieved from the context case-base (Level 1).
2. This set of cases is used to create the index for the retrieval of cases from the perspective and lower level case-bases.
3. Case retrieval from these case-bases follows the process of Fixed Indexing.
4. Initial retrieval (i.e., step I) uses a fixed index whereas retrieval from all other case-bases uses a dynamic index based on the cases retrieved in step i.

Such a retrieval strategy is useful for the initial retrieval of *domain cases*. It is more useful in retrieving adaptable sub cases that are of the same or similar contexts.

NIRMANI has the ability to create cases from accessing design data directly from existing data-bases. For example, where an Elemental Cost Analysis for a building is stored in a design organisations' existing data-base, the data can be directly accessed by NIRMANI to generate cases. This makes the maintenance of case bases easier as the users only have to update their existing data-bases.

5. Case Retrieval & Adaptation Example

5.1 The Case Memory Organisation

NIRMANI is presently being implemented for the estimating and architectural design of warehouse buildings. The present implementation of NIRMANI incorporates the following multi case base structure (Table 2).

5.2 Retrieving Cases

The retrieval process in NIRMANI commences with an initial retrieval of a meta-case from the Context case-base. The system creates an index using features and their weights and adds each feature to the index. The index created can then be used to retrieve cases from the case-base. Depending on the retrieval criteria the system creates a retrieval case. A nearest-neighbour algorithm is then used to retrieve cases from the index created considering only features described in the presented case.

Fixed Indexing uses indices for the subsequent case-bases and are created in a similar manner. But, when Narrowing Dynamic Indexing is used for case retrieval of a meta case-base will be used to create the index for the case-bases at the perspective level. For example if 50 cases were selected from the General Context case-base those 50 cases will be selected to retrieve cases from the Architectural Context case-base. The result is "a set of contextually matching cases". Similarly, the filtered set of cases retrieved at the Architectural Context case-base is used to retrieve cases from the sub case bases storage area and office area. the results at this level will be contextually compatible set of cases within the same architectural context. The same process of retrieval is used to retrieve cases for estimating.

A scoring mechanism is used to indicate the degree of match of a case to the problem case. Each case will contain a case-base weighted score and a non-weighted score. The user is allowed the opportunity to prioritise retrieval of one case-base over another by using the weighting system. On completion of the retrieval process from all the case-bases the cases retrieved will have a total score indication. This indicates the sum of scores gained by a domain case or a family of cases from each case base. The recommended case is the highest scoring case. However, the user has the opportunity of selecting a different case if required.

The multimedia features of the system lets the user view related documents, images and drawings pertaining to a case. For example, a domain case contains a detailed cost analysis, cost report and specifications among other documents. Some cases have still images (photographs) of the building, video clips explaining the real time function of the building.

No.	Case-Base	Type	Brief Description
1.	Project Context	Meta Case-Base	Consists of features describing the project

			context.
2.	Architectural Context	Perspective Case-Base	Case features describe the Architects view on the design.
3.	Estimating Context	Perspective Case-Base	Case features describe the context in which the estimate can be built up.
4.	Storage Area	Sub Case-Base (Architectural Context)	Case features describe the function and constraints of functional spaces
5.	Offices Area	Sub Case-Base (Architectural Context)	Case features describe the function and constraints of functional spaces.
6.	Aesthetic Requirements	Sub Case-Base (Architectural Context)	Details of the specific overall aesthetic considerations for the project.
7.	Substructure	Sub Case-Base (Estimating Context)	Describes the specifications and costs of the respective element of the building. It contains constraints that influence an estimated value for the element.
8.	Superstructures	Sub Case-Base (Estimating Context)	
9.	Finishes	Sub Case-Base (Estimating Context)	
10.	Services	Sub Case-Base (Estimating Context)	
11.	Furniture & Fittings	Sub Case-Base (Estimating Context)	
12.	External Works	Sub Case-Base (Estimating Context)	

Table 2 Case-Bases in NIRMANI

Some cases contain design critiques, user evaluations, reports, design and construction problems encountered or references to published cost indices and relevant documents in the world wide web (WWW). The user can view any of these media files from within the NIRMANI system, upon retrieval of a case.

5.3 Adaptation of a Case

Once a case is selected by the user, the system maps all features of the selected case to the retrieval case where there is a mismatch in features, user confirmation will be sorted. Once the solution case is created in this manner, the system prompts the user to adapt features that mismatch according to the retrieval criteria used.

In order to adapt features that dont match, the user can use the case-base or data-base query facilities. In order to retrieve sub cases that can be used for adaptation previously created indexes can be used where the user needs to retrieve a case having the same general context. If the user needs to retrieve cases that have the same perspective based context, (e.g. Architectural Context or Estimating Context), an index created on Narrowing Dynamic Indexing approach can be used. If the user wishes to retrieve a case without considering the general or perspective-based context, then an index created using Fixed Indexing approach can be used. For example, assume the user needs to find a case that has a storage area with a dock-leveller system. If there are no matching cases to be found on the cases retrieved using Narrowing Dynamic Indexing or Partial Dynamic Indexing approaches, then the user can create a context independent index with the use of Fixed Indexing. A storage area sub-case retrieved can then be used to adapt the selected domain cases storage area sub-case.

Adaptation in NIRMANI is not automated. The complexity of design cases does not warrant automated adaptation. However, the system supports user directed adaptation where case-base support is provided by the system. The underlying knowledge base of the system guides the user by checking against mistakes created by the user. It will prompt the user where adaptation breaches a design or estimating rule. Once a case is adapted to a satisfactory level, the user can accept the case and add it to the case-base. The accepted case will contain information as to its source case(s). The list of source cases in a system created case provides an indication of the use of cases in the system. With time it can be used to prune the case-base.

6. Conclusions

It is clearly evident that both Architects and Surveyors use previous solutions in the process of creating new design solutions and their cost estimates. Thus the paradigm of CBR can be applied to both estimating and design through the development of a multi-agent collaborative case-based estimating and design system. NIRMANI integrates design, estimating & construction by integrating estimation and design in a single system and recording the construction experience of a design. The use of multimedia to present cases to the user enhances the usability of the system improving the understanding of complex designs by the client and the understanding of the requirements of the client by the design team. It also brings the design team to a single design platform with the client thereby improving communication.

This paper discusses problems relating to Case Memory Organisation and retrieval in multi-task multi-perspective complex domains. The NIRMANI system overcomes most of these problems. The multi-task/multi-perspective base hierarchical case memory organisation structure provides an efficient and effective case memory organisation for the complex building design domain. It is efficient because retrieval does not need to search the entire case-base to find a solution. It filters case-bases through a multi-stage retrieval process. It is effective because it considers multi-perspectives of the same design space and provides adaptable sub-cases for design and cost adaptation. The system has achieved data integration by directly accessing data for cases from existing traditional data bases.

NIRMANI encourages and assists collaboration in the design task by integrating the multi-perspectives of the construction domain in a single platform. For the first time it provides an elemental estimate of a building along with the design specification and an exemplar design (the source design as a CAD layout design). Design alternatives can therefore be developed cost effectively.

The system grows with the experience of the organisation providing libraries of corporate experience for the design organisation. The use of previous design and construction experience helps to achieve *"best practice"* in design by avoiding repetition of previous mistakes. An interesting facet of the system is that it allows the development of the design brief from a basic set of requirements to a detailed and comprehensive brief. Moreover, the entire CBD process is interactive, giving the designers sufficient authority and control to guide the design process and achieve creativity.

Knowledge based design adaptation remains an important area where case-based systems need to concentrate more in future. Complete automation of design adaptation may not be feasible in the foreseeable future because of the complexity of design and the diversity of requirements of each individual building client. Further, it is also not a feature that the experts in the construction industry look forward to. They prefer the adaptation of design cases to remain at the discretion of the designer [11]. This view can also be supported by the fact that achieving creativity in design remains a task out of reach of present computational capabilities. However, case representation characteristics can be further improved with the use of virtual reality to dynamically present CAD based adaptations to the user. Moreover, emphasis should be given to the integration of the numerous specialists of the construction industry through such systems; not to replace their knowledge but rather to aid their performance to achieve greater client satisfaction. The systems for the construction industry should, therefore, encourage collaboration of the diverse experts of the industry.

NIRMANI is presently implemented as a prototype system using ART*Enterprise (an object oriented knowledge based system development environment by Inference Corporation). The retrieval module of NIRMANI is completed and was demonstrated at the 2nd. UK Workshop on CBR [42].

References

[1] Aouad, G.F., Ford, S., Kirkham, J.A., Brandon, P.S., Brown, F.E., Child T., Cooper G.S., Oxman, R.E., and Young, B. (1994), *"An Approach to Knowledge Elicitation in the Construction Industry"*, Construction Management and Economics, Vol.12, No.3, pp 384 - 397.

[2] Kartam, N.A., (1994), A knowledge-intensive database system for making effective use of construction lessons learned, ASCE.

[3] Brandon, P.S.,(1984), Cost vs. quality : A zero sum game ? In : Powell, J.A., et. al. (ed.) Designing for building utilisation, E & FN SPON, London, pp. 108-123.

[4] Harper, D.R. (1978), *"Building : The process and the product"*, Construction Press. London.

[5] Alshawi, M., and Underwood, J., (1994), Information analysis approach for integrating design and construction, Dept. of Surveying, University of Salford.

[6] Brandon, P.S., Basden, A., Hamilton, I., Stockley, J., (1988), Expert systems, The strategic planning of construction projects, RICS.

[7] Watson, I.D., Brandon, P.S., and Basden, A., (1991), EMMY: an expert system for predicting maintenance for housing associations, In: Bezelga, A., and Brandon, P.S. (ed.) *Management quality and economics in building*, pp. 1603-1611, E & FN SPON.

[8] Maher, M.L. (1988), *"An Expert System for Preliminary Structural Design "*, Expert System for Engineering Design , Edited: M.D.Rychener , pp 37-52.

[9] Goel, A., Kolodner, J. L., Pearce, M., Billington, R.; and, and Zimring, C, (1991), *"Towards a case-based tool for aiding conceptual design problem solving"*, Proceedings : Workshop on case-based reasoning (DARPA), , Edited: E.R.Bareiss, Morgan Kaufmann . Washington, D.C. San Mateo, CA.

[10] Hua, K.,and Faltings, B., (1993), Exploring case-based building design-Cadre, In: *AI EDAM, Vol. 7, No.2*, pp. 135-143, Academic Press Limited.

[11] Hunt, J. and, Miles, R. (1995), *"Towards an Intelligent Architectural Design Aid"*, Expert Systems, Vol.12, No.3, pp 209 -218.

[12] Slade, S.,(1991), Case-based reasoning: A research paradigm, AI magazine, pp. 42-55.

[13] Kolodner, J.L.,(1993), Case-Based Reasoning, Morgan Kaufmann Publishers, Inc..

[14] Riesbeck, C.K.,and Schank, R.S., (1989), Inside case-based reasoning. *Northvale, NJ: Erlbaum*.

[15] Aamodt, A., and Plaza Enric, (1994), Case-based reasoning: Foundational issues, methodological variations and system approaches, *The European Journal on Artificial Intelligence, AICOM Vol. 7 No.1*, pp. 39-59.

[16] Perera, R.S. (1994), *"A Case-Based Design Approach for the Integration of Design & Estimating"*, MSc Dissertation, Department of Surveying. University of Salford, Salford, UK.

[17] Watson, I.D., and Marir, F., (1994), Case-based reasoning: A review, in The Knowledge Engineering Review, Vol. 9, No 4.

[18] Simon, H.A., (1973), The structure of the ill structured problems, *Artificial Intelligence 4*, pp. 181-201.

[19] Akin, O. (1988), *"Expertise of the Architect"*, Expert Systems for Engineering Design, Edited: Rychener, M., pp 173-196. Academic Press. London.

[20] Bartsch-Sporl (1995), *"Towards the Integration of Case-Based, Schema-Based and Model-Based Reasoning for Supporting Complex Design Tasks"*, Proceedings of the ICCBR 95, Portugal.

[21] Schmitt, G., (1993), Case-based design and creativity, In: *Automation in construction, Vol. 2*, pp. 11-19.

[22] Maher, M.L.,and Zhang, D.M., (1993), CADSYN: A case-based design process model, In: *AI EDAM, Vol. 7, No. 2*, pp. 97-110,.

[23] Maher, M.L.,and Balachandran, B., (1994 c), Multimedia approach to case-based structural design, In: *The journal of computing in civil engineering, Vol. 8, No. 3*, pp. 359-376, July.

[24] Nisbet, J. (1970), *"It's Elemental My Dear Bill Said the QS to the Builder"*, London.

[25] Cartlidge, D.P. and, and Mehrtens, I.N. (1982), *"Practical Cost Planning"*, Hutchinsons. London.

[26] Ferry, D.J., and, and Brandon, P.S. (1991), *"Cost planning of buildings"*, 6th edition, BSP.

[27] Bisio, R. and, and Malabocchia, F. (1995), *"Cost Estimation of Software Projects through Case-Based Reasoning"*, Proceedings of ICCBR-95, Portugal.

[28] Mukhopadhyay, T., Vicinanza, S.S. and, Prietula, M.J. (1992), *"Examining the Feasibility of a Case-Based Reasoning Model for Software effort estimation (Technical)"*, MIS Quarterly, Vol.16, No.2,

[29] Eliot, L.B. (1993), *"Case-Based Reasoning in Estimation"*, AI Expert, Vol.8, No.5, Miller Freeman.

[30] Klein, G.A. and, and Calderwood, R., (1988), *"How do people use analogues to make decisions?"*, Proceedings of Workshop on CBR (DARPA), Edited: Kolodner, J.L., Morgan Kaufmann. Clearwater, Florida, CA, USA..[31] Klein, G.A.;, Whitaker, L.A. and, King, J.A. (1988), *"Using analogues to predict and plan"*, Proceedings of Workshop on CBR (DARPA), Edited: Kolodner, J.L., Morgan Kaufmann. Clearwater, Florida, CA, USA.

[32] Stottler, R.H. (1992), *"Case-Based Reasoning for Bid Preparation"*, AI Expert, Vol.7, No.3, Miller Freeman.

[33] Watson, I.D., and Perera, R.S., (1995), *"NIRMANI : A Case-Based Expert System for Integrated Design & Estimating"*, Accepted for Publication in the Proceedings of the Expert Systems 1995, Cambridge, U.K..

[34] Schank, R.C.,(1982), *Dynamic memory: A theory of learning in computers and people*. New York: Cambridge Univ. Press.

[35] Schank, R.C.,and Abelson, R.P., (1977), *Scripts, plans, goals and understanding.* Erlbaum, Hillsdale, New Jersey, US.

[36] Oxman, R.;, and Oxman, R. (1993 b), *"PRECEDENTS: Memory structure in design case libraries"*, CAAD Futures '93, Edited: Flemming, U., , and Wyk, S.V., pp 273-287.

[37] Sycara, K. P. (1989), *"Argumentation: Planning Other Agents' Plans"*, Proceedings of IJCAII-89, Michigan.

[38] Perera, R. S., Watson, I. D., and Alshawi, M. (1995), *"Case-Based Design Approach for Integration of Design and Estimating"*, Proceedings of the 4th International Conference on the Application of AI to Civil & Structural Engineering, Civil Comp 95 (to be published). Cambridge, UK.

[39] Oxman, R. (1995), *"Data, Knowledge and Experience in Multi-user Information Systems"*, Construction Management and Economics, Vol.13, pp 401 - 409. E. &. F. N. Spon.

[40] Bhat, R.R. (1995), *"Agent Systems for Case Adaptation"*, Proceedings of the International Conference on Computer-Aided Architectural Design, Centre for Advanced Studies in Architecture. National University of Singapore, Singapore.

[41] Woolridge, M. (1992), *"The Logical Modelling of Computational Multi-Agent Systems"*, PhD Thesis, Department of Computation, UMIST - University of Manchester, Manchester, U.K..

[42] Watson, I.D. (1996). *Progress In Case-Based Reasoning*. Proc. of the 2nd. Uk Workshop on Case-Based Reasoning. University of Salford, UK.

Perera, R.S., and Watson, I. (1996), "Multi-Agent Collaborative Case-Based Estimating and Design in NIRMANI: Organising a Multi-Perspective Case Memory", Kumar, B. (Ed.), Information Processing in Civil and Structural Engineering Design, pp. 53-64, Civil-Comp Press, Scotland, U.K.