

# Modeling CBR Representation for Architectural Design Reuse

Paul Raduma, Lic. Tech., M.Arch., B.Arch.

Helsinki University of Technology, Espoo 02150 Finland, E-mail: Paul.Raduma@hut.fi

## Abstract

The design of a good representation scheme is an important step towards facilitating the construction of case-based reasoners. The paper presents a design case representation scheme based on a conceptual CBR model for architectural design re-use, and discusses relevant lessons from case representation in other CBR models. The model presented decomposes a house design, and uses relative room positioning, co-ordinates and orientation to represent the basic spaces within the house. The author considers cases stored as memory of the design field. These cases are treated with a retrieval process (selection and adaptation) from the memory. The originality of this process is in the dynamic cycle of selection and adaptation. This paper uses the conceptual model to introduce and present a process of the dynamic cycle of selection and adaptation of design cases.

## 1. Introduction

Architects frequently use experiential knowledge in the problem solving process. Case-Based Reasoning (CBR) is based on the use of situational experiences in solving new problems. There is growing interest in CBR and knowledge-based systems in the design professions. Applications inspired by this interest include ARCHIE (Domeshek and Kolodner 1997) JANUS (Fisher, McCall, and Morch 1989), SEED (Flemming et al. 1997), FABEL (Voss 1997), and PRECEDENT (Oxman 1996). HouseCAD, a conceptual CBR model discussed here, is part of an ongoing research at Helsinki University of Technology. The HouseCAD model uses relative room positioning, co-ordinates and orientation to represent the basic spaces within the house. It is based on snapshots from the design process, using the house's plan view in its representation. The model assumes human-architect's intervention in the adaptation process, based on the proposals from iterative retrievals by the case based reasoner.

Design cases, like other cases in general have three basic components: a situation or *problem description*, a *solution*, and an *outcome*. The *problem description* records the state of the world at the time the case was taking place, and if appropriate, what problem was to be solved at that time.

The *solution* records the solution to the problem specified in the problem description. The *outcome* records the resulting state of the world after the solution was carried out (Kolodner 1993). Cases with situation description and outcome can be used in evaluating new situations. This is especially important in a continuous environment like architectural design where it is not obvious where a case starts and where it ends. The design process of a building is continuous, and changes are inevitable in all stages – from preliminary to detailed design and construction phase. A snapshot of a stage in the design process can be represented as a case if it teaches a lesson for the future.

Case storage in CBR systems should reflect the conceptual view of what is represented in the case. In HouseCAD, design-case representation for CBR indexing starts with extracting indexes from the bubble diagrams architects employ in conceptual design. This information is later used by the nearest-neighbor retrieval module, which compares the problem brief with design cases in the case memory. The Case-based reasoner recommends to the human architect the design cases suitable for further adaptation and storage.

In CBR, the case-base should be organized into a manageable structure that supports efficient search and retrieval methods. A balance has to be found between storing methods that preserve the richness of cases and their indexes and methods that simplify the access and retrieval of relevant cases. The methods are largely theoretical and hardly apply in the commercially available CBR tools. Instead, the commercial CBR tools either store cases as simple and flat file data-structures, or within conventional relational database structures and use indexes to reference the cases. The commercial CBR tools generally use two techniques: nearest neighbor retrieval and inductive retrieval. This is the process of incorporating what is useful to retain from the new problem-solving episode into the existing knowledge. The learning from success or failure of the proposed solution involves how to index the case for later retrieval from similar problems, and how to integrate the new case in the memory structure. Hence, the need to organize the case-base into a manageable structure.

## 2. Representing a Design Case

### 2.1 Bubble Representation

Architects love graphic representations, and often develop their early design ideas using bubble diagrams. The HouseCAD model proposes case-representation derived from the bubble diagram. A bubble is drawn to represent each space; dark thick lines are drawn to indicate strong relationships, while light thin lines represent weaker relationship (Figure 1).

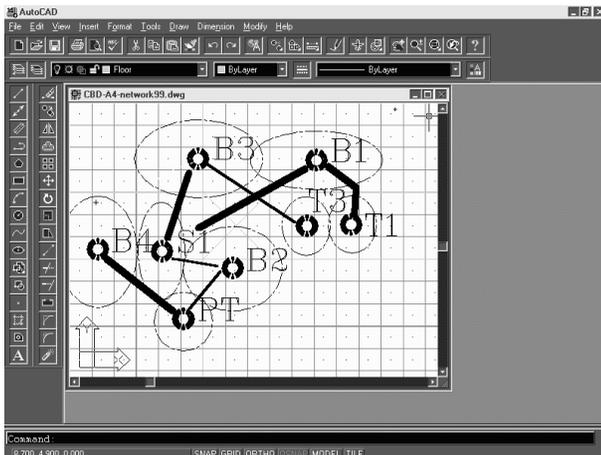


Figure 1. Bubble Representation of a Design Scenario

In HouseCAD, the bubble diagrams are defined by ellipses whose centers are stored as 3D co-ordinates, and have their attributes stored in a CAD program, and correspond to the center point of the respective space. The scenario shows a stage in the design process of the upper floor of a two-story house. The master bedroom is represented by B1, while B2, B3, and B4 represent the second, third and fourth bedrooms respectively. T1, T3 and PT represent the master bathroom, third bathroom and patio/terrace respectively.

In ARCHIE-II – a case-based design aid for the conceptual design of buildings - for example, a bubble editor assists the architect with conceptual design in three ways (Griffin and Domeshek 1996). First, to improve on a representational technique architects are already using by automating editing and preserving an on-line record of their work. Second, to help the architects describe features of the spaces. Third, the bubble editor helps to form queries to the case-based design aiding system. CBR systems use indexes to speed-up retrieval. A good representation scheme should reflect the conceptual view of what is represented in the case. The CLAVIER system for autoclave loading (Hennessey and Hinkle 1991, Hinkle and Toomey 1994) is one of the first commercially fielded CBR applications. It has its case representation such that each layout case is described

in terms of the name of parts; tables on which the parts were placed; relative position of other parts; and production statistics such as start and finish times, pressure, and temperature. In HouseCAD, the bubble diagram records one interpretation of the brief: the *problem*. The case library stores design-cases: *solutions* that are represented partly based on the bubble diagram.

### 2.2 Choosing the Design Cases

To determine the relative frequency of various spaces or rooms in home designs, an analysis was done on 612 contemporary styles. These contemporary houses were taken out of an on-line case base of 5,005 from various designers. These had two stories or more, attached garages and had floor area of between 92 and 278 m<sup>2</sup>. Out of these 46 had floor areas of were between 255 and 275 m. This group was first studied to determine space frequency, which in turn would determine the 32 spaces and rooms to be considered for indexing in the HouseCAD design-case representation scheme (Raduma 1999).

### 2.3 Trimming the Case Base

The selection criteria from the case-base of the 5,005 was modified such that only single story cases with 3 or 4 bedrooms, attached garage, and with floor area of 135 to 185 m<sup>2</sup> would be retrieved. The resultant was a case base of 141, indexed 1001 to 1141. Sixteen of these cases were removed from the case base if, in the author's opinion, they were not good cases. The final case base used for the representation scheme had 125 cases. The cases were drawn from a total of 18 design companies. By limiting the design cases to single story cases with 3 or 4 bedrooms, attached garage, and with floor area of 135 to 185 m<sup>2</sup>, it makes it possible to avoid noisy cases. It also eliminates cases that would otherwise make the design case base too large manually enter into a case memory.

## 3. Decomposing a Design Case

Bubble representation of a transformation of the brief is shown in Figure 2. The scenario shows the house divided into four quartiles - northeastern, southeastern, southwestern and northwestern quartiles. The plan north is for design case representational and retrieval purposes only. Later discussion shows how the best match for this transformation compares with other case matches in the simulation (Figure 6).

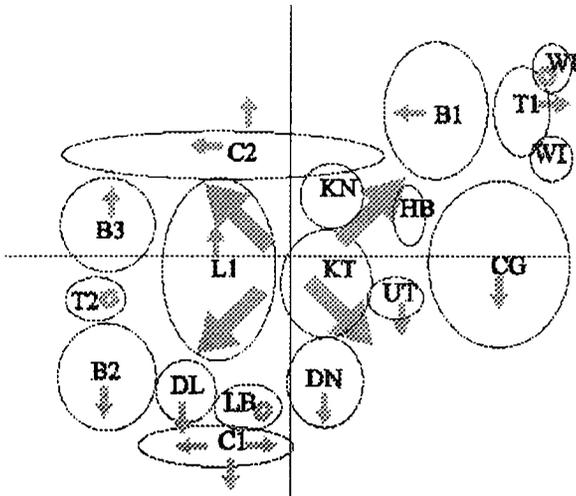


Figure 2. Space Orientation and Location: Brief 2

The case representation is an interpretation of the brief that calls for a detached single-story, one-family house, for a retired couple. It should have three bedrooms, and an extra room that may be used as an office. There is a spectacular view to the north. The living room, a covered porch or deck, and one or two bedrooms should take advantage of this view. Vehicular access to the house will be from the southwest. It should accommodate a two-car garage, and the lady of the house prefers it to be easily accessible to the kitchen and utilities area. Two separate walk-in closets are required in the master bedroom

An earlier model of HouseCAD divides the house into four quartiles - northeastern, southeastern, southwestern or northwestern (Raduma 1999). It considers only one major orientation or view. A space has to be oriented to the north (N), south (S), east (E), west (W), oriented inward (I) or non-existent (X). The model represents each house plan in sets of four representation schemes, transformation by reflection of the original design-case. The transformation of each plan yields three additional versions of the original. The three additional cases may be necessary if orientation with respect to the sun, wind, slope; view and other site constraint dictate, while other factors governing layout remain constant. Depending on the transformational plan retrieved, the living room of the same design-case original may be indexed to be in the NEQ, SEQ, SWQ, or NWQ, all of which are correct, and may apply in different site contexts. Transformation by reflection or rotation is not uncommon in site planning, and is made easy by CAD programs. A record of such transformation is important, and shows, after further adaptation how each unique design has its own unique model. Each design process includes several design solutions that can also be represented as cases. As designers continually re-use and modify their

previous designs, it is important to model and analyze such record (Raduma 1999). This increases the case memory from which iterative solutions of design problems can be applied.

ARCHIE uses specific space types based on courthouse experiences. SEED relies on two central constructs - design units and functional units in its indexing and retrieval. In the FABEL domain, a case is a layout, and its features are attribute-value pairs, or keywords that are associative. A CBR tool in the architectural design domain should aim to develop a decision support system for architects. FABEL for example, explores various methods for combined reasoning and mutual support of different knowledge types. HouseCAD uses major spaces and rooms in a house, and their orientation for indexing.

#### 4. Case Descriptors

The HouseCAD represents its design cases by concentrating on the solution. This bears some similarity with JULIA (Hinrichs 1992) and KRITIK (Goel 1989), which are also design problem solvers. JULIA's design task is in creating menu solutions. Most of the problems are too large, and hence are broken into parts. KRITIK's problem domain is physical devices (e.g. simple electrical circuits), and is smaller than JULIA's. Its functional representation shows how the pieces are connected to each other, and shows the general effects of those connections.

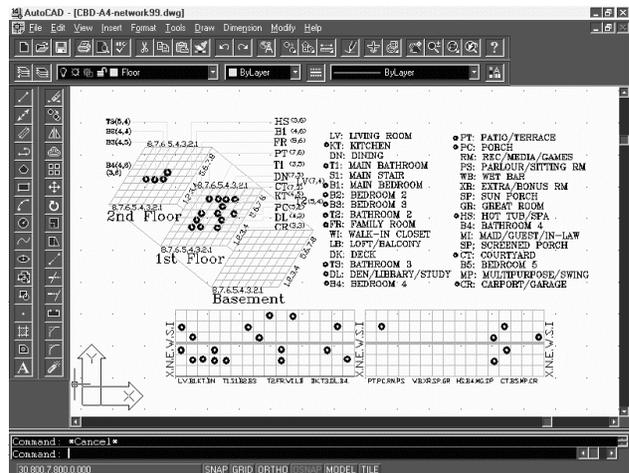


Figure 3. Design case representation

The representation scheme uses the 32 case descriptors as discussed above. For simplicity, this scheme uses spaces and rooms on the main floor in its representation. As shown in Figure 3, it is however, possible to use a more elaborate 3-floor representation scheme (Raduma 1999). This scenario shows 3-D representation of the case design

scriptors, to show whether they represent spaces on the basement (level 0) the main floor (level 1) or the upper floor (level 2). This scenario is based on a study that used a 3-metre by 3-metre grid. The model decomposes a house design, and uses relative room positioning, co-ordinates and orientation to represent the basic spaces within the house.

The earlier HouseCAD design case representation model has two priority sets, each with 16-room or space descriptors (Figure 3). The first set of 16 contains the descriptors for the living room (LV), master-bedroom (B1), kitchen (KT), dinning room (DN), and main bathroom (T1). Other are the main stairs (S1), bedrooms 2 (B2) and 3 (B3), second bathroom (T2), family room (FR), walk-in closet (WI), loft/balcony (LB), Deck (DK), third bathroom (T3), Den/Library/Study/Office (DL), fourth bedroom (B4) and patio/terrace (PT). The model uses simplified 2D co-ordinate system, in addition to the three planes representing the basement, first and second floors. This in effect is a 3D model that uses relative room positioning to determine the location of the respective rooms within an eight by eight grid. Master bedroom width determines the grid dimension – one unit on the X and Y-axes, while 1 in the Z-axis refers to level 1. Basement floor is level 0, while the upper floor is level 2 in the Z-axis.

## 5. Retrieval and Match

Assuming that the bubble diagram (Figure 1) is an interpretation of the brief – the problem to be solved - in terms of the required layout, the retrieval may yield a nearest neighbor such as the one shown (in Figure 4). In this hypothetical example, the retrieved case has some similarity in the positioning of master bedroom B1, bedrooms 2, 3 and 4, bathrooms T1 and T2. In real life scenario, HouseCAD may retrieve a design case from the case memory, which is not the nearest neighbor. CAD Layer use makes it possible to compare this retrieved case with the first bubble diagram. The spatial relationship between the two bubble diagram representations is checked.

In terms of orientation, the case satisfies the desired conditions in the master bedroom 1, bedrooms 2, 3, 4, second bathroom T2, and to some extent, the master bathroom T1. One other aspect, that is, the distance between comparable rooms or spaces is computed. The distance between the respective main stairs S1, master bedroom B1, and bedrooms B2, B3, and B4 work out to be about 4.5 m, 3 m, 2.2 m, 3.2 m and 5 m respectively. The human-architect considers that these dimensions are too big, probably removes the design case from the case memory and another search process is instituted.

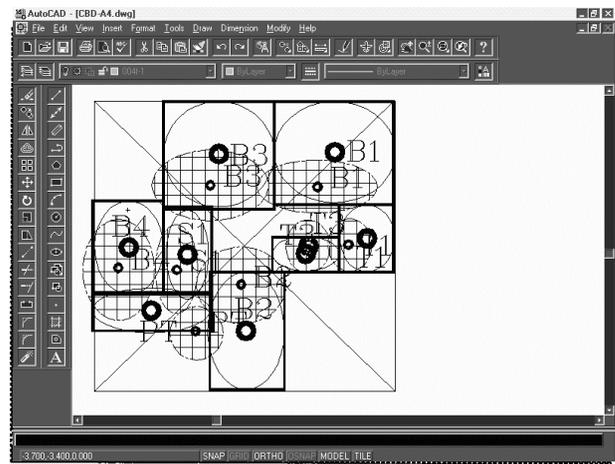


Figure 4. Case Match

After several searches from the existing case base, a better design case is retrieved, whose bubble diagram representation is again compared with the original bubble representation of the brief (Figures 5). The distances are much more acceptable (about 0.9 m, 1.8 m, 0.8 m, 2.1 m, and 2.5 m). Using weighted nearest-neighbor retrieval technique, the latter distances (2.1 m, 2.5m), that between respective bedrooms 3 and 4, HouseCAD advises that the retrieved case is acceptable for adaptation.

In the current HouseCAD representation scheme, each design case is transformed by reflection and/or by rotation. This transformation yields an additional 7 design case representations. There are two alternatives: Each of the 125 design cases with 8 design case representations and stored in the case-base, or a second alternative of representing the brief with additional 7 design case representations. The latter has been used in the simulation. Hence, we have a total of 8 design case representations namely the BRIEF, BRIEF 2, 3, 4, 5, 6, 7 and 8. The design Brief 7, for example, refers to the brief mirrored along X-axis with 90° rotation.

Each of these representations was matched with each of the 125 design cases in the case base. The HouseCAD representation scheme uses the 32 case descriptors discussed in step 4 above. An analysis of the 125 design cases revealed that the first 8 descriptors accounted for 985 out of 2,106 instances (46.8%). The next 8 descriptors accounted for 796 instances (37.8%) and the last 16 accounted for 325 (15.4%). This information was used in weighed matching. Hence design case 1,082 has a 23% match with the brief. It has 2 matches using the first 8 descriptors, 1 match with the next 8, and one match with the last 16 case descriptors.

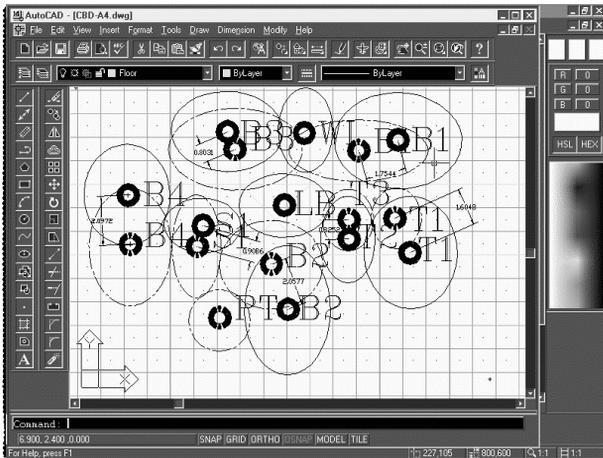


Figure 5. Case Comparison

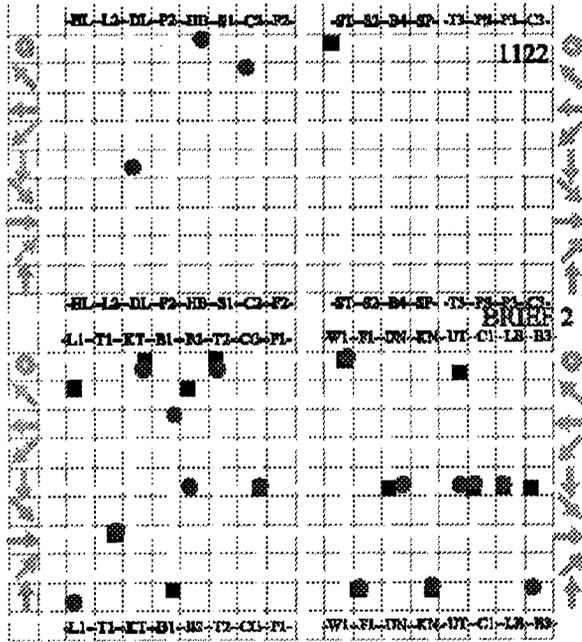


Figure 6. Case Retrieval 1122 (37% match)

The brief has a total of 18 case descriptors distributed as follows: 7 out of 8 in the first set; 7 out of 8 in the second set, and 3 out of 16 in the third set. Using the design case representation of the brief, there was an average match of 10.9%. The best four retrievals had matches of 39%, 34%, 28% and 28%. There was no match at all in 18 out of the 125 cases. The design Brief 2, a transformation of the brief - mirrored along Y-axis had an average match of 12.7% - better than the original brief. The best four retrievals had matches of 37%, 37%, 36% and 35% (Figure 6, Table 1).

There was no match at all in 16 out of the 125 cases. Brief 3, a transformation of the brief mirrored along X-axis had an average match of 6%. Brief 4, a transformation of the brief mirrored along Y-axis & with 90° rotation had an average match of 5.6%. The best four retrievals had matches of 18%, 16%, 16% and 16%. There was no match at all in 39 out of the 125 cases Brief 5, a transformation of the brief mirrored along X-axis had an average match of 5.9%. Brief 6, a transformation of the brief with 90° rotation had an average match of 4.3%. Brief 7, a transformation of the brief mirrored along X-axis with 90° rotation had an average match of 6.1%. The best four retrievals had matches of 23%, 23%, 17% and 16%. There was no match at all in 42 out of the 125 cases. Brief 8, a transformation of the brief with 180° rotation had an average match of 3.4%.

BRIEF 1				BRIEF 2			
	max. match	total		max. match	total		
	7 8 3	18		7 8 3	18		
weight	47 38 15	100		47 38 15	100		
CASE	case match	%	CASE	case match	%		
1065	3 4 0	39	1102	2 5 0	37		
1074	3 3 0	34	1122	2 5 0	37		
1026	2 3 0	28	1123	1 4 2	36		
1047	2 3 0	28	1037	3 2 1	35		
1052	1 4 0	26	1062	3 3 0	34		
1007	3 1 0	25	1063	3 3 0	34		
1008	3 1 0	25	1075	3 3 0	34		
1009	3 1 0	25	1106	1 4 1	31		
1086	3 1 0	25	1112	1 4 1	31		
1082	2 1 1	23	1107	1 5 0	30		
1006	2 2 0	23	1051	3 2 0	30		
1121	2 2 0	23	1081	2 3 0	28		
1028	1 2 1	21	1104	2 3 0	28		
1010	1 3 0	21	1068	1 3 1	26		
1032	1 3 0	21	1101	1 3 1	26		

Table 1. Best 15 matches in Brief 1 and Brief 2

The result of the simulation, especially of the Brief and Brief 2 is significant. It should be noted that the case base was restricted to single story houses with 3 or four bedrooms and an attached garage. Each of the 32 case descriptors can be represented to be in any of the four quartiles (NEQ, SEQ, SWQ and NWQ). If represented in each of the four quartiles, the case descriptor may be represented to open to one of the nine orientations: the north, northeast, east, southeast, south, southwest, west, northwest, or inward. This implies that there are 4\*9 alternatives in representing each descriptors, or 36 possibilities. The brief had 18 spaces, each of which has 36 orientation alternatives.

The number of combinations and permutations is huge. On average, the 125 design cases had a little less than 18 case descriptors, but some had as little as 14 (case 1022). To achieve an exact match, the correct design case should have 18 case descriptors in a specific orientation. The chance for getting an exact match in any of the 125 design cases is close to zero. The success of our case base is due to the fact that only cases relevant to the narrow domain (single story houses with 3 or four bedrooms and an attached garage) was used. Further the cases were represented such that the entrance lobby (and front porch) were oriented to the south, thus limiting any random location of the lobby (LB) and in some cases, the front porch (C1) to the south.

## 5. Adaptation Question

Adaptation is widely accepted as the biggest challenge in the CBR process. It is yet to be seen if case-based reasoners in the design domain will develop as autonomous systems that carry out retrieval, adaptation and evaluation process by themselves. We have a lot to learn from, for example, CHEF, JULIA, PLEXUS, CASEY, PROTOS and HYPO, case-based reasoners considered by some to be autonomous systems (Leake, 1996). Human-machine system that works along with architects to solve design problems or interpret design scenarios is proposed in the HouseCAD model. CLAVIER, considered successful in the design domain by most CBR scientists, was designed to adapt autoclave layouts, but the engineers on the ground did not like the adaptation facility. They opted to adapt the layouts themselves, using CLAVIER to check that their human adaptation will not repeat a failed autoclave layout.

The adapted design-case, from its case representation or indexing, is compared with other indexes in the case memory. In one scenario, one design case compared very closely with the original bubble diagram – the brief (Figure 1), especially the relative position and orientation of master bedroom B1. It also compared very well with the adjacent bedroom B3, bedroom B4 with its "western" orientation, position of the patio/terrace PT and the position of the master bathroom T1. The human-architect may accept this as the final retrieval from which minor and final adaptation will be made. The architect may alternatively retrieve another case, compare it with the original bubble diagram, transform it, and adapt it. In order to satisfy the constraints set out earlier, the architect converts the master bedroom B1 to B2, then enlarges B2 to the size of B1, and converts the master bathroom T1 to patio/terrace PT. The resultant goes through transformation by reflection to yield a total of four new designs. This is stored as part of the case-base (Figures 7, 8).

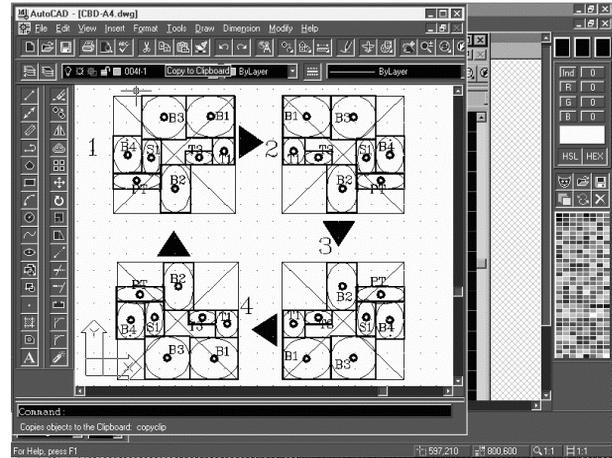


Figure 7. Transformation

The adaptation method in HouseCAD is not autonomous, but based on the human architect's intervention. The method is a combination of specialized search and case-based substitution. Both auxiliary knowledge structures and case memory are queried, and specialized search heuristics are used to guide the memory search. Case-based substitution uses other cases to suggest substitutions as in CLAVIER and JULIA.

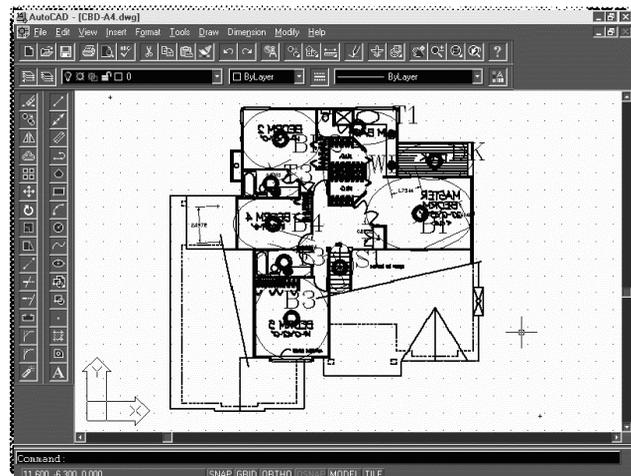


Figure 8. Case Adaptation

## 6. Discussion

Architectural design re-use is one avenue of CBR research that can be of direct benefit in housing design for large population groups. The traditional white table approach to low-income housing design cannot yield user-specific designs if there is limited time and financial resources committed to architectural design. The white table approach in

housing design for such large population groups may result in duplicating inappropriate prototypes in hundreds or thousands, not suited to varying user profiles. The design of housing has to be seen from the design of the individual unit and as part of a larger urban planning project. It is therefore useful to compare lessons from case-based planning. In the design and planning of housing, CBR as a knowledge-based system is feasible as a knowledge assistant, and an intelligent consultant (Yeh and Shi 1999). In case-based planning (CBP), previously generated plans are stored as cases in memory and can be reused to solve similar planning problems in the future. CBP can save considerable time over planning from scratch (generative planning), thus offering a potential (heuristic) mechanism for handling intractable problems. One drawback of CBP systems has been the need for a highly structured memory that requires significant domain engineering and complex memory indexing schemes to enable efficient case retrieval. This problem can be alleviated, like in CaPER (Hendler 1995), using a CBP system that is based on a massively frame-based AI language that can do extremely fast retrieval of complex cases from a large, non-indexed memory.

The performance of Case-Based Reasoners, like other intelligent systems, depends on factors including inherent design decisions and domain characteristics. In the architectural design domain, breaking the task in to small manageable pieces best solves the problem. This does not preclude looking at the big picture- how the individual parts relate to each other and with the outside world. The HouseCAD model proposes a design case representation model in the domain of residential architecture based on the decomposition of a house design, and the use of relative room positioning, co-ordinates and orientation to represent the basic spaces within the house.

## 7. References

- Domeshek, E., and Kolodner, J. 1997. The Designer's Muse. In *Issues and Applications of Case-Based Reasoning in Design*, M. L. Maher and Pu, P. eds. Hillsdale, N.J: Lawrence Erlbaum Associates: 11-38.
- Fisher, G., McCall, R., and Morch, A. 1989. Design Environments for Constructive and Argumentative Design. In, *Proceedings, Human Factors in Computing Systems Conference*, New York: ACM Press: 269-275.
- Flemming, U., Aygen, Z., Coyne, R., and Snyder J., 1997. Case-Based Design in a Software Environment that supports the Early Phases in Building Design. In *Issues and Applications of Case-Based Reasoning in Design*: 61-86.
- Griffith A.L. and Domeshek, E. 1996. Indexing Evaluations of Buildings to Aid Conceptual Design. In Leake, D. B., ed. *Case-Based Reasoning. Experiences, Lessons & Future Directions* Cambridge, MA: AAI Press/ MIT Press
- Goel, A. 1989. Integration of case-based reasoning and model-based reasoning for adaptive problem solving. Ph.D. dissertation, Dept. of Computer and Information Science, The Ohio State University.
- Hendler, J. A. 1995. Evaluating the CAPER Planning System, Proceedings, Israeli Symposium on Artificial Intelligence, Jerusalem, Israel, Jan 1995.
- Hennesey, D. and Hinkle, D. 1991. Initial Results from Clavier: A case-based autoclave loading assistant. In *Proceedings of the DARPA Case-Based Reasoning Workshop*, edited by R. Bareiss. San Francisco: Morgan Kauffman Publishing.
- Hinkle, D., & Toomey, C. 1994. Clavier: Applying Case-Based Reasoning to Composite Part Fabrication. In *Innovative Applications of Artificial Intelligence 6*, 54-62. Cambridge, MA: AAI Press/ MIT Press.
- Hinrichs, T.R. 1992. Problem Solving in open worlds. A Case Study in Design. Northvale, NJ: Erlbaum.
- Kolodner, J. 1993. Case-Based Reasoning. San Mateo, CA: Morgan Kaufmann.
- Leake, D. B. 1996. CBR in Context: The Present and Future. In Leake, D. B. ed.. *Case-Based Reasoning. Experiences, Lessons & Future Directions* Cambridge, MA: AAI Press/ MIT Press
- Oxman, R. E. 1996. Case-based design support: supporting architectural composition through precedent libraries. *Journal of Architectural and Planning Research* 13, 242-255.
- Raduma, P. 1999. Designing a CBR Representation Scheme for Architectural Design Reuse. *Arpakannus 1/99*, Newsletter of the Finnish Artificial Intelligence Society, Special Issue on the IJCAI'99 affiliate event Networks'99, Espoo, Finland August 8-19, 1999.
- Voss, A. 1997. Case Design Specialists in Fabel. In Maher M.L & Pu, P. eds. *Issues and Applications of Case-Based Reasoning in Design*, Lawrence Erlbaum
- Yeh, A. G. O., and Shi, X. 1999. Applying case-based reasoning to urban planning. A new planning-support system tool. *Environment and Planning B: Planning and Design* 1999, volume 26 pages 101-115.