Application of case-based reasoning techniques to the automation of single-family residential property appraisals

Raymond Laureano-Ortiz
University of Central Florida, raymond.laureano@gmail.com

APPLICATION OF CASE-BASED REASONING TECHNIQUES TO THE AUTOMATION OF SINGLE-FAMILY RESIDENTIAL PROPERTY APPRAISALS

BY

RAYMOND LAUREANO-ORTIZ
B.S.E.E., Polytechnic University of Puerto Rico, 1988

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Computer Engineering
in the Graduate Studies Program
of the College of Engineering
University of Central Florida
Orlando, Florida

Fall Term
1990

ABSTRACT

Case-based reasoning has emerged as an alternative to rule-based reasoning techniques for the design of expert systems. This paper concentrates on the issues involved in the application of the case-based reasoning techniques to a specific domain, property appraisal. Case-based reasoning has been recently favored because it seems to resemble more closely to the psychological process humans follow when trying to apply their knowledge to the solution of problems: People adapt solutions of similar problems they handled in past experiences to address present situations.

Property appraisal or valuation is a domain characterized by having a single parameter in its solution, that is, the value of the property being appraised. This makes it differ from most of the domains in which case-based reasoning have been attempted. Those other domains require the satisfaction of multiple goals, which are related to one another in some type of explanation or plan. Because of the fact that property appraisal has a single goal, it is particularly important to find the best possible answer for that solution. In addition to this, the achievement of consistency is also essential in this domain in which different experts may reach different answers even having the same data at their disposition.

By modelling the market data approach of appraisal, using adaptations of case-based reasoning techniques, such as the similarity links and the critics, and integrating other techniques, such as the use of comfort factors, a case-based reasoner for property appraisal is implemented addressing the issues just mentioned above.

ACKNOWLEDGEMENTS

Most of the background information and heuristic knowledge about property appraisal in this work was provided by Curtis Fieldson, a professional property appraiser from Chicago, Illinois. Also, Kathleen Shearer, a property appraiser from Altamonte Springs, Florida, as well as Aida Diaz, a real estate agent from Deltona, Florida, gave me access to some very useful sources of sales data. I thank them for allowing me share with them some of their precious time to gain an understanding about the procedures they follow in doing their job.

In addition, I express my gratitude to the following group of persons: Dr. Avelino J. Gonzalez, for sharing his ideas with me and for trusting me unconditionally; Dr. H. R. Myler and Dr. S. Khajenoori, for taking some of their time to evaluate my work; the staff of STAC and the Interlibrary Loan Office, for the great service they provided to assist in the location of references; Linda Wills and Dr. Wills, for their help in maintaining the Symbolics equipment in working conditions; all those who have given me their love and moral support, especially Mami, Papi, Normita, and Eliomer, and also Maria, Gilberto, and Tere; the Supreme Being, for allowing us to freely decide what we want to accomplish and blessing us with the means to do it.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION: CASE-BASED REASONING, AN ALTERNATIVE

Heuristic knowledge compiled from many experiences is not the only way we use experience in reasoning. Frequently, a specific previous experience that is stored in our memory acts as a guide in allowing us to construct a solution to a new situation. If we recall a previous case similar to the problem we are trying to solve, we can use it as a guideline for solving the new case (Kolodner and Riesbeck 1986). This view of intelligence is known as case-based reasoning (CBR).

Case-based reasoning techniques involve the search of solutions to present situations by looking back at precedents, that is, old cases. They entail the retrieval of old cases to illuminate aspects of the current problem and adapt the old solutions to solve the new situation. Examples of case-based reasoning can be found in the legal system and at financial institutions. When faced with the task of deciding on a sentence for a person that has been declared guilty of a crime, a judge uses laws to obtain general guidelines for the type of crime committed. However, it is by researching previous cases or precedents that the judge obtains a view of specific applications of the law. Judges then adapt the differing features of the previous case in relation to the current case being considered. In this way, the old sentence is changed to reflect the needs of the present case or situation. Other group that utilizes case-based reasoning when solving problems is comprised by loan approval officials at financial institutions. They might have general guidelines provided by the bank policies to approve loans, but also they use the records of previous customers that have the most similar characteristics to the present loan applicant. The officials observe the loan repayment history of these old customers and, based on this and taking into account the differences between the old customer and the new applicant, they decide about the approval of the current loan.

1.1 Modelling Human Reasoning

In the last few years, there has been an increasing interest in research into the general area of case-based reasoning. Case-based reasoning has been proposed as a more psychologically precise model of the reasoning of an expert than the more widely used rule-based systems, which are the basis of the expert systems that began to be commercially available in the past decade (Kolodner 1988; Kolodner and Riesbeck 1986; Riesbeck and Schank 1989).

As discussed in (Riesbeck and Schank 1989), learning in fields such as law and business means learning the cases, and reasoning in these subjects means being able to make new decisions by abstracting the essentials from an appropriate prior case. Thus, the essence of thinking, in these fields, is the storage and retrieval of cases. This follows the more general premise that case-based reasoning is the essence of how human reasoning works (Kolodner and Riesbeck 1986). People reason from experience. They use their own experiences if they have a relevant one or they make use of the experience of others to the extent that they can obtain information about such experiences.

Case-based reasoning is thus an example of reasoning by analogy (Carbonell 1982, 1986; MacKellar and Maryanski 1988; Winston 1980, 1982). Analogical problem solving per se is one form of learning because it learns from previous experience how to solve similar problems. Use of analogies produces comparison-based predictions. Thus, analogical reasoning can occur in every situation in which people are required to make judgements and predictions. This was illustrated previously in the examples about the judges and the bank officials.

1.2 Case-Based Reasoning vs. Rule-Based Reasoning

Case-based reasoning is an alternative to rule-based reasoning for building expert systems. In case-based reasoning, the problem solver makes its inferences based directly on previous cases rather than by the more traditional approach of using rules. The case-based approach, however, has its advantages and disadvantages.

Rule-based systems solve problems by chaining rules of inference together. These systems can be flexible and produce good answers if the rules based on experience cover most of the possible situations in a domain; however, they can be slow and prone to errors, especially if the rule chain is long and and the problem to be solved involves many input factors. In a case-based system, experience will be more explicit since the complete description of how a problem was solved is stored as a separate entity (a case). In rule-based systems, an experience is stored as pieces of problem-solving knowledge scattered in a group of rules. However, the case-based system solutions will be restricted to the variations on known situations, that is, variations on whatever is found in its case base, thus producing approximate answers. On the other hand, it can provide quicker answers because there is usually a close connection between the input case and the retrieved solution, that is, the delay associated with the long rule chain in complex domains does not exist. If the retrieved solution from memory does not work for the current situation, it is adapted taking into account those features that are different from the current situation. Also, the answer in a case-based reasoner is better supported since it can be traced directly to an actual previous experience.

As domains become larger and more complex, rules are more difficult to obtain. A domain could involve many situations with a variety of outcomes and many combinations of inputs that will require a large amount of rules to make a working system. The development of such system may be a tedious, costly, and time-consuming task. With case-based reasoning, meanwhile, all possible situations do not need to be present since the available ones can be adapted to solve those that do not appear in the case base. Thus, the development time of a case-based system may be shorter than that taken to develop a rule-based system.

Frequently, knowledge engineers find themselves forcing the expert into generalizing pieces of his own problem-solving knowledge to be able to fit them into the so-called rules of thumb that are necessary to develop a rule-based system. On the other hand, case-based reasoning tries to follow the natural way in which experts reason by asking them for knowledge in the form of previous experiences. Sometimes, the cases are already available, especially in those domains, such as law, that require the recording of problems considered and their respective solutions. Thus, the knowledge acquisition process is facilitated. The library of previous cases also eases the construction of explanations or justifications for the solutions given to a problem by providing specific support data.

In summary, case-based reasoning can be used when it is difficult to formulate domain rules, but example cases are easy to get, or when a case library is already available. Even if rules for a domain can be formulated, case-based reasoning can be utilized if rules require more input information than what is normally available, or if using rules is expensive because the rule base is large or the average rule chain is long. Case-based reasoning can be particularly advantageous when cases with similar solutions have similar problem statements, that is, there exists a similarity metric that can be calculated for problem statements and a corresponding set of adaptation rules.

The purpose of the work reported here is to apply techniques of case-based reasoning to a specific domain, as explained in the next chapter, and study the issues involved in such application. Chapter 3 overviews case-based reasoning techniques and concepts and specifies the ones to be used for our target domain. Chapter 4 explains the functioning of the prototype that implements the techniques discussed in the previous chapter. The results of the prototype testing are shown in chapter 5. Before the conclusions of chapter 7, avenues for future research are presented in chapter 6.

CHAPTER 2

PROPERTY APPRAISAL AND CBR: PROBLEM STATEMENT AND PROPOSED SOLUTION

Property appraisal is a time-consuming task that requires a lot of research and may be quite expensive to whomever ask for such a service. If the expertise of appraisers could be captured in a system, it could be used by the appraisers themselves to cut down the time required to prepare an appraisal. The system could be seen as someone giving advice or suggestions based on the heuristics that try to emulate the expert's knowledge or experience. The system could also be used as a training tool to new appraisers.

There are several methods of appraisal: the cost approach, the market data or sales comparison approach, and the income approach. The cost approach is based upon the reproduction cost of the building plus the value of the land; the market data approach is based upon what similar properties are selling for in the market; and the income approach is based upon the amount of net income the property can produce. For a more detailed description of the methods, see (Creteau 1974; American Institute of Real Estate Appraisers 1988; Boyce and Kinnard 1984). An appraiser uses a combination of these methods to make a property valuation. However, the most popular, especially in the appraisal of residential properties, is the market data approach. The other two have a more limited application. Therefore, an attempt to automate property appraisal should follow the market data approach.

2.1 Market Data Approach of Appraisal

The market data approach of appraisal works on the premise that an informed buyer would pay for a property no more than the cost of acquiring an existing property with the same utility (Boyce and Kinnard 1984). The sale price in a transaction is then the reflection of the knowledge that both the seller and the buyer have of the market. This justifies the use of comparable or similar properties that have been sold recently to determine the market value of a property that needs to be appraised.

It is very unusual that the comparable properties be identical to the property to be appraised, or subject property, in every aspect. Adjustments must be made to account for any difference. In estimating the amount of adjustment to make for the presence and absence of any factor or for varying quantities of any factor in the comparable sales property as compared with the subject property, the only valid measure is evidence of the market reactions of buyers to such a difference. The principle involved is that each factor or element of comparison in a property has a contribution to value, and this contribution may be reflected in a sales price differential.

Cost is not always a good source of adjustments. If you added a new room to your house, you might want to add its cost to the value of the house. However, if comparable properties have been selling for less in your community, you will not be able to recover what you invested in your new room. Your room will have less value (with respect to its cost) in the eyes of buyers or the market in general. Cost to install or build is thus not the appropriate measure of the difference between two properties because it may or may not equal the sales price differential reflected in the market behavior of buyers.

Therefore, by studying transactions in the market area, an appraiser learns how much effect on value each different element of comparison produces. The comparables are analyzed in light of these differences and their sales prices are adjusted to reflect the value of the subject. When the differences are minor, adjusted sales prices of comparable properties provide a persuasive indication of value. When differences are more substantial, greater adjustments are required and the results are less reliable. Limits on the number and magnitude of adjustments are recommendable for this reason.

When making a comparison between two properties, it is not practical to take into account every possible difference between them. A set of elements of comparison should be defined. Elements of comparison are property characteristics that are important enough to make a significant difference in the value of a property. They usually include legal characteristics; transactions characteristics such as financing terms, conditions of sale (motivation), and market conditions (time); physical characteristics such as size and condition; and locational features such as neighborhood and site. However, the appraiser uses his own judgement to decide the actual set of elements of comparison he is going to use (Almy, Gloudemans, and Denne 1978).

After adjustments are made for financing terms and market conditions (time), if they were necessary, an appraiser can isolate the effects of physical and locational characteristics by comparing prices of pairs of properties that are similar except for a single physical or locational difference. In practice, several matched pairs should be isolated from the sales of comparable or similar properties so that the appraiser's conclusion will be based on an adequate sample. Several methods can be used to study market data for this purpose. One of them is the paired data set analysis, which is facilitated with the use of market data grids, which organize the subject property data and the information of the comparables in adjacent columns. A simple example of their use is shown in chapter 6. Market data grids may help to identify which comparables have the fewest differences from the subject and should be given the most weight in reconciliation. They can reveal pairs of comparables that differ in only one feature, thus helping to determine the value of the adjustment corresponding to the dissimilar feature. Also, they ease the totaling of adjustments to calculate the value differences between the subject property and each comparable.

The market data approach is thus an attempt to measure the reactions of typical buyers and sellers. In this way, it becomes, to some extent, a simulator of market behavior.

2.2 Proposing Case-Based Reasoning

A purely rule-based system in property appraisal could be difficult to implement because rules need to be written on generalizations based on observations of the market, which is a dynamic entity. Since it is dynamic, rules might have to be updated frequently after studying market data. On the other side, if a case-based system is used, new recent cases are added, and the dynamic nature of the market is analyzed by the system itself each time a new appraisal is done. A particular generalization is done for the current instance or situation. Besides, the availability of sales records and databases, which can be readily converted to case bases, gives an advantage to case-based reasoning.

Property appraisal lends itself well to case-based reasoning. Since the method to be followed is the market data approach, the case base will consist of descriptions of all kinds of properties previously appraised and sold. The expert will provide the heuristic knowledge necessary to adapt or adjust the values from the real estate properties in the case base that are the most similar in terms of characteristics or features to the property being appraised.

Case-based reasoning is not concerned with mathematical models, which, since the 1960's, have been the basis of the efforts to automate property appraisal, especially mass property appraisal for government purposes. This field has been known as computer-assisted valuation (CAV). Mathematical modelling techniques, including multiple regression analysis and feedback analysis, have been used for the market data or sales comparison approach of appraisal (Woolery and Shea 1985; Adair and McGreal 1988; Carbone 1987). However, again, the purpose of this case-based system is to capture the heuristic knowledge utilized by the property appraiser to handle his previous experiences or the experiences of others in this field, and integrate that into an expert system that uses case-based techniques as opposed to other artificial intelligence reasoning techniques as the rule-based and the model-based. If statistical methods were used to figure out the general trends in the market, concrete examples to backup the decisions and provide explanations would not be available.

In this paper, a case-based prototype system for automated property appraisal is thus demonstrated. The domain is limited to single-family residential property appraisal because whatever is developed for this subset of the domain may be easily extended later to the other types of properties and other types of appraising tasks. The prototype was developed in the Symbolics 3640 Lisp Machine. LISP was used as the language for system development because of its facilities for incremental prototyping. The investigation is directed toward the issues involved in the application of case-based reasoning techniques to the automation of property appraisal. The development of the prototype is intended to illustrate whether a case-based system provides fairly good solutions (in this case, fair appraisals) utilizing a relatively small case base, thus avoiding the large rule base and the long rule chains necessary for this domain if pure rule-base reasoning is used.

CHAPTER 3

CASE-BASED REASONING CONCEPTS AND THEIR APPLICABILITY TO PROPERTY APPRAISAL

The general procedure of case-based reasoning, as presented by (Kass and Leake 1988), follows these four steps:

1. Retrieve a case from the case memory.
2. Compare the retrieved case to the current situation. Evaluate the relevance of the past experience to the current situation.
3. If necessary, adapt the retrieved case in order to generate a case description that applies to the current situation.
4. Use the previous case to generate inferences that can be transferred to help process the current input.

The generality of this procedure, however, raises several important questions about the specific techniques that can be used to implement each step:

1. What is a case? What are the components of cases?
2. How are cases retrieved? How is the memory organized for such effect?
3. How is the applicability of an old case determined?
4. How does a case adaptation proceed?

5. How are insights from the old case applied to the new situation?

In our case, the interest is in determining which CBR techniques among the ones available will serve better the purpose of developing the case-based appraising prototype.

3.1 Case Representation

The most popular means of representation of cases in CBR is the MOPs or Memory Organization Packages (Riesbeck and Schank 1989). MOPs are used to organize events related to a particular domain by means of the combination of AI concepts, such as frames, abstraction, and inheritance. They are particularly useful to represent the dynamic nature of the knowledge bases of most domains, especially design domains, such as architecture, programming, and plan generation, which require the construction of a solution that must satisfy several goals.

Instead of events per se, the cases to be represented in property appraisal prototype consist of the descriptions of properties sold during a specific period of time in a specific geographic area. So, they do not require the complexity of MOPs. The frames that are used to represent cases in the prototype include slots for each of a set of elements of comparison, that is, the features or characteristics of a real estate property that are deemed important for the determination of value.

Even when the elements of comparison may vary somewhat from appraiser to appraiser and from market to market (Weber, 1990), a set of elements of comparison was chosen after some interviews with appraisers (Shearer 1990; Fieldson 1990a) and the study of several references (Boyce and Kinnard 1984; American Institute of Real Estate Appraisers 1988):

* Living area in square feet,

* No. of bedrooms,

* No. of bathroom,

* Style of the house,

* Age of the house,

* Location (neighborhood),

* Date of sale,

* Type of cooling equipment,

* Type of heating equipment,

* Type of garage,

* Site or lot size, and

* Availability of a swimming pool.

This is a small set for the purpose of prototype development. However, eventually the additional elements included in the Uniform Residential Appraisal Report (URAR) may be used. The URAR is the most popular standard form used by appraisers to present the results of their work, that is, the appraisal.

3.2 Case Retrieval: Memory Organization

The way cases are organized in memory influences the speed and effectiveness with which cases are retrieved from it. There is a great variety of memory organizations that go from flat memory to hierarchical organizations, and from serial search to parallel search.

Since finding the best match in a relatively small case base is our goal, flat memory is the best way to organize the case base. This is done in the case-based reasoner HYPO (Ashley and Rissland 1987, 1988), whose domain is patent law. Also, this method was used in the version 0 of the CBR Tool of Cognitive Systems, Inc. (Riesbeck 1988b), which is being developed using a case base of the battle planning domain. Flat memory consists of simply storing cases in a list or array sequentially.

If the case base were a large one, retrieval from flat memory could be expensive since every case is processed for a match. This could be solved by using some type of hierarchy to organize cases with similar features. Example of this can be found in the shared feature networks of MicroMOPs (Riesbeck and Schank 1989), which is an implementation of a miniature MOP-based memory system. In shared feature networks, cases are stored in the nodes of a tree, which subdivides the space of cases according to the features they share. Another technique to organize cases according to their features to save time on retrieval are the discrimination nets used in case-based reasoners such as CHEF (Hammond 1986, 1987), which is used for recipe generation; CYRUS (Kolodner 1983a, 1983b), which organizes the events related to Cyrus Vance's job as the

U.S. Secretary of State; MEDIATOR (Kolodner and Simpson 1989), used in political dispute mediation; and JULIA (Kolodner 1987), a menu designer for catering.

The flat memory problem with large databases could also be addressed by performing parallel search with flat memory as suggested in the MBR or Memory-Based Reasoning paradigm (Stanfill 1988) discussed in the next section. Parallel search could be attempted with a Connection Machine or a similar parallel architecture equipment by storing a different case as a feature vector on each of the processors of the machine. Retrieval is thus made by parallel matching of feature vectors.

The hierarchical networks and the parallel search algorithm have their benefits, but applying them to our prototype will not improve its performance. Hierarchical networks would utilize much more memory than what is required for the flat memory organization; and this is not justified for a small case base. On the other hand, parallel search requires expensive equipment, which might not be justified for our application.

3.3 Case Retrieval: Looking for the Best Case

When looking for similar cases in memory, for some domains, it will suffice to find any similar case. However, for our target domain, property appraisal, the goal is to find the best cases, that is, the properties that among all the properties in memory are the most similar to the subject property. For this, a best-match algorithm similar to the one presented by the MBR or Memory-Based Reasoning paradigm needs to be applied (Stanfill 1988; Stanfill and Waltz 1986, 1988).

MBR is a variation of case-based reasoning that is not so concerned with memory organization since it uses flat memory. Each case (frame) has the same structure, but pointers between cases (as are found in MOPs) are not allowed at this time. MBR does not spend time organizing memory in some type of network, because, usually, with this type of memory organization, the global best matches cannot be obtained, especially if the indexes that are chosen to group the cases are not the most appropriate. Indexes are those elements of comparison between cases that determine the relevance of cases. MBR differs from our prototype in that it uses parallel search because it is directed to large case bases.

We are particularly interested in the global best matches in property appraisal because we are looking for the most precise figures of value we can find, even when the appraised value is still an estimate or approximation. In both MBR and our prototype, the best-match algorithm takes the target case, computes a similarity metric between the target case and every case in memory and retrieves the best matches from memory. It is unusual to find an exact match for a case since this means that every feature in the case was matched exactly. So, most of the cases retrieved from memory are partial matches. The degree of partial match, or the aggregate match score, depends on the relative importance and degree of match or relative similarity of corresponding features. The computation of this aggregate match score is implemented in the prototype by using the technique known as static evaluation, in which both the relative importance and degree of match of features are assigned numerical values. The best cases are the ones with the highest aggregate match scores. The version 0 of the CSI's CBR Tool, which was referenced previously, is an example of a case-based reasoner using static evaluation; however, its implementation of the similarity metric is different from the one in our prototype.

For the actual matching of values of corresponding features in two cases, there are two possibilities in the prototype. If the values are numeric, they are matched if both of them are found within a pre-specified range. Meanwhile, if the values are concepts (by this, we refer to any other non-numeric values), the degree of similarity is known if a similarity link was defined for the particular pair of concepts. Though they are implemented differently, similarity links and the similarity networks described in (Bailey, Thompson, and Feinstein 1987, 1988) have the same purpose. Both relate pairs of concepts with a match factor. The higher the factor, the greater the degree of similarity. In similarity networks, the concepts related are each an object, which is represented by a frame or other similar representation. The similarity links that our prototype uses include the pairs of concepts and their respective match factor in a list. At the same time, this list is part of a larger list carried by a slot of a frame representing a feature of a property. More details about similarity links are discussed in the next chapter, in which the prototype is described.

3.4 Case Adaptation

Once the best cases in memory, that is, the cases with the highest aggregate scores, are identified, they are retrieved from memory to become the official comparable properties. Then, the adaptation phase starts. This phase takes care of the application of adjustments to the sale price of comparables to get a better indication of the value of the subject property. Even though appraisers choose only three comparable properties to be adjusted and be included in their report of an appraisal job, our prototype chooses ten comparables for the reasons explained later in section 3.5.

According to (Riesbeck and Schank 1989), probably, the best understood adaptation technique is one called parameterized solutions. What is going to be adapted is determined by the differences between problem specifications, that is, between the subject property and its comparable from the case base. When a case is retrieved for an input situation, the old and new problem descriptions are compared along the specified parameters. The differences are then used to modify the solution parameters in the appropriate directions. For example, in JUDGE (Riesbeck and Schank 1989), a crime has parameters such as "heinousness," "seriousness of motive," and "remorse." Likewise, there are parameters involved in the solution, such as length of imprisonment, availability of parole, fines, and others.

In the property appraisal domain, the parameters of comparison for cases were discussed in section 3.1. Among them, we have living area, style of house, and type of garage. The differences in these parameters are used to decide what adjustments are needed to "adapt" the sale price of each comparable to the value of the subject property. Thus, the solution parameters are the sale prices of the comparables.

However, to get the dollar amount of the adjustments needed to compensate for the differences between the previous and the current case, we need to use another adaptation method. From among the adaptation methods currently developed, as surveyed in (Kolodner and Riesbeck 1989), the most appropriate for this is the critic application. Critics are implemented as a rule-based system and indexed by the feature that triggers them. Care is exercised to keep under control the number of critics because like rules in a rule-based system, their efficiency degrades as the set gets large.

Therefore, like in PERSUADER (Sycara 1988), a case-based reasoner used in labor contract dispute resolution, the critics in our prototype are used for parameter adjustment.

The critics provide the dollar-amount adjustment necessary to compensate for the difference of the comparable case with respect to the subject property. The adjustment amounts for differences in each feature are obtained from the expert.

The use of parameterized solutions does not imply that there is a simple formula for getting some set of problem parameters to a solution. In each program, the parameterized solutions are of value in modifying an existing solution, not creating a new solution from scratch. Combined with the critic application, it is a simple but powerful way to augment a case library, but it is not a replacement for a good set of cases (Riesbeck and Schank 1989). Adaptation critics are essentially mini-problem solvers. In case-based reasoning, adaptation critics are primarily a labor and space-saving device. They allow minor variants of stored cases to be generated dynamically as needed. The core of case-based reasoning, however, depends on the case library, not on the adaptation rules (Riesbeck 1988a).

3.5 Determination of a Solution

Once all adjustments are obtained, they are added or subtracted, as appropriate, from the sale price of the corresponding comparable property. In this way, an adjusted value, which better reflects the value of the subject property, is produced for each one of the three comparables. This is done in the prototype for a total of ten comparable properties retrieved from memory.

It may seem redundant to adjust so many properties if appraisers just need three to support their report and if the three best comparables could be identified by their aggregate match scores during the case retrieval phase. However, even though the most similar cases may be identified, some type of technique is needed to make sure that the sale prices of the comparables chosen are a reflection of the average market prices for that type of property, and not the result of a sale under unusual conditions. Having more than three comparables allows for a better analysis of the tendencies of the prices in the market and for the elimination of those comparables that do not fall within the typical range of prices. Even those that are among the first three best comparables, which are the ones that would logically be chosen to calculate the final appraised value of the subject property if this test for market consistency were not being made, could be eliminated from consideration if their sale prices are far from the typical value assigned by the market to the type of property being considered.

The goal of our program is then to come up with a single appraised value of the subject property. To do so, the three comparables whose adjusted values better reflect the value of the subject are combined in a weighted average, whose implementation is explained in section 4.4. The method to select these three comparables out of the ten available is also discussed in the next chapter.

The use of more than one of the best matches to be adapted and obtain a solution serves as a consistency checker in property appraisal as well as in case-based reasoning in general. In addition to our program, other case-based reasoner that uses this technique is the BURN Sizer (Kolodner and Riesbeck 1989), which estimates the computer resources of an organization. BURN Sizer uses a case libray of organizations that includes the description of each organization and the computer resources it needs. By comparing the current organization or company with the ones in the case library and choosing the two most similar organizations in the library, two computer resource specifications are obtained. This case-based reasoner then adapts the resource needs of the best two matching organizations to fit the current input. If the two answers are close, then BURN Sizer has a higher confidence in its answer.

In the next chapter, the implementation of the techniques presented in this one is explained in more detail. The program written as a prototype is described.

CHAPTER 4

CASE-BASED APPRAISER: PROGRAM DESCRIPTION

This chapter describes the prototype of the program that implements the market data approach in property appraisal. The prototype is referenced here as the CBA or Case-Based Appraiser. The program was developed using the LISP language in a Symbolics 3640 Machine.

The object-oriented facilities of the Symbolics Machine were used to organize the data. Each object is called a flavor and is organized in a frame-based structure that has a set of slots associated with it. Flavors are used because they ease the access of data during program execution.

The following description starts with a view of the flavors defined in the program. After that, the procedures followed for case retrieval and adjustment determination are discussed. Then, the guidelines to choose the best comparables to obtain the appraised value of the subject property are explained. Finally, the user interface facilities are presented very briefly.

4.1 Flavor Definitions

CBA uses five different flavors. Their names are PROPERTY, CASE, COMPARABLE, MATCHING-FACTORS, and CURRENT-

CASE. PROPERTY serves as a base flavor for CASE and CURRENT-CASE. This means that CASE and CURRENT-CASE inherit variables and access functions, among other things, from the flavor PROPERTY. Meanwhile, COMPARABLE is built on flavor CASE. Throughout the program, instances of all these flavors are used.

4.1.1 PROPERTY Flavor

The PROPERTY flavor was created to serve as a base flavor. It is used to group a common set of traits or characteristics shared by the other flavors.

The set of slots for this flavor is composed of the following:

* LIVING-AREA,
* BEDROOMS,
* BATHROOMS,
* STYLE,
* YEAR,
* LOCATION,
* SALE-DATE,
* COOLING,
* HEATING,
* GARAGE,
* SITE, and
* POOL.

These slots correspond to the elements of comparison that were identified in section 3.1.

During the case retrieval stage, a similarity metric is calculated to determine how close is a case in memory to the subject property in terms of their characteristics. For this metric, a weight needs to be assigned to each one of the features or elements of comparison of a property to reflect their relative contribution to value and their relative importance when searching for comparables. Therefore, even though the original intention was to make PROPERTY exclusively a base flavor, it was observed that by making an instance of this flavor, the weights needed for the case retrieval stage could be stored. This particular instance of PROPERTY is stored in the variable *WEIGHTS*. The weights are assigned by the expert in integer numbers.

4.1.2 CASE Flavor

The CASE flavor is intended to represent all the cases in memory, that is, the case library. The cases are descriptions of properties that have been sold during a specific period of time at a particular geographic area. All the instances of this flavor are stored in the global variable *CASES* and are named by hyphenating the word "case" and a number (for example, CASE-1, CASE-2, CASE-3, and so on).

In addition to the slots associated to the elements of comparison, inherited from flavor PROPERTY, CASE contains the following slots:

* ADDRESS,

* SALE-PRICE,

* AGGREGATE-MATCH-SCORE,

* LIVING-AREA-MATCH-FACTOR,

* BEDROOMS-MATCH-FACTOR,

* BATHROOMS-MATCH-FACTOR,

* STYLE-MATCH-FACTOR,

* YEAR-MATCH-FACTOR,

* LOCATION-MATCH-FACTOR,

* SALE-DATE-MATCH-FACTOR,

* COOLING-MATCH-FACTOR,

* HEATING-MATCH-FACTOR,

* GARAGE-MATCH-FACTOR,

* SITE-MATCH-FACTOR, and

* POOL-MATCH-FACTOR.

As seen, a slot for the address and a slot for the sale price of each property in the case base is added. Also, a slot for the match factor for each feature or element of comparison of the property is added. This match factor will represent the degree of similarity between the value of the feature of the property in the case base and the value of the same feature in the subject property. The value of the match factor is obtained from the data stored in the instance of the MATCHING-FACTORS flavor discussed later. The AGGREGATE-MATCH-SCORE slot is used during the case retrieval stage to store the value assigned to the property after the calculation of the similarity metric used to compare it to the subject.

4.1.3 COMPARABLE Flavor

The COMPARABLE flavor is used to represent the ten properties (cases) that are chosen as the best matches to the subject property. A separate flavor is used for comparables because they will have a larger number of slots than the rest of the cases. Comparables will comprise a fraction of the case base and it is not justified to have such a large number of slots for all the cases. The instances of this flavor are stored in the global variable *COMPARABLES*, and their names are formed by hyphenating the word "comparable" with a number (for example, COMPARABLE-1, COMPARABLE-2, and COMPARABLE-3).

This flavor inherits all the slots in CASE, which in turn, includes all the slots in PROPERTY. In addition, COMPARABLE adds these slots:

* SOURCE,
* LIVING-AREA-ADJUSTMENT,
* BEDROOMS-ADJUSTMENT,
* BATHROOMS-ADJUSTMENT,
* STYLE-ADJUSTMENT,
* YEAR-ADJUSTMENT,
* LOCATION-ADJUSTMENT,
* SALE-DATE-ADJUSTMENT,
* COOLING-ADJUSTMENT,
* HEATING-ADJUSTMENT,
* GARAGE-ADJUSTMENT,
* SITE-ADJUSTMENT,

* POOL-ADJUSTMENT,

* ADJUSTED-VALUE,

* GROSS-ADJUSTMENT,

* NET-ADJUSTMENT, and

* COMFORT-FACTOR.

The slot SOURCE will point to the case from which the property information came from for the particular comparable. The GROSS-ADJUSTMENT slot holds the sum of the absolute value of all the adjustments in a comparable; meanwhile, the NET-ADJUSTMENT slot is used to store the sum of the adjustments taking into account the signs. The amount saved in NET-ADJUSTMENT is the same that is either added or subtracted, as appropriate, from the sale price of the property to get the adjusted value of the comparable, which is then stored in slot ADJUSTED-VALUE.

The COMFORT-FACTOR slot contains a number between 0 and 1.0 that is assigned after adjustments are generated and before the final calculation of the appraised value for the subject property; section 4.4 explains in detail the role of these comfort factors in CBA. The rest of the slots store whatever adjustment is needed for the specific feature they represent during the adjustment generation stage. These adjustments are dollar amounts obtained from the critics stored in the instances of the MATCHING-FACTORS flavor.

4.1.4 CURRENT-CASE Flavor

The flavor CURRENT-CASE represents the subject property, that is, the property to be appraised. So, there is only one instance of this flavor and it is assigned to the variable *MY-PROPERTY*.

Like flavor CASE, CURRENT-CASE inherits the slots from PROPERTY; however, it only adds two more slots to its instances: ADDRESS and APPRAISED-VALUE. This flavor does not need any slots for match factors because the subject property does not need to be compared to itself. Instead of the SALE-PRICE slot, CURRENT-CASE adds APPRAISED-VALUE to hold the final result of the appraisal.

4.1.5 MATCHING-FACTORS Flavor

The MATCHING-FACTORS flavor contains information that needs to be associated with each of the features or elements of comparison of a real estate property. Consequently, an instance of MATCHING-FACTORS is made for each feature: LIVING-AREA, BEDROOMS, BATHROOMS, STYLE, YEAR, LOCATION, SALE-DATE, COOLING, HEATING, GARAGE, SITE, and POOL. The following slots may be found in the MATCHING-FACTORS flavor:

* VALID-VALUES,

* VALID-RANGE,

* CONCEPT-MATCHING-PAIRS,

* DIFFERENCE-MATCHING-RANGES, and

* ADAPTATION-CRITICS.

The contents of these slots varies depending of the type of values the feature has. The features that are being used may be classified as those with numeric values and those that have concepts as values. By concepts, it is meant anything else, but numbers. LIVING-AREA, BEDROOMS, BATHROOMS, YEAR, SALE-DATE, and SITE are in the first group. STYLE, LOCATION, COOLING, HEATING, GARAGE, and POOL are in the latter. Let us take LIVING-AREA and GARAGE as examples of their respective groups.

Figure 1 shows an example of the contents of the slots of the instance LIVING-AREA of the flavor MATCHING-FACTORS.

```
LIVING-AREA is an instance of flavor MATCHING-FACTORS.
     VALID-VALUES: 'number
     VALID-RANGE: '(500 3000)
     CONCEPT-MATCHING-PAIRS: nil
     DIFFERENCE-MATCHING-RANGES: '((-50 50 .95)
                                   (50 100 .85)
                                   (-100 -50 .85)
                                   (100 150 .5)
                                   (-150 -100 .5))
     ADAPTATION-CRITICS: '(50 1000)
```

FIGURE 1. Example of the Contents of the Slots of the Instance LIVING-AREA of Flavor MATCHING-FACTORS.

Since this feature has numeric values, the slot VALID-VALUES contains the atom NUMBER and the slot CONCEPT-MATCHING-PAIRS is NIL. The VALID-RANGE slot contains a list of the two limits of the range of valid values for the feature. If the numeric feature has several numeric values, like SALE-DATE, which has values for the day, the month, and the year, the slot VALID-RANGES may contain a list of lists. Each sublist includes, in that case, the limits for the range of one of the values.

When comparing a feature with a numeric value to the corresponding feature in the subject property, the difference of both values is obtained. This difference is used to decide the degree of matching of both features with the help of the list of similarity links included in the slot DIFFERENCE-MATCHING-RANGES. Similarity links were introduced to our discussion in section 3.3. Each similarity link is implemented as a list, whose first two numbers establish a range and whose third number represents the match factor to be assigned if the calculated difference falls within the range. There can be any number of links or no links at all in the list stored in this slot. This matching procedure is further explained in the case retrieval section.

The ADAPTATION-CRITICS slot of this type of feature includes a single list with two elements; the first one represents one unit of difference between the pair of values of the feature and the second is the corresponding dollar amount of adjustment to be made per unit of difference. This second element could also be a percentage per unit of difference to be applied to the sale price to get the dollar amount of adjustment. The adaptation procedure that utilizes these critics is discussed in the section on adjustment generation.

Figure 2 shows an example of the contents of the slots of the instance GARAGE of the flavor MATCHING-FACTORS. In the case of features with concepts as values, the VALID-VALUES slot lists all the possible values the feature may have. Both the DIFFERENCE-MATCHING-RANGES slot and the VALID-RANGE slot are always NIL because there are no numbers in these concept-based features to match against a range. In figure 2, the valid values for the feature GARAGE are a one-car garage, a two-car garage, a carport, or no garage at all (none). In the CONCEPT-MATCHING-PAIRS slot, a list of similarity links is stored. Like in features with numeric values, the similarity links are implemented with a list of three elements. However, the first two elements of this list correspond to the two concepts being compared and the third element corresponds to the match factor of both concepts. More details about the utility of these similarity links are presented in the section of case retrieval.

```
GARAGE is an instance of flavor MATCHING-FACTORS.
     VALID-VALUES: '(1-car 2-car carport none)
     VALID-RANGE: nil
     CONCEPT-MATCHING-PAIRS: '((1-car 2-car .4))
     DIFFERENCE-MATCHING-RANGES: nil
     ADAPTATION-CRITICS: '((none 1-car 2000)
                           (none 2-car 4000)
                           (none carport 500))
```

FIGURE 2. Example of the Contents of the Slots of the Instance GARAGE of Flavor MATCHING-FACTORS.

The critics in this type of feature are implemented as a list of lists in the ADAPTATION-CRITICS slot. The first element on each of the sublists is always the same value and is known as the base concept. The rest of the valid values of the feature take turns to occupy the second position of the sublists. No value can be repeated in the second position of the sublists. Each one of the valid values should be used once and only once. The third element of each sublist is the dollar amount of adjustment to be made if that pair of concepts is encountered. The amount of the adjustment should correspond for a situation in which the comparable property has the base concept as the value of its feature and the subject property has any of the others. Other combinations of values are derived from this information as explained in the adjustment generation section.

The information needed to establish the similarity links, the adaptation critics, and the weights for each feature is obtained from the expert. These measures comprise the heuristic knowledge necessary to handle the information in the case base. The user interface takes care of storing the information of these slots in the proper manner. Once this matching and adaptation information and the case base are available, the process of case retrieval may start.

4.2 Case Retrieval

The process of case retrieval in CBA is performed in two steps: first, the calculation of the aggregate match score for all the cases in memory and, second, the selection of the comparables.

4.2.1 Calculation of Aggregate Match Scores

The aggregate match score is the result of the similarity metric calculated to identify which cases in memory best resemble to the property currently being appraised. The first step taken by CBA to obtain these global scores is to take each case in memory and identify how close is the value of each of its features to matching the value of the corresponding feature in the subject property.

The value of a specific feature of a case in memory, which is a previously appraised and sold property, is referenced by the variable PREVIOUS-VALUE in CBA. Meanwhile, the value of the same feature in the current case, that is, the property to be appraised, is referenced by the variable CURRENT-VALUE. The degree of similarity of PREVIOUS-VALUE and CURRENT-VALUE is reflected in the match factor assigned to the relationship. This match factor is stored in the case in memory at the appropriate match factor slot. For example, if CASE-10 and the subject property, *MY-PROPERTY*, are being compared in terms of the type of garage they have, the match factor is stored in the GARAGE-MATCH-FACTOR slot of CASE-10. If both PREVIOUS-VALUE and CURRENT-VALUE are exactly equal, the match factor to be stored will be equal to 1.

However, if PREVIOUS-VALUE and CURRENT-VALUE are not exactly equal, to perform partial matching we need to use the information provided by the instances of the MATCHING-FACTORS flavor. Remember that each instance of this flavor correspond to a feature in cases. By checking the VALID-VALUES slot of the MATCHING-FACTORS instance, it can be determined whether the feature has numbers or concepts as its values. Once this is known, the right slot may be accessed for the matching information, which is in the form of similarity links.

If the feature has numeric values, the DIFFERENCE-MATCHING-RANGES provide the similarity links. As discussed in section 4.1.5, each similarity link associates all values within a range with a particular match factor. However, before using the information in the links, PREVIOUS-VALUE is subtracted from CURRENT-VALUE to obtain a difference, which is the one checked against all the ranges in the similarity links. If this difference falls within a range, the match factor associated with that range is the one stored appropriately. For example, referring to the instance LIVING-AREA of flavor MATCHING-FACTORS as shown in figure 1, if the difference falls between -50 and 50, the match factor is .95; if it falls between 50 and 100, the factor is .85; and so it follows. If the difference does not fall within any range, the match factor is 0.

If the feature, otherwise, has concepts as values, the CONCEPT-MATCHING-PAIRS slot contain the necessary similarity links. If the pair of concepts represented by PREVIOUS-VALUE and CURRENT-VALUE is equal to any of the pairs in the links, no matter if they are in the right order or the reverse order, the match factor associated with the pair is properly stored. In the instance GARAGE of figure 2, the only similarity link available is (1-car 2-car .4). If for example, PREVIOUS-VALUE and CURRENT-VALUE correspond to 1-CAR and 2-CAR or vice versa, the match factor to be stored in the GARAGE-MATCH-FACTOR slot of the case in memory is 0.4. Again, if the pair of concepts does not match any pair in the links, the match factor is 0.

If for some reason, no information is found in either the DIFFERENCE-MATCHING-RANGES slot or the CONCEPT-MATCHING-PAIRS slot when they are accessed, then the match factor to be stored is 0. Usually, when there is no information in these slots, it is because the expert thinks that no partial matching is adequate for the specific feature. In that case, only exact matches produce a match factor different from 0.

Once all the match factors have been determined the slots of a case might look like in figure 3. The values in the match factors slots will vary depending on what values the features of the subject property have. The values shown in figure 3 are just for illustrative purposes. At this stage, every slot in a case has a value, except for the AGGREGATE-MATCH-SCORE slot.

In order to get this aggregate score, we need all the match factors and also the weights that are assigned by the expert to each feature to indicate their relative importance. These weights are stored in the instance *WEIGHTS* of flavor

PROPERTY. The contents of the variable *WEIGHTS* might look like in figure 4.

```
CASE-10 is an instance of flavor CASE.
     ADDRESS: "888 Stratton Ave., Deltona"
     SALE-PRICE: 60000
     AGGREGATE-MATCH-SCORE: unbound
     LIVING-AREA: 1500
     BEDROOMS: 3
     BATHROOMS: 2
     STYLE: 'ranch
     YEAR: 80
     LOCATION: Deltona/Area-18
     SALE-DATE: '(7 15 90)
     COOLING: 'central
     HEATING: 'central
     GARAGE: '2-car
     SITE: '(80 125)
     POOL: 'no
     LIVING-AREA-MATCH-FACTOR: 0.8
     BEDROOMS-MATCH-FACTOR: 1.0
     BATHROOMS-MATCH-FACTOR: 1.0
     STYLE-MATCH-FACTOR: 1.0
     YEAR-MATCH-FACTOR: 0.5
     LOCATION-MATCH-FACTOR: 1.0
     SALE-DATE-MATCH-FACTOR: 1.0
     COOLING-MATCH-FACTOR: 1.0
     HEATING-MATCH-FACTOR: 1.0
     GARAGE-MATCH-FACTOR: 0
     SITE-MATCH-FACTOR: 0
     POOL-MATCH-FACTOR: 0
```

FIGURE 4. Example of the Contents of the Slots in an Instance of Flavor CASE after Match Factors Have Been Determined.

To finally calculate the aggregate match factor, an addition is performed on the products of the match factor and the weight corresponding to each feature. For example, for the feature LIVING-AREA and referring to figures 3 and 4, the product (0.8 * 3) must be performed. This product is added to the products of the other features and the result is stored in the AGGREGATE-MATCH-SCORE slot of the corresponding case. The process is repeated for every case.

```
*WEIGHTS* is an instance of flavor PROPERTY.
     LIVING-AREA: 3
     BEDROOMS: 2
     BATHROOMS: 2
     STYLE: 2
     YEAR: 2
     LOCATION: 3
     SALE-DATE: 3
     COOLING: 1
     HEATING: 1
     GARAGE: 1
     SITE: 2
     POOL: 1
```

FIGURE 4. Example of the Contents of the Instance *WEIGHTS* of Flavor PROPERTY.

4.2.2 Selection of the Best Comparables

It is important that the most similar cases are chosen to avoid large adjustments: the larger the adjustments, the less precise the appraised figures are. This is the justification to calculate the aggregate match score for every case in the global variable *CASES*. To choose the top cases, the cases are sorted in descending order according to the value of their aggregate match score and stored in the global variable *SORTED-CASES*.

The cases with the highest aggregate match scores are the best matches or comparables. This means that the first ten cases in the variable *SORTED-CASES* are the ones that provide the necessary information for the ten instances of the flavor COMPARABLE. After the instances are created, each comparable has filled slots for each of the features of the property, the address, the sale price, the source (that is the CASE instance from which the information was obtained), the adjusted price (initially equal to the sale price), and the match factors for each feature. The adjustment slots for each feature are unbound at this stage.

4.3 Adjustment Determination

The steps for adjustment determination are comprised by the identification of those features in each comparable that do not match the subject property, the activation of the appropriate critics to identify the amount of adjustment, and the application of the adjustment amounts to the sale price of each comparable to establish its adjusted value.

4.3.1 Identification of Differing Features in Comparables

The first step in adjustment determination is to identify which features in the comparables need adjustment. Not all differing features require an adjustment. Appraisers use their judgement to decide which differences are important enough to call for an adjustment.

CBA uses the match factor determined for each feature during case retrieval to decide whether to make an adjustment. Each match factor is taken and compared to a global threshold value called *MATCH-FACTOR-THRESHOLD*. Any match factor greater or equal to the threshold is changed to be 1. On the other hand, any match factor under the threshold is changed to

0. In this way, all the features that need adjustment have a match factor of 0. The threshold is determined by the expert.

4.3.2 The Role of Critics

The dollar amounts of adjustments are obtained from the expert and saved as adaptation critics in the instances of the MATCHING-FACTORS flavor. An alternative, which is left as a possible avenue for future research, to obtain the adjustments is the automated adjustment generation (AAG). Impressions about AAG are presented in chapter 6.

An adaptation critic can be viewed as a type of rule that is accessed when the particular feature it is associated with needs an adjustment and is triggered when the condition it represents is met. As explained in section 4.1.5, the critics are grouped by features and their application varies depending on the type of values the feature has. So, when two numeric features are compared, the difference in their values is divided by the predetermined unit of difference to decide how many times the dollar amount of adjustment is going to be applied (added or subtracted as appropriate) from the corresponding comparable property sale price.

For features with numeric values, we can refer to figure 1. These features contain a single critic, which gives a unit of difference and a fixed dollar amount to serve as adjustment per unit of difference. For example, the subject property, *MY-PROPERTY*, and, say, COMPARABLE-2 may be differing in their feature LIVING-AREA, which has the information of figure

1 associated with it. *MY-PROPERTY* has 1700 square feet of living area and COMPARABLE-2 has 1500, which means that there is a difference of 200 square feet. The critic in figure 1 indicates that for every 50 square feet of difference, an adjustment of 1000 dollars should be applied. Consequently, in this example, an adjustment of 4000 dollars is made to the comparable. The adjustment is stored in the slot LIVING-AREA-ADJUSTMENT of COMPARABLE-2.

It is important to note that the difference between the features is obtained by subtracting the value in the comparable from the value in the subject property. In this operation, the sign is important because it determines if the adjustment to the comparable is negative or positive. A negative difference means that the comparable is superior to the subject property and its value should be adjusted downward to equate to the value of the subject property. A positive difference means the opposite, that the comparable is inferior to the subject in the feature being observed and its value should be adjusted upward to be put at the same level as the subject property. In our previous example, since the difference is positive, the adjustment is also stored as a positive value.

For features with concepts as values, we can refer to figure 2. These features contain a list of critics, each associating a pair of concepts with a dollar amount of adjustment. The first concept that appears in the critic is known as the base concept as explained in section 4.1.5. The amount of the adjustment is assigned by the expert assuming a situation in which the comparable property has the base concept as the value of its feature and the subject property has any of the others. Assume now that *MY-PROPERTY* and COMPARABLE-2 differ in the feature GARAGE; *MY-PROPERTY* has a 1-car garage and COMPARABLE-2 has no garage. According to figure 1, if the given comparable has value NONE and the subject is 1-CAR, the adjustment is 2000. 2000 is then stored in the GARAGE-ADJUSTMENT slot of COMPARABLE-2.

Other combinations of values are derived from this information in figure 1. For that, CBA associates each concept of the feature with an adjustment figure. The base concept is always associated with 0 and the rest of the concepts is associated with the adjustment figure of the critic in which they appear. Thus, in the case of GARAGE, this results:

* NONE (base concept): 0,

* 1-CAR: 2000,

* 2-CAR: 4000,

* CARPORT: 500.

Again here, the sign for the adjustment is also important. So, to obtain the correct sign and magnitude of the adjustment, take the adjustment figure associated with the concept of the feature in the subject and subtract from it the adjustment amount associated with the concept in the comparable. Some examples are shown in table 1.

4.3.3 Determination of Adjusted Value of Comparables

After all the features whose match factor was 0 obtained an appropriate adjustment, the adjustments of a specific comparable may be totaled and applied to the sale price of the comparable. This is done by going through all the adjustment slots, that is, LIVING-AREA-ADJUSTMENT, BEDROOMS-ADJUSTMENT, BATHROOMS-ADJUSTMENT, and so on, for each comparable. The slots with no adjustment are ignored, but the figures that appear in the filled slots are added together taking into account the signs. Once added, the total net adjustment amount of the particular comparable is applied to its sale price and the result stored in the ADJUSTED-VALUE slot of the comparable.

TABLE 1

EXAMPLES OF THE ADJUSTMENT DETERMINATION FOR FEATURES HAVING CONCEPTS AS THEIR VALUES
(IN THIS CASE, THE FEATURE IS GARAGE)

Feature Value in Subject	Associated Adjustment Figure	-	Feature Value in Comparable	Associated Adjustment Figure	=	Adjustment to Be Applied to Comparable
1-CAR	2000		NONE	0		2000
NONE	0		1-CAR	2000		-2000
2-CAR	4000		1-CAR	2000		2000
CARPORT	500		1-CAR	2000		-1500

4.4 Determination of the Appraised Value

Once the adjusted values of the ten comparables are calculated, the three best comparables are selected to combine their adjusted values in a weighted average for the calculation of the final appraised value of the subject property. Before explaining the weighted average procedure, the guidelines followed to choose those three best comparables out of the ten are presented in this section.

4.4.1 Missing Adjustments

It is possible that a comparable be identified during the calculation of the adjusted values as having one or more missing adjustments. A comparable might need an adjustment because of a differing feature, but it could be possible that the information to adjust it be unavailable. The user might have forgotten to enter the critics for a specific feature or the subject property might have an unusual feature value. Anyway, whatever the reason is for the missing adjustment, the comparable is considered invalid and is marked as "unadjusted." These comparables are not considered any further during the process.

4.4.2 Adjustment Limits

There are two total adjustment figures that are calculated for each comparable; they are the total net adjustment and the total gross adjustment, which are stored in the slots NET-ADJUSTMENT and GROSS-ADJUSTMENT, respectively, of the corresponding comparable. As discussed before, each comparable may have both upward and downward adjustments, that is, positive and negative adjustments. While the net adjustment is calculated by adding all these adjustments and taking into account their signs, the gross adjustment is obtained by adding the absolute value of each adjustment.

In the appraising domain, there are limits on the magnitude of these two figures. The limits are expressed as percentages of sale price. In CBA, they are stored in the global variables *GROSS-ADJUSTMENT-LIMIT* and *NET-ADJUSTMENT-LIMIT*. These limits help in the identification of those comparables that might be "overadjusted." One has to keep in mind that the larger the adjustments, the less precise the adjusted value may be. Even when the ten comparables are the most similar cases, any of them may have a differing feature value that is so far off the value of the subject that it may require a large adjustment that goes over the acceptable limit making it an invalid comparable.

4.4.3 Comfort Factor Calculation

To determine if the adjustments are within the appropriate limits, the percentages of net adjustment to sale price and gross adjustment to sale price are calculated for each comparable. If the percentages are over the limit percentages indicated in the global variables, the comparables are considered invalid and marked as "overadjusted." They are not considered any further after this point.

After the "unadjusted" and "overadjusted" comparables are eliminated from the list of ten candidates, the comfort factor for each of the remaining comparables is obtained. The comfort factor indicates how comfortable the program is with the adjusted value of the comparable property. It reflects the number and magnitude of adjustments made to the comparable. A comfort factor of 100% thus means that no adjustment was made to the sale price of the comparable, making it a precise reflection of the market. The higher the number of adjustments made to a comparable and the larger their magnitude, the lower is the comfort factor. It is important to note that the comfort factor is a measure to discriminate among the comparables that are left at this point. A low comfort factor does not eliminate the comparable because in terms of adjustments all of the remaining comparables are within the acceptable range as explained before.

The comfort factor is able to reflect all what is described above because it is derived directly from the percentage of gross adjustment to sale price for each comparable. It is inversely proportional to that percentage, reflecting how far the percentage is from the *GROSS-ADJUSTMENT-LIMIT*. The farther the percentage is from the limit, the less adjustments were made and the higher the comfort factor.

4.4.4 Market Consistency Heuristic

Now we have the remaining comparables with their respective comfort factors. If, at this point, we have three or less valid comparables left, there is no selection to make among them and the appraised value of the subject can be calculated. If the valid comparables are two or three, the weighted average explained in the next section is used to obtain the appraised value. If there is only one valid comparable, its adjusted value automatically becomes the appraised value. If no valid comparables are available, this means that there is no data to support an appraisal. This latter situation could happen with a property that has atypical feature values or if properties having the subject property characteristics are missing from the case library. It should be clear that the less valid comparables available, the less precise the appraisal could be, especially when there are less than four available comparables because, in that case, the market consistency heuristic, to be described below, cannot be applied.

As expressed in section 3.5, some type of technique is needed to make sure that the comparables have sale prices and adjusted values within the typical range in the market of properties of a similar type. For this purpose, a heuristic was also implemented to help in the selection of the three best comparables to be used in the calculation of the final appraised value. It is obvious that the heuristic is only applied when four or more valid comparables are available. When this is the case, first, an average is calculated of the adjusted values of the all the remaining valid comparables. Once this is done, the comparables are ranked by their absolute closeness to this average. From the three closer comparables to this average, the one with the best comfort factor is chosen. The process continues with the ranking of the comparables by their closeness to the adjusted value of this chosen comparable. Finally, the two closer comparables to the comparable chosen first are also selected to get the three needed.

In this way, three adjusted values that are as close as they can be are obtained, providing a stronger support to the appraisal. They are also close to the typical or average market prices and have relatively good comfort factors. As shown in the next chapter, this heuristic proved to give better results than other methods to select the best three comparables.

4.4.5 Appraised Value as a Weighted Average

If there are two or three valid comparables, or if there are four or more valid comparables and the selection of the three best is done, then the appraised value of the subject property can be calculated by using a weighted average. Each of the three adjusted values (or two, if that is what is available) is multiplied by its comfort factor, and the sum of these products is divided by the sum of the comfort factors.

The comfort factor thus serves as a weight to give a higher importance to those comparables that had less adjustments. The goal of obtaining a value for the subject property is finally accomplished.

4.5 User Interface

The user interface is important to obtain the necessary information from the expert in the proper format. However, the facilities provided by CBA are still very basic since this was not a priority in our endeavor. The program has been implemented as an activity in the Symbolics machine. Each one of the modules is called from the menu that is continuously shown on the screen. The menu includes the following options:

- CURRENT-PROPERTY,
- EDIT,
- CASE-RETRIEVAL,
- CRITIC-BASED ADJUSTMENT DETERMINATION,
- AUTOMATED ADJUSTMENT GENERATION.

The CURRENT-PROPERTY option allows the user to enter the values for the features of the property to be appraised. For each one of the features, this is the type of value accepted:

- LIVING-AREA: a string
- BEDROOMS: a number
- BATHROOMS: a number
- STYLE: choice from menu of concepts
- YEAR: number
- LOCATION: choice from menu of concepts

* SALE-DATE: a list of three numbers

* COOLING: choice from menu of concepts

* HEATING: choice from menu of concepts

* GARAGE: choice from menu of concepts

* SITE: a list of two numbers

* POOL: choice from menu of concepts

For those features that accept their values from a menu of concepts, the alternatives presented in the menu are taken from the VALID-VALUES slot of the corresponding MATCHING-FACTORS instance. For numeric features, the user is asked to enter numbers within a range or ranges, which the program obtains from the VALID-RANGE slot of the corresponding feature. Examples of the format of the values that these features hold were seen in figure 4.

The EDIT option allows to change (temporarily) the values of the following things:

* the *MATCH-FACTOR-THRESHOLD*,

* the valid values or valid range of any feature,

* the similarity links of any feature,

* the adaptation critics of any feature,

* the weight assigned to a specific feature,

* the *GROSS-ADJUSTMENT-LIMIT*, and

* the *NET-ADJUSTMENT-LIMIT*.

Every time the program is initialized, however, the default values are loaded again. With the EDIT option, a new case base can also be loaded.

The CASE-RETRIEVAL option finds in memory the best matches to the property currently being analyzed. After CASE-RETRIEVAL is used, either CRITIC-BASED ADJUSTMENT DETERMINATION or AUTOMATED ADJUSTMENT GENERATION may be used to come up with an appraisal figure. The CRITIC-BASED ADJUSTMENT DETERMINATION option generates the adjustments from the critics containing the adjustments suggested by the expert, while the AUTOMATED ADJUSTMENT GENERATION option tries to generate the adjustments from the data provided by the comparables themselves. This latter option, however, gives partial results and is left as a future field of investigation as presented in chapter 6.

The next chapter illustrates the application of the program described in this chapter to some specific input problems. Also, some statistics on system performance with respect to value determination are presented.

CHAPTER 5

CASE-BASED APPRAISER: TESTING

In this chapter, the CBA program is run with a specific example. After the system output is shown for this problem, some observation are made about the general performance of the program when asked to appraise a group of seventy different test properties.

5.1 Sample Run

As a first step to run CBA, parameters, such as the weights, the match factors, the adjustment amounts for critics, and the adjustment limits, should be properly set, and the property descriptions for the case base or library should be fairly complete. Even though the user have access to the parameters to change them, he should be careful when doing it because the information stored in them was given by experts in the field. If the user changes any of the parameters, the change should be made based on information from the expert or an official source in the appraising field.

The current parameters, including the match factors in the similarity links, the weights for the features, and the *MATCH-FACTOR-THRESHOLD*, were derived from conversations with property appraisal experts (Fieldson 1990a; Shearer 1990).

The critics, including the adjustment amounts and percentages, as well as the adjustment limits, were provided in writing by a group of professional property appraisers led by Curtis Fieldson (Fieldson 1990b). The valid ranges for the numeric features are shown in table 2 and the valid values for the concept-based features are shown in table 3. The contents of similarity links for numeric features and concept-based features is shown in tables 4 and 5, respectively. The three dashes (---) used in both tables mean that no links were defined for that feature. Meanwhile, the contents of critics is shown in table 6 for numeric features and in table 7 for concept-based features. Remember that the word "difference" in the numeric feature tables refers to the result of subtracting the feature value of the comparables from the feature value of the subject. The current value of the weight for each feature is shown in table 8 and the values of the other CBA parameters are presented in table 9.

The current case base was obtained from a MLS (Multiple Listing Service) manual with descriptions of single-family residential properties sold during a period of about five months in the area of Deltona, Florida. The manual was prepared by the members of the DeLand and West Volusia Board of Realtors, Inc. The case library thus contains a total of 107 property descriptions representing the variety of different types and values of the properties in the Deltona area. Appendix A shows this current CBA case base. In this

TABLE 2

VALID RANGES FOR NUMERIC FEATURES

Feature	Valid Range of Numbers	Unit
LIVING-AREA	(500 3000)	square feet
BEDROOMS	(1 4)	rooms
BATHROOMS	(1 3)	rooms
YEAR	(60 90)	years
SALE-DATE	(1 12) (1 31) (90 90)	months days years
SITE	(50 400)	square feet

TABLE 3

VALID VALUES FOR CONCEPT-BASED FEATURES

Feature	Valid Values
STYLE	ranch, raised-ranch, split-level, two-story
LOCATION	Deltona/Area-16, Deltona/Area-18, Deltona/Area-17, Deltona/Area-19
COOLING	central, not-central, none
HEATING	central, not-central, none
GARAGE	1-car, 2-car, carport, none
POOL	yes, no

TABLE 4

CONTENTS OF SIMILARITY LINKS FOR NUMERIC FEATURES:
INFORMATION FOR PARTIAL MATCHING

Feature	Range of Values for the Difference	Associated Match Factor
LIVING-AREA	(-50 50)	.95
	(50 100)	.85
	(-100 -50)	.85
	(100 150)	.50
	(-150 -100)	.50
BEDROOMS	---	---
BATHROOMS	---	---
YEAR	(-5 5)	--- 0.95
SALE-DATE	(0 6)	1.0
	(6 12)	.75
SITE	(0 0)	1.0
	(-1000 1000)	.95
	(1000 2000)	.85
	(-2000 -1000)	.85

TABLE 5

CONTENTS OF SIMILARITY LINKS FOR CONCEPT-BASED FEATURES: INFORMATION FOR PARTIAL MATCHING

Feature	Pair of Concepts	Associated Match Factor
STYLE	(ranch raised-ranch)	.8
	(split-level two-story)	.8
LOCATION	(Deltona/Area-16 Deltona/Area-18)	.5
	(Deltona/Area-17 Deltona/Area-18)	.8
COOLING	(not-central none)	.5
HEATING	(not-central none)	.5
GARAGE	(1-car 2-car)	.4
POOL	---	---

TABLE 6

CONTENTS OF CRITICS FOR NUMERIC FEATURES: INFORMATION FOR ADJUSTMENT DETERMINATION

Feature	Unit of Difference	Dollar Amount Adjustment
LIVING-AREA	50	1000
BEDROOMS	1	2000
BATHROOMS	1	1500
YEAR	1	.005 of sale price
SALE-DATE	6	2000
SITE	2000	1500

TABLE 7

CONTENTS OF CRITICS FOR CONCEPT-BASED FEATURES:
INFORMATION FOR ADJUSTMENT DETERMINATION

Feature	Base Concept	Second Concept	Dollar Amount Adjustment
STYLE	ranch	raised-ranch	0
		split-level	0
		two-story	0
LOCATION	Deltona/Area-17	Deltona/Area-18	1000
		Deltona/Area-16	2500
		Deltona/Area-19	5000
COOLING	none	not-central	0
		central	2000
HEATING	none	not-central	0
		central	1000
GARAGE	none	1-car	2000
		2-car	4000
		carport	500
POOL	no	yes	6000

TABLE 8

ASSIGNED WEIGTHS

Feature	Weight
LIVING-AREA	3
BEDROOMS	2
BATHROOMS	2
STYLE	2
YEAR	2
LOCATION	3
SALE-DATE	3
COOLING	1
HEATING	1
GARAGE	1
SITE	1
POOL	1

TABLE 9

OTHER CBA PARAMETERS

Parameter	Current Value
MATCH-FACTOR-THRESHOLD	.8
GROSS-ADJUSTMENT-LIMIT	.25
NET-ADJUSTMENT-LIMIT	.15

appendix, the property descriptions appear as instances of flavor CASE.

The current, subject, or test property for this sample run was take from the same MLS manual the case base was obtained. However, it is a house that have not been sold yet, but was appraised for the price of $43,000. The description of this property is shown in table 10.

When this information of the subject property is entered into the program, the case retrieval phase starts. The aggregate match score for each property in the case library is calculated as described in section 4.2.1 and the ten cases with the highest aggregate match scores are chosen as shown in table 11. This is the output of the case retrieval phase. All the comparable features and their respective values are included as well as the match factor (in square brackets) for each feature in each comparable to indicate how close the value in the comparable is of the subject value.

A closer look to the match factor assignment and the aggregate match score calculation can be take by observing how these were performed for COMPARABLE-7. The justification for the match factor (mf) assigned to each feature is indicated below:

* LIVING-AREA: mf = 0 because the difference between subject and comparable is (807 - 1120) or -313 square feet, and this number does not fall within any of the

TABLE 10

DESCRIPTION OF THE SUBJECT PROPERTY

Feature	Value
ADDRESS	949 W. Embassy Dr., Deltona
LIVING-AREA	807
BEDROOMS	2
BATHROOMS	1
STYLE	ranch
YEAR	75
LOCATION	Deltona/Area-19
SALE-DATE	today's date (11 20 90)
COOLING	central
HEATING	central
GARAGE	carport
SITE	80 x 125 or (80 125)
POOL	no

TABLE 11

OUTPUT OF THE CASE RETRIEVAL PHASE:
DESCRIPTIONS OF THE TEN CASES
CHOSEN AS COMPARABLES

Property	Address	Match Score	Sale Price	Living Area
Subject	949 Embassy Dr.			807
COMPARABLE-1	871 Henderson	19.8	32500	836 [.95]
COMPARABLE-2	1076 Deltona Blv.	18.9	46000	1400 [0]
COMPARABLE-3	1680 Nesbitt St.	18.4	52700	874 [.85]
COMPARABLE-4	647 Merrimac	17.8	53800	836 [.95]
COMPARABLE-5	1620 Brady Dr.	17.55	42000	874 [.85]
COMPARABLE-6	795 Chippendale St.	16.85	36000	840 [.95]
COMPARABLE-7	1039 Pioneer Dr.	16.75	45500	1120 [0]
COMPARABLE-8	8358 Blytheville Ave.	16.0	46000	1000 [0]
COMPARABLE-9	1062 Providence Blvd.	16.0	40000	1113 [0]
COMPARABLE-10	1073 Abagail Dr.	15.9	47900	1000 [0]

TABLE 11 -- CONTINUED

Property	Bedrooms	Bathrooms	Style	Year	Location
Subject	2	1	ranch	75	Deltona/Area-19
COMP-1	2 [1.0]	1 [1.0]	ranch [1.0]	66 [0]	Deltona/Area-19 [1.0]
COMP-2	2 [1.0]	1 [1.0]	ranch [1.0]	73 [.95]	Deltona/Area-19 [1.0]
COMP-3	2 [1.0]	1 [1.0]	ranch [1.0]	70 [.95]	Deltona/Area-18 [0]
COMP-4	2 [1.0]	1 [1.0]	ranch [1.0]	63 [0]	Deltona/Area-19 [1.0]
COMP-5	2 [1.0]	1 [1.0]	ranch [1.0]	67 [0]	Deltona/Area-19 [1.0]
COMP-6	3 [0]	1 [1.0]	ranch [1.0]	64 [0]	Deltona/Area-19 [1.0]
COMP-7	3 [0]	1 [1.0]	ranch [1.0]	76 [.95]	Deltona/Area-19 [1.0]
COMP-8	2 [1.0]	1 [1.0]	ranch [1.0]	83 [0]	Deltona/Area-19 [1.0]
COMP-9	2 [1.0]	1 [1.0]	ranch [1.0]	86 [0]	Deltona/Area-19 [1.0]
COMP-10	2 [1.0]	2 [0]	ranch [1.0]	76 [.95]	Deltona/Area-19 [1.0]

TABLE 11 -- CONTINUED

Prop.	Sale Date	Cooling	Heating	Garage	Site	Pool
Subj.	(11 20 90)	central	central	carport	(80 125)	no
COMP-1	(7 17 90) [1.0]	central [1.0]	central [1.0]	carport [1.0]	(100 110) [.95]	no [1.0]
COMP-2	(7 13 90) [1.0]	central [1.0]	central [1.0]	carport [1.0]	(80 125) [1.0]	no [1.0]
COMP-3	(8 25 90) [1.0]	central [1.0]	central [1.0]	carport [1.0]	(100 100) [1.0]	no [1.0]
COMP-4	(8 30 90) [1.0]	central [1.0]	central [1.0]	none [0]	(75 100) [0]	no [1.0]
COMP-5	(7 18 90) [1.0]	central [1.0]	central [1.0]	1-car [0]	(75 100) [0]	no [1.0]
COMP-6	(6 29 90) [1.0]	central [1.0]	central [1.0]	carport [1.0]	(75 100) [0]	no [1.0]
COMP-7	(8 4 90) [1.0]	central [1.0]	central [1.0]	carport [1.0]	(83 135) [.85]	no [1.0]
COMP-8	(7 1 90) [1.0]	central [1.0]	central [1.0]	1-car [0]	(80 125) [1.0]	no [1.0]
COMP-9	(6 6 90) [1.0]	central [1.0]	central [1.0]	none [0]	(80 125) [1.0]	no [1.0]
COM-10	(9 6 90) [1.0]	central [1.0]	central [1.0]	1-car [0]	(80 125) [1.0]	no [1.0]

ranges provided by the similarity links in table 4 to at least be partially matched.

* BEDROOMS: mf = 0 because the difference between subject and comparable is of one room, and there are no similarity links for this feature as shown in table 4.
* BATHROOMS: mf = 1 because the comparable matches the subject exactly in this feature.
* STYLE: mf = 1 because the comparable matches the subject exactly in this feature.
* YEAR: mf = .95 because the difference between subject and comparable is (75 - 76) or one year, and this number falls within the range associated with match factor .95 in table 4.
* LOCATION: mf = 1 because the comparable matches the subject exactly in this feature.
* SALE-DATE: mf = 1 because the difference between subject and comparable is (11 - 8) or 3 months, and this number falls within the range associated with match factor 1.0 in table 4.
* COOLING: mf = 1 because the comparable matches the subject exactly in this feature.
* HEATING: mf = 1 because the comparable matches the subject exactly in this feature.
* GARAGE: mf = 1 because the comparable matches the subject exactly in this feature.

- * SITE: mf = .85 because the difference between subject and comparable is ((80 * 125) - (83 * 135)) or -1205 square feet, and this number falls within the range associated with match factor .85 in table 4.
- * POOL: mf = 1 because the comparable matches the subject exactly in this feature.

After this, the aggregate match score is calculated. The match factor of each feature is multiplied by the corresponding weight in table 8 and the products are added together to get the score. For COMPARABLE-7, this was the operation used to get the score:

$$(0 * 3) + (0 * 2) + (1 * 2) + (1 * 2) + (.95 * 2)$$
$$(1 * 3) + (1 * 3) + (1 * 1) + (1 * 1)$$
$$(1 * 1) + (.85 * 1) + (1 * 1) = 16.75.$$

This score gave this property from the case base the opportunity to become COMPARABLE-7, one of the best ten comparables from memory.

The next phase is the adjustment determination, whose output is observed in the extended table 12. All the features that have match factor over the *MATCH-FACTOR-THRESHOLD*, which, in this example, is 0.8, are marked with an "OK," indicating that they do not need any adjustment. The rest of the features received the appropriate adjustments following the critic-based procedures described in the previous chapter for both numeric and concept-based features. Immediately comes the calculation of adjusted values. The net adjustment

TABLE 12

OUTPUT OF THE ADJUSTMENT DETERMINATION PHASE: ADJUSTED VALUES OF THE TEN COMPARABLES

Property	Sale Price	Total Gross Adj.	Total Net Adj.	Adjusted Value	Comfort Factor	Living Area
Subject						807
COMP-1	32500	1500 [5%]	1500 [5%]	34000	.8	836 [OK]
COMP-2	46000	11000 [24%]	-11000 [24%]	35000	over adjusted	1400 [-11000]
COMP-3	27000	4000 [15%]	4000 [15%]	31000	.4	874 [OK]
COMP-4	38000	4300 [11%]	4300 [11%]	42300	.56	836 [OK]
COMP-5	42000	4700 [11%]	1700 [4%]	43700	.56	874 [OK]
COMP-6	36000	5500 [15%]	1500 [4%]	37500	.4	840 [OK]
COMP-7	45500	8000 [18%]	-8000 [18%]	37500	over adjusted	1120 [-6000]
COMP-8	46000	6300 [14%]	-6300 [14%]	39700	.44	1000 [-3000]
COMP-9	40000	8700 [22%]	-7700 [19%]	32300	over adjusted	1113 [-6000]
COMP-10	47900	6000 [13%]	-6000 [13%]	41900	.48	1000 [-3000]

TABLE 12 -- CONTINUED

Property	Bedrooms	Bathrooms	Style	Year	Location
Subject	2	1	ranch	75	Deltona/Area-19
COMP-1	2 [OK]	1 [OK]	ranch [OK]	66 [1500]	Deltona/Area-19 [OK]
COMP-2	2 [OK]	1 [OK]	ranch [OK]	73 [OK]	Deltona/Area-19 [OK]
COMP-3	2 [OK]	1 [OK]	ranch [OK]	70 [OK]	Deltona/Area-18 [4000]
COMP-4	2 [OK]	1 [OK]	ranch [OK]	63 [2300]	Deltona/Area-19 [OK]
COMP-5	2 [OK]	1 [OK]	ranch [OK]	67 [1700]	Deltona/Area-19 [OK]
COMP-6	3 [-2000]	1 [OK]	ranch [OK]	64 [2000]	Deltona/Area-19 [OK]
COMP-7	3 [-2000]	1 [OK]	ranch [OK]	76 [OK]	Deltona/Area-19 [OK]
COMP-8	2 [OK]	1 [OK]	ranch [OK]	83 [-1800]	Deltona/Area-19 [OK]
COMP-9	2 [OK]	1 [OK]	ranch [OK]	86 [-2200]	Deltona/Area-19 [OK]
COMP-10	2 [OK]	2 [-1500]	ranch [OK]	76 [OK]	Deltona/Area-19 [OK]

TABLE 12 -- CONTINUED

Prop.	Sale Date	Cooling	Heating	Garage	Site	Pool
Subj.	(11 20 90)	central	central	carport	(80 125)	no
COMP-1	(7 17 90) [OK]	central [OK]	central [OK]	carport [OK]	(100 110) [OK]	no [OK]
COMP-2	(7 13 90) [OK]	central [OK]	central [OK]	carport [OK]	(80 125) [OK]	no [OK]
COMP-3	(8 25 90) [OK]	central [OK]	central [OK]	carport [OK]	(100 100) [OK]	no [OK]
COMP-4	(8 30 90) [OK]	central [OK]	central [OK]	none [500]	(75 100) [1500]	no [OK]
COMP-5	(7 18 90) [OK]	central [OK]	central [OK]	1-car [-1500]	(75 100) [1500]	no [OK]
COMP-6	(6 29 90) [OK]	central [OK]	central [OK]	carport [OK]	(75 100) [1500]	no [OK]
COMP-7	(8 4 90) [OK]	central [OK]	central [OK]	carport [OK]	(83 135) [.85]	no [OK]
COMP-8	(7 1 90) [OK]	central [OK]	central [OK]	1-car [-1500]	(80 125) [OK]	no [OK]
COMP-9	(6 6 90) [OK]	central [OK]	central [OK]	none [500]	(80 125) [OK]	no [OK]
COMP-10	(9 6 90) [OK]	central [OK]	central [OK]	1-car [-1500]	(80 125) [OK]	no [OK]

and gross adjustment for each comparable is determined and their percentages with respect to sale price are obtained. These percentage are used to check if any comparable is "overadjusted" and to calculate the comfort factor for the rest. In this example, COMPARABLE-9, COMPARABLE-7, and COMPARABLE-2 are overadjusted because they went over the 15% limit for the percentage of net adjustment with respect to sale price. The rest obtained their comfort factors as seen in table 12.

Now, instead of choosing comparable 1, 4, and 5 for the final appraised value calculation because they have the highest comfort factors, the market consistency heuristic is applied. The average adjusted value is approximately $38,586. The closest three to this average are comparables 6, 8, and 10. Of these three, the best comfort factor corresponds to COMPARABLE-10. Consequently, COMPARABLE-10 and comparable 4 and 5, whose adjusted values are the closest to the one of COMPARABLE-10, are chosen to calculated the appraised value of the subject property. It can be observed that the comparable with the highest comfort was not chosen. Even though it has the value less affected by adjustments, it is not close enough to the market typical or average price in order to be chosen for the final calculation.

The weighted average for the appraised value uses the comfort factors and the adjusted values of the chosen comparables 4, 5, and 10 as follows:

((42300 * 0.56) + (43700 * 0.56) + (41900 * 0.48))

/ (0.56 + 0.56 + 0.48) = 42670.

Thus, the appraised value for the subject property, always rounded to the nearest hundred, is therefore $42,700.

5.2 Test Results and Influencing Issues

When comparing the CBA appraised value for the subject property of $42,700 and the list price of $43,500 given by the MLS manual, a difference of $800 is noted, which is about 2% of the list price. To have a more general idea of how the program can perform, a group of seventy test cases was obtained. The test cases were also take from the same MLS manual. In addition to sold properties, MLS manuals list properties that are available for sale, and they are appraised to get a list price to be included in their descriptions. The test properties appear in Appendix B as LISP functions that bind the different feature values of the instance *MY-PROPERTY* of flavor CURRENT-PROPERTY. The list prices of test cases, established by either realtors or appraisers (Diaz 1990), can be used to compare the appraised values given by CBA against them. Some interesting observations can be made by doing such comparisons as shown in table 13.

This table shows the difference between list price and the CBA appraised value for each one of the test properties. It can be seen that two properties out of the seventy did not obtain an appraised value because all the comparables retrieved to get a value were overadjusted. The values of

TABLE 13

COMPARISON OF CBA APPRAISED VALUES AND LIST PRICES
OF THE TEST PROPERTIES

Test No.	List Price	CBA Appraised Value	Number of Valid Comparables	Difference	Percent. of List Price
70	159900	NONE	0	NONE	NONE
69	127500	94600	9	2900	0.26
68	122900	141300	2	-18400	0.15
67	119900	104600	1	15300	0.13
66	112000	100000	6	12000	0.11
65	109500	84900	8	24600	0.22
64	104900	98800	3	6100	0.06
63	99900	79300	6	20600	0.21
62	98500	NONE	0	NONE	NONE
61	95000	99700	9	-4700	0.05
60	94900	85000	9	9900	0.1
59	94500	82000	9	12500	0.13
58	92000	76400	10	15600	0.17
57	89000	69100	7	19900	0.22
56	86500	74300	10	12200	0.14
55	85000	78700	6	6300	0.07
54	82500	76900	8	5600	0.07
53	79900	70100	10	9800	0.12
52	79900	75300	9	4600	0.06
51	79900	90600	5	-10700	0.13
50	79200	71500	10	7700	0.1
49	77500	83300	7	-5800	0.07
48	75000	69700	9	5300	0.07
47	74900	76200	10	-1300	0.02
46	73900	74600	10	-700	0.01
45	73000	81600	8	-8600	0.12
44	71900	67800	10	4100	0.06
43	69900	65400	9	4500	0.06
42	69900	71400	9	-1500	0.02
41	69500	73200	10	-3700	0.05
40	68500	46200	2	22300	0.33
39	67900	58700	10	9200	0.14
38	67000	69400	6	-2400	0.04
37	66900	62700	10	4200	0.06
36	64900	56100	9	8800	0.14
35	64900	68500	10	-3600	0.06
34	64000	67700	10	-3700	0.06
33	63500	58900	10	4600	0.07
32	63000	64700	10	-1700	0.03

TABLE 13 -- CONTINUED

Test No.	List Price	CBA Appraised Value	Number of Valid Comparables	Difference	Percent. of List Price
31	61900	53500	10	8400	0.14
30	60750	57000	9	3750	0.06
29	59900	58500	10	1400	0.02
28	59900	62700	8	-2800	0.05
27	59500	57000	10	2500	0.04
26	58900	61100	9	-2200	0.04
25	57900	59400	10	-1500	0.03
24	57700	60300	3	-2600	0.05
23	56900	58800	10	-1900	0.03
22	56000	56500	10	-500	0.01
21	55000	66400	3	-11400	0.21
20	54900	54000	10	900	0.02
19	54500	43800	2	10700	0.2
18	53900	54800	10	-900	0.02
17	53900	51500	10	2400	0.04
16	52900	56400	7	-3500	0.07
15	49900	56600	10	-6700	0.13
14	49900	52800	10	-2900	0.06
13	49500	44700	5	4800	0.1
12	49000	47600	8	1400	0.03
11	47500	44100	5	3400	0.07
10	45900	37100	6	8800	0.19
9	45000	44400	1	600	0.01
8	44900	42100	8	2800	0.06
7	43900	39700	9	4200	0.1
6	43500	42700	7	800	0.02
5	42900	43400	6	-500	0.01
4	39900	41200	9	-1300	0.03
3	39500	61300	2	-21800	0.55
2	34900	36300	7	-1400	0.04
1	30500	32000	3	-1500	0.05

these properties are in the 100's, a range in which few properties are actually sold in Deltona. An analysis of the results for the rest of the test properties is presented in table 14 under the column headed "68 Tests." In 37% of the tests that obtained appraised values, CBA gave values with a difference percentage of 5% or less, and in 68% of the tests, it gave values with a difference percentage with respect to list price of 10% or less. 90% of the values given by CBA have 20% or less as their difference percentage. The average difference is of $6,800 and the average difference percentage is of 9%.

TABLE 14

ANALYSIS OF TEST RESULTS

Indicator	68 Tests	66 Tests
Properties with percentage of difference of .05 or less	.37	.38
Properties with percentage of difference of .10 or less	.68	.70
Properties with percentage of difference of .15 or less	.85	.88
Properties with percentage of difference of .20 or less	.90	.92
Average difference	6800	6400
Average percentage of difference	.09	.08

The above results are an indication that CBA is providing fairly good results and, at the same time, these results give some direction to the future efforts of improving the CBA algorithm. The following sections discuss some of the issues that influence these results.

5.2.1 Number of Represented Features in a Property

Given that the expert that provided the information for CBA is different from all the people that appraised the list prices of the test properties, the testing results can be considered acceptable because the CBA appraised values are fairly consistent and close to the list prices. These results also reflect that the features chosen to represent each property are indeed critical in the determination of value. However, on the other hand, the small set of features used to represent properties is precisely the cause for most of the big differences between list prices and CBA appraised values, especially those over the 20% of difference. This can be seen especially in two test properties, TEST-40 and TEST-3.

TEST-40 and TEST-3 have the worst CBA appraised values. Their difference percentages are 33% and 55%, respectively. When a closer look is given to the description of these two properties in the MLS manual, it is realized that some other features were needed to represent properties in general in order to get an acceptable appraised value from CBA. For example, TEST-40 is a lakefront home and none of the comparables that CBA chose is; this is because there is currently no representation of this characteristic in properties. Consequently, TEST-40 obtained an appraised value under the one it really has. On the other side, TEST-3 is a property with a large lot and a large house. However, the house is a bungalow style; it is a vacationing cottage that was not built to last as much as a regular family house. Since there is no feature representing the construction style and material of houses in the property descriptions used in CBA, the program compares this property to others with large lots and large houses of the regular type, arriving at an appraised value over the real value of the property. The elimination of these two properties, which for the just discussed reasons are way off the typical performance of CBA, gives new improved figures for the analysis in table 14 under the column titled "66 Tests."

If more features are included for each property, a better differentiation between properties can be made. Therefore, the values appraised should be more precise.

5.2.2 Lack of Data or Unusual Feature Values

There is a possibility of having incorrect information for some feature values in the properties in the case base or the test cases may be another reason for the larger differences in values (list price and appraised price). This could be solved by obtaining more reliable sources of data. Even though the MLS manual has done well as a data source, other sources like the property tax records, which should have more correct information, can be used.

Properties that have unusual combination of the features for the list price that was assigned to them have larger difference percentages. The features presented in the property point to an appraised value different from the list price. If there are not enough similar properties or properties similar enough the case base, a small number of valid comparables is obtained or an appraisal could not even be made, as in the case of TEST-62 and TEST-70. It can be observed from the column of valid comparables available in table 11 that most appraised values of the tests in which there were three or less valid comparables have a relatively large difference percentage. This unavailability of valid comparables does not allows the check for market consistency of the prices of the comparables themselves giving more margin for error. So, the less number of valid comparables that are available, the less precise the appraised value is; the less valid comparables, the larger the margin of difference.

5.2.3 Market Consistency Check

If the market consistency heuristic can be applied, it helps to reduce these differences between list price and appraised value. In table 15, the statistics for two other methods used to select the three comparables that are finally used to compute appraised values are presented with the one previously shown for the market consistency heuristic in table

14. The 3-best-comparables method chooses the three comparables that had three highest aggregate match scores in the case retrieval phase. The third method chooses the comparable with the best comfort factor and then the two comparables with the closest adjusted values to the value of the one chosen first. The three methods were tried with the seventy test cases and it is clear from table 15 that the market consistency heuristic is superior.

TABLE 15

PERFORMANCE OF DIFFERENT METHODS TO SELECT THE BEST THREE COMPARABLES

Indicator	Market Consistency Heuristic	Three First Comparables	Best CF and 2 Closest
Properties with percentage of difference of .05 or less	.37	.34	.37
Properties with percentage of difference of .10 or less	.68	.63	.54
Properties with percentage of difference of .15 or less	.85	.81	.71
Properties with percentage of difference of .20 or less	.90	.93	.88
Average difference	6800	7200	7800
Average percentage of difference	.09	.10	.11

5.2.4 Subjective Appraisal Problem

A point that should be clear is that the same property may be valued by three different appraisers, and each open of them may come up with a different figure. All these figures are considered acceptable as long as they are supported by market data. Thus, it can be said that this domain of property appraisal involves a subjective problem. In domains where right and wrong are basically impossible to determine, it is important to be consistent. Since no fixed values can be obtained as the absolute correct answers to compare our results against, at least some consistency should be reached to give answers within an acceptable range, in which the answers from experts can be found. In this respect, we believe that CBA does a good job given the limitations it has in its condition of prototype in its early stage: small case base, small number of represented figures, and lack of better sources of cases (even though MLS has done fairly well). Therefore, the tests are not a measure of preciseness or correctness, but of consistency in the use of market data. Answers are not classified as correct or incorrect, but as fair or unacceptable if they are within a range.

CHAPTER 6

AVENUES FOR FUTURE RESEARCH

In this chapter, some ideas are presented to pursue the idea of automatically generating the adjustments needed during the appraising process from the market data itself. Improvements and extensions that could possibly be made to CBA are also discussed.

6.1 Automated Adjustment Generation (AAG)

As mentioned previously, AAG could be considered as an alternative to produce the dollar amount adjustments. Instead of asking the expert to provide the figures, they could be obtained from the market data. If this is possible, the dynamic nature of the market and the knowledge in property appraisal could be emulated even better. AAG could even be considered as a way to further implementing machine learning. In the next three subsections, the theoretical justification, the algorithm and the possible directions for development of AAG are discussed.

6.1.1 Theoretical Base

As explained in section 2.1, property appraisal, especially its market data approach, holds that any adjustments made to the value of a comparable need to be supported by market data. The behavior of buyers and sellers in the market determine the value of a property; and each feature of a property has an individual contribution to value that makes possible the isolation of its effect by observing properties that have been sold and differ in only one feature. Now, to determine the actual dollar amount of adjustment for specific feature, one of the most popular techniques used by appraisers is comprised by the market grids and the paired data set analysis. Let us observe the following simplified example.

In table 16, a small market data grid is shown. It includes the subject property, three comparable sales, and an extremely small set of elements of comparison just to illustrate the principle involved in paired data set analysis. First, the appraiser notes the significant differences between each comparable property and the subject property. If a comparable is identical to the subject in a given respect, "same" is indicated on the grid. Then, the appraiser tries to find a pair of comparables that differ in only one respect. Sale 1 and 2 differ in two features, and the other two possible pairs of comparables differ in only one feature. Sale 2 and 3 are chosen to be paired because they differ only in the condition of the property.

The next step is to determine whether the presence of the feature in question is an advantage or a disadvantage, and how much value the market ascribes to it by using paired data set analysis, that is by subtracting the price of one sale from the other. In this example, the good condition of sale 2 is an advantage valued at $6,000. The adjustment is made only to the comparable that differs from the subject. An upward adjustment is made if the comparable is inferior to the subject, and a downward adjustment is done otherwise. Thus, sale 1 and 3 receive their appropriate upward adjustment.

TABLE 16

USING MARKET DATA GRIDS FOR PAIRED DATA SET ANALYSIS: PART I

Feature or Element of Comparison	Subject	Sale 1	Sale 2	Sale 3
Price	?	$101,000	$109,800	$103,800
Site Shape	Irreg.	Irreg. (same)	Irreg. (same)	Irreg. (same)
Condition	Good	Poor +$6,000	Good (same)	Poor +$6,000
Garage	1-car	1-car (same)	2-car	2-car

Now, in table 17, we observe that the prices of the comparables have all been adjusted and now those features that received adjustments are going to be treated as if they were identical to the corresponding feature in the subject. Therefore, sales 1 and 2, which originally had 2 differences, now are considered to have 1 difference. So, as more adjustments are made, the differences among comparables decrease, providing the possibility of finding more pairs of comparables with one difference. By pairing sales 1 and 2 or sales 1 and 3, the downward adjustment of $2,800 is obtained.

TABLE 17

USING MARKET DATA GRIDS FOR PAIRED DATA SET ANALYSIS: PART II

Feature or Element of Comparison	Subject	Sale 1	Sale 2	Sale 3
Price	?	$101,000	$109,800	$103,800
Adjusted Value		$107,000	$109,800	$109,800
Site Shape	Irreg.	Irreg. (same)	Irreg. (same)	Irreg. (same)
Condition	Good	Poor (adjusted)	Good (same)	Poor (adjusted)
Garage	1-car	1-car (same)	2-car -$2,800	2-car -$2,800

This procedure may be extended to larger market data grids with more comparables and more elements of comparison. However, this extension becomes more complex, especially when, for example, the features or elements of comparison have more than two possible values. In the example of tables 2 and 3, each feature could have only one of two values (regular or irregular, poor or good, 1-car or 2-car).

The following section presents the general algorithm followed by CBA to implement the generation of adjustments from pairs of properties that have a single difference following the principles of the paired data set analysis.

6.1.2 Algorithm

CBA follows the algorithm presented in this section to try to implement AAG. Currently, ten comparables, retrieved from the case base, are used in the process of AAG.

1. Each comparable is taken and each one of its features that matches exactly to the subject, or partially within the allowed range, is marked as being "already adjusted."
2. To identify the pairs of comparables that have a single difference, a list of lists is made in which each sublist is composed of a pair of comparables with their corresponding number of differences. To determine the differences between two comparables, the same guidelines followed to compare each comparable to the subject during case retrieval are also used to compare the pair of comparables. They are compared feature by feature. Even when both features are marked as being "already adjusted," they must be compared because even when they match to the subject, they might not match between them. This is especially seen when the features are numeric and use ranges in their matching process.

3. For each one of the pairs that differ in only one feature and for the feature in which they differ, check if one and only one of the comparables has the same value as the subject. If so, there is a comparable with a feature similar to the subject and another with a feature differing from subject. Given this, the correct value of the adjustment, both in sign and magnitude, is found by subtracting the value of the property with the differing feature from the value of the property with the similar feature. The adjustment is stored in a slot associated with the property with the differing value.

   If both comparables, for the given feature, have the same value as the subject or both have values different from the one in the subject, the adjustment is not calculated. In the first case, both comparables are "already adjusted" in that feature; they do not need the adjustment. In the latter, there is no way to calculate the adjustment that each comparable needs; the adjustment may be different for each comparable.

4. If an adjustment amount is calculated and stored, all the other comparables that have the same differing feature receive the adjustment and store it.

5. Every time an adjustment is received, it is stored by the comparable and associated with the appropriate

feature. When more than one adjustment figure is received for a particular feature, all of them are stored in a list to average them at a later step.

6. After all pairs with a single difference have been checked to produce adjustments, all comparables are checked to see if at least three of them are fully adjusted, that is, all their differences have been reconciled by receiving an adjustment. If so, the program stops and totals the adjustments for each of the three top comparables that are fully adjusted. When totaling the adjustments in a comparable, care must be exercised when more than one adjustment figure is found for a single feature. These figures should be averaged to a single figure and then added to the rest of the adjustments in the comparable. Each total adjustment figure is applied to sale the price of the corresponding comparable. Finally, the adjusted prices of the three comparables are averaged to get the final appraised value figure for the subject property.

7. If the three fully adjusted comparables have not been obtained, adjustments are still totaled and the adjusted values of properties updated, but the flow continues by going back to step 2. Before that, all the features that received adjustments are marked as "already adjusted." The loop continues in steps 2-7

if the three fully adjusted comparables are not obtained unless in one of the cycles no adjustments are generated. This causes the program to stop and display a message indicating that the data available does not provide sufficient support to make an appraisal.

AAG uses many of the comparison routines developed for case retrieval in CBA and, as the routine that produces the critic-based adjustments, it uses the match factors provided by the case retrieval stage to determine initially which features in the comparables need adjustment.

Access to the AAG procedure is gained through the AUTOMATED ADJUSTMENT GENERATION option of the main menu, but, at this point, it only gives partial results. To try to improve its effectiveness, some more research is needed. In the next section, some observations are made about this respect.

6.1.3 Possible Directions for Development

AAG seems to need very large databases to perform better. Still, the challenge of AAG is to produce adjustments avoiding the use of statistical methods, but using heuristic knowledge from property appraisal experts. However, before reaching any conclusion about the effectiveness of AAG, some more investigation should be done on several areas.

A closer look to the mental process of the appraiser during adjustment generation is needed. From the observations made, some more heuristics to increase the amount of adjusting information obtained from the case base itself might be produced. Up to this point, the heuristics developed limit the number of adjustments produced. The current implementation of AAG works by extracting information from pairs of comparables with a single difference. Right now, a group of ten comparables is being used to perform the adjustment extraction. Since these comparables are the most similar to the subject among all the cases in memory, most of them differ from the subject in only one or a few features. However, usually the right combination of feature values is not present to apply effectively the paired data set analysis and identify pairs with a single difference. Sometimes, a group of 20 or more comparables is needed to start getting fully adjusted comparables; but even in this situation, the fully adjusted comparables obtained usually are not the top matches, that is, they are not among the best three or even the best ten comparables. Also, the larger the number of features included in a property and the broader the range of values for each feature, the larger the case base should be since it needs to provide a more vast variety of combinations of feature values.

To address these problems and still try to work with a modestly-sized case base, observations should be made to the way in which the appraiser handles data that do not include many pairs of comparables with a single difference. In the real world, it is difficult to find such pairs and the appraisers evidently have some method of assigning the relative contribution of each differing feature to the money difference of two properties. Thus, a weight system might be an alternative to implement the credit assignment scheme that appraisers have. Care should be exercised if this weight scheme or any other technique is used to extract the adjustment information from pairs of comparables with two or more differences because the validity of the approximations may degrade to the eyes of the appraiser and the client.

why

If a scheme using weights is utilized for credit assignment, then for each feature, the level of advantage of each of its values must be identified. This should help in identifying the combinations of differing features that may be used to extract adjustments. Suppose that two properties, COMPARABLE-3 and COMPARABLE-4, differ in only two features. COMPARABLE-3 has a two-car garage and a pool, while COMPARABLE-4 has a 1-car garage and no pool. If a two-car garage and the availability of a pool are identified as the most advantageous values in their respective features, then our pair of comparables could be used to get the adjustments by using the scheme of weights. COMPARABLE-3 has the most advantageous values for the differing features and the direction (upward or downward) and magnitude of the adjustments can be determined. However, if COMPARABLE-3 has a two-car garage and no pool and COMPARABLE-4 has a one-car garage and a pool, there is no way to determine, first, which one has the most advantageous values and, then, which direction and what magnitude the adjustment for each feature has.

Another problem in AAG to address is what to do when a negative adjustment is obtained for a value of a feature that is considered an advantage. It could be accepted if we assume that buyers at some point could react adversely to a feature in a house that traditionally has been considered an advantage. However, the final decision of accepting or discarding the adjustment should be based on what appraisers do in practice.

There is no guarantee that AAG can possibly produce adjustments efficiently, but this could only be proven after the appropriate investigation of these issues.

6.2 Possible Improvements and Extensions to CBA

The following are improvements to CBA that could be incorporated in the near future:

- For now, the case base needs to be entered manually into the proper format in a file. Each property is made an instance of flavor CASE. Some facilities, transparent to the user, to allow the loading of other files with case bases should be added.
- Better editing facilities could be developed, especially to handle the critics and similarity links obtained from the expert.

* More elements of comparison could be added to describe properties more clearly and differentiate them better. The goal should be to incorporate all the feature included in the URAR (Uniform Residential Appraisal Report) form, which is the most popular standard form used by professional appraisers to report on their findings.

* Hidden features could be incorporated into the program. Hidden features are those that are not given directly by the information in the case base, instead they are derived from the combination of input features, that is, those given by the case base. For example, in the URAR form, there is a feature called functional utility, which is not given directly by the descriptions of properties found in different places. Therefore, this feature should be made hidden and its value should be obtained by a predetermined combination of input feature values, such as the number of bedrooms, the number of bathrooms, the availability of a family room or a porch, etc. Functional utility is a measure of the attractiveness and usefulness of the property.

* For consistency, the flavor MATCHING-FACTORS could be eliminated and each slot of the flavor could be made an instance of flavor PROPERTY. *WEIGHTS* is already an

instance of PROPERTY. This change does not make the code more efficient, but it does make it clearer.

These other points are extensions to CBA that could be incorporated in the long run:

* A combination of the adjustments produced by the critics and AAG, if this one is further developed, could be made.
* A natural language processing system (NLP), as the one used with JULIA (Shinn 1988) could be used to interpret problem descriptions into the appropriate frame-based format of the case base.
* An interface could be made with some type of computerized service that provide access to databases of properties sold to obtain the information for the case base. Examples of these services is the MLS, organized by the different local boards of realtors, and the SREA Market Data Center, maintained by a national organization of real estate appraisers.
* An interface could be added to allow the user to define the number and the names of the elements of comparison he wants to use. In this way, the user may have more flexibility in the adaptation of the system to the target market. With this interface, the program becomes more generic; currently, the code is highly dependent on a fixed number of elements of comparison.

CHAPTER 7

SUMMARY AND CONCLUSION

As seen, case-based reasoning is a viable alternative to construct expert systems as opposed to purely rule-based systems. In the particular case of property appraisal, case-based reasoning seems to adapt more naturally to the actual way in which property appraisers do their work. CBR is a paradigm that may ease the task of knowledge acquisition for expert systems. A case base may be produced from the experts' own experiences, which may be obtained directly by interviewing the experts or indirectly by accessing some type of database or printed source with such events. In our case, the case base was obtained through the latter method. However, the information necessary to process such case base was obtained by means of interviews.

The most important information needed from the expert, after the case base, is the set of adaptation critics he uses to apply the solutions of previous problems to the present situation. These critics are usually implemented in the form of rules. However, they may be a lot simpler than the rules of a purely rule-based system, and also they may be smaller in number. Rule-based systems have to build a solution from scratch by using rules, while the critics in the case-based systems are intended to make small changes to solutions that were already given to other problems. The actual size of the set of rules to perform partial matching and adaptation, implemented in this work as similarity links and critics, depends on the completeness of the case base: the less comprehensive the case base is, greater is the reliance on the partial matching information and the adaptation information and greater could be their size.

Property appraisal is an example of judgement appraisal domain, in which a solution may have a broad range of valid values and, though different experts may give different solutions, they may be right if data support them. This type of domain has been successfully represented using case-based reasoning, but property appraisal differs from the rest of the domains in this category in that its solution involves a single parameter whose value is numeric. Most of the other domains in which case-based reasoning has been applied include the preparation of a plan or an explanation with multiple parameters. The fact that there is a single parameter in the solution for the property appraisal domain makes it extremely important to have the best similar cases possible from memory. This is accomplished in the work presented in this paper through the particular combination of several techniques: best-match algorithm (MBR), similarity links, weights, match scores, adjustment limits, comfort factors, critics, and market consistency heuristic.

The automated adjustment generation for the adaptation phase of the case-based reasoner presented in this work is left as an open avenue for further investigation. Its actual usefulness and effectiveness in the search of solutions could only be determined after further research on its implementation methods.

The implementation presented here is based on the premises of traditional appraising practices. Instead to recurring to statistical methods to process large amounts of data, appraisers analyze a relatively small number of properties in a market to learn about the tendencies in that market. However, there is some literature (Jaffe 1985) that points out that the trend in the appraising profession could change in the near future. The methodology of property valuation could change to the use of the microcomputer to apply statistical measures to the study of market data so that their work could meet the demand for more scientific approach. However, this could take time since the people that have been longer in the profession resist change. The validity of the use of this expert system presented here will then lay on the trend that is later followed. However, the system, especially if AAG is incorporated to it, could still be used as a training tool, so that the new appraiser can observe the process followed in traditional appraising.

APPENDIX A

CURRENT CBA CASE LIBRARY

```
(setf case-1
      (make-instance 'case
               :address "2273 Flamingo Ave., Deltona"
               :sale-price 29000
               :living-area 700
               :bedrooms 1
               :bathrooms 1
               :style 'ranch
               :year 69
               :location 'Deltona/Area-17
               :sale-date '(8 31 90)
               :cooling 'not-central
               :heating 'not-central
               :garage 'carport
               :site '(75 100)
               :pool 'no))


(setf case-2
      (make-instance 'case
               :address "972 Chippendale St., Deltona"
               :sale-price 30000
               :living-area 800
               :bedrooms 3
               :bathrooms 1
               :style 'ranch
               :year 66
               :location 'Deltona/Area-19
               :sale-date '(8 9 90)
               :cooling 'none
               :heating 'not-central
               :garage 'carport
               :site '(80 125)
               :pool 'no))


(setf case-3
      (make-instance 'case
               :address "935 Vivian Terr., Deltona"
               :sale-price 30600
               :living-area 700
               :bedrooms 1
               :bathrooms 1
               :style 'ranch
               :year 68
               :location 'Deltona/Area-18
               :sale-date '(6 27 90)
               :cooling 'not-central
               :heating 'not-central
               :garage 'none
               :site '(99 134)
               :pool 'no))
```

```
(setf case-4
      (make-instance 'case
               :address "661 Rookery Ave., Deltona"
               :sale-price 59500
               :living-area 1100
               :bedrooms 2
               :bathrooms 2
               :style 'ranch
               :year 86
               :location 'Deltona/Area-18
               :sale-date '(6 28 90)
               :cooling 'central
               :heating 'central
               :garage  '2-car
               :site '(80 125)
               :pool 'no))


(setf case-5
      (make-instance 'case
               :address "643 N. Firwood Ave., Deltona"
               :sale-price 56500
               :living-area 1131
               :bedrooms 2
               :bathrooms 2
               :style 'ranch
               :year 82
               :location 'Deltona/Area-16
               :sale-date '(8 7 90)
               :cooling 'central
               :heating 'central
               :garage '1-car
               :site '(80 125)
               :pool 'no))


(setf case-6
      (make-instance 'case
               :address "1016 Fountainhead Dr., Deltona"
               :sale-price 66000
               :living-area 1539
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 77
               :location 'Deltona/Area-19
               :sale-date '(9 8 90)
               :cooling 'central
               :heating 'central
               :garage '2-car
               :site '(125 94)
```

```
                 :pool 'no))


(setf case-7
      (make-instance 'case
                 :address "1931 E. Chapel, Deltona"
                 :sale-price 69000
                 :living-area 1472
                 :bedrooms 3
                 :bathrooms 2
                 :style 'ranch
                 :year 88
                 :location 'Deltona/Area-16
                 :sale-date '(6 29 90)
                 :cooling 'central
                 :heating 'central
                 :garage '2-car
                 :site '(80 125)
                 :pool 'no))


(setf case-8
      (make-instance 'case
                 :address "595 S. Glancy Dr., Deltona"
                 :sale-price 70000
                 :living-area 1455
                 :bedrooms 3
                 :bathrooms 2
                 :style 'ranch
                 :year 86
                 :location 'Deltona/Area-16
                 :sale-date '(9 5 90)
                 :cooling 'central
                 :heating 'central
                 :garage '2-car
                 :site '(80 125)
                 :pool 'no))


(setf case-9
      (make-instance 'case
                 :address "1119 Madura, Deltona"
                 :sale-price 72000
                 :living-area 1817
                 :bedrooms 4
                 :bathrooms 2
                 :style 'ranch
                 :year 84
                 :location 'Deltona/Area-19
                 :sale-date '(8 2 90)
                 :cooling 'central
                 :heating 'central
```

```
                :garage '2-car
                :site '(80 125)
                :pool 'no))


(setf case-10
      (make-instance 'case
                :address "1271 Section Line Trail, Deltona"
                :sale-price 73000
                :living-area 1600
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 79
                :location 'Deltona/Area-19
                :sale-date '(8 14 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 125)
                :pool 'yes))


(setf case-11
      (make-instance 'case
                :address "880 Halstead St., Deltona"
                :sale-price 31900
                :living-area 660
                :bedrooms 2
                :bathrooms 1
                :style 'ranch
                :year 64
                :location 'Deltona/Area-19
                :sale-date '(8 4 90)
                :cooling 'not-central
                :heating 'not-central
                :garage 'carport
                :site '(75 100)
                :pool 'no))


(setf case-12
      (make-instance 'case
                :address "871 Henderson, Deltona"
                :sale-price 32500
                :living-area 836
                :bedrooms 2
                :bathrooms 1
                :style 'ranch
                :year 66
                :location 'Deltona/Area-19
                :sale-date '(7 17 90)
```

```
                 :cooling 'central
                 :heating 'central
                 :garage 'carport
                 :site '(100 110)
                 :pool 'no))


(setf case-13
      (make-instance 'case
                 :address "1680 Nesbitt St., Deltona"
                 :sale-price 27000
                 :living-area 874
                 :bedrooms 2
                 :bathrooms 1
                 :style 'ranch
                 :year 70
                 :location 'Deltona/Area-18
                 :sale-date '(8 25 90)
                 :cooling 'central
                 :heating 'central
                 :garage 'carport
                 :site '(100 100)
                 :pool 'no))


(setf case-14
      (make-instance 'case
                 :address "2473 Kimberly Dr., Deltona"
                 :sale-price 35750
                 :living-area 902
                 :bedrooms 2
                 :bathrooms 1
                 :style 'ranch
                 :year 77
                 :location 'Deltona/Area-17
                 :sale-date '(8 29 90)
                 :cooling 'not-central
                 :heating 'not-central
                 :garage 'none
                 :site '(75 100)
                 :pool 'no))


(setf case-15
      (make-instance 'case
                 :address "1026 Cobblestone Dr., Deltona"
                 :sale-price 29000
                 :living-area 874
                 :bedrooms 2
                 :bathrooms 1
                 :style 'ranch
                 :year 66
```

```
               :location 'Deltona/Area-16
               :sale-date '(8 31 90)
               :cooling 'central
               :heating 'central
               :garage 'none
               :site '(80 100)
               :pool 'no))


(setf case-16
      (make-instance 'case
               :address "647 Merrimac, Deltona"
               :sale-price 38000
               :living-area 836
               :bedrooms 2
               :bathrooms 1
               :style 'ranch
               :year 63
               :location 'Deltona/Area-19
               :sale-date '(8 30 90)
               :cooling 'central
               :heating 'central
               :garage 'none
               :site '(75 100)
               :pool 'no))


(setf case-17
      (make-instance 'case
               :address "795 Chippendale St., Deltona"
               :sale-price 36000
               :living-area 840
               :bedrooms 3
               :bathrooms 1
               :style 'ranch
               :year 64
               :location 'Deltona/Area-19
               :sale-date '(6 29 90)
               :cooling 'central
               :heating 'central
               :garage 'carport
               :site '(75 100)
               :pool 'no))


(setf case-18
      (make-instance 'case
               :address "102 Amigos Rd., DeBary"
               :sale-price 37500
               :living-area 1025
               :bedrooms 3
               :bathrooms 1
```

```
                  :style 'ranch
                  :year 62
                  :location 'DeBary
                  :sale-date '(7 28 90)
                  :cooling 'not-central
                  :heating 'none
                  :garage 'none
                  :site '(150 75)
                  :pool 'no))


(setf case-19
      (make-instance 'case
                  :address "808 Merrimac St., Deltona"
                  :sale-price 38500
                  :living-area 1032
                  :bedrooms 2
                  :bathrooms 1
                  :style 'ranch
                  :year 66
                  :location 'Deltona/Area-19
                  :sale-date '(6 30 90)
                  :cooling 'central
                  :heating 'not-central
                  :garage 'carport
                  :site '(75 100)
                  :pool 'no))


(setf case-20
      (make-instance 'case
                  :address "1684 Brady Dr., Deltona"
                  :sale-price  39000
                  :living-area 874
                  :bedrooms 2
                  :bathrooms 1
                  :style 'ranch
                  :year 67
                  :location 'Deltona/Area-18
                  :sale-date '(8 28 90)
                  :cooling 'central
                  :heating 'central
                  :garage 'carport
                  :site '(75 100)
                  :pool 'no))


(setf case-21
      (make-instance 'case
                  :address "1664 Brady Dr., Deltona"
                  :sale-price 39900
                  :living-area 900
```

```
                :bedrooms 2
                :bathrooms 1
                :style 'ranch
                :year 67
                :location 'Deltona/Area-18
                :sale-date '(7 24 90)
                :cooling 'central
                :heating 'central
                :garage 'carport
                :site '(75 100)
                :pool 'no))


(setf case-22
      (make-instance 'case
                :address "2379 Lake Helen-Osteen, Deltona"
                :sale-price 41900
                :living-area 1101
                :bedrooms 2
                :bathrooms 2
                :style 'ranch
                :year 73
                :location 'Deltona/Area-17
                :sale-date '(7 28 90)
                :cooling 'none
                :heating 'central
                :garage 'carport
                :site '(90 100)
                :pool 'no))


(setf case-23
      (make-instance 'case
                :address "1062 Providence Blvd., Deltona"
                :sale-price 40000
                :living-area 1113
                :bedrooms 2
                :bathrooms 1
                :style 'ranch
                :year 86
                :location 'Deltona/Area-19
                :sale-date '(6 6 90)
                :cooling 'central
                :heating 'central
                :garage 'none
                :site '(80 125)
                :pool 'no))


(setf case-24
      (make-instance 'case
                :address "1620 Brady Dr., Deltona"
```

```
               :sale-price 42000
               :living-area 874
               :bedrooms 2
               :bathrooms 1
               :style 'ranch
               :year 67
               :location 'Deltona/Area-19
               :sale-date '(7 18 90)
               :cooling 'central
               :heating 'central
               :garage '1-car
               :site '(75 100)
               :pool 'no))


(setf case-25
      (make-instance 'case
               :address "2830 Thornberry Ct., Deltona"
               :sale-price 43900
               :living-area 910
               :bedrooms 2
               :bathrooms 1
               :style 'ranch
               :year 83
               :location 'Deltona/Area-18
               :sale-date '(8 31 90)
               :cooling 'central
               :heating 'central
               :garage '1-car
               :site '(80 125)
               :pool 'no))


(setf case-26
      (make-instance 'case
               :address "1387 Hartley Ave., Deltona"
               :sale-price 44500
               :living-area 984
               :bedrooms 3
               :bathrooms 1
               :style 'ranch
               :year 63
               :location 'Deltona/Area-19
               :sale-date '(8 14 90)
               :cooling 'central
               :heating 'central
               :garage 'carport
               :site '(75 100)
               :pool 'no))

(setf case-27
      (make-instance 'case
               :address "2860 E. Canal Rd., Deltona"
               :sale-price 40000
               :living-area 1450
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 73
               :location 'Deltona/Area-18
               :sale-date '(8 31 90)
               :cooling 'central
               :heating 'central
               :garage '1-car
               :site '(75 100)
               :pool 'no))


(setf case-28
      (make-instance 'case
               :address "1567 Ft. Smith, Deltona"
               :sale-price 42500
               :living-area 874
               :bedrooms 2
               :bathrooms 1
               :style 'ranch
               :year 69
               :location 'Deltona/Area-18
               :sale-date '(8 4 90)
               :cooling 'central
               :heating 'central
               :garage 'carport
               :site '(76 125)
               :pool 'yes))


(setf case-29
      (make-instance 'case
               :address "1378 Ft. Smith Blvd., Deltona"
               :sale-price 42000
               :living-area 1139
               :bedrooms 3
               :bathrooms 1
               :style 'ranch
               :year 75
               :location 'Deltona/Area-18
               :sale-date '(9 8 90)
               :cooling 'central
               :heating 'central
               :garage '1-car
               :site '(125 130)
               :pool 'no))
```

```
(setf case-30
      (make-instance 'case
                :address "669 Courtland Blvd., Deltona"
                :sale-price 43000
                :living-area 1027
                :bedrooms 2
                :bathrooms 2
                :style 'ranch
                :year 84
                :location 'Deltona/Area-18
                :sale-date '(6 29 90)
                :cooling 'central
                :heating 'central
                :garage '1-car
                :site '(81 125)
                :pool 'no))


(setf case-31
      (make-instance 'case
                :address "800 Abby Terr., Deltona"
                :sale-price 39000
                :living-area 1315
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 74
                :location 'Deltona/Area-19
                :sale-date '(7 31 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 125)
                :pool 'no))


(setf case-32
      (make-instance 'case
                :address "1517 N. Providence Blvd., Deltona"
                :sale-price 45900
                :living-area 1320
                :bedrooms 3
                :bathrooms 1
                :style 'ranch
                :year 71
                :location 'Deltona/Area-18
                :sale-date '(7 26 90)
                :cooling 'central
                :heating 'central
                :garage 'none
                :site '(75 100)
                :pool 'no))
```

```
(setf case-33
      (make-instance 'case
                :address "904 Mentmore, Deltona"
                :sale-price 44500
                :living-area 850
                :bedrooms 2
                :bathrooms 1
                :style 'ranch
                :year 85
                :location 'Deltona/Area-18
                :sale-date '(8 24 90)
                :cooling 'central
                :heating 'not-central
                :garage '1-car
                :site '(80 125)
                :pool 'no))


(setf case-34
      (make-instance 'case
                :address "358 Blytheville Ave., Deltona"
                :sale-price 46000
                :living-area 1000
                :bedrooms 2
                :bathrooms 1
                :style 'ranch
                :year 83
                :location 'Deltona/Area-19
                :sale-date '(7 1 90)
                :cooling 'central
                :heating 'central
                :garage '1-car
                :site '(80 125)
                :pool 'no))


(setf case-35
      (make-instance 'case
                :address "1039 Pioneer Dr., Deltona"
                :sale-price 45500
                :living-area 1120
                :bedrooms 3
                :bathrooms 1
                :style 'ranch
                :year 76
                :location 'Deltona/Area-19
                :sale-date '(8 4 90)
                :cooling 'central
                :heating 'central
                :garage 'carport
                :site '(83 135)
```

```
               :pool 'no))


(setf case-36
      (make-instance 'case
               :address "667 Ft. Smith Blvd., Deltona"
               :sale-price 42000
               :living-area 1064
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 85
               :location 'Deltona/Area-18
               :sale-date '(8 1 90)
               :cooling 'central
               :heating 'central
               :garage '1-car
               :site '(80 125)
               :pool 'no))


(setf case-37
      (make-instance 'case
               :address "1319 Anderson St., Deltona"
               :sale-price 44900
               :living-area 1140
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 83
               :location 'Deltona/Area-19
               :sale-date '(7 27 90)
               :cooling 'central
               :heating 'central
               :garage '1-car
               :site '(80 125)
               :pool 'no))


(setf case-38
      (make-instance 'case
               :address "1073 Abagail Dr., Deltona"
               :sale-price 47900
               :living-area 1000
               :bedrooms 2
               :bathrooms 2
               :style 'ranch
               :year 76
               :location 'Deltona/Area-19
               :sale-date '(9 6 90)
               :cooling 'central
               :heating 'central
               :garage '1-car
               :site '(80 125)
               :pool 'no))


(setf case-39
      (make-instance 'case
               :address "1076 Deltona Blvd., Deltona"
               :sale-price 46000
               :living-area 1400
               :bedrooms 2
               :bathrooms 1
               :style 'ranch
               :year 73
               :location 'Deltona/Area-19
               :sale-date '(7 13 90)
               :cooling 'central
               :heating 'central
               :garage 'carport
               :site '(80 125)
               :pool 'no))


(setf case-40
      (make-instance 'case
               :address "1358 W. Hartley Cir., Deltona"
               :sale-price 48000
               :living-area 1241
               :bedrooms 3
               :bathrooms 1
               :style 'ranch
               :year 64
               :location 'Deltona/Area-19
               :sale-date '(8 30 90)
               :cooling 'central
               :heating 'central
               :garage '1-car
               :site '(86 100)
               :pool 'no))


(setf case-41
      (make-instance 'case
               :address "950 Hanford Ln., Deltona"
               :sale-price 48900
               :living-area 1132
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 84
               :location 'Deltona/Area-18
               :sale-date '(8 10 90)
```

```
                  :cooling 'central
                  :heating 'central
                  :garage 'none
                  :site '(83 139)
                  :pool 'no))


(setf case-42
      (make-instance 'case
                  :address "2349 Otis Ave., Deltona"
                  :sale-price 49900
                  :living-area 1350
                  :bedrooms 2
                  :bathrooms 2
                  :style 'ranch
                  :year 77
                  :location 'Deltona/Area-17
                  :sale-date '(8 22 90)
                  :cooling 'central
                  :heating 'central
                  :garage 'carport
                  :site '(80 125)
                  :pool 'yes))


(setf case-43
      (make-instance 'case
                  :address "1030 Hemingway Dr., Deltona"
                  :sale-price 52000
                  :living-area 1100
                  :bedrooms 2
                  :bathrooms 2
                  :style 'ranch
                  :year 85
                  :location 'Deltona/Area-19
                  :sale-date '(8 2 90)
                  :cooling 'central
                  :heating 'central
                  :garage '1-car
                  :site '(80 125)
                  :pool 'no))


(setf case-44
      (make-instance 'case
                  :address "1025 Anderson St., Deltona"
                  :sale-price 51000
                  :living-area 1400
                  :bedrooms 3
                  :bathrooms 2
                  :style 'ranch
                  :year 73
```

```
                :location 'Deltona/Area-19
                :sale-date '(7 17 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 125)
                :pool 'no))


(setf case-45
      (make-instance 'case
                :address "2029 Apricot Dr., Deltona"
                :sale-price 50000
                :living-area 1137
                :bedrooms 2
                :bathrooms 2
                :style 'ranch
                :year 81
                :location 'Deltona/Area-16
                :sale-date '(7 20 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 125)
                :pool 'no))


(setf case-46
      (make-instance 'case
                :address "1363 Whitewood Dr., Deltona"
                :sale-price 52000
                :living-area 1622
                :bedrooms 2
                :bathrooms 2
                :style 'ranch
                :year 65
                :location 'Deltona/Area-19
                :sale-date '(7 28 90)
                :cooling 'central
                :heating 'central
                :garage 'carport
                :site '(100 101)
                :pool 'no))


(setf case-47
      (make-instance 'case
                :address "2889 Arredonda, Deltona"
                :sale-price 56550
                :living-area 1150
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 89
                :location 'Deltona/Area-18
                :sale-date '(8 31 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 125)
                :pool 'no))


(setf case-48
      (make-instance 'case
                :address "460 Oslo Dr., Deltona"
                :sale-price 55000
                :living-area 1147
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 89
                :location 'Deltona/Area-18
                :sale-date '(9 1 90)
                :cooling 'none
                :heating 'central
                :garage '2-car
                :site '(80 125)
                :pool 'no))


(setf case-49
      (make-instance 'case
                :address "1160 McCormick, Deltona"
                :sale-price 52900
                :living-area 1700
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 79
                :location 'Deltona/Area-19
                :sale-date '(7 25 90)
                :cooling 'central
                :heating 'central
                :garage 'none
                :site '(84 151)
                :pool 'yes))


(setf case-50
      (make-instance 'case
                :address "1130 E. Normandy Blvd., Deltona"
                :sale-price 56000
                :living-area 1689
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 79
                :location 'Deltona/Area-19
                :sale-date '(8 28 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(85 125)
                :pool 'no))


(setf case-51
      (make-instance 'case
                :address "2171 Dumas Dr., Deltona"
                :sale-price 55000
                :living-area 1285
                :bedrooms 2
                :bathrooms 2
                :style 'ranch
                :year 83
                :location 'Deltona/Area-18
                :sale-date '(6 30 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(87 125)
                :pool 'no))


(setf case-52
      (make-instance 'case
                :address "1700 N. Normandy Blvd., Deltona"
                :sale-price 53000
                :living-area 1700
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 76
                :location 'Deltona/Area-19
                :sale-date '(8 28 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(128 102)
                :pool 'no))


(setf case-53
      (make-instance 'case
                :address "2090 Roseway Dr., Deltona"
```

```
               :sale-price 52400
               :living-area 1580
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 80
               :location 'Deltona/Area-16
               :sale-date '(5 10 90)
               :cooling 'central
               :heating 'central
               :garage '1-car
               :site '(88 125)
               :pool 'no))


(setf case-54
      (make-instance 'case
               :address "766 E. Lacy Cir., Deltona"
               :sale-price 58000
               :living-area 1334
               :bedrooms 2
               :bathrooms 2
               :style 'ranch
               :year 77
               :location 'Deltona/Area-19
               :sale-date '(7 14 90)
               :cooling 'central
               :heating 'central
               :garage '2-car
               :site '(78 137)
               :pool 'no))


(setf case-55
      (make-instance 'case
               :address "1011 Prescott Blvd., Deltona"
               :sale-price 59900
               :living-area 1130
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 86
               :location 'Deltona/Area-18
               :sale-date '(8 25 90)
               :cooling 'central
               :heating 'central
               :garage '1-car
               :site '(80 125)
               :pool 'no))

(setf case-56
      (make-instance 'case
               :address "1401 Providence Blvd., Deltona"
               :sale-price 55000
               :living-area 1366
               :bedrooms 2
               :bathrooms 2
               :style 'ranch
               :year 81
               :location 'Deltona/Area-19
               :sale-date '(8 17 90)
               :cooling 'central
               :heating 'central
               :garage '2-car
               :site '(92 125)
               :pool 'no))


(setf case-57
      (make-instance 'case
               :address "2755 Candler Dr., Deltona"
               :sale-price 57000
               :living-area 1096
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 86
               :location 'Deltona/Area-16
               :sale-date '(6 15 90)
               :cooling 'central
               :heating 'central
               :garage '1-car
               :site '(80 125)
               :pool 'no))


(setf case-58
      (make-instance 'case
               :address "2709 Derby Dr., Deltona"
               :sale-price 59900
               :living-area 1150
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 86
               :location 'Deltona/Area-17
               :sale-date '(8 31 90)
               :cooling 'central
               :heating 'central
               :garage '1-car
               :site '(80 125)
               :pool 'no))
```

```
(setf case-59
      (make-instance 'case
                  :address "1911 Maderia Ave., Deltona"
                  :sale-price 60900
                  :living-area 1415
                  :bedrooms 3
                  :bathrooms 2
                  :style 'ranch
                  :year 86
                  :location 'Deltona/Area-17
                  :sale-date '(8 11 90)
                  :cooling 'central
                  :heating 'central
                  :garage '2-car
                  :site '(80 125)
                  :pool 'no))


(setf case-60
      (make-instance 'case
                  :address "2129 E. Gloria Dr., Deltona"
                  :sale-price 61200
                  :living-area 1204
                  :bedrooms 3
                  :bathrooms 2
                  :style 'ranch
                  :year 80
                  :location 'Deltona/Area-16
                  :sale-date '(9 8 90)
                  :cooling 'central
                  :heating 'central
                  :garage '2-car
                  :site '(80 125)
                  :pool 'no))


(setf case-61
      (make-instance 'case
                  :address "1842 Giles, Deltona"
                  :sale-price 60000
                  :living-area 1209
                  :bedrooms 3
                  :bathrooms 2
                  :style 'ranch
                  :year 83
                  :location 'Deltona/Area-16
                  :sale-date '(5 7 90)
                  :cooling 'central
                  :heating 'central
                  :garage '2-car
                  :site '(80 125)
                  :pool 'no))
```

```
(setf case-62
      (make-instance 'case
                :address "2889 Gallup Ct., Deltona"
                :sale-price 58400
                :living-area 1061
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 87
                :location 'Deltona/Area-18
                :sale-date '(9 1 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(92 144)
                :pool 'no))


(setf case-63
      (make-instance 'case
                :address "2419 Beck Cir., Deltona"
                :sale-price 61900
                :living-area 1190
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 87
                :location 'Deltona/Area-18
                :sale-date '(7 26 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(128 190)
                :pool 'no))


(setf case-64
      (make-instance 'case
                :address "1617 Ross, Deltona"
                :sale-price 62000
                :living-area 1150
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 86
                :location 'Deltona/Area-18
                :sale-date '(9 8 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 125)
```

```
               :pool 'no))


(setf case-65
      (make-instance 'case
               :address "3053 Lagoon Ave., Deltona"
               :sale-price 62900
               :living-area 1381
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 87
               :location 'Deltona/Area-17
               :sale-date '(7 27 90)
               :cooling 'central
               :heating 'central
               :garage '2-car
               :site '(91 125)
               :pool 'no))


(setf case-66
      (make-instance 'case
               :address "3221 Sardinia Terr., Deltona"
               :sale-price 62900
               :living-area 1172
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 87
               :location 'Deltona/Area-18
               :sale-date '(8 15 90)
               :cooling 'central
               :heating 'central
               :garage '2-car
               :site '(80 125)
               :pool 'no))


(setf case-67
      (make-instance 'case
               :address "2822 Fayson Cir., Deltona"
               :sale-price 62000
               :living-area 1480
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 88
               :location 'Deltona/Area-18
               :sale-date '(7 31 90)
               :cooling 'central
               :heating 'central
```

```
                :garage '2-car
                :site '(125 80)
                :pool 'no))


(setf case-68
      (make-instance 'case
                :address "1350 Bailey Ave., Deltona"
                :sale-price 62500
                :living-area 1340
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 86
                :location 'Deltona/Area-19
                :sale-date '(6 30 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 125)
                :pool 'no))


(setf case-69
      (make-instance 'case
                :address "960 Stratton St., Deltona"
                :sale-price 64000
                :living-area 1404
                :bedrooms 2
                :bathrooms 2
                :style 'ranch
                :year 84
                :location 'Deltona/Area-19
                :sale-date '(8 31 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(85 125)
                :pool 'no))


(setf case-70
      (make-instance 'case
                :address "769 Stratton St., Deltona"
                :sale-price 62500
                :living-area 1500
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 88
                :location 'Deltona/Area-16
                :sale-date '(9 11 90)
```

```
                 :cooling 'central
                 :heating 'central
                 :garage '2-car
                 :site '(80 125)
                 :pool 'no))


(setf case-71
      (make-instance 'case
                 :address "1432 Summit Hill Dr., Deltona"
                 :sale-price 62500
                 :living-area 1304
                 :bedrooms 3
                 :bathrooms 2
                 :style 'ranch
                 :year 88
                 :location 'Deltona/Area-16
                 :sale-date '(6 27 90)
                 :cooling 'central
                 :heating 'central
                 :garage '2-car
                 :site '(80 125)
                 :pool 'no))


(setf case-72
      (make-instance 'case
                 :address "878 Abbott Ave., Deltona"
                 :sale-price 67900
                 :living-area 1200
                 :bedrooms 2
                 :bathrooms 2
                 :style 'ranch
                 :year 89
                 :location 'Deltona/Area-19
                 :sale-date '(6 5 90)
                 :cooling 'central
                 :heating 'central
                 :garage '2-car
                 :site '(80 125)
                 :pool 'no))


(setf case-73
      (make-instance 'case
                 :address "494 Glenhaven, Deltona"
                 :sale-price 67000
                 :living-area 1625
                 :bedrooms 3
                 :bathrooms 2
                 :style 'ranch
                 :year 85
```

```
               :location 'Deltona/Area-18
               :sale-date '(6 30 90)
               :cooling 'central
               :heating 'central
               :garage '2-car
               :site '(100 150)
               :pool 'no))


(setf case-74
      (make-instance 'case
               :address "2574 Tryon Ave., Deltona"
               :sale-price 68850
               :living-area 1500
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 88
               :location 'Deltona/Area-16
               :sale-date '(6 6 90)
               :cooling 'central
               :heating 'central
               :garage '2-car
               :site '(108 116)
               :pool 'no))


(setf case-75
      (make-instance 'case
               :address "2930 Grimes St., Deltona"
               :sale-price 70900
               :living-area 1674
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 87
               :location 'Deltona/Area-18
               :sale-date '(8 25 90)
               :cooling 'central
               :heating 'central
               :garage '2-car
               :site '(80 125)
               :pool 'no))


(setf case-76
      (make-instance 'case
               :address "98 Fordham St., Deltona"
               :sale-price 71000
               :living-area 1600
               :bedrooms 3
               :bathrooms 2
```

```
                :style 'ranch
                :year 83
                :location 'Deltona/Area-19
                :sale-date '(8 29 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 125)
                :pool 'no))


(setf case-77
      (make-instance 'case
                :address "1001 Alladin Dr., Deltona"
                :sale-price 68500
                :living-area 1567
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 69
                :location 'Deltona/Area-19
                :sale-date '(8 15 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(125 137)
                :pool 'no))


(setf case-78
      (make-instance 'case
                :address "2257 E. Union Cir., Deltona"
                :sale-price 71500
                :living-area 1147
                :bedrooms 2
                :bathrooms 2
                :style 'ranch
                :year 74
                :location 'Deltona/Area-16
                :sale-date '(8 25 90)
                :cooling 'central
                :heating 'central
                :garage '1-car
                :site '(100 144)
                :pool 'yes))


(setf case-79
      (make-instance 'case
                :address "1954 Viking Ave., Deltona"
                :sale-price 72500
                :living-area 1500
```

```
                  :bedrooms 3
                  :bathrooms 2
                  :style 'ranch
                  :year 83
                  :location 'Deltona/Area-19
                  :sale-date '(7 18 90)
                  :cooling 'central
                  :heating 'central
                  :garage '2-car
                  :site '(80 125)
                  :pool 'no))


(setf case-80
      (make-instance 'case
                  :address "1554 Bavon Dr., Deltona"
                  :sale-price 74000
                  :living-area 1440
                  :bedrooms 3
                  :bathrooms 3
                  :style 'ranch
                  :year 83
                  :location 'Deltona/Area-16
                  :sale-date '(7 28 90)
                  :cooling 'central
                  :heating 'central
                  :garage '2-car
                  :site '(102 169)
                  :pool 'no))


(setf case-81
      (make-instance 'case
                  :address "1161 Algoma St., Deltona"
                  :sale-price 72000
                  :living-area 1556
                  :bedrooms 3
                  :bathrooms 2
                  :style 'ranch
                  :year 78
                  :location 'Deltona/Area-19
                  :sale-date '(7 3 90)
                  :cooling 'central
                  :heating 'central
                  :garage '2-car
                  :site '(80 125)
                  :pool 'yes))


(setf case-82
      (make-instance 'case
                  :address "826 Sweetbriar Dr., Deltona"
```

```
                :sale-price 73000
                :living-area 1454
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 88
                :location 'Deltona/Area-19
                :sale-date '(8 15 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(97 125)
                :pool 'no))


(setf case-83
      (make-instance 'case
                :address "834 N. Fourth Dr., Deltona"
                :sale-price 74900
                :living-area 1750
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 88
                :location 'Deltona/Area-19
                :sale-date '(7 1 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 125)
                :pool 'no))


(setf case-84
      (make-instance 'case
                :address "409 W. Taylorville St., Deltona"
                :sale-price 74900
                :living-area 1500
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 88
                :location 'Deltona/Area-16
                :sale-date '(6 30 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 125)
                :pool 'no))

(setf case-85
      (make-instance 'case
               :address "727 Vicksburg St., Deltona"
               :sale-price 76000
               :living-area 1704
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 88
               :location 'Deltona/Area-16
               :sale-date '(8 29 90)
               :cooling 'central
               :heating 'central
               :garage '2-car
               :site '(125 104)
               :pool 'no))


(setf case-86
      (make-instance 'case
               :address "2159 Shadowridge, Deltona"
               :sale-price 67500
               :living-area 1774
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 75
               :location 'Deltona/Area-16
               :sale-date '(8 11 90)
               :cooling 'central
               :heating 'central
               :garage '2-car
               :site '(115 100)
               :pool 'no))


(setf case-87
      (make-instance 'case
               :address "1175 S. Brickell, Deltona"
               :sale-price 77900
               :living-area 1586
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 88
               :location 'Deltona/Area-19
               :sale-date '(7 31 90)
               :cooling 'central
               :heating 'central
               :garage '2-car
               :site '(80 125)
               :pool 'no))
```

```
(setf case-88
      (make-instance 'case
                :address "2682 N. Timberlake Ave., Deltona"
                :sale-price 75000
                :living-area 1494
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 88
                :location 'Deltona/Area-16
                :sale-date '(6 16 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 125)
                :pool 'no))


(setf case-89
      (make-instance 'case
                :address "1319 Providence Blvd., Deltona"
                :sale-price 79000
                :living-area 1717
                :bedrooms 4
                :bathrooms 2
                :style 'ranch
                :year 83
                :location 'Deltona/Area-16
                :sale-date '(7 28 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 100)
                :pool 'no))


(setf case-90
      (make-instance 'case
                :address "1240 Humphrey Blvd., Deltona"
                :sale-price 78900
                :living-area 1835
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 87
                :location 'Deltona/Area-18
                :sale-date '(7 5 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 125)
                :pool 'no))
```

```
(setf case-91
     (make-instance 'case
                :address "944 Feather Dr., Deltona"
                :sale-price 84900
                :living-area 1611
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 80
                :location 'Deltona/Area-16
                :sale-date '(8 29 90)
                :cooling 'central
                :heating 'not-central
                :garage '2-car
                :site '(80 125)
                :pool 'no))


(setf case-92
     (make-instance 'case
                :address "1966 S. Old Mill Rd., Deltona"
                :sale-price 94000
                :living-area 1706
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 80
                :location 'Deltona/Area-16
                :sale-date '(8 31 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 125)
                :pool 'yes))


(setf case-93
     (make-instance 'case
                :address "1172 Peak Cir., Deltona"
                :sale-price 84900
                :living-area 1675
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 84
                :location 'Deltona/Area-18
                :sale-date '(8 31 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(100 150)
                :pool 'no))


(setf case-94
      (make-instance 'case
                :address "372 Oslo Dr., Deltona"
                :sale-price 89500
                :living-area 1620
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 87
                :location 'Deltona/Area-19
                :sale-date '(8 28 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 125)
                :pool 'no))


(setf case-95
      (make-instance 'case
                :address "1222 Feather Dr., Deltona"
                :sale-price 85000
                :living-area 1821
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 85
                :location 'Deltona/Area-16
                :sale-date '(6 28 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 125)
                :pool 'no))


(setf case-96
      (make-instance 'case
                :address "2884 Bardahl Ct., Deltona"
                :sale-price 85000
                :living-area 1739
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 86
                :location 'Deltona/Area-18
                :sale-date '(7 1 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(100 150)
                :pool 'no))


(setf case-97
      (make-instance 'case
                :address "675 Jena Dr., Deltona"
                :sale-price 89900
                :living-area 1556
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 78
                :location 'Deltona/Area-19
                :sale-date '(8 25 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 140)
                :pool 'yes))


(setf case-98
      (make-instance 'case
                :address "1582 Fentress, Deltona"
                :sale-price 87500
                :living-area 1700
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 85
                :location 'Deltona/Area-18
                :sale-date '(7 26 90)
                :cooling 'central
                :heating 'central
                :garage '2-car
                :site '(80 125)
                :pool 'yes))


(setf case-99
      (make-instance 'case
                :address "2001 Dixie Belle, Deltona"
                :sale-price 90500
                :living-area 1800
                :bedrooms 3
                :bathrooms 2
                :style 'ranch
                :year 80
                :location 'Deltona/Area-16
                :sale-date '(7 5 90)
```

```
                  :cooling 'central
                  :heating 'central
                  :garage '2-car
                  :site '(118 109)
                  :pool 'no))


(setf case-100
      (make-instance 'case
                  :address "801 Sylvia Dr., Deltona"
                  :sale-price 85000
                  :living-area 2176
                  :bedrooms 3
                  :bathrooms 2
                  :style 'ranch
                  :year 83
                  :location 'Deltona/Area-16
                  :sale-date '(8 18 90)
                  :cooling 'central
                  :heating 'central
                  :garage '2-car
                  :site '(85 125)
                  :pool 'no))


(setf case-101
      (make-instance 'case
                  :address "1920 W. Chapel, Deltona"
                  :sale-price 92500
                  :living-area 1875
                  :bedrooms 3
                  :bathrooms 2
                  :style 'ranch
                  :year 81
                  :location 'Deltona/Area-16
                  :sale-date '(8 24 90)
                  :cooling 'central
                  :heating 'central
                  :garage '2-car
                  :site '(125 128)
                  :pool 'yes))


(setf case-102
      (make-instance 'case
                  :address "2079 E. Prairie Cir., Deltona"
                  :sale-price 97500
                  :living-area 2000
                  :bedrooms 3
                  :bathrooms 2
                  :style 'ranch
                  :year 85
```

```
               :location 'Deltona/Area-16
               :sale-date '(5 17 90)
               :cooling 'central
               :heating 'central
               :garage '2-car
               :site '(80 125)
               :pool 'no))


(setf case-103
      (make-instance 'case
               :address "559 E. Lehigh Dr., Deltona"
               :sale-price 95500
               :living-area 1538
               :bedrooms 2
               :bathrooms 2
               :style 'ranch
               :year 79
               :location 'Deltona/Area-18
               :sale-date '(8 4 90)
               :cooling 'central
               :heating 'central
               :garage '2-car
               :site '(124 344)
               :pool 'yes))


(setf case-104
      (make-instance 'case
               :address "1239 Giovanni, Deltona"
               :sale-price 99000
               :living-area 1839
               :bedrooms 3
               :bathrooms 2
               :style 'ranch
               :year 86
               :location 'Deltona/Area-19
               :sale-date '(8 15 90)
               :cooling 'central
               :heating 'central
               :garage '2-car
               :site '(80 125)
               :pool 'yes))


(setf case-105
      (make-instance 'case
               :address "346 Magnoia Pl., Deltona"
               :sale-price 95000
               :living-area 2214
               :bedrooms 3
               :bathrooms 2
```

```
                 :style 'two-story
                 :year 87
                 :location 'Deltona/Area-19
                 :sale-date '(8 20 90)
                 :cooling 'central
                 :heating 'central
                 :garage 'carport
                 :site '(100 276)
                 :pool 'no))


(setf case-106
      (make-instance 'case
                 :address "1450 Saxon Blvd., Deltona"
                 :sale-price 99000
                 :living-area 2000
                 :bedrooms 3
                 :bathrooms 3
                 :style 'ranch
                 :year 76
                 :location 'Deltona/Area-16
                 :sale-date '(6 27 90)
                 :cooling 'central
                 :heating 'central
                 :garage 'none
                 :site '(80 125)
                 :pool 'yes))


(setf case-107
      (make-instance 'case
                 :address "2241 E. Union Cir., Deltona"
                 :sale-price 149900
                 :living-area 2091
                 :bedrooms 3
                 :bathrooms 2
                 :style 'ranch
                 :year 82
                 :location 'Deltona/Area-16
                 :sale-date '(8 21 90)
                 :cooling 'central
                 :heating 'central
                 :garage '2-car
                 :site '(100 125)
                 :pool 'no))
```

APPENDIX B

TEST CASES

```
(defun test-1 ()
  (setf *list-price* 30500)
  (setf (living-area *my-property*) 690)
  (setf (bedrooms *my-property*) 1)
  (setf (bathrooms *my-property*) 1)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 64)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'not-central)
  (setf (heating *my-property*) 'not-central)
  (setf (garage *my-property*) 'carport)
  (setf (site *my-property*) '(78 125))
  (setf (pool *my-property*) 'no))


(defun test-2 ()
  (setf *list-price* 34900)
  (setf (living-area *my-property*) 766)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 1)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 66)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'not-central)
  (setf (heating *my-property*) 'not-central)
  (setf (garage *my-property*) 'carport)
  (setf (site *my-property*) '(100 110))
  (setf (pool *my-property*) 'no))


(defun test-3 ()
  (setf *list-price* 39500)
  (setf (living-area *my-property*) 1200)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 84)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'not-central)
  (setf (heating *my-property*) 'not-central)
  (setf (garage *my-property*) 'none)
  (setf (site *my-property*) '(80 241))
  (setf (pool *my-property*) 'no))


(defun test-4 ()
  (setf *list-price* 39900)
  (setf (living-area *my-property*) 905)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 1)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 64)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '1-car)
  (setf (site *my-property*) '(82 100))
  (setf (pool *my-property*) 'no))


(defun test-5 ()
  (setf *list-price* 42900)
  (setf (living-area *my-property*) 988)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 1)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 74)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'none)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '1-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-6 ()
  (setf *list-price* 43500)
  (setf (living-area *my-property*) 807)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 1)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 75)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) 'carport)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-7 ()
  (setf *list-price* 43900)
  (setf (living-area *my-property*) 881)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 1)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 80)
  (setf (location *my-property*) 'Deltona/Area-17)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) 'none)
  (setf (site *my-property*) '(100 80))
  (setf (pool *my-property*) 'no))


(defun test-8 ()
  (setf *list-price* 44900)
  (setf (living-area *my-property*) 872)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 1)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 80)
  (setf (location *my-property*) 'Deltona/Area-17)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) 'carport)
  (setf (site *my-property*) '(75 125))
  (setf (pool *my-property*) 'no))


(defun test-9 ()
  (setf *list-price* 45000)
  (setf (living-area *my-property*) 1029)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 75)
  (setf (location *my-property*) 'Deltona/Area-17)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'not-central)
  (setf (heating *my-property*) 'not-central)
  (setf (garage *my-property*) '1-car)
  (setf (site *my-property*) '(75 100))
  (setf (pool *my-property*) 'no))


(defun test-10 ()
  (setf *list-price* 45900)
  (setf (living-area *my-property*) 1000)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 1)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 72)
  (setf (location *my-property*) 'Deltona/Area-17)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'none)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) 'none)
  (setf (site *my-property*) '(75 100))
  (setf (pool *my-property*) 'no))


(defun test-11 ()
  (setf *list-price* 47500)
  (setf (living-area *my-property*) 954)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 1)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 84)
  (setf (location *my-property*) 'Deltona/Area-18)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '1-car)
  (setf (site *my-property*) '(116 120))
  (setf (pool *my-property*) 'no))


(defun test-12 ()
  (setf *list-price* 49000)
  (setf (living-area *my-property*) 912)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 84)
  (setf (location *my-property*) 'Deltona/Area-16)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '1-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-13 ()
  (setf *list-price* 49500)
  (setf (living-area *my-property*) 988)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 1)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 80)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '1-car)
  (setf (site *my-property*) '(117 125))
  (setf (pool *my-property*) 'no))
```

```
(defun test-14 ()
  (setf *list-price* 49900)
  (setf (living-area *my-property*) 1065)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 80)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '1-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-15 ()
  (setf *list-price* 49900)
  (setf (living-area *my-property*) 1030)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 82)
  (setf (location *my-property*) 'Deltona/Area-18)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '1-car)
  (setf (site *my-property*) '(93 125))
  (setf (pool *my-property*) 'no))


(defun test-16 ()
  (setf *list-price* 52900)
  (setf (living-area *my-property*) 1173)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 1)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 75)
  (setf (location *my-property*) 'Deltona/Area-16)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) 'none)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'yes))


(defun test-17 ()
  (setf *list-price* 53900)
  (setf (living-area *my-property*) 1146)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 80)
  (setf (location *my-property*) 'Deltona/Area-17)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '1-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-18 ()
  (setf *list-price* 53900)
  (setf (living-area *my-property*) 1137)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 84)
  (setf (location *my-property*) 'Deltona/Area-18)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-19 ()
  (setf *list-price* 54500)
  (setf (living-area *my-property*) 1296)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 1)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 64)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'not-central)
  (setf (garage *my-property*) 'carport)
  (setf (site *my-property*) '(93 134))
  (setf (pool *my-property*) 'no))


(defun test-20 ()
  (setf *list-price* 54900)
  (setf (living-area *my-property*) 1092)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 79)
  (setf (location *my-property*) 'Deltona/Area-16)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '1-car)
  (setf (site *my-property*) '(79 125))
  (setf (pool *my-property*) 'no))


(defun test-21 ()
  (setf *list-price* 55000)
  (setf (living-area *my-property*) 1296)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 1)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 71)
  (setf (location *my-property*) 'Deltona/Area-16)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) 'carport)
  (setf (site *my-property*) '(75 100))
  (setf (pool *my-property*) 'yes))


(defun test-22 ()
  (setf *list-price* 56000)
  (setf (living-area *my-property*) 1230)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 84)
  (setf (location *my-property*) 'Deltona/Area-16)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '1-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-23 ()
  (setf *list-price* 56900)
  (setf (living-area *my-property*) 1428)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 76)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-24 ()
  (setf *list-price* 57700)
  (setf (living-area *my-property*) 1000)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 1)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 86)
  (setf (location *my-property*) 'Deltona/Area-17)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 130))
  (setf (pool *my-property*) 'yes))


(defun test-25 ()
  (setf *list-price* 57900)
  (setf (living-area *my-property*) 1225)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 83)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '1-car)
  (setf (site *my-property*) '(145 116))
  (setf (pool *my-property*) 'no))


(defun test-26 ()
  (setf *list-price* 58900)
  (setf (living-area *my-property*) 1500)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 74)
  (setf (location *my-property*) 'Deltona/Area-16)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))
```

```
(defun test-27 ()
  (setf *list-price* 59500)
  (setf (living-area *my-property*) 1350)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 78)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '1-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-28 ()
  (setf *list-price* 59900)
  (setf (living-area *my-property*) 1236)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 82)
  (setf (location *my-property*) 'Deltona/Area-17)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) 'carport)
  (setf (site *my-property*) '(196 100))
  (setf (pool *my-property*) 'no))


(defun test-29 ()
  (setf *list-price* 59900)
  (setf (living-area *my-property*) 1250)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 81)
  (setf (location *my-property*) 'Deltona/Area-16)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-30 ()
  (setf *list-price* 60750)
  (setf (living-area *my-property*) 1368)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 82)
  (setf (location *my-property*) 'Deltona/Area-17)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '1-car)
  (setf (site *my-property*) '(106 150))
  (setf (pool *my-property*) 'no))


(defun test-31 ()
  (setf *list-price* 61900)
  (setf (living-area *my-property*) 1124)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 85)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-32 ()
  (setf *list-price* 63000)
  (setf (living-area *my-property*) 1330)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 86)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(85 125))
  (setf (pool *my-property*) 'no))


(defun test-33 ()
  (setf *list-price* 63500)
  (setf (living-area *my-property*) 1205)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 83)
  (setf (location *my-property*) 'Deltona/Area-16)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-34 ()
  (setf *list-price* 64000)
  (setf (living-area *my-property*) 1400)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 86)
  (setf (location *my-property*) 'Deltona/Area-18)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-35 ()
  (setf *list-price* 64900)
  (setf (living-area *my-property*) 1350)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 87)
  (setf (location *my-property*) 'Deltona/Area-17)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'yes))


(defun test-36 ()
  (setf *list-price* 64900)
  (setf (living-area *my-property*) 1254)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 83)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-37 ()
  (setf *list-price* 66900)
  (setf (living-area *my-property*) 1360)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 85)
  (setf (location *my-property*) 'Deltona/Area-17)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-38 ()
  (setf *list-price* 67000)
  (setf (living-area *my-property*) 1582)
  (setf (bedrooms *my-property*) 4)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 86)
  (setf (location *my-property*) 'Deltona/Area-17)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '1-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-39 ()
  (setf *list-price* 67900)
  (setf (living-area *my-property*) 1190)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 84)
  (setf (location *my-property*) 'Deltona/Area-16)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))
```

```
(defun test-40 ()
  (setf *list-price* 68500)
  (setf (living-area *my-property*) 1310)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 1)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 73)
  (setf (location *my-property*) 'Deltona/Area-18)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) 'carport)
  (setf (site *my-property*) '(100 195))
  (setf (pool *my-property*) 'no))


(defun test-41 ()
  (setf *list-price* 69500)
  (setf (living-area *my-property*) 1604)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 79)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 137))
  (setf (pool *my-property*) 'no))


(defun test-42 ()
  (setf *list-price* 69900)
  (setf (living-area *my-property*) 1743)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 3)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 74)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(85 125))
  (setf (pool *my-property*) 'no))


(defun test-43 ()
  (setf *list-price* 69900)
  (setf (living-area *my-property*) 1324)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 81)
  (setf (location *my-property*) 'Deltona/Area-16)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-44 ()
  (setf *list-price* 71900)
  (setf (living-area *my-property*) 1500)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 1)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 89)
  (setf (location *my-property*) 'Deltona/Area-16)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(85 125))
  (setf (pool *my-property*) 'no))


(defun test-45 ()
  (setf *list-price* 73000)
  (setf (living-area *my-property*) 1844)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 79)
  (setf (location *my-property*) 'Deltona/Area-17)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) 'none)
  (setf (site *my-property*) '(125 125))
  (setf (pool *my-property*) 'no))


(defun test-46 ()
  (setf *list-price* 73900)
  (setf (living-area *my-property*) 1700)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 85)
  (setf (location *my-property*) 'Deltona/Area-16)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-47 ()
  (setf *list-price* 74900)
  (setf (living-area *my-property*) 1720)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 85)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(90 145))
  (setf (pool *my-property*) 'no))


(defun test-48 ()
  (setf *list-price* 75000)
  (setf (living-area *my-property*) 1536)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'two-story)
  (setf (year *my-property*) 87)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'none)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-49 ()
  (setf *list-price* 77500)
  (setf (living-area *my-property*) 1816)
  (setf (bedrooms *my-property*) 4)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 87)
  (setf (location *my-property*) 'Deltona/Area-17)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(114 121))
  (setf (pool *my-property*) 'no))


(defun test-50 ()
  (setf *list-price* 79200)
  (setf (living-area *my-property*) 1402)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 77)
  (setf (location *my-property*) 'Deltona/Area-16)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'yes))


(defun test-51 ()
  (setf *list-price* 79900)
  (setf (living-area *my-property*) 1712)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 74)
  (setf (location *my-property*) 'Deltona/Area-18)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(170 125))
  (setf (pool *my-property*) 'yes))


(defun test-52 ()
  (setf *list-price* 79900)
  (setf (living-area *my-property*) 1386)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 85)
  (setf (location *my-property*) 'Deltona/Area-18)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(100 125))
  (setf (pool *my-property*) 'yes))
```

```
(defun test-53 ()
  (setf *list-price* 79900)
  (setf (living-area *my-property*) 1548)
  (setf (bedrooms *my-property*) 4)
  (setf (bathrooms *my-property*) 3)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 87)
  (setf (location *my-property*) 'Deltona/Area-18)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) 'carport)
  (setf (site *my-property*) '(105 125))
  (setf (pool *my-property*) 'no))


(defun test-54 ()
  (setf *list-price* 82500)
  (setf (living-area *my-property*) 1700)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 86)
  (setf (location *my-property*) 'Deltona/Area-18)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-55 ()
  (setf *list-price* 85000)
  (setf (living-area *my-property*) 1674)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 88)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'yes))


(defun test-56 ()
  (setf *list-price* 86500)
  (setf (living-area *my-property*) 1648)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 3)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 78)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'yes))


(defun test-57 ()
  (setf *list-price* 89000)
  (setf (living-area *my-property*) 1500)
  (setf (bedrooms *my-property*) 2)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 77)
  (setf (location *my-property*) 'Deltona/Area-16)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(100 194))
  (setf (pool *my-property*) 'no))


(defun test-58 ()
  (setf *list-price* 92000)
  (setf (living-area *my-property*) 1800)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 83)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'yes))


(defun test-59 ()
  (setf *list-price* 94500)
  (setf (living-area *my-property*) 1870)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 87)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(80 125))
  (setf (pool *my-property*) 'no))


(defun test-60 ()
  (setf *list-price* 94900)
  (setf (living-area *my-property*) 1839)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 86)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'not-central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(120 125))
  (setf (pool *my-property*) 'no))


(defun test-61 ()
  (setf *list-price* 95000)
  (setf (living-area *my-property*) 1935)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 81)
  (setf (location *my-property*) 'Deltona/Area-16)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(95 151))
  (setf (pool *my-property*) 'yes))


(defun test-62 ()
  (setf *list-price* 98500)
  (setf (living-area *my-property*) 2700)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 70)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'not-central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(120 280))
  (setf (pool *my-property*) 'no))


(defun test-63 ()
  (setf *list-price* 99900)
  (setf (living-area *my-property*) 1913)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 87)
  (setf (location *my-property*) 'Deltona/Area-18)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(125 85))
  (setf (pool *my-property*) 'no))


(defun test-64 ()
  (setf *list-price* 104900)
  (setf (living-area *my-property*) 2487)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 88)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) 'none)
  (setf (site *my-property*) '(86 125))
  (setf (pool *my-property*) 'no))


(defun test-65 ()
  (setf *list-price* 109500)
  (setf (living-area *my-property*) 1935)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 83)
  (setf (location *my-property*) 'Deltona/Area-16)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(90 125))
  (setf (pool *my-property*) 'no))
```

```
(defun test-66 ()
  (setf *list-price* 112000)
  (setf (living-area *my-property*) 1956)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 81)
  (setf (location *my-property*) 'Deltona/Area-16)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(125 170))
  (setf (pool *my-property*) 'no))


(defun test-67 ()
  (setf *list-price* 119900)
  (setf (living-area *my-property*) 2265)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 3)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 72)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(100 140))
  (setf (pool *my-property*) 'yes))


(defun test-68 ()
  (setf *list-price* 122900)
  (setf (living-area *my-property*) 2176)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 86)
  (setf (location *my-property*) 'Deltona/Area-17)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(170 125))
  (setf (pool *my-property*) 'yes))


(defun test-69 ()
  (setf *list-price* 127500)
  (setf (living-area *my-property*) 1810)
  (setf (bedrooms *my-property*) 3)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 85)
  (setf (location *my-property*) 'Deltona/Area-18)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'not-central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(90 220))
  (setf (pool *my-property*) 'yes))


(defun test-70 ()
  (setf *list-price* 159900)
  (setf (living-area *my-property*) 2400)
  (setf (bedrooms *my-property*) 4)
  (setf (bathrooms *my-property*) 2)
  (setf (style *my-property*) 'ranch)
  (setf (year *my-property*) 71)
  (setf (location *my-property*) 'Deltona/Area-19)
  (setf (sale-date *my-property*) '(11 20 90))
  (setf (cooling *my-property*) 'central)
  (setf (heating *my-property*) 'central)
  (setf (garage *my-property*) '2-car)
  (setf (site *my-property*) '(150 125))
  (setf (pool *my-property*) 'yes))
```

REFERENCES

Adair, A. S., and W. S. McGreal. 1988. Computer assisted valuation of residential property: A Belfast case study. Real Estate Appraiser and Analyst 54 (Winter): 18-21.

Almy, Richard R., Robert J. Gloudemans, and Robert C. Denne. 1978. Assessment practices in the U.S. Chicago: International Association of Assessing Offices, Research and Technical Services Department. NTIS, 696881 PB-291 297/0.

American Institute of Real Estate Appraisers. 1988. Appraising residential properties. Chicago: American Institute of Real Estate Appraisers.

Ashley, Kevin D., and Edwina L. Rissland. 1987. Compare and contrast, a test of expertise. In Proceedings of the AAAI-87, 273-278.

________. 1988. A case-based approach to modeling legal expertise. IEEE Expert 3 (Fall): 70-77.

Bailey, David, Donna Thompson, and Jerald Feinstein. 1987. Similarity networks as a knowledge representation for space application. In Third conference on artificial intelligence for space application, part I, Hustsville, Alabama, November 2-3, 1987. Washington, D.C.: National Aeronautics and Space Administration, 279-284.

________. 1988. Similarity networks: A knowledge representation for information analysis and retrieval. PC AI 2 (July-August): 29-32.

Boyce, Byrl N., and William N. Kinnard, Jr. 1984. Appraising real property. Lexington, Massachusetts: D.C. Heath and Company, and Society of Real Estate Appraisers.

Carbone, Robert. 1987. Notes on the Carbone-Longini feedback system. Property Tax Journal 6 (September): 161-172.

Carbonell, Jaime G. 1982. Experiential learning in analogical problem solving. In Proceedings of AAAI-82, Pittsburgh, Pennsylvania, 168-171.

________. 1986. Derivational analogy: A theory of reconstructive problem solving and expertise acquisition. In Machine intelligence, edited by R. Michalski, J. Carbonell, and T. Mitchell. Los Altos, California: Morgan-Kaufmann, 371-392.

Diaz, Aida, real estate agent. 1990. Personal interview by author, 26 September, Deltona, Florida.

Creteau, Paul G. 1974. Real estate appraising (step-by-step), 2nd ed. Portland, Maine: Castle Publishing Co.

Fieldson, Curtis, real estate appraiser. 1990. Telephone interview by author, 24 September and 12 November, from Winter Park, Florida to Chicago.

________. 1990. Questionnaire sent by author, 22 October, from Winter Park, Florida to Chicago.

Hammond, Kristian J. 1986. CHEF: A model of case-based planning. In Proceedings of AAAI-86, Philadelphia, Pennsylvania, 267-271.

________. 1987. Explaining and repairing plans that fail. In Proceedings of IJCAI-87, 109-114.

Jaffe, Austin J. 1985. The future of appraising. The Appraisal Journal 53 (October): 621-625.

Kass, Alex M., and David B. Leake. 1988. Case-based reasoning applied to constructing explanations. In Proceedings of a workshop on case-based reasoning, edited by Janet Kolodner, 190-208. San Mateo, California: Morgan Kaufmann Publishers, Inc.

Kolodner, Janet L. 1983. Maintaining organization in a dynamic long-term memory. Cognitive Science 7: 243-280.

________. 1983. Reconstructive memory: A computer model. Cognitive Science 7: 281-328.

________. 1987. Extending problem solver capabilities through case-based inference. In Proceedings of the fourth international machine learning workshop, 167-178.

________, ed. 1988. Proceedings of a workshop on case-based reasoning. San Mateo, California: Morgan Kaufmann Publishers, Inc.

Kolodner, Janet, and Chris Riesbeck. 1989. Case-based reasoning: Tutorial notes. Detroit: Eleventh International Joint Conference on Artificial Intelligence.

Kolodner, Janet L., and Christopher K. Riesbeck, eds. 1986. Experience, memory, and reasoning. Hillsdale, New Jersey: Lawrence Erlbaum Associates, Publishers.

Kolodner, Janet L., and Robert L. Simpson. 1989. The MEDIATOR: analysis of an early case-based problem solver. Cognitive Science 13: 507-549.

MacKellar, Bonnie, and Fred Maryanski. 1988. Reasoning by analogy in knowledge base systems. In Proceedings - Fourth international conference on data engineering in Los Angeles California, February 1-5, 1988. New York, New York: IEEE, 242-249.

Riesbeck, Christopher K. 1988. An interface for case-based knowledge acquisition. In Proceedings of a workshop on case-based reasoning, edited by Janet Kolodner, 312-326. San Mateo, California: Morgan Kaufmann Publishers, Inc.

________. 1988. Knowledge acquisition for case-based reasoning systems. In Lyndon B. Johnson Space Center, second annual workshop on space operations automation and robotics, November 1988. Washington, D.C.: National Aeronautics and Space Administration, 289-292.

Riesbeck, Christopher K., and Roger C. Schank. 1989. Inside case-based reasoning. Hillsdale, New Jersey: Lawrence Erlbaum Associates, Publishers.

Shearer, Kathleen, real estate appraiser. 1990. Personal interview by author, 27 September, Altamonte Springs, Florida.

Shinn, Hong S. 1988. Abstractional analogy: A model of analogical reasoning. In Proceedings of a workshop on case-based reasoning, edited by Janet Kolodner, 370-387. San Mateo, California: Morgan Kaufmann Publishers, Inc.

Stanfill, Craig. 1988. Learning to read: A memory-based model. In Proceedings of a workshop on case-based reasoning, edited by Janet Kolodner, 402-413. San Mateo, California: Morgan Kaufmann Publishers, Inc.

Stanfill, Craig, and David L. Waltz. 1986. Toward memory-based reasoning. Communications of the ACM 29 (December): 1213-1228.

________. 1988. The memory-based reasoning paradigm. In Proceedings of a workshop on case-based reasoning, edited by Janet Kolodner, 414-424. San Mateo, California: Morgan Kaufmann Publishers, Inc.

Sycara, Katia. 1988. Using case-based reasoning for plan adaptation and repair. In Proceedings of a workshop on case-based reasoning, edited by Janet Kolodner, 425-434. San Mateo, California: Morgan Kaufmann Publishers, Inc.

Weber, Bruce R. 1990. Application of geographic information systems to real estate market analysis and appraisal. Appraisal Journal 58 (January): 127-132.

Winston, Patrick H. 1980. Learning and reasoning by analogy. Communications of the ACM 23 (December): 689-703.

__________. 1982. Learning new principles from precedents and exercises. Artificial Intelligence 19: 321-350.

Woolery, Arlo, and Sharon Shea, eds. 1985. Introduction to computer assisted valuation. Boston: Oelgeschlager, Gunn & Hain, Publishers, Inc., and Lincoln Institute of Land Policy.