MODELING A PROJECT SCOPE USING A CASE-BASED REASONING APPROACH

Alfredo F. Serpell, José M. Rueda
Department of Construction Engineering and Management, Universidad Católica de Chile, Chile

ABSTRACT: The availability of a good, complete scope definition in the early stages of a project is widely recognized by industry practitioners as a key factor for overall project success. This paper presents a Project Scope Modeling Methodology for computerized decision support during the definition of a new project scope. The methodology is based on the effective reutilization of historical project scope definitions through the application of Case Based Reasoning (CBR), an Artificial Intelligence approach. In CBR, the previous experiences are reused in solving new situations reducing the complexities of modeling reasoning processes. By using CBR, the scope modeling methodology helps to find and reuse the most relevant historical information, allowing to easily consult and combine information from multiple scope definitions in a computerized environment. The resulting scope definitions are ready to serve as input information for different planning purposes. The application for conceptual cost estimating is discussed.

KEYWORDS: project, scope, planning, modeling, case-based reasoning, methodology.

1 INTRODUCTION

As many other early project management activities, the definition of a project's scope is usually carried out with constrained resources and under stiff time restrictions. In order to define the scope of a new project, project managers often reuse scope related information from previous projects. The reutilization process may combine a series of computer and non-computer activities, several data sources, tools, methods and assumptions, all driven by the project managers' experience.

Over the last decades, diverse approaches for computerized assistance have been developed to improve information handling processes and easing problem solving. A number of innovative approaches have been used to build Decision Support Systems (DSS), which are interactive, computer-based systems helping users to solve various semi-structured and unstructured problems (Sage, 1991). As pointed out by Pal and Shiu (2004), successful decision support systems have been employed in various domains, such as law, medicine, e-commerce, finance, and engineering. As in other fields of application, possibilities for improving the scope definition process exist through the intelligent gathering and use of information following a learning process.

This paper presents a Project Scope Modeling Methodology for computerized decision support during the definition of the scope of a new project. The methodology is built upon Case-based Reasoning, an Artificial Intelligence approach to problem solving by means of reusing information. In addition, the application of the methodology is also discussed in the context of CASEST, a prototypical system for scope modeling and conceptual cost estimating (Rueda, 2006).

2 CASE BASED REASONING (CBR)

Case-based reasoning (CBR) is a thriving paradigm for reasoning and learning in Artificial Intelligence (Leake, 1996). According to Pal and Shiu (2004), Case-based reasoning may be defined as a model of reasoning that incorporates problem solving, understanding, and learning. Bergmann et al. (2004) define CBR as a problem solving methodology that models human reasoning and thinking.

In Case-based reasoning, a new problem is solved by recalling and adapting a solution that was successfully applied to a previous problem with similar characteristics. People routinely use Case-based reasoning instead of employing methods or solving procedures. Nevertheless, as stated by Kolodner (1991), they may suffer from an inability to consistently recall the appropriate prior solutions, distinguish between important and unimportant features, recall prior experiences under time pressures, and deal with incomplete and uncertain information in new problems.

Case-based Decision Support Systems work with a database of past experiences named cases. As defined by Mubarak (2004), cases are episodic couplings of problems and solutions. This concept can be described as a function between domain P (problems) and domain S (solutions) where cases are the pairs (problem, solution). The problem part of a case is often represented as a list of descriptive parameters, while the solution is represented in a number of ways such as text, diagrams, graphics, trees or hierarchical structures, among others. In many situations, an evaluation of the solution is provided with the case. A domain of evaluations (E) can be added and the case will be the triplet (P, S, E). As explained by Pal and Shiu (2004), the problem-solving life cycle in a Case-based reasoning system consists essentially of the following four steps:

1. Given a new problem, the system retrieves similar previously experienced cases (e.g., problem–solution–evaluation) whose problem part is judged to be similar.
2. The case(s) retrieved are reused by copying or integrating its solution parts. An initial solution is found.
3. The initial solution is modified or adapted in an attempt to solve the new problem.
4. The new solution obtained is retained in the database for future use (along with its problem and evaluation if available) once it has been confirmed or validated.

3 CASE REPRESENTATION

Case representation is an important activity for Case-based Reasoning. As pointed out by Aamodt and Plaza (1994), it involves finding an appropriate structure for describing, visualizing and using case contents. Figure 1 presents a building project example (core/shell project) that shows the case representation proposed for the Project Scope Modeling Methodology described later.

The Scope Modeling Cases contain a problem and a solution part. The problem part is represented as a "project description" or list of project parameters in the form of attribute-value pairs. These descriptive parameters specify a particular need that is satisfied by the solution part through a single project scope definition. The solution stores all the scope related information, which is represented as a tree or hierarchical structure similar to a Project Breakdown Structure. The elements of the structure with a plus sign (+) have their children elements collapsed or hidden. Conversely, the elements with a minus sign (-) have all their children elements expanded or shown.

SCOPE MODELING CASE

PROBLEM - Project Description

Attribute	Value
Basement area [m^2]	960
Building footprint [m^2]	1.930
Building area [m^2]	11.800
Building levels	6
Average level height [m]	2,7
Building structure	Reinforced Concrete

SOLUTION - Scope Definition

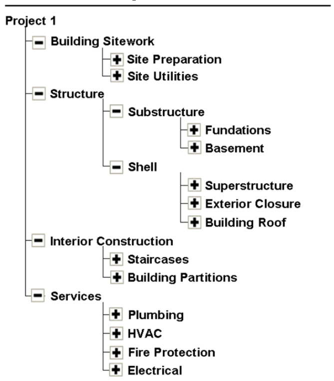

Figure 1. Case Representation for Project Scope Modeling Methodology.

4 CASE ATTRIBUTES

In a CBR problem solving cycle, the case attributes (along with its values) allow to evaluate the similarity between cases in order to retrieve appropriate information. The selection of the attributes for a Case-based Decision Support System is a domain dependent activity. The example of the Figure 1 shows a set of attributes that may be useful for describing building projects. As explained by Watson (1997), attributes must be predictive in a useful manner, influencing the outcome of the process and describing the circumstances in which a case is expected to be retrieved in the future.

Several methods or algorithms to choose attributes are described in the literature (see, for instance, Pal and Shiu 2004). However, as stated by Kolodner (1993), for practical applications attributes should be chosen using expert criteria.

Once selected, the attributes are incorporated into the decision support system. All cases of a certain type use the same set of attributes and are generally clustered into the database.

5 PROJECT SCOPE MODELING METHODOLOGY

The Project Scope Modeling Methodology is designed to be executed by a computerized system to assist the user's decision making during the definition of the scope of a new project. The methodology describes an interactive work process in which the user makes decisions while the system suggests useful information.

The Scope Modeling Methodology proposes a three step process to prepare a new scope definition: in first place, specifying a "project description" for a new scope modeling case; secondly, evaluating similarity and retrieving an initial solution or breakdown structure; and lastly, adapting the initial breakdown structure retrieved.

5.1 *Specification of the "project description" for the new scope modeling case*

The first step in the process is to create a "project description" or problem part for the new scope modeling case. The decision support system presents the user a set of attributes (previously determined according to the type of project) while the user enters a corresponding set of values that describe the characteristics of the new project. The list of attribute-value pairs specifies the problem part for a new scope modeling case, making possible to start the evaluation of the similarity between the new situation and the stored cases.

5.2 *Evaluation of the similarity and information retrieving*

The next step in the process is to obtain an initial solution or breakdown structure by retrieving historical information from the database. In order to find a useful solution, it is necessary to compare the project description of the new scope modeling case with all project descriptions stored in the database. These comparisons are performed by the decision support system using the Nearest Neighbor Algorithm.

As explained by Pal and Shiu (2004), the Nearest Neighbor Algorithm allows evaluating the similarity between two cases using similarity functions. A global similarity function is usually built up from a number of socalled local similarity functions, one for each descriptive attribute involved. In this way, the global similarity between two cases is calculated as a weighted sum of local similarities. The following equation represents the Nearest Neighbor Algorithm:

$$Similarity(P,C) = \sum_{i=1}^{n} w_i \times sim_i(P_i, C_i) \qquad (1)$$

The equation describes a situation for which P and C are two cases compared for similarity, n is the number of attributes in each case, i is an individual attribute from 1 to n, w_i is the feature weight of attribute i, and sim_i is a local similarity function. Similarities are usually normalized to fall within the range 0 to 1, where 1 means a perfect match and 0 indicates a total mismatch. The Table 1 shows the application of local similarity functions for numerical and text attributes:

Table 1. Local similarity functions for numerical and text attributes.

Attributes	Local Similarity Function	Application Example
Numerical	$sim(p_i, c_i) = 1 - \frac{dist(p_i, c_i)}{R_{max}}$ $dist(p_i, c_i) = Abs(p_i - c_i)$ $R_{max} = c_{max} - c_{min}$ c_{max} = Maximum value stored in database for a given attribute c_{min} = Minimum value stored in database for a given attribute	Calculating the similarity for the "basement area" attribute, in square meters: $p_i = 1{,}300$ $c_i = 1{,}520$ c_{max} = 1,790 (Maximum basement area in database) c_{min} = 1,050 (Minimum basement area in database) $dist(p_i, c_i) = Abs(1{,}300 - 1{,}520) = 220$ Distance must be normalized by R_{max} in order to have a similarity between [0 and 1]. $R_{max} = 1{,}790 - 1{,}050 = 740$ $sim(p_i, c_i) = 1 - 220/740 = 0.70 =$ **70%**
Text	Whenever possible, text attributes are ordered and each one receives a numerical value, then text attributes are treated as numerical. If ordering is not possible then: $sim(p_i, c_i) = \begin{cases} 1 & \text{when } p_i = c_i \\ 0 & \text{else} \end{cases}$	The attribute "space restrictions" may have the following values: "none" = 1 "low" = 2 "medium" = 3 "high" = 4 "very high" = 5 p_i = "low", c_i = "very high" c_{max} = "very high" (Maximum value in database) c_{min} = "none" (Minimum value in database) $dist(p_i, c_i) = Abs(2 - 5) = 3$ $R_{max} = 5 - 1 = 4$ $sim(p_i, c_i) = 1 - 3/4 = 0.25 =$ **25%**

The Table 2 shows the application of the Nearest Neighbor Algorithm to measure the similarity between two cases:

Generally, the similarity calculation continues until all cases in the database have been compared, and ranked according to their similarity to the new target problem. As shown in the example of table 2, attributes that are considered more important may have their importance denoted by weighting them more heavily in the case-matching process. The strategies for determining the weights assigned to each attribute are diverse, ranging from the application of weights provided by experts, the use of techniques such as Genetic Algorithms (for an overview, see Craw and Jarmulak 1999), and in some cases the use of weights provided by the users. In addition, researchers and implementers can decide not to use weights at all.

After all Nearest Neighbor comparisons are performed, the breakdown structure of the most similar case is retrieved from database as an initial solution and is presented to the user for the realization of an interactive adaptation process, which is also assisted by Case-based Reasoning.

5.3 *Adaptation of the initial breakdown structure*

Once retrieved, the solution found must be adapted in order to better fit the new situation. The user of the system drives the adaptation process using his domain knowledge to revise and modify the initial breakdown structure. Adjustments are carried out whether adding or deleting elements. In turn, the system uses Case-based reasoning to suggest suitable scope information for adaptation.

Figure 2 illustrates the adaptation approach used. The element "Site Utilities" is shown as selected for revision and adjustment. Each time an element is selected for adaptation, the system presents to the user a list of possible elements that can be added to the structure as children of that particular node. The elements of the list come from an Item Library, which is a compilation of elements extracted from all breakdown structures stored in the database. In order to help the user to decide additions or deletions of elements, the system also displays percentages indicating the occurrence of each element across the k (i.e. k=3) most similar projects ranked through Nearest Neighbor Algorithm.

Table 2. Application example of Nearest Neighbor Algorithm.

Attribute	New Case	Stored Case	Distance	c_{max}	c_{min}	Maximum Range	Local Similarity	Weighted Local Similarity
Basement area [m^2]	960	1,260	300	1,750	760	990	69.6 / 100	12.5 / 18
Building footprint [m^2]	1,930	1,550	380	2,300	860	1,440	73.6 / 100	11.04 / 15
Building area [m^2]	11,800	14,050	2,250	21,300	4,400	16,900	86.6 / 100	25.1 / 29
Building levels	6	9	3	15	5	10	70 / 100	11.2 / 16
Average level height [m]	2.7	2.75	0.05	2.68	3	0.32	84.3 / 100	6.7 / 8
Building structure *	Reinforced Concrete	Reinforced Concrete	-	-	-	-	100 / 100	14 / 14
							Total Similarity	$\sum = 80.54$ %

* Text values have not been ordered and treated as numerical

The adaptation of elements can be performed following a top-down revision, with the adjustments starting from the top level (project level) to the highest detail levels, and checking all the elements of a level before going to the next one.

When all the adjustments conclude, a scope definition for the new project is obtained. This new breakdown structure is then ready to be used for planning purposes.

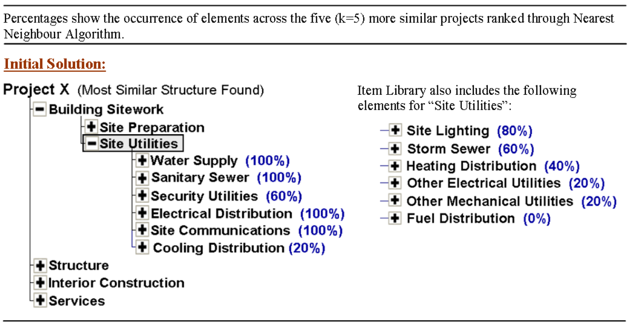

Figure 2. Case Adaptation.

6 OVERVIEW OF CASEST

CASEST is a prototypical system for scope modeling and conceptual cost estimating. It automatically generates construction cost estimates using the breakdown structures obtained with the Project Scope Modeling Methodology. Figure 3 depicts the system architecture. CASEST is conformed by three basic types of components: the user interfaces, the system program modules, and the system databases.

In CASEST, a program module for Project Scope Modeling executes all the support tasks to assist the preparation of a new project scope definition. The automated approach of the Scope Modeling Methodology makes possible to carry out the scope definition process in a short time and with a limited effort.

Once a new breakdown structure have been prepared, CASEST generates project quantities through a series of parametric relationships incorporated into a parametric model. The values established in the project description are used as input information for the model. Numerous parametric relationships process these values into quantities for the elements of the structure located at the highest levels of detail. Afterwards, the Costing program module multiplies all project quantities by its corresponding unit construction costs using data from a cost database expressed on a per-unit basis. Estimating reports are then generated and presented to the user.

In such a way, the system allows to easily obtain detailed estimates starting from a simple project description at a conceptual level. The resulting estimates allow knowing, in addition to the total construction cost of the project, the detailed costs of the main project components and of any other project items.

7 SYSTEM VALIDATION

Data for the study was collected from 17 building projects constructed in the city of Santiago, Chile between the years 2003 and 2005. The buildings are in the range of 7 to 27 floors and between 1 and 3 basement levels. The total constructed area for these buildings are between 4,615 and 22,020 square meters. All projects have a structure based on reinforced concrete walls. In table 3, detailed information of each project is shown.

Table 3. Descriptive information of collected cases.

Project	Basement Levels	Area Basement Levels [m²]	Floors	Total Floors Area [m²]	Building Footprint [m²]	Total Area [m²]
1	1	1,000	7	3,615	512	4,615
2	2	2,180	15	6,638	437	8,818
3	3	3,150	27	18,250	659	21,400
4	2	1,350	8	5,200	635	6,550
5	2	4,380	24	16,248	663	20,628
6	3	1,334	12	5,189	413	6,523
7	1	3,014	12	11,392	929	14,406
8	2	1,800	13	7,150	542	8,950
9	2	5,624	20	16,396	809	22,020
10	1	1,130	9	4,070	432	5,200
11	3	2,960	14	11,368	760	14,328
12	1	1,450	19	6,124	316	7,574
13	3	3,413	18	13,104	720	16,517
14	3	4,430	*	16,905	450	21,335
15	2	3,530	11	13,270	1,190	16,800
16	3	1,525	20	5,793	310	7,318
17	2	2,362	13	9,092	683	11,454

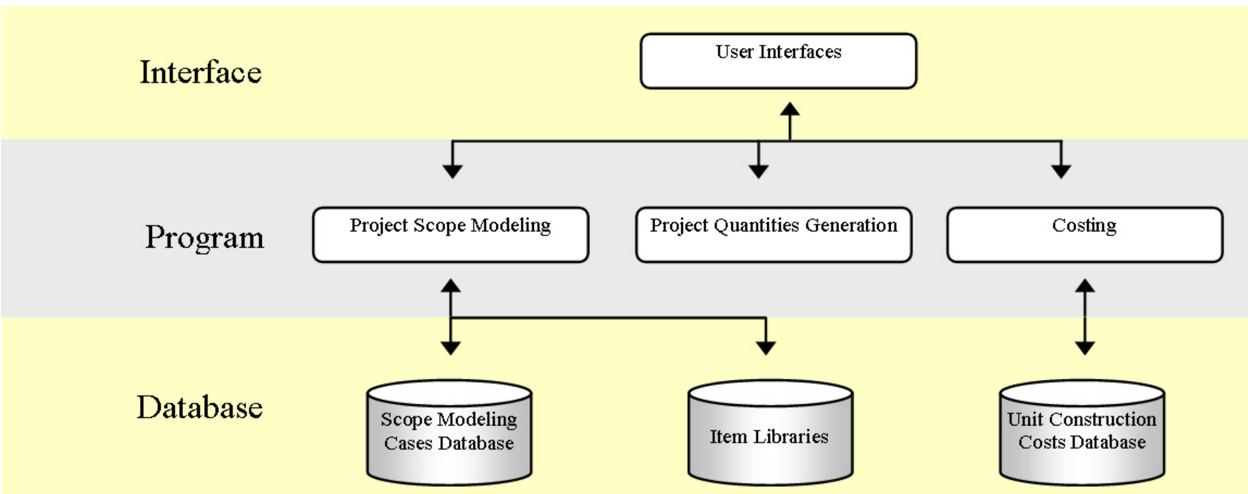

Figure 3. Architecture of CASEST.

The validation process consisted in estimating the collected construction project costs using the prototype of CASEST. The proposed automated scheme allowed carrying out the conceptual estimating processes with a low effort and time in each opportunity. The estimated costs were compared later with the real construction costs to obtain accuracy results.

The projects used to develop the Case Base for modeling received a special treatment during the test process. Each time one of these projects was going to be estimated, its information was retired from the case base including both, its set of descriptive attributes and its work breakdown structure or scope definition. In this way it was avoided that for each case where one of these projects were estimated, the system would deliver the same project as the result of the search and evaluation of similarity, a fact which would damage the validity of the application of the modeling system.

The similarity was evaluated at the project level because in this case the capability of the system for composing scopes was not used. The attributes selected for evaluating the similarity were: number of basement levels, total basement area, area of levels over ground, number of floors, building footprint, and type of structure of the building. The predetermined weighting values for each of these attributes are shown in table 4. These weights were determined asking experienced personnel of the construction company that built the projects.

Table 4. Predetermined weighting values for similarity evaluation.

Attribute	Weight
Basement levels	9.1%
Total basement area	21.7%
Over ground area	46.9%
Number of floors	14.3%
Building footprint	2.9%
Building structure type	5.1%

The results obtained from the evaluation of similarity are shown in table 5. In the table are included the results of total similarity, local similarity and weighted local similarity. In the case of local weighted similarities, the maximum values that an attribute could present are presented.

Table 5. Results of projects' similarity evaluation.

SL Local Similarity SLP Weighted Local Similarity		Similarity percentage for attributes % Basement Levels	Basement Levels Area	Building Area	Buildings Floors	Buildings Foot print	Building Structure
Project 1	SL	50 / 100	82.7 / 100	75.8 / 100	70 / 100	91.9 / 100	100 / 100
Total Similarity: 75.8%	SLP	4.5 / 9.1	17.9 / 21.7	35.5 / 46.9	10 / 14.3	2.6 / 2.9	5.1 / 5.1
Project 2	SL	100 / 100	73.1 / 100	98.9 / 100	80 / 100	60.7 / 100	100 / 100
Total Similarity: 89.6%	SLP	9.1 / 9.1	15.8 / 21.7	46.4 / 46.9	11.4 / 14.3	1.7 / 2.9	5.1 / 5.1
Project 3	SL	100 / 100	91.7 / 100	96.5 / 100	90 / 100	71.7 / 100	100 / 100
Total Similarity: 94.3%	SLP	9.1 / 9.1	19.9 / 21.7	45.2 / 46.9	12.8 / 14.3	2.08 / 2.9	5.1 / 5.1
Project 4	SL	100 / 100	91.7 / 100	96.5 / 100	90 / 100	71.7 / 100	100 / 100
Total Similarity: 94.3%	SLP	9.1 / 9.1	19.9 / 21.7	45.2 / 46.9	12.8 / 14.3	2.08 / 2.9	5.1 / 5.1
Project 5	SL	50 / 100	73.4 / 100	86.3 / 100	85 / 100	98.9 / 100	100 / 100
Total Similarity: 81.1%	SLP	4.5 / 9.1	15.9 / 21.7	40.4 / 46.9	12.1 / 14.3	2.8 / 2.9	5.1 / 5.1
Project 6	SL	100 / 100	73.1 / 100	98.9 / 100	80 / 100	60.7 / 100	100 / 100
Total Similarity: 88.4%	SLP	8.5 / 8.5	19.8 / 27.1	43.1 / 43.6	10.6 / 13.3	1.6 / 2.7	4.8 / 4.8
Project 7	SL	50 / 100	74.9 / 100	71.1 / 100	95 / 100	41.4 / 100	100 / 100
Total Similarity: 74%	SLP	4.5 / 9.1	16.2 / 21.7	33.3 / 46.9	13.5 / 14.3	1.2 / 2.9	5.1 / 5.1
Project 8	SL	50 / 100	79 / 100	78.5 / 100	70 / 100	84.6 / 100	100 / 100
Total Similarity: 76.1%	SLP	4.5 / 9.1	17.1 / 21.7	36.8 / 46.9	10 / 14.3	2.4 / 2.9	5.1 / 5.1
Project 9	SL	100 / 100	90.2 / 100	86.6 / 100	75 / 100	75 / 100	100 / 100
Total Similarity: 87.3%	SLP	9.1 / 9.1	19.5 / 21.7	40.6 / 46.9	10.7 / 14.3	2.1 / 2.9	5.1 / 5.1
Project 10	SL	50 / 100	85.8 / 100	94.2 / 100	75 / 100	65.8 / 100	100 / 100
Total Similarity: 84.9%	SLP	4.5 / 9.1	18.6 / 21.7	44.1/ 46.9	10.7 / 14.3	1.9 / 2.9	5.1 / 5.1
Project 11	SL	50 / 100	89.9 / 100	86.6 / 100	95 / 100	65.3 / 100	100 / 100
Total Similarity: 85.2%	SLP	4.5 / 9.1	19.5 / 21.7	40.6 / 46.9	13.5 / 14.3	1.8 / 2.9	5.1 / 5.1
Project 12	SL	50 / 100	73.7 / 100	71 / 100	95 / 100	-4 / 100	100 / 100
Total Similarity: 72.4%	SLP	4.5 / 9.1	16 / 21.7	33.3 / 46.9	13.5 / 14.3	-0.1 / 2.9	5.1 / 5.1
Project 13	SL	100 / 100	97.1 / 100	96.8 / 100	90 / 100	78.4 / 100	100 / 100
Total Similarity: 95.8%	SLP	9.1 / 9.1	21 / 21.7	45.4 / 46.9	12.8 / 14.3	2.2 / 2.9	5.1 / 5.1
Project 14	SL	50 / 100	84.2 / 100	96.4 / 100	80 / 100	67.4 / 100	100 / 100
Total Similarity: 86.3%	SLP	4.5 / 9.1	18.2 / 21.7	45.2 / 46.9	11.4 / 14.3	1.9 / 2.9	5.1 / 5.1
Project 15 A	SL	50 / 100	51.3 / 100	98.3 / 100	95 / 100	96.5 / 100	100 / 100
Total Similarity: 77.8%	SLP	7.2 / 14.5	14.1 / 27.6	36.2 / 36.9	13.7 / 14.5	4.9 / 5.1	1.4 / 1.4
Project 15 B	SL	0 / 100	52.8 / 100	92.1 / 100	85 / 100	96.5 / 100	100 / 100
Total Similarity: 90.2%	SLP	0 / 0.8	0.4 / 0.8	60.8 / 66.1	17.1 / 20.2	9.3 / 9.7	2.4 / 2.4
Project 15 C	SL	50 / 100	78.3 / 100	90.3 / 100	80 / 100	83.3 / 100	100 / 100
Total Similarity: 87.4%	SLP	0.4 / 0.8	0.6 / 0.8	59.7 / 66.1	16.1 / 20.2	8 / 9.7	2.4 / 2.4
Project 16	SL	100 / 100	81.6 / 100	79.6 / 100	35 / 100	-41.6 / 100	100 / 100
Total Similarity: 73%	SLP	9.1 / 9.1	17.7 / 21.7	37.3 / 46.9	5 / 14.3	-1.2 / 2.9	5.1 / 5.1
Project 17	SL	100 / 100	87.8 / 100	86.7 / 100	100 / 100	62.1 / 100	100 / 100
Total Similarity: 90%	SLP	9.1 / 9.1	19 / 21.7	40.6 / 46.9	14.3 / 14.3	1.8 / 2.9	5.1 / 5.1

In some cases like projects 12 or 16, the obtained results were negative when evaluating the similarity of building footprint. This occurs when the value of the attribute is very far away from the range of values in the case database for this attribute. This is due to the way of calculating the similarity using the closer neighbor algorithm and it simply states numerically a lacking of similarity.

As can be appreciated in the table, the total similarity values always presented satisfactory results. The average of total similarity so obtained was 84.4%. The assigned admissible similarity limit was set at 65%, meaning that in no case was necessary to use the scope definition system starting from zero information.

Even with a reduced number of modeling cases the obtained results were positive. This fact demonstrates what is informed in the CBR literature that the important issue for the case base is not the number of cases but the variety of them. In table 6 are presented the variation ranges for the descriptive attributes of the modeling cases. It can be seen the great variety of the project cases.

Table 6. Range of variation for attributes of modeling cases.

Attributes	**Variation range**
Basement levels	1 – 3
Total basement area [m^2]	1,000 – 5,624
Over ground floors	7 – 27
Total area of over ground levels [m^2]	3,615 – 18,250
Total area [m^2]	4,615 – 22,020

7.1 *Results of estimated costs*

In table 7 the conceptual cost estimates obtained through the application of the scope modeling methodology are presented.

Table 7. Conceptual cost estimates obtained using the scope modeling methodology.

Project	**Real Cost (UF)**	**Estimated Cost UF**	**Accuracy %**
1	15,647	16,895	+7.98
2	30,988	28,942	-6.60
3	74,402	70,277	-5.54
4	25,560	23,703	-7.27
5	67,087	72,951	+8.74
6	27,720	23,482	-15.29
7	55,080	50,704	-7.94
8	24,802	29,115	+17.39
9	73,865	74,759	+1.21
10	21,240	18,592	-12.47
11	48,657	51,811	+6.48
12	27,600	25,730	-6.78
13	56,090	59,864	+6.73
14	83,520	74,308	-11.03
15	72,720	55,421	-23.79
16	26,922	28,041	+4.16
17	46,821	38,567	-17.63
Average of absolute accuracy			9.8

The results correspond to the direct structure of construction costs expressed in UF which is a Chilean indexed currency. Each UF is approximately equal to US$ 36.

All results so obtained are considered acceptable and fall within the range for conceptual estimating of building projects of ±30% as suggested by Siqueira (1999) and Abourizk et al. (2002). The average absolute accuracy was 9.8 %. The results confirm the fact that if one has good scope information for input then the accuracy of cost conceptual estimating is improved to a great extent.

8 CONCLUSIONS

Case Based Reasoning decision support was proposed for Project Scope Modeling. A prototypical system for scope modeling and conceptual cost estimating was implemented as an application tool. The automation and support of CBR problem solving makes possible to carry out the scope definition process in a short time and with a limited effort. Each stage of the process can be assisted without the participation of manual information handling. Large amounts of scope related information from numerous projects can be quickly evaluated for similarity, retrieved, and combined.

The similarity measures used in CBR also reduce the subjectivity in searching for information, allowing the user to access most similar cases. Case Based Reasoning decision support formalizes adequately the reuse of information in the scope definition process, reflecting naturally the human reasoning processes.

The breakdown structures obtained through the application of the methodology can be used in a computer environment as input information for planning purposes. Scope definitions were used for automatically generating construction costs estimates in this case.

By means of the development and application of the Scope Modeling Methodology for the conceptual estimating of construction costs it is possible to conclude the following:

- The process proposed for the evaluation of the similarity between projects, allows an estimator to have the best information contained in the system memory in each opportunity and to make decisions on the basis of expressed objective information in numerical terms. This way the subjective appreciations were eliminated, formalizing the information process search.
- The visual ordering and aspects of the hierarchic structures to the interior of the computational system contribute to the modeling process, allowing to suitably representing the information of the project, to organize the content of the scope definition, and to conduct operations of elimination, addition and replacement of its elements easily.
- The organization and storage of scope modeling cases in flat memory constitute the simplest way to implement the data bases of a scope modeling system based on cases. The increase in the time of evaluation of the similarity as the database of cases increases is not significant given the great processing speed of information of current computational tools.
- With the application of the scope modeling methodology it is possible to obtain detailed cost estimations that allow knowing in addition to the total cost of the

project, the detailed costs of the main components and of any component.
- The use of a conceptual cost estimation system based on the application of the scope modeling methodology tends to be simple and it does not require greater learning efforts. This is due to the fact that suitably formalizes the management of information and knowledge by means of the application of the CBR, reflecting in a natural way, the common practices in conceptual estimation and human reasoning.
- Historical information is a fundamental element for the CBR and also for the conceptual estimation of costs. It turns out indispensable to have historical information to apply the scope modeling methodology. Although this aspect can be considered like a restriction for CBR, the use of previous experiences allows finding solutions to new problems without having the necessity to use a high level of explicit knowledge.

REFERENCES

Aamodt A. and Plaza E. (1994), CBR: Foundational Issues, Methodological Variations, and System Approaches, Artificial Intelligence Communications, Vol. 7, No. 1, 39-59.

Abourizk S., Babey G. y Karumanasseri G. (2002), Estimating the Cost of Capital Projects: An Empirical Study of Accuracy Levels for Municipal Government Projects, Canadian Journal of Civil Engineering, Vol. 29, No. 5, 653-661.

Bergmann R., Althoff K-D., Breen S., Göker M., Manago M., Traphöner R. and Wess S. (2004), Developing Industrial Case-Based Reasoning Applications: The INRECA Methodology, Springer-Verlag, Berlin.

Craw S. and Jarmulak J. (1999), Genetic Algorithms for Feature Selection and Weighting, IJCAI-99 Workshop on Automating the Construction of Case-Based Reasoners, Stockholm, Sweden.

Kolodner J. (1991), Improving Human Decision Making through Case-Based Decision Aiding, AI Magazine, Vol. 12, Issue 2, 52-68.

Kolodner J. (1993), Case-Based Reasoning, Morgan Kauffman Publishers Inc., San Francisco, CA.

Leake D. (1996), Case-Based Reasoning: Experiences, Lessons, and Future Directions, AAAI Press, Menlo Park, CA.

Mubarak K. (2004), Case Based Reasoning for Design Composition in Architecture, PhD Thesis, Carnegie Mellon University, Pittsburgh, PA.

Pal S. and Shiu S. (2004), Foundations of Soft Case-Based Reasoning, Wiley-Interscience, Hoboken, NJ.

Rueda, J.M. (2006), Application of Case-Based Reasoning to Modeling Projects Scope for Conceptual Cost Estimating, M.Sc. Thesis, Pontificia Universidad Católica de Chile, Santiago, Chile.

Sage A. (1991), Decision Support Systems Engineering, Wiley-Interscience, Hoboken, NJ.

Siqueira I. (1999), Neural Network-Based Cost Estimating, Concordia University, Montreal, Canada.

Watson I. (1997), Applying Case-Based Reasoning: Techniques for Enterprise Systems, Morgan Kauffman, San Francisco, CA.