AN EXPERT SYSTEM FOR ESTIMATING CONSTRUCTION COST

by

Zohar Herbsman

University of Florida

Gainesville, Florida, U.S.A.

and

Richard A. Wall

United States Navy

Washington, District of Columbia, U.S.A.

Introduction.

One area of construction that inherently lends itself to automation is project estimating. There are currently a number of programs in this area that are commercially available. These programs, which exist in a database management environment, nominally use either a flat, indexed sequential or random file architecture to store the detailed information required for an estimate. Typical of the information included are line items for labor, material, equipment and subcontractors along with their associated quantities and costs. A nominal estimating session consists of selecting the appropriate line items required for the job from this list of candidate items. We have carried this concept one step further and developed a system that employs a hierarchical structure to define the required components of labor, material, equipment and subcontractors. While not employing AI techniques, the architecture that we have designed and implemented does lay the foundation and provides a skeleton structure around which an expert system can be built. This system is currently implemented in the database management environment of Rbase 5000, and includes five levels of data and relationship hierarchy that define the elements that an estimator would use when drafting an estimate.

Contained within this paper is a description of the structure implemented in the Rbase 5000 DBMS, and a discussion of the architecture of an expert system that would use this database as the foundation for its knowledge base.

Architecture of the Project Estimating Database.

The structure of the estimating database contains five levels of hierarchy. The base level, which is referred to as the COMPONENT level, contains the information nominally found as line items in the commercially available estimating programs. Contained herein are data relative to the cost and quantity, as applicable, of labor, material, equipment and subcontractors. This data can be as definitive as required. For example, the cost of labor, which is broken down by trade, can also be described by expertise within a trade (i.e.,-journeyman, apprentice, etc.) and can include such factors as overtime, hazard pay and fringes. This level is the only area within the database where detailed cost information is available. While conceptually identical, the four areas that comprise the COMPONENT level (labor, material, equipment and subcontractors) are contained within separate files. This design allows ease of updating and a more rapid program response as the individual components migrate up the hierarchy to the goal state. Figures 1, 2 and 3 depict representative samples of the labor, material and equipment COMPONENT files, respectively.

Figure 1 shows a typical labor component file.

DESCRIPTION	UNIT	COST/UNIT
Bricklayer	Hr	$20.50
Bricklayer Helper	Hr	$16.00
Carpenter	Hr	$20.00
Carpenter Foreman	Hr	$22.00
Cement Finisher	Hr	$19.20
Equipment Operator (light)	Hr	$19.45
Equipment Operator (heavy)	Hr	$20.60
Equipment Operator (crane)	Hr	$21.05
Equipment Oiler	Hr	$17.50
Laborer, Building	Hr	$15.90
Laborer, Foreman	Hr	$17.90
Mechanic	Hr	$21.80
Plumber	Hr	$23.05
Plumber Foreman	Hr	$25.05
Rodman	Hr	$21.75
Rodman Foreman	Hr	$23.75
Structural Steel Worker	Hr	$21.70
Structural Steel Foreman	Hr	$23.70
Truckdriver (light)	Hr	$16.35
Truckdriver (heavy)	Hr	$16.60
Welder	Hr	$22.55

Figure 1.
Partial Listing of the LABOR File at the COMPONENT Level (5).

Figure 2 shows a typical material component file.

DESCRIPTION	UNIT	COST/UNIT
Asphalt	CY	$58.80
Concrete, field mix 2250 psi	CY	$44.60
Concrete, field mix 3000 psi	CY	$47.45
Concrete, bituminous	Ton	$27.30
Concrete Block, 4X8X16	EA	$ .53
Concrete Block 8X8X16	Ea	$ .78
Gravel, bank run	Ton	$2.25
Lumber, 2X4	MBF	$315.00
Lumber, 2X6	MBF	$320.00
Lumber, 2X10	MBF	$365.00
Lumber, 1X6	MBF	$720.00
Lumber, 1X12	MBF	$890.00
Perlite	CF	$1.04
Sand, bank run fill	Ton	$3.50
Sand, mortar	Ton	$7.00
Stone, 3/8"-1/2"	Ton	$8.75
Stone, 3/4"-1 1/2"	Ton	$7.75

Figure 2.
Partial Listing of the MATERIAL File at the COMPONENT LEVEL (5).

Figure 3 shows a typical equipment component file.

DESCRIPTION	UNIT	COST/UNIT
Aggregate Spreader	Day	$51.20
Chipping Machine	Day	$137.00
Dozer, 75 hp	Day	$206.00
Dozer, 200 hp	Day	$634.80
Dozer, 300 hp	Day	$931.00
Dozer, 410 hp	Day	$1157.00
Dumptruck, 12 ton	Day	$220.20
Dumptruck, 16 ton	Day	$279.60
Excavator, hydraulic, 0.5 CY	Day	$262.00
Excavator, hydraulic, 0.75 CY	Day	$340.20
Excavator, hydraulic, 1.0 CY	Day	$443.80
Excavator, hydarulic, 1.5 CY	Day	$579.60
Excavator, hydraulic, 2.0 CY	Day	$801.00
Excavator, hydraulic, 3.5 CY	Day	$1412.00
Loader, backhoe, 48 hp	Day	$156.00
Loader, backhoe, 80 hy	Day	$234.00
Loader, frontend, 1.5 CY	Day	$271.00
Loader, frontend, 2.25 CY	Day	$421.00
Loader, frontend, 2.5 CY	Day	$600.00
loader, frontend, 5.5 CY	Day	$845.00
Paving Machine	Day	$544.40
Roller, sheepsfoot, 130 hy	Day	$290.60
Roller, tandem, 5 ton	Day	$85.00
Roller, tandem, 10 ton	Day	$175.20
Roller, vibrating	Day	$281.40
Scaper, towed	Day	$89.95
Tractor Trailer, lowbed	Day	$224.30
Trailer, platform	Day	$96.00

Figure 3.
Partial Listing of the EQUIPMENT File at the COMPONENT Level (5).

The four levels above the COMPONENT level serve to define relationships that aggregate certain components into a specific unit. These levels are the CREW level, the TASK level, the ITEM level and the JOB level.

Figure 4 shows the graphical representation of this hierarchy.

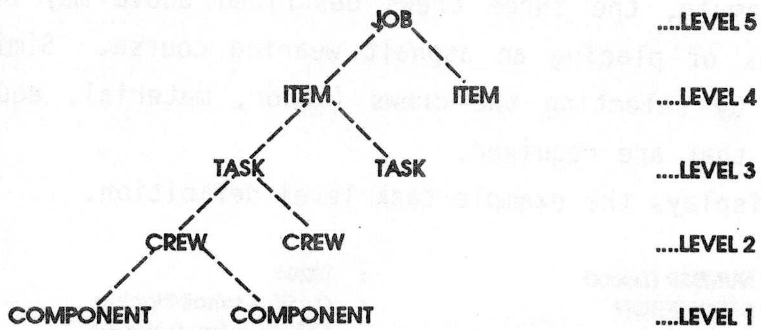

Figure 4.
Estimating Database Hierarchy

At the CREW level, certain COMPONENTs are grouped to specify the make-up of a crew. For example, an asphalt placing crew may include 1 labor foreman, 7 building laborers and 2 equipment operators (5). All the specific information for this crew is drawn from the labor COMPONENT database.

Figure 5 depicts this structure and the information contained therein for the example crew.

CREW NUMBER
(LCxxxx for LABOR CREWS
MCxxxx for MATERIAL CREWS
ECxxxx for EQUIPMENT CREWS
SCxxxx for SUBCONTRACTOR CREWS) : LC0104
CREW DESCRIPTION : (LABOR, Asphalt Placing)
QUANTITY OF PRODUCTION : 3050
UNIT OF PRODUCTION : SY

COMPONENT #1:	(LABOR, Laborer, Building)	QUANTITY #1: 56 hr
COMPONENT #2:	(LABOR, Laborer, Foreman)	QUANTITY #2: 8 hr
COMPONENT #3:	(LABOR, Equip Operator (heavy)	QUANTITY #3: 16 hr

Figure 5.
Crew Level Structure

Likewise, an equipment crew to place asphalt may be defined from the equipment COMPONENT database to include a paving machine and a tandem roller. Finally, a material crew to place asphalt would include the delivered asphalt.

At the TASK level, crews are conglomerated to specify construction tasks. For example, the three crews described above may be used to specify the task of placing an asphalt wearing course. Similar tasks may be defined by selecting the crews (labor, material, equipment or subcontractors) that are required.

Figure 6 displays the example task level definition.

TASK NUMBER (Txxxx)	:	T0206
TASK DESCRIPTION	:	(TASK, Asphalt Placing (1.5" Wearing Course))
QUANTITY OF PRODUCTION	:	1000
UNIT OF PRODUCTION	:	SY

CREW # 1	LABOR, Asphalt Placing)	QUANTITY #1: 0.33 crewday
CREW # 2	(EQUIPMENT, Asphalt Placing)	QUANTITY #2: 0.33 crewday
CREW # 3	(MATERIAL, Asphalt (1/5"))	QUANTITY #3: 45 CY

Figure 6.
Task Level Structure

Using similar methodology, the ITEM level consolidates tasks into items. As an example, the task of placing the asphalt wearing course can be combined with the task of placing a concrete base course to define an item that describes a road surface.

The top level of the hierarchy defines the job by grouping items. These definitions are contained at the JOB level. A length of road surface combined with the items for site preparation and grade adjustment may describe the job of constructing a roadway.

While each definition is unique as to its constituent elements, the key to this structure is that similar definitions exist at each level. For example, a number of material crews for asphalt could be defined at the crew level, each one detailing a different wearing course thickness.

This difference would affect the quantity of asphalt required for a given surface area. From these options, the most appropriate is then selected and added to the estimate.

As can be seen, this architecture allows the individual components of construction to migrate up the hierarchy to define the labor, material, equipment and subcontractor requirements for a job that is being estimated. In practice, the definitions contained within the JOB, ITEM, TASK and CREW levels actually specify requirements that migrate down the hierarchy until they can be satisfied at the COMPONENT level. The detailed information identified with the component (i.e.,-quantity, cost, etc.) is then attached to the requesting unit and conveyed in turn up the hierarchy.

While not a production system, this program has been used to validate the concept of this architecture with a large measure of success. The transformation of this database into a knowledge base is seen as the next logical step, as the relationships that exist in the four topmost levels fulfill the intent of a frame as used in expert systems (1). For example, the definition of a task can easily be transformed into a TASK frame, where the requirements for crews are the slots and the candidate variables become the crews, which, in themselves are also frames. The crew frames, then, fill their slots with variables from the COMPONENT level. While some other modifications to the basic frame configuration will also be discussed, this represents the foundation of the knowledge base for use with the construction project estimating expert system.

Overview of the Project Estimating Expert System.

While it is well understood that expert system implementation should not be applied across all disciplines, the domain of construction estimating satisfies the six classic requirements that are used to gauge a domain's suitability to application of an expert system. These six necessary criteria are (2):

1) GENUINE EXPERTS MUST EXIST. Within all disciplines of the construction industry, there are numerous professional estimators whose accuracy is documented daily.

2) THE EXPERTS MUST GENERALLY AGREE ABOUT THE CHOICE OF AN ACCEPTABLE SOLUTION. There exists widespread consensus on the majority of engineering practices that are used to formulate construction estimates. While this accord may not exist on estimates for special or exotic construction, these areas need not be included within the scope of the expert system's capabilities.

3) THE EXPERTS MUST BE ABLE TO ARTICULATE AND EXPLAIN THEIR PROBLEM SOLVING METHODOLOGY. The inherently algorithmic strategies used in estimating provide a fairly straight-forward methodology that lends itself to adaptation in an automated environment.

4) THE PROBLEMS OF THE DOMAIN MUST REQUIRE COGNITIVE, NOT PHYSICAL SKILLS. The task of the expert system will not be to accomplish the task, but rather, to plan how it will be accomplished.

5) THE TASK CANNOT BE TOO DIFFICULT. The scope of the expert system's capabilities will, by necessity, be limited to those engineering practices that are industry standard.

6) THE PROBLEM SHOULD NOT REQUIRE COMMON SENSE OR GENERAL WORLD KNOWLEDGE. This criteria speaks mainly to the size and complexity of the knowledge base and sophistication of the control knowledge. While the system will include a fair amount of control knowledge, it is not envisioned to operate as a stand alone expert. Rather, the nominal application will be as an expert assistant to an human estimator, who will be invited to intervene whenever common sense or general world knowledge is required to continue pursuit of the solution.

A seventh informal criteria mandates that the cost of implementation be justified against the potential use of the system. In other words, the system must have a payback (primarily in terms of cost savings) that warrants its original implementation. To satisfy this requirement, it is important to consider the low profit margin associated with project estimating. Depending upon the type of construction that a particular contractor practices, only one out of every ten estimate/bids may ultimately generate a profit producing job. It is therefore crucial to a contractor's financial health that estimates not only be accurate, but also producible with a minimum of time and effort.

A final factor that recommends this domain to implementation is concurrent research in expert system utilization in other areas of construction. For example, systems are currently being designed to deal with the administrative and financial aspects of construction, purchasing and inventory control (6) and job activity scheduling (4). Since it appears that the execution of projects will soon exploit AI technology, it is only natural that the planning and estimating functions employ similar methodology. This has the advantage of providing a single, consolidated knowledge base for all functionally connected expert systems.

Even though the previously described relational database does provide an automated job estimating capability, the principal component is still the human estimator who prepares an estimate by selecting construction activities from various menus. As such, there exists very limited control knowledge within the program itself. To transition this structure to an expert system knowledge base, certain enhancements are required to alleviate the dependency on the human estimator and supplement this function with an inference engine and the associated control structure. These enhancements affect three areas of the previously described database/frame structure: 1) a more general description of the activity definitions to the system, 2) the type of knowledge contained within a frame, and 3) the manner in which control knowledge is communicated and utilized.

The database architecture derives what flexibility it has from the fact that similar activity definitions can exist at discrete levels of the hierarchy; recall that a number of material crews for various thicknesses of asphalt can be defined at the CREW level, each one differing in the quantity provided. This allows the estimator to select the crew that is appropriate to the particular job being estimated. However, the use of this material is not as idealized as the database structure would indicate. In one scenario, the costs associated with the asphalt may include transportation charges to the job site. As such, the asphalt would be defined at the CREW level. In another scenario, the contractor may elect to provide its own transportation of the asphalt. Under this circumstance, the asphalt (material crew) would be combined with a dump truck (equipment crew) and a truck driver (labor crew) to now define the asphalt at the TASK level. In still another scenario, a large contractor (or subcontractor) may operate a batch plant and desire to use the constituent components of the batch plant as the material CREW definition. In this case, the production of the batch plant would provide the TASK level definition, and the delivered asphalt would be an ITEM level definition once the TASK of transportation was included. From this it can be seen that the same result (asphalt on the job site) can manifest at three different levels of the hierarchy. To gain the flexibility of selection described above, it is therefore necessary to allow similar definitions to exist at many levels of the hierarchy and not to just multiple occurrences at only one level.

With a structure such as this, the notion of a global hierarchy dissolves. What remains are subgoals that depend upon a hierarchy to the COMPONENT level, with each subgoal sufficient unto itself. Under this structure, a subgoal (which was earlier defined as a CREW, TASK, ITEM or JOB in the database environment) will rest at the top of its own hierarchy, the depth of which is determined exclusively by the number and complexity of the constituent parts that describe its definition.

When operating in the database environment, the CREW, TASK, ITEM and JOB definitions contain the routing information that directs the program to the constituent components of the hierarchy at the levels indicated, as Figures 5 and 6 show. In the absence of the levels, however, another method is required to permit the inference engine to isolate for consideration all the candidate frames that may satisfy the current subgoal. Toward this end, we propose to supplant the level structure with a data dictionary that contains the frame descriptions in a (verb.noun) couplet. This technique, which is borrowed from the field of Value Engineering, is used to describe the essential function of the construction activity whose component parts constitute the frame. The essential function is that activity which is necessary to fulfill the minimum need of the user (7). By using this technique, the number and complexity of the descriptors will be minimized, thereby simplifying the search algorithms of the system. In addition, since the couplets define only the essential function of the activity, the estimate will be driven toward a no frills (least cost) solution. For example, a frame that describes adjusting the elevation of the grade may be titled (RAISE.GRADE). The REQUIREMENT slots would specify (PROVIDE.FILL) and (SPREAD.FILL) as the component parts. The (PROVIDE.FILL) requirement may then be satisfied by a subcontractor frame that is titled (PROVIDE.FILL) or by another frame, with the same title, that has as its REQUIREMENTS (BUY.FILL), the inference engine is able to determine the field of candidate frames from which it can select one to satisfy the subgoal state. Which frame the inference engine ultimately selects will depend upon the control knowledge in the candidate frames and in the context, which are the subjects of subsequent enhancements.

The database scheme allows for communication between the COMPONENT level and the top four levels. However, for an expert system to function with this information, all frames must have the capability to communicate among themselves and pass control knowledge, as it is required or generated, to the inference engine. To this end, three additional slots are added to each frame. The first two describe the instances, scenarios and/or environments of the problem state relative to the usage of the frame. The first is called USEGOOD and contains factors that describe when the frame should be used. In a manner similar to frame descriptions, factors are described by using a (noun.condition) couplet. For example, a frame describing the construction of a gypsum board wall system on metal studs may contain a USEGOOD factor of (COST.LOW). This would serve to recommend this frame for use whenever the (COST.LOW) factor appeared in the expert system context, indicating that minimal cost was a design criteria of the estimate. Likewise, the second slot, USE-

BAD, contains factors whose presence would deny the use of the frame. In the example above, the USEBAD slot may include the factor (FIRE RATING.retardant), indicating that the frame incorporating metal studs should not be invoked when there is a requirement for a fire rated wall. These two slots are allowed to contain as many factors as required to comprehensively describe the domain conditions affecting the desirability of invoking the frame. As such, they contain the control knowledge that must be matched to the knowledge in the context to direct the search toward the goal or subgoal state.

Where the USEGOOD and USEBAD slots interrogate the context for knowledge, the third slot allows for the addition of knowledge to the context. Called the EFFECT slot, it contains factors that are added to the context whenever the parent frame is invoked. For example, the selection of the wall system with metal studs described above may carry with it the requirement that all electrical cables routed through the studs be encased in rigid conduit so as to prevent rubbing on the knockouts of the metal studs. This condition is transmitted to the context by the inclusion of the factor (CONDUIT.RIGID) in the EFFECT slot. Once added to the context, the result would be to force the program to select rigid conduit encasement for all electrical wires placed within the wall system. This slot is only used when the parent frame is invoked, and its purpose is to add control knowldege to the context for use by the USEGOOD and USEBAD slots at subsequent decision points.

Figure 7 synopsizes the structure of the resulting frame that includes these additional slots.

FRAME:	(VERB. NOUN)
REQUIREMENTS:	(VERB, NOUN (QUANTITY)) (.........., (.........))
USEGOOD:	(NOUN, CONDITION), (................),
USEBAD:	(NOUN, CONDITION), (................),
EFFECT:	(NOUN, CONDITION), (................)

Figure 7.
Generic Frame Structure with Enhancements

In addition to these three slots, the expert system for estimating also incorporates a comprehensive knowledge editor that allows expansion of the component level as well as an ability to generate frames on-line as the requirements of the estimate dictate. We feel this to be of particular importance in this domain since new materials and techniques are constantly being introduced. An inert knowledge base, which is static in terms of both the elements available (components) and the techniques of construction (top level frames), will not long remain a viable tool for the estimator. Instead, the system must possess the capability of continual improvement through interaction with the estimator.

Architecture of the Project Estimating Expert System.

The architecture of this expert system mirrors that of many currently available shells in that the goal state is externally defined to be one of the higher level frames (3). While this will be the normal usage, a goal sate can also be defined at a much lower level than that which would nominally describe a job. For example, specifying the construction of 100 square feet of drywall as the goal state would cause the expert system to stop once this activity had been estimated. This flexibility has the advantage of allowing partial construction or renovation jobs to be estimated.

The context, which is the system's short term memory, contains a description of the variables within the problem domain. At the beginning of an estimating session, the user answers generic questions about the type of structure/facility to be estimated, cost requirements, ambient job site conditions, etc. At each decision point, the inference engine interrogates the context to determine which frame to select by matching the factors contained therein with the control knowledge in the USEGOOD and USEBAD slots of the frames. Additional knowledge is added to the context whenever a frame is selected and the EFFECT slot is fired.

The method of search employed throughout is depth-first. The first value in the REQUIREMENT slot of the goal frame is used as the first subgoal. This requirement then migrates to the next lower level where, upon selection by the inference engine of the suitable frame, the first value in the REQUIREMENT slot of this new frame becomes the active subgoal. This process continues until elements at the COMPONENT level are identified, at which time the second value in the REQUIREMENT slot of the controlling frame becomes active. From this, it can be seen that while only one subgoal is active at any given time, a number of higher level subgoals may be idle pending resolution of the active subgoal at the COMPONENT level.

While this search strategy is straight-forward and does, to a large extent, mimic the nominal problem solving strategy used by most professional estimators, it also forces the solution to early commitment of subgoals. In this sense, the system behaves more like an opportunistic planner than a hierarchical planner in that potential subgoal interactions at higher levels are not reviewed before a frame is invoked (8). At the time that a frame is chosen, the EFFECT slot will add its factor(s) to the context. There exists no safeguard to ensure that the EFFECT factor(s) added to the context may not nullify from consideration a frame that was previously selected. If this occurs, the system must resolve the conflict. One possible technique is to backtrack to the decision point where the previously chosen frame was selected. This frame is then selectively masked from the subsequent search that will begin at the preceding decision point. This method provides for good flexibility in that the system will never be trapped into a solution that was cast in an early decision. Unfortunately, this method also demands that a large amount of overhead be committed to tracking decision points and referencing them back to the factors resident in the context. Further, there are no doubt combinations of factors that would cause the system to backtrack endlessly upon itself, never arriving at the goal state. For these reasons, we have developed a second method of subgoal conflict resolution.

This technique still requires the system to identify the presence of a conflict by matching the USEBAD factors from all the currently chosen frames against the context whenever an EFFECT slot is fired. If a conflict is discovered, the system then masks the most recently chosen frame (i.e.,-the one that caused the conflict), removes its EFFECT factor(s) from the context and proceeds with the search from the preceding decision point. Even though this strategy encourages early-commitment, sensitivity analysis on the solution can still be performed, if desired, by requiring the system to select subgoals in a different sequence.

Summary and Conclusions.

While the development of the expert system described in this paper would require a substantial effort, the database at the COMPONENT level already exists in a number of commercially available configurations. In as much as possible, the expert system should be designed to accommodate these existing structures. The impact of this will be a decrease in the cost and time required to develop the system. In addition, standardization of the knowledge base will allow the job estimating expert system to be integrated into a larger assemblage of programs that deal with and direct a number of construction activities.

References

1. Cohen, Paul R., and Fiegenbaum, Edward A., Eds., The Handbook of Artificial Intelligence, Volume 3, HeurishTech Press, Stanford, 1982, pp. 22-25.
2. Dankel, Douglas D., "Class Notes for Expert Systems (CAP6627)", Department of Computer Science, University of Florida, Gainesville, FL, 1986.
3. Harmon, Paul, and King, David, Expert Systems, John Wiley and Sons, Inc., New York, 1985, pp. 55-57.
4. Levitt, Raymond E., and Kunz, John C., "Using Knowledge of Construction and Project Management for Automated Schedule Updating," Project Management Journal, Vol. XVI, No. K, December, 1985, pp. 57-76.
5. Means Site Work Cost Data, R. S. Means Company, Inc., Kingston, MA, 1986.