Knowledge Acquisition (1991) 3, 421–441

Knowledge extraction using content analysis

RITU AGARWAL
Department of MIS and Decision Sciences, The University of Dayton, Dayton, OH 45469–2130, USA

MOHAN R. TANNIRU
School of Management, Syracuse University, Syracuse, NY 13244–2130, USA

(Received 12 September 1990 and accepted in revised form 5 May 1991)

Expert knowledge elicited by manual knowledge acquisition techniques does not constitute an adequate model of expert behavior until it has been further analysed, decomposed and synthesized. This paper argues that there are two distinct tasks that constitute the knowledge acquisition exercise: information acquisition and knowledge extraction. The extraction of knowledge from documented information (interview transcripts, texts etc.) requires a precise specification of the form of expert knowledge. A model of the decision making process is utilized as a theoretical basis for developing knowledge constructs used in expert decision making and the data reduction methodology of content analysis is applied to perform the task of knowledge extraction. Content analysis exhibits the desirable characteristic of independence from both information acquisition and subsequent knowledge representation. The application of the methodology to a resource allocation domain is demonstrated.

1. Introduction

A term used in the context of developing expert systems, knowledge acquisition has been defined as the process of extracting problem solving knowledge from a knowledge source (usually a human being) with the objective of capturing it in a technical artifact that can reasonably simulate the problem solving behavior (Buchanan *et al.*, 1983). Recently, interest in the notion of knowledge-based systems indicates that the capture and machine replication of specialized knowledge from diverse inanimate sources such as texts, procedure manuals etc. can also serve a useful purpose. Though the benefits obtainable from expert and knowledge-based systems are tangible, the incipient nature of the technology and the inherently experimental approach to expert system development has precluded, to a large extent, the diffusion of this technology into commerical environments. Despite the proliferation of techniques for eliciting expert knowledge, the prime cause of this slow diffusion has been attributed to the knowledge acquisition bottleneck, as the process of acquiring knowledge is a data-intensive, cumbersome and poorly understood task (Wilkins *et al.*, 1984).

Belkin *et al.* (1987) classify knowledge acquisition methodologies into three categories: interviewing the expert, verbal protocol analysis and observational studies. All of these methodologies generate large quantities of data in one form or another. Interviews generate records of conversations between knowledge engineers and experts, verbal protocols produce monologues where an individual introspects

1042-8143/91/040421 + 21$03.00/0

and talks aloud either during task performance or retrospectively, and observation techniques lead to a record of the observer's insights into problem solving behavior. The process of knowledge acquisition does not conclude with this data generation; the data have to sifted, decomposed and synthesized into a coherent description of a domain model and expert reasoning strategies (Motta *et al.*, 1990). For example, a transcript of a verbal protocol does not form the basis for the encoding of expertise, this protocol must be analysed further before it allows a novice to reach expert decisions. Hayward *et al.* (1987) describe this as a "huge conceptual gap between knowledge as expressed by an expert and the encoding of expertise in a software system". It becomes important then to distinguish between two distinct tasks that constitute the knowledge acquisition exercise—*information acquisition* and *knowledge extraction*—which must be performed on the information acquired before it can meaningfully be called knowledge. Our research focuses on this structuring phase of the knowledge acquisition cycle (Figure 1).

The crucial distinction between information and knowledge is important for another reason. The literature abounds with references to problems related to the "representation mismatch" (Buchanan *et al.*, 1983) and emphasizes how critical it is for the knowledge engineer to separate knowledge extraction from knowledge representation. The underlying principle here mirrors the prevalent view in structured systems analysis and design of undertaking and completing analysis prior to any form of logical design (Yourdon & Constantine, 1979). The knowledge extraction phase must thus deliver a product that is in no way dependent on its eventual translation into a computer-based system. It is evident that a necessary characteristic of any knowledge extraction technique must be independence from both the information acquisition phase (to avoid acquisition bias) and the knowledge representation phase (to avoid representation bias).

Some attempts have been reported at developing analytical logics that assist a knowledge engineer in the extraction and modeling of knowledge from information. These include discourse analysis (Belkin *et al.*, 1987), multi-dimensional scaling, cluster analysis (Cooke & McDonald, 1987) etc. Motta *et al.* (1990) provide a current review of analytic knowledge extraction methods. Johnson *et al.* (1987) provide an interesting specification of expertise that is based on an analysis of a verbal protocol generated by an expert as she/he solves a problem. Their method consists of identifying three syntactic categories of behavior from the protocol record and then performing a semantic analysis on these data to generate the specification. This specification is representation-independent. Our research subscribes to the same general principles and our primary interest is in a systematic extraction of knowledge from information, independent of both the acquisition exercise used to generate the information and its eventual structuring into a formal representation scheme.

In our research we have conceptualized the constituents of expert knowledge using a decision making paradigm. This formalization has been used in conjunction with the data reduction methodology of content analysis to extract knowledge from transcripts of interviews with industry experts. The techniques of content analysis are potentially applicable for effectively extracting and modelling knowledge from other types of textual materials. They satisfy the underlying principles of scientific inquiry: those of objectivity and validity, address what has previously been

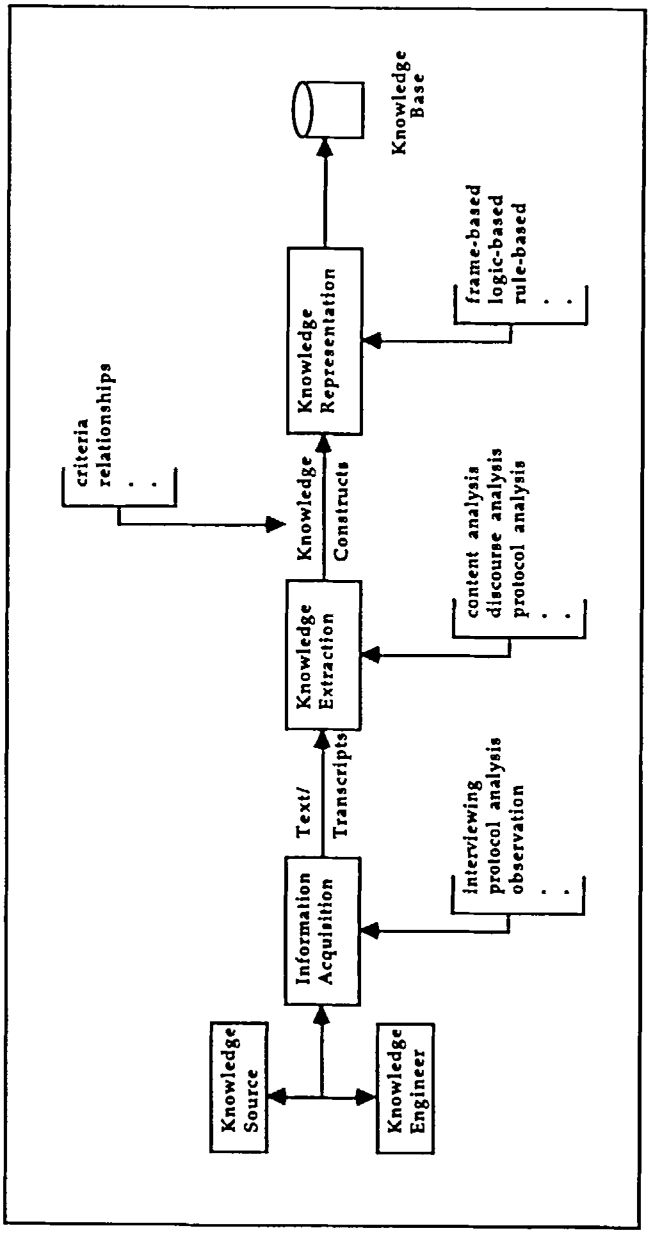

FIGURE 1. Information acquisition and knowledge extraction.

perceived as an intuitive, non-replicable task and produce a set of knowledge that is independent of both the elicitation exercise used to generate it and its eventual representation.

The organization of the remainder of this paper is as follows. The next section examines the decision making process and identifies categories of knowledge constructs. The third section introduces the secondary observation methodology of content analysis that has been applied successfully in a number of social science settings but whose application to knowledge extraction has been limited. The application of content analysis to a specific situation where knowledge was acquired from a number of decision makers in one business domain is illustrated in the fourth section. The final section presents our concluding observations and develops avenues along which this research may be extended.

2. Formalization of decision making knowledge constructs

A prerequisite to the extraction of knowledge from a knowledge source is a specification of the precise form of the objects that constitute knowledge. Such a formalization of knowledge constructs is useful for a variety of reasons. First, it provides the knowledge engineer with a template which he/she can use when analysing descriptions of problem solving behavior, regardless of the manner in which these descriptions were generated. Second, cognisance of knowledge constructs can serve as an input into the process of information acquisition, thus reducing the dichotomy between information and knowledge and shortening the knowledge acquisition cycle. Finally, the use of such a formalization makes the knowledge extraction process independent of its eventual representation, thereby reducing the possibility of a representation mismatch. The specification generated by knowledge extraction can then be further refined and structured to implement the expert system using any representation scheme and/or software development tool (Motta *et al.*, 1990).

A fair amount has been written in the literature about the nature of expert knowledge: that it is a collection of experiences, that it is primarily in the form of heuristics (Hayes-Roth *et al.*, 1983; Feigenbaum, 1979), that it is analogical in nature, that it is judgmental in nature etc. An on-going debate in the AI community and amongst philosophers focuses on the epistemology of knowledge and related concerns. Our interest is more pragmatic and we are concerned with the definition of knowledge from the point of view of encapsulating it in computer software.

Newell (1982) defines knowledge as "whatever can be ascribed to an agent, such that its behavior can be computed according to the principle of rationality". Note, however, that rationality is a subjective, context-sensitive phenomenon that is not amenable to an objective definition. Further, rationality has the connotation of a normative or prescriptive decision making model, and that is not the objective of an expert system which essentially contains a descriptive model. Kornell (1987) evocatively characterizes this as the difference between "formal" and "narrative modes" of thinking. It is important to note, however, that there is an alternate view opposed to the subjective view of knowledge which argues that a certain level of normative reasoning is necessary to avoid problems related to the system providing erroneous advice.

Prior research has *described* expert knowledge in a variety of ways (Johnson *et al.*, 1987; Alexander *et al.*, 1987; Kim & Courtney, 1988). Our work focuses on knowledge constructs at a different level of granularity; and our primary interest is in examining "unstructured" business decision making domains, characterized by a significant amount of heuristic reasoning and the utilization of qualitative, subjective data (PremKumar, 1989).

Accordingly, we begin by analysing popular conceptualizations of the nature of the managerial decision making process in order to isolate discrete steps in this process and to examine the knowledge required to perform each step. Though the literature contains several descriptions of the decision making process (e.g. Simon, 1965; Mason, 1981), we use Archer's (1987) nine-step framework as the starting point. The framework, shown in Figure 2, along with a brief description of each of these steps (Table 1), subsumes a variety of well know decision models described in the literature.

In general, a decision making activity is required whenever the outcome of an activity does not correspond to what was expected or desired (problem identification). When a particular situation has been found to be problematic, an individual generates alternate courses of action to remedy this situation and estimates their impact on organizational operations (problem analysis). Using a set of criteria that are considered appropriate in the decision making environment, the "best" course of action is selected for implementation (problem resolution).

TABLE 1
Basic steps in a decision making progress

1. Monitor (the decision environment): observe internal/external factors that impact an individual's decision making environment for possible deviation.
2. Define (the problem or opportunity): extract the essential details of the problem (where and when did this problem occur, who is responsible, what is its effect on future activities, what are its costs, etc.).
3. Specify (the decision objectives): determine the expectations on the part of the decision maker with regard to the decision outcome, risks involved, and constraints imposed.
4. Diagnose (the problem or opportunity): use deductive and inductive reasoning to uncover the cause/critical factor that created the problem.
5. Develop (alternative solutions/courses of action): using replicative information from past experiences, expert consultation, creativity, and ingenuity, or appropriate tools, generate, without evaluation, various alternatives.
6. Establish (the criteria for appraisal): identify single or multiple criteria, quantitative and/or qualitative models, and objective and subjective information that will have a bearing on performance of an alternative.
7. Appraise (the alternatives): This appraisal may be step-wise (preliminary screening versus detailed evaluation) and iterative (non-fulfillment of appraisal criteria established before may require a redevelopment of alternatives).
8. Choose (the best course of action): select a course of action that "best" meets the criteria established.
9. Implement (the course of action): Using planning, organizing, staffing, directing, and controlling activities, ensure that the decision made is carried out successfully.

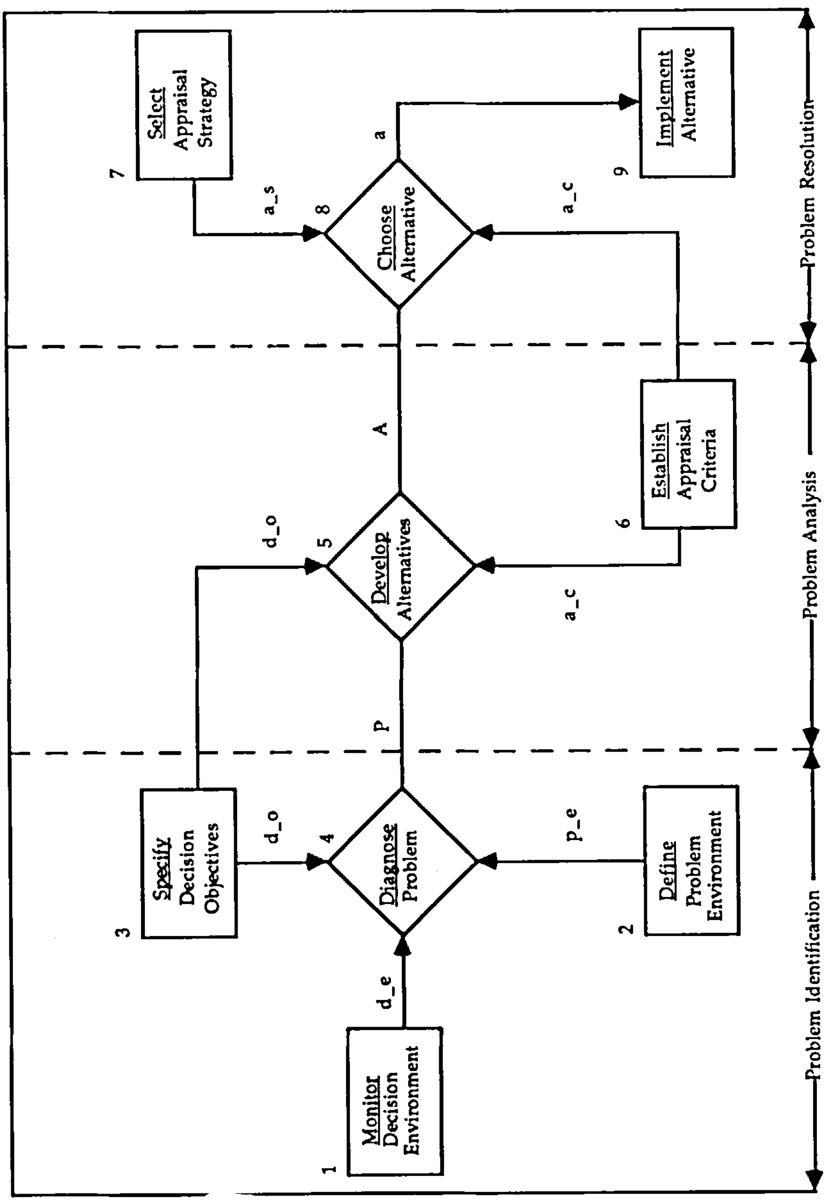

FIGURE 2. A 3-phase, 9-step decision making framework.

Figure 2 maps Archer's nine steps on to the three phases: problem identification, problem analysis and problem resolution. The first three steps identify the decision environment (d_e), problem environment (p_e) and decision objectives (d_o), and lead to a diagnosis of the problem state (P). These four steps comprise the "problem identification" phase. The decision objectives, and appraisal criteria (a_c) are used to generate a set of alternatives for evaluation (A). These two steps are grouped under the "problem analysis" phase. The alternatives are then evaluated using appraisal criteria (a_c) and an appraisal strategy (a_s) to choose a particular alternative (a) for implementation. The last three steps are grouped under the "problem resolution" phase.

In order to make a decision, knowledge about the problem identification criteria (d_e, d_o, and p_e) and their relationships (f_pi) is required. Similarly, knowledge needed to analyse a problem includes identifying problem analysis criteria (P, a_c, and d_o) and their relationships (f_pa). Finally, knowledge needed to resolve a problem includes the identification of problem resolution criteria (A, a_c, and a_s) and their relationships (f_pr). Further, note that not all of these steps may be present in every decision making process. For example, in a resource allocation decision, the alternatives may have been already provided by various business units, i.e. (A) is known. In such a case, the emphasis is on the identification of knowledge (or criterion variables and their relationships) that will allow the manager or groups of managers to select one of these as a part of the problem resolution phase. However, if the objective of the decision making process is to diagnose a defect in a part, then the knowledge set includes criteria that will allow for the formulation and testing of various hypotheses as a part of the problem identification phase.

Complete knowledge of the criteria and their relationships is a characteristic of structured or programmable decisions, while incomplete knowledge about some of these results in a semi-structured decision situation. The extent to which a decision making process is structured or unstructured depends on the subset of criteria and their relationships that is not well defined. For example, if the criteria and their relationships are not well defined in the problem identification phase, but the criteria used for problem analysis and its resolution, along with their relationships, are well defined, then the decision making process exhibits unstructure in the problem identification phase and structure in the other two phases, thus exhibiting semi-structure when viewed as a whole. Semi-structure may also result *within* a single decision making task such as problem analysis where the appraisal criteria and problem state are well defined, but their relationships, i.e. what alternatives are available to address these problems, are not known. Traditionally, decision support systems have been built to support such semi-structured environments where the system models what is known and leaves the rest to the decision maker's judgement, while operations or transaction-based systems have been designed to support structured decisions.

Since the functionality desired of expert systems is for them to draw appropriate (i.e. identical to those of the expert used for their construction) inferences given a set of criteria values, the design of such systems requires that we extract knowledge about all the criteria and their relationships that are germane to the domain being modeled. It is possible that an expert will not be able to provide all the criteria that have an impact on the decision making process, may not be able to identify what values these criteria will take, or how these impact on the decision making process as a whole. This may be partly due to his/her inability to recall past experiences effectively, due to cognitive limitations (Tulving, 1979), or partly because certain scenarios (combinations of criteria values) may have not occurred in the past, thus, rendering it difficult to anticipate the appropriate action. We can extract as much knowledge about criteria and their relationships as the expert can provide in order to replicate his/her performance at that time. In addition, if the knowledge extracted includes situations which are feasible but have no specific action associated with them due to their anticipatory nature, then such knowledge will allow the expert system to have self-knowledge about its limitations as well (addressing the "brittleness" concern—Boose & Gaines, 1988).

We have conceptualized the knowledge that needs to be extracted in the design of business expert systems through the following knowledge constructs: (1) *Criteria*: these criteria can play different roles such as assist in selecting a best alternative (in the problem resolution phase), allow the detection of a problem (in the problem identification phase) etc. In order for the expert system to be complete in its consideration of all possible scenarios, it is critical that the knowledge about the *values* these criteria can and will take is extracted as well; and (2) *Relationships among criteria*: these are used to define how, when and in what form these criterion values are used to make appropriate decisions such as infer that a problem exists, propose a viable alternative or choose a particular alternative for implementation.

Criteria and relationships constitute basic knowledge constructs utilized in decision making behavior. While these constructs appear generic, we have found the conceptualization useful in extracting and recording expertise in a specific business domain captured through structured and unstructured interviews. In addition to these two constructs, an important aspect of modelling expert knowledge is to be able to define a *boundary* around what is known and what is unknown at the time of knowledge elicitation (Woodward, 1990). This notion of bounding the expert's knowledge is captured through a third construct: an equivocal situation. Equivocal situations are scenarios where a specific criterion's role in reasoning and its relationship to *other* criteria is unknown. We explain the concept of an equivocal situation in greater detail subsequently.

Notice that the commonly employed knowledge representation schemes, viz. frame-based, rule-based and logic-based, also require that knowledge be structured into constructs that are *isomorphic* to criteria and relationships. Frame-based representations require knowledge to be conceptualized in the form of objects or classes of objects (Fikes & Kehler, 1985). Associated with these objects are certain attributes and the system employs a definition-by-specialization paradigm for the inheritance of properties among the hierarchical ordering of objects. The hierarchical structure implicitly admits relationships among the domain objects, which themselves constitute a more specialized subclass of criteria, as we define them. Frame-based representations also permit other types of relationships such as associations and causal links which are not captured explicitly through the initial domain model constructed with criteria, relationships and equivocal situations. They can, however, be identified when the initial domain model is refined further. We illustrate one such refinement procedure (categorization) subsequently. Rule-based systems require an explication of the antecedent and consequent parameters, or situation—action pairs (Hayes-Roth, 1985), with the relationship between them being expressed in the form of an inference rule. Again, the analogy to criteria and relationships is evident. Genesereth & Ginzberg (1985) describe logic-based systems as a realization of an application-independent inference procedure. Thus, constructing a logic-based system is tantamount to finding a suitable description of the application area, and descriptions of application areas are developed by identifying domain objects and the relationships satisfied by those objects. A logic program then is simply a set of statements about the objects and their relationships in a suitable formal language, and a prerequisite to writing such a program is the identification of objects and relationships.

The preceding discussion on representation is brief as our intent in this article is to focus on knowledge extraction. In this section we have used the decision making process as a theoretical basis for developing knowledge constructs that model the manner in which individuals reason. It is important to note that the domain model constructed by extracting criteria, relationships and equivocal situations is at an extremely *coarse* level of granularity. Additional refinement such as hierarchically ordering the constructs, categorizing criteria and relationships into classes, establishing causal links etc., is essential to structure the knowledge in a form amenable to machine manipulation. Motta *et al.* characterize a model such as ours as "digested data" where the knowledge is yet to be bound together with a unifying model of the domain and the task. They do, however, recognize the significance of this preliminary data analysis task.

In order to extract knowledge constructs from information documented in written transcripts, we need to establish their formal semantics, i.e. the *meaning and form* of the constructs have to be delineated in an unambiguous manner. The next section provides a brief description of content analysis which can assist in this definition and extraction procedure.

3. An analytical methodology for extracting knowledge

We have applied the techniques of content analysis to the task of knowledge extraction. Content analysis is a data filtering methodology that allows data generated by a communication (written, verbal, visual etc.) to be transformed into a form that is conducive to subsequent analysis. It is essentially a secondary observation method for text and has been been used for a variety of applications in the social sciences. Originally developed as a technique for analysing documentary materials (such as newspapers—Berelson, 1954), content analysis has become more widely applicable to research settings where it is desired to code open-ended questions in surveys, describe trends in communication content etc. The essence of the methodology is best illustrated through the following definitions presented in the literature:

> "Content analysis is any research technique for making inferences by systematically and objectively identifying specific characteristics in text" (Stone *et al.*, 1966)
> "Content analysis is a research technique for making valid and replicable inferences from data to their context" (Krippendorff, 1980)

The central idea reflected is that of a transformation process that allows large volumes of the textual material being analysed to be classified into much fewer content categories. This data reduction process permits the extraction of the constructs that are of interest to the researcher, while maintaining the objectivity and validity of the exercise. The primary components of content analysis are of two kinds: recording units, which may be words, phrases, paragraphs etc., and context units, which is the largest body of context that may be examined in characterizing a recording unit. The context unit may be needed to understand the semantics of a recording unit by referencing the larger context in which it appears. Note that the definition and meaning of recording units is a function of the research questions being addressed through content analysis, and may be different for different problems.

The process of defining recording units is called unitization. Since all subsequent inferences made from the text are dependent on the quality of the unitization, this process is perhaps the most critical component of content analysis, particularly when the units of interest are not well-formed grammatical structures such as individual words or entire sentences, but may occur in any structural form. Unitization requires the development and *iterative refinement* of a unitization scheme that explicitly lays down the rules for identifying a recording unit. The objective of the unitization scheme is to eliminate bias in the identification of recording units, and the scheme permits a shared meaning among multiple individuals who may then apply the rules to the textual materials.

Once recording units (knowledge constructs) have been defined, a coding scheme describing categories of interest is developed and, as before, tested and refined iteratively. A coding scheme simply consists of categories or "pigeon-holes" and rules for placing a recording unit in a category. Categories provide a principled and objective way to obtain a successively finer refinement of domain knowledge. The specific categories to be used in a particular situation depend upon the nature of the domain being examined. Figure 3 provides an overall description of the process of content analysis.

Content analysis is a powerful methodology for knowledge extraction. If implemented rigorously, the results yielded by it are reliable in that they are capable of verification by independent observers (Krippendorf, 1980). The process of achieving reliability in the unitization scheme ensures the existence of a shared meaning between the researcher and the rest of the world. This externalization of semantics is essential for the development of expert systems which are capable of being verified by the experts they purport to model. Such an externalization can also help eliminate *subjectivity* in the interpretation of an expert's utterances, a problem that can become significant if there is a dichotomy in the frames of reference of the knowledge engineer and the expert. Unitization is particularly useful as it facilitates the process of achieving consensus between knowledge engineers and experts on definitional issues, thus allowing for more effective communication.

Our interest was in identifying techniques for knowledge extraction that satisfied the principles of scientific enquiry, and were, at the same time, not too esoteric so as to require specialized knowledge and training to implement. An additional motivation underlying this exercise was the belief that it is important to differentiate clearly between the analysis and design phases of software development, especially

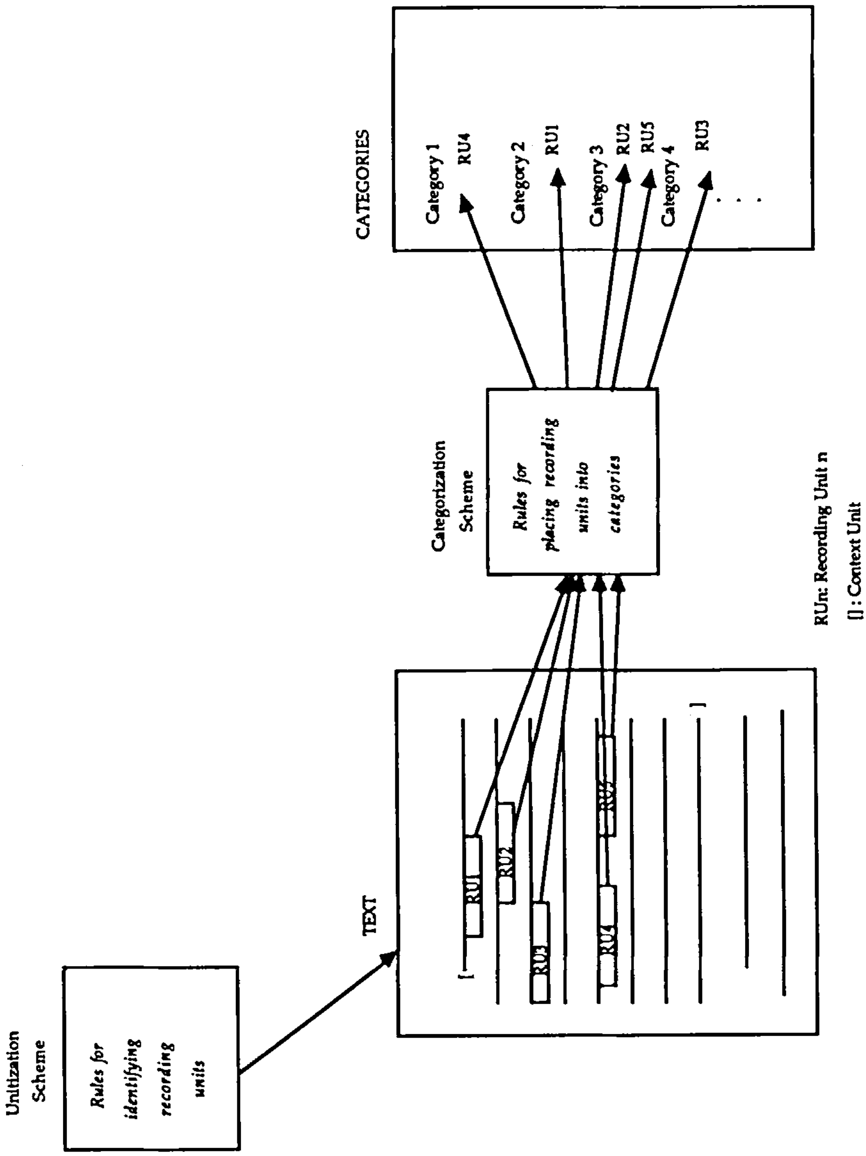

FIGURE 3. The process of content analysis.

when the software is complex, as expert systems tend to be. Content analysis appeared to be one such methodology that could be used without too much additional overhead in terms of knowledge engineer skill requirements. A formal experiment using interviews as the primary information acquisition method was conducted, and the transcripts of the interviews analysed using content analysis to extract the relevant knowledge. The application of the methodology in this specific context is described in the following section.

4. An application of content analysis to knowledge extraction

Thirty decision makers from industry responsible for the resource allocation/capital budgeting decision were interviewed to understand the processes underlying this decision (Agarwal & Tanniru, 1990). Each interview lasted for approximately 1·5 h and generated between 20 and 30 pages of interview transcript. All transcripts were formatted identically, with line numbers assigned to each line to facilitate the referencing of important utterances.

We have developed and tested unitization schemes for the extraction of the knowledge constructs identified in the previous section. Two primary schemes were utilized: one for the identification of criteria and the other for the identification of relationships. A third scheme, for equivocal situations, was considered important from the point of view of coverage of the expert system to be developed subsequently; in order to delineate the limits within which the expert system can make reasonable recommendations.

The unitization schemes are presented in Appendices A–C. Each unitization scheme was developed in an iterative fashion, with multiple coders using the schemes to extract knowledge from a random sample of the transcripts. The results of this extraction were compared, and the schemes revised until a reliability (or inter-judge coding agreement) of over 90% had been achieved. The level of reliability indicates the extent to which the objective of a shared meaning has been attained. Krippendorf's (1980) percentage agreement test was used as an index of reliability. Percentage agreement is computed as a ratio of the intersection of recording units identified by coders and their union. The content analysis literature suggests that theoretically, once an acceptable level of reliability has been obtained, *any* coder should be able to utilize the coding scheme to extract recording units. In practice, we utilized coders with some preliminary domain knowledge, i.e. they all had a rudimentary knowledge of the resource allocation domain. It is not obvious what the results of unitization will be if, in fact, coders are completely unfamiliar with the domain. Once the schemes were defined and tested, two coders extracted knowledge constructs from all the interview transcripts.

Notice that the unitization schemes identify the recording units of interest at two levels: at the *syntactic level*, by describing the form of the grammatical utterance within which the relevant construct can exist, and at the *semantic level*, by specifying the meaning associated with the construct. Such multiple levels of definition further enhance the validity and replicability of the exercise. The definitions of the constructs are generic and perhaps applicable to other unstructured problem domains, though that generalization is not substantiated by our research. The examples used by us in the unitization schemes pertain to the specific decision

00108 Then there would be expenditure decisions that would be made that might
00109 be
00110 for *environmental reasons*. That might be made to comply with *government*
00111 *regulations*. And in a case like that, I would not see it as a case of
00112 whether
00113 or not we are going to do it, if it's *mandated*, we have to do it to be in
00114 compliance with the law. The question is, IF we have to it, are we doing
00115 it in the appropriate way? Are we *minimizing our expense*, are we getting
the *best deal on the hardware* we need to purchase, on the *amount of*
effort
that we are going to expend to do the project?

Criteria identified using the criterion unitization scheme are italicized.

FIGURE 4. Identification of criteria in interview transcripts.

situation being investigated. The construction of such examples for other domains would be a responsibility of the knowledge engineer and would require at least a rudimentary knowledge of the problem domain.

The actual extraction process is demonstrated on fragments of interview transcripts in Figures 4–6. The relevant constructs are italicized in the criteria and rule unitization and the entire paragraph in Figure 5 constitutes an equivocal situation. In this particular decision making scenario, the problem was clearly identified, i.e. the decision makers were required to select among competing investment opportunities, given a pre-determined level of permissible spending. However, the manner in which the problem was analysed by each expert was largely a function of organizational and environmental characteristics, and also dependent, to some extent, on the individual decision maker's style and preferences.

The criterion unitization scheme was developed to extract criteria that were germane to the problem resolution phase, viz. characteristics of investment opportunities, such as costs, benefits, congruence with strategic objectives etc. that the expert perceived as pertinent for classifying them as desirable or undesirable. These criteria were apparent in experts' statements of the form:

"....we first need to know what it's going to cost us.."
"I look at the individual proposing the project, what kind of track record does he have...?"

00382 Q: The bigger you get, I guess the more formal it ends up being. You said
00383 there is a committee now that is actually working on establishing
00384 guidelines, payback periods, that type of thing?
00385 A: They are looking at the entire Major Expenditure Proposal process. One
00386 of the things I hoped would come out of it would be more formalized
00387 guidelines regarding what kinds of return on investments or payback periods
00388 have to be hit for projects to be considered for approval. Because we don't
00389 have those formal guidelines. A lot of it just comes down to plain gut call.
00390 It really is, that's the way it is.

Entire excerpt constitutes an equivocal situation.

FIGURE 5. Identification of equivocal situations in interview transcripts.

A: Earlier I alluded to this ranking process that we have with
ranks from one to six- not a perfect match- but it ended up
that we were only able to fund those projects ranked one,
two, three, and like, half of the fours. So the ones that didn't
make it....
Q: What was the criteria then? What gave something a one
versus a two or three?
A: *One was something that was safety related. Two was*
imminent failure with long term consequences.
Q: What would be a six then?
A: *Something with a positive benefit to cost ratio but*
payback greater than one year. It was either a three or a
four, I'm not sure exactly, if a project had a *positive*
benefit to cost ratio, it was considered a good project and if
it paid itself off in one year then it became a four. So
that ended up being one that would get done. In essence we
did not do the ones- you known they might be good projects,
but they would have a longer term benefit and they did not
make the cut list.

Rules identified using the rule unitization scheme are underlined.

FIGURE 6. Identification of rules in interview transcripts.

"...we are constantly monitoring our competition to see what they are up to..."

The scheme extracted variables that constituted the appraisal criteria (a_c), i.e. goals and objectives against which the analysis criteria were traded-off, manifest in utterances such as "cost reduction", "strategic fit", "mandatory regulations" etc.

Relationships were operationalized through the rule unitization scheme. Though we used an IF–THEN construct to clarify the semantics of a relationship, coders were made aware that relationships would not necessarily appear in the transcripts in that form. Thus, coders were instructed to look for relationships in the *vicinity* of criteria, requiring that criteria be extracted prior to relationships. Vicinity here represents the physical proximity of a relationship to a criterion in the formatted interview transcript. Each relationship was conceptualized as consisting of at least two objects: either a criterion variable and a decision variable or more than one criterion. Relationships were seen to exist at two levels: at the *domain problem solving knowledge level* such as how the decision maker obtained the value of a criterion, how a criterion and its associated value impinged on the final selection of investment opportunities, and at the *meta-knowledge level* such as the ordering or prioritization of criterion consideration in the problem resolution phase. These relationships constituted the appraisal strategy (a_s) and the functional dependencies among criteria (f_pr). The two levels of relationships provide insights into how the knowledge manifest in relationships might eventually be hierarchically structured in a knowledge base.

Criteria and relationships constituted the core of the problem solving behavior exhibited by the experts we interviewed. For the reasons highlighted earlier, the additional construct of equivocal situations was also extracted. At a conceptual level, an equivocal situation was defined as one where it would not be possible to generate more information about the usage of a criterion by going back and querying the expert for a second time. The extraction of these situations is useful not only because it imbues the expert system with self-knowledge, but also because it saves valuable knowledge engineer and expert time in that these lines of reasoning are not pursued in subsequent interactions. Equivocal situations could occur in the transcripts in one of two ways: either the knowledge engineer asked the expert if a particular criterion affects his/her decision and the response was of the form ". . . .not sure what effect it would have. . . " or ". . . .I do not have a procedure for addressing that. . . .", or the expert identified a criterion as being relevant to the resource allocation decision, but not currently used by him/her. Notice the distinction between an uncertain situation (one characterized by incomplete information), where the expert did not explain the usage of a variable adequately, and an equivocal one, where the expert was not aware of its utilization.

The methodology of content analysis for knowledge extraction also serves the purpose of making subsequent interactions with the expert more focused. The products of analysis at each stage, viz. criteria and relationships, form the inputs to the next stage, where the knowledge engineer can determine if the criteria, as articulated by the expert, exhibit any natural classification or categorization. For the problem situation we investigated, we performed one such categorization which classified criteria along qualitative and quantitative dimensions. Appendix D contains the categorization scheme. The categorization procedure helped enrich the model of expert behavior by indicating the portions of the decision making process that were algorithmic in nature and the portions which required the individual's subjectivity and expertise. Categorization is of particular value in complicated domains where a collapsing of criteria into fewer meaningful classes can help reduce the complexity that the knowledge engineer has to contend with.

The analysis of the 30 interview transcripts yielded a rich set of knowledge pertaining to the resource allocation domain. The high degree of inter-judge coding agreement suggests that our unitization schemes were valid descriptions of the knowledge constructs that we sought. On the average, over 70 criteria, six relationships, and one equivocal situation was extracted from each transcript (Agarwal & Tanniru, 1990). The outcome of the analysis, viz., knowledge constructs and knowledge categories, can prove to be useful for two purposes: to serve as the basis for the elicitation of further knowledge, and to provide insights into how the knowledge might be eventually structured in a formal knowledge representation scheme. For example, in the domain we investigated, a large number of relationships did appear in a production-rule format, suggesting that a rule-based representation might be appropriate.

5. Conclusions

In this paper we have presented content analysis as a viable methodology for extracting and modeling expertise from diverse sources of information such as interview transcripts, texts, procedure manuals etc. The advantages of the methodology are exemplified in its objectivity and replicability. Depending upon the domain of interest, appropriate unitization and coding schemes can assist in the extraction of any type of knowledge construct.

The method of content analysis as we have applied it for knowledge extraction does have its limitations. Its major limitation is also one of its strengths, that it presupposes that a precise specification of knowledge constructs (as identified in the unitization schemes) exists. It may be the case that in certain domains it is impossible or extremely difficult to define such knowledge constructs *a priori*. Also, the process of defining and testing unitization schemes is long and time-consuming, thus further extending the knowledge acquisition phase of system development. Finally, what we have presented is clearly the first step in knowledge acquisition—the so called "data digestion" (Motta *et al.*, 1990) phase—which provides an essentially declarative view of the domain. Beyond this step the knowledge engineer is faced with the task of obtaining a more unified and comprehensive model of the problem domain by identifying specific problem solving strategies used by the expert.

One of the benefits of structured analysis in the development of management information systems is its ability to make the analyst use the first set of information (usually in the form of a data flow diagram) to guide further acquisition of relevant information, all the time focusing the effort towards defining a logical view of the system. Similarly, content analysis of the first interview allowed us to extract an *initial* set of criteria and relationships which formed the basis for obtaining additional criteria and relationships. However, choosing the appropriate representation technique needs additional, domain specific information such as the similarity between the way the knowledge is viewed by the expert and the way it will be stored in the system (this is critical if the explanation facility of an expert system is to play a major role in making the system effective), the amount of interaction this system will have with other non-expert (or traditional) systems, and the degree to which the domain requires a handling of uncertainty in the decision making process.

Though automated procedures for implementing content analysis do exist, they tend to be oriented more toward the extraction of structured types of recording units such as specific words or phrases. An interesting knowledge-based application would be to investigate the automation of content analysis in the context of the knowledge constructs presented here. We note, however, that much additional work is required before such automation is feasible.

References

AGARWAL, R. & TANNIRU, M. R. (1990). Knowledge acquisition using structured interviewing: an empirical investigation. *Journal of Management Information Systems*, **7**, 123–140.

ALEXANDER, J. H., FREILING, M. J., SHULMAN, S. J., REHFUSS, S. & MESSICK, S. L. (1987). Ontological analysis: an ongoing experiment. *International Journal of Man–Machine Studies*, **26**, 473–485.

ARCHER, E. R. (1987). How to make a business decision: an analysis of theory and practice. In S. WITHANE, Ed. *Organization Design: Theory and Applications*. Littleton, MA: Copling Publishing.

BELKIN, N. J., BROOKS, H. M. & DANIELS, P. J. (1987). Knowledge engineering using discourse analysis. *International Journal of Man–Machine Studies*, **27**, 127–144.

BERELSON, B. (1954). Content analysis. In G. LINDZEY, Ed. *Handbook of Social Psychology*, Vol. 1, pp. 488–522. Cambridge, MA: Addison-Wesley.

BOOSE, J. H. & GAINES, B. R. (1988). Knowledge acquisition for knowledge-based systems. *Tutorial, AAAI 88*.

BUCHANAN, B. G., BARSTOW, D., BECHTEL, R., BENNET, J., CLANCEY, W., KULIKOWSKI, C., MITCHELL, T. & WATERMAN D. A. (1983). Constructing an expert system. In F. HAYES-ROTH, D. A. WATERMAN & D. B. LENAT, Eds. *Building Expert Systems*. Reading, MA: Addison-Wesley.

COOKE, N. M. & MCDONALD J. E. (1987). The application of psychological scaling techniques to knowledge elicitation for knowledge based systems. *International Journal of Man–Machine Studies*, **26**, 533–550.

FEIGENBAUM, E. A. (1979). Themes and case studies in knowledge engineering. In D. MICHIE, Ed. *Expert Systems in the Micro-Electronic Age*. Edinburgh University Press.

FIKES, R. & KEHLER, T. (1985). The role of frame-based representation in reasoning. *Communications of the ACM*, **28**, 904–920.

GENESERETH, M. R. & GINSBERG, M. L. (1985). Logic programming. *Communications of the ACM*, **28**, 933–941.

HAYES-ROTH, F. (1985). Rule-based systems. *Communications of the ACM*, **28**, 921–923.

HAYES-ROTH F., WATERMAN D. A. & LENAT, D. B. (1983). An overview of expert systems. In F. HAYES-ROTH, D. A. WATERMAN & D. B. LENAT, Eds. *Building Expert Systems*. Reading; MA: Addison-Wesley.

HAYWARD, S. A., WIELINGA, B. J. & BREUKER, J. A. (1987). Structured analysis of knowledge. *International Journal of Man–Machine Studies*, **26**, 487–498.

HOFFMAN, R. R. (1987). The problem of extracting the knowledge of experts from the perspective of experimental psychology. *The AI Magazine*, **8**, 53–67.

JOHNSON, P. E., ZUALKERMAN, I. & GARBER, S. (1987). Specification of expertise. *International Journal of Man–Machine Studies*, **26**, 161–181.

KIM, J. & COURTNEY, J. F. (1988). A survey of knowledge acquisition techniques and their relevance to managerial problem domains. *Decision Support Systems*, **4**, 269–284.

KORNELL, J. (1987). Formal thought and narrative thought in knowledge acquisition. *International Journal of Man–Machine Studies*, **26**, 203–212.

KRIPPENDORFF, K. (1980). *Content Analysis: An Introduction to its Methodology*. Beverly Hills, CA: Sage.

MASON, R. O. (1981). Basic concepts for designing management information systems. In R. MASON & E. B. SWANSON, Eds. *Measurement for Management Decision*, pp. 81–95, Reading, MA: Addison-Wesley.

MOTTA, E., RAJAN, T. & EISENSTADT, M. (1990). Knowledge acquisition as a process of model refinement. *Knowledge Acquisition*, **2**, 21–49.

NEWELL, A. (1982). The knowledge level. *Artificial Intelligence*, **18**, 87–127.

PREMKUMAR, G. (1989). A cognitive study of the decision-making process in a business context: implications for design of expert systems. *International Journal of Man–Machine Studies*, **31**, 557–572.

SIMON, H. A. (1965). *The Shape of Automation for Men and Management*. New York: Harper and Row.

STONE, P. J., DUNPHY, D. C., SMITH, M. S. & OGILVIE, D. M. (1966). *The General Enquirer: A Computer Approach to Content Analysis*. Cambridge, MA: MIT Press.

TULVING, E. (1979). Relation between encoding specificity and levels of processing. In L. S. CERMAK & F. I. M. CRAIK, Eds. *Levels of Processing in Human Memory*. Hillsdale, NJ: Lawrence Erlbaum.

WILKINS, D. C., BUCHANAN, B. G. & CLANCEY, W. J. (1984). Inferring an expert's reasoning process by watching. *Proceedings of the 1984 Conference on Intelligent Systems and Machines*.

WOODWARD, B. (1990). Knowledge acquisition at the front end: defining the domain. *Knowledge Acquisition*, **2**, 73–94.

YOURDON, E. & CONSTANTINE, L. L. (1979). *Structured Design*. Englewood Cliffs, NJ: Prentice-Hall.

Appendix A: Unitization scheme: criteria

The respondent is describing the resource allocation/ capital investment process as it is executed in his/her organization. The following scheme will help extract CRITERIA or factors that the respondent has articulated as being germane to this decision making process.

• A CRITERION is something that the respondent uses as a justification or rationale for a certain kind of behavior. It is used to guide the decision making process and is a determinant of the choices that are made. The criterion provides direction and resolution to the decision process.

e.g.
"One of our primary objectives it to reduce cost and therefore any investment that contributes to achieving that objective will naturally be attractive to us".

In the above statement, the criterion is "amount of cost reduction" and it is used to implicitly prioritize investments.

• A CRITERION is a noun which may have one or more adjectives qualifying it.

e.g.
"Anything that the unique requirements of a contract require us to do we must do".

The criterion here is "requirements" which is qualified by "unique" and "of a contract".

• A CRITERION is any piece of information that is used when making the resource allocation decision.

e.g.
"Well, we try to see what our competition has done in this area. . ."

Here the respondent has clearly identified that he/she will obtain information on the actions of the organization's competitors and use that information when making decisions.

• Any goals or objectives identified by the respondent also constitute CRITERIA since decisions for allocating resources are made on the basis of the extent to which a particular investment or allocation contributes to that goal achievement.

e.g. "We want to be market leaders in this particular line of products. . . ."

In the above example, the objective and the criterion is "market leader".

• A respondent may reference the same CRITERION a number of times during the interview. The line numbers of all references should be included when unitizing the criterion.

• For something to be coded as a CRITERION, it is not necessary for the respondent to specify exactly *how* it is used, only that it is a driver of the decision in some way.

Appendix B: Unitization scheme: rules

This scheme will help extract decision rules from the knowledge acquisition interviews. Note that criterion unitization must be done prior to rule unitization.

• RULES guide the decision making process by identifying appropriate actions under different sets of circumstances.

• A RULE is a speech construct that consists of two parts: the antecedent and the consequent.

• The antecedent, or "IF" part of the RULE specifies a certain state of the world. The consequent, or "THEN" part specifies the appropriate action to be taken or inference to be made if the antecedent state of the world is true.

> e.g.
> IF the economy is in an inflationary state
> THEN adjust discount factor used to calculate rate of return

• The antecedent of the RULE contains a CRITERION that the respondent uses as a rationale for behavior. Thus, every criterion has one or more rules associated with it. The number of rules associated with a criterion depends on the number of LEVELS of the criterion that the respondent identifies.

> e.g.
> IF investment amount is less than $25,000
> THEN investment can be approved locally
> IF investment amount is greater than $25,000
> THEN approval must be obtained from the chairman

The criterion in the above example is "investment amount" and the two levels are less than $25,000 and greater than $25,000.

• A RULE must clearly define *how* a criterion is used in the decision making process. The respondent must identify various objectives and goals that he/she would like to accomplish, but these would not constitute rules unless a specific level of the objective was clearly identified.

> e.g.
> "One of our objectives is to reduce cost and increase capacity utilization".

The above statement does not constitute a decision rule because the respondent has not identified how the criteria of cost and capacity utilization are used in making a decision.

• A RULE may occur at a meta-level in the sense that it may specify the *order* in which criteria have to be considered.

> e.g.
> "If the investment is in equipment, we would consider the payback period first and then the rate of return".

In the above example, "payback period" and "rate of return" are both criteria and the rule identifies the order in which these criteria are to be applied.

• A RULE may specify how the respondent obtains the *value* of a criterion.

> e.g.
> "Amount of effort required for a project is calculated by assigning a dollar value of $32 for analyst time and $24 for programmer time".

Here the criterion is "amount of effort" and the respondent has clearly identified how it is to be calculated.

• RULES will not appear in the transcript explicitly in the form of "IF–THEN" statements. They must be deduced from the statements of the respondent in the vicinity of a criterion.

Appendix C: Unitization scheme: equivocal situations

This scheme is to be used to extract EQUIVOCAL SITUATIONS from the knowledge acquisition interviews. These situations can be identified in one of the following ways:

• The interviewer asks the respondent if a particular factor or criterion is considered by him/her while making decisions and the respondent says:

"We don't use it here
I don't know how this factor will affect my decisions"

If the respondent clearly says that the factor *does not* affect any decision, then it is not an equivocal situation.

• The respondent mentions a factor that he/she knows as affecting the resource allocation decision, but that factor is not being used as a criterion in his/her particular situation.

Appendix D: Categorization scheme: criteria

This categorization scheme is to be appplied to the CRITERIA extracted from the knowledge acquisition interviews. Each criterion must be classified into one of the two categories defined below.

Category: *Qualitative criteria*
• A qualitative criterion is one which has inherently subjective assumptions built into its *estimation* and *utilization*.

e.g
user satisfaction
market needs
productivity of employees
vendor support
strategic requirements
risk

• All criteria relating to opinions, preferences, perceptions, and utilities derived by respondent are qualitative in nature.

e.g.
"Sometimes we feel a certain project would be just *nice to have*".
"We *want to be leaders* in this particular business".
"Before I commit any resource I ask myself 'is this *where I want the company to go?*' "
"Often simply the *way in which a project is presented* may have a lot to do with its acceptance".

• A qualitative criterion in one where the possibility of two individuals perceiving the same phenomenon and deriving two different conclusions exists.

e.g.
commitment of users
philosophies set by the top

• Any criterion which involves subjectivity in the assessment of its *effect* is qualitative.

e.g.
impact of promotion campaign on image of company
impact of investment on strategic goals

• The *nature* of the organization, if it is service rather than product oriented, is a qualitative criterion.

e.g. service organization

Category: *Quantitative criteria*

• A quantitative criterion is one which is tangible and observable, or measurable.

e.g.
rate of return
capacity utilization
labor requirements

• A quantitative criterion may be attached to some physical entity.

e.g.
capability of vendor's product

• There exist objective procedures to measure the value of a quantitative criterion.

e.g.
number of man years to complete the project
equipment requirements

• All criteria that relate to activities that are mandated are quantitative.

e.g.
regulatory requirements