Knowledge Acquisition Using Multiple Domain Experts in the Design and Development of an Expert System for Disaster Recovery Planning

by

Frank W. Nasuti

Preliminary proposal for dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy

School of Computer and Information Sciences
Nova Southeastern University

2000 (Revised)

Problem Statement and Goal

Disaster Recovery Planning (DRP)[1] and Business Resumption Services (BRS)[2] are both time consuming and costly processes. These, as well as the feeling that "we have never had a problem," have made companies hesitate in implementing plans to resume business activities to prepare for a catastrophic event. However, several factors have caused companies to change their attitude towards their need for DRP and BRS. Disasters such as earthquakes, tornadoes and hurricanes have awakened management to the realization that it could happen to them (Cerullo & McDuffie, 1994; Hiles, 1992; Garcia-Molina & Polyzois, 1990). The increasing dependence of organizations on data processing to perform the basic functions of corporate America, the fear of lawsuits by shareholders for their failure to prepare, recent accounting regulations that address an entity's ability to survive as an "ongoing" entity (AICPA, 1988) in event of a disaster have caused management to address the need to develop DRP and BRS, and federal regulations that call for the existence or development of DRP and BRS plans (Garcia-Molina & Polyzois, 1990).

Business opportunities in both the consulting and services areas for DRP and BRS have increased tremendously (Jacobs & Weiner, 1997; Rudolph, 1990). However, several factors are hindering the ability to provide the services needed by companies. These include:

---

[1] Disaster Recovery Planning is a term that is used to describe activities that cover computer/data-processing activities.

[2] Business Resumption Services is a term that is used to describe all critical business functions including computer/data processing, telecommunications, support functions such as accounting (billing/accounts receivable, payroll, etc.), etc.

1. The lack of experienced consultants needed to evaluate organizational needs for DRP and BRS, or to evaluate and review existing plans;
2. The need for multiple skills ranging from auditing, assessment, communications, etc.; and
3. The need for expertise in specific areas including hardware, software, telecommunications, general business and industry specific concerns.

Although consultants and vendors specializing in disaster recovery planning are available, the number is limited and the quality of their services is questionable in many cases. The information gathering process between the consultants and the client is a time consuming process and in most cases requires the use of multiple vendor experts, as well as various resources within the customer's organization.

This research proposes, as a solution to address these deficiencies, the design and development of an expert system to assist in the determination of the needs of an organization for disaster recovery and business interruption services, as well as the evaluation of existing plans.

This research will attempt to design an expert system to develop a disaster recovery plan. It will include the knowledge acquisition processes necessary to elicit information from multiple domain experts. The specific goals of this research include:

1. Knowledge acquisition specific to the problems of using multiple domain experts;
2. Design and development of a prototype expert system for disaster recovery planning; and
3. Validation of the prototype expert system.

Relevance, Significance and Brief Review of Literature

Disaster Recovery Planning

Disaster Recovery Planning and Business Resumption Services are critical in all organizations. However, the ability to resume the minimum functions and operations necessary to ensure continuing operations after a disaster, becomes questionable in all but the smallest entities. Research has shown that over 70% of organizations that experience a catastrophic event close within two years (Andrews, 1994).

Recent events have demonstrated the need for disaster recovery capabilities by all organizations relying on data processing capabilities. Disaster, as it relates to data processing, is defined as an event which causes the loss of computing service, or of a significant part of it, or of systems, communications or applications, for a length of time which prevents the impacted organization from achieving its mission or which imperils the business (Hiles, 1992). Disasters such as Hurricane Andrew, the World Trade Center bombing, the Northridge Earthquake, and Hurricane Hugo (Cerullo & McDuffie, 1994; Griswold, Lightle, & Lovelady, 1990) left many companies in California and Florida helpless. More recent disasters such as Hurricane Floyd and the El Nino weather incidents have made many companies realize that both DRP and BRS are necessary. This concern is critical in organizations where sizeable computer processing is present, whether as a service or support function. Added to this are various federal regulations that require key organizations such as the banking industry to implement and test a disaster recovery plan (Garcia-Molina & Polyzois, 1990). The result of this is the need for the implementation of disaster recovery planning for data processing operations, as well as the extension of this plan to other critical business functions and operations (Cerullo & McDuffie, 1994). Evidence strongly supports the concept that companies can survive a disaster if they: (1) plan for the possibility of a disaster; (2) formulate strategies for recovering critical business functions; (3) implement technology to aid the recovery of automated functions and systems; and (4) train employees to implement the strategies (Rudolph, 1990).

Knowledge Acquisition

In designing expert systems, the process of eliciting information has been termed knowledge acquisition. According to Hoffman (1987), knowledge acquisition (elicitation) involves extracting problem-solving expertise from knowledge sources, which are usually domain experts. Waterman (1985) defines knowledge acquisition as the process of extracting, structuring and organizing knowledge from several sources, usually human domain experts, so it can be used in a program. This knowledge acquisition process involves one or more knowledge engineers interacting with one on more domain experts, each of which brings a certain set of attributes to this interaction with the goal of developing a shared representation or model, of the expert's problem solving processes.

The acquisition of knowledge is a major and critical phase in the development of an expert system. Knowledge acquisition is considered by many to be the most difficult and precarious stage in the knowledge engineering process. Rook & Croghan (1989) and Hwang (1994), amongst others, stated that knowledge acquisition has often been described as the bottleneck in knowledge based systems development.

Byrd, Cossick & Zmuk (1992) noted that knowledge acquisition is nothing but an extension of requirements analysis, which involves end users and systems analysts interacting in an effort to recognize and specify the data and information needed to develop an information system. Their research showed that many of the entities and processes in knowledge acquisition and requirements analysis are almost identical. Although most systems analysts do receive training in requirements analysis, little or no training is available at the college or professional level in knowledge acquisition (Kemp & Purvis, 1996).

O'Neil (1989) attempted through surveys to demonstrate why the vast majority of expert systems fail. Some of the reasons noted by him and others include:

1. The lack of user participation in design (Rees, 1993);
2. The lack of structure and organization of knowledge acquisition (McGraw & Harbison-Briggs, 1989);
3. Communication problems between the knowledge engineer and the domain expert (O'Neil, 1989; Jeng, Lieng & Hong, 1996);
4. Failure in identifying the right candidates for knowledge acquisition (Stein, 1993);
5. Failure of verification and validation.

The knowledge engineer may have little knowledge about the domain and may not understand the jargon used by the domain expert (Jeng, Lieng & Hong, 1996).

Rook and Croghan (1989) suggest a systems engineering conceptual framework to address the knowledge acquisition bottleneck in the knowledge based system development rather than focusing on specific knowledge acquisition methods and techniques. They noted that what is critical to overcoming the difficulties of knowledge acquisition is a framework that "structures" knowledge acquisition steps within the context of the expert system development phase. Specifically, they outlined a structured knowledge acquisition process in order to: (1) specify the goals of knowledge acquisition as they relate to the expert system development cycle phases; (2) identify the specific steps or tasks involved in each phase; (3) specify the specific goals of each knowledge acquisition step; (4) identify constraints or barriers to successful knowledge acquisition activities; and (5) provide a knowledge acquisition task structure that will facilitate the selection of appropriate knowledge acquisition methods.

Knowledge Acquisition Techniques

Key to the success of the design and development of the DRP expert system is the choice of the correct or most appropriate technique for knowledge acquisition. As noted by Niederman (1996), much research is available discussing the various techniques. Several knowledge acquisition techniques include structured and unstructured interviews, "constrained-processing tasks" including both simulation of familiar tasks and the use of scenarios, and the use of "tough cases". Byrd, Cossick and Zmud's (1992) list includes observation of the expert in action, unstructured elicitation which corresponds to unstructured interviews, mapping domain knowledge as a combined effort between the knowledge engineer and expert to develop a cognitive map, formal analysis of the domain, and structured elicitation corresponding to structured interviewing. Byrd, Cossnick & Zmud (1992) agree that it be best to use the above techniques in some combination, depending on what works in a particular environment (Niederman, 1996).

Choices include interviewing, observations and protocol analysis (Wolfgram, Dear & Galbraith, 1987), among other techniques. This paper will evaluate these and other techniques and attempt to utilize the most appropriate technique.

Multiple Domain Experts

The key parties in expert systems design and knowledge acquisition are the knowledge engineer and the domain expert(s). A domain expert is defined as an articulate, knowledgeable person with a reputation for producing good solutions to problems in a particular field (Waterman, 1985). The selection of domain experts may depend on several criteria. As the definition states, selection ideally should be based on expertise, experience or reputation of the individual(s). In addition, the individual(s) should possess business and personal attributes, which lend themselves to the knowledge acquisition process. Thirdly, the individual(s) must be available during both the knowledge acquisition process and expert system development process.

Most early expert systems were developed using a single domain expert (O'Neil & Morris, 1989). Few involved multiple experts and the problems of knowledge acquisition that occur with the use of multiple domain experts. They noted that the main reason for this seemed to be that it was easier both to elicit the knowledge and to avoid contentious issues and conflicting opinions. However, knowledge engineers soon realized, in many cases, that expertise is not resident in the knowledge of a single domain expert. Multiple experts can provide the mix of knowledge that is required in a complex structure, such as disaster recovery planning, and provide coverage for the many problems and solutions (Money & Harrald, 1995).

The problems of using multiple domain experts include the issues of conflict between the experts, and the failure of the knowledge engineer to express the relationship between multiple views in requirements specifications which may overlap, complement and contradict each other (Nuseibeh, Kramer & Finkelstein, 1994). Hwang (1994) noted that one of the most difficult problems of knowledge acquisition is to integrate domain knowledge from multiple experts, especially when inconsistencies and conflicts need to be resolved. Hwang also noted that as the experts of most application domains seem to be very busy, it is impossible to get them together. Added to this is the often different physical location of the domain experts and travel costs to get them all in one place.

O'Neil & Morris (1989) noted ways to avoid conflict and dissent. They include (1) asking the experts to provide documentary back-up evidence; (2) by the use of probabilities and uncertainty factors to express degrees of agreement or disagreement amongst the experts; and (3) by creating systems modularly so that different experts could be specifically called on for particular sections and consultations.

In attempting to integrate multiple requirements specifications, overlaps must be identified and expressed, complementary participants made to interact and cooperate and contradictions resolved (Nuseibeh, Kramer & Findelstein, 1994).

Flowcharts/Decision Trees/Decision Tables

Three key tools in the design of expert systems utilizing rule-based systems are flowcharts, decision trees and decision tables. These tools were prominent in the early years of expert system development (O'Neil & Morris, 1989) and continued to be used for fuzzy logic applications in the nineties (Chen, 1993).

Flowcharts can be used to represent the flow of the logic including the order of asking for data needed during the expert system's consultation process and the branching of the flow based on the data received or calculated by the application. Graphic based flowchart-editing tools such as Flowtool (Watkins, Dimopoulos, Neville, & Li, 1993), Flowchart, etc allow the knowledge engineer to quickly create and edit the flowcharts.

Decision trees and decision tables will be used to represent the decision-making process based on all of the data inputted or calculated. Decision tables are easily updated and converted into rules. Both of these tools can be used for design purposes and can be further utilized as verification and validation tools.

Expert System Shells

During the early years of expert system development, most expert systems were developed using one of several programming languages specifically developed for such use, such as Prolog and LISP. It was soon realized that most expert systems utilized the same logic and that the key differences were the knowledge and data required by the inference engine. By stripping away the knowledge and data from existing expert systems, the resulting "shell" could be used to develop new expert systems. Expert system shells are software packages containing a generic inference engine, a user interface, and a collection of other tools that enable users to develop and use expert systems. Using the shell's tools, a knowledge engineer can develop new knowledge bases and can structure, all, delete, and modify the knowledge contained in them (Stylionou, Smith, & Madey, 1995).

Expert system shells were available during the late eighties and early nineties. Most were developed under the DOS environment and those that survived were modified for use under Windows. However, the loss of interest in these shells left many good ones, such as VP-Expert, dropped by the developers or failed to be supported.

Which expert system shell to use is a key decision to be made by the knowledge engineer. Using the wrong shell can result in project failure. There is agreement in the literature that different application types require different expert system shell capabilities. In their research, Stylionou, Smith & Madey (1993) defined taxonomy of expert system application types with similar capability requirements into categories with each category called a generic application category. In subsequent research Stylionou et al (1995) noted that the selection of a shell depends on factors such as application and project characteristics, capabilities of the shell, user sophistication, ease of integration with existing software and hardware, and vendor support. Although backward chaining has been viewed as the most efficient method, expert systems for design applications should utilize forward-chaining since its desired goals are not known in advance or are too numerous to list and is most useful in creating data driven processes and expert systems with unknown goals (Song, Strum & Medsker, 1991).

Verification and Validation Process

Many expert systems in the past have failed to include any verification and validation phase and as a result have failed. Similar to the testing phase of conventional structured application design, the verification and validation process are used to ensure that the system does what it is supposed to do. The verification and validation process must include all of the players involved, including the domain experts, the knowledge engineer and the users.

Although verification and validation are related, they are different processes and often confused with each other. Validation refers to building the system right; that is, substantiating that the system performs with an acceptable level of accuracy; whereas, verification refers to building the system "right"; that is, substantiating that a system correctly implements its specifications (O'Keefe, Balci & Smith, 1990). Whereas, verification tests for the consistency and completeness of the expert system, validation analyzes the knowledge base and decision-making capabilities of the expert system for content validity, the level of expertise and the reliability of the expert system. O'Keefe, Balci & Smith (1990) noted the major problems in validating an expert system's performance include: (1) what to validate; (2) what to validate against; (3) what to validate with; (4) when to validate; (5) how to control the cost of validation; (6) how to control the bias; and (7) how to cope with multiple results. They noted that the alternatives of "what to validate" include intermediate results, or the final results or conclusions, or the reasoning of the logic, or any combination of the three. The choices for "what to validate against" include known results or expected performance. As noted by O'Leary (1990), validation is important to the decision-making success of an expert system and to the continued use of the expert system. An expert system that does not make the correct decisions will lead to a loss of confidence in it and non-use. Therefore, expert systems must be validated prior to implementation in the field for real world use. O'Leary also noted that validation requires: (1) ascertaining what the system knows, does not know, or knows incorrectly; (2) ascertaining the level of expertise of the system; (3)

determining if the system is based on a theory for decision-making in the particular domain; and (4) determining the reliability of the system. Nguyen, Perkins, Laffey, & Pecora (1990) noted particular problems in the knowledge base when checking for consistency and completeness. They listed five problems for consistency including redundant rules, conflicting rules, subsumed rules, unnecessary IF conditions, and circular rules. In their research they indicated that any one of four situations can be indicative of gaps or missing rules in the knowledge base, including unreferenced attribute values, dead-end goals, unreachable conclusions and dead-end IF conditions.

Tools utilized in the verification and validation process include those specific to expert systems, as well as some of the tools used in conventional application design. These include the explanation facility of expert system shells, flowcharts, decision trees and decision tables. The later has been used in a vast majority of cases to provide for extensive verification and validation assistance. It easily allows the knowledge engineer to check for contradictions, inconsistencies, incompleteness, redundancy, etc in the problem specification (Vanthienen & Robben, 1993; Nguyen et al, 1990).

Barriers and Issues

Contingency planning service is the ability to provide a multi consultant approach to the customer's needs for various business resumption services in the event of a disaster. This concept requires the consulting services of several domain experts including, at a minimum, a DRP/BRS consultant, a telecommunications consultant, a recovery site expert, and an industry expert. This multi consultant/expert approach presents several problems, including:

1. The need for each consultant/expert to meet with each customer, to evaluate the customer's environment and needs;
2. Consistency of the consultant/expert at each meeting;
3. The availability of all consultants/experts at the same time or when needed;
4. The need to meet with several different customer personnel; and
5. The need for speedy turnaround time from meetings/consultations to providing the written proposal to the customer.

Some of problems in completing a disaster recovery plan relate to the difficulty of gathering the essential information to be detailed in the body of the plan (Jacobs & Weiner, 1997). This is complicated by the lack of identified or available experts (which is viewed in the industry as a critical lack of qualified personnel), the available expert/consultant being viewed as costly, or the expert/consultant not being readily available at a particular location. Other barriers include: (1) the problems inherent in knowledge acquisition, and (2) the complications of this difficult process when using multiple domain experts.

Approach

This expert system design and development will utilize a phased approach:

Phase 1:

This initial phase will entail the knowledge acquisition process from the 3 domain experts identified in the resources section of this proposal, to elicit the information needed (requirements specifications) by the various disaster recovery vendors to determine the data processing environment of the customer. This knowledge acquisition process will be based on the best or most appropriate method or methods of knowledge acquisition, as identified in a review of the literature in knowledge acquisition and requirements analysis. Emphasis will be made on interviewing techniques, review of examples, and case studies.

Phase 2:

This phase will include the design of the decision trees and decision tables needed by the expert system to determine the appropriate disaster recovery plan services for the customer. Flowcharting tools will be used to document the flow and branching of questions during the expert system consultation. Both forward and backward chaining will be considered.

Phase 3:

During this phase the prototype expert system will be developed. Initial design plans call for the use of VP Expert, an expert system shell development tool with the final version compiled for use as a standalone application. However, since it is recognized that VP Expert is a DOS based tool, other shells may be evaluated to determine if a tool is available for use on Windows based systems.

Phase 4:

During this final phase, the prototype expert system developed in phase 3 will be validated. Attempts will be made to validate the decision-making capability of the expert systems by comparing its conclusions with those of domain experts during both real-life walkthroughs and/or simulations and/or case studies.

The specific steps will include:

1. Research in the area of knowledge acquisition, including knowledge acquisition techniques, knowledge acquisition methodology, and knowledge acquisition validation and verification methods. Specifically, the literature will be reviewed for general information on knowledge acquisition, including problems in the elicitation process, theories, comparisons with requirements analysis, elicitation methods, and case studies of real-life situations and simulations.

   Knowledge acquisition techniques will be identified, including the advantages and disadvantages of each. The use of these techniques, singularly or in combination, will be evaluated to determine the best or most appropriate method or methods to use in the prototype.

   Knowledge acquisition methodologies will be identified and evaluated. A model will be developed to be used in the design and development of the prototype expert system.

   Knowledge acquisition validation and verification methods will be identified and evaluated to determine which methods will be used in the validation and verification of the prototype. Methods for use during design, as well as the final prototype, will be researched.

2. Research in the area of multiple domain experts and identification of associated problems. Although many existing expert systems have been developed using a single domain expert, the proposed expert system will utilize multiple domain experts. It has been noted that the use of multiple single domain experts has presented more complications than experienced with single experts. The problems associated with multiple domain experts will be identified and measures taken to avoid or minimize their affects during the application development.

3. Research in the area of disaster recovery planning. Specifically, analysis of past disasters will address the need for the expert system. Case studies will be reviewed to insure the accuracy of the disaster recovery plan recommended during the consultation process of the expert system.

4. Research in expert systems in general, including development methodology and tools. Specifically, identification of development methodologies and models recommended in the literature to determine what expert system methodology will be used in the design of the prototype.

5. Identification and selection of domain experts. Key to the success of the expert system will be the accuracy, content, completeness, etc. of the knowledge acquisition process. Such success will depend on the knowledge

engineer's ability to identify all available domain experts and ensure that those selected have the knowledge, etc. needed to design a working prototype.

6. Knowledge acquisition using domain experts. During this step, the knowledge acquisition process will take place as outlined in Table 1.

7. Design of the expert system for disaster recovery planning. During this step, the expert system will be developed using the expert system shell development tool.

8. Validation and verification of the knowledge acquisition process and the expert system prototype. During this step, the prototype expert system will be validated and verified in several ways as noted in Table 2. At the completion of the expert system development, attempts will be made to validate the results of the conclusions of the expert system against real conditions. Alternatives will include: (1) validating against domain experts in real-life situations; (2) validating against simulations; and (3) validating against case studies.

Milestones

The literature will be reviewed to determine the appropriate expert system / knowledge based system design methodology. The milestones and dates to be completed are noted in Table 3.

Resources

Three domain experts and two software tools will be used to design and document the expert system. Descriptions of these resources are described below.

Domain Experts

Expert one is a CPA and a Senior Managing Director for Navigator Associates and a former Director in the Information Technology Group of Coopers & Lybrand. As an EDP auditor and consultant he had extensive experience in the auditing of contingency planning and disaster recovery plans and is presently involved in designing crisis management planning for major organizations. His former clients included Fortune 500 companies, including Bell Atlantic, Conrail and Booze-Allen-Hamilton. He will serve as the domain expert in the hardware, software, maintenance and recovery team areas.

Expert two is presently an independent consultant specializing in physical, data and telecommunications security and crisis management. He was the former Vice President for Security for DIVX, a subsidiary of Circuit City and a retired Secret Service agent responsible for telecommunications security. He has over twenty-five years experience in telecommunications, including the design and implementation of backup planning in telecommunications. He will serve as the domain expert for telecommunications and contribute in other areas of his expertise.

Expert three is a Managing Director for Kroll Associates specializing in computer security and integrity controls. He holds a doctorate in Computer Science and is an expert in computer security, data processing controls, telecommunications and disaster recovery. He has over 25 years experience. He will serve as the domain expert in design of DRP systems, recovery site selection and plan maintenance and will assist in telecommunications.

Tools

VP Expert, an expert system development tool, will be used for designing the expert system. A flowcharting tool will be used to flowchart the design of the expert system and used as the tool to validate the logic of the expert system.

Table 1. The Knowledge Acquisition Process

1. Select domain experts.
2. Meet with all domain experts as a group to explain the purpose of the expert system, the knowledge acquisition process, etc.
3. Each domain expert develops an initial list of questions in their area of expertise.
4. The knowledge engineer reviews #3 and returns to the domain experts for follow-up action.
5. Each domain expert is asked to write explanations for why their questions were being asked and/or why the information is needed. (This will be used as a validation method and used in the explanation facility of the expert system).
6. Knowledge engineer reviews above and eliminates any delicate questions.
7. Final questions are modularized by the knowledge engineer and distributed to the domain experts.
8. Knowledge engineer meets with the main experts as a group to discuss the latest version.
9. Knowledge engineer makes final revisions and flowcharts the expert system.
10. Knowledge engineer meets with domain experts and individually or as a group for final approval of final version of questions, explanations and flowcharts. Revisions made immediately for implementation.

Table 2. Validation & Verification of Knowledge Acquisition & Expert Systems.

TOOL/TECHNIQUE	REASON FOR USE
Explanation Facilities	• Used to validate during knowledge acquisition phase the need for question and data requested. • Used to assist client during consultation. • Used to provide client with reason question is being asked. • Can be used as a training tool.
Flowcharts/Decision Trees	• Used to verify flows and branching of questions and expert system during consultation.
Decision Tables	• Used to validate and verify decision-making based on rules.
Walkthroughs	• Used to validate results by comparing decisions making or conclusions by expert systems to those of experts during actual consultation..
Simulations	• Used to validate results by comparing decisions making or conclusions by expert systems to those of experts during simulated walkthroughs.
Case Studies	• Used to validate results by comparing decisions making or conclusions by expert systems to those of experts using case studies.

Table 3. Milestones

Task	Dates to be completed
1. Selection of domain experts	**Completed**
2. Initial briefing to all domain experts	**Completed**
3. Knowledge acquisition process begins. Each domain expert completes version 1 of general questions & questions specific for their area of expertise	**1/31/00**
4. Knowledge engineer (KE) reviews all questions, categories and eliminates duplicates	**2/7/00**
5. Domain experts reviews product in step 4 above, make comments, add new questions, etc. (=version 2)	**2/14/00**
6. KE reviews all questions, categories and eliminates duplicates for version 2	**2/21/00**
7. KE finalizes above & develops flowcharts	**2/28/00**
8. Domain experts review final version and flowcharts for signoff before programming	**3/5/00**
9. KE codes programs for expert system using shell	**3/31/00**
10. Expert system is debugged and tested & revisions made	**3/31/00**
11. Expert system is tested by domain experts & revisions made	**4/15/00**
12. Final product approval	**4/30/00**

Annotated Bibliography

American Institute of Certified Public Accountants (AICPA). 1988. The auditor's consideration of an entity's ability to continue as a going concern. *Statement on Auditing Standards No. 59*. New York, NY: AICPA

Statement is an auditing standard to be followed by CPA's in auditing the financial statements of an entity. Generally accepted accounting principles are based on the going-concern concept, which means that the entity is expected to continue in operation and meet its obligations as they become due, without substantial disposition of assets outside the ordinary course of business, restructuring debt, externally forced revisions of its operations, etc. Hence an opinion that financial statements are in conformity with GAAP means that continued existence may be presumed for a reasonable time not to exceed one year beyond the date of the financial statements. Today, this going-concern concept is applied to the entity's ability to resume data processing services (disaster recovery planning and business resumption planning), especially in entity's where data processing ability is critical to operations.

Andrews, R. A. (1994). An ounce of prevention: Guidelines for preparing a disaster recovery plan. *Proceedings of the IEEE 1994 National Aerospace and Electronics Conference, 2*, 802-806

The effects of disasters can render an organization helpless. Over 70% of organizations that experience a serious emergency close within two years. Virtually all high-tech organizations are computer-dependent and most cannot perform day-to-day operating activities without their computer systems. In defense software development efforts, the loss of months or years of development work could end a program. The most difficult task of the recovery stage is resuming computer operations. The computer system(s) of an organization must be reconfigured quickly if the organization is to avoid long-term damage. Not only must new hardware and system software be acquired, but also the entire system must be configured to the original "pre-disaster" state. This rebuilding process includes restoring the required communication links and restoring the application software to each designated machine. This assumes that the systems are backed-up regularly, and the media are stored offsite. The objective of this paper is to heighten the awareness of the need for every business organization, or laboratory project, to have a disaster recovery (or business resumption) plan. Experience has shown that organizations which have contingency plans in place prior to an emergency have a much greater chance for survival than those organizations which only prepare the plan after a disaster has occurred. The focus is on preparing for emergencies involving an organization's computer systems. Steps for putting together a recovery plan, including issues that must be addressed in coordinating such a plan, are discussed

Byrd, T., Cossick, K. & Zmud, R. (1992). A synthesis of research on requirements analysis and knowledge acquisition techniques. *MIS Quarterly*, 16(1), 117-138.

Article compares techniques for requirements analysis used in conventional applications and knowledge acquisition used in expert systems. Authors note that each is the most critical step in their respective systems development process. Article presented an initial categorization scheme to facilitate the merging of research across these two areas and divides the techniques into groups by method of elicitation. Article also demonstrates that techniques in RA and KA share similarities in characteristics and purpose. With expert systems adopting some of the features of information systems, and vice versa, it seems imperative for knowledge engineers and analysts to be aware of a variety of elicitation techniques, regardless of their origin, to be able to develop advanced systems.

Chen, S. (1993). A knowledge acquisition scheme for ruled-based systems. *Proceedings IEEE Region 10 Conference on Computer, Communication, Control and Power Engineering*, 2, 621-625.

Article discusses KA schemes for rule-based systems called fuzzy Petri nets to represent the fuzzy production rules of a rule-based system, where the domain expert or KE can add new knowledge or update the existing knowledge by inserting new fuzzy production rules into the file or by updating the fuzzy production rules in the file.

Cerullo, M. J., & McDuffie, R. S. (1994). Planning for disaster. *CPA Journal, 64*(6), 34-37

Authors are accounting professors and discuss disaster recovery planning. They note that recent disasters (World Trade Center, Hurricanes Andrew & Hugo, San Francisco earthquake) are calls to industry for the need of DRP. Includes: areas of concern for recovery (facilities, communications); risk analysis; matters to consider (e.g., DR manager, voice & telecommunications, vital records, etc.); benefits. Article concludes with discussion of SAS 60 (Communication of Internal Controls Structure Related Matters in an Audit) which notes that the lack of a DRP may be a reported condition. Key element of article is reference to survey conducted of 71 companies in Charleston, SC after Hurricane Hugo. Results: 41 responded; 44% had DRP before hurricane; 56% did not; of the latter, downtown was 1-15 days for 20 companies, 16-60 days for 2 companies, and 2-4 months for 1 company.

Garcia-Molina, H. & Polyzois, C. (1990). Issues in disaster recovery. *COMPCON Spring '90. Intellectual Leverage. Digest of Papers. Thirty-fifth IEEE Computer Society International Conference*, 573-77.

Articles discusses several issues in disaster recovery including motivation for remote backups and their applicability to various systems, issues involved in the design, implementation, and evaluation of remote backup mechanisms, and some criteria for evaluating and comparing various remote backup mechanisms.

Authors define a disaster as total loss of processing power and access to data at one site for a period of time and possible permanent loss of data at that site. Disasters include natural (earthquakes, power outages), malicious acts, hardware failure, operator errors, etc. They noted that a remote backup can ensure continuous operation even in the presence of extensive failures that may render an entire site non-operational for which local replication may be inadequate.

Also noted that some federal regulations call for the existence or development of a disaster recovery plan (e.g. banks).

Griswold, J. S., Lightle, T. L., & Loverlady, J. G. (1990). Hurricane Hugo: effect on state government communications. *IEEE Communications, 28*(6), 12-17

Article discusses the impact of Hugo on state government communications in South Carolina and activities in the wake of the disaster. The state of South Carolina had been chosen, along with California, as a model state for development of the National Communications System/Telecommunications Services Priority (NCS/TSP) model plan for emergency restoration of telecommunications services. This effort is being undertaken in conjunction with the National Communications System Organization in Washington, DC. They are assisting the state in development of the telecommunications services priority (TSP) plan, to be used in the future for coordinating restoration of critical federal, state, and local telecommunication services. A private sector partnership has been initiated with the telecommunications and utility companies in South Carolina to provide fiber-optic alternate routing and other communications backup capability to each of the emergency preparedness offices in South Carolina counties in future disasters. An exchange of disaster recovery plans with other states has been undertaken. Information regarding each of these efforts is provided.

Hiles, A. (1992). Surviving a computer disaster. *Engineering Management Journal*, 2(6), 271-274.

Most organizations are so reliant on their computing service that loss of real-time systems could have devastating effects. The first step to improving this situation is to identify what risks exist and how significant they are within the specific environment of the organization. Disaster recovery plans can then by formulated in order to restore permanent computing service.

Disaster is defined as an event which causes the loss of the computing service, or of a significant part of it, or of systems, communications or applications, for a length of time which prevents the impacted organization from achieving its mission or which imperils the business. Risk include fire, flood, lightning strikes, malicious damage, power failures, impact, subsidence, loss of supplied services or special consumables, rodent damage, insect infestation, contamination, telecommunications failure, industrial action,…. A risk review will identify the key threats and includes a critical component failure analysis and business interruption analysis. Elements of a DRP include: immediate restoration procedures, restoration of computing infrastructure and of applications, resumption of business processing under emergency arrangement, and restoration of the permanent computing service.

Hoffman, R. (1987). The problem of extracting knowledge of experts from the prospective of experimental psychology. *AI Magazine, 8*, 53-64

Author discusses his research on generating methods for extracting the knowledge of experts from the experimental psychology side of expert systems. He notes, as others have, that knowledge acquisition is a significant bottleneck in the system development process. His research offers: (1) a working classification of methods for extracting knowledge (table 1); (2) some ideas about the types of data that the methods yield; (3) a set of criteria by which the methods can be compared relative to the need of the system development (table 5); (4) some salient advantages and disadvantages of the various methods table 4); (5) comparison of the results of four of the methods in terms of efficiency (table 7); and (6) some steps for extracting and characterizing knowledge prior to the construction of an expert system (table 8). The discussion highlights certain issues, including the contrast between the empirical approach taken by experimental psychologists and the formalism-oriented approach that is generally taken by cognitive scientists.

Hwang, G. (1994). Knowledge elicitation and integration from multiple experts. *Journal of Information Science and Engineering*, 10, 99-109.

Article discusses knowledge acquisition from multiple experts. Author notes that knowledge acquisition is known to be a critical bottleneck for building expert systems and that this is complicated when multiple experts are needed due to conflicts and inconsistencies. The paper discusses the use of a KA method called MERGE (multiple expert repertory grid elicitation) to cope with these problems by integration of the expertise.

Jacobs, J., & Weiner, S. (1997). The CPA's role in disaster recovery planning. *CPA Journal, 67*(11), 20-25

Article discusses opportunities for consulting services by CPAs. Notes SAS 60 as justification for bringing DR issues into the audit process. Includes: Types of companies that used DRP; Disaster planning alternatives; Steps in the formation and implementation of DRP (11 steps); Benefits; Overcoming plan preparation difficulties ("some of the problems encountered in completing a client's DRP relate to the difficulty of gathering the essential information to be detailed in the body of the plan"). Key point: 1993 industry study commissioned by then DEC determined that "90% of companies that experience a catastrophic loss of data and equipment and do not have a DRP are out of business within 2 years." Some of the problems encountered in completing a client's DRP relate to the difficulty of gathering the essential information to be detailed in the body of the plan.

Jeng, B., Lieng, T.-P. & Hong, M. (1996). Interactive induction of expert knowledge. *Expert Systems with Applications*, 10(3), 393-401.

Article discusses knowledge acquisition. Authors note that KA is a bottleneck in developing KBS. They note that it is a manual approach that elicits domain knowledge by interviewing human experts and that this process has problems because it the experts are often unable to articulate their reasoning. They also note that attempted automated approaches that induce knowledge from a set of training cases also suffer from the unavailability of sufficient case studies. The authors present an integrated approach that combines the strengths of both methods to compensate for their weaknesses.

Authors note that there are many different methods for KA, which can be classified as manual, semi-automatic and automatic. Manual methods are highly labor intensive and include structured/unstructured interviews, analysis of protocol, observations, etc.

Induction is a process of general inference from particular instances, whereas, rule induction refers to a concept learning process by which a set of rules is created from training cases or case studies to explain or to predict a problem solving behavior When the induced structure of knowledge is represented in the form of a decision tree, it is also called (decision) tree induction. A decision tree is considered as a set of rules in a compact form, which can easily be transformed into rules.

Kemp, E. & Purvis, M. (1996). The role of the individual project in teaching knowledge acquisition. *Proceedings of the International Conference in Software Engineering: Education and Practice,* 138-143.

Knowledge acquisition is that stage of knowledge based systems development, which is equivalent to the analysis and design phases of the conventional software life cycle. An account is given of the experience gained when teaching a postgraduate course on this topic. One of the aims of the course, "Topics in Knowledge Acquisition", is to provide students with the opportunity to discover for themselves what this stage of the knowledge based system development life cycle involves. Eliciting, analyzing and modeling domain knowledge, the main activities of the knowledge acquisition process, are fraught with problems. Each student undertakes the task of developing a small expert system in an appropriate domain. The emphasis in this project is not on the implementation but on the processes the students follow. Students are asked to evaluate the elicitation, analysis and modeling techniques they use. The paper describes the information available to the students before they commence the project, gives an account of their experiences during the knowledge acquisition stage, discusses their findings and concludes with lessons learned for the future.

Liebowitz, J. (1993). Educating knowledge engineers on knowledge acquisition. *IEEE International Conference on Developing and Managing Intelligent Systems Projects*, 110-117.

Article notes that although many universities are offering programs in knowledge engineering (?), its is rare that a course in knowledge acquisition is included. Paper discusses the need for courses in KA that will contribute to improving the state-of-the-art in expert system development as well as a general background in KA.

Authors note that KA bottleneck has been well documented; that one day of the expert's time is usually needed for every four days of the KE's time; because of the difficulty in acquiring and eliciting knowledge, potential KE's need a stronger foundation in the fundamentals, methodologies, techniques, and tools for acquiring knowledge; that George Washington University is one of the few schools that do offer such a course. They noted that the KA process includes: (1) discovery of prelim knowledge and problem range; (2) identifying sources of information; (3) acquiring detailed knowledge from sources; & (4) analyzing, coding and documenting knowledge. They quote Cullen for KA problems: i.e. quality of expertise, communication, etc. They also quote (?) that a major reason why KA is a difficult process relates to the Paradox of Expertise which states that the more competent domain experts become, the less able they are to describe the knowledge they use to solve problems. Specifically: experts have a tendency to state their conclusion and the reasoning behind them in general terms that are too broad for effective machine analysis; the pieces of basic knowledge are assumed and are combined so quickly that it is difficult for the expert to describe the process; that in explaining his conclusion or hunch, the expert will repeat only the major steps, leaving out most of the smaller ones (note: this is a concern in my research). They also quote (?) for other problems such as: Experts may lack time or may be unwilling to cooperate; Methods for KA many be poorly defined; Use of one source when relevant knowledge is scattered across several sources; Builders may attempt to collect documented knowledge rather than use experts; difficulty in recognizing specific knowledge from irrelevant data; Experts may change their behavior when they are observed or interviewed; and Interpersonal communication factors between KE and expert.

Liou, Y.I. (1999). Expert system technology: knowledge acquisition. In J. Liebowitz Ed.), *The Handbook of Applied Expert Systems*. (pp. 2.1 – 2.11). New York: CRC Press.

Article presents chapter on knowledge acquisition. Author discusses selection of experts based on (1) domain expertise, experience and reputation (2) personal characteristics & attributes, & (3) availability. He references McGraw's book for problems when using one expert; references himself for problems using multiple experts; and references himself et al for the positive effects of using multiple experts.

Problems when using one expert include: (1) difficulty in allocating time by a key individual in the organization, that may create a bottleneck; (2) personal bias that may affect the performance of the system; and (3) limitation to a single line of reasoning that may affect the usefulness of a system since expert systems that are developed based on a single, perhaps narrow line of reasoning, do not emulate most real-life decision making; and (4) incomplete domain expertise that may affect the performance of a system.

Problems with multiple experts: (1) difficulty of merging each individuals expert's knowledge into one group structure that provides the underlying problem solving expertise of the ES (i.e., conflict resolution); and (2) difficulty of the generation of group knowledge that does not reside in any one individual expert but resolves as a result of group interaction. I will note that a solution to this is a strong KE with strong knowledge in the domain.

Positive effects of multiple effects include: (1) it assures that the KB can be complete; (2) improves the likelihood of obtaining specialized knowledge in subdomains of the problem (important for my research); (3) increases the quality of the acquired knowledge; (4) assures that the facts that are included in the KB are important ones; (5) enhances understanding of the domain knowledge through discussion, debate and exchange of the hypothesis between members of the expert team; and (6) encourages interactions among experts and creates a synergy such that the acquired group knowledge is sum of the individual's knowledge.

McGraw, K. L., & Harbison-Briggs, K. (1989). *Knowledge acquisition: Principles and guidelines*. Englewood Cliffs, NJ: Prentice-Hall.

Textbook written for novice knowledge engineers. Presents a practical approach to the KA process. KA methodologies (Chapter 1-3) and techniques (Chapter 4-11) are detailed, including the benefits of working with multiple domain experts and the most effective means for resolving problems that may arise. Authors note that KA is the most critical element in the development of an expert system and that it is both people and time intensive (p.2). They state that most of the literature related to ES development alludes to the troublesome nature of KA, even when only a single domain expert has contributed to the knowledge base. The already difficult KA process becomes even more complicated when the ES being developed requires that knowledge engineers interact with multiple experts (p. 242). Complex, real life problems are seldom so simple that they can be solved based on consultations with a single expert. More often than not, a complex problem requires access to different types of knowledge and thus, to more than one expert (p. 243).

Money, W. & Harrald, J. (1995). The identification of group support systems to knowledge acquisition for disaster recovery planning. *Proceedings of the Twenty-eighth Hawaii International Conference on System Sciences*, 4, 468-474.

This paper describes the results of the use of a Group Support System (GSS) and two "hindcasting" scenario exercises. The GSS is used to implement the hindcasting, knowledge mapping, and knowledge representation techniques as disaster planning tools. This work improves our understanding of the knowledge engineering process in the areas of knowledge acquisition, validation, and representation when the knowledge must be collected from the multiple experts, and transferred to a knowledge base. The experiences from the exercises illustrate how the collection of the knowledge of experts may be supported by a GSS when the experts perceive their task to be one of collaboratively deducing a rational causal explanation for a complex event. The results indicate that managers may improve their ability to integrate multiple expert's knowledge into the disaster preparedness planning process using GSS processing techniques.

Nguyen, T., Perkins, W., Laffey, T. & Pecora, D. (1987). Knowledge base verification. In J. S. Chandler & T.-P. Liang (Eds.), *Developing Expert Systems for Business Applications*. (pp. 69-77). Columbus, OH: Merrill Publishing.

Article describes CHECK, a program whose function is to check a knowledge base for consistency and completeness, i.e., knowledge base verification. Verification is defined as a technique or methodology for testing the consistency and completeness of a rule set. The program detects several potential problems in verification, including redundant rules, conflicting rules, subsumed rules, unnecessary IF conditions, and circular rules. CHECK also attempts to verify completeness in the knowledge base by looking for potential gaps, including unreferenced attributed values, illegal attribute values, missing rules, unreachable conclusions, and deal-end goals. Authors in their research note that many changes and additions to the rule sets occur during the development of a knowledge base.

Niederman, F. (1996). Acquiring knowledge about group facilitation: research propositions. *Proceedings of the 1996 conference on ACM SIGCPR/SIGMIS.* 58-67.

This paper argues for the desirability of investigating the expertise that facilitators use to lead groups. It reviews the importance and difficulties in studying meetings. It reviews the nature of knowledge acquisition techniques and suggests a knowledge acquisition model for developing a map of the expertise in the domain of group facilitation. Sets of propositions regarding (1) the domain of knowledge for expert facilitators and (2) differentiation of facilitators by level of expertise are presented. Finally, potential benefits of further investigation of facilitator expertise are discussed.

Nuseibeh, B., Kramer, J., & Finkelstein, A. (1994). A framework for expressing the relationship between multiple views in requirements specifications. *IEEE Transactions on Software Engineering,* 20(10), 760-773

Article discusses multiple viewpoints or perspectives in requirements specifications. Authors note that even in a single activity such as requirements elicitation, it is likely to involve multiple development participants who will hold multiple perspectives of a single domain. Multiple perspectives may overlap, complement or contradict each other. In their use of ViewPoints, they note that in attempting to integrate multiple requirements specifications, overlaps must be identified and expressed, complementary participants made to interact, and cooperate, and contradictions resolved.

O'Keefe, R., Balci, O, & Smith, E. (1990). Validating expert system performance. In J. S. Chandler & T.-P. Liang (Eds.), *Developing Expert Systems for Business Applications*. (pp. 91-102). Columbus, OH: Merrill Publishing.

Article seeks to establish validation as an important concern in expert system research and development. It discusses problems in expert system validation and presents the qualitative and quantitative methods for validating expert systems. It was noted that engineers typically have validated expert system performance by running test cases through a system and comparing results (i.e., the classification, final certainty factors, and advice given) against known results or expert opinion. The authors note problems with this process. Validation is often confused with verification. Simply stated, validation refers to building the right system (i.e., substantiating that a system performs with an acceptable level of accuracy) whereas verification refers to building the system "right"; (i.e., substantiating that a system correctly implements its specifications).

O'Leary, D. (1990). Validation of expert systems with applications to auditing and accounting expert systems. In J. S. Chandler & T.-P. Liang (Eds.), *Developing Expert Systems for Business Applications*. (pp. 78-90). Columbus, OH: Merrill Publishing.

Papers presents a theory-based framework that is useful for guiding the validation of an expert system and also for eliciting other validation issues. The framework addresses validation in terms of validity, objectivity, cost-benefits and accuracy. Validation is important to the decision-making success of an expert system and to the continued use of the expert system. An expert system that does not make the correct decisions will lead to a loss of confidence in it and non-use. Therefore, expert systems must be validated prior to implementation in the field for real like use. Author also noted that validation requires: (1) ascertaining what the system knows, does not know, or knows incorrectly; (2) ascertaining the level of expertise of the system; (3) determining if the system is based on a theory for decision-making in the particular domain; and (4) determining the reliability of the system.

O'Neil, M. & Morris, A. (1989). Expert systems in the United Kingdom and evaluation of development methodologies. *Expert Systems, 6*, 90-99.

Article discusses survey performed in late 1987, early 1988, of expert system developers which was aimed to establish the nature and scope of expert system projects in Britain, the skill set and methodologies used in their development and the background and experience of the personnel employed to develop them. Major part of survey focused on the human input and organizational impact of the expert systems. It includes details of the organizational background of those working in the area, review of the nature, number and scope of expert system projects, the knowledge engineering task and methodologies used, the skills and attributes of the KE, and issues of updating and extending expert systems.

Noted in survey were:
Only 22% of expert systems were designed using multiple experts. Overriding reason for the use of a single expert seemed to be that it was easier both to elicit the knowledge and to avoid contentious issues and conflicting opinions. Several developers had tried to use multiple experts but had faced problems of conflicting information, team disagreements and individual experts trying to assert their own authority over the group. However, they noted that in cases where it was impossible to obtain all the information needed from one expert, it was necessary to use multiple experts and that the KE needed to take steps to avoid conflict or dissent by: (1) asking experts to provide documentary backup evidence; (2) the use of probabilities and uncertainty factors to express degrees of agreement or disagreement amongst experts; and (3) creating systems modularly so that different experts could be specifically called on for particular sections and consultation. I will take these recommendations into my KA/expert system design approach by requiring all experts to state the reasons they need the information they are asking for (which I will also use as a validation technique and for the explanation facility of VP-Expert). In addition I will design the ES modularly and have each modular done by a specific domain expert, although all will eventually review the others to ensure all matters are covered.

Rees, P. L. (1996). User participation in expert systems. *Industrial Management & Data Systems, 93*(6), 3-7

Article discusses the need for user involvement in the design and implementation of expert systems and emphases that the lack of such is a reason for failure of expert systems. The article first outlines some of the work that has been carried out in the area of participative design of new technology. It then goes on to describe three expert system projects and uses them to illustrate the difficulty of participative design and the dangers of not undertaking it. The article concludes that, just because participation is difficult, this is insufficient justification for rejecting it as an important goals.

Author notes DEC's XSEL as one of the most successful use of participative design and discusses Mumford's methodology for participative design, ETHICS (Effective Technical and Human Implementation of Computer-based Systems), which provides practical guidance for user participation.

Rook, F. & Croghan, J. (1989). The knowledge acquisition activity matrix: a systems engineering conceptual framework. *IEEE Transaction on Systems, Man & Cybernetics*, 19(3), 586-597.

Paper presents a systems engineering conceptual framework that provides a structure for the specification of KA activities throughout the SDLC. Objectives include: (1) review general KA requirements for the development of applied knowledge-based systems; (2) present a SE conceptual framework in the form of the KA acquisition activity matrix (KAAM), for structuring KA activities; and (3) discuss the specific goals, constraints, & output of KA steps in the context of the SDLC. Authors also note the KA has often been described as the bottleneck in the KBS development and that most responses to countering this bottleneck phenomenon focus on specific knowledge elicitation methods and techniques. Authors suggest a SE conceptual framework (KAAM) to address the bottleneck.

Rudolph, C. Business continuation planning / disaster recovery: a marketing perspective. *IEEE Communications Magazine*, 28(6), 25-28.

Article discusses requirements for restoring telecommunication services to businesses following a disaster such as the central office fire in the Chicago suburb of Hinsdale in 1988 and several California earthquakes. Also discussed was US West's disaster recovery strategies and disaster recovery marketing tool (called Preparedness Audit). An illustrative case study is presented. Author noted that evidence strongly supports the idea that companies can survive disaster if they: (1) plan for the possibility of a disaster; (2) formulate strategies for recovering critical business functions; (3) implement technologies to aid the recovery of automated functions and systems; and (4) train employees to implement those strategies.

Survey conducted at the University of Minnesota revealed that many businesses cannot survive with data processing capabilities beyond 5 days. Maximum Time To Belly Up (MTTBU)=term is being used in the industry to denote the time frames that contingency planning coordinators have to restore not only the DP applications, but the associated telecommunications networks as well. Reality dictates that proactive, preplanned solutions for restoring communications capability be a part of the contingency planning process for companies which rely on telecommunications for critical business functions.

Song, I. -Y., Strum, S. & Medsker, C. (1991). Design and implementation of a database design aid using VP-Expert. *Proceedings of the IEEE/ACM International*

Conference on Developing and Managing Expert System Programs, 1991, 15-23.

Article discusses the design and implementation of a database modeling aid called Database Designer using VP-Expert. Authors discuss their approach to system design and implementation issues specific to the expert system design shell, VP-Expert, such as combining backward chaining and forward chaining (this combination will be used in my research project), procedural techniques, and transparency and visibility of VP-Expert statements. Authors note that VP-Expert allows one dimensional arrays and forward chaining; that an ES for design applications should utilize forward chaining, since its desired goals are not known in advance or are too numerous to list; that forward chaining is most useful in creating data driven processes and ES's with unknown goals.

Stein, E. W. (1993). A method to identify candidates for knowledge acquisition. *Journal of Management Information Systems, 9*(2), 161-178

Article talks about network analysis as a method for selecting candidates for knowledge acquisition. Network analysis is based on the measurement of interactions or relations between individuals. Author noted that network analysis can help the knowledge engineer and managers identify experts that can be identified based on their centrality as information providers within the organization.

Network analysis seems to be time consuming and looks broadly at organization's members to determine who has knowledge. However, validation seems to be based on what other members of the organization think about an individual. Question=do they really have the expertise? For example, an accountant may know the most about accounts receivable in his company, but is he an expert in accounts receivable? These are really two different questions. Note: Network analysis may work if the knowledge acquisition is specific to the organization itself.

Stylionou, A., Smith, G., & Madey, R. (1993). *Selection criteria for expert system shells. A sociotechnical framework.* Communications of the ACM, 35(10), 30-48.

The evaluation and selection of a specific shell are important parts of expert system development projects. Use of the wrong shell could lead to major compromised or project failure. It is difficult to sort through the hype and evaluate many shells and vendor claims; therefore many evaluate and select arbitrarily and unsystematically. The objective of the article was to provide a systematic process for the selection and evaluation.

Authors note benefits of expert systems including: cost reduction, increased output, improved quality, consistency of employee output, reduced overtime, captured scarce expertise, flexibility in providing service, easier operation of equipment, increased reliability, faster response, ability to work with incomplete and uncertain information, improved training and increased ability to solve complex problems.

The shell evaluation and selection process include (1) identification of required capabilities and features; (2) identification of potential shells; (3) evaluation of shells based on required capabilities and features; and (4) selection of the appropriate shell. Based on their survey, they determined the shell capabilities considered to be the most important are: embeddability (refers to the ability of ES shells to be built into conventional applications thereby providing these applications with the advantages of a KB system), rapid prototyping, backward chaining, explanation facility, ability to customize explanations, linkage to databases, and documentation comprehensiveness/readability.

Stylionou, A., Smith, G., & Madey, R. (1995). An empirical model for the evaluation and selection of expert system shells. *Expert systems with applications*, 8(1), 143-155.

Authors develop an empirical model for selecting and evaluating expert system shells. The model was based on a survey of 271 knowledge engineers and end-user and can be used to identify critical ES shell attributes and capabilities, which can then be used as evaluation criteria. On page 153, authors show a diagram of the expert system shell evaluation process (model) they recommend. Authors define ES shells as software packages containing a generic inference engine, a user interface and a collection of other tools that enable users to develop and use expert systems. Using the shell's tools, a KE can develop new knowledge bases and can structure, add, delete, and modify the knowledge contained in them.

There are more than 100 commercial ES shells available in the market (note: this has decreased tremendously in 1999 since interest in expert systems is fading). The selection of a specific ES shell for a particular application is an important decision. If the wrong shell is selected it could result in an inefficient or ineffective system or even in project failure. The difficulties encountered during the process of selection and evaluation are not unlike those for other software packages. The selection depends on factors such as: application characteristics (e.g. problem domain); project characteristics (developer's sophistication with AI techniques); capabilities of shell, sophistication of users; ease of integration with existing software and hardware; and vendor support.

Vanthienen, J. & Robben, F. (1993). Developing legal knowledge based systems using decision tables. *Proceedings of the fourth international conference on artificial intelligence*, 282-291.

Paper describes the methodology, tools used and experience in developing an expert system used to determine financial benefits for the disabled in Belgium. Authors discuss use of decision tables for KA, representation and for testing consistency (verification and validation) of legal knowledge. Includes definition of decision tables and steps for constructing decision tables.

Waterman, D.A. (1985). *A guide to expert systems*. Reading, Mass: Addison-Wesley Publishing.

Book is one of the early textbooks on the subject of expert systems and is well known as an excellent introduction to expert systems. It is a good source of definitions in the subject as well as a reference for expert system tools, knowledge representation and methodology. Also includes: knowledge acquisition (chap 14), difficulties in development (chap 16), and common pitfalls in planning (chap 17), in dealing with domain experts (chap 18) and in development (chap 19).

He defines a domain expert as an articulate, knowledgeable person with a reputation for producing good solutions to problems in a particular field. He defines knowledge acquisition as the process of extracting, structuring and organizing knowledge from several sources, usually domain experts, so it can be used in a program.

Watkins, A., Dimopoulos, N., Neville, S. & Li, K. (1993). Flowtool: a procedural-knowledge acquisition tool. *IEEE Pacific Rim Conference on Communications, Computers & Signal Processing*, 1, 31-34.

Authors present FLOWTOOL, a procedural-knowledge acquisition tool developed specifically to acquire procedural knowledge associated with the diagnosis and calibration procedures of engineering systems. It uses a graphical user interface to acquire procedural knowledge and delivers a knowledge base complete with links to related hypermedia document(s). FLOWTOOL has been implemented and operates in a Unix/X-Windows environment and creates rules for NEXPERT OBJECT and PROLOG. FLOWTOOL has been used to capture diagnostic and calibration knowledge pertaining to C-COR main trunk amplifiers.

Wolfgram, D., Dear, T., & Galbraith, C. (1987). *Expert systems for the technical professional*. New York: John Wiley & Sons.

Book is a general text on expert systems. Authors discuss introduction to expert systems, structures, components, and construction of expert systems. Key to text is chapter on knowledge engineering and knowledge acquisition including definitions, techniques and tools. Authors also discuss types of applications suitable for expert systems and stages of development. Expert systems (ES) are defined as "computer systems, comprising both hardware and software that mimic an expert's thought processes to solve complex problems in a given field" (domain). ES address and solve knowledge-intensive problems (large information sets) that can have multiple correct or acceptable answers. Suitable applications for ES include: interpreting and identifying, predicting, diagnosing, designing, planning, monitoring, debugging and testing, instructing and training, and controlling. ES are appropriate where there are not established theories, where human expertise is scarce or in high demand, and where the information is cloudy or fuzzy. ES analyze and present the best possible answer with advice and recommendations. According to Barr & Feigenbaum (1981), production rules are best used in domains of diffused knowledge. These are domains consisting of many facts, such as clinical medicine.

Authors note that one of the most critical responsibilities of the knowledge engineer is KA. KA is defined as the process of identifying, extracting, documenting and analyzing the information-processing behavior of domain experts in order to define an expert system's knowledge base and inference engineer. ES will be only as good as the expertise it is built on---quality of knowledge is key. For some domains, it is sufficient that only a minimal level of expertise be achieved; in other domains, such as medical diagnostics, it is absolutely necessary that the best expertise be incorporated. KA techniques include: interviews (unstructured, open-ended, etc.), protocol analysis, walkthroughs, questionnaires, expert reports. Protocol analysis (AKA, think-aloud method), particularly a set of techniques known as verbal protocol analysis, is by far the most common method by which the KE acquires detailed knowledge from the expert.

References

American Institute of Certified Public Accountants (AICPA). 1988. The auditor's consideration of an entity's ability to continue as a going concern. *Statement on Auditing Standards No. 59*. New York, NY: AICPA

Andrews, R. A. (1994). An ounce of prevention: Guidelines for preparing a disaster recovery plan. *Proceedings of the IEEE 1994 National Aerospace and Electronics Conference, 2*, 802-806.

Byrd, T., Cossick, K. & Zmud, R. (1992). A synthesis of research on requirements analysis and knowledge acquisition techniques. *MIS Quarterly*, 16(1), 117-138.

Chen, S. (1993). A knowledge acquisition scheme for ruled-based systems. *Proceedings IEEE Region 10 Conference on Computer, Communication, Control and Power Engineering*, 2, 621-625.

Cirullo, M. J., & McDuffie, R. S. (1994). Planning for disaster. *CPA Journal, 64*(6), 34-37.

Garcia-Molina, H. & Polyzois, C. (1990). Issues in disaster recovery. *COMPCON Spring '90. Intellectual Leverage. Digest of Papers. Thirty-fifth IEEE Computer Society International Conference*, 573-77.

Griswold, J. S., Lightle, T. L., & Loverlady, J. G. (1990). Hurricane Hugo: Effect on state government communications. *IEEE Communications, 28*(6), 12-17.

Harmon, P. & King, D. (1985). *Expert systems, artificial intelligence in business*. New York: Wiley.

Hiles, A. (1992). Surviving a computer disaster. *Engineering Management Journal*, 2(6), 271-74.

Hoffman, R. (1987). The problem of extracting knowledge of experts from the prospective of experimental psychology. *AI Magazine, 8*, 53-64.

Hwang, G. (1994). Knowledge elicitation and integration from multiple experts. *Journal of Information Science and Engineering*, 10, 99-109.

Jacobs, J., & Weiner, S. (1997). The CPA's role in disaster recovery planning. *CPA Journal, 67*(11), 20-25.

Kemp, E. & Purvis, M. (1996). The role of the individual project in teaching knowledge acquisition. *Proceedings of the International Conference in Software Engineering: Education and Practice,* 138-143.

Liou, Y.I. (1999). Expert system technology: knowledge acquisition. In J. Liebowitz Ed.), *The Handbook of Applied Expert Systems*. (pp. 2.1 – 2.11). New York: CRC Press.

McGraw, K. L., & Harbison-Briggs, K. (1989). *Knowledge acquisition: Principles and guidelines*. Englewood Cliffs, NJ: Prentice-Hall.

Money, W. & Harrald, J. (1995). The identification of group support systems to knowledge acquisition for disaster recovery planning. *Proceedings of the Twenty-eighth Hawaii International Conference on System Sciences*, 4, 468-474.

Nguyen, T., Perkins, W., Laffey, T. & Pecora, D. (1990). Knowledge base verification. In J. S. Chandler & T.-P. Liang (Eds.), *Developing Expert Systems for Business Applications*. (pp. 69-77). Columbus, OH: Merrill Publishing.

Niederman, F. (1996). Acquiring knowledge about group facilitation: research propositions*. Proceedings of the 1996 conference on ACM SIGCPR/SIGMIS.* 58-67.

Nuseibeh, B., Kramer, J., & Finkelstein, A. (1994). A framework for expressing the relationship between multiple views in requirements specifications. *IEEE Transactions on Software Engineering,* 20(10), 760-773.

O'Keefe, R., Balci, O, & Smith, E. (1990). Validating expert system performance. In J. S. Chandler & T.-P. Liang (Eds.), *Developing Expert Systems for Business Applications*. (pp. 91-102). Columbus, OH: Merrill Publishing.

O'Leary, D. (1990). Validation of expert systems with applications to auditing and accounting expert systems. In J. S. Chandler & T.-P. Liang (Eds.), *Developing Expert Systems for Business Applications*. (pp. 78-90). Columbus, OH: Merrill Publishing.

O'Neil, M. & Morris, A. (1989). Expert systems in the United Kingdom and evaluation of development methodologies. *Expert Systems, 6*, 90-99.

Rees, P. L. (1996). User participation in expert systems. *Industrial Management & Data Systems, 93*(6), 3-7.

Rook, F. & Croghan, J. (1989). The knowledge acquisition activity matrix: a systems engineering conceptual framework. *IEEE Transaction on Systems, Man & Cybernetics*, 19(3), 586-597.

Rudolph, C. Business continuation planning / disaster recovery: a marketing perspective.

IEEE Communications Magazine, 28(6), 25-28.

Song, I. -Y., Strum, S. & Medsker, C. (1991). Design and implementation of a database design aid using VP-Expert. *Proceedings of the IEEE/ACM International Conference on Developing and Managing Expert System Programs*, 1991, 15-23.

Stein, E. W. (1993). A method to identify candidates for knowledge acquisition. *Journal of Management Information Systems, 9*(2), 161-178.

Stylionou, A., Smith, G., & Madey, R. (1993). *Selection criteria for expert system shells. A sociotechnical framework.* Communications of the ACM, 35(10), 30-48.

Stylionou, A., Smith, G., & Madey, R. (1995). An empirical model for the evaluation and selection of expert system shells. *Expert systems with applications*, 8(1), 143-155.

Vanthienen, J. & Robben, F. (1993). Developing legal knowledge based systems using decision tables. *Proceedings of the fourth international conference on artificial intelligence*, 282-291.

Waterman, D.A. (1985). *A guide to expert systems*. Reading, Mass: Addison-Wesley Publishing.

Watkins, A., Dimopoulos, N., Neville, S. & Li, K. (1993). Flowtool: a procedural-knowledge acquisition tool. *IEEE Pacific Rim Conference on Communications, Computers & Signal Processing*, 1, 31-34.

Wolfgram, D., Dear, T., & Galbraith, C. (1987). *Expert systems for the technical professional*. New York: John Wiley & Sons.