

SIMPLE, SMALL-SCALE KNOWLEDGE ENGINEERING FOR PRESERVING INSTITUTIONAL EXPERTISE

A CASE STUDY

Kenton Graviss, Ph.D.
Indiana University Southeast
Dept. of Computer Science

ABSTRACT

Knowledge engineering has matured to the point that there are numerous tools and software packages designed specifically to organize, make accessible, and distribute an expert's or organization's proprietary know-how. As a subdiscipline of AI, knowledge engineering was initially applied as a paradigm of expert systems over twenty-five years ago. It provided methods to capture and organize knowledge in a useful format accessible to non-experts. This paper presents an example of how an organization's expertise, structured into a knowledge domain, can be preserved and managed. The purpose is to provide a simple approach for identifying and organizing that expertise and to preserve it in a readily accessible format. Today, businesses, particularly small businesses, must operate at ever-increasing speed in functions critical to their success and continued viability. The refrain "operating at the speed of business" has never been truer. Keeping track of, or just identifying what serves as an organization's competitive edge (i.e., its proprietary expertise) often gets lost in the daily battle to maintain and grow the business. The advent of e-commerce has leveled the playing field, making the need to capture, organize, and distribute knowledge a critical capability that could make the difference between business success and failure.

A simple approach is given for a small-scale knowledge engineering application to show how an organization can safeguard against the potentially high cost of key personnel and knowledge losses. A commercially available knowledge tool was used to capture, and organize the engineering-domain knowledge of a small business. The result was the creation of a user-friendly neural-network based knowledge system capable of providing material selection recommendations for compressor seal applications. Formerly non-centralized and generally inaccessible knowledge was reformatted, structured, and made explicitly available for use by critical personnel. Benefits achieved include: identification of critical expertise, its organization and preservation, revelation of the importance of specific knowledge, discovery of knowledge gaps, identification of areas needing technology maintenance, and creation of an accessible knowledge base to support consistent recommendations and communications. In addition, the results have the potential for supporting business functions related to web commerce, promotion, training, and research, and development.

1. INTRODUCTION

The collective expertise of an organization is its lifeblood. Employee education, training programs, new equipment, and computer upgrades are approaches commonly used to stem the technology tide. The success of these efforts depends heavily on attracting, motivating, and keeping valued employees. In today's acutely competitive job market employers are harder-pressed to find and keep people with high-level skills and experience. People with these capabilities have options and, in some fields, can exercise them at will. The void they leave behind can become a vacuum from which small and medium businesses may not quickly recover.

Reasons for capturing and preserving an organization's expertise include: identification of its proprietary knowledge base, protection of expertise critical to a competitive business position, communication and distribution of consistent, reliable information, quality assurance for engineering and production, sales and marketing trumpeting, and to guide development and research. Seldom however, are such qualitative (but substantive) reasons converted to explicit goals for active pursuit, particularly in small businesses. This is likely due to constraints on time, personnel, and resources in addition to: the difficulty of quantifying bottom-line benefits, the perceived esoteric nature of the endeavor, and a belief that time-consuming and expensive tools (software) and personnel would be required. These preclude many organizations from even considering an attempt.

A simple, cost-effective approach to knowledge engineering is presented. An inexpensive (yet powerful) tool was used to structure and present the "embodied" technical expertise of a small company. The company is a 113-year-old business and though small (< \$50MM annual sales) it has historically been and continues to be, a global industry-leader in the development and manufacturing of high quality compressor seals¹. The knowledge tool used is a neural-net based program with knowledge capturing, structuring, and presentation features with a Windows?-like interface. The tool provided a convenient scheme for knowledge representation with a user interface and inferencing approach that was clear and understandable. It requires mapping knowledge to a set of case examples that then serve as inputs for training a neural net².

Figure 1 illustrates the impact on productivity that such knowledge losses can have on the aggregate learning curve of the business. The right-shift of curve A to B represents the time required to recover the lost expertise and bring it back to the level achieved by the original experts. The shift factor, τ , has a greater impact on a small business than a large or medium-sized organization due to the lack of back-up personnel and resources available to cover such contingencies. For a business, it can be important to preserve in as convenient a form as possible, the knowledge lost as senior and key people move on.

There is also an obvious need for an organization to provide consistent and reliable information regarding its services, products, and capabilities. In many cases engineering personnel are responsible for making a wide range of technical decisions to production plants, sales and marketing departments, and customers. Often such departments serve key functions with a nonuniform distribution of knowledge and expertise. As a result the most senior and knowledgeable personnel are besieged to provide information and assistance. This ties them up intermittently, or continually, serving the business on one hand but depriving it on the other by not freeing them to apply their expertise in functions critical to business viability (and possibly survival) such as in product/process development and research.

¹ C. Lee Cook, A Dover Resources Company, Louisville, KY, www.cleecook.com.

² Intellix Designer (KnowMan? Basic Suite '99), Rev. 3, by Intellix A/S, Copenhagen, Denmark, www.intellix.com.

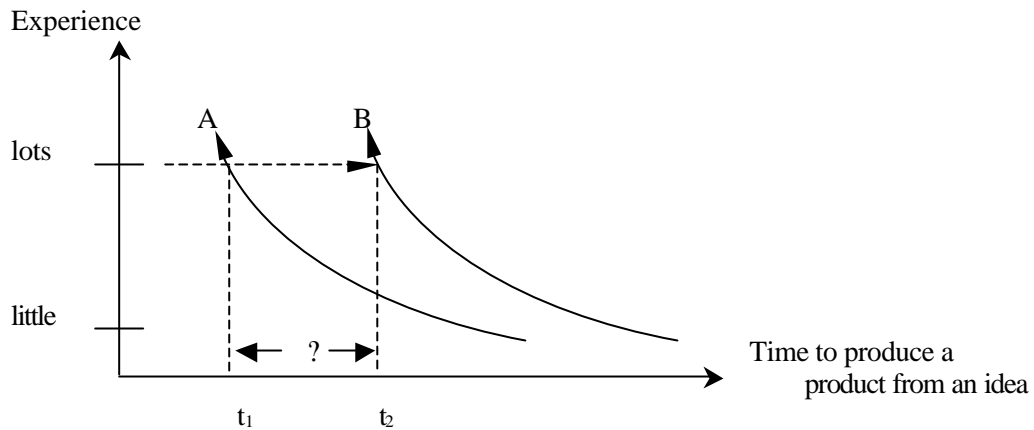


Fig. 1 – learning curve shift from loss of expertise.

Situations like this occur in all organizations but the impact is greatest in small ones where resources are the most dear. Ironically, such businesses see the least need for knowledge engineering and perceive themselves the least capable of doing anything about it. Yet these have the most to lose by neglecting the preservation of knowledge and expertise.

Knowledge preservation makes information more consistent, reliable, and ultimately more *meaningful* to an organization. The value of “meaningfulness” should not be underestimated. It can transform otherwise useless *data* from *information* into useful *knowledge*, building a foundation that can be used to maintain control of its business destiny.

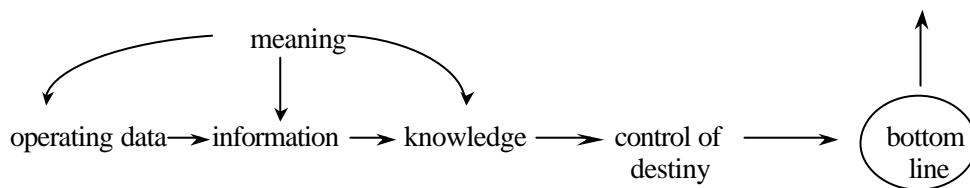


Fig. 2 – transforming data into productive knowledge.

2. BACKGROUND

Expert systems are derived from the field of artificial intelligence (AI). AI is a multidisciplinary branch of computer science that has contributed to many other fields as diverse as engineering, cognitive science, behavioral science, mathematics, biology, and medicine [1]. Knowledge-based systems grew out of AI research and form the broader domain encompassing expert systems. Such systems deal with the nature of expertise and the representation of knowledge in a computer program. The general minimum requirements for a computer program to be considered useful as a knowledge-based system are that it store knowledge (not just data or information) and be capable of solving problems with it. Expert systems have other requirements though such as being able to simulate reasoning about a problem of technical or commercial value using knowledge representations and heuristics (rules of thumb) instead of linear, algorithmic, or statistical means [2]. Since the end of World War II knowledge systems have received much research attention. Well-known examples of successful systems include programs designed for language understanding (SHRDLU), robotics (STRIPS), medicine (ONCOCIN, MYCIN), and telephone switching

diagnostics (COMPASS) to name a few [3-7]. In general, engineering a knowledge system requires at least three ingredients to be present: known expertise, a tractable, quantifiable knowledge domain, and a method for referencing and inferencing it with a computer. For the first of these, two questions must be addressed,

- i) are there *available* experts?
- ii) are they *willing* and *able* to articulate and share it?

The second ingredient deals with the nature of the application. Relative to the vast knowledge-storing capacity of the human brain, knowledge tools have a limited capacity for representing knowledge. Certain tasks and domains are not amenable to computer representation such as those that,

- o rely heavily on commonsense reasoning (speech interpretation, physical causality, awareness, perception, intuition, notions), and
- o those requiring sophisticated sensory-motor skills (beyond those of current robots and computer vision systems).

Fortunately, tools are available for the third ingredient and these include AI languages, shells, and editors that have been developed to assist with acquiring and representing knowledge in a useful form. They include programs such as KRL (Knowledge Representation Language), LISP, PROLOG, CLIPS (C Language Integrated Production System), EMYCIN, PROTEGE, and TEIRASIS, [8-11]. Knowledge tools like these pioneered the application of knowledge-based methodologies for solving real-world problems.

3. KNOWLEDGE TOOL SELECTION

Selection of the knowledge tool was decided on the basis of its capacity to meet the subject project's long and short term requirements. It was important that the software have specific features and capabilities like the following,

- i) a clear and familiar user interface (Windows? -based preferred)
- ii) learning/adaptation capability – to accommodate new knowledge with old
- iii) convenient means of acquiring knowledge – through either passive or active methods
- iv) convenient editing facilities – for easy knowledge revising and updating
- v) web-integration – Internet distribution capability
- vi) efficient – in terms of learning to use it, knowledge representation, and throughput
- vii) dynamic sizing – for knowledge expansion
- viii) cost-effective – reasonable base system and upgrade costs
- ix) reliable technical support
- x) plug-in functionality – for integrating external programs and applications software
- xi) symbolic and numeric processing – to allow different knowledge formats

Intellix A/S's Designer software [12,13] provided many (but not all) of the above capabilities. The domain knowledge, engineering materials, was used to address a wide range of compressor seal applications in the natural gas and petroleum/chemical industries. The tool selected fit the company's needs well here because it provided a qualitative, non-intimidating representation system to preserve the knowledge base of the company's experts. It relies on an n-tuple based neural net and cross-validation techniques to condition inputs used for its training [14]. A case-based reasoning (CBR) approach is used to teach the neural net a set of

examples that codifies the expert's knowledge [15]. Examples are obtained from the expert in passive or active knowledge-acquisition modes. They are stored as factors (questions), values (answers), and results (answers to a set of factors). Knowledge documents are defined to contain the examples. The documents can be linked to other knowledge documents to allow chaining of related sub areas of the knowledge domain. Specific material selections defined by the expert form the results made available at the end of a series of questions. Links can also be created between factor values so that related supplemental questions contained in other knowledge documents can be used. The material recommendation results are important pieces of information that guide the company's business plan with regard to filling and obtaining orders for their products and services. Historically, this information was contained in scattered internal documents, handbooks [16,17], and the minds of their senior-level technical employees.

The resulting small-scale system described next, is suitable for providing knowledge relevant to technical support of material recommendations. A case study is provided focusing on its development while highlighting key features and specific aspects of the knowledge engineering process. Key benefits were obtained through identification, preservation, and centralization of the knowledge needed to maintain and advance business interests.

4. DOMAIN STRUCTURING AND KNOWLEDGE ACQUISITION

Prior to capturing and representing expertise in a knowledge tool the knowledge domain must be known and be accessible. Knowledge sources must be located and, if possible, gathered together. Once assembled, the task of structuring the domain can begin. A blueprint is needed to guide the knowledge engineer for mapping the domain to a useful representation. This framework can be any relevant organizational tool that assists and clarifies the mapping process, such as an outline of the knowledge base, a conceptual sketch that breaks down the domain into manageable pieces, flowcharts, logic maps, or hierarchical tree-structured diagrams. This creates a starting point for the knowledge engineering process and serves as the roadmap for all subsequent development and implementation. Like any viable engineering design it will, of necessity, undergo revisions as development proceeds. Figure 3 gives the initial tree structure used to clarify the scope of the domain knowledge needed to select materials for the company's seal products. Three dominant application characteristics were identified as essential to picking suitable materials: (1) cylinder and rod conditions, (2) gas composition, and (3) compressor operating conditions.

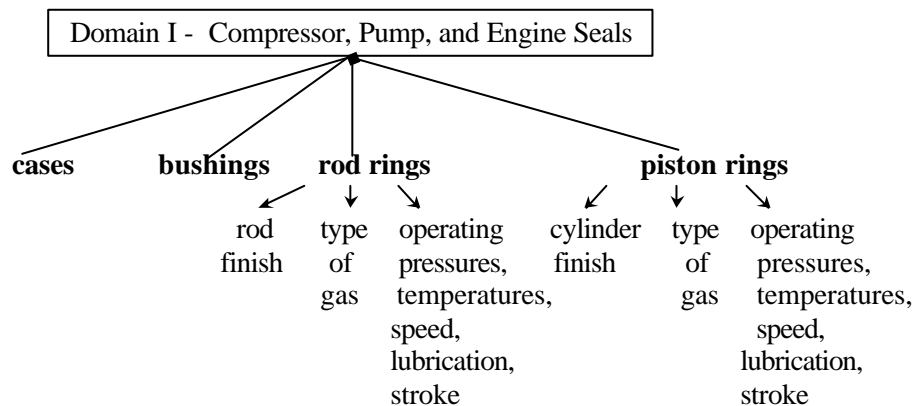


Fig. 3 – structure of material selector knowledge domain.

The next step was to acquire the knowledge critical for determining the best available materials for piston and rod rings used in reciprocating compressors. These two sealing product lines were considered similar in function and were combined in the same domain. The knowledge process has historically meant interviewing and probing an expert, or experts, to obtain their decision-making reasoning and results. Typically this has been the bottleneck in the knowledge engineering process [18] which, depending on the expert, has required the most time and effort. This led researchers to pursue more efficient ways for acquiring an expert's knowledge, such as automated knowledge elicitation (transference of knowledge via dialogue with the machine), and machine learning (through examples). Here, the knowledge tool used provided two modes for knowledge acquisition: (1) the input of case examples thought up by the expert, and (2) a passive knowledge-acquisition mode in which the tool "interviews" the expert and requests specific information for use as examples. In the latter case, the tool is typically used to identify and fill knowledge gaps in the example set provided in the first case.

The knowledge was acquired through numerous interviews and discussions with the expert, resulting in the definition of critical factors, values, and material selections input to the tool. The structuring of the knowledge domain required several revisions after its initial conception (Fig. 3). Due to the quite-natural and iterative process of deciding which knowledge is critical and identifying how it is used, stops and starts are inherent in the process. It was the most nonlinear at the start of knowledge acquisition, when the nature of the task was newest to both knowledge acquirer and acquiree. It also resulted in veering away from initial assumptions about the breadth and depth of the knowledge domain itself, which proved to be wider and deeper than originally thought. Discovery of the true scope of the knowledge domain came as a small surprise to both expert and knowledge engineer.

As the system evolved, revelation of the expert's decision process, necessitated the need to incorporate a computational model as a partial basis for selecting materials. The information needed to determine the application's underlying wear factor (a critical determiner of suitable materials) was reduced to a proprietary set of calculations based on specific physical and material parameters and the expert's heuristics. Since the knowledge tool relied heavily on use of a qualitative knowledge representation, additional mapping of quantitative information to ranges of qualitative factor values was required. This is viewed as a shortcoming of the knowledge tool that could and should be remedied in later versions³.

5. KNOWLEDGE EXAMPLES

Each example used in the knowledge base represents pertinent questions and answers that lead to a desired material selection. The questions and answers are stored as sets of factors and their possible values. The knowledge domain can be structured to partition examples into sets of knowledge contained in distinct knowledge documents. Each knowledge document can represent separate, but related, areas of questioning. Editors are available to create, structure, and link factors and results between knowledge documents, as needed, to define the knowledge domain. The document structure is formatted to resemble the familiar Windows® directory/folder structure as shown in Figure 4. Four documents are shown with two of them expanded to reveal their factors (Friction document) and values (Application document). A partial view of the examples defined for the Friction knowledge document are shown on the right side of the Figure. Some of the example factors (questions) are given at the top of each

³ Designer 4.0 has reportedly expanded its functionality to include provisions for external application program plug-ins.

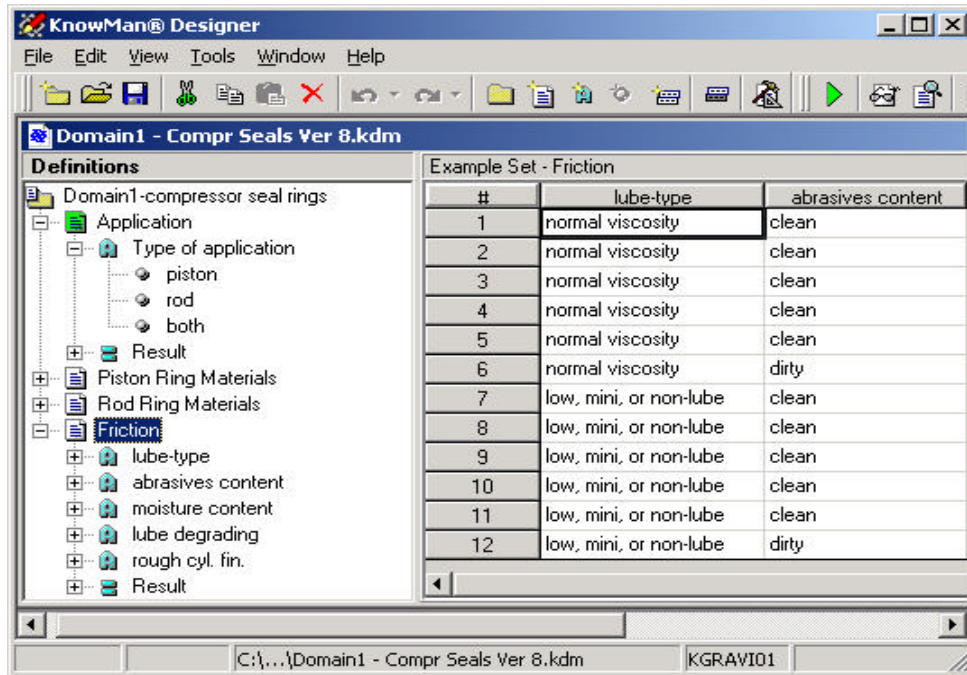


Fig. 4 – material selector knowledge document structure.

column, while the rows contain the values (answers). Results (material selections) are not shown, but are listed in a separate column to the far right for each of the examples.

Testing of the example set can be done as the knowledge base is entered. The examples serve the function of production-rules typically used in rule-based systems. Such rules can be cumbersome and difficult to define if the knowledge domain is heavily layered with decision-making. An example of a production-rule that would be needed to represent the function provided by the Friction level knowledge document (shown at the bottom of Fig. 4) is,

```

If (lubrication = normal viscosity) Then
  if (abrasives = clean) and (moisture = dry) and not (lube-degrading = y) and (finish = n) then
    Friction level = low
  endif
  elseif (abrasives = clean) and (moisture = wet) and (lube-degrading = y) or (finish = y) then
    Friction level = normal
  endif
  elseif (moisture = bone dry) or (abrasives = dirty) then
    Friction level = high
  endif
Else
  if (abrasives = clean) and (moisture = dry) and not (lubrication = y) and (finish = n) then
    Friction level = normal
  else
    Friction level = high
  endif
Endif;

```

Fig. 5 – production rule to determine friction level

Activating the examples in the tool initiates the neural net training process in which it learns by example to produce desired results. The results are given a confidence level (low, medium, high, and match or 100% confidence) during testing of the system, which is an indicator of the completeness of the example set for producing reliable material selections. Each result has a confidence rating and the developer can view it during testing. The example space is filled (complete) when the number of examples provided equals the total number of possible combinations of all the factor values. The highest confidence results (i.e., a perfect match with an example) occur most frequently in example sets having the maximum number of examples to cover all the combinations of factors and values. However, in many cases the result of an example does not depend on a unique answer to each particular question. This is because examples can be defined such that the outcome (i.e., result) is the same, regardless. The tool allows specifying a “don’t care” value for these factors which can significantly reduce the number of examples needed for training the system to yield acceptable results. This was the case for the Friction level document which required 48 examples (2 lubrication values x 2 abrasive values x 3 moisture values x 2 lube-degrading values x 2 finish values) to complete its example space.

Four examples from the complete example set are shown in Table 1. The “*” represents a “don’t care” answer. Note that in these cases, the expert has defined the friction level as high regardless of cylinder or rod surface finish. Therefore the number of examples can be reduced to one using “*” for the lube-degrading and finish factors in these cases. Examples can also be deactivated which precludes them from being used in neural net training.

TABLE 1 – redundant examples from the piston ring knowledge document

Ex. #	lubrication ?	abrasives ?	moisture ?	lube-degrading ?	finish ?	friction level
25	low, min, non	clean	wet	y es (*)	y es (*)	high
26	low, min, non	clean	wet	y es (*)	n o (*)	high
27	low, min, non	clean	wet	n o (*)	y es (*)	high
28	low, min, non	clean	wet	n o (*)	n o (*)	high

In addition to this, the neural net can be configured for optimal learning from the example set and is capable of returning results with medium to high confidence levels with only 10% of the example space filled. Conflicts in the example set occur when two or more examples with the same values are given a different result. These examples can be revised or removed to eliminate the conflict. Or the conflict can be allowed, lowering the confidence in the resulting material selections. Table 2 shows the example set sizes and number of actual examples used in each of the knowledge documents with no example conflicts.

TABLE 2 – example set sizes

knowledge document	example set size	actual examples used	% completeness
Friction level indicator	48	12	25
piston ring selector	64	64	100
rod ring selector	192	109	56.8

The Friction Indicator required only 12 examples (out of a possible 48) to sufficiently define the example space so that it would give material selections that were a perfect match (100% confidence) to the answers for the given questions (factors). The piston ring materials Selector required a full example set of examples (64). For the rod ring materials Selector a 57% complete set of examples was required to obtain high confidence results. The percentage of completeness reflects the complexity of the knowledge and reasoning needed to make material selections by the respective documents.

6. RESULTS

Testing of the examples sets and results provides a look at several key features of the knowledge system: example conflicts, result confidence ratings, relative significance of results, factor weighting, and knowledge redundancy and gaps. Figure 6 illustrates some of these features. It gives typical results from the tool's test window for the piston ring Selector at the end of querying. The material selections are given on the left hand side in the form of numeric designations. Only three 1st/2nd material combinations would be needed for any particular application, as specified by the expert. The tool's input tracking function was toggled on so that it decided the optimal order of questioning that would lead to the quickest result. The bar graph indicates that a perfect match (100%) was found in the example set for the user's answers for materials 05-0/09-0. The questions and answers are given at the bottom of the window. Of the six questions provided for selecting a material, only five were used by the system. It had determined that the question related to the compressor's stroke length was superfluous and did not influence piston ring material selection. The bars highlighted for the other two material combinations show their weighting relative to the winning result's; 51% for 09-0/05-0 and 49% for 05-0/05-1. The "rod" result is only a link to the rod seal document and provides a jump to the questions prepared for selecting rod ring materials.

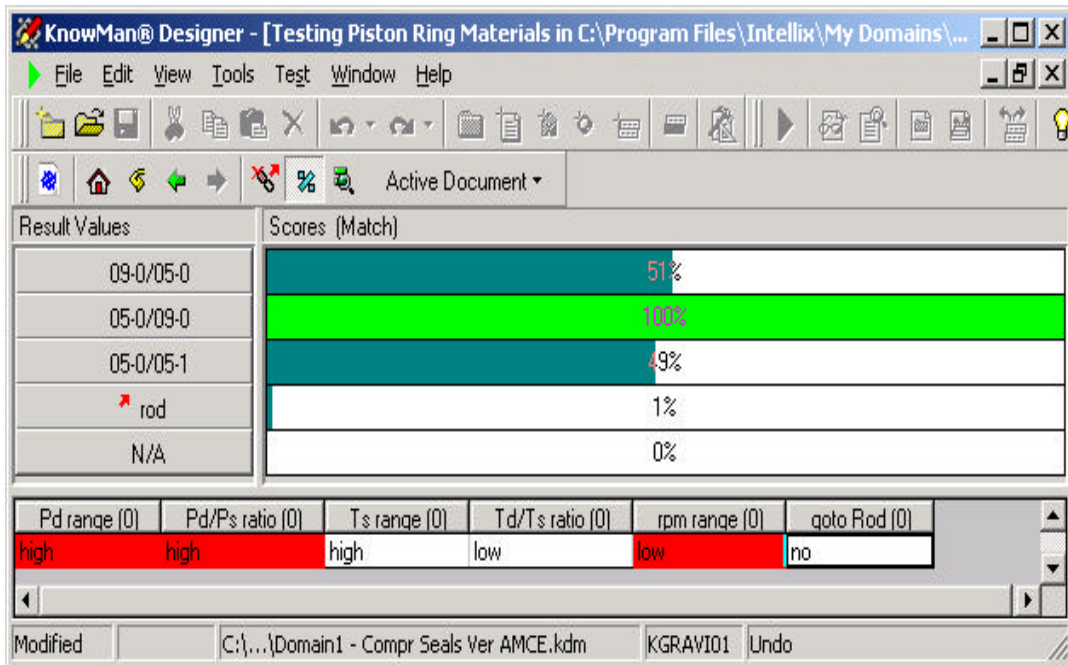


Fig. 6 – test results for the piston ring material selector.

Testing is done on a step-by-step basis. As the questions are answered, the knowledge engineer can follow the status and progress of the system as it leads to a result. Initially, before the first question is answered, the test window exhibits a bar graph showing the result with the most input examples having the highest percentage. As the questioning progresses, the bar graph fluctuates to reflect the percentages of the competing results. The result that is “in the lead” will have the highest percentage at the time, but this does not guarantee it will “win” the race to a result. That is decided by the user’s answers for the remaining questions. The questions listed at the bottom of the window are given a numeric score before they are answered. This score indicates how much the answer to this factor will improve the confidence of the results. When a result is provided the factor scores all become zero. For the piston ring material selector the results are all directly related to examples in the small sample set, and thus are all of high confidence.

The rod ring material Selector was based on a more complex example set. It’s knowledge document also required linking to the Friction level document prior to querying the user for compressor operating condition information (this additional information was originally Inked to the piston ring document, but the tool determined it was not needed). The knowledge system required the use of all six operating condition questions. To show the complexity of selecting a rod ring material, Fig. 7 shows its results distribution, as provided by the tool’s document analyzer. It can be seen that, of the 11 material results, most examples refer to the material designated as 05-02/09-01. This does not diminish the importance of the other results and examples however, as they are also needed for other applications in this domain.

Figure 8 gives additional statistics provided by the tool that indicate the usage of the example space. The graph shows that over 99% of the example space is covered (i.e., the examples provided will cover nearly all of the possibilities defined for the given results) even

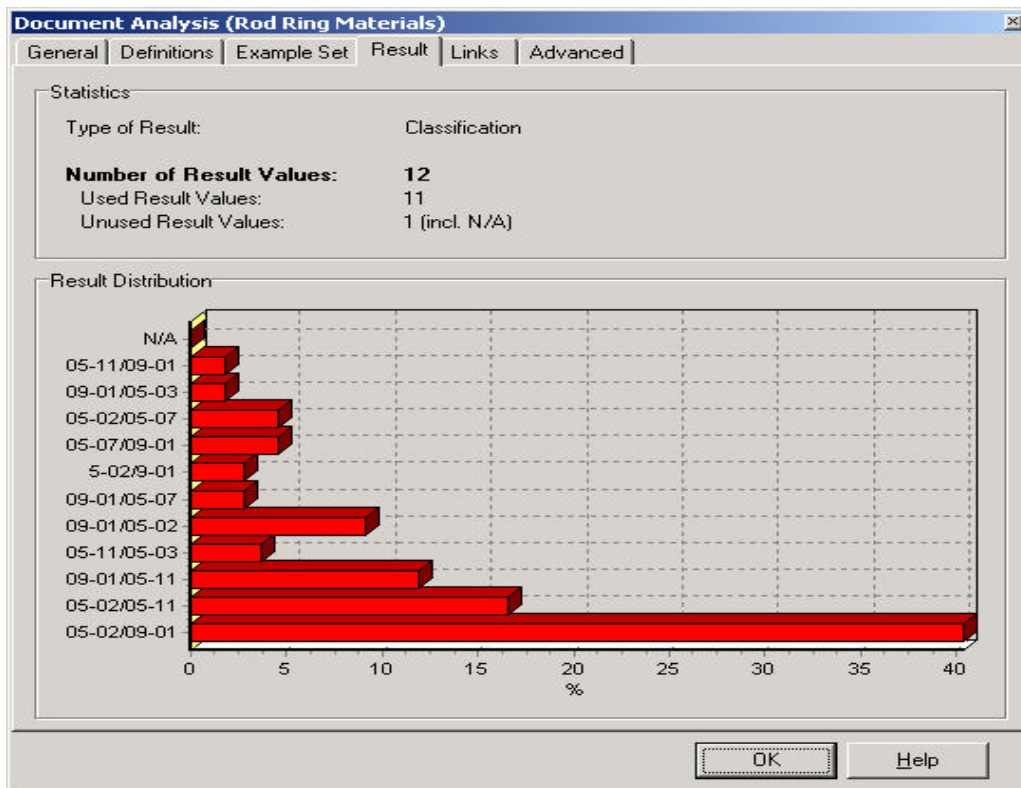


Fig. 7 – results distribution for the rod ring material selector.

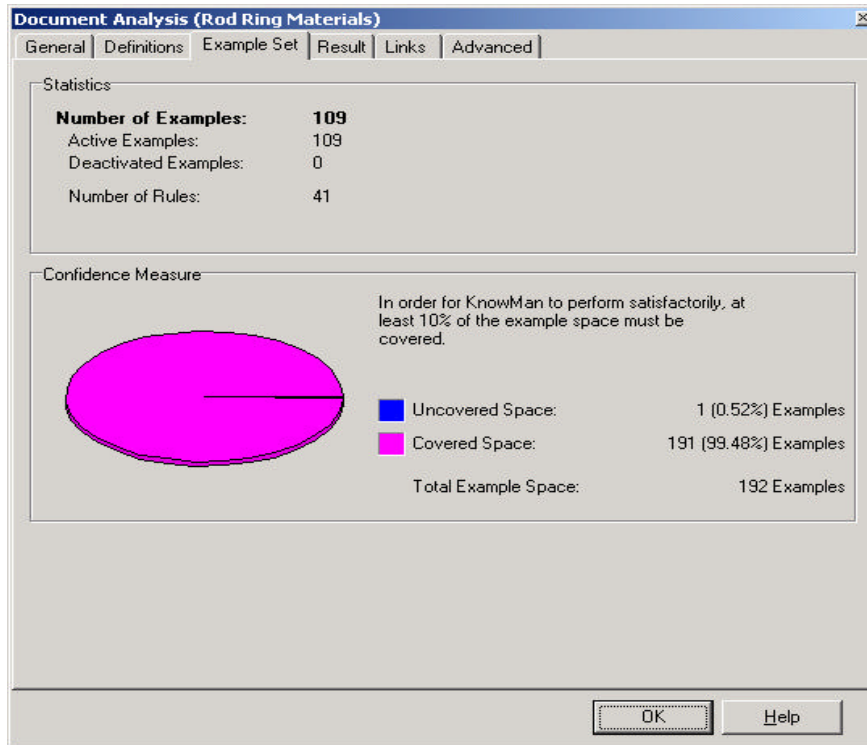


Fig. 8 – example-space data for the rod ring material selector.

though only 109 examples out of a possible 192 are provided in the example set. High confidence is thus also assured for the rod ring material selections. The example space for the Friction level indicator document is 100% covered. The tool also performs optimization of the factors associated with the example set, and reports on the information-gain achieved as a

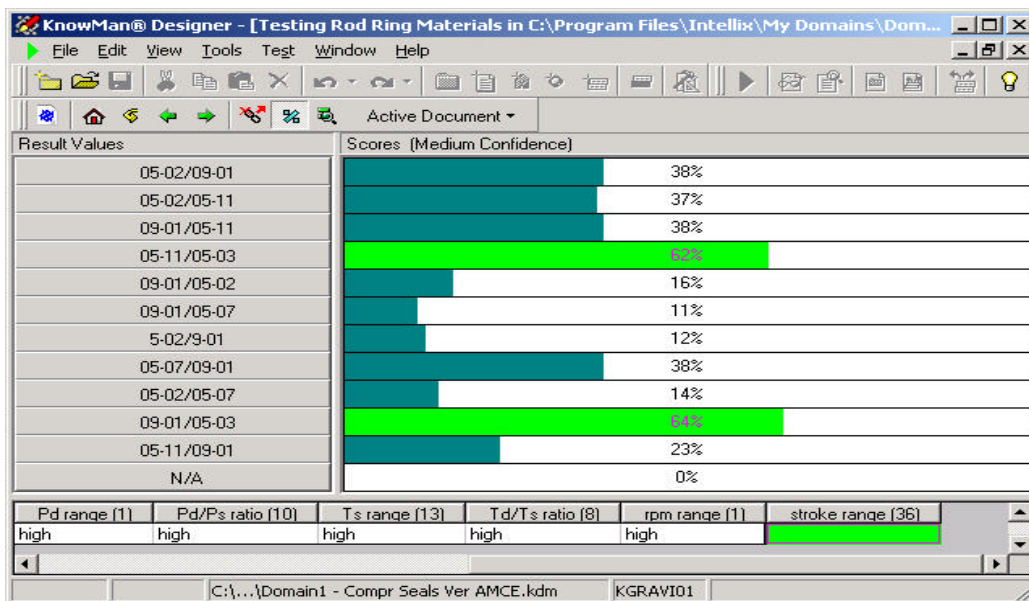


Fig. 9 – rod ring document test status, midway to a result

result of cross-validation, factor deactivation, and fine tuning of the neural net training examples. A typical testing session for the rod ring document is shown Figure 9. The testing is one question away from a final selection and shows that there are two viable material combinations at this point in the questioning. Depending on the application's stroke range, the knowledge tool would subsequently choose either 05-11/05-03 (for a low stroke range) or 09-01/05-03 (high stroke range) as the final selection.

A user-interface is provided apart from the implementation screens shown here. The interface can include additional explanation text and graphic displays to assist the user in navigating the system and in clarifying the questions and results.

In its current state the knowledge system has received preliminary approval by the primary expert involved in its development. Further examination and review by additional company experts awaits it before final approval is granted. Once qualified, this knowledge "stub" will serve as the backbone for further development of other proprietary knowledge bases. This includes the development of a seal ring style Chooser and an engineering dimension Recommender. Beyond that there is the potential for distributing the expertise across the Internet for which the tool has built-in capability and functionality.

7. CONCLUSIONS

A relatively simple, small-scale knowledge system has been developed using an available, off-the-shelf knowledge tool. In addition to the benefits gained from preserving the company's technical expertise, centralization and documentation of critical knowledge was stored as a resource for technical, training, sales, and marketing functions. The knowledge engineering process itself brought several tangible benefits which included defining the knowledge and finding knowledge gaps where none were previously thought to exist. The discovery that friction-level knowledge and stroke-range doesn't affect piston ring material selection is an example. The quantification of the expert's knowledge using material wear calculations also uncovered issues that may deserve more technical attention and validation such as material wear data.

Although the knowledge tool lacked specific features the author felt would make it more useful, it was found to have the features that made it a good fit for capturing and implementing the required knowledge domain. Its ease of use, familiar interface, and feature-content assisted the knowledge engineering effort. The capability for Internet distribution of the results were not explored here but this is a critical capability needed by any commercially-viable knowledge tool. As noted earlier, the elucidation of the expertise by an expert can be a challenging, time-consuming process. A good and powerful knowledge tool, along with the requisite willing and pliable expert and knowledge engineer, are essential ingredients for a successful effort. The results given are intended to demonstrate that small-scale knowledge engineering is no longer only an option for larger corporations, but is an important and achievable endeavor for small businesses and organizations.

8. ACKNOWLEDGEMENTS

Special thanks is due Paul C. Hanlon, technical engineering expert at the C. Lee Cook Company, Louisville, KY. He provided expertise and guidance for development of the knowledge system. Thanks also to Don York, Vice President of C. Lee Cook for having the vision and providing resources that made its development possible. James Lake of Intelx A/S, Copenhagen, Denmark, and Mark Klein of the Tacit Knowledge Corporation,

Minneapolis, MN, supplied valuable technical assistance and insight for the system's implementation.

9. REFERENCES

1. Barr A. and Feigenbaum A. E., eds (1981), *The Handbook of Artificial Intelligence*, Vol. 1, Los Altos, CA: Morgan Kaufmann.
2. Buchanan B. G., Barstow D., Bechtel R., Bennet J., Clancey W., Kulikowski C., Mitchell T. M., and Waterman D. A. (1983), Constructing an expert system. In *Building Expert Systems* (Hayes-Roth F., Waterman D. A. and Levat D., eds), Ch. 5, Reading, MA: Addison-Wesley.
3. Winograd T. (1972), Understanding natural language, *Cognitive Psychology*, 1, 1-191.
4. Fikes R. E. and Nilsson N. J. (1971), STRIPS: a new approach to the application of theorem proving to problem solving, *Artificial Intelligence*, 2, 189-208.
5. Shortliffe E. H., Scott A. C., Bischoff M. B., van Melle W. and Jacobs C. D. (1981), ONCOCIN: an expert system for oncology protocol management. In *Proc. 7th International Joint Conference on Artificial Intelligence*, pp. 876-81.
6. Shortliffe E. H. (1976), *Computer-Based Medical Consultations: MYCIN*, New York, Elsevier.
7. Prerau D. S. (1990), *Developing and Managing Expert Systems*, Reading, MA: Addison-Wesley.
8. Bobrow D. G. and Winograd T. (1977), An overview of KRL, a knowledge representation language, *Cognitive Science*, 1(1).
9. van Melle W. J. (1981), *System Aides in Constructing Consultation Programs*, Ann Arbor, MI: UMI Research Press.
10. Musen M. A., Gennari J. H. and Wong W. W. (1995), A rational reconstruction of INTERNIST-I using PROTÉGÉ-II, Knowledge Systems Laboratory, Medical Computer Science, KSL -95-46.
11. Davis R. (1980b), Application of meta-language knowledge to the construction, maintenance and use of large knowledge bases. In *Knowledge-Based Systems in Artificial Intelligence* (Davis R. and Lenat D., eds) pp. 229-490, New York: McGraw-Hill.
12. KnowMan? Basic Suite, Introductory Paper (Copyright 1993-2000), Intellix A/S, Copenhagen, Denmark.
13. KnowMan? Basic Suite, White Paper (Copyright 1993-2000), Intellix A/S, Copenhagen, Denmark.
14. Linneberg C. and Jorgensen T. M., (1998), *Cross-validation techniques for n-tuple based neural networks*, Intellix A/S, Copenhagen, Denmark.
15. Jackson P. 3rd ed (1998), *Introduction to Expert Systems*, Ch. 22, Reading, MA: Addison-Wesley.
16. York D. F., (Rev. 1999), *Non-Metallic Selection Guide*, Control No. 36, C. Lee Cook, Louisville, KY.
17. Hanlon P. (2001), *Compressor Handbook*, McGraw-Hill, New York, NY.
18. Feigenbaum E. A. (1977), The art of artificial intelligence: themes and case studies of knowledge engineering. In *Proc. 5th International Joint Conference on Artificial Intelligence*, pp. 1014-29.