THE DIPMETER ADVISOR SYSTEM
A Case Study In Commercial Expert System Development

Reid G. Smith James D. Baker

Schlumberger-Doll Research
Old Quarry Road
Ridgefield, Connecticut 06877

ABSTRACT

We discuss commercial expert system development, using our experience with the *Dipmeter Advisor* system as a case study. While the data is too sparse for definitive results, several ideas have emerged as important and suggestive as guidelines for subsequent commercial expert system undertakings.

During the past four years, the *Dipmeter Advisor* system has migrated from an initial experiment in application of expert system techniques in well-log interpretation to a candidate commercial interpretation system. The system has undergone substantial change: It has been implemented in different configurations, in different languages, and on different computer systems. Our ability to experiment with the system has been greatly enhanced by the tools and ideas of rapid prototyping. We have also observed an oscillation in thrust over time between *(i)* expansion and change in the domain knowledge, and *00* selection and design of appropriate expert system tools. Finally, we have found several of the maxims of expert system development to be valid, but question a number of others.

1. INTRODUCTION

The past decade has seen the development of a number of expert systems, mostly by AI researchers for use in research environments. To date, few have been utilized for *industrial* applications. As a result, we have little experience with which to characterize either the nature of commercial expert systems or their development process.

The *Dipmeter Advisor* system is the result of a four year effort by Schlumberger to apply expert systems technology to problems of well-log interpretation. We have observed during this effort that the development of a commercial expert system imposes a substantially different set of constraints and requirements in terms of characteristics and methods of development than those seen in the research environment.

This paper is intended as a case study. We briefly describe the dipmeter interpretation problem and the evolution of the *Dipmeter Advisor* system. During its development a number of ideas have surfaced which we believe to be characteristic of this type of effort, given the current state of the technology. While the data is too sparse for definitive results, these ideas are thought to be important and suggestive as guidelines for subsequent commercial expert system undertakings.

2. THE PROBLEM

Oil-well logs are made by lowering tools into the borehole and recording measurements made by the tools as they are raised to the surface. The resulting logs are sequences of values indexed by depth. Logging tools measure a variety of petrophysical properties. The *dipmeter* tool in particular measures the conductivity of rock in a number of directions around the borehole. Variations in conductivity can be correlated and combined with measurements of the inclination and orientation of the tool to estimate the magnitude and azimuth of the *dip* or tilt of various formation layers penetrated by the borehole (Figure 1.).

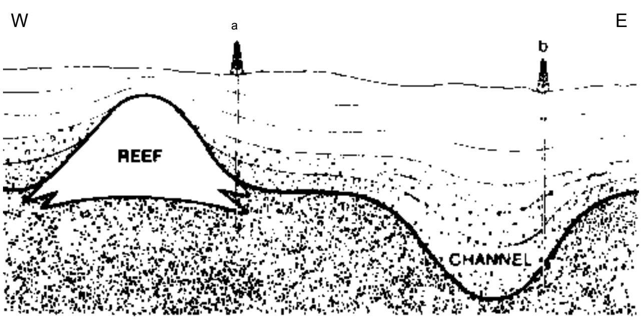

Figure 1. Dip in Subsurface Formations

Because the dipmeter tool has high resolution in the vertical direction (0.1-0.2 in.), it provides the petroleum geologist with detailed information on relatively fine-structured sedimentary beds. This type of information is invaluable in defining hydrocarbon reservoir structure and designing methods to drain such reservoirs.

Knowledge of the dip variations as a function of depth in the vicinity of the borehole does not in itself identify geologic features. However, when combined with knowledge of local geology and rock properties measured by other logs (e.g., lithology [sand, shale, …]), the characteristic dip patterns *(signatures)* of geologic events in the depositional sequence can be interpreted.

The right channel of Figure 2 is an interval of a dipmeter log. Dip estimates are shown as *tadpoles.* Dip magnitude increases to the right of the graph, and the down dip direction is indicated by the tail on each tadpole. The vertical axis is depth. (Hollow tadpoles indicate lower confidence dip estimates than solid tadpoles.) The left channel is a gamma ray log. (It measures natural gamma radiation in the formation-a rudimentary lithology indicator.)

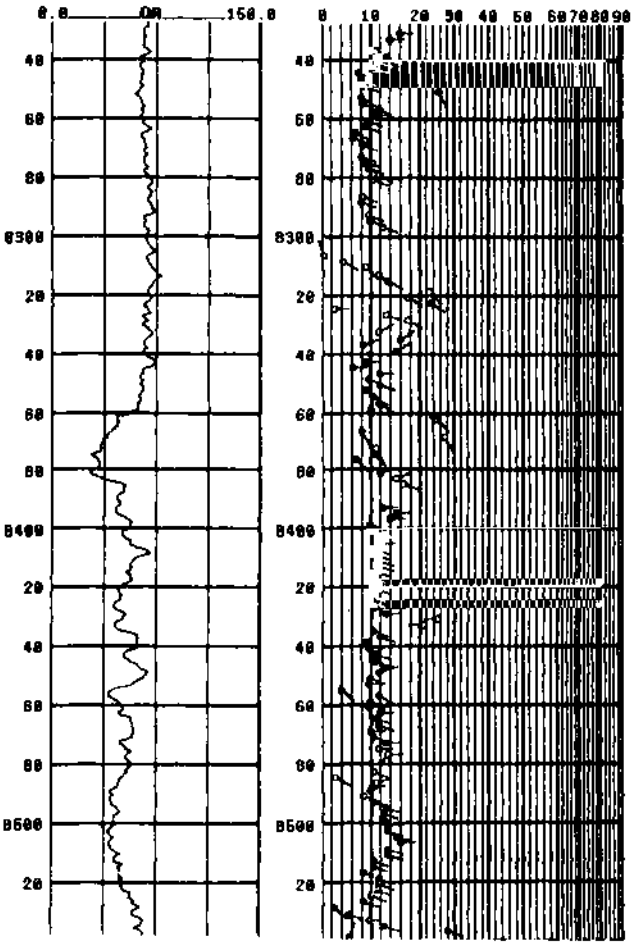

Figure 2. Gamma Ray and Dipmeter Logs

Sequences of tadpoles can be grouped together in patterns. Three of the characteristic dip patterns are described below [Schlumberger, 1981].

- Green Pattern: An interval (zone) of constant dip magnitude and azimuth. This pattern is characteristic of structural dip—caused by large-scale tectonic disturbance that occurs long after deposition and compaction of sediment.
- Red Pattern: A zone of increasing dip magnitude with constant azimuth over depth. This pattern is indicative of down dip thickening, which may be associated with distortions near structural features (e.g., faults), differential compaction of sediment over buried topographic features (e.g., reefs), or channel filling.
- Blue Pattern: A zone of decreasing dip magnitude with constant azimuth over depth. This pattern is indicative of down dip thinning, which may be associated with distortions near structural features, differential compaction beneath denser overlying deposits (e.g., sand lenses), or sediment transport by water or wind.

From this localized data, a skilled interpreter is often able to make comprehensive deductions about the geological history of deposition, the composition and structure of the beds, and the optimum locations for future wells.

3. SYSTEM OVERVIEW

The Dipmeter Advisor system attempts to emulate human expert performance in dipmeter interpretation. It utilizes dipmeter patterns together with local geological knowledge and measurements from other logs. It is characteristic of the class of programs that deal with what has come to be known as signal to symbol transformation [Nii, 1982].[1] The program is written in 1NTERL1SP and operates on the Xerox 1100 Scientific Information Processor (Dolphin).

The system is made up of four central components: (i) a number of production rules partitioned into several distinct sets according to function (e.g., structural rules vs stratigraphic rules); (ii) an inference engine that applies rules in a forward-chained manner, resolving conflicts by rule order; (iii) a set of feature detection algorithms that examines both dipmeter and open hole data (e.g., to detect tadpole patterns and identify lithological zones); and (iv) a menu-driven graphical user interface that provides smooth scrolling of log data.

Conclusions are stored as instances of one of 65 token types, with approximately 5 features/token, on a blackboard that is partitioned into 15 layers of abstraction (e.g., patterns, lithology, stratigraphic features). There are 90 rules and the rule language uses approximately 30 predicates and functions. The rules have the familiar empirical association flavor. A sample is shown below.

```
IF
    there exists a delta-dominated, continental-she(f marine zone, and
    there exists a sand zone intersecting the marine zone, and
    there exists a blue pattern within the intersection
THEN
    assert a distributary fan zone
        top — top of blue pattern
        bottom <— bottom of blue pattern
       flow — azimuth of blue pattern
```

The system divides the task of dipmeter interpretation into 11 successive phases as shown below. After the system completes its analysis for a phase, it engages the human interpreter in an interactive dialogue. He can examine, delete, or modify conclusions reached by the system. He can also add his own conclusions. In addition, he can revert to earlier phases of the analysis to refer to the conclusions, or to rerun the computation.

1. Initial Examination: The human interpreter can peruse the available data and select logs for display.
2. Validity Check: The system examines the logs for evidence of tool malfunction or incorrect processing.

1. Early versions of the program are described in (Davis, 1981], and [Gershman, 19821.

2. This sample is similar to the actual interpretation rule, but has been simplified somewhat for presentation.

3. Green Pattern Detection: The system identifies zones in which the tadpoles have similar magnitude and azimuth.
4. Structural Dip Analysis: The system merges and filters green patterns to determine zones of constant structural dip.

\+ 5. Preliminary Structural Analysis: The system applies a set of rules to identify structural features (e.g., faults).

6. Structural Pattern Detection: The system examines the dipmeter data for red and blue patterns in the vicinity of structural features.[3]

\+ 7. Final Structural Analysis: The system applies a set of rules that combines information from previous phases to refine its conclusions about structural features (e.g., strike of faults).

8. Lithology Analysis: The system examines the open hole data (e.g., gamma ray) to determine zones of constant lithology (e.g., sand and shale).

\+ 9. Depositional Environment Analysis: The system applies a set of rules that draws conclusions about the depositional environment. For example, if told by the human interpreter that the depositional environment is marine, the system attempts to infer the water depth at the time of deposition.

10. Stratigraphic Pattern Detection: The system examines the dipmeter data for red, blue, and green patterns in zones of known depositional environment.

\+ 11. Stratigraphic Analysis: The system applies a set of rules that uses information from previous phases to draw conclusions about stratigraphic features (e.g. channels, fans, bars).

For the phases shown above, "+" indicates that the phase uses production rules written on the basis of interactions with an expert interpreter. The remaining phases do not use rules.[4]

Figure 3 shows a sample Xerox 1100 screen following the stratigraphic analysis phase. On the extreme right the system displays a summary log of dip magnitude for the entire well. The black box indicates the region of the well that is expanded in the second window from the right. This window shows the dipmeter data together with the deviation of the borehole itself. The next window displays two other logs, GR (gamma ray) and ILD (a resistivity log).

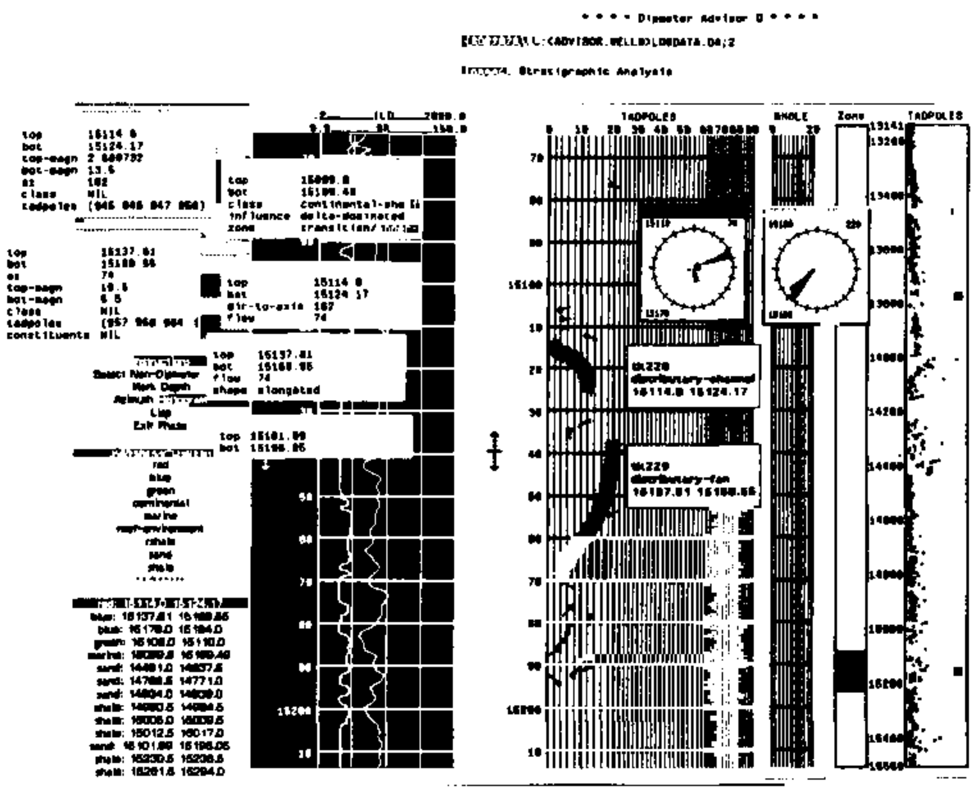

Figure 3. Stratigraphic Analysis Interaction

3. The algorithms used by the system to detect dip patterns are beyond the scope of this paper. It is worth noting, however, that the *textbook* definitions given earlier do not provide sufficient specification. The problem is complicated by local dip variations and occasional gaps in the data.

4. The rules obtained to date are due to J. A. Gilreath of Schlumberger Offshore Services, New Orleans, LA. The feature detectors and signal processing algorithms were written independently by project members.

The system summarizes relevant conclusions in the (scrolling) windows in the lower left hand part of the screen. The user (a dipmeter interpreter) has selected a number of conclusions to be examined in greater detail and shown as annotations on the dipmeter log. Also shown is the dip azimuth trend before and after structural dip removal.[5]

4. ON COMMERCIAL EXPERT SYSTEMS

In this section we summarize some of our observations on the development and evolution of the *Dipmeter Advisor* system. We discuss the nature of commercial expert systems as we see them in the coming decade, characteristics of the evolution process, development methods, and finally we compare our experience with some of the *traditional wisdom* of expert system development.

4.1 EMBEDDED SYSTEMS

Domain practitioners don't care which methods are used to help them solve their problems. What matters is utility and performance. Indeed it is unlikely that traditional AI methods alone will solve real problems. They are likely to be augmented by techniques from signal processing and pattern recognition, to name but two possibilities. This implies that the knowledge engineer involved in commercial expert system development must be prepared to solve problems that involve a variety of disciplines and techniques.

Thus it is our view that the expert system kernel is likely to be only a (perhaps even relatively small) component embedded in a larger system. The particular suite of problems common to signal understanding problems may, of course, bias our outlook, but we believe that it is difficult to avoid the conclusion that acceptance and real use of expert systems depend on far more than a knowledge base and inference engine.[6]

Indeed our experience has been that these traditional parts of an expert system are not the predominant parts of the overall system either in terms of the amount of code or the resources required for system development. It is instructive in this regard to examine the relative amounts of code devoted to various functions in the *Dipmeter Advisor* system:

Inference Engine:	*8%*
Knowledge Base:	*22%*
Feature Detection:	*13%*
User Interface:	*42%*
Support Environment:	*15%*

This breakdown cannot be used, of course, as a direct measure of programming effort or as an indicator of where the system gets its power. It does, however, demonstrate the importance of a good programming language, given the relative amount of code that is devoted to the user interface. It also indicates that traditional programming skills will still be required for the development of commercial expert systems.

5. The scrolling graphics code was written by Paul Barth. Extensions to the INTERLISP-D menu package were written by Eric Schoen.
6. Oaschnig has made a similar observation in the context of the PROSPECTOR system [Oaschnig, 1982].

4.2 SYSTEM EVOLUTION

Based on our experience, we hypothesize an oscillating focus of attention in commercial expert system development projects. Initially, the focus is demonstration of feasibility; acquiring the knowledge for a constrained problem and finding the appropriate set of expert system tools with which to encode and apply the knowledge. This phase could be relatively short. It is followed by a phase of expansion of the domain knowledge-during which the expert system tools remain relatively constant. There will likely come a point at which the intial tools do not provide sufficient power to allow continued expansion of the system's expertise. At that point, the focus will move away from domain problems and towards selection-more likely development-of new expert system tools. Once a new set of more powerful tools has been constructed, then the focus will again return to the domain problems at hand.

Naturally any particular system may not pass through very many of these oscillations. The focus in the RI project, for example, didn't appear to oscillate at all [McDermott, 19811. We believe this is due to the nature of the task. There was little of the uncertainty about the nature of the problem that is evident in the the signal understanding or diagnosis tasks. Consequently the initial tools were in fact sufficiently powerful to handle the problem.

In the MYCIN project we seem to be observing the beginnings of an oscillation. The initial system was constructed. Then the rule base was expanded, leaving the initial expert system tools intact. More recently a new design, NEOMYCIN, has appeared--a new set of tools [Clancey, 1981].

Along with the oscillating focus, we hypothesize a performance vs time curve. We expect this curve to show periods of high positive slope-corresponding to implementation of new expert system tools, followed by periods of lower slope-corresponding to expansion of domain knowledge, followed by periods of level or even decreasing slope—corresponding to reaching (or surpassing) the amount of domain knowledge and generality that can be supported by the tools.

At any given point in time, then, an expert system will suffer from two kinds of weakness, due to *(i)* insufficient domain knowledge; and *(ii)* inadequate expert system tools. Just as the focus of the project will vary, depending on which of the two types of weakness is most troublesome, the type of person required to improve the system will also vary.

Improvements in the first area can be made to a large extent by people primarily knowledgeable in the domain, but not necessarily knowledgeable in the design of expert systems.[7] For example, the *Dipmeter Advisor* system is familiar with a relatively small number of different lithologies. The performance of the system could be improved in this area without redesign. Similarly, the coverage of the rules could be extended to handle more environments, or specialized to handle local anomalies.

7. We have already noted, however, the likelihood that traditional programming skills will continue to be required.

Weaknesses due to inadequate expert system tools cannot be corrected without redesign. This type of effort requires a person who can actually build expert systems, as opposed to one who can use the framework to expand capabilities. For example, the *Dipmeter Advisor* system deals with uncertainty in a rudimentary way. It uses rule order to help circumvent potential multiple interpretations for the same interval in the well, or simply draws multiple conclusions for the same zone, leaving the problem to be sorted out by the human interpreter. Similarly, the system has a very local view of consistency in the vertical sequence. This is directly attributable to the fact that it is reasoning from sets of empirical rules and has no model of the underlying geological processes that lead to the rules. These deficiencies cannot be overcome without redesign.

4.3 SYSTEM DEVELOPMENT

We have attempted a critical review of the development side of the *Dipmeter Advisor* system. Although we are as yet unable to abstract a development methodology, several observations stand out. Almost every major issue and decision in the evolution of the *Dipmeter Advisor* system addressed one or more of the following.

1. *Demonstration of Feasibility*
2. *Demonstration of Utility and Performance*
3. *Evaluation of Utility and Performance*

Demonstration of Feasibility: The problem of dipmeter interpretation was initially selected as a vehicle for investigating the applicability of expert systems techniques to well-log interpretation. Until feasibility could be demonstrated, other questions were somewhat secondary.

As a first step, a substantial effort was expended on acquisition of dipmeter interpretation knowledge. This effort was carried out over a 12 to 18 month period using standard techniques (protocols, video tape, discussion, representative examples, and so on). A single expert was studied in detail, again adhering to standard practice.

The implementation of a prototype system followed data acquisition and was carried out in approximately four months (completed in December 1980). The rule Dase and inference engine were written in INTERLISP (245 Kbytes of source code) and ran on a DEC 2020. The user interface was graphical, written in FORTRAN (450 Kbytes of source code), and ran on a Ramtek 9400 connected to a VAX 11/780. The VAX and 2020 were linked via a CHAOSnet. The rule base was made up of aproximately 30 rules. There were also several feature detectors and signal processing algorithms.

Demonstration of Utility and Performance: The prototype system demonstrated to the expert that significant analyses were possible. To determine commercial viability, other issues must be addressed. Does the system solve enough of the problem to be interesting and useful? Can the system perform with the efficiency and interactivity necessary in a field environment without overutilizing available computing resources.

Two examples demonstrate the problem. The initial prototype, had no means of actually detecting the red and blue patterns and the lithology zones that are required to perform an unaided interpretation. It did not solve enough of the problem to be useful. This resulted in implementation of algorithms for simple detection of tadpole patterns and lithologic zones.

Second, the detection of green patterns and determination of structural dip took approximately 18 minutes in the first test well. This was unacceptable for actual use—later effort reduced the time to under 2 minutes.

Evaluation of Utility and Performance: This is the area of field evaluation, and several concerns exist here. First, is the rule base sufficiently complete to correctly solve a wide variety of problems in the geological environments for which it was developed. Second, what changes and effort would be required when working in other geological environments? And third, does the rule base sufficiently capture the thinking of enough dipmeter interpreters to be useful?

To date, this has been the most difficult area. The above questions need to be answered by people in the engineering and field groups. To accomplish this, the prototype system must be capable of operating in their existing environment-possibly upgraded with modest investment.

One of the difficulties with the initial prototype was the unusual architecture of linked computers, which was not a standard company configuration. In an effort to facilitate testing, the system was reimplemented in FRANZLISP (except for the graphical interface), totally on the VAX 11/780. Unfortunately this did not solve the problem. The VAX/Ramtek configuration, as a shared resource in a generally overloaded situation, required an excessively long time to complete a case. Under worst conditions, it took several hours. (In an unloaded VAX environment, it could be completed in one-half hour or less.)

At this point, new technology came to the rescue, and the system was re-implemented on the Xerox 1100, which has both a dedicated processor and sophisticated graphics. In this implementation the graphical interface code was integrated into the remainder of the system. The result was approximately 612 Kbytes of INTERLISP source code. This implementation was robust enough and fast enough to allow transfer to a Shlumberger Interpretation Engineering group for testing in a non-research environment.

4.3.1 RAPID PROTOTYPING AND TECHNOLOGY TRANSFER

In the beginning of a commercial expert system development project, it is important to demonstrate the feasibility of the system. Rapid prototyping seems to be an appropriate strategy—especially given the usual vagueness of the understanding of what can be accomplished.

8. Somewhat surprising, our field organization seems to be prepared to believe that with enough effort, the system can be made sufficiently *intelligent,* but that system efficiency must be closely monitored.

The main concern in such an approach is a flexible and powerful development environment. Traditionally, such an environment is not even closely related to the commercial computational environment. This leads to the problems noted above. With the advent of inexpensive personal workstations, however, there is real hope that the situation may be changing (as has been our experience with the *Dipmeter Advisor* system).

Significant questions still remain. One of the problems of rapid prototyping is that it provides a good start toward system development, but does not offer clear guidance on how to produce a well-engineered commercial product (see, for example [Sheil, 1983]). Traditionally this is viewed as a problem in technology transfer.

Our experience with the *Dipmeter Advisor* system may suggest a different strategy. We have seen an evolution through *successive refinement.* Through the different stages, functionality has changed, as well as the target systems, development environments, and personnel. Basically, what we have seen at each stage is the introduction of new features, the solving of old problems, and the consolidation of existing code which does not require substantial change.

Is it possible that the traditional transfer from research to engineering may involve successive releases, corresponding to successive prototypes? If true, then we must somehow convey to our engineering organizations a more accurate perception of the expected lifetimes of our prototypes. In addition, this methodology suggests early transfer to engineering rather than late, with the expectation that several such transfers will be made. Furthermore it forces the prototype designers to pay even more attention to user interfaces than our earlier figures would suggest. If the systems are going to be changing rapidly then they must have especially convenient and easy-to-learn interfaces.

Expert systems technology by its nature will be difficult to transfer. Such systems require skills that are possessed by a very small number of individuals. Furthermore, the rapid prototyping development methodology makes traditional transfer even more difficult-the systems are in a constant state of flux. As a result it is fair to say that for the foreseeable future, greater than normal responsibility will lie with the research organizations to ensure successful transfer.

4.4 SOME OBSERVATIONS ON THE TRADITIONAL WISDOM

For the remainder of this section we consider a number of *maxims* of expert system development in the light of our experience in the commercial environment. (See [Barstow, 1981], [Buchanan, 1982], or [Davis, 1982] for good summaries of the traditional wisdom of expert systems development.)

A common maxim of expert system development is that we should throw away the code for the *Mark-I* version of the system as soon as it demonstrates feasibility and get started on *Mark-IL* In the commercial environment, there is great reluctance to throw away code. As a result, a more likely scenario involves a series of progressive releases of the system to the expert and possibly to the engineering organization for development and use. The fact is that even though the knowledge engineer knows all too well the limitations of *Mark-I,* and even has ideas on how to overcome them, *Mark-1* may still provide some useful service. We do not yet know how to manage this type of progressive and evolutionary technology transfer.

It is well accepted that expert system development is an incremental process. Usually we understand this to mean that the performance of the system improves incrementally. There is, however, another kind of change that may occur; namely, our experts are themselves moving targets-partially as a result of the perspective gained through experience in expert system development! This has been apparent during the *Dipmeter Advisor* project. For example, we have seen an increasing geological awareness in our expert dipmeter interpreter. This has led to a series of changes in the way stratigraphic analysis is handled in the system. Not all of these changes have proved useful-the expert appeared to be using the program at times as a test bed for his own evolving ideas.

It is traditional wisdom that the task should be very carefully defined before the system is designed. Our experience has been that this is quite difficult. In consonance with our comments on the rapid prototyping development strategy, it is not clear that task definition can be done in a rigorous fashion. We suggest a contingent definition-one that is clear for a time, but can be easily changed. We should note that the evolving performance of the system itself at least partially fuels changes in the task definition.

It is generally accepted that construction of the *Mark-1* system should be commenced as soon as one example of the intended behavior is understood. We now believe that we spent too much time in knowledge acquisition before actually starting to build a system. This had the effect of slowing our rate of progress. We could not move forward in formalizing the knowledge that had been gained, because we could not demonstrate in concrete terms our understanding of it.

Some of the development team also deemed themselves to have acquired more expertise than was warranted. This is a natural tendency. It was partially due to infrequent interactions with the expert. More responsibility fell on the shoulders of the knowedge engineers to organize the domain knowledge than appears prudent. This infrequency also led to a problem of validation-how to be sure that we were on the right track. On a related note, we can testify to the necessity of an adequate set of generic examples with which to test the system as it evolves.

One piece of traditional wisdom might be questioned. It is common to deal with a single expert during the development of an expert system. The perceived danger is that it is difficult enough to capture what a single expert is doing, let

9. This is a good illustration of a conflict that can arise as a result of somewhat different goals of research and of development in expert systems. The former is concerned with continued exposition and machine implementation of human expert reasoning methods, while the latter is concerned with construction of products that utilize already understood and implemented methods.

alone a number of experts. In the particular context of dipmeter interpretation, however, it might have been useful to involve a number of different experts from the outset. We now understand that there are many schools of thought on the problem. There is also a variety of perspectives that can be brought to bear on it--dipmeter interpretation expertise and geological expertise are not necessarily co-located in the same person. While the rules for a first approach are most appropriately phrased by a dipmeter interpreter, we might have been well-advised to obtain the necessary geological vocabulary and structure from a geologist. In future systems, we will attempt to synthesize these overlapping points of view.

In a similar vein, we have noted a difficulty that can arise when a single expert is used *and* when he provides all examples with which to test the system. When working with familiar examples our expert does indeed appear to apply forward-chained empirical rules-kind of compiled inferences. Recently, however, we have participated in experiments with a number of interpreters (and examples) from around the world. During these experiments we noted that our expert resorted to a different mode of operation when faced with completely unfamiliar examples. He appeared to reason from underlying geological and geometric models-abandoning the rules. In some sense, this is of course to be expected. It was instructive, however, to actually document the change. We believe that dealing with multiple experts would have provided concrete evidence of this phenomenon much sooner in the life of the project.[10]

We have also noted a lurking danger in dealing with experts. It appears to be possible to give an expert a false sense of comfort with a particular formalism (e.g., rules). At times we had a sense that the expert was trying to make us happy by expressing what he was doing in terms of the rule framework we had offered—perhaps at the cost of accuracy. We would be well-advised to avoid over-reliance on the rule (or any other presently known) framework. We don't want to convince the expert that this simple idea covers everything he does, or that system failures are necessarily the result of incorrect or missing rules.

With regard to acceptance of the expert systems approach, our experience has been somewhat different from that of the RI designers [McDermott, 1981]; that is, for RI there was general relatively rapid acceptance of the ideas within the organization. From early in the project concerns revolved almost totally around *performance* and *utility* in the problem domain.

We have seen a substantial increase in the size of the rule base (approximately tripled) and the functionality required of the system before we could consider field evaluation. McDermott has described a similar experience with RI. The size of its rule base tripled during the *development* phase [McDermott, 1981].

The traditional wisdom notes the importance of early construction of a flexible user interface. For the *Dipmeter Advisor* system the interface is graphical. It has proved invaluable in testing and user acceptance. Furthermore, as has been noted elsewhere [Buchanan, 1982], expert systems that are actually used by people trying to solve problems in their own domains of interest (as opposed to being used by researchers as vehicles for experimentation with AI techniques) must pay particular attention to human interface issues. For the *Dipmeter Advisor* system, it was only after we constructed a personal workstation implementation that was flexible, robust, and fast that it became possible to seriously consider testing by the Schlumberger engineering organization.

One final observation worth noting relates to the impact of an expert system on the domain experts. As has been found in other applications of expert systems [Feigenbaum, 1980], the existence of an expert system is helping to identify the real knowledge used in the field-the kind of knowledge that is rarely found in textbooks. A program that captures some of it at least gives a concrete basis for comparing the methods of different experts. As Gaschnig has noted [Gaschnig, 1982], it can also help a group to reach some form of consensus. The *Dipmeter Advisor* system has stimulated .an examination of current dipmeter interpretation methods that promises to improve quality.

10. Actually seeing the change in reasoning was further complicated by the fact that our expert has extremely broad experience. Hence, rinding a completely unfamiliar example was quite difficult.

5. CONCLUSIONS

The current *Dipmeter Advisor* system has provided substantial demonstration of the feasibility of using expert system techniques in commercial well-log interpretation. Additional analysis and evaluation of the system will certainly further define the the strengths and weaknesses of its approach. The experience gained to date has also helped to suggest characteristics of commercial expert system development as well as properties of a development methodology.

Acknowledgements: The latest implementation is a product of the efforts of a number of people. They include: Dave Barstow, Paul Barth, Jim Callan, Roger Duffey, Tony Passera, Paul Pruchnik, and Eric Schoen. A number of helpful suggestions were made by Stan Vestal, and by Bob Langley and Bob Young (both with Houston Interpretation Engineering). Barstow, Vestal, J. A. Gilreath, and Tom Mitchell also made helpful suggestions on this paper.

REFERENCES

D. R. Barstow and B. G. Buchanan, *Maxims for Knowledge Engineering.* SDR AI Memo No. 10, Schlumberger-Doll Research, May 1981.

B. G. Buchanan, New Research On Expert Systems. In J. E. Hayes, D. Michie, and Y-H Pao (Eds.), *Machine Intelligence 10.* New York: Wiley & Sons, 1982, pp. 269-299.

W. J. Clancey and R. Letsinger, NEOMYCIN: Reconfiguring A Rule-Based Expert System For Application To Teaching. *Proceedings of the Sixth International Joint Conference on Artificial Intelligence,* August, 1981, pp. 829-836.

R. Davis, H. Austin, I. Carlbom, B. Frawley, P. Pruchnik, R. Sneiderman, J. A. Gilreath. The Dipmeter Advisor: Interpretation of Geologic Signals. *Proceedings of the Sixth International Joint Conference on Artificial Intelligence,* August, 1981, pp. 846-849.

R. Davis, Expert Systems: Where Are We? And Where Do We Go From Here? The AI Magazine, Vol. 3, No. 2, Spring 1982, pp. 3-22.

E. A. Feigenbaum, Knowledge Engineering: The Applied Side Of Artificial Intelligence. STAN-CS-80-812 (HPP-80-21), Dept. of Computer Science, Stanford University, September 1980.

J. Gaschnig, Application of the PROSPECTOR System to Geological Exploration Problems. In J. E. Hayes, D. Michie, and Y-H Pao (Eds.), Machine Intelligence 10. New York: Wiley & Sons, 1982, pp. 301-323.

A. Gershman, Building A Geological Expert System For Dipmeter Interpretation. Proceedings of the European Conference on Artificial Intelligence, July 1982, pp. 139-140.

J. McDermott, RI: The Formative Years. The AI Magazine, Vol. 2, No. 2, Summer 1981, pp. 21-29.

H. P. Nii, E. A. Feigenbaum, J. J. Anton, and A. J. Rockmore, Signal-to-Symbol Transformation: HASP/SIAP Case Study. The AI Magazine, Vol. 3, No. 2, Spring 1982, pp. 23-35.

B. Sheil, Power Tools For Programmers. Datamation, February 1983, pp. 131-144.

Schlumberger, Dipmeter Interpretation: Volume 1– Fundamentals. 1981.