CMU-CS-85-116

MUD: A Drilling Fluids Consultant

Gary Kahn and John McDermott

Department of Computer Science
Carnegie-Mellon University
Pittsburgh Pennsylvania 15213

23 March 1985

Abstract

This paper reports on MUD, a drilling fluids consultant developed at Carnegie-Mellon University. MUD is able to diagnose drilling fluid problems and recommend treatments for their correction. MUD's functionality, its approach to diagnosis, and its treatment strategies are discussed. In addition, we examine why MUD's approach to diagnosis is successful given domain constraints, and draw several conclusions with respect to knowledge acquisition strategies.

Table of Contents

List of Figures

1. Introduction

The designer of a diagnostic system must identify the domain knowledge that allows observed problems to be explained in terms of their causes and must decide how this knowledge is to be represented and used. Surveying existing diagnostic systems, one sees a continuum of possibilities. At one end, exemplified by the EMYCIN family of systems [vanMelle 81], evidential support functions are applied to evidence whose weight, with respect to diagnostic conclusions, is explicitly represented in the program. The support function takes as its arguments the degree of support contributed by each of a number of weighted evidential considerations and returns a value indicating the degree to which the evidence combines to support a diagnostic conclusion. Such systems are often said to have *compiled* diagnostic knowledge; this means that the intermediary steps in the causal path from hypothesized problem to evidential consideration are not represented, and that the degree of evidential support provided by each consideration is subjectively assigned.

At the other end, exemplified by recent work on computer fault diagnosis [Davis 83], diagnosis rests on an explicit consideration of articulated causal pathways, which are either statically represented as a network of causally related parts or dynamically generated on the basis of more general causal knowledge. A diagnostic conclusion is achieved by successfully tracing causal pathways that unite observations with their hypothesized causes. Here problem solving exploits a world model of events, states, and causal relations. The attempt is to produce a well-connected causal account, or in the extreme, to simulate a pattern of results which corresponds to those observed. Unlike the EMYCIN type of system, the diagnostic significance of any observation is a function of its place in the network of instantiated causal relations.

Along the continuum from 'evidentially' to 'causally' oriented systems are those in which the causal and functional structure of the modelled entity becomes more and more prominent in the representations and search heuristics of the diagnostic system though support functions may still be used to assign evidential relevance and achieve diagnostic conclusions. Systems such as CADUCEUS [Pople 82], CASNET [Weiss 78], and ABEL [Patil 81] occupy different points along the continuum.

MUD[1], a drilling fluid[2] diagnostic and treatment consultant recently developed at Carnegie-Mellon University in cooperation with NL Baroid, occupies a place near the evidential end of this continuum. MUD was developed for a number of reasons:

1. To achieve some clarity about the desirability and implications of attacking a diagnostic task using an evidential as opposed to a causal approach.
2. To explore the appropriateness of existing approaches to diagnosis in a novel domain.
3. To determine how far the flexibility of pattern matching production system languages, such as OPS, could be exploited in a diagnostic and treatment task.

This paper discusses the MUD system and our findings with respect to the above goals. Although MUD has entered field testing, as of the writing of this paper it is too early to report on results. During development, however, MUD was confronted by 25 or so test cases. In these cases it demonstrated a level of competence comparable to that of expert mud engineers on the 20 or so types of mud problems it knew about at the time. Thus, we are confident that our design decisions, as discussed below, have led to a workable system.

MUD is designed as a production system and is implemented in OPS5 [Forgy 81]. A production system is composed of conditional, if-then, rules called productions. The *if* part of each rule is a statement of conditions, and the *then* part a statement of actions. Rules are instantiated when their conditions match expressions, or data objects, in a global working memory. Instantiated rules are said to be members of a conflict set. The production system interpreter chooses a rule to fire from among those in the conflict set. When a rule is fired, its actions are carried out. This typically causes changes to working memory resulting in a new conflict set. While the examples presented below assume limited familiarity with OPS5, this is not necessary to an understanding of the text.

---

[1] Many people besides ourselves have contributed to the development of MUD: Randall Brooks, Steven Downes-Martin, David Geller, Kinson Ho, John Hutter, and Jeff Stout deserve special mention. Jeff Stout also deserves thanks for collecting data on the nature of MUD's rules.

[2] Drilling fluids are often composed of clay, which gives them a muddy appearance; hence, drilling fluids are referred to as mud and drilling fluids engineers as mud engineers.

2. MUD and Its Domain

MUD serves as a diagnostic and treatment consultant to mud engineers. Mud engineers know how to test for and regulate drilling fluid properties that influence characteristics of the bore hole, as well as aspects of the drilling operation. A well maintained drilling fluid serves to optimize:

1. Hole cleaning (the removal of cuttings and cavings).
2. The suspension of cuttings and weight material during interruptions in circulation.
3. The removal of sand and cuttings from the mud at the surface.
4. The prevention of caving and sloughing of the hole.
5. The control of subsurface pressures.
6. Filtrate control.
7. Transmission of hydraulic pressure to the bit.
8. The cooling and lubrication of the bit and drill string.
9. The support of the drill string and casing.
10. Well logging.

MUD's diagnostic conclusions and treatment recommendations must be sensitive to the composition, or type, of mud in use. At the point that MUD moved from CMU to NL Baroid, it could find the causes of mud problems and recommend treatments to correct these problems for 2 of the 10 standard mud types it will eventually handle. A mud problem is defined as a deviation from expected measures for one or more of about 20 mud properties typically monitored by mud engineers;[3] these properties include density, solids content, rheology, and filtrate characteristics, among others. Deviations are recognized from disparities between current measures and optimal target levels established by a mud plan[4] that provides expectations of the state of the drilling fluid. MUD has some capacity to generate its own mud plan,[5] but more typically the plan is provided from an external data source or interactively by the user.

Diagnosing a mud problem entails finding the causes for deviant test results. Possible causes include contaminants, high temperatures, high pressures, and inadequate corrective treatments, including the under-

---

[3] The traditional approach has been to enter this data on an API (American Petroleum Institute) drilling mud report form; see Appendix A.

[4] In the drilling fluids domain this is typically called a mud *program*; it is referred to here as a *plan* in order to avoid a software connotation.

[5] This is done by a companion program to MUD, called TARGET SETTER. Work on the latter system is primarily due to the efforts of Steven Downes-Martin.

use of solids-removal equipment and the unsatisfactory use of chemical additives. MUD may arrive at more than one hypothesis about possible causes for any set of test results. In this case, hypotheses are ranked by confidence; MUD is able to explain its level of confidence in each hypothesis. MUD suggests treatments which may require either restoring or altering mud properties through the addition of chemical additives or the operation of special equipment. When alternate treatments are available, MUD evaluates them and chooses the best. In what follows some representative output from the MUD system is described.[6]

2.1 Diagnosis

Whenever MUD is provided with information about mud properties, it produces a list of those properties that are either low (L) or high (H) with respect to a desired target value. For example, MUD might produce the following list:

THE FOLLOWING ARE ABOVE OR BELOW SET TARGETS

DENSITY	L
PLASTIC VISCOSITY	L
YIELD POINT	L
HTHP CAKE THICKNESS	H
SOLIDS CONTENT	L
LIQUID CONTENT	H
ELECTRICAL STABILITY	H
OIL WATER RATIO	H
LOW SPECIFIC GRAVITY SOLIDS	L

In addition to recognizing deviations with respect to threshold values, MUD is also sensitive to the degree of change in a mud property and its proximity to a problem threshold. When a property deviates more than a certain percentage from either its previous level or its reading on a previous day, a warning is issued. A warning is also issued if the current reading is within a certain degree of a problem threshold. These warnings are merged by a primitive discourse manager.

WARNING

THE CURRENT READING OF YIELD POINT [6 LBS/ 100 SQ-FT]
IS 67 % HIGHER THAN THE PREVIOUS READING,
BUT IS MOVING TOWARD THE TARGET VALUE OF 11.5 LBS/ 100 SQ-FT

WARNING

[6] See Appendix B for an example of a complete interaction with MUD.

THE CURRENT READING OF PLASTIC VISCOSITY [38 CP]
IS 16 % LOWER THAN THE PREVIOUS READING,
AND IS MOVING AWAY FROM THE TARGET VALUE OF 42.5 CP

WARNING
THE CURRENT READING OF TOTAL CHLORIDES [280000 MG/L]
IS 27 % HIGHER THAN THE PREVIOUS READING
AND RIGHT AT THE MINIMUM ACCEPTABLE VALUE OF 280000 MG/L,
BUT IS MOVING TOWARD THE TARGET VALUE OF 290000.0 MG/L

Having considered the evidence and perhaps after requesting more information from the user, MUD reports its strongest conclusions, provided that they are supported by a combined evidential weight exceeding a set threshold. At the same time, any observation not consistent with the hypothetical evaluation is reported.

THERE IS CONSIDERABLE EVIDENCE THAT:
1: THERE IS AN INFLUX OF HYDROCARBONS
ALTHOUGH THERE IS ENOUGH EVIDENCE TO ACCEPT THE HYPOTHESIS,
CONTRARY TO EXPECTATIONS:
THERE IS NO DECREASE IN ELECTRICAL STABILITY

At this point in an interaction, the user is presented with a menu offering several kinds of explanatory displays. Most importantly, one can examine MUD's reasons for its assessment of any hypothesis considered during a diagnostic session.

WHICH HYPOTHESIS WOULD YOU LIKE EXPLAINED [1]: 1
THAT THERE IS AN INFLUX OF HYDROCARBONS CAN BE ACCEPTED BECAUSE
THERE IS AN INCREASE IN SYSTEM VOLUME
THERE IS A DECREASE IN LOW SPECIFIC GRAVITY SOLIDS
THERE IS AN DECREASE IN DENSITY
THERE IS A DECREASE IN PV
AND MORE SPECIFICALLY
THE OIL-WATER RATIO IS UP

In addition to explanations of the above sort, a user can find out which hypotheses the system considered during its analysis (and how confident it was in each), receive a summary of what each of these hypotheses would have accounted for, and ask, for any particular symptom (or deviant result), which hypotheses would have explained it. Finally, the user can ask if there is any deviant data that is not accounted for by some accepted hypothesis.

2.2 Treatment

Having drawn its conclusions about the causes behind observed property deviations, MUD is prepared to provide treatment recommendations. MUD deals with treatments at two levels of specification. At the first level, it provides a treatment plan in which the nature of recommended additives and their consequences are described. Since many of the effects of the diagnosed problem are secondary results of certain deviant properties, MUD's treatment plan covers only what it believes to be the primary property deviations.

```
PROBLEM: AN INFLUX OF FORMATION HYDROCARBONS
EFFECTS:
    DECREASE IN 10 SECOND GEL-STRENGTH
    DECREASE IN 10 MINUTE GEL-STRENGTH
    DECREASE IN DENSITY
    DECREASE IN LOW SPECIFIC GRAVITY SOLIDS
    DECREASE IN PLASTIC VISCOSITY
    DECREASE IN YIELD POINT
    INCREASE IN OIL WATER RATIO
TREATMENTS:
    ADD WEIGHT-MATERIAL TO INCREASE DENSITY
    ADD EMULSIFIER TO INCREASE ELECTRIC STABILITY
    ADD WATER TO DECREASE OIL WATER RATIO
```

At the second level of description, MUD recommends specific additives and their amounts; the cost of each addition is given, together with the total cost of the treatment plan. An asterisk next to an additive means that the system knows about other additives with the same function.

MATERIAL SPECIFICATIONS

PRODUCT	RECOMMENDED TREATMENT	CURRENT INVENTORY	COST OF MATERIALS
EZ-MUL	4 DRUMS	1800 DRUMS	$ 3234
INVERMUL	21 DRUMS	1800 DRUMS	$ 17427
* BARGAIN	670 SX	2800 SX	$ 3786
WATER	53 GALS	2800 GALS	$ 13
GELTONE	20 SX	800 SX	$ 1453
TOTAL COST OF THE ABOVE MATERIALS IS:			$ 25914
RESULTING VOLUME = 2064 BBL			

The user may ask for two kinds of explanations about the recommended treatment plan. If the user asks to have the recommended amounts explained, the nature of the explanation offered depends on the function of the additive. For instance, if an emulsifier is added, the standard recommended dosage at a particular temperature is indicated, together with any factors that lead to a modification of that dosage. If oil is added, the amount is explained in terms of that needed to increase the oil/water ratio from some current value to a desired target value. The user may also examine the alternatives to an asterisked additive to find out why it was not MUD's choice; MUD decides among alternatives on the basis of a parametric function which takes into account cost of treatment, expected side effects, and inventory.

```
[M]: EXPLAIN EZ-MUL
    AT 325 DEGREES A DOSAGE OF 3 PPB IS REQUIRED. 1 DRUM IS SLIGHTLY MORE THAN 26.25 LBS.
    21 LBS. SHOULD BE ADDED TO COVER AN ADDITIONAL SYSTEM VOLUME OF 7 BBL.
    AN ADDITIONAL 25% OF EZ-MUL IS ADDED TO THE RECOMMENDED TREATMENT WHEN
    DRILLING THROUGH SALT, BRINGING THE TOTAL TO 26.25 LBS.
[M]: ALTERNATIVES TO BARGAIN
    BAROID IS AN ALTERNATIVE TO BARGAIN:
        BAROID        950 SX            1500 SX        $      4010
    BARGAIN WAS PREFERRED BECAUSE:
        IT IS CHEAPER
        IT HAS FEWER SIDE EFFECTS
```

3. Diagnosis

3.1 MUD's Approach

There are several substantial design decisions that must be made early in the development of a diagnostic system. Perhaps most important is the decision of where to locate a system on the continuum of evidential to causal approaches. MUD is located near the evidential end of this spectrum for a number of reasons discussed more fully elsewhere [Kahn 84a].

In summary, MUD has little need of the two features that often make a model-driven approach desirable, namely a constrained search of diagnostic considerations and explanations of why such considerations are evidentially relevant. An INTERNIST-like causally-constrained search from symptoms to deep causes offers no advantages because the number of potential diagnostic conclusions is limited due to the initial availability of diagnostically discriminative evidence. Deep explanations are not required since the typical users of MUD have little understanding of mud chemistry; they are trained to recognize significant patterns of evidence and to draw conclusions from these rather than from an understanding of how problematic events cause the observed symptoms. Moreover, in any particular run-time context, it would have been unreasonable to gather all the information that a model-driven diagnosis would have required.

Once one decides to take an evidential approach, there is a quite natural structure to diagnostic problem solving:

1. Generate a set of plausible hypotheses.
2. Order the hypotheses for investigation.
3. For each hypothesis, determine what information is required in order to accept or reject it.
4. Seek out this information.
5. Evaluate each hypothesis on the basis of the available evidence.

MUD follows this approach fairly closely, as do INTERNIST [Pople 82], and to a somewhat lesser degree, EMYCIN [vanMelle 81]. But within this general approach, the designer of a diagnostic system, is presented with a number of design decisions bearing on

1. the representation of diagnostic knowledge
2. the search of the problem space
3. the evaluation of evidence and hypotheses
4. the explanation of diagnostic conclusions.

3.2 Representation

MUD relies on compiled diagnostic knowledge and evidential support functions. Figure 3-1 provides an example of a rule expressing the relation between a hypothesis and the evidence which supports it. Each diagnostic rule is a production that may be fired when MUD decides to investigate a relevant hypothesis. A description of the evidence supporting a hypothesis is then entered into its global working memory. Other more general rules constitute an inference engine and provide the capabilities for seeking and evaluating evidence, as well as deciding among hypotheses.

```
(P HYPOTHESIS::FORMATION-SOLIDS-CONTAMINATION
    (HYPOTHESIS ↑NAME FORMATION-SOLIDS ↑STATUS OPEN)
    (DATA ↑NAME MUD-TYPE ↑VALUE INVERMUL)
    -->
    (BIND <NEWLABEL>)
    (MAKE REASON ↑FOR FORMATION-SOLIDS ↑LABEL <NEWLABEL> ↑TYPE RESULT
        ↑NAME LOW-SPECIFIC-GRAVITY-SOLIDS-UP ↑POSITIVE-SUPPORT 9 ↑NEGATIVE-SUPPORT 8)
    (MAKE DATAFOR ↑FOR <NEWLABEL> ↑OBJECT DATA ↑OBJECT-NAME LOW-SPECIFIC-GRAVITY SOLIDS
        ↑RELEVANT-ATTRIBUTE DIRECTION ↑CONDITION HIGH))
```

English translation:[7]
When assessing the possibility of formation solids contamination using an invermul mud
consider the following evidential relations:
If the percent of formation solids in the system is higher than expected under normal conditions, there is considerable reason (.9) for believing that there is formation solids contamination.
If the percent of formation solids in the system is not higher than expected under normal conditions, there is considerable reason (.8) for disbelieving that there is formation solids contamination.

Figure 3-1: A sample rule

The effect of the diagnostic rules is to generate a tree of evidential considerations or reasons below each hypothesis. At the top node of each tree is a unique HYPOTHESIS. Below each HYPOTHESIS are one or more REASONS. A REASON represents a consideration with a positive and negative evidential weight, which may be used in confirming and/or disconfirming the HYPOTHESIS. In other words, the evidential focus of the rule can be considered both in terms of its sufficiency and necessity *vis a vis* deducing that the hypothesized state holds.

---

[7] All of the rule translations in this report are hand generated. The translations emphasize the main import of the rule, and do not correspond directly to the condition elements or actions of the rules.

Each REASON is linked by a network to data that can potentially justify the consideration. In the simple case, there is a single link, expressed in the DATAFOR, between the REASON and one 'observation' (the value of an attribute of a specified data object). In other cases, a REASON may be linked to a set of observations as is the case in the rule in figure 3-2. A truth value is returned to the REASON by a Boolean function over the set of associated observations.

In figure 3-1, the REASON provides support for the hypothesis that formation solids are building up in the drilling fluid. The evidence which grounds the REASON is described in the DATAFOR working memory element as the value of the DIRECTION attribute of the DATA object whose name is LOW-SPECIFIC-GRAVITY-SOLIDS. When the test specified in the ↑CONDITION field is true for the value of ↑RELEVANT-ATTRIBUTE, the REASON is justified and may be used in support of the hypothesis to which it is linked. Thus, in the example, if the value of ↑DIRECTION in the specified data object is HIGH, then the REASON linked to the DATAFOR is justified. When a rule is justified, the value specified in the ↑POSITIVE-SUPPORT field will be used in computing an overall measure of belief for the hypothesis in question. The value specified in the ↑NEGATIVE-SUPPORT field will be used in computing an overall measure of disbelief when a REASON is unjustified (see section 3.4).

In figure 3-2, the top node of the DATAFOR network is a conjunctive condition on subsequent DATAFORS, each of which is denoted by having a ↑SIBLING field of <SIBLING>. The DATAFOR which links the REASON to the current value of nacl-ppm (parts per million of dissolved sodium chloride) is of ↑TYPE RELEVANCY. The consequence of this is that the supported reason will be ignored unless this DATAFOR evaluates to true, which will be the case if and only if the current amount of dissolved sodium chloride (represented as the ↑VALUE attribute of the data object named NACL-PPM) is less than 380,000 ppm.

3.2.1 Big rules and little rules

Diagnostic rule-based expert systems differ in regard to the amount of evidential knowledge represented in a rule. As figure 3-3 shows, the typical MYCIN rule demands that several pieces of evidence, including several symptoms, be present before any conclusion can be drawn. While MYCIN includes rules that combine distinct evidential considerations, it also uses an algorithmic procedure of evidential combination. In this respect, MYCIN differs from systems developed using EXPERT [Weiss 78], in which a distinct rule for every element in

```
(P DEMON::HYPOTHESIZE-SALT-NACL-PPM-UP
   (HYPOTHESIS ↑NAME SALT ↑STATUS OPEN)
   (DATA ↑NAME MUD-TYPE ↑VALUE INVERMUL)
   -->
   (BIND <NEWLABEL> (GINT))
   (MAKE REASON ↑FOR SALT ↑LABEL <NEWLABEL> ↑TYPE CAUSE
      ↑NAME NACL-PPM-UP ↑POSITIVE-SUPPORT 2 ↑NEGATIVE-SUPPORT 9)
   (BIND <SIBLING> (GINT))
   (MAKE DATAFOR ↑FOR <NEWLABEL> ↑CONDITION AND <SIBLING>)
   (MAKE DATAFOR ↑FOR <NEWLABEL> ↑EXPLAINED-BY SALT ↑OBJECT DATA ↑OBJECT-NAME NACL-PPM
      ↑RELEVANT-ATTRIBUTE DIRECTION ↑CONDITION = HIGH ↑SIBLING <SIBLING>)
   (MAKE DATAFOR ↑FOR <NEWLABEL> ↑TYPE RELEVANCY ↑OBJECT DATA ↑OBJECT-NAME NACL-PPM
      ↑RELEVANT-ATTRIBUTE VALUE ↑CONDITION < 380 ↑SIBLING <SIBLING>))
```

English translation:
When assessing the possibility of a salt formation being drilled when using an invermul mud consider the following evidential relations:
If the current amount of dissolved salt is less than 380,000 ppm and
is present in amounts higher than expected
there is a small reason (.2) for believing that a salt dome is being drilled.
If the current amount of dissolved salt is less than 380,000 ppm and
is not present in amounts higher than expected
there is a considerable reason (.8) for disbelieving that a salt dome is being drilled.

Figure 3-2: A logically complex rule

the powerset of evidential considerations is required.[8] MUD is at the opposite extreme from systems developed with EXPERT. Domain experts are asked only to specify and weight rules whose conditions cannot be broken up into more elementary evidential considerations. An elementary evidential consideration (referred to as an evidential focus) is typically a single symptom together with one or more background (or contextual) considerations which affect the diagnostic significance of observing that symptom. The rule of combination discussed in section 3.4 is used to combine evidence across different rules.

There are several reasons for pushing down to an elementary level. The first is that the knowledge acquisition task is easier, simply because one needs to inquire about fewer rules. That is, given a set of symptoms {S} where each $s \in S$ can take one of three values -- true (has occurred), false (has not occurred), or unknown -- there are 3^n rules that can be defined on {S}. However, when evidential foci are teased apart,

[8] The position occupied by MYCIN appears to reflect the propensity of physicians to pull together considerations that fall along the same path in a differential diagnosis. Since there may be several such paths underlying a diagnostic conclusion, there are several such rules.

IF (1) THE SITE OF THE CULTURE IS BLOOD, AND
(2) THE GRAM STAIN IS POSITIVE, AND
(3) THE SUSPECTED PORTAL OF ENTRY OF THE ORGANISM IS THE GASTRO-INTESTINAL TRACT, AND
(4) THE LOCUS OF INFECTION IS THE ABDOMEN OR THE PELVIS
THEN THERE IS STRONGLY SUGGESTIVE EVIDENCE (.9) THAT
THERAPY SHOULD COVER ENTEROBACTERIACEAE

Figure 3-3: A typical MYCIN rule

there are only 3n possible rules, as the evidential contribution of each s ϵ S will depend only on the truth value of s. Once these contributions are determined, they can be composed using accepted algorithms for the combination of evidence.

Secondly, complex rules can generate unintuitive results under MYCIN-like assumptions that beliefs are non-complementary and that uncertainties are propogated using the conjuctive/disjunctive rules of standard fuzzy logic. For instance, consider the two sets of rules in Figure 3-4, where F represents Bernoulli's rule of evidential combination, variants of which are used by both MYCIN and MUD.

Rules C.1, C.2, and C.3 say that when A and B are the case, believe H to degree (.8), but believe H to degree (.3) if only B is true and to degree (.6) if only A is true. Rules S.1 and S.2 have virtually the same effect as C.2 and C.3 when either A or B is false. However, if both A and B, are true, then both rules will be instantiated in the simple case. Their combined weight, with respect to the hypothesis they support, will be tallied by the function F, which in this case, returns .72 .

COMPLEX RULES
- C.1: A & B --> H (.8)
- C.2: A & ~B --> H (.6)
- C.3: ~A & B --> H (.3)

SIMPLE RULES
- S.1: A --> H (.6)
- S.2: B --> H (.3)
- TOGETHER WITH: F(A.B) --> H(.72)

Figure 3-4: Two representations of evidential knowledge

Now consider what happens with the complex representation when uncertainty enters the picture. Again, following MYCIN, the certainty of a conjunction is taken to be the minimum of the certainties on each of the conjuncts. Thus if B is believed true to degree (.6), ~B to degree (0), and A to degree (1), the overall certainty of the conjunction C.1 is (.6). The contribution of C.1 to a belief in H is then this value (.6) times the confidence factor (.8) which represents the strength of the rule on the condition that its supporting evidence is certain. C.2 makes no contribution to H, as the certainty on ~B is 0. Thus, in this case, we would be left with a contribution of (.48). On the other hand, the procedure for combining the contributions of the simple rules calls for diminishing the contribution in proportion to the uncertainty of the evidence on which the rule is conditional. Thus, the overall belief in H given S.1 and S.2 is F((.6)*(.3), .6) = .67. In this case, the uncertainty on B only effects the contribution of B; A's contribution remains intact. Although it is hard to make general conclusions here, the latter procedure appears to generate more intuitive results in the mud domain.

Despite the above considerations, complex rules may appear desirable insofar as they reflect a kind of domain expertise, namely, the ability of the domain expert to better combine evidence than the combinatoric function. However, conflation of this sort makes it difficult to discern the real content of rules.

For instance, one typical conflation that we found when experts volunteered rules was a 'jackpot effect'; that is, the weight of several observations together far exceeded the weight that would be assigned using a reasonable function over their individual weights. The reason for this, in general, was that the observations together constituted a superset of the evidence for some competing hypothesis. Some additional piece of evidence, not very significant in itself, was very significant in the context of discerning a difference between two well supported hypotheses. What allows this assumption to be buried in a complex evidential rule is the additional assumption that the total set of observations are not due to two distinct hypotheses. While experts may be trusted to have this knowledge within a domain of predictable possibilities, in more open domains, such as MUD's, where bore holes may be through a range of lithologies with quite different problem profiles, no expert is likely to have enough experience to be sure of such an assumption. Thus, in these cases, we have explored two options. One is to define ways of using the diagnostic rule representation to explicitly state that some other hypothesis has been ruled out; the other is to define higher level rules that look at the distribution of evidence across hypotheses, and where reasonable, reject hypotheses whose evidence can be properly subsumed by some other hypothesis. So far we have found no diagnostic loss in limiting rules to an expression of the evidential significance of a single evidential focus, provided that the diagnostic strategies inherent in assessing combinations of symptoms are made explicit.

Teasing out the different factors which may be naively compounded in a complex rule has led to some pleasing results. For one, we have been able to gain some insights about where confidence factors come from, that is, about the objective considerations behind the domain experts' subjective assignments. These insights have proved valuable both to the discernment of errors in the rule set, as well as to enhancing the effectiveness of our knowledge acquisition interviews (see section 5.2 below).

3.3 Search

The problem space in a diagnostic system can be viewed as consisting of hypotheses, reasons (in light of which one judges hypotheses), and the factual data required to support a reason's application under particular circumstances. In systems that actively seek confirmatory or disconfirmatory data for potentially true hypotheses, the interactive burden on the user increases with the number of hypotheses investigated. There are two ways to reduce this burden. One is to begin by passively collecting significant observations on the basis of which a relatively small set of initial hypotheses may appear as plausible candidates. The other alternative is to constrain search by pruning from the set of hypotheses (sometimes all those in the knowledge base) before all the evidence is in. The latter strategy requires a careful consideration of which reasons will provide the most leverage.

INTERNIST relies on both these strategies, MYCIN [Shortliffe 76] only on the second. MUD relies on the first strategy; it activates a hypothesis if a relevant 'diagnostically significant' event is known to have occurred. A diagnostically significant observation is typically a deviation from an expected value for a mud property. In interviewing mud engineers we discovered that although they recognize a number of potential consequences for each possible hypothesis, only some of these consequences are considered 'diagnostically significant'. If none of the diagnostically significant consequences occur, the associated hypothesis will not even be considered. Since the search space is small (usually fewer than 6 hypotheses are evoked), this approach works well.

Figure 3-5 shows a rule of the kind that generates a hypothesis for consideration during a diagnostic session. The conditional side of each such rule has three condition elements. The first is the task name; all hypotheses are genererated in the task called DIAGNOSIS. Since the significance of deviant properties varies across mud types, each rule specifies the mud systems to which the rule applies; in this example, the mud system is INVERMUL. The current mud system is always represented as shown; i.e., it is the ↑VALUE field of a DATA working memory element with the ↑NAME MUD-TYPE. The third condition element lists the kinds of data, which, if deviant, would indicate that the hypothesis generated by the rule should be considered a potential cause. Hypothesis generation rules come in pairs, with one listing data which is significant when observed to be high with respect to expectations and the other listing data which is significant when low. The ↑DIRECTION attribute distinguishes these cases. The value of this field is set by rules which determine if the current value of the DATA is above or below specifications, as defined by the mud plan.

```
(P DIAGNOSIS::FORMATION-SOLIDS-LOW
    (TASK ↑NAME DIAGNOSIS)
    (DATA ↑NAME MUD-TYPE ↑VALUE INVERMUL)
    (DATA ↑NAME DENSITY ↑DIRECTION LOW)
    -->
    (MAKE HYPOTHESIS ↑NAME FORMATION-SOLIDS ↑CLASS SUPER))
```

English translation:
If the current task is diagnosis
and an invermul mud is in use
and the density is lower than expected,
then consider the possibility of formation solids contamination.

Figure 3-5: Rule to evoke a hypothesis

The action side of the rule simply enters a representation of a particular hypothesis into working memory. Each hypothesis is a member of a class, specified as the value of the ↑CLASS atribute. Class can be SUPER, MATERIAL, or the name of a hypothesis whose class is super. All hypotheses involving an under- or over-dosage of a mud treatment are ↑CLASS MATERIAL. All hypotheses which are super-ordinates (or generalizations) of other hypotheses are ↑CLASS SUPER. Thus, the hypothesis for formation-solids has the ↑CLASS SUPER, while the hypothesis for bentonitic buildup in the system would have as its ↑CLASS FORMATION-SOLIDS, since bentonitic buildup is a variant of formation solids buildup. As described below, class specifications are used in ordering hypotheses for evaluation. In addition, certain strategies regarding the acceptance and rejection of hypotheses entail knowing the ↑CLASS of the hypothesis under consideration.

By using rules of this sort, MUD greatly reduces the number of interactions with the user and the amount of processing that would have been required to examine all known hypotheses. However, there is an exhaustive consideration of all hypotheses put into working memory. We have avoided using further pruning strategies which could lead to the rejection or acceptance of a hypothesis on the basis of erroneous information. Since the mud domain is one in which data gathering procedures are executed under less than ideal conditions, test results are often in error and open to question. Since uncertainty is best assessed in light of as much other data as possible, MUD does not terminate its investigation of hypotheses until all potentially useful data is considered.

3.3.1 Ordering the search

Although MUD evaluates all the REASONs for all evoked hypothesis, it pursues the information it requires in a way that mud engineers find natural. This means (1) it considers the REASONs for the more general SUPER hypotheses before the REASONs for their sub-ordinate variants; and (2) it considers REASONs for hypothesized sub-surface problems before those for treatment problems.

As figure (3-6) shows, each strategy is represented by a single rule. In all such rules, a subordinate hypothesis has the name of its logical superordinate as its ↑CLASS specification. The effect of setting the ↑STATUS attribute of a HYPOTHESIS to OPEN is to evoke rules of the kind shown above in figure 3-1 that add REASONs and DATAFORS to working memory. When a data object described by a DATAFOR is unknown to MUD, a data schema is created and MUD infers or asks about the current value of this object. This is in effect a depth first search: as each reason is added to working memory, MUD seeks out its supporting data.

3.4 Evaluation

Evaluation occurs on several levels and in several different ways in MUD. HYPOTHESES, REASONs and DATAFORs all must be evaluated. A terminal DATAFOR, one that points directly to a data object, is evaluated as true, false, or unknown depending on the value returned by the condition test. A logical DATAFOR, one whose ↑CONDITION is of the form <BOOLEAN OPERATION> <LABEL>, where <LABEL> is a pointer to the daughters of the node, is evaluated as true, false, unknown, or irrelevant depending on the value returned by the operation taken over all the daughter DATAFORs, namely, those whose ↑SIBLING field is <LABEL>.

Tests that appear as the ↑CONDITIONs of terminal DATAFORs can be defined as OPS rules. In many cases, each test can be defined as a single OPS rule. In order to increase run-time efficiency, many of these rules have been specialized to be sensitive to the object and fields being tested. As can be seen in figure 3-7, tests can also be defined on a relation between two values in the same or differing working memory elements. In this example, the DATAFOR asks if the current amount of invermul in the drilling system is greater than the

```
(P ORDER-HYPOTHESES::SUPERORDINATES-FIRST
   (TASK ↑NAME ORDER-HYPOTHESES)
   (HYPOTHESIS ↑CLASS {<CLASS> <> UNDERTREATMENT <> OVERTREATMENT} ↑STATUS NIL)
  - (HYPOTHESIS ↑NAME <CLASS> ↑STATUS NIL)
   -->
   (MODIFY 2 ↑STATUS OPEN))
```

English translation:
If the current task is to order the hypotheses
and there is an unevaluated non-treatment hypothesis
and that hypothesis has no unexamined superordinate
then mark the hypothesis as ready for evaluation.

```
(P ORDER-HYPOTHESES::TREATMENTS-LAST
   (TASK ↑NAME ORDER-HYPOTHESES)
   (HYPOTHESIS ↑CLASS << UNDERTREATMENT OVERTREATMENT >> ↑STATUS NIL)
  - (HYPOTHESIS ↑CLASS {<> UNDERTREATMENT <> OVERTREATMENT} ↑STATUS NIL)
   -->
   (MODIFY 2 ↑STATUS OPEN))
```

English translation:
If the current task is to order the hypotheses
and there is an unevaluated treatment hypothesis
and all non-treatment hypotheses have been evaluated
then mark the hypothesis as ready for evaluation.

Figure 3-6: Ordering rules

amount targeted. Once terminal DATAFORs are evaluated, simple rules representing Boolean functions propogate truth values upwards through a network of logical relations for any evaluated DATAFOR.

```
(DATAFOR ↑FOR <NEWLABEL> ↑SIBLING <FJCT>
   ↑OBJECT DATA
   ↑OBJECT-NAME INVERMUL
   ↑RELEVANT-ATTRIBUTE VALUE
   ↑OBJECT-2 DATA
   ↑OBJECT-NAME-2 INVERMUL
   ↑RELEVANT-ATTRIBUTE-2 TARGET
   ↑CONDITION >)
```

Figure 3-7: A binary datafor condition test

Once the top node of a DATAFOR network is evaluated, this value can be passed on to the corresponding REASON. REASONs that have been evaluated as true make a contribution to a measure of belief in a hypothesis, and those that are false to a measure of disbelief. Irrelevant reasons are ignored, as are those that are evaluated as unknown.[9] A non-Bayesian function similar to that used in MYCIN [Shortliffe 76] combines the evidential weights contributed by each REASON. These weights are determined as a result of interviews with domain experts. The function that combines the weights is represented by the two rules in figure 3-8.

```
(P EVAL::EVAL-MEASURE-OF-BELIEF
    (TASK ↑NAME EVAL)
    (HYPOTHESIS ↑NAME <NAME> ↑STATUS OPEN ↑MEASURE-OF-BELIEF <MB> ↑MEASURE-OF-DISBELIEF <MD>)
    (REASON ↑FOR <NAME> ↑POSITIVE-SUPPORT <SUPPORT> ↑SIGN 1 ↑CONFIDENCE <CONFIDENCE>)
    -->
    (BIND <NEW-CONTRIBUTION> (COMPUTE <SUPPORT> * <CONFIDENCE>))
    (BIND <MB> (COMPUTE <MB> + (<CONTRIBUTION> * (10 - <MB>))))
    (MODIFY 2 ↑MEASURE-OF-BELIEF <MB> ↑BELIEF (COMPUTE <MB> - <MD>)))
```

English translation:
If there is new positive evidence for a hypothesis,
then tally it into an accumulator for measure of belief
and reflect this new evidence in the overall belief in the hypothesis.

```
(P EVAL::EVAL-MEASURE-OF-DISBELIEF
    (TASK ↑NAME EVAL)
    (HYPOTHESIS ↑NAME <NAME> ↑STATUS OPEN ↑MEASURE-OF-BELIEF <MB> ↑MEASURE-OF-DISBELIEF <MD>)
    (REASON ↑FOR <NAME> ↑POSITIVE-SUPPORT <SUPPORT> ↑SIGN -1 ↑CONFIDENCE <CONFIDENCE>)
    -->
    (BIND <NEW-CONTRIBUTION> (COMPUTE <SUPPORT> * <CONFIDENCE>))
    (BIND <MD> (COMPUTE <MD> + (<CONTRIBUTION> * (10 - <MD>))))
    (MODIFY 2 ↑MEASURE-OF-DISBELIEF <MD> ↑BELIEF (COMPUTE <MB> - <MD>)))
```

English translation:
If there is new negative evidence for a hypothesis,
then tally it into an accumulator for measure of disbelief
and reflect this new evidence in the overall belief in the hypothesis.

Figure 3-8: Evaluation Rules

An overall measure of belief is taken to be the difference of ↑MEASURE-OF-BELIEF and ↑MEASURE-OF-DISBELIEF. After this has been obtained for all hypotheses, hypotheses are then accepted or rejected in light of acceptance and rejection thresholds, together with limited strategies for cross evaluating

[9] A more sophisticated handling of the latter is desirable, and there are a number of planned enhancements.

competing hypotheses. Belief can run from 0 to 10. If the ↑MEASURE-OF-DISBELIEF is greater than 6, or if ↑BELIEF is less than 0, the hypothesis is rejected. If ↑BELIEF is greater than 6, the hypothesis is accepted. If everything that could be explained by an unaccepted hypothesis is explained by one or more accepted hypotheses, then the former is rejected. If everything explained by a treatment hypothesis (regarding an over or under-treatment) results from an accepted hypothesis referring to a sub-surface problem, the treatment hypothesis is overlooked in favor of reporting the subsurface problem, or root cause, of the drilling fluid property deviations.

3.5 Explanation

Examples of some of the types of explanation MUD can provide have been shown above in section 2. When presented with diagnostic conclusions, the user can ask for an explanation. In order to be able to explain its conclusions, MUD leaves a trace of its behavior; as MUD's reason-generating rules fire, they create working memory elements containing explanatory text.[10]

When presented with a prompt for information, the user can ask why the question is being asked. An answer to this question is given on the basis of the pathway that links the data being requested back to a HYPOTHESIS. MUD's answer is simply that the information is valuable in assessing the hypothesis. Again the representation in MUD provides opportunities to build more sophisticated explanation capabities.

Explanation in many diagnostic systems is limited to preserving a trace of the rules that were used to lend support to or disconfirm a particular hypothesis. When this is the only explanatory goal, a rule-based approach offers no particular advantage. This is especially true when the rule interpretor, as in the OPS5 language, does not make a trace of fired rules accessible to applications programs.

Our goals for MUD, however, included other kinds of explanation capabilities. We wanted to answer questions, such as, "What was the most significant consideration that led to accepting H1 instead of H2?"; "Why was treatment T1 chosen rather than treatment T2?"; "How would I know if H1 were occurring?"; and "What are the symptoms that characterize H1 but not H2?". These questions cannot be answered by

---

[10] We have been working on more sophisticated explanation capabilities in the context of the MEX program described below in section 3.5.1.

maintaining a trace of rules fired during a diagnostic session. In fact, the latter two questions are ones that might well be asked outside of a diagnostic session. Answering questions such as the first two requires the ability to analyze and compare traces of rules fired in support of different hypotheses or treatments.

We found the pattern matching capabilities of OPS of particular use in answering these kinds of questions. With respect to explaining the choice of a treatment plan, we were able to define comparison rules sensitive to features such as cost, side-effect, and inventory availability, the attributes with respect to which alternative treatments were evaluated. We have also used the pattern matching capabilities to answer the last two kinds of questions. This is currently the responsibility of MEX, a companion system to MUD.

3.5.1 MEX

Although MUD can explain how it arrived at a diagnostic conclusion, it cannot answer questions such as, "How would I know if there was a <type-x> problem?", and "How would I know the difference between a <type-x> problem and a <type-y> problem?". Such questions are the province of MEX, which provides access to MUD's rule base outside of a diagnostic session. For example, figure 3-9 shows how MEX displays the supporting reasons for the hypothesis that there is an influx of water. Figure 3-10 shows MEX's response to a query with respect to the differences between reasons bearing on two competing hypotheses, namely, gyp/anhydrite contamination and cement contamination.

```
THE FOLLOWING CONDITIONS COULD ESTABLISH THAT THERE IS AN INFLUX OF WATER:
  R1:  THERE IS AN INCREASE IN YIELD POINT,
       AND/OR 10 MINUTE GEL-STRENGTH OR 10 SECOND GEL-STRENGTH
  R2:  THERE IS A DECREASE IN DENSITY, ELECTRICAL STABILITY, AND/OR OIL WATER RATIO
  R3:  EITHER THERE IS AN INCREASE IN SYSTEM VOLUME OR IT IS NOT TRUE
       THAT THERE IS AN INCREASE IN SYSTEM VOLUME BECAUSE THERE HAS BEEN LOST CIRCULATION
```

Figure 3-9: An example of MEX output

MEX provides several benefits. For one, it permits drilling fluids engineers to determine the consequences of expected problems, thus providing information that may allow for the preventive pre-treatment of the drilling fluid. Secondly, it allows access to MUD rules in an instructional context. And finally it provides a substantial debugging tool. Without a tool for interpreting the rule base, the misrepresentation of expert diagnostic knowledge may not become evident until a test case generates unacceptable results. While many such misrepresentations are unintentional, others result from uncertainty on the part of programmers about

CONSIDERATIONS WHICH SUPPORT ONLY THE HYPOTHESIS THAT GYP/ANHYDRITE IS BEING DRILLED ARE:

R1: THERE IS A DECREASE IN BICARBONATE, AND/OR CARBONATE

R2: IF THERE IS AN INCREASE IN TOTAL HARDNESS AS CALCIUM, THEN CONSIDER IF:
THERE IS AN INCREASE IN SULFATE

R3: IF THERE IS NOT AN INCREASE IN PH , THEN CONSIDER IF:
THERE IS AN INCREASE IN TOTAL HARDNESS AS CALCIUM, FILTRATE API, YIELD POINT, AND/OR 10 MINUTE GEL-STRENGTH OR 10 SECOND GEL-STRENGTH.

CONSIDERATIONS WHICH SUPPORT ONLY THE HYPOTHESIS THAT CEMENT IS BEING DRILLED ARE:

R1: THERE IS AN INCREASE IN TOTAL HARDNESS AS CALCIUM AND PH

R2: IF THERE IS AN INCREASE IN TOTAL HARDNESS AS CALCIUM AND PH, THEN CONSIDER IF:
THERE IS AN INCREASE IN 10 SECOND GEL-STRENGTH OR 10 MINUTE GEL-STRENGTH, PLASTIC VISCOSITY, AND/OR FUNNEL VISCOSITY

THE FOLLOWING CONSIDERATIONS PROVIDE EVIDENCE ONLY AGAINST THE HYPOTHESIS THAT CEMENT IS BEING DRILLED:

R1: THERE IS NO INCREASE IN PH

THE FOLLOWING CONSIDERATIONS PROVIDE EVIDENCE ONLY AGAINST THE HYPOTHESIS THAT GYP/ANHYDRITE IS BEING DRILLED:

R1: THERE IS AN INCREASE IN PH

CONSIDERATIONS WHICH PROVIDE EVIDENCE AGAINST BOTH HYPOTHESIS ARE:

R1: THERE IS NO INCREASE IN TOTAL HARDNESS AS CALCIUM

Figure 3-10: A sample of MEX's ability to differentiate among hypotheses

how they ought to encode evidential considerations in terms of the representational devices provided by the program. MEX provides programmers who are adding rules to MUD's knowledge base with a means of checking the meaning of new rules.

In what follows we survey the special knowledge required to provide an adequate explanation of MUD's knowledge base. Details regarding MEX's implementation are provided in [Kahn 85]. MUD's representation of diagnostic knowledge is expressed within networks of DATAFOR working memory elements (see section 3.2). A semantic interpretation of these networks requires:

1. the ability to describe primitive evidential considerations, as represented in a single DATAFOR;
2. the ability to recognize logical and conceptual relations expressed within a single network of DATAFORs, dominated by the same REASON;
3. the ability to recognize logical and conceptual relations across distinct networks, each of which is dominated by a different REASON;

Each of the above abilities is provided by rules designed to recognize the meaning inherent in the representations used to model diagnostic knowledge in the mud domain.

3.5.2 Primitive evidential considerations

Much of MEX's text generation capacity is based on rules which generate a phrase (represented as a string) given the DATAFOR specification of a working memory element type, attribute field, and condition test.

The DATAFOR shown in figure 3-11, for example, is easily translated to "temperature is greater than or equal to 325 degrees farenheit" by the rule shown in figure 3-12. This rule recognizes that a ↑RELEVANT-ATTRIBUTE instantiation of VALUE means that the condition test is a test of the value of the named data object. The translation for ">=" is embodied in the rule. There are similar rules for other condition tests.

```
(DATAFOR
    ↑OBJECT DATA
    ↑OBJECT-NAME TEMPERATURE
    ↑RELEVANT-ATTRIBUTE VALUE
    ↑CONDITION >= 325)
```

Figure 3-11: A simple datafor

```
(P EXPLAIN::VALUE-GREATER-OR-EQUAL-THAN
    (TASK ↑NAME MAKE-STRING)
    (HYP ↑NAME <HYP> ↑STATUS OPEN)
    (REASON ↑FOR <HYP> ↑LABEL <NEWLABEL>)
    (DATAFOR ↑FOR <NEWLABEL> ↑OBJECT DATA ↑OBJECT-NAME <NAME>
        ↑RELEVANT-ATTRIBUTE VALUE ↑CONDITION >= <VAL>)
    (DATA ↑NAME <NAME> ↑LONGNAME <LONG> ↑UNITS <UNITS>)
    -->
    (CALL IMPLODE |GREATER THAN OR EQUAL TO | <VAL> | | <UNITS>)
    (BIND <IMP-COMP> (ANSW))
    (MAKE STRING ↑FOR <NEWLABEL> ↑LABEL <SLABEL> ↑SIBLING <SIBLING> ↑NAME <LONG> ↑TYPE <TYPE>
        ↑PREFIX NIL ↑SUBJECT <LONG> ↑VERB |IS| ↑COMPLEMENT <IMP-COMP> ↑PHRASE-TYPE SVC)
```

English translation:
If there is, with respect to the current hypothesis, a reason grounded on the observation
that the value of a given datum is greater than or equal to a given constant
then make a note of that fact

Figure 3-12: A rule for translating simple DATAFORs

The string that is created in figure 3-12 has fields for a prefix, a subject, a verb, and a complement. These fields allow the information held in individual strings to be merged into a linguistically correct phrase. The value of ↑PHRASE-TYPE indicates how to to compose a gramatically correct sentence from the parsed representation.

Many of MEX's elementary string generation rules are more specialized than the rule shown in figure 3-12; they can recognize special conditions which call for a non-literal translation. For instance, there is rule that recognizes that the condition test, "= H", applied to the ↑DIRECTION attribute of a working memory element representing a mud property, means "there is an increase in <that mud property>" rather than "the direction <of the mud property> equals H". Similarly, when the condition test on the ↑VALUE field of a working memory element representing a material additive is "> 0", MEX recognizes that this means not that the material is greater than 0, but rather that the material "is present in the system".

3.5.3 Single reason complex networks

MEX must also interpret networks which express logical or conceptual relations between two or more factual considerations underlying a single reason. In some cases MUD need only combine strings associated with each component DATAFOR, taking care to preserve logical scope as it injects logical operators such as "and" and "or".

However, besides recognizing simple logical relations, MEX must be sensitive to both procedural and conceptual relations expressed by DATAFOR networks. Procedural relations involve the conditional use of evidence. MEX displays these relations in the following way:

R1: IF TARGET LEVEL FOR DENSITY IS LESS THAN OR EQUAL TO 10 PPG, THEN CONSIDER IF:
THERE IS AN INCREASE IN DENSITY

Alternatively, multiple evidential considerations underlying a single reason may implicitly represent a critical conceptual relation. One of MEX's most significant tasks is to provide a mechanism that allows these conceptual relations to be made explicit. This cannot be done with MEX's rule of logical combination, which would, for example, intepret the "and" node underlying a supporting reason for an underdosage of invermul as:

THERE IS AN INCREASE IN HIGH TEMPERATURE / HIGH PRESSURE FILTRATE
AND THE AMOUNT OF DURATONE IS BELOW THE DESIRED TARGET

rather than with the more accurate phrase:

THERE IS AN INCREASE IN HIGH TEMPERATURE / HIGH PRESSURE FILTRATE
THAT IS NOT DUE TO INSUFFICIENT AMOUNTS OF DURATONE

Text such as this is generated by a rule which fires whenever a supporting reason for an undertreatment of a material is supported by a conjunction in which a mud property which has shown an increase is logically conjoined to one which tests to see if the amount of some material in the system is less than that targeted.

3.5.4 Logical and conceptual relations across reasons

MEX also recognizes logical and conceptual relations between different REASONS. Logical relations are those that are recognized on the basis of the form of the DATAFOR network alone; that is, without additional information regarding the meaning of the facts referred to. On the other hand, recognizing conceptual relations requires bringing additional information to bear.

MEX recognizes a logical relation holding across different relations when, for instance, it merges reasons which differ in that one makes a factual reference to the occurrence of a particular fact, while the other refers to the non-occurence of the same fact. For example, MEX recognizes that the two reasons:

R1: SYSTEM VOLUME IS UP
R2: SYSTEM VOLUME IS NOT UP AND THERE IS LOST CIRCULATION

can be combined as:

R1: EITHER THERE IS AN INCREASE IN SYSTEM VOLUME OR IT IS NOT TRUE
THAT THERE IS AN INCREASE IN SYSTEM VOLUME BECAUSE THERE HAS BEEN LOST CIRCULATION

In more general terms, MEX looks for strings that have their source in DATAFORs which are similar except for their condition tests, which are logical complements. One and only one of these DATAFORs will be conjoined to another DATAFOR. MEX assumes that the effect of this associated DATAFOR is to mask or neutralize the evidential observation expressed by its conjuct's complement, that is the solitary DATAFOR. Actions such as this do not require any background knowledge about the facts in particular.

On the other hand, there are times when an appropriate interpretation can make use of additional background knowledge. For instance, in order to generate the string:

R1: THERE IS AN INCREASE IN DISSOLVED SODIUM CHLORIDE WHEN THE MUD IS UNDERSATURATED
OR AN INCREASE IN UNDISSOLVED SODIUM CHLORIDE WHEN IT IS OVERSATURATED

MEX must recognize one reason that is relevant to a saturated state and another relevant to an unsaturated mud solution. Having specific knowledge about some causal properties of the MUD system allows MEX to see such relations when they occur.

4. Treatments

Although questions related to diagnosis dominated our research interests, MUD also required a capacity to recommend treatments for mud problems. Beyond the recommendation of correct treatments, we had the following design goals for this component of MUD:

1. Treatment strategies were to be flexible, recognizing that different responses may be required to the same problem under different circumstances.
2. There should be room for both heuristic and algorithmic knowledge in calculating how much of a particular material additive is to be added to the drilling fluid system.
3. There should be the capacity to formulate alternative treatment strategies and the ability pick the best one.
4. There should be a capacity to explain treatment recommendations.

The current version of MUD succeeds to some extent in meeting all of these goals. In the following sections, we discuss MUD's approach to treatments. Formulating a treatment plan requires:

1. Generating a functionally appropriate treatment plan
2. Choosing specific chemical additives or equipment which will meet the specified functional requirements
3. Determining the amount of an additive to be used or the duration with which to run drilling fluids equipment
4. Evaluating and choosing between alternative treatments with the same function

4.1 Treatment Plan Generation

Anytime an event occurs that can cause mud properties to deviate from their desired target levels, the drilling fluid system requires treatment. Each of these events, represented as a HYPOTHESIS from MUD's diagnostic point of view, is associated with a treatment plan in MUD's rule base. A treatment plan describes an appropriate action to take if the hypothesized event has occurred and mud properties have deviated in a particular way. The rule in figure 4-1, for instance, indicates that if the problematic event is salt contamination, an emulsifier should be added if either the measurements for electrical stability or high pressure-high temperature filtrate are below a minimally acceptable target level or a decision has already been made to add water. It is important to note that at this level of description, the corrective action is described in functional terms, namely, as that of adding an emulsifier. It is only later that MUD chooses a specific product and determines the amount to use.

The representation used to express a treatment plan is very similar to that used to relate REASONS to supporting evidence (see section 3.2). In this case, however, it is a TREATMENT working memory element which is linked by a DATAFOR network to the evidence on the basis of which one would recommend the actions specified in the ↑OPERATOR/↑OPERAND attributes of this working memory element. The rule in figure 4-1, for instance, makes the decision to add an emulsifier only if the datafor network dominated by the TREATMENT evaluates as true. In this case at least one of the subordinate disjuncts must be true. If none are, MUD will not recommend adding an emulsifier even though there is salt contamination. Rules of this kind typically generate several TREATMENTs, each with a set of supporting conditions or DATAFORS. Taken together the set of TREATMENTs constitutes a *treatment plan.*

```
(P GENERATE-TREATMENT::SALT
   (TASK ↑NAME GENERATE-TREATMENT ↑CONTROL SALT ↑ATTRIBUTE {<COUNTER> <> NIL})
   (DATA ↑NAME MUD-TYPE ↑VALUE INVERMUL)
   -->
   (BIND <LABEL> (GINT))
   (MAKE TREATMENT ↑HYPOTHESIS SALT ↑PLAN <COUNTER> ↑NAME ADD-EMULSIFIER
      ↑LABEL <LABEL> ↑ACCEPT NIL ↑FOR E-STABILITY ↑DIRECTION INCREASE
      ↑OPERATOR ADD ↑OPERAND EMULSIFIER)
   (BIND <SIBLE> (GINT))
   (MAKE DATAFOR ↑PLAN <COUNTER> ↑FOR <LABEL> ↑CONDITION OR <SIBLE> ↑TV NIL)
   (MAKE DATAFOR ↑PLAN <COUNTER> ↑FOR <LABEL> ↑SIBLING <SIBLE> ↑OBJECT DATA
      ↑OBJECT-NAME E-STABILITY ↑RELEVANT-ATTRIBUTE DIRECTION ↑CONDITION = LOW)
   (MAKE DATAFOR ↑PLAN <COUNTER> ↑FOR <LABEL> ↑SIBLING <SIBLE> ↑OBJECT DATA
      ↑OBJECT-NAME HTHP-FILTER ↑RELEVANT-ATTRIBUTE DIRECTION ↑CONDITION = HIGH)
   (MAKE DATAFOR ↑PLAN <COUNTER> ↑FOR <LABEL> ↑SIBLING <SIBLE> ↑OBJECT TREATMENT
      ↑OBJECT-NAME ADD-WATER ↑RELEVANT-ATTRIBUTE ACCEPT ↑CONDITION = YES)
   (MAKE ADDS ↑ORDEROF CALCULATION ↑ARE WATER WEIGHT-MATERIAL EMULSIFIER
      ↑PLAN <COUNTER>))
```

English translation:
When generating a treatment program for salt contamination of an invermul mud
suggest the addition of emulsifier
if the electrical stability is lower than expected
or if the high temperature/high pressure filtrate is higher than expected
or if the addition of water has already been recommended
Note that emulsifier has its major effect on electrical stability
and that it should be added only after water and weight material additions

Figure 4-1: A specification of a treatment plan

The first condition element of this sample rule indicates that a decision has been made to generate a treatment plan to deal with the fact that a salt formation is being drilled. Since the relevant set of considerations for each treatment varies across the different mud systems, the second condition element specifies the mud systems to which the rule applies.

Each TREATMENT created by a rule must have several critical attribute fields instantiated. ↑HYPOTHESIS has the name of the HYPOTHESIS on which the TREATMENT bears; ↑NAME provides an abbreviated name for the TREATMENT itself; ↑OPERATOR is used to classify the TREATMENT taken; ↑OPERAND describes in generic or functional terms, the kind of thing which is being added, or in general, is the recipient of the action specified as the ↑OPERATOR. ↑DIRECTION indicates how the action will affect the mud property specified in the ↑FOR field; ↑LABEL provides a unique numeric identifier for the TREATMENT; and ↑PLAN provides a link between all TREATMENTs which are part of a single plan.

The DATAFOR working memory element links the TREATMENT to the evidence that will warrant its recommendation. If the DATAFOR network dominated by the TREATMENT evaluates to true, the TREATMENT is accepted. This is indicated by setting ↑ACCEPT to YES. In the above example, the TREATMENT will be accepted if any of terminal DATAFORs evaluate to true. DATAFORs are evaluated as described above in section 3.4.

In addition to generating a functional description of a treatment plan, rules such as those shown in figure 4-1 create an ADDS working memory element which indicates the order in which the amount of each recommended additive must be calculated. In the provided example, the order of calculation is water, weight material, and emulsifier. Ordering is important as each treatment must be sensitive to other actions taken on the drilling fluids system. Ordering guarantees that MUD will know about these effects at the correct time. Weight material, for instance, must be added after dilution occurs in order to bring the density of the drilling fluid back to its desired state.

As with our approach to diagnosis, we chose to represent treatment knowledge as data generated by the action part of a rule, rather than as conditions on the execution of an action. In other words, we chose not to write rules in which the conditions expressed by a DATAFOR network were represented in the conditional part of a rule and the action as the creation of a TREATMENT. The reasons for this are very similar to those discussed below in section 5.3.

A treatment is typically pursued when doing so will reduce the difference between the current measurement of a mud property and the desired target. Thus, before evaluating the local adequacy of a TREATMENT, MUD considers if it is necessary to specify new targets for mud properties, rather than to merely restore the system to previous target specifications. For example, if there is an influx of water into the borehole, it is typically desirable to raise the MUD's target density by a half pound per gallon. The rule responsible for this action is shown in figure 4-2.

```
(P ADJUST-TARGET::DENSITY-FOR-FLOW
   (TASK ↑NAME ADJUST-TARGETS
       ↑CONTROL {<NAME> <<  FORMATION-HYDROCARBONS HYDROCARBONS
                            WATERFLOW FRESH-WATER-FLOW SALT-WATER-FLOW >>})
   (DATA ↑NAME DENSITY ↑TARGET {<TARGET> <> NIL} ↑VALUE {<VALUE> <> NIL})
  {<T> (TREATMENT ↑FOR DENSITY ↑TARGET NIL)}
   (HYPOTHESIS ↑NAME <NAME> ↑PROBLEM-PERSISTING YES)
   -->
   (CALL COMP <VALUE> + 0.5)
   (BIND <VALUE> (ANSW))
   (CALL TEST <VALUE> > <TARGET>)
   (BIND <ANSW> (ANSW))
   (MODIFY <T> ↑TARGET (IFTHENELSE <ANSW> TRUE <VALUE> <TARGET>)))
```

English translation:

If adjusting targets because there has been an influx of hydrocarbons or water
and no new target has been provided for density
and the problem is persisting,
then make the targeted density the greater of the current target or the current density incremented by .5 ppg

Figure 4-2: A target adjustment rule

4.2 Specifying Treatments

Once a decision has been made to pursue a treatment, MUD chooses specific chemical additives or equipment which will meet the functional requirements of a treatment plan. Since the selection of appropriate drilling fluids equipment differs little from the selection of a material additive, we discuss only the latter.

Chemical additives are selected by consulting a database of inventory information. In the initial version of MUD, an external ascii file was used to hold this information.[11] MUD creates INVENTORY working memory

[11] NL Baroid is in the process of creating a database to maintain this information.

elements, shown in figure 4-3, from data in this file. In the example, the key fields for information about Barite, a trade name product, are shown. In addition to this information the INVENTORY working memory element has data regarding standard dosages, cost, inventory availability, and a number of conversion factors which allow MUD to easily translate between different units in which the amount and cost of Barite might be described.

```
(INVENTORY
    ↑PRODUCT BAROID
    ↑FUNCTION WEIGHT-MATERIAL
    ↑GENERIC-NAME BARITE
    ↑SPECIFIC-GRAVITY 4.2)
```

Figure 4-3: An INVENTORY working memory element

As shown in figure 4-4, MUD searches the set of INVENTORY working memory elements, looking for products that could meet the functional goals of each treatment. When the ↑FUNCTION attribute of an INVENTORY working memory element corresponds to the ↑OPERAND field of a TREATMENT, MUD creates a DO working memory element as a potential treatment specification. Values of the ↑FUNCTION/↑OPERAND fields include things such as emulsifier, weight-material, and so forth.

```
(P ADD::GET-INVENTORY
    (TASK ↑NAME ADD ↑CONTROL <FUNCTION> ↑ATTRIBUTE <PLAN>)
    (DO ↑OPERATOR ADD ↑OPERAND <FUNCTION> ↑SPECIFICATION NIL ↑PLAN <PLAN>)
    (INVENTORY ↑PRODUCT <NAME> ↑FUNCTION <FUNCTION>)
  - (DO ↑SPECIFICATION <NAME> ↑PLAN <PLAN>)
    -->
    (MAKE DO ↑2 (SUBSTR 2 2 INF) ↑LABEL (GINT))
    (MODIFY 2 ↑SPECIFICATION <NAME>))
```

English translation:
If determining what materials to add
and inventory items that can accomplish a current goal have not yet been selected,
then select those items

Figure 4-4: Selecting from inventory

The value of the TREATMENT's ↑PRODUCT field becomes the value of the DO's ↑SPECIFICATION field. Barite, the name of a specific product, would be a possible value of an INVENTORY ↑PRODUCT field. When there are several additives (INVENTORY working memory elements) of the same functional type, MUD will create several corresponding DO working memory elements.

For each material instantiated in a DO working memory element, MUD calculates the amount required for an effective treatment. When there are multiple DOs with the same ↑OPERAND (representing alternative treatments), MUD chooses the best one, as described below.

4.3 Determining Amounts

Mud engineers use a variety of procedures to determine how much of a material additive ought to placed into the drilling fluid system to bring mud properties back to their desired target levels. In some cases, heuristic considerations are used to come up with a per barrel dosage, as in figure 4-5. In other cases, the deviation between a current measure on the mud system and a desired target can be used to arrive at the exact amount of material needed to eliminate the difference, as in figure 4-6.

In interviewing mud engineers we found that they used many heuristic procedures for determining amounts. For example, while engineers could tell us that the dosage of invermul increased with temperature, they could not give us an algorithm for calculating an exact dosage given a particular bottom hole temperature. They were, however, able to tell us that the the rate at which they increased a dosage depended on whether they were drilling under high or low bottom hole temperature conditions. Dosage increased more rapidly when the temperature was over 350 degrees farenheit. We also found that under high temperature conditions, engineers held to a minimum dosage of 14 ppg. Since mud engineers did not use algorithms for calculating the dosage of invermul, but rather used an "intuitive feel" for an appropriate dosage, we had to find algorithms that would approximate their subjective assessment of dosage requirements. The rule in figure 4-5 resulted from this attempt. The action part of the rule specifies a function which computes a dosage, given temperature as an argument. The result of this function approximates the amounts drilling fluid engineers would recommend.[12] A rule similar to this one was used to calculate an invermul dosage under low temperature conditions. Of general interest here is the way heuristic knowledge and algorithmic procedures are combined. We use the mud engineers' descriptions of the factors which affect treatment as condition elements, while attempting to design an empirically sound algorithm for each such set of factors.

In other cases, mud engineers did use relatively precise algorithms to calculate dosage, or the total desired

[12] We used an informal procedure with a limited sample to arrive at this formula. We expect formulas of this type to be reassessed during field testing of the system.

```
(P ADD::INVERMUL-HIGH-TEMPERATURE
    (TASK ↑NAME ADD ↑CONTROL EMULSIFIER ↑ATTRIBUTE <PLAN>)
    (DATA ↑NAME MUD-TYPE ↑VALUE INVERMUL)
   {<DO-INVERMUL>
    (DO ↑OPERATOR ADD ↑SPECIFICATION INVERMUL ↑DOSAGE NIL ↑AMOUNT NIL ↑PLAN <PLAN>)}
    (DATA ↑NAME TEMPERATURE ↑VALUE {<TEMP> > 350})
    -->
   •(CALL COMP [ [ 2 * <TEMP> ] / 25 ] - 18)
    (BIND <ANSW> (ANSW))
    (CALL TEST <ANSW> < 14)
    (BIND <TEST> (ANSW))
    (BIND <DOSAGE> (IFTHENELSE <TEST> TRUE <ANSW> 14))
    (MODIFY <DO-INVERMUL> ↑DOSAGE <DOSAGE>))
```

English translation:
If adding invermul
    and no dosage has yet been calculated
    and the temperature is greater than 350 degrees farenheit,
then use the lesser of the prescribed dosage at that temperature or 14 lbs per gallon

Figure 4-5: A dosage setting rule

amount of an additive. Figure 4-6 shows a rule in which the deviation between the current oil/water ratio and the desired oil/water ratio is used to arrive at the exact amount of oil needed to eliminate the difference. In this rule MUD makes use of a 'virtual' representation of the drilling fluid system. As MUD makes decisions about the amount of material that is to be added, it has to update a 'virtual' representation of the drilling fluid. In particular, the effects of each additive on density, volume, and oil-water ratio must be carefully tracked because this information is used in the calculation of subsequent treatments and may reach levels where problems, such as fracturing and pit overflow, can occur.[13] A virtual representation of a drilling fluid property with respect to a partially executed treatment plan is maintained in a vector working memory element at a position indexed by the value of ↑PLAN. In the example shown, the volume of the system after oil is added is maintained by modifying the working memory element, VIRTUAL-VOLUME. Since MUD may be generating alternative treatment plans, the change in volume must be associated with the current treatment plan alone. This is done by modifying the value indexed by <COUNTER>, the current treatment plan's identifier.

---

[13] Adequate actions in response to these problems have not yet been implemented.

```
(P ADD::AMOUNT-OF-OIL
   (TASK ↑NAME ADD ↑CONTROL OIL)
   (DATA ↑NAME MUD-TYPE ↑VALUE INVERMUL)
  {<DO> (DO ↑OPERATOR ADD ↑OPERAND OIL ↑SPECIFICATION <> NIL ↑PLAN <COUNTER> ↑AMOUNT NIL)}
   (TREATMENT ↑OPERAND OIL ↑PLAN <COUNTER> ↑TARGET <NEW-R>)
   (DATA ↑NAME SOLIDS ↑VALUE {<S> <> U} ↑VERIFIED YES)
  {<VOLUME> (VIRTUAL-VOLUME)}
  {<OIL-WATER> (VIRTUAL-OIL-WATER-RATIO)}
   -->
   (BIND <VT> (SUBSTR <VOLUME> <COUNTER> <COUNTER>))
   (BIND <R> (SUBSTR <OIL-WATER> <COUNTER> <COUNTER>))
   (MODIFY <OIL-WATER> ↑<COUNTER> <NEW-R>)
   (CALL COMP <VT> * [ 1 - [ <S> / 100 ] ])
   (BIND <VL> (ANSW))
   (CALL COMP <VL> * [ <NEW-R> - <R> ] / [ <R> + 1 ])
   (MODIFY <DO> ↑AMOUNT (ANSW)))
```

English translation:
If adding oil to an invermul mud
and no amount has yet been calculated,
then calculate the amount of addition using
the new targeted and current oil/water ratio, the current solids content, and the current volume

Figure 4-6: An amount setting rule

4.4 Evaluating Alternative Treatments

Once amounts are determined for each additive, the best choice among additives with the same function is determined. A heuristic polynomial evaluation function is used to determine a weight that predicts the best choice. As figure 4-7 shows, this weight a function over the cost of using the additive, its potential side-effects, and its availability (i.e., whether it is in stock or not).

```
WEIGHT =  - (COST-WGT) (COST-OF-ADDITIVE) +
          - (SIDE-EFFECTS-WGT) (SIDE-EFFECTS) +
          (AVAILABILITY-WGT) (AVAILABILITY)
```

Figure 4-7: The best choice heuristic

MUD uses a linear function in which the coefficients of the function (the -WGT terms) are provided as inputs to MUD. The substantive parameters, COST-OF-ADDITIVE, SIDE-EFFECTS, and AVAILABILITY are calculated as values on on a 0 - 100 scale. Where side effects are relative to some quantitative characteristic, such as amount, it is not difficult to find a reasonable mapping to a quantitative parameter representing the side effects in our heuristic equation. Where this is not the case, we let mud engineers provide quantitative subjective assessments of the side effects associated with the use of alternative products.

4.5 Explanation

As discussed in section 2.2, MUD can provide two kinds of explanations about its recommended treatment plan. If the user asks to have the recommended amounts explained, the nature of the explanation offered depends on the function of the additive. For instance, if an emulsifier is added, the standard recommended dosage at a particular temperature is indicated, together with any factors that lead to a modification of that dosage. If a weight material is added, the amount is explained in terms of that needed to increase the density from some current value to a desired target value.

MUD includes specialized rules for each of a number of canonical types of explanation. The condition elements for each of these rules often include references to special purpose working memory elements that maintain information that is relevant to the explanation but would not otherwise be preserved by the MUD system given its existing data representations. While this approach gave us a great deal of flexibility during development, especially as we had to learn gradually about what was desired in an explanation, the approach leads in the end to too many unwieldy interdependencies between rules.

The user may also ask MUD why it chose the additive it did from among the alternatives. Answers to this question are easily produced by rules which compare the values assigned each alternative additive with respect to each of the factors in MUD's heuristic evaluation function, figure 4-7.

We also wanted to provide explanations to questions, such as, "What would I need to do if salt contamination occurred?" As discussed with respect to the explanation of diagnostic reasoning, conditional explanations of this kind allow access to MUD's knowledge base outside of a performance context. Since the structure of MUD's treatment generation rules corresponds to the structure of its diagnostic rules, an explanation of treatments at a functional level could be easily provided by minor enhancements to MEX, the explanation system described above in section 3.5.1. Explaining how MUD calculates how much of a material additve to use would, however, be much more difficult, as many different kinds of rules are used to do this. Implementing this capacity would probably require substantial modifications to MUD.

5. Discussion

5.1 Why MUD Works

The current version of MUD assumes that all data entered, as a well as the recognition of diagnostically significant observations, is certain. So far this has not degraded performance. This is surprising. We had expected that MUD would need a quantitative way of both recognizing the likelihood of a deviation in a mud property and of transmitting evidential uncertainty to its hypothetical conclusions. In fact MUD is designed to allow this functionality, with few modifications, if it becomes desirable.

MUD seems able to succeed, with its assumption of evidential certainty, because its diagnostic procedure is robust in two respects. First, there are typically several diagnostically significant evidents which can evoke a hypothesis. If a problem occurs, it is likely to push at least one mud property across a detection threshold. Thus, uncertainty in the data is unlikely to cause MUD to miss the occurrence of a disruptive event. Secondly, as MUD weighs several evidential considerations together in coming to a conclusion with respect to any hypothesis, small errors in some fraction of these observations may wash out given a preponderance of evidence for or against the hypothesis.[14] Indeed this might explain why MUD engineers themselves do not need to rely on mathematical models for handling uncertainty. The following analysis of MUD's diagnostic procedure and knowledge base supports this conclusion.

The strength of a diagnostic conclusion in MUD is a function of the difference between accumulated measures of belief and disbelief. As discussed above, each measure results from an incremental function which operates over the positive or negative evidential weights associated with each REASON. These weights range from 0 to 10, with 0 indicating no contribution to the relevant belief measure. These values are subjectively assigned by domain experts.

If a high measure of belief results from at least one highly weighted REASON, one can accept the hypothesis with confidence, provided that there is an absence of contradictory evidence. However, if a strong belief

[14] One place where it is necessary to be careful is when a mud property with a high negative-support value is near a detection threshold. In these cases, MUD warns the user, but does not alter its diagnostic conclusion as the detection threshold, set by the engineer, should take into account the desirable tradeoff between false-positive and false-negative responses. If the latter is of concern, the threshold can be lowered.

results from the accumulation of many small weights, one can be sure, only under special conditions, that the results are due to the hypothesized problem and not to one or more other problems.

First of all, confidence is warranted when there is a high degree of differentiation in the consequences of the potential problems responsible for diagnostic symptoms. The more differentiation, the more likely it is that belief will accumulate more toward one hypothesis than another.

Secondly, one can be confident if it proves possible to reject alternative hypotheses. A hypothesis can be ruled out when expected consequences of the problem fail to materialize. When evidential considerations are assigned large negative support values, a rule out strategy becomes a powerful diagnostic tool. Since MUD's evidential support function generates an additive measure of disbelief, a rule out strategy can also be used when there is an absence of several consequences, each of which has a low negative support value.

Finally, if one expects few concurrent problems with overlapping results, one can legitimately see significance in marginal differences between potential explanations. Thus, if this condition is met, one can accept a hypothesis grounded on many weak evidential sources, provided the measure of belief is marginally above any alternative.

MUD's ability to achieve high levels of confidence in its diagnostic conclusions will thus rest largely on how domain experts assign weights to evidential considerations. MUD's performance will depend on the extent to which each potential hypothesis has some evidential considerations with high positive and negative support weights. In addition the amount of differentiation between hypotheses with respect to associated evidential considerations and the likelihood of concurrent problems will affect MUD's performance.

And indeed, it appears that MUD performs well and robustly because the above conditions hold extremely well across most hypotheses. Most hypotheses appear to have at least one consideration that carries significant positive import. In 17 out of the 20 problem types MUD currently knows about, one evidential consideration has a weight greater than or equal to 8, MUD's threshold of acceptance under normal conditions.[15] There is also a substantial degree of differentiation. Of the 20 problems, there are only 3 for which there are alternative hypotheses that would explain at least half of their potentially supportive

[15] When lacking evidence, or when faced with unexplained inconsistencies, MUD resorts to more complicated decision rules. Some of these capabilities are still under development.

evidence. This means that evidence for the correct conclusion is unlikely to lend much credence to alternative hypotheses. Thus, even when some potential evidence is degraded or absent, there tend to be other discriminating considerations, sufficient to drive a diagnostic conclusion in the right direction. In addition, MUD's diagnostic conclusions are driven by a rule-out strategy that is supported by the high expectation of observing symptoms associated with particular problems. For 17 of the 20 problems, the failure to observe a key consequence would lead to the belief that that problem had not occurred. In 2 of the remaining cases, the potential exists for rejecting a hypothesis on the failure to observe more than one expected consequence.

In summary, a consideration of evidential weights proves to be a robust diagnostic procedure in the mud domain. One expects either to find observations that strongly support a unique conclusion, find that the evidence converges on a unique hypothesis, or successfully rule out competing, but false, hypotheses. These results do more than confirm some obvious intuitions -- they provide a basis for pushing the analysis one step farther. Just what kind of evidence can push confidence factors in the directions required for robust diagnostic performance?

5.1.1 Where confidence factors come from

During interviews with mud engineers, the most significant factor in determining positive-support values appeared to be the number of alternative hypotheses that could account for the same symptom. Where only one hypothesis could explain a symptom, the assigned weight tended to be at the ceiling (10). Otherwise, the assigned weight generally declined with the number of alternative explanations. But if the number of alternative hypotheses were the only consideration, we would expect the symptom to be assigned the same weight with respect to any hypothesis that could explain it. This turns out not to be the case. The assignment of weights also seems to depend on the relative frequency with which a particular symptom is due to one problem as opposed to another, with the higher weight assigned to the REASON linked to the more likely hypothesis.

In contrast, negative-support weights seem to reflect the degree to which a symptom can be expected, given the occurrence of a particular problem. This expectation is thought to reflect the frequency with which a particular problem leads to a particular symptom. When these transitional likelihoods are high, the failure to observe a symptom is significant evidence that the hypothesized problem has not occurred. A look at the actual negative-support weights assigned by domain experts suggests, however, that transitional likelihood is not the only factor of concern. There is also an element of caution that appears to have its source in procedural uncertainties inherent in taking mud samples and doing field tests. One must take any observation with a grain of salt. This has the effect of lowering negative-support values.

5.2 Implications for Knowledge Acquisition

One of the major problems for a knowledge engineer in a diagnostic domain is learning enough about the domain to drive the interview process effectively [Boose 84], [Davis 82]. In light of our analysis of MUD and discussions with domain experts, we found ourselves relying on a small set of interview strategies that appeared to rapidly lead to a more powerful rule base. These strategies were based on our understanding of MUD's performance, as described above.

For purposes of discussion these strategies can be labelled as:

- differentiation
- frequency conditionalization
- symptom distinction
- symptom conditionalization
- path division
- path differentiation
- test differentiation
- test conditionalization

As a simplification, we describe these strategies in terms of symptoms, diagnosable events, and background conditions. A *symptom*, or symptomatic event, is any event or state consequent to the occurrence of a *diagnosable event*, also referred to as a cause or hypothesis. A *background condition* is any other event or state that affects the diagnostic significance of a symptom. Included here are further differentiating characteristics of the symptom, itself. A reported symptom is a symptom already pointed out by a domain expert and incorporated into the growing knowledge base.

The first three of these strategies are well known to knowledge engineers. *Differentiation* implies seeking for symptoms that provide leverage in distinguishing among diagnosable events. Most powerful in this respect are symptoms which result from a unique diagnosable event. These symptoms have maximally high positive-support values. However, increased differentiation in the knowledge base also results from incorporating symptoms which are explainable by a set of causes different (at least in part) from those underlying previously reported symptoms. For instance, in the MUD domain, both an influx of water and an insufficient use of emulsifier can have the same effects on measurable mud properties. However, an increase in mud volume is usually associated with the former. While this effect can also result from a hydrocarbon influx, other shifts in mud properties distinguish hydrocarbon from water influxes. Thus, the knowledge base can be further differentiated by adding the fact that an increase in volume is a confirming observation with respect to a water influx. Correspondingly, this increases the likelihood that evidence will converge on the actual cause.

Frequency conditionalization is a matter of determining if there are background conditions under which a particular cause is more or less likely to occur. The more these conditions lead to the expectation of a particular cause, the greater the confirmatory significance of a related symptom. For instance, in the MUD domain, an increase in viscosity often results from drilling through one of a number of contaminants, some of which may be expected, others unexpected, in the location being drilled. Thus, one would like the evidential significance of a symptom, such as an increase in viscosity, to be dependent on local knowledge about the likelihood of encountering various contaminants.

Symptom distinction requires seeking out special characteristics of a symptom that identify it as having been caused by one as opposed to other causal events. For instance, in the MUD domain both an influx of water and an increase in low specific gravity solids can cause a decrease in density. However, if density has decreased rapidly, it is more likely to have been due to an influx of water.

The remaining five strategies, as far as we know, are less familiar to knowledge engineers. *Symptom conditionalization* provides a way to increase the negative-support or disconfirmatory values of existing symptoms and consequently allows greater reliance on a rule-out strategy. Negative-support values, as discussed above, are proportional to the expectation that a diagnosable event will indeed give rise to a particular symptom. This expectation can be low if, for instance, the appearance of a symptom requires the co-occurence of a background condition. In the MUD domain, for example, some viscosity effects normally associated with salt contamination of a water based drilling fluid will appear only if the fluid has not been pretreated with surfactant thinners. If there has been a pretreatment of this kind, the failure of viscosity symptoms to appear cannot count as evidence against the hypothesis of salt contamination. However, if one knows that the system has not been pretreated in this way, then the disconfirmatory significance of failing to observe these viscosity symptoms is much greater than it would be otherwise. Put more generally, symptom conditionalization involves interviewing domain experts for the nature of conditions whose occurrence may either be required for, or may thwart, the appearance of a particular symptom. In the latter case, knowledge that the condition has not occurred would be reason for attributing greater significance to the absence of the symptom.

Path division also leads to stronger rule-out oportunities. This strategy requires eliciting a symptomatic event that lies on a causal path from the diagnosable event to an already reported symptom. The new symptom must be selected such that it is more expected, given the cause, than the reported symptom.[16] As such, the failure to observe it will be of greater disconfirmatory value *ceteris paribus* than failing to observe symptoms later in the causal chain. In the MUD domain, for example, the failure to observe an increase in viscosity is less disconfirmatory with respect to shale contamination than the failure to observe a significant increase in free bentonite through the use of a methylene blue test. An increase in bentonite can be considered an intermediate step between shale collapsing into the bore hole and a change in viscosity.

Path differentiation is a means of finding symptoms with high positive-support values. With this strategy, the knowledge engineer determines if a symptom, which may result from one of several causes,[17] does so via (at least partially) non-overlapping causal pathways. Intermediary events on non-overlapping portions of these pathways are expected to have a higher positive-support value than symptomatic events on shared pathways. For instance, in the MUD domain, an increase in plastic viscosity in an oil mud can result from either salt or water contamination. These effects, however, do not result in entirely the same way. Shale contamination causes an increase in plastic viscosity by increasing the percentage of solids in the mud system; water causes an increase by its behavior in a partially emulsified solution. The mud engineer can determine which of these mechanisms accounts for increased plastic viscosity through the use of additional tests. These tests measure the amount of unemulsified water and the solids content of the mud. Positive results on these tests provide stronger confirmation of the respective causes than does the shared symptom of increased plastic viscosity.

---

[16] This will be the case provided that the expectation of the reported symptom given the new symptom is not 1 and there are not alternative pathways from the cause to the reported symptom, some which do not pass through the new symptom.

[17] Such symptoms have low positive-support values.

Test differentiation and test conditionalization provide ways of strengthening the confidence in the observation that a symptom has occurred or not. As discussed above, the significance of a symptom with respect to confirming or disconfirming hypotheses is sometimes less than it might be because the procedure for determining its occurrence is unreliable. In these cases, the knowledge engineer can seek out conditions under which the reliability of the observation can be more readily taken for granted.

Test differentiation is a matter of distinguishing the reliability of different tests. In the MUD domain, for instance, the significance of changes in pH level differ slightly depending on whether pH is measured by litmus paper or the more accurate pH meter. Test conditionalization is a matter of determining the conditions under which the use of a particular test, or eye observation, is more accurate. Some tests, for instance, are more accurate when measurements are made within a particular range of values. Other possibly relevant conditions include consistency with other measures and the experience of the performing technicians.

5.2.1 Automating knowledge acquisition

The fact that our attempts at knowledge acquisition in the MUD domain became more structured over time and converged on the eight distinct strategies discussed above encouraged us to explore the potential for an automated interviewer. Such a tool would have the potential of not only relieving the knowledge engineer of the interview burden, but also of allowing a rapid assessment of the strength of the current knowledge base, and a selection of interview questions designed to compensate for these weaknesses. In addition, by maintaining a mapping between the knowledge base and rules, the system could potentially recognize rules with unexpected weight assignments. Our recent work in this area is reported in [Kahn 84b].

5.2.2 The adequacy of MUD's knowledge

The above analysis gives us a tool for pursuing further knowledge acquisition. We can expect to improve the performance of an evidential system by

1. finding observations with higher transitional likelihoods,
2. finding observations with fewer potential explanations,
3. collecting additional observations with an eye toward increasing the evidential differentiation of alternative hypotheses.

Although the amount of knowledge in MUD already appears high enough to achieve competent levels of performance, expanding knowledge in the directions suggested should result in still higher levels of performance. Unfortunately, we encounter several limitations to so doing.

Where the negative weight of an observation is low, identifying an intermediate event on the pathway from initiating cause to the former observation provides an evidential consideration with a potentially stronger transitional likelihood.[18] The attempt to identify such intermediary events in the MUD domain, however, typically fails for two reasons. The first is that in general there is an absence of procedures for detecting these events under field conditions. For example, many contaminants, such as shale cuttings and salt, affect gelation and viscosity properties of the drilling fluid by altering its molecular structure. The essential events that occur between the onset of contamination and the field tests which measure gelation and viscosity take place at the molecular level. Detecting these events in a way that would allow discrimination among potential contaminants would be a powerful aid to diagnosis. Unfortunately, the appropriate tests require equipment and expertise that would be impractical in the field. A similar situation generally holds for all properties that reflect chemical changes induced by contaminants.

In addition, the pursuit of further causal knowledge is hindered by the many significant evidential considerations that involve the recognition of physical events that are closely linked to the occurrence of a problem. For instance, a significant drop in mud density is a strong indicator of an influx of gas or water under some conditions. Where events have this kind of physical contiguity, there just are no further intermediary events to draw on.

Although the potential for causal refinements is limited, negative weights associated with already recognized symptoms can be increased by identifying the absence of factors that would normally interfere with their expression. For instance, MUD has a slight disposition to deny that an influx of water has occurred when there is no observed increase in mud volume. This disposition is weak because influxes at one depth can be masked by lost circulation at another. However, if the possibility of lost circulation has been rejected, then when there is no increase in volume, MUD can confidently deny that there has been an influx of water.

In many cases, we have been able to strengthen MUD's performance by attempting to clarify the reasons why domain experts assign weak negative weights to potential symptoms. This procedure, however, can only be carried so far. It is not possible to push these weights to a negative certainty for several reasons. For one, the observation of a consequence often has to do with the degree to which a problem is occurring. Secondly,

---

[18] This will be true unless the negative weight of the former observation reflects transitions along multiple causal pathways.

domain experts are unable on many occasions to exhaustively list all the potential reasons for failing to observe an expected consequence. Third, procedures for ascertaining that an interferring condition has occurred or not frequently do not exist. And finally, domain experts seem to map procedural uncertainties into negative weights as well. Since mud samples may not accurately relect down hole conditions or may be unreliablely measured, the expert diagnostician cautiously assigns diagnostic significance to many tests.

Finding events or states to which one can assign greater positive support would proceed by distinguishing pathways by which alternative problems result in shared consequences. Events on the distinct pathways would presumably be explained by fewer hypotheses, and thus have higher positive weights. Distinguishing such pathways and events, however, fails for the very reasons discussed above with respect to achieving higher transitional likelihoods. Pathways are either indistinguishable, or events on these pathways are undetectable.

In the above we have suggested the kind of analysis required to improve diagnostic performance. As a result of examining the potential for expanding MUD's knowledge of diagnostically significant observations, our view is that MUD does about the best job it can, with respect to the problems it knows about.

5.3 MUD as a Production System

MUD is implemented in OPS5, a general purpose production system language [Forgy 81]. Unlike many OPS programs, MUD does not take full advantage of the OPS interpreter, as can be seen from the structure of the sample diagnostic rules. The significant part of the content of such a rule is in its action part; the effect of applying the rule is to place a description of the evidence supporting a hypothesis into working memory. Other more general rules match this data and provide the capabilities for seeking and evaluating evidence, as well as deciding among hypotheses. Alternatively, one can imagine rules that exploited the OPS pattern matcher by placing the evidential descriptions in the conditional part of the rule and using the action part of the rule to evaluate these with respect to their bearing on a REASON. The decision here can be stated as one between explicitly representing evidential relations in working memory (the more declarative approach), and implicitly representing them in the conditions and actions of specialized rules (the more procedural approach). We found several reasons for preferring the former: it makes it easier to expand and maintain the knowledge base, explore alternative approaches to various problems, and access the knowledge base outside of a performance context.

Expanding the knowledge base manually is easier the less one has to be concerned with issues of consistency and control. These problems can arise in many places in a diagnostic system, but most readily occur in guaranteeing that the data required to draw a diagnostic conclusion is available when desired. Backward chaining systems solve this problem by examining the conditions of the rules that would provide support to the hypothesis in question and instantiating those that have unknown values. OPS does not provide this capability directly. Instead, for the procedural approach to work, there must be rules that generate data when it is needed by other rules. Since these rules are distinct from those that actually use the data to make inferences, the burden of maintaining consistency and adequate sequential control is on the programmer. This is especially worrisome when novice programmers will be augmenting the knowledge base with new diagnostic rules -- a situation we faced with MUD. With the declarative approach taken by MUD, neither consistency nor control issues need be of concern when new evidential rules are entered into the system. Since there is only one rule for each evidential consideration, the consistency issue goes away. Since the general rules which interpret the diagnostic rules provide all the functionality of a backward chaining inference engine, no additional control issues arise.

The declarative approach also makes it easier to explore alternative solutions to critical problems. One of these, for example, is that of handling ignorance -- what to do when some desired datum is uninferable by the system and unknown to a user. Under such conditions one would like to make an intelligent guess or point out to the user just what information is required to make an accurate diagnostic assessment. This cannot be done when the missing datum is represented in the conditional part of a rule; the description is simply inaccessible. Again, with the procedural approach, one can compensate for this by creating rules whose conditions constitute the powerset of potential partial matches. Each rule indicates what to do under the conditions of ignorance defined in its conditional part. Not only does this result in a lack of conciseness in the program, but it proves a tremendous obstacle to exploring alternative global strategies for dealing with ignorance. Instead of replacing a very few general rules, as with the declarative approach, it is necessary to modify all the rules which were formulated to handle ignorance in many particular cases. By creating a working memory representation of evidential relations, a few general rules can provide a global strategy to problems of ignorance. In MUD some of these rules function as part of a multi-valued logic for assigning truth values to evidential networks; others to assess the pattern of ignorance, deciding what information is unnecessary or necessary to achieve an adequate diagnostic conclusion.

A declarative approach is also required if knowledge about evidential relations is to be accessed outside of actually running a diagnosis over some set of symptoms.. For instance, in the MUD domain, there are conditions under which engineers would like to ask conditional questions, such as, "How would one know if a salt formation were being drilled?". With the more procedural approach, answering such questions requires using canned text or an artificial trace which could be inconsistent with the rules that perform the actual diagnosis. MUD's more declarative approach allows us to answer such questions by letting specialized explanation rules read the same working memory elements that would drive the system during a diagnostic session.[19]

While our solution to the above problems is a workable one, it has not been efficient. A large working memory of evidential relations and a few general interpretive rules puts a strain on the OPS pattern matcher. One way of gaining the benefits in consistency and conciseness promised by the declarative approach as well as the efficiency promised by the procedural approach is to use both approaches, each at its appropriate time. We are currently exploring the possibility of developing rule generators that would essentially provide a mapping between the declarative and procedural representations. Building such a generator for diagnostic knowledge would allow for the creation of specialized instantiations of the general rules currently deployed by MUD; the input to such a generator would be a representation of the evidential relations required for diagnosing mud problems.

5.3.1 An account of MUD's rules

When released to NL Baroid, the MUD system had 826 rules. Of these, approximately 1/3 represented a general control mechanism, or procedures that made no assumptions about the domain in particular. Of the remaining 2/3's about half involved *domain-specific* knowledge. The remaining half, while not domain-specific, were *domain-dependent*; that is, they entailed assumptions about the representation of information in MUD, a representation that was constrained by requirements of the domain.

A classification of MUD's 268 domain-specific rules is given in figure 5-1. The large class of data related rules is composed of rules which either create a *data schema* for a particular datum, perform a procedure for

[19] These rules can also be used to provide a run-time explanation of diagnostic decisions. However, they have not yet been properly integrated into MUD. Thus, MUD still relies on canned text to provide run-time explanations.

inferring the value of datum, or create a set of requests for data from which a desired value can be inferred. A data schema is a working memory element that carries information about a particular kind of datum. This information includes the default units of an instantiated value, a descriptive phrase for the datum, a question to be used in asking for the current value of the datum, and constraints on acceptable values. We expect these rules to be augmented gradually as the scope of evidence MUD considers increases.

Data related rules:	126
Diagnostic rules:	68
Treatment rules:	52
Inconsistency checking:	19
Search Control:	3

Figure 5-1: MUD's domain-specific knowledge

The class of diagnostic rules includes both hypothesis generation rules and evidential rules (those which create a working memory representation of the evidential requirements for evaluating a hypothesis). We expect considerable growth in these rules as MUD's knowledge base is extended to handle additional mud types. We also expect considerable refinement of the current set of evidential rules. These refinements will be refinements in the logical description of evidential considerations.

Treatment rules generate treatment plans, calculate the amount or degree of a treatment, or generate explanation schemata for different kinds of treatments. We expect the number of rules which generate treatment plans to increase rapidly as MUD is used for other mud systems. We also expect considerable refinement of the existing treatment rules as the scope of considerations bearing on treatments is extended. For similar reasons, we expect rules which calculate the amount or degree of a treatment to increase substantially.

Inconsistency checking rules check for unlikely or impossible combinations of data. This rule set is likely to grow somewhat, but not to a great extent.

The small class of search control rules currently controls the order in which hypotheses are evaluated. As the MUD system comes to recognize larger classes of relevant hypotheses, it may be necessary to modify or add to these rules. However, we do not expect this to occur as MUD's early pruning strategies seem to successfully restrict the set of candidate hypotheses.

A classification of MUD's 303 domain-dependent rules is given in figure 5-2. The domain-dependent rules differ from the domain-specific rules in that they provide a variabilized schema of condition elements. While these rules are generalized, their form is considerably constrained by the requirements of the mud domain. Undoubtedly there are other diagnostic and treatment domains to which many of these rules would apply. However, their present form is a direct consequence of constraints imposed by the mud domain. While some of these rules correspond to domain knowledge, more typically, the rules exist to manipulate internal representations, control the behavior of the program, or explain the behavior of the program to the user.

Data base access and transfers	100
Trend analysis	63
Treatment handling	67
Evidential Assessment	35
User interface	38

Figure 5-2: MUD's domain-dependent knowledge

A large number of data base access and transfer rules are used to provide an interface between Digital's DBMS codycil database and MUD. Most of these rules map between the relatively simple database currently being used, and MUD's more complex representational structure.

Trend analysis rules are largely computational. They make considerable assumptions about the kind of data that requires analysis. While these rules are not restricted to the mud domain per se, they assume a representational structure constrained by the domain considerations which guided MUD's design.

Treatment handling rules provide generic procedures for selecting treatments and providing explanations. A user interface of 179 rules provides a general mechanism for interrogating the user and providing menus; 38 of these rules are used to tailor this interface to the requirements of the MUD domain.

6. Conclusions

The MUD system demonstrates the applicability of production system architectures to diagnostic and treatment tasks. Both these tasks are sufficiently well structured to allow incremental growth through the addition of context specific rules.

MUD's diagnostic approach, like that of many previous sytems, is to rely on evidential rules rather than a deep causal model. This was much preferred in the drilling fluids domain. MUD took a somewhat unusual approach to the representation of declarative knowledge in the context of a forward chaining system. So doing created the opportunity to provide explanation facilities not usually found in forward chaining systems, while at the same time making it more difficult to introduce inconsistencies into the knowledge base.

MUD's approach to providing treatments for diagnosed problems makes use of heuristic knowledge at several points within what is basically a well-structured task. Generating a treatment plan is well-structured in that there is an exact sequence of required tasks, namely, that of generating a functionally appropriate treatment plan, finding material additives or equipment with the required functions, figuring out how much of a material is to be used or for how long a piece of equipment is to be run, and choosing among alternative treatments with the same function. The rules for each of these subtasks, however, require considerable contextual sensitivity. A rule-based approach allows incremental growth and flexibility in achieving higher levels of competence with respect to each of these subtasks.

An analysis of why MUD works has shed some light on the prerequisites for the success of evidential approaches to diagnosis, and has resulted, as well, in a better understanding of knowledge acquisition strategies for diagnostic tasks.

References

[Boose 84] Boose, J.
Personal construct theory and the transfer of human expertise.
In *Proceedings of the National Conference on Artificial Intelligence.* Austin, Texas, 1984.

[Davis 82] Davis, R. and D. Lenat.
Knowledge-Based Systems in Artificial Intelligence.
McGraw-Hill, 1982.

[Davis 83] Davis, R.
Diagnosis via causal reasoning: paths of interaction and the locality principle.
In *Proceedings of the National Conference on Artificial Intelligence.* Washington, D.C., 1983.

[Forgy 81] Forgy, C.L.
OPS5 user's manual.
Technical Report, Carnegie-Mellon University, Department of Computer Science, 1981.

[Kahn 84a] Kahn, G.
On when diagnostic systems want to do without causal knowledge.
In *Proceedings of Advances in Artificial Intelligence.* Pisa, Italy, 1984.

[Kahn 84b] Kahn, G., S. Nowlan, and J. McDermott.
A foundation for knowledge acquisition.
In *Proceedings of IEEE Workshop on Principles of Knowledge-based Systems.* Denver, Colorado, 1984.

[Kahn 85] Kahn, G. and D. Geller.
MEX: An OPS-based approach to explanation.
Expert Systems Memo, Department of Computer Science, Carnegie-Mellon University. 1985

[Patil 81] Patil, R., P. Szolovits and W. Schwartz.
Causal understanding of patient illness in medical diagnosis.
In *Proceedings of the International Joint Conference on Artificial Intelligence.* IJCAI, 1981.

[Pople 82] Pople, H.
Heuristic methods for imposing structure on ill-structured problems.
In Szolovits, P. (editor), *Artificial Intelligence in Medicine.* Westview Press, 1982.

[Shortliffe 76] Shortliffe, E.
Computer-Based Medical Consultation: Mycin.
Elsevier, 1976.

[vanMelle 81] van Melle, W., A.C. Scott, J.S. Bennet and M.A. Peairs.
The EMYCIN manual.
Technical Report, Stanford University, Heuristic Programming Project, 1981.

[Weiss 78] Weiss, S., K.B. Kern, C.A. Kulikowski and S. Amarel.
A model-based method for computer-aided medical decision-making.
Artificial Intelligence (11), 1978.

Appendix A

NL Baroid

DRILLING MUD REPORT

REPORT NO.	DATE 19	DEPTH

API WELL NO.	STATE	COUNTY	WELL	S/T

OPERATOR	CONTRACTOR	RIG NO.
ADDRESS	ADDRESS	SPUD DATE
REPORT FOR MR.	REPORT FOR MR.	SECTION, TOWNSHIP, RANGE

WELL NAME AND NO.	FIELD OR BLOCK NO.	COUNTY, PARISH OR OFFSHORE AREA	STATE OR PROVINCE
BC #5			

OPERATION	CASING	MUD VOLUME (BBL)	CIRCULATION DATA	
PRESENT ACTIVITY	SURFACE IN. at FT.	HOLE / PITS	PUMP SIZE X IN.	ANNULAR VEL. (FT./MIN.) OPPOSITE DP
BIT SIZE (IN.) / NO.	INTERMEDIATE IN. at FT.	TOTAL CIRCULATING VOLUME	PUMP MAKE MODEL	OPPOSITE COLLAR
DRILL PIPE SIZE / TYPE	PRODUCTION OR LINER IN. at FT.	IN STORAGE	BBL/STROKE STROKE/MIN	CIRCULATING PRESSURE PSI
DRILL COLLAR SIZE / LENGTH	MUD TYPE INVERMUL OIL MUD		BBL./MIN.	BOTTOMS UP (MIN.) SYSTEM TOTAL (MIN.)

Sample from ☒ Flowline ☐ Pit
Flowline Tempeture °F 122.

MUD PROPERTIES	Min	Max	Obs
Time Sample Taken			
Depth (ft)	14000		
Weight ☒ (ppg) ☐ (lb/cu ft)	14.0	14.5	14.0
Mud Gradient (psi/ft)	.727	.752	.727
Funnel Viscosity (sec/qt) API at °F	40	50	39
Plastic Viscosity cp at 120 °F	40	45	39
Yield Point (lb/100 sq ft)	8	15	1
Gel Strength (lb/100 sq ft) 10 sec/10 min	2/5	5/10	0/2
pH ☐ Strip ☐ Meter			
Filtrate API (ml/30 min)	0	0	0
API HP-HT Filtrate (ml/30min) 300 °F	10	15	20
Cake Thickness 32nd in API ☐ HP-HT ☒	2	4	8
Alkalinity, ~~[illegible]~~ Lime lb/bbl	2	3	2
Alkalinity, Filtrate (P_f/M_f)	/	/	/
Salt ☒ ppm ☐ gpg Chloride ☐ ppm ☐ gpg X 1000	280	300	290
Calcium ☐ ppm ☐ Gyp (ppb)			
Sand Content (% by Vol)			
Solids Content (% by Vol)	25	28	25
Oil Content (% by Vol)	60	68	60
Water Content (% by Vol)	15	17	15
Methylene Blue Capacity (me/ml mud)			
Electrical Stability, volts	500	1200	1000
Oil/Water - ratio	80/20	88/19	80/20
Low specific gravity solids, % by vol.	11.0	15.0	10.0
Volume, surface bbls	1000	1000	1000

EQUIPMENT

	SIZE	HRS/TOUR				SIZE	HRS/TOUR		
Centrifuge					Desilter				
Degasser					Shaker				
Desander					Other				

DAILY COST	CUMULATIVE COST

MUD PROPERTIES SPECIFICATIONS

WEIGHT	VISCOSITY	FILTRATE

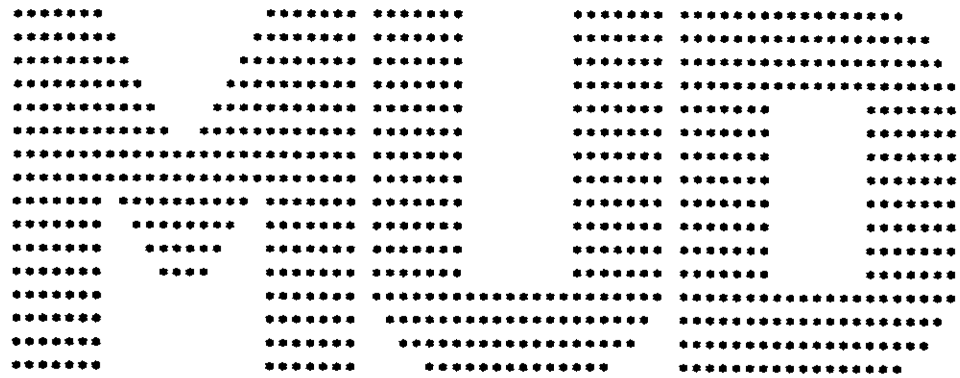
BY AUTHORITY ☐ OPERATOR'S WRITTEN ☐ DRILLING CONTRACTOR ☐ OPERATOR'S REPRESENTATIVE ☐ OTHER

MATERIALS RECOMMENDED

Add DURATONE AND GELTONE II

BAROID is settling and filtration high.

Appendix B



```
TYPE ANYTHING TO CONTINUE [ C ] :
                                WELCOME TO MUD
                               --------------
          To exit the program from any menu, type QUIT


                                  MODES
        To enter data                                       DATA
        Analyze data and form hypotheses                    ANALYSIS
        How to remedy problems                              TREATMENT
        To set options                                      OPTIONS
        To display this menu                                MENU
        To explain an mode                                  HELP [mode]
        To exit from the program                            EXIT

MODE [ MENU ] :d

What kind of mud are you using:invermul
What is the down hole depth:1300
What is the down hole temperature [ degrees f. ] :324

                           AVAILABLE SUBMODES
        For control information                             CON
        Information on the mud system                       API
        Inventory information                               INV
        Materials used and equipment changes                MAT
        Information on the bore hole                        BORE
        Information on the drilling system                  RIG
        Logging problems encountered                        PLOG
        Trends in mud system values                         TREND
        To create or modify a mudspec                       MUDSPEC
        To escape to the top menu                           EXIT

DATA [ MENU ] :api


                                API MODES
        To cycle through all entries                        CYCLE
        To enter a specific piece of data                   DATUM
        Escape to higher level                              EXIT
        Explain what a particular mode does                 HELP [mode]
        Display this menu                                   MENU
        To read data from a file                            READ
        To display values of data                           SHOW
        To write all data to a file                         WRITE
```

```
API [ MENU ] :r

                    DATA FILE TYPE TO READ
        Data is stored in a separate file               FILE
        Data is stored in a database                    DBASE
        Escape to higher level                          EXIT

READ [ MENU ] :f

Give file name [ API.DAT ] :[gsk.data]bc.2

   API data for well BC.2 for depth 3375 FT.
   Last updated on 28-JUNE-83 at 9:15


It is impossible for pf-filtrate to be above 0
when ph is below 8.3.
Since FILTRATE ALKALINITY (Pf) is .7 Pf and pH is 7,
a measurement error has probably been made.
Do you want to reenter data [ Y ] :n


READ [ MENU ] :e


API [ MENU ] :e

DATA [ MENU ] :mudspec


                         MUDSPEC MODES
        To cycle through all entries                    CYCLE
        To enter a specific piece of data               DATUM
        Escape to higher level                          EXIT
        To generate target values                       GEN
        Explain what a MUDSPEC mode does                HELP [mode]
        Display this menu                               MENU
        To read data from a file                        READ
        To display values of data                       SHOW
        To write all data to a file                     WRITE

MUDSPEC [ MENU ] :r

                    DATA FILE TYPE TO READ
        Data is stored in a separate file               FILE
        Data is stored in a database                    DBASE
        Escape to higher level                          EXIT

READ [ MENU ] :f

Give file name [ MUDSPEC.DAT ] :[gsk.data]bcplan.2

   MUDSPEC data for well BC.2 for depth 3200-3500 FT.
   Last updated on 25-JUNE-83 at 12:00


READ [ MENU ] :e


MUDSPEC [ MENU ] :e

DATA [ MENU ] :e


                              MODES
        To enter data                                   DATA
        Analyze data and form hypotheses                ANALYSIS
        How to remedy problems                          TREATMENT
        To set options                                  OPTIONS
        To display this menu                            MENU
        To explain an mode                              HELP [mode]
        To exit from the program                        EXIT

MODE [ MENU ] :a

Is there a database available for trend analysis ? (Y or N)  [ Y ] :n
 No trend analysis will be performed.
```

```
USE ↑ TO RETURN TO DATA MODE

USE < TO MODIFY DATA

Have we gained any volume:y
Are all surface additions of oil accounted for :y
Is there water in the hthp filtrate:y

REVIEW [ MENU ] :e


VALUES FOR THE FOLLOWING ARE ABOVE OR BELOW SET TARGETS

         LIME                                      L
         LOW SPECIFIC GRAVITY SOLIDS               L
         OIL WATER RATIO                           H
         ELECTRICAL STABILITY                      H
         LIQUID CONTENT                            H
         SOLIDS CONTENT                            L
         HTHP CAKE THICKNESS                       H
         10 MINUTE GEL-STRENGTH                    L
         10 SECOND GEL-STRENGTH                    L
         YIELD POINT                               L
         PLASTIC VISCOSITY                         L
         FUNNEL VISCOSITY                          L
         DENSITY                                   L

TYPE ANYTHING TO CONTINUE [ C ] :

THERE IS CONSIDERABLE EVIDENCE THAT:

 1:   THERE IS AN INFLUX OF HYDROCARBONS

 2:   THERE IS AN INFLUX OF FORMATION-HYDROCARBONS

 3:   THERE IS AN UNDER-DOSAGE OF GELTONE

      ALTHOUGH THERE IS SUFFICIENT EVIDENCE TO ACCEPT THE HYPOTHESIS,
      IT SHOULD BE NOTED THAT CONTRARY TO EXPECTATIONS:

      THE PPB AMOUNT OF GELTONE IS NOT LESS THAN THE AMOUNT TARGETED

 4:   THERE IS AN UNDER-DOSAGE OF LIME


TYPE ANYTHING TO CONTINUE [ C ] :


               EXPLANATION MENU
        DIAGNOSES                                          D
        SCOPE                                              S
        REASONS                                            R
        DATA UNACCOUNTED FOR                               X
        EXIT                                               E
[ M ] :d


H 1:  IT IS EXTREMELY LIKELY THAT
THERE IS AN INFLUX OF HYDROCARBONS.
THIS IS SUFFICIENT TO ACCEPT THE HYPOTHESIS.


H 2:  IT IS VERY LIKELY THAT
THERE IS AN INFLUX OF FORMATION-HYDROCARBONS.
THIS IS SUFFICIENT TO ACCEPT THE HYPOTHESIS.


H 3:  IT IS VERY LIKELY THAT
THERE IS AN UNDER-DOSAGE OF GELTONE.
THIS IS SUFFICIENT TO ACCEPT THE HYPOTHESIS.


H 4:  IT IS VERY LIKELY THAT
THERE IS AN UNDER-DOSAGE OF LIME.
THIS IS SUFFICIENT TO ACCEPT THE HYPOTHESIS.


H 6 :   IT IS VERY UNLIKELY THAT
```

THERE IS AN UNDER-DOSAGE OF INVERMUL.
THIS IS SUFFICIENT TO REJECT THE HYPOTHESIS.

H 6 : IT IS EXTREMELY UNLIKELY THAT
THERE IS AN INFLUX OF WATER.
THIS IS SUFFICIENT TO REJECT THE HYPOTHESIS.

H 7 : IT IS VERY UNLIKELY THAT
THERE IS AN UNDER-DOSAGE OF EZ-MUL.
THIS IS SUFFICIENT TO REJECT THE HYPOTHESIS.

H 8 : IT IS VERY UNLIKELY THAT
THERE IS AN OVER-DOSAGE OF EZ-MUL.
THIS IS SUFFICIENT TO REJECT THE HYPOTHESIS.

H 9 : IT IS EXTREMELY UNLIKELY THAT
THERE HAS BEEN AN OVERDILUTION WITH OIL.
THIS IS SUFFICIENT TO REJECT THE HYPOTHESIS.

H 10 : IT IS VERY UNLIKELY THAT
FORMATION SOLIDS ARE BUILDING UP IN THE SYSTEM.
THIS IS SUFFICIENT TO REJECT THE HYPOTHESIS.

H 11 : IT IS VERY UNLIKELY THAT
THERE IS AN OVER-DOSAGE OF INVERMUL.
THIS IS SUFFICIENT TO REJECT THE HYPOTHESIS.

H 12 : IT IS VERY UNLIKELY THAT
THERE IS AN OVER-DOSAGE OF DURATONE.
THIS IS SUFFICIENT TO REJECT THE HYPOTHESIS.

H 13 : IT IS VERY UNLIKELY THAT
THERE IS AN OVER-DOSAGE OF LIME.
THIS IS SUFFICIENT TO REJECT THE HYPOTHESIS.

H 14 : IT IS VERY UNLIKELY THAT
THERE IS AN OVER-DOSAGE OF GELTONE.
THIS IS SUFFICIENT TO REJECT THE HYPOTHESIS.

H 15 : IT IS EXTREMELY UNLIKELY THAT
THERE IS AN UNDER-DOSAGE OF DURATONE.
THIS IS SUFFICIENT TO REJECT THE HYPOTHESIS.

EXPLANATION MENU

DIAGNOSES	D
SCOPE	S
REASONS	R
DATA UNACCOUNTED FOR	X
EXIT	E

[M] :r

Which hypothesis would you like explained [1] :

H 1: THAT THERE IS AN INFLUX OF HYDROCARBONS
CAN BE ACCEPTED BECAUSE

R 1 : THERE IS AN INCREASE IN SYSTEM VOLUME

R 2 : THERE IS A DECREASE IN ELECTRICAL STABILITY

R 3 : THERE IS A DECREASE IN YP

R 4 : THERE IS A DECREASE IN LOW SPECIFIC GRAVITY SOLIDS

R 5 : THERE IS AN DECREASE IN DENSITY

R 6 : THERE IS A DECREASE IN PV

```
R 7 :   THERE IS A DECREASE IN GEL STRENGTH

      AND MORE SPECIFICALLY -

C 1 :   THE OIL-WATER RATIO IS UP

Which hypothesis would you like explained [ 2 ] :2

H 2:  THAT THERE IS AN INFLUX OF FORMATION-HYDROCARBONS
      CAN BE ACCEPTED BECAUSE

R 1 :   THERE IS EVIDENCE OF A HYDROCARBON INFLUX

      AND MORE SPECIFICALLY -

C 1 :   THE OIL WATER RATIO IS ABOVE EXPECTED GIVEN SURFACE OIL ADDITIONS

Which hypothesis would you like explained [ 3 ] :

H 3:  THAT THERE IS AN UNDER-DOSAGE OF GELTONE
      CAN BE ACCEPTED BECAUSE

R 1 :   THERE IS A DECREASE IN GEL STRENGTH

R 2 :   THERE IS AN INCREASE IN CAKE SIZE

R 3 :   THERE IS A DECREASE IN DENSITY

Which hypothesis would you like explained [ 4 ] :8

H 8:  THAT THERE IS AN OVER-DOSAGE OF EZ-MUL
      CAN BE REJECTED BECAUSE

R 1 :   THE PPB AMOUNT OF EZ-MUL IS NOT GREATER THAN THE AMOUNT TARGETED

Which hypothesis would you like explained [ 9 ] :

H 9:  THAT THERE HAS BEEN AN OVERDILUTION WITH OIL
      CAN BE REJECTED BECAUSE

C 1 :   THE OIL WATER RATIO IS ABOVE EXPECTED GIVEN SURFACE OIL ADDITIONS

Which hypothesis would you like explained [ 10 ] :

H 10:  THAT FORMATION SOLIDS ARE BUILDING UP IN THE SYSTEM
      CAN BE REJECTED BECAUSE

R 1 :   THERE IS NO INCREASE IN LOW SPECIFIC GRAVITY SOLIDS

Which hypothesis would you like explained [ 11 ] :6

H 6:  THAT THERE IS AN INFLUX OF WATER
      CAN BE REJECTED BECAUSE

R 1 :   THE OIL-WATER RATIO IS NOT DOWN

R 2 :   THERE IS NO INCREASE IN GEL

R 3 :   THERE IS NO INCREASE IN Y-POINT

R 4 :   THERE IS NO DECREASE IN ELECTRICAL STABILITY

Which hypothesis would you like explained [ 7 ] :4

H 4:  THAT THERE IS AN UNDER-DOSAGE OF LIME
      CAN BE ACCEPTED BECAUSE

R 1 :   THE PPB AMOUNT OF LIME IS LESS THAN THE AMOUNT TARGETED

Which hypothesis would you like explained [ 5 ] :e


                 EXPLANATION MENU
      DIAGNOSES                                          D
      SCOPE                                              S
      REASONS                                            R
      DATA UNACCOUNTED FOR                               X
      EXIT                                               E
[ M ] :x
```

```
THERE IS NO ACCEPTED HYPOTHESES WHICH CAN EXPLAIN THE FOLLOWING:

        THE INCREASE IN ELECTRICAL STABILITY
        THE INCREASE IN OIL WATER RATIO
        THE DECREASE IN YIELD POINT
        THE DECREASE IN PLASTIC VISCOSITY
        THE DECREASE IN LOW SPECIFIC GRAVITY SOLIDS
        THE DECREASE IN DENSITY
        THE DECREASE IN 10 MINUTE GEL-STRENGTH
        THE DECREASE IN 10 SECOND GEL-STRENGTH

TYPE ANYTHING TO CONTINUE [ C ] :

                 EXPLANATION MENU
        DIAGNOSES                                        D
        SCOPE                                            S
        REASONS                                          R
        DATA UNACCOUNTED FOR                             X
        EXIT                                             E
 [ M ] :s

By Hypothesis or by Property [ H ] :

IF THERE IS AN INFLUX OF HYDROCARBONS,
THIS WOULD EXPLAIN:
        THE INCREASE IN ELECTRICAL STABILITY
        THE INCREASE IN OIL WATER RATIO
        THE DECREASE IN YIELD POINT
        THE DECREASE IN PLASTIC VISCOSITY
        THE DECREASE IN LOW SPECIFIC GRAVITY SOLIDS
        THE DECREASE IN DENSITY
        THE DECREASE IN 10 MINUTE GEL-STRENGTH
        THE DECREASE IN 10 SECOND GEL-STRENGTH

IF THERE IS AN INFLUX OF FORMATION-HYDROCARBONS,
THIS WOULD EXPLAIN:
        THE DECREASE IN 10 SECOND GEL-STRENGTH
        THE DECREASE IN 10 MINUTE GEL-STRENGTH
        THE DECREASE IN DENSITY
        THE DECREASE IN LOW SPECIFIC GRAVITY SOLIDS
        THE DECREASE IN PLASTIC VISCOSITY
        THE DECREASE IN YIELD POINT
        THE INCREASE IN OIL WATER RATIO
        THE INCREASE IN ELECTRICAL STABILITY

IF THERE IS AN UNDER-DOSAGE OF GELTONE,
THIS WOULD EXPLAIN:
        THE DECREASE IN 10 SECOND GEL-STRENGTH
        THE DECREASE IN 10 MINUTE GEL-STRENGTH
        THE DECREASE IN DENSITY
        THE INCREASE IN HTHP CAKE THICKNESS


                 EXPLANATION MENU
        DIAGNOSES                                        D
        SCOPE                                            S
        REASONS                                          R
        DATA UNACCOUNTED FOR                             X
        EXIT                                             E
 [ M ] :e

                            MODES
        To enter data                                    DATA
        Analyze data and form hypotheses                 ANALYSIS
        How to remedy problems                           TREATMENT
        To set options                                   OPTIONS
        To display this menu                             MENU
        To explain an mode                               HELP [mode]
        To exit from the program                         EXIT

 MODE [ MENU ] :t

 What is the current volume of the mud system [ BBL ] :2000
 WHAT WAS THE EXPECTED VOLUME OF THE MUD SYSTEM:1890
```

```
H 1 :   THERE IS AN INFLUX OF HYDROCARBONS

H 2 :   THERE IS AN INFLUX OF FORMATION-HYDROCARBONS

H 3 :   THERE IS AN UNDER-DOSAGE OF GELTONE

H 4 :   THERE IS AN UNDER-DOSAGE OF LIME

WHICH HYPOTHESIS DO YOU WISH TO TREAT:1

IS IT STILL TRUE THAT
THERE IS AN INFLUX OF HYDROCARBONS, THAT IS,
IS THIS CONDITION STILL PERSISTING [ Y N U ] [ Y ] :

PROBLEM:   THERE IS AN INFLUX OF HYDROCARBONS

EFFECTS:  INCREASE IN ELECTRICAL STABILITY
          INCREASE IN OIL WATER RATIO
          DECREASE IN YIELD POINT
          DECREASE IN PLASTIC VISCOSITY
          DECREASE IN LOW SPECIFIC GRAVITY SOLIDS
          DECREASE IN DENSITY
          DECREASE IN 10 MINUTE GEL-STRENGTH
          DECREASE IN 10 SECOND GEL-STRENGTH

TREATMENTS:  ADD WEIGHT-MATERIAL
                              TO INCREASE DENSITY
             ADD EMULSIFIER   TO INCREASE ELECTRICAL STABILITY
             ADD WATER        TO DECREASE OIL WATER RATIO

TYPE ANYTHING TO CONTINUE [ C ] :

MATERIAL SPECIFICATIONS

PRODUCT                 RECOMMENDED      CURRENT           COST OF
                         TREATMENT      INVENTORY         MATERIALS

  EZ-MUL                   2 DRUMS     1800 DRUMS        $     1617
  INVERMUL                15 DRUMS     1800 DRUMS        $    12448
 *BARGAIN                670 SX        2800 SX           $     3786
  WATER                   53 GALS      2800 GALS         $       13
  GELTONE                 20 SX         800 SX           $     1453

TOTAL COST OF THE ABOVE MATERIALS IS:                    $    19317

PILOT: NO             RESULTING VOLUME = 2058 bbl.

                TREATMENT ELABORATION MENU
        ALTERNATIVES < PRODUCT >                         A
        EXPLAIN < PRODUCT >                              EX
        EXIT                                             E

ELABORATION: [ M ] :ex ez-mul

AT 324 DEGREES A DOSAGE OF 2.9600 PPB IS REQUIRED.
2 DRUMS IS SLIGHTLY MORE THAN 458.8000 LBS WHICH RESULTS AS FOLLOWS:
 458.8000 LBS SHOULD BE ADDED TO COVER AN ADDITIONAL SYSTEM VOLUME OF 156 BBL.

                TREATMENT ELABORATION MENU
        ALTERNATIVES < PRODUCT >                         A
        EXPLAIN < PRODUCT >                              EX
        EXIT                                             E

ELABORATION: [ M ] :ex water

52.9941 GALS OF WATER WERE ADDED TO BRING THE OIL-WATER RATIO
FROM 80 / 20 TO 77 / 23
```

```
                 TREATMENT ELABORATION MENU
        ALTERNATIVES < PRODUCT >                             A
        EXPLAIN < PRODUCT >                                  EX
        EXIT                                                 E

ELABORATION: [ M ] :a bargain

BAROID IS AN ALTERNATIVE TO BARGAIN :

BAROID                    703 SX          2800 SX            $    5547

DO YOU WANT TO KNOW WHY IT WAS NOT SELECTED [ Y ] :

BARGAIN WAS PREFERRED BECAUSE:

        IT HAS LESS DAMAGING SIDE-EFFECTS
        IT IS CHEAPER

TYPE ANYTHING TO CONTINUE [ C ] :


                 TREATMENT ELABORATION MENU
        ALTERNATIVES < PRODUCT >                             A
        EXPLAIN < PRODUCT >                                  EX
        EXIT                                                 E

ELABORATION: [ M ] :e

H 1 :   THERE IS AN INFLUX OF HYDROCARBONS

H 2 :   THERE IS AN INFLUX OF FORMATION-HYDROCARBONS

H 3 :   THERE IS AN UNDER-DOSAGE OF GELTONE

H 4 :   THERE IS AN UNDER-DOSAGE OF LIME

WHICH HYPOTHESIS DO YOU WISH TO TREAT:
↑C

[CTRL-C: RETURN TO TOP]
```