23

A Representation Scheme Using Both Frames and Rules

Janice S. Aikins

Much of artificial intelligence research has focused on determining the appropriate knowledge representations to use in order to achieve high performance from knowledge-based systems. The principal hypothesis being explored in this chapter is that there are many advantages to a system that uses both framelike structures and rules to solve problems in knowledge-intensive domains. These advantages can be grouped into two broad categories: those dealing with the knowledge base representation itself, and those dealing with the system's reasoning and performance. In order to test this hypothesis, a knowledge representation was designed that uses a combination of frames and rules in a data structure called a *prototype*. The domain chosen was that of pulmonary physiology. The task was to interpret a set of pulmonary function test results, producing a set of interpretation statements and a diagnosis of pulmonary disease in the patient.[1] Initially, a MYCIN-like production rule system called PUFF (Kunz et al., 1978) was written to perform pulmonary function test interpretations. Problems with the production rule formalism in PUFF and similar rule-based systems motivated the creation of a prototype-directed system, called CENTAUR. See Aikins (1980; 1983) for more detailed discussions of this system.

CENTAUR uses prototypes that characterize the typical features of each pulmonary disease. Each feature is called a component of the pro-

---

This chapter is based on a technical memo (HPP-79-10) from the Heuristic Programming Project, Department of Computer Science, Stanford University. Used with permission.

[1]It should be noted, however, that the methodology used is not domain-specific; the task that was chosen is not important for the comparisons made between various knowledge representation schemes.

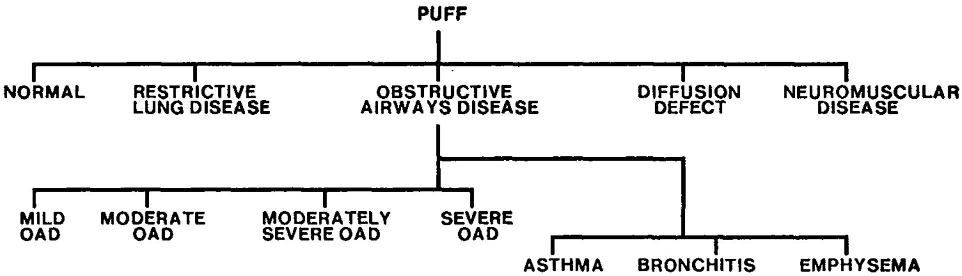

FIGURE 23-1 A portion of the prototype network.

totype. Associated with each component are rules used to deduce a value for the component. The prototypes focus the search for new information by guiding the invocation of the rules and eliciting the most relevant information from the user. These prototypes are linked together in a network in which the links specify the relationships between the prototypes. For example, the obstructive airways disease prototype is linked to the asthma prototype with a SUBTYPE link, because asthma is a subtype of obstructive airways disease (see Figure 23-1).

This chapter discusses the problems of a purely rule-based system and the advantages afforded by using a combination of rules and frames in the prototype-directed system. A complementary piece of research (Aikins, 1979), not discussed here, deals with the problems of a frame-based system. Previous research efforts have discussed systems using frames [see, for example, Minsky (1975) and Pauker and Szolovits (1977)] and systems using a pure rule-based approach to representation (Chapter 2). Still other systems have used alternate knowledge representations to perform large knowledge-based problem-solving tasks. For example, INTERNIST (Pople, 1977) represents its knowledge using a framelike association of diseases with manifestations. Each manifestation, in turn, is associated with the list of diseases in which the manifestation is known to occur. In PROSPECTOR (Duda et al., 1978a), the framelike data structures have been replaced by a semantic network. Few researchers, however, have used both frames and production rules or have attempted to draw comparisons between these knowledge representation methodologies. CENTAUR offers an appropriate mechanism with which to experiment with these representation issues.

This paper presents an example of the CENTAUR system performing an interpretation of a set of pulmonary function test results and focuses on CENTAUR's knowledge representation and control structure. In addition, some advantages of the prototype-directed system over the rule-based approach for this problem are suggested.

23.1 The CENTAUR System

CENTAUR is a consultation system that produces an interpretation of data and a diagnosis based on a set of test results. The inputs to the system are the pulmonary function test results and a set of patient data including the patient's name, sex, age, and a referral diagnosis. The output consists of both a set of interpretation statements that serve to explain or comment on the pulmonary function test results and a final diagnosis of pulmonary disease in the patient.

CENTAUR uses a hypothesis-directed approach to problem solving where the hypotheses are represented by the prototypes. The goal of the system is to confirm that one or more of the prototypes in the prototype network match the data in an actual case. The final set of confirmed prototypes is the system's solution for classifying the data in that case. The prototypes represent the various pulmonary diseases, their severity, and their subtypes, with the result that the set of confirmed prototypes represents the diagnosis of pulmonary disease in the patient.

The system begins by accepting the test and patient data. Data entered in the system suggest or "trigger" one or more of the prototypes. The triggered prototypes are placed on a hypothesis list and are ordered according to how closely they match the data. The prototype that matches the data most closely is selected to be the current prototype, the system's current best hypothesis about how to classify the data in the case.

In the example in Figure 23-2, the prototype that represents a pulmonary function consultation (PUFF) has been selected as the initial current prototype.[2] Initial data are requested and the user's responses (in boldface and following the asterisks) are recorded. The system attempts to fill in values for the components of a prototype, which may cause rules to be invoked, or, if no rules are associated with a component, the system will ask the user for the value. When all of the prototype components have values, the system decides whether the given data values are sufficiently close to those expected for the prototype to confirm that the prototype matches the data.[3] Another prototype is then selected as the current pro-

[2]Just as the pulmonary disease prototypes represent typical ranges of values for the pulmonary function tests for patients with that disease, the pulmonary function prototype states some of the typical features of a pulmonary function consultation. For example, for any pulmonary function consultation, an initial set of test and patient data is required, and both a final interpretation and pulmonary diagnosis are generated. Similarly, the prototype network of the CENTAUR system includes a prototype called MYCIN, which states typical features of a MYCIN infectious disease consultation. Above both of these prototypes is a third prototype, CONSULTATION, which states some domain-independent features of any consultation. For example, the CONSULTATION prototype contains a component called STRATEGY, which allows the user to specify whether a confirmation strategy (to confirm the most likely hypothesis) or an elimination strategy (to disprove the least likely hypothesis) is desired.

[3]The system maintains a confirmed list of prototypes that have been shown to match the data in the case and a disproved list of prototypes that have been proved not to match the data.

totype, and the process repeats. The system moves through the prototype network confirming or disproving disease prototypes. The attempt to match data and prototypes continues until each datum has been explained by some confirmed prototype or until the system has concluded that it cannot account for any more of the data. A portion of the prototype network for the pulmonary function application is given in Figure 23-1. Details of the knowledge representation and control structure for the CENTAUR system are given in Section 23.2 and Section 23.3.

Figure 23-2 is an example of an interpretation of a set of pulmonary function test results for one patient. Comments are in italics. Many additional lines of trace are printed to show what CENTAUR is doing between questions.

```
*CENTAUR
14-Jan-79 13:54:07

        CURRENT PROTOTYPE: PUFF

        The current hypothesis is that an interpretation of the pulmonary function tests is desired.

        [Control slot of PUFF prototype being executed ...]

        --------PATIENT-7446--------
                                                        (The initial data given by the user.)
1) Patient's identifying number:
** 9007
2) referral diagnosis:
** ASTHMA
        [Trigger for ASTHMA and CM 900]
      Prototype ASTHMA is triggered by the value ASTHMA for the referral diagnosis. The certainty measure (CM)
      indicates on a numerical scale the degree of certainty with which the prototype is indicated by the data.)

3) RV/RV-predicted:
** 261
4) TLC(body box)observed/predicted:
** 139
5) FVC/FVC-predicted:
** 81
        [Trigger for Normal and CM 500]
                      (The questioning continues and other prototypes are triggered by the data values.)
```

FIGURE 23-2 An example of the interpretation of a set of patient test results. Debugging information (not routinely provided) is shown to help demonstrate the process of prototype invocation. (Key: ALS = amyotrophic lateral sclerosis; FEV1 = forced expiratory volume at one minute; FVC = forced vital capacity; MMF = maximal midexpiratory flow; OAD = obstructive airways disease; RDX = referral diagnosis; RLD = restrictive lung disease; RV = residual volume; TLC = total lung capacity.)

6) FEV1/FVC ratio:
** **40**
7) the DLCO/DLCO-predicted:
** **117**
[Trigger for NORMAL and CM 700]
8) Change in FEV1 post-dilation - pre-dilation:
** **31**
9) MMF/MMF-predicted:
** **12**
[Trigger for OAD and CM 900]
10) The slope (F50-obs-F25-obs)/FVC-obs:
** **9**
[Trigger for OAD and CM 900]

TRIGGERED PROTOTYPES
PROTOTYPE: ASTHMA, CM: 900, REASON: RDX was ASTHMA
PROTOTYPE: NORMAL, CM: 500, REASON: FVC was 81
PROTOTYPE: NORMAL, CM: 700, REASON: DLCO was 117
PROTOTYPE: OAD, CM: 900, REASON: MMF was 12
PROTOTYPE: OAD, CM: 900, REASON: F5025 was 9
(A list of the prototypes that have been triggered is given. The CM and the value that caused the trigger are also listed.)

Confirmed List: PUFF

It is confirmed that an interpretation of the pulmonary function tests is desired.

(The primary purpose of the pulmonary function prototype is to acquire the initial data from the user. It has no components and is thus confirmed when it has completed its task.)

MoreSpecific Prototypes chosen: NORMAL OAD
(Although there are five possible, more specific disease prototypes for PUFF, only the two that were triggered by the initial data are selected as possibilities to pursue.)

[New prototypes being filled in ... NORMAL OAD]
(These prototypes are filled in with the data values that are already known in the case.)

! Surprise Value ! 261 for RV in NORMAL, CM: 700
! Surprise Value ! 139 for TLC in NORMAL, CM: 400
! Surprise Value ! 40 for FEV1/FVC in NORMAL, CM: -166
! Surprise Value ! 12 for MMF in NORMAL, CM: -499
! Surprise Value ! 9 for F5025 in NORMAL, CM: -699

(Any data values that are not consistent with the values expected for that disease prototype are noted as surprise values, and the CM for that prototype is lowered. In this case, five of the data values are not consistent with the NORMAL pulmonary function prototype.)

Hypothesis List: (OAD 990) (NORMAL -699)
(The hypothesis list of triggered prototypes is then ordered according to the CM of the prototypes and a new current prototype is chosen.)

CURRENT PROTOTYPE: OAD

The current hypothesis is that there is an interpretation of Obstructive Airways Disease.

Components of OAD chosen to trace: F25 D-RV/TLC

FIGURE 23-2 continued

(In order to instantiate the OAD prototype, two more components must have values. These are asked of the user if there are no rules associated with the components that can be used to deduce their values.)

11) The flow F25:
** **UNKNOWN**
12) RV/TLC Observed-Predicted:
** **25**

Confirmed List: OAD PUFF

It is confirmed that there is an interpretation of Obstructive Airways Disease.

(The OAD prototype is confirmed. Control information associated with the prototype specifies that the degree of OAD should be determined next, followed by the subtype of OAD.)

MoreSpecific Prototypes chosen: MILD-OAD MODERATE-OAD
MODERATELY-SEVERE-OAD SEVERE-OAD

(No degree prototypes were triggered by the data values, so all of them are selected as possible hypotheses to be filled in along with the data values in the case.)

[New prototypes being filled in ... MILD-OAD MODERATE-OAD
MODERATELY-SEVERE-OAD SEVERE-OAD]
...

(More surprise values are noted, and the hypothesis list is ordered, leading to a current prototype of SEVERE-OAD.)

CURRENT PROTOTYPE: SEVERE-OAD

The current hypothesis is that there is an interpretation of Severe Obstructive Airways Disease.

Components of SEVERE-OAD chosen to trace: FEV1

13) FEV1:
** **42**

(The consultation continues with new components being asked and classified.)

Confirmed List: SEVERE-OAD OAD PUFF

It is confirmed that there is an interpretation of Severe
Obstructive Airways Disease.

MoreSpecific Prototypes chosen: ASTHMA

Hypothesis List: (ASTHMA 900)

(Next the subtypes of OAD are explored. ASTHMA is chosen from three possible subtypes because it was triggered by the initial data.)

CURRENT PROTOTYPE: ASTHMA

The current hypothesis is that there is an interpretation of Asthma.

Components of ASTHMA chosen to trace: DEG-REV

14) The change in resistance pre-dilation - post-dilation:
** **20**

Confirmed List: ASTHMA SEVERE-OAD OAD PUFF

It is confirmed that there is an interpretation of Asthma.

[Facts marked Accounted For by ASTHMA, SEVERE-OAD, OAD]

(There are no further degrees or subtypes to be explored for this case. Thus data values that can be accounted for by one of the confirmed prototypes are marked. If there are data values remaining that cannot be accounted for by the confirmed prototypes, the system will attempt to determine if there are multiple diseases in the patient.)

[Action slot of OAD prototype being executed ...]

(At this point the system begins executing any actions associated with the confirmed prototypes. There are no actions for ASTHMA or SEVERE-OAD, so the OAD action is the first to be executed.)

OAD Action Clause
Display the findings about the diagnosis of obstructive airways
  disease
Conclusions: the findings about the diagnosis of obstructive
  airways disease are as follows:
  Elevated lung volumes indicate overinflation.
  The RV/TLC ratio is increased, suggesting a SEVERE degree of air trapping.
  Forced Vital Capacity is normal but the FEV1/FVC ratio is
    reduced, suggesting airway obstruction of a SEVERE degree.
  Low mid-expiratory flow is consistent with severe airway
    obstruction.
  Obstruction is indicated by curvature of the flow-volume loop
    which is of SEVERE degree.
  Reversibility of airway obstruction is confirmed by improvement in airway resistance following
    bronchodilation.

[Action slot of PUFF prototype being executed ...]

PUFF Action Clause
Display the conclusion statements about this interpretation
Conclusions: the conclusion statements about this interpretation
  are as follows:
    Smoking probably exacerbates the severity of the patient's
      airway obstruction.
    Discontinuation of smoking should help relieve the symptoms.
    Good response to bronchodilators is consistent with an
      asthmatic condition, and their continued use is indicated.

PUFF Action Clause
Display the summary statements about this interpretation
Conclusions: the summary statements about this interpretation are as
follows:

SEVERE Obstructive Airways Disease, Subtype ASTHMA

Do you wish advice on another patient?
** **NO**

FIGURE 23-2 continued

23.2 Knowledge Representation in CENTAUR

Knowledge is represented in CENTAUR by both rules and prototypes. Each prototype contains two kinds of information: domain-specific *components* that express the substantive characteristics of each prototype, and domain-independent *slots* that specify information used in running the system. Each component may, in turn, have slots of information associated with it, including a RULES slot that links the component to rules that determine values of the component. Thus the outline of a prototype can be viewed as shown in Figure 23-3.

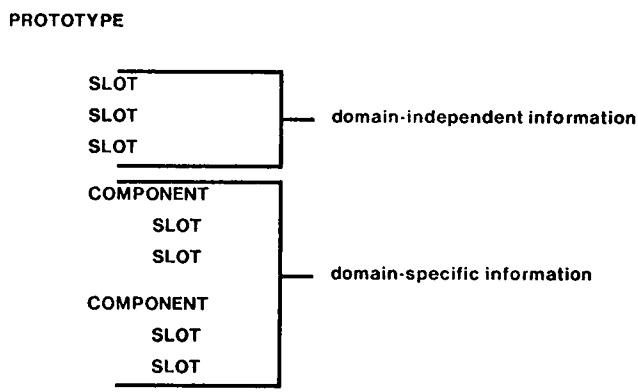

FIGURE 23-3 Prototype outline.

The rules consist of one or more premise clauses followed by one or more action clauses. An example is given in Figure 23-4.[4] In general, the premise clauses specify a set of value ranges for some of a prototype's components, and the action clauses make conclusions about the values of other components. Besides these static data structures, there are also data structures that give information about the actual data values obtained during the consultation. These are called *facts* and are discussed in Section 23.2.3.

23.2.1 Prototypes and Components

Most of CENTAUR's prototypes represent the characteristic features of some pulmonary disease. For example, there is a prototype for obstructive airways disease (OAD), a portion of which is shown in Figure 23-5. In the

[4]As in MYCIN, the rule is stored internally in the Interlisp form shown; the English translation is generated from that.

```
RULE013
------
  PREMISE:     ($AND ($OR ($AND (LESSP* (VAL1 CNTXT MMF)
                                         20)
                                 (GREATERP* (VAL1 CNTXT FVC)
                          ($AND (LESSP* (VAL1 CNTXT MMF)
                                         15)
                                 (LESSP* (VAL1 CNTXT FVC)
                                         80))]
  ACTION:      (DO-ALL (CONCLUDE CNTXT DEG<-MMF SEVERE TALLY 900)
                       (CONCLUDETEXT CNTXT FINDINGS<-OAD (TEXT $MMF)
                                       TALLY 1000))
```

RULE013

[This rule applies to any patient, and is tried in order to find out about the degree of obstructive airways disease as indicated by the MMF or the findings about the diagnosis of obstructive airways disease.]

If: 1) A: The MMF/MMF-predicted ratio is less than 20, and
B: The FVC/FVC-predicted ratio is greater than 80, or
2) A: The MMF/MMF-predicted ratio is less than 15, and
B: The FVC/FVC-predicted ratio is less than 80

Then: 1) *There is strongly suggestive evidence (.9) that the degree of obstructive airways disease as indicated by the MMF is severe, and*
2) It is definite (1.0) that the following is one of the findings about the diagnosis of obstructive airways disease: Low midexpiratory flow is consistent with severe airway obstruction.

FIGURE 23-4 A sample rule in CENTAUR in both Interlisp and English versions.

OAD prototype, there are components for many of the pulmonary function tests that are useful in characterizing a patient with OAD; two of these are shown in the figure. For example, the total lung capacity of a patient with OAD is typically higher than that of a person with normal pulmonary function. Thus there is a component, TOTAL LUNG CAPACITY, with a range of *plausible values* that are characteristic of a person with OAD.

In addition to a set of plausible values, that is, values consistent with the hypothesis represented by the prototype, the components may have additional information associated with them. (The ways in which this information is used are discussed in Section 23.3.) There may be one or more *possible error values*, that is, values that are inconsistent with the prototype or that might have been specified by the expert to check what he or she considers to be a measurement error. Generally, both a reason for the error and a possible fix for the error are specified. For example, the expert may specify that one of the pulmonary function tests be repeated to ensure accuracy. A component may also have a *default value*. Thus all of the components in a disease prototype, with their default values, form a picture of the typical patient with the disease. Finally, each component has an *importance measure* (from 0 to 5) that indicates the relative importance of a particular component in characterizing the disease.

In addition to the domain-specific components, each prototype con-

PROTOTYPE GENERAL INFORMATION	Obstructive Airways Disease (OAD)
--Bookkeeping Information	Author: Aikins Date: 27-OCT-78 Source: Dr. Fallat
--Pointers to other prototypes (link prototype)	Pointers: (degree MILD-OAD) (degree MODERATE-OAD) ... (subtype ASTHMA) ...
--English phrases	Hypothesis: "There is an interpretation of OAD."
COMPONENTS Plausible Values Default Value Possible Error Values Rules Importance of value to this prototype . . .	TOTAL LUNG CAPACITY Plausible Values: >100 Importance: 4 REVERSIBILITY Rules: 19,21,22,25 Importance: 0 (value not considered)
CONTROL INFORMATION	Deduce the degree of OAD Deduce the subtype of OAD Deduce any findings associated with OAD
ACTION INFORMATION	Print the findings associated with OAD

FIGURE 23-5 A sample prototype showing possible slots on the left and values of those slots for OAD on the right.

tains slots for general information associated with it. This includes bookkeeping information (name of the prototype, its author, date on which the prototype was created, and source for the information contained there) and English phrases used in communicating with the user. There are also pointers to other prototypes in the prototype network, which are useful, for example, when either more general disease categories or more specific subtypes of disease are indicated. Some control information is represented explicitly in slots associated with the prototype (Section 23.3). This information includes what to do in order to confirm the prototype and what to do when the prototype has been confirmed or disproved. Each prototype also has associated with it a *certainty measure* (from −1000 to 1000) that indicates how certain the system is that the prototype matches the data in each case.

23.2.2 Rules

The CENTAUR knowledge base also includes rules, which are grouped into four sets according to their functions. They refer to values for components in their premise clauses and make conclusions about values of components in their action clauses. An example of one of the rules is given in Figure 23-4. The RULES slot associated with a component contains a list of all rules that make a conclusion about that component. These may be applied when a value is needed for the component.[5]

Many of the rules are classified as *patient rules,* rules dealing with the patient. Besides the patient rules, there are three other sets of rules. Those rules whose actions make summary statements about the results of the pulmonary function tests are classified as *summary rules;* rules that refer to values of components in their premises and suggest general disease categories in their actions are classified as *triggering rules.* These are used to "trigger" or suggest the disease prototypes. Those rules that are used in a second stage of processing, after the system has formulated lists of confirmed and disproved prototypes are called *refinement rules;* they are used to refine a preliminary diagnosis, producing a final diagnosis about pulmonary disease in the patient. The refinement rules constitute a further set of domain expertise; they test the system's tentative conclusions, which may result in a modification of these conclusions. For example, if two diseases can account for a given pulmonary function test result and both have been confirmed in that case, a refinement rule may determine which disease process should account for the test result in the final interpretation.

23.2.3 Facts

In CENTAUR, each piece of case-specific data that has been acquired either initially from the patient's pulmonary function test results or later during the interpretation process is called a *fact.* Each fact has six fields of information associated with it. When a fact is first introduced into the system, its name, value, and certainty factor[6] fields are instantiated. For example, if the user specifies that the total lung capacity of the patient is 126 with a certainty factor of 0.8, then a fact is created:

```
NAME: Total Lung Capacity
VALUE: 126
CERTAINTY FACTOR: .8
```

The fourth field associated with the fact indicates where it was obtained: from the user (this includes the initial pulmonary function test results), from the rules, or as a default value associated with a prototype component. Thus, in the fact about total lung capacity, the fourth field would have the value USER.

The fifth field of each fact becomes instantiated once fact values are classified as being plausible values, possible error values, or surprise values

[5]If no rules are associated with the component, the user will be asked for the value. If the user responds UNKNOWN and the component has a default value, that value will be used.

[6]The certainty factor is just MYCIN's CF—a number ranging from −1 to 1 that indicates the importance of the given value.

for a given prototype. Surprise values are all of those values that are neither plausible values nor possible error values. They indicate facts that cannot be accounted for by the hypothesis represented by the prototype. In the fact about total lung capacity, the fifth field might contain the classification (PV OAD) and (SV NORMAL) meaning that the value of 126 for the total lung capacity of a patient would be a plausible value if the patient had obstructive airways disease, but would be a surprise value if the patient were considered to have normal pulmonary function.

The last field associated with a fact indicates which confirmed prototypes can account for the given value. When a prototype is confirmed, all of the facts that correspond to components in the prototype and whose values are plausible values for the component are said to be "accounted for" by that prototype. When the OAD prototype is confirmed, for a patient with total lung capacity of 126, for example, the last field of the sample fact for total lung capacity would be filled in with the prototype name OAD.

23.3 Control Structure for CENTAUR

The control information used by CENTAUR is contained either in slots that are associated with the individual prototypes or in a simple interpreter. Some control strategies are specific to an individual prototype and need to be associated with it, while more general system control information is more efficiently expressed in the interpreter.

Basically, the interpreter attempts to match one or more of the prototypes with the data in an actual case. At any one time there is one current prototype that the system is attempting to match to the facts of the case. Attempting a match for this prototype entails finding values for the prototype components, i.e., instantiating the prototype. The exact method to be used in instantiating the prototype depends on the individual prototype and is expressed in one of the prototype control slots.

When all of the facts have been accounted for by some confirmed prototype, or when no prototype can account for a known fact,[7] the system has completed the hypothesis-formation stage. The confirmed list of prototypes then represents the system's hypothesis about how to classify the facts. At this point, additional knowledge may be applied before generating the final pulmonary function interpretation and diagnosis. Some of this knowledge is represented in the refinement rules associated with the confirmed prototypes. Further information may be sought from the user at

[7]This statement oversimplifies the actual matching criteria used by the system. Some tolerance for a mismatch between known fact values and plausible values in the prototype is allowed.

this stage. For example, further lab tests may be suggested or additional test results may be required before a final diagnosis is given.

The result of executing the refinement rules is a final set of confirmed prototypes and a list of all facts with an indication of which prototypes account for which facts. The system then executes the clauses specified in the action slot of each confirmed prototype. Typically, these clauses express a clean-up chore such as executing summary rules associated with the prototype[8] or printing interpretation statements. The action slot of the PUFF prototype itself causes the final interpretation and pulmonary diagnosis to be printed.

23.3.1 Prototype Control Slots

Four of the slots associated with a prototype contain clauses that are executed by the system at specific times to control the consultation. Each clause expresses some action to be taken by the system at different stages: (a) in order to instantiate the prototype (CONTROL slot), (b) upon confirmation of the prototype (IF-CONFIRMED slot), (c) in the event that a prototype is disproved (IF-DISPROVED slot), and (d) in a clean-up phase after the system processing has been completed (ACTION slot).

When a prototype is first selected as the current prototype, the system executes the clauses in the CONTROL slot of that prototype. The information in this slot indicates how to proceed in order to instantiate the prototype, usually specifying what data should be acquired and in what order they should be acquired. Therefore, executing these clauses will cause values to be obtained for the prototype components. The CONTROL slot can be thought of as a rule whose implicit premise is "if this prototype is selected as the current prototype" and whose action is the given set of clauses. If no CONTROL slot is associated with a prototype, the interpreter will attempt to fill in values for the prototype components in order according to their importance measures.

When all of the clauses in the CONTROL slot have been executed and the prototype has been instantiated, a decision is made as to whether the prototype should be confirmed as matching the facts of the case.[9] The system then checks either the IF-CONFIRMED slot or the IF-DISPROVED slot to determine what should be done next. These slots can be viewed as rules whose implicit premise is either "if this prototype is confirmed as matching the data" or "if this prototype is proved not to match the data." The appropriate actions are then indicated in the set of clauses contained in the slot.

---

[8]Recall that the premise of a summary rule typically checks the values for one or more parameters and that the action generates an appropriate summarizing statement.

[9]It would be possible to associate such a confirmation criterion with each individual prototype, but this has not been found to be necessary for the pulmonary diagnosis problem. Instead, the system uses a general algorithm, applicable to all of the prototypes, that checks the values of the components and their importance measures to determine if the prototype should be marked as confirmed.

The fourth slot specifying clauses to be executed is the ACTION slot. The implicit premise in this slot is "if the system has completed its selection of confirmed prototypes and this prototype is confirmed." Thus the clauses in the ACTION slot are the last ones to generate summary statements or print data interpretations.

23.4 Advantages of the Prototype-Directed Approach

One question addressed by this research is this: in what ways are both frames and rules superior to either alone? Comparisons can be drawn between purely rule-based systems, such as PUFF, at one end of the spectrum and purely frame-based systems at the other. This section states some of the advantages of the prototype-directed approach used in CENTAUR for the pulmonary function interpretation task, as compared to the purely rule-based approach used in PUFF. The next chapter discusses a purely frame-based approach to the same problem. These advantages can be grouped into two broad categories: those dealing with knowledge base representation, and those dealing with reasoning and performance.

23.4.1 Knowledge Representation

Specific advantages of using prototypes in the pulmonary function domain include the following:

A. *Rules attached to prototypes are used to represent only medical expertise, not computational information.* In the PUFF system, there are rules that guide computation by controlling the invocation of other rules. This feature can be very confusing to the medical experts since they do not know which rules are intended to represent medical expertise and which rules serve a necessary computational function. For example, a PUFF rule necessary to determine whether there is obstructive airways disease (OAD) in the patient is

> If an attempt has been made to deduce the degree of OAD, and an attempt has been made to deduce the subtype of OAD, and an attempt has been made to deduce the findings about OAD, then there is an interpretation of potential OAD.

This rule expresses some of the control structure of the system, namely, that when there is an interpretation of OAD, then the degree, subtype, and findings associated with the OAD should be determined. The rule is confusing because it implies that finding out the degree, subtype, and findings leads to an interpretation of OAD—which might be misinterpreted as medical expertise. In fact, this rule is executed for every case and causes all of the other OAD rules to be invoked, even when no OAD is present.

In CENTAUR, rules that guide computation have been removed from the rule base, leaving a less confusing, more uniform rule base, where each rule represents some "chunk" of medical expertise. Computation is now guided by the prototypes. For example, the CONTROL slot represents information dealing with how to instantiate the prototype. For the OAD prototype, this CONTROL slot specifies that deducing the degree, subtype, and findings of obstructive airways disease are the steps to take in instantiating that prototype.

B. *Prototypes represent more clearly some of the medical expertise formerly contained in rules.* In some cases, medical expertise that has been represented in the production rules is more clearly represented in the prototype. Consider, for example, the following PUFF rule:

> If the degree for OAD is NONE, and the degree for OAD by the MMF is greater than or equal to MILD, then the degree for the OAD is MILD.

The medical expertise expressed in this rule is not apparent. In order to understand this rule, it is necessary to see it as one part of a group of several other rules, all of which together help to determine the degree of obstructive airways disease in the patient. The first clause of the rule, "If the degree for OAD is NONE," is partly a description of the medical context, indicating that the degree of OAD has not been established. However, it is also control information in that it requires that the degree for OAD be determined, which, in turn, invokes the other rules. Yet part of the motivation for using rules is that each rule should be a single "chunk" of knowledge, understandable in its own right. Further, what is really being said in this rule is that in determining the degree of OAD in the patient, there are several pulmonary function measurements to be considered, but, of these, the MMF measurement should be given somewhat more weight. In CENTAUR, this fact is represented explicitly in the OAD prototype by giving the MMF component an importance measure higher than those of the other measurement components.

C. *Knowledge is represented explicitly by prototypes.* As was indicated in paragraphs A and B above, making knowledge explicit is one of the advantages of the prototype representation. Not only is knowledge about how to instantiate the prototype represented explicitly, but knowledge about what to do if the prototype is confirmed or disproved, as well as what are appropriate clean-up actions to perform for the prototype, e.g., printing findings or summarizing data, is also represented. Other information, such as the importance measure to assign to one of the prototype components when matching prototypes to data, is also made explicit. All of this specifies to those working with the knowledge base precisely what information is represented and what role that information plays in the computation.

D. *Additional knowledge is represented by prototypes.* By adding a set of disease prototypes, some new knowledge about pulmonary disease can be represented. In MYCIN additional knowledge can be added as properties of *rules,* but it is difficult to add new knowledge about diseases. For example, plausible ranges of values for each of the pulmonary function tests for each disease, as well as the relative importance of each measurement in a particular disease prototype, can be listed.

23.4.2 Reasoning and Performance of the System

A second category of advantages deals with the way the system reasons about the problem. This is evident in part by watching the performance of the system, that is, the questions that are asked and the order in which information is acquired. Some of the advantages of a prototype-directed system are the following:

E. *Consultation flow follows the physician's reasoning.* The consultation begins with specific test results suggesting or "triggering" some of the prototypes. The prototypes serve as tentative hypotheses about how to classify the data in a given case. They also guide further inquiry. As new information is acquired, these hypotheses are revised, or, in CENTAUR's terms, prototypes are confirmed or disproved and new prototypes may then be suggested. The process of medical problem solving has been discussed by many researchers [e.g., Elstein et al. (1978)], and it is widely felt that this sequence of suggesting hypotheses, acquiring further information, and then revising the hypotheses is, in fact, the problem-solving process used by most physicians. Thus there is increased conceptual clarity, in that the user can understand what the program is doing. Other advantages that accrue from this approach include: (a) the knowledge base is easier to modify and extend, and (b) the system can offer the user a more intelligible explanation of its performance during the consultation. Giving the system the ability to explain its knowledge and performance has been a primary design goal of the present research efforts. Since the prototype-directed system reasons in a manner more like a human user, its behavior seems more natural and transparent and thus is more likely to be accepted by physicians.

F. *The order in which questions are asked can be controlled.* In a rule-based system such as PUFF, questions are asked of the user as rules are invoked that contain clauses referring to information that is not yet known. The designers of PUFF, or any EMYCIN system, control the order in which the questions are asked only by writing rules to enforce some order. As has been discussed, this procedure results in a potentially confusing rule base where some rules represent medical expertise and others guide computation. In the prototype-directed system, the expert specifies the order in which information is to be acquired for each prototype in the CONTROL slot. Thus control information is labeled explicitly as such, and the rule base remains uniformly a body of medical expertise. The expert can also specify what information must be acquired and what information is optional, using the importance measure associated with each component.[10]

G. *Only relevant questions are asked.* Another advantage of CENTAUR over the rule-based version of PUFF is that only those hypotheses suggested by the initial data are explored. For example, if the total lung capacity (TLC) for the patient is 70, then CENTAUR would begin exploring the possibility of restrictive lung disease (RLD) because a low TLC would trigger the RLD prototype.[11] In the PUFF program, the first disease tried is always OAD, so the PUFF program would begin asking questions dealing with OAD. These questions would seem irrelevant considering the data, and, indeed, if there were no data to indicate OAD, such questions would not be asked by CENTAUR.

H. *Inconsistent information is indicated.* During a consultation, it is also possible to point out inconsistent or possibly erroneous data as they are entered, so that a technician can repeat a test immediately or at least decide if it is worth the time to continue analyzing the case. This feature is invoked when possible error values are detected for a component of a prototype, or when no prototype can be determined to account for a given value.[12]

23.5 Summary

CENTAUR was designed in response to problems that occurred while using a purely rule-based system. The CENTAUR system offers an appropriate environment in which to experiment with knowledge representation issues such as determining what knowledge is most easily represented in rules and what is most easily represented in frames. In summary, much research remains to be done on this and associated knowledge representation issues. This present research is one attempt to make explicit the art of choosing the knowledge representation in AI by drawing comparisons between various approaches and by identifying the reasons for selecting one fundamental approach over another.

---

[10]Optional information is indicated by assigning a component an importance measure of 0.

[11]A low TLC is consistent with a hypothesis of RLD; a high TLC is consistent with OAD.

[12]It is also possible that there is an overly restricted range of plausible values for a prototype component, in which case the user may extend the range to encompass the indicated value.