(4)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1 REPORT NUMBER	2 GOVT ACCESSION NO	3 RECIPIENT'S CATALOG NUMBER
AIM 1183		

LE (and Subtitle)	5 TYPE OF REPORT & PERIOD COVERED
Consensus Knowledge Acquisition	memorandum
	6 PERFORMING ORG. REPORT NUMBER

HOR(s)	8 CONTRACT OR GRANT NUMBER(s)
Andy Trice, Randall Davis	N00014-85-K-0124

FORMING ORGANIZATION NAME AND ADDRESS	10 PROGRAM ELEMENT PROJECT, TASK AREA & WORK UNIT NUMBERS
Artificial Intelligence Laboratory 545 Technology Square Cambridge, MA 02139	

TROLLING OFFICE NAME AND ADDRESS	12 REPORT DATE
Advanced Research Projects Agency 1400 Wilson Blvd. Arlington, VA 22209	December 1989
	13 NUMBER OF PAGES 25

NITORING AGENCY NAME & ADDRESS(if different from Controlling Office)	15 SECURITY CLASS. (of this report)
Office of Naval Research Information Systems Arlington, VA 22217	UNCLASSIFIED
	15a DECLASSIFICATION/DOWNGRADING SCHEDULE

16 DISTRIBUTION STATEMENT (of this Report)

Distribution is unlimited

17 DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18 SUPPLEMENTARY NOTES

None

19 KEY WORDS (Continue on reverse side if necessary and identify by block number)

Knowledge acquisition
Multiple experts
Knowledge based systems
Debugging

20 ABSTRACT (Continue on reverse side if necessary and identify by block number)

We have developed a method and prototype program for assisting two experts in their attempts to construct a single, consensus knowledge base. We show that consensus building can be effectively facilitated by a debugging approach that identifies, explains, and resolves discrepancies in their knowledge. To implement this approach we identify and use recognition and repair procedures for a variety of discrepancies. Examples of this knowledge are illustrated with sample transcripts from

(cont.)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered

Block 20 continued:

CARTER, a system for reconciling two rule-based systems. Implications for resolving other kinds of knowledge representations are also examined.

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY

A.I. Memo No. 1183 December 1989

Consensus Knowledge Acquisition

Andrew Trice
Randall Davis

Abstract

We have developed a method and prototype program for assisting two experts in their attempts to construct a single, consensus knowledge base. We show that consensus building can be effectively facilitated by a debugging approach that identifies, explains, and resolves discrepancies in their knowledge. To implement this approach we identify and use recognition and repair procedures for a variety of discrepancies. Examples of this knowledge are illustrated with sample transcripts from CARTER, a system for reconciling two rule-based systems. Implications for resolving other kinds of knowledge representations are also examined.

This report describes research done at the Artificial Intelligence Laboratory and the School of Management of the Massachusetts Institue of Technology. Support for the Laboratory's artificial intelligence research is provided in part by the Advanced Research Projects Agency of the Department of Defence under Office of Naval Research contract N00014-85-K-0124. Additional support for this research was provided by the International Financial Services Research Center of the School of Management, as well as Digital Equipment Corporation, McDonnel Douglas Space Systems, and General Dynamics Corp.

1 INTRODUCTION

There is a curious contradiction in the current state of practice of knowledge acquisition: At a time when the view is widely shared that knowledge in organizations is distributed among multiple experts, and information systems are seen as an effective way to coordinate the activities of groups, common practice in knowledge acquisition still focuses on acquiring the knowledge of a single individual. Research in both artificial intelligence (Davis, 1982; Mittal and Dym, 1985) and information systems (Fellers, 1987; Fedorowicz and Manheim, 1986; Mumford, 1987) has identified this gap as a major barrier to the development of more powerful knowledge systems.

Until now, expert system developers have dealt with this difficulty either by refraining from building multi-expert systems entirely; by appointing one of the experts as "knowledge czar," thereby giving him the final word in any dispute; or merely by requiring experts to achieve consensus on their own, without any systematic assistance. Multi-expert acquisition techniques that have been proposed to date have tended to be either very restrictive mathematical formulations (Gaglio et al., 1985), adaptations of established group decision-making techniques (Jagannathan and Elmaghraby, 1985), or methods that focus on simply using knowledge from multiple sources rather than finding and resolving the conflicts and inconsistencies in that knowledge.

We call a process by which multiple experts attempt to construct a single consensus knowledge base "consensus knowledge acquisition" (CKA). The objective of our research is to develop ideas and tools to facilitate this activity. Specifically, we have drawn on and extended work in artificial intelligence, information systems design, and negotiation, to create a debugging system capable of aiding two (or more) experts in systematically identifying, explaining, and resolving discrepancies in their knowledge.

We begin discussion of the issues by outlining several approaches to acquiring and using multiple bodies of expertise. We then argue for an approach focused on debugging and present a set of ideas in this vein. We describe the mechanisms we have developed for detecting and reconciling knowledge base discrepancies, illustrating these procedures with sample transcripts from our prototype system. Finally, we calibrate the contribution of our work and suggest promising future directions.

2 HOW CAN WE HANDLE MULTIPLE EXPERTS?

The problem of reconciling multiple points of view has been an issue of study for some time in areas as widespread as group decision making, mathematical psychology, and management science. One interesting way to view these disparate approaches is to categorize them according to whether they are descriptive or normative, and where they focus their efforts at consensus: on outcome, process, or knowledge.

2.1 Descriptive Approaches

Descriptive approaches to this problem are fundamentally concerned with understanding how groups of decision makers actually behave when required to produce a single answer. Behavior has been studied in both field settings (e.g. Janis, 1982) and various controlled laboratory conditions (e.g. Davis, 1980; Hammond, 1975). A commonly observed phenomenon is the existence of psychological barriers to effective decision-making: factors such as conformity pressure, shyness, unequal distribution of power, and others can all affect both the process of coming to a decision and the quality of the decision that results.

2.2 Outcome Combination Methods

Work aimed at combining outcomes is illustrated by ideas like voting (Miner, 1984), averaging (Aczel and Saaty, 1983), and decomposition and re-synthesis (Brehmer and Hagafors, 1986). The objective is to arrive at a decision which, while not necessarily reflecting a consensus of the experts, is still better than any single expert could have arrived at alone.

These methods are largely normative — concentrating on how judgments ought to be combined rather than on what typically happens in groups, and are focused on outcome — it is the experts' final recommendations that are combined.

The effectiveness of these methods depends on the validity of their assumptions about both the nature of the outcome and the skill mix of the experts. Nature of the outcome matters because, for example, voting is appropriate when the scale of outcome values is nominal, while averaging is suitable when it is a ratio. Assumptions about skill mix are crucial because averaging makes no sense unless expert errors are distributed randomly, while decomposition and re-synthesis assumes that they vary systematically across subproblems (i.e., experts have different sub-specialties).

The fundamental problem with these methods is their focus on outcome rather on than the reasoning used to determine it. We believe it is premature to combine results before even attempting to achieve consensus on the underlying knowledge used to arrive at those results. Exploring that knowledge may reveal key differences in reasoning, vocabulary, or problem assumptions which, once reconciled, remove the outcome discrepancy entirely. There are also ownership issues to consider: If we combine results without allowing discussion of the underlying rationale, the experts are more likely to be unhappy with or unwilling to take responsibility for the result.

These methods may prove useful in cases where experts have discussed the rationales and still cannot reach agreement, or in situations where the knowledge bases exist but the experts responsible for them are unavailable.

2.3 Argumentation

A second approach, argumentation methods, centers on helping people make explicit the logical structure of their positions. Structured frameworks for analyzing

arguments (Toulmin, 1958; Fogelin, 1982), for instance, enable different parties in a debate to cooperate in constructing and making precise the arguments for and against a particular assertion. These ideas have recently been embodied in computer-based tools (e.g., Smolensky et al., 1987; Stefil et al., 1987; Lowe, 1985; Nunnamaker et al., 1988) that aid users in constructing and manipulating the arguments, and sometimes offer spreadsheet-like capabilities that facilitate exploring the impact of changing an assumption.

These tools are normative in their approach to consensus building and almost entirely process oriented: they assist experts in the process of deliberating and debating, but, importantly, do not suggest resolutions to inconsistencies. As such they introduce an element of rigor into the deliberation process, but offer little guidance in resolving differences between the experts.

2.4 Debugging the Knowledge

We do not want to focus on outcome alone, because we believe that the fundamental task is to reach consensus on the knowledge itself: differences in outcome may simply be symptoms of a disagreement about what to know. In that case dealing with outcome is treating the symptoms rather than the cause, while dealing with the differences in knowledge solves the root problem and may eliminate all the symptoms.

We choose not to focus on formal argumentation in the belief that the knowledge representation in use — in this case rules — provides sufficient basic structure to the discussion.

Instead we seek to assist the experts in detecting, deliberating over and reconciling discrepancies between them. Our approach is normative and focused on the underlying knowledge used by each expert: we want to understand how experts ought to come to agreement and we want that agreement to be about the thing we consider to be fundamental to this undertaking — the knowledge used to make the decisions. Debugging is a technique well suited to our goals because it centers on the detection, explanation, and repair of defects in symbolic systems. As a result we use the phrase "debugging" the knowledge to characterize both the focus of our efforts and the primary technique we employ.

3 SOME USEFUL IDEAS

Given this perspective, three research areas provide relevant concepts. Artificial intelligence (AI) offers the literature on knowledge-based systems and a body of work on debugging; information systems provides general guidelines for synthesizing multiple points of view; while work in negotiation and conflict resolution suggests the role of a third party facilitator.

From AI we exploit the notion that the knowledge representation in use can assist consensus-building by providing a structure and vocabulary for comparing arguments and the knowledge on which they are based. One familiar example is the explanation

facility provided by rule-based systems (Davis et al., 1977). These allow a user to trace the steps the program followed in reaching a particular conclusion, providing a representation of the argument a domain expert would put forward in support of his recommendation. This provides a concrete and specific focus to the discussion. The differences between two such reasoning chains can then be described using the vocabulary provided by the representation, in this case the notion of if/then rules, attribute-object-value triples, strengths of certainty, etc. This helps to establish the agenda for discussion between the two experts.

Program debugging research takes this idea a step further. Many debugging systems (Brown and Burton, 1978, Kuper, 1989) have developed bug taxonomies that specify the kinds of things that can go wrong, the probable causes underlying them, and the corresponding repairs. A key idea here is that knowledge about the program being debugged can itself be used to help guide the repair. Davis (1979), for example, used knowledge about knowledge base structure to support individual knowledge acquisition. Our research can be viewed as the extension of this work to the multiple expert case.

From information systems design we adapt methodologies used to resolve conflicting points of view (e.g., Mumford, 1987; Mason and Mitroff, 1981; Hammond et al., 1984). These methodologies advocate, first, full and active participation from all involved parties. This suggests that we should structure the CKA process so that the two experts are likely to have equal influence on design decisions. Second, both adversarial and conciliatory activities are needed to maximize the validity of the final design (Henderson, 1987). This implies that we require tools both for enabling experts to understand how they differ and for suggesting ways to resolve their conflicts. Third, it is more effective to focus expert discussion on decision criteria rather than on outcomes (Hammond et al., 1984). This has helped encourage our focus on knowledge rather than process. Finally, the resulting consensus system must be based on a foundation of commonly understood terms, because agreement on the higher-level behavior of the system critically depends on this mutual understanding: If the basic vocabulary differs, the two participants are speaking different, possibly incommensurate languages.

From negotiation, we use the metaphor of the third party mediator. A program for facilitating CKA can be thought of as a facilitator whose job is to aid in resolving discrepancies between two experts. Although CKA is somewhat different from a traditional negotiation situation, there is still a useful resemblance. First, negotiation gives us a vocabulary for characterizing the range of roles a CKA program attempts to fill (e.g., "non-binding arbitrator", "process consultant"). Second, it can help us understand the probable consequences of various discrepancy resolution strategies. For instance, if mediators attempt to resolve easy issues before hard ones, they may create a cooperative climate between the parties, but risk alienating parties who view discussing trivial issues as a waste of time (Rubin, 1981).

4 BOUNDING THE PROBLEM

We make several assumptions to help bound the task we take on here. First, we assume the expertise to be reconciled is homogeneous in the sense that both experts are capable of solving the entire problem. This enables us to focus on resolving discrepancies rather than combining knowledge from distinct fields.

A second, related assumption of our approach is that the experts already have a shared frame of reference, some basic set of assumptions in common. Without that, determining where they agree and disagree would be difficult, not only for our system, but for any human attempting the task.

Third, we assume the experts have constructed individual knowledge bases (KBs) prior to the start of the process. This ensures that the experts can explain the reasoning they used to arrive at their answers and that that reasoning can be adequately captured by a known reasoning process. This in turn allows us to focus on debugging the knowledge — detecting and resolving differences — rather than knowledge acquisition.

Fourth, experts involved in CKA are assumed to have equal influence on the process. The intent is that the content of the consensus KB be determined by rational deliberation rather than political or organizational factors. A related assumption is that any conflict between the experts arises from disagreements about facts and judgments rather than from conflicting interests, as in a bargaining situation.

Finally, as simplifying assumptions at the outset we consider only rule-based representations of knowledge, and only two experts, as a way of providing a foundation for our initial efforts.

Two other points will help to set the context for our work. First, it is a fundamental premise of the work that a consensus KB can perform better than an individual expert's KB. Our hypothesis is that unearthing and resolving differences between two experts will be fundamentally synergistic, removing limitations and defects in both of their KBs. This is plausible but of course not guaranteed: some consensus knowledge bases may not be as good as either of the originals.

Second, our point of view is normative rather than descriptive, unlike much of the work in group decision making, which attempts to describe the complex set of psychological phenomena that occur in such settings (e.g., Janis, 1982). Rather than asking what does happen when groups of experts interact, we ask how two experts should behave to maximize the benefit from collaboration. This is illustrated in part by our assumption above that the multiple experts have equal influence on the process. As with any normative group decision making process, we look for ways of proceeding that attenuate the psychological barriers. We believe that focusing discussion on repairing specific discrepancies in knowledge is one useful mechanism for achieving this.

5 CARTER

We are developing a prototype system for facilitating CKA, dubbed CARTER (Conflict AnalyzeR for Targeted Expert Resolution). The system plays the role of a non-binding arbitrator mediating between two experts (Figure 1).

CARTER examines each expert's KB, looking for matches and conflicts between them, deciding which discrepancy to try to resolve, and suggesting possible resolutions. The two experts discuss the suggested resolution and can choose to update their KBs as suggested, update them in some other manner, or not update them at all. Whatever the decision, the agreed-upon knowledge is added to the third, consensus knowledge base. The experts' KBs are then analyzed anew, with the cycle repeating until those two agree exactly, or no further areas of consensus can be found. In practice, the process is slightly more complex than this, but this gives a sense of the basic structure. We use transcripts of CARTER in operation to illustrate some of our discrepancy resolution techniques.

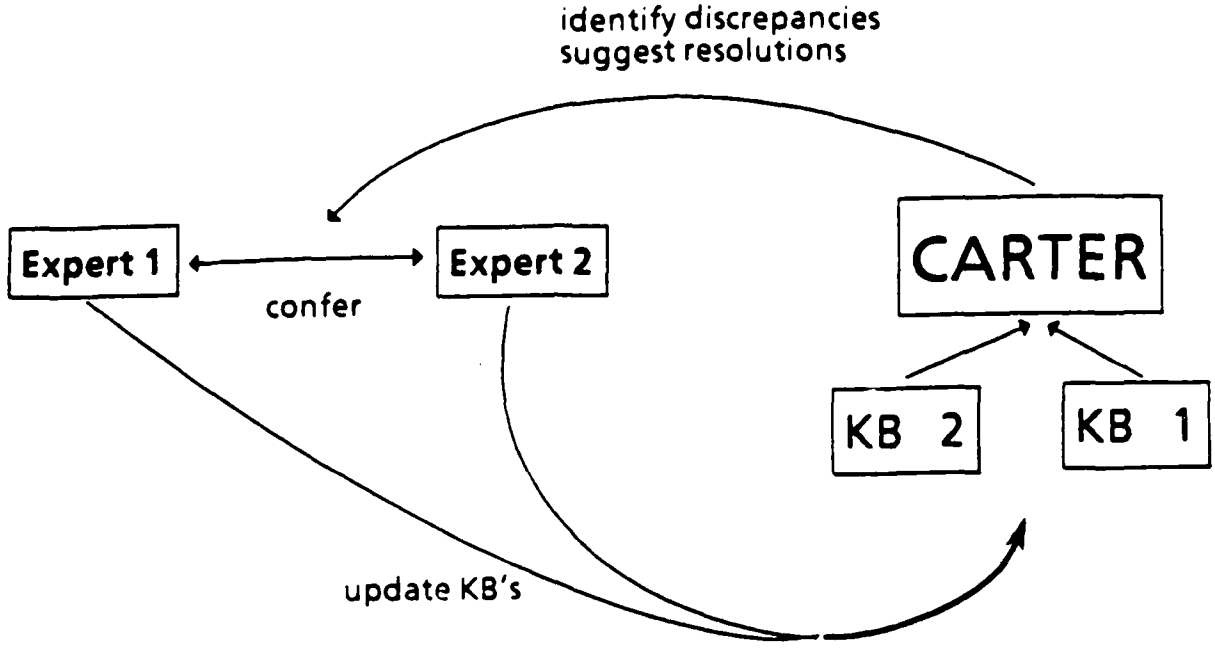

FIGURE 1.
CARTER Scenario

5.1 USING DISCREPANCY KNOWLEDGE

Our initial efforts at CKA focus on information derived from the KBs themselves: CARTER examines the two KBs to detect discrepancies and to determine how they might be made consistent. The KBs consist of rules expressed in terms of object, attribute, and value triples that supply a topology of relationships between the concepts. CARTER's knowledge lies in detecting specific kinds of discrepancies and linking them with one or more potential resolutions.

As an example, imagine that two wine experts, Kevin and Mary, have each constructed a KB that recommends a specific wine to go with dinner, and now wish to create a single, consensus KB. Among the discrepancies they might encounter are:

1. differences in the nature of the outcome: one expert may specify a wine grape (e.g., Pinot-Blanc) while the other specifies both grape and vintage (e.g., Pinot-Blanc '83).

2. differences in vocabularies: one expert may refer to the **body** of the wine, while the other refers to its **robustness**.

3. differences in pattern of inference: the experts may agree on the overall vocabulary, but interconnect them differently, as for instance if one expert uses the **character** of the meal (spicy or bland) to help infer which wine to select, while the other relies on the **category** of the main dish (e.g., meat or fish).

4. differences in the rules: the experts may agree on the vocabulary and interconnection between terms, but suggest different specific values, as for instance if one expert reasons that a turkey dish suggests a white wine, while the other reasons that a turkey dish suggests a rose wine. Both are reasoning from the type of the main dish to the color of wine, but come out with different values.

CARTER's overall strategy is to attack these in the order given. This approach is motivated by both the computational and negotiation character of the task. The computational task faced by the system is one of matching two collections of rules that are at one level simply directed graphs; any useful guidance about where to start the matching process will vastly improve the system's chances of making intelligent suggestions. Expressed in these terms, we anchor the search at the end of the graph, trying first to match the outcomes, then working backwards, matching the nodes connected to the outcome, and continuing to work backward from there. Starting with the outcome is sensible because it relies on the heuristic that two KBs about the same topic are likely to have the same attribute as their goal

Starting with the outcome is also sensible from the negotiation point of view: it is difficult to imagine an effective discussion about the details if the two knowledge bases are trying to arrive at different kinds of conclusions.

Figure 2 shows the beginning of this process. CARTER starts by determining the goal of each KB, a simple task since it is by definition the sole attribute that appears

only in the conclusions of rules (i.e., it is inferred by rules but nothing further is inferred from it). CARTER identifies Kevin's goal as `wine-region` and Mary's as `wine-name`.

```
Expert 1, what is your name?  KEVIN
Expert 2, what is your name?  MARY

KEVIN and MARY, the first thing I want to do is get some basic
agreement on what the goal of the consensus KB should be.  I am
analyzing your individual KB's    an attempt to match them up.

OK.  Here are the results of my analysis.

KEVIN has goal WINE-REGION
MARY has goal WINE-NAME
```

Figure 2: Identifying goals.

The system's next task is to decide as best it can whether these two things represent identical concepts. The judgment about the real meaning of these two terms can only come from the experts, but the system can make a surprisingly good guess by examining three kinds of circumstantial evidence available in the knowledge base:

- Are the concept labels the same? In this case they are not (`wine-region` vs. `wine-name`), but this can of course be an artifact of name-choice or (in other circumstances) variations in spelling or abbreviation. Conversely, a match in labels is useful evidence but no guarantee of match in meaning.
- In the case of attributes, are the values the same? Once again here the answer is no (e.g., `california, rhone,` etc., vs. `chablis, gamay` etc.).
- Are they inferred from the same concepts and are they in turn used to infer the same concepts? That is, do the occupy similar places in the topology of the knowledge base? Once again in this case the answer is no.

Note that the last form of evidence makes the process recursive: to determine whether two concepts in the conclusion of a rule are the same, we need to determine whether the concepts mentioned in the premise are the same, thereby starting the process all over again with the premise concepts.

Weighing the evidence in the case at hand (Figure 3), CARTER concludes that **wine-region** and **wine-name** are not identical concepts. In response it tries a different tactic, invoking the heuristic that the goal attribute of one KB might be found on the route to the goal attribute of the other KB. That is, since the two endpoints in the graphs do not match, perhaps the endpoint in one matches with one of the intermediate points (conclusions) in the other. To explore this possibility, CARTER tries to match **wine-region** of Kevin's KB (KKB) with the attributes that determine **wine-name** of Mary's KB (MKB), using the same criteria of label, value, and topological correspondence.

As it turns out, this too fails, so CARTER tries it the other way around, attempting to match **wine-name** of MKB with the attributes determining **wine-region** in KKB. One of these attributes, **wine-grape**, happens to match rather well with **wine-name**. Although their names are not the same, they share eight different values, along with three attributes used to determine them. This is strong evidence that **wine-grape** and **wine-name** are the same concepts.

```
I'm afraid these goals do not match.
Also, KEVIN's goal doesn't match up with any concepts in MARY's KB.

However, MARY'S goal does seem to correspond to an attribute
determining KEVIN's goal, in particular, WINE-GRAPE, because
   The domain of WINE-GRAPE of KEVIN's KB and
   the domain of WINE-NAME of MARY's KB match up:
   Values in common are:  CHABLIS CHARDONNAY CHENIN-BLANC RIESLING
                          GAMAY PINOT-NOIR ZINFANDEL SAUVIGNON-BLANC
   Values only KEVIN has are:  PINOT-GRIS SEMILLION NAPA-GAMAY
                               PINOT-BLANC
   Values only MARY has are:  BURGUNDY VALPOLICELLA SOAVE.

   In addition,
   WINE-COLOR, WINE-SWEETNESS, and WINE-BODY of KEVIN's KB, which
        determine WINE-GRAPE, match with
   WINE-RECOMMENDEDCOLOR, WINE-RECOMMENDEDSWEETNESS, and
   WINE-RECOMMENDEDBODY of MARY's KB, which determine WINE-NAME.

Therefore, it would seem that WINE-REGION is the result of an extra
operation on the data KEVIN's KB performs that MARY's KB does not.
```

Figure 3: Matching goals.

The strong match between **wine-grape** and **wine-name** enables CARTER to propose a specific diagnosis about the specific discrepancy between the two KBs. Since (Figures 3 and 4) **wine-name**, the goal attribute of MKB, seems to match the concept **wine-grape**, a concept on the route to the goal of KKB, CARTER concludes that **wine-region** is a concept that reflects an additional inference that only Kevin's KB performs, thereby providing a more specific recommendation (not just a wine grape, a grape grown in a particular region).

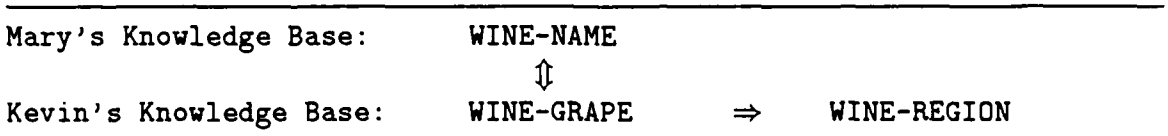

Figure 4: Aligning the two KBs.

Now that CARTER has analyzed the discrepancy, it can propose a plan to reconcile it (Figure 5). One of the repair actions suggests that, when one KB has an attribute the other lacks, the discrepancy can be resolved either by adding the attribute to one KB or deleting it from the other. That is, the experts need to decide whether the consensus KB should use the extra attribute.

```
I recommend that you do the following to reach agreement on the
goals of the consensus KB.

 First, decide whether WINE-REGION should be included in the
 consensus KB.

 If so, MARY adds the WINE-REGION attribute to her KB and generates
 anew set of rules for determining WINE-REGION.

 If not, KEVIN deletes the WINE-REGION attribute from his KB, and we
 consolidate the attributes WINE-GRAPE and WINE-NAME.

Do you approve of this plan? Y
```

Figure 5: The repair plan.

At this point, CARTER asks the experts to decide. Whatever the outcome, there

is additional work to do. Since **wine-name** and **wine-grape** are the same underlying concept, they must be reconciled. In addition, if the experts decide to include **wine-region** in the consensus KB, Mary must provide her version of the rules that take the additional inference step, determining **wine-region** from **wine-name**, then those rules must be compared with Kevin's and any discrepancy resolved.

The experts indicate this plan as reasonable. CARTER then provides information about **wine-region** to assist them (Figure 6), and after some discussion, the experts choose not to include it. It is deleted from KKB.

```
Here is some information cn WINE-REGION:
Attribute name:  WINE-REGION
KB's found in:  KEVIN's
Possible values:  CALIFORNIA RHONE RED-BURGUNDY ST-EMILLION BEAUJOLAIS
                  RED-BURGUNDY WHITE-BURGUNDY COTES-DU-RHONE
Objects attribute affects:  None
Attributes determined by:  WINE-GRAPE WINE-COLOR

Now, given this information, do you wish to retain WINE-REGION for
the consensus KB? (Take all the time you want to discuss it) N

OK.  I'll delete WINE-REGION from KEVIN's KB now, OK? Y

WINE-REGION deleted.
```

Figure 6: Resolving the discrepancy

As indicated by the plan CARTER supplied, the next task is to consolidate **wine-grape** and **wine-name**. Once again CARTER knows about this kind of discrepancy and how to resolve it: the experts should choose a shared attribute name and agree on a consolidated list of values for the attribute (Figure 7). The experts choose **wine-type** as the shared term.

```
Next, I am going to try to help you consolidate WINE-GRAPE and
WINE-NAME.

Given how they match up, are WINE-GRAPE and WINE-NAME the same
concept? Y

What do you two want to call this concept? WINE-TYPE

Fine.  I will change the names.
```

Figure 7: Consolidating two concepts.

Reconciling the values is more involved, since we are not sure how the mismatched values correspond. CARTER tries to solve the problem by using the topology of the KBs to attempt to match the values. As with attribute labels, if two values with different labels are linked to other values that do correspond (i.e., they have the same label), the different labels may in fact be referring to the same underlying concept. The heuristic here is that rules with similar premises typically have similar conclusions and that the apparent difference in conclusions can arise simply because the experts are using different names (or varying spelling) for the same concept.

To test this theory, CARTER retrieves the rules that determine **wine-type** in both KBs and picks out those with identical premise value labels but differing conclusions (Figure 8). This process picks out **pinot-gris** in KKB and **soave** in MKB, because they are both inferred by rules that reason from a wine whose **sweetness** is **dry**, and whose **color** is **white**.

```
Now, let's establish the consensus values for WINE-TYPE.
   So far, we have the following set of consensus values:
   CHABLIS CHARDONNAY CHENIN-BLANC RIESLING GAMAY PINOT-NOIR ZINFANDEL
   SAUVIGNON-BLANC

   Values still to be accounted for are
   PINOT-GRIS SEMILLION NAPA-GAMAY PINOT-BLANC in KEVIN's KB and

   BURGUNDY VALPOLICELLA SOAVE in MARY's KB.

Some of these values may be corresponding concepts with different
labels; others may needed to be added to or deleted from the
consensus KB.

Would you like some suggestions about how the labels might
correspond?  Y

PINOT-GRIS may correspond to SOAVE since they both appear in similar
rules:
         KEVIN's KB                          MARY's KB

IF WINE-COLOR is WHITE and          IF WINE-RECOMMENDEDCOLOR is WHITE and
   WINE-SWEETNESS is DRY               WINE-RECOMMENDEDSWEETNESS is DRY
THEN WINE-TYPE is PINOT-GRIS        THEN WINE-TYPE is SOAVE

Are PINOT-GRIS and SOAVE two different names for the same concept? N
Ok.

I have no further suggestions for how any of the unmatched values
might correspond.
The possible correspondences are:
1) different names for the same concept
2) different concepts that should be added to the consensus KB
3) different concepts that should be deleted from the consensus KB.

Please indicate which category each value belongs in, and I'll help you
make the necessary changes.
```

Figure 8: Attempting to match values.

As it turns out, in this case the strategy fails, since pinot-gris and soave are in fact different types of wine. At this point CARTER has run out of useful suggestions about how to pair off the unmatched values in each knowledge base. Some of the other values may correspond, but CARTER cannot detect them because the experts have named them differently and used different premises to conclude them. The remaining non-shared values are simply missing from one of the knowledge bases. The system now offers one final piece of assistance, informing the experts of the kinds of discrepancies that can occur (accidental naming mismatches and genuine differences in the coverage of each KB), and asks them to determine the correct analysis for each of the non-shared values.

After reaching agreement on the goal for the consensus knowledge base, the system then retrieves the attributes that determine the goal in each of the individual KBs and uses the techniques just illustrated to get agreement on these attributes. Once the system has agreement on these related attributes, it begins to detect and remove incompleteness and inconsistency in the rules that link them. Figure 9 shows an example of resolving two inconsistent rules: two rules with identical premises but different conclusions.

```
Now, let's try to get agreement on the rules determining WINE-COLOR.

I have noticed the following discrepancy:

        KEVIN's KB                              MARY's KB

IF ENTREE-NAME is TURKEY                IF ENTREE-NAME is TURKEY
THEN WINE-COLOR is WHITE                THEN WINE-COLOR is ROSE

Since you two have already discussed the vocabulary involved here, I
am pretty sure this problem is not due to a failure to consolidate
WHITE and ROSE, or a misunderstanding about the meaning of
ENTREE-NAME or WINE-COLOR.

Would you like to include both of these rules in the consensus
KB?  N
```

Figure 9: Detecting and analyzing inconsistent rules.

CARTER knows three ways in which this can happen:

1. a misunderstanding about the vocabulary: white and rose could be synonyms.

This is ruled out because the experts have already agreed on the vocabulary.

2. there is not really a mismatch because both rules should be in both knowledge bases (each expert forgot one rule that the other remembered).
   CARTER offers them this option, but they decline.

3. both rules are over-generalized as stated: they are both missing an attribute whose value constitutes an important unstated assumption that the experts know but forgot to make explicit.

When CARTER inquires about this last possibility (Figure 10), the experts agree that they omitted information about the **entree-sauce**. When they elaborate their rules to include this, we discover that they had different assumptions about the corresponding value, so the rules are in fact mutually compatible and the experts agree that they should both be included in the consensus KB. The process continues with the system guiding the experts in resolving remaining details about the new attribute **entree-sauce**.

```
In that case, let me propose another solution.

It is possible that you two are making different assumptions about
some other attribute which is important for determining the
conclusion of the rule, but that neither of you have included in
your KB.  Take a moment to think about and discuss what other
factors might be important in determining WINE-COLOR.

Can you think of any other factors? Y

What do you want to call this factor? ENTREE-SAUCE

OK.  KEVIN, what value for ENTREE-SAUCE did you have in mind when
you wrote your rule?  NONE

MARY, what value for ENTREE-SAUCE did you have in mind when you
wrote your rule?  SWEET

If we included ENTREE-SAUCE in the two rules, they would look like
this:
        KEVIN's KB                                MARY's KB

IF ENTREE-NAME is TURKEY      and        IF ENTREE-NAME is TURKEY and
   ENTREE-SAUCE is NONE                     ENTREE-SAUCE is SWEET
THEN WINE-COLOR is WHITE                 THEN WINE-COLOR is ROSE

Are both of you willing to include both of these rules in the
consensus KB?  Y

Great!  Now, to complete the repair process, we have to determine
the set of consensus values for ENTREE-SAUCE, and add it into the
other rules linking ENTREE-NAME and ENTREE-SAUCE to WINE-COLOR as
appropriate.  Then, we can go to work on the discrepancies which
arise in these modified rules.
```

Figure 10: *Resolving the difference.*

5.2 CATALOGING DISCREPANCY KNOWLEDGE

CARTER's expertise lies in recognizing and repairing a variety of discrepancies. This knowledge is organized in a catalog currently containing ten entries, each of which consists of a discrepancy detection procedure and a corresponding set of resolution procedures. This simple, detection-resolution style organization of the catalog makes it easier to add new entries as we gain more experience with consensus knowledge acquisition.

To determine the kinds of discrepancies we needed to cover, we systematically compared a number of KBs at three different levels of abstraction. Viewing a knowledge base as a functional relationship leads us to focus on inputs (test data) and outputs (the generated recommendations). Viewing it in terms of individual rules centers on the detailed relationships between attribute-object-value triples. Those triples in turn define the vocabulary of the experts. Studying KBs from each of these three points of view gives us some assurance that we have achieved reasonable coverage of the set of possible discrepancies.

We also found that discrepancies can be resolved through four general mechanisms that cut across these levels of abstraction: (i) negation, (ii) incorporation, (iii) compromise, and (iv) elaboration. Negation suggests that one expert should change something to remove a defect in his knowledge base, when the other expert convinces him that one of his judgments is incorrect. Incorporation suggests that one expert should add something to his KB that the other already has (or conversely that the other expert should remove it). This is useful when one KB has an incomplete set of objects, rules, or test cases (or the other KB has extraneous objects, rules, or test cases). Figure 6 provides an illustration with the deletion of `wine-region`.

Compromise suggests that both experts change their KBs. It is helpful when the experts wish to establish a shared vocabulary or negotiate an intermediate settlement. One example is the decision to use `wine-type` as the shared name in Figure 7.

Elaboration suggests that both experts add something to the KBs to remove discrepancies not otherwise resolvable. It is needed when a problem is not localizable in either KB individually, as when `entree-sauce` had to be added in Figure 10.

The discrepancy catalog is a domain-independent source of knowledge that systemizes our approach to CKA. We can account for the differences between KBs in terms of the catalog entries and attempt to remove them through an associated resolution mechanism. The result is a tool for partitioning the CKA problem and supporting the solution of each of the subproblems.

6 RELATED WORK

Two previous efforts are similar in general spirit to ours. A previous use of debugging in this general area is the Delphi technique (Helmer and Rescher, 1959; Jagannathan and Elmaghraby,1985), used to achieve consensus among a group of experts on a specific issue. It is a three-step, iterative process involving, (i) submitting individual

opinions and their supporting reasoning to a skilled facilitator, (ii) preparation of a summary report by the facilitator, and (iii) forwarding of the report back to the experts. Here the facilitator plays a role in debugging by attempting to clarify the specific areas of disagreement among the experts. Although entirely manual, this role demonstrates one possible activity of a CKA debugging program, that of "setting the agenda" for discussion between experts.

More recently, work by Boose (1986) and Plaza et al. (1987) has been focused on using knowledge of multiple experts. They concentrate for the most part on a number of schemes for using the knowledge rather than resolving discrepancies. They suggest combining expertise simply by adding both experts' knowledge to a single knowledge base, tagging each rule with its author, and then allowing a number of basic strategies. In one case the user simply has to decide which expert to believe, in another the user can weight the experts' opinions, etc. The guidance they do offer in reaching consensus on the knowledge is relatively modest. They proceed from the repertory grid notion (Kelly, 1955) that underlies their work and suggest that all the vocabulary terms used by each expert individually to construct his own grid be combined to form a single, larger vocabulary that will then be used by each expert to construct a new grid. They acknowledge that the experts may be unfamiliar with each other's terms and "may have to 'guess' what was meant" when they encounter an unfamiliar term in the grid.

Another recent study (Klein et al., 1989) addresses the issue of resolving conflicting design specifications. Through direct observation of architects cooperatively developing a design for a house, they developed a conflict class hierarchy for identifying and resolving differences between design alternatives. Although this typology of conflicts is similar in some respects to our discrepancy catalog, one important difference is in the content: their primary focus is on reconciling the designs themselves rather than design knowledge.

7 CONTRIBUTIONS, EXTENSIONS, LIMITATIONS

The primary contribution of this work is the store of detailed information we have codified for facilitating CKA. It represents a small but growing and relatively systematic expression of knowledge that was previously informal, experiential, and largely tacit.

A second contribution arises from the surprisingly effective degree of bootstrapping the system displays. The system must make its best guess about the meaning of a term from the way it is used in a knowledge base, it can gather only circumstantial evidence of the sort we reviewed above, and it must, paradoxically, gather that evidence from the very same knowledge bases it is attempting to modify to reach consensus. It it thus striking how effective the system's heuristics are at guiding it, allowing it to make plausible judgments about which concepts match and so that even when it has to ask the experts, the questions are for the most part sensible and well

chosen.

A third contribution is the role of our work as a general model for constructing systems that detect and resolve knowledge-level discrepancies. While our current system removes discrepancies in rules and attribute-object-value triples, we believe debugging and repair strategies can equally well be organized around other kinds of knowledge structures, including decision trees, frames, and database schemas. The fundamental process involves three steps: identify the various elements of the representation (e.g., alternatives, events, payoffs, probabilities), develop a taxonomy of how the representations can differ across these elements (e.g., one expert has an additional alternative), and finally prescribe possible resolutions for each of these discrepancies (e.g., one expert has to add an alternative). The three resolution mechanisms described above may provide additional guidance in this last step.

The fourth contribution is the ability of the debugging approach to support the early phase of CKA. Recall that the other techniques for reconciling multiple experts—combination and argumentation—are most effective only after we have established that a conflict exists and that it is difficult or impossible to resolve. Our technique is useful in the important previous stage when we are still trying to make sense of how the KBs compare. It would be unwise for experts to argue about their differing positions before they had established that a real conflict existed. The size of the discrepancy catalog suggests that it is surprising how many inconsistencies are reconcilable without resorting to argument or outcome combination methods.

7.1 Future Work

Although the discrepancy catalog is an important and effective first step, considerable work remains. One of the most important areas for future research is the question of discrepancy resolution strategy. While the strategy discussed in Section 5.1 (starting at the outcome and working backward) is very useful, it is only one of many possibilities. One problem is that this may be a bit too myopic to be effective in a large scale knowledge base. The system in effect immediately dives into the details and its needs a better sense of the larger picture. Our next task is thus to generate a number of strategies and evaluate them in terms of (i) the efficiency and effectiveness with which they increase the degree of consensus, and (ii) the naturalness and coherence of the dialogues they produce.

We will investigate strategies organized around two kinds of approaches. The first approach relies on systematic traversal of the KB. One example of this was illustrated earlier (working backward from the goal); we intend to examine two others that are also likely to be effective: forward from inputs and working in both directions from any agreed on point. We expect that working forward from inputs should be effective on the grounds that the two KBs are likely to work from the same basic information. We believe that begining at an intermediate point of agreement and expanding in both directions will exploit the strategy of emphasizing what the experts already agree on and building from this foundation.

The other approach involves assigning a score to each discrepancy based on its severity (how dissimilar the two concepts are), the value of resolving it (how much resolution would increase the degree of consensus), and the likelihood that it will not be resolved through the removal of another discrepancy (e.g., when the resolution of an attribute discrepancy reconciles their corresponding rules). The system's choice of discrepancy could then be guided by a hill-climbing strategy, always choosing the discrepancy with the highest score. Another form of guidance can be supplied by precedence relationships between both knowledge base elements (e.g., attributes should be resolved before rules) and resolution mechanisms (e.g., attempt incorporation before elaboration). Thus far we have implemented scores based on severity, determining which attributes to match up next.

Several other extensions to the system may also be desirable. First, additional knowledge, not available from the structure of the KBs themselves, will likely provide the system with additional power. Clancey (1986), for example, notes that many current rule-based systems employ a problem solving technique called structured selection, characterized by abstracting from specific data (e.g., classifying a patient based on patient data), followed by heuristic matching (associating a patient class to a disease), and then solution refinement (refining from disease category to specific disease). Each of these three subtasks is carried out by different sets of rules. If CARTER could determine which rules belonged to each subtask, it could use this knowledge to characterize discrepancies more precisely and organize its presentation of choices to the expert. It would as a result be using knowledge about the character of the task (structured selection), in addition to its existing knowledge about rules, attribute-object-value triples, etc.

Second, for the cases in which debugging alone fails to result in a consensus KB, it would be helpful to give CARTER the ability to support formal argumentation between the experts or suggest resolutions based on combination methods (e.g., averaging certainty factors). Finally, we might streamline the resolution process, as in the instances in which the incrementalism of the debugging approach is inefficient. For example, the system may prescribe a number of isolated modifications to the KBs when it would be easier simply to redo an entire section all at once. It would be nice to be able to recognize such situations.

The bootstrapping nature of the system has substantial implications for its performance. In general, the more any bootstrapping program knows, the more effectively it can perform, and conversely. CARTER will perform well when a large number of similarities exist from which to gain a foothold, but will degrade significantly when few are found. Seemingly trivial differences like different abbreviations in the labels used for values can make matching very difficult.

Our attempt to discern meaning of terms by bootstrapping from the existing knowledge base can also run into trouble in circumstances that are unusual, but not impossible. The question of whether two concepts mean the same thing is in fact deep and in general extremely difficult to answer with assurance. Even the best human

mediator working with two cooperative experts may find out only after considerable time has elapsed that two terms thought to be synonymous in fact had importantly different shades of meaning. The best we can do here is accumulate all available circumstantial evidence and use it in the most effective order (comparing names, values, and topology, then eventually asking the experts). In doing so we reduce the chance of being misled, but must remain aware of the possibility of it happening.

8 CONCLUSION

We have described a novel approach to and prototype system for facilitating consensus knowledge acquisition. The key contributions of this work include the development of a detailed store of knowledge for detecting and resolving discrepancies in rule-based systems and a general procedure for developing similar systems for other representations. We expect the next advance in this area to come from implementing improved discrepancy resolution strategies. This work will serve as the starting point for understanding more generally how experts reach consensus and how we can best support them in their efforts to do so.

REFERENCES

Aczel, J., and Saaty, T. "Procedure for Synthesizing Ratio Judgements," *Journal of Mathematical Psychology*, Volume 27, Number 1, March 1983, 93–102.

Brehmer, B., and Hagafors, R. "The Use of Experts in Complex Decision Making: A Paradigm for the Study of Staff Work," *Organizational Behavior and Human Decision Processes*, Volume 38, Number 2, October 1986, pp. 181–196.

Boose, J. *Expertise Transfer for Expert Systems Design.* Amsterdam: Elsevier,1986.

Brown, J., and Burton, R. "Diagnostic Models for Procedural Bugs in Basic Mathematical Skills," *Cognitive Science*, Volume 2, Number 2, April–June 1978,pp. 155–192.

Clancey, W. "Heuristic Classification," in Kowalik, J., ed,. *Knowledge-Based Problem-Solving.* New York: Prentice Hall, 1986.

Davis, J. H., "Group Decision and Procedural Justice," in Fishbein, M., ed., Progress in Social Psychology, Vol. 1., Hillsdale, NJ: Erlbaum, 1980.

Davis, R. et al. "Production Rules as a Representation for a Knowledge-Based Consultation Program", *Artificial Intelligence*, Volume 8, Number 1, February 1977, pp. 15–45.

Davis, R. "Interactive Transfer of Expertise," *Artificial Intelligence*, Volume 12, Number 2, August 1979, pp. 121–157.

Davis, R. "Expert Systems: Where are We? And Where Do We Go From Here?" MIT AI Lab Memo 665, 1982. Also reprinted in AI Magazine, Summer 1982.

Fedorowicz, J., and Manheim, M. "A Framework for Assessing Decision Support Systems and Expert Systems," *Proceedings of the Sixth International Conference on Decision Support Systems*, 1985, pp. 116–127.

Fellers, J. "Skills and Techniques for Knowledge Acquisition: A Survey, Assessment, and Future Directions," *Proceedings of the Eighth International Conference on Information Systems*, 1987, pp. 118–132.

Fogelin, R.*Understanding arguments: An introduction to informal logic.* New York: Harcourt, Brace, Javanovich, 1982.

Gaglio, S. et al. "Multiperson Decision Aspects in the Construction of Expert Systems," *IEEE Trans. on Sys., Man, and Cybernetics*, Volume 15, Number 4,July-August 1985, pp. 536–539.

Hammond, K. et al. "Social Judgment Theory," in *Human Judgment and Decision Processes*, Kaplan, M., and Schwartz, S., eds. New York: Academic Press,1975.

Helmer, O., and Rescher, N. "On the Epistemology of the Inexact Sciences," *Management Science*, Volume 6, Number 1, October 1959, pp. 25–52.

Henderson, J. "Managing the IS Design Environment: A Research Framework," Center for Information Systems Research Working Paper #158, MIT, May 1987.

Janis, I. *Groupthink: Psychological Studies of Policy Decisions and Fiascos.* Boston: Houghton Mifflin, 1982.

Jagannathan, V., and Elmaghraby, A. MEDKAT: Multiple Expert Delphi-Based Knowledge Acquisition Tool, Engineering Mathematics and Computer Science Department Technical Report, Univ. of Louisville, 1985.

Kelly, G. *The Psychology of Personal Constructs.* Boston: Norton, 1955.

Klein, M., et al. "Towards a Theory of Conflict Resolution in Cooperative Design," Proceedings of the 9th Workshop on Distributed Artificial Intelligence, 1989, pp. 329-349.

Kuper, R., Dependency-Directed Localization of Software Bugs, MIT-AI-TR-1053, MIT AI Lab, May, 1989.

Lowe, D. "Co-operative Structuring of Information: The Representation of reasoning and debate," *Int. J. Man-Machine Studies*, Volume 23, Number 2, August 1985, pp. 97–111.

Mason, R.O., and Mitroff, I. *Challenging Strategic Planning Assumptions.* New York: Wiley, 1981.

Miner, F. "Group vs. Individual Decision Making: An Investigation of Performances Measures, Decision Strategies, and Process Losses/Gains," *Organizational Behavior and Human Decision Processes*, Volume 33, Number 1, February 1984, pp. 112–125.

Mittal, S., and Dym, C.L. "Knowledge Acquisition from Multiple Experts," *The AI Magazine*, Volume 6, Number 2, Summer 1985, pp. 32–36.

Mumford, E. "Managerial Expert Systems and Organizational Change: Some Critical Research Issues," in *Critical Issues in Information Systems Research*, Boland, R., and Hirschheim, R., eds. New York: Wiley, 1987.

Nunnamaker, J., et al. "Computer-aided Deliberation: Model Management and Group Decision Support," *Operations Research*, Volume 36, Number 6, November–December 1988, pp. 826–848.

Plaza, E. et al. "Consensus and Knowledge Acquisition," in *Uncertainty in Knowledge-Based Systems.* Bouchon, B., and Yager, R., eds. Berlin: Springer-Verlag, 1987. pp. 294–306.

Rubin, Z., ed. *Dynamics of Third Party Intervention: Kissinger in the Middle East.* New York: Praeger, 1981.

Smolensky, P. et al., "Constraint-Based Hypertext for Argumentation," Dept. of Comp Sci. and Linguistics, University of Colorado WP #CU-CS-358-87, 1987.

Stefik M, et al., Beyond the chalkboard, *CACM*, **30**:1, Jan 1987, pp. 32–47.

Toulmin, S. *The Uses of Argument.* Cambridge, England: Cambridge University Press, 1958.