AI Magazine Volume 25 Number 4 (2004) (© AAAI)

Project Halo
Towards a Digital Aristotle

Noah S. Friedland, Paul G. Allen, Gavin Matthews, Michael Witbrock, David Baxter, Jon Curtis, Blake Shepard, Pierluigi Miraglia, Jürgen Angele, Steffen Staab, Eddie Moench, Henrik Oppermann, Dirk Wenke, David Israel, Vinay Chaudhri, Bruce Porter, Ken Barker, James Fan, Shaw Yi Chaw, Peter Yeh, Dan Tecuci, and Peter Clark

■ Project Halo is a multistaged effort, sponsored by Vulcan Inc, aimed at creating Digital Aristotle, an application that will encompass much of the world's scientific knowledge and be capable of applying sophisticated problem solving to answer novel questions. Vulcan envisions two primary roles for Digital Aristotle: as a tutor to instruct students in the sciences and as an interdisciplinary research assistant to help scientists in their work.

As a first step towards this goal, we have just completed a six-month pilot phase designed to assess the state of the art in applied knowledge representation and reasoning (KR&R). Vulcan selected three teams, each of which was to formally represent 70 pages from the advanced placement (AP) chemistry syllabus and deliver knowledge-based systems capable of answering questions on that syllabus. The evaluation quantified each system's coverage of the syllabus in terms of its ability to answer novel, previously unseen questions and to provide human-readable answer justifications. These justifications will play a critical role in building user trust in the question-answering capabilities of Digital Aristotle.

Prior to the final evaluation, a "failure taxonomy" was collaboratively developed in an attempt to standardize failure analysis and to facilitate cross-platform comparisons. Despite differences in approach, all three systems did very well on the challenge, achieving performance comparable to the human median. The analysis also provided key insights into how the approaches might be scaled, while at the same time suggesting how the cost of producing such systems might be reduced. This outcome leaves us highly optimistic that the technical challenges facing this effort in the years to come can be identified and overcome.

This article presents the motivation and long-term goals of Project Halo, describes in detail the six-month first phase of the project—the Halo Pilot—its KR&R challenge, empirical evaluation, results, and failure analysis. The pilot's outcome is used to define challenges for the next phase of the project and beyond.

Aristotle (384–322 BCE) was remarkable for the depth and scope of his knowledge, which included mastery of a wide range of topics from medicine and philosophy to physics and biology. Aristotle not only had command over a significant portion of the world's knowledge, but he was also able to explain this knowledge to others, most famously, though briefly, to Alexander the Great.

Today, the knowledge available to humankind is so extensive that it is not possible for a single person to assimilate it all. This is forcing us to become much more specialized, further narrowing our worldview and making interdisciplinary collaboration increasingly difficult. Thus, researchers in one narrow field may be completely unaware of relevant progress being made in other neighboring disciplines. Even within a single discipline, researchers often find themselves drowning in new results. MEDLINE,[1] for example, is an archive of 4,600 medical publications in 30 languages, containing over 12 million publications, with 2,000 added daily.

Making the full range of scientific knowledge accessible and intelligible might involve anything from simply retrieving facts to answering a complex set of interdependent questions and providing appropriate justifications for those answers. Retrieval of simple facts might be achieved by information-extraction systems searching and extracting information from a large corpus of text, such as Voorheese (2003). But aside from the simplicity of the types of questions such advanced retrieval systems are designed to answer, they are only capable of retrieving "answers"—and justifications for those answers—that already exist in the corpus. Knowledge-based question–answering systems, by contrast, though generally more computationally intense, are capable of generating answers and appropriate justifications and explanations that are not found in texts. This capability may be the only way to bridge some interdisciplinary gaps where little or no documentation currently exists.

Project Halo is a multistaged effort aimed at creating Digital Aristotle (DA), an application encompassing much of the world's scientific knowledge and capable of answering novel questions through advanced problem solving. DA will act both as a tutor capable of instructing students in the sciences and as a research assistant with broad interdisciplinary skills, able to help scientists in their work. The final DA will differ from classical expert systems in four important ways.

First, in speed and ease of knowledge formulation. Classical expert systems required years to perfect and highly skilled knowledge engineers to craft them; Digital Aristotle will provide tools to facilitate rapid knowledge formulation by domain experts with little or no help from knowledge engineers.

Second, in coverage. Classical expert systems were narrowly focused on the single topic for which they were specifically designed; DA will over time encompass much of the world's scientific knowledge.

Third, in reasoning techniques. Classical expert systems mostly employed a single inference technology; DA will employ multiple technologies and problem solving methods.

Fourth, in explanations. Classical expert systems produced explanations derived directly from inference proof trees; DA will produce concise explanations, appropriate to the domain and the user's level of expertise.

Adoption by communities of subject matter experts of the Project Halo tools and methodologies is critical to the success of DA. These tools will empower scientists and educators to build the peer-reviewed, machine-processable knowledge that will form the foundation for Digital Aristotle.

The Halo Pilot

The pilot phase of Project Halo was a six-month effort to set the stage for a long-term research and development effort aimed at creating Digital Aristotle. The primary objective was to evaluate the state of the art in applied KR&R systems. Understanding the performance characteristics of these technologies was considered to be especially critical to DA, as they are expected to form the basis of its reasoning capabilities. The first objectives were to identify and engage leaders in the field and to develop suitable evaluation methodologies; the project was also designed to help in the determination of a research and development roadmap for KR&R systems. Finally, the project adopted principles of scientific transparency aimed at producing understandable, reproducible results.

Vulcan undertook a formal bidding process to identify teams to participate in the pilot. Criteria for selection included a well-established and mature technology and a world-class team with a track record of government and private funding. Three teams were contracted to participate in the evaluation: a team led by SRI International with substantial contributions from Boeing Phantom Works and the University of Texas at Austin; a team from Cycorp; and a team from Ontoprise.

Significant attention was given to selecting a proper domain for the evaluation. It was important, given the limited scope of this phase of the project, to adapt an existing, well known evaluation methodology with easily understood and objective standards. First a decision was made to focus on a "hard" science and, more specifically, on a textbook presentation of some part of that science. Several standardized test formats were also examined. In the end, a 70-page subset of introductory college-level advanced placement (AP) chemistry was selected because it was reasonably self-contained and did not require solutions to other hard AI problems, such as representing and reasoning with uncertainty, or understanding diagrams (Brown, LeMay, and Bursten 2003). This latter consideration, for example, argued against selecting physics as a domain.

Table 1 lists the topics in the chemistry syllabus. Topics included stoichiometry calculations with chemical formulas; aqueous reactions and solution stoichiometry; and chemical equilibrium. Background material was also identified to make the selected chapters more fully self-contained.[2]

This scope was large enough to support a large variety of novel, and hence unanticipated, question types. One analysis of the syllabus identified nearly 100 distinct chemistry laws, suggesting that it was rich enough to require complex inference. It was also small enough to be represented relatively quickly—which was essential because the three Halo teams were allocated only four months to create formal encodings of the chemistry syllabus. This amount of time was deemed sufficient to construct detailed solutions that leveraged the existing technologies, yet was too brief to allow significant revisions to the teams' platforms. Hence, by design, we were able to avoid undue customization to the task domain and thus to create a true evaluation of the state of the art of KR&R technologies.

Nevertheless, at the outset of the project it was completely unclear whether competent systems could be built. In fact, Vulcan's secret intent was to set such a high bar for success that the experiment would expose the weaknesses in KR&R technologies and determine whether these technologies could form the foundation of DA. The teams accepted the challenge with trepidation caused by several factors, including the mystery of working in a new domain with the novel performance task of answering hard, and highly varied, advanced placement questions; and generating coherent explanations in English—all within four months.

The Technology

The three teams had to address the same set of issues: knowledge formation, question answering, and explanation generation (Barker et al. 2004; Angele et al. 2003; and Witbrock and Matthews 2003.) They all built knowledge bases in a formal language and relied on knowledge engineers to encode the requisite knowledge. Furthermore, all the teams used automated deductive inference to answer questions. Despite these high-level similarities, the teams' approaches differed in some interesting ways, especially with respect to explanation generation.

Knowledge Formation

Each system achieved significant coverage of the parts of the domain represented by the syllabus and was able to use that coverage to answer substantial numbers of novel questions. All three systems used class taxonomies, such as the one illustrated in figure 1, to organize concepts such as acids, physical constants, and reactions; represented properties of classes using relations; and used rules to represent complex relationships.

Subject	Chapters	Sections	Pages
Stoichiometry: Calculations with Chemical Formulas	3	3.1 – 3.2	75 - 83
Aqueous Reactions and Solution Stoichiometry	4	4.1 – 4.4	113 - 133
Chemical Equilibrium	16	16.1 – 16.11	613 - 653

Table 1. Course Outline for the Halo Challenge.

Domain-Driven Versus Question-Driven Knowledge Formation

Recall that Vulcan released a course description consisting of 70 pages of a chemistry textbook and 50 sample questions. The teams had the choice of building knowledge bases either starting from the syllabus text or from the sample questions or working from both in parallel. Ontoprise and Cycorp approached knowledge formation in a target-text-driven approach, and SRI approached knowledge formation in a question-driven approach.

The Ontoprise team encoded knowledge in three phases. During the first phase team members encoded the knowledge within the corpus into the ontology and rules without considering any sample test questions. They then tested this knowledge on test questions that appeared in the textbook—which were different from the sample set released by Vulcan. In the second phase, they tested the sample questions released by Vulcan. The initial coverage they observed was around 30 percent. During this phase, they refined the knowledge base until coverage of around 70 percent was reached. In the second phase, they also coded the explanation rules. In the third phase, they refined the encoding of the knowledge base and the explanation rules.

Cycorp used a hybrid approach by first concentrating on representing the basic concepts and principles of the corpus and gradually shifting over to a question-driven approach. The intent behind this approach was to avoid overfitting the knowledge to the specifics of the sample questions available. This strategy met with mixed success: in the second phase, considerable reengineering of the knowledge was required to meet the requirements of the questions without compromising the strategy's generality. This was partly because the textbook adopted an example-based approach with somewhat varied depth, whereas the process of knowledge formation would have benefited from a more systematic and uniform coverage.

The SRI team's approach for knowledge formation was highly question-driven. Starting from the 50 sample questions, team members worked backwards to identify what pieces of knowledge would be needed to solve them. In-

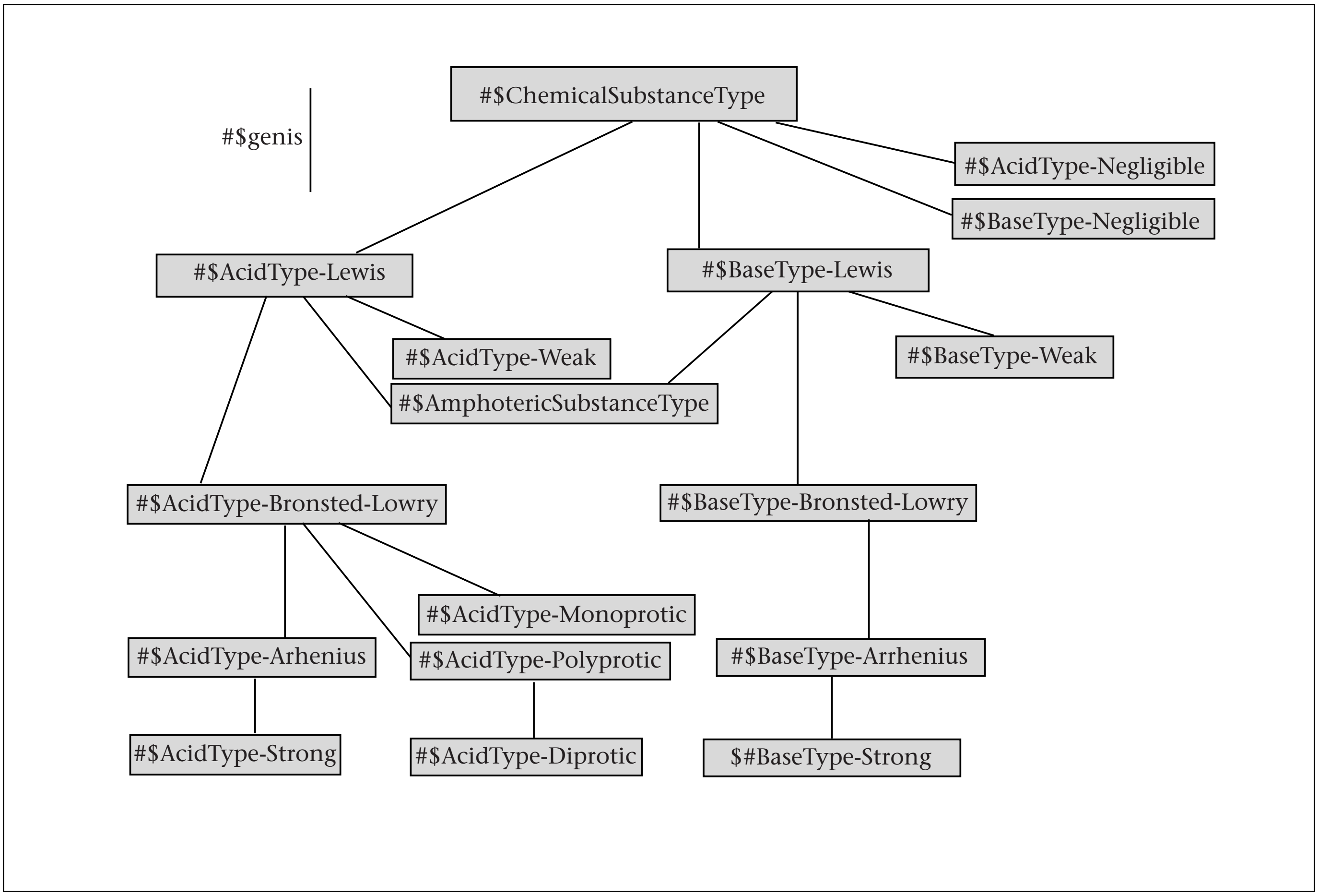


Figure 1. Extract from Cyc's Ontology of Acids and Bases.

These nodes represent second-order collections for organizing specific substance types; edges represent subsuption relationships.

terestingly, the initial set of questions was found to require coverage of a substantial portion of the syllabus. Once the coverage for the sample set of questions was achieved, they looked for additional sample questions from the available AP tests. Working with this additional set of sample questions, they ensured the robustness of their initial coverage.

Reliance on Domain-Independent Ontologies

Both Cycorp and SRI relied on their preexisting knowledge base content. Ontoprise started from scratch. Not surprisingly, the top-level classes in the Ontoprise knowledge base are chemistry concepts such as elements, mixtures, and reactions. Interestingly, the Ontoprise knowledge base did not draw on well-known ontological distinctions such as object type versus stuff type. We describe here in more detail how SRI and Cycorp leveraged their prior knowledge base and the issues that arose in doing so.

For several years the SRI team has been building a library of representations of generic entities, events, and roles (Barker, Porter, and Clark 2001) and they were able to reuse parts of this for the Project Halo pilot. In addition to providing the types of information commonly found in ontologies (class-subclass relations and instance-level predicates), their representations include sets of axioms for reasoning about instances of these classes. The portion of the ontology dealing with properties and values was especially useful for the Halo pilot. It includes representations for numerous dimensions (for example, capacity, density, duration, frequency, quantity) and values of three types: scalars, cardinals, and categoricals. This ontology also includes methods for converting among units of measurement (Novak 1995), which the SRI team's system used to align the representation of questions with representations of terms and laws, even if they are expressed with different units of measurement.

Cycorp publishes an open-source version of Cyc[3] that was used as a platform for the OpenHalo system. Cyc's knowledge consists of terms, relations, and assertions. The assertions are organized into a hierarchy of microtheories that permit the isolation of specific assumptions into a specific context. OpenHalo utilized OpenCyc's 6,000 concepts but was augmented for Project Halo with 1,000 new concepts and 8,000 existing concepts selected from the full Cyc knowledge base. A significant fraction of the latter formed part of the compositional explanation-generation system.

Reliance on Domain Experts

Cycorp and Ontoprise relied on their knowledge engineers to do all the knowledge formation, while SRI relied on a combined team of knowledge engineers and chemistry domain experts.

Team SRI used four chemists to help with the knowledge formation process, which was done in the following steps. First, ontological engineers designed representations for chemistry content, including the basic structure for terms and laws, chemical equations, reactions, and solutions. Second, chemists consolidated the domain knowledge into a 35-page compendium of terms and laws summarizing the relevant material from 70 pages of a textbook. While doing this, the chemists were asked to start from the premise to be proven and trace the reasoning in a backward-chaining manner to make it easy for knowledge engineers to encode this in the knowledge base. Third, knowledge engineers implemented that knowledge in the KM frame language, creating representations of about 150 laws and 65 terms. While doing so, they compiled a large suite of test cases for individual terms and laws as well as combinations of them. This test suite was run daily. Fourth, the "explanation engineer" augmented the representation of terms and laws to generate English explanations. Finally, the domain experts reviewed the output of the system for correctness and understandability.

Ontoprise knowledge engineers learned the domain and built the knowledge base—mostly starting with understanding and modeling the examples given in the textbook. They compiled a set of 41 domain concepts, 582 domain instances, 47 domain relations, and 345 axioms used for answering the questions. In addition, they added 138 rules in order to provide explanations for the answers produced.

Explanation Generation

The three teams took quite different approaches to explanation generation. These differences were based on the teams' available technologies (recall that the project allowed little time to develop new technologies), their longer-term goals, and their instincts of what might work.

The Ontoprise System. OntoNova, the Ontoprise system, was based on the representation language F(rame)-Logic (Kifer, Lausen, and Wu 1995) and the logic programming-based inferencing system OntoBroker (Angele et al. 2003). For answer justification, OntoNova used metainferencing, as follows. While processing a query OntoBroker produced a log file of the proof tree for any given answer. This proof tree, which was represented in F-Logic and contained the instantiated rules that were successfully applied to derive an answer, acted as input for a second inference run to produce English answer justifications.

We illustrate this approach with a sample question. The question asks for the Ka value of a substance, given its quantity in moles and its pH. The following is an extract from the log file of the proof tree:

```
a15106:Instantiation[ofRule->>kavalueMPhKa;
instantiatedVars->>{i(M,0.2),i(PH,3.0),…].
```

This log file extract states that the rule *kavalueMPhKa* was applied at the point in time logged here. Then, the variables *M* and *PH* were instantiated by 0.2 and 3.0 respectively. Rules important for justifying results, for example "kavalueMPhKa," were applied in the second, meta-inference run. Explanation rules were specified by their reference to an inference rule used to derive the answer, the instantiations of the variables of that rule, and a human-authored explanation template referring to those variables. These explanation rules resembled the explanation templates of the SRI system. The corresponding explanation rule for kavalueMPhKa was:

```
FORALL I,M1,PH1 explain(EX1,I) <-
I:Instantiation[ofRule->>kavalueMPhKa;
instantiatedVars->>{i(M,M1),i(PH,PH1)}] and
EX1 is ("The equation for calculating the acid-
dissociation.").
```

These explanation rules were applied to the proof tree for the example to produce the following justification output:

> The equation for calculating the acid-dissociation constant Ka for monoprotic acids is **Ka=[H+][A-]/[HA].** For monoprotic acids the concentrations for hydrogen [H+] and for the anion [A-] are the same: **[H+]=[A-].** Thus, we get **Ka= 0.0010 * 0.0010 / 0.2 = 5.0E-6** for a solution concentration **[HA] = 0.2 M.**
>
> The equation for calculating the pH-value is **ph=-log[H+].** Thus we get the pH-value, **ph = 3, H+ concentration [H+] = 0.0010.**

This two-step process for creating explanations allowed the application of OntoBroker to generate explanations. For OntoNova, the Onto-

```
ComputeConcentrationOfIons(C)
        if C is a strong electrolyte
                return(Max)                   [expl-tag-1]
        else
                …                             [expl-tag-2]

[expl-tag-1]
        entry: "If a solute is a strong electrolyte, the
               concentration of ions is maximal"
        exit:  "The concentration of ions in" C "is" Max
        dependencies:    electrolyte-status(C)
```

Figure 2. The Law for Computing the Concentration of Ions in a Chemical.

prise team developed an entire knowledge base for this purpose. Running short of time, they could not fully exploit the flexibility of this approach and thus mostly restricted it to an approach similar to template matching. In the future, however, they plan to apply additional inference rules to (1) integrate additional knowledge, (2) reduce redundancies of explanations, (3) abstract from fine-grained explanations, and (4) provide personalized explanations.

Team SRI's System. Team SRI's system was based on KM, a frame language with some similarities to KRL and KL-ONE systems.[4] During reasoning, KM records which rules are used in the derivation of ground facts. These proof tree fragments could be presented as an "explanation" of the derivation of a fact. Experience with expert systems, however, has taught us that proof trees and inference traces are not comprehensible explanations for most users.

To provide better explanations, KM allows the knowledge engineer to supply explanation templates for each knowledge base rule. These explanation templates provide control over which proof tree fragments are presented in the explanation and what English text is used to describe them. In particular, the knowledge engineer specifies what text to display when a rule is invoked ("entry text"), what text is displayed when the rule has been successfully applied ("exit text"), and a list of any other facts that should be explained in support of the current rule (dependent facts). The three parts of the explanation template can contain arbitrary KM expressions, allowing the knowledge engineer considerable control over explanation generation.

Consider the law for computing the concentration of ions in a chemical solution, shown in figure 2. When the explanation of a rule application is requested, the entry text is formed and displayed, followed by the nested explanation of dependent facts, followed by the exit text. The explanation generated for the computation of concentration of ions in NaOH is as follows:

> If a solute is a strong electrolyte, the concentration of ions is maximal
> Checking the electrolyte status of NaOH.
>   Strong acids and bases are strong electrolytes.
>   NaOH is a strong base and is therefore a strong electrolyte.
>   NaOH is thus a strong electrolyte.
> The concentration of ions in NaOH is 1.00 molar.

A more complete description of team SRI's system can be found in Barker et al. (2004).

The Cycorp System. All the systems generated their explanations by appropriate filtering and transformation of the inference proof tree. The primary difference in Cycorp's approach was that Cyc was already capable of providing natural language explanations for any detail, however minor, whereas the other systems required the addition of template responses for each rule and fact deemed important. The result of this was that much of the effort expended on explanations by Cycorp concerned judicious strengthening of the filters, and Cyc's output consequently erred on the side of verbosity. Moreover, Cyc's English, being built up compositionally by automatic techniques rather than being handcrafted for a specific project, exhibits a certain clumsiness of expression.

A specific example of where Cyc had a lot of trouble generating readable explanations was when the use of a mathematical equation required a lot of arithmetic. While a step-by-step exposition of every mathematical step involved is technically correct, it makes most readers recoil in horror. Cyc's lower scores for explanations may therefore be ascribed not to any errors contained within, nor to the complete absence of an explanation, but to the fact that the key chemistry principles involved tended to be buried amid more trivial reasoning. In the Halo pilot challenge, the graders appeared to value conciseness of expression over either correctness or completeness.

In the Halo pilot challenge, Cyc produced explanations for every question it answered correctly, and it was rare for any of the graders to find any fault with the explanations' correctness. Comments like "calculations are correct but once again buried" and "not well focused" were common. It was clear at the end of the pilot phase that Cyc required significant work in explanation filtering; substantial progress has been made during subsequent projects.

Evaluation

At the end of four months, knowledge formulation was stopped, even though the teams had not completed the task. All three systems were sequestered on identical servers at Vulcan. Then the challenge exam, consisting of 100 novel AP-style English questions, was released to the teams. The exam consisted of three sections: 50 multiple choice questions and two sets of 25 multipart questions—the detailed answer and free-form sections. The detailed answer section consisted mainly of quantitative questions requiring a "fill in the blank" (with explanation) or short essay response. The free-form section consisted of qualitative, comprehension questions, which exercised additional reasoning tasks such as metareasoning, and relied more, if only in a limited way, on commonsense knowledge and reasoning.

Due to the limited scope of the pilot, there was no requirement that questions be input in their original, natural language form. Thus, two weeks were allocated to the teams for the translation of the exam questions into their respective formal languages. Upon completion of the encoding effort, the formal question encodings of each team were evaluated by a programwide committee to guarantee high fidelity to the original English. The criterion of fidelity was as follows:

> Assume that a student was fluent in both English and the formal language in question. If she is able to infer additional facts from the formal encodings either through omission of detail or because new material details were provided that were not available in the English description of the question, then a fidelity violation had occurred.

Once the encodings were evaluated, Vulcan personnel submitted them to the sequestered systems. The evaluations ran in batch mode. The Ontoprise system completed its processing in 2 hours, the SRI system in 5 hours and the Cycorp system in a little over 12 hours. Each of the three systems produced an output file in accordance with a predefined specification. For each question, the format required the specification of the question number; the full English text of the question; a clear answer, either in prose or letter form for multiple choice questions; and an explanation of how the answer was derived—even for multiple choice questions. See the sidebar "Examples of System Outputs and Grader Comments" for more details.

Vulcan engaged three chemistry professors to evaluate the exams. Adopting an AP-style evaluation methodology, they graded each question for both correctness and the quality of the explanation. The exam encompassed 168 distinct gradable components consisting of questions and question subparts. Each of these received marks, ranging from 0 to 1 point each for correctness and separately for explanation quality, for a maximum high score of 336. All three experts graded all three exams. The scoring of all three chemistry experts was aggregated for a maximum high score of 1008.

Sodium azide is used in air bags to rapidly produce gas to inflate the bag. The products of the decomposition reaction are:

(a) Na and water;
(b) Ammonia and sodium metal;
(c) N2 and O2;
(d) Sodium and nitrogen gas;
(e) Sodium oxide and nitrogen gas.

Figure 3. An Example of a Multiple Choice Section Question, MC3.

Empirical Results

Vulcan was able to run all of the applications during the challenge despite minor problems associated with each of the three systems.[5] Results were compiled for the three exam sections separately and then aggregated to form the total scores. Despite significant differences in approach, all three systems performed remarkably well, above 40 percent for correctness for most of the graders—a score comparable to an AP-3 (out of 5)—close to the mean human score of AP-2.82!

The multiple choice (MC) section consisted of 50 questions, MC1 through MC50. Each of these questions featured five choices, lettered "a" through "e." The evaluation required both an answer and a justification for full credit, even for MC questions. Figure 3 provides an example of one of the multiple choice questions, MC3.

Figure 4 depicts the correctness (on the left) and answer-justification (on the right) scores for the multiple choice section as a percentage of the 50-point maximum. The Cycorp, Ontoprise, and SRI scores are depicted by the different gray bars. Bars are grouped by the grading chemistry professors, SME1 through SME3, where SME stands for subject matter expert. SRI and Ontoprise both scored about 70 percent correct in this section, while Cycorp scored slightly above 50 percent. Cycorp applied a metareasoning technique to evaluate multiple choice questions. First, Cycorp's OpenHalo attempted to find a correct answer among the

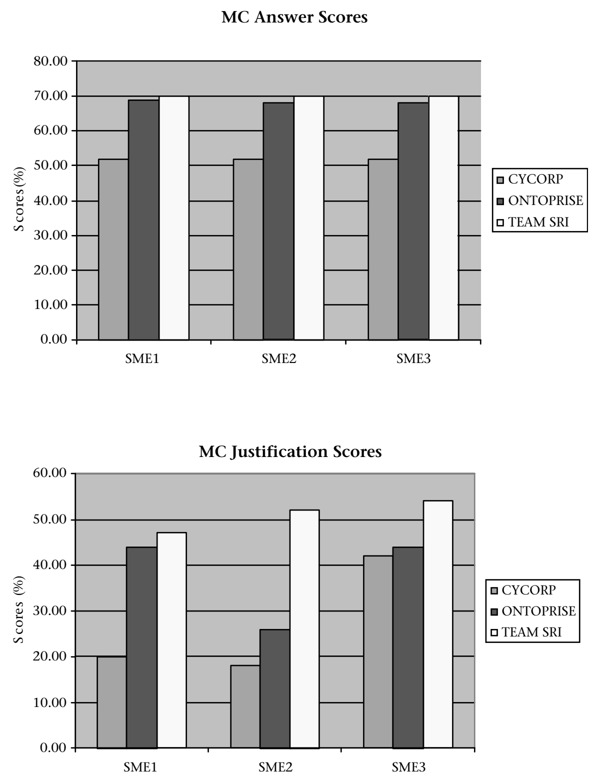


Figure 4. Correctness and Answer-Justification Scores for the Multiple Choice Section as a Percentage of the Maximum Score of 50 Points.

Balance the following reactions, and indicate whether they are examples of combustion, decomposition, or combination

(a)$C_4H_{10} + O_2 \rightarrow CO_2 + H_2O$

(b)$KClO_3 \rightarrow KCl + O_2$

(c)$CH_3CH_2OH + O_2 \rightarrow CO_2 + H_2O$

(d)$P_4 + O_2 \rightarrow P_2O_5$

(e)$N_2O_5 + H_2O \rightarrow HNO_3$

Figure 5. An Example of a Detailed Answer Section Question, DA1.

five. If it failed to do so, it would attempt to determine which of the five options were provably wrong. This led to some questions returning more than one letter answer, none of which received credit from the subject matter experts. In contrast, the other two teams hard-coded the approach to be used—direct proof versus elimination of obvious wrong answers—and appeared to fare better.

The answer-justification scores were all considerably lower than the correctness scores. Note that these two measurements were not independent. Systems that were unable to produce an answer did not produce a justification, and systems that produced incorrect answers were rarely able to produce convincing answer justifications. The answer justification scores were also far less uniform than the correctness scores,[6] with the scoring for SRI appearing to be the most consistent across the three evaluators. All the evaluators found the SRI justifications to be the best, while the Cycorp generative-English was the least comprehensible to the subject matter experts.

The detailed answer (DA) section had 25 multipart essay questions, DA1–DA25, representing a total of 80 gradable answer components. Figure 5 depicts an example of a DA section question, DA1. Figure 6 depicts the correctness and answer-justification scores for the DA section. The correctness assessment shows a slight advantage to the Cycorp system in this section. OpenHalo may have fared better here because it was not penalized by its multiple choice strategy in this section.

The free-form (FF) section also had 25 multipart essay questions, FF1–FF25, representing 38 gradable answer components. Figure 7 depicts an example of a FF question, FF2. Figure 8 shows the correctness and answer-justification scores for the FF section. This section was designed to include questions that were somewhat beyond the scope of the defined syllabus. Some required metareasoning and, in some cases, limited commonsense knowledge. The objective was to see how well the systems performed faced with such challenges and whether the additional knowledge constructs available to SRI and Cycorp would translate into better results. The outcome of this section showed a marked advantage to the SRI system, both for correctness and for justification. We were surprised that the Cycorp system did not do better, given its many thousands of concepts and relations and the rich expressivity of CycL. This result may reflect the inability of their knowledge engineering team to leverage knowledge in Cyc for this particular challenge.

Figure 9 provides the total challenge results, as percentages of the 168-point maximum scores, for answer correctness and justifications. The correctness scores show a similar trend for the three subject matter experts, with team SRI slightly outperforming Ontoprise and Ontoprise slightly outperforming Cycorp. By contrast, the justification scores display a significant amount of variability. We are considering changes in our methodology to address this issue, including training future subject matter experts to produce more consistent scoring.

All subject matter experts found some answer justifications that they liked. The subject matter experts provided high-level comments, mostly focused on the organization and conciseness of the justifications. In some instances, justifications were quite long. For example, Cycorp's generative English produced some justifications in excess of 16 pages in length. The subject matter experts also complained that many arguments were used repetitively and that proofs took a long time to "get to the point." In some multiple choice questions, proofs involved invalidating all wrong answers rather than proving the right one. All the teams appeared to rely on instance-based solutions to prove generalized comprehension-oriented questions, indicating a limited ability to reason with concepts. Gaps in knowledge coverage were also evident. For example, many of the teams had significant gaps in their knowledge of net ionic equations. Detailed question-by-question scores are available on the project Web site.

Problematic Questions

Despite the impressive overall performance of the three systems, there were questions on which each of them failed. Most interestingly, there were questions on which all three systems failed dramatically. Five prominent and interesting cases—DA10, DA22, FF1, FF8, and FF22—are shown in figure 10. We examine these questions more closely.

The first issue to address is whether these questions share properties that explain their difficulty. An initial hypothesis is that all five questions require that a system be able to represent and reason about its own problem-solving procedures and data structures—that is, that it be reflective or capable of metarepresentation and metareasoning. That property would explain the difficulty all three systems had, at least to the extent that the systems can be said to lack such reflective capabilities.

DA10 seems to probe a system's strategy for solving any of a general class of problems; indeed, it seems to ask for an explicit description of that strategy. DA22 implies that the pH calculation that a problem solver is likely to use will generate an unacceptable result in this case (a value of greater than 7 for an acid) and then asks for an explanation of what went wrong, that is, of why the normal pH calculation leads to an anomalous result here. These two questions both seem to require the system to represent and reason about, indeed to explain the workings of, its own problem-solving procedures.

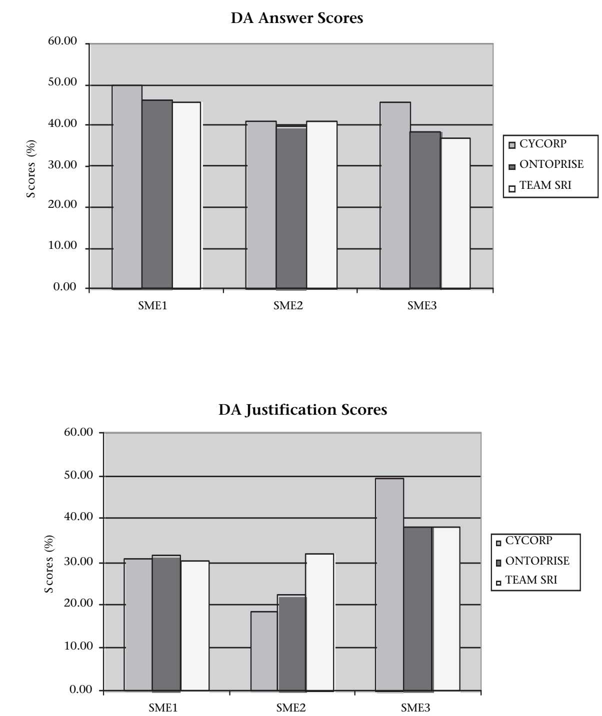


Figure 6. Correctness and Answer-Justification Scores for the Detailed Answer Section as a Percentage of the Maximum Score of 80 Points.

Pure water is a poor conductor of electricity, yet ordinary tap water is a good conductor. Account for this difference.

Figure 7. An Example of a Free-Form Section Question, FF2.

FF22 seems similar in that it asks about the applicability of approximate solutions for a certain class of problems and about the reasons for the limits to that applicability. On reflection, though, it is really probing the system's knowledge of certain methodological principles used in chemistry rather than the system's knowledge of its own inner workings. What seems to be missing is knowledge about chemistry—not about chemical compounds but rather about methods used in chemistry, in particular about approximate methods and their scope and limits. And, of course, these latter methods may or may not be built into the system's own problem-solving routines.

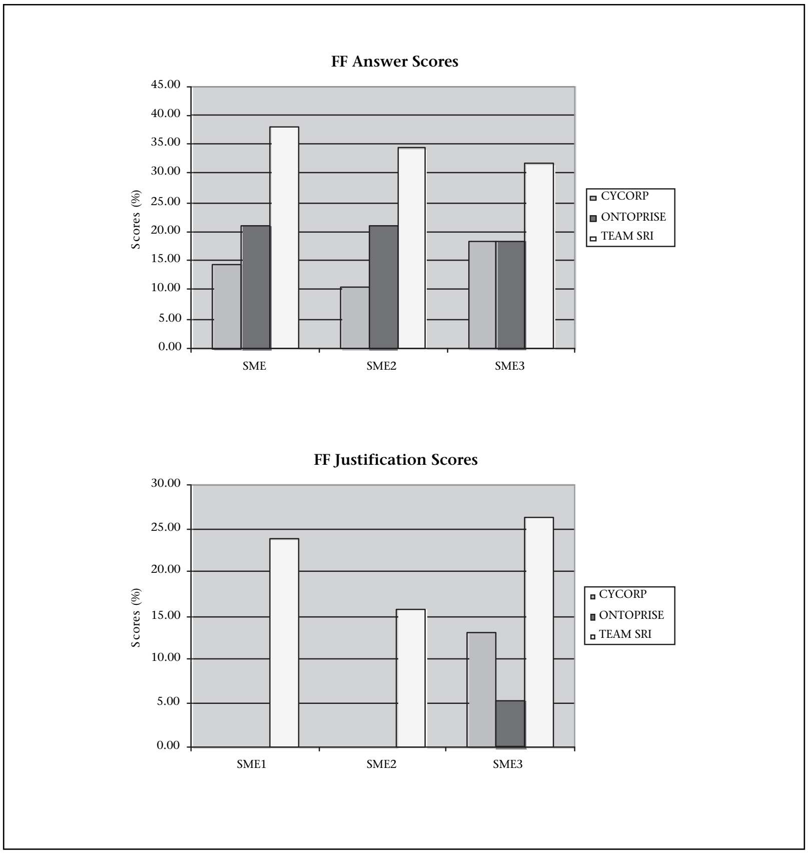


Figure 8. Correctness and Answer Justification Scores for the Free-Form Section as a Percentage of the Maximum Score of 38 Points.

Note that SRI fared significantly better both in the correctness and justification scoring.

FF1 and FF8 are similar in that one asks for similarities and the other for differences, and in both cases, the systems did represent the knowledge but did not support the reasoning method to compute them.

FF1 is a question about the language of chemistry, in particular about the abstract syntax or type conventions of terms for chemical compounds. All three systems had some knowledge encoded so that the differences could be computed but lacked the necessary reasoning method to compute them.

A Note on Performance

The Halo pilot challenge was run over the course of a day and a half on sequestered systems at Vulcan, by Vulcan personnel. As noted above, we did encounter minor problems with all three systems that were resolved over this period of time. Among other issues, the batch files containing the formal encodings of the challenge questions needed to be broken into two to facilitate their processing on all three systems. The Ontoprise system proved to be the fastest and most reliable, taking approximately 2 hours to complete its batch run. The SRI system ran the challenge in approximately 5 hours, and the Cycorp system completed its processing in over 12 hours. In this latter case, a memory leak on the sequestered platform caused the server to crash, and the system was rebooted and run until the evaluation time limit expired.

All three teams undertook modifications and improvements to the sequestered systems and ran the challenges again. In this case the Ontoprise system was able to complete the challenge in 9 minutes, the SRI system took 30 minutes, and the Cycorp system took approximately 27 hours to process the challenge. Both the sequestered and the improved systems are freely available for download off the Project Halo Web site.

Analysis

The three systems did well—better than Vulcan expected. Nevertheless, their performance was far from perfect, and the goal of the pilot project was to go beyond evaluations of KR&R systems to an analysis of them. Therefore, we wanted to understand why these systems failed when they did, the relative frequency of each type of failure, and the ways these failures might be avoided or mitigated.

Based on our collective experience building KR&R systems, at the beginning of Project Halo we designed a taxonomy of failures that fielded systems might experience. Then, at the end of the project, we analyzed every point lost on the evaluation in an attempt to identify the failure and place it within the taxonomy. We studied the resulting data to draw lessons about the taxonomy, the systems, and (by extrapolation) the current state of KR&R technologies for building fielded systems. See Friedland et al. (2004) for a comprehensive report of this study.

In particular, our failure analysis suggests three broad lessons that can be drawn across the board for the three systems: modeling, answer justification, and scalability for speed and reuse.

Modeling. A common theme in the modeling problems across the systems was that the incorrect knowledge was represented, or some domain assumption was not adequately factored-in, or the knowledge was not captured at the right level of abstraction. Addressing these problems requires us to have direct involvement of the domain experts in the knowledge-engineering process. The teams involved such experts to different extents and at different times during the course of the project. The SRI team, which involved professional chemists from the beginning of the project, appeared to benefit substantially. This presents a research challenge, since it suggests that the expositions of chemistry in current texts are not sufficient for building or training knowledge-based systems. Instead, a high-level domain expert must be involved in formulating the knowledge appropriately for system use. Two approaches to ameliorating this problem being pursued by participants are (1) providing tools that support direct manipulation and testing of KR&R systems by such experts, and (2) providing the background knowledge required by a system to make appropriate use of specialized knowledge as it is presented in texts.

Answer Justification. Explanation, or, more generally, response interpretability, is fundamental to the acceptance of a knowledge-based system, yet for all three state-of-the-art systems, it proved to be a substantial challenge. Since the utility of the system will be evaluated end to end, it is to a large degree immaterial whether its answers are correct, if they cannot be understood. Reaching the goals of projects

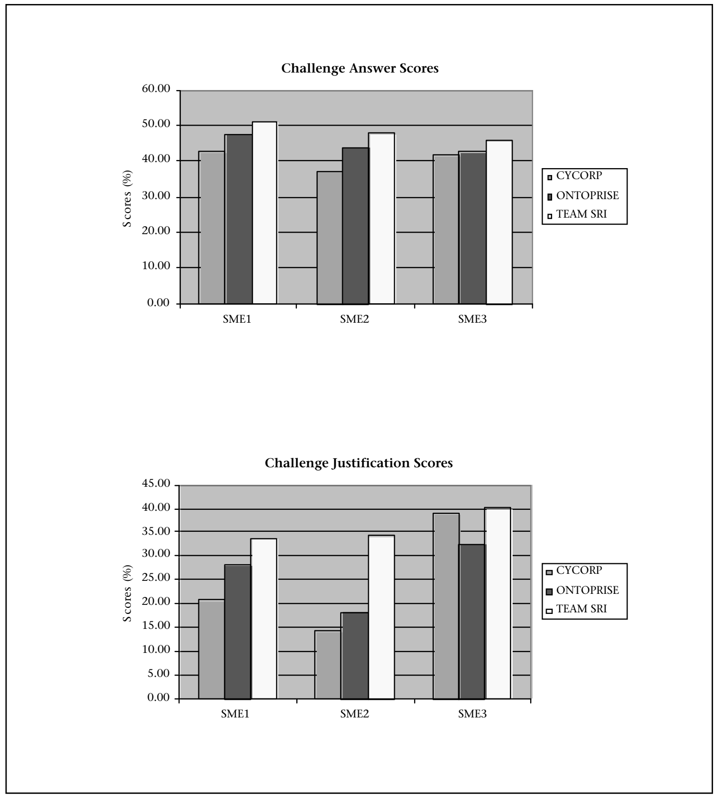


Figure 9. Total Correctness and Justification Scores as a Percentage of the Maximum Score of 168 Points.

DA10: HCl, H2SO4, HClO4, and HNO3 are all examples of strong acids and are 100% ionized in water. This is known as the "leveling effect" of the solvent. Explain how you would establish the relative strengths of these acids. That is, how would you answer a question such as "which of these acids is the strongest?"

DA22. Phenol, C6H5OH, is a very weak acid with an acid equilibrium constant of Ka = 1.3 x 10-10. Determine the pH of a very dilute, 1 x 10-5 M, solution of phenol. Is the value acceptable? If not, give a possible explanation for the unreasonable pH value.

FF1. What is the difference between the subscript 3 in HNO_3 and a coefficient 3 in front of HNO_3?

FF8. Although nitric acid and phosphoric acid have very different properties as pure substances, their aqueous solutions possess many common properties. List some general properties of these solutions and explain their common behavior in terms of the species present.

FF22. When we solve equilibrium expressions for the [H3O+] approximations are often made to reduce the complexity of the equation thus making it easier to solve. Why can we make these approximations? Would these approximations ever lead to significant errors in the answer? If so give an example of an equilibrium problem that would require use of the quadratic equation.

Figure 10. Examples of Chemistry Questions That Proved to Be Problematic for All Three Teams.

A Brief History of Evaluating Knowledge Systems

One unique aspect of the Halo pilot is its rigorous scheme of evaluation. It uses an independently defined and well-understood test, specifically, the advanced placement test for chemistry, on a well-defined scope, specifically, 70 pages of a chemistry textbook. Though such rigorous evaluation schemes have been developed in the areas of shallow information extraction (the MUC conferences) information retrieval and simple question answering (the TREC conferences) for quite a while, the corresponding task of evaluating the kind of knowledge-based systems deployed in the Halo Pilot had appeared to be too difficult to be approached in one step.

Thus, previous efforts at measuring the performance of knowledge-based systems such as in high-performance knowledge bases (HPKB) and rapid knowledge formation (RKF) constituted important stepping stones towards rigorous evaluation of knowledge-based systems, but the Halo pilot represents a significant advance. To substantiate this summary, we shall review some of the details of developments in these various areas.

Retrieving Answers from Texts

Question-answering via information retrieval and extraction from texts has been an active area of research, with a progression of annual competitions and conferences, especially the 7 message-understanding conferences (MUCs) and the 12 text-retrieval conferences (TRECs) from 1992–2003, sponsored by NIST, IAD, DARPA, and ARDA. TRECs were initially aimed at retrieving relevant texts from large collections and then at extracting relevant passages from texts (Voorheese 2003). The earlier systems had virtually no need for inference-capable knowledge bases and reasoning capabilities. In recent years the question-answering tasks have become more challenging, for example, requiring a direct answer to a question rather than a passage containing the answer. The evaluation schemes are very well defined, including well worked out definitions of the tasks and answer keys that are used to compute evaluation measures including precision and recall.

Recently there has been a surge of interest in the use of domain knowledge in question answering (for example, see Chaudhri and Fikes [1999]). ARDA's current advanced question and answering for intelligence program (AQUAINT), started in 2001, is pushing text-based question-answering technology further, seeking to address a typical intelligence-gathering scenario in which multiple, interrelated questions are used to fulfill an overall information need rather than answer just single, isolated, fact-based questions. AQUAINT has both adopted TREC's approach to the evaluation of question-answering and tried to extend it to encompass more complex question types, for example biographical questions of the form "Tell me all the important things you know about Osama bin Laden." The fundamental difference between the Halo evaluation and the AQUAINT evaluation is that the AQUAINT evaluations are designed to test the question-answering capability on huge bodies of text on widely ranging subjects using very limited reasoning capabilities. In contrast, the Halo evaluation is focused on evaluating deep reasoning in the field of sciences. The eventual goal of Halo is to do significant coverage of sciences, but the current phase was limited to only 70 pages of a chemistry textbook.

Building and Running Knowledge-Based Systems

In the area of knowledge-based systems, DARPA, AFOSR, NRI, and NSF jointly funded the knowledge sharing effort in 1991 (Neches et al. 1991). This was a three-year collaborative program to develop "knowledge sharing" technologies to facilitate the exchange and reuse of inference-capable knowledge bases among different groups. The aim was to help reduce costs and promote development of knowledge-based applications. This was followed by DARPA's High Performance Knowledge Base (HPKB) program (1996–2000), designed to push knowledge-based technology further and demonstrate that very large (100k+ axiom) systems could be built quickly and be usefully applied to question-answering tasks (Cohen et al. 1998). The evaluation in HPKB was aimed simply at the hypothesis that large knowledge-based systems can be built at all, that they can accomplish interesting tasks, and that they do not break—as a toy system would and as many of the initial knowledge-based systems did—when working with a realistically sized knowledge base.

Evaluating Knowledge-Based Systems

There have been few efforts so far at documenting and analyzing the quality of fielded KR&R systems (Brachman et al. 1999; Keyes 1989; and Batanov and Brezillon 1996). RKF made significant efforts to analyze and document the quality of knowledge base performance (Pool et al. 2003). Specifically, an evaluation in DARPA's Rapid Knowledge Formation project, which was roughly comparable to the one used in Project Halo, was based on approximately 10 pages from a biology textbook and a set of test questions—however, this was not an independently established test. The Halo pilot, reported on here, improves upon these evaluations by being more systematic and usable for cross-system comparisons. The Halo pilot has adopted an evaluation standard that is comparable to the rigor of the challenges for retrieving answers from texts. It provides an exact definition of the scope of the domain—an AP chemistry test setting that has proven its validity in many years with many students—as well as an objective evaluation by independent graders. We conjecture that the Halo evaluation scheme is extensible enough to support a coherent long-term development program.

like Digital Aristotle will require an investment of considerably more resources into this aspect of systems to realize robust gains in their competence. Constructing explanations directly from the system's proof strategy is neither straightforward nor particularly successful, especially if that strategy has not been designed with explanation in mind. One alternative is to use explicit representations of problem-solving methods (PSMs) so that explanations can include statements of problem-solving strategy as well as statements of facts and rules (Clancey 1983). Another is to perform more metareasoning over the proof tree to construct a more readable explanation.

Scalability for Speed and Reuse. There has been substantial work in the literature on the trade-off between expressiveness and tractability, yet managing this trade-off, or even predicting its effect in the design of fielded systems over real domains, is still not at all straightforward. To move from a theoretical to an engineering model of scalability, the KR community would benefit from a more systematic exploration of this area driven by the empirical requirements of problems at a wide range of scales. For example, the three Halo systems, and more generally, the Halo development and testing corpora, can provide an excellent test bed to enable KR&R researchers to pursue experimental research in the trade-off between expressiveness and tractability.

Discussion

All three logical languages, KM, F-Logic, and CycL, were expressive enough to represent most of the knowledge in this domain. F-Logic was by far the most concise and easy to read, with syntax most resembling an object-oriented language. F-Logic also yielded very high-fidelity representations that appear to be easier and more intuitive to construct. Ontoprise was the only team to conduct a sensitivity study of the impact of different question encodings on system performance. In the case of the two questions they examined, their system produced similar answers with slightly different justifications. For the most part, the encoding process and its impact on question-answering stability remain an open research topic.

SRI and Ontoprise yielded comparably sized knowledge bases. OntoNova was built from scratch using no predefined primitives, while SRI's system leveraged the Component Library, though not as extensively as the team had initially hoped. SRI's use of professional chemists in the knowledge-formulation process was a huge advantage, and the quality of their outcome is reflected by this fact. The other teams have conceded that, if they had the opportunity to revisit the challenge, they would adopt the use of subject matter experts in knowledge formation. Cycorp's OpenHalo knowledge base was two orders of magnitude larger than the other teams'. They were unable to demonstrate any measurable advantage in using this additional knowledge, even in example-based questions, where they exhibited metareasoning brittleness similar to that observed in the other systems. The size of Cycorp's knowledge base does, however, explain some of the significant run-time differences. They have also yet to demonstrate successful, effective reintegration of Halo knowledge into the extended Cyc platform. Reuse and integration appear to remain open questions for all three Halo teams.

The most novel aspect of the Halo pilot was the great emphasis put on answer explanations, which served two primary purposes: first, to exhibit and thereby verify that deep reasoning was occurring, and second, to validate that appropriate domain explanations can be generated. This is an area that is still open to significant improvement. SRI's approach produced the best-quality results, but it leaves open many questions regarding how well it might be scaled, generalized, and reused. Cycorp's generative approach may eventually scale and generalize, but the current results were extremely verbose and often unintelligible to domain experts. Ontoprise's approach of running a second inference process appears to be very promising in the near term.

Vulcan Inc. and the pilot participants have invested considerable efforts in promoting the scientific transparency of the Halo pilot. The project Web site provides all the scientifically relevant documentation and tutorials, including an interactive results browser and fully documented downloads representing both the sequestered systems and improved Halo pilot chemistry knowledge bases. We eagerly anticipate comment from the AI community and look forward to its use by universities and other researchers.

Finally, the issue of cost must be considered. We estimate that the per page expense for each of the three Halo teams was on the order of \$10,000 per page for the 70-page syllabus. This cost must be significantly reduced before this technology can be considered viable for Digital Aristotle.

In summary, all of the Halo systems scored well on a very difficult challenge: extrapolating the results of team SRI's system on the limited 70-page syllabus to the entire AP syllabus yielded the equivalent of an AP-3 score for answer correctness—good enough to earn course cred-

Examples of System Outputs and Grader Comments

The Halo pilot evaluation was intended to assess deep-reasoning capabilities of the three competing systems in the context of a well-known evaluation methodology. Seventy pages of the advance placement (AP) chemistry syllabus were selected. Systems were required to produce coherent answers and answer justifications in English to each of the 100 AP-style questions posed. In the case of multiple choice questions, a letter response was required. The evaluation team consisted of three chemistry professors, who were instructed to grade the exams using AP guidelines. Answer justifications were thus required to conform to AP guidelines to receive full credit.

System outputs were required to conform to strict formats. The question number needed to be clearly indicated, followed by the original English text of the question. This was to be followed by the answer. Multiple choice questions required a letter answer. Finally, the answer justification was required. Justification guidelines required that the answers be clear, concise, and appropriate for AP exams.

The system outputs were rendered into hardcopy and distributed to three chemistry professors for evaluation. These subject matter experts were asked to apply AP grading guidelines to assess the system outputs and to provide lots of written comments.

Question examples 1 through 3 present Ontoprise, SRI, and Cycorp system outputs, respectively. Examples 1 and 2 depict responses to multiple choice question 20, while example 3 depicts the response to multiple choice question 34. These figures also contain graders' remarks.

Question 20

Original question

The pH of a 1.0M solution of HCl is:

a. 1.0
b. 0.1
c. 0.0
d. less than zero
e. between 0 and 1

Answer

C

Justification

- The equation for calculating the ph–value is **ph=–log[H+]**. Thus we get ph–value **ph = 0, H+ concentration [H+] = 1.**
- In the given mixture **the concentration of H3O–molecules(atoms) is 1 M.**
    - ◊ **HCl is a strong monoprotic acid** that completely splits into H+ and Cl–. A 1–molar mixture of HCl has therefore a 1–molar concentration of H+ and Cl– molecules.

This is a logical, well ordered presentation.

Question Example 1. Ontoprise's OntoNova Application's Output for Multiple Choice Question 20.

Note the output's format: the question number is indicated at the top; followed by the full English text of the original question; next, the letter answer is indicated; finally, the answer justification is presented. The graders written remarks are included. OntoNova employed a second inference step to derive the answer justification, using human-authored templates, the proof tree, and combination rules to assemble the English text.

```
20. The pH of a 1.0M solution of HCl is:
(a) 1.0
(b) 0.1
(c) 0.0
(d) less than zero
(e) between 0 and 1
```

Answer

The output of is/are:

Good, brief presentation in logical order.

- (c) 0.0
- What is the pH of a 1.0M solution of HCl?.
    - The pH of a chemical depends on the concentration of H+.
        - The pH can be computed as -log10[H+].
            - For strong acids, the concentration of H+ is equal to the concentration of acid.
                - The equilibrium concentration of strong acid reactants in equilibrium reactions is zero.
                - The equilibrium concentration of HCl is thus 0 molar.
            - HCl is a strong acid, so [H+] = 1.00 molar.
        - The pH of a solution of HCl is therefore -log10(1.00) = 0.00e+0.
    - The pH of HCl strong-electrolyte-solution is therefore 0.00e+0.
- The pH of a 1.0M solution of HCl is: (c) 0.0.

Question Example 2. SRI's SHAKEN Output.

Human-authored templates associated with chemical methods were combined during the back-chaining process to produce the English text. The templates specify the salient subgoals to be elaborated as indented "sub explanations." This resulted in generally superior answer justifications.

MC34: The acid ionization constant for benzoic acid, C6H5COOH, is 6.3E-5. The conjugate base of this acid has an ionization constant of: (a) 6.3E-5; (b) 1E-14; (c) 1.59E-10; (d) 6.3E-19; (e) 6.3E-9.

Answer:

C

Reformulation of Question:

If

- the acid-dissociation constant for benzoic acid is $6.3 * 10^{-5}$
- and benzoic acid and some *BASE* form a conjugate acid-base pair,

what values of *ANSWER* are there such that there is some *KA* such that

- the base-dissociation constant for *BASE* is *KA*
- and *ANSWER* is a good choice among these sentences:
  - A. $KA \approx 6.3 * 10^{-5}$
  - B. $KA \approx 1 * 10^{-14}$
  - C. $KA \approx 1.59 * 10^{-10}$
  - D. $KA \approx 6.3 * 10^{-19}$
  - E. $KA \approx 6.3 * 10^{-9}$

Reasoning:

Answer C is the right answer because:

Given (from the question):
The acid-dissociation constant for benzoic acid is 6.3E-5.
Benzoic acid and *BASE* form a conjugate acid-base pair.

Applicable Rule:
If

- ***ACID* and *BASE* form a conjugate acid-base pair**
- **and the acid-dissociation constant for *ACID* is *KA*,**

then the base-dissociation constant for *BASE* is the ratio of Kw to *KA*. Good identification of important relationship, required to answer question.
-- *from Section 16.8 of Chemistry: The Central Science*

Rule Application:
The ratio of Kw to 6.3E-5 = 1.5873E-10. ✓

Conclusion:
The base-dissociation constant for *BASE* is 1.5873E-10.

Trivially: 1.5873E-10 ≈ 1.59E-10.

Question Example 3. The Output of Cycorp's OpenHalo Application for Multiple Choice Question 34.

Note the grader's remarks. OpenHalo used the proof tree and Cyc's generative English capabilities to produce the English answer. This example illustrates one of the better outcomes—some questions produced many pages of generative English, which were far less intelligible to the graders.

Comments on SRI Output
Very good, brief presentation in logical order
Good use of selective elimination
This is a good first effort, but still looks a little like what would be expected from a student taking this exam.
Generally, when a calculation was required, the program did not follow what I would expect from a student: namely, a setup, substituted numbers, followed by a solution to the problem.
Comments on Ontoprise Output
Good logical flow, set up with substitutions shown
Well done, and to the point
The logic is more readable (and shorter!) than the KB justifications, but it often seems to be presented backwards — arguing from the answer as opposed to arguing to the answer.
There was a common error in the use of significant figures. Answers are often given in as many figures as the program could generate. This is also a common problem with students so I guess we could claim that the computer is being more human-like in these responses
Comments on Cycorp Output
Main strength of the program is that it did a fairly good job of arriving at the correct answers
A good approach for replacing the "reasoning" used in this program is to use "Show Method of Calculation" (for all questions which involve the calculation of a numerical answer) and "Explain Reasoning" (for all questions which do not involve a calculated answer)
In general, the program appears to take a quantitative approach when answering questions and does not know how to take a qualitative approach. For example, when the activity series was part of a question, the program would use cell potentials
The "reasoning" are much, much, much too long. A sufficient "reasoning" to any of the questions on the exam never requires more than the 1/2 page.

Comments on Output.

This figure shows a small, if representative, subset of verbatim comments that subject matter experts made on the answers to the questions produced by the system. The subject matter experts had well-defined expectations of system behavior, for example setting up a problem before actually presenting its solution or the number of significant digits used in the output. These comments did not reflect, for the most part, on the correctness of the computation, but rather were indicators of "how things were to be done on an AP chemistry exam." This highlights the importance of understanding domain-specific requirements in answer and answer-justification formation and generation by question-answering systems.

it at many top universities. The Halo teams believe that with additional, limited effort they would be able to improve the scores to the AP-4 level and beyond. Vulcan Inc. has developed two additional challenge question sets to validate these claims at a future date.

Conclusions and Next Steps

As we noted at the beginning of this article, Project Halo is a multistaged effort. In the foregoing, we have described phase one, which assessed the capability of knowledge-based systems to answer a wide variety of unanticipated questions with coherent explanations. Phase two of Project Halo will examine whether tools can be built to enable domain experts to build such with an ever-decreasing reliance on knowledge engineers—a goal that was pursued in DARPA's Rapid Knowledge Formation project. Empowering domain experts to build robust knowledge bases with little or no assistance from knowledge engineers will (1) dramatically decrease the cost of knowledge formulation; (2) greatly reduce the type of errors observed in the Halo Pilot that resulted from lack of understanding of the domain on the part of knowledge engineers; and (3) facilitate a growing, peer-reviewed body of machine-processable knowledge that will form the basis for Digital Aristotle. A critical measure of success is the degree to which the relevant scientific communities are willing to adopt these tools, especially in their pedagogies.

At the core of the knowledge formulation approach envisioned in phase two is a document-rooted methodology in which the domain expert uses an existing document such as a textbook as the basis for the formulation of a knowledge module. Tying knowledge modules to documents in this way will help determine the scope and context of each module, the types of questions they can be expected to answer, and the appropriate depth and resolution of the answers. The 30-month phase-two effort will be undertaken in three stages. First, a six-month, analysis-driven design process will examine the complete AP syllabi for chemistry, biology, and physics (B). The objective of this analysis will be to determine requirements on effective use by domain experts of a range of knowledge-acquisition technologies. The results of this study should allow us to (1) determine the gaps in "coverage" of current state-of-the-art knowledge-acquisition techniques and define targeted research to fill those gaps; (2) understand and prioritize the methods, techniques, and technologies that will be central to the Halo 2 application; (3) understand which portions of the syllabi would be best suited to demonstrate the broadest proof of concept, given our resources, and better understand the ways in which the content requirements of the different disciplines can be mutually leveraged; and finally, (4) produce a coherent, well motivated design.

A 15-month implementation stage will follow. Here, the detailed designs will be rendered into working systems, and these systems will be subject to a comprehensive user evaluation to understand their viability and the degree to which the empirical data from actual use by domain experts fits the models developed during the design stage. Finally, a nine-month refinement stage will attempt to correct the shortcomings detected in the implementation stage evaluation, and a second evaluation will be undertaken to validate the refinements.

Future work will focus on tactical research to fill gaps identified in Halo 2 that will lead to greater coverage of the scientific domains. These efforts will investigate both automated and semiautomated methods to facilitate formulation of knowledge and posing of questions and provide better tools for evaluation and inspection of the knowledge-formulation and question-answering processes. We will also be focusing on reducing brittleness and other systemic failures of the Halo phase-two systems that will be identified by a comprehensive failure analysis of the sort we developed for phase one. We will be seeking the assistance of the KR&R community to standardize our extended failure taxonomy for use in a wide variety of knowledge-based applications.

Throughout phase two, Project Halo will be conducting an ongoing dialogue with domain experts and educators, especially those from the three target scientific disciplines. Our aims are to better understand their needs and to explain the potential benefits of the availability of high-quality machine-processable knowledge to both research and education. For example, once a proof of concept for our knowledge formulation approach has been established, Project Halo will consider how knowledge modules might be integrated into interactive tutoring applications. We will also examine how such modules might assist knowledge-driven discovery, as part of the functionality of a digital research assistant.

Acknowledgments

Full support for this research was provided by Vulcan Inc. as part of Project Halo. For more information, see www.projecthalo.com.

Notes

1. MEDLINE is the National Library of Medicine's

premier bibliographic database covering the fields of medicine, nursing, dentistry, veterinary medicine, the health care system, and the preclinical sciences.

2. Sections 2.6–2.9 in chapter 2 provide detailed information. Chapter 16 also requires the definition of moles, which appears in section 3.4, pages 87–89, and molarity, which can be found on page 134. The form of the equilibrium expression can be found on page 580, and buffer solutions can be found in section 17.2.

3. Available from http://www.opencyc.org/.

4 The system code and documentation are available at http://www.cs.utexas.edu/users/mfkb/km.html.

5. After the challenge evaluation was complete, the teams put in a considerable effort to make improved versions of their application for use by the general public. These improved versions address many of the problems encountered on the sequestered versions. Vulcan Inc has made both the sequestered and improved versions available for download on the project Halo Web site.

6. One explanation for this is that, although agreed guidelines exist for marking human justifications, the Halo systems can create justifications unlike any that the graders have seen before (for example, with extensive verbatim repetition), and for which no agreed scoring protocol has been established.

References

Angele, J.; Mönch, E.; Oppermann, H.; Staab, S.; and Wenke, D. 2003. Ontology-based Query and Answering in Chemistry: Ontonova @ Project Halo. In *Proceedings of the Second International Semantic Web Conference (ISWC2003)*. Berlin: Springer Verlag.

Barker, K.; Porter, B.; and Clark, P. A Library of Generic Concepts for Composing Knowledge Bases. In *Proceedings of the First International Conference on Knowledge Capture (K-Cap'01)*, 14–21. New York: Association for Computing Machinery Special Interest Group on Artificial Intelligence.

Barker, K.; Chaudhri, V. K.; Chaw, S. Y.; Clark, P. E.; Fan, J.; Israel, D.; Mishra, S.; Porter, B.; Romero, P.; Tecuci, D.; and Yeh, P. 2004. A Question-Answering System for AP Chemistry: Assessing KR&R Technologies. In *Proceedings of the Ninth International Conference of Knowledge Representation and Reasoning*. Menlo Park, Calif.: AAAI Press.

Batanov, D.; and Brezillon, P., eds. 1996. *Proceedings of the First International Conference on Successes and Failures of Knowledge-based Systems in Real World Applications*. Bangkok, Thailand: Asian Institute of Technology.

Brachman, R. J.; McGuinness, D. L.; Patel-Schneider, P. F.; and Borgida, A. 1999. Reducing CLASSIC to "Practice": Knowledge Representation Theory Meets Reality. *Artificial Intelligence Journal* 114(1–2): 203–237.

Brown, T. L.; LeMay, H. Eugene; and Bursten, B. 2003. *Chemistry: The Central Science*. Englewood Cliffs, NJ: Prentice Hall.

Chaudhri, V.; and Fikes, R., eds. 1999. *Question Answering Systems: Papers from the AAAI Fall Symposium*. Technical Report FS-99-02. Menlo Park, California: American Association for Artificial Intelligence.

Clancey, W. J. 1983. The Epistemology of a Rule-Based Expert System: A Framework for Explanation. *Artificial Intelligence Journal* 20(3): 215–251.

Cohen, P.; Schrag, R.; Jones, E.; Pease, A.; Lin, A.; Starr, B.; Gunning, D.; and Burke, M. 1998. The DARPA High Performance Knowledge Bases Project. *AI Magazine* 19(4): 5–49.

Friedland, N.; Allen, P. G.; Witbrock, W.; Matthews, G.; Salay, N.; Miraglia, P.; Angele, J.; Staab, S.; Israel, D.; Chaudhri, V.; Porter, B.; Barker, K.; and Clark, P. 2004. Towards a Quantitative Platform-independent Quantitative Analysis of Knowledge Systems. In *Proceedings of the Ninth International Conference of Knowledge Representation and Reasoning*. 2004. Menlo Park, Calif.: AAAI Press.

Keyes, J. 1989. Why Expert Systems Fail? *IEEE Expert* 4(11): 50–53.

Kifer, M.; Lausen, G.; and Wu, J. 1995. Logical Foundations of Object Oriented and Frame Based Languages. *Journal of the ACM*, 42(4): 741–843.

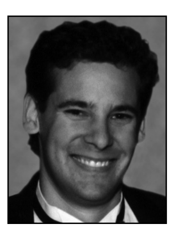
Neches, R.; Fikes, R.; Finin, T.; Gruber, T.; Patil, R.; Senator, T.; and Swartout, W. R. 1991. Enabling Technology for Knowledge Sharing. *AI Magazine* 12(3): 6–56.

Novak, G. 1995. Conversion of Units of Measurement. *IEEE Transactions on Software Engineering* 21(8): 651–661.

Pool, M.; Murray, K.; Mehrotra, M.; Schrag, R.; Blythe, J.; Kim, J.; Miraglia, P.; Russ, T.; and Schneider, D. 2003. Evaluation of Expert Knowledge Elicited for Critiquing Military Courses of Action. In *Proceedings of the Second International Conference on Knowledge Capture (KCAP-03)*. New York: Association for Computing Machinery.

Voorheese, E. M., ed. 2003. *The Twelfth Text Retrieval Conference (TREC2003)*. Washington, DC: Department of Commerce, National Institute of Standards and Technology (http://trec.nist.gov/pubs/trec12/t12_proceedings.html).

Witbrock, M.; and Matthews, G. 2003. *Cycorp Project Halo Final Report*. Austin, TX.: Cycorp.



Noah Friedland, the former Project Halo program manager at Vulcan Inc., led the development of Digital Aristotle—a knowledge application that will be capable of assisting scientific educators and researchers in their work in a growing number of scientific disciplines. Friedland holds a Ph.D. in computer science from the University of Maryland, where he studied under the late Azriel Rosenfeld. He also holds degrees in aeronautical and electrical engineering from the Technion in Haifa, Israel. Prior to joining Vulcan in 2002, Friedland held a variety of leadership roles in Seattle-area high-tech startups. His website is www.noahfriedland.com.

Investor and philanthropist **Paul G. Allen** creates

and advances world-class projects and high-impact initiatives that change and improve the world. He cofounded Microsoft with Bill Gates in 1976, remained the company's chief technologist until he left Microsoft in 1983, and is the founder and chairman of Vulcan Inc. and the chairman of Charter Communications (a broadband communications company). Named one of the top ten philanthropists in America, Allen gives back to the community through his charitable foundations. Learn more about Allen online at www.vulcan.com.

Bruce Porter (porter@cs.utexas.edu) is a professor in the Department of Computer Science at the University of Texas at Austin and the director of the Artificial Intelligence Laboratory. His research and teaching focuses on knowledge-based systems and the contributing technologies of KR&R, machine learning, explanation generation, and natural language understanding.

Ken Barker (kbarker@cs.utexas.edu) investigates knowledge-based systems as a research associate in the Department of Computer Sciences at the University of Texas at Austin. He received a Ph.D. in computational linguistics from the University of Ottawa in 1998.

James Fan (jfan@cs.utexas.edu) is a Ph.D. student at the University of Texas at Austin. He has an M.S. (2001) and a B.S. (1996) in computer sciences from the University of Texas. His research interests include knowledge-based systems, ontologies, semantic web, natural language processing, and machine learning.

Dan G. Tecuci (tecuci@cs.utexas.edu) is currently a Ph.D. student in computer sciences at the University of Texas at Austin. He received a M.Sc. in computer sciences from the same university in 2001. His main research interest is artificial intelligence with a focus on knowledge-based systems.

Peter Z. Yeh (pzyeh@cs.utexas.edu) is currently a Ph.D. student in computer sciences at the University of Texas at Austin. He received his B.S. and M.S. in computer science from the same institution. His research interests are in the areas of inexact and semantic graph matching, knowledge-base systems, and knowledge representation.

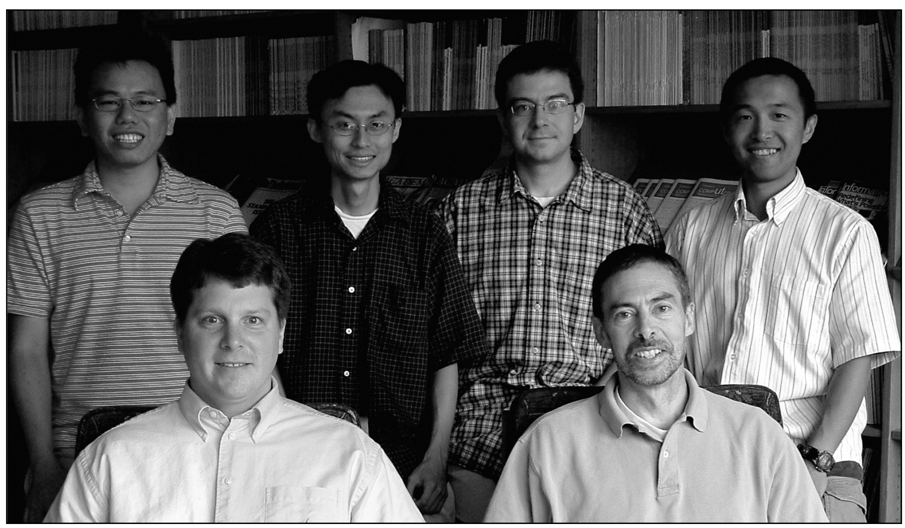
Shaw Yi Chaw (jchaw@cs.utexas.edu) is currently a Ph.D student in computer sciences at the University of Texas at Austin. He has a B.Comp in computer science from the National University of Singapore. His research interests center on advancing knowledge-based systems as a feasible tool for knowledge management and evolution.

David Israel (David.Israel@sri.com) is director of the natural language program in the Artificial Intelligence Center at SRI International.

Vinay Chaudhri (Vinay.Chaudhri@sri.com) is the program manager for ontology management in the Artificial Intelligence Center at SRI International. His research focuses on knowledge acquisition, ontologies, and inference for large shared knowledge bases.

Peter Clark (Peter.E.Clark@boeing.com) is a research scientist in the Intelligent Information Systems group, Boeing Phantom Works, leading research in the areas of knowledge-based systems, machine reasoning, and controlled language processing.

SRI Team. Front row (left to right): Lisa Ahlberg, Vinay Chaudhri, Irina Beylin, David Israel, Tomas Uribe. Back row (left to right): Ken Murray, Pedro Romero, Anne Venturelli, and David Wilkins.



Knowledge Based Systems Group at University of Texas at Austin. Sitting (left to right): Ken Barker, Bruce Porter. Standing (left to right): Jason Chaw, Peter Yeh, Dan Tecuci, James Fan.

Cycorp Team. Front row (left to right): Michael Witbrock, Gavin Matthews. Back row (left to right): David Baxter, John Curtis, Blake Shaperd.

At Left: Ontoprise Team. Sitting (left to right): Eddie Moench, Jurgen Angele. Standing (left to right): Dirk Wenke, Henrik Oppermann, Steffen Staab.

Juergen Angele (angele@ontoprise.de) is the chief executive officer and cofounder of Ontoprise GmbH. He has published some 75 papers on ontologies, frame-based logics, reasoning, and semantic web technologies. He is the inventor of the inference engine, OntoBroker.

Steffen Staab, PD Dr. rer. nat. (staab@ontoprise.de), is a senior lecturer at the University of Karlsruhe and cofounder of Ontoprise GmbH. He is currently on the editorial board of *IEEE Intelligent Systems, International Journal of Human-Computer Studies, Journal of Web Semantics,* and *Journal of IT&T.*

Dirk Wenke, Dipl. Math. Oec. (wenke@ontoprise.de), is a senior software developer and knowledge engineer at Ontoprise GmbH, Karlsruhe. He is responsible for the development of the ontology editing tools.

Eddie Moench, Dipl. Wi.-Ing. (moench @ontoprise.de), is software engineer at Ontoprise GmbH. He is responsible for the development of SemanticMiner, the ontology-based knowledge-retrieval application.

Henrik Oppermann, Dipl.-Ing. (oppermann@ontoprise.de), is vice president of professional services at and shareholder of Ontoprise GmbH. His focus is on industrial application of ontologies and reasoning technologies.

Gavin Matthews (gmatthew@cyc.com) is a project manager with Cycorp. He holds an M.A. in computer science and mathematics from the University of Cambridge and graduated in 1991. He has been with Cycorp since 2002. He has experience in software engineering management at Geodesic Systems, Harlequin Group plc, and Electronic Data Systems, specializing in memory management and 3-D solid modeling.

Michael Witbrock (witbrock@cyc.com) is currently vice president for research at Cycorp. Previously Witbrock was a principal scientist at Terra Lycos, a research scientist at Just Systems Pittsburgh Research Center, and a system scientist at CMU. He has a Ph.D. in computer science from Carnegie Mellon University and is the author of more than two dozen papers and two granted patents.

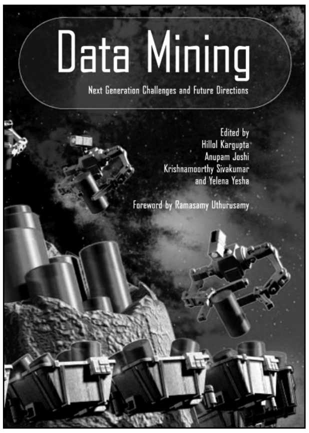
John Curtis (jonc@cyc.com) received his Ph.D. in philosophy at the Ohio State University in 1999, where he specialized in the history of analytic philosophy and the philosophy of logic. With Cycorp since 1999, he has worked primarily at the intersection of NLP and ontology, designing semantic models for parsing, and overseeing the development and implementation of strategies for resolving intermediate-level representations (Cyc's semantic layer between natural language and inferentially salient CycL).

Pierluigi Miraglia (plmiraglia@yahoo.com) earned his doctorate in philosophy at the Ohio State University. After working for several years at Cycorp, he is now a senior ontologist with Convera Corporation. He lives in Austin, Texas.

David Baxter (baxter@cyc.com) holds a Ph.D. in linguistics from the University of Illinois. He has worked for Cycorp since 1998 and specializes in natural language parsing and generation.

Blake Shepard (blake@cyc.com) is an ontologist at Cycorp. He received his Ph.D. in philosophy from the University of Texas at Austin in 2000 and has been with Cycorp since 1999.

www.aaai.org/Press/Books/



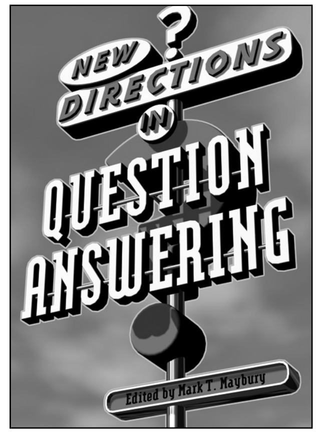
Data Mining: Next Generation Challenges and Future Directions

Edited by Hillol Kargupta, Anupam Joshi, Krishnamoorthy Sivakumar and Yelena Yesha

Data mining, or knowledge discovery, has become an indispensable technology for businesses and researchers in many fields. Drawing on work in such areas as statistics, machine learning, pattern recognition, databases, and high performance computing, data mining extracts useful information from the large data sets now available to industry and science. This collection surveys the most recent advances in the field and charts directions for future research.

The first part looks at pervasive, distributed, and stream data mining, discussing topics that include distributed data mining algorithms for new application areas, several aspects of next-generation data mining systems and applications, and detection of recurrent patterns in digital media. The second part considers data mining, counter-terrorism, and privacy concerns, examining such topics as biosurveillance, marshalling evidence through data mining, and link discovery. The third part looks at scientific data mining; topics include mining temporally-varying phenomena, data sets using graphs, and spatial data mining. The last part considers web, semantics, and data mining, examining advances in text mining algorithms and software, semantic webs, and other subjects.

ISBN 0-262-61203-8 576 pp, index.
$40.00 softcover Copublished by The MIT Press



New Directions in Question Answering

Edited by Mark Maybury

Question answering systems, which provide natural language responses to natural language queries, are the subject of rapidly advancing research encompassing both academic study and commercial applications, the most well-known of which is the search engine Ask Jeeves. Question answering draws on different fields and technologies, including natural language processing, information retrieval, explanation generation, and human computer interaction. Question answering creates an important new method of information access and can be seen as the natural step beyond such standard Web search methods as keyword query and document retrieval. This collection charts significant new directions in the field, including temporal, spatial, definitional, biographical, multimedia, and multilingual question answering. After an introduction that defines essential terminology and provides a roadmap to future trends, the book covers key areas of research and development. These include current methods, architecture requirements, and the history of question answering on the Web; the development of systems to address new types of questions; interactivity, which is often required for clarification of questions or answers; reuse of answers; advanced methods; and knowledge representation and reasoning used to support question answering. Each section contains an introduction that summarizes the chapters included and places them in context, relating them to the other chapters in the book as well as to the existing literature in the field and assessing the problems and challenges that remain.

ISBN 0-262-63304-3 352 pp., index.
$40.00 softcover Copublished by The MIT Pres