AI Magazine Volume 11 Number 2 (1990) (© AAAI)

Term Subsumption Languages in Knowledge Representation

Peter F. Patel-Schneider, AT&T Bell Laboratories, Murray Hill, New Jersey

Bernd Owsnicki-Klewe, Philips Research Laboratories, Hamburg, West Germany

Alfred Kobsa, University of Saarbrücken, West Germany

Nicola Guarino, LADSEB-CNR, Padova, Italy

Robert MacGregor, USC/Information Sciences Institute, Marina del Rey, California

William S. Mark, Lockheed AI Center, Menlo Park, California

Deborah L. McGuinness, AT&T Bell Laboratories, Murray Hill, New Jersey

Bernhard Nebel, DFKI GmbH, Saarbrücken, West Germany

Albrecht Schmiedel, Technical University of Berlin, West Germany

John Yen, Texas A&M University, College Station, Texas

The Workshop on Term Subsumption Languages in Knowledge Representation was held 18–20 October 1989 at the Inn at Thorn Hill, located in the White Mountain region of New Hampshire. The workshop was organized by Peter F. Patel-Schneider of AT&T Bell Laboratories, Murray Hill, New Jersey; Marc Vilain of MITRE, Bedford, Massachusetts; Ramesh S. Patil of the Massachusetts Institute of Technology (MIT); and Bill Mark of the Lockheed AI Center, Menlo Park, California. Support was provided by the American Association for Artificial Intelligence and AT&T Bell Laboratories.

This workshop was the latest in a series in this area. Previous workshops have had a slightly narrower focus, being explicitly concerned with KL-One, the first knowledge representation system based on a term subsumption language (TSL), or its successor, NIKL. Two of these workshops were held in 1981 (Schmolze and Brachman 1982) and 1986 (Moore 1986).

The workshop brought together 34 researchers and students in the field from the United States, Germany, Austria, Italy, Canada, and Korea. Its primary goal was to review the field after about 10 years of work, exchange ideas, and set up research directions for the future.

Term Subsumption Languages

Term subsumption languages are knowledge representation formalisms that employ a formal language with a formal semantics for the definition of terms (more commonly referred to as concepts or classes) and that deduce whether one term subsumes (is more general than) another. These formalisms generally descend from the ideas presented in KL-One (Brachman and Schmolze 1985). TSLs are a generalization of both semantic networks and frames. One result of the workshop was to standardize use of the term terminological logics to describe these formalisms; term subsumption languages was chosen as a neutral term for describing the workshop.

In the last few years, many knowledge representation systems have been built using TSLs, including Krypton (Brachman et al. 1985), KL-Two (Vilain 1984), NIKL (Robbins 1986; Kaczmarek, Bates, and Robbins 1986), Back (Peltason et al. 1989; Nebel and vonLuck 1988), Meson (Edelmann and Owsnicki 1986), SB-One (Kobsa 1990), Loom (MacGregor and Bates 1987), Quirk (Bergmann and Gerlach 1987), and Classic (Borgida et al. 1989). These systems go beyond a bare TSL in various ways: Almost all of them incorporate assertional languages that enable the systems to reason about instances of terms, some of them allow for retraction of told facts, and so on. The workshop not only concerned TSLs but also TSL-based knowledge representation systems and their use in larger AI systems.

Outline of the Workshop

The workshop was designed to encourage discussion. To aid this approach, no formal talks were presented, and no proceedings is being produced.

For a large portion of the workshop, the attendees were divided into working groups of 7 to 15 participants. Each working group was devoted to in-depth discussion of particular topics. Moderators were chosen to keep the discussions flowing and on track and were assisted by preselected discussants who presented short position statements. Ample time was left for intensive discussion, although several of the discussions could not be completed within their allotted time and had to be continued in the evening. Moderators reported the results of the working groups in plenary sessions that also allowed for further discussion of the topics covered.

Personal Viewpoints

The workshop started with a set of personal viewpoints outlining the problems in the field. The viewpoints were designed to provide a basis for discussion throughout the workshop.

Ramesh Patil, a major user of TSL-based systems, spoke on behalf of the system users. He advocated the idea of knowledge representation as a service (Doyle and Patil 1989): Knowledge representation systems should offer specialized services for the application designer in terms of definitions, assertions, and queries. His claim was that knowledge representation systems suffer from a lack of knowledge about what the user wants to do with an answer and, thus, are forced to do almost complete deduction, even in cases where an incomplete but fast answer would be preferable. He pointed out that it is not sufficient to look only for local run-time optimization; instead, the overall performance of the system should be improved.

0738-4602/89/$4.00 ©1990 AAAI

Bill Swartout of USC/Information Sciences Institute (USC/ISI), Marina del Rey, California, reported on his experience with the NIKL system, which he used to provide enhanced explanation facilities for expert systems. Like many participants, he questioned an overly strict distinction between terminological and assertional reasoning within these systems as hard to comprehend and an obstacle to rich expressiveness. He stated that classification has been overrated and questioned (as did Ramesh Patil) the point of complete inferences that require limited expressiveness. Limiting expressiveness requires model hacks by the user that can make it impossible to generate explanations. Swartout also presented some design ideas of his Hi-Fi system (Smoliar and Swartout 1988), which is supposed to be a user-configurable problem solver.

Bob MacGregor, the main developer of Loom, took up Swartout's critique of classification and pointed out that modern TSL-based knowledge representation systems tend to put strong emphasis on capabilities other than classification. Such services include forward-chained inferences, semantic unification, inconsistency detection, object-oriented programming, and the integration of highly tuned and fast special-purpose reasoners. He emphasized that this technology could serve as a good basis for both next-generation expert systems and high-level programming languages, the main obstacle being the lack of integration with existing tools, databases, and programming languages. He also pointed out that in the last few years, TSLs have become increasingly similar to the Omega language (Attardi and Simi 1981) and suggested that this relationship should be more thoroughly investigated.

Bernhard Nebel (DFKI GmbH, Saarbrücken, West Germany), a developer of Back, reviewed the broad theoretical work on TSLs, which in past years focused on analyzing computability and computational complexity of the subsumption problem for different languages (for example, Nebel 1988). Recent results have shown that all reasonable terminological languages have the property of being worst-case intractable; that is, a complete algorithm can come up with an exponential time behavior. He proposed the analysis of normal and average cases as a new direction because exponential time consumption occurs rarely, if ever, in practical applications.

TSLs are a generalization of both semantic networks and frames.

The Use of Term Subsumption Languages

Several of the workshop sessions discussed topics having to do with the use, in applications, of systems based on TSLs and with properties of TSLs driven by applications.

The plenary session on using systems based on TSLs, moderated by Ramesh Patil, gave some impressions of the structure and the use of applications based on TSLs. Lewis Norton of the Unisys Paoli Research Center, Paoli, Pennsylvania, briefly reported on a knowledge base in the KNET formalism for hardware configuration. This knowledge base contains about 2000 concepts and 150 different roles, each concept having an average of about 35 roles. The depth of the hierarchy is below 10; concepts are manually placed in the hierarchy (KNET does not have a classifier). Ramesh Patil estimated that in a large knowledge base he is familiar with, 75 percent of the concepts are primitive because of the limited expressive power of TSL that is employed (Doyle and Patil 1989). Therefore, he strongly opposed any attempt to further restrict the expressiveness of TSL systems. Instead, he proposed that such systems be configured on a pay-as-you-go basis: If the application uses only a small portion of the expressive power of TSL, then the system will be fast; if more expressive power is used, then the system might slow down but will still be able to represent and reason with the knowledge given it.

Marc Vilain reported that in his natural language parser, nearly 100 percent of the interaction with the terminological knowledge base was classification. Alfred Kobsa of the University of Saarbrücken, West Germany, estimated that in the Xtra natural language system (which also comprises a natural language generator and a user-modeling component), this amount does not exceed 20 percent to 30 percent.

Eric Mays of the IBM T. J. Watson Research Center, Yorktown Heights, New York, reported on a financial marketing expert system using a TSL system. In this system, the TSL system includes a Lisp-based procedural component for computing the values of roles. However, the TSL system still maintains control over the semantics of objects as opposed to the situation in commercial frame-based systems, where the system-supplied semantics of objects can be subverted by users.

In the plenary session entitled "What Expressive Power Is Desirable?" moderated by Jim Schmolze of Tufts University, Medford, Massachusetts, a great number of new representational elements were listed that one would like to see in enhanced TSLs to make them more usable in applications. These elements include representations for time, belief, goals, plans, disjunction and negation, quantification, lambda abstraction, time intervals, and sequences. However, most people agreed that the concept- and object-centered approach of TSLs has advantages over arbitrary assertions.

The problem is not really what to add but how to effectively add it. One approach, suggested by Bill Swartout and incorporated in Hi-Fi, is to allow essentially unrestricted expressive power and turn over all nontrivial deduction to user-specified inference procedures. This approach puts the burden of maintaining efficiency and completeness on the shoulders of application builders, who should know more about what sorts of inferences are needed in particular applications. In this way, application systems can be complete (in the sense of performing all inferences required for the application) without being theoretically complete. Another approach is to use incomplete but expressively powerful systems, where the inferences performed by the system are crafted to be reasonably quick yet still cover the inferences needed by applications. (Loom is a good example of this approach.) This approach has the benefit of removing control of inference from application builders, who might not be capable of determining reasonable sets of inferences. A third approach is to build expressively powerful and also complete systems (modulo undecidability problems). The efficacy of this approach rests on the observation that applications are generally well behaved and do not

Incorporating defaults into the next generation of TSLs was generally considered an important step.

exhibit the computationally worst case.

The session on nonterminological extensions to standard TSLs, moderated by Albrecht Schmiedel of the Technical University of Berlin, consisted of discussions about three distinct kinds of extension: default inferences, in addition to the standard inferences based on definitions and implications; explicit incorporation of time into the term language; and the representation of multiple agents.

Incorporating defaults into the next generation of TSLs was generally considered an important step. Lee Spector of the University of Maryland opened the discussion with some remarks on defaults in the Parka system, which are basically default role values that can later be overridden. Spector's remarks led to a discussion on representing part-whole relationships. In Parka, this representation is done with aggregation node types and constraints to trigger additional subsumption links between concepts representing parts (for example, car-part ISA road-vehicle-part) dependent on compound roles.

A different approach to integrating defaults was presented by Bernhard Pfahringer of the University of Vienna. He captures defaults using two new kinds of links between concepts, typically-implies and not-a-typical, that are based on the notion of skeptical inheritance.

One of the main problems with this approach, which was brought up in the discussion, concerned the necessity of dealing with multiple extensions even when skeptical inheritance is adopted because of the propagation of defaults across roles. Bob McGregor conjectured that computing skeptical inheritance would be intractable, if not undecidable, when propagation across roles was taken into account. This conjecture is at odds with what defaults are good for in the first place—to provide quick and unambiguous intelligent guesses in the absence of other information. In Loom, therefore, where defaults are handled with the same machinery as strict implications and, thus, as efficiently, a single extension as a result of applying defaults is guaranteed. However, there is a price to be paid: This extension might be inconsistent.

Albrecht Schmiedel proposed an extension to TSLs allowing explicit reference to time. A new term-forming operator is introduced, inside of which predications valid for certain times—represented by time interval variables—can be made; time interval variables can be constrained by absolute times, durations, and their interrelations. A special variable, Now, can be used to relate to the time of validity. Concepts without temporal operators are generally valid Now; concepts with temporal operators that don't refer to the Now variable are universally valid. Subsumption can be defined straightforwardly, taking into account temporal properties of predications such as liquidity; however, algorithms for computing subsumption and complexity results are still missing.

Sangki Han of the Samsung Advanced Institute of Technology, Suwon, South Korea, was concerned with the representation of multiple agents within the framework of Sphinx. Sphinx is a hybrid knowledge representation system developed in Korea that is designed to handle multiagent problem solving. Han described an architecture of multiple Sphinxes, one for each agent that represents the agent's particular beliefs and one that represents the real world and the relations between the agents. Each has its own set of term definitions and individuals and a set of horn clauses for inferences. Queries involving the epistemic states and relationships of the different agents can be evaluated using an extension of Konolige's deduction model.

System Issues

A number of workshop sessions were concerned with implementation and integration issues of TSL-based systems. One of these sessions covered the area of system implementations (moderated by Robert MacGregor) and involved discussions about implementational details. The first half of this session was devoted to descriptions of the software architecture of four different systems. The central reasoning component in each of these systems is a term classifier. The second half of this session explored some recent research on the possibility of improving the performance of terminological reasoners by using parallel hardware implementations.

Bob MacGregor described the Loom knowledge representation system being developed at USC/ISI. Loom features the ability to revise both terminological and assertional knowledge. Also, Loom extends the traditional terminological knowledge representation paradigm to include horn logic and a default logic.

Lori Alperin Resnick of AT&T Bell Laboratories described the Classic knowledge representation system being developed there. Classic was recently upgraded to support revision of assertional knowledge. Both Classic and Loom appear to have developed an object-based technology for truth-maintaining assertional knowledge that is distinct from the traditional justification-based and assumption-based truth maintenance approaches.

Albrecht Schmiedel described the status of the Back system being developed at the Berlin Technical University. Back is currently being redesigned, with a new emphasis on (1) interfacing Back's terminological reasoner to a database management system (DBMS) and (2) extending the terminological and assertional components to represent and reason with temporal knowledge.

Bernd Owsnicki-Klewe of Philips Research Laboratories, Hamburg, West Germany, described the Meson system developed at Philips. Meson extends the traditional TBox and ABox hybrid architecture: Knowledge that represents monadic implications, individuality, and disjointedness is considered to belong to a UBox, and a separate hierarchy for describing nonconcepts (for example, integer ranges, strings) is labeled the DBox. Overall, these systems exhibit a surprising amount of underlying uniformity, which means either that they are all working in a promising direction or that they have all missed the mark.

Under the topic of parallel architectures, Owsnicki-Klewe presented some ideas on how a hardware network (analogous to a network that implements a shuffle-sort algorithm) could be configured to compute subsumption relationships in parallel. Lee Spector described the architecture of the Parka system, which implements a knowledge representation facility on a connection machine. Simulations on randomly generated nets showed that parallel processing yielded an exponential savings of time for many kinds of queries, including simple property inheritance. Spector claimed that the architecture could also efficiently perform concept classification for the special case of concepts whose representation can easily be reduced to a canonical form.

In the discussion that ensued after the last presentation, the general consensus was that significant performance gains using massively parallel (SIMD) architectures might be possible when answering some specific kinds of queries. However, the general problem of concept classification is probably amenable only to speedups using MIMD machines and is not well suited to a SIMD architecture.

Deborah McGuinness, AT&T Bell Laboratories, moderated the session on user interfaces to TSL-based systems. Participants began by recognizing the distinction between user interfaces for knowledge base designers and those for end users and maintainers of applications built using TSLs. The discussion also addressed user interfaces for system developers. Alfred Kobsa presented generally agreed-on design criteria for these interfaces. Requirements include consistency of operations, direct manipulation, direct feedback, multimodality, customizability, system and user control, an undo mechanism, a navigation mechanism, browsing and query support, information reduction and abstraction tools, and explanation aids.

There was not a general consensus on how to implement all these goals. Alfred Kobsa, John Yen (Texas A&M University), and Lew Norton presented user interfaces on which they had worked. Respectively, they each focused on a user interface for graphic natural language applications, retrieval by reformulation, and the assistance of maintainability as a primary goal. Some participants supported highly graphic interfaces; it was argued, however, that the presentation issues such as the appropriate use of interactive graphics and user versus system control of layout must then be addressed. The point was made that for very large knowledge bases, issues of fast layout, navigation by description, and explanation of classification results are important but not yet satisfactorily solved in many systems. Another claim was made that there are at least four levels of interface: graphic (probably for less experienced users), textual, term definition, and abbreviated keystroke (for users most familiar with the system). Many sophisticated designers of large knowledge bases prefer text editors over interactive user interfaces.

Current user interfaces display a large variety of features, and many desired support mechanisms are not implemented in most systems. Graphic browsing is a necessary but not sufficient ingredient for a usable system. The key for future interfaces is to include all the necessary components in a coherent, integrated manner. Possibly, user modeling will ease this task.

The session on TSL-based systems and other computing paradigms, moderated by John Yen, discussed the current and future impacts on TSLs on rule-based programming and object-oriented programming paradigms (aspects of databases were explicitly excluded and treated in a separate session). Presentations on Consul, by William Mark, SB-One, by Alfred Kobsa, and Loom, by John Yen, showed different approaches to

. . . a formal semantics is extremely useful for knowledge representation systems. Formal semantics gives a better understanding of what TSL is able to express . . .

both the integration and the use of rules in a TSL context.

Two major contributions to rule-based paradigms were identified. The first contribution involves the philosophy of the use of rules (Consul [Mark 1981]). Rules in Consul map one description to another until a description with an associated action is reached. Hence, the system's inferential activity is completely directed at formulating an actionable description of an entity that does not come into the system described in some immediately actionable way. Although Consul's transformation rules are not as general as production rules, some researchers believe that they demonstrate a better use of rules. Similar transformation rules have been used to map linguistic structures to domain-specific knowledge in a natural language application built using SB-One.

The second contribution of term subsumption systems to the rule-based paradigm is in improving the cleanliness of rule-based systems. As demonstrated in Consul and KL-Two, using terms defined in the terminological knowledge base to describe a rule's condition improves the reusability of definitional knowledge and the understandability of rules. This benefit becomes more significant as the expressiveness of a terminologically based rule system moves toward that of a production system (for example, conjunctive patterns with multiple free variables).

In addition, the idea of classifying concepts in KL-One has recently been carried over to a rule-based paradigm for classifying patterns in Clasp (Yen, Neches, and MacGregor 1989), a production system built on top of Loom. As a result, production rules in a knowledge base can be organized based on a principled measure of the specificity of their conditions. Regarding the contribution to the object-oriented programming paradigm, Clasp's pattern-classification capability allows methods in object-oriented programming to be extended to describe complex situations to which they apply.

Aspects of integrating classification-based inference services with other ways of computing seem to be an important point if these systems are to be widely useful. Yet, ways of performing this integration—as far as systems care about this point at all—are still quite different. Loom offers all reasoning capabilities in one shell, making it, as one participant put it, a strong competitor to Kee. Other systems, such as Meson, provide a variety of programming interfaces that let the user embed this service into the special problem-solving or computing environment.

The session on TSLs and databases, moderated by Rich Fritzson of the Unisys Paoli Research Center, centered on topics of data-knowledge persistency, namely, mass data and shareability, on one hand, and access of a knowledge representation system to existing databases on the other. It was proposed that we view a database as a kind of instance server, where database entries are transformed into a nonpersistent internal form. It was additionally stated that in a normal DBMS, the notion of a query (few queries, large answers) differs widely from the one usually employed in a knowledge representation system (many queries, small answers). Thus, in a knowledge representation system, issues of query optimization do not make much sense.

The problems of database integration are still basically unsolved. This situation seems to stem from two facts: First, knowledge persistency and concurrent access to knowledge bases are becoming necessary for large applications. Thus, it seems reasonable to use standard database techniques for these problems. Second, standard databases are expressively too weak to support nontrivial hybrid inferences.

Finally, it was noted that if research begins focusing on new data models that support persistency, concurrency, and expressiveness, the aspect of accessing already existing databases can be omitted.

Applications in Natural Language Processing

The session on TSLs and natural language processing, moderated by Marc Vilain, featured presentations on feature grammars and classification-based representation by Bob Kasper of USC/ISI; referent finding by Manfred Gehrke of Siemens AG, Munich, West Germany; and lexical acquisition by Howard Beck of the University of Florida, Gainsville. The session showed the usefulness of classification-based systems for a broad area of natural language processing.

The discussion centered around the degree to which linguistic knowledge should be encoded in TSL (for example, should a terminological knowledge base contain lexical items?) and whether proximity of concepts in a terminological knowledge base would correspond to linguistic proximity of the denoted objects (an important issue for anaphora resolution). It was generally agreed that hybrid architectures, including TSLs, can be usefully employed for tasks of discourse analysis, which can make heavy use of terminological and assertional inference. TSLs are well suited to natural language processing because they do not make the closed-world assumption common to databases and many knowledge representation systems.

Sessions on Formal Issues

Several sessions of the workshop discussed formal issues of TSLs. Peter Patel-Schneider moderated the session on aspects of formal semantics and analysis in TSLs. All participants agreed that a formal semantics is extremely useful for knowledge representation systems. Formal semantics gives a better understanding of what TSL is able to express and probably also reveals points where problems might occur as well as serves as a reference point for implementations. The formal semantics need not be complete to be useful; systems that implement partial deduction (as do many TSL systems) still benefit from a formal semantics.

A model-theoretic semantics is generally preferable to a proof-theoretic one because a semantics should capture our intuitions about how the world is structured rather than describe the algorithms used in the knowledge representation system. Nevertheless, a proof theory for a system can also be useful, particularly where the system is incomplete with respect to its model-theoretic semantics.

TSLs need more than just a standard static semantics, however. There is a need for a formal semantics and analysis of retraction of facts, temporal updates, inconsistency and incoherency, and defaults. The analysis of inconsistency and incoherency will be particularly important in large knowledge bases to characterize how systems prevent pollution of the knowledge base when inconsistency or incoherency is detected.

Bernhard Nebel moderated the session on the consequences of intractability and undecidability. Discussion focused on questions such as why tractability and decidability problems are interesting to the work in the TSL field, what could be done in cases of intractable problems, and why undecidability seems to be a more severe problem than intractability.

There was a consensus among the discussants that TSL should be efficient and principled; that is, it should provide answers in a reasonable amount of time, and the behavior should be justified by something other than the code implementing the system. However, interpreting the two terms efficient and principled as worst-case tractability and soundness and completeness with respect to the standard model-theoretic semantics leads us to conclude that we should not try to implement TSLs at all; as was shown recently, even in Krypton, subsumption is worst-case intractable. Furthermore, as we now know, including so-called role-value maps in TSL results in the undecidability of subsumption.

Proposals for coping with this problem include (1) using weak semantics to characterize the (tractable) system behavior (however, it didn't seem clear how useful this approach is in practice); (2) using a large set of inference rules to characterize the behavior (this approach was criticized because such a specification gives neither a good intuitive characterization of the system nor a formal one); (3) identifying normal cases for which tractability or decidability can be proven (the usefulness of this approach has to be shown in evaluating existing knowledge bases); (4) empirically identifying tractable special cases (which might be hard to characterize from a formal point of view but has been done, for instance, in the area of unification grammars); and (5) applying a decision-theoretic analysis to the services of a knowledge representation system, a solution that requires a deep analysis of the tasks an overall system is supposed to perform.

The last three proposals would greatly benefit from a corpus of terminological knowledge bases, as was discussed in the plenary session afterwards, especially because the negative complexity and decidability results contrast with the results from actual running systems. As Nebel put it in his introductory statement, worst cases tend to not occur.

In the session on hybrid architectures, moderated by Nicola Guarino of LADSEB-CNR, Padova, Italy, discussion concentrated on a conceptual characterization of the differences between *terminological knowledge*, which has to do with terminology, and *assertional knowledge*, which has to do with assertions, rather than on a distinction between different formalisms.

A preliminary issue concerned the real utility of this distinction. Jim Schmolze argued that if you think of a sort of lingua franca for knowledge representation, you can't be committed to the difference between terminological and assertional knowledge or even between roles and concepts. Ramesh Patil observed that the distinction actually depends on the knowledge engineer, in the sense that from the outside, it is not easy to tell who did what. In this respect, both Bill Swartout and Jim Schmolze brought up an interesting point about the opportunity of decoupling expressiveness from reasoning: The knowledge representation language might well consist of a single, uniform formalism accessed by different special-purpose reasoners.

A less radical perspective that was well received consisted of an extension of what is today accounted for as terminological knowledge. Although some people thought that we have to stick to definitions, others felt that in the medium term, we should be able to incorporate some information about individuals into terminological knowledge. In other words, the word terminological should gain a broader interpretation —as a synonym for taxonomic or intensional instead of just definitional or analytic.

Individuals are often part of the

. . . achieving success will require great attention to the integration of TSL-based systems into the existing technology base.

terminology of a domain, in the sense that we often perform implicit taxonomic classification on them. Nicola Guarino cited the example of Frisch's substitutional approach to many-sorted logics, where the *sortal theory* (which contains taxonomic knowledge about terms, including inviduals) can be compared with the terminological knowledge, and the *object theory* (which describes arbitrary relations among terms) can be compared with assertional knowledge. Individuals crop up in terminological knowledge in three different ways: (1) as background knowledge within a definition (for example, when we want to define the class of people who work in specific institutions), (2) when a concept definition depends on the assertional properties of its instances (as with gray elephants, for example), and (3) when we need to perform intensional reasoning on properties of individuals (as in the statement, Pat knows Mike's telephone number). In all these cases, a broader definition of terminological knowledge (as in the case of the DRL system [Guarino 1989])—or at least a deeper interaction between assertional and terminological knowledge—might be highly desirable, as recognized by Ramesh Patil, Yeona Yang of MIT, Massimo Poesio of the University of Rochester, and Peter Patel-Schneider.

Future Directions

The concluding session, on future directions, was moderated by Bill Mark. The purpose of this session was to establish directions for the future development and use of terminological knowledge representation systems. The discussion focused on four issues: (1) assessment (what has been the value of the systems developed to date; that is, what claims can be made for TSL knowledge representation systems?), (2) goals (what are reasonable goals for the future; that is, what claims do we want to make?), (3) approaches (how should we go about achieving our goals?), and (4) evaluation (how will we know if we succeed?).

The session found broad consensus on most of the issues. It was generally agreed that work on TSL-based systems has resulted in a better understanding of existing knowledge representation formalisms and knowledge representation in general. Peter Patel-Schneider noted that formal analysis of the completeness and tractability of subsumption algorithms has played a key role and noted the difference between TSL-based knowledge representation and other approaches.

The unanimous opinion of the session participants was that it is now time to build significant applications using TSL-based systems, and that the success or failure of this effort will constitute the most valuable metric of progress in research in TSLs. In fact, routine use of TSLs or TSL-based knowledge representation system AI applications was agreed on as definite evidence of success. Jon Doyle of MIT pointed out that this metric should be refined by an analysis of the applications to determine which kinds are amenable to which techniques, including TSL-based systems.

It was recognized that as with any AI tool technology, achieving success will require great attention to the integration of TSL-based systems into the existing technology base (interfaces to DBMSs, high-quality training and documentation, and so on). To some extent, TSL-based knowledge representation tools will be seen as alternatives to commercial shells such as Art and Kee. Widespread use in AI applications will, therefore, depend on the ability of the builders of TSL-based knowledge representation systems to clearly explain the advantages of their approach over that of the commercial shells. Bob MacGregor, Bill Mark, and others saw this integration as a key part of their work in the near future. In the medium term, Peter Patel-Schneider argued that the community should reformulate and extend the epistemological basis of terminological knowledge along the lines discussed in the session on hybrid architectures.

Finally, Marc Vilain and others saw the systematic compilation of a knowledge representation corpus as an important goal of application building. This corpus could be used as a standard for comparing various systems and to delineate research problems.

Conclusions

Despite being physically separated for many years, researchers in TSLs seem to have reached a good common standard of thinking about the field, and the systems have shown remarkable compatibility in terms of expressiveness and implementation. Systems seem to differ mainly with respect to the additional services they provide. Such extra services include assertional reasoners and integration with rule-based and object-oriented programming.

This compatibility gave rise to a great amount of agreement in the sessions; controversial discussions were rare! Thus, to a certain extent, the workshop might mark a historical event in TSL research: henceforth, thinking might focus on the issues of integration, transfer, and applications; theoretical considerations will take into account normal cases rather than worst cases.

The goal of having a strongly discussion-oriented workshop rather than another miniconference seemed to be met by the workshop's participants. Certainly, the atmosphere and fabulous cuisine of the 100-year-old Victorian country inn that hosted the workshop contributed to the workshop's success. The organizers even managed to keep the weather cold and rainy during the entire workshop (so that everybody was forced to work) except for one afternoon when they permitted the sun to shine for three hours, allowing a hike in the beautiful White Mountains.

Acknowledgments

This report is based on a report by Bernd Owsnicki-Klewe and Alfred Kobsa and includes reports from various session moderators from the workshop.

Thanks are owed to Bernhard Pfahringer and Manolis Koubarakis of the University of Toronto for their comments on previous versions of this article.

References

Attardi, G., and Simi, M. 1981. Consistency and Completeness of OMEGA, a Logic for Knowledge Representation. In Proceedings of the Seventh International Joint Conference on Artificial Intelligence, 504–510. Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.

Bergmann, H., and Gerlach, M. 1987. QUIRK: Implementierung Einer tbox zur Reprasentation von Bekgrifflichem Wissen. Technical Report, WISBER Memo, 11, Project WISBER, Dept. of Computer Science, Univ. of Hamburg.

Borgida, A.; Brachman, R.; McGuinness, D.; and Resnick, L. 1989. CLASSIC: A Structural Data Model for Objects. In Proceedings of the 1989 ACM SIGMOD International Conference on Management of Data, 59–67. Association for Computing Machinery.

Brachman, R., and Schmolze, J. 1985. An Overview of the KL-ONE Knowledge Representation System. *Cognitive Science* 9(2): 171–216.

Brachman, R.; Gilbert, P. V.; and Levesque, H. 1985. An Essential Hybrid Reasoning System: Knowledge and Symbol Level Accounts of KRYPTON. In Proceedings of the Ninth International Joint Conference on Artificial Intelligence, 532–539, Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.

Doyle, J., and Patil, R. 1989. Language Restrictions, Taxonomic Classification, and the Utility of Representation Services. MIT/LCS/TM-387, Laboratory for Computer Science, Massachusetts Institute of Technology.

Edelmann, J., and Owsnicki, B. 1986. Data Models in Knowledge Representation Systems: A Case Study. In Proceedings of GWAI-86: The 10th German Workshop on Artificial Intelligence, eds. C. Rollinger and W. Horn, 69–74.

Guarino, N. 1989. Nature and Structure of Terminologic Knowledge. In Proceedings of the First Conference of the Italian Association for Artificial Intelligence, 20–31. Trento, Italy: Italian Association for Artificial Intelligence.

Kaczmarek, T.; Bates, R.; and Robbins, G. 1986. Recent Developments in NIKL. In Proceedings of the Fifth National Conference on Artificial Intelligence, 978–985. Menlo Park, Calif.: American Association for Artificial Intelligence.

Kobsa, A. 1990. The SB-ONE Knowledge Representation Workbench. In *Formal Aspects of Semantic Networks,* ed. J. Sowa. San Mateo, Calif.: Morgan Kaufmann. Forthcoming.

MacGregor, R., and Bates, R. 1987. The Loom Knowledge Representation Language. Technical Report, ISI/RS-87-188, USC/Information Sciences Institute, Marina del Rey, Calif.

Mark, W. 1981. Representation and Inference in the Consul System. In Proceedings of the Seventh International Joint Conference on Artificial Intelligence, 375–381. Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.

Moore, J. 1986. NIKL Workshop Summary, USC/Information Sciences Institute, Marina del Rey, Calif.

Nebel, B. 1988. Computational Complexity of Terminological Reasoning in BACK. *Artificial Intelligence* 34(3): 371–383.

Nebel, B., and vonLuck, K. 1988. Hybrid Reasoning in BACK. In *Methodologies for Intelligent Systems*, Volume 3, eds. Z. W. Ras and L. Saitta, 260–269. New York: North-Holland.

Peltason, C.; Schmiedel, A.; Kindermann, C.; and Quantz, J. 1989. The BACK System Revisited. KIT Report, 75, Dept. of Computer Science, Technical University of Berlin.

Robbins, G. 1986. The NIKL Manual, USC/Information Sciences Institute, Marina del Rey, Calif

Schmolze, J., and Brachman, R., eds. 1982. Proceedings of the 1981 KL-One Workshop, Bolt Beranek and Newman Inc., Jackson, N.H.

Smoliar, S., and Swartout, W. 1988. A Report from the Frontiers of Knowledge Representation, USC/Information Sciences Institute, Marina del Rey, Calif.

Studer, R., and Borner, S. 1988. An Approach to Managing Large Inheritance Networks within a DBS Supporting Nested Relations. In *Nested Relations and Complex Objects in Data Bases* (Lecture Notes in Computer Science 361), eds. S. Abiteboul, P. C. Fischer, and H.-J. Scheck, 229–239. New York: Springer-Verlag.

Vilain, M. 1984. KL-TWO, A Hybrid Knowledge Representation System, BBN Laboratories., Cambridge, Mass.

Yen, J.; Neches, R.; and MacGregor, R. 1989. Using Terminological Models to Enhance the Rule-Based Paradigm. In Proceedings of the Second International Symposium on Artificial Intelligence.