Statement of Interest

James G. Schmolze
Dept. of Computer Science
Tufts University
Medford, MA 02155 USA
Internet: schmolze@cs.tufts.edu

15 June 1989

Brief Statement of my General Interests:

My previous work in the KR area focused mainly on terminological systems (although we didn’t call them that then), namely KL-ONE and NIKL [1,2,3,4,11]. My dissertation [5,6] articulated knowledge that a robot would need when performing certain everyday tasks. In specifying this robot knowledge, I very quickly went beyond the expressibility of both KL-ONE and NIKL, and finally resorted to a standard first order language. This experience, along with others, motivated me to enlarge my interests beyond terminological systems, see Positions 2 and 3 below. Also, a recent paper of mine [10] argues for embracing n-ary terms in terminological systems, see Position 1 below.

Position 1: Most terminological KR systems restrict the arity of terms that are directly representable to 1 or 2, and this restriction is unnecessary. Most constructs that exist in these terminological KR languages can easily be generalized to constructs that support terms of any arity \( \geq 0 \). In doing so, one gains clarity and simplicity in the representation of terms whose arity exceeds 2, and one avoids a certain assertional "clutter" for these terms. Moreover, the cost of direct representation of n-ary terms appears to be between zero and very little. A full presentation of my case appears in [10], so I will not discuss it further here.
Position 2:

Terminological systems must fit within larger representational frameworks. It is therefore time to assess how well previous and current KR systems have integrated terminological, (grounded) propositional, equational, rule-based and other representations and reasoners. Moreover, it is a good time to set goals for future work in this area. Even though the workshop is focused on terminological systems, it is still important to address the larger context in which such systems are placed and used. I suggest that we take stock of systems such as Rhet [13], Cake [14] and others and consider their successes in meeting general representational needs. I also suggest that we analyze them with respect to my third position.

Position 3:

We must re-examine certain popular assumptions regarding the role of a KR system within a larger knowledge-based (KB) system. Specifically, we must abandon the notion that the services provided by a KR system depend only on the truth theory of the representation language (as urged in [15]). By abandoning this notion, we allow KR systems to reason both unsoundly and incompletely (which is urged by many researchers, including Levesque [19]). Moreover, we are forced to move away from the view that a KR system is a black box with a simple functional interface as found in Krypton [16]. As a consequence, I argue for the development of "expandable hybrid systems", which is a slight twist on current hybrid systems. (1) The representational language should be widely expressive in order to include (nearly) all of what users need to represent. (2) The system should smoothly integrate expansions in either the representation language or in the suite of inferential algorithms the system performs. Regarding (1), the inability of the KR system to draw inferences using a given type of construct should not necessarily prohibit its inclusion in the representation language (although this will surely have some influence). Item (2) is intended to allow for expansions in the KR system without affecting the form of one's representation at the "knowledge level" [17]. I do not minimize the difficulty of (2), but I do urge that future extensions be planned for as much as possible. These points will be elaborated below.

In [15], Brachman and Levesque argue that a KR system should (A) be a utility that is not available for inspection or control (i.e., a black box), and
(B) answer queries based on entailment (i.e., if the system is asked whether 
"x" is true, it should say "yes" if and only if the current knowledge base 
entails "x"). As a result, for a KR system to be correct, entailment must be 
computed. Algorithms for determining entailment, hence, must be tractable, 
and so Brachman and Levesque argue that the languages of KR systems 
should be limited. Of course, they recognize that the needs for representing 
partial knowledge seem to demand languages for which entailment is not 
tractable. They hope "to integrate limited forms of languages and reasoning, 
with the goal of forging a powerful system out of tractable parts."

While the work that Brachman and Levesque present and suggest for the 
future is important for KR, more progress can be made by relaxing both (A) 
and (B). I begin by assuming that (i) in general, KB systems need widely 
expressive representation languages — languages in which entailment is not 
tractable. It would be nice if this was not the case, but so far the evidence 
is overwhelming (e.g., see [15] and [18]). Further, (ii) a KR system should 
(ttry to) meet all of the representational needs of a KB system. Let's face it, 
users will do what they need to, the only question is whether or not the KR 
system helps them do it. Yet, (iii) the KR system should be principled in that 
there is an overall truth theory for the representation language and that all 
inferences drawn should be understandable with respect to that theory. Note 
that I specifically do not require that all inferences be sound (though many 
will be), nor that the set of inferences drawn be complete (though certain sub-
languages may have the complete entailment relation computed). Further, 
(iv) the performance of the inferential algorithms should be predictable to 
whatever extent is possible. Thus, decidability results and, when appropriate, 
complexity measures should be available for each of the inferential algorithms 
supported. For example, deduction is understandable and, depending on the 
language, its computation may or may not be predictable (i.e., may or may 
not be tractable). Methods for combining evidence in expert systems are 
understandable (though not necessarily deductive) and the performance of 
such combination methods is predictable.

Note that I have relaxed (B), and in doing so, have proposed a KR system 
that could be dangerous to use because it could easily fall off the computa-
tional "cliff." One solution ("approach" might be a better word) is to relax 
(A) and to open up the KR black box for KB systems' users to control. For 
now, I suggest only a little of this manual control (though time and experience 
may change that). Imagine a hybrid KR system with a single, uniform repre-
sentation language. When a user "tells" the KR system something, the user can also instruct the system as to which sub-systems to store it in, and/or which of a set of alternative internal representations to use. When a user "asks" the KR system something, the user can also instruct the system to use certain sub-systems, and/or to use certain inferential algorithms, and/or to use a certain amount of resources. In such a situation, there are a number of possible answers to a query such as "always" (valid), "never" (unsatisfiable), "sometimes" (satisfiable), "don't know" (can't be determined using the given algorithms), "out of resources", or combinations of the preceding (e.g., it's not "always" but don't know whether it is "never" or "sometimes"). Of course, the KR system will need reasonable default behavior for cases in which a user does not specify sub-systems or algorithms or resource limits.

In summary, my third position is that KR systems should embrace the representational needs of KB systems as much as possible. This means having representational languages where entailment is not tractable. To deal with this, hybrid systems should embrace many limited inference sub-systems which may or may not be sound and may or may not complete. They should embrace intractable inferential algorithms for users who need them and are willing to pay for them. The key is to expand the interface to KR systems by offering more control to users and to construct expandable hybrid systems that can incorporate future representational sub-systems. With this perspective, current KR systems that do not compute complete entailment are not failures, as argued in [15], but are examples of limited successes. Further, this perspective embraces all previous work involving limited inference for KR systems and it builds in the ability to support and assimilate future work in this area.

Conclusions:

Terminological KR systems have interesting and difficult problems in and of themselves (e.g., my first position), and no doubt the workshop should and will attend to some of them. However, it is clear that many current terminological KR systems fall far short of the representational needs of KB systems, and so I urge consideration of the larger contexts I have presented in either my second or third positions.

Acknowledgements:
These ideas have been stewing for several years. However, a recent conversation with Jon Doyle and Ramesh Patil brought these thoughts to a light that was clear enough for me to write about.

References:

* Items 1-12 constitute my bibliography.


Association for Artificial Intelligence, St. Paul, Minn., August 1988.


