STATEMENT OF INTEREST

Manolis Koubarakis
Department of Computer Science
University of Toronto
Toronto, Ontario, M5S 1A4
koubarak@ai.toronto.edu

1989

In [KMSB89] the knowledge representation language Telos is presented. Telos treats individuals and attributes in a uniform way and emphasizes the structuring principles of classification, specialization and aggregation. It also offers an assertion language which can be used to introduce domain specific constraints and rules in the knowledge base. Finally, to allow the user to represent and reason with respect to temporal knowledge, Allen’s interval calculus [AH85] is tightly integrated into Telos’ representational framework.

The form of specialization employed by Telos is rather simple compared to the subsumption mechanism of languages in the KL-ONE family (e.g., KRYPTON [BFL83] or CLASSIC [BBML89]). Classes can be specialized and therefore inherit attributes of their superclasses. The transitive closure of the Isa graph gives us all the specialization relationships in the knowledge base. However, when rules having the predicate isa in their head are allowed in the language then sufficiency conditions for specialization relationships can be expressed. Thus although specialization is built into the language, one can assert conditions under which two classes are Isa related. Our proposal lies in between systems like KL-ONE, CLASSIC or KRYPTON and systems based on predicate logic which treat Isa as just another predicate. The systems in the former category are haunted by the fact that the complexity of the subsumption operation can easily get out of hand while the systems in the latter category have none of the advantages which an inheritance
mechanism can provide. According to our proposal, a minimum of necessary specialization machinery must be built into the language. Then the user is the one who will decide whether he wants to have classes to be IsA related whenever certain conditions hold1.

In contrast with languages in the KL-ONE family, Telos makes heavy use of the classification dimension: objects2 can be classified as tokens, simple classes, meta-classes, meta-meta-classes and so on. The classification dimension is also exploited for integrity enforcement: attributes of objects should belong to appropriate classes (i.e., be of appropriate type) but must also satisfy other domain specific constraints attached to the classes where they belong.

I am interested in investigating the query processing problem for Telos. Up to now, knowledge representation languages have relied on a simple TELL/ASK interface for communication with the knowledge base3. I believe that this view of knowledge bases is fundamental but more elaborate query languages are needed. Research in database systems is the best place to look for intuition as to how to proceed from the existing simple query languages. Query language designers for database systems usually try to meet the following goals:

- Query languages should be declarative (preferably).
- Query languages should assume a computational model based on bulk operations (e.g., the unit of communication between the data base and the query processor should not be a single tuple of objects satisfying a predicate but rather the set of all the satisfying tuples).
- Query expressions can be composed. For example, the result of a query on a set of relations should be a relation itself. This relation might then be used in the creation or update of another relation and so on.
- Query language constructs should bear a natural relation to the data model assumed by the data base (in our case, the query language should

1 And it is perfectly reasonable that he might decide not to do so if these conditions are very complex.
2 I.e., individuals or attributes.
3 This has been the case for Telos as well.
have natural ways of capturing the constructs supported by the representational framework: complex objects with unique identity, classification, specialization, and time).

- Queries should be computable.

I am currently working on developing a query language in the style of DATALOG [UL88] or LDL [Zan88] trying to meet as many of the above goals as possible. The work on object models and languages by Abiteboul and Beeri (see [Bee89], [AB88]) and Kifer (see [KW89], [KL89]) has been a significant source of inspiration for my work so far. After developing the query language for Telos, I will continue with a study of query optimization. In this workshop, I would like to discuss my research with people who face similar or related problems for term subsumption systems.

References


