The Role of Ontologies in the Verification and Validation of Knowledge Based Systems

Trevor J.M. Bench-Capon
Department of Computer Science
The University of Liverpool
Liverpool,
England

Abstract

In this paper I give some preliminary examination of the ways in which an ontology - an explicit specification of the conceptualisation of the domain - can support the verification and validation of a knowledge based system. The discussion is focussed on a simple, well known, example relating to the identification of animals. Key elements of the support provided by the ontology relate to attempting to give coherence to the domain conceptualisation; making the role of experts in verification and validation more structured and less at the mercy of interpretation,; constraining the number of test cases required to give good coverage of the possible cases; and structuring the testing to give better assurance of its efficacy, and a possible basis for greater automation of the testing process. Finally I make some brief remarks on the relation between the ontology of a knowledge based system, and a database and its schema.

1. Introduction

Ontologies, best characterised as "the explicit specifications of the conceptualisation of a domain" (Gruber 1995), have, in recent years, made a significant impact on thinking about the design and development of knowledge based systems (KBS). Typically a number of advantages are said to result from the use of ontologies including:

- Facilitating sharing of knowledge between systems;
- Facilitating reuse of knowledge in new systems;
- Aiding knowledge acquisition; and,
- Improving the verification and validation of knowledge based systems.

Much has been written about the first three of these topics, but as yet little detail has been advanced with respect to the fourth topic. In this paper I shall attempt to outline some of the things that ontologies can do for verification and validation.

I shall do this by considering an example. The example I shall use is *ZOOKEEPER*, a very simple rule base described in Winston's AI textbook (Winston 1992). There are several reasons for choosing this example: it is small enough that the complete system can be given in a short paper; it deals with a domain familiar to all; and the example itself is very well known, and for many people is their first encounter with a rule base. Section 2 will recapitulate this rule base. In section 3 I will discuss how the rule base as it stands might be verified and validated. In section 4 I will give an ontology for the domain, and in section 5 I will show some additional possibilities for verification and validation that this allows. In section 6 I will give some concluding remarks.

2. A Toy Animal Classification System

The rulebase for ZOOKEEPER is given in Winston (1992), page 121-4. It is explicitly limited to the identification of seven animals: a cheetah, tiger, giraffe, ostrich, penguin and an albatross. It has 15 rules, enabling identification of these seven animals, often in several ways, to allow for some observations being unobtainable.

The rules (expressed here in Prolog form) are:

```
Z1: mammal(X):-hair(X) .

Z2: mammal(X):-givesMilk(X) .

Z3: bird(X):-feathers(X) .

Z4 bird(X):-flies(X),
         laysEggs(X).

Z5: carnivore(X):- mammal(X),
               eats(X,meat).
```

```
Z6: carnivore(X):- mammal(X),
               teeth(X,pointed),
               has(X,claws) ,
             eyes(X,forwardPointing).

Z7: ungulate(X):- mammal(X),
             has(X,hoofs).

Z8: ungulate(X):- mammal(X),
              chewsCud(X).

Z9: cheetah(X):-carnivore(X) ,
            colour(X,tawny),
            spots(X,dark) .

Z10: tiger(X):-carnivore(X),
          colour(X,tawny),
          stripes(X,black).

Z11: giraffe(X):-ungulate(X),
                legs(X,long),
               neck(X,long),
               colour(X,tawny),
            spots(X,dark).

Z12: zebra(X):-ungulate(X),
          colour(X,white),
          stripes(X,black).

Z13: ostrich(X):-bird(X),
            not flies(X),
              legs(X,long),
               neck(X,long),
        colour(X,blackandwhite).

Z14: penguin(X):-bird(X),
            swims(X),
            not flies(X),
        colour(X,blackandwhite).

Z15: albatross(X):- bird(X),
               flies(X,well).
```

These rules can be used either to identify an animal given a set of observations, or to test a hypothesis that an animal is of a particular species. Now let us consider how we might go about verifying and validating *ZOOKEEPER*.

3. Verifying and Validating ZOOKEEPER

Here I shall use Boehm's well known distinction between verification and validation (Boehm 1981):

- Verification "Are we building the product right?"
- Validation: "Are we building the right product?"

As usually interpreted in the context of knowledge based systems, verification would include checking that the rule base is structurally sound (free of subsumed rules, contradictions and dead end rules and the like), while validation would be effected by supplying sets of typical observations and checking that identification was possible and correct. Additionally we might present the rules to an expert and ask for confirmation of their correctness.

It is probable that if we were to run such tests, ZOOKEEPER would pass them quite well. There appear to be no structural problems, each rule looks plausible enough to accept on inspection, and appropriate sets of facts will lead to the correct answers.

The only problems would arise through an inability to respond to certain sets of observations: there are certain sets of observations which would not give an answer (such as a white carnivore); and there is a need to give more information than is strictly necessary (such as the orientation of eyes as well the pointedness of the teeth). Both of these possible defects may, however, be acceptable: since we are limited to seven animals, no white carnivores will be observed, and Winston argues for the inclusion of extra information on the grounds that "there is no need for information in rules to be minimal. Moreover, antecedents that are superfluous now may become essential later as new rules are added to deal with other animals" (Winston 1992, p123).

So shall we conclude that the rule base is entirely satisfactory? I don't think we should, particularly if we are going to take seriously the possibility of extending the system to cater for, possibly a good many, more animals.

What I am suggesting is that "building the system right" needs to encompass more than a simple absence of structural defects. What we want, in addition, is some kind of *conceptual coherence* to the representation. In reaching the final rule base distinctions were proliferated as and when they were needed in order to discriminate amongst the seven particular animals, and without much regard for distinctions that had already been made. If a system is to be built correctly, it should make principled distinctions, and make them in a justifiable manner. For example, spots are "dark" and stripes are "black". Do we want a distinction between "dark" and "black"? What other varieties of spots and stripes might there be? Is there really a good difference between being white in colour with black stripes, black in colour with white stripes and whiteandblack in colour?

We also need the observations to be relatively easy to obtain. Some of those required by this rule base need judgement to be applied. A particular example of this is the requirement be that an albatross flies “well”. This might well raise differences of opinion and interpretation. Others are rather hard to obtain: “lays eggs” is an occasional thing which might be hard to observe (and not observable at all in the case of a male of the species).

Much of the problem derives from the failure to initially conceptualise the domain in a coherent fashion. The strategy is first to classify an animal as a mammal or a bird, then sub-divide mammals into carnivores and ungulates, and then to discriminate members of these categories in terms of some observable features which are indicative of the particular animals in the collection. The higher level distinctions are theory driven, and the rules are determined by theory: for example Z4 is justified on the grounds that “some mammals fly and some reptiles lay eggs, but no mammal or reptile does both” (Winston, p122). But in the context of use of the system, Z4 is applicable only to the albatross, since the other two birds are flightless, and if it can fly it is an albatross, so its oviparity is neither here nor there. On the other hand, if we were to take the notion of extensibility seriously Z9 would be inadequate since it describes leopards and jaguars as well as cheetahs. As it stands here the rules are defective, with respect to the standards of a well constructed system because they derive from conflicting conceptualisations of the domain, and conflicting ideas of how the system will be used.

The problems above of course derive from the lack of a clear specification as a starting point. Viewed simply from the standpoint of its real use, as an example rule base to illustrate forward and backward chaining, it is adequate. It is only when we project it into standing as a real application that we would need to specify whether it was supposed to identify only seven or an indefinite range of animals; whether it is meant to incorporate known theory about animal classification, or to restrict itself to what can be seen; what kind of judgements the user of the system can be expected to make, and the like.

In the next section I will provide a crude ontology for a system which intended to identify animals on the basis of observations made by a non-expert, and which intended to cover the seven animals given, but also to be extensible to other animals.

4. An Ontology for ZOOKEEPER

I shall use as the foundation for the ontology the observations that can be identified from the *ZOOKEEPER* rules. We can identify the following observable predicates:

1) has hair
2) gives milk
3) has feathers
4) flies
5) lays eggs
6) eats meat
7) long legs
8) long neck
9) tawny colour
10) dark spots
11) white colour
12) black stripes
13) black and white colour
14) swims
15) flies well
16) pointed teeth
17) claws
18) eyes point forward
19) hoofs
20) chews cud

Some of these seem to present alternatives, so we can group them accordingly.

1) skin covering {hair,feathers}
2) colour{white, tawny,black and white}
3) markings{spots,stripes}
4) movesBy{swims,flies}
5) feet{hoofs,claws}

In other cases there seem to be implicit alternatives:

1) teeth{pointed,?}
2) eats{meat,?}
3) legs{long,?}
4) neck{long,?}
5) stripes{black,?}
6) spots{dark,?}
7) flies{well,?}
8) eyes{point forward,?}

In other cases we have only a true or false decision:

1) gives milk
2) lays eggs
3) chews cud

We now need to perform some rationalisation on this; for example flying and swimming are not exclusive, so these predicates must be separated. Moreover, flying appears to be a qualitative thing rather than a simple boolean. We can make the markings and colour situation more coherent by saying that an animal has a basic colour, and markings, which may be lighter or darker than the basic colour. Where we have gaps, these need to be filled.

We could now arrive at the situation where we can identify the attributes, and the possible values they can take, shown in Table 1. This will provide us with a well defined vocabulary with which to construct a set of rules.

To complete the ontology we need to add some axioms, stating combinations which are impossible. For example:

A1 Not (eats meat and chews cud)
A2 Not (Material feathers and chews cud)
A3 Not (Pattern none and shade not n/a)

Coat:
  Material {hair,feathers}
  Colour {white,tawny,black}
  Markings :
    Pattern{spots,stripes,irregular,none}
    Shade {light,dark,n/a}

Facial Features
  Eyes {forward,sideways}
  Teeth{pointed,rounded}

Feet {claws,hoofs}

Flies{no,poorly,well}

Eats{meat,plants,both}

Size:
  Neck{long,normal}
  Legs{long,normal}

Gives Milk {true,false}
Lays Eggs {true,false}
Chews Cud {true,false}
Swims {true,false}

Table 1: Attributes and Values for ZOOKEEPER

We could now construct a table of our example animals, and fill in the values for the predicates where they have been given. Where there are gaps, we need to find the information required to complete them. This may itself identify additional possibilities for some of the attributes (for example birds do not have teeth). Table 2 is such a table. (Some of the answers are conjectural - I am not an expert).

Predicate	Cheetah	Tiger	Zebra	Giraffe	Ostrich	Penguin	Albatross
Material	hair	hair	hair	hair	feathers	feathers	feathers
Colour	tawny	tawny	white	tawny	black	black	white
Pattern	spots	stripes	stripes	spots	irregular	irregular	none
Shade	dark	dark	dark	dark	light	light	n/a
Eyes	forward	forward	sideways	sideways	sideways	forward	sideways
Teeth	pointed	pointed	rounded	rounded	none	none	none

Feet	claws	claws	hoofs	hoofs	toes	toes	toes
Neck	normal	normal	normal	long	long	normal	normal
Legs	mormal	mormal	normal	long	long	short	normal
Gives Milk	true	true	true	true	false	false	false
Flies	no	no	no	no	no	no	well
Eats	meat	meat	no	no	yes	yes	yes
Lays Eggs	no	no	no	no	true	true	true
Chews Cud	no	no	true	true	no	no	no
Swims	no	yes	yes	no	false	true	no

Table 2: Attributes of Animals in *ZOOKEEPER*

5. Using the Ontology in Validation and Verification

We now have an ontology which we can use to verify and validate a knowledge base built on it.

The role of the expert is changed significantly. The expert no longer examines rules, but instead the vocabulary, and the table of attributes. With respect to the vocabulary the expert should check:

- that the attributes represent sensible distinctions
- that the values are exclusive
- that the values are exhaustive

The point about values can be addressed from two standpoints: either from the point of view of the existing collection, or from the point of view of a potentially extended collection. The first will indicate what is needed to test the rule base against its current operation, and the other will provide an indication of its extensibility.

Also, to facilitate testing the expert should indicate whether observations are always available, or only sometimes available.

The expert might modify Table 1 to give Table 3. Here always observable attributes are indicated in bold, as are values required by the current seven animals.

Coat:
Material {**hair,feathers**,scales}
Colour {**white,tawny,black,**grey,russet}
Markings :
Pattern{**spots,stripes,irregular,none**}
Shade {light,**dark, n/a**}

Facial Features
Eyes {**forward,sideways**}
Teeth{**pointed,rounded,none**}

Feet {**claws,hoofs,toes**}
[Comment: feet are hard to observe (Winston)]

Flies{**no**,poorly,**well**}

Eats{**meat,plants**,both}
[Comment: meat includes fish]

Size:
Neck{**long,normal**}
Legs{**long,normal**}

Gives Milk {**true,false**}
Lays Eggs {**true,false**}
Chews Cud {**true,false**}
Swims {**true,false**}
[Comment: could have {no,poorly,well}]

Table 3: Validated Attributes and Values for ZOOKEEPER

The expert should also examine the table of attributes (Table 2), to confirm that these entries are correct. The table can be further verified by ensuring that it does not conflict with any of the axioms. Under this scheme the role of the expert is much more well defined, and more systematic so that there is less possibility of interpretation allowing errors to go unnoticed.

We can now use this information to test the rule base. In the original *ZOOKEEPER* there were 20 predicates each of which appeared capable of being true or false independently, giving more than a million possible combinations. Assuming exhaustive testing to be impossible, test data was selected by using plausible combinations, but there was no system about the generation of these, and so coverage was not ensured. If, however, we confine ourselves to testing only the attributes always available as observations, and only the values actually used by our current collection, we have only 1152 combinations. These can be further pruned by using the axioms of the ontology; for example by excluding all combinations where pattern is none and shade is not n/a. This is a very significant reduction, particularly in view of the fact that the correctness of the answers can be decided by reference to Table 2. Thus the ontology provides the essential input for an automated test harness.

Testing against these cases may identify cases where the system produces:
1) an incorrect answer;
2) multiple answers;
3) no answer.

Case (1) requires amendment to the offending rule. Case (2) indicates either that some rule must be made more specific (possibly using not always available features), or - potentially - that the current ontology is inadequate and requires another predicate to discriminate the cases. Case (3) requires expert inspection: it may be that:

- there should be an axiom in the ontology excluding such cases; or
- that animals exist satisfying this set of attributes, but are not in the collection, in which case the rule base is correct, and the combination should not occur in practice; or
- that identification is reliant on some not always available feature. For example the identification of the albatross turned on its flying power, which might not be observable.

The third case is most problematic. If removing the offending antecedent creates case (1) or case (2) problems, then we have to reconcile ourselves to a certain incompleteness, or find some always available discriminating observation. Case (2) will drive us to *introduce* antecedents relating to intermittently observable features, whereas case (3) may motivate us to *remove* them.

In addition to these possibilities for verification and validation against the ontology, normal structural checks should, of course, be applied. The quasi-random testing for validation is, however, unnecessary given the more structured approach permitted by the ontology.

6. Conclusions

In this paper, I have used a simple example to indicate how the availability of an ontology can aid verification and validation. For verification, we enable the expert to check the vocabulary in a structured manner, and in a way which discriminates between currently needed information and information which will permit some straightforward extensions to the collection of animals. For validation we have shown how the testing process can be structured using an ontology.

Since the integration of knowledge based and database systems is becoming increasingly topical we may also make some remarks here on the relation between databases and ontologies. Table 2 is effectively a database recording the attributes of a test set. Table 3 corresponds well with a database schema with its identification of attributes and domains for those attributes. Thus if we are founding a knowledge based system on an existing database, much will be there - particularly with regard to the attributes we categorised as always available. Some difference between the attributes required by a database and a KBS may be motivated by the expert’s conceptualisation, or by the need for intermediate predicates to support problem solving, but the database schema should provide an excellent starting point.

To summarise the thrust of this paper: ontologies can help drive verification and validation by;
- allowing the expert to inspect the distinctions made rather than make judgement calls on the rules, permitting a greater assurance of conceptual coherence in the knowledge base,;
- providing a means of structuring testing;
- suggestion appropriate responses to flaws indicated by testing.

References

Boehm, B.W., (1981). *Software Engineering Economics*. Prentice Hall.

Gruber, T.R., (1995).*Towards Principles for the Design of Ontologies Used for Knowledge Sharing*. International Journal of Human-Computer Interaction, Vol 43, pp 907-928.

Winston, P.H, (1992). *Artificial Intelligence*, Third Edition. Addison Wesley: Reading, Mass