The Web Is Not Well-Formed

Guus Schreiber, *University of Amsterdam*

The debate about what a Web ontology language should look like is reminiscent of past neat–scruffy struggles. Knowledge modelers want expressiveness, logicians stress decidability. The main difference is that the Semantic Web actually forces us to make some choices: there is a strong need for real-world knowledge representation.

Expressivity requirements

The people arguing for an expressive language have a strong case. For example, suppose a person wants to find images of red apes on the Web. Most photos of orangutans (which generally have a red-orange color) will satisfy this query. However, you can't expect the indexer of every orangutan photo to explicitly state the ape's color (this also leads to unwanted inter-indexer variability). You really want them to annotate the photo with the class *orangutan*, and possibly only specify the color if it is not red or orange (old animals can be brown or gray, some orangutans are albino, and so forth). This requires the ability to express default knowledge, a language property logicians are not very fond of because it requires nonmonotonic reasoning. However, if specification of default color values is disallowed, we will not be able to retrieve a significant number of relevant photos. Actually, making this match is what the Semantic Web is all about (as it enables a match between a query *red ape* and a photograph that is annotated with neither of these terms). Also note that from the search perspective, 100-percent correctness (precision) is not needed, just a sufficiently high percentage.

In the same animal domain, we find another example of a frequently occurring form of knowledge that is difficult to capture in description logic—namely, the fact that there is in practice no hard borderline between instances and classes. If we look in a biology book at the definition of an orangutan, we will find something like this:

Orangutan

Latin name:	*Pongo pygmaeus*
kingdom:	Animalia
phylum:	Chordate
class:	Mammalia
order:	Primates
family:	Hominidae
genus:	Pongo

From the viewpoint of the biological taxonomy of species, an orangutan is an instance of a species class, while at the same time it represents a collection of animal instances. *Orangutan* can thus be considered both a class and an instance. Note that specifying orangutan as a subclass of species (and defining the values above as slot-value restrictions) is incorrect. An individual orangutan is not an instance of species (it is not an animal type).

The methodological guidelines should encourage user groups to agree on a small set of metaclasses to handle their particular expressivity requirements, which fall outside the DL core.

This notion of classes as instances of (meta)classes comes up during conceptual modeling of almost any domain with some degree of complexity. Another example is a Boeing 747, which denotes both a collection of individual aircrafts but also is itself a member of a collection of aircraft types, with other members such as Airbus 310. Both interpretations are needed to semantically annotate Web pages of aircraft industry.

The metaclass mechanism

An ontology language on top of RDF Schema only makes sense if it introduces some formal semantics to the ontology definitions. Undoubtedly, description logic provides a well-researched basis for such a semantics. Subsumption, which forms the basis of description logic, is a natural way for people to express domain knowledge. However, if we just ignore expressivity requirements, people will simply not use the Web ontology language, and the whole effort will become a failure. Logic is an ideal, well-formed world—the Web is not.

To cater to this, the language should have an extendible metalevel, which enables users to describe additional interpretations of classes and properties. The *class-as-instance* mechanism can actually fulfill this role (as it does in RDF Schema). In addition, the Web ontology language should provide methodological guidelines for using the base language plus the metalevel mechanism in an appropriate manner. For example, one can solve the default problem by defining a class *orangutan* with necessary property restrictions and, in addition, a subclass such as *archetypical orangutan* to define default property restrictions. This leads to correct logical interpretations by DL reasoners. Subsequently, we can define a metaclass such as *archetype* and make the subclass an instance of this class. This enables index and search programs to treat this class in a special manner.

Of course, the metaclass mechanism could potentially open a can of worms. If used only in an ad hoc fashion, it will lead to messy ontologies. The methodological guidelines should encourage user groups to agree on a small set of metaclasses to handle their particular expressivity requirements, which fall outside the description logic core. In the future, we might even want to define standards for metaclasses to be used. ■

Acknowledgments

This essay benefited from discussions with other WebOnt Group participants and with various colleagues in Amsterdam. The author is supported by the ICES-KIS project Multimedia Information Analysis, funded by the Dutch government and by the IST Project IBROW funded by the European Union.