4.2.4 Reasoning Support

Formal semantics is a prerequisite for *reasoning support*. Derivations such as the preceding ones can be made mechanically instead of by hand. Automatic reasoning is important because it allows us to check the correctness of the ontology. For instance:

- check the consistency of the ontology
- check for unintended relations between classes
- check for unintended classifications of instances

Automated reasoning support allows one to check many more cases than could be checked manually. Checks like the preceding ones are extremely valuable for designing large ontologies, for cases where multiple authors are involved, and for integrating and sharing ontologies from various sources.

We can provide formal semantics and reasoning support to an ontology language by mapping it to a known logical formalism, and by using automated reasoners that already exist for those formalisms.

It is clear that we need an ontology language that is richer than RDF Schema, a language that offers these features and more. In designing such a language one should be aware of the trade-off between expressive power and efficient reasoning support. Generally speaking, the richer the logical formalism, the less efficient the reasoning support becomes, often crossing the border of decidability; that is, reasoning on such logics is not guaranteed to terminate. We therefore need a compromise, a language that can be supported by reasonably efficient reasoners, while being sufficiently expressive to represent a large variety of knowledge.