(d) Find out how to remotely call the inference engine to be used.

Proof Layer

The aim of these projects is to realize a proof layer, the vision of which was briefly outlined in chapter 1. Note that there is not one proof layer, but rather a proof layer for each selected Semantic Web reasoning system (logic). Still, some considerations are common to all such systems. Two possible logical systems for which one could implement a proof layer are

- a simple monotonic rules language, such as Datalog (Horn logic without function symbols), for which the reasoning tool Mandarax[10] could be used;
- a nonmonotonic rules language, as discussed in this chapter, for which DR-DEVICE[11] or DR-Prolog[12] could be used.

3. This basic project can be done by two or three people in about two months. The aim is to develop an interactive system that provides explanation to the user. Important aspects to be addressed include the following:

(a) Decide how to extract relevant information from the overall proof trace. You could consult the automated reasoning and logic programming literature for ideas.

(b) Define levels of granularity for representing proofs. Should whole proofs be shown, or only metasteps? These could then be refined if the user questions a certain step.

(c) Ultimately, the "leaves" of a proof will be RDF facts, rules, or inference conditions used.

[10] mandarax.sourceforge.net/.
[11] lpis.csd.auth.gr/systems/dr-device.html.
[12] www.csd.uoc.gr/~bikakis/DR-Prolog/.