2.8 A Direct Inference System for RDF and RDFS

As stated, the axiomatic semantics detailed in section 2.7 can be used for automated reasoning with RDF and RDF Schema. However, it requires a first-order logic proof system to do so. This is a very heavy requirement and also one that is unlikely to scale when millions (or billions) of statements are involved (e.g., millions of statements of the form $Type(?r, ?c)$).

For this reason, RDF has also been given a semantics (and an inference system that is sound and complete for this semantics) directly in terms of RDF triples instead of restating RDF in terms of first-order logic, as was done in the axiomatic semantics of section 2.7.

This inference system consists of rules of the form

IF	E contains certain triples
THEN	add to E certain additional triples

(where E is an arbitrary set of RDF triples).

Without repeating the entire set of inference rules (which can be found in the official RDF documents), we give here a few basic examples:

IF	E contains the triple $(?x, ?p, ?y)$
THEN	E also contains the triple $(?p, \texttt{rdf} : \texttt{type}, \texttt{rdf} : \texttt{property})$

This states that any resource $?p$ that is used in the property position of a triple can be inferred to be a member of the class rdf:Property.

A somewhat more interesting example is the following rule:

IF	E contains the triples $(?u, \texttt{rdfs} : \texttt{subClassOf}, ?v)$
	and $(?v, \texttt{rdfs} : \texttt{subclassOf}, ?w)$
THEN	E also contains the triple $(?u, \texttt{rdfs} : \texttt{subClassOf}, ?w)$

which encodes the transitivity of the subclass relation.

Closely related is the rule