The OWL2 RL rules can be categorized in four (non-disjoint) categories: triple pattern rules, inconsistency rules, list rules, and datatype rules.

Triple Pattern Rules These rules derive certain RDF triples from a conjunction of RDF triple patterns. Translation of these rules to RIF (using Frame formulas) is straightforward using rules of the form:[4]

```
Group(
    Forall ?V1 ... ?Vn(
        s[p->o] :- And(s1[p1->o1]... sn[pn->on]))
)
```

Inconsistency Rules Such rules indicate inconsistencies in the original RDF graph (of course w.r.t. the existing OWL knowledge). These rules can be easily represented in RIF as rules with conclusion *rif:error*, a predicate symbol within the RIF namespace that can be used to express inconsistency. For example, an inconsistency occurs when two predicates have been declared to be disjoint, but connect the same entities. This can be expressed in RIF as follows:

```
Group(
    Forall ?P1 ?P2 ?X ?Y(
        rif:error :- And(
            ?P1[owl:propertyDisjointWith ?P2] ?X[?P1->?Y] ?X[?P2->?Y]))
)
```

List Rules A number of OWL2 RL rules involve processing OWL expressions that include RDF lists (for example *owl:AllDifferent*). Two approaches are possible to express these rules in RIF. One may use recursive rules to traverse RDF graphs at run time, yielding a uniform representation. Or one may take a preprocessing approach

[4]For improved readability, these rules are given in Prolog (backward) notation, instead of the If-Then (forward) notation used so far.