Their typing information is

$$PropVal(Subject, ?st, ?r) \longrightarrow$$
$$(Type(?st, Statement) \wedge Type(?r, Resource))$$

$$PropVal(Predicate, ?st, ?p) \longrightarrow$$
$$(Type(?st, Statement) \wedge Type(?p, Property))$$

$$PropVal(Object, ?st, ?v) \longrightarrow$$
$$(Type(?st, Statement) \wedge (Type(?v, Resource) \vee Type(?v, Literal)))$$

The last axiom says, if $Object$ appears as the property in an RDF statement, then it must apply to a reified statement and have as its value either a resource or a literal.

Containers

All containers are resources:

$$Type(?c, Container) \longrightarrow Type(?c, Resource)$$

Containers are lists:

$$Type(?c, Container) \longrightarrow list(?c)$$

Containers are bags or sequences or alternatives:

$$Type(?c, Container) \longleftrightarrow$$
$$(Type(?c, Bag) \vee Type(?c, Seq) \vee Type(?c, Alt))$$

Bags and sequences are disjoint:

$$\neg(Type(?x, Bag) \wedge Type(?x, Seq))$$

For every natural number $n > 0$, there is the selector _n, which selects the nth element of a container. It is a functional property