1.1.3 Basic Technology for the Semantic Web

The aforementioned three design principles have been translated into actual technology, and much of this book will be devoted to describing just that technology:

1. use *labeled graphs* as the data model for objects and their relations, with objects as nodes in the graph, and the edges in the graph depicting the relations between these objects. The unfortunately named "Resource Descripion Framework" *RDF*[3] is used as the formalism to represent such graphs.

2. use *web identifiers (Uniform Resource Identifiers - URI)* to identify the individual data-items and their relations that appear in the datasets. Again, this is reflected in the design of RDF.

3. use *ontologies* (briefly: hierarchical vocabularies of types and relations) as the data model to formally represent the intended semantics of the data. Formalisms such as *RDF Schema* and *The Web Ontology Language* (OWL) are used for this purpose, again using URIs to represent the types and their properties.

1.1.4 From Data to Knowledge

It is important to realize that in order to really capture the intended semantics of the data, a formalism such as RDF Schema and OWL are not just data-description languages, but are actually lightweight *knowledge representation* languages. They are "logics" that allow the inference of additional information from the explicitly stated information. RDF Schema is a very low expressivity logic that allows some very simple inferences, such as property inheritance over a hierarchy of types and type-inference of domain and range restrictions. Similarly, OWL is somewhat richer (but still relatively lightweight) logic that allows additional inferences such as equality and inequality, number restrictions, existence of objects and others. Such inferences in RDF Schema

[3] Perhaps "Rich Data Format" would be a better name.