

7.7.3 Knowledge Maintenance

Besides basic storage and retrieval functionality, a practical Semantic Web repository provides functionality for managing and maintaining the ontology: change management, access and ownership rights, and transaction management.

Besides lightweight ontologies that are automatically generated from unstructured and semistructured data, there must be support for human engineering of much more knowledge-intensive ontologies. Sophisticated editing environments can be used to retrieve ontologies from the repository, allow a knowledge engineer to manipulate them, and place them back in the repository.

7.7.4 Knowledge Use

The ontologies and data in the repository are to be used by applications that serve an end user. We have already described a number of such applications. In addition, external applications may access the knowledge through the exposure of data through one or all of the following approaches: a SPARQL endpoint, Linked Data, or RDFa.

7.7.5 Applying the Architecture

In the On-To-Knowledge project,²⁴ the architecture of figure 7.1 was implemented with very lightweight connections between the components. Syntactic interoperability was achieved because all components communicated in RDF. Semantic interoperability was achieved because all semantics was expressed using RDF Schema. Physical interoperability was achieved because all communications between components were established using simple HTTP connections, and all but one of the components (the ontology editor) were implemented as remote services. When operating the On-To-Knowledge system from Amsterdam, the ontology extraction tool, running in Norway, was given a London-based URL of a document to analyze; the resulting RDF and RDF Schema

²⁴<http://www.ontoknowledge.org/>.