Architecting a Search Engine for the Semantic Web

David E. Goldschmidt and Mukkai Krishnamoorthy

Rensselaer Polytechnic Institute
Troy, New York, USA
{goldsd, moorthy}@cs.rpi.edu

Abstract

Since its emergence in the early 1990s, the World Wide Web has rapidly evolved into a global information space of incomparable size. Keyword-based search engines such as Google™ index as many webpages as possible for the benefit of human users. Sophisticated as such search engines have become, they are still often unable to bridge the gap between HTML and the human. Tim Berners-Lee envisions the *Semantic Web* as the web of machine-interpretable information that complements the existing World Wide Web, providing an automated means for machines to truly traverse the Web on behalf of their human counterparts. A cornerstone application of the emerging Semantic Web is the search engine that is capable of tying components of the Semantic Web together into a traversable landscape. This paper describes both an architecture for and a prototype of a *Semantic Web Search Engine* (*SWSE*) using Jena that provides more sophisticated searching with more exacting results. To compare keyword-based search via Google with semantics-based search via the SWSE prototype, we utilize the *Google CruciVerbalist* (*GCV*), a system we developed that attempts to solve crossword puzzles via a generic search interface.

Introduction

For many, a search engine is the starting point for locating new information on the Web. Among companies specializing in Web search technologies, Google currently enjoys a top spot in terms of both coverage and reliability (Elgin 2004). Google's search technologies rely on the linked structure of the Web to rank webpages based on their popularity. Other search engines use a variety of word-frequency and clustering techniques, as well as additional keyword-based approaches. Regardless of the underlying architecture, users specify keywords that match words in huge search engine databases, producing a ranked list of URLs and snippets of webpages in which the keywords matched.

While such technologies have been successful, users are still often faced with the daunting task of sifting through multiple pages of results, many of which are irrelevant. Surveys indicate that almost 25% of Web searchers are unable to find useful results in the first set of URLs that are returned (Roush 2004). Such results are designated for human consumption rather than machine processing.

Tim Berners-Lee, the inventor of the World Wide Web, defines the *Semantic Web* as "The Web of data with meaning in the sense that a computer program can learn enough about what the data means [in order] to process it" (Berners-Lee 1999). Rather than a Web filled only with human-interpretable information, Berners-Lee's vision includes an extended Web that incorporates *machine-interpretable* information, enabling machines to process the volumes of available information, acting on behalf of their human counterparts (Fensel et al 2003).

In this paper, we present the building blocks of the Semantic Web and describe a scalable architecture for a *Semantic Web Search Engine* (*SWSE*) using Jena. A prototype implementation of the SWSE is also presented, including sample ontologies and results. Keyword-based search results from Google are compared to SWSE search results via the *Google CruciVerbalist* (*GCV*), a system developed to solve crossword puzzles using only the results of Google or another such search engine interface (Goldschmidt and Krishnamoorthy 2004).

Building Blocks of the Semantic Web

Much of the infrastructure of the Semantic Web has already been defined (see Berners-Lee et al 2001, Fensel et al 2003, Hjelm 2001, Heflin et al 2002, and others).

Uniform Resource Identifier (URI). *Uniform Resource Identifiers* (*URIs*) are used to represent tangible objects, people, places, abstract relationships, intangible or fuzzy concepts—just about anything (Connolly 2003). Syntactically, URIs resemble URLs. Defining URIs enables the development of ever-expanding machine-interpretable vocabularies. These are the nouns, verbs, and other language constructs that make up the Semantic Web.

Resource Description Framework (RDF). The *Resource Description Framework* (*RDF*) is used to combine URIs together to form machine-interpretable statements. Akin to simple prose, an RDF statement consists of a subject, a predicate, and an object. In general, the subject is a resource, the predicate is a property, and the object is either a resource or a literal value (see Lassila and Swick 1999, Manola and Miller 2002).

RDF Schema (RDFS). Using basic RDF constructs, rich machine-interpretable vocabularies may be developed. *RDF Schema* (*RDFS*) is an example of a widely used vocabulary language based on RDF that allows you to define classes, subclasses, properties, and subproperties (see Brickley and Guha 2002).

Web Ontology Language (OWL). Incorporating RDF and RDF Schema, the *Web Ontology Language* (*OWL*) further enriches the family of ontology languages available for use on the Semantic Web. OWL provides such constructs as: (1) relations (e.g. equivalence, disjointness); (2) cardinality; (3) richer typing; (4) characteristics of properties (e.g. symmetry, transitivity); and (5) enumerated classes (see McGuinness and van Harmelen 2004).

Ontologies. Using the aforementioned languages, domains of knowledge called *ontologies* are defined. An ontology "formally defines a common set of terms that are used to describe and represent a domain," thus making the terms and knowledge therein reusable (Fensel et al 2003).

Jena Framework. Designed and implemented by Brian McBride et al of HP Labs, the *Jena Framework* is a set of Java APIs devoted to Semantic Web application development. Jena supports ontologies developed using RDF, OWL, and RDFS. Based on these ontology languages, Jena provides a reasoning subsystem that supports RDFS and the *OWL Lite* subset, though OWL support is described as being "preliminary and still under development" (McBride et al 2005). Since the given ontology languages allow the specification of constraints, a means of validating an RDF model is provided via Jena's validation inference interface.

Sample Ontologies

Organized ontologies and other RDF documents are currently being created to support Semantic Web applications. Not surprisingly, much of these efforts are catalogued on the World Wide Web (see SchemaWeb 2005, Swoogle 2005). We constructed numerous ontologies as part of our SWSE implementation, including vocabularies for genealogy, mythology, United States Presidents, the Solar System, geography, and so on.

WordNet® Ontology. In an effort to bring a voluminous and practical vocabulary to the Semantic Web, we looked to *WordNet®*, an open lexical reference of over 200,000 English words and phrases, including their semantic relationships with one another. WordNet has been developed by the Cognitive Science Laboratory at Princeton University under the direction of Professor George A. Miller (WordNet 2005).

In early 2001, Melnik and Decker converted the WordNet vocabulary to RDF and RDFS (Melnik and Decker 2001). We have further translated WordNet to OWL, enabling the use of the WordNet vocabulary as both resources and properties. WordNet defines nouns, verbs, adverbs, and adjectives. We translated these vocabulary constructs to OWL as both properties and resources, appending the corresponding part of speech to distinguish usage. For example, the concept of life is translated to the `life-Noun` resource and the `life-Noun-as-Verb` property, enabling specific use as subject, predicate, or object in RDF statements.

Semantic Webgraphs. Information on the Semantic Web may be represented via *semantic webgraphs*. Such constructs are graphs in which resources and literal values are represented as nodes, and properties as either nodes (see Figure 1) or directed edges (see Figure 2). Semantic webgraphs provide a graph-based human-readable format, and enable software agents to traverse the Semantic Web via well-known graph traversal algorithms.

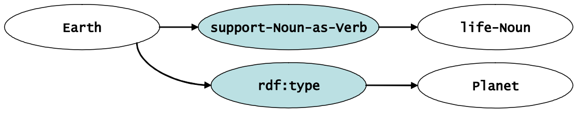

Figure 1. Semantic webgraph showing Earth's support for life

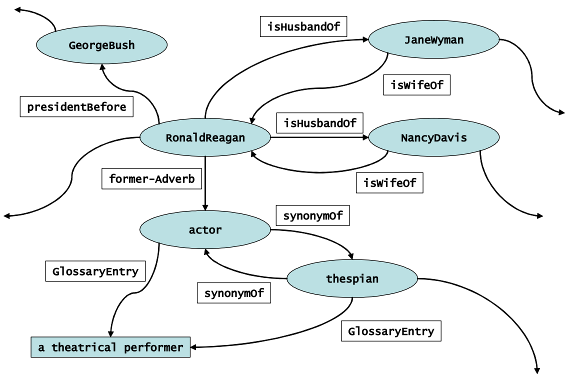

Figure 2. Semantic webgraph focused on Ronald Reagan, including vocabulary from the genealogy, WordNet, and Presidents ontologies

Implementing a Semantic Web Search Engine

SWSE Architecture and Prototype

From a bird's-eye view, the architecture of the *Semantic Web Search Engine* resembles that of traditional keyword-based search engines. Queries are accepted and results generated based on summarized data in a central database.

Search Queries. The basic search query form is plaintext, which supports intuitive features much like that of Google. Text may be quoted to treat multiple words as a single unit, and *stop words* (e.g. "is," "the," "of," etc.) will—to some degree—be ignored. Starting with a plaintext query form maximizes the flexibility of future search enhancements.

Using Search Keywords to Identify URIs. In the SWSE architecture, keyword-based search still plays a role. Given a search query Q in plaintext form, phrases of Q are matched via case-insensitive string matching against the `<rdfs:label>` and `<rdfs:comment>` elements of all available RDF documents, as well as all `<rdfs:Literal>` elements, as identified by property definitions. Results of this string-matching phase are URIs of both properties and non-properties, weighted according to frequency.

Forming RDF Queries. Once potential properties and resources are identified, they are combined to form RDF queries against the RDF knowledge base. This is the heart of the SWSE architecture in which the various combinations of properties and resources are queried and results collected and ranked.

As an example, if a given plaintext query results in potential properties p_1 and p_2, and potential non-property resources n_1, n_2, and n_3, a query will be generated for each combination in which either zero or one element is missing (e.g. $n_1\ p_1\ n_2$; $n_1\ p_1\ n_3$; $n_1\ p_1\ ?$; $?\ p_2\ n_1$; etc.). This "fill-in-the-blank" RDF statement detection process is repeated with those new elements discovered forming a "hop" in the sense of a breadth-first search algorithm.

Given example query "wife of President before Adams," we match substrings against the SWSE knowledge base. From the genealogy ontology, we match the `isWifeOf` property; from the US Presidents ontology, we match the `presidentBefore` and `presidentAfter` properties, as well as the `JohnAdams` and `JohnQuincyAdams` resources. During the first pass, we detect statements shown in Figure 3.

```
GeorgeWashington presidentBefore JohnAdams.
JamesMonroe presidentBefore JohnQuincyAdams.
ThomasJefferson presidentAfter JohnAdams.
AndrewJackson presidentAfter JohnQuincyAdams.
AbigailSmith isWifeOf JohnAdams.
LouisaJohnson isWifeOf JohnQuincyAdams.
```

Figure 3. Detected RDF statements shown in weighted order

For each new resource detected, the process repeats itself, yielding new RDF statements (see Figure 4).

```
MarthaCurtis isWifeOf GeorgeWashington.
ElizabethKortwright isWifeOf JamesMonroe.
MarthaSkelton isWifeOf ThomasJefferson.
RachelRobards isWifeOf AndrewJackson.
JamesMonroe presidentAfter JamesMadison.
ThomasJefferson presidentBefore JamesMadison.
etc.
```

Figure 4. RDF statements detected during the second pass

Though this process may be repeated many times, we limit the number of repetitions—i.e. the maximum breadth—to a small number such as two or three, otherwise results will be flooded with irrelevant inferences.

Traversing Semantic Webgraphs. The RDF statements discovered via the aforementioned querying process form a semantic webgraph. In an attempt to match multiple RDF statements to a given plaintext query, we number each word of the plaintext query, as shown in Figure 5.

```
wife of President before Adams
1    2  3         4      5
```

Figure 5. Sample plaintext query with order specified

We then count the number of potential properties, n_p, detected for the plaintext query string, deduplicating based on location. For the given example, the word "wife" in location `1` matches the `isWifeOf` property, whereas the word "President" in location `3` matches both the `presidentAfter` and `presidentBefore` properties. Counting location `3` only once, n_p is two, indicating that our results should combine at most two RDF statements. More specifically, we traverse only two properties in our semantic webgraph. See Table 1 for example results.

SWSE Prototype Results

As described above, we have successfully implemented and tested an SWSE prototype. Rather than use a relational database, the SWSE prototype stores all of its knowledge in memory.

Sample SWSE Query Results. With the aforementioned ontologies, the SWSE prototype provides results to queries in a fashion reminiscent of Prolog. Example results appear in Table 1.

Plaintext Query	Top SWSE Results
Wife of President before Adams	Martha Curtis | Is Wife Of | George Washington | President Before | John Adams. Elizabeth Kortwright | Is Wife Of | James Monroe | President Before | John Quincy Adams.
Daughter of wife of Norse God of Mischief	Hel | Is Daughter Of | Angrboda | Is Wife Of | Loki | Is God Of | Mischief.
Who wrote the Gettysburg Address?	Abraham Lincoln | wrote | Gettysburg Address. Abraham Lincoln | penned | Gettysburg Address.

Table 1. Example query results using SWSE

Google CruciVerbalist. A fundamental goal of the Semantic Web is to enable machines to communicate with one another via machine-processable vocabularies. In an effort to compare keyword-based search and semantics-based search, we constructed the *Google CruciVerbalist* (*GCV*), a system that attempts to solve crossword puzzles using Google or the SWSE prototype to answer clues.

GCV utilizes numerous keyword-based "tricks" to translate crossword puzzle clues into "Google-friendly" or "search-friendly" query strings. For each query string, candidate answers are obtained from the list of top ten query results. None of the actual HTML pages are fetched (Goldschmidt and Krishnamoorthy 2004).

Comparative Crossword Puzzle Results. A set of theme-based children's crossword puzzles were used to compare keyword-based searching via Google to semantics-based searching via the SWSE prototype.

As shown in Table 2, the keyword-based approach yields many irrelevant results, whereas the semantics-based approach is much more exacting.

Crossword puzzle	Candidate answers per clue via Google (min / avg / max)	Candidate answers per clue via SWSE (min / avg / max)
Norse Mythology	15 / 68.44 / 124	1 / 1.36 / 3
US Presidents	25 / 107.69 / 206	1 / 5.31 / 12
Solar System	34 / 89.45 / 192	1 / 3.09 / 8
US Geography	4 / 42.00 / 72	1 / 1.10 / 2

Table 2. Comparing the number of candidate answers per clue

Given the set of candidate answers for each clue, GCV attempts to fill in the crossword grid. Each candidate answer is assigned a confidence value based on word frequency; such confidence values drive the depth-first search algorithm used to populate the grid. The fewer incorrect candidate answers, the higher the success rates, as shown in Table 3.

Crossword puzzle	Correctly placed words via Google	Correctly placed words via SWSE
Norse Mythology	9 / 25 (36%)	25 / 25 (100%)
US Presidents	8 / 13 (62%)	12 / 13 (92%)
Solar System	11 / 11 (100%)	11 / 11 (100%)
US Geography	4 / 10 (40%)	10 / 10 (100%)

Table 3. Comparing the success of solving crossword puzzles

Conclusions

Though Google searches occur by the thousands every second (Elgin 2004), technologies for searching the World Wide Web are reaching a plateau. New developments and advancements in keyword-based search technologies will continue to improve search services on the Web; however, the growth rate of these improvements will likely be slight. Problems of imprecise and irrelevant results will continue to hinder Web searchers, especially with the continued expansion of the Web.

A new, semantically based approach is necessary not only to reduce the "information overload" problem of the day, but also to enable more effective and productive services over the Web. By providing a viable architecture and prototype for a Semantic Web search engine, our research aims to help open the floodgates of the emerging Semantic Web.

References

Berners-Lee, T. 1999. *Weaving the Web: the Original Design and Ultimate Destiny of the World Wide Web by Its Inventor*. New York: HarperSanFrancisco.

Berners-Lee, T. et al 2001. The Semantic Web. *Scientific American*. May 2001.

Brickley, D. and Guha, R. eds. 2002. RDF Vocabulary Description Language 1.0: RDF Schema. W3C, http://www.w3.org/TR/rdf-schema.

Calishain, T. and Dornfest R. 2003. *Google Hacks: 101 Industrial-Strength Tips & Tools*. Sebastopol, Calif.: O'Reilly & Associates, Inc.

Connolly, D. et al 2003. Web Naming and Addressing Overview. W3C, http://www.w3.org/ Addressing.

Elgin, B. 2004. Why the world's hottest tech company will struggle to keep its edge. *BusinessWeek*. May 3, 2004.

Fensel, D. et al eds. 2003. *Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential*. Cambridge, Mass.: MIT Press.

Goldschmidt, D. and Krishnamoorthy, M. 2004. Solving Crossword Puzzles via the Google API. In *Proceedings of the IADIS International Conference WWW/Internet 2004*, 382-389. Madrid, Spain: IADIS Press.

Heflin, J. et al eds. 2002. Requirements for a Web Ontology Language. W3C, http://www.w3.org/TR/webont-req.

Hjelm, J. 2001. *Creating the Semantic Web with RDF*. John Wiley & Sons, Inc.

Lassila, O. and Swick, R. eds. 1999. Resource Description Framework (RDF) Model and Syntax Specification. W3C, http://www.w3.org/TR/REC-rdf-syntax.

Manola, F. and Miller, E. eds. 2002. RDF Primer. W3C, http://www.w3.org/TR/rdf-primer.

McBride, B. et al 2005. HP Labs Semantic Web Research, http://www.hpl.hp.com/semweb/.

McGuinness, D. and van Harmelen, F. eds. 2004. OWL Web Ontology Language Overview. W3C, http://www.w3.org/TR/owl-features.

Melnik, S. and Decker, S. 2001. WordNet via RDF, http://www.semanticweb.org/library/.

Roush, W. 2004. Search beyond Google. *Technology Review*. March 2004. http://www.technologyreview.com/articles/print_version/roush0304.asp.

SchemaWeb 2005. http://www.schemaweb.info/.

Swoogle 2005. http://swoogle.umbc.edu/.

WordNet 2005. http://wordnet.princeton.edu/.