

# 1 Computational Thinking Leading to a Web of Knowledge

To think about turning raw data into accessible knowledge in a *Web of Knowledge* (*WoK*), we first ask some fundamental questions: What is data? What are facts? What is knowledge? How does one know? Philosophers have pursued answers to these questions for millennia; and although we do not pretend to be able to contribute to Philosophy, we can use their ideas about *ontology*, *epistemology*, and *logic* to guide us in how to build a WoK algorithmically.

- *Ontology* is the study of existence. It asks: “What exists?” In our quest to build a WoK, we must find computable solutions the question: “What concepts, relationships, and constraints exist?” Our answer: computational ontology can provide formal conceptual models of some domain of knowledge by declaring the relevant concepts along with the relationships among these concepts and the constraints over these concepts and relationships.
- *Epistemology* is the study of the nature of knowledge. It asks: “What is knowledge?” and “How is knowledge acquired?” To build a WoK, we provide computational answers to “What is digitally stored knowledge?” and “How does raw data become algorithmically accessible knowledge?” by populating conceptual models. We turn raw data into knowledge by embedding facts in the concepts and relationships in accord with the constraints.
- *Logic* comprises principles and criteria of valid inference. It asks: “What is known?” and “What can be inferred?” In the computational context of a WoK, it can answer the question: “What are the known facts (both given and implied)?” We ground our conceptual model in a description logic—a decidable fragment of first-order logic. To make this logic practical for non-logicians, we must and do add a query generator whose input consists of ordinary free-form textual expressions or ordinary fill-in-the-blank query forms.

To actualize these ideas, we propose a way to turn raw symbols contained in web pages (or other source documents) into knowledge and to make this knowledge accessible by average web users. The key research problems are: (1) How do we make ontology creation—conceptual-model creation—easy enough to be usable by typical human knowledge workers? (2) How do we make epistemology—content annotation with respect to an ontology—easy enough to require little, if any, training for human annotators? (3) How do we make logic—query specification—easy enough for unspecialized web users? Not only do these activities need to be easy enough, they also have to be good enough. Without a resolution to these problems, the barrier to WoK content creation and usage will remain too high, and the WoK will remain elusive. Our research proposal innovatively addresses these problems and shows how a paradigm-shifting advance can enable a WoK.

## 2 From a Web of Pages to a Web of Knowledge

We use two examples to show how we propose to turn a web page into a page of queryable data. Figure 1 shows part of two ordinary, human-readable web pages about cars for sale. The facts in these pages are obvious: e.g., a ’93 NISSAN is for sale; it is sweet cherry red, has air conditioning, and sells for \$900. Facts on other pages are much less obvious (e.g., Figure 2), but a specialist can see a myriad of facts: Chromosome 17 starts at location 1,194,558, ends at 1,250,267, and has 55,709 bases. Users would like to be able to query the facts on these pages directly: “Find me a red Nissan for under \$5000; it should be a 1990 or newer and have less than 120K miles on it.” Or, “Tell me the location and size of chromosome 17.” We cannot, however, directly access these facts with the current structure of the web. Our proposal makes these facts visible from outside the page and directly accessible to query engines (as opposed to search engines).

The screenshot shows the City Weekly Classifieds website. At the top, there's a navigation bar with links for news, arts & entertainment, event listings, dining listings, classifieds, best of utah, and about. Below this is a search bar and a section for classified ads. The main content area is titled "Classifieds: Autos" and shows 16-24 results from 24 total results. Several car listings are visible, including a '97 MITSUBISHI Montero LS, a '93 NISSAN Model XE, and a '97 SAAB. A sidebar on the left contains a navigation menu with categories like Classifieds, Real Estate For Rent, Employment, Financial, Rec. Vehicles, Merchandise, Garage/Yard Sales, Agricultural, Pets & Livestock, Personals, Announcements, Legal Notices, Service Directory, Marketplace, Homes, Jobs, Autos, Business Directory, OnlineAthens, News, UGA News, Obituaries, Police Central, and Sports. A large advertisement for "OnlineAthens" is overlaid on the right side of the page, featuring a search icon and the text "BONA FIDE CLASSIFIED".

Figure 1: Sample Car Ads Web Pages.

## 2.1 From Symbols to Knowledge—Ontological and Epistemological Tools

To make facts available to query engines, we first map out a guiding pathway to turn raw symbols into knowledge. *Symbols* are characters and character-string instances (e.g., \$, mileage, red, chromosome, 55,709). *Data* builds on *symbols* by adding conceptual meta-tags (e.g., Price: \$900, Color: red, Size: 55,709). *Conceptualized data* groups data tagged with conceptual identifiers into the framework of a conceptual model (e.g., an ordinary, extended ER model, although in our work the conceptual models we use are fact-oriented). We have *knowledge* when we populate a conceptual model with correct<sup>1</sup> conceptualized data.

To specify ontological descriptions, we need a conceptual-modeling language. We use OSM [EKW92], which lets us classify and describe things that exist as object sets, relationships among these things as relationship sets, and constraints over these object and relationship sets. OSM is equivalent to an *ALCN* description logic [BN03], which gives it the formal properties we need.

A minimal necessary tool is an editor that allows users to construct conceptual models. Building ontologies by hand, however, becomes a bottleneck in the process of turning data into knowledge [BCL06]. Can we automatically construct an ontology for a domain of knowledge from raw source domain information? If so, how?

<sup>1</sup> *Correct* is interesting. How do we know whether conceptualized data is correct? Humans struggle to know; machines may never know. For the system we are proposing, we rely on evidence and provenance by always linking conceptualized data back to its original source, the human-readable web page from which it was extracted.

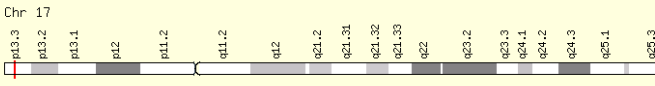

<p><b>Genomic Location</b> (According to <a href="#">GeneLoc</a> and/or <a href="#">HGNC</a>, and/or <a href="#">Entrez Gene (NCBI build 36)</a>, and/or <a href="#">miRBase</a>, Genomic Views According to <a href="#">UCSC</a> and <a href="#">Ensembl</a>) <a href="#">About This Section</a></p> <p>Jump to Section...</p>	<p>Chromosome: <b>17</b> Entrez Gene cytogenetic band: <b>17p13.3</b> Ensembl cytogenetic band: <b>17p13.3</b></p> <p>Gene in genomic location: bands according to <a href="#">Ensembl</a>, locations according to <a href="#">GeneLoc</a> (and/or <a href="#">Entrez Gene</a> and/or <a href="#">UCSC</a>)</p>  <p>GeneLoc gene densities for chromosome 17</p> <p><a href="#">GeneLoc location for GC17M001194</a> (about GC identifiers)</p> <p>Start: <b>1,194,558</b> bp from pter End: <b>1,250,267</b> bp from pter Size: <b>55,709</b> bases Orientation: <b>minus</b> strand</p> <p><a href="#">Exon Structure</a> </p> <p><b>1 alternative location:</b> <b>Chr7+</b>, CRA_TCAChr7v2 63,208,480-63,232,567</p> <p>RefSeq genomic assemblies: <a href="#">NT_079593.2</a> <a href="#">NC_000017.9</a> <a href="#">NT_010718.15</a></p> <p>Genomic View: <a href="#">UCSC Golden Path with GeneCards custom track</a></p>
<p><b>Proteins</b> (According to <a href="#">UniProt</a>, and/or <a href="#">Ensembl</a>, Phosphorylation sites)</p>	<p><b>UniProt/Swiss-Prot:</b> <a href="#">1433E_HUMAN_P62258</a> (See protein sequence)</p> <p><b>Size:</b> 255 amino acids; 29174 Da <b>Subunit:</b> Homodimer. Interacts with NDEL1 (By similarity). Interacts with HCV core protein <b>Subcellular location:</b> Cytoplasm (By similarity), Melanosome. Note=Identified by mass spectrometry</p>

Figure 2: Sample Molecular-Biology Web Page.

Attempts to extract ontologies from natural-language text documents have been unsuccessful [BCL06]. Although the jury is still out, attempts to extract ontologies from semi-structured documents, such as the ones in Figures 1 and 2, appear promising [TEL<sup>+</sup>05]. Thus, we build tools to use the very web pages we wish to turn into knowledge as sources to help us construct an ontology. This works by recognizing that the data is formatted in a particular way (e.g., the table in the OnlineAthens ads in Figure 1) and by using reverse-engineering techniques to construct a conceptual model (e.g., to discover that *Price*, *Year*, *Make*, and *Model* in the table in Figure 1 are concepts for a car-sales conceptual model or to discover that *Start*, *End*, *Size*, and *Orientation* are all related concepts describing a chromosome location in a gene ontology).

Since we cannot fully count on these automatic ontology-building tools, we also provide a way to build ontologies that leverages the idea of an ordinary form. People generally know how to design forms with single- and multiple-entry blanks. And we know how to algorithmically turn form-like nested structures into conceptual models [AK07].

Although we can cross the barrier to building conceptual models, these ontological descriptions are not enough. We also need a way to link raw facts in web pages with ontological descriptions. We need epistemological tools as well as ontological tools.

A way to link the actual facts with an ontology is to annotate a web page with respect to that ontology. A data value  $V$  in a web page annotated for an object set  $S$  in an ontology  $O$  is a mapping from  $V$  to  $S$  in  $O$ . Likewise, we annotate related pairs (and, more generally, related groups) of values in a web page by mapping them to a relationship set in an ontology. In [DEL06], we describe an automatic annotation tool we have built, and our web site [DEG] includes a working demo that annotates a web page with respect to an ontology. In the demo, when a user causes the mouse cursor to hover over an annotated value (e.g., over the annotated value *160K* in Figure 1), the demo system highlights the value and displays the connecting link to the ontology.

Although it is possible to annotate a web page with respect to an ontology by hand, this is likely to be too tedious and time consuming to be practical for many applications.<sup>2</sup> We have therefore augmented ontologies with instance recognizers (ontologies augmented with instance

<sup>2</sup>Although tedious, we do foresee hand-annotation as a viable way to create annotated content. Moreover, we can also annotate images like some commercial enterprises do (e.g., [Foo07]), but with respect to ontologies so that they can be queried in the WoK.

Red Nissan for under \$5000 – a 1990 or newer with less than 120K miles on it

Figure 3: Free-Form Query

recognizers are called *extraction ontologies*). Instance recognizers contain regular expressions that recognize common textual items such as dates, times, prices, and numbers. They also contain lexicons that match with items such as car makes and models and protein names and functions. Much can and has been said about how to build and use these instance recognizers embedded within extraction ontologies (e.g., [ECJ<sup>+</sup>99, DEL06] among several others).

Building instance recognizers is laborious. We have four answers to this legitimate observation: (1) We already have a library of common instance recognizers that can be used directly or specialized for use in some domain. (2) We need not have perfect recognizers because we can augment extraction ontologies with pattern-based extractors—once a few values have been recognized, pattern-based extractors can determine the pattern of a semi-structured page and thereby correctly recognize values beyond those observed by instance recognizers. (3) We can automatically update lexicon recognizers by adding additional values found by pattern-based extractors. And (4) neither instance recognizers nor lexicons are absolutely necessary because we can let users provide a few sample mappings from a page to an ontology, and from these sample mappings generate pattern-based annotators.

## 2.2 Querying Knowledge—Logic Tools

After building tools to turn raw symbols in web pages into knowledge, we next need to provide appropriate query capabilities. Given that we have data on a web page annotated with respect to an ontology, we can immediately generate subject-predicate-object triples in RDF, the W3C standard for representing ontological data. We can then directly use the SPARQL query language, also a W3C standard, to write and execute queries over this RDF data.

If everyone could write SPARQL queries, we would basically have what we need to enable users to search the WoK. Since users should not be asked to learn SPARQL, however, we provide a query system in which users can pose queries in their own terms. Figure 3 shows a free-form query for a tool we have built<sup>3</sup> to query WoK content [AME06, AME07]. The key to making these free-form queries work is not natural-language processing (at least not in the usual sense of natural-language processing), but rather is the application of extraction ontologies to the queries themselves. This lets us align user queries with ontologies and thus with facts in annotated web pages. One can also think about these free-form queries as keyword queries, except that the system also recognizes and uses applicable operations over recognized keywords. As Figure 3 shows, the WoK query engine highlights words, values, phrases, and operations it recognizes (e.g., the context keyword *miles*, the value *Red*, and the operation *under* applied to the value *\$5000*). The highlighting provides feedback to users, letting them know which words, values, phrases, and operations the WoK search engine recognizes.

Anyone can readily pose free-form queries. To be successful, however, a user does have to guess what keywords, values, and constraint expressions might be available in an extraction ontology for the domain of interest. This is similar to users having to guess keywords and values for current search-engine queries. Since arbitrary free-form queries may not always be successful, we also provide a form query language, based on the ontology, that allows a user to fill out a form and submit it to pose queries in much the same way users currently pose queries by filling in forms currently on the web. Interestingly, these query forms are automatically derivable from domain

<sup>3</sup>See the on-line “SerFR” demo at our web site [DEG].

ontologies, and thus need not be specified by developers. Instead of reverse-engineering a form to create an ontological structure, we can forward-engineer (derive) forms from the ontology and use them in a natural-forms query language [Emb89].

Finally, we must make all of this scale globally. We have defined semantic indexing [AMELT07], with which we can quickly find applicable ontologies for user queries, and we have considered large-scale caching, following the lead of modern search engines. We realize that we will not be able to build a full-scale system on our own, but we can show the way and provide the technical answers to make it all work.

### 3 Research Plan—Innovations Yielding Project Outcomes

The work we propose here presents a grand vision—a vision that others share [BL07]. What is different and innovative, however, is that we have developed a practical plan to realize this vision. In answer to the key research problems in our quest to build a WoK, we propose the following:<sup>4</sup>

1. Make ontology creation easy enough to be usable by typical human knowledge workers by (a) computationally automating ontology creation for many semi-structured knowledge repositories,<sup>†</sup> (b) enabling ontology-creation-by-form-specification for customizable ontology creation,<sup>‡</sup> (c) providing for automated self-improvement of extraction ontologies,<sup>‡</sup> and (d) making it worthwhile for interested knowledge workers to spend valuable time creating, modifying, and establishing the basis for improved ontologies.<sup>‡</sup>
2. Make epistemological content annotation with respect to an ontology require no training or only minimal training by (a) computationally automating epistemological content annotation,<sup>†</sup> (b) providing a by-example way to annotate content for large, semi-structured information repositories,<sup>‡</sup> and (c) making it worthwhile for interested people to annotate content, not automatically or semi-automatically annotatable.<sup>‡</sup>
3. Make logic (i.e., query specification) easy enough for average web users by (a) enabling free-form queries for unspecialized web users,<sup>†</sup> (b) expanding the keyword-search paradigm to not just content keywords but also to meta-keywords and to simple computations, equalities, comparisons, and aggregations,<sup>‡</sup> and (c) providing on-the-fly-generated, fill-in-the-form query interfaces.<sup>‡</sup>

Our claims lend themselves to rigorous evaluation. We initially intend to work in four application areas representing diverse fields of interest: geopolitical data (government), microbiology (science), items for sale (retail business), and genealogy (historical documents). For each, we must demonstrate that the body of facts extracted by experts from specific web pages, can be obtained with our tools by computer-literate non-specialists (e.g., government employees, biology students, prospective buyers, and family history enthusiasts). This must be accomplished with minimal training and modest effort, in reasonable time, and with acceptable reliability. To this end, we will use evaluation techniques similar to those we have used in earlier work [ZN04]. We will monitor, via logging routines embedded in our software tools, the time taken by the subjects and appropriateness of the answers obtained to a set of factual queries.

A resolution of the research problems as proposed here should break down the barrier to both accessible-fact creation and fact access. It should enable us to “WoK” the vast store of heterogeneous, digital data on the web.

---

<sup>4</sup>We mark research projects with a dagger (†) that are well underway based on prior NSF-sponsored research and mark research projects with double-dagger (‡) that we propose as part of this CDI-sponsored research.

## 4 List of Participants—Intellectual Partnerships & Synergism

David W. Embley

Department of Computer Science

Brigham Young University

Provo, Utah

Research Interests: conceptual modeling, information extraction, ontology generation, and user-friendly query languages.

Stephen W. Liddle

Information Systems Department

Director of the Kevin and Debra Rollins Center for eBusiness

Marriott School of Management

Brigham Young University

Provo, Utah

Research Interests: conceptual modeling, software engineering environments and tools, data extraction, information retrieval, multiparadigm software development environments and tools, and e-business.

Deryle W. Lonsdale

Department of Linguistics & English Language

Brigham Young University

Provo, Utah

Research Interests: formal syntax and semantics, computational linguistics, information extraction, and Salish languages.

George Nagy

Department of Electrical, Computer, and Systems Engineering Rensselaer Polytechnic Institute

Troy, New York

Research Interests: recognition systems that improve with use, document image analysis, design and evaluation of automated document entry (text, tables, diagrams, maps), and pattern recognition.

The four Co-PI's come from four different departments at two universities. Their research interests are complementary and synergistic with respect to the proposed research project. Nagy is particularly strong in design and evaluation of recognition systems like the ones we need to generate and populate ontologies; Lonsdale is particularly strong in computational linguistics which we need for reverse-engineering semi-structured text into meaningful conceptual models; Liddle is particularly strong in software engineering environments which we need to build the fairly large-scale demonstration system we envision; and Embley is particularly strong in user-friendly query languages which are needed for average users of the WoK. All of us have done research on ontology-generation and information-extraction projects. As a group, we have all been Co-PI's on one or the other or both of the NSF-sponsored projects that directly support the current proposal (NSF grants 0083127, 0414644, & 0414854). We have worked together for many years; and although our joint research interests are diverse, it is our unique approach to combining them in new and groundbreaking ways that leads to a potential solution for "generating ... knowledge from [the] wealth of heterogeneous digital data" by creating a web of knowledge.



## References

- [AK07] R. Al-Kamha. *Conceptual XML for Systems Analysis*. PhD dissertation, Brigham Young University, Department of Computer Science, June 2007.
- [AME06] M.J. Al-Muhammed and D.W. Embley. Resolving underconstrained and overconstrained systems of conjunctive constraints for service requests. In *Proceedings of the 18th International Conference on Advanced Information Systems Engineering (CAiSE'06)*, pages 223–238, Luxembourg City, Luxembourg, June 2006.
- [AME07] M. Al-Mumammed and D.W. Embley. Ontology-based constraint recognition for free-form service requests. In *Proceedings of the 23rd International Conference on Data Engineering (ICDE'07)*, pages 366–375, Istanbul, Turkey, April 2007.
- [AMELT07] M.J. Al-Muhammed, D.W. Embley, S.W. Liddle, and Y. Tijerino. Bringing web principles to services: Ontology-based web services. In *Proceedings of the Fourth International Workshop on Semantic Web for Services and Processes (SWSP'07)*, pages 73–80, Salt Lake City, Utah, July 2007.
- [BCL06] P. Buitelaar, P. Cimiano, and B. Loos. Preface. In *Proceedings of the 2nd Workshop on Ontology Learning and Population: Bridging the Gap Between Text and Knowledge, (COLING-ACL 2006)*, page v, Stroudsburg, Pennsylvania, 2006. Association for Computational Linguistics.
- [BL07] T. Berners-Lee. Future of the world wide web, March 2007. Testimony of Sir Timothy Berners-Lee Before the United States House of Representatives Committee on Energy and Commerce Subcommittee on Telecommunications and the Internet.
- [BN03] F. Baader and W. Nutt. Basic description logics. In F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider, editors, *The Description Logic Handbook*, chapter 2, pages 43–95. Cambridge University Press, Cambridge, UK, 2003.
- [DEG] Homepage for BYU Data Extraction Group. [www.deg.byu.edu](http://www.deg.byu.edu).
- [DEL06] Y. Ding, D.W. Embley, and S.W. Liddle. Automatic creation and simplified querying of semantic web content: An approach based on information-extraction ontologies. In *Proceedings of the First Asian Semantic Web Conference (ASWC'06)*, pages 400–414, Beijing, China, September 2006.
- [ECJ+99] D.W. Embley, D.M. Campbell, Y.S. Jiang, S.W. Liddle, D.W. Lonsdale, Y.-K. Ng, and R.D. Smith. Conceptual-model-based data extraction from multiple-record web pages. *Data & Knowledge Engineering*, 31(3):227–251, November 1999.
- [EKW92] D.W. Embley, B.D. Kurtz, and S.N. Woodfield. *Object-oriented Systems Analysis: A Model-Driven Approach*. Prentice Hall, Englewood Cliffs, New Jersey, 1992.
- [Emb89] D.W. Embley. NFQL: The natural forms query language. *ACM Transactions on Database Systems*, 14(2):168–211, June 1989.
- [Foo07] Footnote.com. <http://www.footnote.com>, 2007.
- [TEL+05] Y.A. Tijerino, D.W. Embley, D.W. Lonsdale, Y. Ding, and G. Nagy. Toward ontology generation from tables. *World Wide Web: Internet and Web Information Systems*, 8(3):261–285, September 2005.
- [ZN04] J. Zou and G. Nagy. Evaluation of model-based interactive flower recognition. In *Proceedings of the 17th International Conference on Pattern Recognition*, volume II, pages 311–314, Cambridge, United Kingdom, August 2004.