

# Computerizing Computer Science

**T**here are two possible scenarios for the future of relations between humans and computers: *user friendliness* and *computer friendliness*.

In the user-friendliness scenario, computers become smart enough to communicate in natural language and, in general, adapt to human conventions. In the computer-friendliness scenario, humans radically adapt their practices in order to communicate with and take advantage of computers.

Correspondingly, there are two complementary approaches to managing textual data on the Web: *natural language texts* and *computer-friendly* encodings. Natural language texts store raw textual language with the expectation that computers will be smart enough to extract useful information. Computer-friendly encodings use metadata, markup, and formalized or controlled languages ([www.uilots.let.ruu.nl/Controlled-languages](http://www.uilots.let.ruu.nl/Controlled-languages)) to make texts easier to process by computers. I believe that while both approaches are worth pursuing, there is too little appreciation of the desirability (indeed, perhaps even the inevitability) of the second, computer-friendly approach. I argue, first, that the use of computer-friendly encodings is symptomatic of a revolutionary trend toward the computerization of human knowledge. Second, I argue that computer scientists should aim to set an example, by organizing computer science resources into a standardized, online “archive of computer science knowledge,” using computer-friendly encodings to enable easy processing by computer.

## The Computerization of Knowledge and Language

Computers and the Internet are changing the way people do all sorts of things. As a consequence, much

of human knowledge has come to involve the use of computers and their accompanying formalisms: relational and object-oriented databases, query languages, programming languages, scripting languages, search engines, hypertext, browsers, user interfaces, visualization tools, and application-specific languages (for spreadsheets, word processors, CAD packages, expert systems, and so forth). These computer formalisms complement the already widespread use of mathematical and logical formalisms.

In this sense, artificial languages have already partially displaced natural languages as the means for storing and representing human knowledge. (Is this something that can be quantified?)

Some day, perhaps, computers will be intelligent enough to conform to human standards and conventions. But unless and until this occurs, humans will instead need to adjust their habits to conform to computer standards, by using specialized, computer-oriented formalisms.

Moreover, even if computers do become intelligent enough to conform to some of our conventions, it seems likely we will find it more convenient and precise to communicate in specialized, computer-oriented languages, since so much of human knowledge will already be expressed in such languages, and since natural languages suffer from ambiguity and imprecision. Natural language is certainly not ideal for many descriptive and performative tasks, such as describing mathematical, graphical, and computational objects, and specifying complex actions.

The lingua franca of the future may very well be a computer language.

In particular, I believe computer scientists will increasingly come to use formalized languages for

## THE LINGUA FRANCA OF THE FUTURE

may very well be a computer language.

## • Viewpoint

storing and representing their online material, and I wish to accelerate this trend toward computerization.

### Four Levels of Computer-Friendly Encoding

I identify four levels of computer-friendly encoding. First is the level of metadata (see the Dublin Core Metadata Element Set; [purl.oclc.org/metadata/dublin\\_core](http://purl.oclc.org/metadata/dublin_core) and a Review of Metadata; [www.ukoln.ac.uk/metadata/DESIRE/overview](http://www.ukoln.ac.uk/metadata/DESIRE/overview).) This level contains texts accompanied by metatags documenting (superficial) information such as author, file format, date of creation, subject, location, maintainer, keywords, and other similar facts about the text.

The next level of encoding text is markup, exemplified by LaTeX, HTML, and TEI Lite. Already, researchers in many fields write scholarly documents in text processing languages like LaTeX that make explicit some of the logical and rhetorical structure of the document. LaTeX and HTML are the preferred formats for the physics community (see e-print archive; [xxx.lanl.gov](http://xxx.lanl.gov)), while LaTeX is a de facto standard for the mathematics community (see e-math home page; [www.ams.org](http://www.ams.org)). HTML is widely used for representing information on the Web. TEI Lite is used for annotating literary and other works in several digital library projects, including the Oxford Text Archive.

Text can be even more marked up than the typical LaTeX, HTML, or TEI Lite document. The third level of encoding I call the semantic level, wherein the actual content of the document is expressed in a formal, computer-friendly knowledge representation language. The Mizar language ([www.mizar.org/language/](http://www.mizar.org/language/)) is being used to develop a database of computer-verified mathematics. It is one component of the QED project ([www.mcs.anl.gov/qed](http://www.mcs.anl.gov/qed)), that aims “to build a single, distributed, computerized repository that rigorously represents all important, established mathematical knowledge.”

These three levels—metadata, markup, and semantic encoding—represent a continuum along the axis of increasing amounts of explicitness and formalization of content. There is an orthogonal approach to the computer-friendly encoding of text: controlled languages. The aim here is to restrict the syntax and vocabulary of natural language texts to support automatic processing by computer. Con-

trolled languages are already being used to enable machine translation of software documentation.

The widespread use of computer-friendly encodings will enable intelligent access to digitally stored information. As standards evolve and as progress is made with knowledge representation languages, more and more useful information will be contained in the formalized languages and less in the natural language text itself.

### Organizing Computer Science Resources

**T**he Web needs standards if people and computers are to communicate productively. It is unfortunate that Postscript is the de facto standard format for storing computer science reports on the Web. Not only is Postscript difficult to use for information retrieval (it is computer-unfriendly); it is also verbose. Postscript is in many ways the opposite of marked-up text (marked-down text?) because it describes the physical appearance of text without making explicit the logical and rhetorical structure.

Distributed on the Web are numerous computer science resources of various qualities and styles: home pages for projects, home pages for individuals, FTP sites containing software and papers, bibliographies, report archives, mailing lists, course information, news groups, information stored at commercial and industrial sites, and so on.

Currently, the most popular tools for Web searching are general-purpose search engines such as Alta Vista, Excite, HotBot, and Infoseek. These search engines rely on various statistical tricks to find matching documents. But the search engines are inadequate. Often they return too many hits, often they return too few hits, and often the ratio of useful hits to irrelevant hits is low.

The search engines' mediocre utilities are likely to continue until either substantial progress is made in natural-language understanding or people use computer-friendly encodings to help search engines find relevant material.

Aside from the aforementioned search engines, there are subject directories like Yahoo, Magellan, and NewHoo that contain (manually) organized directories of resources. Analogous to these directories, there is a real need to organize the various computer science resources on the Internet into a coherent framework—a unified computer science

archive. The advantages of such an organization would be better access, less redundancy, and higher quality.

**Better access.** If documents are classified by subject and represented in a computer-friendly encoding, this will enable easier knowledge extraction by humans, by search engines, and by other text processing systems.

**Less redundancy.** Numerous digital library projects and tech-report archives worldwide are duplicating efforts by designing similar systems performing similar tasks. Multiple copies of documents are stored in a haphazard manner throughout the Web. Moreover, in the absence of a standardized way to find resources, different researchers rediscover the same technique or theorems and overlook work relevant to their own research.

**Higher quality.** A standardized computer science archive will speed up dissemination of knowledge and encourage researchers to be more organized and formal about exactly what they are working on. The easier access to previous work will, like market pressures in the monetary economy, lead to less wasted effort and more concentration on achieving useful, novel results.

I envision the computer science archive as a consolidated information source to assist with learning and literature review in computer science. The idea is to have an organized storehouse of computer science knowledge, with links to dictionaries, tutorials, papers, bibliographies, theories, proofs, software, standards, documentation, language definitions, home pages, background, pointers to literature, commentary, FTP sites, mailing lists, open problems, calls for papers, calls for participation, commercial services, publishers, intelligent search tools, digital tutors, and so on.

Though the archive will have to be largely hand-crafted, this does not mean there cannot be tools to assist users in organizing the information, by doing some half-intelligent clustering of the data and then allowing for incremental, manual configuration.

Compared to the mathematics ([www.ams.org](http://www.ams.org)), physics ([xxx.lanl.org](http://xxx.lanl.org)), and engineering ([www.eevl.ac.uk](http://www.eevl.ac.uk)) communities, computer science has been slow to computerize its scholarly practices. Will computer science take the lead in computerizing its knowledge and resources?

Within computer science there are several related efforts under way, including Hypatia ([\[qmw.ac.uk\]\(http://qmw.ac.uk\)\) and an archive ACM is scheduled to announce. What I want to emphasize is the great potential for such technology to transform not just the superficial conventions of scholarly publication \(for instance, whether research is published on paper or online\), but also the very way knowledge is represented, accessed, and acquired.](http://hypatia.dcs.</a></p></div><div data-bbox=)


## The Challenge: Representing Computer Science Knowledge

There are numerous technical and political issues to resolve to realize this vision for a successful archive of computer science knowledge. Foremost among these is the development of standard computer-friendly encodings for representing and organizing computer science knowledge and resources.

In the short term, a major goal should be to represent the superficial meta content of computer science resources in order to enable intelligent searching and retrieval. A small amount of markup or metadata—along with quite a bit of manual organization of materials—should be sufficient for this purpose.

In the long term, the goal should be to represent the formal (internal) content of the stored materials, in a detailed markup language, knowledge representation language, or controlled language. Conventions of writing, research, and maybe even thinking would need to change drastically, to accommodate the requirements of computer-friendly encoding. The difficulty of this endeavor probably varies directly in proportion to the amount of content to be formally represented.

Deciding on such languages and formalisms for encoding computer science knowledge is a significant technical problem but one both challenging and worth pursuing.

The broader goal is to use computer science as a testbed for technology to enable the computerization of many areas of human knowledge. Indeed, the trend toward computerization has already begun, and I believe we have reason to look forward with great hope to its mature fruition. 

---

**DON SMITH** ([dsmith@cs.waikato.ac.nz](mailto:dsmith@cs.waikato.ac.nz)) is a lecturer at Waikato, Hamilton, New Zealand.