UT Libraries Electronic Document Delivery Service

Thank you for using ILLiad and UT's electronic document delivery service. Attached, please find a portable document format (pdf) version of an item you requested through Interlibrary Services (www.lib.utk.edu/~ils).

- This item will remain in your ILLiad account for thirty (30) days. After 30 days, ILLiad will automatically delete the file, removing the item from your account. You may delete this item any time prior to end of the 30-day expiry period.

- As Interlibrary Services (ILS) performs only minor editing of electronic documents prior to delivery (e.g. rotating, resorting pages), ILS delivers items 'as is.' ILS can't control the quality of the initial imaging performed by the lending library. ILS passes along what the lending library provides. If the quality of this document fails to meet your needs, let ILS know and ILS will find a remedy.

- Refer questions and comments concerning electronic delivery and ILLiad to ILS at ilsmail@aztec.lib.utk.edu or (865) 974-4240.

Markup Meets the Mainstream: The Future of Content-Based Processing

Charles Hill
Department of English
University of Wisconsin
Oshkosh, WI 54901
hill@uwosh.edu http://www.uwosh.edu/departments/english/

In their 1990 article, DeRose et al. stated that there are now clear signs that OHCO-based text processing will soon be reaching the general text processing markets (DeRose et al., p. 18). The authors did not mean that millions of office workers and school children would be learning to type tags into their documents. Rather, they were predicting that new WYSIWYG editors would make content-based markup languages transparent and easy to implement. Once these editors made content-based coding as simple as using a word processor, the text-producing world would give up word processing, which treats text as a stream of characters, and come to see text as it really is—an Ordered Hierarchy of Content Objects.

In 1997, it is certainly true that content-based markup systems (usually referred to as "descriptive markup") have become familiar to those who create or produce large, complex documents for a living. However, to the rest of the world—that is, in the thousands of offices and schoolrooms in which vast amounts of text are being produced and distributed for a variety of purposes—descriptive markup systems remain largely unknown. This may be partly because WYSIWYG editors for SGML markup systems are not widely marketed. However, the real obstacles preventing descriptive markup systems from penetrating the general text processing market may be more complex.

The authors make a strong case for conceiving of text as a hierarchy of content objects, but they perhaps do not appreciate what a dramatic paradigm shift they are asking of today's readers and writers to begin conceiving of text this way. Our conceptions of the world are shaped by the tools we use to manipulate it, and our conceptions of the nature of text are shaped by the tools we use to create, alter, distribute, and access texts. The generations growing up with word processors know that text can be formatted in various ways for various purposes, but most writers (even good ones) cannot always articulate their reasons for using a particular font or for making certain words bold because, in everyday practice, they dont have to. They just tell the word processor to bold a particular word, and it does it. The tools that most writers use emphasize the way the text is displayed, not the functions of various parts of the text or the ways in which those parts relate to each other. I am not saying that writers never think about a text's structure or content. Rather, the tools they have grown up with have not prompted them to think of a text's structure and content as being inherently related to the formatting decisions they make.

DeRose et al. discuss many practical benefits for using content-based markup languages, but they do not really discuss the effects on the writer that a shift from a character-based to a content-based paradigm would entail. If all beginning writers were taught to specify the function of various text objects and defer all formatting decisions until the last stage of the writing process, then how might this affect their conceptions of text—of what text is? And how might such a paradigm shift affect textual practice, or writers' decisions about page design and formatting? One possible outcome of such a shift might be more deliberate and careful formatting, since all formatting would be driven by previous decisions about text structure and function. Anyone who has taught introductory courses in technical writing knows that beginning writers often overuse new formatting abilities as they learn them, and their first efforts are often awash with various fonts, graphics, and formatting tricks, incorporated for no apparent reason. (A quick browse through the World Wide Web reveals even more elaborate instances of unbridled formatting, complete with blinking text, multiple frames, and useless animations.) Perhaps—just perhaps—writers who are exposed to content-based markup systems early in their educations would be more likely to understand and

appreciate the teacher's often-repeated and often-ignored pronouncement, "form follows function." There is some anecdotal evidence to indicate that this may be the case; even professional writers report that, when they begin using SGML-based markup systems, they benefit from being forced to think explicitly about the structure and function of the various textual elements being marked.

SGML applications are finding their way into the technical writing classroom, so those students should already be realizing the benefits of being exposed to a content-based text processing system. However, in their article, DeRose et al. were especially targeting academic scholars, who produce prodigious amounts of text, yet don't consider themselves "professional writers" or "technical writers." And if scholars should be using content-based systems, then so should their students, as well as other professionals who produce a lot of text. Yet it remains to be seen whether the majority of the text-producing population will make the switch from word-processing to content-based processing anytime soon.

There is, of course, one markup system that has spread far beyond the workplaces of professional and technical writers—HTML, the language of the World Wide Web. However, HTML is not a content-based, or descriptive, markup system. It is a procedural system—i.e., it is used to tell web browsers how to display text and graphics, and what to do when the user clicks on an object. This distinction must be explained (sometimes many times) to writers whose first enounter with a markup language is with HTML. Because it looks like a descriptive system but is not one, the widespread dissemination of HTML may actually hinder any efforts to lead writers through a paradigm shift to content-based text processing.

Because it is a procedural system, HTML reinforces, rather than challenges, the writers tendency to think of the text as a stream of formatted characters. To make matters worse, HTML is a mixture of descriptive tags that define content objects (e.g., <TITLE>, <ADDRESS>, <CAPTION>) and "procedural" tags that define formatting commands (e.g., <B>, <I>). If the goal of using markup systems is to prompt students to think about texts as content objects rather than as streams of characters, the muddled nature of HTML can only make this shift even more difficult to attain.

Because it looks like a descriptive system but is not one, the widespread dissemination of HTML may actually hinder any efforts to lead writers...to content-based text processing.

Early exposure to HTML may also discourage writers from learning other markup systems. Because it is used (many say misused) as a formatting language rather than as a descriptive markup system, HTML can seem unnecessarily crude and limiting—a Rube Goldberg approach to accomplishing what would be relatively simple and straightforward with a good desktop publishing program. New WYSIWYG editors for HTML appear every week, holding out the promise that coding web pages really will be as simple as word processing. However, these editors have difficulty keeping up with the constantly changing HTML standard, so most people who want more than a very basic web page still end up doing some coding manually. (Many writing instructors agree that students learning web authoring should have some knowledge of basic HTML tags before they are allowed to use a WYSIWYG editor. Clearly, the day of "transparent" HTML is not yet here.) Most web authors view HTML as something to be endured in order to be able to publish on the web, not as a superior system for producing and manipulating texts.

Finally, because is not a true descriptive system, HTML offers almost none of the benefits that DeRose

et al. ascribe to content-based markup languages. For instance, the authors maintain that content-based systems can make it easier to reformat documents, and allow for more sophisticated searches of the document and retrieval of specific information. However, reformatting raw HTML documents is certainly no easier than reformatting a document with a good word processor or page design package, and while search engines for the Web continue to improve, they still do not provide the type of advanced, sophisticated searching that the authors promise. The end result of all of this is that we have an entire generation of writers growing up with a markup system, but because it is a procedural system, it does not encourage writers to learn other markup systems, and it does not prompt writers to think of text as an Ordered Hierarchy of Content Objects (DeRose et al., 1990, p. 22). If anything, it reinforces writers' tendencies to treat text as simply "a stream of characters" that may be formatted in various ways (DeRose et al., 1990, p. 9).

However, HTML may yet turn into a descriptive system. The latest version of the Microsoft Explorer browser gives web authors the ability to use stylesheets, and Netscape is planning to offer stylesheet capability in a future release. A stylesheet is a separate document that may be referred to by one or more HTML documents; the stylesheet allows the author to, in effect, redefine the HTML tags for specified pages. Reformatting an entire website may entail simply revising the one stylesheet that is referred to by all of the pages on that site. This is the way descriptive markup systems were designed to work, and it may prompt web authors (with some helpful guidance) to use HTML tags to specify the function and content of the elements in their web pages, relying on the stylesheets to tell the user's browser how to display these various elements. When stylesheets are as common as browsers, then the popularity of HTML may actually facilitate, rather than hinder, the conceptual shift from a character-based to a content-based text processing system.

There is another development in the HTML world that may raise a whole new set of questions about the process of designing and formatting documents. In a working draft of a report still in progress, the World Wide Web Consortium recommends that future versions of web browsers should give users the ability to turn off the web author's style sheet, and users should be able to create their own personal styles for displaying the pages that they access (WWW, 1997).

In other words, after the web author has painstakingly defined the stylesheets in order to create the formatting characteristics that he or she feels are most effective, the user can then replace all of these formatting decisions with his or her own prefer ences. Taking the control over the "look and feel" of a document away from the producer of the text and giving it to the user is a dramatic shift in the dynamics of the reading/writing process. Giving users such control over the text was unheard of in the print-dominated world, but it will become more common as the shift to a screen-dominated paradigm continues to accelerate.

Of course, the Web is just one medium for distributing texts, but once users of documents experience this level of control over the web pages that they read, they will come to expect and demand it from other document sources. As electronic reading and the distribution of texts over computer networks become more common, inexorably pushing print to the margins of textual practice, users will gain more and more control over a text's characteristics. This control will include (but will not be limited to) format and design characteristics (Hill & Mehlenbacher, 1996). The effects of this shift of control from producers to users will be profound. Among other things, the writer's formatting decisions may become irrelevant, or a whole new technology may be developed that will allow writers to anticipate and take advantage of the types of formatting decisions that might be made by the user. In any case, this shift in the locus of control will be facilitated by the ability to remove formatting commands from the content document and place them in a separate stylesheet; in turn, as users demand more control over texts, this separation of design from content will become a necessity rather than an option.

Content-based markup languages do not solve all of our problems, and may not be appropriate for all writing tasks. For the production of some types of texts—e.g., short documents which must be produced quickly, or documents with many complex graphics—it may make more sense to take advantage of the formatting capabilities of a good word processor or desktop publishing program, rather than go through the relatively cumbersome process of defining a specific SGML system, or accepting the

limitations of an existing one. When visual impact is important and a document must be produced quickly, it may make more sense for an experienced writer to simply say, "let's put this line in 18-point Times Roman and bold it, without having to define a descriptive tag for that one line.

Descriptive markup systems clearly have benefits for some types of writing; the authors describe several of these benefits in their article, and the accuracy of their assertions is demonstrated by the rising popularity of SGML systems among users in business, industry and government. But the authors also predicted that the entire populace that makes up the "general text processing markets" will and should turn their backs on word processing in favor of content-based markup systems. In response to this prediction, I have tried to concentrate on the promises of such a revolution and some of the challenges to it, speculating on what may happen when content-based markup systems spread into the mainstream, beyond the worksites of professional writers, and into the lives of the millions of others who produce text as just a part of their professional and personal lives.

While we do not yet know how long the World Wide Web will survive in its present form, the Web will certainly continue to influence the nature of textual practice for the near future, and it provides one model of the "mainstreaming" of textual markup systems. Therefore, it seems useful to examine the problems and successes of HTML, and to look at some of the current trends in the HTML world, to see how markup languages in general may fare as they continue to work their way into the mainstream culture.

References

Derose, S. J., Duran, D. G., Mylonas, E., & Renear, A. H. (1990). What is text, really? *Journal of Computing in Higher Education*, 1(2), 3-26.

Hill, C., & Mehlenbacher, B. (1996). Readers' expectations and writers' goals in the late age of print. Proceedings of SIGDOC'96 (pp. 257-266). Research Triangle Park, NC: Association for Computing Machinery.

World Wide Web Consortium (1997). HTML and style sheets. <http://www.w3.org/pub/WWW/TR/WD-style> (1997, May 15, in progress).