From Generic to Descriptive Markup: Implications for the Academic Author

Tuija Sonkkila
Helsinki University of Technology Library
Otaniementie 9
FIN-02150 Espoo
Finland

Abstract—Electronic publishing is confronted with a multitude of demands and hopes, expressed by users on one hand, and by institutions on the other. One of the key issues concerns long-term availability of digital information. In addition, research findings indicate that users would like to place more detailed full-text information retrieval requests. Due to differing interests, some users focus their attention to figure captions, others in tables or bibliographies, to name just a few examples. Furthermore, there is a wish to deliver publications on many platforms, which asks for suitable mechanisms of combining information with different sets of output specifications. In all these three cases, the capabilities of todays desktop editors fall short. Yet they are among the most frequently used tools to produce scientific publications.

It is claimed that the answer would lie in the use of structure-oriented editors and descriptive, platform-independent markup. But the move is not a trivial one. One of the first big challenges is the author himself. To what extent is he willing to modify his working habits? Does he accept the possibility of letting someone else define the layout of his work? Another major issue is the publication process. The nature of changes in work-flow are as much organizational as they are technical.

This paper describes some of the lessons learned in *HUTpubl*, a project conducted by the Helsinki University of Technology (HUT) Library. The goal of the project is to establish an SGML-based (Standard Generalized Markup Language) publishing model for HUT scientific publication series. The paper further elaborates on findings in other related projects and research activities.

INTRODUCTION

An electronic document has no inherent structure other than that of a linear character string. Some basic techniques have to be established, if any parts of the document are to be identifiable. Markup - or tagging - is one such technique. Marcoux and Sévigny [1] distinguish between three types of markup: hardware-oriented, generic, and descriptive. In hardware-oriented markup such as in the use of escape sequences, the form and meaning of the tags are dictated by the equipment used. Generic, or procedural markup, refers to e.g. formatting codes generated by software such as word processors. Codes are translated to hardware-oriented markup by drivers. Descriptive markup on the other hand identifies all structural elements in a document but does not impose any procedures upon them. Content and processing are thus separated. The way elements are treated depends on the application that is used to process the elements.

Descriptive markup is used in applications based on the SGML metalanguage (Standard Generalized Markup Language, ISO 8879:1986). These applications are also called document-type definitions (DTD). They define all elements that a certain class of documents may include and the rules that markup of these elements has to follow.

SGML applications are primarily used in industry where documentation is either a competitive factor, central for product maintenance, or subject for extensive data exchange. Literature on SGML consists of numerous books, conference proceedings, journal articles, and online resources. The reader is advised to consult the most comprehensive and up-to-date source for all SGML-related information, Robin Cover's SGML/XML WWW (Word Wide Web) page [2].

By definition SGML provides a standardized way to build document repositories that are independent of the constantly changing software and hardware environment. But the task is not a trivial one. The Task Force on Archiving of Digital Information notes in its Report [3]: "Refreshing digital information by copying will work as an effective preservation technique only as long as the information is encoded in a format that is independent of the particular hardware and software needed to use it and as long as there exists software to manipulate the format in current use". The report further claims that much of the recent work on digital libraries has been notably silent on the archival issues, and goes on encouraging e.g. librarians to take an active role in their parent institutions in issuing guidelines for digital preservation, and adopting data standards.

PROJECT DESCRIPTION

In early 1997 the Helsinki University of Technology (HUT) Library launched a project called HUTpubl. The project had four technical objectives. First, to design an SGML application, a DTD, for HUT publications. Second, to test the usability of the DTD against a number of real-life examples. Third, to evalute two different word processors in structured editing. Fourth, to design conversion procedures from SGML to HTML (HyperText Markup Language) format for WWW delivery. In this paper only the first two are discussed in more detail.

In addition to these specific objectives the project targeted to two unwritten, more abstract ones. First, to gather knowledge about SGML implementation issues. Second, to promote the importance of publishing standards at HUT. It should be noted here that the HUTpubl project was practical in nature rather than based on strict research principles. Methods and findings described below should therefore be evaluated against this background.

0-7803-5101-0/98/$10.00 © 1998 IEEE

Helsinki University of Technology (HUT) is the biggest technical university in Finland. It is also the oldest, celebrating its 150th anniversary in 1999. HUT consists of 12 faculties, and the number of under- and postgraduate students is roughly 12 000.

As a publisher of traditional print publications HUT is a big non-commercial one. Annually, the number of individual titles exceeds 400, published in over 200 different scientific publication series, consisting of technical reports, theses, etc. The publishing process is decentralized. In other words, there is no central publishing unit at HUT, no "HUT University Press", as it were. In this respect, HUT differs from many other Finnish universities. Traditionally also, the twelve faculties at HUT are very indepent when it comes to publishing procedures. The only HUT-wide publishing standard concerns layout of the cover of the print version.

DTD TEST SETUP

The DTD was designed in 2,5 months after a one month document analysis. At this stage, there was no user participation. The overall correctness of the DTD was tested when a pilot document - a licentiate thesis - was structured by the project staff. The first objective evaluation of the DTD took place during the test period that lasted for five weeks.

The test group consisted of three volunteers, one male researcher, and two female secretaries, representing three different departments of HUT. All had a background of having used PCs for several years in various work-related activities. They were now asked to use the HUTpubl DTD by adding, retrospectively, a structure into three documents, one document each. All documents were already in an electronic format and paper versions were also available. The testgroup was told that document output properties were specified in document template files to be used. The idea was to concentrate in the document structure. One of the secretaries was to work with Microsoft Word on a laptop computer. The other two had FrameMaker+SGML (FMSGML) from Adobe installed on their desktop machines.

In the case of Word, adding structure meant selecting appropiate styles from a template file to logically different parts of the document, e.g. titles, sub-titles, paragraphs, and lists. These styles would then be converted to respective SGML elements with FMSGML by the project staff.

FMSGML is a hybrid program where a structure-oriented word processor has been added to a desktop publishing environment. FMSGML provides different views to the document, e.g. a hierarchical one in the form of a tree with collapsible branches. The DTD is not visible to the author but acts rather like a supervisor on the background, reminding him or her about which elements could be chosen in any given time. Because of a built-in SGML parser, or DTD validator, the author can validate the conformity of the text against the DTD chosen.

SGML was a new concept for the testgroup. Therefore, a 1,5 day tutorial on SGML principles in general and the HUTpubl DTD in particular was arranged, added with a half-day training session with FMSGML.

The test arrangements included weekly sessions, an email list for questions and answers, and a Frequently Asked Questions file kept on the local WWW server. At the end of the five week editing period, the testgroup was given a questionnaire comprising of both closed and open questions about the DTD, about Word and FMSGML, and about the test in general. During the closing session an informal interview was performed based on the results of the questionnaire.

Initially, the test was expected to shed light upon two practical questions. First, was the DTD suitable for authoring, i.e. were element names understandable, were some elements missing, some unnecessary? Second, were FMSGML and Word feasible in adding structure to the document? Possible findings to these questions should then be used as practical guidelines in determining whether the HUTpubl DTD needed major updates - minor had been made during the test already - and in deciding which kind of word processors the project should consider using in the future.

FINDINGS AND DISCUSSION

During the first weekly sessions it seemed reasonable to assume that the test objectives would be met in schedule. There seemed to be no bigger problems related to the DTD, and initial technical difficulties in FMSGML installment were under control. FMSGML users showed interest in looking at different views of their document and used actively the ability to manipulate text in structural blocks. The only prevailing negative issue was Word. The testgroup member using Word on the laptop claimed that selecting styles from the long, scrollable drop-down list was cumbersome. She also disliked the feeling of "working in the dark", i.e. not being able to validate her work.

Although technical aspects of structured editing seemed to be manageable, at least in test settings, at the same time the challenge of implementing HUTpubl deliverables at HUT became bigger. In other words, the non-technical aspects of SGML-based publishing grew in importance. The following dicussion sums ups the issues that were dealt with in testgroup sessions, during the closing interview, and in a number of informal, undocumented discussions between the project staff and HUT faculty.

The paper metaphor

The member of the testgroup who was to use Word had no working knowledge of Word styles beforehand. This came as no surprise. Even though there is little empirical research on the actual use of styles in word processors, there is recent evidence [4] that styles are overlooked. Part of the problem is said to be the paper metaphor communicated by WYSIWYG (What You See Is What You Get), because in the paper format the use of styles makes little sense. Styles are thought to be paper output properties only [ibid.].

The paper metaphor was present during testgroup sessions also. All testgroup members consulted frequently the printed version of their document which led them to ask about how some typographical part of the text ought to be tagged.

Note that although one of the basic principles of SGML is to keep content and output separate, in practice this is not categorically the case. DTDs often include elements that are used e.g. for emphasis. This has some interesting implications to the division of work. Assuming that tagging would be done not by the author himself but by a technical writer or coder, it should be made clear as to how much freedom for judgement the coder is allowed to have in interpreting the text. Which text fragments ought to be taken as significant "rhetorical" elements and given a respective tag, and which not? In practice this would mean frequent discussions between the coder and the author.

Closely related to the issue of coding is a more fundamental question: when is the work considered published? If it is officially declared published before tagging begins, are modifications in the document structure allowed at all? Electronic publishing policy might perhaps need new concepts in this respect. One alternative is to apply concepts from the print era. Linköping University Electronic Press in Sweden for instance has stated that 'publishing' in their case coincides with the paper version coming out off the press [5].

DTD and intellectual work

The HUTpubl testgroup did not make experiments in using the DTD or styles while writing a new document. For that much more further testing and research is needed. Indeed, the most serious criticisms towards SGML applications have been expressed by those who claim that the way the DTD dictates the inherent structure of the work endangeres the heuristic writing process of the author [6]. In categories of bias in computer system design, Friedman and Nissenbaum [7] refer to the same danger by defining formalization of human constructs as one sub-type of technical bias.

It is true that the DTD stands in a central position. The author is not able to add new elements or apply elements where they are not allowed. All the author can do when confronting a DTD barrier is try to use some other element. He or she may also choose to ask for a modification of the DTD to gain more freedom or, to get the desired new element added. Whether this succeeds (and how quickly) depends on the complexity of the SGML production line. This in turn is dependable on the size of the organization, volume of the SGML document base, etc.

The SGML standard itself has been a target of criticism. It is claimed to be too complex and to include constructions that real-life SGML applications should avoid. In the late 1990s criticism lead to concrete work organized by the World Wide Web Consortium (W3C) for designing a lighter standard especially for WWW delivery. XML (eXtensible Markup Language) was given the status of W3C Recommendation in February 1998 [8]. One of the novelties of XML is the optionality of the DTD. The author might like to omit the DTD entirely, or create it afterwards when the structure of the text has become clearer [9]. It is tempting to imagine that this model might gain foothold in academe provided that future SGML/XML tools are capable of storing, querying, retrieving and delivering data in a limitless number of different DTDs. Another future prospect could be to use XML for designing more flexible DTDs.

Note that unlike in technical documentation where SGML markup is primarily used for ensuring data correctness, SGML in academic publication systems ought to be a mechanism for building semantically richer document databases for future research, not a tool for supervision. Like C. M. Sperberg-McQueen eloquently said in his closing remarks at SGML'92 [10]: "...part of its [SGML's] accomplishment is that by solving one set of problems, it has exposed a whole new set of problems. Notation is a tool of thought, and one of my main concerns is to find ways in which markup languages can improve our thought by making it easier to find formulations for thoughts we could not otherwise easily have."

DTD modifications

Modifications of the DTD are inevitable but like changes in a database structure they are costly and pose risks to data integrity. A SGML database, whether it is a actual database (e.g. relational) or a collection of files in SGML format, has to be parsable against the DTD. This means that after every change to the DTD the administration has to be sure that all existing data conforms to this new version of the DTD, that data is valid.

DTD modifications may also lead to potential occurences of differing interests. From a technical point of view, the stricter the DTD is at the beginning, the easier it is to add more freedom of choice into it afterwards [11]. But from authors' point of view a more relaxed DTD right from the start might perhaps be more desirable. Finally, looking at the issue from the organization management's point of view, it is assumable that it would be in their interest to ensure that the SGML implementation phase does not generate too much negative feedback. It is thus fairly clear that user participation in DTD design and evaluation is crucial.

In an 18 month project conducted by the University of Oslo a system for decentralized course catalog production was created, evaluated and established [12]. During the pilot phase different opinions arose between system developers and authors about the nature of the DTD. System developers, consisting of staff from the local computing centre, proposed a fairly strict DTD whereas authors were in favour of a more flexible one. Authors, working partly under the supervision of the central administration, came from different faculties and departments of the university. Author's opinions were gathered during unofficial discussions and through participatory observation, and the DTD was modified accordingly, based both on these findings and on later editions of the course catalog.

The HUTpubl DTD was also modified during the test period. It should be stressed here though, that given the DTD

will be taken into production, a more comprehensive evaluation of the DTD is probably necessary. Like Kondrach reminds [13], the testgroup ought to have a clear understanding of the whole SGML publishing process and the status of the DTD in it. As discussed above, the HUTpubl DTD formed a part of a technical pilot project where e.g. workflow issues were not a major issue.

Only one DTD might prove to be insufficient altogether. Multiple DTDs, all tailored to different authoring clientele and for different functions might offer a solution to the difficult problem of finding a suitable DTD for all [14]. The drawback is that more than one DTD will substantially increase the burden of DTD version control and subsequently SGML database management.

Researchers as authors

There exists hardly any literature about researchers as authors of structured text. One reason for this may be the small number of SGML implementations in higher education in general. Note that the role of SGML in projects launched and propagated by the Text Encoding Initiative (TEI) in humanities [15] is a central tool for research rather than a publishing tool for documents. Project deliverables themselves are usually SGML databases such as text corpuses. But whether any e.g. research documents published by these projects are generated through a SGML production line, is not known.

Contrary to administrative or clerical staff as in the example of Oslo above, researchers in higher education are considered "free" actors. As such they are sensitive to guidelines and recommendations for the process of their intellectual work, especially placed by the university administration or information management bodies. Nevertheless, researchers do accept guidelines coming from outside, notably from the area of scholarly journals. To submit material researchers have to follow detailed format descriptions expressed by the Editorial Board.

One suggested alternative for getting faculty interested in and becoming familiar with structured editing, its tools, methods and possibilities, is to arrange special occasions for coding [16] or even possibilities to launch experimental publishing projects of its own [e.g. 17].

CONCLUSIONS

In this paper, the initial test results of the HUTpubl DTD have been presented, added with remarks and thoughts, induced by discussions with HUT faculty, about what kind of consequences structured editing would have on the academic author.

SGML is a mature international standard for document processing, and as such offers a robust technical solution for e.g. archiving documents in higher education. But technology is only one side of the coin. Division of labor asks for careful considerations, and user participation early in the design process is essential. Futhermore, the heart of the SGML application itself, the DTD, poses a dilemma. Academic freedom may not tolerate a strict DTD, not even a flexible one. Frequent modifications to the DTD on the other hand may prove fatal to SGML document management. The XML standard developed by W3C gives promises in this respect, but much further research is still needed, especially in finding out how academic authors produce text, and for what purposes.

ACKNOWLEDGMENT

This work was made possible thanks to the kind support of Dr. Sinikka Koskiala, the Head of the Helsinki University of Technology Library. The author wishes to thank Jörgen Westerling for the excellent HUTpubl prototype, and Sanna Haapio, Tiina Hartikainen, and Pekka Eloranta for active participation in the HUTpubl DTD testing. Special thanks to the HUTpubl Advisory Board, and to Timo Vendelin from Remtec Systems Ltd for insightful consultation. Finally, thanks to the SGML Users' Group of Finland for many thought-provoking seminars.

REFERENCES

[1] Y. Marcoux, and M. Sévigny, "Why SGML? Why now?," *Journal of the American Society for Information Science*, vol. 48, pp. 584-592, July 1997.

[2] R. Cover, *The SGML/XML Web Page*. [Online]. Available: <URL: http://www.sil.org/sgml/sgml.html>

[3] Task Force on Archiving of Digital Information, *Preserving Digital Information, Final Report and Recommendations*. [Online]. (May 20, 1996). Available: <URL: http://lyra.rlg.org/ArchTF/index.html> [1998, April 13].

[4] P. Sørgaard, and T. I. Sandahl, "Problems with styles in word processing: a weak foundation for electronic publishing with SGML." Cop. IEEE 1996. [Online]. Available: <URL: http://internet.adb.gu.se/publications/6/pap.html> [1998, April 7].

[5] E. Sandewall, "Strategies and policies of Linköping University Electronic Press," in *Linköping Electronic Articles on Academic Policies and Trends*. Vol. 1(1996): 1. [Online]. Available: <URL: http://www.ep.liu.se/ea/apt/1996/001/> [1998, April 12].

[6] S. A. Selber, "The OHCO model of text: merits and concerns," *The Journal of Computer Documentation*, vol. 21, pp. 26-31, August 1997.

[7] B. Friedman, and H. Nissenbaum, "Bias in computer systems," in *Human values and the design of computer technology*, B. Friedman, Ed. New York, NY: Cambridge University Press, 1997, pp. 21-40. [CSLI Lecture Notes Number 72].

[8] W3C, *Extensible Markup Language (XML) 1.0*. [Online]. Available: <URL: http://www.w3.org/TR/1998/REC-xml-19980210> [1998, April 7].

[9] S. J. DeRose, D. Durand, E. Mylonas, and A. H. Renear, "Further context for 'What is text, really?'," *The Journal of Computer Documentation*, vol. 21, pp. 40-44, August 1997.

[10] C. M. Sperberg-McQueen, "Back to the frontiers and edges, " in *Closing remarks at SGML'92: the quiet revolution*. [Online]. Available: <URL: http://www.sil.org/sgml/sgml92sp.html> [1998, April 5].

[11] B. E. Travis, and Dale C. Waldt, *The SGML implementation guide, a blueprint for SGML migration*. Berlin : Springer, 1996.

[12] A. E. Jenssen, and T. I. Sandahl, "Conflicts between the possibilities and the reality in the field of structured electronic documents, experiences from a large-scale SGML-project." [Online]. (No date).

Available: <URL: http://internet.adb.gu.se/publications/14/> [1998, March 18].

[13] G. Kondrach, "DTD testing : find the devils in the details." In *SGML/XML'97 Conference Proceedings, December 8-11, 1997, Washington, D.C.* Alexandria, VA: Graphic Comunications Association 1997, pp. 133-141.

[14] K. F. Best, "Just how many DTDs do you need?" In *SGML'96 Conference Proceedings, November 18-21, 1996, Boston, MA.* Alexandria, VA: Graphic Comunications Association 1996, pp. 131-139.

[15] The Text Encoding Initiative. [Online]. Available: <URL: http://www.uic.edu/orgs/tei/>.

[16] E. Øverby, "Which demands do an SGML-organisation (users) have to the developers of the SGML-system," In *SGML'96 Conference Proceedings, November 18-21, 1996, Boston, MA.* Alexandria, VA: Graphic Comunications Association, 1996, pp. 597-600.

[17] *The Electronic Text Center.* [Online]. Available: <URL: http://etext.lib.virginia.edu/>.