BIJLAGE FF

SGML and TEX in scientific publishing

N.A.F.M. Poppelier
Elsevier Science Publishers
Physical Sciences and Engineering Division
R&D Department
n.poppelier@elsevier.nl

Abstract

Elsevier Science Publishers has for a few years investigated the possibility of accepting compuscripts, a manuscript in electronic form, created with TEX, LATEX and a few other text processing systems, and converting these to *SGML* form. This paper will discuss the current status of these activities, the reasons for converting compuscripts to *SGML* form, and the various ways in which TEX is used.

Introduction

Until a few years ago the results of scientific research were always published in the following way: the author writes an article using whatever method he prefers. Some authors use TEX, some authors a word processor, others prefer a typewriter, and there are still authors who consider the pen to be the best desktop publishing tool that is available.

When the author is satisfied with his work, he or she submits the manuscript, say, to the editor of a journal, usually in duplicate or triplicate. The editor enters the manuscripts into his administration, and sends a copy to one or more "referees," who review and judge the manuscript. This is one of the places on the long road from manuscript to published paper where the publisher adds something to the paper. The system of peer reviewing, which provides a kind of scientific certification, is one example of "added value." The quality of the printed product and the world-wide distribution of an article in a journal can also be labelled "added value."

The referee(s) can either recommend acceptance of the manuscript for publication, possibly with suggestions for alterations, or rejection of the manuscript. If it is accepted, and after the author has put it into final form, the manuscript is sent to the publisher, where it is adapted to the style of the journal by a technical editor. Then the manuscript is re-typed (typeset) and, in most cases, stored in electronic form. Because of the fact that the material has been re-typed, it may contain errors. These errors are removed by proof-reading, which is a task that can be performed, for example, by the technical editor or the author.

Finally, the paper is published, together with other papers, in the journal and after a while the author sees his article in print.

The time that passes between submission of the manuscript and publication of the article in printed form can vary between a few weeks and a year, or even longer. In some fields of science this period of time is too long and authors sometimes ask publishers why this time cannot be reduced.

Modern Times

The conventional method of publishing, which I have sketched above, can probably not be made any more efficient by using conventional means. However, many authors use their own text processing systems to produce their papers and reports. This text can be transmitted in electronic form. In principle, this makes it possible to

- publish the paper within a shorter length of time;
- extract bibliographic information and abstracts, and store this information in a database.

In the first case we see that, by making use of manuscripts in electronic form, which I will call *compuscripts* in the rest of this paper, that is by making use of modern electronic means, the conventional method of publishing scientific journals *can* be made more efficient. Furthermore, the production of secondary publications, such as abstract journals, can be made more efficient since the abstracts do not have to be re-typed.

An important consequence is that the publisher can now add extra value to the journal or to his entire range of scientific publications, by making the material available in various electronic forms, such as hypertext books, databases on CD-ROM, etc.

However, a publisher can only accept a compuscript if the instructions of the text processing system of the author can, in one way or another, be converted to that of the publisher, which in most cases is a computer-driven phototypesetter. Since a publisher wants to work as efficiently as possible, this conversion must be automatic or nearly automatic. This means that the compuscript must be prepared by the author according to a set of rules. In most cases, of course, a technical editor still has to read a draft printed on paper to mark appropriate changes to the text, the typography or the layout. Especially if the paper contains tables or mathematical formulae, this is an intricate task.

Research at ESP

Elsevier Science Publishers (*ESP*) has for a few years been investigating the possibility of accepting compuscripts created with modern text processing systems. The aim of our present research is to develop a method for manipulating the compuscripts in such a way that the material can be published more efficiently than before, and is also stored for re-use at a later stage.

We take the view that the transition from manuscript-based to compuscript-based publishing must at least preserve the flexibility of the current submission procedures and, where possible, improve upon it. This means, for instance, that the author should not be bothered with compuscript preparation instructions that are significantly different between publishers, or even between the various journals of one publisher. It also means that publishers should be able to accept compuscripts produced by a range of text processing systems, and submitted either by electronic network or on a diskette, via ordinary mail.

SGML

As Figure 1 shows, the Standard Generalized Markup Language (SGML) [1, 2, 3] plays a central role in our approach. SGML promises to become a very important tool, since it allows us to (i) separate the logical structure of the document from its visual structure, i.e. its appearance; (ii) treat all accepted compuscripts, once they have been converted to SGML form, in a uniform working environment; (iii) handle the compuscripts independently of output medium and output format; (iv) store the text, wholly or partially, in a database and use the database contents again at a later stage for various other forms of publications; (v) standardize compuscript input formats between the various scientific publishers, thus helping to preserve the flexibility of submission mentioned above.

It is not our intention to ask authors to submit SGML-coded compuscripts yet. However, we foresee that, as a consequence of the fact that the departments of the US Government as well as major industries have adopted SGML as a standard for document interchange, developers of text processing software will provide their customers with facilities for conversion from their particular text format to SGML. Authors will then be able to produce their compuscripts with the text processing system of their choice and to provide the publisher with an SGML version without additional effort.

Until then, in order to have all material submitted in the form of compuscripts available in SGML-coded form, conversion facilities must be developed. The scheme we would like to realize is indicated in Figure 1.

On the one hand, Elsevier Science Publishers wants to allow authors to use their favorite text processing system. On the other hand we know that there are dozens of text processing systems that are being used by authors in various disciplines.

This means that we have to make a choice. Two factors influence this choice: (i) the text processing systems chosen must correspond to a fair proportion of the total number of authors, and (ii) a compuscript prepared with a certain text processing system must, ideally, be convertible to SGML form. The first point is something that should be investigated separately for every discipline. The second point implies that we must choose text processing systems that, whenever possible, produce texts where the logical structure is, at least partially, indicated. In other words: a Word compuscript based on a style sheet is to be preferred over a plain Word file, and LaTeX is preferred over plain TeX.

TeX Compuscripts

TeX will be one of the text processing systems included in our electronic publishing scheme. Later this year, as a first step towards having a TeX compuscript scheme for all appropriate journals, *ESP* will start accepting LaTeX-coded compuscripts for a few of their physics journals. For various reasons we have chosen LaTeX as the most convenient variety of TeX.

- Since according to Lamport [4], "the primary function of almost all the LaTeX commands . . . [is] to describe the logical structure of [a] document" conversion of a LaTeX-coded compuscript to SGML form is possible.
- The idea of a document style, which is fundamental to LaTeX, enables us to produce camera ready copy for a particular journal by changing only the `\documentstyle` command in the submitted compuscript.
- LaTeX is relatively easy to learn.
- LaTeX is described in a book, so that it can be considered as a *de facto* standard.
- Especially for large review articles and for books, the following LaTeX tools come in very handy
  - cross-referencing with symbolic keys,
  - automatically generated table of contents,
  - index and bibliography tools.

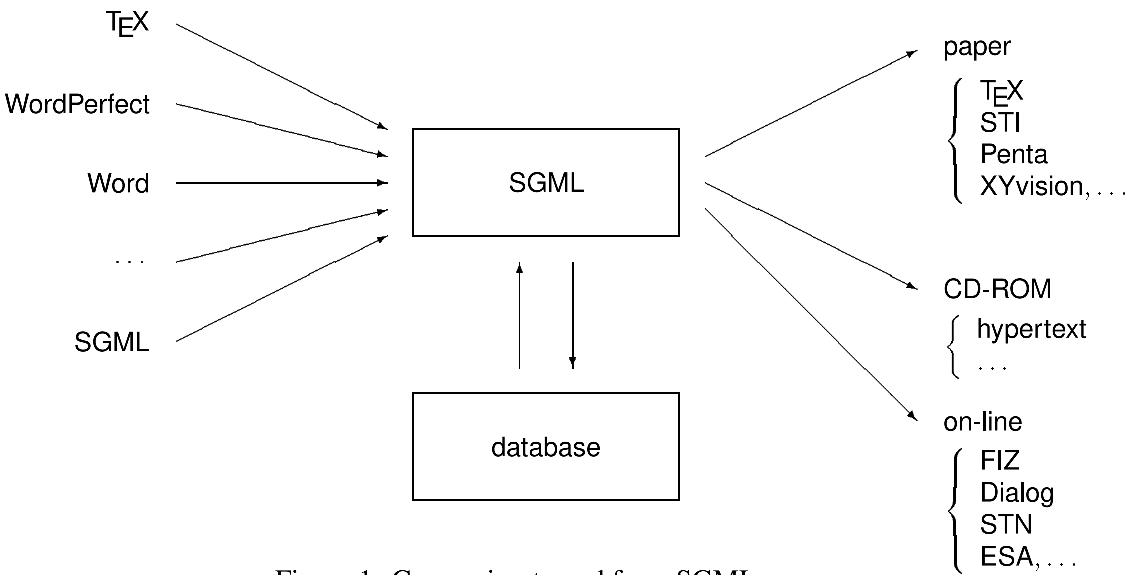

Figure 1: Conversion to and from SGML.

Status Quo

So far we have achieved several things. First of all, we have created LaTeX document styles for four of *ESP*'s scientific journals. These new document styles exactly reproduce the layout of the corresponding journals, which so far are still produced in the conventional way.

We have also been able to develop a set of programs for complete automatic conversion of LaTeX documents with mathematical formulae to SGML form. To be more precise: we convert LaTeX documents with mathematical formulae to a *document type definition* [1, 2] for scientific papers that is based on the one developed by the *Association of American Publishers* (*AAP*).

A document type definition is a description of the logical structure of a certain class of documents, using a method that resembles the specification of the syntax of a programming language. As an example, the document type definition of a novel is represented in a diagrammatic manner in Figure 2.

Conversion of LaTeX-coded tabular material to SGML form, i.e, to the *AAP* document type definition or possibly a different document definition, is currently under investigation. Another automatic conversion that is currently under investigation is the conversion from Word, with an appropriate style sheet, to SGML.

Book Projects with TeX

Recently the Physical Sciences and Engineering Division of *ESP* has published several books that have been written in plain TeX or in LaTeX, and the number of TeX-coded and LaTeX-coded books that we will publish in the near future will probably increase. Our experience is that LaTeX is an excellent tool for these larger projects due to the separation of logical structure and visual structure, the ability to do cross-referencing with symbolic keys and the presence of tools for index and bibliography; these advantages were already mentioned under the section *TeX Compuscripts*.

However, we have also noticed that making a TeX-coded or LaTeX-coded book ready for publication requires more time when the authors do not have to follow a set of instructions, but are free to type things the way they think is best.

Other Uses of TeX

Lots of programs, such as database programs, have output capabilities that are somewhat meager. After all, a database program is meant to store data, not to print it in a sophisticated manner. The fact that TeX is a programmable text formatting system makes it an excellent "print engine" for such programs. As long as the database program is capable of inserting some fairly simple texts in its output – most database programs have report generators that are able to accomplish this – TeX's capabilities as a text formatter can be used to print the output in any desired way.

An example of this approach is symbolic mathematics packages that can compute complicated formulae and generate the TeX codes for formatting them. Another example is provided by the database-publishing activities of the *Excerpta Medica Publishing Group* (*EMPG*), which is part of the Biomedical Division of *ESP*.

The *EMPG* employs a database of article openings, abstracts and citation lists that have been derived from articles in a selected set of biomedical journals. The information in this database is stored in a structured form. With this information the *EMPG* produces abstract journals that are called the *Core Journals in* $\langle \ldots \rangle$,

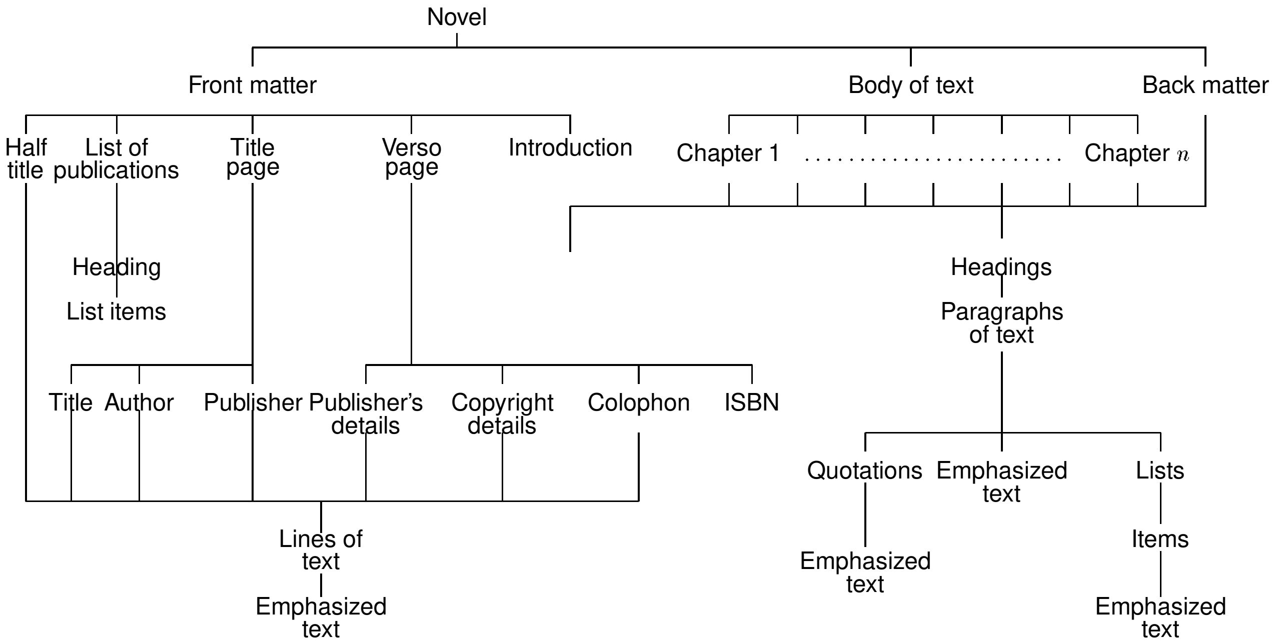

Figure 2: Document type definition of a novel.

where $\langle \ldots \rangle$ is, for instance, *Opthalmology*, *Cardiology* or *Neurology*. The *EMPG* has been experimenting with using LATEX for the production of *Core Journal*s: the relevant portions of the database are extracted and converted to LATEX. The LATEX-coded form of this information can then, in principle, be used to produce the camera ready copy for the journal, using a document style that was developed in-house.

In the near future, starting this year, a similar system will be implemented for the article openings of all journals that are published by *ESP*. The purpose of this project, which is called *CAPCAS*, is to have the article openings, i.e. title, author(s), abstract, keywords and publication history, available in SGML-coded form and to store this information in a database. From this database we can then create secondary publications of various kinds, such as volume indexes, author indexes and abstract journals. Also for this type of database-publishing, TEX is used to format the structured information on paper.

Conclusion

In this paper I have sketched the ideas that *Elsevier Science Publishers* have developed concerning the handling of author prepared manuscripts in electronic form. In this scheme, SGML will play a central role, and TEX, in one or more of its varieties, will also play a part.

Furthermore, I have given a few examples of other ways in which TEX's text formatting capabilities can be put to good use, namely in the fields of book production and database publishing.

Summarizing, I think it's no exaggeration to say that, because of its programmability, TEX is eminently suitable for large-scale text production, both directly as a text processing system, and indirectly as the output component of a database-publishing system. Considering that

- the quality of material typeset in TEX is considered satisfactory even by very demanding users,
- TEX runs on an impressive number of different types of computers, and
- TEX output can be printed on a wide range of output devices,

the number of uses of TEX will undoubtedly increase in the years to come.

Bibliography

[1] International Standard ISO 8879: *Standard Generalized Markup Language (SGML)*. ISO (1986).

[2] Martin Bryan: *SGML, an author's guide to the Standard Generalized Markup Language*. Addison Wesley, (Workingham, 1988)

[3] Eric van Herwijnen: *Practical SGML*. Kluwer Academic, (Dordrecht, 1990).

[4] Leslie Lamport: *LATEX, a document preparation system*. Addison Wesley, (Reading MA, 1986).

BIJLAGE GG

The future of TeX and METAFONT

Donald E. Knuth

3 october 1990

My work on developing TeX, METAFONT, and Computer Modern has come to an end. I will make no further changes except to correct extremely serious bugs.

I have put these systems into the public domain so that people everywhere can use the ideas freely if they wish. I have also spent thousands of hours trying to ensure that the systems produce essentially identical results on all computers. I strongly believe that an unchanging system has great value, even though it is axiomatic that any complex system can be improved. Therefore I believe that it is unwise to make further "improvements" to the systems called TeX and METAFONT. Let us regard these systems as fixed points, which should give the same results 100 years from now that they produce today.

The current version number for TeX is 3.1, and for METAFONT it is 2.7. If corrections are necessary, the next versions of TeX will be 3.14, then 3.141, then 3.1415, . . . , converging to the ratio of a circle's circumference to its diameter; for METAFONT the sequence will be 2.71, 2.718, . . . , converging to the base of natural logarithms. I intend to be fully responsible for all changes to these systems for the rest of my life. I will periodically study reports of apparent bugs, and I will decide whether changes need to be made. Rewards will be paid to the first finders of any true bugs, at my discretion, but I can no longer afford to double the size of the reward each year. Whenever I have created a new version, I will put it in the official master TeX archive, which currently resides at Stanford University. At the time of my death, it is my intention that the then-current versions of TeX and METAFONT be forever left unchanged, except that the final version numbers to be reported in the "banner" lines of the programs should become

```
TeX, Version $\pi$
```

and

```
METAFONT, Version $e$
```

respectively. From that moment on, all "bugs" will be permanent "features."

As stated on the copyright pages of Volumes B, D, and E, anybody can make use of my programs in whatever way they wish, as long as they do not use the names TeX, METAFONT, or Computer Modern. In particular, any person or group who wants to produce a program superior to mine is free to do so. However, nobody is allowed to call a system TeX or METAFONT unless that system conforms 100% to my own programs, as I have specified in the manuals for the `TRIP` and `TRAP` tests. And nobody is allowed to use the names of the Computer Modern fonts in Volume E for any fonts that do not produce identical `tfm` files. This prohibition applies to all people or machines, whether appointed by TUG or by any other organization. I do not intend to delegate the responsibility for maintainance of TeX, METAFONT, or Computer Modern to anybody else, ever.

Of course I do not claim to have found the best solution to every problem. I simply claim that it is a great advantage to have a fixed point as a building block. Improved macro packages can be added on the input side; improved device drivers can be added on the output side. I welcome continued research that will lead to alternative systems that can typeset documents better than TeX is able to do. But the authors of such systems must think of another name.

That is all I ask, after devoting a substantial portion of my life to the creation of these systems and making them available to everybody in the world. I sincerely hope that the members of TUG will help me to enforce these wishes, by putting severe pressure on any person or group who produces any incompatible system and calls it TeX or METAFONT or Computer Modern—no matter how slight the incompatibility might seem.