Editorial

In a previous issue my co-editor revealed a truth hitherto apparent only to the most expert students of textual and stylistic analysis — the EP-odd editorials, though signed by both editors, are primarily individual efforts. If there are any lingering doubts as to which editor is hammering the keyboard this time those same students can reflect that the last editorial was signed 'Furuta and Stotts' — so it must be somebody else's turn — and can then proceed with their detective work via a careful study of what follows, based on the orthography of certain non-occurring words (e.g. 'centred', 'orientated' and 'colourful').

In this issue we have included two papers (one by Warmer and van Vliet and the other by Mamrak and Barnes) which explore different aspects of the practical usage of SGML and the development of tools to facilitate the mapping of SGML tags into other forms of markup. The final paper, from Harrison and Munson, is an interesting analysis of possible algorithms for the automatic numbering of document components such as figures, tables, equations and so on.

Readers may also recall that in this new volume of the journal we are introducing an occasional unrefereed series of short features entitled 'EP-odds and ends'. Subject always to the laws of libel, and the dictatorial whims of the editors, this column will range over anything and everything to do with electronic publishing, from newsworthy 'inside information' through to philosophical speculations about the true nature of EP. The first of these columns is very much in the former category and will appear in the next issue. It consists of an article by Chuck Bigelow and Kris Holmes on the redesign of the Apple screen fonts for System 7 on the Macintosh. This, in turn, is part of an issue which will be devoted entirely to papers on typography. However, in the issue after that, we go to the opposite, 'philosophical', extreme, with a thought-provoking article from Furuta and Stotts about the future of hypertext. To set the scene for this latter article, and as a coda to the two special issues on hypertext that have just gone by, the writer of this editorial offers the following thoughts (though these are definitely under the guise of Editorial Privilege, and from a non-expert viewpoint as far as hypertext is concerned).

The topic of hypertext, and allied themes such as active documents, do seem to represent the Leading Edge (or the Wild Frontier?) of EP. They stretch the current notions of 'document abstraction' to breaking point, showing us that we still have only an inkling of what a 'document' *could* be once all the underlying technologies are mature, let alone what it *should* be. Is a hypertext to be just a static database structure which is browsed by some entirely external program? Or is the program to be bound in with the hypertext, in such a fashion that the whole document becomes a metaprogram, which can be selectively 'executed' in a multitude of ways? Indeed, if this metaprogram view has any substance to it then it follows that the true potential of hypertext will only be apparent when computer scientists have discovered much more about the fundamental nature of programs and programming. So far, hypertext, and EP generally, have revealed all the same intriguing but infuriating problems encountered in the wider arena of computer science. A suitable level of abstraction confers enormous benefits in devising simple and powerful documents (but so often the abstraction only feels '90% correct',

leaving the remaining 10% maddeningly awkward and inelegant). Nonetheless, just as computer scientists have had to solve the problem of mapping abstract functional languages onto unyielding and unhelpful hardware, so also will the EP community have to grapple with the difficulties of 'link-editing' and presenting suitable subsets of some highly abstract hypertext on conventional devices (including plain paper!).

Of course, the intriguing question is whether our abstract schemes for programs and/or documents are true pointers to some powerful universal truth, or whether the sense of elegance and economy in a good approach is just a delusion. One suspects that most computer scientists would like to believe that elegant data structures for organising all forms of information, together with powerful formalisms for manipulating and reasoning about it, have always pre-existed in some sense, and were just waiting to be discovered. (In the same way, most mathematicians reject the purely formalist, symbolic, view of mathematics and opt instead for a Platonic picture that ascribes an 'existence' to such abstract notions as integers, irrational numbers and groups.) Whatever the final verdict may be, EP has shown itself, yet again, to be more than just another area of applied computer science. It continues to offer tantalising glimpses of a future where programs and documents, taken together, are part of some bigger story.

<div align="right">DAVID BRAILSFORD AND RICHARD FURUTA</div>