HTML & TEX: Making them sweat[*]

Peter Flynn

Computer center, University College,
Cork, England
cbts8001@iruccvax.ucc.ie

Abstract

HTML is often criticised for its presentation-oriented conception. But it does contain sufficient structural information for many everyday purposes and this has led to its development into a more stable form. Future platforms for the World Wide Web may support other applications of SGML, and the present climate of popularity of the Web is a suitable opportunity for consolidation of the more stable features. TEX is pre-eminently stable and provides an ideal companion for the process of translating HTML into print.

1 Markup

HTML, a HyperText Markup Language[1], is the language used to structure text files for use in the World Wide Web, an Internet-based hypertext and multimedia distributed information system. HTML is an application of SGML, the Standard Generalized Markup Language, ISO 8879[3]. Contrary to popular belief, neither SGML nor HTML is new: SGML gained International Standard status in 1986 and HTML has been in use since 1989.

SGML is a specification for writing descriptions of text structure. In itself SGML does not *do* anything, any more than, say, Kernighan and Ritchie's specification of the C language[4] *does* anything: users and implementors have to do something *with* it. It has been slow to achieve popularity, partly because writing effective Document Type Descriptions (DTDs) is a non-trivial task, and partly because software to make full use of its facilities has traditionally been expensive. It was therefore seen as a 'big business only' solution to text-handling problems until the popularisation of HTML owing to increased use of the World Wide Web. Since 1992 the software position has also improved considerably — an extensive list of tools is maintained by Steve Pfeffer at UIO[6].

2 The World Wide Web

WWW (W3 or just 'the Web') is a client-server application on the Internet. Users' clients ('browsers') request files from servers run by information providers and display them, using the HTML markup embedded in the text to render the formatting. Some of the markup can provide filenames for the retrieval of graphics as illustrations, or act as anchor-points for links to other documents, which can be further text, or graphics, sound or motion video. This latter capability gives the Web a hypertext and multimedia dimension, and allows crosslinking of files almost anywhere on the Internet.

Because the HTML files are plain text with embedded plain text markup, in traditional SGML manner, they are immediately portable between arbitrary makes and models of computer or operating system, making the Web one of the first genuinely portable, multiplatform applications of its kind.

2.1 HTML Markup

An example of simple markup and an appropriate rendering is illustrated in Figure 2. The conventions of SGML's Reference Concrete Syntax[3] are used, so markup 'tags' are enclosed in angle brackets (less-than and greater-than signs), in pairs surrounding the text to which they refer, with the end-tag being preceded by a slash or solidus immediately after its opening angle bracket.

The rendering is left almost entirely to the user's client program, as there are almost no facilities within HTML for the expression of appearance apart from a minimal indication of font change (italics, boldface and typewriter-type). Indeed, most recent browsers allow the *user* arbitrary control over which fonts, sizes and colours should be used to instantiate the tagged elements of text.

2.2 Implementation

HTML was devised for the Web by non-SGML-experts who saw it as an ideal mechanism for implementing plain-text portability while preserving sufficient structural information for online rendering: one of the classical reasons for adopting SGML. It is now becoming standardised by an IETF working group who have produced a draft specification in the form of a formal DTD[1]. Because of the need to allow this specification to model existing 'legacy' documents (most of which would be regarded as fragments

[*] Reprint from the Annals of the UK TEX Users Group **Baskerville**, Volume 5.2, March 1995. Published with permission of both Baskerville editor and author. Presented at the UK TEX Users Group conference 'Portable Documents: Acrobat, SGML, and TEX', on 19 January 1995, London, England.

```
#! /bin/sh

echo Content-type: text/html
echo

cat <<EOH
<html><head><title>Date and time</title></head><body><p>It is now
EOH
date
cat <<EOT
</p></body></html>
EOT
```

Figure 1: *Example of a Unix shell script to return the date and time as a HTML file*

```
<html>
  <head>
    <title>Fleet Street Eats</title>
  </head>
  <body>
  <h1>Where to eat in Fleet Street</h1>
  <p>There are many restaurants in the City, from
     fast-food joints to <i>haute cuisine</i>.</p>
  ...
```

Document title: `Fleet Street Eats`

Where to eat in Fleet Street

There are many restaurants in the City, from fast-food joints to *haute cuisine*.

...

Figure 2: *Example of HTML markup and possible rendering*

rather than document instances), as well as provide for more robust usage, the current DTD has two modes: a non-rigorous ‘deprecated’ mode for describing the legacy and a ‘recommended’ mode for creating and maintaining files in conventional form.

HTML is sufficient for minimal documents, providing the structural and visual features shown in Figure 3. A future version (3.0) is being developed by the IETF Working Group, which will allow the description of mathematics, tables and some additional visual- and content-oriented features.

Despite the coming improvements, HTML is likely to be joined in the Web by other DTDs in future. One well-known SGML software house already has a prototype browser which can handle instances of arbitrary DTDs, given sufficient formatting information. This would make it possible to use the Web for transmission and display of documents using other SGML applications such as CALS (US Military), DocBook (O’Reilly/Davenport), the TEI (Text Encoding Initiative) and corporation-specific DTDs (such as those of Elsevier).

The next version of the DTD, HTML3, contains specifications for mathematics, tables and some additional elements for content-descriptive material, as well as a few extra visual keys such as an `ALIGN` attribute for positional specification. Most of this work is being implemented on a test basis in the Arena browser (Unix/X only at the moment) at CERN.

Although Web browsers can reference files by any of several methods (HTTP, the Web’s ‘native’ protocol; FTP; Telnet; Gopher; WAIS; and others) by using the URL (Universal Resource Locator: a form of file address on the Internet), the most powerful tool lies at the server end: the ability of servers to execute scripts, provided their output is HTML. A trivial example is shown in Figure 1, which returns the date and time.

Such a script can contain arbitrary processing, including the invocation of command-line programs and the passing of arguments. Data can be gathered from the user either with the `<isindex>` tag in the header, which causes a single-line data-entry field to appear, or with the more complex `<form>` element with scrollable text boxes, checkboxes, radio buttons and menus. In this manner, complete front-ends can be manufactured to drive data-retrieval engines of any kind, provided that they operate from the command line, and that the script returns their output in HTML. The user (and the browser) remain unaware that the result has been generated dynamically.

Structural		**Descriptive**		**Visual**	
`html`	document type	`a`	hypertext link anchor-point	`b`	bold type
`head`	document header	`cite`	citations	`br`	forced line-break
`title`	document title	`code`	computer code	`hr`	horizontal rule
`base`	root address for incomplete hypertext references	`em`	emphasis	`i`	italics
`meta`	specification of mapped headers	`kbd`	keyboard input	`tt`	typewriter type
`link`	relationship of document to outside world	`samp`	sample of input	`img`	illustrations
`isindex`	specifies a processable document which can take an argument	`strong`	strong emphasis		
		`var`	program variable		
`body`	contains all the text				
`h1`...`h6`	six levels of section heading				
`p`	paragraph				
`pre`	preformatted text				
`blockquote`	block quotations	**Form-fill**		**Obsolete:**	
`address`	addresses	`form`	contains a form	`listing`	use `pre`
`ol`	ordered lists	`textarea`	free-text entry	`xmp`	use `pre`
`ul`	unordered lists	`input`	input field (text, checkbox, radio button, *etc*	`plaintext`	use `pre`
`menu`	menu lists	`select`	drop-down menu	`nextid`	editing control
`dir`	directory lists	`option`	menu item	`dfn`	definition of term
`li`	list item				
`dl`	definition lists				
`dt`	definition list term				
`dd`	definition list description				

Figure 3: *Markup available in HTML 2.0 (indentation implies the item must occur within the domain of its [non-indented] parent)*

2.3 Presentation

HTML is criticised for being 'presentation-oriented', but as can be seen from Figure 3, the overwhelming majority of the markup is structural or content-descriptive. However, this does not prevent the naïve or sophisticated author from using or abusing the markup in attempts to coerce browsers into displaying a specific visual instantiation, primarily because none of the browsers (with the partial exceptions of Arena and `w3-mode` for GNU Emacs) performs any form of validation parsing, and will thus display any random assemblage of tags masquerading as HTML. This behaviour has misled even some eminent authorities to dismiss HTML as 'not being SGML'.

There is thus a conflict between the SGML purist on the one side, who decries any attempt at encoding visual appearance; and the uninformed author on the other, who has been unintentionally misled into thinking that HTML and the Web constitute some kind of glorified networked DTP system.

The purists are few in number but eloquently vocal: however, in general, they acknowledge that visual keys can be included if they are carefully coded. A perceived requirement to allow an author to recommend the centering of an element is thus achieved in HTML3 by the `align="center"` attribute, rather than the unnecessary `<center>` element proposed by the authors of Netscape.

The demands of the author are at their most marked in the approach of publishers and marketing users, who have been accustomed for the last 550 years to exert absolute control over the final appearance of their text. But the Web is not paper, and the freedoms and constraints of the Press do not apply: it is as much a new medium as radio or television. For such an author to insist that she must be able to control the final display to the same extent as on paper is as pointless as insisting that a viewer with a black-and-white television must be able to see the colours in a commercial.

The paradigm has been established that the browser controls the appearance, using the markup as guidelines. There is indeed no reason at all why attributes could not be added so that an author could write

```
<h1 color=green font=LucidaBrightBoldItalic
                        size=24 shading=50>
```

but the user of Lynx or WWW (two popular text-only browsers for terminal screens) would still only see the heading in fixed-width typewriter characters. The habit of insisting that everyone 'must' see a particular typographic instantiation is an unfortunate result of a misinterpretation of the objective of the Web: to deliver information in a compact, portable and arbitrarily reprocessable form.

But publishers accustomed to paper, insistent on 'keeping control', have of course an entirely valid point, one with which the present author has great sympathy. Why should a carefully-prepared document be made a hames of by a typographically illiterate user who has set `<h2>` to display as 44pt Punk Bold in diagonal purple and green stripes?

The solution probably lies in the implementation of style sheets, perhaps along the lines of those discussed by the authors of Arena[5]. They would in any case only be recommendations: not every user has a CD-ROM of Adobe or Monotype fonts. In any event, if 100% control is essential, as in the display of typographic examples, all graphical browsers can be configured to spawn a window to display PostScript file, although the download time may be a strong disincentive.

It is entirely possible that the control of content will ultimately prove a more attractive option than the control of appearance.

3 Publishing with HTML

Setting aside the unresolved questions of display, there are more pressing business problems about publishing on the Web.

The authentication of users is being addressed at several levels, from simple, non-authoritative checks using `identd` to the more complex username-and-password systems employed on some Web pages. From the user's end, the authentication of the data being accessed is equally important. The openness of the Internet in its raw form allows 'spoofing' in both directions, so the emergence of protocols to provide checks is to be welcomed.

The security of network-accessible texts from break-ins remains a concern to anyone providing high-value merchandise, and Web text is in this sense no different from any other computer data. Normal precautions must therefore be taken to prevent theft through other channels (such as remote login), as distinct from theft perpetrated by falsification of Web access.

There is a need for robust solutions to charging and billing for usage, and the secure transmission of financial data, including credit card numbers, digital signatures, and perhaps even EFT transactions. The Secure HTTP (SHTTP) mechanism being marketed by MCom and others is becoming popular as a way of achieving some of this, but the Internet must shed some of its image of lax controls and sloppy housekeeping if it is to achieve sufficient 'respectability' to attract the business of those who are not networking specialists.

The handling of copyright and the intellectual property of electronic texts remains, as ever, an unsolved problem. While copyright law can be used to provide a remedy for breach, the difficulty lies in preventing the breach occurring in the first place. The reason is that (as with other electronic material), copying and reproduction is fast, cheap and easy, once the material is in the hands of the customer. While a supplier may use SHTTP to protect the details of the transaction, once a print file has been sent to someone, the supplier retains no control whatsoever over its use, reuse and abuse. Copies could be sent to dozens others, or printed many times, in the space of minutes.

3.1 Printing from HTML

The demand for printed copies of Web material is surprisingly high. Although in some cases it is reminiscent of those people who insist on printing their email, it is undeniable that there is a serious requirement for good quality print from Web documents.

Existing solutions to printing SGML text are usually application-specific, being embedded in SGML editors or DTP systems, but there are also some more generic packages:

- `Format` by Thomas Gordon (LATEX)
- `HTMLtoPS` by Jan Kårrman (PostScript)
- `SGML2TeX` and `WebSet` by Peter Flynn (TEX/LATEX)
- `SimSim` by Jonathan Fine (TEX)

The use of TEX systems for most of these seems to indicate that the similarity of markup concepts has not gone unnoticed by practitioners. The author's own contributions are experimental, but the second of them is planned as an interactive Web service, to be introduced in the summer of 1995. Emailing a URL to the point of service will cause it to be retrieved, typeset, and the output returned to the user by email in PostScript form. As a form of email browser, the control of appearance may lie in the hands of the user, but suggestions for how implement this are currently being sought[2].

3.2 Problems

Implementing a professional level of typesetting from HTML raises some interesting questions:

- most HTML files are invalid
- most HTML authors don't understand SGML
- most HTML authors couldn't care less
- most World Wide Web users couldn't care less

The handling of missing, damaged or abused tags in a gracious manner is not a feature of most SGML parsers. At the best, a typesetter-browser can only be expected to report to the user that a file is invalid, and while it may be displayed by browsers which do not make any claim to typographic quality, an attempt to make a respectable print job of an invalid file is unlikely to succeed.

4 Development

The future of the World Wide Web and HTML is uncertain. While development continues, and while new users are anxious to start surfing the net, the existing designs and implementations will suffice. In the longer term, a coalescing of services is likely to occur, but for this to happen, a number of changes need to take place:

- The Web will start to make use of other DTDs, as outlined above. Any file containing a `<!doctype...>` at the beginning could cause a browser to retrieve the DTD specified, along with a style sheet, and work much as any SGML-conformant DTP system would.
- Browsers will become pickier, able to offer better services at the expense of rejecting invalid or badly broken files. Arena already perfoms a form of consistency check on the HTML code of files, and displays 'Bad HTML' in the top corner when an offender is spotted.
- Users will become pickier, demanding better response from the browser, better response from the server, and better facilities from both. As users become more educated about the use of SGML, developers will no longer be able to hide the deficiencies of products under the cover of technical detail.
- This presupposes more user education, which is inevitable in a developing technology. 100 years ago, motor cars appeared on the roads, but few passengers in them understood the use of the levers and rods which controlled them. With some minor exceptions, it is now expected that a driver knows that turning the wheel clockwise turns the car to the right, and *vice versa*. It will not take us that long to perceive the innards of HTML, but it can only be done by training and education.
- At some stage, investment is always needed. Many companies have put substantial sums into the development of Internet resources, and those that have done so with forethought and planning deserve to reap a rich reward. It is a long-term investment, more akin to a partnership, but support is always needed by those who undertake the developments, especially as much of it is done in personal time and at personal expense.

There is still some way to go before we achieve the ease of use of the telephone or the radio, but the path is becoming easier with each new development.

References

[1] Berners-Lee T & Connolly D, *HyperText Markup Language Specification — 2.0*, Internet Draft, IETF Working Group on HTML, December 1994.

[2] Flynn P, *Typographers' Inn*, TEX and TUG NEWS, **4**, 1, March 1995.

[3] Goldfarb C, *The SGML Handbook*, OUP, 1990, ISBN 0–19–853737–9.

[4] Kernighan BW & Ritchie DM, *The C Programming Language*, Prentice-Hall, 1978.

[5] Lie H *et al*, *HTML Style sheets*, `http://www.w3.org/hypertext/WWW/Style/`

[6] Pepper S, *The Whirlwind Guide: SGML tools and vendors*, `ftp://ftp.ifi.uio.no/pub/SGML/SGML-Tools/SGML-Tools.txt`