DOCUMENT RESUME

ED 411 868 IR 056 695

AUTHOR	Dixon, Ross
TITLE	How Can Documents Be Put onto the Web?
PUB DATE	1996-00-00
NOTE	6p.; In: Online Information 96. Proceedings of the International Online Information Meeting (20th, Olympia 2, London, England, United Kingdom, December 3-5, 1996); see IR 056 631.
PUB TYPE	Guides - Non-Classroom (055) -- Reports - Evaluative (142) -- Speeches/Meeting Papers (150)
EDRS PRICE	MF01/PC01 Plus Postage.
DESCRIPTORS	Access to Information; *Authoring Aids (Programming); Case Studies; Computer Mediated Communication; Computer Software Evaluation; Computers; *Electronic Publishing; *Electronic Text; Foreign Countries; Hypermedia; *Information Dissemination; Online Searching; Online Systems; *World Wide Web
IDENTIFIERS	*Adobe Acrobat; *HTML; SGML; United Kingdom

ABSTRACT

The specific characteristics of the World Wide Web mean that publishing documents using this medium has a number of unique requirements and presents a different challenge to anyone wishing to put documents on the Web. Key issues which have to be resolved include determining the nature of the documents to be published, identifying whether documents should be published directly or from an intermediate form, and deciding what sort of document creation tool should be used. Three types of documents can be published on the Web: formatted documents--documents which cannot be edited or have their contents searched, such as images; processable documents--the contents of which can be searched and/or edited, such as text documents; and formatted processable documents--documents that are content searchable and allow hypertext linking, but do not allow editing and require that documents exist as a collection of pages, such as documents produced in Adobe Acrobat. The paper defines HTML (hypertext mark-up language) and Adobe Acrobat, explains how HTML and Acrobat documents are created, and describes key characteristics of the two and to what Web applications each are suited. It also addresses converting documents that exist only in paper form to Web readable pages. Two case studies illustrate the conversion of paper based documents directly to Acrobat, and the use of SGML (standard generalized mark-up language) as a basis for output. (Author/SWC)

How can documents be put onto the Web?

Ross Dixon
Pindar PLC, UK

Abstract: *The World Wide Web is rapidly maturing as a medium for publishing documents online. This makes it important to both publishers and other organisations with an online publishing requirement. The specific characteristics of the Web mean that publishing using this medium has a number of unique requirements and presents a different challenge to anyone wishing to put documents on the Web. Key issues which have to be resolved include determining the nature of the documents to be published, identifying whether documents should be published directly or from an intermediate form, and deciding what sort of document creation tool should be used. All of these issues are addressed within this paper explicitly and with the use of case studies.*

Keywords: Internet, Web, documents, Acrobat, HTML

1. Introduction

The World Wide Web, or simply the Web, has grown out of the Internet and become a significant publishing medium. Estimates of beyond 50 million users have been made and many organisations (publishers and others) have identified the Web as being integral to their approach to electronic publishing. Clearly the Web has many applications and most Web sites are aimed at marketing rather than serious publishing. However as the understanding, size and infrastructure of the Web continues to increase then scope for its use as an online publishing medium will continue to grow. The more recent intranet phenomenon aimed at internal publishing within organisations further moves the emphasis from marketing to serious publishing use of the Web.

Already the Web has a number of significant publishing sites (many of them in their pilot stages). The experience gained by Web publishers shows that various pitfalls exist for the potential Web publisher. This paper aims to define a brief route map for anyone wanting to get their documents onto the Web and is based on experience from a variety of projects.

2. What types of documents can be published on the Web?

A key decision for any Web publisher is the form of the documents to be published. While many forms of document can be published these can be categorised in three ways:

2.1. Formatted documents

These cannot be edited or have their contents searched. A typical example of such documents are images, i.e. TIFF images. The rather unintelligent nature of such documents is well known and not well suited to the Web. In particular their requirement for indexing (no content searching), no scope for hypertext linking and their data intensive nature (i.e. slow to download) are particularly problematic. For these reasons, this type of document is less common on the Web and is not considered further in this paper.

2.2. Processable documents

The content of these documents can be processed, i.e. searched and/or edited. Processable documents are what we typically think of when we talk of electronic documents — text (i.e. from a word processor) is the obvious example. In comparison with formatted documents, processable documents appear to be much more suited to the Web, i.e. they are content searchable, allowing hypertext linking and less data intensive. An often overlooked characteristic of processable documents is that they do not have to exist as a collection of pages and this non-page approach is supported by the Web. In fact these types of documents are by far the most common on the Web and a specific approach to 'marking-up' the text for the Web known as HTML (Hyper Text Mark-up Language) is the commonest approach to text publishing on the Web.

2.3. Formatted processable documents

This is a somewhat more unusual concept and can be thought of as freezing a processable document such that it is still content searchable and allows hypertext linking but usually does not allow editing and requires that documents do exist as a collection of pages. As expected, this hybrid approach has characteristics of the other two document types but most interestingly maintains most of the benefits of processable documents while forcing the familiar page approach to be retained. The most popular implementation of the formatted processable approach is Adobe Acrobat.

3. What is HTML (hypertext mark-up language)?

HTML is the native language of the Web. The HTML approach requires that document contents are tagged (marked-up) with codes to denote the document presentation and structure. This allows the marked-up contents to be presented and displayed by a Web browser according to the intended style of the Web publisher. In addition to arranging text as required, HTML codes can also be used to make hyperlinks (connecting together non-contemporaneous document sections or different documents), creating forms and referencing graphic data. The benefit of this approach is that the publisher has great flexibility regarding what can be published and has absolute control over the published material. Most important of all, since HTML is the native language of the Web, all Web browsers understand (some versions of) HTML and display HTML pages. Largely because of these characteristics, the vast majority of Web pages are HTML pages.

4. How are HTML documents created?

Three methods of generating HTML documents are relevant and are described below:

4.1. HTML documents

These can be created by any word processor because the HTML codes (i.e. the tags) used are defined using the standard ASCII character set which can be input directly from the keyboard into any word processor document. While this may appear to be the simplest approach (making use of existing applications), it is not commonly used because it requires a high degree of HTML expertise from authors and in comparison with the other approaches is cumbersome and error prone.

4.2. HTML editing packages

Specific HTML editing packages exist which eliminate much of the tedium associated with creating HTML pages directly from a word processor. HTML editors may be associated with word processors or other applications such that the editor is an add-on application which ‘HTML-enables’ the word processor. This approach has the benefit of familiarity for the word processor user and is therefore suitable for casual and low volume HTML document production. However, this approach is limited by the characteristics of the word processor which was almost certainly not designed for HTML output. The alternative approach for higher volume HTML document creation is to use a dedicated HTML editor which is not associated with a word processor.

4.3. Generate HTML files automatically

The final approach is to generate HTML automatically from other data files. Once again two methods are possible here: the first is to use a document creation package such as a word processor or desktop publisher which can export documents to HTML form. At present many such packages do not have these facilities and those that do suffer from the same problems as the add-on editors describe above. The second method is to export HTML data from SGML (standard generalised mark-up language) data. SGML is the prevalent data format used in electronic publishing and is closely related to HTML. In fact SGML inspired the creation of HTML and HTML is often described as a sub-set of SGML. The benefit of this second method is that data in SGML form is relatively easy to export to HTML, can be used in non-Web publishing applications (i.e. to CD-ROM or paper) and benefits from a relatively mature infrastructure which has developed in the 10 years since SGML was formally standardised.

5. What is Adobe Acrobat?

Adobe Acrobat is aimed at allowing users to distribute, interchange, view, annotate and print most forms of electronic document, i.e. a text file from a word processor, a spreadsheet file, DTP graphics and document images. This approach involves converting documents to a proprietary form which can be viewed by a viewer running on most desktop computing platforms. While Acrobat is the most popular approach in this class it is worth noting that comparable products exist, notably Envoy (from Corel) and World View (from Interleaf). For the sake of brevity only Acrobat is discussed in detail in this paper.

Acrobat (and other document viewers) aims to overcome the traditional problems of electronic document interchange such as Word Perfect users attempting to read documents from various versions of MS Word. Historically the only approaches available here were to standardise on application packages or to use an interchange format. The problem here is that enforcing standardisation is difficult even within an organisation and across different organisations it is impossible. Also the main interchange formats are restricted to specific data types, i.e. RTF for text, IGES for CAD drawings, TIFF for document images and so on. The Acrobat approach is a development of the interchange format which allows practically all electronic documents to be held in a single format which can be viewed using a single viewer.

So why is this important? Well, the Acrobat approach (remembering that other products offer similar functionality) allows distribution and interchange of electronic documents across most computing platforms without getting bogged down in compatibility issues.

6. How are Acrobat documents created?

A cornerstone of Acrobat is that any document which can be printed to a PostScript printer (most printers fall into this category) can be input into the Acrobat format known as PDF (Portable Data File). Two methods of creating the PDF from the electronic document are possible:

- PDF output can be generated directly from the document creation application (such as a word processor). This will usually involve the use of an additional Acrobat package known as PDF Writer;
- PostScript data (generated by the document creation package) can be converted to PDF by the use of Acrobat Distiller.

Both the above approaches exclude paper documents. A further package (known as Acrobat Capture) caters for this situation and is described later. One of the key features of the Acrobat approach is that specific software (from Adobe) is required and in this sense it is not considered as 'open' as the HTML approach.

7. Key characteristics of Acrobat/HTML and to what Web applications are each suited?

Acrobat PDF maintains the concept of pages and does not allow editing of contained data — it does allow simple re-use of this data (cut & paste) and annotation of pages. Graphics are included as part of the PDF file. This makes Acrobat a useful form for page publishing on the Web.

HTML does not require the page metaphor to be maintained and allows contained data to be edited for reuse and repurposing. HTML documents can refer to other data such as graphics which are separate from the HTML document. This makes HTML suited to most forms of Web publishing and in particular where complicated documents with various document components are involved.

8. What if your documents only exist in paper form?

It is possible to convert paper documents directly to either PDF or HTML forms, or to convert to SGML and then output to PDF or HTML. For all of these approaches three general stages are required (unless documents are to be manually re-keyed):

8.1. Conversion from paper to image form

This activity simply involves scanning paper documents to produce (TIFF) images.

8.2. Conversion from image to text

This is a character recognition process which usually involves quality assurance and correction/editing processes. If conversion direct to PDF form is required then the recognition process can be undertaken within an Acrobat application known as Capture.

A key point about this process is that many recognition systems do not have a good concept of page layout — they can recognise collections of characters but for example are not good at recognising columns. While this is catered for within some packages such as Capture (although these packages tend to take a considerable time to recognise layout features), other packages may require some manual intervention here if the layout information of the document is to be maintained.

8.3. Mark up/output

From the recognition output the HTML or SGML tags can be inserted (semi-automatically or manually). In the case of Acrobat, multiple PDF files can be collected together, hyperlinks and index data added.

Experience shows that conversion from paper adds a significant cost and time overhead to the process and wherever possible use of already existing electronic files should be made.

9. Case studies

Case studies of the production of HTML documents or Acrobat documents are very common given the large uptake of these techniques: however conversion direct to Acrobat from Paper or the use of SGML as a basis for output are more recent developments and case studies are less common. Two such projects with which Pindar has been involved are an Acrobat Capture project for Oxford University Press and an online and paper publishing from SGML project for the Institution of Electrical Engineers. These are described briefly below:

9.1. Acrobat Capture case study

Earlier this year Oxford University Press (OUP) identified two separate pilot projects in order to assess the viability of Acrobat as a publishing format and Capture as a tool to make documents available in this form. The two projects related to separate publications. The first is a collection of six discontinued books known as the Physics Monographs. The aim of this project is to allow Web access to a subset of each book such that potential users can identify whether the book is of interest and which sections should be retrieved. Access to these sections will require a subscription which will give the user the key to the encryption allowing access. Many aspects of online electronic publishing are covered in this project but the key issue for Capture is that a discontinued book was brought back to life without involving new print runs and so on.

The second project related to a single book, *Leukocytes*. This is not to be made available over the Web (CD-ROM only) and is therefore not considered further here.

The capture from paper to image form was a fairly simple task because of the relatively small volumes involved, i.e. 3000 pages for the Monographs.

The recognition of the characters from the images was more problematic. This was because of the very processor intensive nature of this task and that at the time of the project the Capture product was still at version 1 and had a number of bugs. Most notable was a practical limit on the number of images which could be batched for recognition. This recognition task is ideally left as a batch process to run over night. However the version in use at the time would not allow more than 140 images to be batched for recognition. This severely limited the scope for batch processing and made the recognition process very slow and expensive.

After the recognition process had been completed the correction stage was undertaken and this took considerable resources for various reasons. Most important were the high standard required by OUP, the relative complexity of the books and a limitation within Capture where much of the text formatting data (bold, italics and so on) is not picked up by the recognition engine or the automated error detection, so this has to be identified and corrected manually. Finally the equivalent of a Contents Page is created within the PDF document by the addition of bookmarks which denote sections within books.

Access to the bookmarks of the Monographs will be freely available on the Web and also to all index, reference, contents, bibliography and symbol pages of the books. The idea here is that users will be able to browse this information and then decide whether they want access to the body of the books for which they will have to take out a subscription.

9.2. SGML case study

The Institution of Electrical Engineers (IEE) publishes several learned journals. Pindar is responsible for the production of two of these, presenting them on paper and making them available online through the Online Computer Library Center (OCLC).

The two journals are *Electronics Letters*, published 25 times a year and *IEE Proceedings*, published six times a year under 11 titles, giving 66 publications annually.

IEE chose SGML as the natural format to provide the base structure of its documents. However it was confronted with the problem that SGML encoding of the text and producing the typeset pages had until then been viewed as two separate processes. If these processes were distinct it would mean two things — firstly that the data would have to be processed twice, at a great cost to IEE, and secondly that there would be a weak relationship between the SGML and the data on the typeset page. It is important to IEE that the data in the electronic version is exactly the same as that in the printed version to ensure a consistent publication.

The approach taken to address this requirement is that articles from IEE are received in paper form, and are scanned to image form and then converted to text using character recognition. The recognition output is manually cleaned and marked-up. Any equations are converted to TeX form and held as EPS (Encapsulated PostScript) files: figures are maintained as TIFF images. The EPS and TIFF files are embedded into the articles as images.

Once the SGML data has been produced the different outputs are generated. For hard copy, the SGML data is imported into a DTP application which formats the articles according to the IEE typesetting specification from which film and printing activities are undertaken. The OCLC online database is populated with the SGML data and finally PDFs are produced for OCLC to provide document output for subscribers.

10. Conclusion

Very wide scope exists for making use of the Web as a publishing medium. As the relevant infrastructure widens and improves it is anticipated that Web publishing will become increasingly important. A result of this is that the serious Web publisher will be confronted with the practical issue of how documents can be converted to a suitable form. In the short term this can be addressed by the use of existing packages for the production of HTML documents. Much more problematic is dealing with documents which only exist in paper form and ensuring that electronic documents are 'future-proofed' for unforeseen versions of HTML or non-Web media. Fortunately the electronic publishing world existed long before the Web and familiar techniques such as document conversion and mark-up provide a credible approach to addressing these issues despite their recent emergence.

Ross Dixon
Pindar PLC
Ryedale Building
60 Piccadilly
York
YO1 1NX
UK
Tel: +44(0)1904 613040
Fax: +44(0)1904 613110
E-mail: R.Dixon@pindar.co.uk